版本 16

JMP 文档库

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.1

All JMP® 16 books are included in this document.

The correct bibliographic citation for this document is as follows: SAS Institute Inc. 2020–2021.JMP® 16 文档库.Cary, NC: SAS Institute Inc.

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

March 2021

August 2021

版本 16

发现 JMP

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020–2021. *Discovering JMP® 16*. Cary, NC: SAS Institute Inc.

Discovering JMP® 16

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

March 2021

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 员工分享使用技巧和使用经历的博客
- 同其他用户讨论 JMP 的论坛

https://www.jmp.com/getstarted

目录

发现 JMP

关于本手册 11

JMP 图形库 13

1 了解 JMP 33

文档和其他资源

JMP 文档中的格式规范 35

JMP 帮助 36

JMP 文档库 36

学习 JMP 的其他资源 42

JMP 教程 42

样本数据表 42

学习统计和 JSL 术语 43

学习 JMP 的提示与技巧 43

JMP 工具提示 43

JMP 用户社区 43

免费在线统计思维课程 43

新 JMP 用户欢迎套件 44

统计学知识门户 44

JMP 培训 44

用户编写的 JMP 手册 44

“JMP 起始页”窗口 44

JMP 技术支持 44

2 JMP 简介 45

基本概念

您应该知道的 JMP 概念 47

JMP 入门帮助 47

启动 JMP 48

使用样本数据 49
了解数据表 51
了解 JMP 工作流程 52
第 1 步：启动 JMP 平台并查看结果 52
第 2 步：从 JMP 报表中删除箱线图 54
第 3 步：请求其他 JMP 输出 54
第 4 步：与 JMP 平台结果交互 55
JMP 与 Excel 有哪些区别？ 57
数据表的结构 57
JMP 中的公式 57
JMP 分析和绘图 58

3 使用数据 59
为绘图和分析准备数据
使数据出现在 JMP 中 61
将数据复制并粘贴到数据表 61
将数据导入数据表 61
在数据表中输入数据 63
将数据从 Excel 传输至 JMP 65
使用数据表 67
编辑数据表中的数据 67
选择、取消选择及查找数据表中的值 69
查看或更改数据表中的列信息 72
使用公式计算值的示例 74
在报表中过滤数据的示例 75
重整数据的示例 77
查看汇总统计量 77
创建子集的示例 81
连接数据表的示例 83
对数据排序 85

4 数据可视化 87
常见图形
在一元图形中分析单个变量 89
将直方图用于连续变量 89

将条形图用于分类变量 92
比较多个变量 94
使用散点图比较多个变量 95
使用散点图矩阵比较多个变量 98
使用并排箱线图比较多个变量 101
使用图形生成器比较多个变量 104
使用气泡图比较多个变量 110
使用叠加图比较多个变量 115
使用变异性图比较多个变量 119

5 分析数据 123
分布、关系和模型
关于本章 125
数据绘图的重要性 125
了解建模类型 128
查看建模类型结果的示例 129
更改建模类型 130
分析分布 131
连续变量的分布 132
分类变量的分布 134
分析关系 137
使用单预测变量回归 137
比较一个变量的平均值 142
比较比例 145
比较多个变量的平均值 147
使用多个预测变量的回归 151

6 概览 157
在多个平台中探索数据
有趣的事实：多个分析的相互链接 159
在多个平台中探索数据 159
在“分布”平台中分析分布 159
在“多元”平台中分析模式和关系 162
在“聚类”平台中分析相似值 166

7 保存和共享您的工作 171

保存和重新创建结果

使用项目 173
创建新项目 174
在项目中打开文件 174
在项目中重新排列文件 174
保存项目 175
项目工作区 176
项目书签 177
项目内容 177
项目中最近使用的文件 178
项目日志 178
将文件移动到项目中 179
编写项目 On Open 脚本 179
创建新项目的示例 180
将平台结果保存在记录中 182
示例：创建记录 182
将分析添加到记录中 183
保存和运行脚本 184
示例：保存和运行脚本 184
关于脚本和 JSL 185
将报表另存为交互式 HTML 185
交互式 HTML 包含数据 186
示例：创建交互式 HTML 186
将报表共享为交互式 HTML 187
将报表另存为 PowerPoint 演示文稿 191
创建仪表板 192
示例：组合窗口 192
示例：创建含两个报表的仪表板 194

8 特殊功能 195

自动分析更新和 SAS 集成

自动更新分析和图形 197
示例：使用自动重新计算 197
更改首选项 201

示例：更改首选项 201
集成 JMP 与 SAS 204
示例：创建 SAS 代码 204
示例：提交 SAS 代码 204
A 技术声明 207

关于本手册

《发现 JMP》提供了有关 JMP 软件概括性的介绍。本指南假设您完全不熟悉 JMP。无论您是分析人员、研究人员、学生、教授还是统计人员，本指南向您全面展示 JMP 的用户界面和功能。

本指南介绍下列信息：

- 启动 JMP
- JMP 窗口的结构
- 准备和操控数据
- 使用交互式图形挖掘数据
- 执行简单的分析以扩充图形
- 定制 JMP 和特殊功能
- 共享结果

本指南共六章。每章都包括一些示例，帮助您加深理解该章中出现的概念。所有统计概念都仅做初步介绍。《发现 JMP》中使用的样本数据附带在软件中。下面是各章的简介：

- 第 2 章，“JMP 简介”概述 JMP 应用程序。您将学习如何组织内容以及如何导航软件。
- 第 3 章，“使用数据”说明如何从各种来源导入数据并准备数据进行分析。此外，概述了数据操控工具。
- 第 4 章，“数据可视化”对您可以用于可视化以及理解数据的图形和图表进行了说明。示例既包括简单的单变量分析，也包括多变量图形，您可以通过这些图形查看多个变量之间的关系。
- 第 5 章，“分析数据”说明多种常用分析技巧。这些技巧既包括不需要使用统计方法的简单技巧，也包括需要使用统计知识的高级技巧。
- 第 6 章，“概览”说明如何在若干平台中分析分布、模式和相似值。
- 第 7 章，“保存和共享您的工作”说明如何与非 JMP 用户在 PowerPoint 演示文稿和交互式 HTML 中共享您的工作。同时还涵盖了如何为 JMP 用户将分析另存为脚本，如何在记录和项目中保存工作。
- 第 8 章，“特殊功能”说明如何在数据发生更改时自动更新图形和分析、如何使用首选项定制报表，以及 JMP 如何与 SAS 交互。

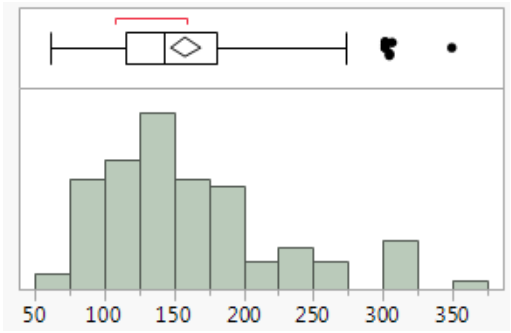
学习了本指南之后，您将可以在 JMP 中轻松地导航和使用数据。

JMP 同时适用于 Windows 和 macOS 操作系统， 但本指南中的材料基于 Windows 操作系统。

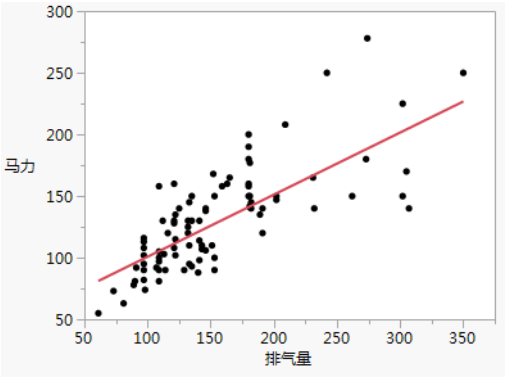
JMP 图形库

各种图形及其平台

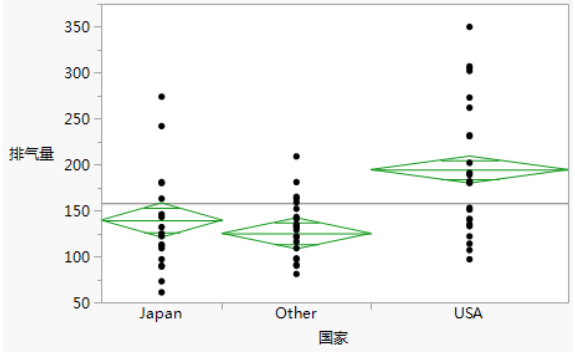
下面是您可以使用 JMP 创建的许多图形的一些图片。每张图片都标注了用于创建它的平台。有关平台及其图形的详细信息，请参见**帮助** > **手册**菜单上的文档。



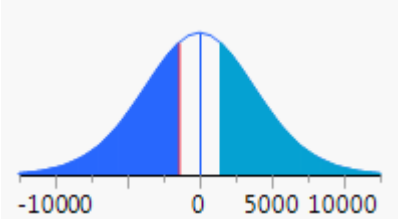
直方图
分析 > 分布



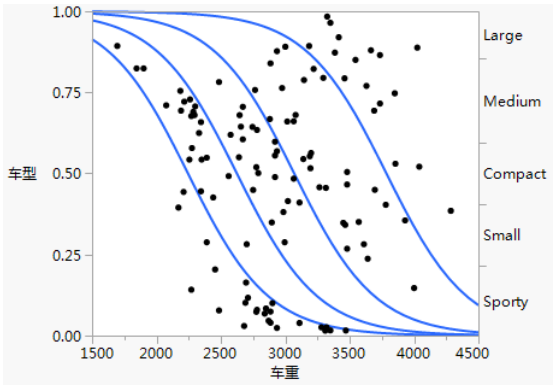
二元
分析 > 以 X 拟合 Y



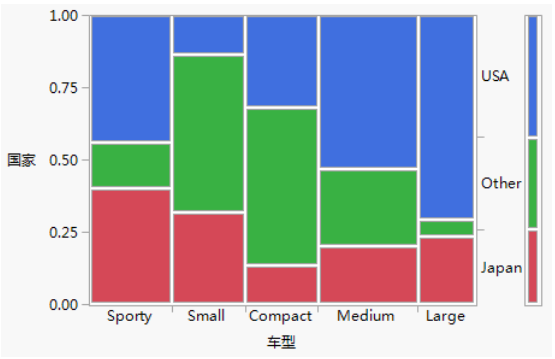
单因子
分析 > 以 X 拟合 Y



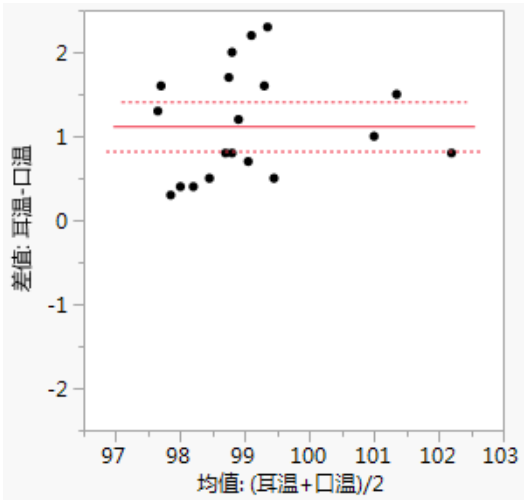
单因子 t 检验
分析 > 以 X 拟合 Y



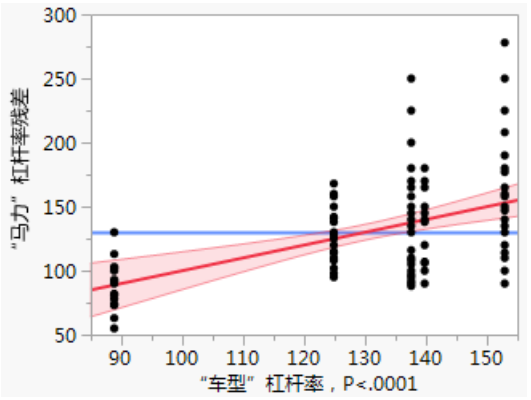
Logistic
分析 > 以 X 拟合 Y



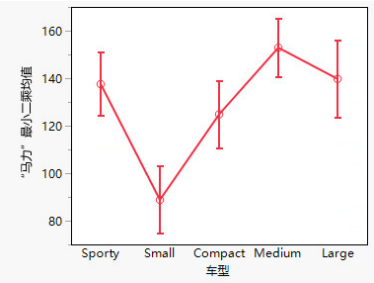
马赛克图
分析 > 以 X 拟合 Y



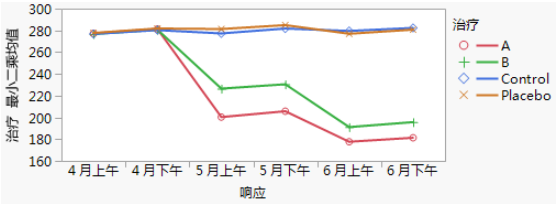
配对
分析 > 专业建模 > 配对



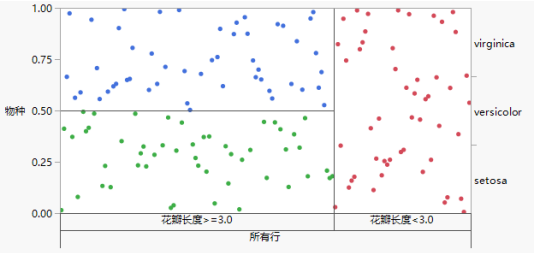
杠杆率图
分析 > 拟合模型



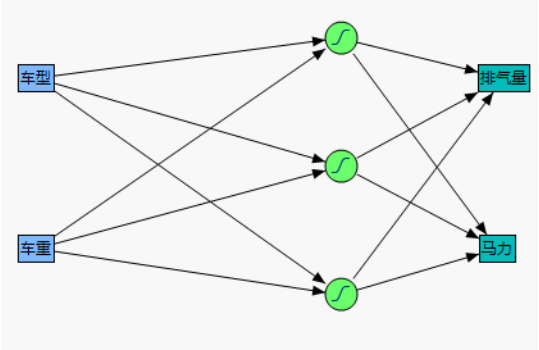
最小二乘均值图
分析 > 拟合模型



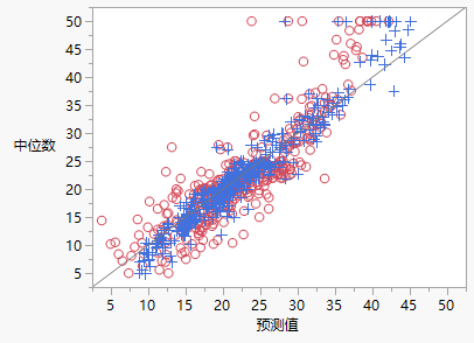
多元方差分析
分析 > 拟合模型



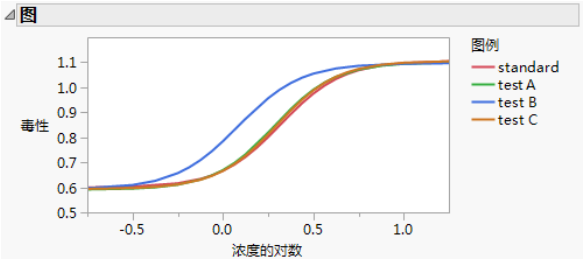
分割
分析 > 预测建模 > 分割



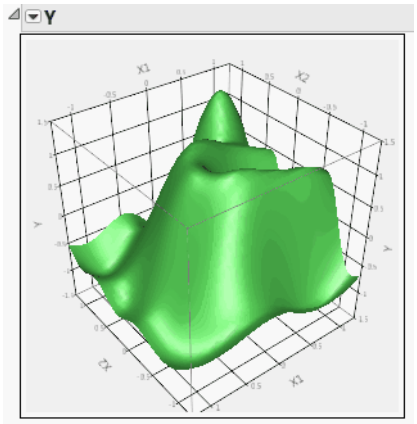
神经关系图
分析 > 预测建模 > 神经



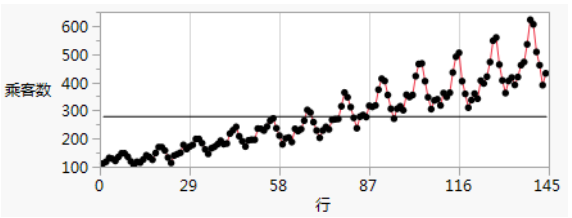
预测值-实际值
分析 > 预测建模 > 模型比较



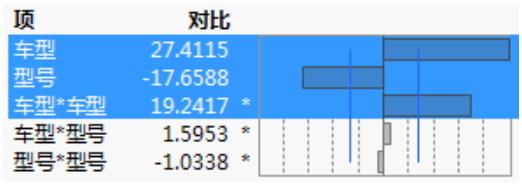
非线性拟合
分析 > 专业建模 > 非线性



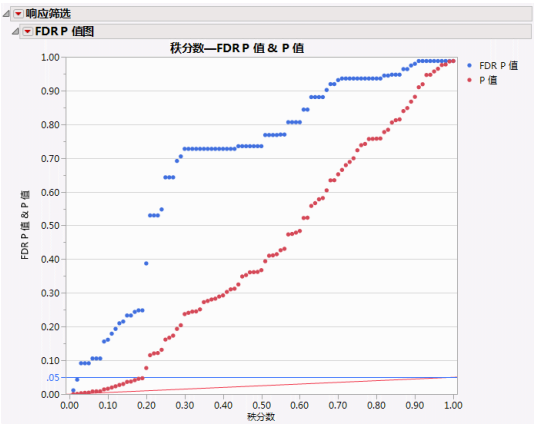
曲面刻画器
分析 > 专业建模 > 高斯过程



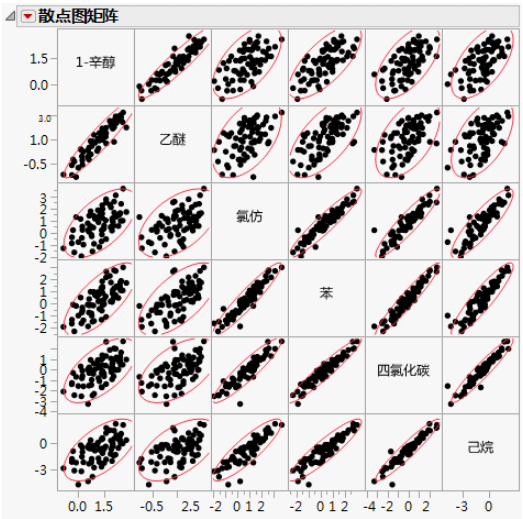
时间序列
分析 > 专业建模 > 时间序列



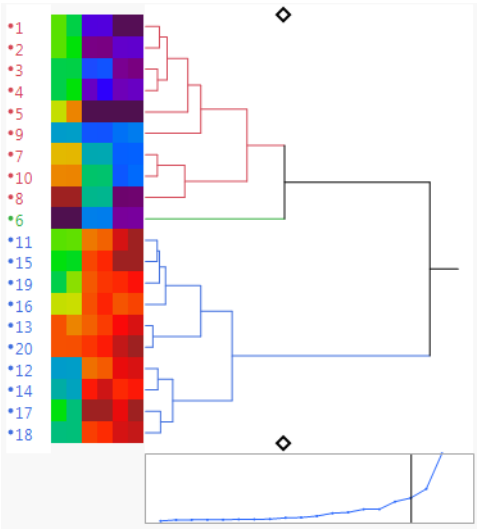
筛选
分析 > 专业建模 > 专业实验设计模型 > 拟合两水平筛选



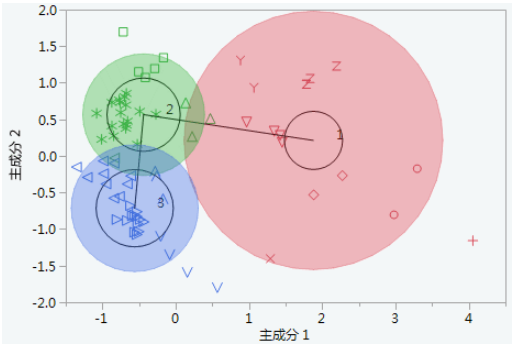
FDR p 值图
分析 > 筛选 > 响应筛选



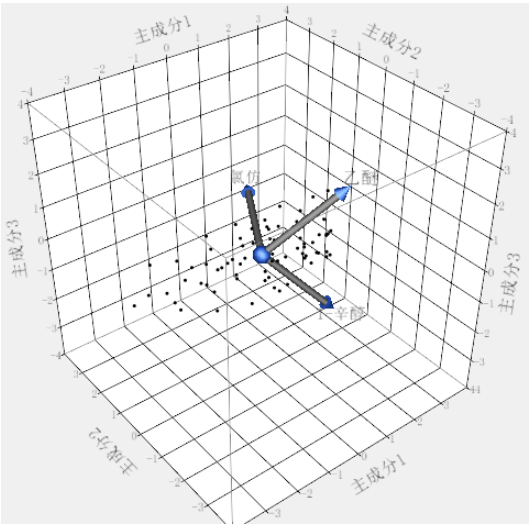
散点图矩阵
分析 > 多元方法 > 多元



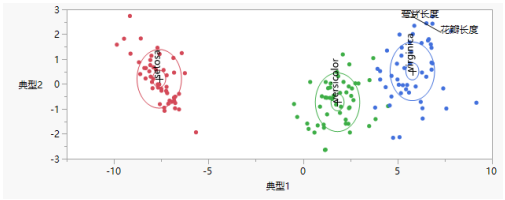
树状图
分析 > 聚类 > 层次聚类



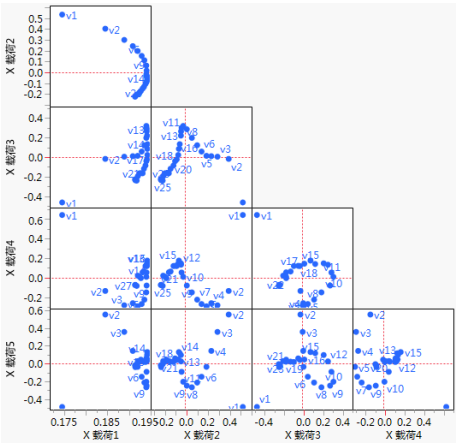
自组织图
分析 > 聚类 > K 均值聚类



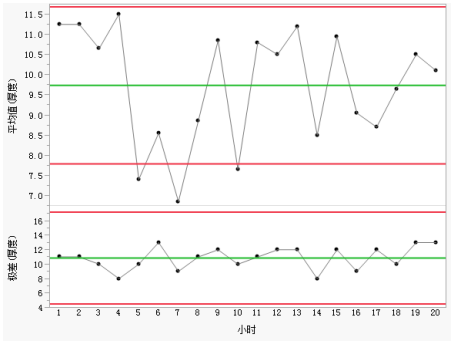
主成分
分析 > 多元方法 > 主成分



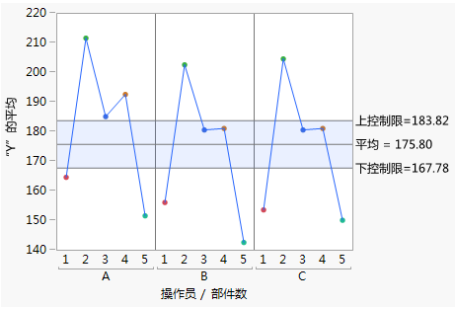
典型图
分析 > 多元方法 > 判别



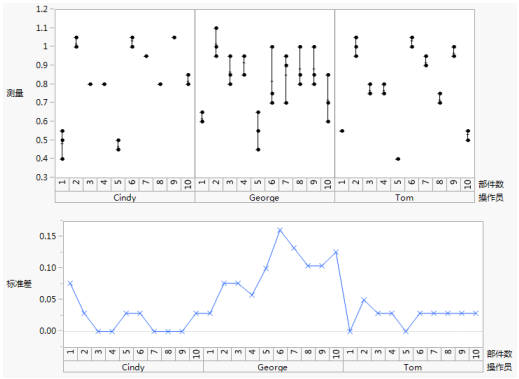
载荷图
分析 > 多元方法 > 偏最小二乘



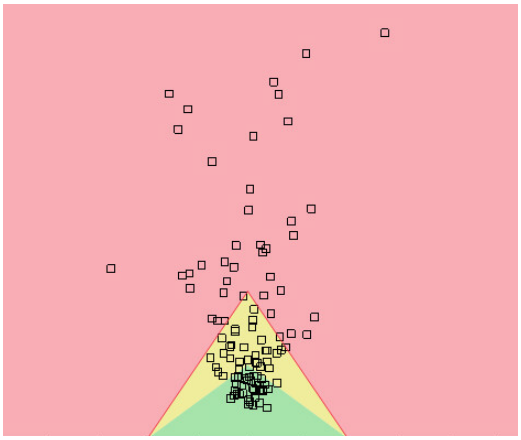
均值图和 R 图
分析 > 质量和过程 > 控制图生成器



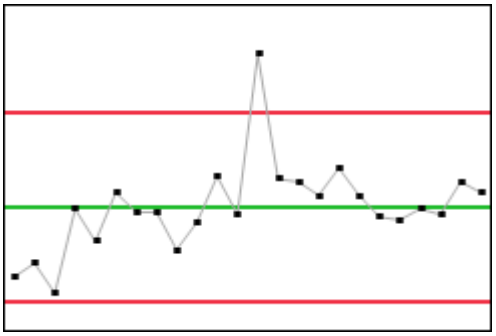
平均图
分析 > 质量和过程 > 测量系统分析



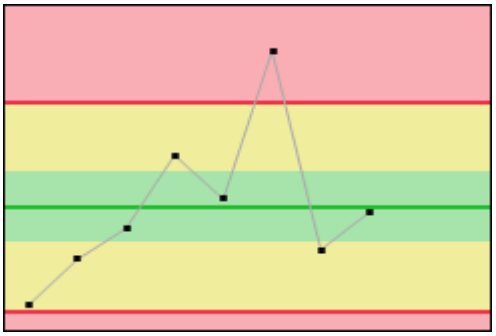
变异性图
分析 > 质量和过程 > 变异性/计数量具图



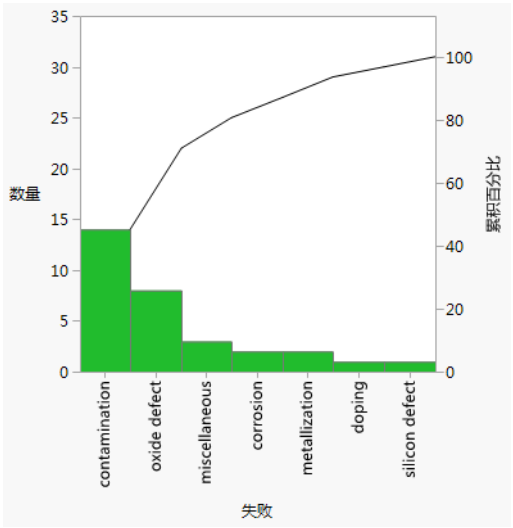
目标图
分析 > 质量和过程 > 过程能力



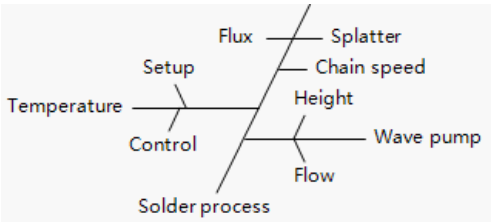
单个度量值图
移动极差图
分析 > 质量和过程 > 控制图 > 单值极差



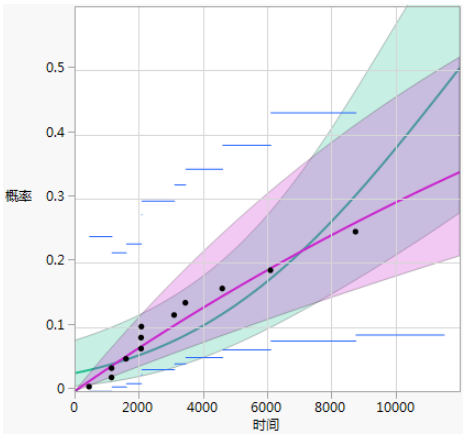
均值控制图
分析 > 质量和过程 > 控制图 > 均值



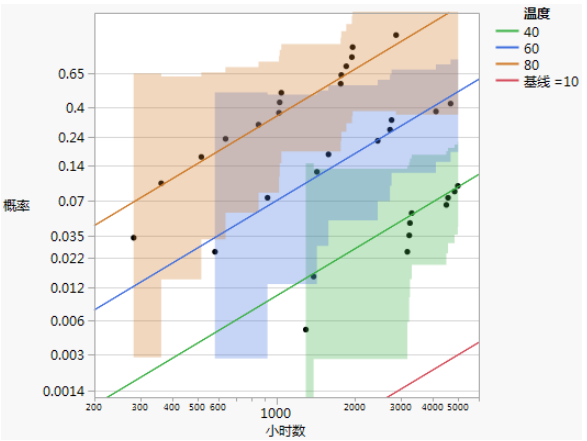
Pareto 图
分析 > 质量和过程 > Pareto 图



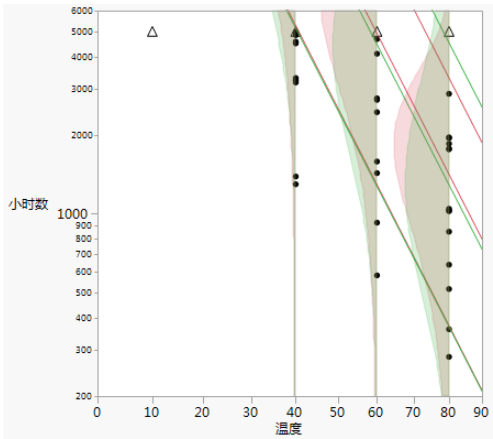
石川图
鱼骨图
分析 > 质量和过程 > 关系图



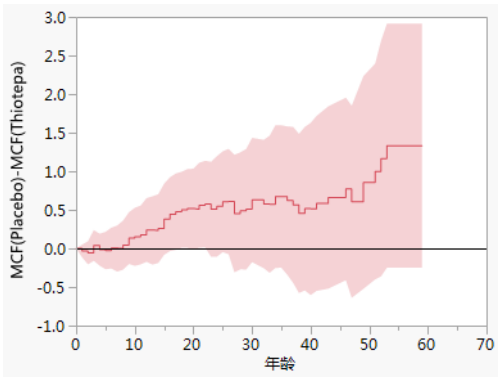
比较分布
分析 > 可靠性和生存 > 寿命分布



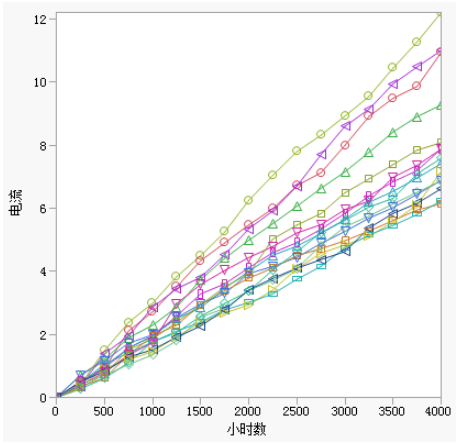
非参数叠加
分析 > 可靠性和生存 > 以 X 拟合寿命



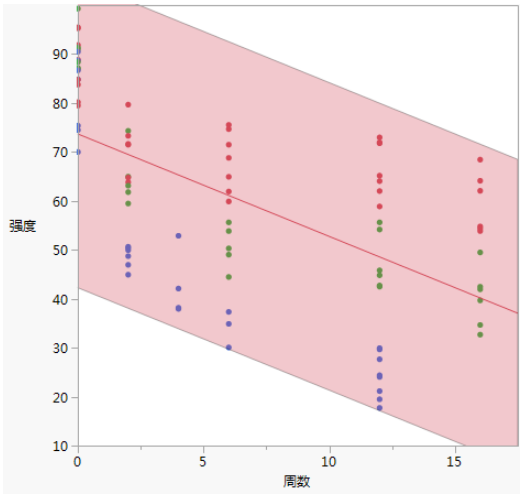
散点图
分析 > 可靠性和生存 > 以 X 拟合寿命



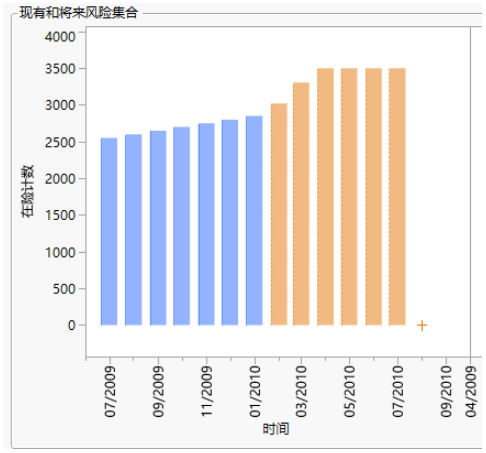
MCF 图
分析 > 可靠性和生存 > 复发分析



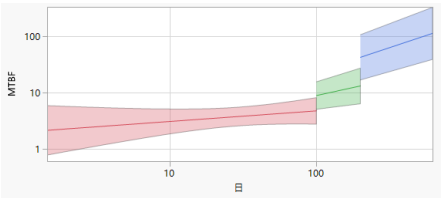
叠加
分析 > 可靠性和生存 > 退化



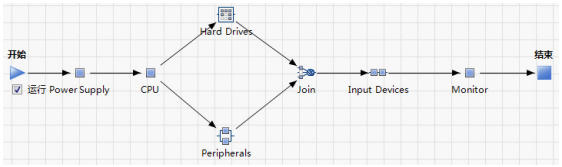
预测区间
分析 > 可靠性和生存 > 破坏性退化



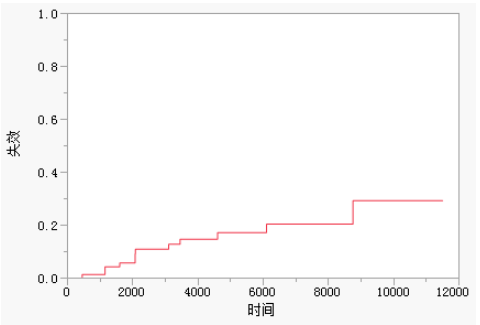
预测
分析 > 可靠性和生存 > 可靠性预测



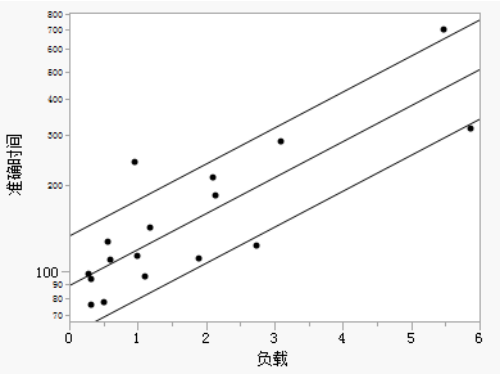
分段 Weibull NHPP
分析 > 可靠性和生存 > 可靠性增长



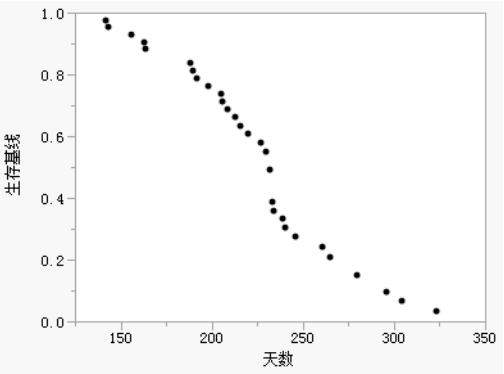
可靠性框图
分析 > 可靠性和生存 > 可靠性方框图



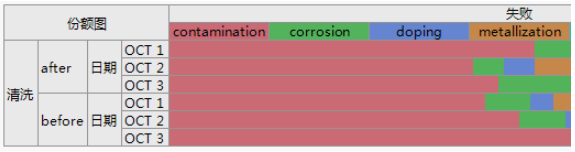
失效图
分析 > 可靠性和生存 > 生存



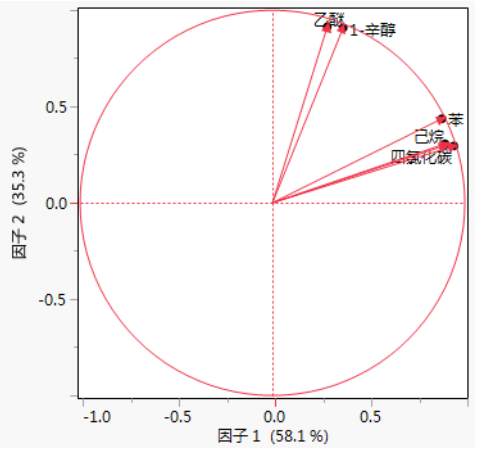
生存分位数
分析 > 可靠性和生存 > 拟合参数生存



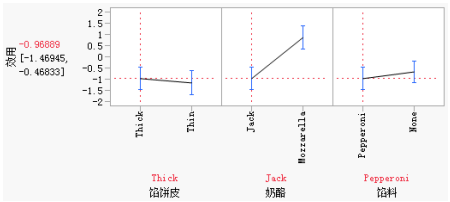
基线生存
分析 > 可靠性和生存 > 拟合比例风险



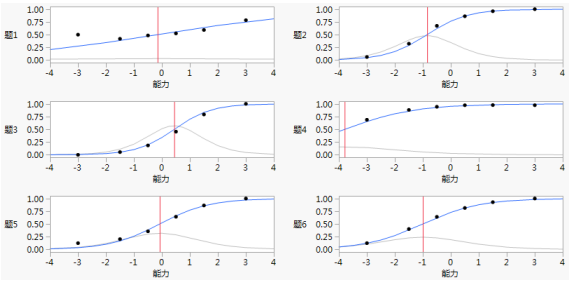
混料刻画器
分析 > 消费者研究 > 分类



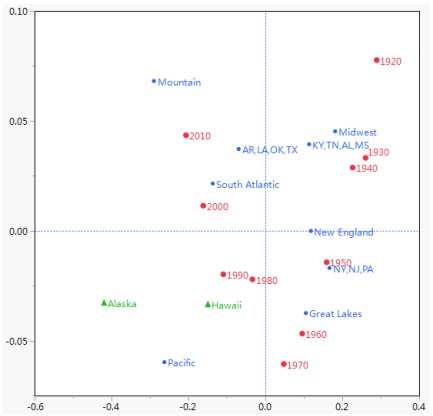
因子载荷图
分析 > 多元方法 > 因子分析



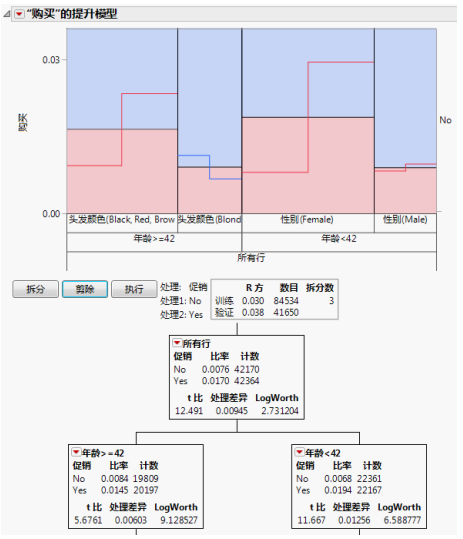
预测刻画
分析 > 消费者研究 > 选择



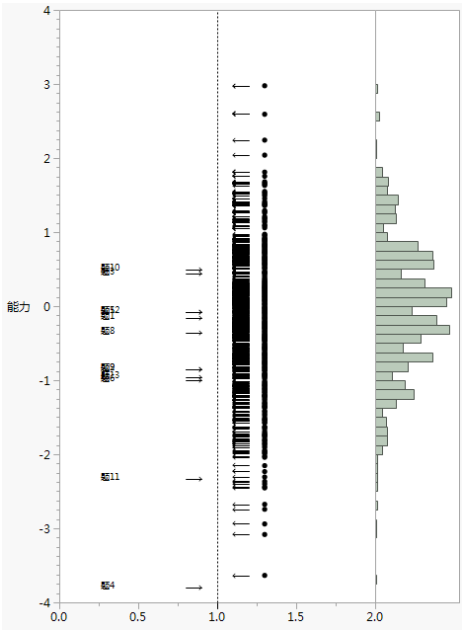
特征曲线
分析 > 多元方法 > 项目分析



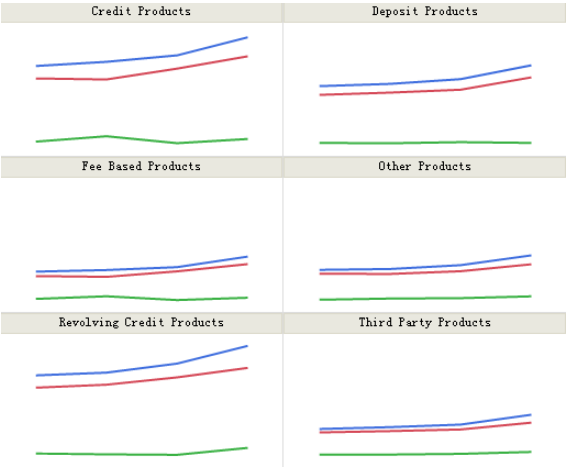
多重对应分析
分析 > 多元方法 > 多重对应分析



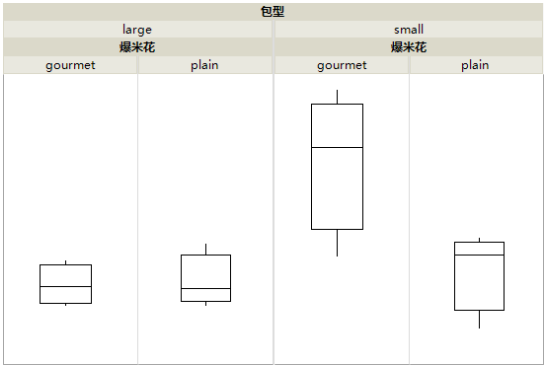
提升模型
分析 > 消费者研究 > 提升



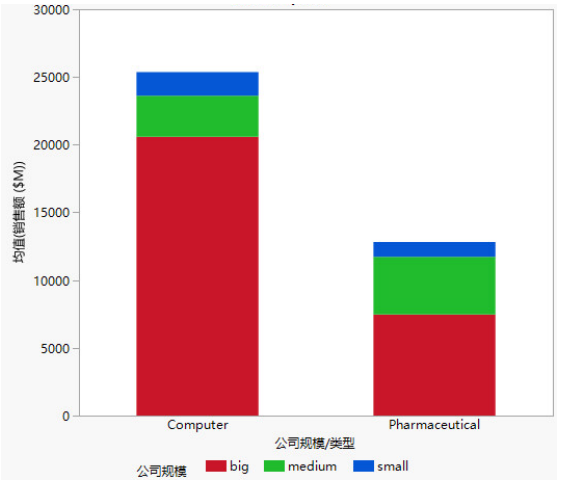
对偶图
分析 > 多元方法 > 项目分析



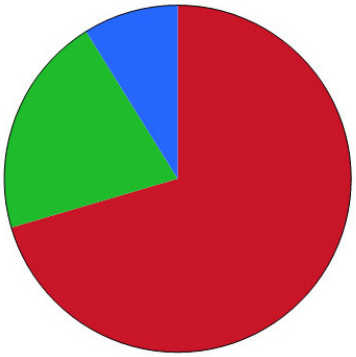
线图
图形 > 图形生成器



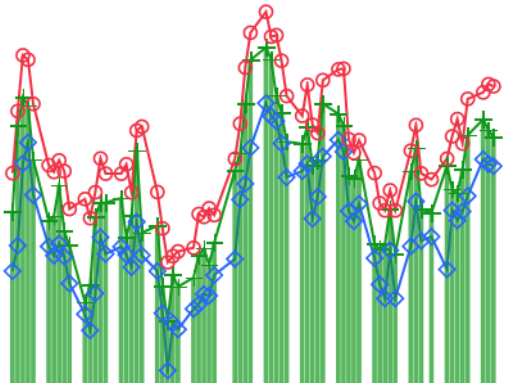
箱线图
图形 > 图形生成器



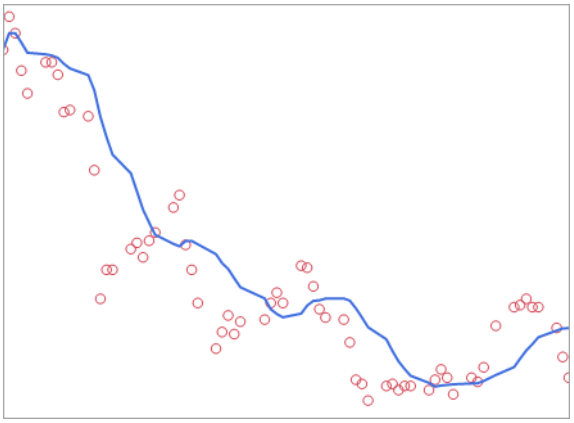
堆叠条形图
图形 > 图形生成器



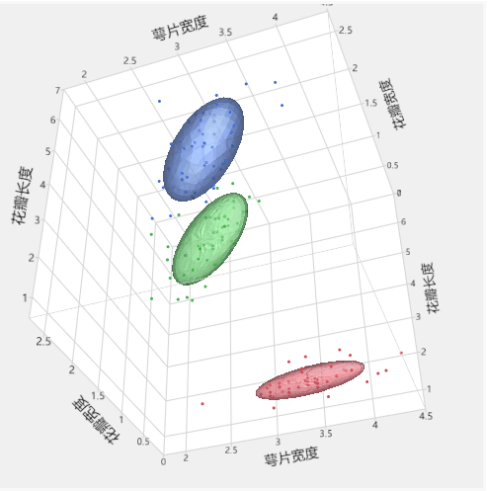
饼图
图形 > 图形生成器



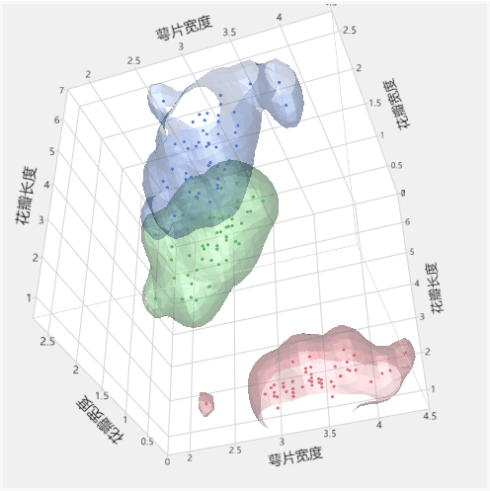
针线图
图形 > 图形生成器



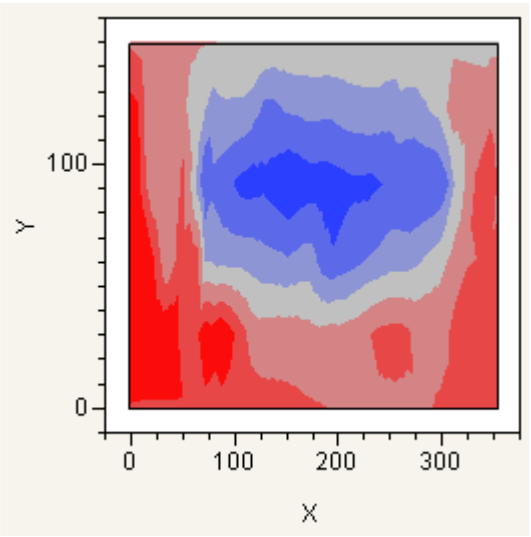
平滑线
图形 > 图形生成器



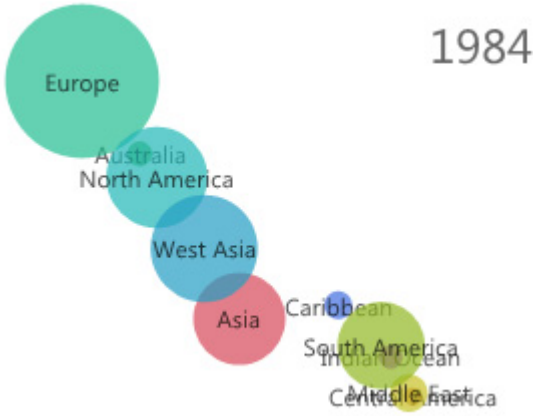
三维散点图
图形 > 三维散点图



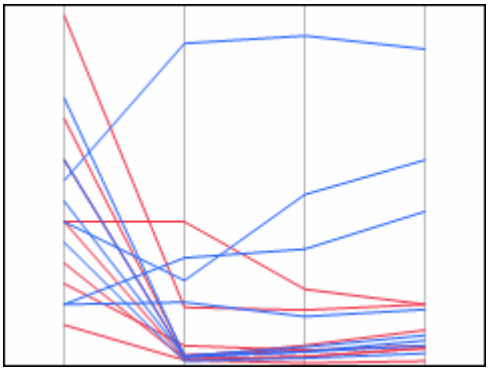
三维散点图
图形 > 三维散点图



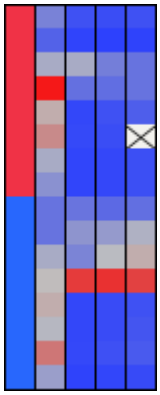
等高线图
图形 > 图形生成器



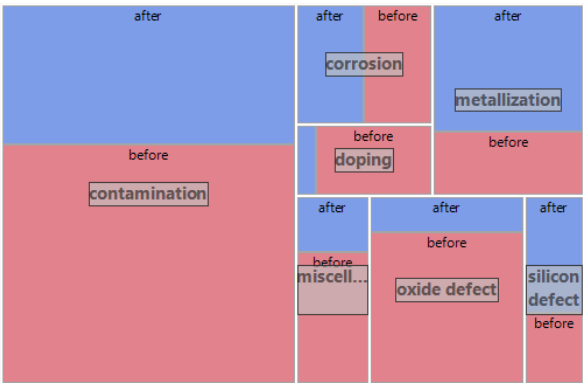
气泡图
图形 > 气泡图



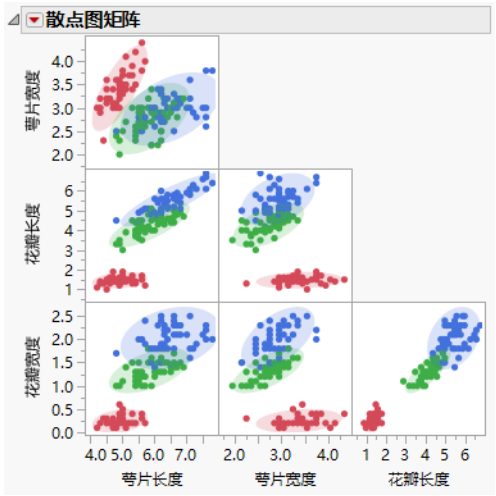
平行图
图形 > 图形生成器



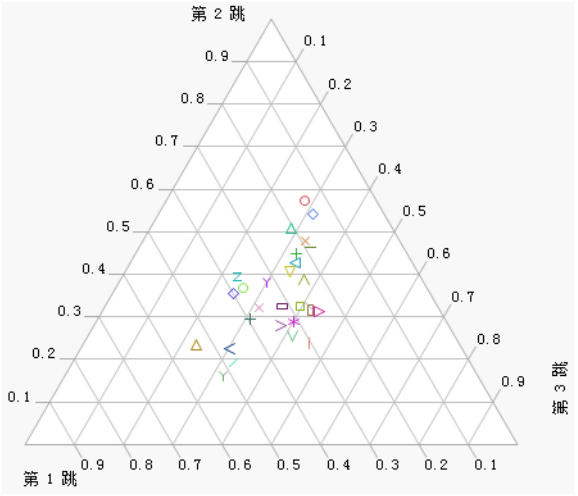
方格图
图形 > 方格图



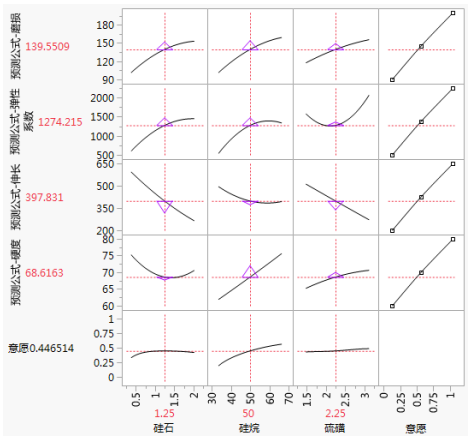
矩形树图
图形 > 图形生成器



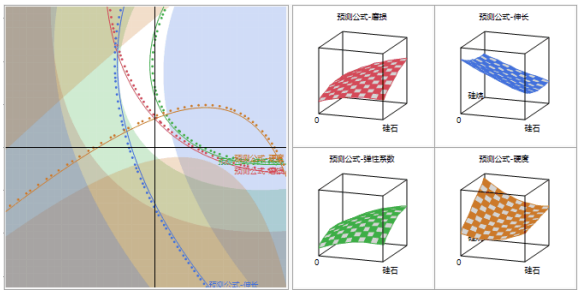
散点图矩阵
图形 > 散点图矩阵



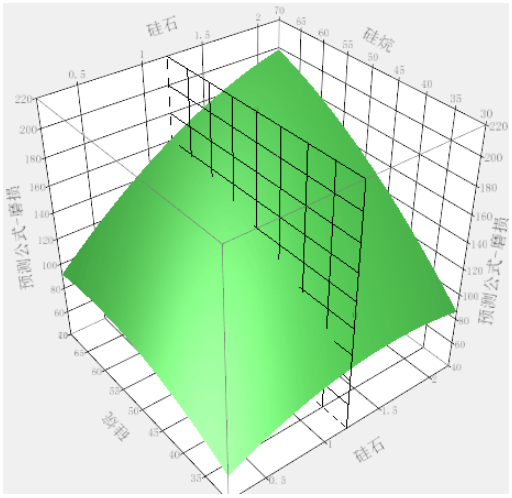
三元图
图形 > 三元图



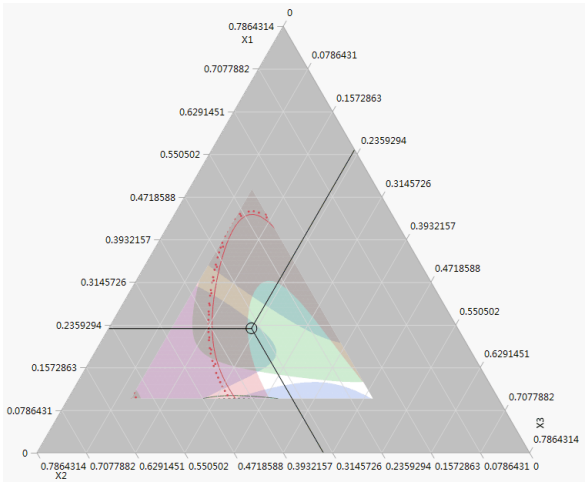
预测刻画器
图形 > 刻画器



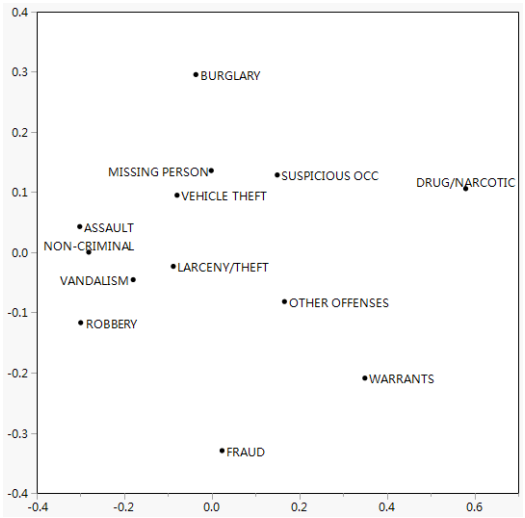
等高线刻画器
图形 > 等高线刻画器



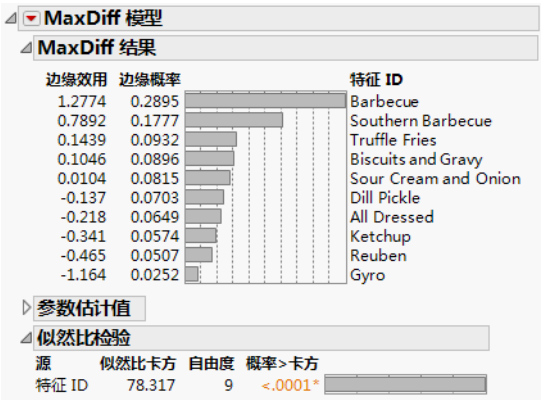
曲面图
图形 > 曲面图



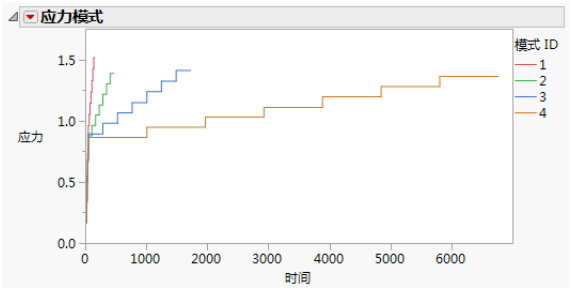
混料刻画器
图形 > 混料刻画器



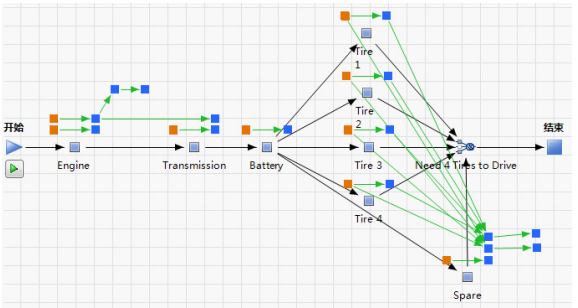
多维尺度化
分析 > 多元方法 > 多维尺度化



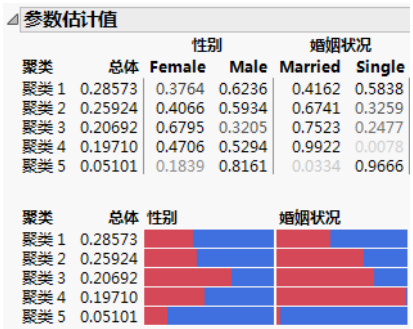
MaxDiff
分析 > 消费者研究 > MaxDiff



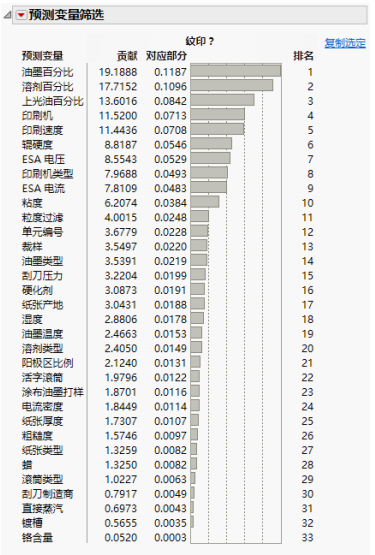
应力模式图
分析 > 可靠性和生存 > 累积损坏



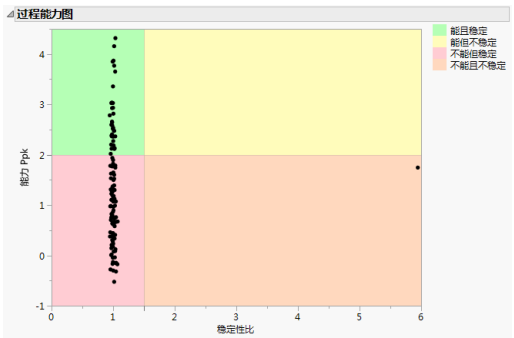
可修复系统模拟
分析 > 可靠性和生存 > 可修复系统模拟



潜在类分析
分析 > 聚类 > 潜在类分析

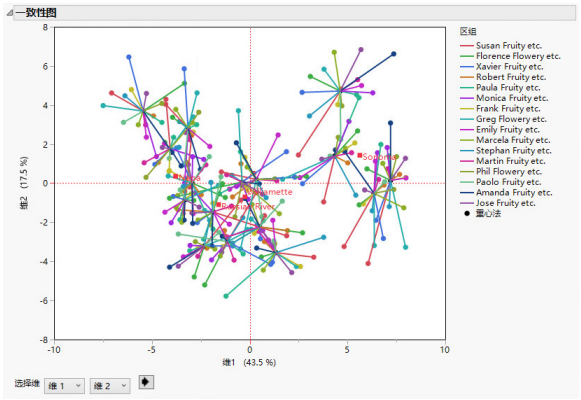


预测变量筛选
分析 > 筛选 > 预测变量筛选

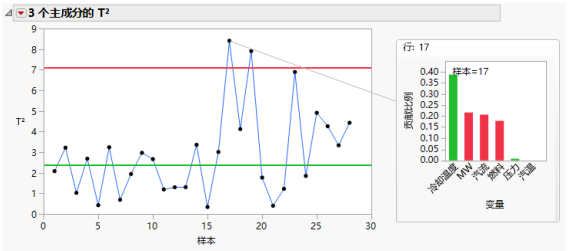


过程筛选
分析 > 筛选 > 过程筛选

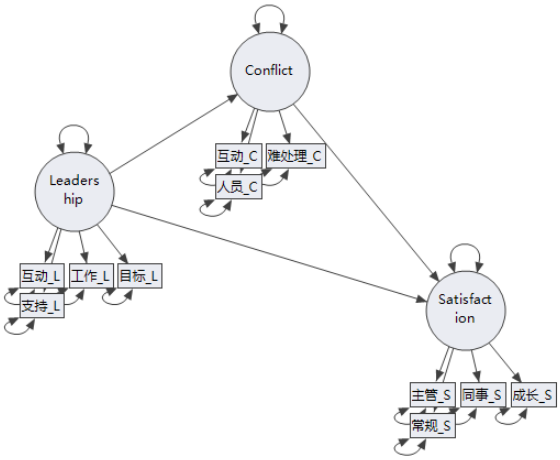
文本分析器
分析 > 文本分析器



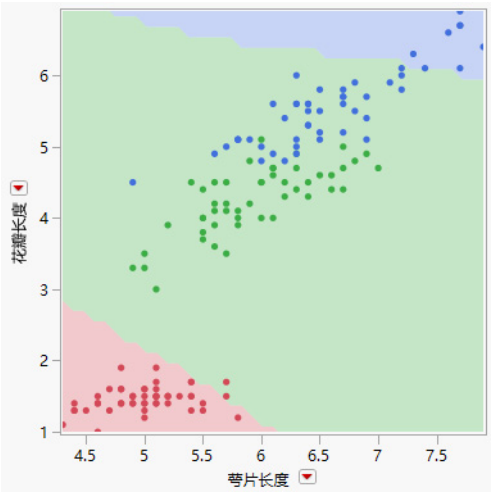
多因子分析
分析 > 消费者研究 > 多重因子分析



模型驱动的多元控制图
分析 > 质量和过程 > 模型驱动的多元控制图



结构化方程模型
分析 > 多元方法 > 结构化方程模型



支持向量机
分析 > 预测建模 > 支持向量机

第 1 章

了解 JMP

文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 35
JMP 帮助 36
JMP 文档库 36
学习 JMP 的其他资源 42
JMP 教程 42
样本数据表 42
学习统计和 JSL 术语 43
学习 JMP 的提示与技巧 43
JMP 工具提示 43
JMP 用户社区 43
免费在线统计思维课程 43
新 JMP 用户欢迎套件 44
统计学知识门户 44
JMP 培训 44
用户编写的 JMP 手册 44
“JMP 起始页”窗口 44
JMP 技术支持 44

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 Lucida Sans Typewriter 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- Helvetica **粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用黑体字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 36 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • Pareto 图 • 关系图 • 管理规格限

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “新 JMP 用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

新 JMP 用户欢迎套件

“新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

JMP 简介

基本概念

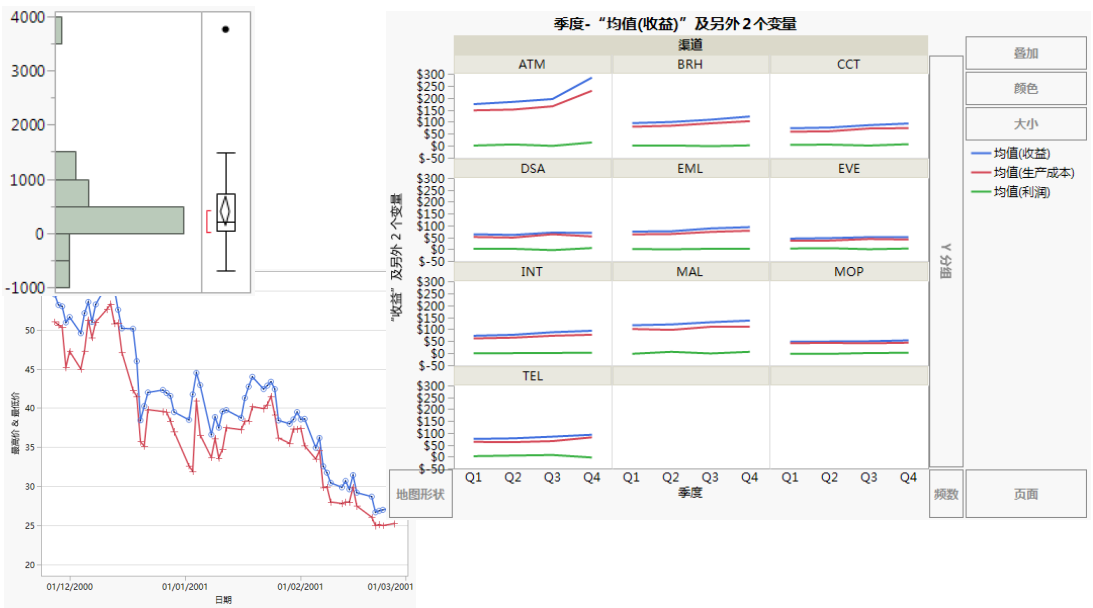
JMP（发音同英文单词 jump）是一款功能强大的交互式数据可视化和统计分析工具。使用 JMP 来执行分析并通过数据表、图形、图表和报表与数据进行交互，可以更多地了解您的数据。

JMP 支持研究人员执行广泛多种的统计分析和建模。业务分析人员同样可以使用 JMP 来快速发现数据的趋势和模式。您不必成为统计专家，借助 JMP 便能从数据中获取信息。

例如，您可以使用 JMP 执行下列操作：

- 创建交互式图形和图表以探索数据并发现数据之间的关系。
- 同时发现多个变量的变异模式。
- 探索和汇总大量数据。
- 开发强大的统计模型以预测未来。

图 2.1 JMP 报表示例



目录

您应该知道的 JMP 概念 47
JMP 入门帮助 47
启动 JMP 48
使用样本数据 49
了解数据表 51
了解 JMP 工作流程 52
第 1 步：启动 JMP 平台并查看结果 52
第 2 步：从 JMP 报表中删除箱线图 54
第 3 步：请求其他 JMP 输出 54
第 4 步：与 JMP 平台结果交互 55
JMP 与 Excel 有哪些区别？ 57
数据表的结构 57
JMP 中的公式 57
JMP 分析和绘图 58

您应该知道的 JMP 概念

在开始使用 JMP 之前，您应该熟悉下列概念：

- 使用 JMP **数据表**输入、查看、编辑和操控数据。
- 从**分析**、**图形**或**实验设计**菜单中选择一个**平台**。平台包含交互式窗口，可用于分析数据和使用图形。
- 平台使用下列窗口：
  - **启动窗口**，用于设置和运行分析。
  - **报表窗口**，显示分析的输出。
- “报表”窗口通常包含下列项：
  - 某种类型的图形（如散点图或图表）。
  - 特定**报表**，您可以使用**展开图标** 显示或隐藏。
  - 平台**选项**，位于**红色小三角菜单** 中。

JMP 入门帮助

JMP 中的一般工作流程很简单：

1. 使数据出现在 JMP 中。
2. 选择一个平台并完成对应的启动窗口。
3. 研究结果并发现数据内涵。

该工作流程在第 52 页的“了解 JMP 工作流程”中有详细介绍。

通常情况下，您在 JMP 中首先会使用图形来可视化各变量以及变量之间的关系。使用图形可以方便地查看该信息，并且可以提出更深层的问题。然后可以使用分析平台深入挖掘问题并寻求解决方案。

- 第 59 页的“使用数据”一章介绍如何把数据带进 JMP 中。
- 第 87 页的“数据可视化”一章介绍如何使用 JMP 提供的一些很有帮助的图形来更好地了解数据。
- 第 123 页的“分析数据”一章介绍如何使用一些分析平台。
- 第 157 页的“概览”一章说明如何在若干平台中分析分布、模式和相似值。

每章都通过示例进行教学。本章中的下列小节介绍数据表和使用 JMP 所需的一般概念。

启动 JMP

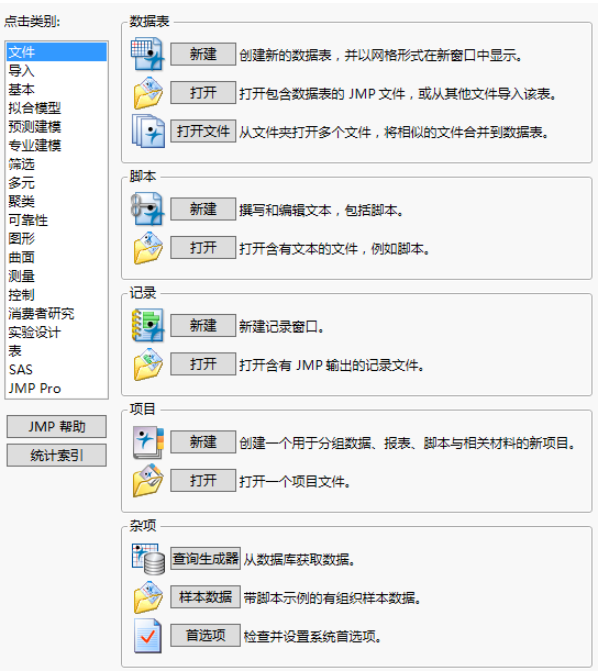
启动 JMP 有两种方式：

- 双击通常位于桌面上的 JMP 图标。该操作将启动 JMP，但不会打开任何现有的 JMP 文件。
- 双击现有的 JMP 文件。该操作将启动 JMP 并打开该文件。

在 Windows 系统中，JMP 的初始视图包括“今日提示”窗口和主窗口；在 macOS 系统中，最初显示“今日提示”和“JMP 起始页”以及主窗口。

“JMP 起始页”窗口使用类别将操作和平台分类。

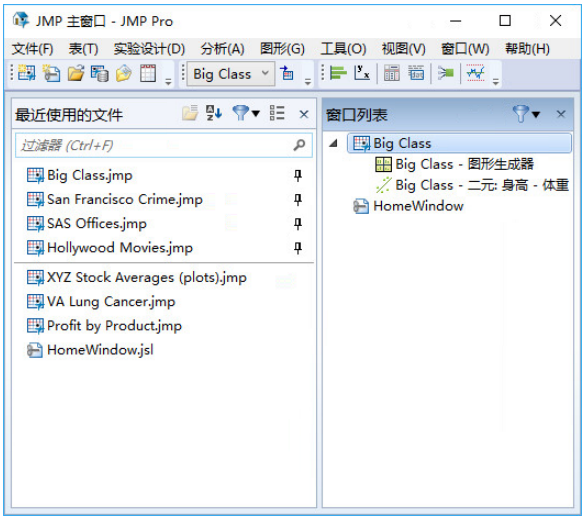
图 2.2 JMP 起始页



左侧是类别列表。点击某个类别可查看与该类别相关的功能和命令。“JMP 起始页”还列出 JMP Pro 功能和平台。

主窗口帮助您在 JMP 中组织和访问文件。

图 2.3 Windows 系统中的主窗口



要打开 Windows 系统中的主窗口，请选择**视图 > 主窗口**。在 macOS 上，选择**窗口 > JMP 主窗口**。主窗口提供下列链接：

- 当前处于打开状态的数据表和报表窗口
- 您最近打开的文件

有关主窗口的详细信息，请参见《使用 JMP》。

几乎所有 JMP 窗口都包含一个菜单栏和一个工具栏。您可以通过三种方式找到大部分 JMP 功能：

- 使用菜单栏
- 使用工具栏按钮
- 使用“JMP 起始页”窗口中的按钮

关于菜单栏和工具栏

菜单和工具栏在许多窗口中都是隐藏状态。要查看它们，请将鼠标悬停在窗口标题栏下方的蓝条上。“JMP 起始页”窗口、主窗口和所有数据表中的菜单始终处于可见状态。

使用样本数据

《发现 JMP》中的示例和其他 JMP 文档使用样本数据表。在 Windows 系统中，样本数据的默认位置是：

C:/Program Files/SAS/JMP/16/Samples/Data

C:/Program Files/SAS/JMPPro/16/Samples/Data

样本数据索引按类别将数据表分组。点击某个展开图标可查看该类别对应的数据表列表，然后点击链接可打开数据表。

macOS 样本数据安装在 /Library/Application Support/JMP/16/Samples/Data 中。

打开 JMP 样本数据表

1. 从**帮助**菜单中，选择**样本数据**。
2. 点击**发现 JMP 中使用的数据表**列表旁边的展开图标将其打开。
3. 点击数据表的名称，以在 《发现 JMP》的示例中使用。

样本导入数据

使用其他应用程序中的文件可以学习如何将数据导入 JMP。

在 Windows 系统中，样本导入数据的默认位置是：

C:/Program Files/SAS/JMP/16/Samples/Import Data

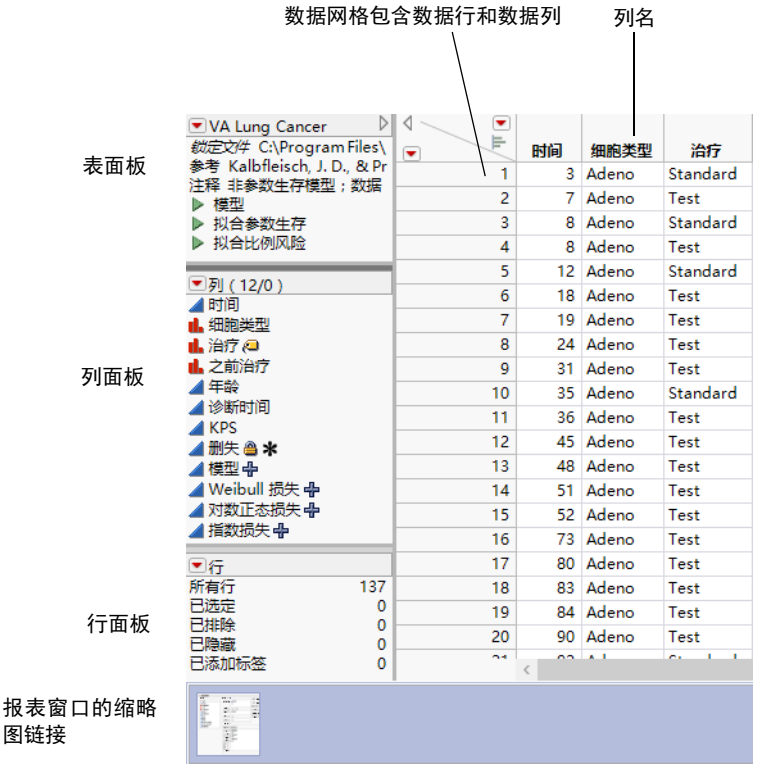
C:/Program Files/SAS/JMPPro/16/Samples/Import Data

了解数据表

JMP 数据表是按行和列组织的数据集合。数据表也可能包含其他信息，例如注释、变量和脚本。这些补充项将在后面章节中介绍。

打开 “VA Lung Cancer”数据表可以查看此处说明的数据表。

图 2.4 数据表



数据表包含下面几个部分：

数据网格 数据网格包含按行和列排列的数据。通常情况下，数据网格中的每一行是一个观测，列（也称为变量）提供关于观测的信息。在图 2.4 中，每行对应一个检验对象，共有十二列信息。尽管数据网格中不能显示全部十二列，但“列”面板会将它们全部列出。每个检验对象的信息包括时间、细胞类型、治疗等。每列包含一个标题或名称。该名称不算在表的总行数中。

表面板 表面板可以包含表变量或表脚本。在图 2.4 中，有一个称为**模型**的保存脚本，它会自动重新创建分析。该表还具有一个名为“注释”的变量，它包含关于数据的信息。表变量和表脚本将在后面的章节中介绍。

列面板 列面板显示列总数（无论是否选定列）以及按名称排序的所有列的列表。括号中的数字 (12/0) 显示有十二列，并且没有选定任何列。每个列名左侧的图标显示列的建模类型。建模类型将在“分析数据”一章中第 128 页的“了解建模类型”中介绍。右侧的图标显示指定给该列的特性。有关这些图标的详细信息，请参见“使用数据”一章中第 72 页的“查看或更改数据表中的列信息”。

行面板 行面板显示数据表中的行数，以及已选定、已排除、已隐藏或已添加标签的行数。在图 2.4 中，数据表中有 137 行。

报表窗口的缩略图链接 该区域基于数据表显示所有报表的缩略图。将鼠标悬停在缩略图上可以看到报表窗口的更大的预览。双击缩略图可以将报表窗口置于最前面。

如何与数据网格交互，包括添加行和列、输入数据和编辑数据，将在第 59 页的“使用数据”一章中介绍。若打开多个数据表，每个数据表会显示在单独的窗口中。

有关 JMP 数据表与 Excel 电子表格差别的详细信息，请参见第 57 页的“JMP 与 Excel 有哪些区别？”。

了解 JMP 工作流程

您的数据放入数据表后，便可以创建图形并执行分析。所有功能都位于平台中，基本上都可以在**分析**或**图形**菜单中找到。它们称为平台，因为它们不仅生成简单的统计结果。平台结果会显示在报表窗口中，具有很高的交互性，与数据表链接并且互相链接。

分析和**图形**菜单下的平台提供各种分析功能和数据探索工具。

生成图形或分析的一般步骤如下所示：

1. 打开一个数据表。
2. 从“图形”或“分析”菜单中选择一个平台。
3. 完成平台启动窗口以设置分析。
4. 点击**确定**以创建包含图形和统计分析的报表窗口。
5. 使用报表选项定制报表。
6. 保存、导出，与他人共享结果。

后面章节将详细介绍这些概念。

以下示例介绍如何分四步执行简单的分析并完成定制。该示例使用 Companies.jmp 文件样本数据表显示变量**利润 ($M)** 的基本分析。

第 1 步：启动 JMP 平台并查看结果

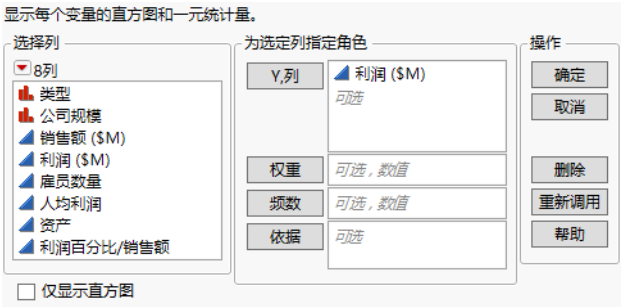
1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**以打开“分布”启动窗口。

3. 在 “选择列”框中选择**利润 ($M)** 并点击 **Y，列**按钮。

   变量**利润 ($M)** 随即出现在 **Y，列**角色中。

   另一种指定变量的方式是从 “选择列”框中点击列并将其拖动到任一角色框中。

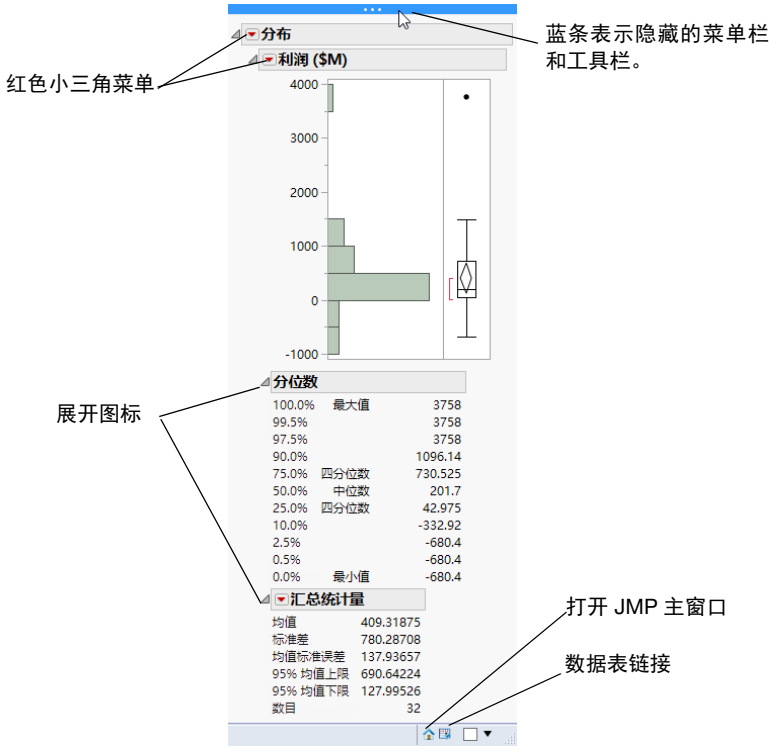
图 2.5 指定**利润 ($M)**



4. 点击**确定**。

   “分布”报表窗口随即出现。

图 2.6 Windows 系统中的 “分布”报表窗口



报表窗口包含基本图形和初步分析报表。结果以分级显示格式显示，您可以通过点击展开图标显示或隐藏任何报表。

红色小三角菜单包含选项和命令，可以随时请求其他图形和分析。

- 在 Windows 系统中，将鼠标悬停在窗口顶部的蓝条上可以查看菜单栏和工具栏。
- 在 Windows 系统中，点击右下角的数据表按钮可以查看用于创建该报表的数据表。在 macOS 系统中，点击报表窗口右上角的**显示数据表**按钮。
- 在 Windows 系统中，点击右下角的 JMP **主窗口**按钮可以查看主窗口。在 macOS 上，选择**窗口** > JMP **主窗口**。

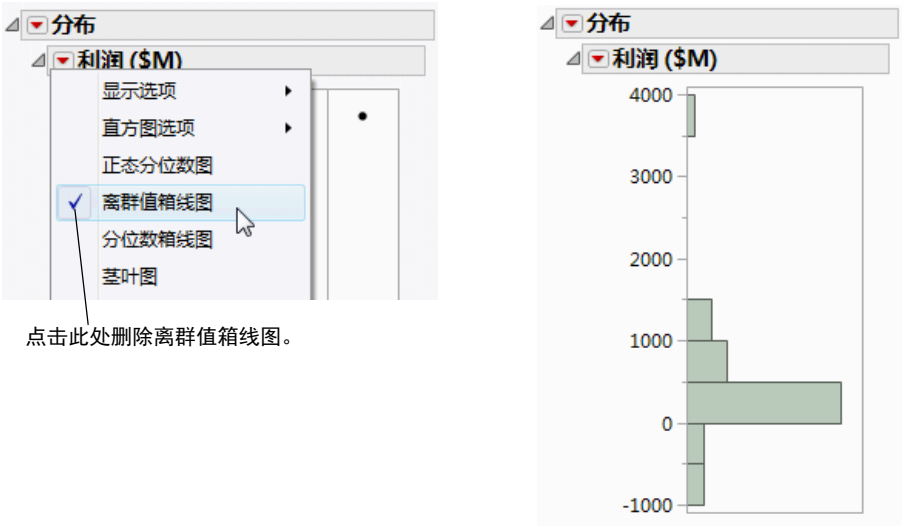
第 2 步：从 JMP 报表中删除箱线图

继续使用您前面创建的 “分布”报表。

1. 点击**利润 ($M)** 旁边的红色小三角，以查看报表选项菜单。
2. 撤销选择**离群值箱线图**以关闭该选项。

   离群值箱线图随即从报表窗口中删除。

图 2.7 删除离群值箱线图



第 3 步：请求其他 JMP 输出

继续使用您在 “第 2 步：从 JMP 报表中删除箱线图”中创建的同一个报表窗口。

1. 点击**利润 ($M)** 旁边的红色小三角并选择**检验均值**。

   “检验均值”窗口随即出现。

2. 在**指定假设均值**框中输入 500。
3. 点击**确定**。

   均值检验随即添加到报表窗口中。

图 2.8 均值检验

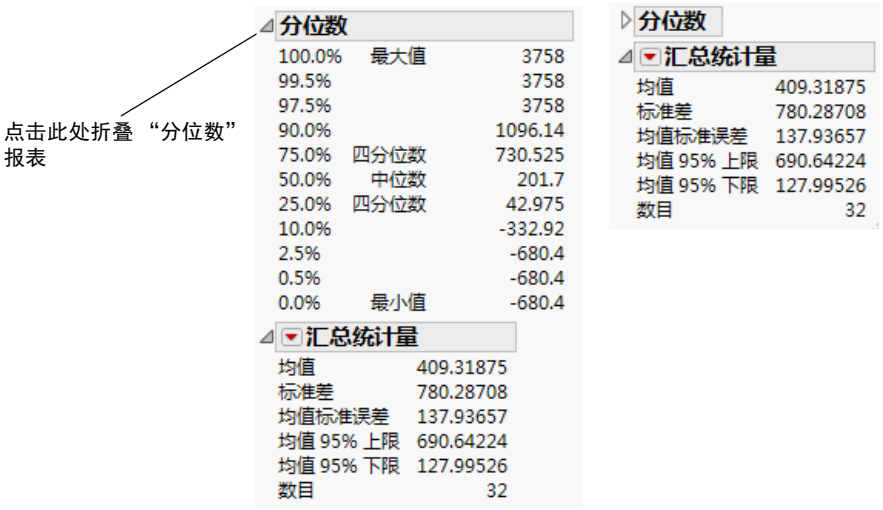
第 4 步：与 JMP 平台结果交互

所有平台生成的结果都是交互的，例如以下结果：

- 可以显示或隐藏报表。
- 可以添加或删除其他图形和统计详细信息以满足您的需要。
- 平台结果可以连接到数据表并且相互连接。

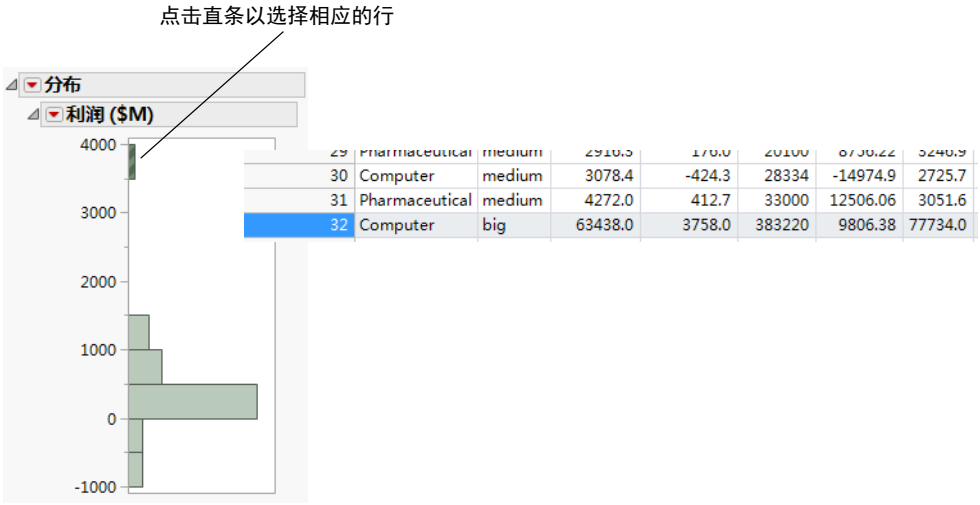
例如，要关闭**分位数**报表，点击**分位数**旁边的展开图标。

图 2.9 关闭分位数报表



平台结果连接到数据表。图 2.10 中的直方图显示一组公司的利润远远高于其他公司。要快速标识该组，点击它们对应的直方图。数据表中相对应的行被选定。

图 2.10 平台结果和数据表之间的连接



在该示例中，组仅包含一家公司，且该行被选定。

JMP 与 Excel 有哪些区别？

JMP 是使用数据表的统计分析程序。Excel 是电子表格应用程序。数据表和电子表格具有不同的结构。

- “数据表的结构”
- “JMP 中的公式”
- “JMP 分析和绘图”

数据表的结构

数据表包含固定的行和列，而电子表格以单元格为基础。在电子表格中，数据、标题或公式可以放置在任何单元格中。在数据表中，结构组织进行分析的数据。该结构由 JMP 分析和绘图平台使用。

列标题 列名即列标题。

列 列包含数据并分配了一个数据类型。基本列为数值型或字符型。若列同时包含字符和数值数据，整列的数据类型为字符，数值会视为字符数据。JMP 还具有专业的列类型来捕获例如图像等内容。JMP 使用列的数据类型来确定分析选项和结果。有关数据类型的详细信息，请参见 “分析数据”一章中第 128 页的 “了解建模类型”。

行 行包含观测。若行没有观测，则该单元格留空。在 JMP 中，点表示缺失数值，空白表示缺失字符值。

有关 JMP 数据表的详细信息，请参见第 51 页的“了解数据表”。有关 JMP 列属性的详细信息，请参见《使用 JMP》手册。

JMP 数据表无法在工作簿（例如在 Excel）中排列。每个 JMP 数据表是一个单独的文件，显示在其自身的窗口中。要合并多个表，请参见《使用 JMP》手册。有关组织 JMP 表和输出的信息，请参见“保存和共享您的工作”一章中第 184 页的“保存和运行脚本”。

提示：要在单个分析中使用两个或多个表中的数据，可以使用“虚拟连接”。有关详细信息，请参见《使用 JMP》手册。

JMP 中的公式

在电子表格中，公式应用于单个单元格，可以利用电子表格中任意单元格（包括工作簿中不同选项卡中的单元格）中的数据。在数据表中，公式应用于整个列。公式可以使用数据表中任何其他列中的数据。列中每一行将基于该行中的数据应用相同的计算。

例如，考虑具有简单求和的数据表，如图 2.11 中所示。列**身高 + 体重**具有公式。该公式为数据表中的所有行将**身高**和**体重**按行相加。

图 2.11 具有公式列的数据表

	姓名	年龄	性别	身高	体重	身高 + 体重
1	KATIE	12	F	59	95	154
2	LOUISE	12	F	61	123	184
3	JANE	12	F	55	74	129
4	JACLYN	12	F	66	145	211
5	LILLIE	12	F	52	64	116
6	TIM	12	M	60	84	144
7	JAMES	12	M	61	128	189

有关 JMP 公式的详细信息，请参见《使用 JMP》手册。

提示：对于基本列汇总统计量，可以使用“分布”平台。请参见《基本分析》手册。

JMP 分析和绘图

JMP 使用平台驱动数据分析。要启动分析，转至“分析”菜单。在平台启动窗口中选择要分析的变量，而分析结果显示在与数据表分开的报表窗口中。这与 Excel 不同，后者将分析插入电子表格中。

绘图选择位于“图形”菜单中。“图形生成器”是开始使用的一个不错的选择。使用“图形生成器”可以拖放列并快速构建图形以探索数据。有关“图形生成器”的详细信息，请参见《基本绘图》手册。

第 3 章

使用数据

为绘图和分析准备数据

在对数据进行绘图或分析之前，数据必须位于数据表中并且使用正确的格式。本章介绍一些基本的数据管理任务，包括以下内容：

- 创建新的数据表
- 打开现有的数据表
- 将数据从其他应用程序中导入 JMP
- 管理数据

图 3.1 数据表示例

Companies
锁定文件 C:\Program Files
注释 数据取自 1990 年

列 (8/0)
类型
公司规模
销售额 ($M)
利润 ($M)
雇员数量
人均利润
资产
利润百分比/销售额

行
所有行 32
已选定 0
已排除 0
已隐藏 0
已添加标签 0

	类型	公司规模	销售额 ($M)	利润 ($M)	雇员数量	人均利润	资产	利润百分比/销售额
1	Computer	small	855.1	31.0	7523	4120.70	615.2	3.63
2	Pharmaceutical	big	5453.5	859.8	40929	21007.11	4851.6	15.77
3	Computer	small	2153.7	153.0	8200	18658.54	2233.7	7.10
4	Pharmaceutical	big	6747.0	1102.2	50816	21690.02	5681.5	16.34
5	Computer	small	5284.0	454.0	12068	37620.15	2743.9	8.59
6	Pharmaceutical	big	9422.0	747.0	54100	13807.76	8497.0	7.93
7	Computer	small	2876.1	333.3	9500	35084.21	2090.4	11.59
8	Computer	small	709.3	41.4	5000	8280.00	468.1	5.84
9	Computer	small	2952.1	-680.4	18000	-37800.0	1860.7	-23.05
10	Computer	small	784.7	89.0	4708	18903.99	955.8	11.34
11	Computer	small	1324.3	-119.7	13740	-8711.79	1040.2	-9.04
12	Pharmaceutical	medium	4175.6	939.5	28200	33315.60	5848.0	22.50
13	Computer	big	11899.0	829.0	95000	8726.32	10075.0	6.97
14	Computer	small	873.6	79.5	8200	9695.12	808.0	9.10
15	Pharmaceutical	big	9844.0	1082.0	83100	13020.46	7919.0	10.99
16	Pharmaceutical	small	969.2	227.4	3418	66530.13	784.0	23.46
17	Pharmaceutical	medium	6698.4	1495.4	34400	43470.93	6756.7	22.32
18	Computer	big	5956.0	412.0	56000	7357.14	4500.0	6.92
19	Pharmaceutical	big	5903.7	681.1	42100	16178.15	8324.8	11.54
20	Computer	medium	2959.3	252.8	31404	8049.93	5611.1	8.54
21	Pharmaceutical	small	1198.3	86.5	8527	10144.25	1791.7	7.22

目录

使数据出现在 JMP 中 61
将数据复制并粘贴到数据表 61
将数据导入数据表 61
在数据表中输入数据 63
将数据从 Excel 传输至 JMP 65
使用数据表 67
编辑数据表中的数据 67
选择、取消选择及查找数据表中的值 69
查看或更改数据表中的列信息 72
使用公式计算值的示例 74
在报表中过滤数据的示例 75
重整数据的示例 77
查看汇总统计量 77
创建子集的示例 81
连接数据表的示例 83
对数据排序 85

使数据出现在 JMP 中

JMP 提供了多种将数据导入 JMP 的方法。

- 要从其他应用程序中复制和粘贴数据，请参见第 61 页的“将数据复制并粘贴到数据表”。
- 要从其他应用程序中导入数据，请参见第 61 页的“将数据导入数据表”。
- 要在数据表中直接输入数据，请参见第 63 页的“在数据表中输入数据”。

您还可以将数据从数据库导入 JMP。请参见《使用 JMP》。

本章使用随 JMP 安装的样本数据表和样本导入数据。要找到这些文件，请参见“JMP 简介”一章中第 49 页的“使用样本数据”。

将数据复制并粘贴到数据表

您可以从其他应用程序中（例如 Microsoft Excel 或文本文件）通过复制和粘贴操作将数据移动到 JMP 中。

1. 在 Microsoft Excel 中打开 VA Lung Cancer.xlsx 文件。该文件位于 Sample Import Data 文件夹中。
2. 选择所有行和列，包括列名。共有 12 列和 138 行。
3. 复制选定的数据。
4. 在 JMP 中，选择**文件** > **新建** > **数据表**以创建一个空表。
5. 选择**编辑** > **带列名一起粘贴**以粘贴数据和列标题。

   若您粘贴到 JMP 中的数据**没有**列名，则可以使用**编辑** > **粘贴**。

将数据导入数据表

您可以从其他应用程序中（例如 Microsoft Excel、SAS 或文本文件）导入数据将其移动到 JMP 数据表中。以下是导入数据的基本步骤：

1. 选择**文件** > **打开**。
2. 浏览到您的文件位置。
3. 若您的文件没有列在“打开数据文件”窗口中，请从**文件类型**菜单中选择正确的文件类型。
4. 点击**打开**。

示例：导入 Microsoft Excel 文件

1. 选择**文件** > **打开**。
2. 导航至 Samples/Import Data 文件夹。
3. 选择 Team Results.xlsx。

请注意数据的起始行和列。该电子表格还包含两个工作表。在本例中，您导入的是 Ungrouped Team Results 工作表。

4. 点击**打开**。

   电子表格在 “Excel 导入向导”中打开，在其中显示数据预览以及导入选项。

   电子表格第一行中的文本为列标题。但是，您希望电子表格第 3 行中的文本转换为列标题。

5. 在**列标题起始于行**旁边，键入 3，然后按下 Enter 键。数据预览中的列标题随之更新。数据第一行的值更新为 4。
6. 仅对该工作表保存设置：
   - 撤销选择窗口左下角中的**用于所有工作表**。
   - 选择窗口右上角中的 Ungrouped Team Results。
7. 点击**导入**按您的指定转换电子表格。

您导入 Excel 文件时， JMP 会预测列标题是否存在，以及列名是否位于第一行。在下列情况下建议使用复制和粘贴方法：

- 列名位于第一行之外的行
- 文件不包含列名并且数据不是从第一行开始
- 文件包含列名并且数据不是从第二行开始

请参见第 61 页的“将数据复制并粘贴到数据表”和《使用 JMP》获取有关导入 Excel 文件的详细信息。

示例：导入文本文件

导入文本文件的一种方式是让 JMP 假设数据格式并将数据放入数据表中。该方法使用您在“首选项”中指定的设置。请参见《使用 JMP》获取有关设置文本导入首选项的信息。

导入文本文件的另一种方式是使用 “文本预览”窗口查看导入后的数据表，并进行调整。下面的示例显示如何使用 “文本导入预览”窗口。

1. 选择**文件 > 打开**。
2. 导航至 Samples/Import Data 文件夹。
3. 选择 Animals_line3.txt。
4. 在 “打开”窗口的底部，选择**数据预览**。
5. 点击**打开**。

图 3.2 初始预览窗口

该文本文件在第一行有标题，列名位于第三行，数据从第四行开始。若您直接在 JMP 中打开该文件，则 “Animals Data” 行将是第一个列名，后面的所有列名和数据将会与文本文件不一致。使用 “预览” 窗口可以在打开文件之前调整设置，并且可以查看调整如何影响最终的数据表。

6. 在**文件在以下行中包含列名**字段中输入 3。
7. 在**数据开始于行**字段中输入 4。
8. 点击**下一步**。

   在第二个窗口中，您可以排除导入的列并更改列的数据建模方式。本示例使用默认设置。
9. 点击**导入**。

新的数据表包含名为 species、subject、miles 和 season 的各列。species 和 season 列是字符数据。subject 和 miles 列是连续数值数据。

提示：您可以同时导入多个文本文件以创建数据表。请参见《使用 JMP》。

在数据表中输入数据

您可以直接在数据表中输入数据。下面的示例显示如何将几个月内收集的数据输入到数据表中。

情境

表 3.1 显示一种新型血压药物的研究数据。每一个体的血压均进行为期六个月的测量。使用了两种药物剂量（300mg 和 450mg），此外还包括对照组和安慰剂组。数据显示每组的平均血压。

表 3.1 血压数据

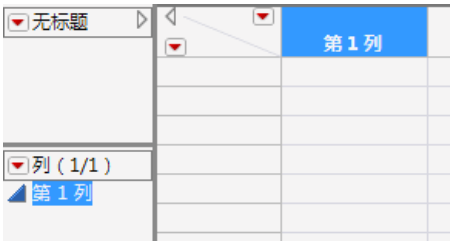
月	控制	安慰剂	300mg	450mg
三月	165	163	166	168
四月	162	159	165	163
五月	164	158	161	153
六月	162	161	158	151
七月	166	158	160	148
八月	163	158	157	150

在新的数据表中输入数据

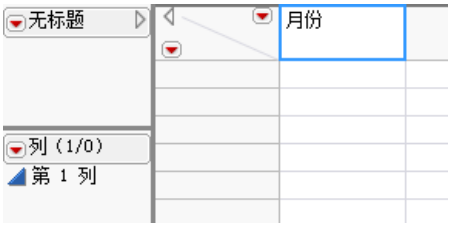
1. 选择**文件 > 新建 > 数据表**以创建空数据表。
   新的数据表包含一列但没有行。
2. 选择列名并将名称改为**月份**。

注意：要重命名列，您还可以双击列名，或选择列并按 Enter。

图 3.3 输入列名



点击一次以选择列。

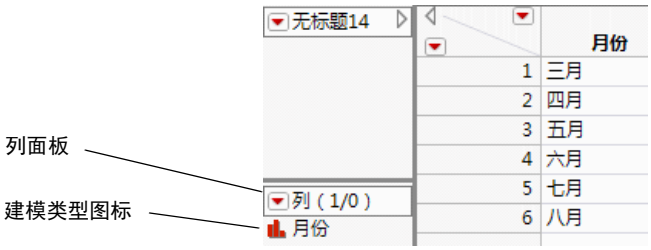


然后键入 “月份”。

3. 选择**行 > 添加行**。
   出现 “添加行”窗口。
4. 因为您想添加六行，键入 6。
5. 点击**确定**。六个空行添加到数据表中。

6. 点击单元格并键入内容，输入**月份**信息。

图 3.4 **月份**列已完成



在列面板中，查看列名左侧的建模类型图标。图标发生了更改，反映出**月份**现在是名义型（以前是连续型）。比较图 3.3 中显示的“列 1”的建模类型和图 3.4 中显示的“月份”的建模类型。该处不同很重要，将在第 72 页的 “查看或更改数据表中的列信息”中讨论。

7. 双击 “月份”列右侧的空白区域，以添加**对照**列。
8. 将名称更改为**对照**。
9. 输入表 3.1 中所示的**对照**数据。您现在的数据由六行和两列组成。
10. 继续按照表 3.1 中所示添加列并输入数据，以创建最终包含六行和五列的数据表。

更改数据表名称

1. 在“表面板”中双击数据表的名称（无标题）。
2. 键入新的名称 （血压）。

图 3.5 更改数据表名称

将数据从 Excel 传输至 JMP

您可以使用 Excel 的 JMP 插件将电子表格从 Excel 传输至 JMP：

- 数据表
- 图形生成器
- “分布”平台
- “以 X 拟合 Y”平台
- “拟合模型”平台

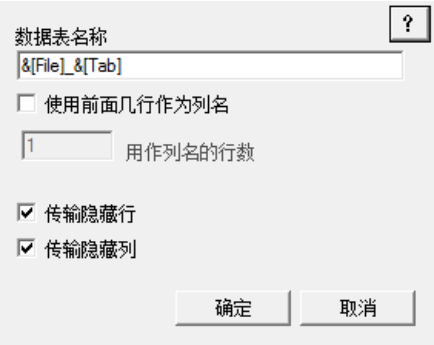
- “时间序列”平台
- “控制图”平台

在 Excel 中设置 JMP 插件首选项

要配置 JMP 插件首选项，请执行以下步骤：

1. 在 Excel 中，选择 JMP > **首选项**。
   “JMP 首选项”窗口随即出现。

图 3.6 JMP 插件首选项



2. 接受默认的**数据表名称**（文件名 _ 工作表名称）或键入一个名称。
3. 若工作表中的第一行包含列标题，请选择**使用前面几行作为列名**。
4. 若选择使用前几行作为列标题，请键入要使用的行数。
5. 若工作表包含要包括在 JMP 数据表中的隐藏行，请选择**传输隐藏行**。
6. 若工作表包含要包括在 JMP 数据表中的隐藏列，请选择**传输隐藏列**。
7. 点击**确定**保存您的首选项。

传输至 JMP

要将 Excel 工作表传输至 JMP，请执行以下步骤：

1. 打开 Excel 文件。
2. 选择要传输的工作表。
3. 选择 JMP，然后选择 JMP 目标：
   - 数据表
   - 图形生成器
   - “分布”平台
   - “以 X 拟合 Y”平台

– “拟合模型”平台
– “时间序列”平台
– “控制图”平台

Excel 工作表将在 JMP 中作为数据表打开，并且随之显示选定平台的启动窗口。

使用数据表

本节介绍使用数据表的基本概念。

- “编辑数据表中的数据”
- “选择、取消选择及查找数据表中的值”
- “查看或更改数据表中的列信息”
- “使用公式计算值的示例”
- “在报表中过滤数据的示例”

提示：考虑将“常规”首选项中的**自动保存超时**值设置为在到达指定的分钟数时自动保存打开的数据表。该自动保存值还适用于记录、脚本、项目和报表。

编辑数据表中的数据

您可以输入或更改数据，一次对几个单元格操作或对整列进行操作。本节包括下列信息：

- “更改数据表单元格中的值”
- “对值重新编码”
- “创建具有一定模式的数据”

更改数据表单元格中的值

要更改一个值，选择单元格并键入更改。也可以双击单元格进行编辑。

注意：双击单元格与选择单元格不是一个操作。单击会选择单元格。您可以同时选择多个单元格，并对选定的单元格执行某些操作。双击只是可以编辑单元格。有关选择行、列和单元格的详细信息，请参见第 69 页的“选择、取消选择及查找数据表中的值”。

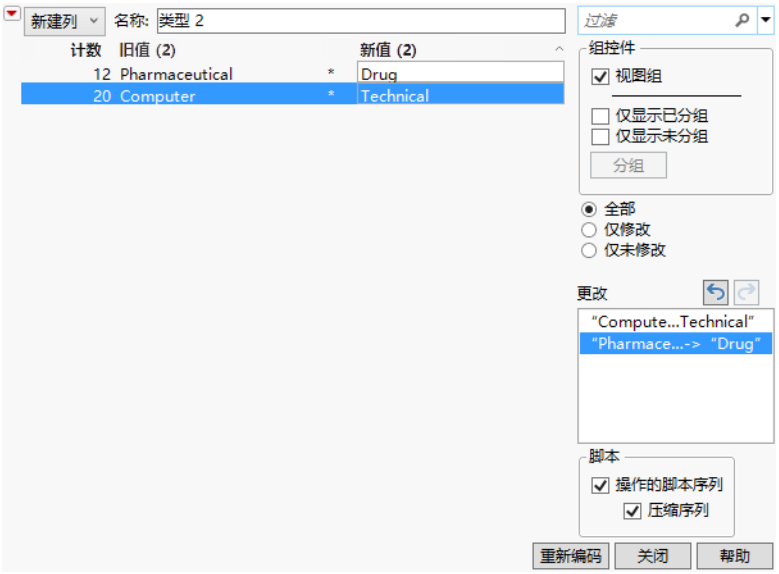
对值重新编码

使用重新编码工具可以同时更改一列中的所有值。例如，假设您对比较计算机和制药公司的销售额感兴趣。您当前的公司标签为“Computer”和“Pharmaceutical”，您想将它们更改为

“Technical”和“Drug”。对 32 行中的所有数据逐步更改所有值将会十分繁琐低效，而且容易出错，特别是您的数据行数过多的情况下。重新编码会是一个更好的选择。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 通过点击列标题一次，选择**类型**列。
3. 选择**列** > **重新编码**。
4. 在“重新编码”窗口的“新值”列中，在“Computer”行中键入“Technical”并且在“Pharmaceutical”行中键入“Drug”。
5. 从“新建列”列表中选择**原位**。
6. 点击**重新编码**。

图 3.7 重新编码窗口



所有单元格都自动更新为新的值。

创建具有一定模式的数据

使用“填充”选项可以使用具有一定模式的数据填充列。若您的数据表很大，并且为每一行键入值会很麻烦的话，“填充”选项将特别有用。

示例：使用具有一定模式的数据填充列

1. 添加一个新列。
2. 在第一个单元格中输入 1，在第二个单元格中输入 2，在第三个单元格中输入 3。
3. 选择这三个单元格，并右击选定单元格中的任何位置以查看菜单。

4. 选择**填充 > 重复序列至表末**。

   剩下的列按 (1, 2, 3, 1, 2, 3, ...) 的顺序填充。

要继续模式 (1, 2, 3, 4, 5, 6, ...) 而不是重复模式，选择**继续序列至表末**。该命令也可用于生成类似于 (1, 1, 1, 2, 2, 2, 3, 3, 3, ...) 的模式。

“填充”选项可以识别简单的算术和几何序列。对于字符数据，“填充”选项仅重复值。

选择、取消选择及查找数据表中的值

您可以在数据表中选择行、列或单元格。例如，要创建现有数据表的子集，您必须首先选择要生成子集的表的某些部分。此外，选择行可以将数据点在图形中突出显示。通过点击、手动选择行和列，或者选择满足特定搜索条件的行。本节包括下列信息：

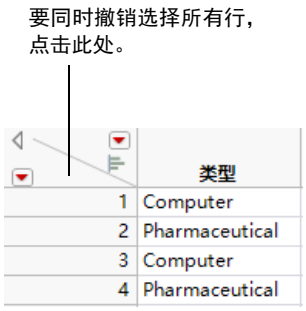
- 第 69 页的“选择和撤销选择行”
- 第 70 页的“选择和撤销选择列”
- 第 71 页的“选择和撤销选择单元格”
- 第 71 页的“搜索值”

选择和撤销选择行

表 3.2 选择和撤销选择行

任务	操作
一次选择一行	点击行号。
选择多个邻接行	点击并拖过行号。 或 选择起始行，按 Shift 键，然后点击最后一个行号。
选择多个非邻接行	选择第一行，按 Ctrl 键，然后点击其他行号。
一次撤销选择一行	按 Ctrl 键，然后点击行号。
撤销选择所有行	点击表左上角的下三角区域（图 3.8）。

图 3.8 撤销选择行

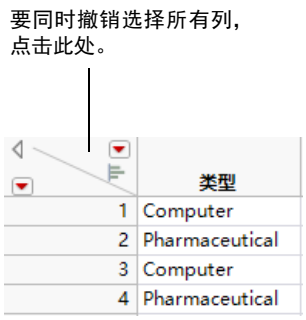


选择和撤销选择列

表 3.3 选择和撤销选择列

任务	操作
一次选择一列	点击列标题。
选择多个邻接列	点击并拖过列标题。 或 选择起始列，按 Shift 键，然后点击最后一个标题。
选择多个非邻接列	选择第一列，按 Ctrl 键，然后点击其他列标题。
一次撤销选择一列	按 Ctrl 键，然后点击列标题。
撤销选择所有列	点击表左上角的上三角区域（图 3.9）。

图 3.9 撤销选择列



选择和撤销选择单元格

表 3.4 选择和撤销选择单元格

任务	操作
一次选择一个单元格	分别点击每个单元格。
选择多个邻接单元格	点击并拖过单元格。 或 选择起始单元格，按 Shift 键，然后点击最后一个单元格。
选择多个非邻接单元格	选择第一个单元格，按 Ctrl 键，然后点击其他单元格。
撤销选择所有单元格	点击表左上角的上三角和下三角区域。

搜索值

在包含成千上万行的数据表中，通过滚动数据表的方式很难找到特定的单元格。若要查找特定的信息，可以使用“搜索”功能进行查找。若找到满足搜索条件的数据，会选定单元格并且数据网格会滚动，以在窗口中显示该单元格。例如，Companies.jmp 数据表包含总销售额为 $11,899 的公司信息。使用“搜索”功能可以查找该单元格。

示例：搜索值

1. 选择**编辑** > **搜索** > **查找**以启动“搜索”窗口。
2. 在**查找内容**框中，输入 11899。
3. 点击**查找**。 JMP 找到第一个值为 11,899 的单元格并将其选定。

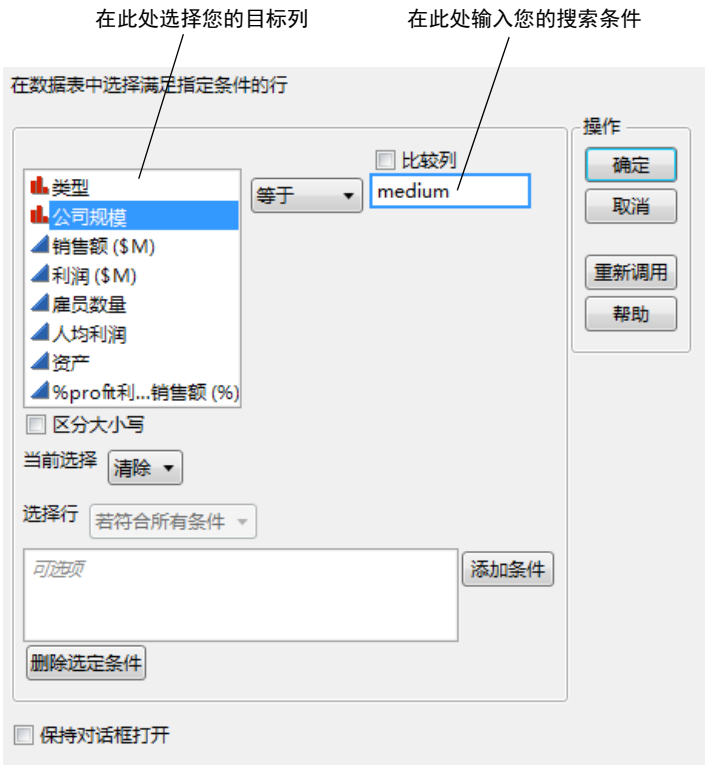
若多个单元格满足搜索条件，再次点击**查找**可查找下一个满足搜索条件的单元格。

您也可以同时搜索多行，每行满足某个条件。

示例：选择对应于中等规模公司的所有行

1. 选择**行** > **行选择** > **选择符合条件的行**以打开**选择行**窗口。
2. 在左侧的列列表框中，选择**公司规模**。
3. 在右侧的文本框中，输入 medium。
4. 点击**确定**。

图 3.10 选择行窗口



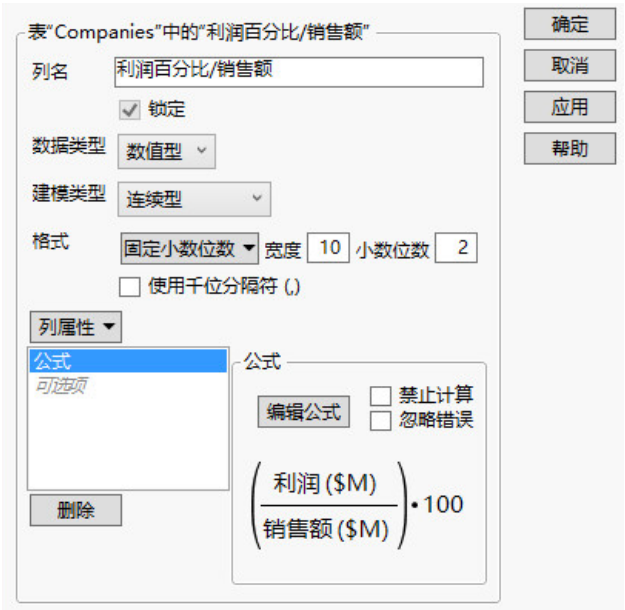
JMP 选择**公司规模**等于“medium”的所有行。共七个。

查看或更改数据表中的列信息

关于数据表列的信息不限于列中的数据。数据类型、建模类型、格式和公式都可以设置。

要查看或更改列特性，双击列标题。或者，右击列标题并选择**列信息**。“列信息”窗口随即出现。

图 3.11 列信息窗口



列名 输入或更改列名。任何两列的列名都不相同。

数据类型 选择下列数据类型之一：

数值 指定列值为数字。

字符 指定列值为非数值，例如字母或符号。

行状态 指定列值为行状态。这是高级主题，请参见《使用 JMP》。

建模类型 建模类型定义在分析中如何使用值。选择下列建模类型之一：

连续型 只能是数字

有序型 可以是数字或字符，并且是有序的类别

名义型 可以是数字或字符，但未排序

格式 选择数值的格式。该选项不适用于字符数据。下面是一些最常见的格式：

最佳 让 JMP 选择最佳的显示格式。

固定小数位数 指定显示的小数位数。

日期 指定日期值的语法。

时间 指定时间值的语法。

货币 指定货币类型和货币值使用的小数点。

列属性 设置特殊的列属性，例如公式、注释和值排序。请参见 《使用 JMP》。

锁定 锁定列，以便无法更改列中的值。

使用公式计算值的示例

使用“公式编辑器”可以创建包含计算值的列。

情境

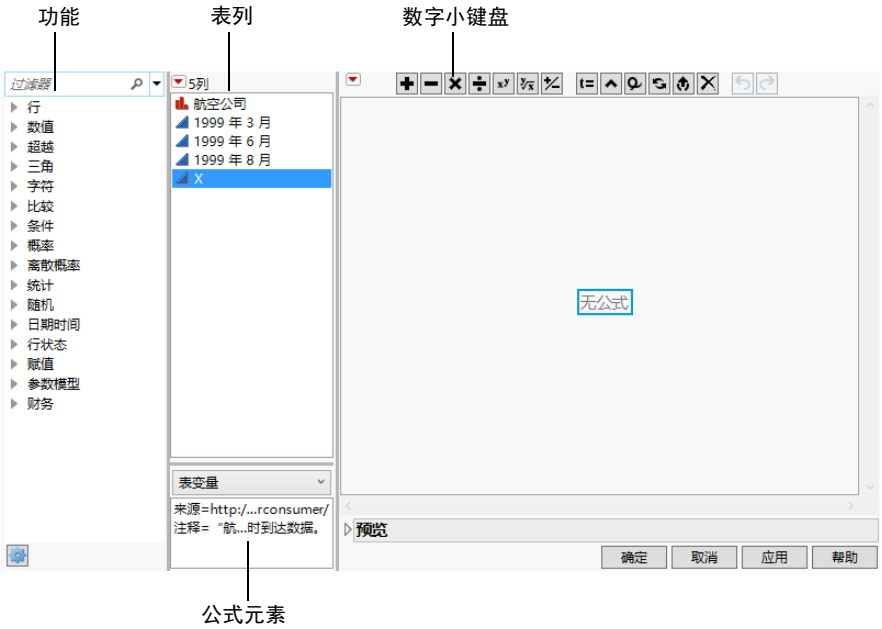
样本数据表 On-Time Arrivals.jmp 反映了几条航线准时到达的百分比。收集了 1999 年 3 月、6 月和 8 月的数据。

创建公式

假设您想创建一个新列，包含每条航线的平均准时到达百分比。

1. 添加一个新列。
2. 右击新列的列标题并选择**公式**。出现 “公式编辑器”窗口。

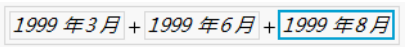
图 3.12 公式编辑器



为每条航线平均准时到达百分比创建公式：

3. 从 “表列”列表中，选择 1999 年 3 月。
4. 点击数字小键盘上的 ✚ 按钮。
5. 选择 1999 年 6 月，然后再跟着另一个 ✚ 号。
6. 选择 1999 年 8 月。

图 3.13 月份总和



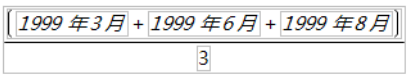
注意：仅选定了 “1999 年 8 月”（被蓝框围住）。

7. 点击围住整个公式的框。

图 3.14 选定了整个公式

8. 点击 ÷ 按钮。
9. 在分母框中键入 3，然后点击公式外部的空白区域。

图 3.15 完成的公式



10. 点击**确定**

新列包含平均值。

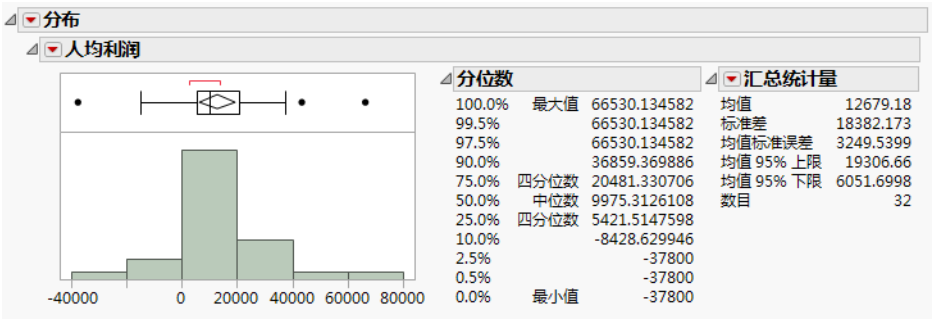
“公式编辑器”有许多内置算术和统计函数。例如，计算平均准时到达百分比的另一种方式是使用“统计”函数列表中的 Mean 函数。有关所有“公式编辑器”函数的详细信息，请参见《使用 JMP》。

在报表中过滤数据的示例

使用**数据过滤器**可以交互式选择复杂的数据子集、在图形中隐藏这些子集，或将其排除不进行分析。例如，查看计算机和制药公司每名雇员的利润。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**人均利润**并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “人均利润”旁边的红色小三角并选择**显示选项** > **水平布局**。

图 3.16 人均利润分布

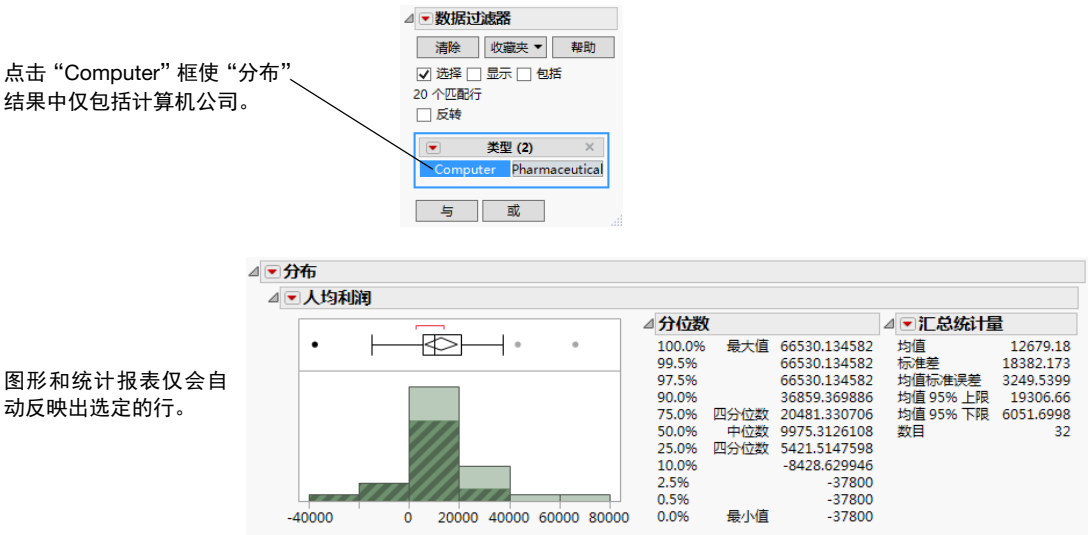


6. 从 “分布” 红色小三角中选择**重新运行 > 自动重新计算**，打开 “自动重新计算” 功能。

   当该选项打开后，您所做的每项更改（例如，隐藏点或排除点）都会使报表窗口本身自动更新。
7. 在数据表中，选择**行 > 数据过滤器**。
8. 选择**类型**并点击**添加**。
9. 确保选中了 “选择”。
10. 要从 “分布” 结果中过滤掉制药公司，使其只包括计算机公司，点击 “数据过滤器” 窗口中的 Computer 框。

    分布结果更新为只包括计算机公司。

图 3.17 计算机公司过滤



相反，要将 “分布” 结果更改为仅包括 “Pharmaceutical” 公司，点击 “数据过滤器” 窗口中的 Pharmaceutical 框。

重整数据的示例

表菜单中的命令 （和**分析**菜单中的 “制表”）可汇总数据表，并将数据表处理成您绘图和分析数据所需的格式。本节说明了五个命令：

汇总 创建一个表，表中包含描述数据的汇总统计量。

制表 提供一个拖动工作区，以创建汇总统计量。

子集 创建一个表，表中包含数据的子集。

连接 将两个数据表中的数据连接成一个新的数据表。

排序 按一列或多列对数据排序。

有关这些命令和其他“表”菜单命令的详细信息，请参见《使用 JMP》。

查看汇总统计量

汇总统计量（例如总和与均值）可以立即提供有关数据的有用信息。例如，若从 32 家公司中查看每家公司的年利润，很难比较大、中、小三种规模的公司利润。汇总可以立即显示该信息。

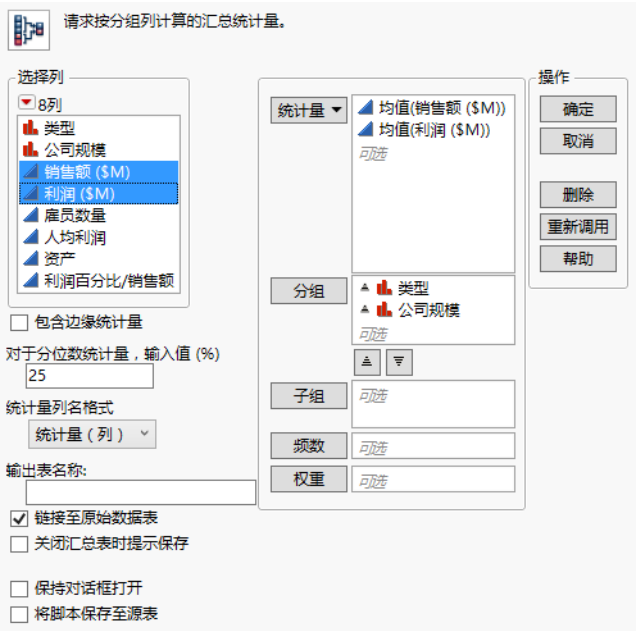
使用**汇总**或**制表**命令，可以创建汇总表。**汇总**命令创建一个新的数据表。对于任何数据表，您可以从汇总表执行分析并创建图形。**制表**命令创建带有汇总数据表的报表窗口。您也可以从“制表”报表创建表。

汇总表示例

汇总表包含分组变量的每个水平的统计量。以计算机和制药公司的财务数据为例，假设您需要计算各种类型和规模的公司的销售额均值和利润均值。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**表** > **汇总**。
3. 选择**类型**和**公司规模**，并点击**分组**。
4. 选择**销售额 ($M)** 和**利润 ($M)**，并点击**统计量** > **均值**。

图 3.18 完成的汇总窗口



5. 点击**确定**。

JMP 为**类型**和**公司规模**的每个组合计算**销售额 ($M)** 和**利润 ($M)** 的均值。

图 3.19 汇总表

	类型	公司规模	行数	均值(销售额 ($M))	均值(利润 ($M))
1	Computer	big	4	20597.48	1089.93
2	Computer	medium	2	3018.85	-85.75
3	Computer	small	14	1758.06	44.94
4	Pharmaceutical	big	5	7474.04	894.42
5	Pharmaceutical	medium	5	4261.06	698.98
6	Pharmaceutical	small	2	1083.75	156.95

汇总表包含下列信息：

- 每个分组变量（在本示例中，**类型**和**公司规模**）都有对应列。
- **行数**列显示原始表中对应于分组变量每个组合的行数。例如，原始数据表包含 14 行对应于小型计算机公司。
- 每个请求的汇总统计量对应一列。在本示例中，**销售额 ($M)** 的均值对应一列，**利润 ($M)** 的均值对应一列。

汇总表链接到源表。在汇总表中选择一行的同时也会在源表中选择相对应的行。

制表示例

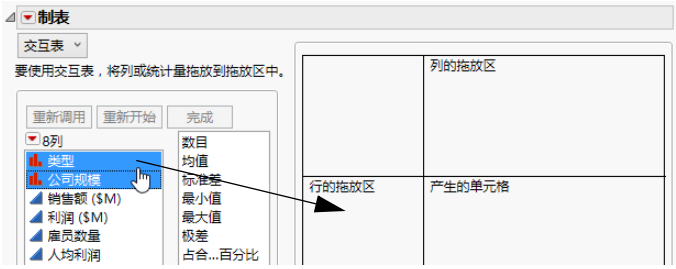
使用 “制表”命令可以将列拖动到工作区中，为分组变量的每个组合创建汇总统计量。该示例显示如何使用 “制表”创建汇总信息，该汇总信息与刚才使用 “汇总”创建的汇总信息相同。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **制表**。

图 3.20 制表工作区

3. 同时选择**类型**和**公司规模**。
4. 将它们拖放到**行的拖放区**。

图 3.21 拖动列到行拖放区



5. 右击标题并选择**嵌套分组列**。

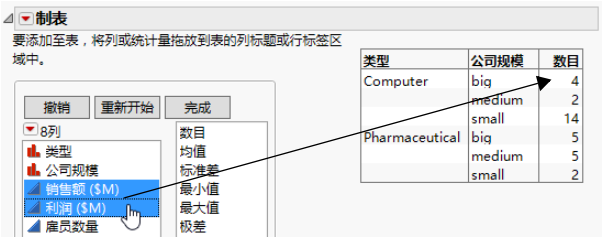
初始制表显示每个分组的行数。

图 3.22 初始制表

类型	公司规模	数目
Computer	big	4
	medium	2
	small	14
Pharmaceutical	big	5
	medium	5
	small	2

6. 同时选择**销售额 ($M)** 和**利润 ($M)**，将它们拖放到表中的**数目**。

图 3.23 添加销售额和利润



制表现在显示每组**销售额 ($M)** 的总和以及**利润 ($M)** 的总和。

图 3.24 总和制表

		销售额 ($M)	利润 ($M)
类型	公司规模	总和	总和
Computer	big	82389.9	4359.7
	medium	6037.7	-171.5
	small	24612.8	629.1
Pharmaceutical	big	37370.2	4472.1
	medium	21305.3	3494.9
	small	2167.5	313.9

7. 最后一步是将总和更改为均值。右击**总和** （任选其一）并选择**统计量 > 均值**。

图 3.25 最终的制表

		销售额 ($M)	利润 ($M)
类型	公司规模	均值	均值
Computer	big	20597.48	1089.9
	medium	3018.85	-85.75
	small	1758.06	44.94
Pharmaceutical	big	7474.04	894.42
	medium	4261.06	698.98
	small	1083.75	156.95

该均值与使用“汇总”命令得到的均值相同。比较图 3.25 和图 3.19。

创建子集的示例

若您仅想特别关注数据表的一部分，可以创建子集。例如，假设您已经比较大、中、小三种规模的计算机和制药公司的销售额和利润，现在仅想查看中等规模公司的销售额和利润。

创建子集需要两步操作。首先选择目标数据，然后将数据提取到新表中。

使用子集命令生成子集

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。

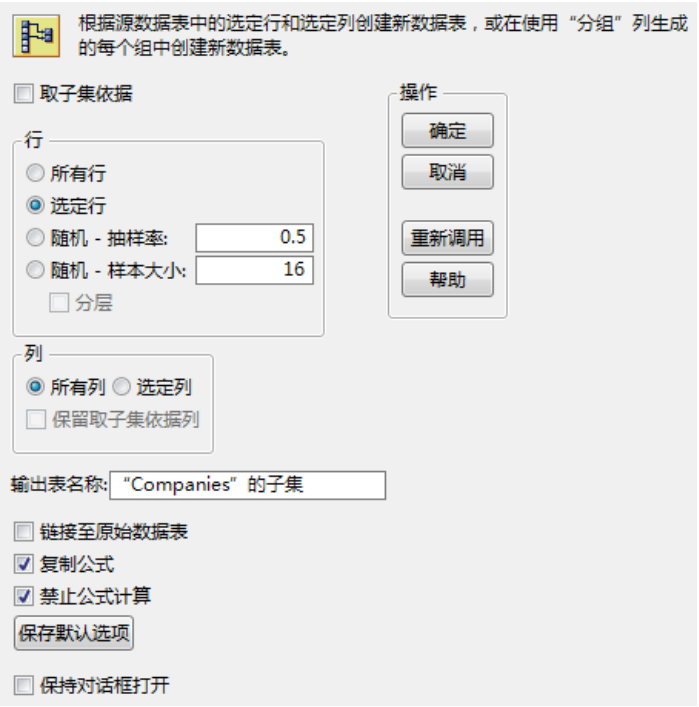
选择您想要生成子集的行和列

2. 选择**行** > **行选择** > **选择符合条件的行**。
3. 在左侧的列表框中选择**公司规模**。
4. 在文本输入框中输入 “medium”。
5. 点击**确定**。
6. 按 Ctrl 键并选择**类型**、**销售额 ($M)** 和**利润 ($M)** 列。

创建子集表

7. 选择**表** > **子集**以启动 “子集” 窗口。

图 3.26 子集窗口



8. 选择**选定列**仅对您所选定的列生成子集。您还可以通过选择其他选项进一步定制您的子集表。
9. 点击**确定**。

得到的子集数据表包含七行和三列。有关“子集”命令的详细信息，请参见《使用 JMP》。

使用“分布”平台生成子集

另一种创建子集的方式是使用平台结果和数据表之间的连接。

示例：使用“分布”命令创建子集

1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 选择**分析 > 分布**。
3. 选择**类型**并点击 **Y，列**。
4. 点击**确定**。
5. 双击代表 “Computer”的直方图，以创建计算机公司的子集表。

警告：该方法创建**链接**的子集表。这意味着，若您对子集表中的数据进行任何更改，源表中的相应值也会更改。

连接数据表的示例

使用“连接”选项可以将多个数据表中的信息合并到一个数据表中。例如，假设您有一个数据表，表中包含爆米花产量实验得到的结果。另一个数据表包含爆米花产量第二次实验的结果。要比较这两个实验或使用两组结果分析实验，则需要将数据放在同一个表中。此外，实验数据不会以相同的顺序输入到数据表中。有一个列具有不同的名称，并且第二个实验不完整。这意味着，您不能将一个表中的数据复制粘贴到另一个表中。

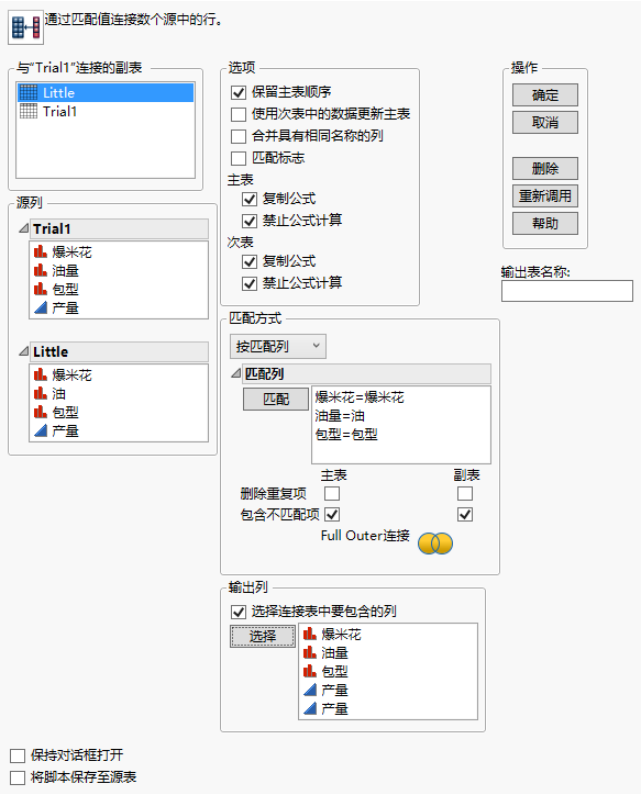
示例：连接两个数据表

1. 选择**帮助** > **样本数据库**，然后打开 Trial1.jmp 和 Little.jmp。
2. 点击 Trial1.jmp 使其成为活动的数据表。
3. 选择**表** > **连接**。
4. 在**与 “Trial1” 连接的副表**框中，选择 Little。
5. 从**匹配方式**菜单中，选择**按匹配列**（若尚未选择该选项）。
6. 在**源列**框中，两个框中都选择**爆米花**，然后点击**匹配**。
7. 同样地，将两个框中的**包型**匹配为**包型**，**油量**匹配为**油**。

   您的匹配列名称不必一定相同。
8. 为两个表选择**包含不匹配项**。

   因为一个实验是不完整的，您想要包含所有行，包括任何带有缺失数据的行。
9. 要避免重复列，选择**选择连接表中要包含的列**选项。
10. 从 Trial1 中，选择所有四列并点击**选择**。
11. 从 Little 中，仅选择**产量**并点击**选择**。

图 3.27 完成的连接窗口



12. 点击**确定**。

图 3.28 连接的表

	爆米花	油量	包型	"Trial1" 中的 "产量"	"Little" 中的 "产量"
1	plain	little	large	8.2	8.8
2	gourmet	little	large	8.6	8.2
3	plain	lots	large	10.4	•
4	gourmet	lots	large	9.2	•
5	plain	little	small	9.9	10.1
6	gourmet	little	small	12.1	15.9
7	plain	lots	small	10.6	•
8	gourmet	lots	small	18.0	•

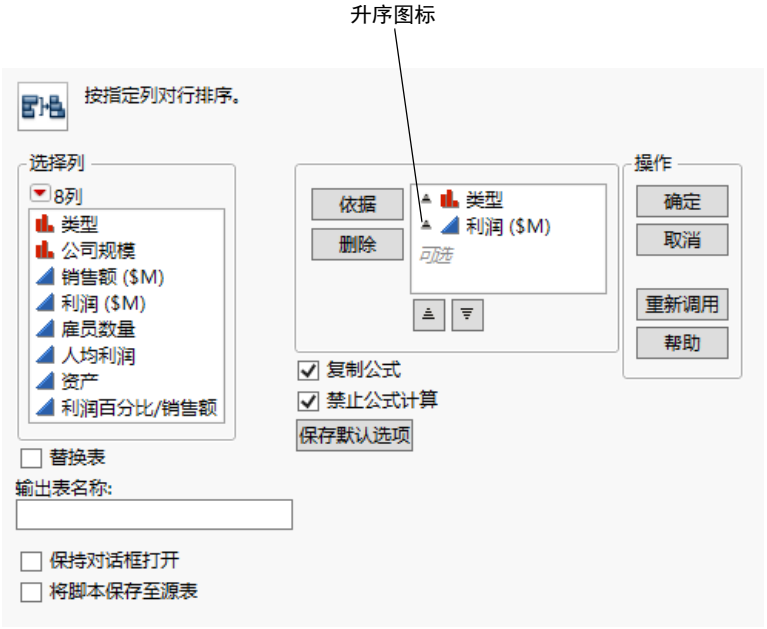
对数据排序

使用 “排序”命令可以按数据表中的一列或多列对数据表排序。例如，查看计算机和制药公司的财务数据。假设您想按**类型**对数据表排序，然后按**利润 ($M)** 排序。此外，您希望**利润 ($M)** 在每个**类型**中为降序。

1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 选择**表 > 排序**。
3. 选择**类型**并点击**依据**，将**类型**指定为排序变量。
4. 选择**利润 ($M)** 并点击**依据**。

此时，两个变量都设置为按升序排序。请参见图 3.29 中变量旁边的升序图标。

图 3.29 升序图标



5. 要将**利润 ($M)** 更改为按降序排序，选择**利润 ($M)** 并点击降序按钮。

图 3.30 更改利润为降序

“利润 ($M)”旁边的图标更改为降序。

6. 选中**替换表**复选框。

   选中后，**替换表**选项告知 JMP 对原始数据表排序而不是创建排好序的新表。若基于原始数据表创建的任何报表窗口处于打开状态，该选项将不可用。在报表窗口打开的情况下对数据表排序，可能会更改某些数据在报表窗口中的显示方式，特别是在图形中。

7. 点击**确定**。

数据表现在按类型的字母顺序排序，在类型中利润合计按降序排序。

第 4 章

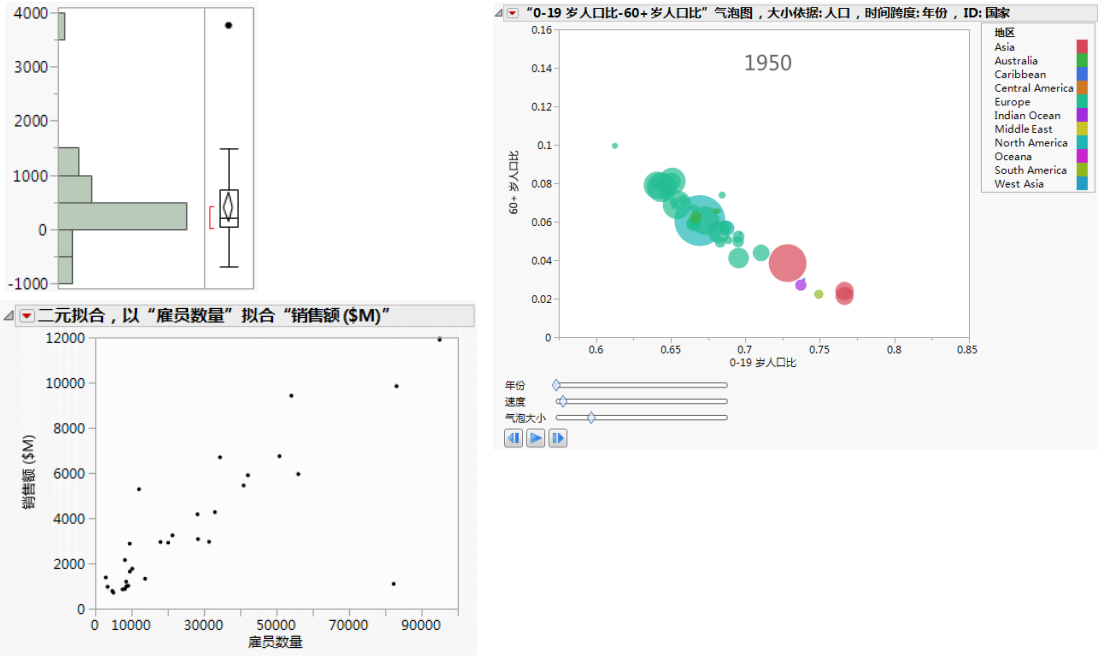
数据可视化

常见图形

将数据可视化是重要的第一步。本章描述的图形可帮助您发现数据的重要细节。例如，直方图可显示数据的形状和范围，帮助您查找异常的数据点。

本章介绍几个最常见的图形，可帮助您在 JMP 中直观地探索数据。本章还将介绍 JMP 的一些图形工具和平台。使用 JMP 能够直观地看到单个变量的分布或多个变量之间的关系。

图 4.1 使用 JMP 可视化数据



目录

在一元图形中分析单个变量 89
将直方图用于连续变量 89
将条形图用于分类变量 92
比较多个变量 94
使用散点图比较多个变量 95
使用散点图矩阵比较多个变量 98
使用并排箱线图比较多个变量 101
使用图形生成器比较多个变量 104
使用气泡图比较多个变量 110
使用叠加图比较多个变量 115
使用变异性图比较多个变量 119

在一元图形中分析单个变量

通过单变量图形（即**一元**图形），您可以一次深度关注一个变量。在开始查看数据时，重要的是先了解每个变量，然后再查看变量之间如何相互影响。通过一元图形可以单独可视化每个变量。

本节包括两个图形来说明单个变量的分布：

- 第 89 页的“将直方图用于连续变量”
- 第 92 页的“将条形图用于分类变量”

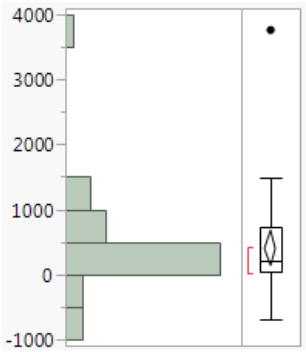
使用“分布”平台可以创建这两个图形。“分布”生成每个变量的图形说明和描述性统计量。

将直方图用于连续变量

直方图是了解连续变量分布的最有用的图形工具之一。使用直方图可以找到数据中的下列信息：

- 平均值和变异性
- 极值

图 4.2 直方图示例



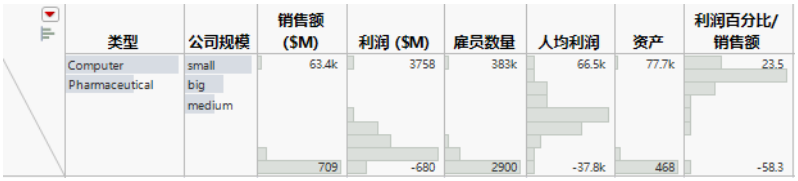
即时直方图

您可以通过点击列标题中的直方图图标即时查看直方图。直方图显示在列标题下方。

图 4.3 即时直方图

	类型	公司规模	销售额 ($M)	利润 ($M)	雇员数量	人均利润	资产	利润百分比/销售额
1	Computer	small	855.1	31.0	7523	4120.70	615.2	3.63
2	Pharmaceutical	big	5453.5	859.8	40929	21007.11	4851.6	15.77
3	Computer	small	2153.7	153.0	8200	18658.54	2233.7	7.10

点击直方图图标。



情境

本示例使用 Companies.jmp 数据表，表中包含一组公司的利润数据。

财务分析人员想探究下列问题：

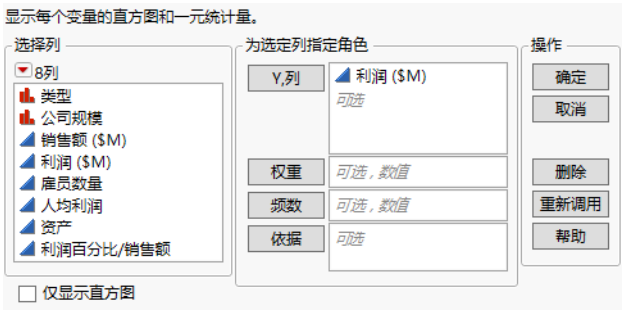
- 通常情况下，每家公司可以获得多大的利润？
- 平均利润是多少？
- 有没有公司的利润明显高于或低于其他公司？

要回答这些问题，使用**利润 ($M)** 的直方图。

创建直方图

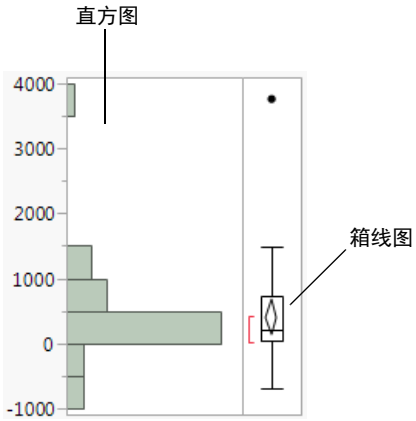
1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**利润 ($M)** 并点击 **Y，列**。

图 4.4 利润 ($M) 的分布窗口



4. 点击**确定**。

图 4.5 利润 ($M) 的直方图



解释直方图

直方图给出下列回答：

- 大多数公司的利润在 $-1000 到 $1500 之间。

  除一个直条以外的所有直条都位于该范围内。此外，相比于其他范围，更多公司的利润在 $0 到 $500 之间。代表该范围的直条比其他直条更长。

- 平均利润略小于 $500。

  箱线图中的菱形中部表示均值。在该示例中，均值略低于 $500 标记处。

- 一家公司的利润明显高于其他公司，可能是一个**离群值**。离群值是从其他常规模式的数据点分离出来的数据点。

  该离群值由直方图顶部的一个非常短的直条表示。直条很小，表示组很小（在该示例中为一家公司），并且它与其他直方图条分开得很远。

除直方图以外，该报表还包括：

- 箱线图（数据的另一种图形汇总）。有关箱线图的详细信息，请参见《基本绘图》。
- **分位数**和**汇总统计量**报表。这些报表将在“分析数据”一章中第 131 页的“分析分布”中讨论。

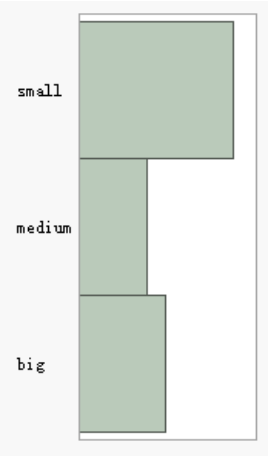
与直方图交互

在 JMP 中，数据表和报表都互相关联。点击直方图条可以选择数据表中相对应的行。

将条形图用于分类变量

使用条形图可以将分类变量的分布可视化。条形图的外观与直方图类似，因为这两种图形中都包含对应于变量水平的直条。条形图显示的直条对应于变量的每个水平，而直方图显示变量的值范围。

图 4.6 条形图示例



情境

本示例使用 Companies.jmp 数据表，表中包含一组公司的规模和类型数据。

财务分析人员想探究下列问题：

- 最常见的公司类型是什么？
- 最常见的公司规模是什么？

要回答这些问题，使用**类型**和**公司规模**的条形图。

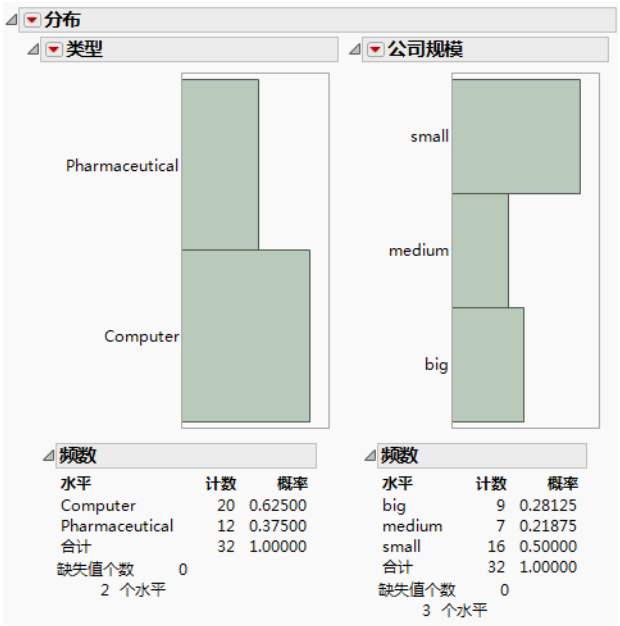
创建条形图

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**类型**和**公司规模**，并点击 **Y，列**。
4. 点击**确定**。

图 4.7 类型和公司规模的条形图

条形图

汇总信息



解释条形图

条形图给出下列回答：

- 计算机公司比制药公司多。

  代表计算机公司的直条比代表制药公司的直条大。
- 小公司最常见。

  代表小公司的直条比代表中等和大规模公司的直条大。

附加的汇总输出提供了详细的频数。该报表将在“分析数据”一章中第 134 页的“分类变量的分布”中讨论。

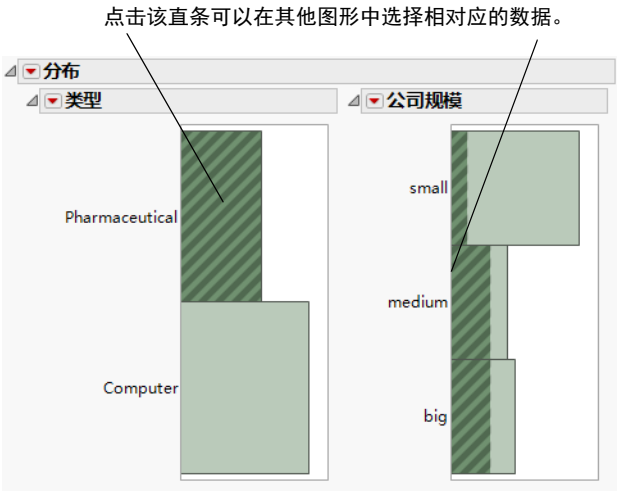
与条形图交互

与直方图一样，点击每个直条会突出显示数据表的行。若创建了多个图形，点击一个条形图中的某个直条会突出显示其他条形图中相对应的一个或多个直条。

例如，假设您想查看制药公司的公司规模的分布。点击**类型**条形图中的“Pharmaceutical”直条，制药公司将在**公司规模**条形图中突出显示。图 4.8 显示出尽管该数据表中的大多数公司是小规模公司，但大多数制药公司是中等规模或大规模公司。

此外，数据表中相对应的行会被选定。

图 4.8 点击直条



比较多个变量

使用多变量图形可以将两个或多个变量之间的关系和模型可视化。本节包括下列图形：

表 4.1 多变量图形

第 95 页的“使用散点图比较多个变量”	使用散点图可以比较两个连续变量。
第 98 页的“使用散点图矩阵比较多个变量”	使用散点图矩阵可以比较多对连续变量。
第 101 页的“使用并排箱线图比较多个变量”	使用并排箱线图可以比较一个连续变量和一个分类变量。
第 119 页的“使用变异性图比较多个变量”	使用变异性图可以在一个或多个分类 X 变量之间比较一个连续 Y 变量。变异性图显示多个分类 X 变量之间均值和变异性的差异。
第 104 页的“使用图形生成器比较多个变量”	使用“图形生成器”可以通过交互方式创建和更改图形。
第 115 页的“使用叠加图比较多个变量”	使用叠加图可以将 Y 轴上的一个或多个变量与 X 轴上的另一个变量比较。若 X 变量是时间变量，则叠加图会非常有用，因为您可以比较两个或多个变量随时间如何变化。

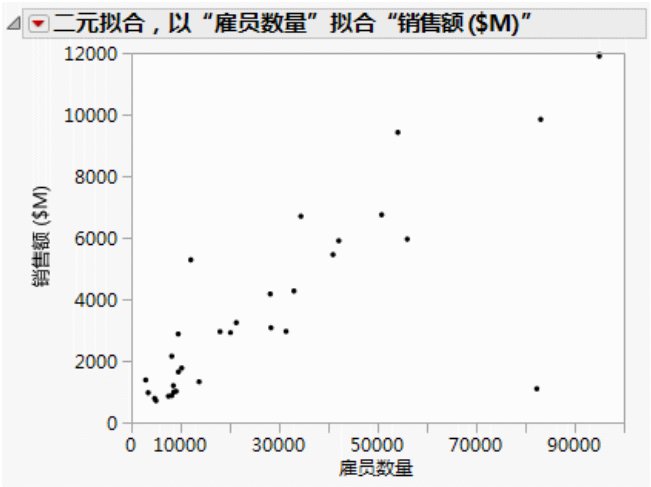
表 4.1 多变量图形 (续)

第 110 页的“使用气泡图比较多个变量”	气泡图是特殊的散点图，使用颜色和气泡大小来同时表示多达五个变量。若某个变量是时间变量，则可以制作图形动画来查看其他变量随时间的变化。

使用散点图比较多个变量

散点图是所有多变量图形中最简单的一个。使用散点图可以确定两个连续变量之间的关系，并且可以发现两个连续变量之间是否**相关**。相关性表示两个变量之间的相关程度。若两个变量高度相关，则一个变量可能会影响另一个变量。或者，两个变量被其他变量以相似方式影响。

图 4.9 散点图示例



情境

本示例使用 Companies.jmp 数据表，表中包含一组公司的销售数字和雇员数量。

财务分析人员想探究下列问题：

- 销售额和雇员数量之间是什么关系？
- 销售额会随雇员数量增长吗？
- 能从雇员数量预测出平均销售额吗？

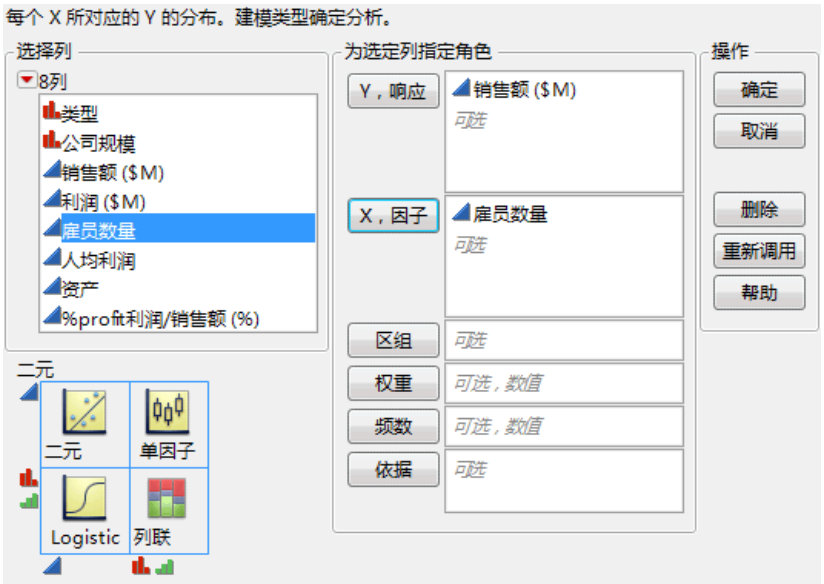
要回答这些问题，使用**雇员数量**-**销售额 ($M)** 散点图。

创建散点图

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **以 X 拟合 Y**。

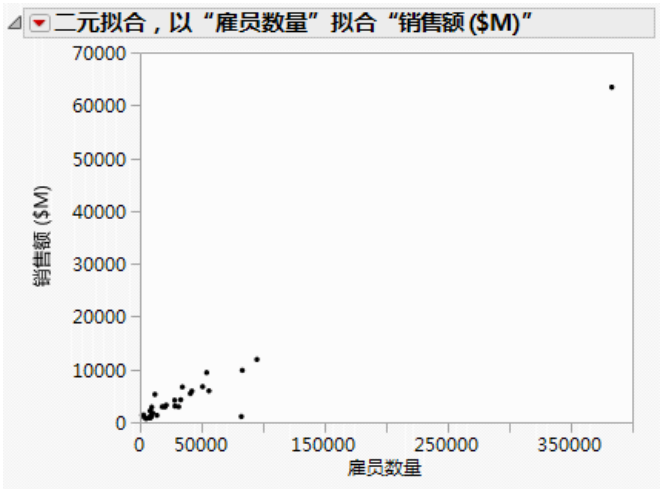
3. 选择**销售额 ($M)** 和 **Y，响应**。
4. 选择**雇员数量**和 **X，因子**。

图 4.10 以 X 拟合 Y 窗口



5. 点击**确定**。

图 4.11 雇员数量 - 销售额 ($M) 散点图



解释散点图

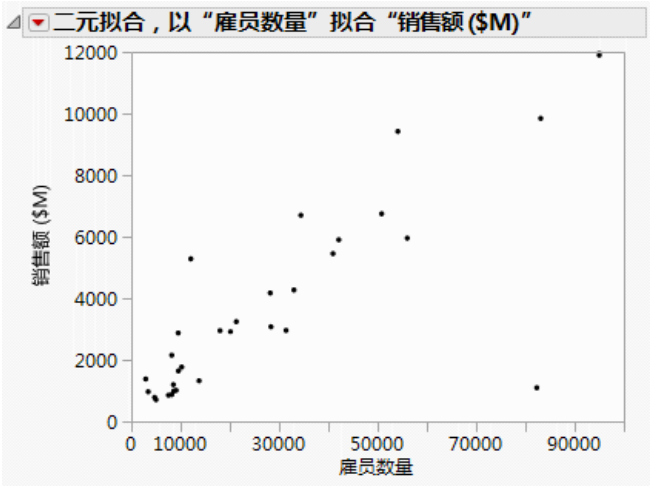
图形右上方的一个点表示有一家公司的雇员数量和销售额都很高。该数据点和其他数据点之间的距离使其他公司之间的关系很难实现可视化。按照下面的步骤，从图形中删除该点并重新创建图形：

1. 点击该点来选上它。
2. 选择**行 > 隐藏和排除**。数据点隐藏并且不再包含在计算中。

   注意：区分隐藏和排除很重要。隐藏某个点会从图形中删除该点但统计计算会继续使用它。排除某个点会从统计计算中删除该点但不会从图形中删除它。若同时隐藏和排除某个点，将从所有计算和所有图形中删除该点。

3. 要重新创建不包含离群值的图形，点击“二元”红色小三角并选择**重新运行 > 重新运行分析**。可以关闭原始报表窗口。

图 4.12 已删除离群值的散点图



更新的散点图给出下列回答：

- 销售额和雇员数量之间存在关系。

  数据点有辨别得出的模式。在图形中数据点不是随机分散，可以靠近大部分数据点绘制一条对角线。

- 销售额随雇员数量增长，并且关系为线性。

  若绘制对角线，它将从左下方向右上方倾斜。这样倾斜表示随着雇员数量增加（底部轴从左到右），销售额也会增长（左侧轴从下到上）。直线靠近大部分数据点，这表示是线性关系。若必须弯曲线条才能靠近数据点，则仍然存在关系（因为数据点的模式），但关系将不是线性的。

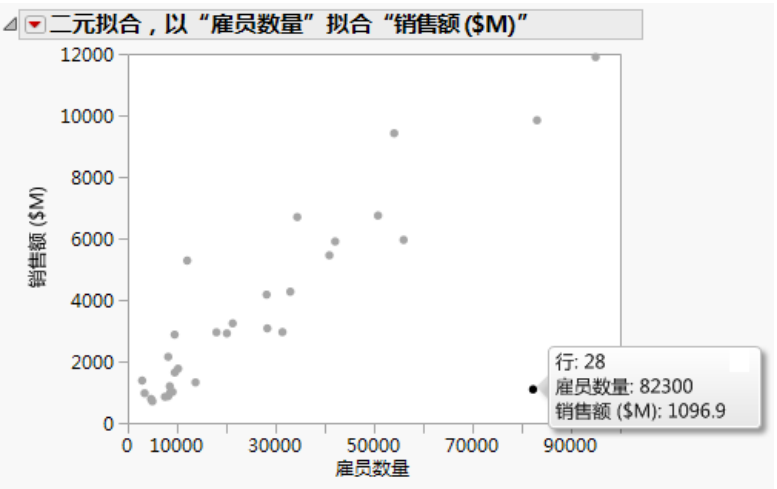
- 可以从雇员数量预测出平均销售额。

散点图显示销售额通常会随雇员数量的增加而增长。若您知道某家公司的雇员数量，则可以预测出该公司的销售额。您的预测落在一条假想线。虽然不会很准确，但会接近实际的销售额。

与散点图交互

与其他 JMP 图形相同，散点图是交互式的。将鼠标悬停在右下角的点上会显示行号以及 x 和 y 值。

图 4.13 将鼠标悬停在某个点上



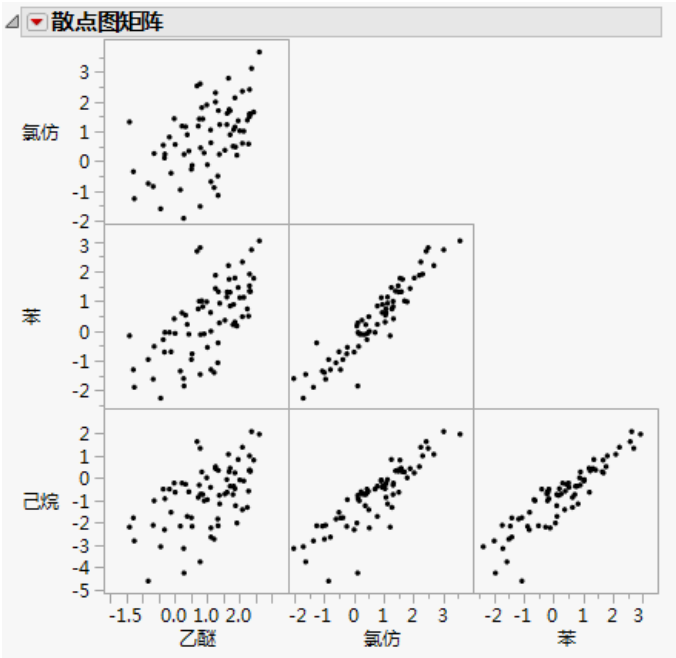
点击某点会在数据表中突出显示相应的行。通过执行下列操作可以选择多个点：

- 点击并将光标拖过这些点。这样操作会选择矩形区域内的点。
- 选择套索工具，然后点击并拖过多个点。套索工具会选择一个形状不规则的区域。

使用散点图矩阵比较多个变量

散点图矩阵是组织成网格（矩阵）形式的散点图集合。每个散点图显示一对变量之间的关系。

图 4.14 散点图矩阵的示例



情境

本示例使用 Solubility.jmp 数据表，表中包含 72 种不同溶质的溶解度测量。

实验室技术人员想探究下列问题：

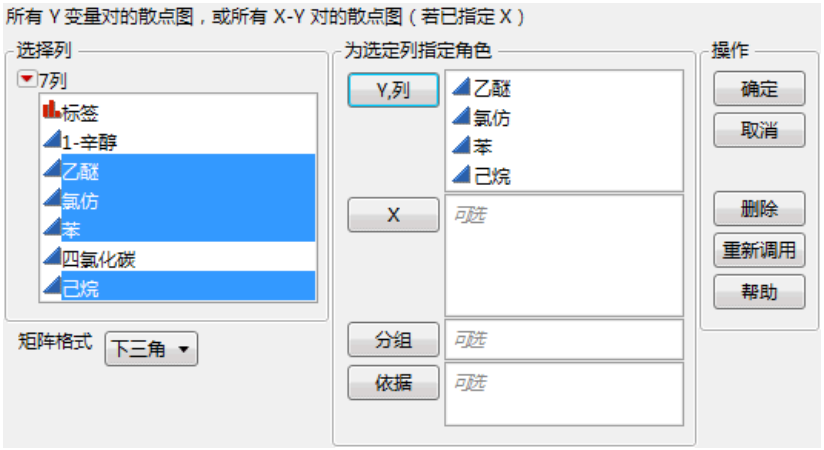
- 任何一对化学物质之间是否存在关系？（有六种配对可能。）
- 哪种配对的关系最强？

要回答这些问题，使用四种溶剂的散点图矩阵。

创建散点图矩阵

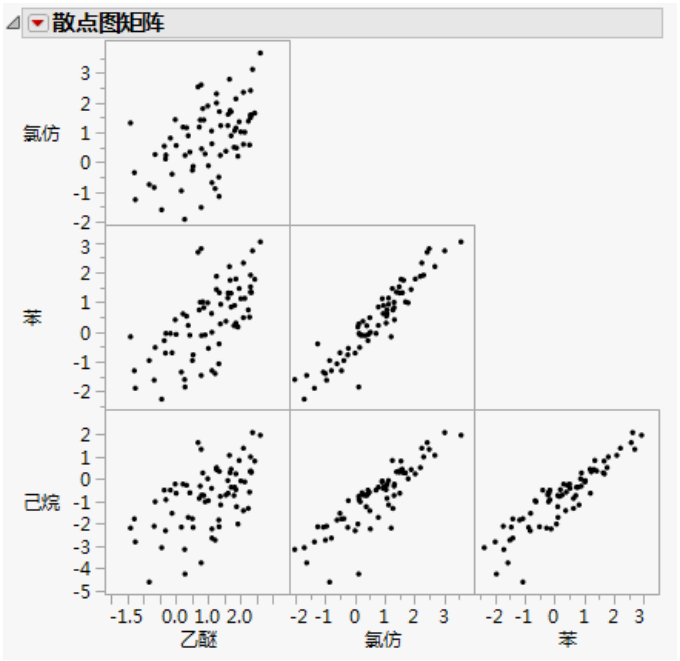
1. 选择**帮助** > **样本数据库**，然后打开 Solubility.jmp。
2. 选择**图形** > **散点图矩阵**。
3. 选择**乙醚**、**氯仿**、**苯**和**己烷**，并点击 **Y，列**。

图 4.15 散点图矩阵窗口



4. 点击**确定**。

图 4.16 散点图矩阵



解释散点图矩阵

散点图矩阵给出下列回答：

- 所有六对变量都是正相关。

随着一个变量增大，其他变量也增大。

- **苯**和**氯仿**之间的关系显示为最强。

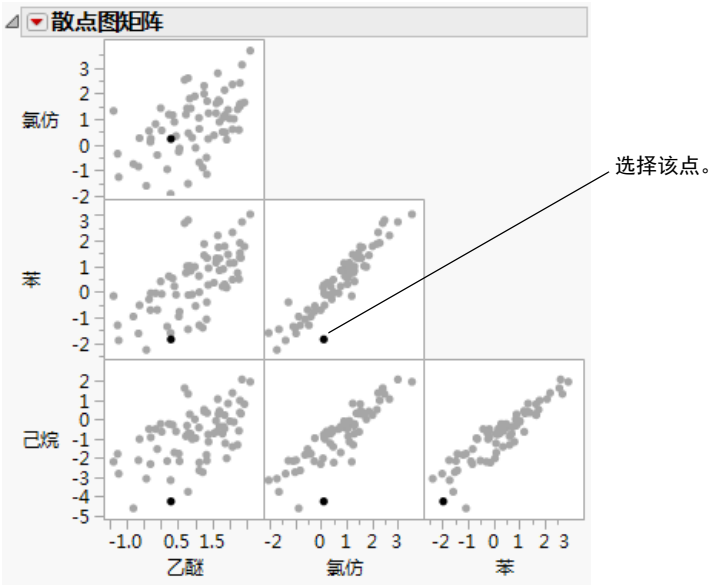
  氯仿 - **苯**散点图中的数据点沿一条假想线极其紧密地聚集。

与散点图矩阵交互

若您在一个散点图中选择某个点，在其他散点图中该点也会被选择。

例如，若在**氯仿**-**苯**散点图中选择某个点，在其他五个图形中同一个点也会被选择。

图 4.17 选定的点



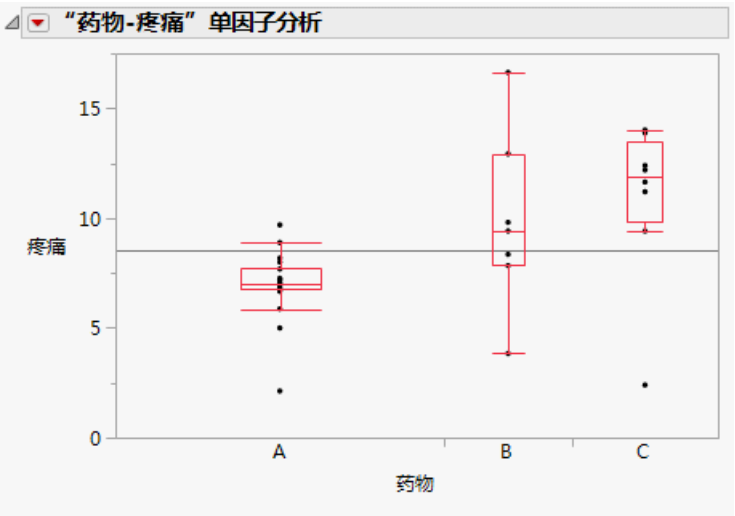
请注意，在其他散点图中也会选定同一个点。

使用并排箱线图比较多个变量

并排箱线图显示下列信息：

- 一个连续变量和一个分类变量之间的关系
- 连续变量在分类变量不同水平下的差异

图 4.18 并排箱线图示例



情境

本示例使用 Analgesics.jmp 数据表，表中包含分别使用三种不同药物的病人的疼痛测量数据。研究人员想探究下列问题：

- 各药物之间的平均止疼疗效有差异吗？
- 每种药物的止疼疗效的**变异性**有差别吗？变异性高的药物不如变异性低的药物可靠。

要回答这些问题，可使用疼痛水平和药物类别的并排箱线图。

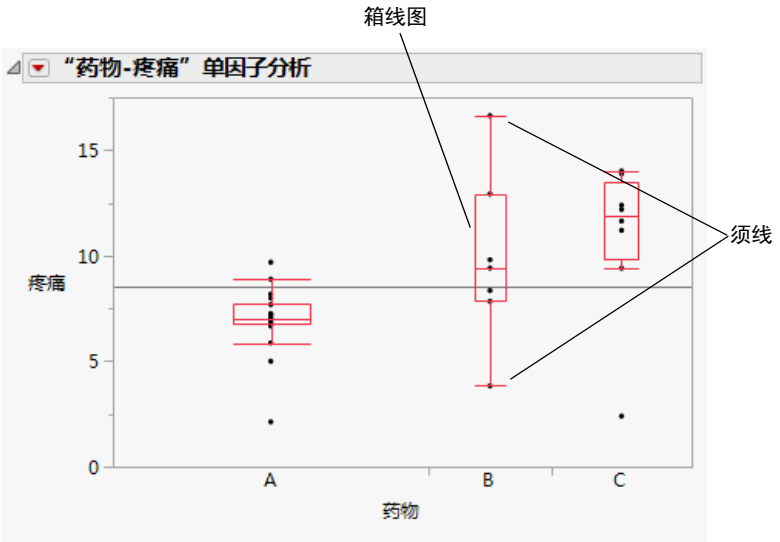
创建并排箱线图

1. 选择**帮助** > **样本数据库**，然后打开 Analgesics.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**疼痛**并点击 **Y，响应**。
4. 选择**药物**并点击 **X，因子**。

图 4.19 以 X 拟合 Y 窗口

5. 点击**确定**。
6. 点击“‘药物 - 疼痛’单因子分析”旁边的红色小三角并选择**显示选项 > 箱线图**。

图 4.20 并排箱线图



解释并排箱线图

箱线图根据下列原则进行设计：

- 贯穿盒体的直线表示中位数。
- 数据的中值落在盒体内。

- 大部分数据落在须线两端之间。
- 落在须线外部的数据点可能是离群值。

图 4.20 中的箱线图给出下列回答：

- 有证据证实服用药物 A 的病人感受到的疼痛更少，因为相比其他药物，药物 A 的箱线图中的疼痛范围较小。
- 药物 B 看起来比药物 A 和药物 C 有更大的变异性，因为箱线图较高。

药物 C 的一个点明显低于药物 C 的其他点。将鼠标悬停在低点上可以看到它是数据表中的第 26 行。该点看起来更像药物组 A 或药物组 B 中的数据。行 26 中的数据需要研究。数据记录时可能发生了录入错误。

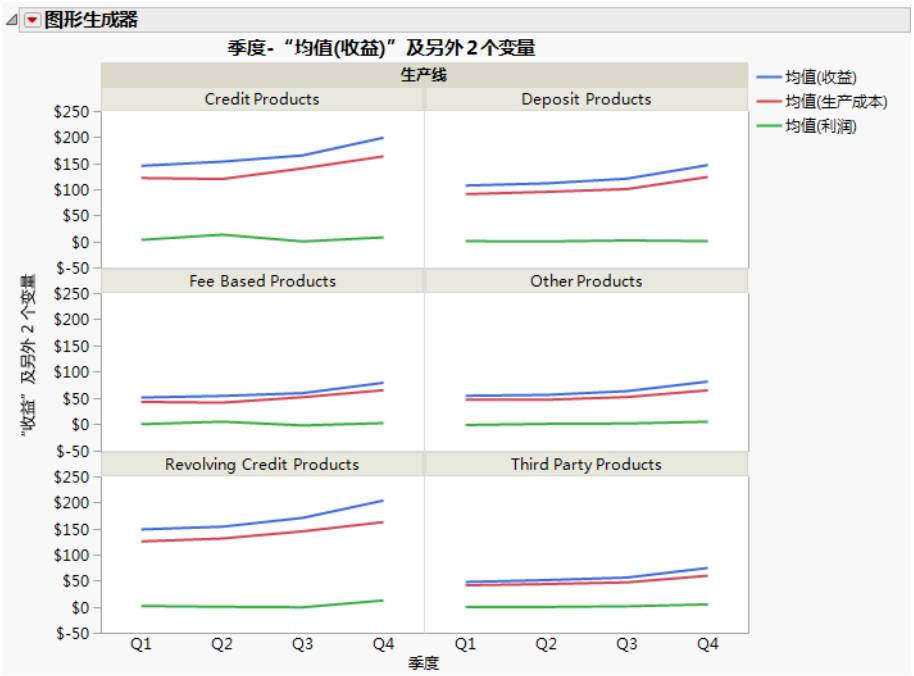
使用图形生成器比较多个变量

使用“图形生成器”可以通过交互方式创建和修改图形。JMP 中的大多数图形通过启动平台并指定变量的方式创建。若您想创建不同类型的图形，可以从“图形”菜单启动特定平台。但是，使用“图形生成器”您可以随时更改变量和图形类型。

使用 “图形生成器”可以完成下列任务：

- 通过将变量拖入和拖出图形来更改变量。
- 通过几下鼠标点击操作创建不同类型的图形。
- 水平或垂直分割图形。

图 4.21 通过 “图形生成器”创建的图形示例



注意：这里仅介绍“图形生成器”的部分功能。请参见《基本绘图》。

情境

本示例使用 Profit by Product.jmp 数据表，表中包含多个产品线的利润数据。

业务分析人员想探究下列问题：

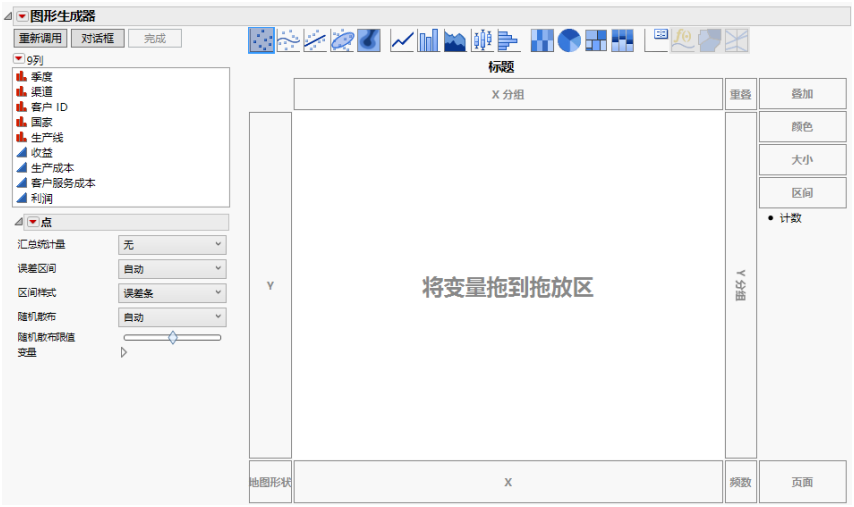
- 不同产品线之间的收益有什么差别？

要回答这个问题，请使用线图显示不同产品线之间的收入、产品成本和利润数据。

创建图形

1. 选择**帮助** > **样本数据库**，然后打开 Profit by Product.jmp。
2. 选择**图形** > **图形生成器**。

图 4.22 图形生成器工作区

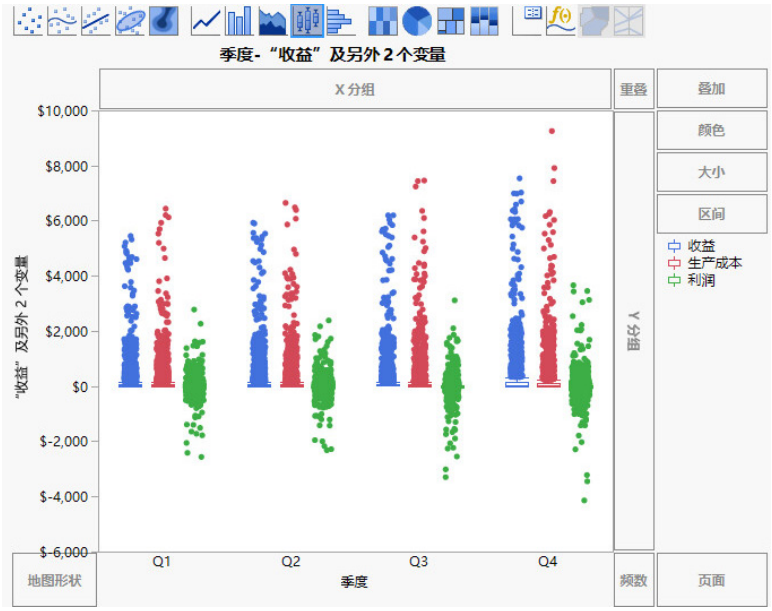


3. 点击**季度**然后将其拖放到 X 区域，从而将**季度**指定为 X 变量。
4. 点击**收益**、**产品成本**和**利润**，并将它们拖放到 Y 区域，从而将这三个变量指定为 Y 变量。

   X 和 Y 区域现在是坐标轴。

> **注意：**您也可以点击这些变量然后点击某个区域以分配它们。但是，在区域变为坐标轴之后，需要将其他变量拖放到坐标轴而非点击变量和坐标轴。

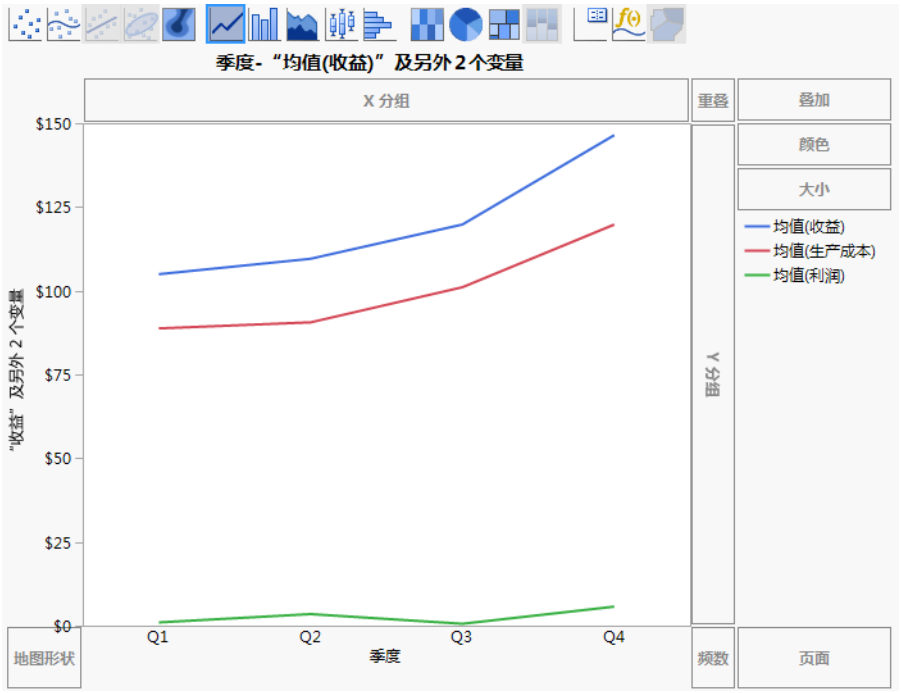
图 4.23 添加 Y 变量和 X 变量之后



根据您使用的变量，“图形生成器”显示并排箱线图。

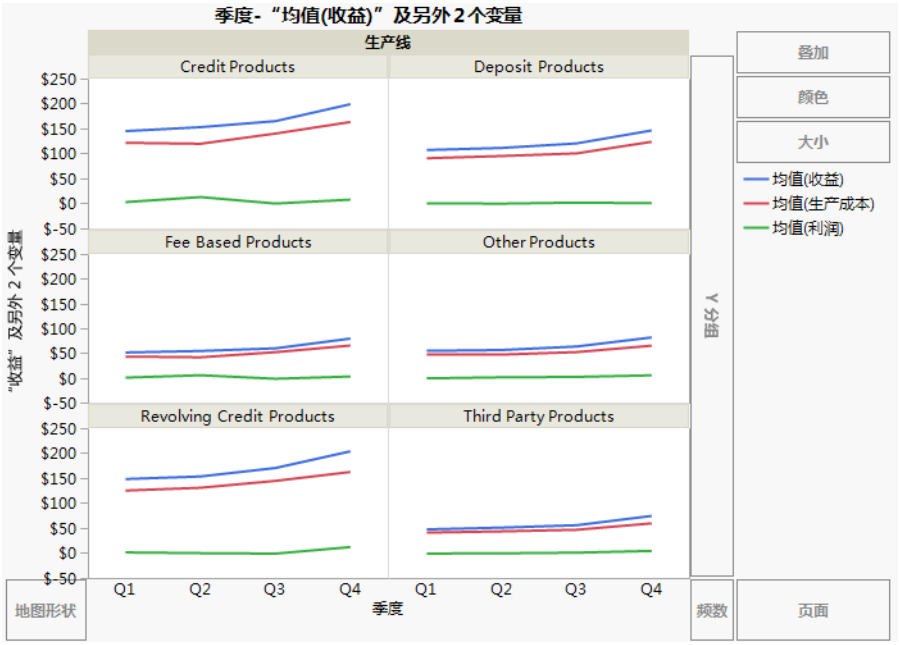
5. 要将箱线图更改为线图，点击“线图”图标 。

图 4.24 线图



6. 要为每个产品创建单独的图形，点击**产品线**，将其拖放到**重叠**区域。
为每个产品创建单独的线图。

图 4.25 最终的线图



解释图形

图 4.25 显示每个产品线的收入、成本和利润。业务分析人员想要查看不同产品线之间的收益差别。图 4.25 中的线图给出下列回答：

- 相比于以费用为基础的产品、第三方产品和其他产品，贷款产品、存款产品和周转贷款产品的收入更高。
- 但是，所有产品线的利润都非常接近。

数据表还包括销售渠道数据。业务分析人员想了解不同销售渠道之间收入、产品成本和利润有什么差别。

1. 要从图形中删除**产品线**，点击图形标题（**产品线**）并将其拖放到“图形生成器”中的空白区域。
2. 要将**渠道**添加为重叠变量，点击**渠道**并将其拖放到**重叠**区域中。

图 4.26 显示销售渠道的线图

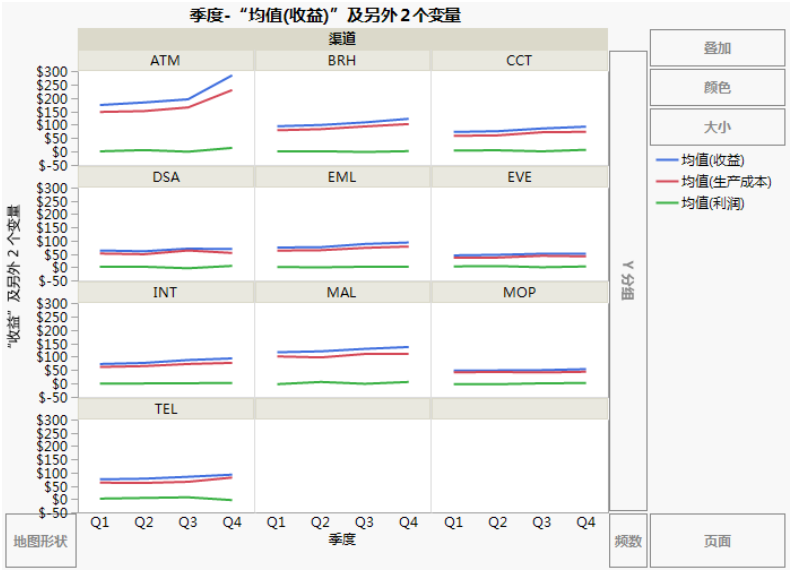
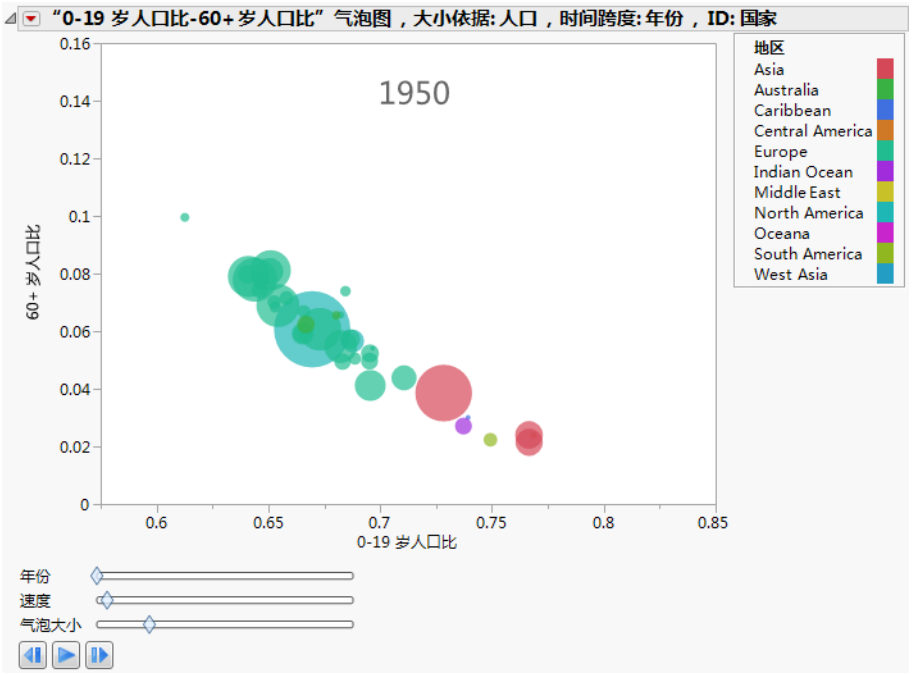


图 4.26 给出这样的回答：ATM 的收入和产品成本最高，并且增长最快。

使用气泡图比较多个变量

气泡图是一个将点表示为气泡的散点图。可以更改气泡的大小和颜色，甚至可以按时间变化将气泡制成动画。由于最多可以表示五维（x 位置、y 位置、大小、颜色和时间），气泡图可以产生动画视觉效果，使数据探索更加方便。

图 4.27 气泡图示例



情境

本示例使用 PopAgeGroup.jmp 数据表，表中包含 116 个国家或地区从 1950 年到 2004 年的人口统计数据。总人口数按年龄组划分，不是每个国家每年都有数据。

社会学家想探究下列问题：

- 世界上人口的年龄在变化吗？

要回答这个问题，查看最老（59 岁以上）和最年轻（不到 20 岁）年龄段人口之间的关系。使用气泡图可以确定该关系如何随时间变化。

创建气泡图

1. 选择**帮助** > **样本数据库**，然后打开 PopAgeGroup.jmp。
2. 选择**图形** > **气泡图**。
3. 选择 **60+ 岁人口比**并点击 **Y**。

   这对应于气泡图中的 Y 变量。
4. 选择 **0-19 岁人口比**并点击 **X**。

   这对应于气泡图中的 X 变量。
5. 选择**国家**并点击 **ID**。

ID 变量的每个唯一水平由图形中的气泡表示。

6. 选择**年份**并点击**时间**。

   这控制制作气泡图动画时的时间指标。

7. 选择**人口**并点击**大小**。

   这控制气泡的大小。

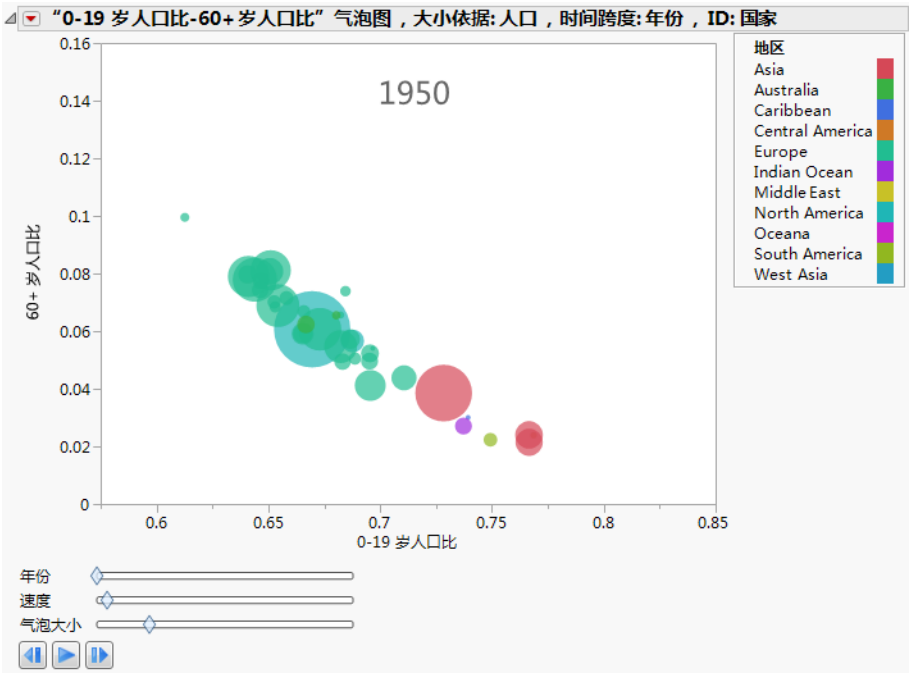
8. 选择**地区**并点击**颜色**。

   “颜色”变量的每个水平都分配了一个独特的颜色。在本示例中，表示位于同一地区的不同国家的所有气泡都是相同的颜色。图 4.29 中显示的气泡图颜色是 JMP 默认颜色。

图 4.28 “气泡图”启动窗口

9. 点击**确定**。

图 4.29 初始气泡图



解释气泡图

因为时间变量 （本示例中为 “年份”）从 1950 开始，初始气泡图显示 1950 年的数据。点击播放 / 暂停按钮可以使气泡图动画地显示所有年份的循环过程。每个连续的气泡图显示对应年份的数据。每年的数据确定下列内容：

- X 和 Y 坐标
- 气泡大小
- 气泡颜色
- 气泡聚合

注意：有关气泡图如何跨多个行聚合信息的详细信息，请参见《基本绘图》。

1950 年的气泡图显示，若不到 20 岁的年轻人比例高，则 59 岁以上的人口比例低。

点击播放 / 暂停按钮播放随年份的推移而变化的气泡图动画。随着时间推进， 0-19 岁人口比人口减少， 60+ 岁人口比人口增加。

播放动画，点击后变为暂停按钮。

暂停动画。

手动控制动画向后一个时间单位。

手动控制动画向前一个时间单位。

年份 手动更改时间指标。

速度 控制动画的速度。

气泡大小 控制气泡的绝对大小，同时保持相对大小。

社会学家想了解世界人口年龄是如何变化的。气泡图表示世界人口在逐渐变老。

与气泡图交互

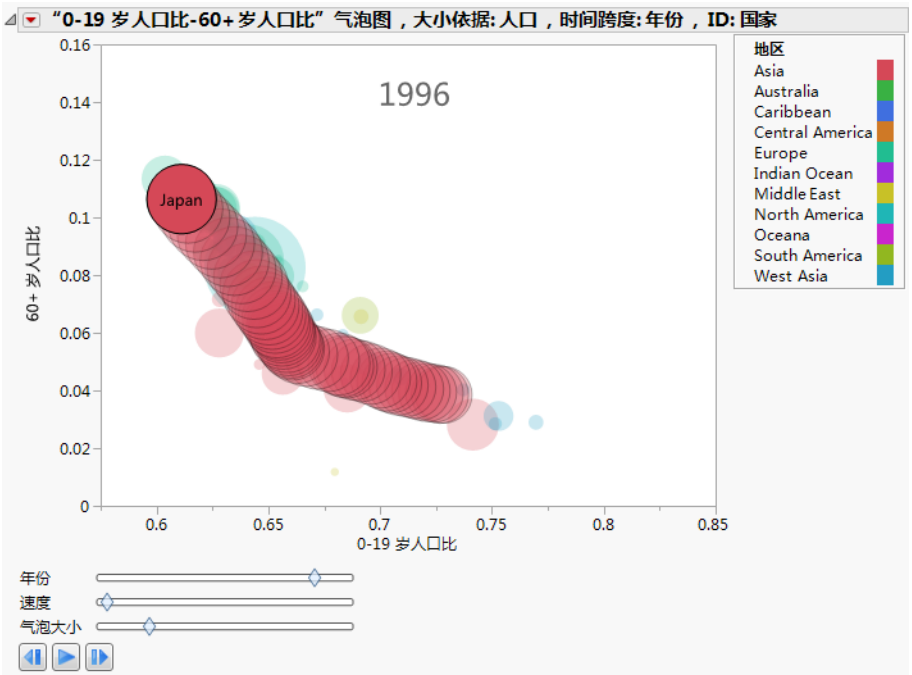
点击可选择某个气泡以查看它随时间的变化趋势。例如，在 1950 年的图形中，中间的大气泡是“Japan”。

查看日本各年份人口变化的模式

1. 点击“日本”气泡的中间将其选定。
2. 点击“气泡图”红色小三角并选择**尾迹气泡 > 已选定**。
3. 点击播放按钮。

随着动画的播放，“Japan”气泡留下了气泡尾迹来演示气泡的历史轨迹。

图 4.30 日本历史人口变化



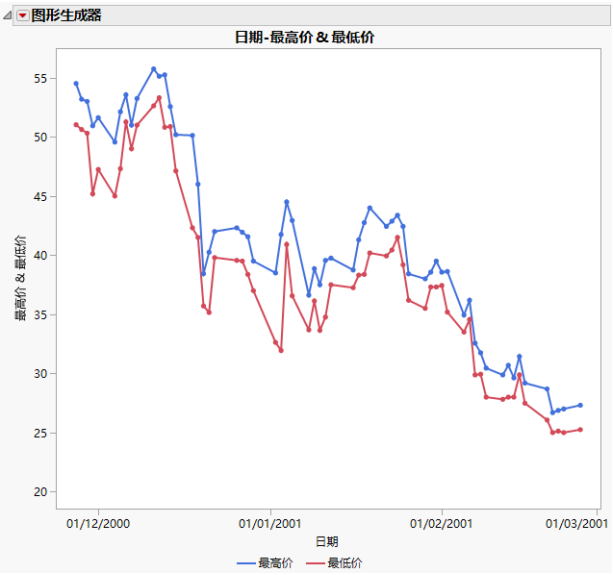
观察“Japan”气泡，您会发现随着时间推进：

- 0-19 岁人口比例降低。
- 60 岁以上人口比例增高。

使用叠加图比较多个变量

与散点图类似，叠加图显示两个或多个变量之间的关系。但是，若某个变量是时间变量，叠加图显示随时间的变化趋势要比散点图的效果好。

图 4.31 叠加图示例



注意：要绘制随时间变化的数据，也可以使用气泡图、控制图和变异性图。有关“图形生成器”和气泡图的详细信息，请参见《基本绘图》。请参见《质量和过程方法》和获取有关控制图和变异性图的信息。

情境

本示例使用 Stock Prices.jmp 数据表，表中包含三个月内的股票价格数据。

潜在投资者想探究下列问题：

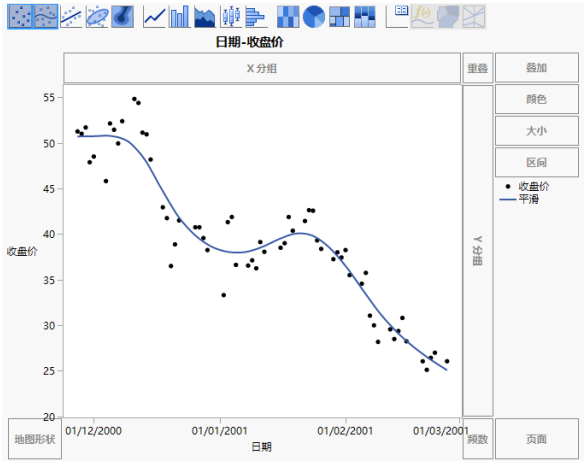
- 在过去的三个月内股票的收盘价格是否发生了变化？
  要回答这个问题，请使用一段时间内股票收盘价格的叠加图。
- 股票的最高价和最低价之间有什么联系？
  要回答这个问题，请使用一段时间内股票最高价和最低价的叠加图。

创建第一个叠加图来回答第一个问题，然后创建另一个叠加图来回答第二个问题。

创建一段时间内股票价格的叠加图

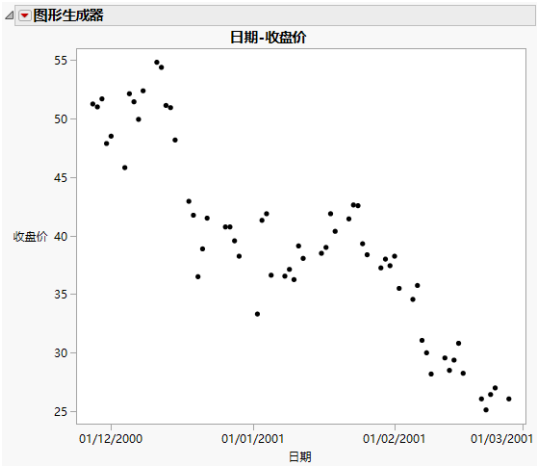
1. 选择**帮助** > **样本数据库**，然后打开 Stock Prices.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**收盘价**并点击 Y。
4. 选择**日期**并点击 X。

图 4.32 带平滑线的叠加图



5. 按 Ctrl 键并点击图形上方的 “平滑线” 图标 以删除平滑线。

图 4.33 一段时间内收盘价的叠加图

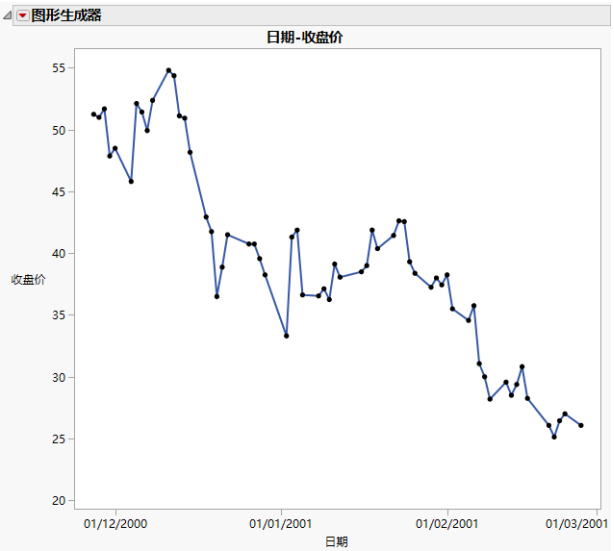


解释叠加图并与其交互

叠加图显示在过去的几个月内股票的收盘价逐渐走低。要清晰地查看趋势，可以连接点。

1. 按 Shift 键并点击图形上方的“线”图标 。

图 4.34 连接的点



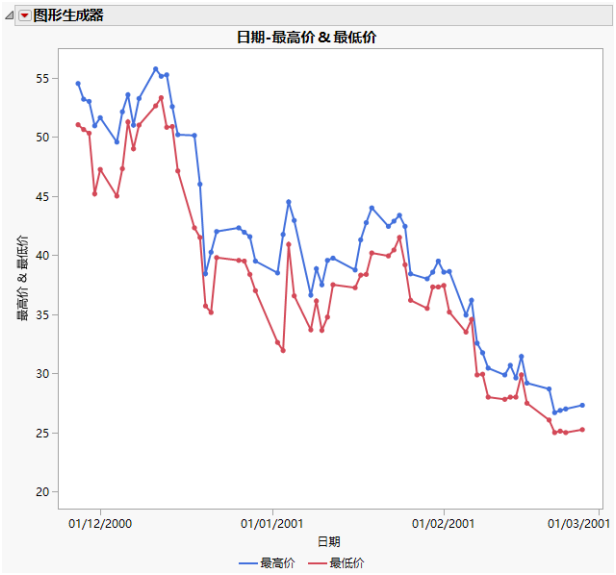
潜在投资者可能注意到尽管股票价格在过去的三个月内上上下下，但总体趋势是向下的。

创建显示股票最高价和最低价的叠加图

使用叠加图可以对多个 Y 变量绘制图形。例如，假设您想在同一个图形中看到最高价和最低价。

1. 按照第 116 页的“创建一段时间内股票价格的叠加图”中的步骤操作，这一次将**最高价**和**最低价**分配给 Y 角色。
2. 按照第 117 页的“解释叠加图并与其交互”中的步骤连接点并添加网格线。

图 4.35 两个 Y 变量



叠加图底部的图例显示了在图形中**最高价**和**最低价**变量所使用的颜色和标记。叠加图显示**最高价**和**最低价**的轨迹非常接近。

回答问题

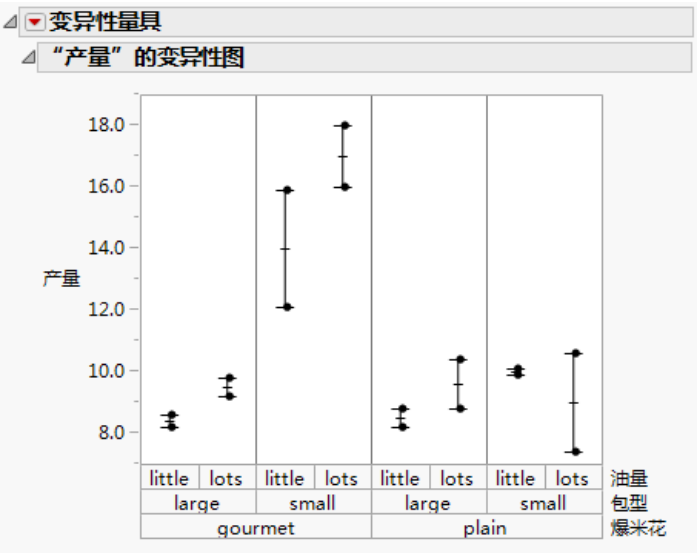
两个叠加图均回答了本示例前面提出的两个问题。

- 第一个叠加图显示股票价格不是保持不变，而是在降低。
- 第二个叠加图显示股票的最高价和最低价相互差别不大。股票价格在给定日期内没有剧烈变化。

使用变异性图比较多个变量

在目前为止介绍的图形中，您仅指定一个 X 变量。使用变异性图可以指定多个 X 变量，并同时查看多个变量之间均值和变异性上的差异。

图 4.36 变异性图的示例



情境

本示例使用 Popcorn.jmp 数据表，数据由爆米花制造商提供。产量（指定数据的玉米粒产出的爆米花数量）根据爆米花类型、包型大小和所用油量的每个组合进行测量。

爆米花制造商想探究下列问题：

- 哪些因素的组合会导致爆米花产量最高？

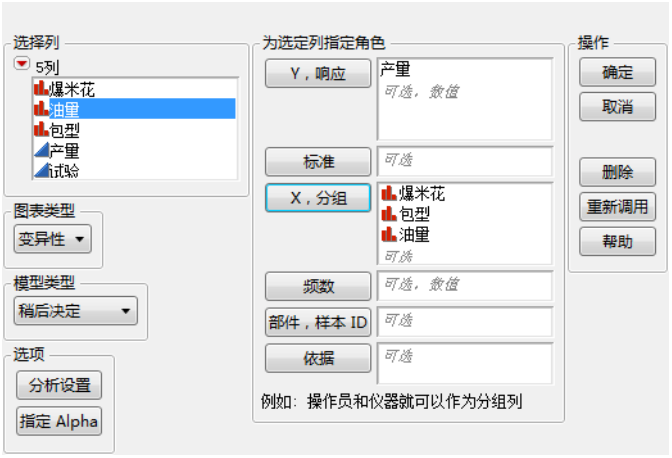
要回答这个问题，请使用产量对类型、包型大小和油量的变异性图。

创建变异性图

1. 选择**帮助** > **样本数据库**，然后打开 Popcorn.jmp。
2. 选择**分析** > **质量和过程** > **变异性 / 计数量具图**。
3. 选择**产量**并点击 **Y，响应**。
4. 选择**爆米花**并点击 **X，分组**。
5. 选择**包型**并点击 **X，分组**。
6. 选择**油量**并点击 **X，分组**。

注意：将变量分配给 **X，分组**角色的顺序很重要，因为在该窗口中的顺序决定了它们在变异性图中的嵌套顺序。

图 4.37 变异性图窗口

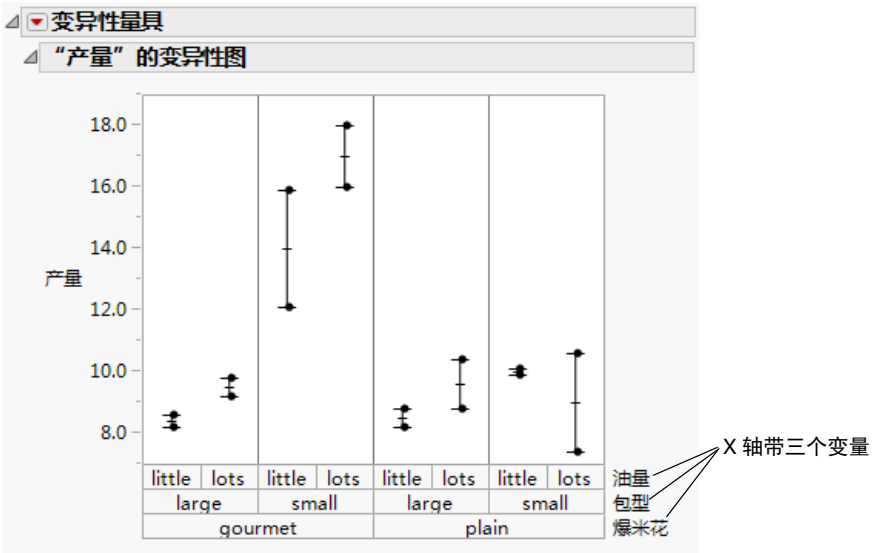


7. 点击**确定**。

   上面的图形是变异性图，显示三个变量每个组合的产量。下面的图形显示三个变量每个组合的标准差。因为下面的图形不显示产量，将它隐藏。

8. 点击 “变异性量具”红色小三角并撤销选择**标准差图**。

图 4.38 结果窗口



解释变异性图

产量的变异性图表示小包装的“gourmet”产量最高。

爆米花制造商可能还会问更具体的问题：产量高是因为包装小还是因为爆米花的种类是“gourmet”？

变异性图显示出：

- 小包装的“plain”产量低。
- 大包装的“gourmet”产量低。

鉴于该信息，爆米花制造商可以推断出仅小包装的“gourmet”产量高。在仅允许使用一个变量的图形中，不可能得出该推断。

第 5 章

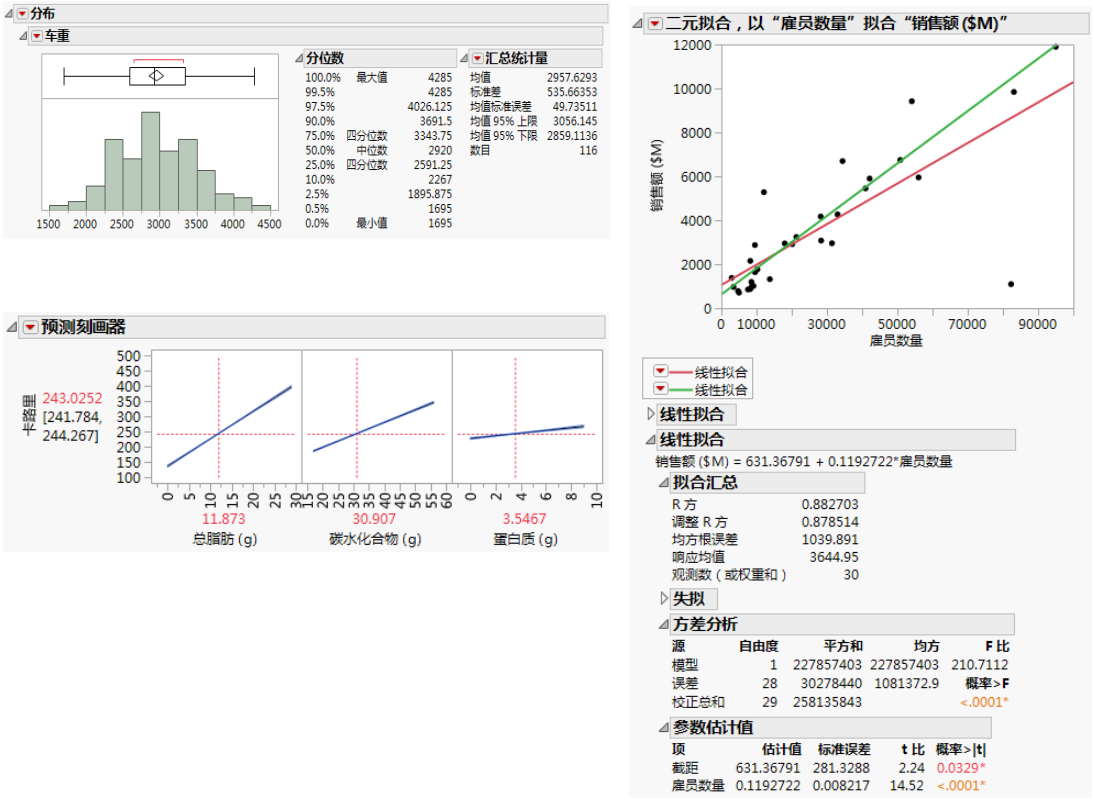
分析数据

分布、关系和模型

在 JMP 中分析数据可以帮助您做出明智的决策。数据分析通常包含下列操作：

- 检查分布
- 揭示关系
- 假设检验
- 构建模型

图 5.1 分析示例



目录

关于本章 125
数据绘图的重要性 125
了解建模类型 128
查看建模类型结果的示例 129
更改建模类型 130
分析分布 131
连续变量的分布 132
分类变量的分布 134
分析关系 137
使用单预测变量回归 137
比较一个变量的平均值 142
比较比例 145
比较多个变量的平均值 147
使用多个预测变量的回归 151

关于本章

开始分析数据之前，请先查看有关基本概念的下列信息：

- 第 125 页的“数据绘图的重要性”
- 第 128 页的“了解建模类型”

本章的剩余部分介绍如何使用 JMP 中的一些基本分析方法：

- 第 131 页的“分析分布”
- 第 137 页的“分析关系”

有关高级建模和分析技巧的说明，请参考以下 JMP 文档：

- 拟合线性模型
- 多元方法
- 预测和专业建模
- 消费者研究
- 可靠性和生存方法
- 质量和过程方法

数据绘图的重要性

对任何数据分析而言，数据绘图 （或可视化）都很重要，应该总是在使用统计检验或构建模型之前完成。要举例说明为何数据可视化是数据分析过程中的首要步骤，请考虑以下示例：

1. 选择**帮助** > **样本数据库**，然后打开 Anscombe.jmp (F. J. Anscombe (1973), American Statistician, 27, 17-21)。

   该数据由四对 X 和 Y 变量组成。

2. 在 “表”面板中，点击**四次方**脚本旁边的绿色小三角。

   该脚本使用**以 X 拟合 Y** 为每对变量创建一个简单的线性回归。**显示点**选项关闭，因此散点图中看不到任何数据。图 5.2 显示了每个回归的模型拟合和其他汇总信息。

图 5.2 四个模型

配对 1 的模型

配对 3 的模型

配对 2 的模型

配对 4 的模型

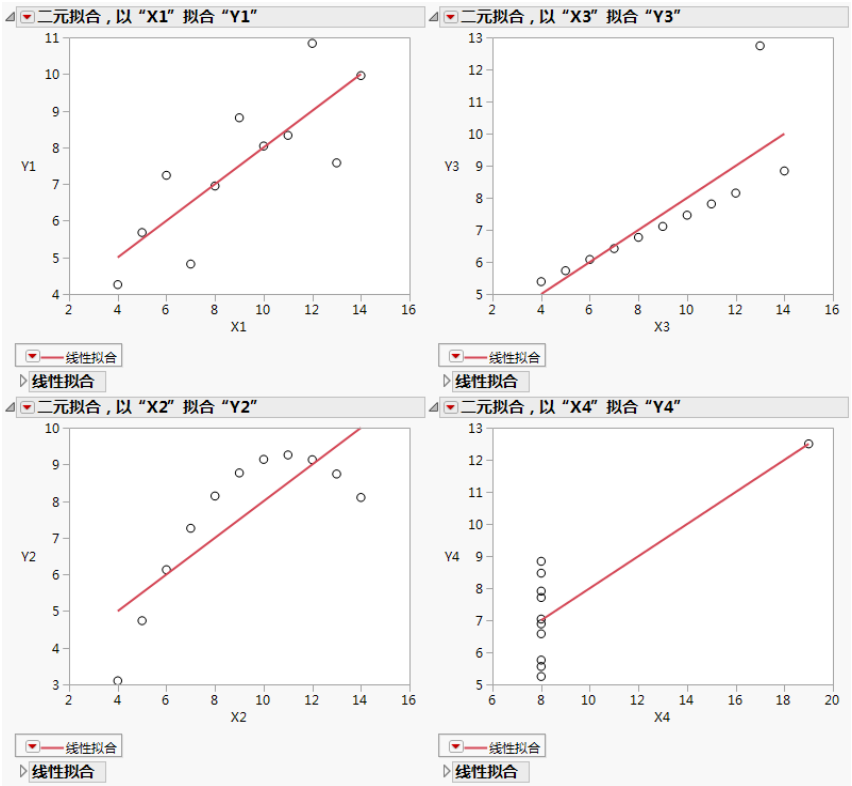
注意：所有四个模型和 R 方值基本相同。每种情况下的拟合模型基本上为 Y = 3 + 0.5X，并且每种情况下的 R 方值基本上为 0.66。若数据分析仅考虑上述汇总信息，您可能会推断出每种情况下 X 和 Y 之间的关系是相同的。但是，这时您还没有可视化数据， 您的推断可能是错误的。

要可视化数据，将各点添加到所有四个散点图中

1. 按 Ctrl 键。
2. 点击任何一个 “二元拟合”旁边的红色小三角，然后选择**显示点**。

图 5.3 添加了点的散点图

配对 1 的图形



配对 3 的图形

配对 2 的图形

配对 4 的图形

散点图显示尽管描述关系的直线相同，但四组配对 X 和 Y 之间的关系不相同：

- 图形 1 表示线性关系。
- 图形 2 表示非线性关系。
- 图形 3 表示线性关系，但一个离群值除外。
- 图形 4 除一个点以外，所有数据的 x 值均为 8。

本示例说明仅根据统计量进行推断是不充分的。直观的数据探索应该是任何数据分析的首要工作。

了解建模类型

在 JMP 中，数据可以是不同的类型。JMP 称之为数据的建模类型。表 5.1 说明了 JMP 中的三种建模类型。

表 5.1 建模类型

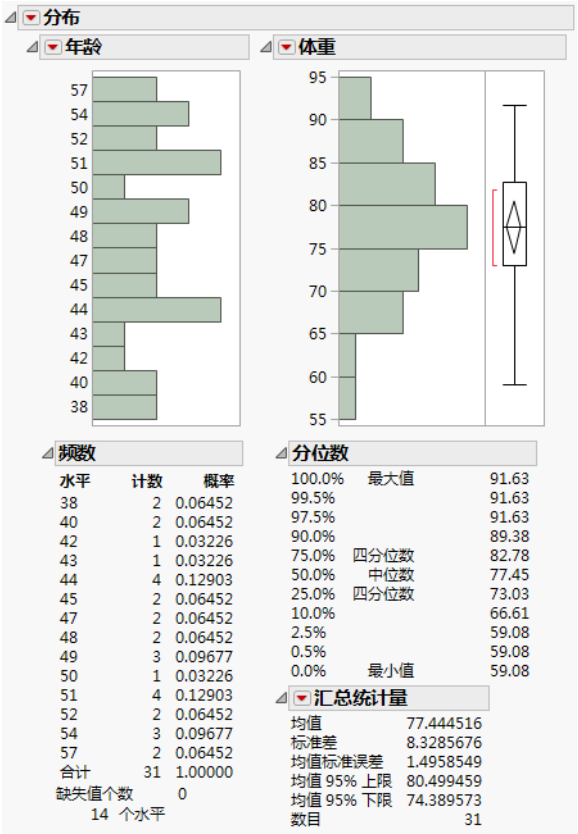
建模类型和说明	示例	具体示例
连续 仅数值数据。在运算中使用，例如总和、均值。	身高 温度 时间	完成测验的时间可能为 2 小时或 2.13 小时。
有序型 数值或字符数据。值属于有序类别。	月份 (1,2,...,12) 字母等级 (A, B,...F) 尺寸（小，中，大）	一年的月份可以是 2（二月）或 3（三月），但不能是 2.13。二月之后是三月。
名义型 数值或字符数据。值属于类别，但顺序不重要。	性别（男或女） 颜色 测试结果（及格或不及格）	性别可以是男或女，没有顺序。性别类别也可以通过数字表示（男 = 1，女 = 2）。
多重响应 仅字符数据。单个单元格中用逗号分隔的非重复条目。	当您刷牙时 大学本科学历 您从事的运动	您一天可能刷几次牙。清早、早餐后、饭后、睡前或以上各项任意组合。
非结构化文本 仅字符数据。通常必须使用“文本分析器”分析的所有唯一值。	产品评论 歌词 调查中的自由回答字段	大多数产品评论都是唯一的，并且“文本分析器”用来确定任何潜在的相似之处。
向量 仅表达式数据。单元格中的值是列或行向量。	预测公式	
无 任何数据类型。在其他建模类型不能很好地表示列的情况下使用。	图片 ID 值	数据表中的图片列将不用于建模，但举例来说，可以用作图形上的标记。

查看建模类型结果的示例

在 JMP 中，不同的建模类型会生成不同的结果。要查看关于差别的示例，请执行下列步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Linnerud.jmp。
2. 选择**分析** > **分布**。
3. 选择**年龄**和**体重**，并点击 **Y，列**。
4. 点击**确定**。

图 5.4 **年龄**和**体重**的分布结果



尽管**年龄**和**体重**都是数值变量，但它们的处理方式不同。表 5.2 比较**体重**和**年龄**结果的差异。

表 5.2 **体重**和**年龄**的结果

变量	建模类型	结果
体重	连续型	直方图、分位数和汇总统计量

表 5.2 体重和年龄的结果 (续)

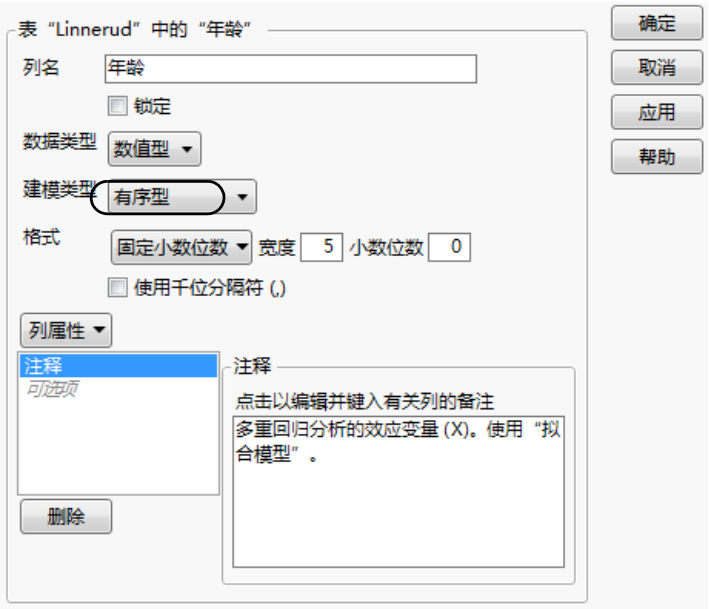
变量	建模类型	结果
年龄	有序型	条形图和频数

更改建模类型

要以不同方式处理变量时，请更改建模类型。例如，在图 5.4 中，年龄的建模类型为有序型。请记住，对于有序型变量， JMP 计算频数统计。假设您想得到平均年龄而不是频数统计，可以将建模类型更改为连续型以显示年龄均值。

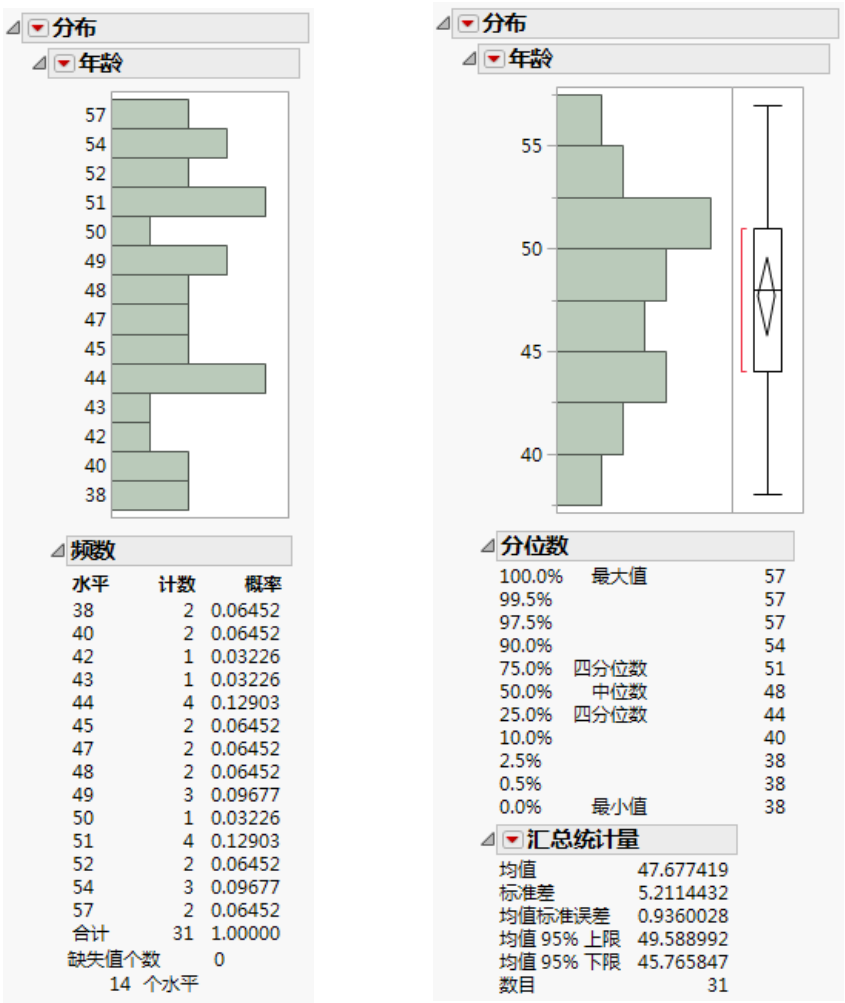
1. 双击**年龄**列标题。“列信息”窗口随即出现。
2. 将 “建模类型”更改为**连续型**。

图 5.5 列信息窗口



3. 点击**确定**。
4. 重复该示例中的步骤 （参见第 129 页的 “查看建模类型结果的示例”）以创建分布。图 5.6 显示**年龄**为有序型和连续型时的分布结果。

图 5.6 年龄的不同建模类型



有序型　　连续型

当年龄为有序型时，您可以查看每个年龄的频数。例如，年龄 48 出现 2 次。当年龄为连续型时，您可以得到年龄均值，该值接近 48 (47.677)。

分析分布

要分析单个变量，可以使用“分布”平台检查变量的分布。根据每个变量是类别变量（名义型或有序型）还是连续变量，变量的报表内容会发生变化。

注意：有关“分布”平台的详细信息，请参见《基本分析》。

连续变量的分布

分析连续变量可能会涉及下列问题：

- 数据形状与任何已知的分布匹配吗？
- 数据中有离群值吗？
- 数据的平均值是多少？
- 平均值从统计意义上来说不同于目标值或历史值吗？
- 数据的分散程度如何？换句话说，标准差是多少？
- 最小值和最大值是多少？

您可以使用图形、汇总统计量和简单的统计检验来回答这些以及其他问题。

情境

本示例使用 Car Physical Data.jmp 数据表，表中包含 116 种不同汽车型号的信息。

铁路公司向设计专家求教，以确定通过火车运输汽车的过程中可能出现的问题。使用该数据，设计专家想探究下列问题：

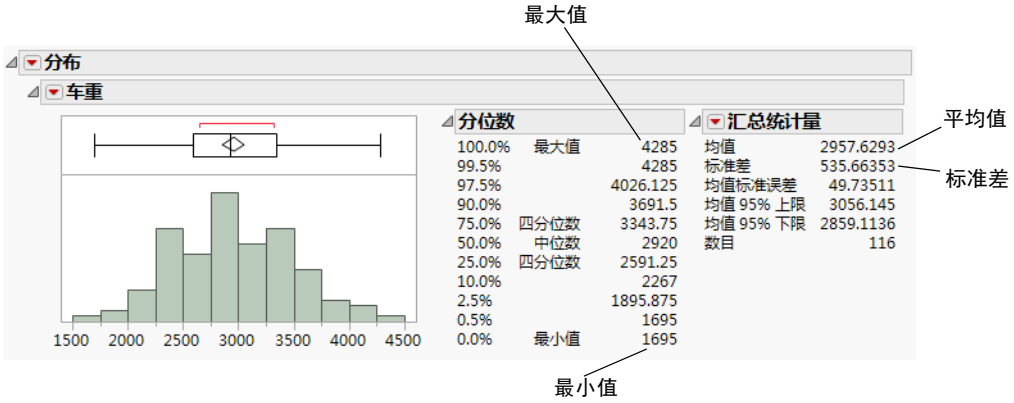
- 汽车平均重量是多少？
- 汽车重量的分散程度（标准差）如何？
- 汽车最小和最大重量是多少？
- 数据中有离群值吗？

使用车重的直方图可以回答这些问题。

创建直方图

1. 选择**帮助** > **样本数据库**，然后打开 Car Physical Data.jmp。
2. 选择**分析** > **分布**。
3. 选择**车重**并点击 **Y，列**。
4. 点击**确定**。
5. 要旋转报表窗口，点击 “车重”红色小三角并选择**显示选项** > **水平布局**。

图 5.7 车重的分布



报表窗口包含三个部分：

- 用于可视化数据的直方图和箱线图。
- 显示分布百分位数的分位数报表。
- 显示均值、标准差和其他统计量的汇总统计量报表。

解释分布结果

使用图 5.7 中提供的结果，设计专家可以回答问题。

汽车平均重量是多少？ “直方图”显示重量在 3,000 磅左右。

汽车重量的分散程度（标准差）如何？ “汇总统计量”显示重量在 2,958 磅左右。“汇总统计量”显示标准差在 536 磅左右。

最小和最大重量是多少？ “直方图”显示最小值在 1,500 磅左右；最大值在 4,500 磅左右。“分位数”显示最小值在 1,695 磅左右；最大值在 4,285 磅左右。

有离群值吗？ 没有。

图 5.7 中的默认报表窗口至少提供图形和统计量。其他图形和统计量可以通过红色小三角菜单得到。

结论

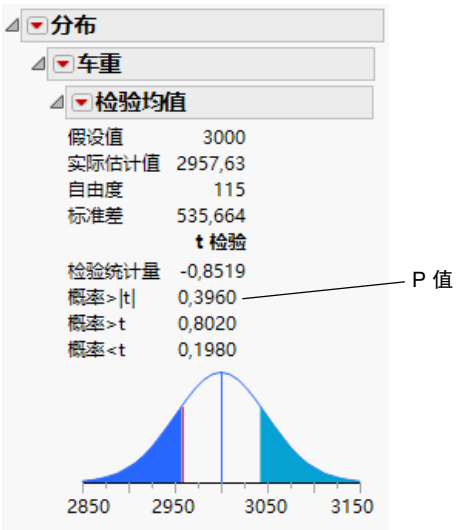
根据其他研究，铁路公司确定平均重量为 3000 磅是最高效的运输重量。现在，设计专家需要确定在想要运输的汽车总体中汽车平均重量是否为 3000 磅。使用 t 检验可以根据这个总体样本得到更大总体的推论。

检验结论

1. 点击“车重”红色小三角并选择**检验均值**。

2. 在出现的窗口中，在 “指定假设均值” 框中键入 3000。
3. 点击**确定**。

图 5.8 检验均值结果



解释 t 检验

t 检验的主要结果是 p 值。在该示例中，p 值为 0.396，分析人员使用显著水平 0.05。因为 0.396 大于 0.05，您不能推断出在更大总体中汽车平均重量与 3000 磅之间存在显著差异。若 p 值小于显著水平，设计专家会推断出在更大总体中汽车平均重量与 3000 磅之间**存在**显著差异。

分类变量的分布

分析分类 （有序型或名义型）变量可能会涉及下列问题：

- 变量有多少个水平？
- 每个水平有多少个数据点？
- 数据是统一分布吗？
- 每个水平表示总体的多大比例？

情境

参见第 132 页的 “连续变量的分布” 中的情境。

现在铁路公司确定了汽车的平均重量与目标重量之间没有显著差异，铁路公司还有其他问题。

设计专家想回答铁路公司的下列问题：

- 汽车是什么类型？

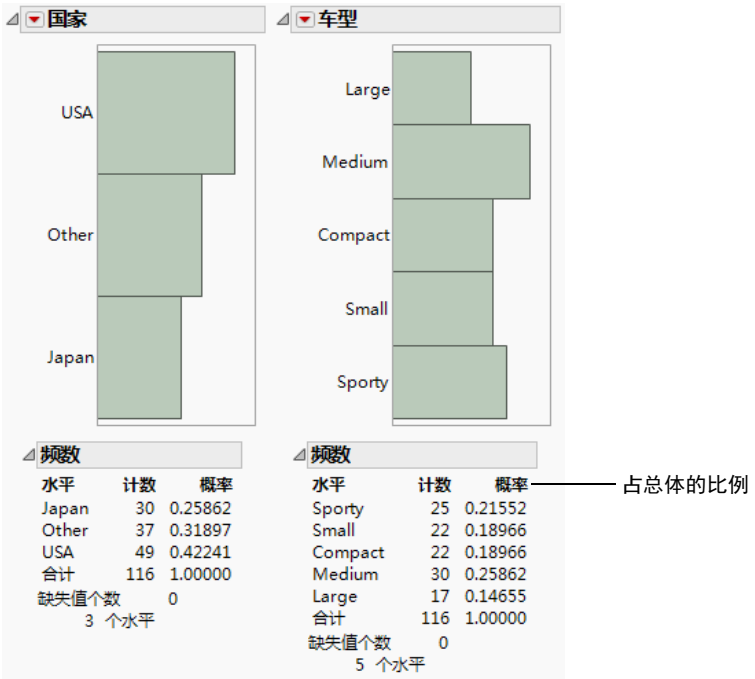
- 原产国是哪里？

要回答这些问题，查看**车型**和**产地**的分布。

创建分布

1. 选择**帮助** > **样本数据库**，然后打开 Car Physical Data.jmp。
2. 选择**分析** > **分布**。
3. 选择**产地**和**车型**，并点击 **Y，列**。
4. 点击**确定**。

图 5.9 产地和车型的分布



解释分布结果

报表窗口包含**产地**和**车型**的条形图和“频数”报表。条形图是“频数”报表中提供的频数信息的图形表示。“频数”报表包含下列信息：

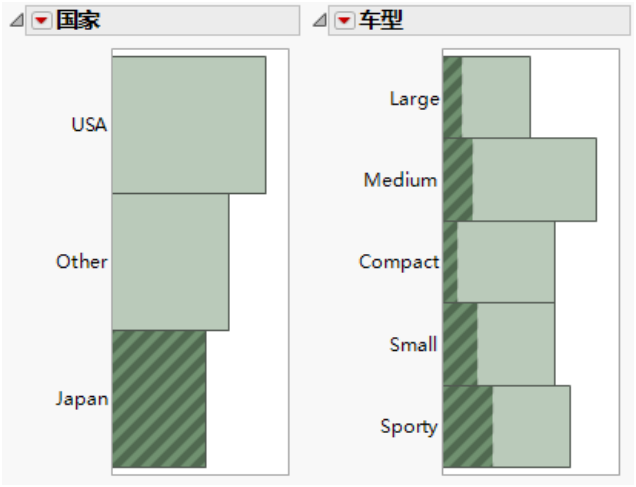
- 数据类别。例如，“Japan”是“产地”的一个类别，“Sporty”是“车型”的一个类别。
- 每个类别的总计数。
- 每个类别占总体的比例。

例如，有 22 辆紧凑型汽车，大约占 116 个观测的 19%。

与分布结果交互

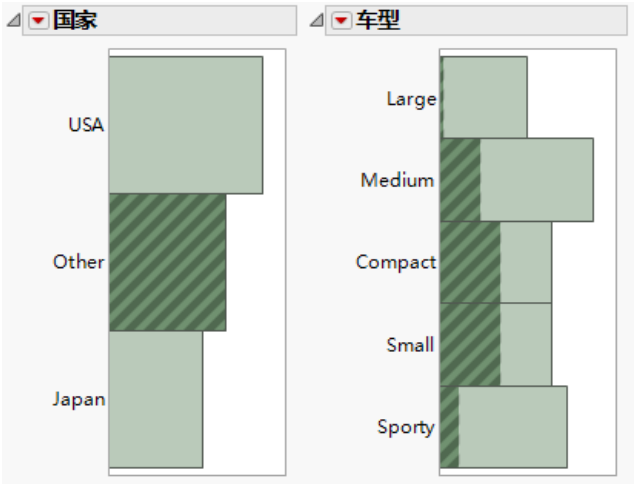
在一个图形中选择某个直条时它也会在其他图形中选择对应的数据。例如，在“产地”条形图中选择“Japan”直条会发现大部分日本车是跑车。

图 5.10 日本车



选择“Other”类别会发现大多数汽车是小型车，几乎没有大型车。

图 5.11 其他汽车



分析关系

散点图和其他此类图形可以帮助您可视化变量之间的关系。可视化关系之后，下一步就是分析这些关系，以便可以通过数字进行描述。用数字描述变量之间的关系称为**模型**。更重要的是，模型还可以根据一个变量 (X) 的值预测另一个变量 (Y) 的平均值。X 变量也称为预测变量。通常情况下，该模型称为**回归**模型。

在 JMP 中，**以 X 拟合 Y** 平台和**拟合模型**平台可以创建回归模型。

注意：此处仅介绍基本的平台和选项。有关所有平台选项的解释，请参见《基本分析》、《基本绘图》和第 125 页的“关于本章”中列出的文档。

表 5.3 显示四个主要关系类型。

表 5.3 关系类型

X	Y	章节
连续	连续	• 第 137 页的“使用单预测变量回归” • 第 151 页的“使用多个预测变量的回归”
分类	连续型	• 第 142 页的“比较一个变量的平均值” • 第 147 页的“比较多个变量的平均值”
分类	分类	第 145 页的“比较比例”
连续型	分类	Logistic 回归是高级主题。请参见《基本分析》。

使用单预测变量回归

情境

本示例使用 Companies.jmp 数据表，表中包含来自医药和计算机行业的 32 家公司的财务数据。

从直觉上来看，相比雇员较少的公司，雇员较多的公司产生的销售收入更高。数据分析人员想根据雇员数量预测每家公司的总体销售收入。

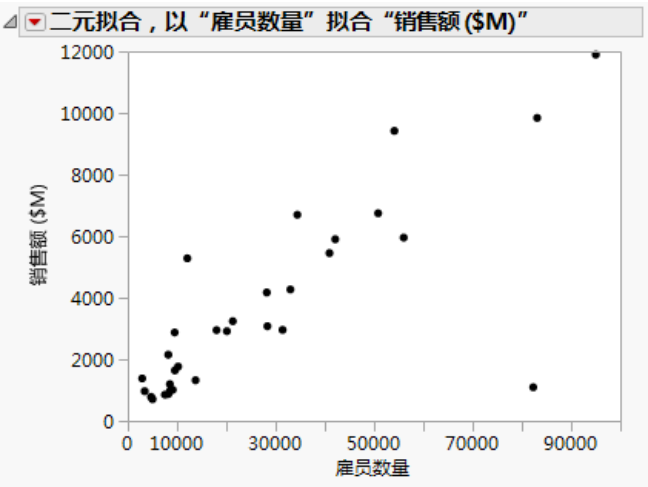
要完成该任务，请执行下列操作：

- 第 138 页的“发现关系”
- 第 138 页的“拟合回归模型”
- 第 140 页的“预测平均销售额”

发现关系

首先，创建一个散点图以查看雇员数量和销售收入之间的关系。该散点图在“数据可视化”一章中第 95 页的“创建散点图”中创建。隐藏并排除一个离群值（雇员数量和销售额明显较高的公司）之后，图 5.12 中的图形显示出结果。

图 5.12 雇员数量 - 销售额 ($M) 散点图

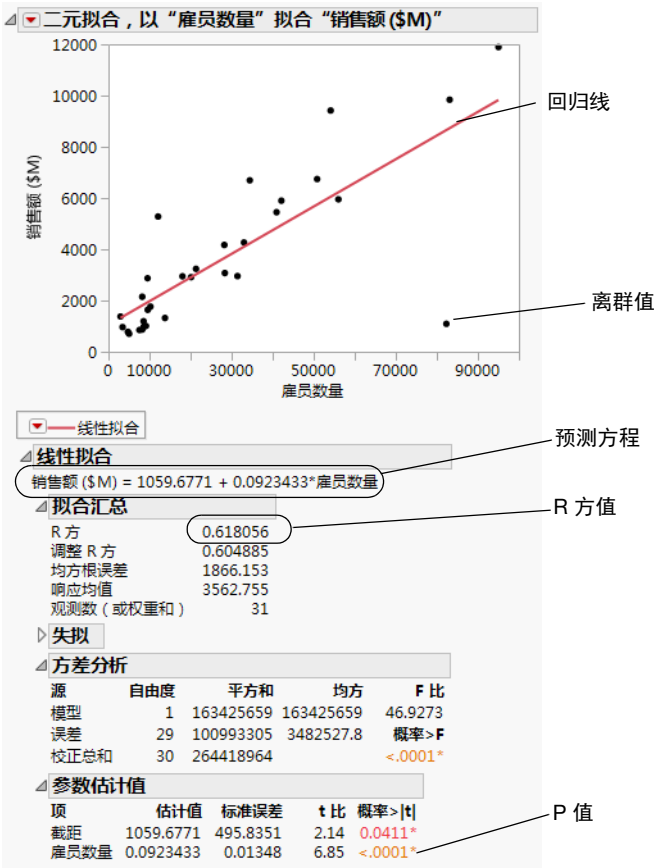


该散点图清晰地显示出销售额和雇员数量之间的关系。和预想的一样，公司的雇员越多，销售额越高。这直观地印证了数据分析人员的猜测，但不能在给定雇员数量的情况下预测出销售额。

拟合回归模型

要根据雇员数量预测出销售收入，可以拟合回归模型。点击“二元拟合”红色小三角并选择**拟合线**。回归线随即添加到散点图中，并且报表添加到报表窗口中。

图 5.13 回归线



在报表中，看到下列结果：

- p 值 <.0001
- R 方值为 0.618

从这些结果中，数据分析人员可以推断出下列结论：

- “雇员数量”模型项的 p 值很小。这支持了在 0.05 显著性水平下“雇员数量”的系数不为零。因此，在预测模型中包括雇员数量可显著改善通过不含雇员数量的模型预测平均销售额的能力。
- R 方值为 0.618 表示该模型解释了销售额中大约 62% 的变异性。R 方值是决定系数，它表示因（响应）变量方差中可以被该模型解释的那部分。R 方值的范围可以从 0 到 1。R 方值为 0 的模型没有解释能力。R 方值为 1 的模型可以完美预测响应。

预测平均销售额

使用回归模型可以预测出公司可能预期的平均销售额（若公司具有一定数量的雇员）。模型的预测方程包含在报表中：

平均销售额 = 1059.68 + 0.092* 雇员数量

例如，在具有 70,000 名雇员的公司中，预测的销售额大约为 $7,500：

$7,499.68 = 1059.68 + 0.092*70,000

在当前散点图的右下方区域，有一个离群值，它与其他公司的一般模式不同。数据分析人员想知道排除该离群值后预测模型是否会发生变化。

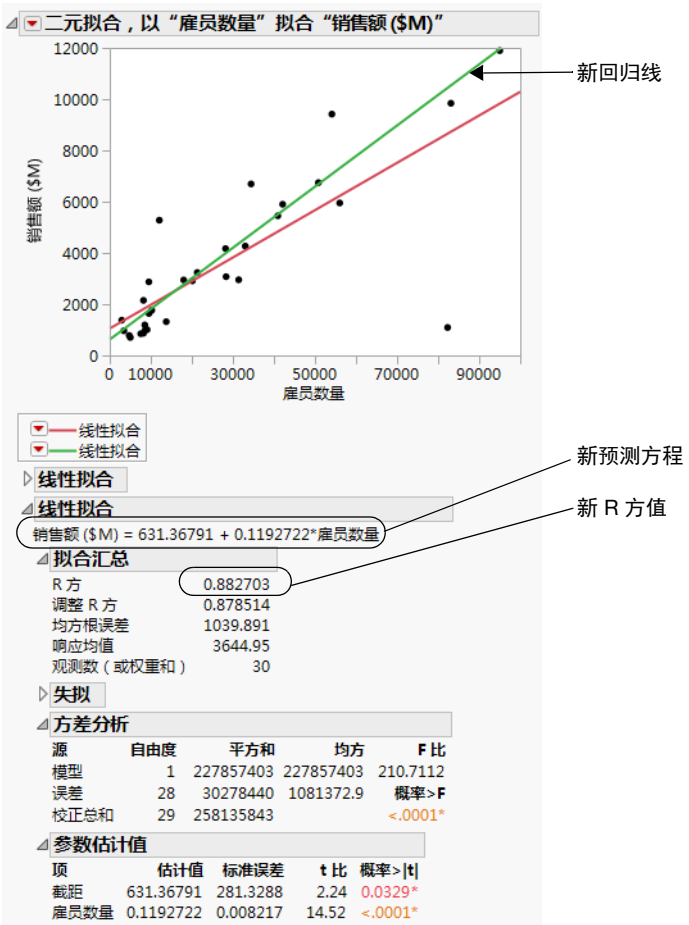
排除离群值

1. 点击离群值。
2. 选择**行 > 排除 / 撤销排除**。
3. 要拟合该模型，点击“二元拟合，以 ‘雇员数量’ 拟合 ‘销售额 ($M)’”旁边的红色小三角，然后选择**拟合线**。

下列项被添加到报表窗口中（图 5.14）：

- 一条新回归线
- 一个新的“线性拟合”报表，包括：
  - 一个新的预测方程
  - 一个新的 R 方值

图 5.14 比较模型



解释结果

使用图 5.14 中的结果，数据分析人员可以得出下列结论：

- 离群值将较大公司的回归线拉低，将较小公司的回归线拉高。
- 不含离群值数据的新模型比第一个模型更强。新的 R 方值为 0.88，与初始分析相比，该值更高且更接近 1。

结论

使用新预测方程，拥有 70,000 名雇员的公司的预测平均销售额可以计算如下：

$8961.37 = 631.37 + 0.119*70,000

第一个模型的预测结果大约为 $7500。第二个模型预测销售总额大约为 $8960，与第一个模型相比增加了 $1460。

删除离群值之后，第二个模型根据雇员数量描述、预测销售总额的效果比第一个模型要好。数据分析人员现在有很好的模型可以使用。

比较一个变量的平均值

若您具有连续 Y 变量和分类 X 变量，则可以比较 X 变量不同水平下的平均值。

情境

本示例使用 Companies.jmp 数据表，表中包含来自医药和计算机行业的 32 家公司的财务数据。

财务分析人员想探究以下问题：

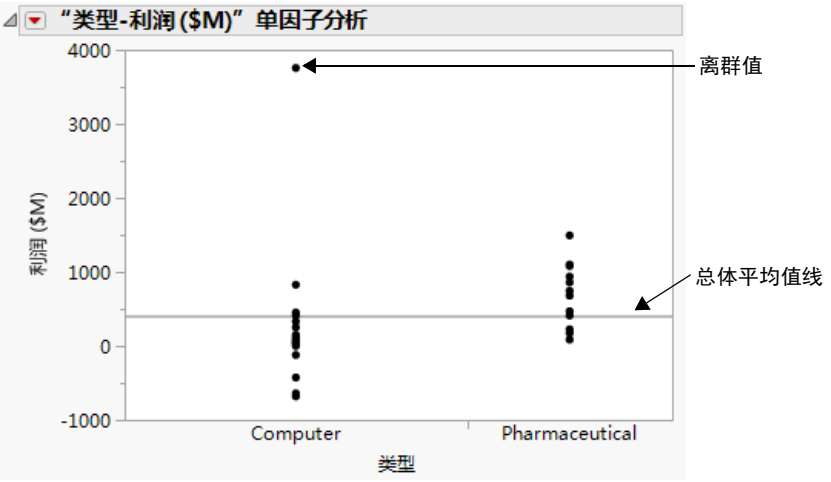
- 计算机公司的利润与医药公司的利润相比怎么样？

要回答该问题，以**类型**拟合**利润 ($M)**。

发现关系

1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 若您仍然打开着 Companies.jmp 样本数据表，则可能排除或隐藏了行。要将这些行恢复为默认状态 （包含所有行并且不隐藏任何行），选择**行 > 清除行状态**。
3. 选择**分析 > 以 X 拟合 Y**。
4. 选择**利润 ($M)** 并点击 **Y，响应**。
5. 选择**类型**并点击 **X，因子**。
6. 点击**确定**。

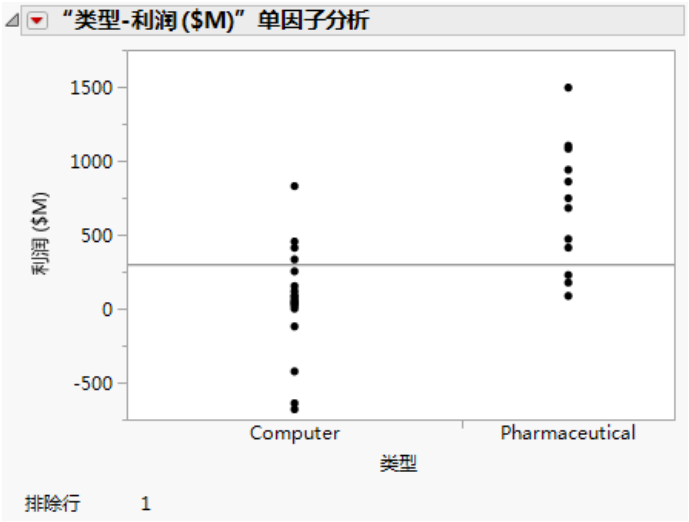
图 5.15 公司类型 - 利润



“计算机”类型中有一个离群值。该离群值扩大了图形的范围，使得利润的比较很困难。排除并隐藏该离群值：

1. 点击离群值。
2. 选择**行 > 排除 / 撤销排除**。数据点不再包含在计算中。
3. 选择**行 > 隐藏 / 撤销隐藏**。数据点不再显示在所有图形中。
4. 要重新创建不包含离群值的图形，点击“‘类型 - 利润 ($M)’单因子分析”并选择**重新运行 > 重新运行分析**。可以关闭原始“散点图”窗口。

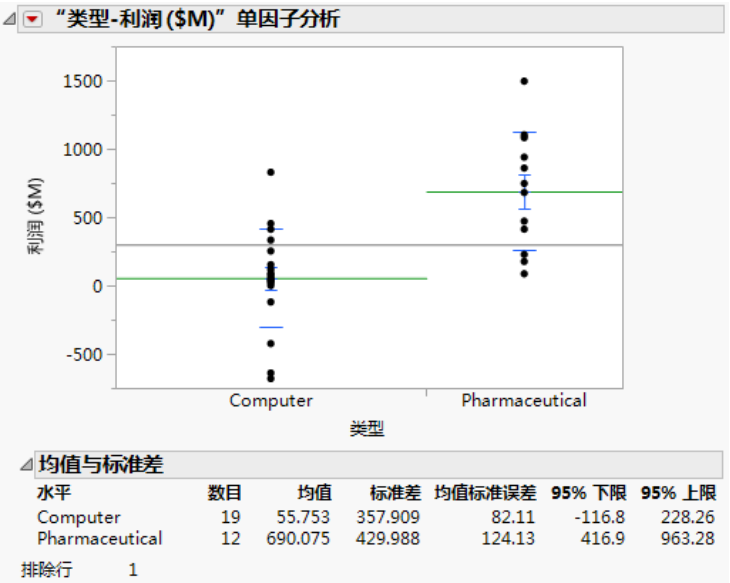
图 5.16 更新的图形



删除离群值可以帮助财务分析人员更加清晰地查看数据。

5. 要继续分析关系，从“‘类型 - 利润 ($M)’单因子分析”旁边的红色小三角中选择下列选项：
   - **显示选项 > 均值线**。这会将均值线添加到散点图中。
   - **均值和标准差**。这将显示一个报表，其中提供平均值和标准差。

图 5.17 均值线和报表



解释结果

财务分析人员想知道计算机公司利润和医药公司利润的比较情况。更新的散点图显示医药公司比计算机公司的平均利润要高。在报表中，若您从一个均值中减去另一个均值，利润差异大约为 $6.35 亿。图形还显示某些计算机公司的利润为负值，而所有医药公司的利润均为正值。

执行 t 检验

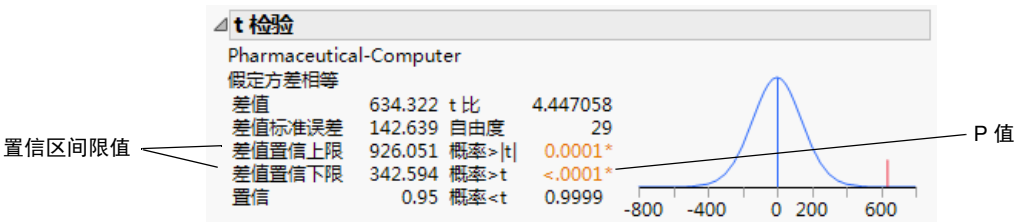
财务分析人员仅查看了诸公司的一个样本数据 （数据表中的公司）。财务分析人员现在想研究下列问题：

- 在更大总体中是否存在差异，或者 $6.35 亿的差异是否因偶然因素而造成？
- 若存在差异，则差异是什么？

要回答这些问题，执行双样本 t 检验。通过 t 检验，可以使用样本中的数据对更大总体进行推断。

要执行 t 检验，点击“单因子分析”红色小三角并选择**均值/方差分析/合并的 t**。

图 5.18 t 检验结果



p 值为 0.0001，小于显著性水平 0.05（表示统计显著性）。因此，财务分析人员可以推断出样本数据的平均利润的观测差异具有统计显著性。这意味着在更大总体中，医药公司的平均利润不同于计算机公司的平均利润。

结论

使用置信区间限值可以确定在两种公司类型的利润中存在的差异程度。查看图 5.18 中的**差异置信上限**和**差异置信下限**值。财务分析人员可以推断出：医药公司的平均利润比计算机公司的平均利润高出的值介于 $3.43 亿和 $9.26 亿之间。

比较比例

若您具有分类 X 和 Y 变量，则可以将 Y 变量内的水平比例与 X 变量内的水平比例进行比较。

情境

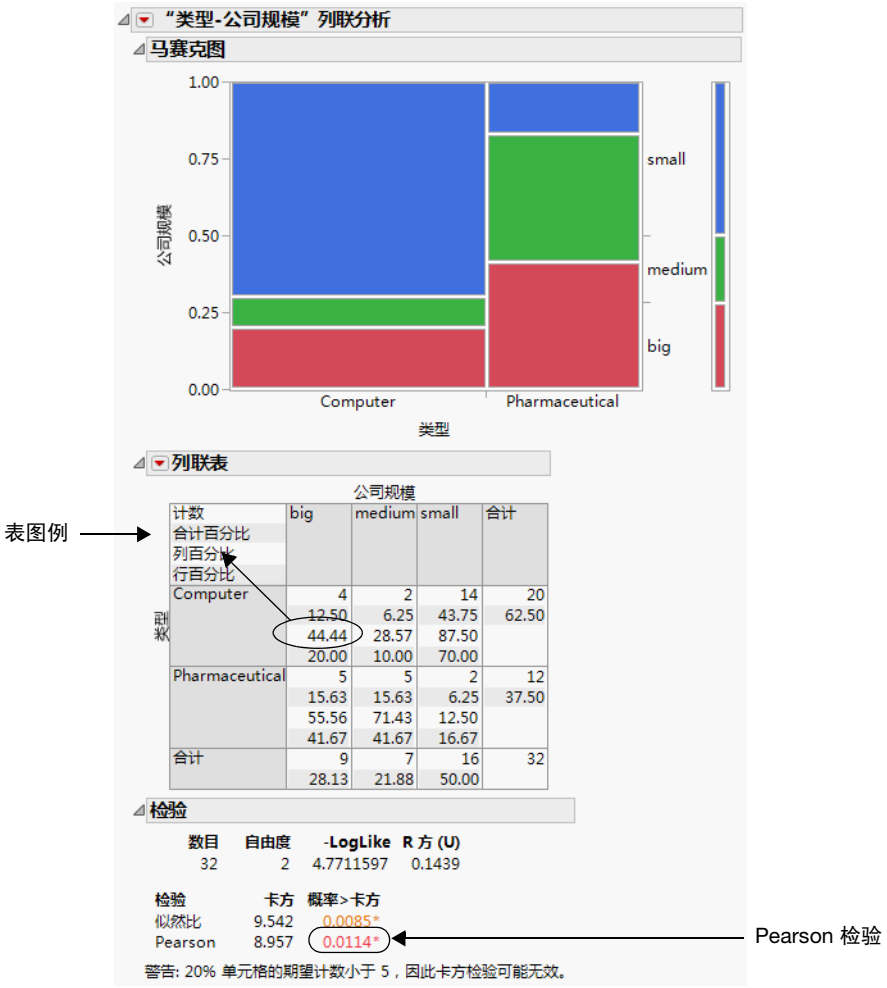
本示例继续使用 Companies.jmp 数据表。在第 142 页的“比较一个变量的平均值”中，财务分析人员确定了医药公司比计算机公司的平均利润高。

财务分析人员想知道某一类公司的规模是否比另一类公司的规模更多地影响利润？但是，在研究该问题之前，财务分析人员需要了解计算机和医药公司总体是否由相同比例的小、中、大规模的公司组成。

发现关系

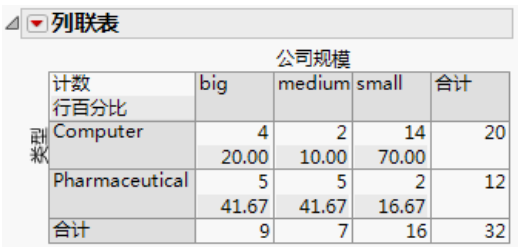
1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 若您仍然打开着上一示例中的 Companies.jmp 数据文件，则可能排除或隐藏了行。要将这些行恢复为默认状态 （包含所有行并且不隐藏任何行），选择**行** > **清除行状态**。
3. 选择**分析** > **以 X 拟合 Y**。
4. 选择**公司规模**并点击 **Y，响应**。
5. 选择**类型**并点击 **X，因子**。
6. 点击**确定**。

图 5.19 公司类型 - 公司规模



列联表包含的信息不适用于该示例。点击“列联表”红色小三角并撤销选择**合计百分比**和**列百分比**以删除该信息。图 5.20 显示了更新的表。

图 5.20 更新的列联表



解释结果

“列联表”中的统计量在马赛克图中以图形表示。“马赛克图”和“列联表”同时比较两个行业中小、中、大规模公司的百分比。例如，“马赛克图”显示计算机行业小规模公司所占比例高于医药行业。“列联表”显示确切的统计量：70% 的计算机公司是小规模的，而小规模的医药公司大约是 17%。

解释检验

财务分析人员仅查看了诸公司的一个样本数据（数据表中的公司）。财务分析人员需要知道在所有计算机和医药公司的更大总体中百分比是否存在差异。

要回答该问题，使用**检验**报表中 Pearson 检验中的 p 值 （第 146 页上的图 5.19）。由于 p 值为 0.011，小于显著性水平 0.05，财务分析人员得出以下结论：

- 样本数据中的差异具有统计显著性。
- 在更大总体中百分比存在差异。

现在财务分析人员知道了小、中、大规模公司的比例是不同的，可以回答这个问题了：某一类公司的规模是否比另一类公司的规模更多地影响利润？

比较多个变量的平均值

第 142 页的“比较一个变量的平均值”这一节比较了分类变量不同水平下的平均值。要同时比较两个或多个变量不同水平下的平均值，请使用**方差分析**法（即 ANOVA）。

情境

财务分析人员可以回答我们在“比较比例”这一节提出的问题，即：根据公司类型（医药或计算机），公司规模对公司利润有更大的影响吗？

要回答该问题，通过以下两个变量比较公司利润：

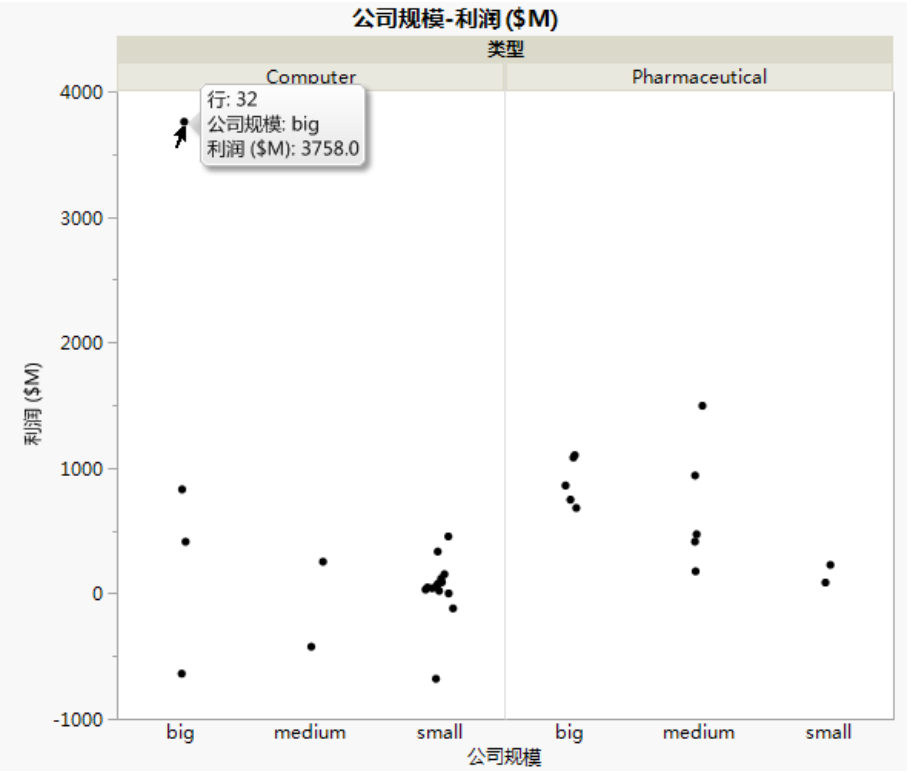
- **类型**（pharmaceutical 或 computer）
- **规模**（small、medium、big）

发现关系

要将类型和规模的所有组合对应的利润差异可视化，请使用图形：

1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 选择**图形 > 图形生成器**。“图形生成器”窗口随即出现。
3. 点击**利润 ($M)** 并将其拖放到 **Y** 区域中。
4. 点击**公司规模**并将其拖放到 **X** 区域中。
5. 点击**类型**并将其拖放到 **X 分组**区域中。

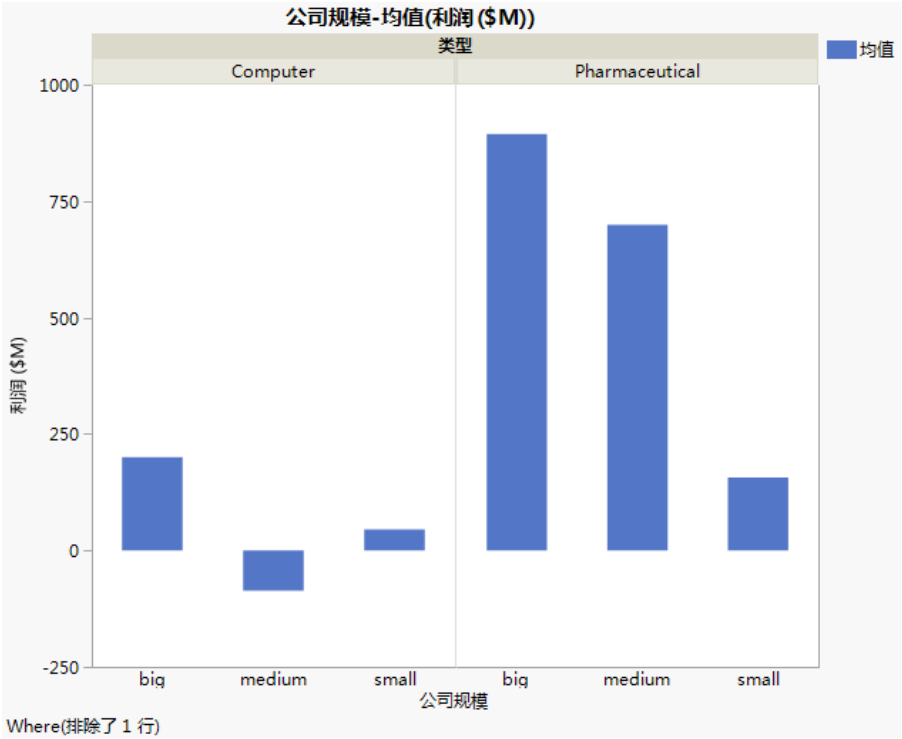
图 5.21 公司利润图形



图形显示一家大规模的计算机公司具有很高的利润。该离群值扩展了图形的尺度，使得比较其他数据点很困难。

6. 选择该离群值，然后右击并选择**行 > 排除行**。该点被删除，并且图形的尺度自动更新。
7. 点击 “条形图” 图标。使用条形图比较利润均值比使用点图更方便。

图 5.22 删除了离群值的图形



更新的图形显示医药公司具有更高的平均利润。此外还显示仅医药公司的公司规模之间存在利润差异。当一个变量（公司规模）的效应因为另一个变量（公司类型）的不同水平发生变化时，这称为**交互**。

量化关系

由于该数据仅是样本，财务分析人员需要确定：

- 差异是否限于该样本且归因于偶然因素

  或

- 在更大总体中是否存在相同的模式

1. 返回已排除了数据点的 Companies.jmp 样本数据表。请参见第 147 页的“发现关系”。
2. 选择**分析 > 拟合模型**。
3. 选择**利润 ($M)** 并点击 **Y**。
4. 同时选择**类型**和**公司规模**。
5. 点击**宏**按钮并选择**完全析因**。
6. 从“重点”菜单中，选择**效应筛选**。

7. 选中**保持对话框打开**选项。

图 5.23 完成的拟合模型窗口

8. 点击**运行**。报表窗口显示模型结果。

要确定利润的差异是真实的，还是由于偶然因素，请检查**效应检验**报表。

注意：有关所有“拟合模型”结果的详细信息，请参见《拟合线性模型》。

查看效应检验

效应检验报表（图 5.24）显示统计检验的结果。在“拟合模型”窗口中有针对模型中包含的每个效应的检验：类型、公司规模和类型*公司规模。

图 5.24 效应检验报表

效应检验

源	参数数目	自由度	平方和	F 比	概率>F
类型	1	1	1401847.4	10.1368	0.0039*
公司规模	2	2	724616.2	2.6198	0.0927
类型*公司规模	2	2	448061.5	1.6200	0.2180

首先，查看模型中对交互作用进行的检验：“类型 * 公司规模”效应。图 5.22 表明不同规模的医药公司会有不同的利润。但是，效应检验指出在对利润的影响上，类型和规模之间没有交互作用。 p 值为 0.218，大于显著性水平 0.05。因此，从模型中删除该效应，并重新运行模型。

1. 返回到“拟合模型”窗口。

2. 在 “构造模型效应” 框中，选择**类型** * **公司规模**效应并点击**删除**。
3. 点击**运行**。

图 5.25 更新的效应检验报表

效应检验

源	参数数目	自由度	平方和	F 比	概率>F
类型	1	1	1356297.9	9.3768	0.0049*
公司规模	2	2	434161.3	1.5008	0.2410

“公司规模”效应的 p 值很大，表示在更大总体中基于规模不存在差异。“类型”效应的 p 值很小，表示在计算机和医药公司数据之间看到的差异不是由于偶然因素造成的。

结论

财务分析人员想知道基于类型 （医药或计算机），公司规模是否对公司利润有较大的影响。财务分析人员现在可以回答该问题：

- 在更大总体中，计算机和医药公司之间的利润存在实际差异。
- 公司规模和类型与公司利润之间没有相关性。

使用多个预测变量的回归

第 137 页的 “使用单预测变量回归” 这一节显示如何构建由一个预测变量和一个响应变量组成的简单回归模型。**多重回归**使用两个或多个预测变量预测平均响应变量。

情境

本示例使用 Candy Bars.jmp 数据表，表中包含糖果条的营养信息。

营养学家想使用下列信息预测卡路里：

- 总脂肪
- 碳水化合物
- 蛋白质

使用**多重回归**可以通过三个预测变量预测平均响应变量。

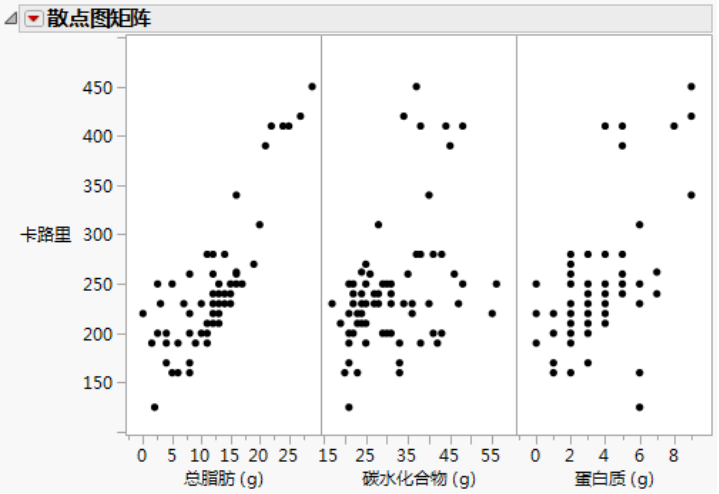
发现关系

要可视化卡路里与总脂肪、碳水化合物和蛋白质之间的关系，创建散点图矩阵：

1. 选择**帮助** > **样本数据库**，然后打开 Candy Bars.jmp。
2. 选择**图形** > **散点图矩阵**。
3. 选择**卡路里**并点击 **Y，列**。

4. 选择**总脂肪 (g)**、**碳水化合物 (g)** 和**蛋白质 (g)**，并点击 X。
5. 点击**确定**。

图 5.26 散点图矩阵结果



散点图矩阵显示卡路里与这三个变量之间存在正相关性。卡路里和总脂肪之间的相关性最强。既然营养学家认识到存在关系，则可以构建多重回归模型以预测平均卡路里。

构建多重回归模型

继续使用 Candy Bars.jmp 样本数据表。

1. 选择**分析** > **拟合模型**。
2. 选择**卡路里**并点击 Y。
3. 选择**总脂肪 (g)**、**碳水化合物 (g)** 和**蛋白质 (g)**，并点击**添加**。
4. 在 “重点”旁边，选择**效应筛选**。

图 5.27 拟合模型窗口

5. 点击运行。

报表窗口显示模型结果。要解释模型结果，请关注以下方面：

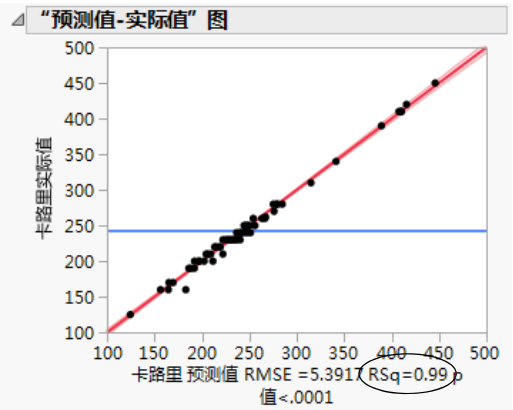
- 第 153 页的“查看“预测值-实际值”图”
- 第 154 页的“解释参数估计值”
- 第 155 页的“使用预测刻画器”

注意：有关所有模型结果的详细信息，请参见《拟合线性模型》。

查看“预测值 - 实际值”图

“预测值-实际值”图显示预测卡路里与实际卡路里的关系图。随着预测值逐渐接近实际值，散点图中的点逐渐落在红色线附近（图 5.28）。因为各点非常接近该线，您可以看到模型根据所选的因子预测卡路里的效果很好。

图 5.28 “预测值 - 实际值”图



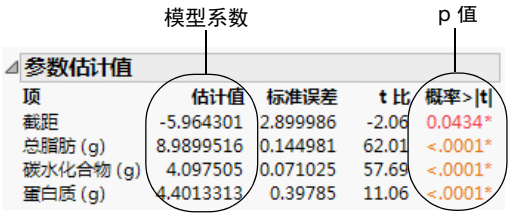
模型准确度的另一个测度是 R 方值（显示在图 5.28 中图形的下方）。R 方值测量该模型中解释的卡路里变异性的百分比。值越接近 1 表示模型预测效果越好。在该示例中，R 方值为 0.99。

解释参数估计值

“参数估计值”报表显示下列信息：

- 模型系数
- 每个参数的 p 值

图 5.29 “参数估计值”报表



在该示例中，p 值都非常小 (<.0001)。这表示所有三个效应（脂肪、碳水化合和蛋白质）在预测卡路里时都做出显著贡献。

您可以使用模型系数预测脂肪、碳水化合物和蛋白质的特定值所对应的卡路里值。例如，假设您想预测具有下列特征的糖果条的平均卡路里：

- 脂肪 = 11 g
- 碳水化合物 = 43 g
- 蛋白质 = 2 g

使用这些值，您可以按照下面的方程计算预测的平均卡路里：

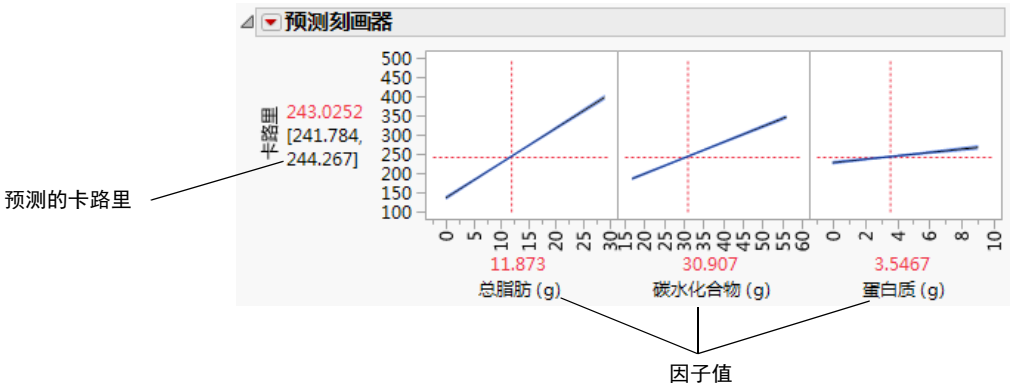
277.92 = -5.9643 + 8.99*11 + 4.0975*43 + 4.4013*2

本示例中的特征与 Milky Way 糖果条（在数据表的 59 行）相同。Milky Way 的实际卡路里是 280，显示出模型预测效果很好。

使用预测刻画器

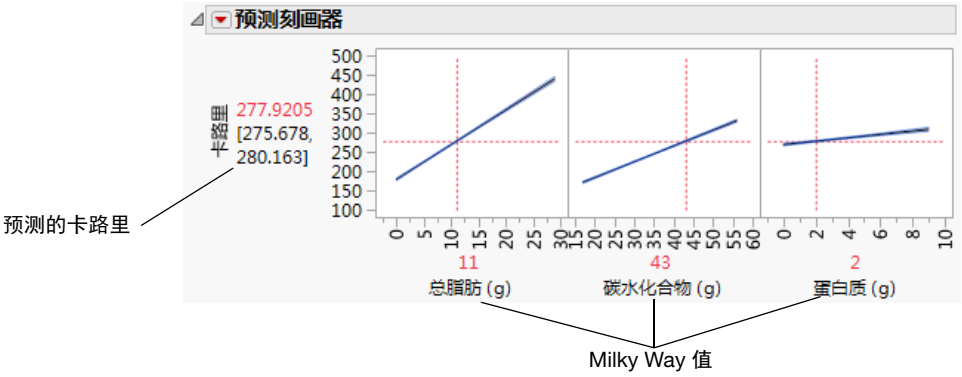
使用“预测刻画器”可以查看因子的变化如何影响预测值。刻画线显示随因子变化卡路里的变化量。代表**总脂肪 (g)** 的线条最陡，表示总脂肪变化对卡路里的影响最大。

图 5.30 预测刻画器



点击并拖动每个因子的垂直线，查看预测值如何变化。也可以点击当前的因子值进行更改。例如，点击因子值并为 Milky Way 糖果条（59 行）键入值。

图 5.31 Milky Way 的因子值



注意：有关“预测刻画器”的详细信息，请参见《刻画器指南》。

结论

营养学家现在有很好的模型来根据糖果条的总脂肪、碳水化合物和蛋白质来预测卡路里。

第 6 章

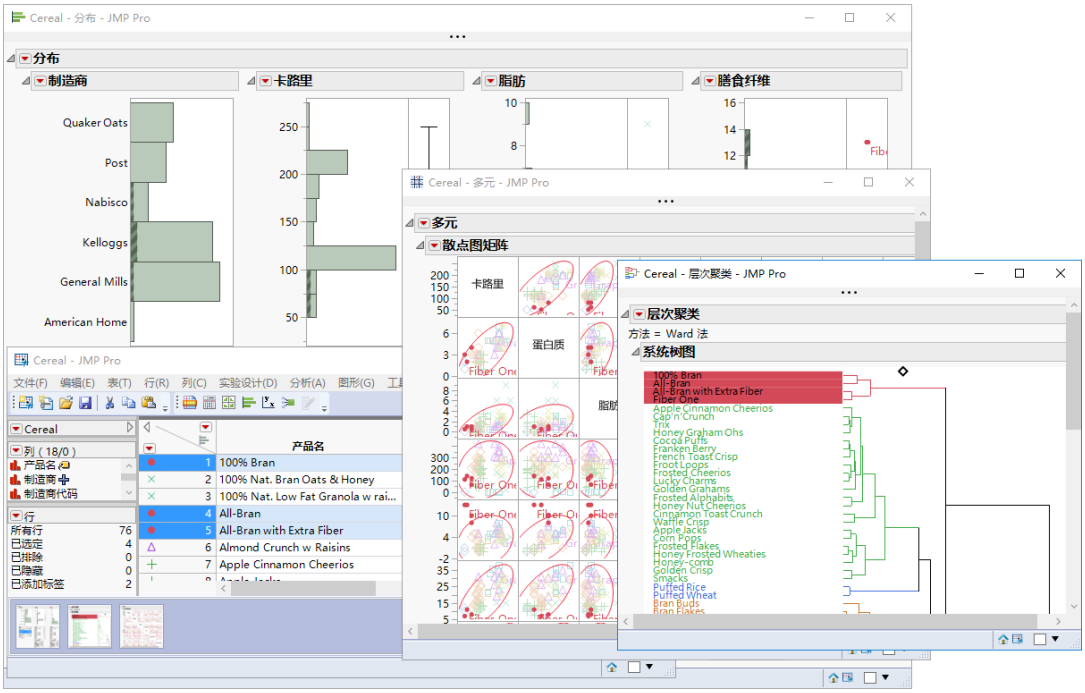
概览

在多个平台中探索数据

JMP 提供众多的统计发现平台，用来协助您探索数据的不同方面。您可以从简单地查看直方图中的单个变量开始，然后过渡到多元和聚类分析以获得更深层次的理解。您会从此过程中的每一步中了解关于数据的更多信息。

本章对随 JMP 安装的 Cereal.jmp 样本数据表进行了逐步分析。您将学习如何在“分布”、“多元”和“层次聚类”等平台中探索数据。

图 6.1 JMP 中多个分析的相互链接



目录

有趣的事实：多个分析的相互链接 159
在多个平台中探索数据 159
在“分布”平台中分析分布 159
在“多元”平台中分析模式和关系 162
在“聚类”平台中分析相似值 166

有趣的事实：多个分析的相互链接

JMP 的一项强大功能就是其多个分析的相互链接的能力。您创建的图形和报表通过数据表彼此链接。如图 6.1 所示，在数据表中选定的数据也会在三个报表窗口中选定。这种多个分析的相互链接可使您在一个窗口中选择数据并查看它在其他窗口中的情况。在使用本章示例的过程中，请保持各个 JMP 窗口处于打开状态，以便亲眼观察这些交互。

在多个平台中探索数据

哪些谷物早餐属于健康饮食？Cereal.jmp 样本数据（从多种流行的盒装谷物早餐收集的真实数据）提供了关于膳食纤维含量、卡路里和其他营养成分的统计数据。要确定最为健康的谷物早餐，您需要逐步解释直方图和描述性统计量、相关性和离群值检测、散点图和聚类分析。

在“分布”平台中分析分布

“分布”平台使用直方图、其他图形和报表演示单个变量的分布（**一元**分析）。**一元**一词就是意味着仅涉及一个变量而不是两个（二元）或许多（多元）变量。不过，您可以在一个报表中检查若干个单个变量的分布。根据该变量是类别变量（名义型或有序型）还是连续变量，变量的报表内容会发生变化。

- 对于分类变量，初始图形是一个直方图。直方图为名义型或有序型变量的每个水平显示一个直条。报表显示计数和比例。
- 对于连续变量，初始图形显示一个直方图和一个离群值箱线图。直方图为分到一组的连续变量值显示一个直条。报表显示选定的分位数和汇总统计量。

一旦了解数据如何分布，您就可以计划下一步应采取的适当分析类型。

注意：有关“分布”平台的详细信息，请参见《基本分析》。

情境

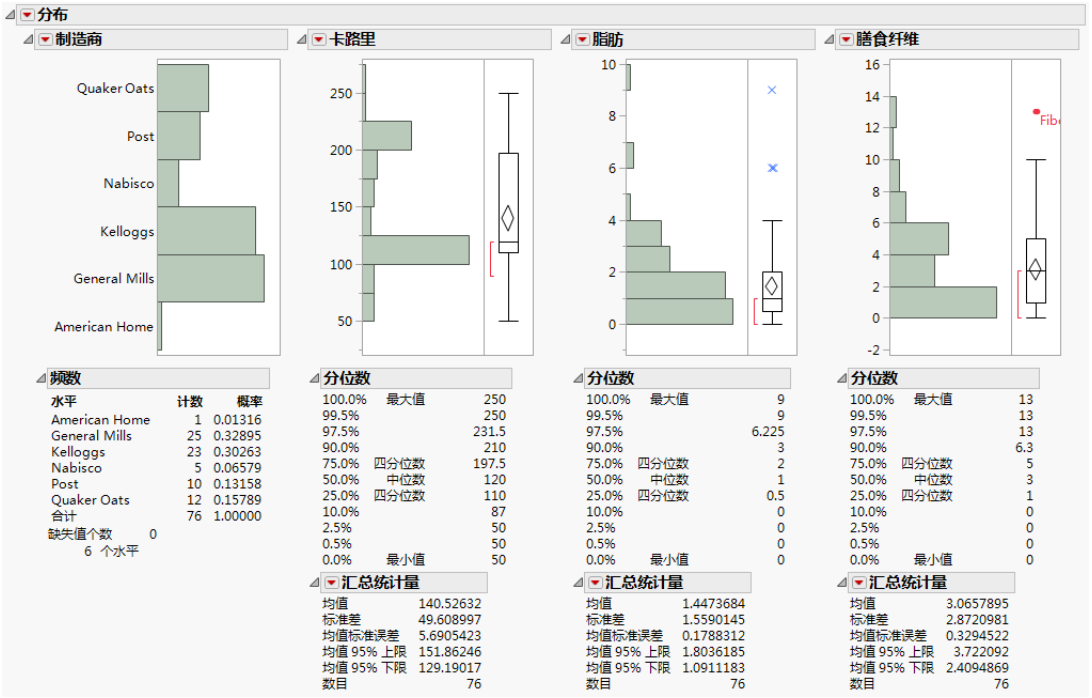
您想要查看谷物早餐的营养价值，让自己的饮食更加健康。分析谷物早餐数据的分布将揭示以下问题的答案：

- 哪些谷物早餐膳食纤维含量最高？
- 平均、最小和最大的卡路里值是多少？
- 脂肪含量的中位数是多少？
- 哪种谷物早餐脂肪含量最高？
- 数据中有离群值吗？

创建分布

1. 选择**帮助** > **样本数据库**，然后打开 Cereal.jmp。
2. 选择**分析** > **分布**。
3. 按 Ctrl 键并点击**制造商**、**卡路里**、**脂肪**和**膳食纤维**。
4. 点击 **Y，列**，然后点击**确定**。

图 6.2 “制造商”、“卡路里”、“脂肪”和 “膳食纤维”的分布



在 “膳食纤维”分布中，请注意以下几点：

– 如“膳食纤维”箱线图所示，Fiber One 和 All-Bran with Extra Fiber 含有的膳食纤维最高。这些谷物早餐属于膳食纤维含量中的离群值。

  Cereal.jmp 中包含 Fiber One 的行添加了标签。该标签在图形中的数据点旁显示谷物早餐的名称。要查看整个标签，需将最右侧的竖边框向右拖。将鼠标悬停于未加标签的数据点上方可看到 “All Bran with Extra Fiber”。

在 “脂肪”分布中，请注意以下几点：

– 将鼠标悬停于“脂肪”箱线图最顶部数据点（x 标记）的上方，可看到 100% Nat. Bran Oats & Honey 脂肪含量最高。

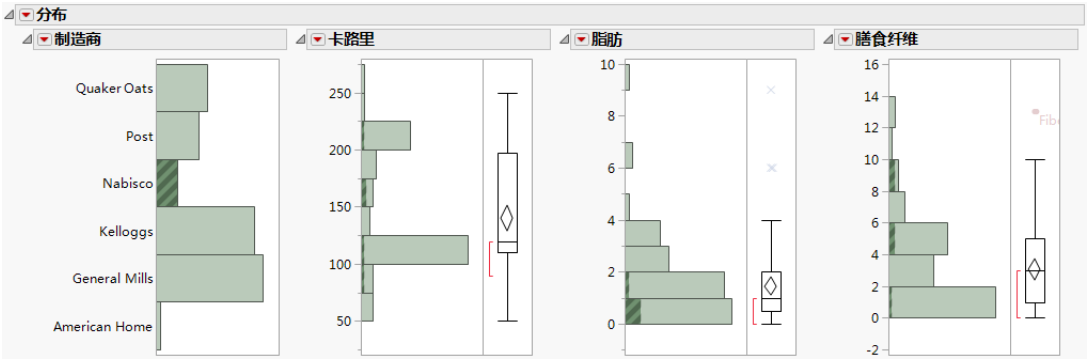
– 在“分位数”报表中，脂肪含量的中位数为 1 克。

在 “卡路里分位数”报表中，请注意以下几点：

– 卡路里最大值为 250。
– 卡路里最小值为 50。

5. 在 “制造商” 直方图中，点击 Nabisco 对应的直条。

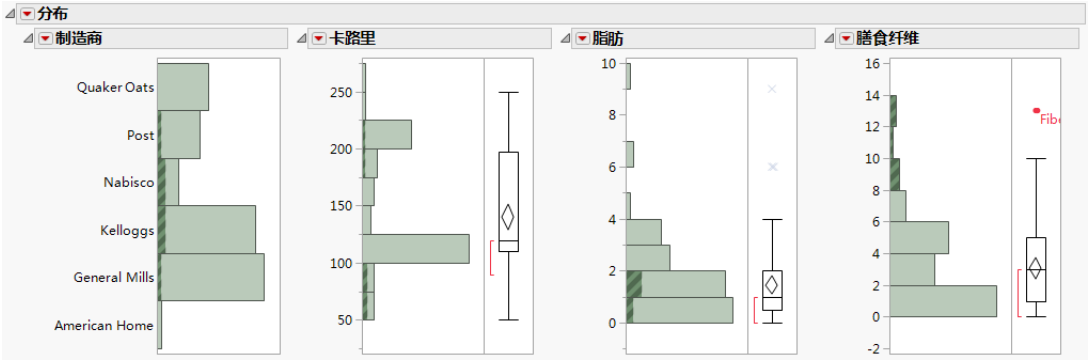
图 6.3 Nabisco 谷物早餐的分布



Nabisco 谷物早餐的 “卡路里”、“脂肪” 和 “膳食纤维” 分布在其他直方图中被突出显示。您可以查看 Nabisco 谷物早餐的 “卡路里”、“脂肪” 和 “膳食纤维” 如何分布 （相对于整体数据的 “卡路里”、“脂肪” 和 “膳食纤维” 分布）。例如， Nabisco 谷物早餐的 “脂肪” 分布看起来低于整体数据的 “脂肪” 分布。

6. 在最后一个 “膳食纤维” 直条下面点击，撤销选择所有直条。
7. 按 Shift 键并点击 “膳食纤维” 直方图中值在 8 以上的所有直方图直条。

图 6.4 膳食纤维含量高的谷物早餐

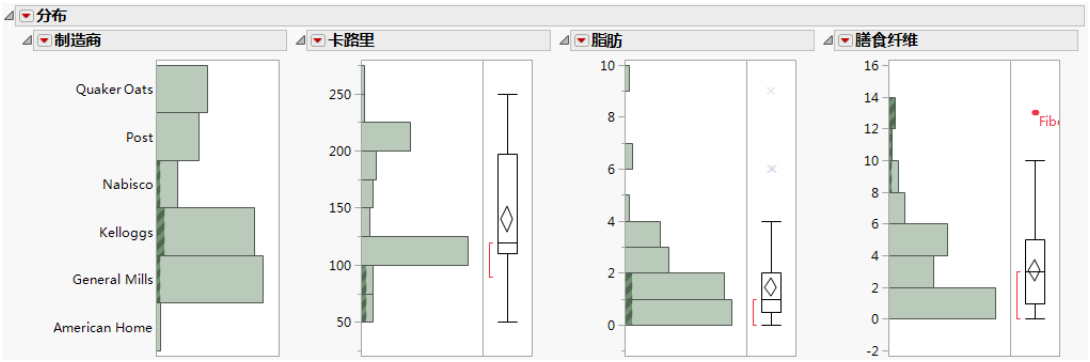


膳食纤维含量最高的谷物早餐在 “卡路里” 和 “脂肪” 直方图中被突出显示。由于各个直方图相互链接，所以要注意，有些膳食纤维含量高的谷物早餐的脂肪含量也较低。

8. 按 Ctrl 键和 Shift 键，并撤销选择 “卡路里” 直方图中值为 200 或接近 200 的两个直条。

将从直方图中去除卡路里含量高的谷物早餐。

图 6.5 膳食纤维含量高且卡路里含量低的谷物早餐



提示：让“分布”报表保持打开状态，之后在聚类分析中您会用到它。请参见第 166 页的“在“聚类”平台中分析相似值”。

解释结果

查看这些结果，您可以回答以下问题：

哪些谷物早餐膳食纤维含量最高？ “膳食纤维”箱线图显示 All-Bran with Extra Fiber 和 Fiber One 的膳食纤维含量最高。这两种谷物早餐是离群值。

平均、最小和最大的卡路里值是多少？ “卡路里”直方图显示卡路里值介于 50 到 275 之间。“卡路里分位数”显示卡路里值介于 50 到 250 之间，卡路里中位数为 120。该分布不均匀。

脂肪含量的中位数是多少？ “脂肪分位数”报表显示脂肪含量的中位数为 1 克。

哪种谷物早餐脂肪含量最高？ “脂肪”箱线图显示 100% Nat. Bran Oats & Honey 脂肪含量最高。该谷物早餐为离群值。

结论

为了增加您饮食中的膳食纤维含量，您决定尝试 All-Bran with Extra Fiber 和 Fiber One，这些谷物早餐的卡路里和脂肪含量更低。多数谷物早餐都不会大量提高饮食中脂肪的含量，但您计划避免食用脂肪含量较高的 100% Nat. Bran Oats & Honey。尽管多数谷物早餐脂肪含量相对较低，但它们的卡路里不一定低。

在“多元”平台中分析模式和关系

您已经确定哪些谷物早餐可以食用，哪些谷物早餐需要避免，接下来就是要了解谷物早餐变量彼此如何相关。“多元”平台支持您观察变量之间的模式和关系。从“多元”报表中，您可以执行以下操作：

- 使用“相关性”表总结每对响应变量之间的线性关系强度

- 使用散点图矩阵确定相依性、离群值和聚类
- 使用其他方法检查多个变量，例如偏相关性、逆相关性和配对相关性、协方差矩阵以及主成分

注意：有关“多元”平台的详细信息，请参见《多元方法》。

情境

您想要查看脂肪和卡路里等变量之间的关系。分析 “多元”平台中的谷物早餐数据揭示了关于以下问题的答案：

- 哪些变量对高度相关？
- 哪些变量对不相关？

创建“多元”报表

1. 在 Cereal.jmp 数据表中，点击“列”面板顶部的底部小三角撤销选择行。

图 6.6 撤销选择行

2. 选择**分析 > 多元方法 > 多元**。
3. 从**卡路里**一直选到**钾**，点击 **Y，列**，然后点击**确定**。

“多元”报表随即出现。该报表默认包含 “相关性”报表和散点图矩阵。“相关性”报表是一个相关系数矩阵，该矩阵总结了每对响应 (Y) 变量之间的线性关系的强度。发暗的数字表明相关度较低。

图 6.7 “相关性”报表

相关性

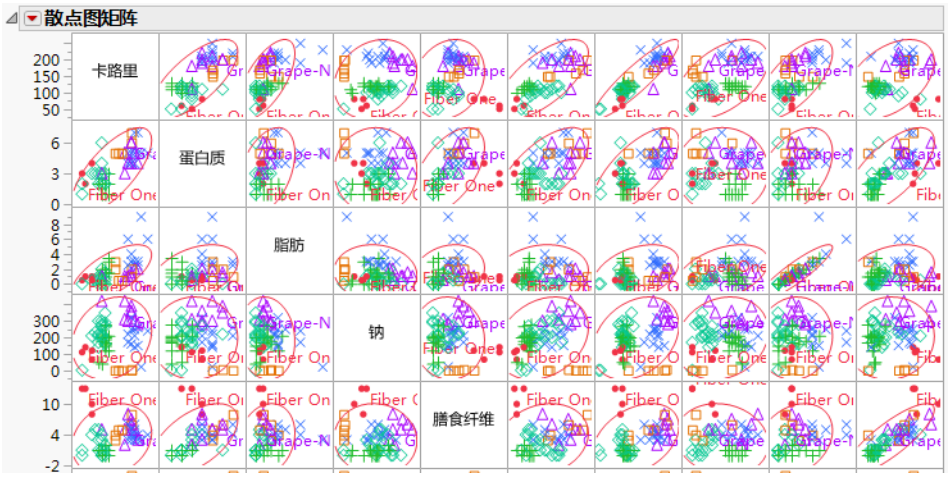
	卡路里	蛋白质	脂肪	钠	膳食纤维	复合碳水化合物	总碳水化合物	糖	脂肪热量	钾
卡路里	1.0000	0.7041	0.6460	0.1996	0.1953	0.6688	0.9076	0.5060	0.6709	0.4451
蛋白质	0.7041	1.0000	0.4080	0.0050	0.5470	0.6486	0.6937	-0.0010	0.4288	0.6782
脂肪	0.6460	0.4080	1.0000	-0.0768	0.1824	0.1037	0.3860	0.4148	0.9013	0.3420
钠	0.1996	0.0050	-0.0768	1.0000	-0.0448	0.2619	0.3066	0.1767	0.0572	0.0459
膳食纤维	0.1953	0.5470	0.1824	-0.0448	1.0000	0.1769	0.3668	-0.1264	0.2553	0.8326
复合碳水化合物	0.6688	0.6486	0.1037	0.2619	0.1769	1.0000	0.7773	-0.1601	0.1558	0.2693
总碳水化合物	0.9076	0.6937	0.3860	0.3066	0.3668	0.7773	1.0000	0.4263	0.4636	0.5375
糖	0.5060	-0.0010	0.4148	0.1767	-0.1264	-0.1601	0.4263	1.0000	0.4369	0.1166
脂肪热量	0.6709	0.4288	0.9013	0.0572	0.2553	0.1558	0.4636	0.4369	1.0000	0.3694
钾	0.4451	0.6782	0.3420	0.0459	0.8326	0.2693	0.5375	0.1166	0.3694	1.0000

请注意以下事项：

- 在“卡路里”列中，卡路里值与钠和膳食纤维之外的所有变量都高度相关。
- 在“膳食纤维”列中，膳食纤维和钾看起来高度相关。
- 在“钠”列中，钠与其他变量不高度相关。

散点图矩阵中的密度椭圆进一步演示了变量之间的关系。

图 6.8 散点图矩阵的一部分



默认情况下，每个散点图中都有一个 95% 二元正态密度椭圆。假定每对变量都服从二元正态分布，该椭圆将覆盖大约 95% 的点。若该椭圆较圆，不向对角方向延伸，则变量之间不相关。若该椭圆较窄且向对角方向延伸，则变量之间相关。

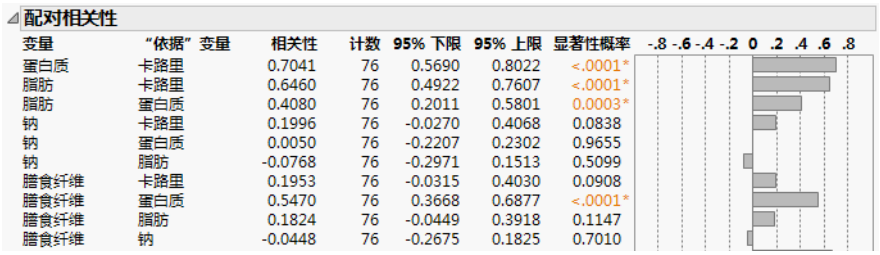
请注意以下几点：

- 在“钠”行中，椭圆较圆，该形状表明“钠”与其他变量不相关。
- 在“脂肪”行中，蓝色 x 标记（代表 Nat. Bran Oats & Honey、Cracklin’ Oat Bran 和 Banana Nut Crunch）显示在椭圆之外。该位置表明该数据是个离群值（因为谷物早餐中的脂肪含量）。

稍后您会进一步探究散点图矩阵。

4. 点击 “多元”红色小三角并选择**配对相关性**以显示 “配对相关性”报表。

图 6.9 “配对相关性”报表的一部分



配对相关性

变量	“依据”变量	相关性	计数	95% 下限	95% 上限	显著性概率
蛋白质	卡路里	0.7041	76	0.5690	0.8022	<.0001*
脂肪	卡路里	0.6460	76	0.4922	0.7607	<.0001*
脂肪	蛋白质	0.4080	76	0.2011	0.5801	0.0003*
钠	卡路里	0.1996	76	-0.0270	0.4068	0.0838
钠	蛋白质	0.0050	76	-0.2207	0.2302	0.9655
钠	脂肪	-0.0768	76	-0.2971	0.1513	0.5099
膳食纤维	卡路里	0.1953	76	-0.0315	0.4030	0.0908
膳食纤维	蛋白质	0.5470	76	0.3668	0.6877	<.0001*
膳食纤维	脂肪	0.1824	76	-0.0449	0.3918	0.1147
膳食纤维	钠	-0.0448	76	-0.2675	0.1825	0.7010

“配对相关性”报表列出每对 Y 变量的 Pearson 乘积矩相关系数。该报表还显示显著性概率并在条形图中比较相关性。

5. 要快速查看哪对变量高度相关，请在报表中右击并选中**按列排序**、**显著性概率**、**升序**复选框，然后点击**确定**。

   相关度最高的变量对显示在报表顶部。变量对的较小 p 值指示存在相关性。最显著的相关性存在于 “总碳水化合物”和 “卡路里”之间。

图 6.10 变量对的较小 p 值

配对相关性

变量	“依据”变量	相关性	计数	95% 下限	95% 上限	显著性概率
总碳水化合物	卡路里	0.9076	76	0.8576	0.9406	<.0001*
脂肪热量	脂肪	0.9013	76	0.8483	0.9365	<.0001*
钾	膳食纤维	0.8326	76	0.7474	0.8908	<.0001*
总碳水化合物	复合碳水化合物	0.7773	76	0.6691	0.8532	<.0001*
蛋白质	卡路里	0.7041	76	0.5690	0.8022	<.0001*
总碳水化合物	蛋白质	0.6937	76	0.5551	0.7949	<.0001*
钾	蛋白质	0.6782	76	0.5344	0.7838	<.0001*
脂肪热量	卡路里	0.6709	76	0.5248	0.7786	<.0001*
复合碳水化合物	卡路里	0.6688	76	0.5221	0.7771	<.0001*
复合碳水化合物	蛋白质	0.6486	76	0.4956	0.7625	<.0001*
脂肪	卡路里	0.6460	76	0.4922	0.7607	<.0001*
膳食纤维	蛋白质	0.5470	76	0.3668	0.6877	<.0001*
钾	总碳水化合物	0.5375	76	0.3551	0.6805	<.0001*
糖	卡路里	0.5060	76	0.3167	0.6566	<.0001*
脂肪热量	总碳水化合物	0.4636	76	0.2659	0.6238	<.0001*
钾	卡路里	0.4451	76	0.2442	0.6094	<.0001*
脂肪热量	糖	0.4369	76	0.2345	0.6029	<.0001*

解释结果

查看这些结果，您可以回答以下问题：

哪些变量对高度相关？ “相关性”报表和散点图矩阵显示卡路里值与钠和膳食纤维之外的所有变量高度相关。“配对相关性”报表显示 “总碳水化合物”和 “卡路里”是最为相关的变量对。

哪些变量对不相关？ “相关性”报表和散点图矩阵显示 “钠”与其他变量都不相关。

结论

您印证了之前的结论：应避免高脂肪含量的 100% Nat. Bran Oats & Honey。这再次表明：尝试 All-Bran with Extra Fiber 和 Fiber One 不失为明智的选择。这两种膳食纤维含量高的谷物早

餐还有额外的好处，卡路里、脂肪和糖份较低，而且钾含量更高。此外，您还要避免碳水化合物含量高的谷物早餐，因为它们很可能含有较高的卡路里。

在“聚类”平台中分析相似值

聚类是一种多元方法，用于将在若干变量之间共享相似值的多个观测归入一组。通过层次聚类可将层次序列（表示为层次树）中的行进行组合。具有某些特征的谷物早餐（如，膳食纤维含量高的谷物早餐）将分组到聚类中，以便您查看谷物早餐之间的相似性。

注意：有关层次聚类的详细信息，请参见《多元方法》。

情境

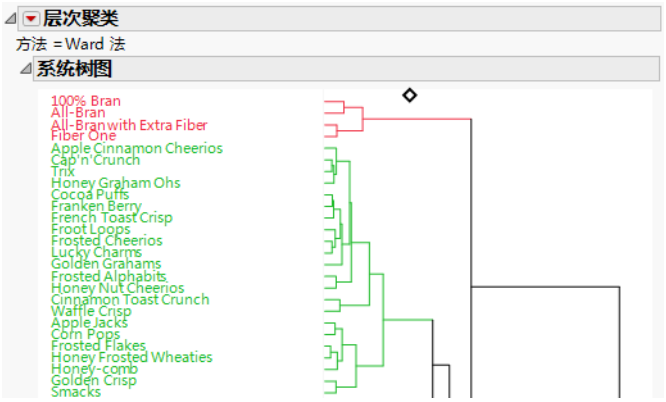
您想要了解哪些谷物早餐彼此相似，哪些不相似。分析谷物早餐数据的聚类将揭示以下问题的答案：

- 哪些谷物早餐聚类提供的营养价值很低？
- 哪些谷物早餐聚类的维生素和矿物质含量高，而糖和脂肪的含量低？
- 哪些谷物早餐聚类的膳食纤维含量高而卡路里低？

创建层次聚类图

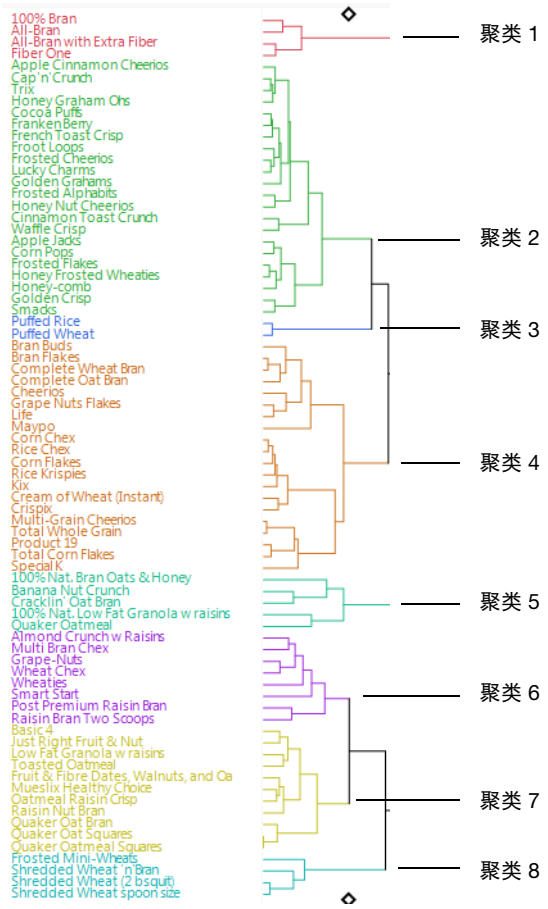
1. 在打开的 Cereal.jmp 中，选择**分析 > 聚类 > 层次聚类**。
2. 从**卡路里**一直选到**强化**，点击 **Y，列**，然后点击**确定**。
   “层次聚类”报表随即出现。聚类根据数据表行状态进行了着色。

图 6.11 “层次聚类”报表的一部分



3. 点击“层次聚类”红色小三角并选择**聚类着色**。
   各聚类根据其在树状图中的关系进行着色。

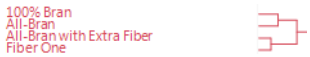
图 6.12 着色的聚类



每个聚类内的谷物早餐具有相似特征。例如，从聚类 1 中谷物早餐的名称可以判断出：谷物早餐的膳食纤维含量较高。

请注意 All-Bran with Extra Fiber 和 Fiber One 如何组合在聚类 1 中。相对于该聚类中的另外两种谷物早餐，这两种谷物早餐彼此更为相似。

图 6.13 聚类 1 中的相似谷物早餐



4. 要选择聚类 1，请点击右侧的红色水平线。

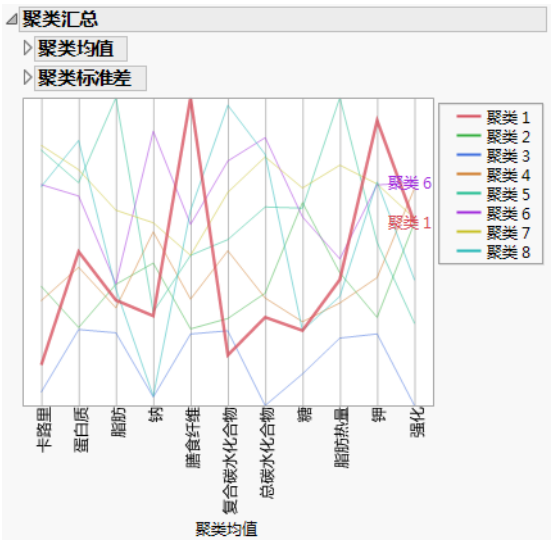
   这四种谷物早餐以红色突出显示。

图 6.14 选择聚类

5. 要查看聚类中的相似特征，点击 “层次聚类”红色小三角并选择**聚类汇总**。

   报表底部的聚类汇总图显示各个变量在各个聚类中的均值。例如，相对于其他聚类中的谷物早餐，该聚类中的谷物早餐含有更多的膳食纤维和钾。

图 6.15 聚类汇总

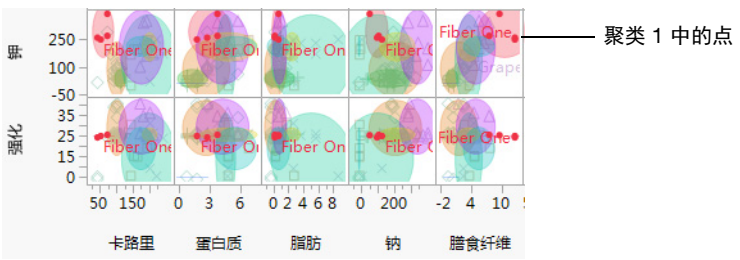


6. 点击 “层次聚类”红色小三角并选择**散点图矩阵**。

   该选项是用于在 “多元”平台中创建散点图矩阵的备用选项。

   请注意 “钾”行中的膳食纤维图。选定的谷物早餐位于图右侧 8 克到 13 克之间。这一位置表明聚类 1 中的谷物早餐的膳食纤维和钾含量都较高。

图 6.16 聚类 1 的特征



注意：若您之前创建的散点图矩阵仍处于打开状态，则也会在其中选定这些点。

解释结果

逐个点击浏览这些聚类并查看 “聚类汇总”报表，您可以发现以下特征：

- 聚类 1 谷物早餐，如 Fiber One 和 All-Bran，膳食纤维和钾的含量较高，卡路里较低。
- 聚类 2 谷物早餐，包括许多儿童喜爱的谷物早餐，糖含量高，膳食纤维、复合碳水化合物和蛋白质的含量都较低。
- 聚类 3 中的谷物早餐（Puffed Rice 和 Puffed Wheat）卡路里低，但营养价值不高。
- 聚类 4 中的谷物早餐（如 Total Corn Flakes 和 Multi-Grain Cheerios）提供的维生素和矿物质可完全满足您每日的需要。其脂肪、膳食纤维和糖的含量都较低。
- 聚类 5 中的谷物早餐蛋白质和脂肪的含量高，钠的含量低。该聚类包括 Banana Nut Crunch 和 Quaker Oatmeal 之类的谷物早餐。
- 聚类 6 中的谷物早餐脂肪含量低，钠和碳水化合物的含量高。传统谷物早餐（如 Wheaties 和 Grape-Nuts）都位于该聚类中。
- 聚类 7 中的谷物早餐卡路里高，膳食纤维含量低。许多含有干果成分的谷物早餐都位于该聚类中（Mueslix Healthy Choice、Low Fat Granola w Raisins、Oatmeal Raisin Crisp、Raisin Nut Bran 和 Just Right Fruit & Nut）。
- 聚类 8 中的谷物早餐钠和糖的含量低，复合碳水化合物、蛋白质和钾的含量高。Shredded Wheat 和 Mini-Wheat 等谷物早餐都位于该聚类中。

通过查看树状图中的结合点，您会发现每个聚类中哪些谷物早餐最为相似。

- 在聚类 1 中，Fiber One 的营养价值类似于 All-Bran with Extra Fiber。100% Bran 和 All-Bran 也十分相似。每对相似谷物早餐都由不同公司生产，所以这些谷物早餐互为竞争产品。
- 在聚类 2 中，Frosted Flakes 和 Honey Frosted Wheaties 很相似，尽管一个是玉米片，另一个是麦片。Lucky Charms 和 Frosted Cheerios 很相似。Cap’n’Crunch 和 Trix 也很相似。

结论

根据您想要摄入更多膳食纤维和更少卡路里的要求，您决定尝试聚类 1 中的谷物早餐。您需要避免聚类 3 中的谷物早餐，其中包含 puffed wheat 和 puffed rice，其营养价值不高。您还需要尝试营养价值高的聚类 4 中的谷物早餐。

第 7 章

保存和共享您的工作

保存和重新创建结果

在您根据数据生成结果之后，JMP 提供多种方式供您和他人共享工作。下面是几种您可以共享工作的方式：

- 将平台结果保存为记录、项目或 Web 报表
- 将结果、数据表和其他文件保存在项目中
- 保存脚本以在数据表中重新生成结果
- 将结果另存为交互式 HTML (.htm, html)
- 将结果另存为 PowerPoint 演示文稿 (.pptx)
- 在仪表板中共享结果

图 7.1 Web 报表的示例

目录

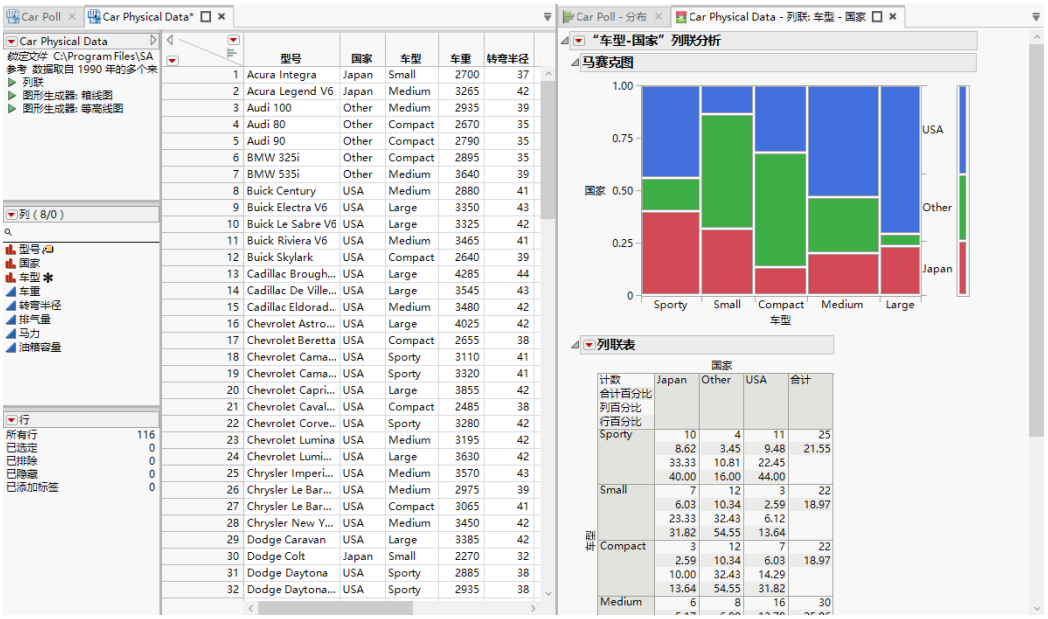
使用项目 173
创建新项目 174
在项目中打开文件 174
在项目中重新排列文件 174
保存项目 175
项目工作区 176
项目书签 177
项目内容 177
项目中最近使用的文件 178
项目日志 178
将文件移动到项目中 179
编写项目 On Open 脚本 179
创建新项目的示例 180
将平台结果保存在记录中 182
示例：创建记录 182
将分析添加到记录中 183
保存和运行脚本 184
示例：保存和运行脚本 184
关于脚本和 JSL 185
将报表另存为交互式 HTML 185
交互式 HTML 包含数据 186
示例：创建交互式 HTML 186
将报表共享为交互式 HTML 187
将报表另存为 PowerPoint 演示文稿 191
创建仪表板 192
示例：组合窗口 192
示例：创建含两个报表的仪表板 194

使用项目

使用 JMP 项目，您可以执行以下操作：

- 通过单个选项卡式 JMP 窗口更有效地探索您的数据
- 快速保存和重新打开一组相关的文件和报表
- 通过将您的表和脚本嵌入独立的项目文件中来轻松地共享您的工作

图 7.2 带数据表和报表的项目文件



本节包括下列信息：

- “创建新项目”
- “在项目中打开文件”
- “在项目中重新排列文件”
- “保存项目”
- “项目工作区”
- “项目书签”
- “项目内容”
- “项目中最近使用的文件”
- “项目日志”
- “将文件移动到项目中”

- “编写项目 On Open 脚本”
- “创建新项目的示例”

创建新项目

要创建新的空 JMP 项目，选择**文件 > 新建 > 项目** (Windows) 或**文件 > 新建 > 新建项目** (macOS)。

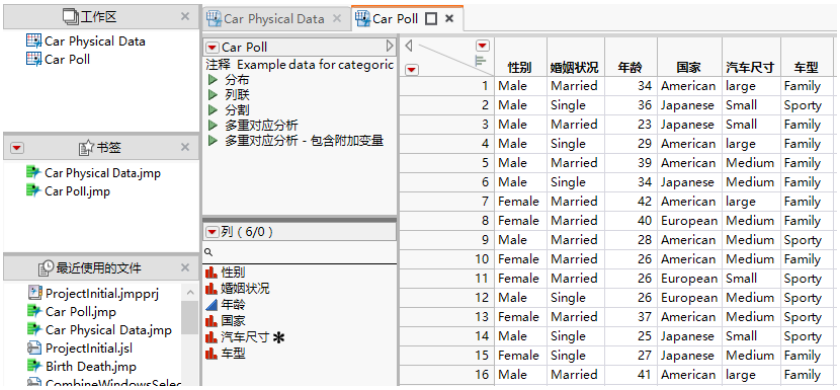
在项目中打开文件

在 JMP 项目中，您可以打开数据表，然后对数据运行分析。每个数据表和分析报表会在新的选项卡中打开。

1. 从项目窗口中，选择**文件 > 打开**，然后导航至您想打开的数据表。
2. 从数据表中，运行分析。

若您的计算机上数据表发生更新，则项目中任何关联的报表会在您重新打开项目时更新。

图 7.3 初始项目



提示：您可以在项目中使用 JMP 键盘快捷方式，例如 Ctrl+S 可以保存文件或 Ctrl+W 可以关闭活动的窗口窗格。有关完整列表，请选择**帮助 > 快速参考卡**。

在项目中重新排列文件

在 JMP 项目中，数据表和报表显示在各自的选项卡中。通过将选项卡拖到停放区中可以重新排列它们。

提示：要撤销或恢复停放项，选择**项目 > 撤销布局**或**项目 > 恢复布局**。要将所有表和报表返回到各自的选项卡，选择**项目 > 重置布局**。

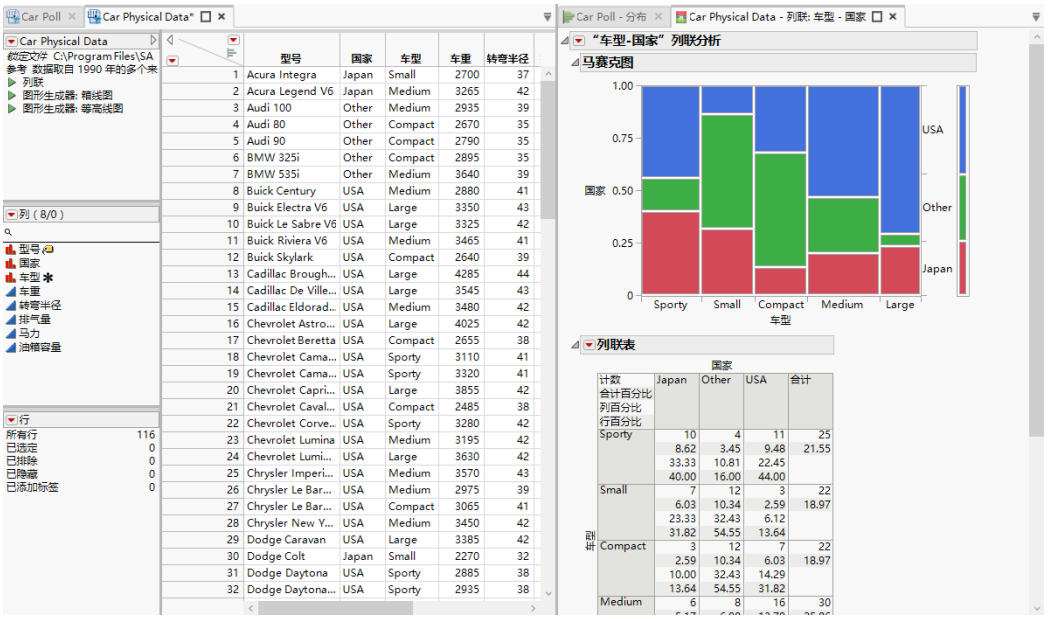
重新排列项目文件的示例

1. 选择**文件 > 新建 > 项目** (Windows) 或**文件 > 新建 > 新建项目** (macOS)。
2. 选择**帮助 > 样本数据库**，然后打开 Car Physical Data.jmp 和 Car Poll.jmp。
3. 在 Car Poll.jmp 数据表中，运行**分布**脚本。
4. 在 Car Physical Data.jmp 数据表中，运行**列联**脚本。
5. 将 Car Poll - **分布**报表选项卡向右拖动并放置在**停放右侧**区中。

   “分布”报表窗格显示在项目窗口的右侧。当您在选项卡之间进行切换时，该报表是停放状态并保持可见。
6. 将 Car Physical Data - **列联选项卡**向 Car Poll - **分布**报表的中间拖动并放置在**“停放”选项卡**区中。

提示：您可以调整数据表窗口的大小以完全显示两个报表。

图 7.4 报表和数据表分组为选项卡



“分布”和“列联”报表现在以选项卡形式显示在一起，两个数据表也是如此。

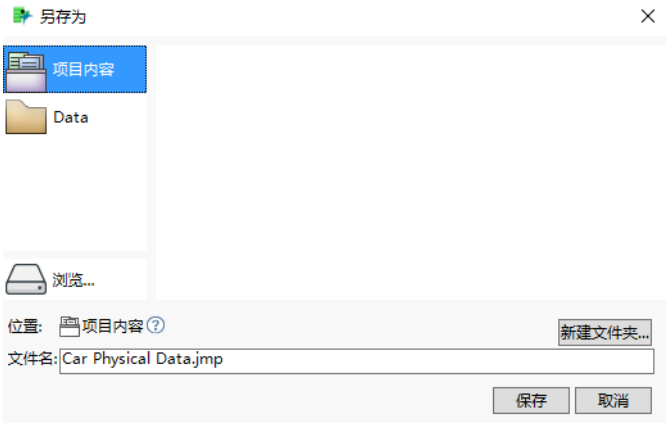
保存项目

若您想共享、分发或存档您的项目，通过将所有数据表和脚本保存至项目内容以创建一个独立的项目。这样会将所有项目文件和文件夹嵌入到您可以与其他 JMP 用户共享的单个项目文件。

1. （可选）要将项目另存为一个独立的项目，通过选择**文件** > **另存为**并选择**项目内容**以将每个数据表和脚本保存在项目中。您可以点击**新建文件夹**在项目内容中创建一个文件夹。

注意：您不需要保存报表，因为它们会自动保存至项目。

图 7.5 将数据表保存至项目内容



2. 选择**文件 > 保存项目**以保存项目文件。

图 7.6 保存项目文件

项目工作区

在 JMP 项目中，您可以在“工作区”工具窗格中查看所有当前打开的文件。报表显示在相应的数据表下方。活动的数据表显示为粗体。

提示：要隐藏或显示“工作区”工具窗格，选择**项目 > 显示工作区**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

在“工作区”工具窗格中，您可以执行以下操作：

- 双击文件使其成为活动状态。
- 右击文件以将其关闭，隐藏（删除选项卡并灰显文件名）、添加书签以及进行其他操作。

项目书签

在 JMP 项目中，您可以在“书签”工具窗格中为文件或文件夹创建快捷方式。您还可以通过创建书签组以组织添加书签的文件和文件夹。

提示：要隐藏或显示“书签”工具窗格，选择**项目 > 显示书签**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

在项目中对打开的文件添加书签

- 右击选项卡并选择**添加书签**。
- 要对所有打开的文件添加书签，右击选项卡并选择**全部添加书签**。

对计算机上的文件或文件夹添加书签

将计算机中的文件或文件夹拖到“书签”工具窗格，或点击“书签”红色小三角菜单并选择**添加文件**或**添加文件夹**。

注意：若您从计算机上添加书签的文件夹中添加或删除文件，则项目中文件夹内容会自动更新。

打开添加了书签的文件

在“书签”工具窗格中，双击文件。

移除书签

在“书签”工具窗格中，右击文件并选择**移除书签**。

创建书签组

1. 点击“书签”红色小三角菜单并选择**新建组**。
2. 命名组并点击**确定**。
3. 将新的或现有的书签拖到组中，或在“书签”工具窗格中，右击组并选择**添加文件**或**添加文件夹**。

项目内容

在一个独立的 JMP 项目中，您保存至项目内容的任何文件会显示在“内容”工具窗格中。

提示：要隐藏或显示“内容”工具窗格，选择**项目 > 显示内容**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

从项目内容打开文件

在“内容”工具窗格中，双击文件。

在项目内容中创建文件夹

1. 点击“内容”红色小三角菜单并选择**新建文件夹**。
2. 命名组并点击**确定**。

将文件移入文件夹

在“内容”工具窗格中，将文件拖到文件夹中。

将计算机中的文件或文件夹复制到项目内容中

1. 点击“内容”红色小三角菜单并选择**将文件复制到项目中**或**将文件夹复制到项目中**。
2. 导航到文件并点击**打开**，或导航到文件夹并点击**选择**。

在项目内容中重命名文件

在“内容”工具窗格中，右击文件并选择**重命名**。

在项目内容中删除文件

在“内容”工具窗格中，右击文件并选择**删除**。

项目中最近使用的文件

在 JMP 项目中，您可以从“最近使用的文件”工具窗格中打开文件。

提示：要隐藏或显示“最近使用的文件”工具窗格，选择**项目 > 显示最近使用的文件**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

打开一个文件

在“最近使用的文件”工具窗格中，双击文件或将其拖到项目中。

注意：保存至项目内容的打开文件不会显示在“最近使用的文件”工具窗格中。

项目日志

在 JMP 项目中，您可以在“日志”工具窗格中查看日志消息。

提示：

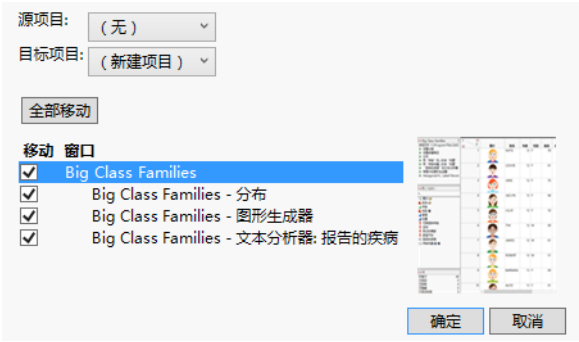
- 要隐藏或显示“日志”工具窗格，选择**项目 > 显示日志**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。
- 默认情况下，项目中出现的日志消息仅显示在项目日志中，不显示在主 JMP 日志中。要更改该设置，选择**文件 > 首选项 > 项目**，然后将**使用项目日志窗格**更新为**条件打开**或**从不**。

将文件移动到项目中

在 JMP 中，您可以将文件移入项目，移出项目或在项目之间移动。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class Families.jmp。
2. 运行 “分布”、“文本分析器”和 “图形生成器”脚本。
3. 从任意窗口中，选择**窗口 > 移至项目 / 从项目移动**。
4. 将 “源项目”保留为 （**无**）以查看在任何项目中均未打开的文件。
5. 将 “目标项目”保留为 （**新建项目**）以将选定文件移动到新创建的项目中。
6. 选中 Big Class Families.jmp 旁边的复选框。与数据表关联的图形也被一起选中。

图 7.7 将窗口移至项目 / 从项目移动窗口



7. 点击**确定**。使用选定表和报表创建了新项目。

编写项目 On Open 脚本

在“项目 On Open 脚本”框中，您可以添加 JSL 脚本，每次特定项目打开时将运行该脚本。例如，您可以创建一个脚本，它创建或修改报表，为用户提示信息，或将使用信息写入日志窗口。

1. 选择**项目 > 项目设置**。
2. 粘贴 JSL 脚本并点击**确定**。

提示：

- 请参见《Scripting Guide》，它解释了当**任何**项目（而非特定项目）打开时如何创建启动脚本。

- 要将现有项目用作新项目的模板，请在**文件 > 首选项 > 项目**下将现有项目指定为新项目模板。

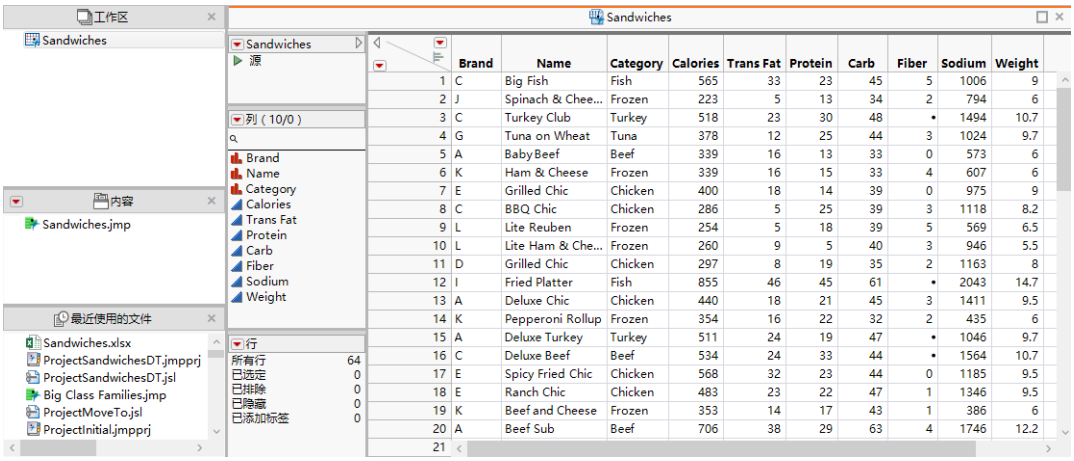
创建新项目的示例

在该示例中，您创建项目，导入数据，生成分析并在项目窗口中停放报表，创建子集表，然后保存、关闭和重新打开项目。

1. 选择**文件 > 新建 > 项目** (Windows) 或**文件 > 新建 > 新建项目** (macOS)。
2. 从项目窗口中，选择**文件 > 打开**。
3. 打开 sandwiches.xlsx 文件，默认情况下它位于此处：
   C:/Program Files/SAS/JMP/16/Samples/Import Data

   提示：在底部，您可能需要将**所有 JMP 文件**更改为 Excel 文件。
4. 点击**导入**。
5. 选择**文件 > 保存**。
6. 确保已选择**项目内容**。
7. 将文件名更改为 Sandwiches.jmp 并点击**保存**。

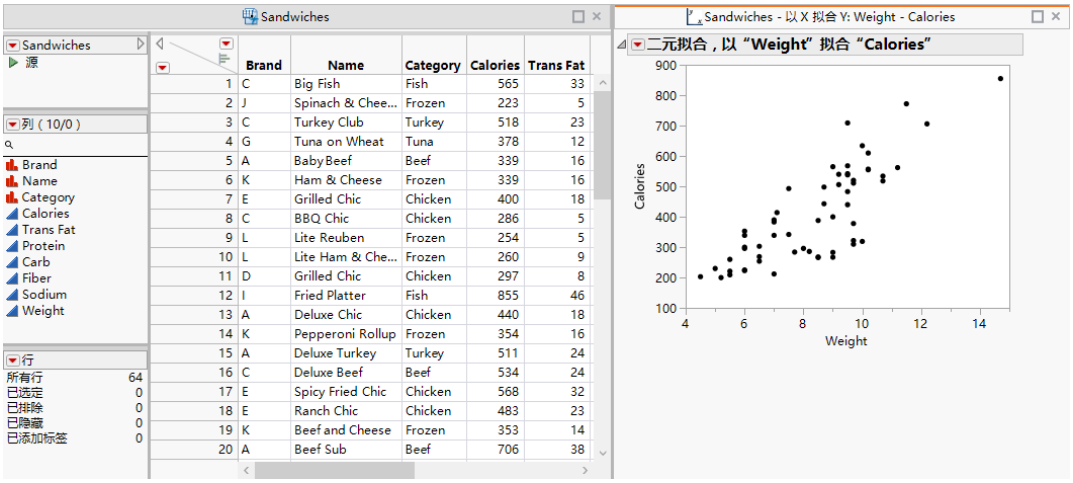
图 7.8 导入了三明治数据的项目



8. 选择**分析 > 以 X 拟合 Y**。
9. 选择 Calories 并点击 **Y，响应**。
10. 选择 Weight 并点击 **X，因子**。
11. 点击**确定**。
12. 将 Sandwiches - 以 X 拟合 Y 选项卡向右拖动并放置在**停放右侧**区中。

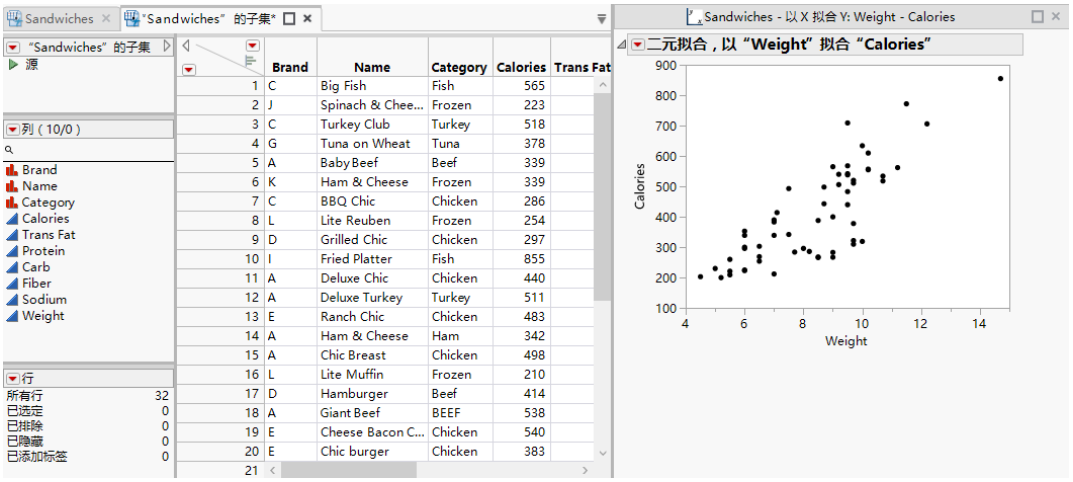
提示：要显示整个报表，您可以将数据表和报表之间的线向左拖动。

图 7.9 停放在右侧的 “以 X 拟合 Y”（二元）报表



13. 选择**表 > 子集**。
14. 在 “行”下方，选择**随机：抽样率 0.5**。
15. 点击**确定**。

图 7.10 包含未保存的子集表的项目



16. 选择**文件 > 保存项目**。
17. 导航至您想保存项目的文件夹，命名项目文件，然后点击**保存**。
18. 关闭项目。

19. 选择**文件** > **打开**并打开您的项目文件。

请注意，您未保存的子集表已自动保存至项目文件。

将平台结果保存在记录中

通过为报表窗口创建记录，可以保存平台报表供以后查看。记录是报表窗口的副本。您可以编辑记录或将其他报表追加到现有记录中。记录不与数据表连接。通过使用记录，您可以轻松地将多个报表窗口中的结果保存在一个报表窗口中供您和他人共享。

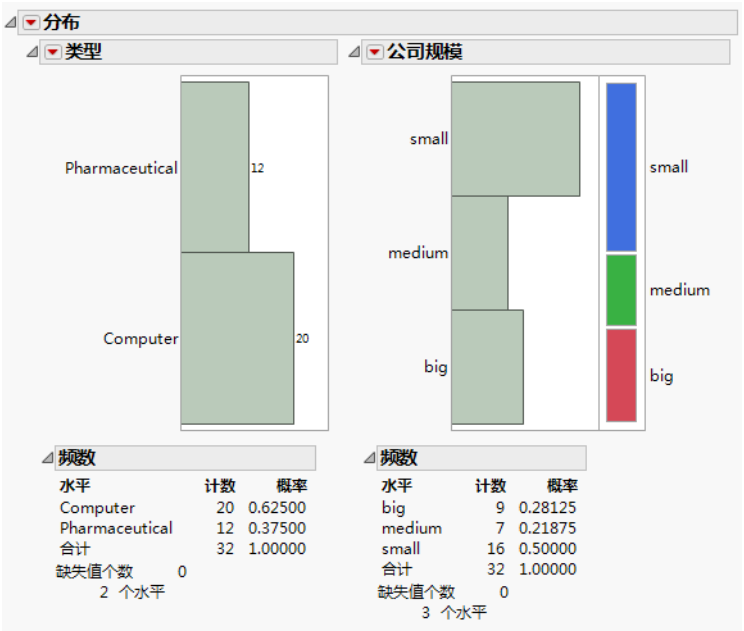
本节包括下列信息：

- “示例：创建记录”
- “将分析添加到记录中”

示例：创建记录

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**类型**和**公司规模**，并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “类型”红色小三角并选择**直方图选项** > **显示计数**。
6. 点击 “公司规模”红色小三角并选择**马赛克图**。
7. 选择**编辑** > **记录**以便记录这些结果。结果被复制到记录窗口中。

图 7.11 分布结果记录



记录中的结果不与数据表连接。在“类型”条形图中，若您点击 Computer 直条，数据表中没有行被选定。

因为记录是结果的副本，大多数红色小三角菜单都不存在。对于您添加到记录中的每个新报表，记录都有一个红色小三角菜单。该菜单包含两个选项：

在新窗口中重新运行 若您有用于创建原始报表的原始数据表，该选项将重新运行分析。结果是一个新的报表窗口。

编辑脚本 该选项将打开一个脚本窗口，窗口中包含 JSL 脚本以重新创建分析。JSL 是更高级的主题，将在 《Scripting Guide》和 《JSL Syntax Reference》中介绍。

将分析添加到记录中

若您执行另外的分析，可以将分析结果添加到现有记录中。

1. 在记录打开的情况下，选择**分析 > 分布**。
2. 选择**人均利润**并点击 **Y，列**。
3. 点击**确定**。
4. 选择**编辑 > 记录**。结果追加到记录的底部。

保存和运行脚本

JMP 中的多数平台选项都是可脚本化的，这表示您执行的多数操作都可以保存为 JMP 脚本语言 (JSL) 脚本。您随时可以使用脚本重新生成操作或结果。

本节包括下列信息：

- “示例：保存和运行脚本”
- “关于脚本和 JSL”

示例：保存和运行脚本

创建报表

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**类型**和**人均利润**，并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “类型” 红色小三角并选择以下选项：
   - **直方图选项** > **显示计数**
   - **置信区间** > 0.95
6. 点击 “人均利润” 红色小三角并选择以下选项：
   - **离群值箱线图**，用于删除离群值箱线图
   - CDF 图
7. 点击 “分布” 红色小三角并选择**堆叠**。

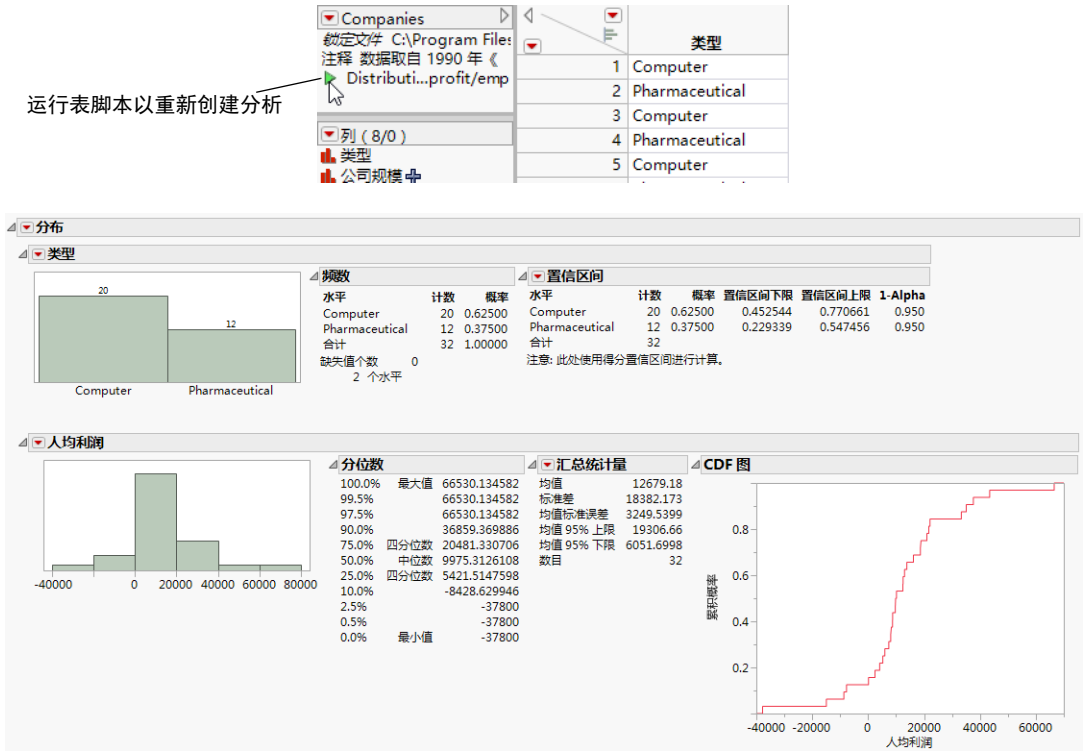
将脚本保存到数据表中并运行脚本

1. 要保存该分析，点击 “分布” 红色小三角并选择**保存脚本** > **至数据表**。新脚本将显示在 “表” 面板中。

图 7.12 分布脚本

2. 关闭 “分布” 报表窗口。
3. 要重新创建分析，点击 “分布” 脚本旁边的绿色小三角。

图 7.13 运行分布脚本



提示：右击表脚本以查看更多选项。

关于脚本和 JSL

您在本节中保存的脚本包含 JMP 脚本语言 (JSL) 命令。JSL 是更高级的主题，将在《Scripting Guide》和《JSL Syntax Reference》中介绍。

将报表另存为交互式 HTML

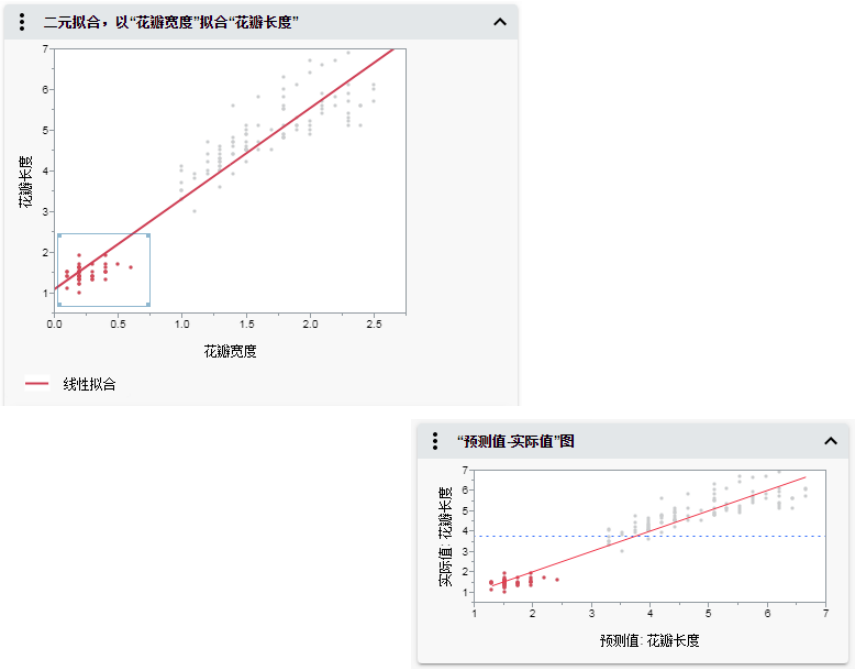
交互式 HTML 支持 JMP 用户共享包含动态图形的报表，这样即使非 JMP 用户也可探索数据。JMP 报表另存为 HTML 5 格式的网页，您可以将其通过电子邮件发送给用户，或在网站上发布。然后用户可以如同在 JMP 中一样探索数据。

交互式 HTML 提供 JMP 中的一部分功能：

- 探索交互式图形功能，如选择直方图直条和查看数据值。
- 通过刷亮方式查看数据。

- 显示或隐藏报表区域。
- 将鼠标悬停在报表上方以获取工具提示。
- 增加标记大小。

图 7.14 在交互式 HTML 中刷亮数据



您对图形（如有序变量、水平直方图、背景颜色和着色数据点）所做的许多更改都保存在网页中。保存内容时关闭的图形和表将在网页中保持关闭状态，直到用户将其打开。

交互式 HTML 包含数据

在 JMP 中将报表另存为交互式 HTML 后，数据将嵌入在 HTML 中。内容未经加密，因为 Web 浏览器无法读取加密数据。要避免共享敏感数据，请将结果另存为非交互式网页。（选择**文件 > 导出 > 数据交互式 HTML 文件**。）

示例：创建交互式 HTML

创建报表

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 分布**。
3. 选择**身高**并点击 **Y，列**。

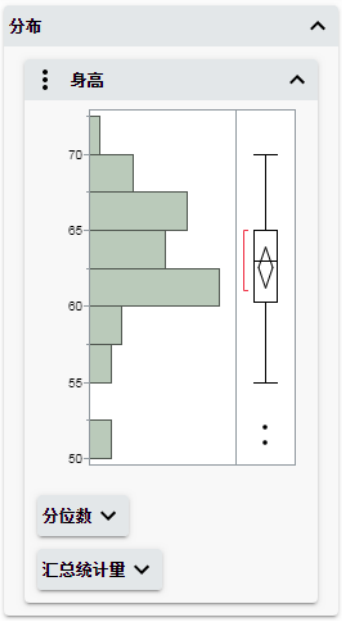
4. 点击**确定**。

另存为交互式 HTML

1. (Windows) 选择**文件 > 导出**，选择**数据交互式 HTML**，然后点击**确定**。
2. (macOS) 选择**文件 > 导出**，选择**数据交互式 HTML**，然后点击**下一步**。
3. 在 “导出”窗口中，选择**保存后打开文件** （若尚未选择该选项）。
4. 命名并保存文件。

   输出将显示在您默认的浏览器中。

图 7.15 交互式 HTML 输出



有关了解交互式 HTML 输出的信息，请访问 https://www.jmp.com/interactive。

将报表共享为交互式 HTML

交互式 HTML 支持您共享包含动态图形的 JMP 报表，这样即便是非 JMP 用户也能探索数据。JMP 报表另存为可以与其他用户共享（例如，在共享网络驱动器上，通过电子邮件或在网站上）的交互式网页。然后用户可以如同在 JMP 中一样探索数据。

交互式 HTML 包含数据

当您将报表导出或发布为交互式 HTML 时，您的数据嵌入在 HTML 中。内容未经加密，因为 Web 浏览器无法读取加密数据。为了避免共享敏感数据，通过改为选择**文件 > 导出 > HTML** 将结果导出为非交互式网页。

以交互方式支持哪些功能

交互式 HTML 提供 JMP 中的一部分功能：

- 若完全支持您的报表中的功能，网页创建时没有警告。
- 若您的报表包含不支持的功能，您会在导出或发布窗口中看到一条消息指出部分实现交互式 HTML。详细信息，请参见 JMP 日志（**视图 > 日志**）。
- 部分实现或不支持的功能在网页中会静态显示。若您将鼠标悬停在网页中不支持的功能上，工具提示会显示该功能还无法交互。

有关了解交互式 HTML 输出的信息，请访问 https://www.jmp.com/interactive。

将单个报表导出为交互式 HTML

要将单个 JMP 报表导出为交互式 HTML，使用“导出”作为“数据交互式 HTML”选项创建单个网页。

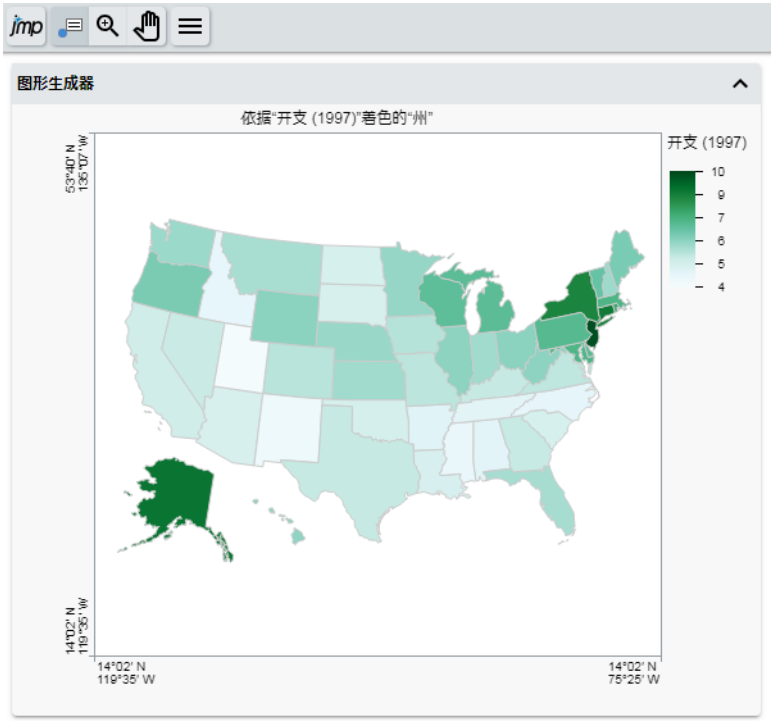
1. 在 JMP 中，创建该报表并使它成为活动窗口。

   注意：若报表包含“本地数据过滤器”，则它在 Web 报表中是静态的，用户无法更改选择。要使“本地数据过滤器”可交互，取消选择**包括**模式。（在“图形生成器”中，要查看**包括**模式，从“本地数据过滤器”红色小三角菜单中选择**显示模式**。）

2. 选择**文件 > 导出**，选择**数据交互式 HTML**，然后点击**下一步**。
3. 为文件命名。
4. （可选）要在导出 HTML 文件后在默认浏览器中打开它，选择**保存后打开文件**。
5. 点击**保存**。

   输出内容保存在选定文件夹中。

图 7.16 单个交互式 HTML 报表的网页



将多个报表发布为交互式 HTML

要将多个 JMP 报表发布为交互式 HTML，使用“发布至文件”选项，它会创建包含报表的索引页面。

提示：当您发布多个 JMP 报表时，请指定保存索引页面和报表文件的位置。您可以选择共享网络文件夹并将该位置提供给其他用户，或选择您的计算机上的文件夹并在共享前压缩文件。若您共享的对象不是 JMP 用户，这将会特别有帮助。

1. 在 JMP 中，创建报表。

   注意：若报表包含“本地数据过滤器”，则它在 Web 报表中是静态的，用户无法更改选择。要使“本地数据过滤器”可交互，取消选择**包括**模式。（在“图形生成器”中，要查看**包括**模式，从“本地数据过滤器”红色小三角菜单中选择**显示模式**。）

2. 从报表窗口中，选择**文件 > 发布 > 发布至文件**。
3. 选择要发布的报表。
4. （可选）更改父级文件夹所在的位置或更改将包含报表的子文件夹的名称。
5. 点击**下一步**。

6. 输入索引页面的标题。您还可以更新报表标题。
7. （可选）更改其他选项。请参见 “交互式 HTML 报表选项”。
8. 点击**发布**。

图 7.17 多个交互式 HTML 报表的索引页面

9. 点击缩略图以打开报表。

有关使用交互式报表的详细信息，请从 HTML 报表中点击 ☰ > **帮助**。这将打开帮助，网址：https://www.jmp.com/interactive。

交互式 HTML 报表选项

标题 为索引页面 （仅多个报表）或报表添加标题。

说明 （可选）将说明添加至索引页面或报表。说明将显示在标题下方。

定制 （仅为多个报表显示）更改网页的外观。您可以更改样式、主题、字体、徽标以及是否显示日期或时间。请参见 “定制索引页面选项”。

发布数据 为交互式报表选择该选项。若您取消选择该选项，报表是静态的。

> **注意:** 要避免共享敏感数据，请将结果另存为非交互式网页。（选择**文件 > 导出 > HTML**。）

添加图像 将图像添加至网页底部。

打开发布的 Web 报表 您点击 “发布” 后，在浏览器中打开网页。

运行后关闭报表 您发布 Web 报表后，关闭 JMP 中的报表。

“删除”图标 （仅为多个报表显示）删除报表。

“箭头”图标 （仅为多个报表显示）更改报表的顺序。

定制索引页面选项

样式格式 确定报表的布局。

大列表 在具有大缩略图的列中显示报表。

小列表 在具有小缩略图的列中显示报表。

网格 在行中显示报表。

定制 CSS 允许您指定一个 CSS 文件来格式化网页。将该 CSS 文件复制到名为 _css 的子文件夹中。指向 CSS 文件的链接是相对的，以便您可以将报表和支持文件发送到另一用户并维护 CSS 格式。

颜色主题 指定网页、标题和边框的颜色。默认网页有一个白色背景、橙色标题和蓝色边框。

更改字体 更改报表中应用的字体。

更改徽标 指定要与报表一起显示的图像。点击图像旁边的向上或向下箭头可以将它移到报表之上或之下。

显示日期 / 时间 显示或隐藏生成网页的日期和时间。

将报表另存为 PowerPoint 演示文稿

通过将 JMP 结果另存为 PowerPoint 演示文稿 (.pptx) 创建一个演示文稿。另存为 .pptx 文件后，在 PowerPoint 中重新排列 JMP 内容并编辑文本。JMP 报表各部分将分别导出至 PowerPoint。

- 报表标题导出为可编辑文本框。
- 图形导出为图像。某些图形元素（如图例）导出为单独的图像。图像会调整大小以适合 PowerPoint 中的幻灯片的尺寸。

使用选择工具选择您想在演示文稿中保存的选择内容。在 PowerPoint 中打开文件后立即删除不需要的内容。

注意：在 Windows 系统中，要打开在 JMP 中创建的 pptx 文件，PowerPoint 要求的版本最低为 PowerPoint 2007。在 macOS 上，PowerPoint 要求的版本最低为 PowerPoint 2011。

1. 在 JMP 中，创建报表。
2. 选择**文件 > 导出**，选择 Microsoft PowerPoint，然后点击**下一步**。
3. 从列表中选择图形文件格式。

   在 Windows 系统中， EMF 是默认格式。在 macOS 上， PDF 是默认格式。

4. 命名并保存文件。（在 macOS 中，命名文件并点击**导出**。）

   该文件将在 Microsoft PowerPoint 中打开，因为默认选定**保存后打开文件**。

注意：在 macOS 上不支持 Windows 上生成的本地 EMF 图形。在 Windows 上也不支持 macOS 上生成的本地 PDF 图形。对于跨平台兼容性，通过选择**文件 > 首选项 > 常规**可更改默认图形文件格式。然后，将 **PowerPoint 图像格式**更改为 PNG 或 JPEG。

创建仪表板

仪表板是一种可视工具，该工具可支持您定期运行和呈现报表。您可以在仪表板上显示报表、数据过滤器、选择内容过滤器、数据表和图形。当您打开仪表板时，仪表板上显示的内容会随之更新。

本节包括下列信息：

- “示例：组合窗口”
- “示例：创建含两个报表的仪表板”

示例：组合窗口

您可以通过在 JMP 中合并若干打开的窗口来快速创建仪表板。组合窗口提供了一些选项，不仅支持查看汇总的统计量，还包含一个选择内容过滤器。

1. 选择**帮助 > 样本数据库**，然后打开 Birth Death.jmp。
2. 运行 “分布”脚本和 “二元”表脚本。
3. 从其中一个报表窗口中，选择**窗口 > 组合窗口**。

   “组合窗口”窗口随即出现。

   提示：在 Windows 系统中，您还可以从 JMP 窗口右下角的 “排列菜单”选项中选择 “组合窗口”。

4. 选择**汇总报表视图**显示图形并省略统计报表
5. 在 “组合”列中，选择 Birth Death - 二元 : 出生率 - 死亡率和 Birth Death - 分布。
6. 在 “过滤依据”列中，选择 Birth Death - 分布。

图 7.18 “组合窗口”选项

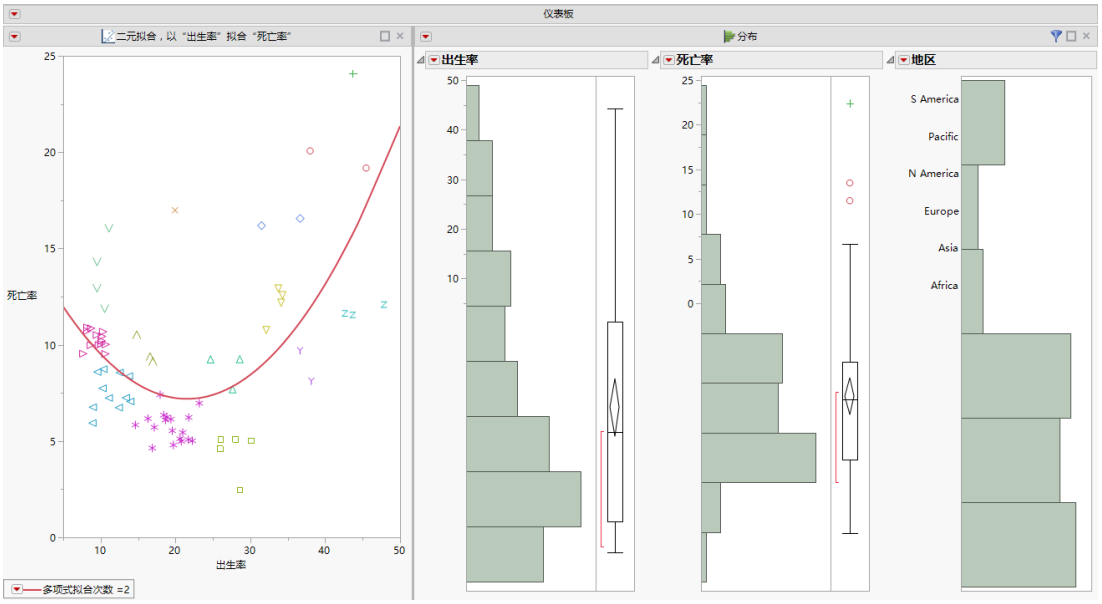
7. 点击**确定**。

   两个报表组合到一个窗口中。请注意“分布”报表顶部的过滤器图标 。选择其中一个直方图中的某个直条后，“二元”图形中的对应数据会被选定。

注意：

- 要在 Windows 上组合报表，您还可以从 JMP 窗口右下角的“排列菜单”选项中选择“组合窗口”。
- 在“组合窗口”窗口中，选择**汇总报表视图**仅查看报表中的图形并省略统计学内容。

图 7.19 组合窗口



示例：创建含两个报表的仪表板

假定您创建了两个报表，并且想要在第二天针对更新的数据集再次运行这些报表。本例显示如何在仪表板生成器中基于报表创建仪表板。

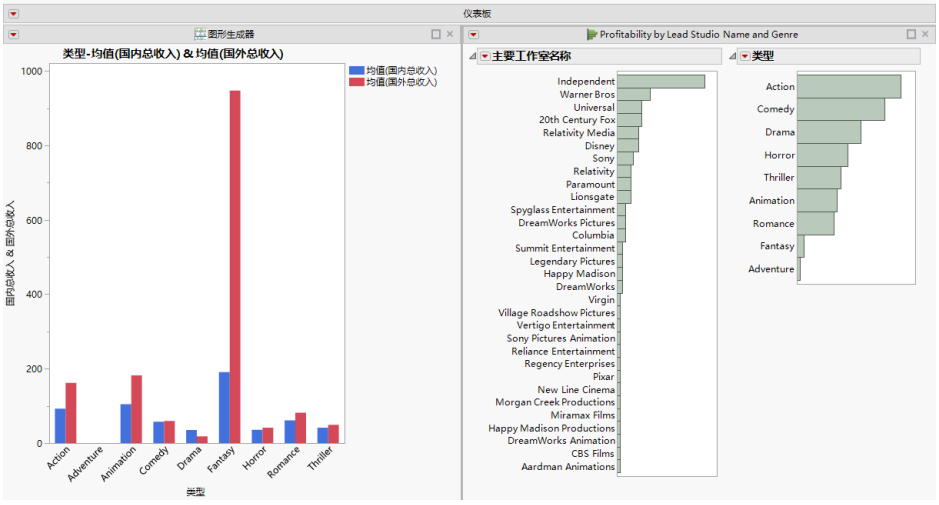
1. 选择**帮助 > 样本数据库**，然后打开 Hollywood Movies.jmp。
2. 运行名为 “分布 : 主要工作室 - 利润和类型 - 利润” 和 “图形生成器 : 类型 - 全球和国内总收入” 的表脚本。
3. 从任意窗口中，选择**文件 > 新建 > 仪表板**。

   随即出现常见布局的模板。
4. 选择 **2x1 仪表板**模板。

   工作区中随即出现可容纳两个报表的框区。
5. 在 “报表” 列表中，双击报表缩略图将其放在仪表板上。
6. 点击 “仪表板生成器” 红色小三角并选择**预览模式**。

   仪表板预览随即出现。请注意，各图形不仅彼此链接还与数据表链接。它们也具有与 “分布” 平台和 “图形生成器” 平台相同的红色小三角选项。
7. 点击**关闭预览**。

图 7.20 包含两个报表的仪表板



有关创建仪表板的详细信息，请参见《使用 JMP》。

第 8 章

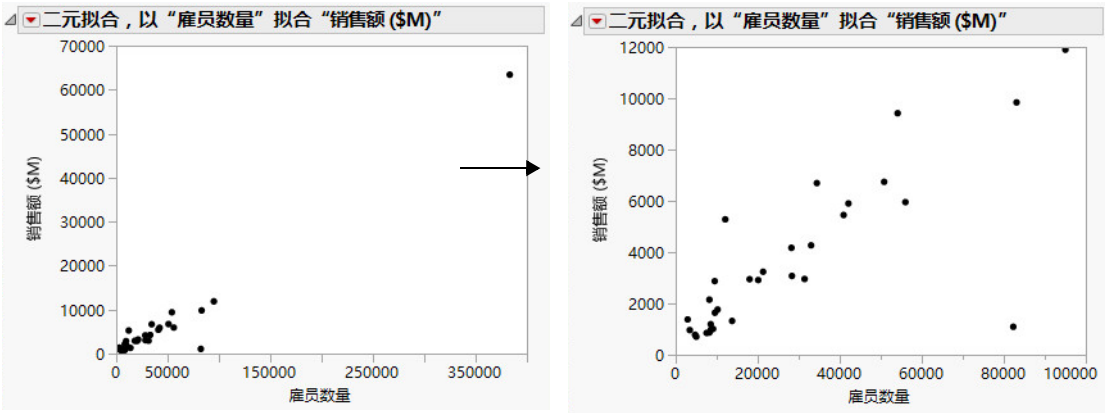
特殊功能

自动分析更新和 SAS 集成

使用 JMP 中的某些特殊功能，您可以：

- 自动更新分析或图形
- 定制平台结果
- 与 SAS 集成以使用高级分析功能

图 8.1 特殊功能示例




```
DATA Candy_Bars; INPUT  Calories Total_fat_g Carbohydrate_g Protein_g; Lines;
310  20  28  6
230  12  27  4
220  12  24  3
170  8  21  3
200  2.5  43  1
260  16  26  5
190  1.5  42  2
190  11  21  2
230  12  28  3
;
RUN;

PROC GLM  DATA=Candy_Bars ALPHA=0.05;
MODEL  Calories =  Total_fat_g Carbohydrate_g Protein_g;
RUN;
```

目录

自动更新分析和图形 197
示例：使用自动重新计算 197
更改首选项 201
示例：更改首选项 201
集成 JMP 与 SAS 204
示例：创建 SAS 代码 204
示例：提交 SAS 代码 204

自动更新分析和图形

当您对数据表进行更改时，可以使用 “自动重新计算”功能自动更新与数据表关联的分析和图形。例如，若您排除、包括或删除数据表中的值，该项更改会立即在相关联的分析或图形中体现。请注意下列信息：

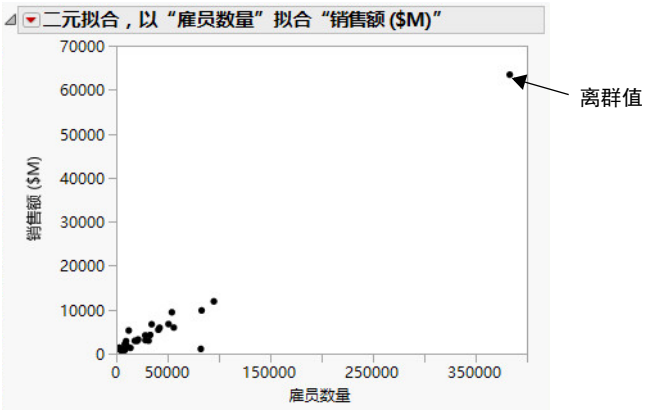
- 某些平台不支持自动重新计算功能。请参见《使用 JMP》。
- 对于**分析**菜单中支持的平台，默认情况下“自动重新计算”处于关闭状态。但是，对于**质量和过程**菜单中支持的平台，默认情况下“自动重新计算”处于打开状态，但是对于“变异性/计数量具图”、“能力”和“控制图”除外。
- 对于**图形**菜单中支持的平台，默认情况下“自动重新计算”处于打开状态。

示例：使用自动重新计算

该示例使用 Companies.jmp 样本数据表，表中包含来自医药和计算机行业的 32 家公司的财务数据。

1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**销售额 ($M)** 并点击 **Y，响应**。
4. 选择**雇员数量**并点击 **X，因子**。
5. 点击**确定**。

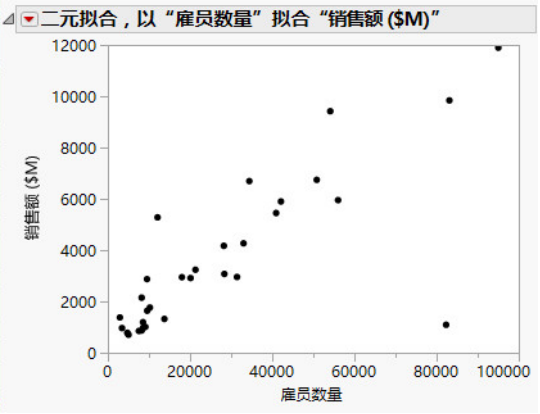
图 8.2 初始散点图



初始散点图显示其中一家公司的雇员和销售额明显多于其他公司。您确定该公司是一个离群值，想要排除该点。在您排除该点之前，需要打开“自动重新计算”功能以便在您更改时散点图可以自动更新。

6. 要开启 “自动重新计算”，点击 “二元拟合，以 ‘雇员数量’ 拟合 ‘销售额 ($M)’ ” 旁边的红色小三角，然后选择**重新运行 > 自动重新计算**。
7. 点击离群值来选上它。
8. 选择**行 > 排除 / 撤销排除**。该点从分析中排除并且散点图自动更新。

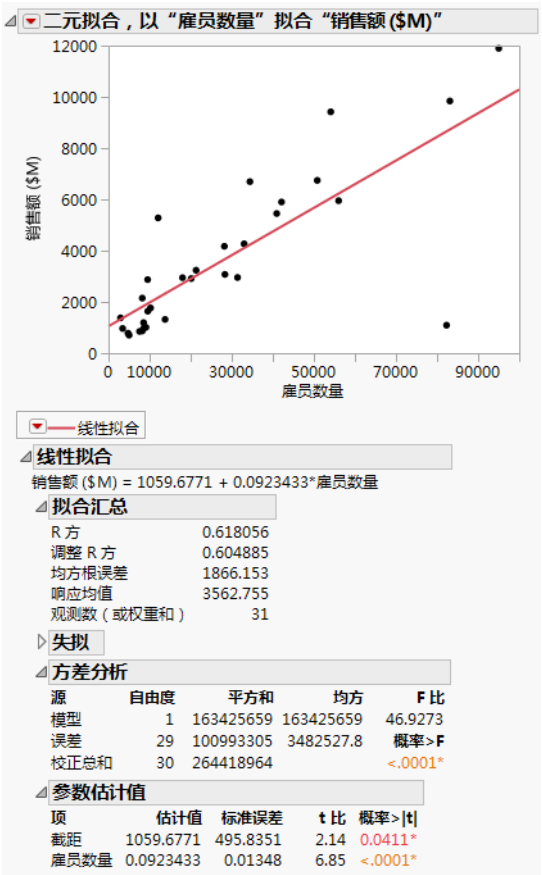
图 8.3 更新的散点图



若您针对数据拟合一条回归线，而右下角的点是一个离群值，它会影响回归线的斜率。若您在打开 “自动重新计算” 的情况下排除该离群值，则会看到回归线的倾斜度发生变化。

9. 要拟合回归线，点击 “二元拟合，以 ‘雇员数量’ 拟合 ‘销售额 ($M)’ ” 旁边的红色小三角并选择**拟合线**。图 8.4 显示了添加到报表窗口的回归线和分析结果。

图 8.4 回归线和分析结果

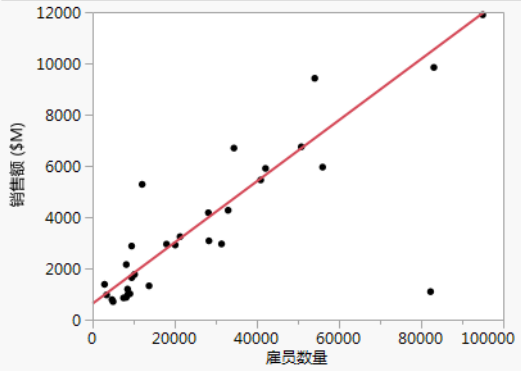


10. 点击离群值来选上它。
11. 选择**行 > 排除 / 撤销排除**。回归线和分析结果将自动更新，反映出排除了该点。

> **提示：**当您排除一个点时，分析会在不包含该数据点的情况下重新计算，但散点图中没有隐藏该数据点。若还要在散点图中隐藏该点，请选择该点，然后选择**行 > 隐藏和排除**。

图 8.5 更新的回归线和分析结果

二元拟合，以“雇员数量”拟合“销售额 ($M)”



线性拟合

线性拟合

销售额 ($M) = 631.36791 + 0.1192722*雇员数量

拟合汇总

R 方	0.882703
调整 R 方	0.878514
均方根误差	1039.891
响应均值	3644.95
观测数（或权重和）	30

失拟

方差分析

源	自由度	平方和	均方	F 比
模型	1	227857403	227857403	210.7112
误差	28	30278440	1081372.9	概率>F
校正总和	29	258135843		<.0001*

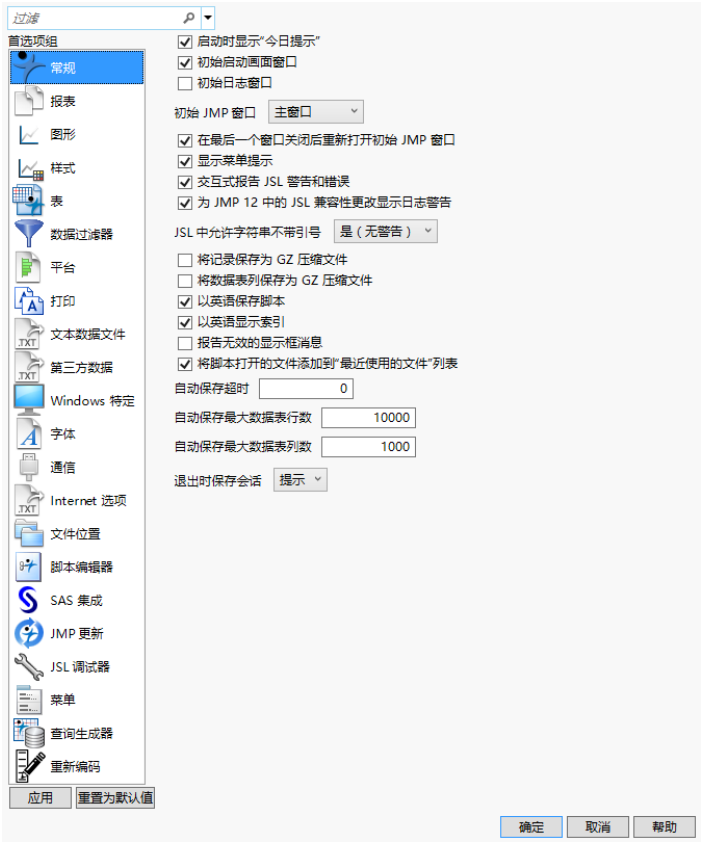
参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	631.36791	281.3288	2.24	0.0329*
雇员数量	0.1192722	0.008217	14.52	<.0001*

更改首选项

您可以在 JMP 中使用“首选项”窗口更改首选项。要打开“首选项”窗口，请选择**文件 > 首选项** (Windows) 或 **JMP > 首选项** (macOS)。

图 8.6 首选项窗口



“首选项”窗口左侧是首选项组的列表。窗口右侧是您可以为选定类别更改的所有首选项。

示例：更改首选项

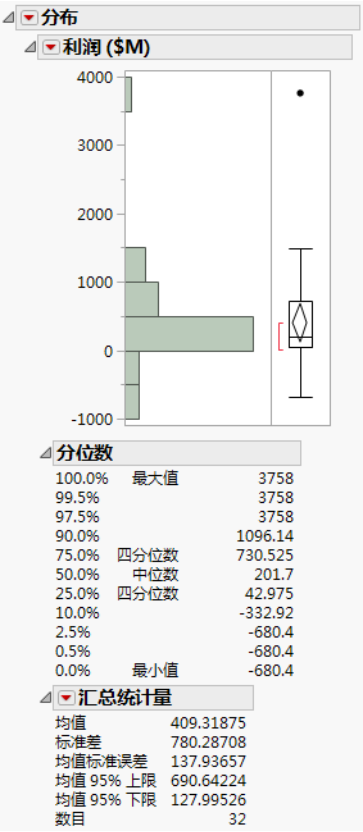
每个平台报表窗口都具有可以打开或关闭的选项。但是，您对这些选项所做的更改在您下次使用该平台时不会被记住。若您希望 JMP 记住您每次使用平台时所做的更改，可以在“首选项”窗口中更改相应的选项。

该示例显示如何设置“分布”平台，以便离群值箱线图不被添加到初始报表中。

使用默认首选项设置创建分布

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**利润 ($M)** 并点击 **Y，列**。
4. 点击**确定**。

图 8.7 “分布”报表窗口



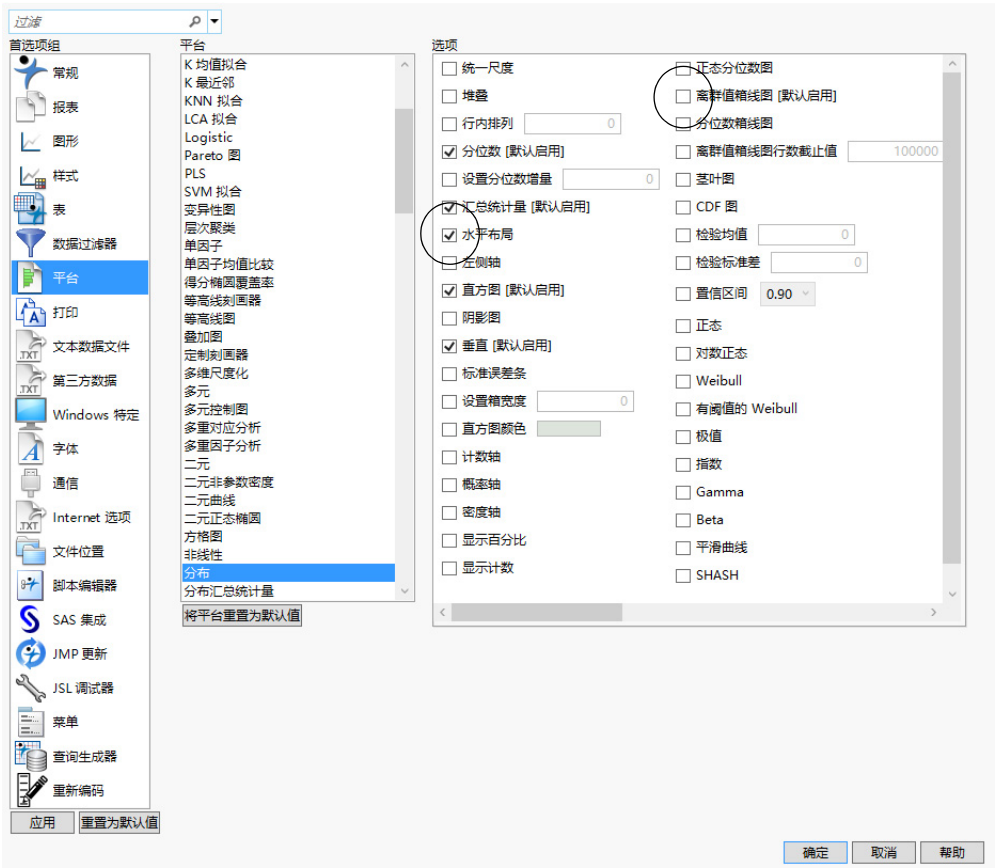
直方图是垂直方向的，并且图形中包含一个离群值箱线图。要将直方图更改为水平方向并且删除离群值箱线图，可以从“利润 ($M)”对应的红色小三角菜单中选择相应的选项。但是，若您希望这些首选项能在每次使用该平台时都有效，则需要在“首选项”窗口中进行更改。

更改离群值箱线图的首选项并重新运行分布

1. 选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)。
2. 从首选项组中选择**平台**。
3. 从 “平台”列表中选择**分布**。

4. 选择**水平布局**选项将其打开。
5. 撤销选择**离群值箱线图**选项将其关闭。

图 8.8 分布首选项



6. 点击**确定**。
7. 重复运行分布分析。请参见第 202 页的 “使用默认首选项设置创建分布”。

直方图现在是水平方向了，并且不显示离群值箱线图了。这些首选项在您下次更改前不会发生变化。

有关所有首选项的详细信息，请参见《使用 JMP》。

集成 JMP 与 SAS

注意：您必须具有从本地计算机或服务器访问 SAS 的权限才能通过 JMP 使用 SAS。

使用 JMP，您可以通过以下方式与 SAS 交互：

- 在 JMP 中编写或创建 SAS 代码。
- 在 JMP 中提交 SAS 代码并查看结果。
- 连接至 SAS 元数据服务器或远程计算机上的 SAS 服务器。
- 连接到本地计算机上的 SAS。
- 打开和浏览 SAS 数据集。
- 检索和查看 SAS 生成的数据集。

有关集成 JMP 和 SAS 的详细信息，请参见《使用 JMP》。

示例：创建 SAS 代码

该示例使用 Candy Bars.jmp 样本数据表，表中包含糖果条的营养数据。

1. 选择**帮助** > **样本数据库**，然后打开 Candy Bars.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**卡路里**并点击 Y。
4. 选择**总脂肪 (g)**、**碳水化合物 (g)** 和**蛋白质 (g)**，并点击**添加**。
5. 点击“模型规格”红色小三角并选择**创建 SAS 作业**。

图 8.9 显示了 SAS 代码。（并未显示所有数据。）

图 8.9 SAS 代码

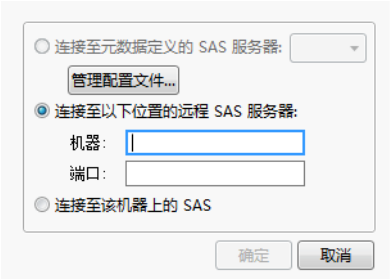
```
DATA Candy_Bars; INPUT  Calories Total_fat_g Carbohydrate_g Protein_g; Lines;
310  20  28  6
230  12  27  4
220  12  24  3
170  8  21  3
200  2.5  43  1
260  16  26  5
190  1.5  42  2
190  11  21  2
230  12  28  3
;
RUN;

PROC GLM  DATA=Candy_Bars ALPHA=0.05;
MODEL  Calories =  Total_fat_g Carbohydrate_g Protein_g;
RUN;
```

示例：提交 SAS 代码

1. 选择**帮助** > **样本数据库**，然后打开 Candy Bars.jmp。

2. 选择**分析 > 拟合模型**。
3. 选择**卡路里**并点击 **Y**。
4. 选择**总脂肪 (g)**、**碳水化合物 (g)** 和**蛋白质 (g)**，并点击**添加**。
5. 点击“模型规格”红色小三角并选择**提交至 SAS**。
6. 在**连接至 SAS 服务器**窗口中（图 8.10），选择连接至 SAS 的方法（若您尚未连接）。对于该示例，选择**连接至该机器上的 SAS**。

图 8.10 连接至 SAS 服务器



7. 点击**确定**。

JMP 连接至 SAS。SAS 运行模型并将结果返回给 JMP。结果可以显示为 SAS 输出、HTML、RTF、PDF 或 JMP 报表格式（您可以使用“JMP 首选项”进行格式选择）。图 8.11 显示了 JMP 报表格式的结果。请参见《使用 JMP》。

图 8.11 JMP 报表格式的 SAS 结果

SAS 系统
GLM 过程

GLM PROCEDURE

数据

观测数

读取的观测数	75
使用的观测数	75

因变量: 卡路里

方差分析

卡路里

总体 ANOVA

源	自由度	平方和	均方	F 值	Pr > F
模型	3	282358	94119.3	3237.58	<.0001*
误差	71	2064.03	29.0709	.	.
校正合计	74	284422	.	.	.

拟合统计量

R 方	变异系数	均方根误差	_5 均值
0.99274	2.21858	5.39174	243.027

“I 型”模型 ANOVA

源	自由度	I 型 平方和	均方	F 值	Pr > F
总脂肪 (g)	1	185260	185260	6372.68	<.0001*
碳水化合物 (g)	1	93540.4	93540.4	3217.67	<.0001*
蛋白质 (g)	1	3557.86	3557.86	122.386	<.0001*

“III 型”模型 ANOVA

源	自由度	III 型 平方和	均方	F 值	Pr > F
总脂肪 (g)	1	111777	111777	3844.97	<.0001*
碳水化合物 (g)	1	96756.1	96756.1	3328.28	<.0001*
蛋白质 (g)	1	3557.86	3557.86	122.386	<.0001*

解

参数	估计	标准 误差	t 值	Pr > |t|
截距	-5.9643	2.89999	-2.0567	0.0434*
总脂肪 (g)	8.98995	0.14498	62.0078	<.0001*
碳水化合物 (g)	4.0975	0.07102	57.6913	<.0001*
蛋白质 (g)	4.40133	0.39785	11.0628	<.0001*

附录 A

技术声明

- Scintilla - Copyright © 1998-2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library - Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages - Copyright © 2009-2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software - Copyright © 1995-2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 996-2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999-2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998 - 2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball - Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako - Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 2006 –2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright (C) 1999-2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright(C) January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

版本 16

使用 JMP

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020–2021. 使用 JMP® 16. Cary, NC: SAS Institute Inc.

使用 JMP® 16

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

March 2021

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 员工分享使用技巧和使用经历的博客
- 同其他用户讨论 JMP 的论坛

https://www.jmp.com/getstarted

目录

使用 JMP

1 了解 JMP 21

文档和其他资源

JMP 文档中的格式规范 23

JMP 帮助 24

JMP 文档库 24

学习 JMP 的其他资源 30

JMP 教程 30

样本数据表 30

学习统计和 JSL 术语 31

学习 JMP 的提示与技巧 31

JMP 工具提示 31

JMP 用户社区 31

免费在线统计思维课程 31

新 JMP 用户欢迎套件 32

统计学知识门户 32

JMP 培训 32

用户编写的 JMP 手册 32

“JMP 起始页”窗口 32

JMP 技术支持 32

2 开始使用 JMP 33

基本功能介绍

剖析 JMP 会话 35

数据表功能 35

数据表面板 37

数据表中的数据网格 43

用于打开数据文件的选项 47

平台和报表 49

平台中的启动窗口 49
报表中的分析结果 51
管理文件和打开窗口 52
Windows 系统中的 “JMP 主窗口” 52
macOS 上的 “JMP 主窗口” 55
在 Windows 上搜索最近打开的文件 57
关闭多个文件 57
显示和排列打开的窗口 57
预览 JMP 文件 58
使用窗口显示功能 59
JMP 起始页窗口 60

3 导入数据 61
创建数据表

关于导入数据 64
导入 Microsoft Excel 文件 65
预览并导入 Microsoft Excel 数据 65
直接导入 Microsoft Excel 文件 76
导入文本文件 77
查找导入数据中的缺失字符 79
“JMP 文本导入向导” 中的选项 79
将文本导入 JMP 之前对其进行编辑 83
从 “脚本编辑器” 导入文本 85
导入多个文件 86
导入 PDF 文件 90
将多个表拼接为一个数据表 90
按列标题拼接表 92
导入 JMP 未能自动检测的表 93
“PDF 预览” 红色小三角选项 95
数据表预览选项 96
PDF 预览右击选项 96
导入 XML 和 JSON 文件 96
预览和导入 XML 和 JSON 数据 96
直接导入 XML 文件 102
直接导入 JSON 文件 103

在“查询生成器”中生成 SQL 查询 104
连接至 SQL 数据库 104
从 SQL 数据库中选择表 105
生成 SQL 查询 109
保存和运行 JMP 查询 121
打开选定的数据 122
“查询生成器”红色小三角选项 122
在“查询生成器”中编写 SQL 语句 123
从 SAS 导入数据 124
Java 运行时环境 (JRE) 要求 124
SAS 菜单选项 124
导入 SAS 数据集 124
通过 JMP 在 SAS 中创建 SAS 传输文件 126
从 JMP 连接至 SAS 127
使用“SAS 查询生成器”打开 SAS 数据集 132
通过 SAS 服务器打开 SAS 数据集 133
使用 SAS 扩展属性导入元数据 140
运行存储过程 141
提交 SAS 代码 142
生成 ODS 结果 143
导入生成的 SAS 数据集 144
导入数据库中的数据 146
连接至数据库 147
从数据库中打开数据 147
编写 SQL 语句以查询数据库 149
结构化查询语言 (SQL)：参考 150
从 Google Sheets 导入数据 155
关于导入 Google Sheets 156
Google Sheet 设置 157
导入 SPSS 文件 157
导入 3S 调查数据 158
导入 HDF5 数据 159
从 MATLAB 导入数据 159
从 R 导入数据 160

使用 Excel 插件导入数据 160
将 Excel 数据传输至数据表 160
关于 Excel 的 JMP 插件 161
卸载 JMP Excel 插件 162
从 Internet 或远程计算机导入数据 162
创建新数据表 163

4 输入和编辑数据 165
执行数据表基本任务

在数据表中输入数据 168
将数据复制并粘贴到数据表中 168
向数据表添加行 168
向数据表添加列 169
使用顺序数据填充数据表列 171
使用数据表列中之前的值替换缺失值 172
将单元格公式输入数据表列 172
选择数据表中的行 173
查找行中的匹配单元格 175
选择包含特定值的行 175
定位下一选定行 177
选择数据表中的列 178
如何选择列 178
在 JMP 列查看器中选择列 179
定位数据表中的下一个选定列和上一个选定列 185
调整列和行的大小 185
在数据表中组织数据 186
删除数据表中的行 186
删除数据表中的列 186
重新排列数据表中的列 187
组合数据表中的列 187
移动数据表中的值 188
为数据表中的单元格着色 188
编辑或删除数据表中的单元格 189
编辑数据表中的列名 189
隐藏和排除数据表中的行 189

排除数据表中的行 190
隐藏数据表中的行 190
删除数据表中的列 191
隐藏数据表中的列 192
查看数据表中缺失数据的模式 192
查找和替换数据表中的单元格值 194
编辑数据表中的单元格 196
行列的右击菜单 197
比较数据表 198
为行和列指定特征 202
如何查找行列选项 203
为行和列添加标签 203
为行指定颜色或标记 204
在图形中创建颜色主题 206
删除图形中的定制颜色主题 210
选择匹配行状态单元格 211
删除行特征 211
锁定列 211
重构数据 211
将数据表列拆分为多列 212
通过文本匹配创建新列 212
生成指示符列 214
组合数据表中的列 214
压缩数据表中的选定列 215
在数据表列中生成分箱公式 217
将标签转换为代码 / 将代码转换为标签 220
生成新公式列 221
变换 JMP 平台中的列 222
对列中数据重新编码 230
编辑数据表 244
重命名数据表 244
锁定数据表 244
压缩数据表 245
复制表脚本 245

使用数据表变量 246
在数据表中创建和保存脚本 247

5 设置 JMP 列属性 251
“列信息”窗口

关于“列信息”窗口 253
关于建模类型 255
短整型格式 256
数值格式 256
在行状态列中存储信息 261
初始化列中数据 262
JMP 中的列属性 263
基本列属性 264
验证列值的属性 264
将信息附加至列值的属性 266
控制列显示的属性 270
在建模和实验设计中使用的属性 271
与控制图和能力相关的属性 272
控制列在平台中的使用方式的属性 273
SAS 属性 280
链接数据表与虚拟连接属性 280
其他列属性 282
由 JMP 分配并控制的属性 284
在各列之间标准化特性和属性 286
对列值重新编码 286
标准化列特性 287
标准化列属性 287
删除列属性 287
关于公式标准化的示例 287
为列指定预选分析角色 288

6 重塑数据 291
创建子集、数据排序等

创建子集数据表 293
分层的子集 294

从报表创建子集数据表 295
数据表排序 296
堆叠列 298
示例：将多列堆叠为一列 300
示例：将多列堆叠为多列 301
拆分列 302
拆分列的示例 304
转置行和列 306
行和列转置的示例 308
拼接数据表 309
拼接数据表的示例 310
拼接数据表和表变量的示例 311
连接数据表 312
连接数据表的示例 314
使用 JMP 查询生成器查询和连接数据表 321
关于指向查询中数据表的链接 324
虚拟连接数据表 324
虚拟连接列的示例 325
示例：虚拟连接的列和显示行状态 328
示例：合并虚拟连接的数据表中的引用数据 330
示例：单个列的链接 ID 和链接引用 332
更新数据表 335
更新数据表的示例 337
匿名数据 338

7 在 JMP 中创建公式 339

公式编辑器

公式概述 342
生成公式 342
在公式中引用数据表值 344
在公式中使用局部变量 346
在公式中合并参数 347
在公式中插入常数 348

将运算符添加至 JMP 公式 349
“公式编辑器”的小键盘参考 350
在公式中使用函数 351
查找“公式编辑器”中经常使用的函数 353
复制和重复公式 355
对公式中的表达式排序 357
按优先顺序生成公式 358
JMP 公式编辑器选项 359
计算 JMP 函数的导数 359
简化复杂公式 359
计算公式 360
忽略公式中的错误 361
查看用 JSL 显示的公式 361
编辑公式 361
更正公式中的错误 362
在公式中选择表达式 362
删除函数公式 362
剪切、复制和粘贴公式 363
点击并拖动公式 363
在 JMP 公式编辑器中定制公式 363
隐藏和显示分级显示的公式 364
更改字体大小公式 364
指定矩阵大小公式 364
更改 JMP 公式的方向 364
打开和关闭参数公式 365
示例：在 JMP 公式编辑器中生成公式 365
使用 JMP 公式编辑器中的基本功能 365
使用局部变量公式 366
使用 Munger 函数公式 367
使用 Match 条件函数公式 368
使用删除表达式按钮公式 369
JMP 公式编辑器的键盘快捷方式 370
JMP 公式编辑器的术语表 371

8 汇总数据 373
表汇总命令
汇总列 375
汇总启动窗口 375
创建汇总表 375
向现有汇总表添加统计量列 378
解释汇总统计量 379
示例：创建带分组的汇总表 381

9 JMP 报表 383
导航和定制报表窗口
导航 JMP 报表 386
使用手形工具 387
JMP 报表的显示选项 387
显示和隐藏 JMP 报表部分 389
合并多个报表 389
重命名报表中的标题 389
增大报表中的字体 390
调整报表窗口的大小 390
JMP 报表中的红色小三角选项 390
报表中的“本地数据过滤器” 391
报表中的“重新运行”菜单 393
报表中的“保存脚本”菜单 393
数据过滤器 396
过滤器列的类型 397
过滤模式 398
“数据过滤器”控制面板 399
“数据过滤器”的红色小三角选项 401
数据过滤器中的动画 405
针对数据过滤器中的变量的红色小三角选项 406
重新格式化报表中的表 409
修改 JMP 报表中的显示框属性 410
将 JMP 报表中的表转换为矩阵 411
在报表中使用条件格式 411

选择图形中的点 416
在图形中选择行 417
在 JMP 图形中选择矩形点区域 417
在 JMP 图形中选择点的不规则形状区域 418
在图形中使用标记 419
更改标记的形状 419
更改标记的颜色 420
更改标记的大小 420
处理拥挤的标记 420
为标记添加轮廓 421
更改标记的选择模式 421
指定标记的透明度 422
排除和隐藏标记 423
将图像用作标记 423
基于值更改标记的形状或颜色 424
更改图形外观 426
调整图形大小 426
放大图形 427
更改图形中的线条粗细 428
更改图形中的背景颜色 428
更改直方图直条的颜色 428
显示图形中的坐标和临时参考线 428
在图形中滚动和调整轴尺度 429
定制图形中的轴和轴标签 429
更改报表中的值顺序 439
更改图形中选定对象的模式和格式 440
将图形添加至数据表 440
定制图形中的标记、线条、文本等元素 441
将图像添加到 JMP 图形或报表 446
将背景图像粘贴到 JMP 图形中 446
在 JMP 报表结尾粘贴图像 446
将图像拖放到 JMP 图形中 447
向 JMP 图形添加地理图像和边界 449
从图像中提取数据 449

向 JMP 报表和图形添加注解和形状 450
向 JMP 报表或图形添加注解 450
向 JMP 报表添加形状 451
定制图形中的悬停标签 454
向悬停标签添加图形或图像 455
向图形中的悬停标签添加富文本 469
更新悬停标签中的名称和值 472
编辑图形中的悬停标签 482
将数据表中的图像添加到悬停标签 487
固定图形中的悬停标签 488
悬停标签的右击选项 489

10 保存和共享您的数据 491
在 JMP 外部利用 JMP 数据

保存和共享数据表 494
将数据表另存为 Microsoft Excel 文件 494
将 JMP 数据另存为 CSV 文件 496
将 JMP 数据另存为文本文件 496
将 JMP 数据另存为 JSON 文件 497
将 JMP 数据另存为 SAS 传输文件 498
将 JMP 数据另存为 Google Sheet 499
将 JMP 数据另存为 SAS 数据集 500
将数据表保存至数据库 501
使用数据表替换数据库 502
保存和共享报表 502
将报表共享为交互式 HTML 504
将报表发布至 JMP Live 或 JMP Public 507
将报表另存为 PDF 文件 516
将报表另存为 PowerPoint 演示文稿 517
设置导出图形的图形 DPI 518
将 JMP 分析另存为脚本 518
以图形格式保存报表部分 519
打印报表 520
复制和粘贴报表 520
使用项目 521

创建新项目 522
在项目中打开文件 522
在项目中重新排列文件 522
保存项目 524
项目工作区 524
项目书签 525
项目内容 525
项目中最近使用的文件 526
项目日志 526
将文件移动到项目中 527
编写项目 On Open 脚本 527
创建新项目的示例 528
创建记录 530
进行演示制作记录的示例 531
新建记录 532
管理共享记录中的链接 533
向记录添加分级显示项并对其进行编辑 534
记录中的右击选项 535
控制记录中分级显示级别的显示 535
向记录添加图形或图像 536
定制记录中的项 536
防止修改记录 536
向记录追加报表 537
以其他格式保存记录 537
删除记录中的项 537
保存 JMP 会话 538
退出 JMP 时保存会话 538
手动保存会话 538
保存日志窗口 539

11 使用仪表板组合和演示报表 541
扩展 JMP

通过创建仪表板来组合 JMP 报表 543
示例：创建包含两个 JMP 报表的仪表板 543
使用“仪表板生成器” 544

“仪表板生成器”中的红色小三角选项 545
从样本仪表板入手 546
编辑仪表板 546
查看运行的仪表板 549
示例：组合窗口以创建仪表板 553
示例：向仪表板添加“选择内容过滤器” 555
示例：创建 JMP 查询仪表板和插件 557
示例：从两个数据表创建仪表板 559

12 对菜单和工具栏进行个性化设置 561
定制 JMP

在 Windows 上对 JMP 工具栏和菜单进行个性化设置 563
对工具栏进行个性化设置 564
对主菜单栏进行个性化设置 567
创建菜单项和工具栏按钮 567
设置定制工具栏和按钮的用户权限 571
重新排列工具栏 571
复制和粘贴菜单、菜单项、工具栏和按钮 572
重新排列定制菜单、菜单项和按钮 573
删除定制项 575
显示或隐藏定制项 576
导入定制设置 576
从 JMP 中删除定制设置 577
在 macOS 上对 JMP 工具栏进行个性化设置 578
在 macOS 上对 JMP 菜单进行个性化设置 579
为插件定制 JMP 菜单和工具栏 580

13 JMP 首选项 581
“首选项”窗口

JMP 首选项概述 583
常规 JMP 首选项 583
JMP 项目的首选项 586
JMP 报表的首选项 587
JMP 图形的首选项 589
JMP 图形和报表的样式首选项 592

数据表的首选项 594
“JMP 数据过滤器”的首选项 598
JMP 平台首选项 599
“打印”首选项 601
用于导入和导出文本文件的首选项 602
第三方数据的首选项 606
Windows 首选项 608
Mac 操作系统首选项 610
“字体”首选项 612
“通信”首选项 614
“Internet 选项”的首选项 614
“文件位置”的首选项 615
“JSL 脚本编辑器”的首选项 617
“SAS 集成”首选项 619
用于更新 JMP 的首选项 622
“JSL 调试器”的首选项 623
“菜单”首选项 623
“查询生成器”的首选项 624
用于对列重新编码的首选项 625
“列切换器”的首选项 627
“日志”首选项 628

A 在公式中引用 JMP 函数 631

公式编辑器中函数的说明

行函数 633
数值函数 635
超越函数 635
三角函数 639
字符函数 639
比较函数 644
条件函数 645
概率函数 650

离散概率函数 657
统计函数 658
随机函数 662
日期时间函数 667
行状态函数 670
赋值函数 675
参数模型函数 675
财务函数 676

B JMP 技术详细信息 679

支持多线程的 JMP 功能 681
将 JMP 特性映射为 SAS 扩展属性的约定 682

C 技术许可声明 689

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 23
JMP 帮助 24
JMP 文档库 24
学习 JMP 的其他资源 30
JMP 教程 30
样本数据表 30
学习统计和 JSL 术语 31
学习 JMP 的提示与技巧 31
JMP 工具提示 31
JMP 用户社区 31
免费在线统计思维课程 31
新 JMP 用户欢迎套件 32
统计学知识门户 32
JMP 培训 32
用户编写的 JMP 手册 32
“JMP 起始页”窗口 32
JMP 技术支持 32

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助 > JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助 > JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 24 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助 > JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • Pareto 图 • 关系图 • 管理规格限

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “新 JMP 用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

新 JMP 用户欢迎套件

“新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

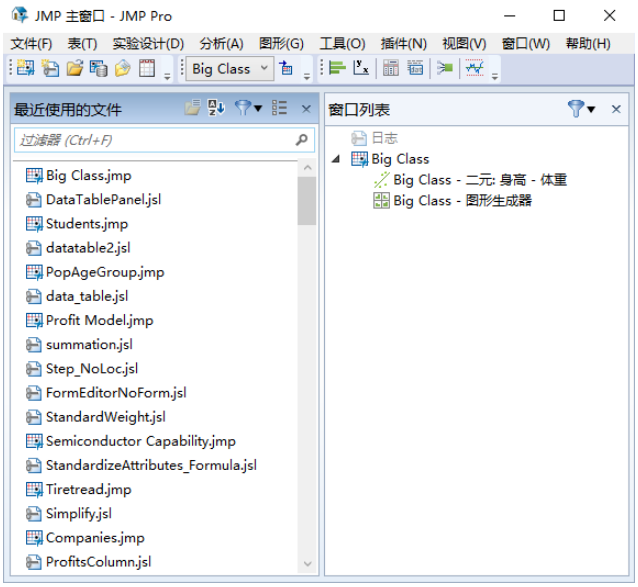
第 2 章

开始使用 JMP
基本功能介绍

为尽快让您了解 JMP，本章介绍了以下主题：

- 了解启动 JMP 时出现的初始窗口
- 了解数据表
- 打开数据文件
- 管理打开的窗口
- 剖析典型的 JMP 用户会话

图 2.1 Windows 系统中的 “JMP 主窗口”



目录

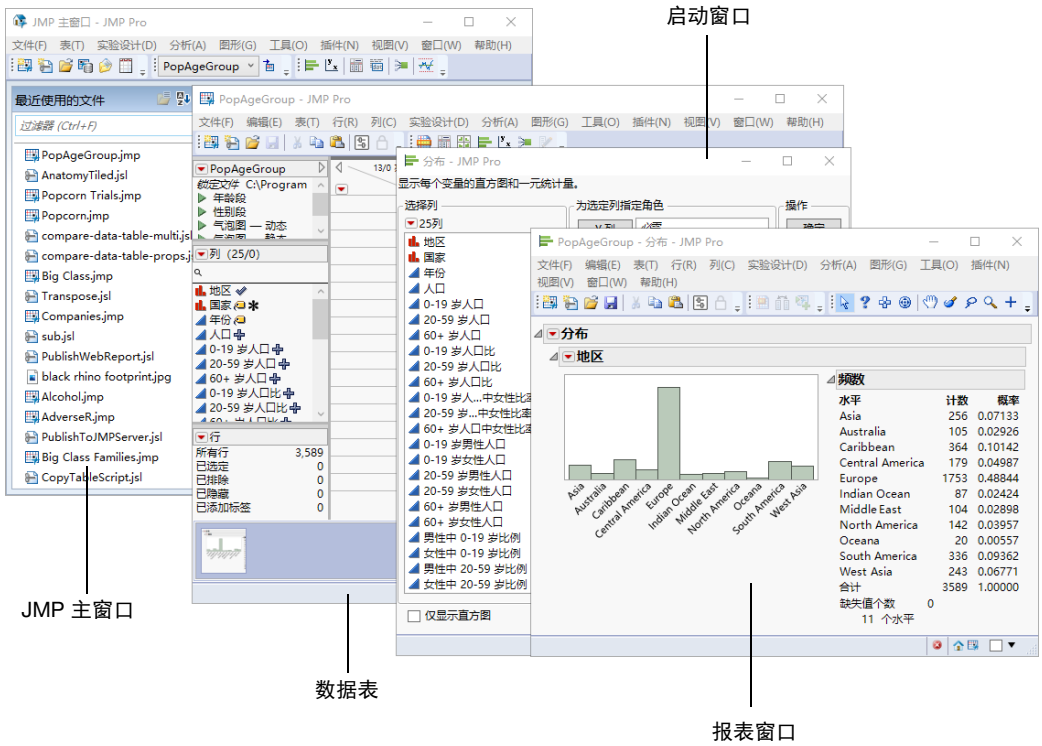
剖析 JMP 会话 35
数据表功能 35
数据表面板 37
数据表中的数据网格 43
用于打开数据文件的选项 47
平台和报表 49
平台中的启动窗口 49
报表中的分析结果 51
管理文件和打开窗口 52
Windows 系统中的“JMP 主窗口” 52
macOS 上的“JMP 主窗口” 55
在 Windows 上搜索最近打开的文件 57
关闭多个文件 57
显示和排列打开的窗口 57
预览 JMP 文件 58
使用窗口显示功能 59
JMP 起始页窗口 60

剖析 JMP 会话

JMP 包含按照统计分析类型组织的多个平台。例如，“分布”平台使用直方图、其他图形和报表生成一元分析。您可以在此“分布”平台中分析数据，然后选择其他平台更为彻底地检查数据。结果导致若干窗口同时打开：至少包括数据表、平台启动窗口和分析结果。

图 2.2 显示了一个典型的 JMP 会话。请注意，这些窗口不会最大化，这样您可以从一个窗口快速切换到另一个窗口。

图 2.2 剖析 Windows 上的典型 JMP 会话



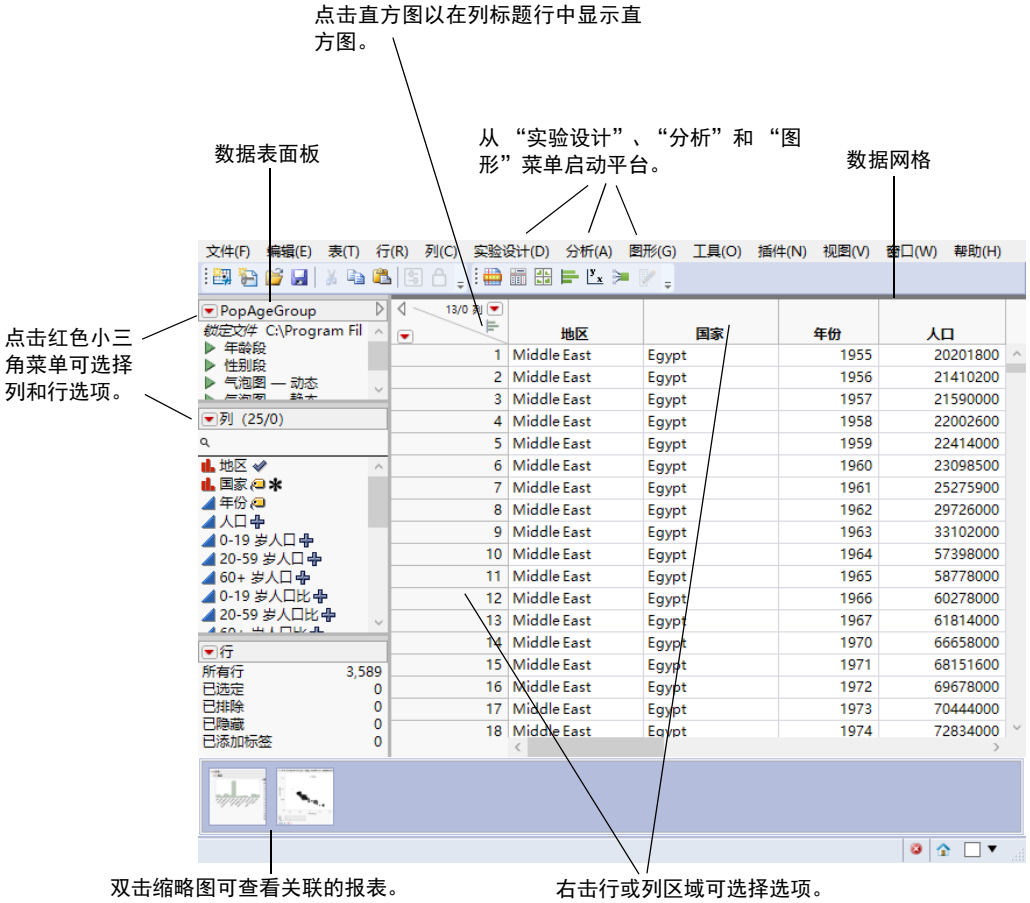
以下各节介绍您通常使用的窗口。

数据表功能

在 JMP 中，数据点被组织成行与列。这些行与列称为**数据表**。数据表有两个部分：左侧的**数据表面板**和右侧的**数据网格**。

您可以使用数据表输入、查看、编辑和操作数据。在数据表中，每个变量都是一列，每个观测都是一行。

图 2.3 数据表功能



数据表具有以下特点：

- 列名可包含任意键盘字符（包括空格），而且可长达 255 个字符。
- 数据表名称的最大长度取决于用户计算机的操作系统。
- 考虑将 “常规” 首选项中的**自动保存超时值**设置为在指定的分钟数自动保存数据表。该自动保存值还适用于记录、脚本、项目和报表。
- 通过选择**文件 > 首选项 > 字体**可以更改名称和值的默认大小和字体。（在 macOS 上，选择 JMP > 首选项 > 字体。）
- 列名在列名区域中自动换行，以适应您指定的列宽。
- 通过移动列边界来扩展列，可查看长值。通过在任意选定列边界上拖动双箭头光标的同时按 Alt 键，可同时调整所有选定列的宽度。
- 数据表中行列的数目仅受到用户计算机内存空间的限制。

- 通过拖动行的一个边框可调整行的大小。所有行会被调整为相同高度。每个单元格内显示的图形会根据行的高度收缩。

  要将列调整为相同宽度，选择相应列并按 Alt (Windows) 或 Option (macOS)；拖动列的一个边框。所有列会被调整为相同宽度。

提示：要关闭基于 Windows 的数据表中的报表缩略图，请按您鼠标上的中间点击按钮。

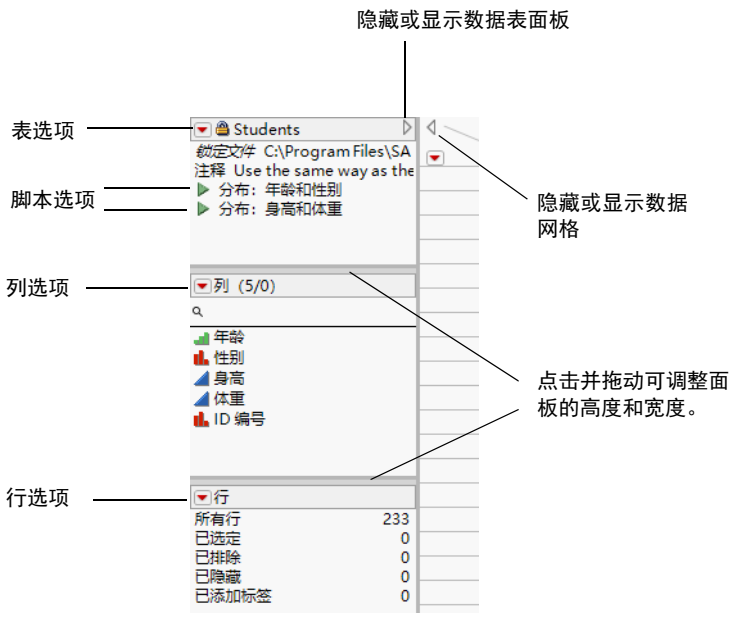
数据表面板

数据表包含三个面板：

- 表面板
- 列面板
- 行面板

这些面板位于数据网格的左侧，其中包含有关表及其内容的信息。每个面板都具有交互区域。

图 2.4 与数据表面板交互



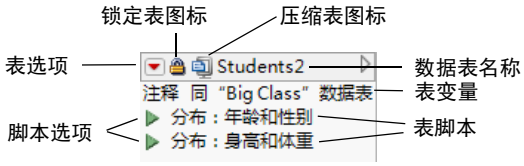
数据表中的表面板

表面板包含以下元素：

- 数据表的名称

- 指示表状态的图标
- 包含表选项的红色小三角菜单
- （可选）表变量
- （可选）表脚本的绿色小三角

图 2.5 表面板示例



表选项

点击数据表名称旁边的红色小三角菜单可显示以下选项：

表 包含与 “表”菜单相同的选项。请参见第 291 页的 “重塑数据”一章。

新建表变量 创建新的表变量，该变量可以是文本，也可以是您希望在数据表中始终可用的任何其他常量字符值。表变量通常用于记录表。请参见“输入和编辑数据”一章中第 246 页的 “使用数据表变量”。

注意：要重命名某个表变量，请双击该变量，然后在 “名称”字段中输入新名称。

新建脚本 创建要与数据表一同保存的 JSL 脚本。选择该命令后，为该脚本命名并键入值（JSL 命令）。点击**确定**后，新脚本将列在 “表”面板中，您可以通过点击其绿色小三角来运行、编辑或删除该脚本。请参见 “输入和编辑数据”一章中第 247 页的 “在数据表中创建和保存脚本”。

组合脚本 将选定的脚本移入组中。点击组名将其重命名。这在您希望组织脚本或是尽量缩短较长的脚本列表时很有帮助。

取消组合脚本 取消组合选定组中的脚本，使其返回顶级。

禁止公式计算 关闭自动计算公式的功能。您可以关闭计算并编写公式的各个部分。之后要检验公式，请取消选择 “禁止公式计算”。

锁定数据表 锁定数据表，使得不可编辑或添加数据和列属性。您仍可以运行分析并指定特征。请参见 “输入和编辑数据”一章中第 244 页的 “锁定数据表”。

保存时压缩文件 保存数据表时压缩该表。保存数据表后，表面板中该数据表名称的旁边会显示一个压缩文件图标 。请参见 “输入和编辑数据”一章中第 245 页的 “压缩数据表”。

保存时压缩文件选项只会缩小文件大小。该命令不影响分析数据所需的内存大小。要减小用于分析的文件大小和内存，请使用**列 > 实用工具 > 压缩选定列**。请参见“输入和编辑数据”一章中第 215 页的 “压缩数据表中的选定列”。

提示：您还可以配置 JMP 在保存表时始终使用 GZ 压缩，方法是选择**首选项 > 常规 > 将数据表列保存为 GZ 压缩文件**。

禁用撤销 删除撤销历史中的所有操作，且不记录将来的操作。仅在数据表打开时禁用撤销；该设置不会与数据表一起保存。该选项可节省内存，特别是当您删除许多行或是对需要大量内存来记录数据的数据表执行其他任务时。

复制表脚本 复制用于重新创建表的脚本。要重新创建表，请将复制的脚本放入新脚本中，然后运行它。请注意，虚拟连接表中的引用列不会包括在脚本中。

复制表脚本 （无数据） 复制用于重新创建表但省略数据的脚本。

重新运行公式 重新计算数据表中包含公式的所有列。

脚本选项

要从数据表面板运行脚本，请点击脚本名称旁边的绿色小三角 。右击脚本名称或绿色小三角可显示这些选项：

运行脚本 运行脚本。

提示：按 Ctrl 键的同时选择您想要运行的表脚本，可一次运行多个表脚本。然后，在表脚本列表下方的空白区域内右击，并选择**运行脚本**。

编辑 在脚本编辑器中打开大多数脚本，以便对其进行编辑。在“应用程序生成器”中打开 JMP 应用程序脚本。

使用重新编码编辑 在脚本编辑器中打开重新编码脚本 （即您通过重新编码列并保存脚本而创建的脚本）。

删除 删除脚本。

组合脚本 将选定的脚本移入组中。点击组名将其重命名。这在您希望组织脚本或是尽量缩短较长的脚本列表时很有帮助。

取消组合脚本 取消组合选定组中的脚本，使其返回顶级。

复制 复制脚本。随后可以将其粘贴到另一个数据表的表面板中。

调试脚本 在 JSL 调试器中打开脚本。请参见 《Scripting Guide》。

粘贴 从其他数据表粘贴该脚本。

其他选项

在表面板中，您还可以执行以下操作：

- 双击某个表变量或脚本名称可编辑其名称和内容。
- 拖动表变量或脚本来重新排列它。

数据表中的列面板

列面板提供了一种查看和移动列的方式。该面板包含以下信息：

- 列选项（包含与“列”菜单中相同的一些选项）
- 数据表中的总列数和选定列数
- 有助于快速查找列的过滤器框（点击放大镜图标可设置搜索过滤器选项和参数）
- 在数据表中显示的列的列表
- 指示每列的建模类型的图标（请参见“设置 JMP 列属性”一章中第 255 页的“关于建模类型”）
- 表示为列指定的特征和属性的图标（未显示，请参见图 2.7）

图 2.6 列面板示例

表示列特征和属性的图标

每个列名右侧的图标指示该列包含的特征和属性。

图 2.7 表示列特征和属性的图标

注意：斜体表示该列已锁在原位。水平滚动时，该列保持可见。

以下是可能出现在“列”面板中的图标：

指示图中与该列对应的数据点使用值而不是行号来标记。请参见“输入和编辑数据”一章中第 203 页的 “为行和列添加标签”。

指示从计算中排除该列。请参见“输入和编辑数据”一章中第 190 页的“排除数据表中的行”。

指示图形中不会包括该列。请参见 “输入和编辑数据”一章中第 190 页的 “隐藏数据表中的行”。

可为 X 或 Y。指示已为该列指定了预选角色 x 或 y。请参见 “设置 JMP 列属性”一章中第 288 页的 “为列指定预选分析角色”。

指示已为该列指定了预选的**验证**角色。请参见 “设置 JMP 列属性”一章中第 288 页的“为列指定预选分析角色”。

指示该列包含一个或多个属性。点击可显示该列包含的属性的列表。

指示列中各值是公式计算的结果。禁止公式计算时，该图标灰显。双击可查看和编辑公式。请参见 “在 JMP 中创建公式”一章中第 359 页的 “JMP 公式编辑器选项”。

指示已启用范围检查或列表检查选项。点击可查看和编辑范围或列表。请参见 “设置 JMP 列属性”一章中第 265 页的 “范围检查”和 “设置 JMP 列属性”一章中第 265 页的“列表检查”。

指示已为该列指定了预选的**权重**角色。请参见 “设置 JMP 列属性”一章中第 288 页的“为列指定预选分析角色”。

指示已为该列指定了预选的**频数**角色。请参见 “设置 JMP 列属性”一章中第 288 页的“为列指定预选分析角色”。

指示列值无法编辑。请参见 “设置 JMP 列属性”一章中第 253 页的 “关于 “列信息”窗口”。

指示表达式列包含可用作标记的图像。请参见“JMP 报表”一章中第 423 页的“将图像用作标记”。

虚拟连接图标

指示列具有 “链接引用”列属性。该属性将当前数据表中的列链接到引用数据表中的 ID 列。蓝色指示已链接引用的数据表。

指示列具有 “链接引用”列属性。灰色指示引用的数据表未打开或未能正确链接。

指示该列具有 “链接 ID”列属性，该属性将数据表中的列标记为 ID 列。也就是说，该数据表的各行通过该 ID 列的值唯一标识。

指示该列链接自引用的数据表。

有关虚拟连接的详细信息，请参见 “重塑数据”一章中第 324 页的 “虚拟连接数据表”。

搜索过滤器选项

若要在列列表中显示搜索过滤器选项，请从列列表的红色小三角菜单中选择“显示过滤器”。这样您就可以定制列搜索，这对于搜索较长的一组列的情况尤其有帮助。

点击搜索框旁边的下箭头按钮以优化搜索。

包含词条 返回包含一部分搜索条件的项。搜索“ease oom”返回如“Release Zoom”这样的消息。

包含短语 返回包含完全搜索条件的项。搜索“text box”返回包含“text”后面直接跟着“box”的条目（例如，“Context Box”和“Text Box”）。

以短语开头 返回以搜索条件开始的项。

以短语结尾 返回以搜索条件结束的项。

整条短语 返回包含整个字符串的项。搜索“text box”返回仅包含“text box”的条目。

正则表达式 允许您在搜索框中使用通配符 (*) 和句点 (.)。搜索“get.*name”查找包含“get”后面跟着一个或多个单词的项。它返回“Get Color Theme Names”、“Get Name Info”、“Get Effect Names”等。

反转结果 返回不匹配搜索条件的项。

匹配全部项 返回同时包含两个字符串的项。搜索“t test”返回包含任一搜索字符串或两者的元素：“Pat Test”、“Shortest Edit Script”和“Paired t test”。

忽略大小写 忽略搜索条件中的大小写。

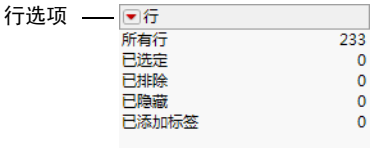
全字匹配 基于“匹配全部项”设置返回包含字符串中每个单词的项。若您搜索“data filter”，并且选择了“匹配全部项”，则返回同时包含“data”和“filter”的条目。

数据表中的行面板

行面板包含以下信息：

- 行选项（与**行**菜单中的选项相同）
- 总行数
- 已选定（突出显示）、已排除、已隐藏和已添加标签的行的数目

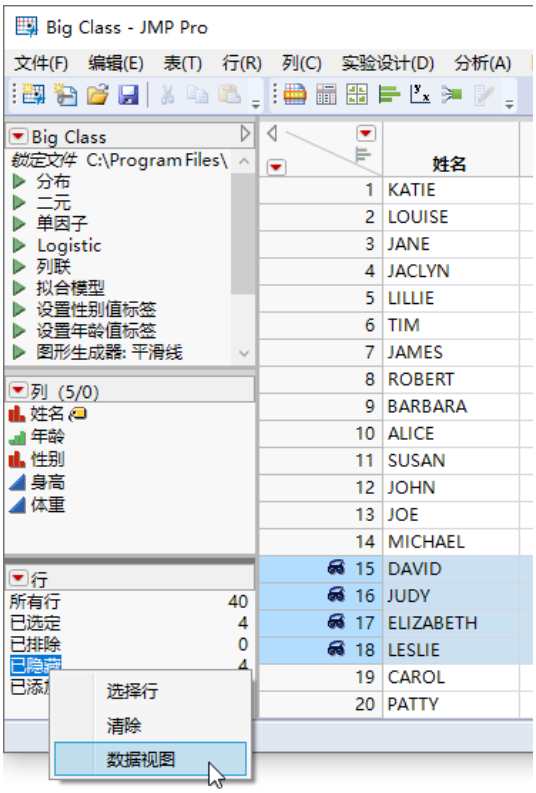
图 2.8 行面板示例



右击行面板中的类别可选择行、清除选定内容或创建**数据视图**。

数据视图创建主数据表的链接子集。例如，若将若干行标记为隐藏，您可能想打开一个仅显示这些隐藏行的窗口。此时可在行面板中右击**已隐藏**并选择**数据视图**。

图 2.9 从行面板创建数据视图

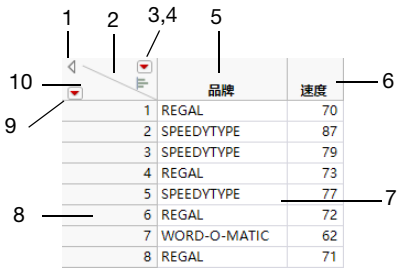


使用数据视图时，可继续在主数据表中执行多数编辑工作。在主数据表或数据视图中所做的更改将体现在这二者中。您可以在数据视图中进行细微更改（如更改某些数据或添加列）。但是，若要做出重大更改（如添加公式），则必须在主数据表中执行这些更改。

数据表中的数据网格

数据网格是数据表的主要组成部分，用来包含数据。图 2.10 显示如何与数据网格交互。另见“输入和编辑数据”一章中第 173 页的“选择数据表中的行”。

图 2.10 与数据网格交互



编号	说明
1	隐藏或显示数据表面板。
2	点击可取消选择所有选定列。要选择所有列，请按 Shift 键并点击。双击以添加多列。
3	红色小三角显示列菜单。
4	直方图显示位于列标题行中的直方图。
5	点击可选择列。 双击可查看“列信息”窗口。请参见第 253 页的“关于“列信息”窗口”。 右击可显示列选项。请参见第 44 页的“列的右击选项”。
6	右击列名以在“列信息”窗口中对其进行编辑，或者选择列名并按 Enter 键。
7	点击并拖动以更改列宽。要更改若干列的宽度，请选择这些列，按 Alt，然后点击并拖动。
8	点击可选择行。 双击可打开“行编辑器”。请参见“输入和编辑数据”一章中第 196 页的“编辑数据表中的单元格”。 右击可显示行选项。请参见第 45 页的“行的右击选项”。
9	显示行菜单。
10	点击可取消选择所有选定行。要选择所有行，请按 Shift 键并点击。双击以添加行。

列的右击选项

在列标题中右击可显示以下选项：

列信息 打开 “列信息”窗口。请参见第 253 页的 “关于 “列信息”窗口”。

标准化特性 将特性（数据类型、建模类型、数值格式等）或属性（公式、注释、列表和范围检查等）从一列复制到其他列。同时从列中删除选定的特性或属性。请参见“设置 JMP 列属性”一章中第 286 页的“在各列之间标准化特性和属性”。

列属性 包含列属性的列表。选择其中一个属性可打开“列信息”窗口并将该属性应用到列。“列信息”窗口也提供该列表。请参见“设置 JMP 列属性”一章中第 263 页的“JMP 中的列属性”。

公式 打开“公式编辑器”。请参见第 339 页的“在 JMP 中创建公式”一章。

重新编码 支持您同时更改某列中的所有值。请参见“输入和编辑数据”一章中第 230 页的“对列中数据重新编码”。

新建公式列 创建公式列。请参见“输入和编辑数据”一章中第 221 页的“生成新公式列”。

插入列 在选定的一个或多个列之前插入一个或多个列。

删除列 删除所有选定列。

添加标签 / 撤销标签 在所有图中为选定列添加标签或撤销标签。请参见“输入和编辑数据”一章中第 203 页的“为行和列添加标签”。

链接 ID 用在虚拟连接数据表中，用来将辅助数据表中的某列标记为 ID 列。也就是说，该数据表的各行通过该 ID 列的值唯一标识。若列为数据表的 ID 列，则“链接 ID”列属性复选框将被选中。请参见“设置 JMP 列属性”一章中第 280 页的“链接数据表与虚拟连接属性”。

链接引用 用在虚拟连接数据表中，用来将主数据表中的某列映射为辅助数据表中的 ID 列。“链接引用”列属性指定辅助数据表的路径名称。请参见“设置 JMP 列属性”一章中第 280 页的“链接数据表与虚拟连接属性”。

排序 按选定列中的值对表中所有行排序。您可以选择按升序或降序对行排序。请参见“重塑数据”一章中第 296 页的“数据表排序”。

复制列属性 复制列属性。

粘贴列属性 粘贴列属性。

复制列 复制脚本以重新创建列。它复制所有特性和属性，但不复制任何数据。

粘贴列 从“复制列”生成的脚本中创建新列。该列将放在选定列之后。若未选定列，则新列将放置在数据表的末尾。新列填充（在目标表中）的行数与脚本中的数据行数相同。

行的右击选项

在行标题中右击可显示以下选项：

隐藏和排除 在所有图形中隐藏选定行并将它们从分析中排除。请参见“输入和编辑数据”一章中第 189 页的“隐藏和排除数据表中的行”。

排除 / 撤销排除 在分析中排除或包括选定行。请参见“输入和编辑数据”一章中第 190 页的“排除数据表中的行”。

隐藏 / 撤销隐藏 在所有图形中隐藏或显示选定行。请参见 “输入和编辑数据” 一章中第 190 页的 “隐藏数据表中的行”。

添加标签 / 撤销标签 在所有图中为选定行添加标签或撤销标签。请参见 “输入和编辑数据” 一章中第 203 页的 “为行和列添加标签”。

颜色 提供调色板。选择一种颜色可将其应用到选定行。该颜色用在图形中。请参见 “输入和编辑数据” 一章中第 204 页的 “为行指定颜色”。

标记 提供标记或符号的选项板。选择一个标记可将其应用到选定行。该标记应用于图形而不是数据点。请参见 “输入和编辑数据” 一章中第 204 页的 “为行添加标记”。

按行状态为行着色 用当前行状态颜色为行着色。请参见 “输入和编辑数据” 一章中第 205 页的 “基于列值为行指定颜色或标记”。

选择匹配单元格 在活动数据表中选择值与选定行匹配的行。请参见 “输入和编辑数据” 一章中第 185 页的 “调整列和行的大小”

反转选择 选择所有先前被取消选择的行，同时取消选择当前选定的所有行。

清除行状态 清除数据表中的所有活动行状态。所有行变为已包括、可见、未添加标签的行，并在图中显示为黑点。该选项不影响保存在行状态列中的行状态。请参见 “输入和编辑数据” 一章中第 211 页的 “删除行特征”。

添加行 将指定数目的行添加至数据表。请参见 “输入和编辑数据” 一章中第 168 页的 “向数据表添加行”。

删除行 从数据表中删除所有选定行。使用**编辑**菜单上的**撤销**命令可撤销意外删除。请参见 “输入和编辑数据” 一章中第 186 页的 “删除数据表中的行”。

光标形状

光标可具有不同形状，具体取决于它在数据网格中的位置。

箭头光标 标准的箭头光标出现在以下位置：

- 在数据表左侧的面板区域中
- 在数据网格左上角三角形的行和列区域中

您可以使用箭头光标执行以下操作：

- 要使用箭头光标选择某列，请在列面板中点击该列的名称。
- 双击列面板中的列名可编辑该列名。
- 点击位于数据网格左上角的三角形区域可取消选择行列。

选择 （大加号）光标 当光标位于列标题顶部或行号区域中时，它会变为一个大加号，指示您可以选择行或列。一旦点击，行或列即被选定并突出显示。点击并拖动可选择多个行或列，按 Ctrl 键并点击可选择不连续的行或列。

- 双击列标题区域可查看该列的 “列信息” 窗口。

– 选择某列可更改列名。该列将突出显示。开始键入（若光标不在锁定列或锁定数据表中）。

– 双击行号区域可使用“行编辑器”编辑行。

I 形光标 选择可编辑的文本时，光标会变为标准的 I 形。要编辑文本，请将 I 形置于突出显示的文本内。点击可标记插入点，拖动可选择要替换的文本。

双箭头光标 光标在处于列或面板边界上时会变为双箭头。向左或向右拖动该光标可更改列或面板的宽度。更改列宽不影响在“列信息”窗口（通过双击列名即可访问该窗口）中指定的列字段宽度。

提示：通过在任意选定列边界上拖动双箭头光标的同时按 Alt 键，可同时调整所有选定列的宽度。

列表检查光标 当您将鼠标移至要求数据验证的列中的值上方时，光标会改变形状。光标会在要求列表检查的列上变为一个指向下方的小箭头。点击时，该值将突出显示，光标会变为 I 形。使用定义为有效文本或有效数字的任意值按常规输入或编辑数据。请参见“设置 JMP 列属性”一章中第 265 页的 “列表检查”。

指针光标 该光标在以下对象上变为指针：

– 用于提供选项的红色小三角菜单

– 用于打开或关闭面板的三角形展开图标

– 要更改的建模类型图标

用于打开数据文件的选项

注意：有关打开文件的详细信息，请参见第 61 页的“导入数据”一章。

要打开某个数据集，请选择**文件 > 打开**，然后选择文件类型。某些文件类型在“打开数据文件”窗口中具有附加功能和选项（表 2.1）。

提示：

- 仅限 Windows：若要每次打开相同的文件类型，请选中**下次调用该窗口时选择该过滤条件**复选框。
- 将文件拖至“JMP 主窗口”可打开该文件。
- 要更改**文件 > 打开**命令所查看的目录，请参见“JMP 首选项”一章中第 615 页的 ““文件位置”的首选项”。

表 2.1 按文件类型划分的“文件 > 打开”选项

文件类型	附加功能和选项
JMP 数据表	显示表注释、列数和行数。 使用**选择列**选项可选择要导入数据表的列。在 Windows 上，点击**打开**旁边的箭头，然后使用“选择列”选项指定要导入数据表的列。在 macOS 上，点击**打开**后，您可以选择要导入的列。
Excel 文件	• 默认在“Excel 导入向导”中导入文件，可让您在导入该文件之前自定义布局和预览数据。 • 点击**打开**旁边的箭头，然后选择其中一个选项。将电子表格的首行转换为列标题。
文本文件	• 要自动确定数据排列，请选择以下选项之一： – **数据（使用首选项）** – **数据（最佳推测）** – **“脚本编辑器”中的文本** • 要手动指定数据排列，请选择**数据（数据预览）**选项。请参见“导入数据”一章中第 79 页的““JMP 文本导入向导”中的选项”。
SAS 数据集	• 将 SAS 变量用作列名 • （仅限 Windows）在您打开受密码保护的数据集时输入密码。 • （仅限 SAS 传输文件）在打开前选择列
SPSS 数据文件	将 SPSS 变量或标签名称用作列名。
XML 数据文件	有关选项的详细信息，请参见“文本文件”。
JSON 数据文件	有关选项的详细信息，请参见“文本文件”。
所有文件	• 要自动确定数据排列，请选择以下选项之一： – 以数据方式打开（使用首选项） – 以数据方式打开（使用最佳推测） – 在脚本窗口中打开纯文本 • 要手动指定数据排列，请选择**数据（数据预览）**选项。 • （仅限 Windows）要使用用户计算机默认的文本编辑器打开文本文件，请选择**使用默认程序打开**。

平台和报表

JMP 包含按照统计分析类型组织的多个平台。例如，“分布”平台使用直方图、其他图形和报表生成一元分析（单个变量的分布）。您可以在此“分布”平台中分析数据，然后选择其他平台更为彻底地检查数据。

平台和报表通过以下方式协同工作：

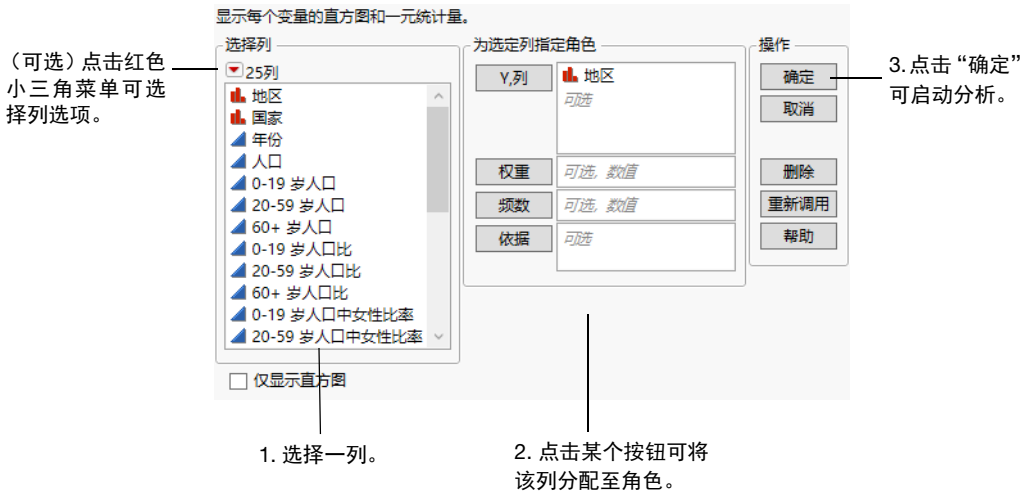
1. 数据表是平台的输入。请参见第 35 页的“数据表功能”。
2. 平台分析在启动窗口中启动。请参见第 49 页的 “平台中的启动窗口”。
3. 平台结果显示在报表窗口中。请参见第 51 页的 “报表中的分析结果”。

第 35 页的图 2.2 “剖析 Windows 上的典型 JMP 会话” 显示平台如何与数据表交互，然后在报表窗口中显示结果。JMP 文档库中提供了关于使用平台的信息。

平台中的启动窗口

启动窗口是您进入平台的入口点，您可在其中指定要分析的列。图 2.11 显示“分布”启动窗口。

图 2.11 启动窗口功能



所有启动窗口都共同提供以下选项：

选择列 列出当前数据表中的所有变量。请注意以下事项：

- 右击列名旁边的建模类型图标可更改建模类型。
- 右击列名可创建变换列。请参见“输入和编辑数据”一章中第 222 页的“变换 JMP 平台中的列”。

– 使用红色小三角菜单中的选项可对列进行过滤和排序。请参见第 50 页的““列过滤器”菜单”。

为选定列指定角色 将选定列移入角色 （如 Y、 X 等。）为列指定变量角色 （就像让演员饰演某个角色）。请参见第 50 页的 ““为选定列指定角色”按钮”。

在 “图形生成器”窗口中，点击左侧的**对话框**按钮，或选择**重新运行 > 重新启动分析**以显示该面板。

操作 以下选项可用：

确定 执行分析。

取消 停止分析并退出启动窗口。

删除 从角色中删除任何选定的变量。

重新调用 用上次执行的分析填充启动窗口。

帮助 将您引导到启动窗口的帮助。

提示：要重新打开上次打开的平台启动窗口，请选择**视图 > 重新调用最后一个平台** (Windows) 或**窗口 > 重新调用最后一个平台** (macOS)。

“为选定列指定角色”按钮

以下按钮在启动窗口中频繁出现。专用于特定平台的按钮将在特定平台的章节中专门介绍。

Y 将某列标识为要研究其分布的响应变量或因变量。

X 将某列标识为自变量、分类变量或说明变量，以预测 Y 变量的分布。

权重 标识数据表列，该列的变量为数据分配权重 （如重要性或影响）。

频数 标识数据表列，该列的值为每行分配频数。在为汇总数据中的每行分配频数时，该选项会很有用。若值为 0 或一个正整数，则在记录多个单位的情况下，该值表示每行的观测频数或计数。

JMP PRO **验证** 标识数据表列，该列的值为拟合模型中进行交叉验证的训练、验证和测试集分配行。

注意：

– 在一些平台中，若您在“验证”列中指定了三个以上的水平，则“K 重”验证可用。

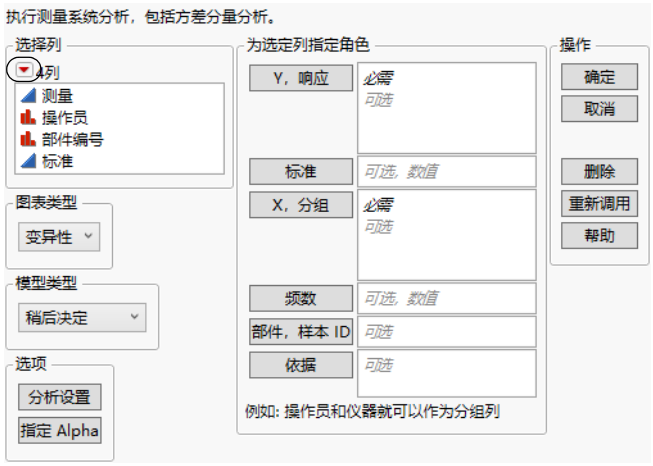
– 若在“选择列”列表中没有选择任何列的情况下点击“验证”按钮，您可以选择创建“验证”列。有关“生成验证列”实用工具的详细信息，请参见《预测和专业建模》。

依据 标识一列，该列创建的报表包含变量每个水平的单独分析。

“列过滤器”菜单

“列过滤器”菜单在多数启动窗口中都显示。通过点击“选择列”面板中的红色小三角菜单可访问“列过滤器”菜单。使用这些选项可以对列排序、显示或隐藏列，或搜索列。

图 2.12 “列过滤器”菜单



重置 将列重置为其原始列表。

显示过滤器 打开一个框，在其中键入要查找的列名。点击该框旁边的箭头并选择选项以定制该过滤器。

按名称排序 按名称的字母顺序以升序、降序或原始顺序对列进行排序。

建模类型 提供用于显示或隐藏具有特定建模类型的列的选项。

数值 显示或隐藏数据类型为“数值”的列。

字符 显示或隐藏数据类型为“字符”的列。

表达式 显示或隐藏数据类型为“表达式”的列。

排除格式 从列选择列表中排除具有特定格式的列。

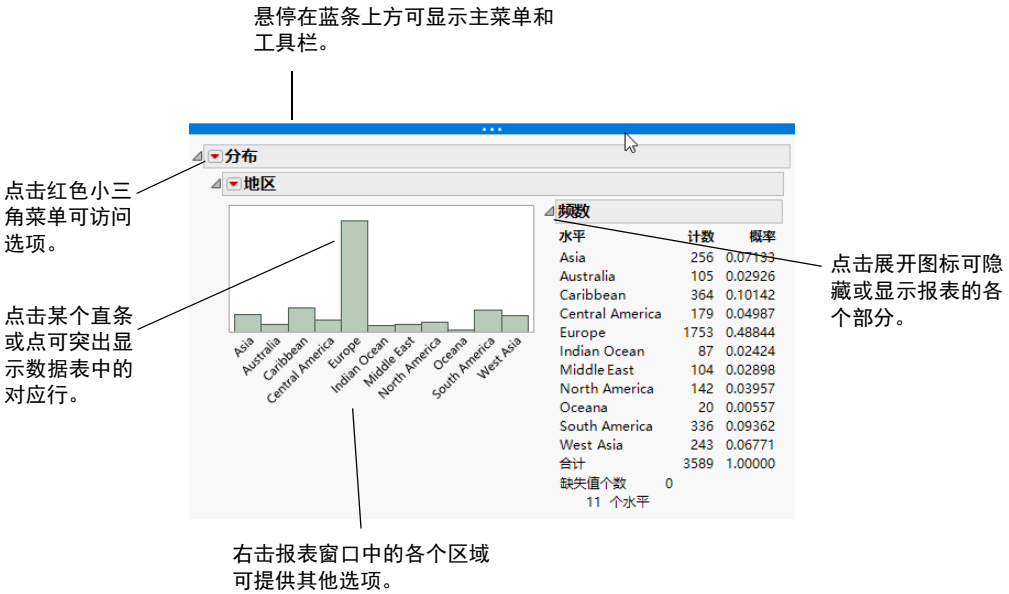
列组 显示或隐藏列组。请参见“输入和编辑数据”一章中第 187 页的“组合数据表中的列”。

未分组列 显示或隐藏未分组的列。

报表中的分析结果

启动分析后，即显示 JMP 报表窗口。报表窗口使用交互式图形和文本报表显示分析的输出。请参见第 383 页的“JMP 报表”一章。

图 2.13 报表窗口功能



管理文件和打开窗口

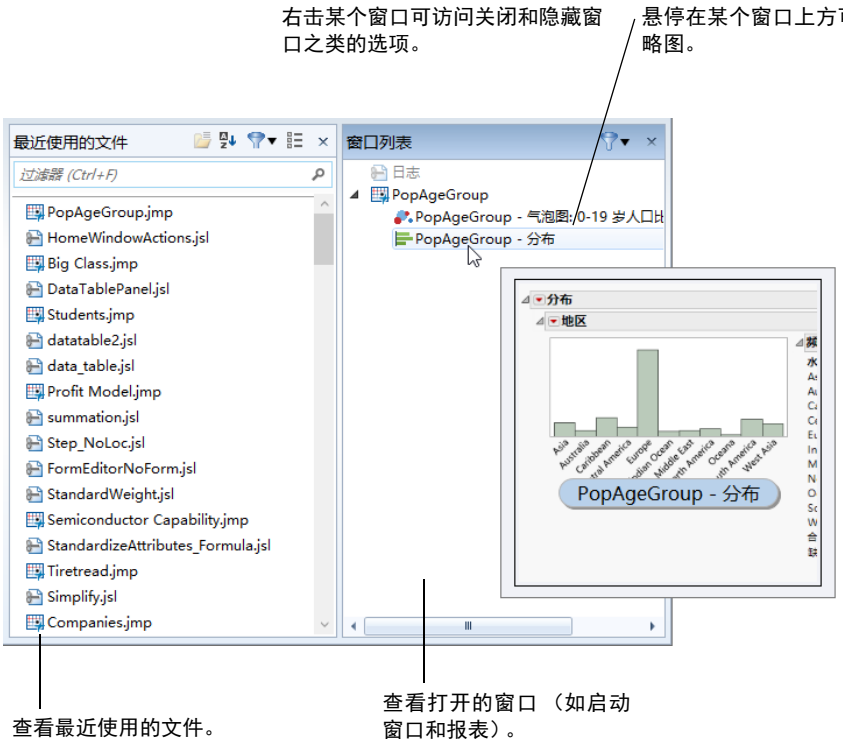
“JMP 主窗口”提供对打开的窗口和文件的即时访问。打开最近打开的文件、关闭窗口和固定常用的文件均为其中的选项。

Windows 系统中的“JMP 主窗口”

在 Windows 上，打开 JMP 时将显示“JMP 主窗口”。

- 打开“最近使用的文件”列表中最近使用的文件或固定常用文件。
- 右击“最近使用的文件”列表中的文件执行常见任务。
- 打开或关闭“窗口列表”中的活动 JMP 窗口。
- 通过从顶部的数据表列表中选择打开的数据表以将其设置为当前数据表。

图 2.14 “JMP 主窗口”操作



“JMP 主窗口”按钮

在 Windows 上，“JMP 主窗口”面板可包含以下按钮：

“打开选定项”按钮用于打开 “最近使用的文件”列表中选定的文件。

“按名称排序”按钮用于按字母顺序对最近使用的文件排序。

“过滤器”按钮用于对 “最近使用的文件”和 “窗口列表”面板中出现的文件类型进行过滤。

加大 “最近使用的文件”列表中项的字体大小和图标大小。

“关闭”按钮用于关闭面板。

最近使用的文件

在 Windows 上，“最近使用的文件”列表提供对最近打开的文件的快速访问。

将常用文件保留在列表顶部以便于快速访问。悬停在文件名上方并点击固定图标 。要取消固定某个文件，请点击固定图标 。

您可以将文件从“最近使用的文件”列表拖入以下位置：

- 拖入“JMP 窗口列表”以打开该文件
- 拖入 Windows 文件夹或您的桌面上以创建该文件的副本
- 拖入 Microsoft 记事本或 Microsoft Word 之类的应用程序以编辑该文件

右击文件可访问以下选项：

- 打开一个文件
- 将位置路径复制到文件
- 在包含文件的文件夹中打开该文件
- 按名称或最近打开的文件的字母顺序对文件排序
- 从列表中删除所选的一个或多个文件
- 删除不再位于同一位置的一个或多个文件（也可以选择这些文件并按 Delete 键。）
- （JSL 脚本）编辑、运行或调试脚本
- （文本文件）更改导入方法或以纯文本方式打开
- （SQL 查询）运行您保存在“查询生成器”中的 SQL 查询。您还可以使用现有查询作为模板来创建新的查询。
- 运行或编辑 JMP 应用程序

  从“最近使用的文件”列表中打开非 JMP 文件时，JMP 使用导入首选项来排列数据。您可以通过右击某一文本文件，更改导入方法或在文本编辑窗口中打该文件。导入首选项在弹出式菜单中以粗体显示。

提示：要防止脚本打开的文件出现在“最近使用的文件”窗口中，请选择**文件 > 首选项 > 常规**，并取消选择**将脚本打开的文件添加到“最近使用的文件”列表**。

在“最近使用的文件”列表中搜索 JMP 文件

要在较长的文件列表中查找最近打开的文件，可在“最近使用的文件”的“过滤器”框中输入部分文件名。

提供以下选项：

- 在文件列表上方的“过滤器”框中输入部分文件名。
- 悬停在“过滤器”框上方可查看搜索提示。
- 过滤文件时，上箭头键或下箭头键可更改选定的文件。按 Enter 键可打开选定的文件，也可以使用光标选择文件。
- 若在“最近使用的文件”列表中选定了某个文件，按 Ctrl+F 即可激活“过滤器”框。
- 键入搜索项后，按 Esc 或点击 X 按钮可退出过滤模式。

窗口列表

在 Windows 上，“窗口列表”显示打开的 JMP 窗口，如数据表、报表和脚本。您可以打开、关闭、重新排列和隐藏 JMP 窗口。

- 若悬停在“窗口列表”中的某个文件上方，此时将出现一个缩略图。
- 要单独在一个窗口中打开“窗口列表”，请选择**视图 > 窗口列表**。
- 要并排显示各个窗口，请右击选定的窗口，然后选择**排列**。
- 若要始终在最大化窗口中自动显示“窗口列表”，请选择**文件 > 首选项 > Windows 特定**，然后选择**在最大化窗口中停放窗口列表**。

右击文件可访问以下选项：

- 查看窗口
- 关闭窗口（也可以选择文件并按 Delete 键。）
- 关闭除当前选定窗口之外的所有窗口（若所选窗口是一个报表窗口，相关的数据表也会保持打开）
- 将某窗口移到最后
- 在 Windows 任务栏中隐藏窗口或取消隐藏窗口
- 选择所有窗口，或清除所有选择内容

注意：当前文件的格式设置为加粗文本。

在主窗口中重新排列面板

要在 Windows 上的主窗口中重新排列面板，请点击并拖动面板的标题栏。将标题栏放到上箭头、下箭头、左箭头或右箭头上，对面板定位。会有蓝框指示面板要放到的位置。要将面板转换为选项卡，请将任意面板拖放到主窗口的中央。

提示：

- 要使主窗口恢复原始排列顺序，请选择**视图 > 主窗口窗格 > 恢复为出厂布局**。
- 若在关闭某面板后又想重新打开，请选择**视图 > 主窗口窗格**，然后选择要打开的面板。
- 要打开“JMP 主窗口”，可以通过选择**视图 > 主窗口**，或点击多数 JMP 窗口右下角的 **JMP 主窗口** 按钮。若看不到 **JMP 主窗口**按钮，请选择**视图 > 状态栏**。
- 若希望在启动后看到“JMP 起始页”或“窗口列表”，可以在“首选项”中进行相关指定（仅限 Windows）。选择**文件 > 首选项 > 常规**，然后从初始 JMP 窗口列表中选择一个选项。

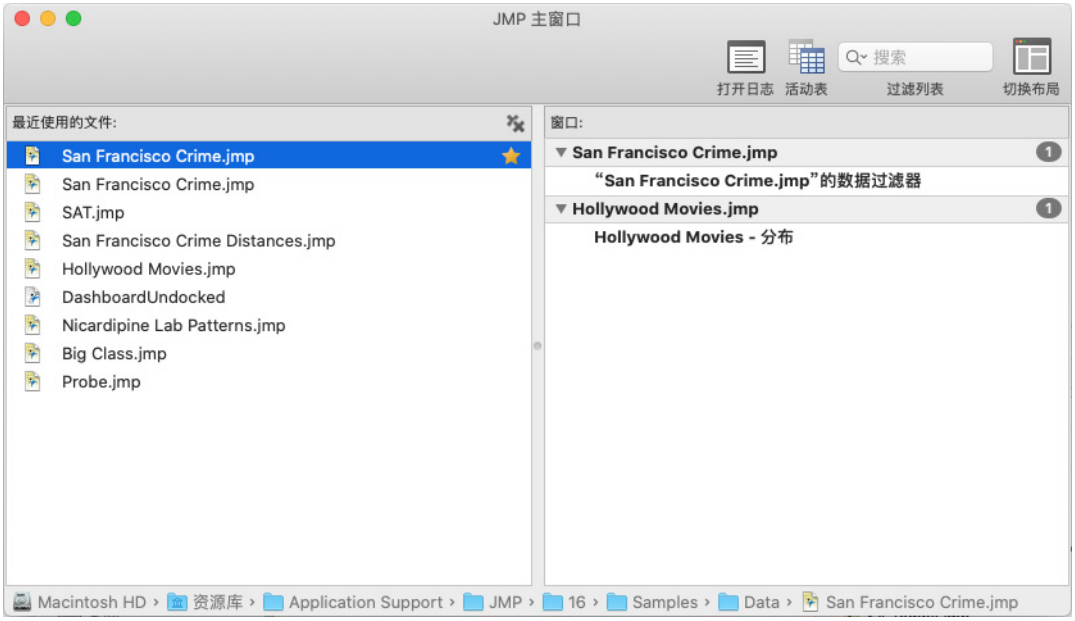
macOS 上的“JMP 主窗口”

使用 macOS 上的“JMP 主窗口”可快速访问 JMP 文件和打开的窗口。

- 打开“最近使用的文件”列表中最近使用的文件。
- 按 Ctrl 键并选择“最近使用的文件”列表中的文件可执行常见任务，如添加或删除收藏项。
- 打开或关闭“窗口”列表中的活动 JMP 窗口。
- 通过从“活动表”列表中选择打开的数据表以将其设置为当前数据表。按 Option 键的同时从列表中选择数据表以激活该数据表，而不将其置前。
- 打开日志。
- 通过在“搜索”框中输入搜索词，在“最近使用的文件”列表中查找文件或在“窗口”列表中查找打开的窗口。
- 点击**切换布局**以水平而不是垂直方式显示“最近使用的文件”和“窗口”列表。

在 macOS 上通过选择**窗口 > JMP 主窗口**，可以打开“JMP 主窗口”（图 2.15）。要在每次打开 JMP 时查看该窗口，请选择 **JMP > 首选项 > 常规 > 主窗口**。

图 2.15 “JMP 主窗口”示例 (macOS)



提示：

- 状态栏区域显示当前选定文件的路径。您可以点击某个文件夹以打文件夹查看其内容。
- 要快速从任意 JMP 窗口查看“JMP 主窗口”，请按 Ctrl 键并点击该窗口，选择**定制工具栏**，然后将“JMP 主窗口”图标拖至工具栏。
- 要将文件从“最近使用的文件”复制到另一个文档或 Finder，请选择该文件，然后选择**编辑 > 复制**。

最近使用的文件

在 macOS 上，“最近使用的文件”列表提供对最近打开的文件的快速访问。

您还可以点击悬停在文件名上方时出现的星形将该文件添加到您的收藏夹。

按 Ctrl 键的同时选择某个文件可访问其他选项。例如，通过在按 Ctrl 键的同时选择“运行脚本”或“调试脚本”，您可以运行或调试 JSL 脚本。

提示：要防止脚本打开的文件出现在“最近使用的文件”窗口中，请选择 **JMP > 首选项 > 常规**，并取消选择**将脚本打开的文件添加到“最近使用的文件”列表**。

“窗口”列表

在 macOS 上，“窗口”列表显示打开的 JMP 窗口，如数据表、报表和脚本。

要关闭某个窗口，请点击当您悬停在“窗口”列表中该窗口上方时所显示的 x 按钮。

若您要关闭的窗口是一个或多个子窗口的父窗口，则系统会提示您**隐藏数据表**、**取消**或**全部关闭**窗口。选择**全部关闭**以关闭父窗口及其所有子窗口。

在 Windows 上搜索最近打开的文件

在 Windows 上，最近打开的文件显示在主窗口中。您还可以从任意 JMP 窗口搜索最近打开的文件。

选择**文件 > 快速打开**或按 Alt+Shift+O 可打开“过滤器”框并输入您的搜索项。该选项还支持您查看文件路径。键入搜索项后，按 Esc 或点击 X 按钮可关闭该窗口。

关闭多个文件

在 Windows 上，可从“JMP 主窗口”一次关闭若干文件。选择“窗口列表”中的文件，右击并选择**关闭**。要保持单个文件打开而关闭其他文件，请选择这个文件，然后选择**关闭其他所有**。

在 macOS上，选择**文件 > 关闭多个**，然后选择您要关闭的文件。

显示和排列打开的窗口

在 JMP 中，通常同时会打开多个窗口（例如，数据表、报表和“JMP 主窗口”）。JMP 提供多种方式来排列和显示这些打开的窗口。

在 Windows 上：

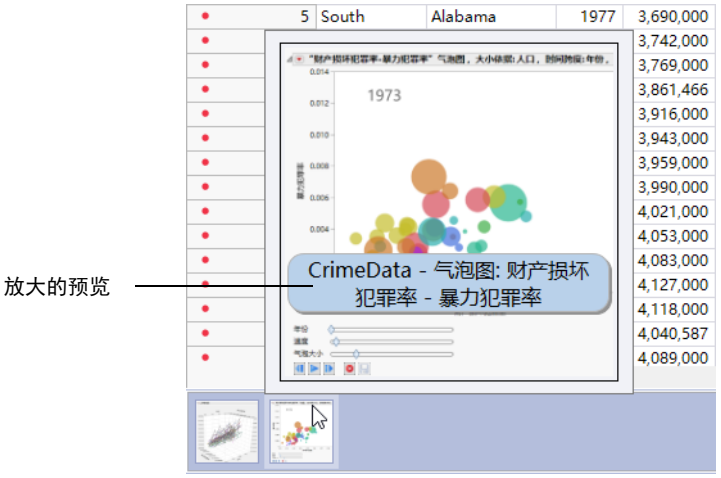
- 按 Ctrl+Tab 可在窗口之间切换。
- 使用“显示”功能（F9 或“窗口 > 显示”）。请参见第 59 页的“使用窗口显示功能”。

- 使用窗口菜单中的选项。请注意，使用多数窗口右下角的“排列菜单”选项也可以提供排列选项。若看不到“排列菜单”、“JMP 主窗口”和“查看相关的数据”这些按钮，请选择**视图 > 状态栏**。

提示：从 Windows 上的“排列菜单”选项中，您可以通过选择“组合窗口”合并打开的窗口。请参见“使用仪表板组合和演示报表”一章中第 553 页的“示例：组合窗口以创建仪表板”。

- 要显示“JMP 主窗口”，请点击多数窗口右下角都有的 **JMP 主窗口** 按钮，或按 Ctrl+1。
- 要显示报表对应的数据表，请点击报表右下角的**查看相关的数据** 按钮。
- 要在查看数据表时打开相关的报表，请在底部窗格中双击该报表的缩略图预览。要放大该预览，请悬停在缩略图上方（图 2.16）。
- 按鼠标上的中间点击按钮，点击缩略图预览以关闭关联的报表。

图 2.16 打开的报表的缩略图预览



在 macOS 上：

- 使用窗口菜单中的选项。
- 使用“显示”功能（F9 或 F3）。

预览 JMP 文件

对于某些 JMP 文件类型，您可以使用“Windows 资源管理器”或 macOS Finder 查看选定文件的一部分。Windows 支持 JMP 数据表、记录和脚本的预览。macOS 支持数据表和脚本的预览。

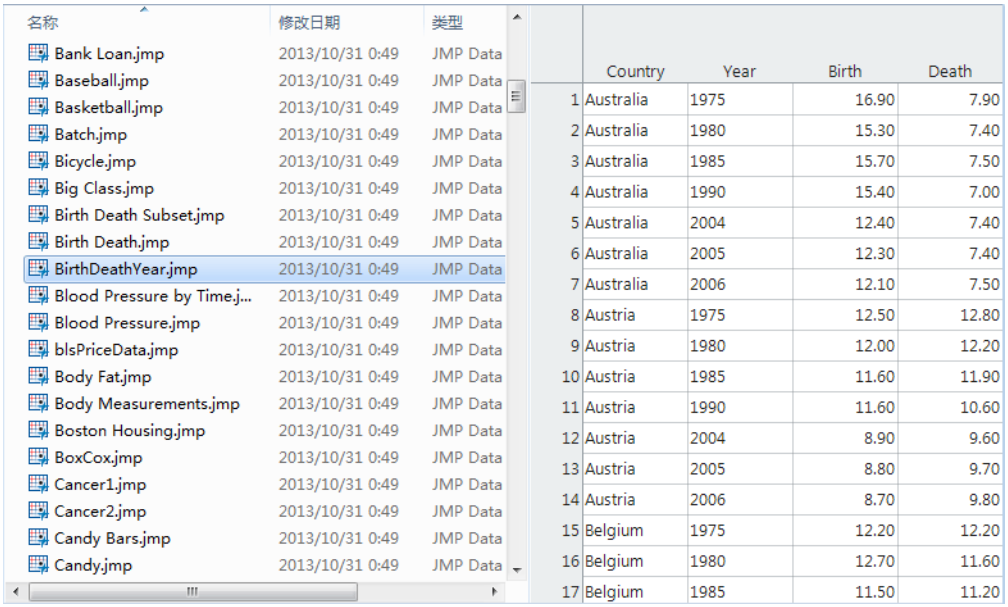
要预览 JMP 文件，请执行以下操作：

1. 在“Windows 资源管理器”或 macOS Finder 中，选择您要预览的文件。
2. 在 “Windows 资源管理器”中，选择右上角的**显示预览窗格** 按钮。对电子邮件程序的支持有所不同。

   在 macOS Finder 中，选择文件并按下空格键，或者选择弹出式菜单或“文件”菜单中的**快速查看**选项。在 Mail 中，选择弹出式菜单或 “文件”菜单中的**快速查看附件**选项。

   预览窗格显示选定文件的一部分。

图 2.17 数据表预览窗格 (Windows)



名称	修改日期	类型
Bank Loan.jmp	2013/10/31 0:49	JMP Data
Baseball.jmp	2013/10/31 0:49	JMP Data
Basketball.jmp	2013/10/31 0:49	JMP Data
Batch.jmp	2013/10/31 0:49	JMP Data
Bicycle.jmp	2013/10/31 0:49	JMP Data
Big Class.jmp	2013/10/31 0:49	JMP Data
Birth Death Subset.jmp	2013/10/31 0:49	JMP Data
Birth Death.jmp	2013/10/31 0:49	JMP Data
BirthDeathYear.jmp	2013/10/31 0:49	JMP Data
Blood Pressure by Time.j...	2013/10/31 0:49	JMP Data
Blood Pressure.jmp	2013/10/31 0:49	JMP Data
blsPriceData.jmp	2013/10/31 0:49	JMP Data
Body Fat.jmp	2013/10/31 0:49	JMP Data
Body Measurements.jmp	2013/10/31 0:49	JMP Data
Boston Housing.jmp	2013/10/31 0:49	JMP Data
BoxCox.jmp	2013/10/31 0:49	JMP Data
Cancer1.jmp	2013/10/31 0:49	JMP Data
Cancer2.jmp	2013/10/31 0:49	JMP Data
Candy Bars.jmp	2013/10/31 0:49	JMP Data
Candy.jmp	2013/10/31 0:49	JMP Data

	Country	Year	Birth	Death
1	Australia	1975	16.90	7.90
2	Australia	1980	15.30	7.40
3	Australia	1985	15.70	7.50
4	Australia	1990	15.40	7.00
5	Australia	2004	12.40	7.40
6	Australia	2005	12.30	7.40
7	Australia	2006	12.10	7.50
8	Austria	1975	12.50	12.80
9	Austria	1980	12.00	12.20
10	Austria	1985	11.60	11.90
11	Austria	1990	11.60	10.60
12	Austria	2004	8.90	9.60
13	Austria	2005	8.80	9.70
14	Austria	2006	8.70	9.80
15	Belgium	1975	12.20	12.20
16	Belgium	1980	12.70	11.60
17	Belgium	1985	11.50	11.20

使用窗口显示功能

“显示”功能可显示一个顶级窗口，其中显示所有打开的 JMP 窗口的缩略图。在 Windows 上，按 F9 可使用“显示”功能。

- 点击某一窗口可将其激活。
- 按空格键可打开“预览”模式，光标之下缩略图的全尺寸视图将出现在屏幕中央。
- 默认情况下，各个窗口依据屏幕上的原始排序从上到下、从左到右排序。按 S 可按名称对这些缩略图排序。

  要退出 “显示”功能，请按 F9、 Esc 或 Alt+Tab，然后选择其他应用程序。

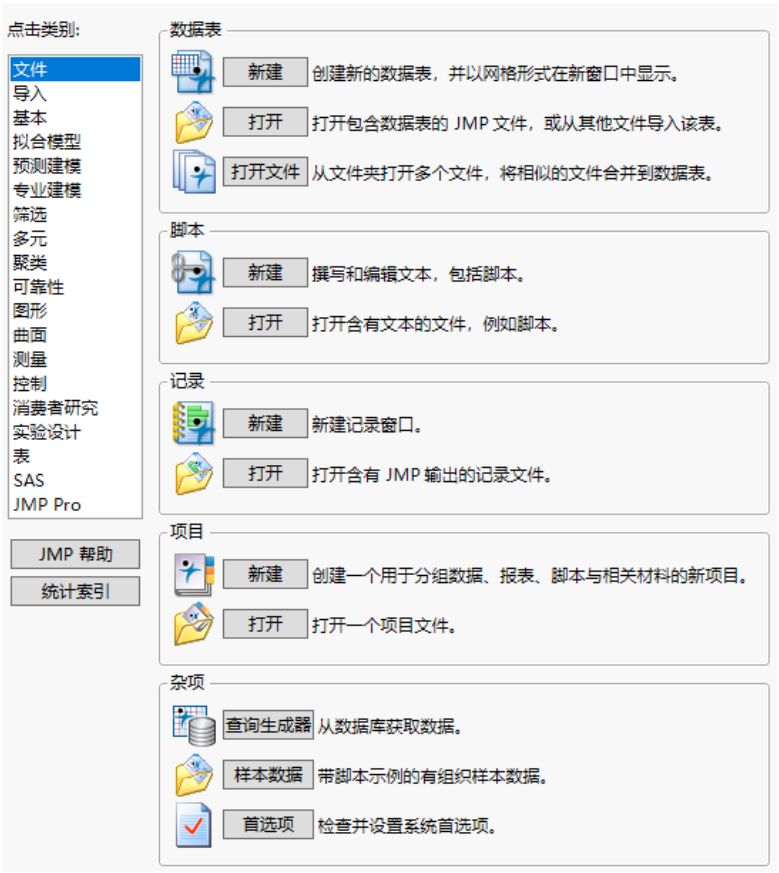
JMP 起始页窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，在 **JMP** > **首选项** > **常规**页面中选择该选项。

图 2.18 显示 Windows 中的“JMP 起始页”窗口。macOS“JMP 起始页”窗口完全相同。

图 2.18 “JMP 起始页”窗口 (Windows)



第 3 章

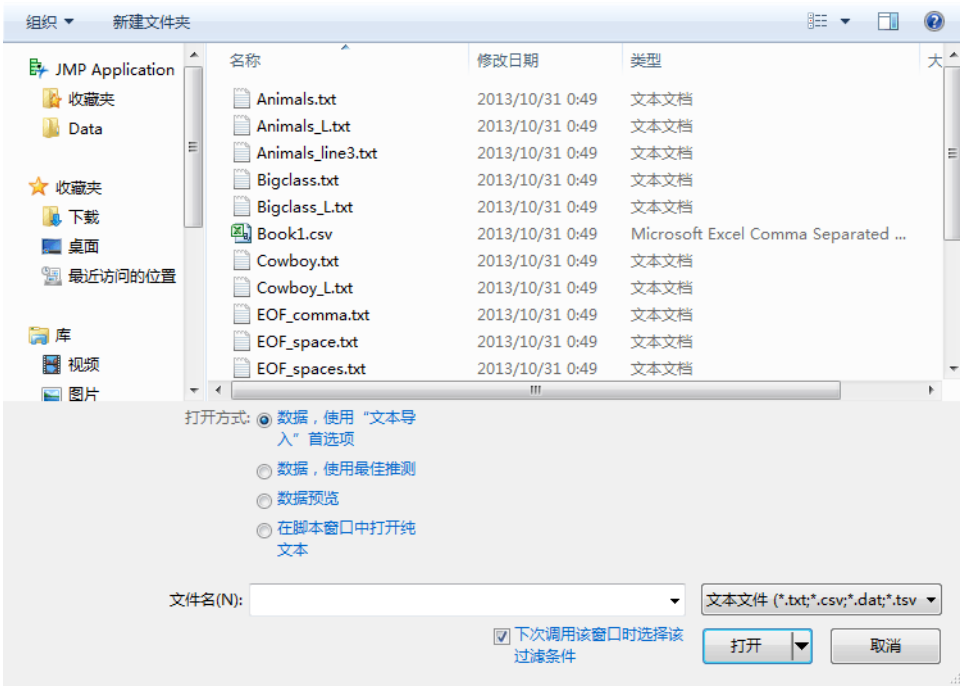
导入数据

创建数据表

本章包含以下主题：

- 如何导入数据，如文本文件、PDF 文件、SPSS 文件、SAS 数据
- 如何传输 Excel 数据
- 如何读入实时数据
- 如何新建数据表

图 3.1 导入文本文件



目录

关于导入数据 64
导入 Microsoft Excel 文件 65
预览并导入 Microsoft Excel 数据 65
直接导入 Microsoft Excel 文件 76
导入文本文件 77
查找导入数据中的缺失字符 79
“JMP 文本导入向导”中的选项 79
将文本导入 JMP 之前对其进行编辑 83
从“脚本编辑器”导入文本 85
导入多个文件 86
导入 PDF 文件 90
将多个表拼接为一个数据表 90
按列标题拼接表 92
导入 JMP 未能自动检测的表 93
“PDF 预览”红色小三角选项 95
数据表预览选项 96
PDF 预览右击选项 96
导入 XML 和 JSON 文件 96
预览和导入 XML 和 JSON 数据 96
直接导入 XML 文件 102
直接导入 JSON 文件 103
在“查询生成器”中生成 SQL 查询 104
连接至 SQL 数据库 104
从 SQL 数据库中选择表 105
生成 SQL 查询 109
保存和运行 JMP 查询 121
打开选定的数据 122
“查询生成器”红色小三角选项 122
在“查询生成器”中编写 SQL 语句 123
从 SAS 导入数据 124
Java 运行时环境 (JRE) 要求 124
SAS 菜单选项 124
导入 SAS 数据集 124
通过 JMP 在 SAS 中创建 SAS 传输文件 126
从 JMP 连接至 SAS 127
使用“SAS 查询生成器”打开 SAS 数据集 132
通过 SAS 服务器打开 SAS 数据集 133

使用 SAS 扩展属性导入元数据 140
运行存储过程 141
提交 SAS 代码 142
生成 ODS 结果 143
导入生成的 SAS 数据集 144
导入数据库中的数据 146
连接至数据库 147
从数据库中打开数据 147
编写 SQL 语句以查询数据库 149
结构化查询语言 (SQL)：参考 150
从 Google Sheets 导入数据 155
关于导入 Google Sheets 156
Google Sheet 设置 157
导入 SPSS 文件 157
导入 3S 调查数据 158
导入 HDF5 数据 159
从 MATLAB 导入数据 159
从 R 导入数据 160
使用 Excel 插件导入数据 160
将 Excel 数据传输至数据表 160
关于 Excel 的 JMP 插件 161
卸载 JMP Excel 插件 162
从 Internet 或远程计算机导入数据 162
创建新数据表 163

关于导入数据

您可以将多种文件格式导入 JMP 并将其另存为数据表。JMP 默认可打开许多文件。JMP 在默认情况下不支持的文件格式需要特定的开放式数据库连接 (ODBC) 驱动程序。

默认支持以下文件格式：

- 逗号分隔文件 (.csv)
- 包含文本的 .dat 文件
- ESRI 形状文件 (.shp)
- 流式细胞术版本 2.0 和 3.0 文件 (.fcs)
- 分层数据格式，版本 5 (.h5)
- HTML（.htm、.html）
- JSON (.json)
- MATLAB（.m、.M）
- 基于 macOS 的 Microsoft Excel 1997 - 2019（.xls、.xlsx）
- 基于 Windows 的 Microsoft Excel 2010 - 2019（*.xlsx、*.xlsm）
- Minitab 可迁移工作表 (.mtp)
- 纯文本 (.txt)
- R (.r)
- SAS 传输文件（.xpt、.stx）
- 基于 macOS 的 SAS 版本 7 - 9（.sas7bdat）
- 基于 Windows 的 SAS 版本 7 - 9（.sas7bdat、sas7bxat）
- SPSS (.sav)
- SQLite 3.0 或更高版本（.sqlite、.db、.sqlite3、.db3）
- 制表符分隔文件 (.tsv)
- Teradata 数据库 (.trd)
- 3S（.sss、.xml）
- xBase 数据文件 (.dbf)
- XML 数据文件 (.xml)

关于 SAS 支持的注释：

在 Windows 和 macOS 上，您可以通过**文件 > 打开**命令直接打开 SAS 数据集。请参见第 124 页的“导入 SAS 数据集”。

另一个选项是通过选择**文件** > SAS > **浏览数据**连接到 SAS 服务器。请参见第 133 页的“通过 SAS 服务器打开 SAS 数据集”。

以下文件需要 ODBC 驱动程序：

- 需要使用兼容 V3+ 的 ODBC 驱动程序支持数据库 (dBASE)（.ndx、.mdx）。.dbf 文件不需要 ODBC 驱动程序。
- 需要使用兼容 V3+ 的 ODBC 驱动程序支持 Microsoft Access 数据库 (.mdb)。

有关使用数据库的详细信息，请参见第 146 页的“导入数据库中的数据”。

用户计算机上可用的内存空间将影响数据导入。极大的文件可能加载较慢或根本无法加载。请考虑将大型文件拆分后再导入。您随后可以连接或拼接各表。请参见“重塑数据”一章中第 309 页的“拼接数据表”和“重塑数据”一章中第 312 页的“连接数据表”。

注意：您可以在 JMP 中打开 R 代码 (.R) 和 SAS 程序文件 (.sas)，但文本将在“脚本”窗口中打开，而不是在数据表中打开。

导入 Microsoft Excel 文件

Microsoft Excel 文件默认在“Excel 导入向导”中打开。该向导中显示数据预览。您可以随后修改这些设置，然后再导入数据。例如，您可以指示从哪行开始导入数据，以及工作表是否包含列标题或隐藏的行或列。Microsoft Excel .xls、.xlsm 和 .xlsx 文件格式均受支持。

有关在向导外打开 Microsoft Excel 文件的信息，请参见第 76 页的“直接导入 Microsoft Excel 文件”。

注意：

- 在 Windows 和 macOS 两个系统中，小数点后面的位数以及导入数据的日期格式可能会有所不同。例如，“10/25/2012”在 macOS 中可能会以“25Oct2012”的格式出现。在 macOS 中，列可能会被导入为字符列，但在 Windows 中则不会。
- 考虑将“常规”首选项中的**自动保存超时**值设置为自动保存打开的数据表。该自动保存值也适用于其他 JMP 文档类型。

预览并导入 Microsoft Excel 数据

导入工作表之前，请在 Excel 中打开电子表格并决定在最终的数据表中结构化数据的方式。例如，您需要知道该工作表是否包括隐藏或合并的单元格。在该向导中，您可以随后排除隐藏的列或行。

要导入包含若干工作表的 Microsoft Excel 文件，请执行以下步骤：

1. 在 Microsoft Excel 中打开工作表。

对于该示例图中显示的数据，我们使用的是 JMP Samples/Import Data 文件夹中的 Team Results.xlsx 文件。该文件具有以下特征：

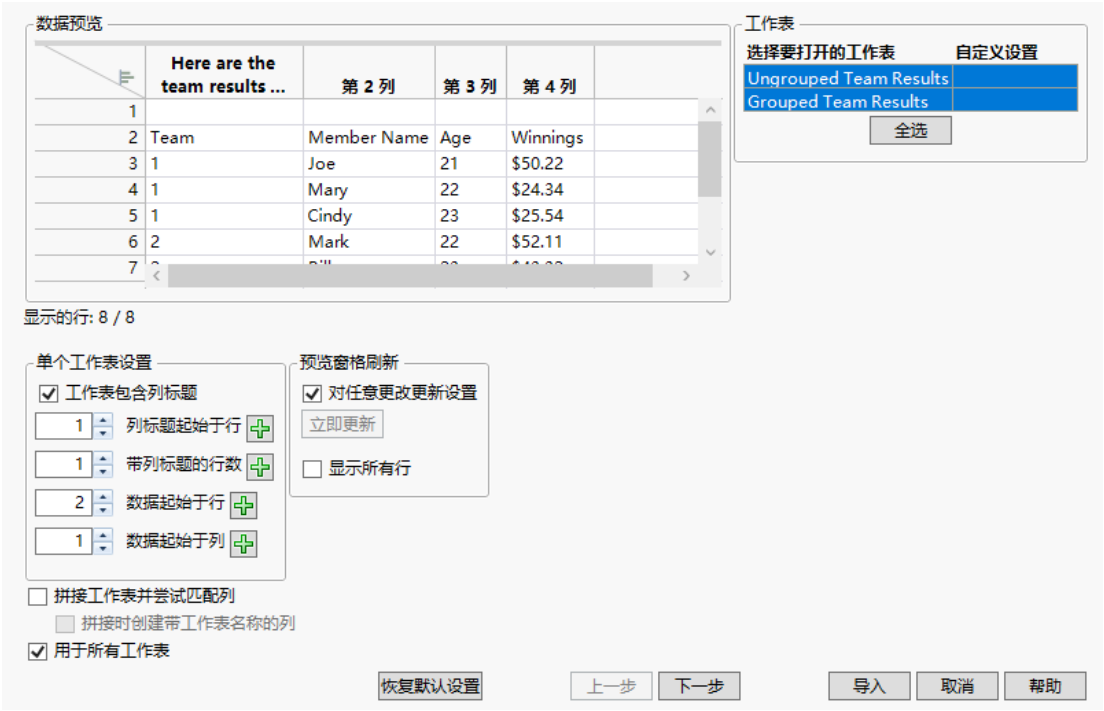
- 数据开始于第 4 行第 2 列，结束于第 9 行第 5 列
- 两个工作表
- 第二个工作表包含两组合并的单元格
- 没有隐藏的行或列

图 3.2 Team Results.xlsx 工作表

A	B	C	D	E	F
	Here are the team results for this year's game.				
	Team	**Member Name**	**Age**	**Winnings**	
	1	Joe	21	$50.22	
	1	Mary	22	$24.34	
	1	Cindy	23	$25.54	
	2	Mark	22	$52.11	
	2	Bill	23	$43.32	
	2	Jennifer	24	$11.23	

2. 要打开 Excel 文件，请选择**文件 > 打开**。
   随即显示 “打开数据文件”窗口。
3. 选择 “文件名”旁边的 **Excel 文件**。
4. 选择 Excel 文件并点击**打开**。
   该工作表在 “Excel 导入向导”中打开，向导中显示有数据预览以及导入选项。

图 3.3 初始数据预览示例



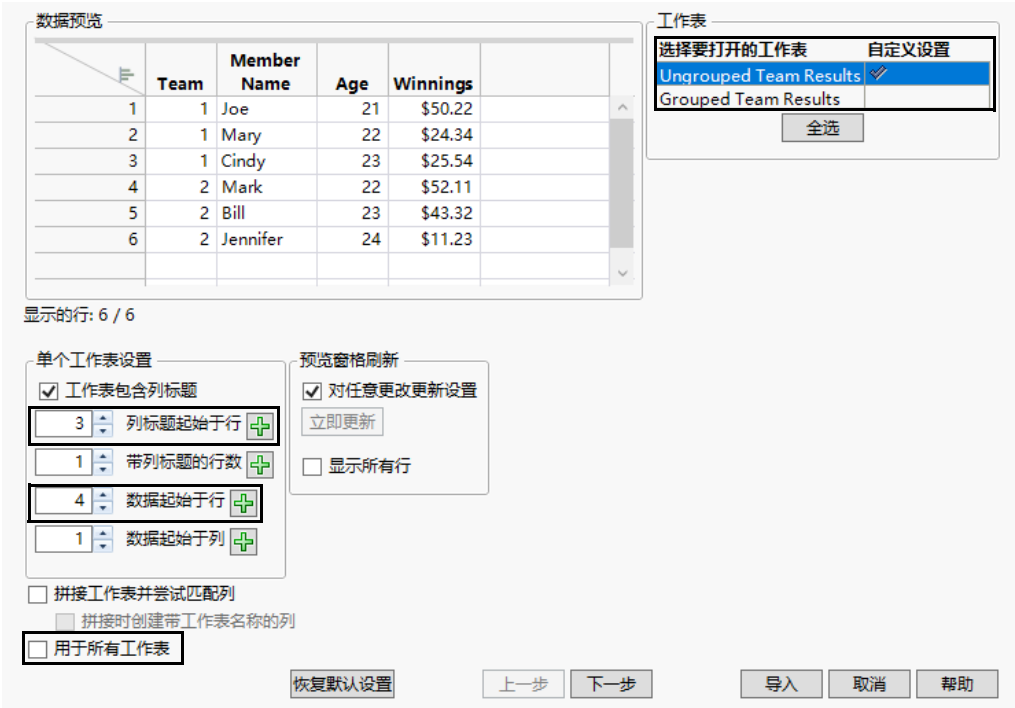
请注意 “数据预览”中的以下特征：

- 在右上角中两个工作表都选定进行导入。
- 第一列已自动删除。
- 工作表第一行中的文本显示为列标题。但是，您想工作表第 3 行中的文本用作列标题。
- 第一个数据行为空。

注意：每次您导入工作表时， JMP 都会记住您之前的更改，即使是在关闭再重新打开 JMP 后也是如此。当您需要多次重新导入同一工作表而且需要试用不同选项时，该功能很有用。要在导入其他工作表时清除这些更改，请点击**恢复默认设置**。

5. 为**列标题起始于行**键入 3。
6. **数据起始于行**更改为 4。
7. 选择 “工作表”窗格中的 Ungrouped Team Results。
   将只导入该工作表。
8. 取消选择**用于所有工作表**。
   这些设置仅应用于 “Ungrouped Team Results”。

图 3.4 选择列标题行



提示：右击“数据预览”窗格中的数值列标题以更改该格式。

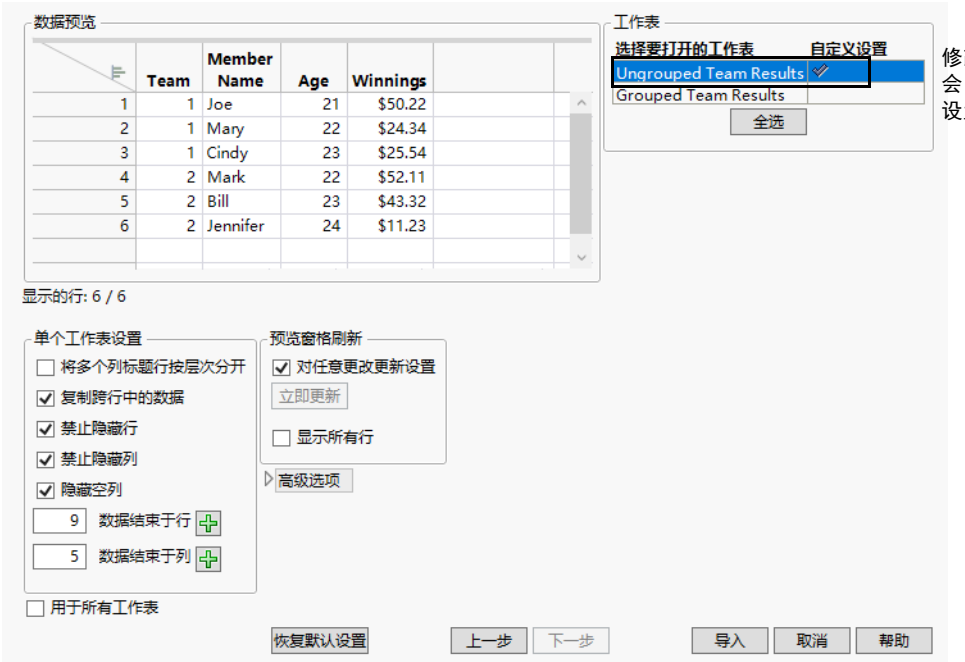
有关所有选项的详细信息，请参见第 69 页的 “单个工作表设置”。

9. 点击**下一步**配置其他导入设置。

   该窗口显示其他导入设置。

10. 为**数据结束于行**键入 9。
11. 为**数据结束于列**键入 5。

图 3.5 指定最后一列

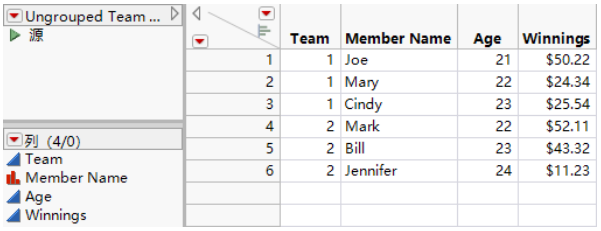


修改导入选项时会自动选定定制设置。

有关所有选项的详细信息，请参见第 70 页的 “其他单个工作表设置”。

12. 点击**导入**按您的指定转换工作表。

图 3.6 最终数据表



以下部分说明了“Excel 导入向导”中的选项。

单个工作表设置

工作表包含列标题 选择工作表是否包含带列标题的行。

列标题起始于行 指示工作表中列标题起始于哪一行。右击向上箭头，直到标题起始于正确的行为止，或者输入行号，然后按 Enter 键。

带列标题的行数 指示工作表是否包含多行作为列标题。点击向上箭头，直到标题行显示正确为止，或者输入行数，然后按 Enter 键。

数据起始于行 指示工作表中数据起始于哪一行。

数据起始于列 指示工作表中数据起始于哪一列。

拼接工作表并尝试匹配列 将所有工作表合并为一个数据表。 JMP 匹配包含相同标题的列。

拼接时创建带工作表名称的列 添加一个新的**源表**列，列出每个导入表的工作表名称。在您选择之前的拼接选项后，该选项才可用。

用于所有工作表 将当前导入设置应用于在右上角选定的所有工作表。

其他单个工作表设置

将多个列标题行按层次分开 指示工作表包含多个行作为列标题，并且您要将这些标题按层次分开。该选项仅适用于可堆叠数据。

复制跨行中的数据 指示在工作表中跨行合并单元格。 JMP 将取消跨越单元格并将单元格内容复制到所有生成的单元格中。默认选定该选项。

若取消选择**复制跨行中的数据**， JMP 将取消跨越单元格并将单元格内容复制到最顶端的单元格。其余的取消跨越单元格将保留为空。

禁止隐藏行 阻止隐藏的行出现在数据表中。

禁止隐藏列 阻止隐藏的列出现在数据表中。

隐藏空列 指示是否导入包含列标题的空列。取消选择该选项可导入列。

数据结束于行 指示工作表中包含数据的最后一行。

数据结束于列 指示工作表中包含数据的最后一列。

高级选项

列名分隔符字符串 若列标题最初位于不同行中，则指示列标题中每个单词之间的分隔符。在“Excel 导入向导”的第一个窗口中指定带列标题的行数。然后在“列名分隔符字符串”框中输入一个字符或空格。使用默认字符串（连字符）的列标题显示如下：“First-Second-Third”。

多序列堆叠 将子类别划分到工作表中带分层标题的单独列中。您还必须选择**将多个列标题行按层次分开**。主类别作为“标签”列导入。

复制跨行中的标题 重复工作表中横跨多个行的每个单元格中的标题文本。例如，图 3.7 中显示的“State”列标题位于工作表中的合并单元格内。在左侧，标题已复制。取消选择该选项可避免复制右侧显示的标题。

图 3.7 Microsoft Excel 文件中的复制标题

复制的标题

	State-State
1	TX
2	TX
3	TX
4	TX

未复制的标题

	State
1	TX
2	TX
3	TX
4	TX

导入单元格颜色 将工作表中的单元格颜色应用到数据表。在 macOS 上，只能确保导入主要和次要颜色。

限值列类型检测 扫描至多 100 行以确定列类型。选择该选项可加快大型工作表的导入。

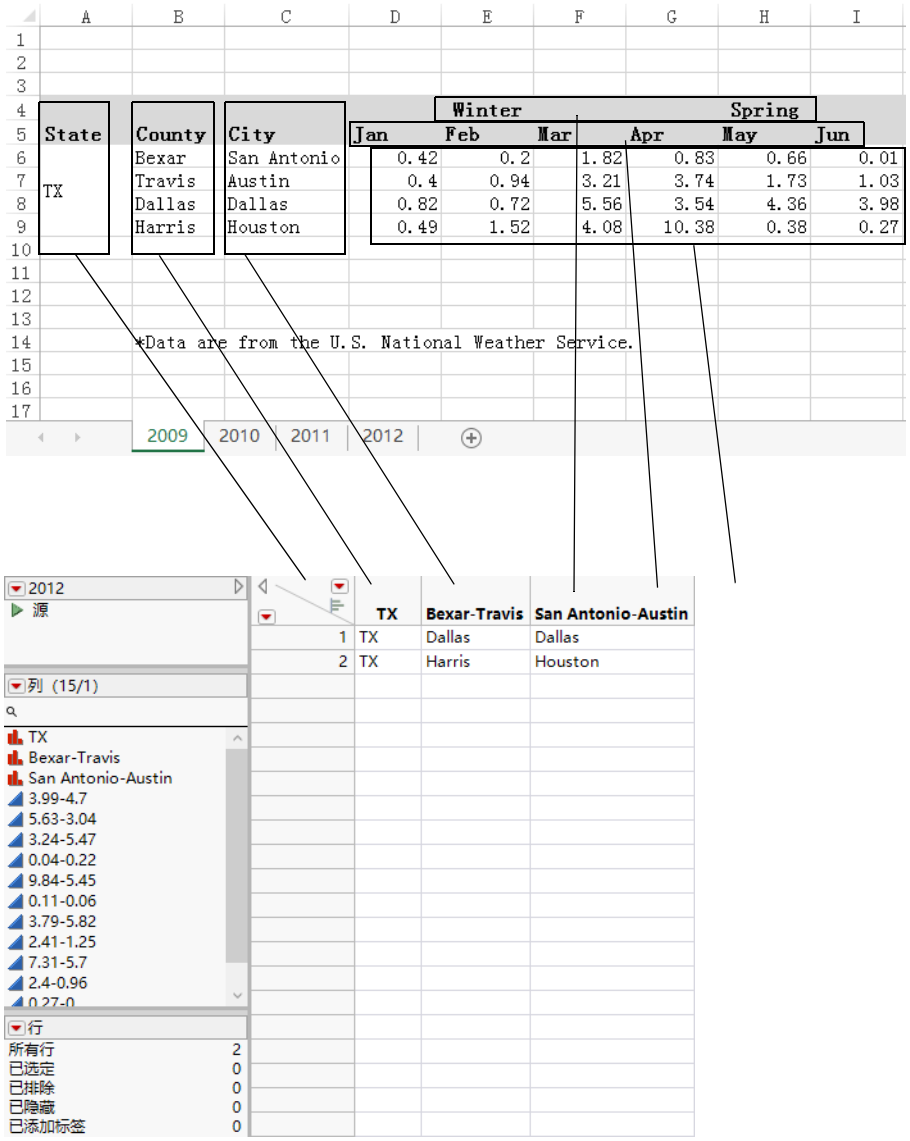
提示：

- 每次您导入工作表时，JMP 都会记住您之前的更改，即使是在关闭再重新打开 JMP 后也是如此。当您需要多次重新导入同一工作表而且需要试用不同选项时，该功能十分有用。要在导入其他工作表时清除这些更改，请点击**恢复默认设置**。
- 您的导入设置保存在名为“源”的数据表脚本中。要使用相同的设置重新导入工作表，请运行该脚本。该脚本包含指向工作表的路径，所以确保其他用户有权访问该位置。
- 要加快大型工作表中的数据预览，请在向导的第一个窗口中取消选择**对任意更改更新设置**。修改该设置，然后点击**立即更新**刷新数据预览。
- 要在“数据预览”窗格中查看所有行，请选择**显示所有行**。预览可能略有延迟，具体取决于电子表格的大小。
- 您可以将同一工作簿中的两个工作表合并为一个数据表。导入时将对列名进行匹配，所以列的顺序无关紧要。

导入包含层次标题的 Microsoft Excel 文件

在 Excel 工作表中，多个标题行可包含一个隐含的层次；第二个标题行包含的数据是第一个标题行的类别。图 3.8 显示了一个示例。在顶部的工作表中，“Winter”和“Spring”位于这两个季节对应月份之上的跨越单元格中。在 JMP 数据表中，您想让这两个季节位于一列中而其对应的月份位于另一列中。

图 3.8 Excel 中的原始数据和 JMP 中有关 2009 年的最终数据



要导入工作表并保持多个列层次，请按照以下步骤进行操作。

1. 在 JMP 中，选择**文件 > 打开**。
2. 在 “打开数据文件”窗口中，选择位于 JMP Samples/Import Data 文件夹中的 Texas Precipitation.xlsx，然后点击**打开**。

   该工作表在 “Excel 导入向导”中打开，向导中显示有数据预览以及导入选项。

图 3.9 “Excel 导入向导”预览

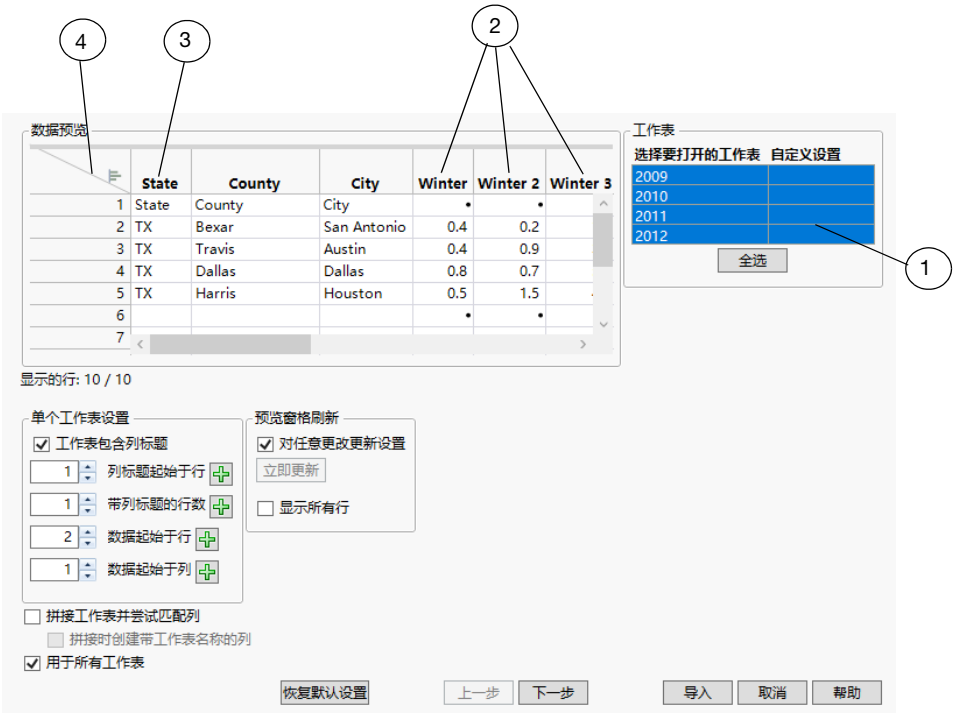


图 3.9 显示了该工作表的默认设置：

1	所有工作表都选定进行导入。按 Ctrl 键并点击某个工作表将其从导入中排除。
2	每一个季节都分成了多个列。
3	工作表第二个标题行中的数据显示在第一行中。
4	已删除顶部的空行。

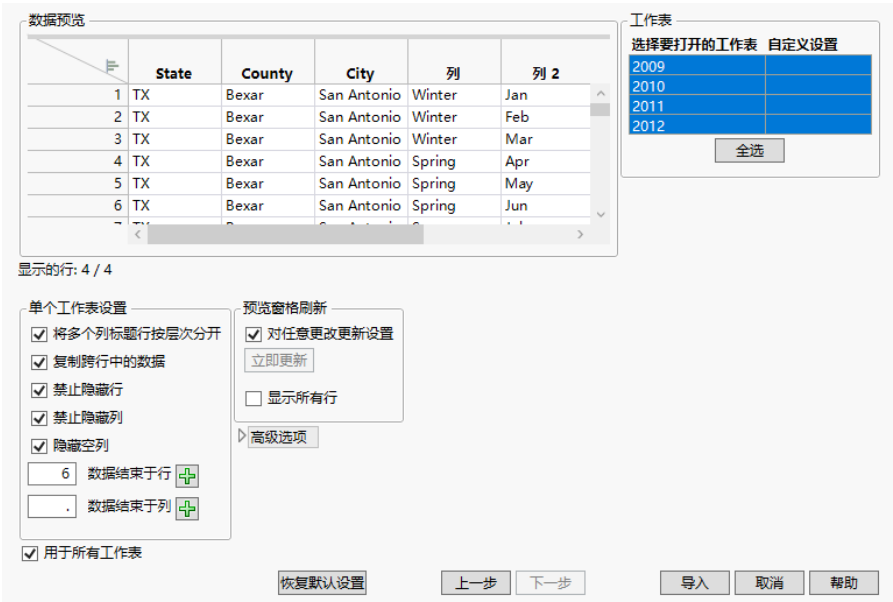
3. 在 “预览窗格刷新”下方，确保已选中**对任意更改更新设置**。
   这意味着当您进行更改时 “数据预览”会自动刷新。
4. 在**带列标题的行数**旁边，点击向上箭头一次。
   请注意，**数据起始于行**自动更新为 3。

图 3.10 第 1 页中更新的设置

5. 点击**下一步**。
6. 在**数据结束于行**旁边键入 6，然后按 Enter 键。
   前三个空行从工作表删除后，数据结束于第 6 行。

> **提示：**您无需键入结束行号，JMP 可帮您计算。在“数据预览”窗格中，选择第 4 行。点击**数据结束于行**旁边的加号 。

图 3.11 第 2 页中更新的设置



请注意，默认选定**复制跨行中的数据**。JMP 将拆分工作表中合并的数据，并将单元格内容作为单独的类别进行复制。“TX”最初出现在合并的单元格中。在 JMP 中，它将被复制到单独的单元格中。

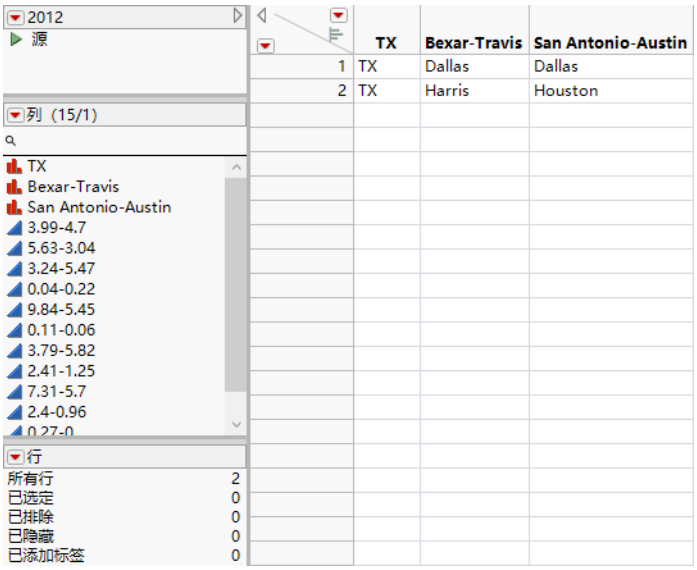
7. 选择**将多个列标题行按层次分开**。

   工作表标题行中的每个季节和月份都将成为数据表中的类别。

8. 点击**导入**。

   四个工作表将作为单独的数据表打开。数据表包含一个“源”脚本，您可以运行该脚本，以使用相同的导入设置将数据导入新的数据表（图 3.12）。您下一次在任何 JMP 会话的“Excel 向导”中打开工作表时，您的导入设置也会被保存。

图 3.12 最终数据



直接导入 Microsoft Excel 文件

Microsoft Excel 文件默认在“Excel 导入向导”中打开。若工作表中数据的结构不规则，该选项很有帮助。例如，您可能需要排除隐藏列或将第三行中的文本转换为列标题。

您可以选择**文件 > 打开**以打开文件，而不是在“Excel 导入向导”中打开电子表格。默认情况下，JMP 将检测第一行是否包含标签，并将标签转换为列标题。您可以在“常规”首选项中更改该设置。“Excel 打开方法”首选项位于“文件”>“首选项”>“常规”(Windows) 和“JMP”>“首选项”>“常规”(macOS) 中。从“使用 Excel 标签作为标题”列表中，选择**始终**或**从不**。

设置 “Excel 打开方法”首选项

要始终在向导外打开 Microsoft Excel 文件，请更改“Excel 打开方法”首选项。选择一次打开所有工作表或从列表中进行选择。

打开 Microsoft Excel 文件 (Windows)

1. 您按照以上方式设置“Excel 打开方法”后，请选择**文件 > 打开**。
2. 要将第一行中的文本转换为列标题，请选择**始终强制使用 Excel 的第 1 行作为标签**旁边的**始终**。
3. 选择 **Excel 文件**文件类型，选择该文件，或输入 URL。
4. 要导入所有工作表，请点击**打开**。

   或

要选择您想打开的工作表，请点击**打开**按钮箭头，然后选择**打开选定的工作表**。选择一个或多个工作表，并点击**确定**。

打开 Microsoft Excel 文件 (macOS)

1. 选择**文件 > 打开**，然后选择文件。
2. （仅限 .xls）要将第一行中的文本转换为列标题，请选择**使用 Excel 标签作为标题**。
3. （仅限 .xls）要打开特定工作表，请选择**选择单个 Excel 工作表**。
4. 点击**打开**。

   若已选择打开特定工作表，请从列表中打开这些工作表，然后点击**确定**。若您改变想法，想要导入所有工作表，还可以点击**全选**。

若您选择了一个 .xlsx 文件，数据预览将会出现在“Excel 向导”中。请参见第 65 页的“预览并导入 Microsoft Excel 数据”。

导入文本文件

您可以打开扩展名为 .txt、.csv 和 .tsv 的文本文件，文本将转换为数据表。包含文本的扩展名为 .dat 的文件也受支持。可以使用几乎任何字符来分隔文本文件，这些文件也可采用固定宽度。

当您将文本文件导入 JMP 时，尾空格将会被删除。

若 JMP 在文本文件中找到大于 9,007,199,254,740,991 的整数，则该列将被视为字符型数据。您可以使用“文本导入向导”窗口或导入脚本将该列设置为数值列。

提示：您可以使用“多个文件导入”选项一次导入多个文本文件。请参见第 86 页的“导入多个文件”。

要调整导入设置，请选择下列其中一个选项

- 选择**文件 > 首选项 > 文本数据文件**更改导入设置，以便 JMP 确定对数据表进行构建和设置格式的最佳方式。
- 在打开文件的同时手动选择导入设置（在本节中说明）。
- 在“脚本编辑器”中打开文件，编辑内容，然后导入内容。在您需要添加文本分隔符或修改文本时，该选项很有帮助。

导入文本文件

1. 选择**文件 > 打开**。
2. 在 Windows 上，您可以将文件类型设置为**文本文件**。
3. 选择要打开的文本文件。

   有关选项的信息，请参见表 3.1 和表 3.2。

表 3.1 在 Windows 上打开文本文件

自动确定数据排列	手动指定数据排列
1. 选择**文件 > 打开**。 2. 从**文件名**旁边的列表中，选择**文本文件**。 3. 要使用首选项中的导入规则，请选择**数据（使用首选项）**。（请参见“JMP 首选项”一章中第 602 页的“用于导入和导出文本文件的首选项”。） 要支持文本导入使用其最佳推测来排列数据，请选择**数据 （最佳推测）**选项。 （可选）选择**下次调用该窗口时选择该过滤条件**选项，默认应用您选定的过滤条件。 4. 选择要打开的文件。 5. 点击**打开**。 **提示：**若最近曾打开过该文件，则“JMP 主窗口”会提供以上步骤的快捷方式。右击“最近使用的文件”列表中的文件，然后选择**导入（首选项）**或**导入 （最佳推测）** 。（导入首选项在右击菜单中以粗体显示。)	1. 选择**文件 > 打开**。 2. 从**文件名**旁边的列表中，选择**文本文件**。 3. 选择窗口底部的**打开方式**旁边的**数据（使用预览）**。 4. 选择要打开的文件。 5. 点击**打开**。 6. 完成“文本导入”窗口中的操作。请参见第 79 页的“‘JMP 文本导入向导”中的选项”。 7. 点击**导入**。 **提示：**若最近曾打开过该文件，则“JMP 主窗口”会提供以上步骤的快捷方式。在“最近使用的文件”列表中右击该文件，然后选择**导入 （预览）**。

表 3.2 在 macOS 上打开文本文件

自动确定数据排列	手动指定数据排列
1. 选择**文件 > 打开**。 2. 选择要打开的文件。 3. 在**打开方式**字段中，选择**数据（最佳推测）**或**数据 （使用首选项）**。 4. 点击**打开**。	1. 选择**文件 > 打开**。 2. 选择要打开的文件。在**打开方式**字段中，选择**数据 （数据预览）**。 3. 点击**打开**。 4. 完成“文本导入向导”窗口中的操作。请参见第 79 页的“ ‘JMP 文本导入向导”中的选项”。 5. 点击**导入**。

注意：

- 在 Windows 上，JMP 可以在用户计算机默认的文本编辑器中打开文本文件。选择**文件 > 打开**，然后从**文件名**列表中选择**所有文件 (*.*)**。选择文本文件，然后选择**使用默认程序打开**。**若不选择该选项，则作为文本打开。**
- 若您在 JMP 扫描数据时取消导入，则将根据 JMP 检测到的内容导入数据。数据可能不会如您预期的那样显示。例如，您可能会看到列中的缺失值被错误地检测为数值。若在扫描数据**后**取消导入，则仅导入读取到这一时点的数据。

有关从“脚本”窗口导入文本的详细信息，请参见第 85 页的“从“脚本编辑器”导入文本”。

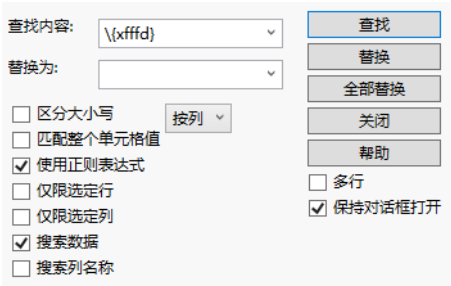
查找导入数据中的缺失字符

JMP 尝试在您的文档中检测字符编码。有时文档可能使用 JMP 无法识别的编码，或者 JMP 可能选择了错误的编码。

JMP 无法识别的文本将替换为 Unicode 字符 ?。通过搜索 `\x{fffd}` 查找文本。

1. 在数据表中，选择**编辑 > 搜索 > 查找**。
2. 在**查找内容**旁边键入 `\x{fffd}`。
3. 确保选定**使用正则表达式**。
4. 确保选定**搜索数据**。
5. 点击**查找**查找第一个实例。
6. 用另一个字符串替换所有实例。

图 3.13 搜索 Unicode 字符



“JMP 文本导入向导”中的选项

打开 JMP 支持的文本文件时，JMP 在将该文件作为数据表打开之前可以显示文本预览。该选项支持您手动排列数据并设置数据格式。例如，您可以指定行尾字符或去除引号。

JMP 检测文件结构，并显示用于导入文本（具有分隔符或固定宽度字段）的选项。若 JMP 选择了错误的文件结构，点击**分隔字段**或**固定宽度字段**单选按钮可导入正确格式的数据。（例如，在文件实际具有分隔符时可能出现固定宽度窗口。）

文本导入向导选项显示在图 3.14 和图 3.15 中。

图 3.14 固定宽度文件的文本导入向导

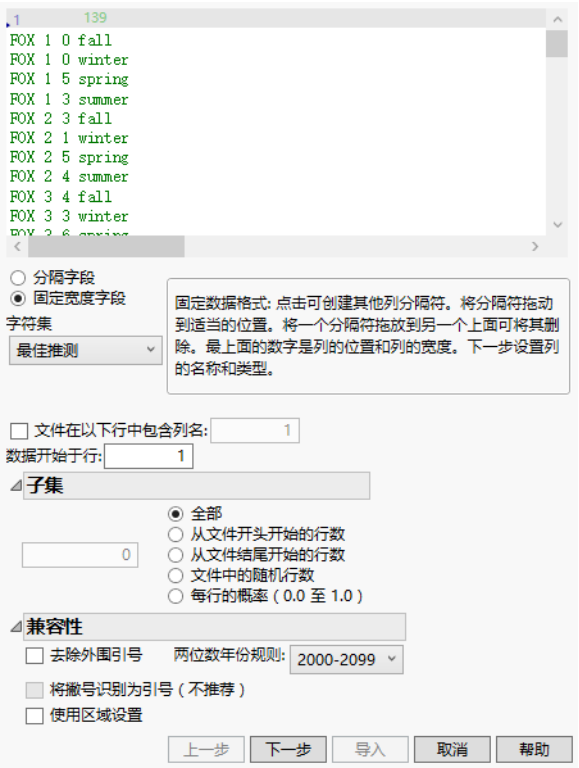


图 3.15 分隔文件的文本导入向导

字符集 选择导入文件中使用的字符集，或者让 JMP 检测字符集。若导入文件中出现不正确的字符，请再次打开该文件，选择另一个字符集。

字段结尾 （仅适用于“分隔导入”窗口。）选中标记字段结尾的字符旁边的复选框。或者，选中**其他**旁边的复选框，并在未列出所需字符的情况下输入一个字符。

行结尾 （仅适用于“分隔导入”窗口。）选中标记行尾的字符旁边的复选框。或者，选中**其他**旁边的复选框，并在未列出所需字符的情况下输入一个字符。请注意，若 JMP 发现双引号，分隔符规则会改为查找结束双引号。其他文本分隔符（包括引号中嵌入的空格）将被忽略，并视为文本字符串的一部分。

文件在以下行中包含列名 告知 JMP 在哪一行中查找要用作列名的数据。例如，若文本文件中的列名位于第 3 行，则选择该选项并在复选框中键入 3。否则，JMP 会将导入文件的第一行中的数据用作 JMP 数据表中的列名，或将第一行视为数据。

数据开始于行 指定包含数据的第一行的编号。

行数 指定要导入的行数。

去除外围引号 （仅适用于固定宽度的导入。）若希望 JMP 去除在文本文件用来引用数据的引号，请选中该复选框。

两位数年份规则 指定如何显示有两位数年份的日期。选择您的日期所属的 100 年范围。例如，若最早的日期是 2/2/79，且年份为 1979，则选择 1970-2069。若最早的日期是 2/2/12，且年份为 2012，则选择 2000-2099。若日期跨不同世纪，则必须在导入数据前使用四位年份对日期重新编码。

将撇号识别为引号 （不推荐） （仅适用于 “分隔导入”窗口）。只有在数据来自非标准数据源，其中使用撇号而不是引号来引用数据字段的情况下，才使用该选项。

使用区域设置 指定导入文本文件时是否使用操作系统的区域设置。若取消选定该选项 （默认设置），用句点表示小数点和用逗号表示值分隔符的文件将正确导入。您必须在 “文本数据文件”导入首选项中指定值分隔符。

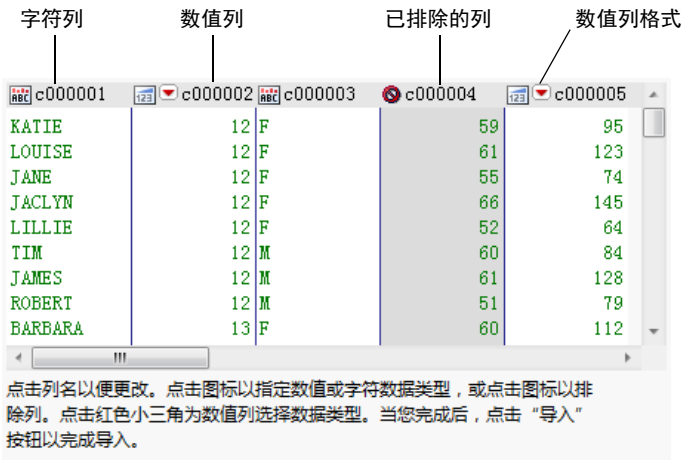
若文件将逗号用作小数点，您还必须设置以下选项：操作系统区域设置必须将逗号用作小数点 （Windows 和 macOS），或是 “Windows 特定显示语言”首选项必须设置为将逗号用作小数点的语言。

选好设置之后，点击**下一步**。下一个窗口显示每列的建模类型。要更改默认建模类型，请执行以下操作之一：

- 点击数据类型图标可将数据类型从数值 ( ) 改为字符 ( )。点击该图标可在建模类型和排除 ( ) 之间切换。排除意味着不导入列。
- 要更改数值列的数据格式，请从红色小三角菜单中选择所需格式。
- 点击列标题可修改文本。

“文本导入”窗口的顶部显示文本文件的预览，该预览与该文本文件导入 JMP 数据文件时的显示相同。点击**导入**按钮可导入数据。

图 3.16 带有列选项的 “文本导入向导”窗口



注意：要指定日期格式，请依次选择列名旁边的红色小三角、**日期**和日期格式。要指定导入数据后日期的显示格式，请右击数据表中的日期列，选择**列信息**，然后更改“格式”旁边的日期。

7. 完成数据设置的操作后，点击**导入**可完成文本导入。

注意：“文本导入向导”和“多个文件导入”均将各列数字默认导入为数字。若您有需要保留前导零的部件号或大于 15 位的部件号这样的数字，则可以强制该列为字符。在“文本导入向导”中，您可以逐列执行该操作（在第二个向导窗格中，列顶部）。在“多个文件导入”中，您可以选中使得**所有**列都作为字符导入的复选框。（否则不会保留前导零，15 位之外的数位将丢失。）有关“多个文件导入”的详细信息，请参见第 86 页的“导入多个文件”。

将文本导入 JMP 之前对其进行编辑

您可以在“脚本”窗口中打开文本文件，在该窗口中编辑文本，然后可以将该文本导入为数据表。在将文本导入为数据表之前，若需重新设置文本格式，该功能十分有用。例如，您可能需要插入正确的分隔符或修改文本。

另一个选项是以文本方式打开 JMP 插件定义 (.def) 文件，然后在“脚本”窗口中对其进行编辑。

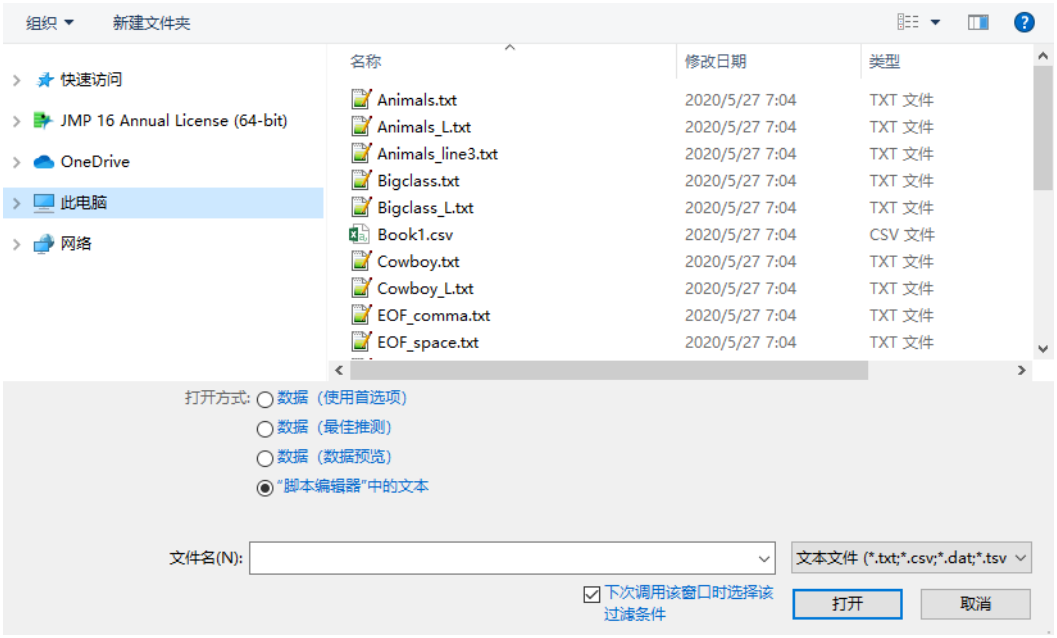
在文本编辑窗口中打开文本文件 (Windows)

最近打开的文件列在“JMP 主窗口”中。对于多数文件，右击文本文件并选择**以纯文本方式打开**即可在文本编辑窗口中打开该文件。无法从“JMP 主窗口”中以纯文本方式打开 JMP 插件定义文件。

首次打开该文件时，请执行以下步骤：

1. 选择**文件 > 打开**。
2. 从**文件名**旁边的列表中，选择**文本文件**。

图 3.17 选择文本文件



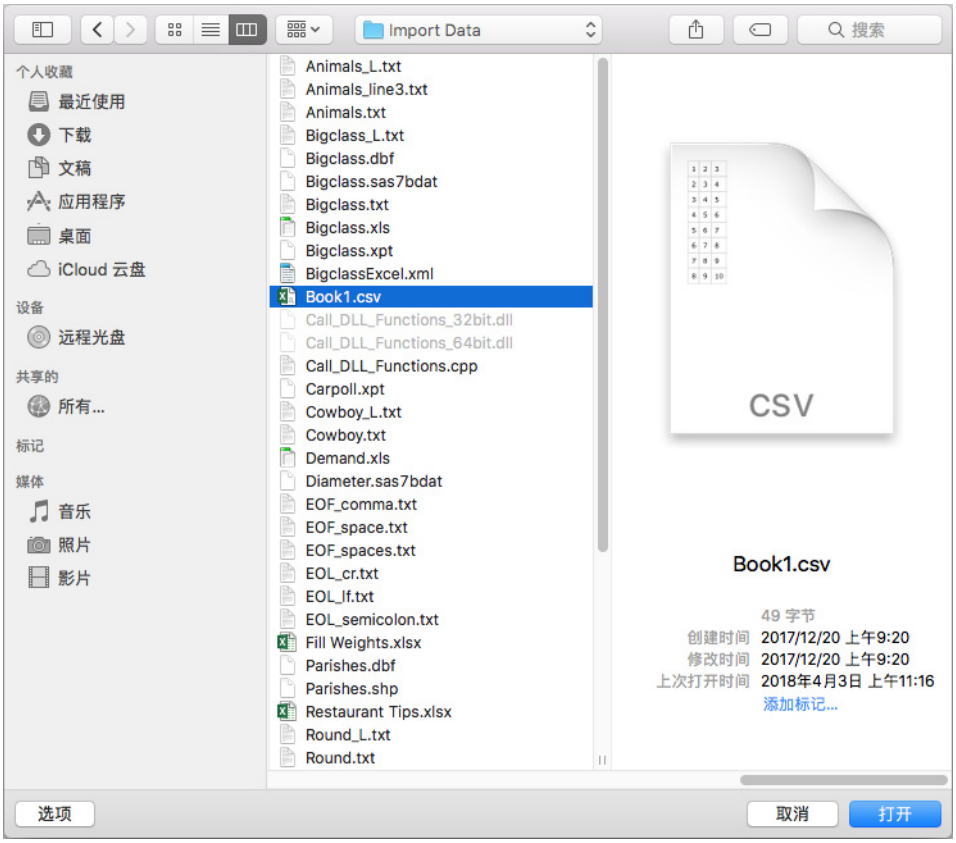
3. （可选）要将默认选项文件类型设置为**文本文件**，请选中**下次调用该窗口时选择该过滤条件**旁边的复选框。
4. 选择文件。
5. 选择**打开方式**旁边的**在脚本窗口中打开纯文本**。
6. 点击**打开**。

   文本显示在 “脚本”窗口中。

在文本编辑窗口中打开文本文件 (macOS)

1. 选择**文件** > **打开**。

图 3.18 在 macOS 上打开文本文档



2. 选择文件。
3. 点击**选项**。
4. 从**打开方式**列表中选择 **“脚本编辑器”中的文本**。
5. 点击**打开**。

   文本显示在 “脚本”窗口中。

有关将文本转换为数据表的详细信息，请按第 85 页的“从“脚本编辑器”导入文本”中的第 3 步操作。

从“脚本编辑器”导入文本

您可以将“脚本编辑器”中的文本导入为数据表。文本可以是表格式（例如，来自 Microsoft Word 文档或网页）或纯文本格式。在将文本导入为数据表之前，若需重新设置文本格式，该功能十分有用。例如，您可能需要插入正确的分隔符或修改文本。

JMP 使用首选项中的导入设置来确定如何构建文本和设置文本格式。有些选项包括删除文本外围引号，以及指定包含列标题和数据的行。请参见“JMP 首选项”一章中第 602 页的“用于导入和导出文本文件的首选项”。

注意：您还可以将整个网页作为数据表导入。请参见第 162 页的“从 Internet 或远程计算机导入数据”。

本节说明如何导入粘贴到“脚本编辑器”中的文本。有关在“脚本编辑器”中打开文本文件的详细信息，请参见第 83 页的“将文本导入 JMP 之前对其进行编辑”。

从 “脚本”窗口导入文本

1. 通过选择**文件** > **新建** > **脚本** (Windows) 或**文件** > **新建** > **新建脚本** (macOS)，打开新的“脚本”窗口。
2. 将文本复制并粘贴到 “脚本编辑器”中。
3. 请执行以下操作之一：
   - 要从“脚本编辑器”导入所有文本，请选择**编辑** > **作为数据导入**。
   - 要导入特定文本，请选择该文本，然后选择**编辑** > **作为数据导入**。

   该文本将导入到一个 JMP 数据表。

注意：要预览从“脚本编辑器”导入的文本，请在选择“文件”>“作为数据导入”之前按 Shift 键。

导入多个文件

多个文件导入 (MFI) 用于导入一个目录中的 CSV 数据和非结构化文本文件，然后将其堆叠在数据表中（或将其导入单独的数据表）。您可以将导入设置另存为脚本，然后在将来更新数据时运行该脚本。

“多个文件导入”支持以下文件类型：

- XLSX
- CSV
- TSV
- DAT
- TXT
- SAS7BDAT
- XML
- JMP

- JSON
- PNG
- JPG
- GIF

多个文件导入还可以仅导入一个文件。表 3.3 可以帮助您决定是使用标准的“文本导入”、“多个文件导入”还是 JSL。

表 3.3 在“文本导入向导”和 MFI 之间选择

导入特征	一个文件	多个文件
CSV、TSV、DAT	“文本导入”提供预览向导来帮助命名列、挑选日期格式以及取数据子集。	MFI 可自动堆叠类似文件。您可以稍后更改格式设置。
非结构化文本	MFI，很可能用 JSL 中的 `Load Text File()` 导入。	MFI
超过 100MB 的大型文件	您可以使用上一次“文本导入”的“源”脚本加快下一次导入。	MFI 以并行方式导入，可能比“文本导入”快得多。
不确定格式	“文本导入”也许能推测出格式。预览向导支持您在导入文本前查看结果。	MFI 不推测。您需要指定制表符分隔符。无预览，但是选择**保持对话框打开**很容易尝试不同的导入设置。
图片	尝试 JSL `Open()` 函数。	MFI 了解 PNG、JPG 和 GIF 文件。您会得到一个包含图片列的表。
JMP、XLSX 和 JSON	尝试 JSL `Open()` 函数。	MFI 使用 JSL `Open()` 打开 JMP、XLSX 和 JSON 文件，并且按需堆叠它们。MFI 在打开 JMP 表时不保留表和列脚本，所以若您需要上述所有脚本，您可能需要使用 `Files In Directory()` 和 `Open()`。

要导入多个文件，请执行以下步骤：

1. 将您要导入的文件放入同一目录。

   本例使用 JMP Samples/Import Data 文件夹中的导入数据。

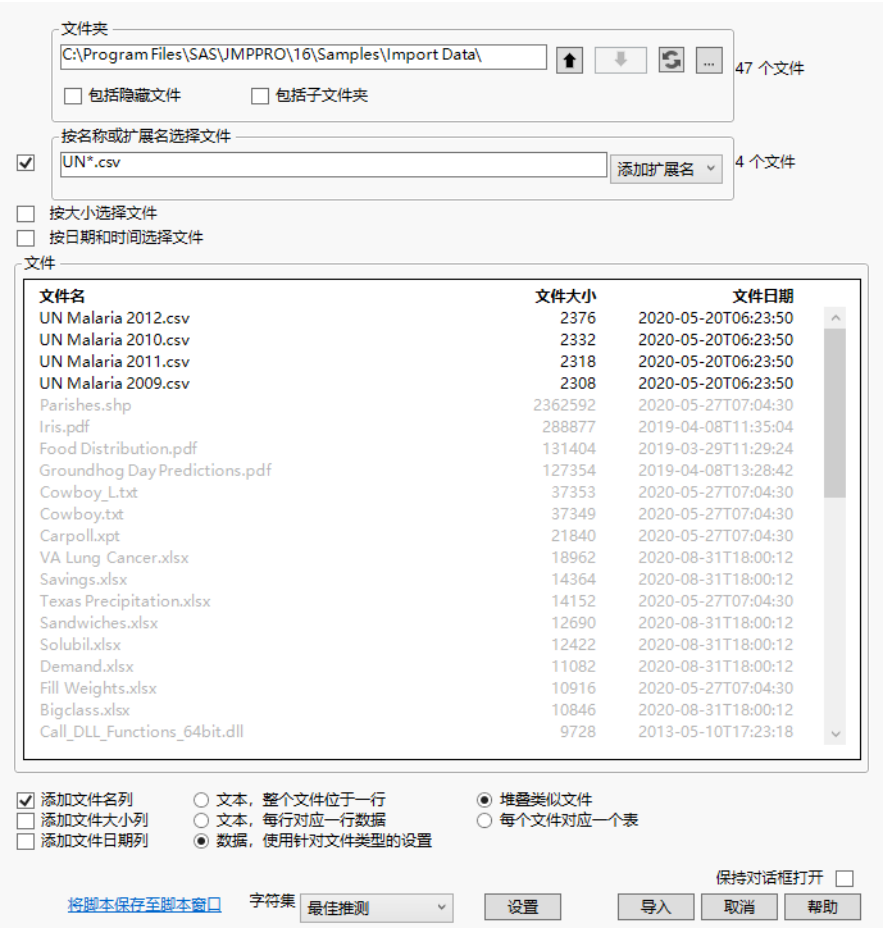
2. 选择**文件 > 导入多个文件** (Windows) 或**文件 > 打开多个** (macOS)。
3. 浏览并选择 JMP Samples/Import Data 文件夹。
4. 在**按名称或扩展名选择文件**框中，键入 UN*.csv 并按 Enter 键。

   在本例中，文件具有 UN 前缀且为 CSV 文件。

   “文件”列表更新以显示选定的那些文件。

   指定多个文件名和扩展名并用分号分隔它们，或点击**添加扩展名**并选择一个扩展名。

图 3.19 选定的 CSV 文件



48 个文件匹配该过滤器的条件。

4 个文件匹配该过滤器的条件。

请注意，若自上次刷新以来目录中的文件已更改，则 “刷新”按钮 将闪动。

5. 指定您要添加到数据表的列。

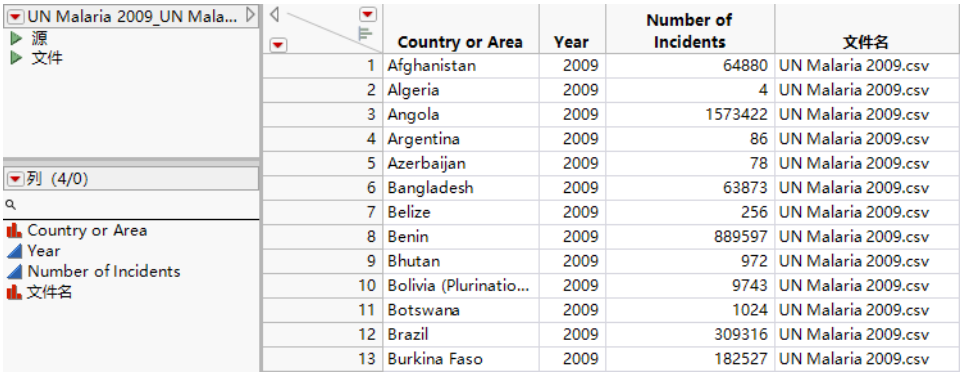
   默认情况下，不添加更多列。您可能需要用 “文件名”列显示导入文件的名称。选择**添加文件名列**。

注意：“文本数据文件”首选项不能确定在 CSV 设置中选定了哪些选项，尽管这些设置很相似。

6. 保持选定**堆叠类似文件**，因为这些 CSV 文件具有相同的列。
7. 点击**导入**。

   JMP 应用文件名、文件大小以及文件日期和时间过滤器来堆叠文件。列中包含相同列标题的数据将导入同一数据表。若各列不匹配，不匹配的列将导入单独的数据表。

图 3.20 导入的 CSV 数据



	Country or Area	Year	Number of Incidents	文件名
1	Afghanistan	2009	64880	UN Malaria 2009.csv
2	Algeria	2009	4	UN Malaria 2009.csv
3	Angola	2009	1573422	UN Malaria 2009.csv
4	Argentina	2009	86	UN Malaria 2009.csv
5	Azerbaijan	2009	78	UN Malaria 2009.csv
6	Bangladesh	2009	63873	UN Malaria 2009.csv
7	Belize	2009	256	UN Malaria 2009.csv
8	Benin	2009	889597	UN Malaria 2009.csv
9	Bhutan	2009	972	UN Malaria 2009.csv
10	Bolivia (Plurinatio...	2009	9743	UN Malaria 2009.csv
11	Botswana	2009	1024	UN Malaria 2009.csv
12	Brazil	2009	309316	UN Malaria 2009.csv
13	Burkina Faso	2009	182527	UN Malaria 2009.csv

注意：

- 要导入隐藏文件，请选择**包括隐藏文件**。
- 要导入子目录中的文件，请选择**包括子文件夹**。
- 要导入特定大小范围内的文件，请在**按大小选择文件**框中，键入文件大小的上限和下限。

  若您不想导入超过一定限值的大型文件，则考虑指定限值。

  以 KB （千字节）为单位指定文件大小。 1 KB 等于 1,000 字节。
- 要导入在特定日期和时间范围内修改的文件，请在**按日期和时间选择文件**框中，点击日历按钮 以指定该范围。
- （非结构化文本文件）基于文件结构选择**文本，整个文件位于一行**或**文本，每行对应一行数据**。

  注意：

  – 这两个选项不支持 XLSX、JSON 和 JMP 文件。
  – 当您选择“文本，整个文件位于一行”时，导入将包括整个文件，并且不会跳过第一个数据行之前的行。
- JMP 创建一个数据表，其中列出无法导入的文件（例如，SAS 数据集）。

导入包含不同引用的结构化数据

有时导出过滤器会在字段周围放置引号之外的符号。您在 CSV 设置中指定分隔符。

CSV 文件中的文本可能包含转义序列，例如，用反斜杠代替双引号。您也是在 CSV 设置中指定分隔符。

提示：

- **保持对话框打开**默认关闭，但您可以在测试时打开该设置。这样您就可以导入文件以查看数据的显示方式，按需编辑源文件，再使用相同的设置重新导入它们。
- 导入不受支持的文件时，将创建名为“@ Unused Files”的数据表。文件名列在“文件”表脚本中。右击该脚本，然后选择“编辑”。要在该数据表中输出不受支持文件的文件名，请在“多个文件导入”窗口中选择**添加文件名列**。
- 右击“文件”列表并选择“按列排序”以更改“文件”列表的排序方式。
- 若没有任何文件匹配您选定的过滤器，则“导入”按钮不可用。
- 当您为“文本分析器”样式的数据选择两个选项中的任一个（底部中央的单选按钮）时，JMP 自动更改 CSV 设置。不指定分隔符，这会将所有数据都加载到一列中。JMP 还会更改 CSV 设置，使其不使用行尾字符，这会将整个文件加载到一行中。您还可以更改 CSV 设置。若想尝试手动操作，务必设置“无标题”并将“第一个数据行”设置为 1。

导入 PDF 文件

通过“PDF 导入向导”，您可以在导入数据之前预览 PDF 文件、调整导入设置并在数据表预览中查看结果。JMP 尝试自动检测 PDF 文件中的数据结构，或者您可以定制该结构。

注意：PDF 文件由包含数据的表构成。在 PDF 预览中，在每页上都显示表的外框并且选定各表。

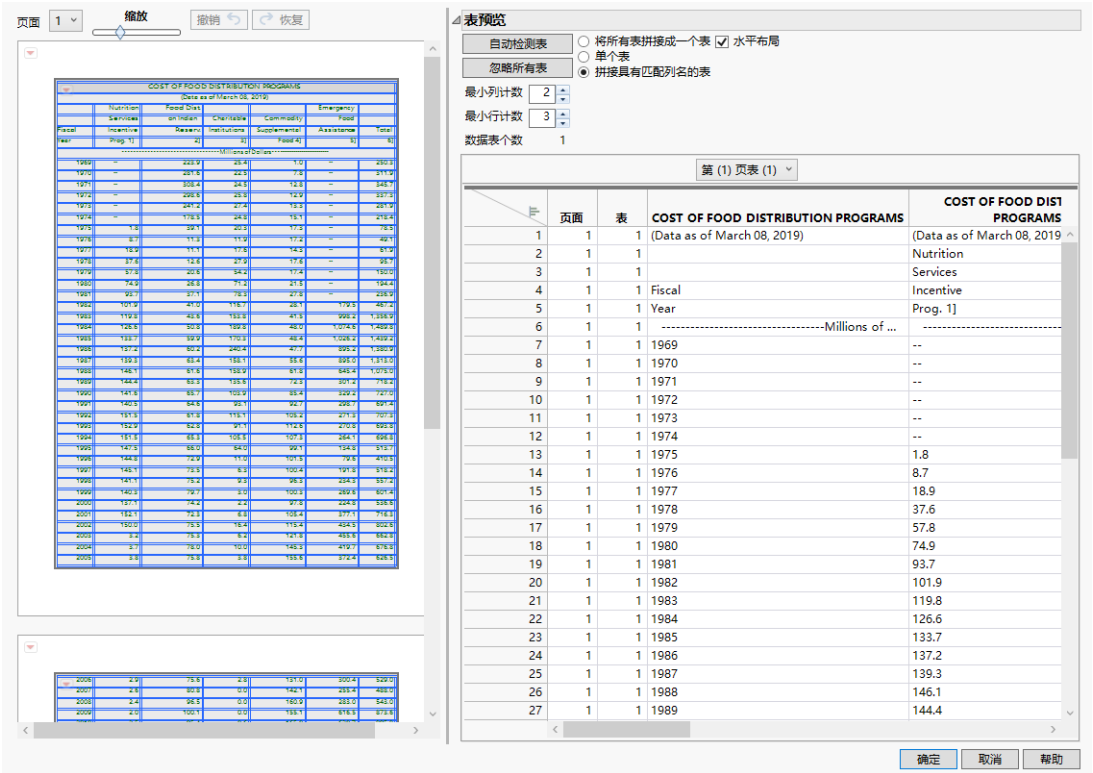
- “将多个表拼接为一个数据表”
- “按列标题拼接表”
- “导入 JMP 未能自动检测的表”
- ““PDF 预览”红色小三角选项”
- “数据表预览选项”
- “PDF 预览右击选项”

将多个表拼接为一个数据表

1. 选择**文件 > 打开**并从“文件名”旁边的列表中选择 **PDF 文件**。
2. 导航至 JMP Samples/Import Data 文件夹并双击 Food Distribution.pdf。
3. 在 “导入 PDF 向导”中， PDF 文件的预览显示在左侧。数据表中数据的预览显示在右侧。

PDF 文件在每页上包含一个表。

图 3.21 初始 PDF 文件和数据表预览



请注意以下几点：

- PDF 顶部有七行您不想导入。
- 您还不想要 PDF 文件最后一行中的脚注。

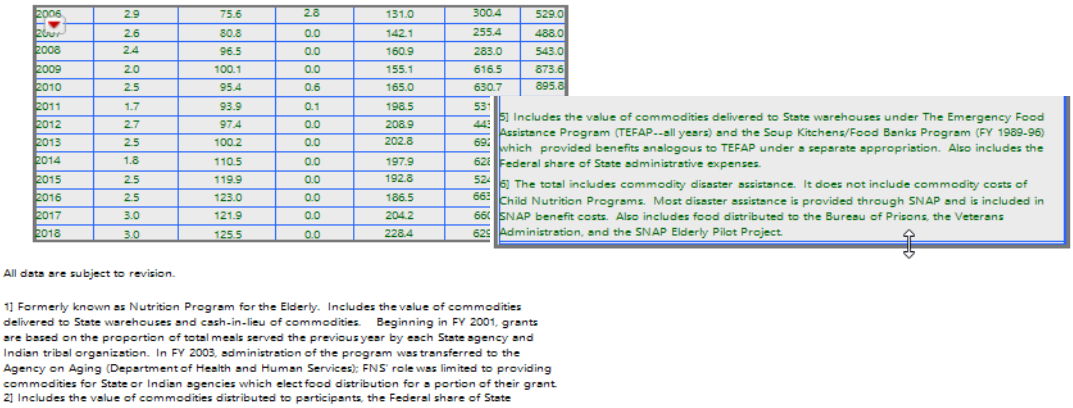
4. 在表的一角，点击红色小三角并选择**用作标题的行数 > 指定行**。
5. 键入 7 并点击**确定**。

   在 PDF 文件预览中，选定了前七行。

   在数据表预览中，数据从第 1 行开始。
6. 在 PDF 文件预览中，滚动到第 2 页。
7. 向上拖动 PDF 表的底部，直到不再选定脚注为止 （图 3.22）。

注意：您还可以简单地从最终数据表中删除行。

图 3.22 拖动表以删除数据



2006	2.9	75.6	2.8	131.0	300.4	529.0
2007	2.6	80.8	0.0	142.1	255.4	488.0
2008	2.4	96.5	0.0	160.9	283.0	543.0
2009	2.0	100.1	0.0	155.1	616.5	873.6
2010	2.5	95.4	0.6	165.0	630.7	895.8
2011	1.7	93.9	0.1	198.5	531	
2012	2.7	97.4	0.0	208.9	443	
2013	2.5	100.2	0.0	202.8	69	
2014	1.8	110.5	0.0	197.9	628	
2015	2.5	119.9	0.0	192.8	524	
2016	2.5	123.0	0.0	186.5	663	
2017	3.0	121.9	0.0	204.2	660	
2018	3.0	125.5	0.0	228.4	625	

5] Includes the value of commodities delivered to State warehouses under The Emergency Food Assistance Program (TEFAP--all years) and the Soup Kitchens/Food Banks Program (FY 1989-96) which provided benefits analogous to TEFAP under a separate appropriation. Also includes the Federal share of State administrative expenses.

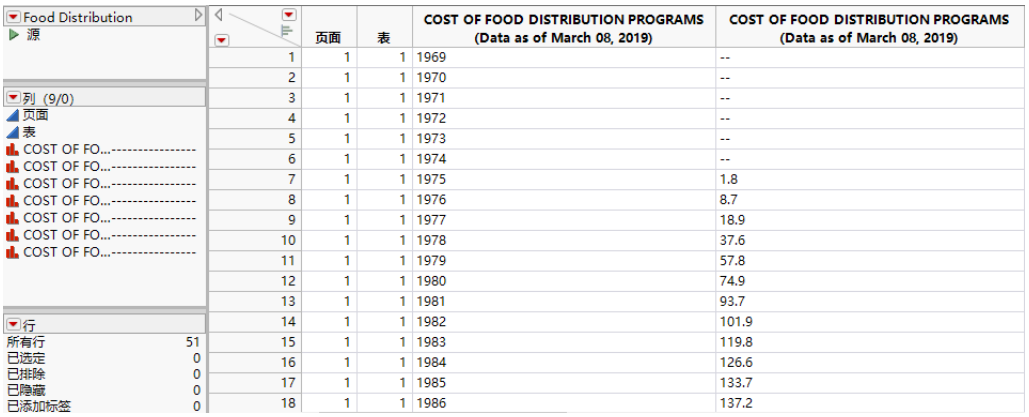
6] The total includes commodity disaster assistance. It does not include commodity costs of Child Nutrition Programs. Most disaster assistance is provided through SNAP and is included in SNAP benefit costs. Also includes food distributed to the Bureau of Prisons, the Veterans Administration, and the SNAP Elderly Pilot Project.

All data are subject to revision.

1] Formerly known as Nutrition Program for the Elderly. Includes the value of commodities delivered to State warehouses and cash-in-lieu of commodities. Beginning in FY 2001, grants are based on the proportion of total meals served the previous year by each State agency and Indian tribal organization. In FY 2003, administration of the program was transferred to the Agency on Aging (Department of Health and Human Services); FNS' role was limited to providing commodities for State or Indian agencies which elect food distribution for a portion of their grant.
2] Includes the value of commodities distributed to participants, the Federal share of State

8. 在右侧的 “表”报表中，选择**将所有表拼接成一个表**。
   拼接数据表时，需要将两个或更多数据表中的行合并到一个数据表中。
9. 在底部，点击**确定**。
   数据将导入数据表。

图 3.23 最终数据表 （部分视图）



Food Distribution
源
列 (9/0)
页面
表
COST OF FO...----------
COST OF FO...----------
COST OF FO...----------
COST OF FO...----------
COST OF FO...----------
COST OF FO...----------
COST OF FO...----------
行
所有行 51
已选定 0
已排除 0
已隐藏 0
已添加标签 0

	页面	表	COST OF FOOD DISTRIBUTION PROGRAMS (Data as of March 08, 2019)	COST OF FOOD DISTRIBUTION PROGRAMS (Data as of March 08, 2019)
1	1	1	1969	--
2	1	1	1970	--
3	1	1	1971	--
4	1	1	1972	--
5	1	1	1973	--
6	1	1	1974	--
7	1	1	1975	1.8
8	1	1	1976	8.7
9	1	1	1977	18.9
10	1	1	1978	37.6
11	1	1	1979	57.8
12	1	1	1980	74.9
13	1	1	1981	93.7
14	1	1	1982	101.9
15	1	1	1983	119.8
16	1	1	1984	126.6
17	1	1	1985	133.7
18	1	1	1986	137.2

按列标题拼接表

当 PDF 文件包含的表格数据在每页上都具有相同的列标题时，可以基于列标题拼接表。

1. 选择**文件 > 打开**并从 “文件名”旁边的列表中选择 **PDF 文件**。
2. 导航至 JMP Samples/Import Data 文件夹并双击 Iris.pdf。
   在 PDF 预览中滚动，注意每页上的列标题都是相同的。
3. 在右侧的 “表”报表中，确保选定**拼接具有匹配列名的表**。

4. 在底部，点击**确定**。

图 3.24 最终数据表 （部分视图）

	Page	Table	Sepal length	Sepal width	Petal length	Petal width	Species
1	1	1	5.1	3.5	1.4	0.2	setosa
2	1	1	4.9	3	1.4	0.2	setosa
3	1	1	4.7	3.2	1.3	0.2	setosa
4	1	1	4.6	3.1	1.5	0.2	setosa
5	1	1	5	3.6	1.4	0.2	setosa
6	1	1	5.4	3.9	1.7	0.4	setosa
7	1	1	4.6	3.4	1.4	0.3	setosa
8	1	1	5	3.4	1.5	0.2	setosa
9	1	1	4.4	2.9	1.4	0.2	setosa
10	1	1	4.9	3.1	1.5	0.1	setosa
11	1	1	5.4	3.7	1.5	0.2	setosa
12	1	1	4.8	3.4	1.6	0.2	setosa
13	1	1	4.8	3	1.4	0.1	setosa
14	1	1	4.3	3	1.1	0.1	setosa
15	1	1	5.8	4	1.2	0.2	setosa
16	1	1	5.7	4.4	1.5	0.4	setosa
17	1	1	5.4	3.9	1.3	0.4	setosa
18	1	1	5.1	3.5	1.4	0.3	setosa

导入 JMP 未能自动检测的表

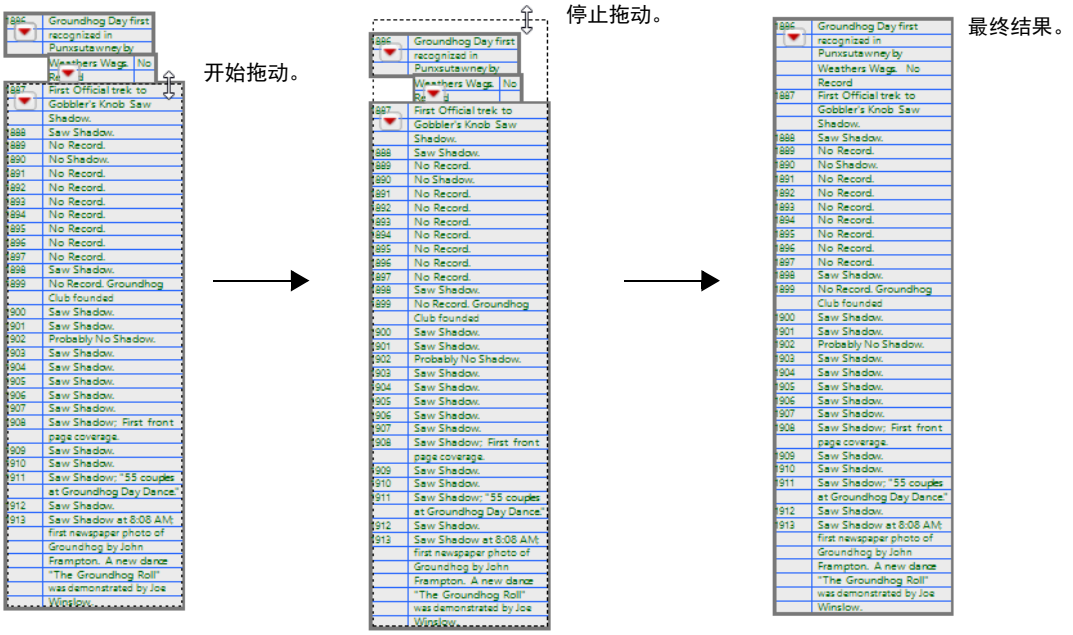
PDF 文件中的表有时无法自动检测。在这些情况下，可以拖动列周围的框来对表进行分组和划分。

图 3.25 PDF 文件的部分预览

请注意，第一列中的行被自动检测为三个表。

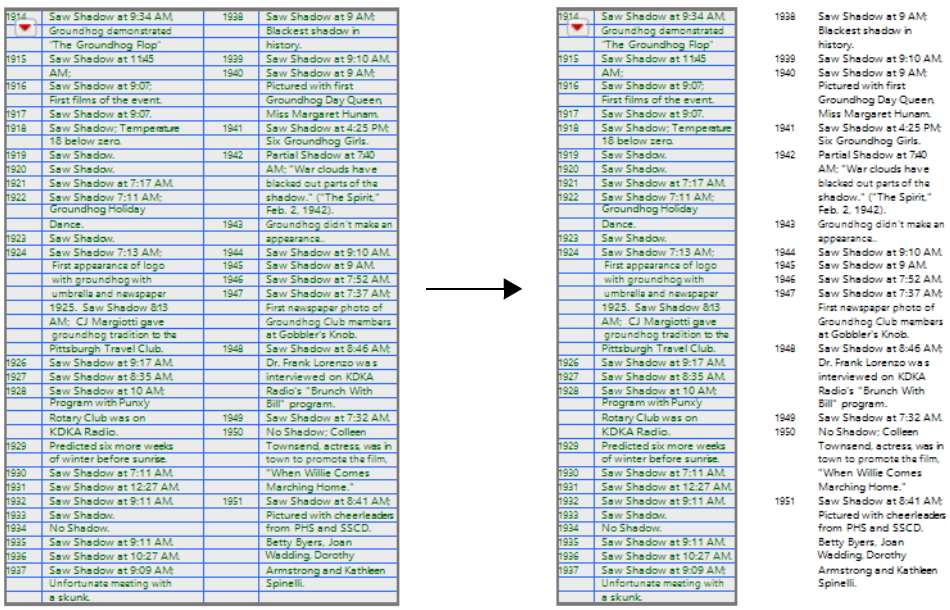
向上拖动第三个表的上边缘，直到它包围两列。

图 3.26 创建一个表



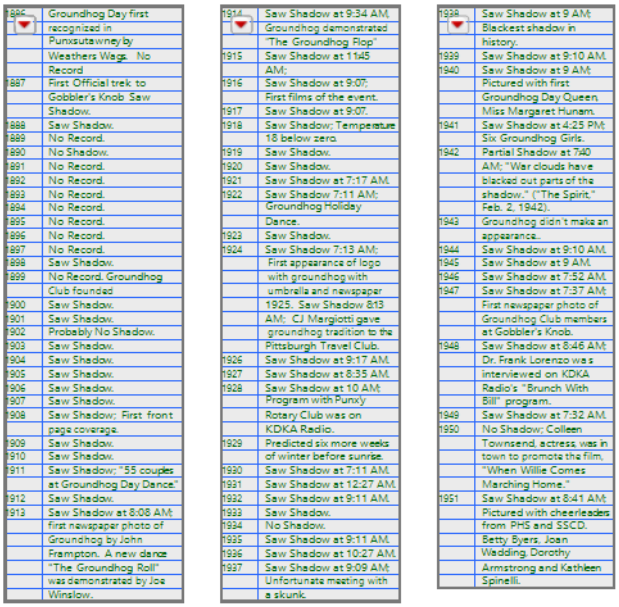
在图 3.27 所示的第一个表中，向左拖动最右边框，直到只有列 1 和列 2 位于同一表中，列 3 和列 4 位于表外（图 3.27）。

图 3.27 拆分表



在剩余的两列周围拖动一个框以将其放入表中（图 3.28）。

图 3.28 第 1 页上的最终表



1886	Groundhog Day first recognized in Punxsutawney by Weathers Wags. No Record
1887	First Official trek to Gobbler's Knob Saw Shadow.
1888	Saw Shadow.
1889	No Record.
1890	No Shadow.
1891	No Record.
1892	No Record.
1893	No Record.
1894	No Record.
1895	No Record.
1896	No Record.
1897	No Record.
1898	Saw Shadow.
1899	No Record. Groundhog Club founded
1900	Saw Shadow.
1901	Saw Shadow.
1902	Probably No Shadow.
1903	Saw Shadow.
1904	Saw Shadow.
1905	Saw Shadow.
1906	Saw Shadow.
1907	Saw Shadow.
1908	Saw Shadow; First front page coverage.
1909	Saw Shadow.
1910	Saw Shadow.
1911	Saw Shadow; "55 couples at Groundhog Day Dance."
1912	Saw Shadow.
1913	Saw Shadow at 8:08 AM; first newspaper photo of Groundhog by John Frampton. A new dance "The Groundhog Roll" was demonstrated by Joe Winslow.
1914	Saw Shadow at 9:34 AM. Groundhog demonstrated "The Groundhog Flop"
1915	Saw Shadow at 11:45 AM;
1916	Saw Shadow at 9:07; First films of the event.
1917	Saw Shadow at 9:07.
1918	Saw Shadow; Temperature 18 below zero.
1919	Saw Shadow.
1920	Saw Shadow.
1921	Saw Shadow at 7:17 AM.
1922	Saw Shadow 7:11 AM; Groundhog Holiday Dance.
1923	Saw Shadow.
1924	Saw Shadow 7:13 AM; First appearance of logo with groundhog with umbrella and newspaper 1925. Saw Shadow 8:13 AM; CJ Margiotti gave groundhog tradition to the Pittsburgh Travel Club.
1926	Saw Shadow at 9:17 AM.
1927	Saw Shadow at 8:35 AM.
1928	Saw Shadow at 10 AM; Program with Punxy Rotary Club was on KDKA Radio.
1929	Predicted six more weeks of winter before sunrise.
1930	Saw Shadow at 7:11 AM.
1931	Saw Shadow at 12:27 AM.
1932	Saw Shadow at 9:11 AM.
1933	Saw Shadow.
1934	No Shadow.
1935	Saw Shadow at 9:11 AM.
1936	Saw Shadow at 10:27 AM.
1937	Saw Shadow at 9:09 AM; Unfortunate meeting with a skunk.
1938	Saw Shadow at 9 AM; Blackest shadow in history.
1939	Saw Shadow at 9:10 AM.
1940	Saw Shadow at 9 AM; Pictured with first Groundhog Day Queen, Miss Margaret Hunam.
1941	Saw Shadow at 4:25 PM; Six Groundhog Girls.
1942	Partial Shadow at 7:40 AM; "War clouds have blacked out parts of the shadow." ("The Spirit," Feb. 2, 1942).
1943	Groundhog didn't make an appearance.
1944	Saw Shadow at 9:10 AM.
1945	Saw Shadow at 9 AM.
1946	Saw Shadow at 7:52 AM.
1947	Saw Shadow at 7:37 AM; First newspaper photo of Groundhog Club members at Gobbler's Knob.
1948	Saw Shadow at 8:46 AM; Dr. Frank Lorenzo was interviewed on KDKA Radio's "Brunch With Bill" program.
1949	Saw Shadow at 7:32 AM.
1950	No Shadow; Colleen Townsend, actress, was in town to promote the film, "When Willie Comes Marching Home."
1951	Saw Shadow at 8:41 AM; Pictured with cheerleaders from PHS and SSCD. Betty Byers, Joan Wadding, Dorothy Armstrong and Kathleen Spinelli.

注意：Samples/Import Data 文件夹中的 Groundhog Day Predictions.pdf 显示在“PDF 导入向导”中自动检测到的数据的清理版本。

“PDF 预览”红色小三角选项

本节介绍 PDF 预览窗格中的红色小三角选项。

“全局”红色小三角

PDF 预览窗口左上角的红色小三角包含两个选项：

自动检测该页面上的表 若以前拆分或合并了表，该选项将还原所做的更改，以便再次自动检测表。

忽略该页面上的表 从页面中删除表。不导入该数据。

每个表上的红色小三角

忽略该表 删除该表。不导入该数据。

用作标题的行数 指定 PDF 文件中将不作为数据导入的标题行数。

自动检测该页面上的表 若以前拆分或合并了表，该选项将还原所做的更改，以便再次自动检测表。

忽略该页面上的表 从页面中删除表。不导入该数据。

数据表预览选项

本节介绍数据表预览窗格中的选项。

自动检测表 若以前拆分或合并了表，该选项将还原所做的更改，以便再次自动检测表。

忽略所有表 从 PDF 文件中删除表。不导入该数据。

最小列计数 避免导入列数少于指定列数的表。

最小行计数 避免导入行数少于指定行数的表。

将所有表拼接成一个表 将两个或更多数据表中的行合并到一个数据表中。

单个表 将数据导入多个表。

拼接具有匹配列名的表 将具有匹配列名的两个或更多数据表中的行合并到一个数据表中。

PDF 预览右击选项

有关多数 PDF 预览右击选项的说明，请参见第 95 页的“每个表上的红色小三角”。以下选项也可用：

添加列分隔线 在选定点创建列分隔线。选择该选项可在数据表中创建新列 （例如，若未能自动检测到列）。

删除列分隔线 在选定点删除列边框。

添加行分隔线 在选定点创建行边框。选择该选项以在数据表中创建新行。

删除行分隔线 在选定点删除列边框 （例如，若无法自动检测行）。

导入 XML 和 JSON 文件

预览和导入 XML 和 JSON 数据

“XML 导入向导”导入两种嵌套文本数据：XML 和 JSON 文件。这两种文件都包含可嵌套以表示层次关系的结构化文本。该向导帮助您选择层次结构中包含值的元素以及确定要生成数据表的行的元素。

可通过一种有用方式来想象嵌套文本文件：文档即一本短篇小说集。有些短篇小说可能有章节，有些则只有段落。章节包含段落。段落包含句子。下面是这样一本书的完整树形结构示例。

```
<book>
  <story name="car poll">
    <wheels>4</wheels>
    <para><name>chev</name></para>
    <para><name>ford</name></para>
    <para><name>volk</name></para>
  </story>
  <story name="big class">
    <para><name>ralph</name><height>6</height></para>
    <para><name>billy</name><height>5</height></para>
  </story>
  <story name="cheese" location="NC">
    <chapter name="american">
      <para>1<price>2.00</price><quantity>2</quantity></para>
      <para>2<price>2.10</price><quantity>4</quantity></para>
      <para>3<price>2.20</price><quantity>3</quantity></para>
    </chapter>
    <chapter name="swiss">
      <para>1<price>3.00</price><quantity>3</quantity></para>
      <para>2<price>3.10</price><quantity>2</quantity></para>
    </chapter>
  </story>
</book>
```

该 XML 文档（本书）包含三个短篇故事；每个短篇故事都是一个不同的数据表。在下例中，您选择一次仅导入其中一个数据表。这个特定的 XML 文件在所有表中都使用段落来表示某个表中的行。不需要按段落排列内容。若您想导入 cheese（奶酪）表（它比其他两个表稍微复杂一些），您可能需要一个 variety（品种）列，其中针对每行包含“american”或“swiss”。您可能还需要 price（价格）列和 quantity（数量）列。而且您可能还需要一个列，该列对于 <para> 的 1、2、3、1、2 值没有明确定义的名称。您想要五行。最后，您可能需要一个表变量或一列来表示位置 NC 值。

JSON 文件包含导入为列标题和数据的名称和值对。在下例中，每个字符串中的第一个名称追加到“Grocery Store Purchases”并变为列标题，如“Grocery Store Purchases.Item”所示。“avocado”是表的第一个单元格中的值。下一列名为“Grocery Store Purchases.Category”，“Produce”是第二列的第一个单元格中的值。

```
{"Grocery Store Purchases":[
    {"Item":"avocado", "Category":"Produce"},
    {"Item":"bread", "Category":"Bakery"},
    {"Item":"chocolate", "Category":"candy"}
]}
```

在 “XML 导入向导”中导入 XML 或 JSON 文件

1. 选择**文件 > 打开**。
2. 浏览到 JMP 的 Samples/Import Data 文件夹。

3. 在 Windows 上，将文件类型设置为 XML 数据文件。
4. 选择 Book.xml 并点击打开。

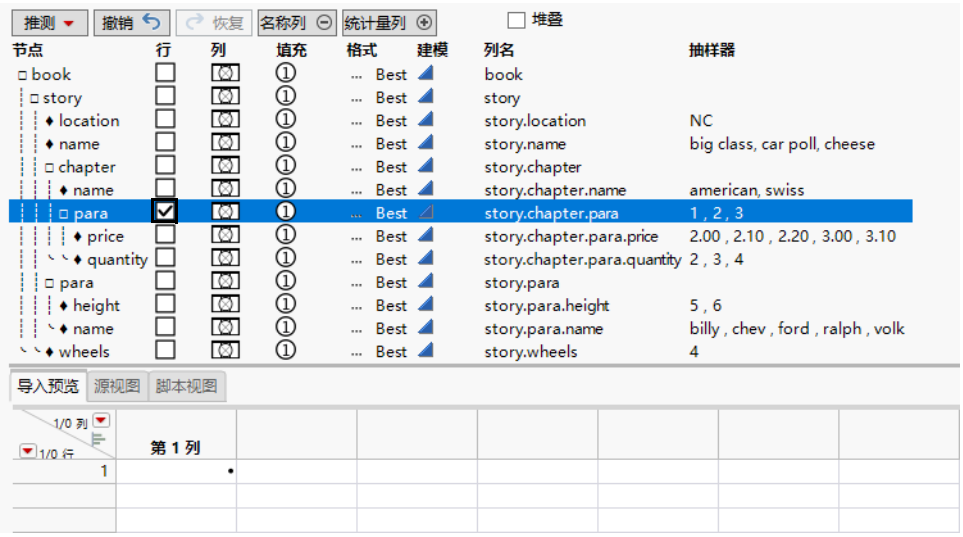
   该文件在 “XML 导入向导”中打开。

注意：在“打开数据文件”窗口中选定了“数据（数据预览）”，以便 XML 文件在向导中自动打开。您还可以直接导入 XML 和 JSON 文件。请参见第 102 页的“直接导入 XML 文件”。

确定如何导入数据

1. 在“行”列中，选择应在数据表中创建新行的元素。在本例中，处理 </para> 时将创建行，并使用行的当前值为该行写入列。

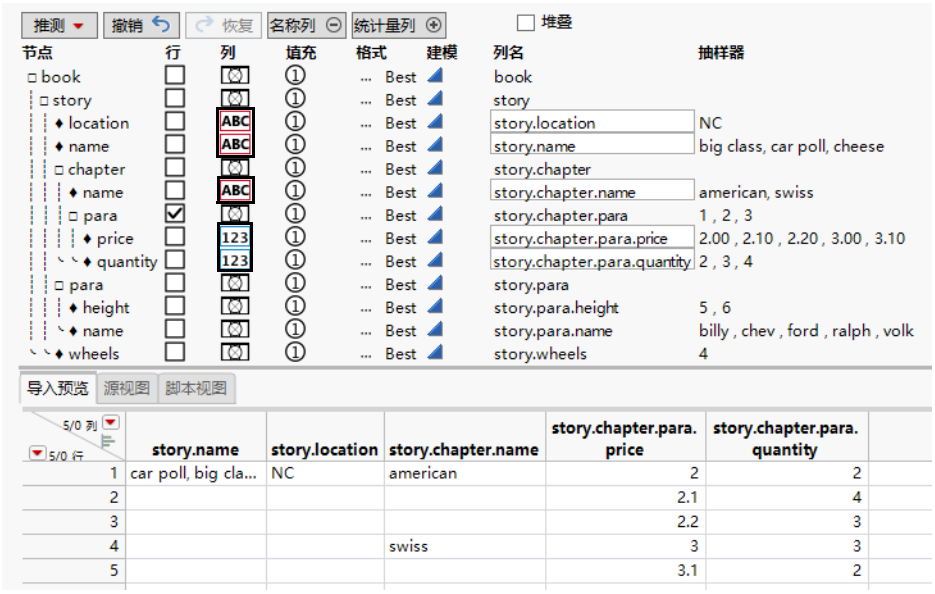
图 3.29 在 “行”列中选定的行元素



2. 在 “列”列中，在 “列”圆圈旁边为该列选择数据类型。
   - 位置：Character
   - 名称：Character（两种情况下皆如此）
   - 价格：Numeric
   - 数量：Numeric

提示：“抽样器”列中的值可让您大致了解应该采用哪种数据类型。

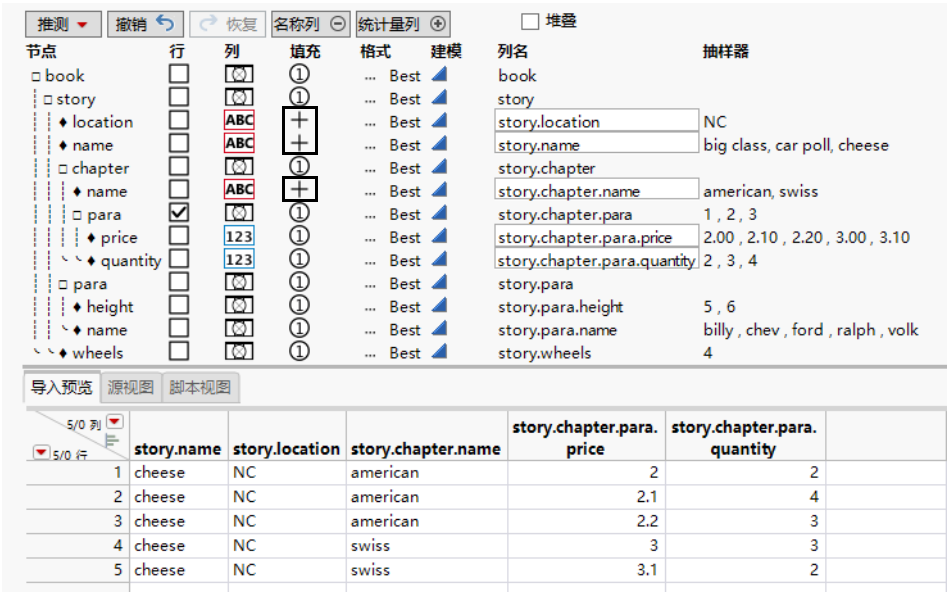
图 3.30 “列”列中分配的数据类型



3. 在 “填充”列中，指定要填充单元格的值。

   对于位置和两种情况下的名称，请点击 “填充”圆圈并选择 +book/story。该设置将使用值填充单元格，直到节点 <book>/<story> 重新开始。

图 3.31 填充在 “填充”列中选定的值

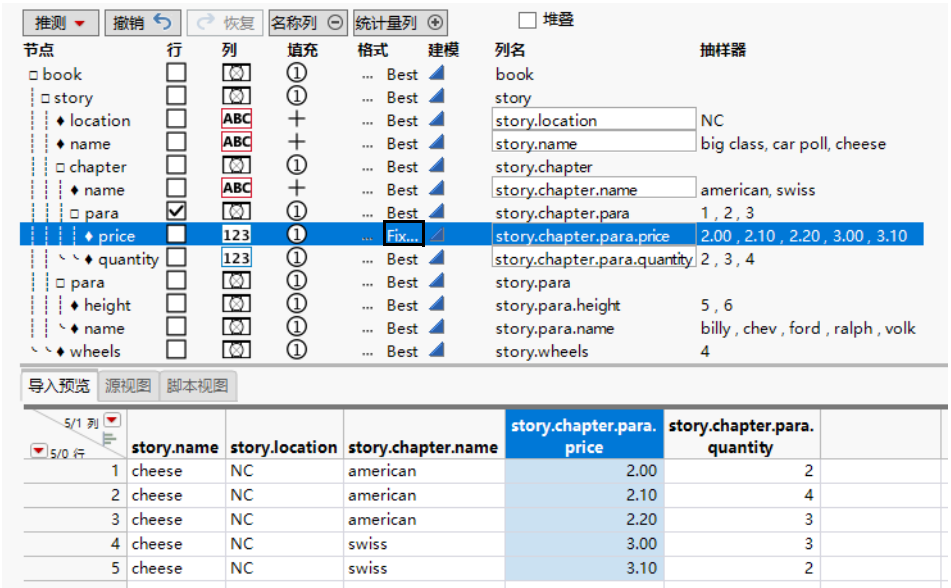


将每一行添加到数据表时，若没有为某列找到新值， JMP 将决定对下一行执行的操作。例如，若没有为每种 cheese 指定章节名称，则在“american”和“swiss”之后跟有缺失值。

- “使用一次”选项意味着只使用一次该值，之后弃用；若不再次提供数据，则第 2、第 3 和第 5 行将缺失。
- “永久填充”选项意味着永远不忘记该值，但是在找到新值时将其替换为新值。
- 您选择本例中的 +book/story 选项。该选项表明：若发现打开的 <book>/<story> 元素，JMP 会填充单元格一直到 <book>/<story> 节点重新开始。-<element> 选项引用该元素的结束标记，这将是该节点终止的点。

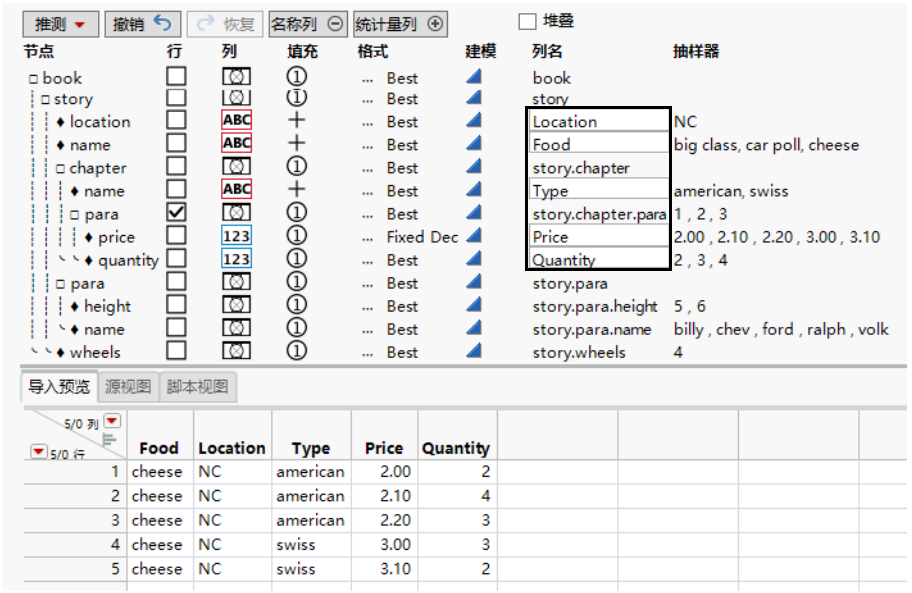
4. 在“格式”列中，点击 price 旁边的 Best，选择 Fixed Dec，为小数位数键入“2”，然后点击**确定**。

图 3.32 “格式”列中选定的格式类型



5. 在“列名”列中，按图 3.33 所示重命名各列。

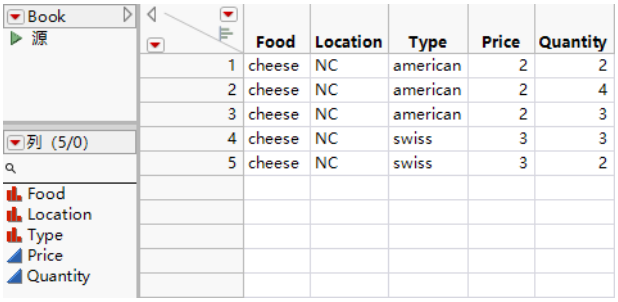
图 3.33 “列名”列中已重命名的列



6. 在该窗口的底部，点击**确定**。

   数据将按照您的指定导入数据表。

图 3.34 最终数据表



查看“XML 导入向导”中的统计量

点击“XML 导入向导”顶部的**统计量列**以显示元素的统计量。

出现次数 元素出现的次数。根元素只出现一次，通常不是从中生成行的元素。您得到的行数将与该元素出现次数相同。

若选择多行，那么实际的行数可能小于“出现”的总次数；在另一个元素结束之后结束的元素不会生成额外的行。例如，若同时将 <para> 和 <chapter> 选作行标记，当这一章就在最

后一段之后结束时，就不会有额外一行。但若在最后一段之后及这一章结束之前有一个用于 <chapter> 的 <footnote>，则会得到额外一行。您必须选择脚注列才会出现这种情况。

写入次数 源具有可以写入的值的次数。若该值为零，则您可能不希望从该元素生成列，因为这些值都将缺失。

唯一值数 唯一值的数目 （小于或等于 “写入”值）。

重复次数 该元素在其父元素中重复的最大次数。较大的重复值可能用于生成行。若该元素重复并用作列，则字符列将生成一个列表；表达式列可能更完善，因为它使用 {lists}（在大括号内）来指示值的停止和开始位置。数值列将只得到其中一个重复值。

抽样器 元素的值。

“XML 导入向导”选项

推测 JMP 根据您选择的数据表的大小选择设置。**高型推测**选择将生成尽可能多行的元素 （默认设置）。**宽型推测**选择将生成尽可能多列的行的元素。**大型推测**选择尽可能多的元素来生成它所需的行，以将所有数据放入表中；它可能比您预期的要大。

撤销 撤销上次对该窗口所做的更改。

恢复 重新调用上一次对窗口进行的更改。

名称列 显示或隐藏 “列名称”列。

统计量列 显示或隐藏树形结构中的元素的统计量。

堆叠 应用于在要创建行的父节点中重复的节点。默认情况下，额外值存储在由逗号分隔的单个表单元格中。若选定此选项，重复值将堆叠在额外的行中。

导入预览 根据您选择的选项显示导入数据的预览。

源视图 显示源 XML 文档。

脚本视图 显示您选好设置的 JSL 脚本。

直接导入 XML 文件

JMP 可以检测 XML 或 JSON 文件的结构，并将其直接导入数据表。若不想在“XML 导入向导”中看到数据预览，请使用此方法。

1. 在 Windows 上：
   - 选择**文件 > 打开**。
   - 将文件类型设置为**数据文件**。
   - 选择文件。
   - 在“打开方式”旁边，选择**数据（最佳推测）**。
2. 在 macOS 上：

– 选择**文件 > 打开**，然后选择文件。
– 点击窗口左下角的**选项**。
– 选择**数据（最佳推测）**。

3. 点击**打开**。

该文件将作为数据表打开。

若您对结果不满意，请考虑使用“XML 导入向导”定制导入。请参见第 96 页的“预览和导入 XML 和 JSON 数据”。

注意：通过选择“编辑”>“作为数据导入”，您可以从“脚本编辑器”窗口导入 XML 或 JSON 文本。若文件中未选择任何内容，则导入将基于文件扩展名。若选定了文本，JMP 则会尝试将文本解析为 JSON 或 XML；否则，文本将导入 CSV 文件。

直接导入 JSON 文件

JSON 文件包含导入为列标题和数据的名称和值对。您可以使用“XML 导入向导”导入 JSON 文件（第 96 页的“预览和导入 XML 和 JSON 数据”），也可以直接导入。

要直接打开 JSON 文件，请执行以下步骤：

1. 在 Windows 上：
   – 选择**文件 > 打开**。
   – 将文件类型设置为 **JSON 数据文件**。
   – 选择文件。
   – 选择**打开方式**旁边的**数据（最佳推测）**。
2. 在 macOS 上：
   – 选择**文件 > 打开**，然后选择文件。
   – 点击窗口左下角的**选项**。
   – 选择**数据（最佳推测）**。
3. 点击**打开**。

该文件将作为数据表打开。

注意：

- 导入的 JSON 文件不包含用于确定列格式的日期、时间、货币、地理、百分比和科学信息。若要使用这些格式中的任何一种，请右击数值列，选择**列信息**，然后在“数据类型”列表中更改格式。
- 名称-值对中的成员名称不区分大小写。
- JSON 导入代码不使用 JSON 文件的数值、布尔值或字符串区分方式。相反，它确定整列是数值还是非数值。若猜测非您所想，则可以将列明确指定为字符列。

- 对于嵌套项，JMP 会为每行重复外部项。
- 该文件不得包含空的数组元素。名称必须用引号引起来，放在名称-值对中的冒号之前。
- JSON 标准接受带小数点的小数。

在“查询生成器”中生成 SQL 查询

“查询生成器”是用于从 SQL 数据库中选择并导入数据而无需编写 SQL 语句的首选方法。将数据导入数据表之前，您可以对数据进行预览。共享您的查询，以便其他用户可定制并运行查询。

“查询生成器”通过使用“文件 > 数据库 > 打开表”功能提供了另外一种方法来编写您自己的查询。不过，您也可以在“查询生成器”中生成查询，然后添加您自己的 SQL 语句。

“SAS 查询生成器”还可用于查询 SAS 服务器上的 SAS 数据集。请参见第 132 页的“使用“SAS 查询生成器”打开 SAS 数据集”。

注意：

- 不支持数据库表名称包含字符 `$# -+/%()&|;?`。
- 不支持“值顺序”列属性。考虑在“查询后脚本”窗口中通过编写脚本来对行排序。

连接至 SQL 数据库

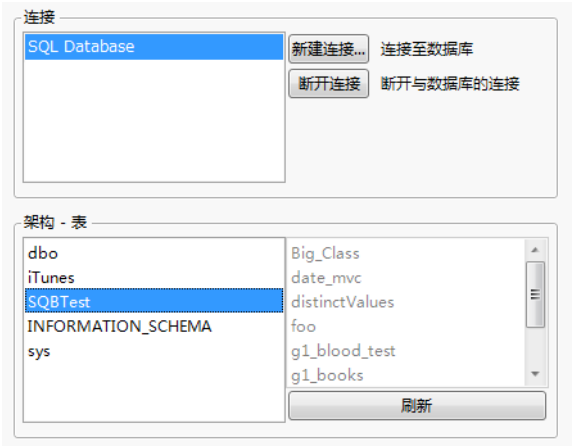
通过 Windows 中的“控制面板”或在 JMP 中设置 ODBC 连接。

1. 选择**文件 > 数据库 > 查询生成器**以显示“选择数据库连接”窗口。

   “连接”框列出了您在当前 JMP 会话中连接到的数据源。
2. 若所需数据源**未**列在“连接”框中，请点击**新建连接**以选择一个数据源。选择数据源的方法取决于您的操作系统和 ODBC 驱动程序。请参见第 147 页的“连接至数据库”。
3. 从“架构 - 表”框中选择一个表或架构，然后点击**下一步**。

   “查询生成器”示例基于名为 SQBTest 的表给出，该表包含电影租赁数据。

图 3.35 选择数据库架构



提示：要在长列表中查找表，请在架构上方的搜索框中输入名称。您还可以在表的列表上方搜索表。红色小三角菜单提供了用于区分大小写和使用正则表达式进行搜索的选项。

从 SQL 数据库中选择表

连接至 SQL 数据库后，选择您想要查询的表。选择一个主表，或连接多个表以查询全部。

默认情况下，JMP 尝试根据表中分配的键关系连接表。

- 主键标识唯一描述数据（例如，客户 ID 编号）的列。主表中的所有行都包括在您的查询中。
- 副表中的外键与其中一个连接表的主键匹配。仅副表中的匹配行包含在您的查询中。

若不存在键，则按列名匹配数据，这会连接两个表。默认情况下，仅副表中的匹配行包含在您的查询中。

该示例显示如何连接多个表。不过，您也可以使用单个表生成查询。在这种情况下，连接不是必需的。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的“连接至数据库”。
2. 从 “架构 - 表”列表中选择所需架构。

   若还选择了表，该表将在您点击 “下一步”后成为主表。
3. 在 “选择要查询的表”窗口中，选择 “可用表”列表中的 g6_Customers，然后点击**主表**。

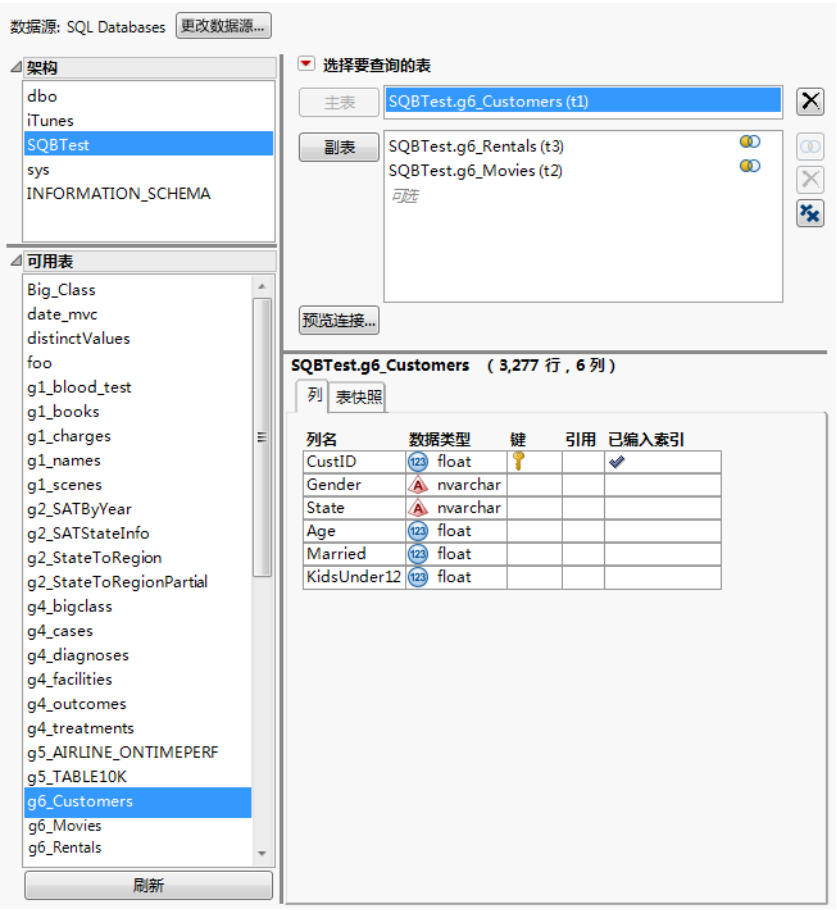
   “列”选项卡显示 CustID 是主键。数据已编入索引，可加快查询速度。
4. 从 “可用表”列表中选择 g6_Movies 和 g6_Rentals，然后点击**副表**。

“左连接”图标 指示这些表已自动连接 （图 3.36）。 CustID 是 g6_Customers 中的主键，且与另外一个表的外键匹配。

提示：添加主表或副表后，点击**添加相关表**可添加包含匹配列的表。当未找到相关表或者未选择主表或副表时，该按钮不可用。

图 3.36 中显示了完成的窗口。

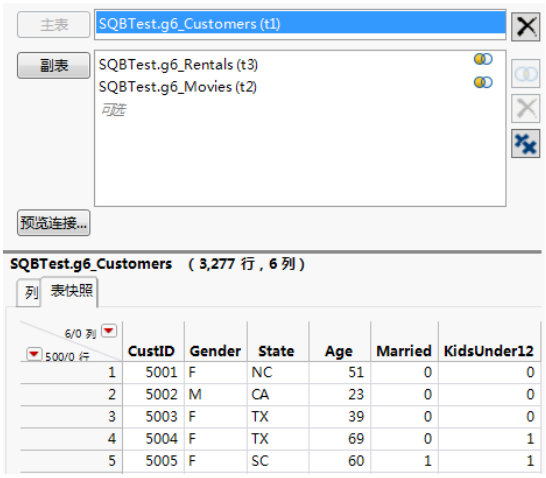
图 3.36 选择主表和副表



提示：要在长列表中查找架构，请在架构上方的搜索框中输入名称。您还可以在表的列表上方搜索表。红色小三角菜单提供了用于区分大小写和使用正则表达式进行搜索的选项。

5. 点击每个表的**表快照**选项卡以预览数据。

图 3.37 “表快照”选项卡



6. 在主表和副表下方，点击**预览连接**以查看从指定连接创建的表预览。

提示：

- 副表旁边的 图标指示该表未在查询中连接。点击**编辑连接**按钮 以指定要连接的列。若您无法找到要连接的列，请点击**删除**按钮 以删除该表。请参见“编辑连接表的条件”。
- 在“列”选项卡中，“键”列可能会显示多个键；有些列可同时为主键和外键。主键旁边不会显示唯一键图标，因为所有的主键都是唯一的。
- 在“列”选项卡中，为与另一个架构中的主键相匹配的外键指定了“引用”。该引用为架构和列的名称。
- 点击**更改数据源**可查询其他架构或数据库。
- 要连接不同源（例如，数据库和 Microsoft Excel）中的数据，请使用“查询生成器”将数据库数据导入某个数据表；将 Excel 数据导入某个数据表；然后使用“表”菜单中的“JMP 查询生成器”查询并连接这两个表。
- 导入包含主键的某个表时，“链接 ID”列属性将添加至数据表中的列。该列属性支持您虚拟连接数据表。请参见“重塑数据”一章中第 324 页的“虚拟连接数据表”。

编辑连接表的条件

在“选择要查询的表”红色小三角菜单中，最初选定了“自动连接数据库表”。JMP 根据键关系或匹配列名自动连接数据库表。

若不存在键，或者列名不匹配，请点击**编辑连接** 以指定要连接的列。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

编辑连接表的条件

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的 “连接至数据库”。
2. 在 “选择要查询的表” 窗口中，选择 g1_books 作为主表， g1_charges 作为副表。

   副表旁边的 图标指示该表未在查询中连接。
3. 选择 “副表” 窗格中的 g1_charges，然后点击**编辑连接** 。

   随即显示 “添加条件” 窗口。
4. 在 “左侧列” 列表中，选择 g1_books。
5. 从 “左侧列” 框中选择 Book ID。
6. 从 “右侧列” 框中选择 ID。
7. 确保在两个框之间选择了等号。

图 3.38 “添加条件” 窗口

8. 点击**下一步**。

   “编辑连接” 窗口显示 g1_books 中的不匹配行将包含在数据表中。仅忽略 g1_charges 中出现的行。

   要进行完全连接并导入所有行，您需要选择**包括下表中的不匹配行 : g1_charges**。若仅有一个不匹配选项可用，数据库将不支持完全连接。
9. 点击**确定**。

注意：在连接所有副表之后，**确定**按钮才可用。

阻止表自动连接

- 从主表上方的 “选择要查询的表” 红色小三角菜单中，取消选择**自动连接数据库表** 。
- 若您频繁查询大型数据库，请取消选择**首选项 > 查询生成器**中的**自动连接添加至查询的表**，以防止出现内存问题。

生成 SQL 查询

选择数据库表后，您可以导入数据或生成查询。“查询生成器”可让您通过交互方式创建数据库查询，而无需编写 SQL 表达式。

选择数据库表和连接表（若有必要）后，点击**生成查询**以打开“查询生成器”窗口。通过选择要包含的列以及指定抽样和过滤条件，您可以继续改进查询。您也可以保存查询，以便以后再次编辑和运行。

所有数据库表中的列将显示在“可用列”列表中。诸如 t1 和 t2 的前缀（也称为**别名**）将每列与相应的数据库表相关联。

要跳过“查询生成器”步骤并导入所有数据，请点击**立即导入**。

注意：“表”菜单中的“JMP 查询生成器”提供许多相同的选项，但支持您查询和连接 JMP 数据表。请参见“重塑数据”一章中第 321 页的“使用 JMP 查询生成器查询和连接数据表”。

从数据库表中选择列

假设您想按电影类型、分级和人口统计数据（例如，婚姻状况和年龄）来查看电影租赁情况。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的“连接至数据库”。
2. 在 “选择要查询的表”窗口中，选择 g6_Customers，然后点击**主表**。
3. 选择 g6_Movies 和 g6_Rentals，然后点击**副表**。
4. 点击**生成查询**以显示 “查询生成器”窗口。
5. 在“可用列”框中，选择 t1.Gender、t1.Age、t1.Married、t1.KidsUnder12、t2.Rating 和 t2.Genre。
6. 在 “已包括的列”选项卡上点击**添加**。

图 3.39 选定列

注意：

- 若查询中包含八个或八个以上的表，搜索框会显示在左上角附近的表名上方。要在长列表中查找一列，请在“可用列”下方的搜索框中输入名称。红色小三角菜单提供了用于区分大小写和使用正则表达式进行搜索的选项。要隐藏搜索框，请从“可用列”红色小三角菜单中取消选择**显示搜索框**。
- 引用列不再显示在“可用列”列表中，因为查询中不支持它们。
- 您可以更改列宽，只需从“已包括的列”列表的“格式”列表中选择一个选项即可。

7. 选择列下方的 SQL 选项卡以查看用于查询的 SQL 语句。在您运行查询后，该代码将保存为数据表属性。
8. 点击右下角的**保存**。

   您的工作将保存为 g6_Customers.jmpquery 文件，您可以稍后打开该文件以返回至此处或运行该查询。
9. 点击**运行查询**以导入数据。

   数据表包括以下脚本：

   - 运行“源”脚本可重新连接至数据库并导入数据。
   - 运行“修改查询”脚本可在“查询生成器”中打开查询。
   - 运行“从数据库更新”脚本可重新导入并刷新数据。

提示：

- 要重命名列，请在“已包括的列”选项卡中双击“JMP 名称”并输入一个新名称。
- 要重命名别名，请在“选择要查询的表”窗口中右击该表，然后选择**更改别名**。别名不区分大小写。
- 查询将在后台运行，除非您在“查询生成器”的 ODBC 首选项中取消选择**在后台运行查询（若可能）**。您还可以通过选择**视图 > 正在运行的查询**来检查所有 ODBC 查询的进度。

注意：对于 “SAS 查询生成器”，所有查询都在前台运行。

- 若预览加载过慢，请取消选择**自动更新预览**。点击“查询预览”选项卡下方的**更新**可更新数据视图。若您经常使用大型数据库，请考虑更改“JMP 查询生成器”首选项中的“预览”选项。考虑限制可以预览的最大行数。在“JMP 查询生成器”首选项中，更改**要预览的最大行数**的值。
- 要忽略数据库中的重复行，请在“已包括的列”选项卡中选择**仅非重复行**。

保持与 JMP 12 兼容

若向查询添加 JMP 16 功能，该查询将不再能在 JMP 12 中加载。若使用的是 JMP 13，但您需要创建仍能在 JMP 12 中运行的查询，请在“查询生成器”首选项中选择**令该查询保持与 JMP 12 兼容**。选择该选项后，导致兼容性问题的功能将在“查询生成器”中隐藏。

准备好将 JMP 12 查询移至之后的版本后，取消选择该首选项。

创建计算列

您可以根据现有列创建新列。您可以计算两列的均值，然后将均值存储在一个新列中。日期时间值的格式可能不正确。从“可用列”红色小三角中，选择**添加计算的列**，然后在公式编辑器中创建新列。

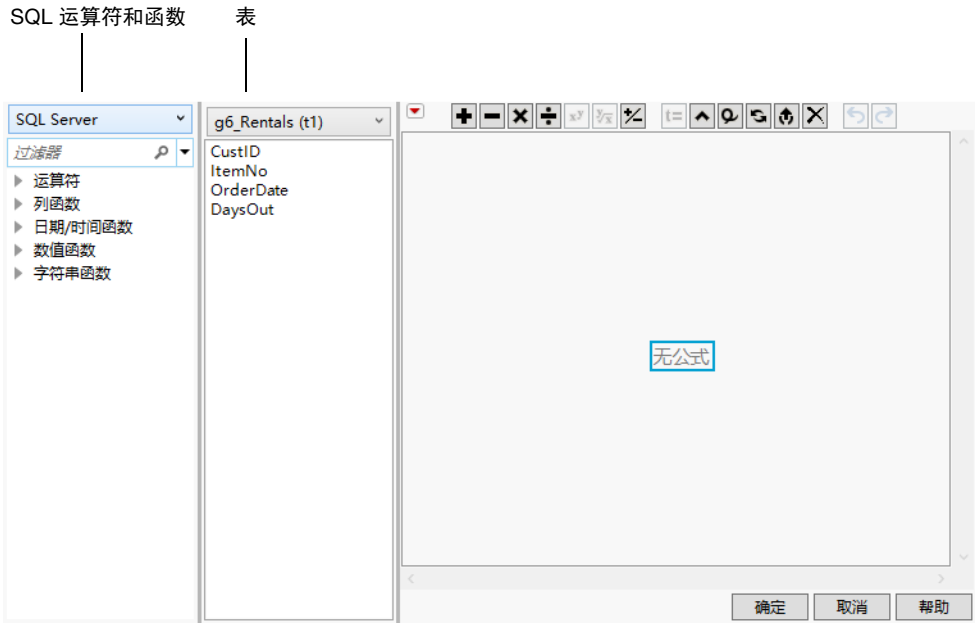
假设您想计算在租赁期内您最多可以观看多少次电影。您要查询的数据库包含每部电影的长度和该电影外借天数。本例显示了如何根据这些数据创建新的计算列。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的“连接至数据库”。
2. 在 “选择要查询的表”窗口中，选择 g6_Rentals 作为主表， g6_Movies 作为副表。
3. 点击**生成查询**以显示 “查询生成器”窗口。
4. 点击 “可用列”红色小三角并选择**添加计算列**。

   随即显示 “计算列”窗口。该窗口包含 “JMP 公式编辑器”。

图 3.40 包含公式编辑器的“计算列”窗口



注意：

- “公式编辑器”左侧的列表中提供了运算符和函数（图 3.40）。在某些实例中，您需要基于数据库更改服务器类型。
- “运算符”列表不提供拼接 (||) 运算符。您必须在“公式编辑器”框中键入公式。

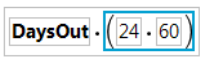
5. 从左侧的 g6_Rentals 列表中选择 Days Out，然后点击乘号 ✖ 按钮。

图 3.41 计算列

6. 选择空白框，键入 (24 * 60，然后按 Enter 键。

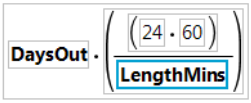
   该公式将一小时内的分钟数乘以一天内的分钟数。请注意，当您键入第一个圆括号时，第二个圆括号将会自动插入。

图 3.42 公式的第一部分



7. 点击外部框以选择整个等式，然后点击除号 ➗ 按钮。
8. 从左侧列表中选择 g6_Movies，然后选择 LengthMins。

图 3.43 公式的第二部分



9. 点击**确定**。

   随即创建了名为 Calc1 的新列。

10. 右击该列，然后选择**重命名列**。
11. 键入 MaximumTimesWatched，然后点击**确定**。
12. 在 “可用列”列表中，选择 MaximumTimesWatched，然后点击**添加**。
13. 选择 t2.Name，然后点击**添加**。

    在 “查询预览”选项卡中，注意到 “Nanny McPhee”可在电影租借期内观看 160 次。

对常用值进行分组

将数据导入 JMP 前，您可以合并常用值（或对其进行分组）。要对常用值进行分组，请选择 “聚合”函数以确定如何计算常用值。

注意：聚合支持基于您的数据库。请参见数据库文档。

假设您关注特定电影的租赁次数。在本例中，将计算每一个项目编号的计数，并将常用电影值分组至单个行中。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。详细信息，请参见第 147 页的 “连接至数据库”。
2. 在 “选择要查询的表”窗口中，选择 g6_Movies 作为主表， g6_Rentals 作为副表。
3. 点击**生成查询**以显示 “查询生成器”窗口。
4. 在 “可用列”框中，选择 t1.Name 和 t2.ItemNo，然后点击**添加**。
5. 选择 t2.ItemNo，然后从 “聚合”列表中选择**计数**。

   t1.Name 对应的 “分组依据”复选框已选中。特定电影名称的所有实例将被分组在一行中。

图 3.44 分组列

6. 点击**运行查询**以导入数据。
7. 在数据表中，右击**计数 -ItemNo** 列，然后选择**排序 > 降序**。
   “Scarface”租借频率最高。

图 3.45 已排序的 “计数 -ItemNo”列的部分视图

	Name	计数-ItemNo
1	Scarface	15991
2	Rambo	15643
3	Nanny McPhee	15289
4	Marmaduke	15255
5	A Bug's Life	15128
6	Diary of a Wimpy Kid	14989
7	Cars	14905

提示：

- 要清除已分组的行，请从列的“聚合”列表中选择**无**。
- DISTINCT 聚合函数仅显示包含非重复值的行。包含重复值的行将被忽略。当数据库中包含很多重复值时，这些函数很有用。

导入数据样本

对于大型数据库，考虑对数据进行抽样。抽样可返回行的子集并缩短查询时间。数据库查询运行，并且根据您在“样本”选项卡上选择的选项导入一小部分数据。

抽样方法因数据库供应商而有所不同。

- SQL 服务器默认支持分区抽样。分区样本包含一整页的行（例如第 1 页和第 5 页上的所有行）。若您选择 1,000 行，则导入约 1,000 行。
- Oracle 和其他数据库支持行抽样。若您选择 5,000 行，则每个样本通常可导入 4,800 至 5,200 行，具体取决于 Oracle 如何循环数据。

对于主要数据库供应商，JMP 可对功能进行检测，并在可能的情况下提供特定于供应商的选项。供应商不支持的功能在“样本”选项卡上不可用。

假设您想导入数据样本。在本例中，您选择 5,000 个随机行。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。详细信息，请参见第 147 页的“连接至数据库”。
2. 在 “选择要查询的表”窗口中，选择 g6_Rentals 作为主表， g6_Movies 作为副表。
3. 点击**生成查询**以显示 “查询生成器”窗口。
4. 在 “已包括的列”选项卡上，点击**全部添加**按钮。
5. 点击**样本**选项卡，然后选择**对该结果集抽样**。
6. 选择**随机 N 行**，然后输入 5,000。

   在 “抽样方式”区域中，根据数据库支持的抽样类型，“分区或页面”是唯一选项。
7. 点击**运行查询**以导入数据。

   新的数据表包含约 5,000 行。通过分区或页面抽样，您获得的样本第一次可能包含 4,900 行，下一次可能包含 5,600 行。

提示：

- 要在每次运行查询时重新创建相同的样本集，请将“种子”值设置为不大于 64,000 的正整数。假设您想按性别查询电影租借情况。键入 1 作为“种子”值，然后运行查询。结果中男性客户的分布过低。键入 2 作为“种子”值，然后再次运行查询。重复该过程以找到生成男性和女性相似分布情况的“种子”值。
- 要在“已包括的列”选项卡中添加单个列，右击该列然后选择**包括列**或点击**添加**按钮。

选择过滤器以导入数据子集

添加过滤器可将选定过滤器中的值的一个子集导入数据表。若查询与 JMP 12 兼容，某些过滤器则不可用。请参见第 111 页的“保持与 JMP 12 兼容”。

用于连续和分类列的过滤器

简单比较 使用指定的运算符匹配值。

年龄 > 14 匹配大于 14 岁的年龄。

范围 使用指定的运算符匹配一定范围内的值。

12 ≤ 年龄 ≤ 17 匹配 12-17 岁之间的年龄。

Is NULL 或 Is Not NULL 匹配缺失值。

NULL 或 not NULL 匹配缺失值和非缺失值。

自定义表达式 支持您自行编写 SQL 表达式。

```
( ( ( t2.Gender IN ( 'F' ) ) AND ( (t2.Age >= 20) AND (t2.Age <= 50)
  ) ) )
```

匹配 F 性别。它还匹配 20-50 岁之间的年龄。

仅用于分类列的过滤器

列表框 显示一个列表框，从中可选择一个或多个值。“列表框”是基于“查询生成器”首选项的分类列的默认过滤器。

手动列表 支持您输入列值。

复选框列表 显示复选框列表。

注意：“列表框”、“手动列表”和“复选框列表”都包括一个**未在列表中**选项，该选项支持您检索与选定值不匹配的行。

包含 匹配包含或不包含指定值的字符串。 Contains Comedy OR Romance 匹配 Comedy 和 Romance。

对于大多数分类列，过滤器默认为一个列表框。对于包含超过 1,000 个水平的列，会自动选择“包含”过滤器。您可以在“查询生成器”首选项中更改水平数。

Like 或 Not Like 匹配与指定值相似或不相似的字符串。支持 % 通配符（0 或更多字符）和 _ 通配符（恰好一个字符）。

Genre Like %com 匹配“com”之前任意个数的字符，就像在“RomCom”中一样。若还要匹配“Comedy”，请使用 %com% 或 Contains com。

匹配列值 匹配指定的列值。先选择表，再选择列。**选择非匹配**选项支持您过滤除选定行之外的所有行。相关示例，请参见第 118 页的“从现有数据表中导入匹配数据”。

本例显示如何导入满足以下条件的数据：年龄为 30 岁以上（包括 30 岁）的客户，类型为 RomCom 和 Comedy 的电影。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

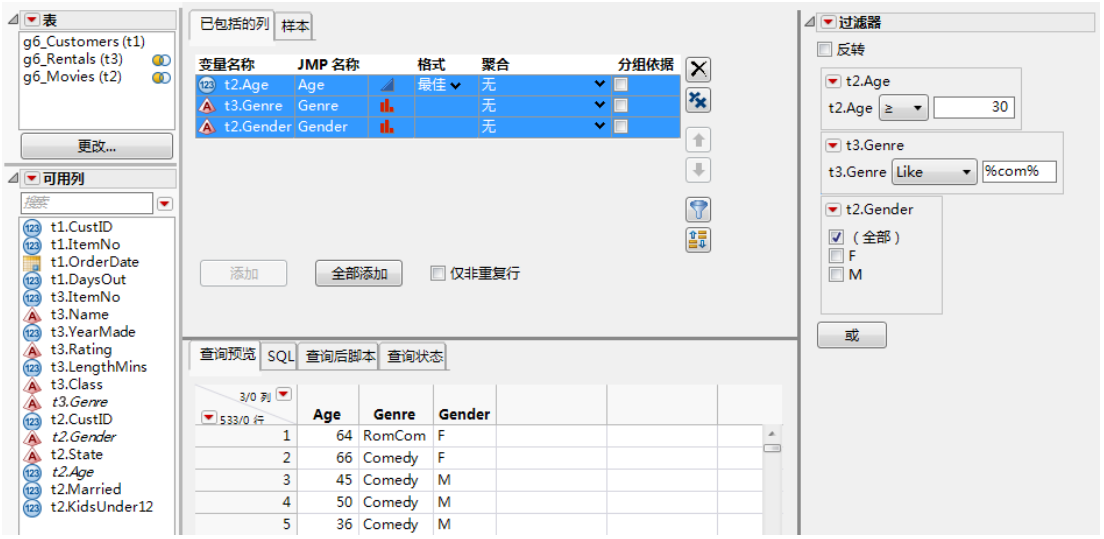
---

1. 选择**文件** > **新建** > **数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的“连接至数据库”。
2. 在 “选择要查询的表”窗口中，选择 “可用表”列表中的 g6_Rentals，然后点击**主表**。
3. 选择 g6_Customers 和 g6_Movies，然后点击**副表**。
4. 点击**生成查询**以显示 “查询生成器”窗口。
5. 在 “可用列”框中，选择 t2.Gender、 t2.Age 和 t3.Genre，然后点击 “已包括的列”选项卡中的**添加**。
6. 选择 “已包括的列”选项卡中的所有列，然后点击**将选定项添加至过滤器** 。

   适用于这些列的过滤器随即出现在 “过滤器”分级显示项中。
7. 将 t2.Age 过滤器设置为 “≥ 30”。
8. 点击 t3.Genre 红色小三角并选择 Like，键入 %com%，然后按 Enter 键。

   % 通配符与 “com”之前和之后的任意字符数匹配。在 “查询预览”选项卡中，注意到类型为 “RomCom”和 “Comedy”的电影会显示出来。

图 3.46 选择过滤器



9. 点击 “过滤器”红色小三角并选择**全部运行时提示**。

   运行查询的用户可定制这些过滤器。
10. 点击**运行查询**。
11. 在 “查询提示”窗口中，点击**确定**以应用预选的过滤器并导入数据。

注意：

- 对于大多数分类列，过滤器默认为一个列表框。对于包含超过 1,000 个水平的列，会自动选择“包含”过滤器。您可以在“查询生成器”首选项中更改水平数。
- 过滤器的红色小三角菜单中的“条件”选项支持您过滤分层类别中的数据。例如，假定您有一个“州”过滤器和一个“城市”过滤器。要选择某个州，然后只显示该州中的城市，请点击“城市”红色小三角并选择“条件”。
- “过滤器”列表顶部的“反转”选项支持您为所有过滤器选择指定行以外的所有行。该选项不可用于选择所有行的过滤器。
- 过滤器列表中的“<空>”指示对于该列，数据库包含缺失值。
- 要针对大量分类数据列创建过滤器，JMP 会尝试确定表中的行数。
  - 默认选定称为**为无法确定大小的表检索类别水平**的“查询生成器”首选项，以便 JMP 自动检索水平。若取消选择该首选项，则会选定“查询生成器”首选项中的“包含”备选过滤器类型。
  - 若分类列超过一百万行，JMP 不会自动检索过滤列的唯一类别水平。称为**表中将要自动检索类别水平的最大行数**的“查询生成器”首选项支持最小值为 -1（无限制），最大值为 10 亿行。
- 除非选定**令该查询保持与 JMP 12 兼容**“查询生成器”首选项，否则分类列的默认过滤器是列表框。

从现有数据表中导入匹配数据

您还可以从打开的数据表中选择与查询中的列相匹配的行。考虑包含航空公司数据的数据库。该数据库中包含飞行时间和机尾编号之类的数据。您还有一个包含机尾编号数据的数据表。使用“匹配列值”过滤器可以仅导入机尾编号匹配的数据。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**帮助 > 样本数据库**，然后打开 Air Traffic.jmp。
2. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的 “连接至数据库”。
3. 在“选择要查询的表”窗口中，选择“可用表”列表中的 g5_AIRLINE_ONTIMEPERF，然后点击**主表**。
4. 点击**生成查询**以显示“查询生成器”窗口。
5. 在“已包括的列”选项卡中，点击**全部添加**。
6. 选择“已包括的列”选项卡中的 t1.TailNum，然后点击**将选定项添加至过滤器** 。
7. 点击“过滤器”列中的 t1.TailNum 红色小三角，选择**过滤器类型**，然后选择**匹配列值**。
8. 选择**匹配表中的值**下方的 Air Traffic。
9. 选择**机尾编号**列，然后选择列表中的**所有行** (38,118)。

“查询预览”选项卡中的数据视图会更新以显示已过滤的值。

10. 点击**运行查询**以导入数据。

    数据表仅包括**机尾编号**列中包含的那些行的数据。

编写自定义表达式以导入数据子集

若不想通过选择所提供的过滤器来生成数据子集，您还可以编写自定义 SQL 表达式来生成数据子集。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

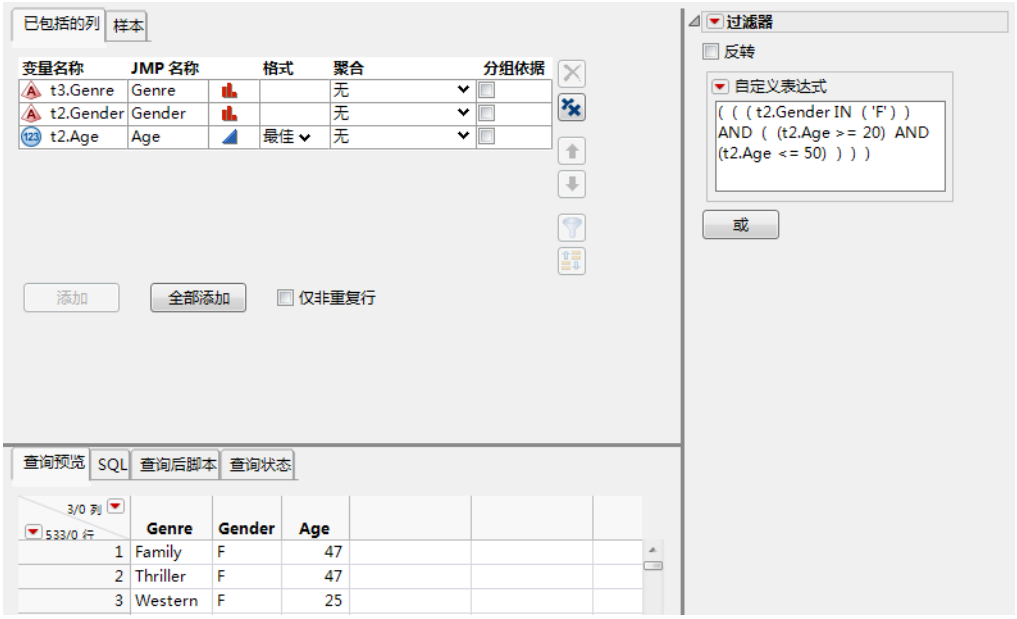
1. 选择您想过滤的列（在第 115 页的“选择过滤器以导入数据子集”中有相关说明）。
2. 点击 “过滤器”红色小三角并选择**添加自定义表达式**。
3. 在 “自定义表达式”框中键入以下文本：

```
( ( ( t2.Gender IN ( 'F' ) ) AND ( (t2.Age >= 20) AND (t2.Age <= 50)
) ) )
```

4. 在 “自定义表达式”框外部点击，以更新 “查询预览”选项卡。

   该表达式匹配 F Gender。它还匹配 20-50 岁之间的 Age。

图 3.47 编写自定义过滤器表达式



对选定数据进行排序

您可以将特定列中的行按值排序，以控制数据在数据表中的显示方式。在本例中，您对“Married”列按降序排序，然后先后按年龄和身高对数据进行排序。

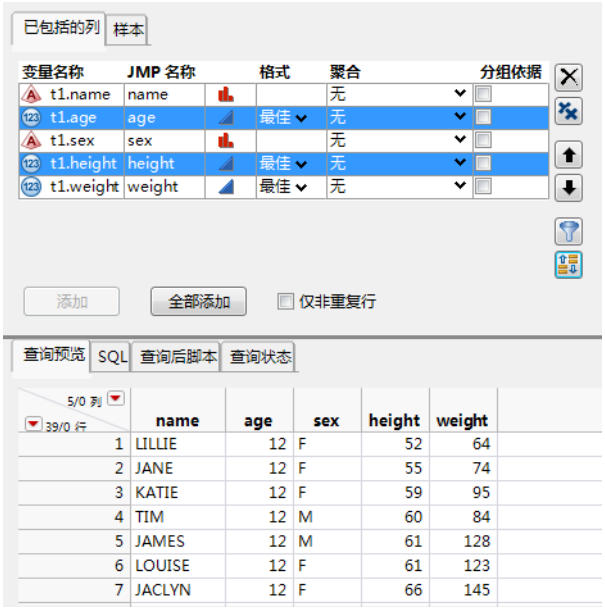
注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。详细信息，请参见第 147 页的“连接至数据库”。
2. 在“选择要查询的表”窗口中，选择 g4_bigclass 作为主表。
3. 点击**生成查询**以显示“查询生成器”窗口。
4. 在“已包括的列”选项卡中，点击**全部添加**。
5. 选择 t1.age 和 t1.height，然后点击**按选定项排序** 。

   这些列显示在右侧列的“排序方式”分级显示项中。

   这些列首先按 age（从小到大）排序，然后按 height（从低到高）排序。

图 3.48 选择“排序方式”列

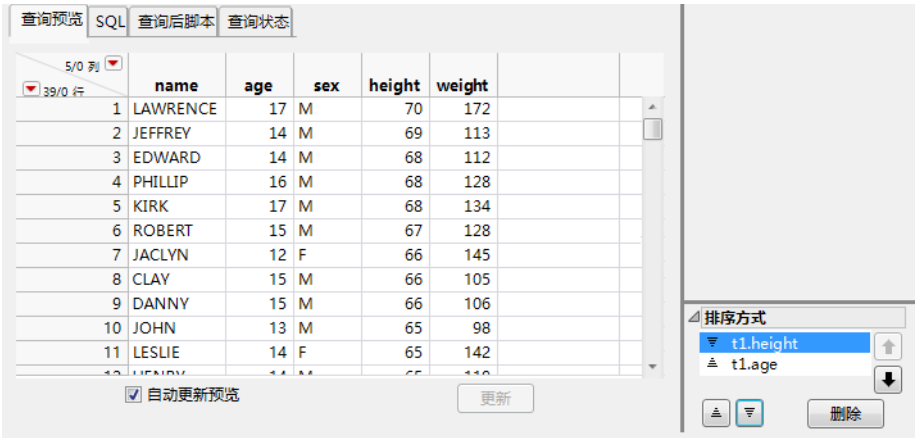


6. 在“排序方式”分级显示项中，选择 t1.height，然后点击这些列下方的**将值按降序排序** 。

   height 列按从高到低进行排序。
7. 选择 t1.height，然后点击**在列表中上移选定项** 。

   height 列先排序。age 列中的值在 height 的每个水平内排序。若 height 为 68，则 age 从 14 到 17 进行排序。

图 3.49 列重新排序后的结果



查看查询状态

在“查询状态”选项卡中，查看后台运行的查询状态。显示查询名称、SQL 语句和已处理的记录数。您可以随时停止查询，仅查看已处理的记录。要从其他 JMP 窗口查看后台查询，请选择**视图 > 正在运行的查询**。若您从“查询生成器”首选项中取消选择**在后台运行查询（若可能）**，则不会提供状态详细信息。

注意：对于“查询生成器”中的 SAS 连接，所有查询和查询预览都在前台运行。

编写查询后脚本

在“查询后脚本”选项卡中编写 JSL 脚本，该脚本将在您运行查询后运行。例如，您可能想要导入数据，然后创建分布。

```
Distribution( Column( :age, :gender ) );
```

该脚本是最终数据表中的“源”脚本的一部分。

保存和运行 JMP 查询

将您的查询保存为 .jmpquery 文件，以便稍后修改或运行该查询。若服务器连接字符串未指定密码，系统将提示您输入密码。.jmpquery 文件也可通过 JSL 脚本打开和运行。

在生成查询后，您可以点击右下角的**保存**将设置保存为一个 .jmpquery 文件。再次点击**保存**将会用您的最新更改覆盖该文件。点击**另存为**可将查询保存在新的 .jmpquery 文件中。

运行 JMP 查询

查询文件默认在“查询生成器”中打开，您可在其中修改该查询，然后按需运行查询以导入数据。若对查询满意，您可以配置好在打开时即运行它，而不是在“查询生成器”中打开。

生成查询时，请从左上角的“查询名称”框旁边的红色小三角菜单中选择**打开即运行**。

要取代“打开即运行”设置，请在从文件系统打开该文件之前先按 Ctrl 键 (Windows) 或 Command 键 (macOS)。在 Windows 上，您还可以在“JMP 主窗口”中右击该查询并选择**编辑查询**。

将已保存的查询用作模板

您还可以将 .jmpquery 文件用作新查询的起始点。该选项可防止您在意外点击**保存**时覆盖掉原始查询。它的功能与您在修改查询后点击**另存为**一样。

1. 在“JMP 主窗口”中的“最近使用的文件”列表中，右击 .jmpquery 文件。
2. 选择**作为模板打开**。
3. 修改查询，然后点击**保存**以创建新的 .jmpquery 文件。

注意：.jmpquery 文件包含数据库登录信息。您设置数据库连接之后才可以运行查询。请参见第 147 页的“连接至数据库”。

打开选定的数据

在指定要从 SQL 数据库导入的列和数据后，请点击**运行查询**以在数据表中打开数据。SQL 语句另存为表变量。以下脚本可用：

源 运行查询。

修改查询 在“查询生成器”中打开查询，您可在此处更改导入的列和数据并进一步定制查询。

从数据库更新 连接至数据库刷新数据，然后运行查询。若数据表和数据库表包含的列数相同，则当数据库刷新时，现有数据表中的值将会更新。

当刷新的最后一列后面只跟有公式列时，现有数据表也会更新。

否则，已更新的数据会出现在新的数据表中。

“查询生成器”红色小三角选项

“查询生成器”红色小三角菜单提供了脚本和自定义 SQL 选项。修改和运行脚本始终自动保存在最终数据表中。

复制修改脚本 将脚本复制到计算机的剪贴板中，让您可以修改查询。

复制运行脚本 将脚本复制到计算机的剪贴板中，让您可以刷新数据并运行查询。

将修改脚本保存至脚本窗口 将“修改查询”脚本保存至脚本窗口。

将运行脚本保存至脚本窗口 将“从数据库更新”脚本保存至脚本窗口。

打开即运行 打开查询文件时即运行该查询。

转换为自定义 SQL 在新的脚本编辑器窗口中显示查询语句。您在选择该选项之前必须删除提示过滤器。

当您从 “自定义 SQL”窗口中保存查询时，系统将保存自定义 SQL。在您定制查询之前就已存在的交互式组件不会被保存。红色小三角菜单中的 “恢复为交互式”也不可用。

令该查询保持与 JMP 12 兼容 令针对该查询的后续更改保持与 JMP 12 兼容。引起与 JMP 12 的兼容性问题的功能在您编辑该查询时隐藏。取消选择该选项并做出只有 JMP 13 和更高版本才支持的更改后，该选项不再可用。

“JMP 查询生成器”首选项包括相同的选项，默认取消选定该选项。在 JMP 16 中打开标记为 JMP 12 兼容的查询时，无论该首选项如何设置都会隐藏引起兼容性问题的功能。

恢复为交互式 在 “查询生成器”窗口中显示交互式查询。恢复时，系统不会保存您在 “自定义 SQL”选项卡上所做的更改。

在“查询生成器”中编写 SQL 语句

通过“查询生成器”，您可以按交互方式创建 SQL 查询，而无需编写 SQL 语句。您还可以在“查询生成器”中生成查询，然后向该查询添加定制语句。

注意：“查询生成器”示例基于的数据库并未随 JMP 安装。

1. 选择**文件 > 新建 > 数据库查询**，连接至数据库，然后选择 SQBTest 架构。请参见第 147 页的“连接至数据库”。
2. 在 “选择要查询的表”窗口中，选择 g6_Rentals 作为主表。
3. 选择 g6_Movies 和 g6_Customers 作为副表。
4. 点击**生成查询**以显示 “查询生成器”窗口。
5. 在 “已包括的列”选项卡上，点击**全部添加**按钮。
6. 点击 “查询生成器”红色小三角并选择**转换为自定义 SQL**，然后点击**确定**。

   “查询生成器”生成的 SQL 在 “自定义 SQL”选项卡中显示。
7. 在分号之前点击鼠标，然后键入以下 SQL 语句：

```
WHERE ( ( ( t2.Gender IN ( 'F' ) ) AND ( (t2.Age >= 20) AND (t2.Age
  <= 50) ) ) )
```

8. 点击**运行查询**以将数据导入 JMP 中。

   数据表脚本包括定制查询。

注意：若您从红色小三角菜单中选择**恢复为交互式**，则系统将不会保存您在“自定义 SQL”选项卡中所做的更改。若您保存定制查询并重新打开它，则“恢复为交互式”不可用。

有关 SQL 语句的简介，请参见第 150 页的“结构化查询语言 (SQL)：参考”。

从 SAS 导入数据

您可以连接到 SAS 服务器，直接处理 SAS 数据集。

Java 运行时环境 (JRE) 要求

- 在 Windows 上，必须在您的计算机上安装 Java 运行时环境 (JRE) 7 或更高版本才能访问 SAS。不过，不必将 JRE 7 指定为当前版本。
- 在 macOS 上，必须安装 JRE 7 或更高版本以便集成 SAS。

注意：在 Windows 上，JRE 的位数必须与 JMP 的位数匹配。例如，若安装了 64 位 JRE，则 64 位 JMP 将与 32 位和 64 位 SAS 同时通信。

SAS 菜单选项

以下选项位于“文件”>“SAS”菜单中。

浏览数据 浏览并导入位于 SAS 服务器上的数据。

SAS 查询生成器 在 SAS 服务器上选择并导入数据，无需编写 SQL 语句。请参见第 104 页的“在 “查询生成器”中生成 SQL 查询”。

将数据导出至 SAS 将 JMP 数据表导出至 SAS 服务器。

浏览 SAS 文件夹 浏览并运行 SAS 存储过程或打开元数据定义的数据表。

新建 SAS 程序 打开一个用于编写和提交 SAS 代码的脚本窗口。

提交至 SAS 将 SAS 代码从 JMP 直接发送至当前活动的 SAS 服务器。

打开 SAS 日志窗口 打开活动 SAS 服务器的 SAS 日志窗口。

打开 SAS 输出窗口 打开活动 SAS 服务器的 SAS 输出窗口。该窗口显示最近的 SAS 输出。

服务器连接 管理与 SAS 服务器的连接。

您还可以查找“JMP 起始页”的 SAS 页上 SAS 选项的快捷方式，同时提供了一个 SAS 工具栏。您可以在“首选项”窗口（**文件 > 首选项**）的“SAS 集成”页上保存与 SAS 集成相关的设置。有关设置“SAS 集成”首选项的详细信息，请参见“JMP 首选项”一章中第 619 页的“ “SAS 集成”首选项”。

导入 SAS 数据集

可采用以下众多 SAS 格式之一来保存 SAS 数据集：

- Windows 支持的格式包括 .sas7bdat 和 .sas7bxat。
- macOS 支持读写 .sas7bdat 文件。

- Windows 和 macOS 支持读写 .xpt 文件

在 JMP 中打开某个数据集后，该文件将作为数据表打开。JMP 默认将 SAS 变量名用作列名。要在 Windows 上使用特定文件中的变量标签，请在打开该文件时选择相应选项（请参见下面的第 5 步）。

.sas7bdat 文件支持以下 ISO 日期格式：B8601DA、B8601DN、B8601DT、B8601DZ、B8601LZ、B8601TM、B8601TZ、E8601DA、E8601DN、E8601DT、E8601DZ、E8601LZ、E8601TM、E8601TZ 和 IS8601DA。若不支持该日期格式，则将以“科学记数法”格式导入数据。

.xpt 文件支持以下日期格式：BEST、DATE、DATETIME、DAY、MMDDYY、DOWNAME、HHMM、HOUR、TIME、YYMMDD、DDMMYY、MMSS、DTDATE、IS8601DA、E8601DA、PERCENT 和 DOLLAR。

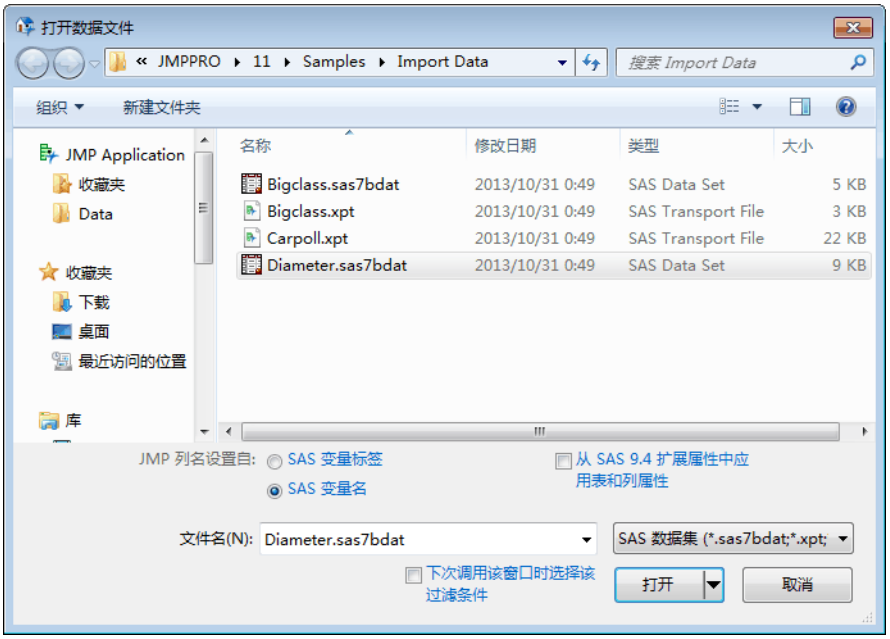
从 JMP 打开 SAS 数据集：

1. 选择**文件 > 打开**。
2. （仅限 Windows）从**文件名**旁边的列表中选择 **SAS 数据集**，如图 3.50 所示。

注意：对 SAS 数据集进行更改后， SAS 变量名和格式会被保留并保存起来。请参见“保存和共享您的数据”一章中第 500 页的 “将 JMP 数据另存为 SAS 数据集”。

3. 选择文件。

图 3.50 打开 SAS 数据集



注意：要选择要导入的列，请点击 “打开”按钮上的箭头并选择**选择列**。

4. （可选）选择以下任意选项：

   SAS 变量标签 将 SAS 变量标签（而不是变量名）用作 JMP 数据表的列名。

   SAS 变量名 将 SAS 变量名（而不是标签）用作 JMP 数据表的列名。

5. （Windows 上的可选项）选择以下任意选项：

   从 SAS 9.4 扩展属性中应用表和列属性 若 SAS 服务器支持扩展属性 (SAS 9.4)，则在存储 JMP 元数据时会包括扩展属性。该设置覆盖“SAS 集成”页上的“SAS 9.4 扩展属性”首选项。

   下次调用该窗口时选择该过滤条件 将默认文件类型选项设置为您在**文件名**列表旁边选择的选项。若选定该选项，则下次调用该窗口时默认文件类型将为 **SAS 数据集**。

6. （可选）为 SAS 传输 (.xpt) 文件选择以下任意选项：

   选择成员 允许您输入要让 JMP 打开的特定成员或表的名称。在 macOS 上，选择**成员表 > 指定**，然后输入名称。

   打开全部成员 打开传输文件中的全部成员或表。在 macOS 上，选择**成员表 > 全部**。

   保存全部成员 打开文件即将其另存为 JMP 文件。该文件保存至打开 SAS 传输文件时所用的目录。在 macOS 上，相应的选项为**全部保存**。

   选择列 指示 JMP 仅打开传输文件中的特定列。从出现的列表中选择要导入的列。在 macOS 上，相应的选项为**打开前选择列**。

7. 点击**打开**。

注意：若要导入 SAS 文件中的日期变量，JMP 会查找 SAS 日期格式并将其转换为 JMP 日期列。

通过 JMP 在 SAS 中创建 SAS 传输文件

JMP 可以打开使用 SAS XPORT 引擎保存的 SAS 传输文件。例如，下面就是一个用于创建名为 test 的传输文件的 SAS 代码示例。

注意：misc 和 work 为 SAS 逻辑库引用名。

```
data test;
input name $ age weight;
cards;
Susan 12 72
Melanie 10 68
Jonathan 11 77
Sheila 13 67
;
libname misc xport 'C:/test.xpt';
```

```
proc copy in=work out=misc;
run;
```

从 JMP 连接至 SAS

您可以连接至“SAS 元数据服务器”或直接连接至“SAS 工作区服务器”。一旦连接至“SAS 元数据服务器”，即可浏览 SAS 服务器、逻辑库和数据集。

注意：SAS 服务器版本 9.4 是 JMP“SAS 集成”首选项中的默认设置，尽管您可以按需将版本改为 9.3。

要开始连接，请选择**文件 > SAS > 服务器连接**。随即显示图 3.51 中所示的“SAS 服务器连接”窗口。可在该窗口中建立所有连接。

图 3.51 SAS 服务器连接

以下小节说明了如何连接至 SAS 服务器。

- 第 128 页的“从 JMP 连接至 SAS 元数据服务器”
- 第 130 页的“从 JMP 连接至远程 SAS 工作区服务器”
- 第 131 页的“从 JMP 连接至 SAS 环境（仅限 Windows）”
- 第 132 页的“从 JMP 连接至您本地机器上的 SAS（仅限 Windows）”

从 JMP 连接至 SAS 元数据服务器

注意：一次只能连接至一个元数据服务器。若要建立另一个连接，将断开第一个连接。

连接至元数据服务器

1. 选择**文件** > **SAS** > **服务器连接**。随即显示第 127 页上的图 3.51 中所示的“SAS 服务器连接”窗口。
2. 选择 SAS 服务器的版本。您的 “SAS 元数据服务器”管理员应了解该信息。

   JMP “SAS 集成”首选项中默认选择 SAS 服务器版本 9.4。
3. 选择要使用的配置文件。

   若不具备配置文件，请参见第 129 页的 “创建或修改 SAS 元数据服务器配置文件”。
4. 点击**连接**。

   若 JMP 无法建立连接，将出现错误消息。常见的原因是用户名或密码无效。若需要更新配置文件的信息，请参见第 129 页的 “创建或修改 SAS 元数据服务器配置文件”。
5. 点击**关闭**。

一旦连接至“SAS 元数据服务器”，即可连接至该元数据服务器所提供的任何“SAS 工作区服务器”。

连接至 “SAS 工作区服务器”（仅限 Windows）

1. 选择**文件** > **SAS** > **服务器连接**。随即显示第 127 页上的图 3.51 中所示的“SAS 服务器连接”窗口。
2. 选择要连接的工作区服务器。

图 3.52 打开与工作区服务器的连接

3. 点击**连接**。

   在 “打开工作区服务器连接”下，该工作区服务器显示为当前活动的连接。

图 3.53 当前活动连接

4. 点击**关闭**。

更改活动连接

注意：活动连接是用于提交 SAS 代码或处理 SAS 脚本命令的连接。

要更改活动连接，首先需要连接至不止一个服务器。按照第 128 页的“连接至“SAS 工作区服务器”（仅限 Windows）”中的说明操作，添加两个或更多服务器连接。

1. 在“打开工作区服务器连接”部分中，点击下拉菜单并选择所需服务器。
2. 点击**设置为活动**。
3. 点击**关闭**。

提示：您可以随时更改活动服务器。

断开与 SAS 工作区服务器连接

1. 在“SAS 服务器连接”窗口中的“打开工作区服务器连接”之下，选择要断开连接的工作区服务器。
2. 点击**断开连接**。

断开与 SAS 元数据服务器连接

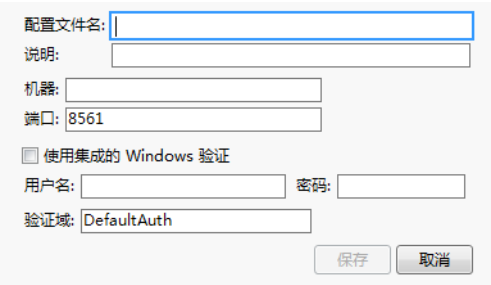
1. 在“SAS 服务器连接”窗口中，选择要断开连接的元数据服务器。
2. 点击**断开连接**。

创建或修改 SAS 元数据服务器配置文件

1. 在“SAS 服务器连接”窗口中，选择 **SAS 服务器版本**。
2. 点击**管理配置文件**。
3. 点击**添加**可添加新的配置文件，点击**修改**可更改配置文件的设置。

   随即显示“创建配置文件”或“修改配置文件”窗口。若是添加新配置文件，除“验证域”字段（其中包含 DefaultAuth）和“端口”字段之外，其他所有字段都为空。若是在修改配置文件，这些字段包含当前信息。

图 3.54 创建或修改元数据服务器配置文件



4. 填写连接 “SAS 元数据服务器”所需的信息。您的 “SAS 元数据服务器”管理员应了解该信息。

   配置文件名 选择该配置文件的名称。该名称显示在配置文件列表中。

   说明 （可选）您可以输入该配置文件的简短说明。

   机器 元数据服务器所在的机器的名称。（示例：myserver.mycompany.com）

   端口 应通过哪一端口连接该机器。（示例：8561）

   使用集成的 Windows 验证 选择该选项可使用您的 Windows 登录 ID 和密码访问服务器。启用后，用户名和密码字段将禁用。默认禁用该选项。

   用户名 用于连接该元数据服务器的用户名。

   密码 您的密码。密码始终显示为星号。

   验证域 您作为用户所属的域。

5. 点击**保存**。

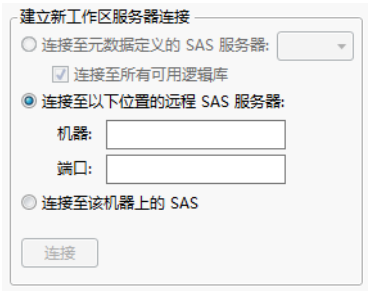
从 JMP 连接至远程 SAS 工作区服务器

您还可以直接连接至“SAS 工作区服务器”，而不必通过元数据服务器连接。

连接至远程 SAS 工作区服务器

1. 选择**文件** > **SAS** > **服务器连接**。随即显示第 127 页上的图 3.51 中所示的“SAS 服务器连接”窗口。
2. 在 “建立新工作区服务器连接”下，选择**连接至以下位置的远程 SAS 服务器**。

图 3.55 打开与远程 SAS 服务器的连接



3. 输入机器名称和端口号。您的 SAS 服务器管理员应了解该信息。
4. 点击**连接**。
5. 在出现的窗口中输入您的用户名和密码。
6. 点击**确定**。
7. 在 “SAS 服务器连接”窗口中点击**关闭**。

断开与远程 SAS 工作区服务器的连接

1. 在“SAS 服务器连接”窗口中的“打开工作区服务器连接”之下，选择要断开连接的服务器。
2. 点击**断开连接**。

从 JMP 连接至 SAS 环境（仅限 Windows）

在 Windows 上，若在 JMP 首选项中选择了 SAS 服务器版本 9.3 或 9.4 并且正确配置了您的计算机或 JMP，则您可以连接至 SAS 中间层（或 **SAS 环境**）。（SAS 服务器版本 9.4 是 JMP “SAS 集成”首选项中的默认设置。）

SAS 安装程序应该已经设置由用户计算机来查找 SAS 环境定义文件。若未设置，则可以在 JMP 首选项中输入该文件的路径。

配置 JMP 首选项

1. 选择**文件 > 首选项 > SAS 集成**。
2. 选择**连接至 SAS 环境**，然后点击**配置**。
3. 要连接至 JMP 已经检测到的环境，请点击**自动发现**，然后按需从列表中选择 URL。
4. 要输入 SAS 环境定义文件的路径，请点击**手动配置**并输入 URL。
5. 点击**确定**。

连接至 SAS 环境

1. 选择**文件 > SAS > 服务器连接**，打开“SAS 服务器连接”窗口。

2. 在 “元数据服务器连接”区域中，选择**连接至 SAS 环境**。

   若该选项不可用，则表示用户计算机或 JMP 未经配置，无法查找该环境。请参见第 131 页的 “配置 JMP 首选项”。

3. 按需从 “环境”列表中选择环境的名称。
4. 点击**连接**。
5. 出现提示时，输入您的用户名和密码。

从 JMP 连接至您本地机器上的 SAS（仅限 Windows）

您还可以直接连接至您本地机器上的 SAS。

连接至您计算机上的 SAS

1. 选择**文件 > SAS > 服务器连接**，打开 “SAS 服务器连接”窗口。
2. 在 “建立新连接”下，选择**连接至该机器上的 SAS**。

   若计算机上未安装 SAS，则该选项为禁用状态。

3. 点击**连接**。
4. 在 “SAS 服务器连接”窗口中点击**关闭**。

与用户计算机上的 SAS 断开连接

1. 在 “SAS 服务器连接”窗口中，选择 “打开的连接”下的**本地**。
2. 点击**断开连接**。

使用 “SAS 查询生成器”打开 SAS 数据集

“SAS 查询生成器”是用于从 SAS 服务器选择并导入数据的首选方法。将数据导入数据表之前，您可以对数据进行预览。您也可以保存这些查询，以便以后进行修改和运行，或在 JSL 脚本中引用。

“SAS 查询生成器”提供了通过 “文件 > SAS > 浏览数据”功能打开 SAS 数据集的替代方法。

要通过 “SAS 查询生成器”打开 SAS 数据集，请执行以下步骤：

1. 选择**文件 > SAS > SAS 查询生成器**。
2. 在 “连接至 SAS 服务器”窗口中，选择元数据服务器或远程服务器。
   - 要连接至您已在 JMP 中设置的元数据服务器，请从**连接至元数据定义的 SAS 服务器**列表中选择服务器。

     “SAS 集成”首选项中默认选择**连接至所有可用逻辑库**。该选项自动连接所有 JMP 会话中的元数据定义的 SAS 逻辑库。当工作区服务器包含大量元数据定义的 SAS 逻辑库时，可以考虑取消选择该选项，以加快您与服务器的连接。

– 要添加或配置元数据服务器，请点击**管理配置文件**，然后按照第 129 页的“创建或修改 SAS 元数据服务器配置文件”中的步骤进行操作。
– 要与远程服务器连接，请输入机器名和端口号。您的“SAS 元数据服务器”管理员应了解该信息。

3. 点击**确定**。

随即显示“SAS 查询生成器”窗口。

有关使用查询生成器的详细信息，请参见第 104 页的“在“查询生成器”中生成 SQL 查询”。

注意：

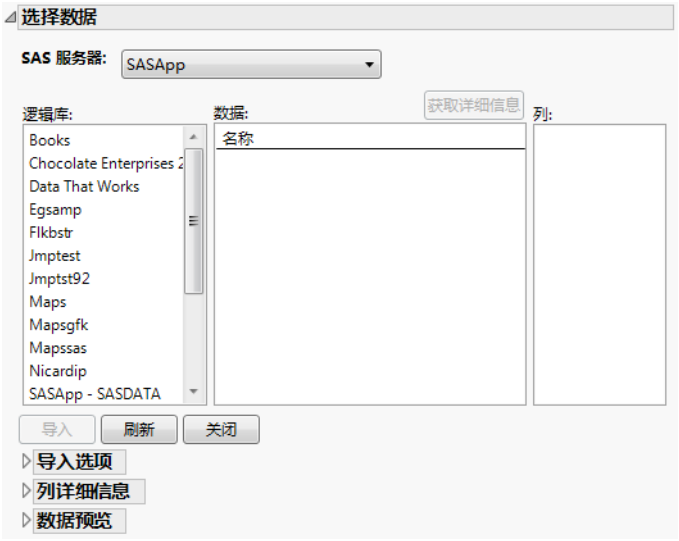
- 所有“查询生成器”的查询都在前台运行。
- 默认情况下不会导入扩展属性。要导入扩展属性，请修改 JMP“SAS 集成”首选项。选择**文件 > 首选项** (Windows) 或 **JMP > 首选项** (macOS)。选择 **SAS 集成**，然后选择**导入时，从扩展属性应用表和列属性**。
- “SAS 查询生成器”不支持本地服务器连接。

通过 SAS 服务器打开 SAS 数据集

一旦连接至某个“SAS 工作区服务器”，即可浏览该服务器上的 SAS 逻辑库并将数据导入 JMP。

要浏览 SAS 服务器上的数据集，请选择**文件 > SAS > 浏览数据**。随即显示“浏览 SAS 数据”窗口。

图 3.56 浏览 SAS 数据



该窗口最初填充有“SAS 元数据服务器”（若已连接）提供的服务器的列表。同时还显示所有物理和本地连接（如第 131 页上的图 3.55 所列）。

- 选择某个服务器可查看该服务器包含的逻辑库的列表。
- 选择某个逻辑库可查看该逻辑库中的数据集的列表。
- 选择某个数据集可查看该数据集中的列的列表。

关闭再重新打开“浏览 SAS 数据”窗口时，以前查看过的逻辑库和数据集都会显示在该窗口中。不过，您可以随时从“SAS 服务器”列表中选择其他服务器，再从中选择逻辑库和数据集。

提示：若某个服务器不可用或是连接失败，该服务器的名称将显示为浅色的斜体文本。点击该名称可尝试重新建立连接。

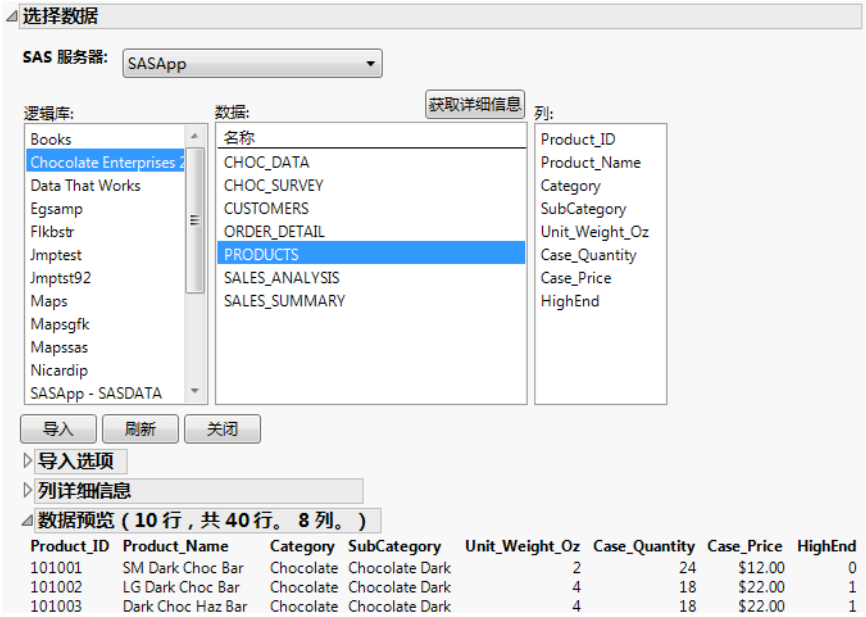
浏览 SAS 数据信息

使用“获取详细信息”、“列详细信息”和“数据预览”等选项，您可以在打开某个 SAS 数据集之前选择该数据集并查看有关其内容的信息。

数据预览

选择某个数据集后，“数据预览”分级显示项会为您显示该数据集中的前十行和前十列。

图 3.57 数据预览



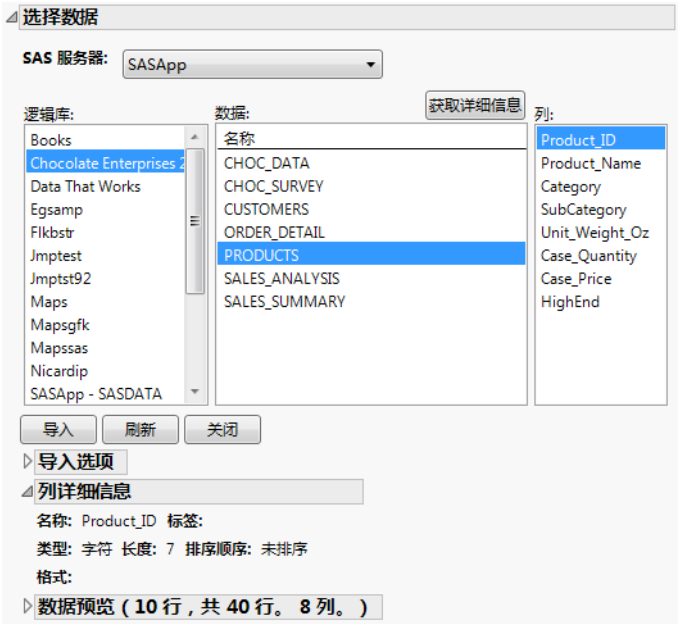
数据集详细信息

点击“浏览 SAS 数据”窗口中的**获取详细信息**可以查看逻辑库中每个数据集的大小和最后修改日期。该选项可帮助您评估您的计算机是否能处理整个数据集。

列详细信息

要查看有关数据集中的特定列的信息，可选择该列。“列详细信息”分级显示项为您显示有关该数据列的一些基本信息。

图 3.58 列详细信息



名称 SAS 数据集中的列名。

标签 描述性的列标签。该标签可以比名称长，通常有助于明确列名的含义。

类型 指定列的数据类型为字符还是数值。

长度 列中数据的字节长度。

排序顺序 列在 SAS 中的排序方式。

格式 SAS 列的格式，如 DOLLAR。该格式字段还包含有关格式化值的宽度和小数位数的信息。

打开 SAS 数据集

您可以将 SAS 数据集直接导入 JMP。

1. 从“浏览 SAS 数据”窗口中，选择一个数据集。
   默认情况下， JMP 指定要导入**所有行**。
2. 点击**导入**。

整个 SAS 数据集将导入 JMP 数据表。导入 SAS 数据后，JMP 会与 SAS 格式作最好的匹配。

导入 SAS 数据集样本

您可以将 SAS 数据集样本直接导入 JMP。

1. 从“浏览 SAS 数据”窗口中，选择一个数据集。
2. 打开 “导入选项”分级显示项。

图 3.59 导入选项

3. 若只想导入数据集的某个部分，您可以执行以下任意操作：
   - 仅选择前 *x* 行。请参见第 137 页的“仅导入前 x 行”。
   - 选择对指定的文件大小自动抽样。请参见第 137 页的“导入指定大小的自动抽样文件”。
   - 选择列的子集。请参见第 137 页的“选择列子集”。
   - 构造一个 WHERE 子句来过滤数据。请参见第 137 页的“使用 WHERE 子句导入”。
   - 抽取数据的定制样本。请参见第 138 页的“导入数据的随机样本”。

以下选项也可用：

使用标签作为导入列的列名 将 SAS 标签转换为列名。

将 SQL 表变量添加至导入的表中 将 SQL 表变量另存为数据表变量。

从 SAS 扩展属性恢复 JMP 元数据 将 SAS 数据集中保存的 JMP 元数据 （表和列属性）作为扩展属性包括在内。

仅导入前 x 行

1. 在“导入选项”部分中，选择**仅前 x 行**，并指定要导入的行数。
2. 在 “浏览 SAS 数据”窗口中，点击**导入**。

   JMP 导入指定数目的行。

导入指定大小的自动抽样文件

1. 在“导入选项”部分中，选择**自动抽样**并指定要导入的数据量 (MB)。
2. 在 “浏览 SAS 数据”窗口中，点击**导入**。

   JMP 导入指定的数据量 (MB)。

选择列子集

1. 在“导入选项”部分中，点击**选择列**。

   随即显示 “选择列”窗口。

图 3.60 选择列

2. 选择要导入的列。

   要一次选择多列，请在按住 Ctrl 的同时点击各列。
3. 点击**添加**。
4. 添加了所需的所有列后，请点击**确定**。
5. 在 “浏览 SAS 数据”窗口中，点击**导入**。

只有您从 SAS 数据集中选定的列才会导入 JMP 数据表。

使用 WHERE 子句导入

1. 点击 Where。
2. 使用 WHERE 子句编辑器构造 WHERE 子句。
3. 点击**确定**返回 “浏览 SAS 数据”窗口。
4. 点击**导入**。

只有与 WHERE 子句匹配的数据才会导入 JMP 数据表。

有关构造 WHERE 子句和使用 WHERE 子句编辑器的信息，请参见第 154 页的“使用 WHERE 子句编辑器”。

注意：若同时使用 WHERE 子句和抽样方法导入数据，则首先应用 WHERE 子句，然后再对过滤的数据进行抽样。

您还可以自行编写 SQL 语句。

使用自定义 SQL 语句导入

您还可以使用自定义 SQL 语句打开 SAS 数据集。

1. 打开“导入选项”分级显示项下的“自定义 SQL”分级显示项。

图 3.61 自定义 SQL

2. 在该窗口中输入您的 SQL 语句。
3. 点击**执行自定义 SQL**。

注意：自定义的 SQL 在所选服务器上运行，但不限于任何选定的逻辑库或数据集。

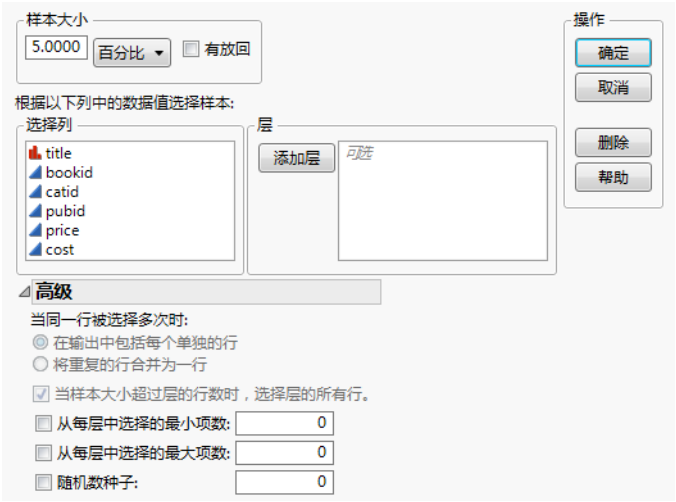
导入数据的随机样本

您还可以导入 SAS 数据集各行的随机样本。

注意：抽样功能要求在 SAS 服务器上安装具有许可证的 SAS/STAT 产品。若未安装 SAS/STAT，抽样将被禁用。

在“导入选项”分级显示项的“对导入数据进行抽样”区域中，请选中**自定义随机样本**复选框。默认情况下导入 5% 的行。要更改随机样本导入设置，请点击**设置**按钮。

图 3.62 抽样设置



在该窗口中，可以指定以下任意设置：

样本大小 您可以按百分比或行数来设置样本大小。要确保每行仅抽样一次，请取消选择**有放回**选项。要确保对任意行都可以抽样并且任意行都可以在导入的数据中多次出现，请选择“有放回”选项。

按列选择 您可以通过将列移入层列表来选择层。

处理多行抽样 若选定**有放回**，则可以指定是将每个重复的行添加为单独一行，还是将所有重复行合并为一行。若选定第二个选项，将向表中添加一列，其中包含每行被抽样的次数。

设置最小和最大选定项数 选择该选项并输入一个数字。

设置随机数种子 选择该选项并输入种子。指定种子后，您可以多次重新生成同一样本。

注意：若同时使用 WHERE 子句和抽样方法导入数据，则首先应用 WHERE 子句，然后再对过滤的数据进行抽样。

导入选项

您还可以使用另一些选项指定如何将 SAS 数据导入 JMP。

使用标签作为导入列的列名 选定该选项后，即可使用列标签来替换列名（列名长度有限，且不易理解）。默认禁用该选项。要将 SAS 数据列的名称用作 JMP 中的列名，请取消选中该框。

将 SQL 表变量添加至导入的表中 选定该选项后，即可将 SQL 查询作为变量添加至数据表面板。默认启用该选项。若禁用该选项，导入数据表时只会添加两个变量：SAS 服务器和数据集。

从 SAS 扩展属性恢复 JMP 元数据

提示：若数据受到密码保护，则最好禁用该选项，因为密码可能显示在 SQL 中。

表变量

导入 JMP 数据表后，表变量显示在数据表的左上部面板中。这些变量显示 SAS 服务器、数据集以及 SQL 查询和抽样设置（若适用）。还会添加一个源脚本，支持您随时重新导入。

打开受密码保护的数据集

JMP 可以打开受密码保护的 SAS 版本 7 或更高版本的数据集。密码不区分大小写。

打开受密码保护的数据集

1. 选择**文件 > 打开**。
2. 从**文件类型**列表中选择 **SAS 数据集**。
3. 选择文件。
4. 点击**打开**。
5. 输入密码，然后点击**确定**。

   若密码不正确，系统会提示您重新输入，直到输入正确。

使用 SAS 扩展属性导入元数据

SAS 扩展属性是您在 SAS 代码中定义的元数据，用于导入表脚本、标签、长度和类型之类的信息。您将扩展属性与某个数据集或变量关联，并以名称-值对的形式定义它们，如 _JMP_TABLESCRIPTNAME_2="OnOpen"。请参见关于扩展属性的 SAS 文档，网址为 http://sas.com/。

从 JMP 导出至 SAS 时，列属性和表脚本之类的项将自动保留。仅当将 SAS 代码导入 JMP 时，才将 SAS 扩展属性映射到 JMP 特性。

以下是在 SAS 代码中定义扩展属性的示例。这些属性定义表脚本名称和表脚本本身。

```
/* specify two table scripts. */
%LET _DS_ATTRIBUTES=_JMP_TABLESCRIPTCOUNT=2
/* define the table script names */
_JMP_TABLESCRIPTNAME_1="Favorite Movie By State"
_JMP_TABLESCRIPTNAME_2="OnOpen"
_JMP_TABLESCRIPTVALUE_2="CurrentDataTable() << RunScript(""Favorite Movie By
  State"");"
/* define the OnOpen table script */
_JMP_TABLESCRIPTVALUE_1=
/* a portion of the OnOpen table script */
"
```

```
Current Data Table();
New Column(""max_name"",
   Character,
   Width(128)
);
```

下例显示如何使用扩展属性组合 PROC DATASETS：

```
%LET _DS_ATTRIBUTES = attr-name=attr-value...attr-name=attr-value;
%LET _VAR_ATTRIBUTES= var-name (attr-name=attr-value...)...var-name
   (attr-name=attr-value...);

PROC DATASETS NOLIST LIB=WORK;
  MODIFY AttributeReferenceTable;
  XATTR OPTIONS MAXCHUNK=100;
  XATTR SET DS &_DS_ATTRIBUTES;
  XATTR SET VAR &_VAR_ATTRIBUTES;
RUN;
QUIT;
```

有关 SAS 扩展属性及其相应的 JMP 特性的详细信息，请参见第 682 页的“将 JMP 特性映射为 SAS 扩展属性的约定”。

两个“SAS 集成”首选项确定是导入还是导出扩展属性。选择**导出时，在扩展属性中存储表和列属性**以导出扩展属性。选择**导入时，从扩展属性应用表和列属性**。默认取消选定这些选项。

要查看关于导出扩展属性的示例，请运行以下脚本：

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
sd = dt << Make SAS Data Step Window( SaveJMPMetadata( 1 ) );
```

运行存储过程

存储过程是保存在所连接的 SAS 服务器上的 SAS DATA 步代码。您可以从 JMP 运行这些存储过程并查看脚本运行的结果。

注意：根据您为 SAS 设置的首选项，错误消息会发送至 JMP 日志或单独的 SAS 日志窗口。

您必须连接至元数据服务器才能查看和运行存储过程。若在未建立上述连接的情况下选择**文件 > SAS > 浏览 SAS 文件夹**，系统会提示您建立连接或取消操作。

选择并运行存储过程

1. 选择**文件 > SAS > 浏览 SAS 文件夹**。
   随即显示“浏览 SAS 文件夹”窗口。
2. 浏览存储过程，找到要运行的存储过程。
3. 选中它，并点击**运行**。

数据将作为 JMP 数据表打开。

在 Windows 上，您还可以右击存储过程并选择**复制元数据路径**。该选项将路径复制到剪贴板。您随后可以将其粘贴至脚本窗口，将其作为 JSL 运算符 `Meta Get Stored Process()` 的参数。请参见《JSL Syntax Reference》。

注意：选定流式输出时，SAS 存储过程返回的结果中可能不显示静态图形。

存储过程默认以 HTML 格式发送报表，但您可以在 JMP 首选项的“SAS 集成”页上选择 RTF 或 PDF。选择**文件 > 首选项** (Windows) 或 **JMP > 首选项** (macOS) 可查看 JMP 首选项。

提交 SAS 代码

您可以将 SAS 代码从 JMP 直接提交至当前活动的 SAS 服务器。若提交的 SAS 代码生成 SAS 列表输出，将从 SAS 服务器自动检索该输出并显示在 JMP 中。此外，还会检索生成的 SAS 日志，若提交代码中有任何错误消息，则将在“SAS 日志”窗口中自动显示 SAS 日志。

JMP 中的所有分析都在 JMP 内本机运行，而不依赖于 SAS 系统。JMP 生成的 SAS 代码是使您能够在初始 JMP 分析后在 SAS 系统中进行单独但相似的分析，或使用 JMP 内部拟合的模型对 SAS 系统中新的观测进行评分。

图 3.63 SAS 代码提交示例

SAS 代码

```
data a;
x=1;

proc print;
run;
```

用于提交 SAS 代码的 JMP 代码

```
SAS Connect("sasmain");
SAS Submit("data a; x=1; proc print; run;");
```

以下 JMP 平台可生成 SAS 代码：

- 标准最小二乘法 - PROC GLM
- REML - PROC MIXED
- 逐步 - PROC GLM
- 名义型和有序型 Logistic - PROC LOGISTIC
- GLM - PROC GENMOD
- 时间序列（ARIMA 和 TFM）- PROC ARIMA
- 神经 - SAS DATA 步评分代码
- 分割（决策树、Bootstrap 森林、提升树）- SAS DATA 步评分代码

注意：使用 JSL 函数 `As SAS Expr( formula );` 以将任意预测公式转变为可在 SAS DATA 步中使用的表达式。请参见《JSL Syntax Reference》。

直接从 JMP 运行 SAS 代码

1. 使用**文件 > 打开**打开现有 SAS 程序，或创建新的 SAS 程序。（通过选择**文件 > SAS > 新建 SAS 程序**并键入 SAS 代码，创建新的 SAS 程序。）
2. 点击**提交至 SAS** 图标 。

   您还可以在 “程序编辑器”窗口中右击，然后选择**提交至 SAS**。该菜单项还包括 SAS 代码要提交至的 SAS 活动服务器的名称。

   您还可以按 F8 （在 macOS 上按 Command+Shift+R）。

使用 JSL 脚本运行 SAS 代码

编写并运行使用 `SAS Submit` 或 `SAS Submit File` JSL 函数的 JSL 脚本。有关编写用于提交 SAS 代码的 JSL 脚本的详细信息，请参见《Scripting Guide》。

查看 SAS 列表输出

若提交的 SAS 代码生成 SAS 列表（文本）输出，作业执行完成后该输出将自动显示在“SAS 输出”窗口中。若需要以后在 JMP 会话中再次查看 SAS 列表输出，请选择**文件 > SAS > 打开 SAS 输出窗口**。“SAS 输出”窗口会保留对 SAS 活动服务器执行的最近 25 次提交的列表输出。

查看 SAS 日志

若提交的 SAS 代码有误，将自动打开 SAS 活动服务器的“SAS 日志”窗口，显示该作业的 SAS 日志。不过，您可以通过选择**文件 > SAS > 打开 SAS 日志窗口**，随时查看最近 25 次提交到活动服务器操作的 SAS 日志。

若希望在完成一次提交后，在 JMP 日志后追加 SAS 日志信息：

1. 选择**文件 > 首选项** (Windows) 或 **JMP > 首选项** (macOS)。
2. 打开 “SAS 集成”类别。
3. 在 “显示 SAS 日志”部分中，选择 **JMP 日志**而不是**单独的窗口**。

同样，在“显示 SAS 日志”部分中，您可以设置应该**始终**、**从不**还是**出错时**（默认设置）显示 SAS 日志。

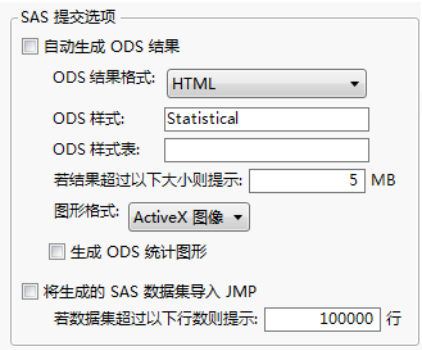
生成 ODS 结果

“SAS 输出交付系统”(ODS) 是一种功能强大的输出机制，用于以 HTML、RTF、PDF 和其他格式生成报表。ODS 输出通常比纯文本格式的 SAS 列表输出更具吸引力，也更容易定制。您可以使用“首选项”对提交的 SAS 代码进行设置，以便生成 ODS 结果而不是 SAS 列表输出。

从提交的 SAS 代码生成 ODS 结果

1. 选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)。
2. 打开 “SAS 集成”类别，找到 “SAS 提交选项”这个大选项组。

图 3.64 “首选项”中的 “SAS 提交选项”



3. 选择**自动生成 ODS 结果**选项。
4. 从 **ODS 结果格式**列表中，选择要采用哪种格式生成 ODS 结果：HTML、PDF、RTF 或 JMP 报表。
5. （可选）您可以使用其他选项指定一种样式或样式表，对结果设置格式或为生成的图形设置格式。请参见 “JMP 首选项”一章中第 619 页的 “SAS 集成”首选项”。

执行以上步骤会使 JMP 生成更多的 SAS 代码，包括用来封装您提交的 SAS 代码的 ODS 语句。您提交的 SAS 代码随后自动以指定格式生成 ODS 结果。这些结果将下载到您的计算机，若有可能，将显示在 JMP 中，或是显示在适当的外部应用程序中。

导入生成的 SAS 数据集

您提交的 SAS 代码可能生成 SAS 数据集。您可以将这些数据集自动导入 JMP，供将来分析。

1. 选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)。
2. 打开 “SAS 集成”类别。
3. 选择**将生成的 SAS 数据集导入 JMP** 选项。

将 JMP 数据表导出至 SAS

您可以将 JMP 数据表导出至“SAS 工作区服务器”。

1. 连接至“SAS 工作区服务器”。
2. 选择要导出的文件。
3. 选择**文件** > **SAS** > **将数据导出至 SAS**。

需要时可使用配置文件中的用户名和密码为您执行自动连接。

4. 从 “选择要导出的数据”下面的已打开数据表的列表中选择要导出至 SAS 的数据表。

图 3.65 将数据导出至 SAS

5. （可选）要只导出数据表中的某些列，请点击**选择列**。请参见第 145 页的“选择要导出的列”。
6. 选择目标服务器。
7. 选择逻辑库。

提示：若未显示所需逻辑库，请参见第 146 页的“在“将数据导出至 SAS”窗口中显示逻辑库”。

随即显示逻辑库中的数据集的列表。

8. 输入您想要在 SAS 逻辑库中显示的名称。
9. （可选）设置您要使用的导出选项。请参见第 146 页的 “导出选项”。
10. 点击**导出**。

选择要导出的列

1. 要只导出数据表中的某些列，请点击**选择列**。
2. 在出现的窗口中，选择要导出的列并点击**添加**。
3. 将所有列添加至 “选定列”列表后，点击**确定**。

导出选项

忽略 “已排除”行状态 （导出所有行） 选择该选项可导出数据表中的所有行。取消选择该选项仅导出未排除的那些行。默认启用该选项。

保留 SAS 变量名 该选项适用于最初从 SAS 导入的数据表。导入 SAS 数据集时，原始 SAS 变量名保存在每列的列属性中。选择该选项可在导出至 SAS 时使用每列的 SAS 变量名。取消选择该选项将导出 JMP 变量名。默认关闭该选项。

保留 SAS 格式 该选项适用于最初从 SAS 导入的数据表。导入 SAS 数据集时，原始 SAS 输出格式和输入格式保存在每列的列属性中。选择该选项可在导出至 SAS 时使用每列的 SAS 输出格式和输入格式。取消选择该选项将改为导出 JMP 格式。默认启用该选项。

在“将数据导出至 SAS”窗口中显示逻辑库

若“将数据导出至 SAS”窗口未显示所需逻辑库，可采用以下方式之一定义逻辑库：

- 使用 JSL，将代码提交至 SAS 服务器。代码将使用 `libname` 命令定义逻辑库。
- 定义 autoexec.sas 文件，在每次调用 SAS 时运行 SAS 代码片段。这将在每次连接至 SAS 时都创建相同的 `librefs`。有关 autoexec.sas 文件的详细信息，请参见 SAS 文档。

无法从“将数据导出至 SAS”窗口访问在元数据中定义的逻辑库（如在 SAS Management Console 下的 Data Library Manager 中定义的逻辑库）。

导入数据库中的数据

若您具有数据库的 ODBC （开放式数据库连接）驱动程序，则可以导入该数据库中的数据，然后将数据存回该数据库。

您可以使用结构化查询语言 (SQL) 语句控制从外部数据库中导入的数据。在 JMP 中打开某个数据库文件时，实际是向该数据库发送了一个 SQL 语句。默认情况下，该语句获取数据库表中的所有列和记录。有关交互式生成查询的详细信息，请参见第 104 页的“在“查询生成器”中生成 SQL 查询”。有关自行编写查询的详细信息，请参见第 149 页的“编写 SQL 语句以查询数据库”。

本节介绍了如何连接至数据库和导入数据。

注意：

- 不支持数据库表名称包含字符 `$# -+/%()&|;?`。
- 不允许与某个数据源建立多个连接。
- 使用 64 位版本的 ODBC 驱动程序以匹配 JMP 的位。

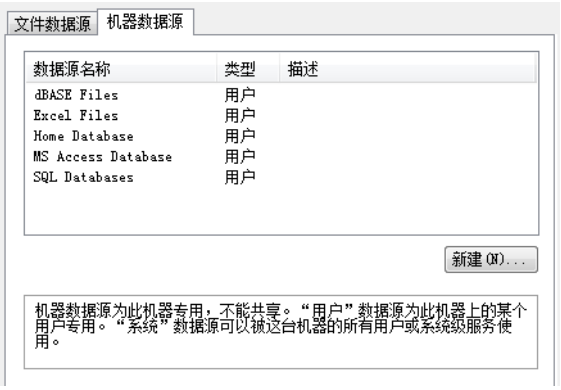
连接至数据库

JMP 可以使用 ODBC 管理器与使用 ODBC 数据源的数据库进行通信。ODBC 管理器和相关驱动程序取决于您的操作系统。您在操作系统软件中创建数据源后，请按照以下步骤在 JMP 中连接至数据库：

1. 选择**文件 > 数据库 > 打开表**。“连接”框列出了您在当前 JMP 会话中已连接的数据源。
2. 点击**新建连接**。
3. (Windows) 在“选择数据源”窗口（图 3.66）中，点击“机器数据源”选项卡，选择数据源，点击**确定**，输入用户名和密码，然后点击**确定**。

   (macOS) 在“选择 DSN”窗口中，选择数据源，输入用户名和密码，然后点击**选择 DSN**。

图 3.66 选择数据源 (Windows)



新连接显示在“从数据库打开表”窗口中。

从数据库中打开数据

在您连接至 ODBC 数据库并选择要导入的表后，数据会在数据表中打开。数据表中包含了几个表脚本。

- 运行“源”脚本可重新连接至数据库。
- 运行“从数据库更新”脚本可重新导入并刷新数据。若数据库表包含的列数与原始 JMP 表的列数相同，则现有 JMP 表中的值将原位更新。不过，若列数不同，则新数据表将在更新数据的存储位置打开。
- 运行“保存至数据库”脚本可将数据表保存至数据库。数据库中的现有数据会被替换。该脚本可能包含用户名和密码。可以设置称为 **ODBC 隐藏连接字符串**的首选项，用于防止包括这一可能的敏感信息。选择**文件 > 首选项 > 表** (Windows) 以查找首选项（或者在 macOS 上选择 **JMP > 首选项 > 表**）。

从数据库中导入数据

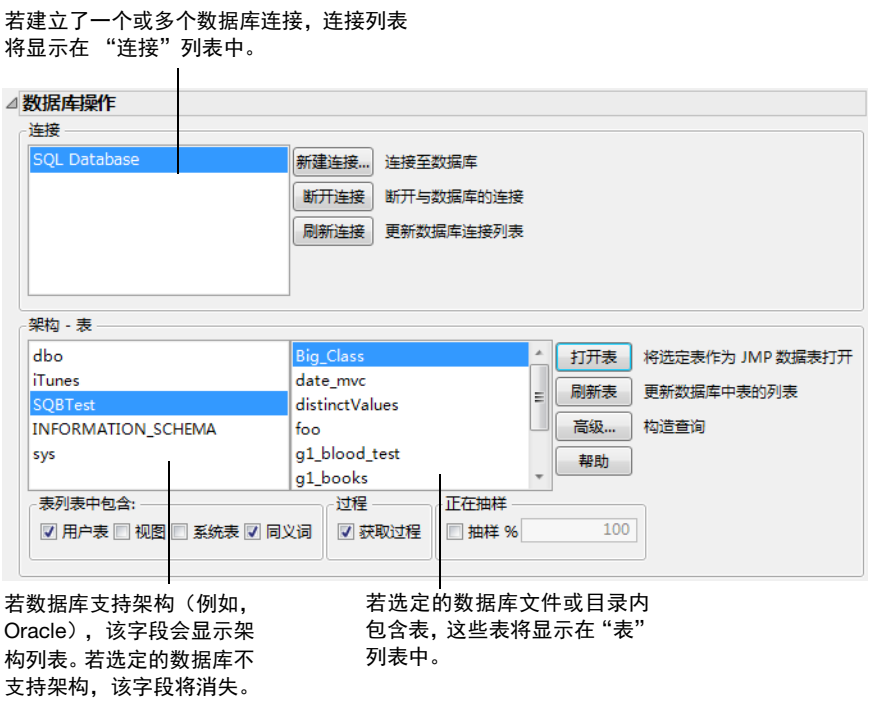
1. 选择**文件 > 数据库 > 打开表**。

   随即显示 “从数据库打开表”窗口 （图 3.67）。

2. 若您已连接至数据库，请在 “连接”框中选择该数据库。请按照第 147 页的 “连接至数据库”中的步骤进行操作。

   “连接”框列出 JMP 所连接的数据源。“架构 - 表”框列出了支持这些数据库的架构。

图 3.67 “从数据库打开表”窗口



注意：若 ODBC 驱动程序不支持获取过程，将禁用**获取过程**复选框。

3. 若所需数据源**未**列在 “连接”框中，请点击**新建连接**以选择一个数据源。选择数据源的方法取决于您的操作系统。
4. 在 “连接”框内选择所需的数据源。“表”框内的表列表随之更新。更新可能需要几秒钟的时间，具体取决于表的数量和连接至数据库的速度。若数据库支持架构，将为列表中的第一个架构加载表，点击其他架构时也会为其他架构加载表。
5. 通过选择**表列表中包含**复选框组中的选项控制所列出的表。

注意：不同驱动程序对这些标签的解释各有不同。

用户表 点击该选项后，将在**表**列表中显示所有可用的用户表。用户表特定于登录到该计算机的用户。

视图 点击该选项后，将在**表**列表中显示“视图”以及其他所有可以打开的文件类型。“视图”是一些虚拟表，即每次打开时都更新的查询结果集。视图用于获取和组合一个或多个表中的信息。

系统表 点击该选项后，将显示**表**列表中的所有可用系统表。系统表是可供所有用户或系统范围内的服务使用的表。

同义词 点击该选项后，将在**表**列表中显示所有可用的 ORACLE 同义词。

正在抽样 输入要显示在表列表中的行数百分比。选择该选项可加快大型数据库的查询速度。JMP 使用数据库所支持的抽样方法。当数据库不支持抽样时该复选框将不可用。

6. 从“表”列表中选择所需表。

注意：若连接了 dBase 数据库，请选择要连接的数据库**文件夹**。单个文件灰显，无法选择。

7. 点击**打开表**可导入选定表中的所有数据，点击**高级**可指定要导入的表子集。一些数据库要求您输入用户 ID 和密码才允许访问数据。

   当您打开大型表时可能会略有延迟。要查看所有活动 ODBC 查询的状态，请选择**视图 > 正在运行的查询**。

注意：若数据之前导出到 JMP 中的数据库并且包含“表达式”列，则该列可能会被导入为“字符”列。选择**列 > 列信息**，然后将“数据类型”改为“表达式”。

编写 SQL 语句以查询数据库

您可以使用结构化查询语言 (SQL) 语句控制从数据库中导入的数据。在 JMP 中打开某个数据库文件时，实际是向该数据库发送了一个 SQL 语句。默认情况下，该语句获取数据库表中的所有列和记录。而有时并不需要如此多的数据。若您只对一部分表数据感兴趣，则可以自定义 SQL 请求，仅请求所需的数据。执行某一 SQL 查询后，该查询的代码将以 SQL 表变量形式存储在数据表中。

本节介绍如何编写 SQL 语句以检索数据。要在不编写 SQL 语句的情况下进行交互式数据查询，可以使用“查询生成器”。您还可以在“查询生成器”中开始创建查询，然后将其添加至您自己的 SQL。请参见第 123 页的“在“查询生成器”中编写 SQL 语句”。

1. 选择**文件 > 数据库 > 打开表**。

   随即显示“从数据库打开表”窗口（图 3.67）。

2. 根据情况连接至数据库或选择现有数据库连接。请按照第 147 页的“连接至数据库”中的步骤进行操作。

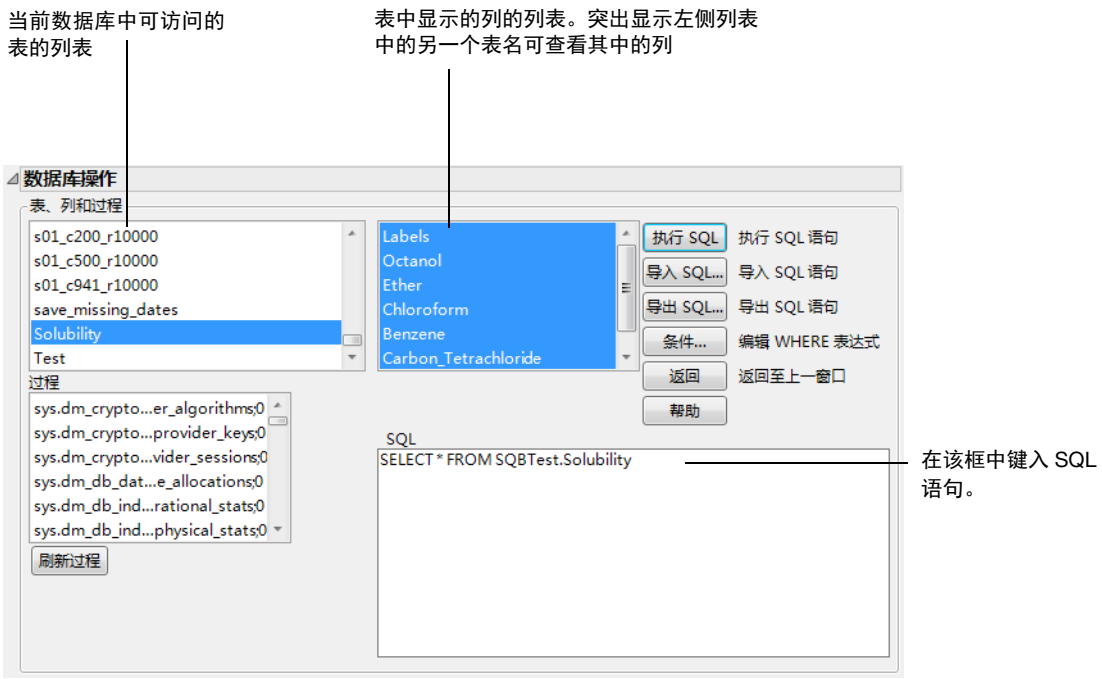
   “连接”框列出 JMP 所连接的数据源。“架构 - 表”框列出了支持这些数据库的架构。

注意：在该窗口中运行的 SQL 查询仅针对该窗口左侧各窗格中显示的表和过程执行。在此运行无关的 SQL 不会生成任何结果。

3. 在 “从数据库打开表”窗口中，点击**高级**按钮，打开特定的表子集。
4. 键入有效的 SQL 语句或修改默认语句。图 3.68 显示适用于选定文件的默认 SQL `Select` 语句。有关可以使用的 SQL 语句的说明，请参见第 150 页的“结构化查询语言 (SQL)：参考”。

   相反，您可以点击 Where 按钮并使用 WHERE 子句编辑器创建表达式，以此来添加表达式。请参见第 154 页的 “使用 WHERE 子句编辑器”。

图 3.68 读取存储在 Excel 文件的 Solubility 表中的所有变量



5. 点击**执行 SQL**。随即显示一个包含选定列的 JMP 数据表。请参见 “输入和编辑数据”一章中第 246 页的 “使用数据表变量”。
6. 要查看所有正在运行查询的状态，请选择**视图 > 正在运行的查询**。

请注意，您可以通过输入任何有效的 SQL 语句并点击**执行 SQL** 来执行该命令。有效的 SQL 因数据源和 ODBC 驱动程序而异。

结构化查询语言 (SQL)：参考

以下各部分是关于 SQL 的简介，可让您深入了解查询的强大功能，但并不是完整的 SQL 参考。

使用 SELECT 语句

JMP 中最重要的 SQL 语句是 SELECT 语句。该语句指示数据库从数据源中获取哪些行。当您完成第 149 页的“编写 SQL 语句以查询数据库”中针对 Solubility.jmp 样本数据表的过程之后，实际就是向数据源发送了以下 SQL 语句：

```
SELECT * FROM "Solubility"
```

* 运算符是“所有列”的简写形式。这样，该语句即向数据库发送了一个请求，请求从指定的数据表中返回所有列。

若不想返回所有行，您可以用数据表中的特定列名取代 *。在 Solubility 数据表示例中，您可以通过提交以下语句仅选择 ETHER、OCTANOL 和 CHLOROFORM 这三列：

```
SELECT ETHER, OCTANOL, CHLOROFORM FROM "Solubility"
```

注意：JMP 不要求使用分号结束 SQL 语句。

JMP 提供一种图形化的方式来构造简单的 SELECT 语句，这样就不必再键入实际的 SQL。要从数据源中选择特定列，请在列列表中突出显示这些列。

突出显示若干行

- 按 Shift 键，然后点击以选择一系列列名
- 按 Ctrl 键再点击 (Windows) 或按 Command 键再点击 (macOS) 可选择单个列名。

请注意，SQL 语句将随着您的选择相应更改。

有时，您希望从数据源中仅获取唯一记录。也就是说，您希望消除重复记录。要启用该功能，请使用 DISTINCT 关键字。

```
SELECT DISTINCT ETHER, OCTANOL, CHLOROFORM FROM "Solubility"
```

对结果排序

您可以按数据库中的一个或多个字段对结果排序。使用 ORDER BY 命令指定要排序的变量。

```
SELECT * FROM "Solubility" ORDER BY LABELS
```

选择所有字段，按 LABELS 变量对生成的数据表排序。若要指定更多变量作为排序依据，请将其添加到逗号分隔列表中。

```
SELECT * FROM "Solubility" ORDER BY LABELS, ETHER, OCTANOL
```

使用 WHERE 语句

使用 WHERE 语句，您可以基于条件获取数据表中的特定行。例如，您可能想要选择列 ETHER 的值大于 1 的所有行。

```
SELECT * FROM "Solubility" WHERE ETHER > 1
```

WHERE 语句放置在 FROM 语句之后，可使用以下任意逻辑运算符。

表 3.4 WHERE 运算符

运算符	含义
=	等于
!= 或 <>	不等于
>	大于
<	小于
>=	大于等于
<=	小于等于
NOT	逻辑 NOT
AND	逻辑 AND
OR	逻辑 OR

对条件求值时，先对整个语句处理 NOT 语句，然后再依次处理 AND 语句和 OR 语句。因此，

```
SELECT * FROM "Solubility" WHERE ETHER > -2 OR OCTANOL < 1 AND CHLOROFORM > 0
```

相当于

```
SELECT * FROM "Solubility" WHERE ETHER > -2 OR (OCTANOL < 1 AND CHLOROFORM >
  0)
```

使用 IN 语句和 BETWEEN 语句

要指定获取一定范围内的值，请随同 WHERE 使用 IN 和 BETWEEN 语句。IN 语句指定值列表，BETWEEN 支持您指定一定范围内的值。例如，

```
SELECT * FROM "Solubility" WHERE LABELS IN ('Methanol', 'Ethanol', 'Propanol')
```

获取 LABELS 列的值为 Methanol、Ethanol 或 Propanol 的所有行。

```
SELECT * FROM "Solubility" WHERE ETHER BETWEEN 0 AND 2
```

获取 ETHER 值介于 0 和 2 之间的所有行。

使用 LIKE 语句

使用 LIKE 语句，您可以选择与给定字符串相似的值。用 % 来表示可包含任何值的字符串。例如，您可能想要从 Solubility 数据中选择所有乙醇类的化学品，即：以 –ol 结尾的化学品。下面的 SQL 语句可完成该任务。

```
SELECT * FROM "Solubility" WHERE LABELS LIKE '%OL'
```

% 运算符可放置于 LIKE 语句中的任何位置。下面的示例提取包含以 M 开头且以 OL 结尾的标签的所有行：

```
SELECT * FROM "Solubility" WHERE LABELS LIKE 'M%OL'
```

使用聚合函数

聚合函数用于获取汇总数据而不是数据本身。可在 SELECT 语句中使用以下任意聚合函数。

表 3.5 SELECT 语句函数

函数	含义
SUM( )	列的总和
AVG( )	列的平均值
MAX( )	列的最大值
MIN( )	列的最小值
COUNT( )	列中的行数

以下是一些示例：

- 以下语句请求对 ETHER 列和 OCTANOL 列分别求和：
  ```
  SELECT SUM(ETHER), SUM(OCTANOL) FROM "Solubility"
  ```
- 下面的语句返回 ETHER 值大于 1 的行数：
  ```
  SELECT COUNT(*) FROM "Solubility" WHERE ETHER > 1
  ```
- 下面的语句可为您计算乙醇类数据的 OCTANOL 平均值：
  ```
  SELECT AVG(OCTANOL) FROM "Solubility" WHERE LABELS LIKE '%OL'
  ```

注意：使用聚合函数时，生成的 JMP 数据表中的列名为 Expr1000、Expr1001，依此类推。最好在完成获取后重命名这些列名。

GROUP BY 和 HAVING 命令

GROUP BY 和 HAVING 命令特别适用于聚合函数。这两个命令支持您基于数据集中某个字段的值多次执行聚合函数。

例如，您可能想要计算数据表中每个 ETHER 值（ETHER=0、ETHER=1 等）的记录数。

- SELECT COUNT(ETHER) FROM "Solubility" GROUP BY (ETHER) 返回一列数据，列中每个条目对应 ETHER 的一个水平。
- SELECT COUNT(ETHER) FROM "Solubility" WHERE OCTANOL > 0 GROUP BY (ETHER) 的作用与上一语句相同，只不过是为 OCTANOL > 0 的行计算记录数。

对某列的聚合函数使用 GROUP BY 时，请将该列本身包含在 SELECT 语句中。例如，

```
SELECT ETHER, COUNT(ETHER) FROM "Solubility" GROUP BY (ETHER)
```

在返回计数之外还返回包含 ETHER 水平的一列。

使用子查询

聚合函数还有助于计算要在 WHERE 语句中使用的值。例如，您可能希望获取大于 ETHER 平均值的所有值。换言之，您希望找到 ETHER 的平均值，然后仅选择值大于该平均值的记录。请记住 SELECT AVG(ETHER) FROM "Solubility" 语句可获取您想要的平均值。所以，正确的 SQL 命令会在 WHERE 条件语句中使用以下语句：

```
SELECT * FROM "Solubility" WHERE ETHER > (SELECT AVG(ETHER) FROM
  "Solubility")
```

保存和加载 SQL 查询

构造查询之后，您可能希望以后重新运行该查询，但不想在每次想要使用该查询时都手动键入它。此时，您可以将该查询导出至外部文件。请在图 3.68 所示的窗口中点击**导出 SQL** 按钮。这将调出一个窗口，您可在其中将 SQL 查询另存为文本文件。

要加载保存的查询，请在图 3.68 所示的窗口中点击**导入 SQL** 按钮。这将调出一个窗口，您可在其中导航到保存的查询。该查询一打开即加载到该窗口中。

使用 WHERE 子句编辑器

JMP 可以帮助在 ODBC 导入期间生成 SQL 查询的 WHERE 子句。它提供 WHERE 子句编辑器，可帮助您生成使用 SQL 常用功能的基本表达式，同时允许使用供应商特定的函数。例如，您不必了解 SQL 是使用“=”还是“==”来进行比较，也不必了解是使用 avg() 还是 average() 来求平均值。

此外，应使用单引号 (‘string’) 而不是双引号 (“string”) 引住字符串文字。

打开 WHERE 子句编辑器

1. 按照第 147 页的“连接至数据库”中的步骤连接至某个数据库。
2. 在**从数据库打开表**窗口中，点击**高级**按钮。
3. 点击 Where 按钮。

使用 WHERE 子句编辑器从工作面板添加以下任何内容：表达式、函数和项。这些内容应用于突出显示的蓝框。

1. 点击“表名浏览器”选择某个表。该表中的列显示在列表中。
2. 点击“SQL 供应商名称浏览器”选择要使用的 SQL 的类型：GenericSQL、Access、DB2、MySQL、Oracle、SQL Server 或以上全部类型。通过点击列表中的某个函数或运算符并从出现的列表中选择运算符，执行所需操作。

注意：不支持以下 SQL Server 数据类型：二进制、地理和几何图形。

3. 通过点击公式编辑区域中的空公式元素将其选中。出现红色外框时即被选中。相对于所点击位置的最小嵌套框内的所有项都将被选中。后续操作应用于这些组合元素。
4. 通过点击小键盘上的按钮向表达式添加运算符。
5. （可选）要定制 WHERE 子句，请从公式编辑器中的红色小三角菜单中选择一个选项：

显示外框 显示或隐藏 WHERE 子句项的外框。

大字体 增大公式的字体大小。

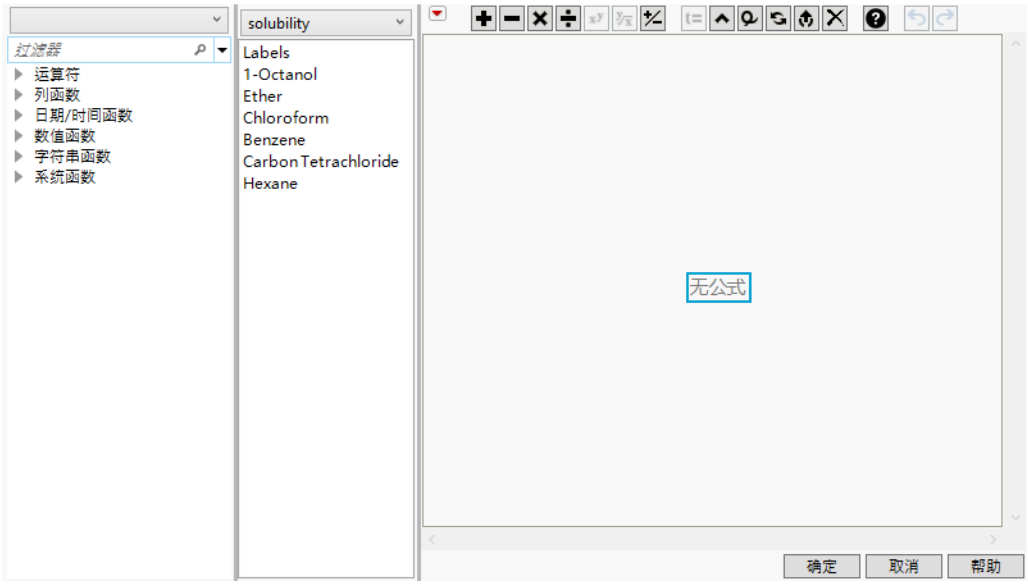
小字体 减小公式的字体大小。

简化 尽量简化 WHERE 子句语句。

重置面板布局 显示图 3.69 中所示的面板。

WHERE 子句编辑器与“公式编辑器”（在第 339 页的“在 JMP 中创建公式”一章中介绍）的工作方式类似。

图 3.69 WHERE 子句编辑器



从 Google Sheets 导入数据

Google Sheet 是一种保存在 Google 帐户中的电子表格（由类似选项卡的工作表构成）。这样您就可以继续编辑电子表格并与其他人共享。

通过登录到您的 Google 帐户、复制电子表格的 URL 并指定数据的构成方式，可以从 Google 电子表格导入数据。例如，若有列标题，您必须知道数据从哪一行开始。

要从 Google 电子表格导入数据，请执行以下步骤：

1. 登录 https://sheets.google.com 并打开该电子表格。
2. 复制浏览器地址栏中的 URL。
3. 在 JMP 中，选择**文件 > 从 Internet 打开 > Google Sheets**。
4. 键入您的 Google 帐户名称。（没必要键入 @gmail.com。）
5. 将该 URL 粘贴到电子表格 URL 或 ID 字段中。
6. 点击**导入**。

   数据将导入 JMP 数据表。

   若您是第一次使用指定的 Google 帐户导入数据，系统将提示您为 JMP 提供对您的 Google 帐户的访问权限。

点击**允许**。

数据将导入 JMP 数据表。

关于导入 Google Sheets

- Google 用来对您帐户进行身份验证的信息存储在 C:/Users/<用户名>/AppData/Roaming/SAS/JMP/OAuthTokens.jmp 中。（您的 Google 密码已经加密。）这支持您使用相同的 Google 帐户导入和导出数据，而无需再次提供身份验证信息。请注意，有权访问您的计算机用户帐户的用户也有权访问您的所有 Google 电子表格。若其中包含机密信息，请重新考虑共享您的用户帐户。
- 帐户名称中不要求包含 @gmail.com。不过，若您使用它，则 <用户名>@gmail.com 和 <用户名> 将被视为不同帐户。导入或导出时，您必须为这两个帐户输入身份验证代码。
- 若其他用户有权访问您计算机的用户帐户，他们则可以将 OAuthTokens.jmp 复制到他们的计算机并导入您的电子表格。
- 若在 Windows 上的 Internet Explorer 中选择了**在兼容性视图中显示 Intranet 站点**，则无法导入Google Sheet。要避免该问题，请执行以下步骤：
  1. 在 Internet Explorer 中，选择“工具”按钮 ，然后选择**兼容性视图设置**。
  2. 取消选择**在兼容性视图中显示 Intranet 站点**。

注意：在 Google Sheets 中，您可以选择**文件 > 发布到 Web** 发布工作表以供用户查看。在这种情况下，指向已发布工作表的链接包含 pubhtml 后缀。使用“文件”>“从 Internet 打开”打开这些工作表。请参见第 162 页的“从 Internet 或远程计算机导入数据”。

Google Sheet 设置

如有必要，更改以下设置，然后点击**导入**导入工作表数据。

工作表包含列标题 选择工作表是否包含带列标题的行。

列标题起始于行 指示工作表中列标题起始于哪一行。右击向上箭头 ，直到标题起始于正确的行为止，或者输入行号，然后按 Enter 键。

数据起始于行 指示工作表中数据起始于哪一行。

选择要打开的工作表 通过按 Ctrl 键并点击工作表，取消选择任何不想导入的工作表。

高级选项

单元格区域 导入指定区域内的单元格 （例如， G5:G17）。

导入单元格颜色 将工作表中的单元格颜色应用到数据表。

隐藏空列 阻止空列出现在数据表中。

导入 SPSS 文件

JMP 将 SPSS 文件作为数据表打开，并保持几项 SPSS 功能：

- 支持具有最基本的格式的常规数值和字符数据。
- 支持 SPSS 日期、日期时间和时间格式。
- 默认情况下，标签转换为列标题。若选择该选项，但数据不包含标签，则列将命名为**第 1 列**、**第 2 列**，依此类推。

  您还可以在打开 SPSS 文件时选择用于转换列标题的方法。所选的方法随后将取代首选项。

  要更改默认转换方法，请选择**文件 > 首选项**（或 macOS 上的 **JMP > 首选项**）。在“常规”页上，取消选择**导入期间将 SPSS 标签用于列名**。变量名随后将自动导入为列标题。
- 在 SPSS 文件中定义的值标签另存为值标签列属性。在每个数据表单元格中显示的是该值标签而不是原始值。有关“值标签”属性的详细信息，请参见“设置 JMP 列属性”一章中第 266 页的“值标签”。
- SPSS 可以对一个变量赋于特定值以便分析中被当作缺失项使用。例如，可以将值 64 视为**身高**变量的缺失值。然后，在计算身高的分布时将忽略值 64。当您导入 SPSS 时，这些值包括在相应变量的“缺失值代码”列属性中。

  支持至多包含 20 个数的缺失值范围。若范围以负数开始，则数字倒数至最大值 （例如 -10 至 -5）。若范围以正数开始，则数字从最小数开始累加 （例如 1 到 12）。

导入时将不保留 SPSS 文件中选定的自定义货币格式。此外，JMP 不能读取包含双字节字符（如，非 Unicode 日文字符）的 SPSS 数据。

打开 SPSS 文件 (Windows)

1. 选择**文件** > **打开**。
2. 从**文件名**旁边的列表中，选择 **SPSS 数据文件 (*.sav)**。
3. 选择 SPSS 文件。
4. （可选）要指定列标题，请从以下 **JMP 列名设置自**选项中选择其一：
   - **SPSS 标签**用于基于 SPSS 标签创建列标题。
   - **SPSS 变量名称**用于基于变量名创建列标题。
5. 点击**打开**。

   JMP 将该文件作为数据表打开。

打开 SPSS 文件 (macOS)

1. 选择**文件** > **打开**。
2. 选择 SPSS 文件。
3. （可选）要指定列标题，请执行以下操作之一：
   - 取消选择**使用 SPSS 标签作为标题**可将变量名转换为列标题。
   - 选择**使用 SPSS 标签作为标题**可将标签转换为列标题。
4. 点击**打开**。

   JMP 将该文件作为数据表打开。

注意：

- 与导入其他文件相同，打开和保存大型 SPSS 文件时可能会遇到延迟。
- 竖线 (|) 现在可将 .sav 文件的 SPSS 标签中的多个响应内的单词分开（如在“detergent|liquid”中）。要更改分隔符，请运行以下 JSL 脚本：`Preference(SPSSMultiResponseDelimiter(":"))`，其中的冒号是新的分隔符。

导入 3S 调查数据

3S (SSS) 是调查数据的一种规格。调查数据存储在成对文件中：.xml 或 . sss，以及 . csv、.dat 或 . asc 文件。. xml 或 . sss 文件可定义变量并描述调查。随附的 . csv、.dat 或 . asc 文件包含收集自每一个响应者的数据。

默认情况下，变量标签将被转换为列标题。要将变量名称转换为列标题，请取消选择 JMP“常规”首选项中的**使用 3S 标签作为标题**。

要导入 3S 文件，请执行以下步骤：

1. 将一对 3S 文件放置于相同的文件夹中，并且对两个文件使用相同的根名称。
2. 选择**文件** > **打开**。

3. 在 Windows 上，选择 “文件” 名称列表中的 **3S 调查文件**。
4. 选择 SPSS 文件。
5. （在 Windows 上可选）要覆盖 3S 列标题首选项，请选择 **JMP 列名设置自** : 旁边的 “变量标签” 或 “变量名称”。
6. 点击**打开**。

   JMP 将该文件作为数据表打开。

注意：默认编码为 UTF-8。编码为 Windows-1252 的 3S 文件将作为 ISO-8859-1 导入。

导入 HDF5 数据

分层数据格式，版本 5 (HDF5) 是用于存储数据的可移植文件格式。HDF5 文件包含组和数据集。导入该文件时，JMP 会打开一个组以显示内部数据集的名称。例如，若该文件包含一组含有曲目和艺术家表的唱片数据，JMP 会提示您打开 “曲目” 或 “艺术家”。若该文件不包含组，JMP 会提示您打开单个表。

JMP 仅可处理含有数值（整型、浮点、双精度）和字符串类型的表，以及具有三个或三个以下维（仅含简单类型）的复合文件。若数据集未显示在导入窗口中，则很可能不支持该文件。

您最多可以导入 1,000,000 列，行数不受限制。

注意：HDF5 文件仅支持 ASCII 字符。法文和中文之类的非 ASCII 字符不受支持。

要导入 HDF5 文件，请执行以下步骤：

1. 选择**文件 > 打开**。
2. 在 Windows 上，确保从 “文件名” 列表中选定了**所有 JMP 文件**或 **HDF5 数据文件**。
3. 选择 HDF5 文件。

   随即显示该文件中的组或数据集列表。
4. 选择您要导入的数据集，然后点击**导入**。

   数据将以数据表的形式打开。

从 MATLAB 导入数据

JMP 提供用于与 MATLAB 通信的脚本编写接口。请参见《Scripting Guide》和《JSL Syntax Reference》。

从 R 导入数据

JMP 提供用于与 R 通信的脚本编写接口。请参见《Scripting Guide》和《JSL Syntax Reference》。

使用 Excel 插件导入数据

Excel 插件为 Windows 上的 JMP 和 Excel 用户提供了新功能：

- 将 Excel 中的选定单元格转换为 JMP 数据表。请参见第 160 页的“将 Excel 数据传输至数据表”。
- 在 Excel 工作簿中将 JMP 刻画器用于计算模型。刻画器工具用于将 JMP 刻画器的强大功能引入 Excel 工作表中的模型。您不必在 JMP 中重建 Excel 模型，只需确认这些模型是正确的，并在 JMP 和 Excel 中维护这些模型。请参见《刻画器指南》。

注意：在 JMP 安装期间，选择 Excel 插件。这将为您的 Microsoft Excel 版本安装插件。

- Microsoft Excel 2010、Microsoft Excel 2013、Microsoft Excel 2016、Microsoft Excel 2019 和 Office 365 Desktop Excel 均受支持。

将 Excel 数据传输至数据表

您可以使用 Excel 的 JMP 插件将 Excel 工作表传输至以下 JMP 目标：

- 数据表
- 图形生成器
- “分布”平台
- “以 X 拟合 Y”平台
- “拟合模型”平台
- “时间序列”平台
- “控制图”平台

您还可以创建用于在 JMP 中进行刻画的模型。请参见《刻画器指南》。

将 Excel 数据传输到 JMP 数据表和平台

1. 在 Excel 工作表中，在功能区中显示 JMP 插件。
2. 点击**首选项**按钮。
3. 接受默认的**数据表名称**（文件名 _ 工作表名称）或键入一个名称。
4. 若工作表中的第一行包含列标题，请选择**使用前面几行作为列名**。

5. 若选择使用前几行作为列标题，请键入要使用的行数。
6. 若工作表包含要包括在 JMP 数据表中的隐藏行，请选择**传输隐藏行**。
7. 若工作表包含要包括在 JMP 数据表中的隐藏列，请选择**传输隐藏列**。
8. 点击**确定**保存您的首选项。
9. 选择要传输到 JMP 的单元格，包括要用作列名的所有单元格。
   若将单元格用作列名，这些单元格需要位于选定内容的前面几行。
   若仅选定一个单元格 （或未选定任何单元格），则将整个 Excel 工作表传输到 JMP。
10. 从工具栏中选择 JMP 目标：
    - 数据表
    - 图形生成器
    - “分布”平台
    - “以 X 拟合 Y”平台
    - “拟合模型”平台
    - “时间序列”平台
    - “控制图”平台

    JMP 打开，选定的数据位于新的 JMP 数据表中，选定的启动窗口随之显示。

注意：

- 空单元格将导入为缺失数据，日期、数字和字符串均可正确识别。
- JMP 窗口可能隐藏在 Excel 窗口之后，特别是在您将 Excel 最大化时。

有关使用各个 JMP 平台的详细信息，请参考相应的 JMP 文档。

关于 Excel 的 JMP 插件

JMP 插件命令分为两组：

传输至 JMP

首选项 设置首选项，以便将数据从 Excel 传输到 JMP。

数据表 将 Excel 文件中的选定数据传输到 JMP 数据表。

图形生成器 将 Excel 文件中的选定数据传输到 JMP 数据表，并启动 “图形生成器”平台。

分布 将 Excel 文件中的选定数据传输到 JMP 数据表，并启动 “分布”平台。

以 X 拟合 Y 将 Excel 文件中的选定数据传输到 JMP 数据表，并启动 “以 X 拟合 Y”平台。

拟合模型 将 Excel 文件中的选定数据传输到 JMP 数据表，并启动 “拟合模型”平台。

时间序列 将 Excel 文件中的选定数据传输到 JMP 数据表，并启动 “时间序列”平台。

控制图 将 Excel 文件中的选定数据传输到 JMP 数据表，并启动 “控制图” 平台。

在 JMP 中刻画

创建 / 编辑模型 设置将 JMP 刻画器用于 Excel 数据的首选项。

运行模型 运行 JMP 刻画器。

卸载 JMP Excel 插件

要卸载 Excel 插件，请双击 JMP 安装程序，点击**修改**，然后取消选择 **Excel 插件**。

从 Internet 或远程计算机导入数据

通过选择**文件 > 从 Internet 打开**，可以从 Internet、FTP 站点或其他计算机导入数据。文件路径的开头可以是 http、ftp 或 file；也可以是盘符或指向网络驱动器的路径（相对或绝对）。

使用 “从 Internet 打开”，您可以从 Internet 或远程计算机导入数据和图像，并将其另存为数据表、网页或文本。您还可以使用该选项打开 SAS 存储过程报表。有关使用 “从 Internet 打开” 导入 Google 电子表格的详细信息，请参见第 155 页的 “从 Google Sheets 导入数据”。

假定您想要导入网页上的表中的数据（和图像），并将其另存为数据表。该特定网页列出按人口划分的国家/地区。

1. 选择**文件 > 从 Internet 打开 > 网页**。
2. 输入以下 URL：
   https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_population。
3. 将**打开方式**设置为**数据**，然后点击**确定**。

   JMP 查找指定页上的若干表。
4. 点击 Rank country (or dependent territory)...，然后点击**确定**。

   JMP 打开包含来自网页表的信息的数据表。
5. （可选）要向数据表添加图像，请运行**加载图片**脚本。

注意：

- 您可以采用以下格式之一打开远程文件：

  数据 将该文件作为数据表打开。

  网页 在 JMP 浏览器中打开网页 (Windows) 或在默认浏览器中打开网页 (macOS)。

  从 Windows 上的 “文件” 菜单中，您随后可以选择保存该文件或将数据导入为数据表。选择该选项可导入由网页脚本和服务器端请求生成的数据。

  文本 在 “脚本编辑器” 中打开该文件。若导入的是 HTML 文件，则显示网页的 HTML 标记。

- 从 FTP 服务器打开某个文件时，将显示一个 FTP 登录窗口。对于匿名帐户，点击**确定**。对于需要身份验证的登录，请输入您的用户 ID 和密码。文件随后按您指定的方式打开。

  注意：一些匿名 FTP 服务器要求提供用户 ID。若数据表未能打开，请尝试在“用户 ID”文本框中键入“ftp”或“anonymous”。将密码文本框留空，然后点击**确定**。

- JMP 会在因错停止导入之前等待 60 秒。您可以在“常规”首选项页上更改“从 Internet 打开超时”设置。
- 若您正在导入的表包含图像，则图像首先会以文本格式导入。要在您的数据表中显示这些图像，可以运行自动生成的名为**加载图片**的表脚本。包含这些图像的新表达式列将添加至数据表。
- 运行数据表中的“源”脚本可支持您重新导入和刷新数据。

将 SAS 存储过程报表作为数据表打开 （仅限 Windows）

1. 选择**文件** > **从 Internet 打开**，然后选择该文件。

   该文件将在 JMP 浏览器中打开。
2. 在 JMP 浏览器中，选择**文件** > **导入表为数据表**。

   出现一个窗口，列出在网页中找到的表。
3. 选择要导入的一个或多个表。
4. 点击**确定**。

   每个表都作为新的数据表打开。

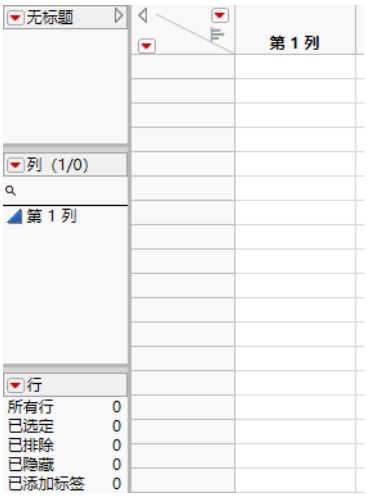
有关存储过程报表的详细信息，请参见第 141 页的“运行存储过程”。

创建新数据表

要通过手动输入数据创建新数据表，请执行以下步骤：

1. 选择**文件** > **新建** > **数据表**。这将显示一个空数据表，不包含任何行，只有一个标记为**第 1 列**的数值列。
2. 将光标移至某一单元格上。
3. 在该单元格中点击。光标在该单元格中显示为一条线。

图 3.70 新的数据表



4. 输入值。

可使用多种方法在表中填充值：

- 创建新的行列，并键入数据或将数据粘贴到数据网格中。（请参见“输入和编辑数据”一章中第 168 页的“在数据表中输入数据”。）
- 构造一个公式来计算列值。（请参见“在 JMP 中创建公式”一章中第 342 页的“生成公式”。）
- 从其他应用程序导入数据。（请参见第 64 页的“关于导入数据”。）
- 复制其他应用程序中的值，并将其粘贴到表中。
- 使用测量设备读取外部度量值。有关数据馈送的详细信息，请参见《Scripting Guide》。
- 将列从一个表拖至另一个表。

有关如何对数据表进行格式设置、编辑和处理的详细信息，请参见第 165 页的“输入和编辑数据”一章。

第 4 章

输入和编辑数据

执行数据表基本任务

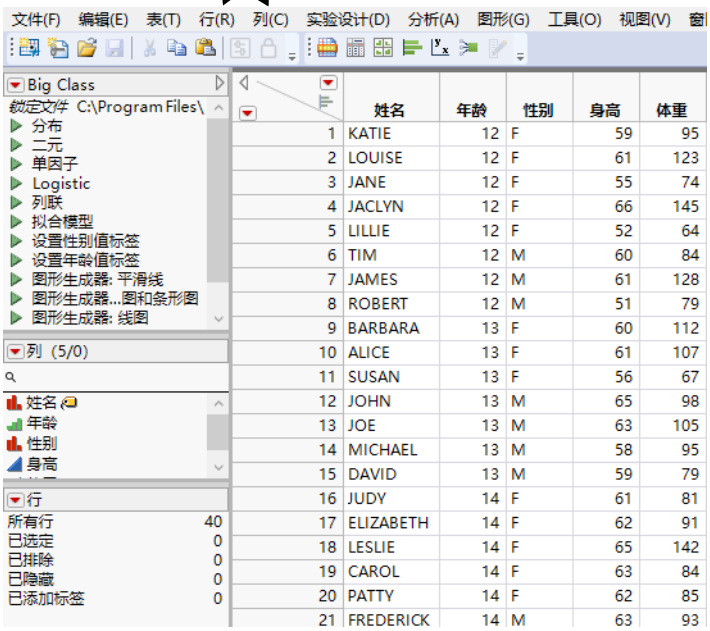
将数据导入 JMP 或创建新的数据表后，您可以设置数据的格式以便为数据分析做好准备。

本章包含以下信息：

- 更改数值的格式
- 添加、删除和选择行列
- 使用“行编辑器”在行中导航并编辑行
- 创建保存至数据表的脚本

图 4.1 “行”菜单和“列”菜单

使用“行”菜单和“列”菜单中的选项输入和编辑数据。



目录

在数据表中输入数据 168
将数据复制并粘贴到数据表中 168
向数据表添加行 168
向数据表添加列 169
使用顺序数据填充数据表列 171
使用数据表列中之前的值替换缺失值 172
将单元格公式输入数据表列 172
选择数据表中的行 173
查找行中的匹配单元格 175
选择包含特定值的行 175
定位下一选定行 177
选择数据表中的列 178
如何选择列 178
在 JMP 列查看器中选择列 179
定位数据表中的下一个选定列和上一个选定列 185
调整列和行的大小 185
在数据表中组织数据 186
删除数据表中的行 186
删除数据表中的列 186
重新排列数据表中的列 187
组合数据表中的列 187
移动数据表中的值 188
为数据表中的单元格着色 188
编辑或删除数据表中的单元格 189
编辑数据表中的列名 189
隐藏和排除数据表中的行 189
排除数据表中的行 190
隐藏数据表中的行 190
删除数据表中的列 191
隐藏数据表中的列 192
查看数据表中缺失数据的模式 192
查找和替换数据表中的单元格值 194
编辑数据表中的单元格 196
行列的右击菜单 197
比较数据表 198
为行和列指定特征 202
如何查找行列选项 203

为行和列添加标签 203
为行指定颜色或标记 204
在图形中创建颜色主题 206
删除图形中的定制颜色主题 210
选择匹配行状态单元格 211
删除行特征 211
锁定列 211
重构数据 211
将数据表列拆分为多列 212
通过文本匹配创建新列 212
生成指示符列 214
组合数据表中的列 214
压缩数据表中的选定列 215
在数据表列中生成分箱公式 217
将标签转换为代码 / 将代码转换为标签 220
生成新公式列 221
变换 JMP 平台中的列 222
对列中数据重新编码 230
编辑数据表 244
重命名数据表 244
锁定数据表 244
压缩数据表 245
复制表脚本 245
使用数据表变量 246
在数据表中创建和保存脚本 247

在数据表中输入数据

本节介绍如何添加行和列、使用顺序数据填充列，以及输入单元格公式。

- “将数据复制并粘贴到数据表中”
- “向数据表添加行”
- “向数据表添加列”
- “使用顺序数据填充数据表列”
- “使用数据表列中之前的值替换缺失值”
- “将单元格公式输入数据表列”

提示：考虑将“常规”首选项中的“自动保存超时值”设置为在指定的分钟数自动保存打开的数据表。该自动保存值还适用于记录、脚本、项目和报表。

将数据复制并粘贴到数据表中

将数据从其他文档粘贴到数据表时，JMP 按需添加行列。点击某个单元格并粘贴该数据。

多行数据

复制一个数据表单元格并粘贴多行数据时，每行都粘贴到它自己的单元格中。

要将所有行都粘贴到一个单元格中，请执行以下操作之一：

- 先双击现有单元格再粘贴。
- 右击现有单元格并选择**粘贴至单元格**。“粘贴至单元格”选项仅适用于字符和表达式列。

注意：若尝试粘贴到不存在的单元格中，文本将粘贴到单独的单元格中。

复制的内容的数据类型需与您粘贴到的单元格的数据类型相匹配。您可以添加新列并粘贴数据、将数据粘贴到数据类型正确的单元格中，或按需更改单元格的数据类型。

注意：要复制整个数值，请选择**编辑 > 以全精度复制**。该选项仅适用于固定小数格式。

向数据表添加行

向数据表添加任意数量的行

1. 选择**行 > 添加行**。
2. 输入要添加的行数。
3. 指定要添加新行的位置 （在数据表的开头或末尾，或在特定行之后）。

4. 点击**确定**。

向数据表末尾添加单个行

- 在最后一行之下，点击单元格中的任意位置，然后开始键入。
- 在最后一行之下，双击空的行号区域。

向数据表添加列

有多种向数据表添加新列的方式：

- 双击数据表最后一列右侧的空白区域可添加一列。有关在添加列后更改特性的详细信息，请参见第 253 页的“关于“列信息”窗口”。
- 选择**列 > 新建列**。使用该选项添加一个或多个列并更改其特性。
- 双击数据网格中行号上方的上部小三角。“添加多列”窗口随即显示。

注意：

- 最初创建列时，您可以选择使用初始数据值填充该列。请参见“设置 JMP 列属性”一章中第 262 页的“初始化列中数据”。 但是，修改单元格后，该选项不再显示。
- 若在数据表中选定了任何行，双击以创建新列时将取消选择这些行。使用“列”>“新建列”创建新列则不会更改行选择。
- 不能编辑“依据”组或链接子集的列中的数据。

使用 “新建列”添加一个或多个列

1. （可选）选择一个或多个新列要添加到其前或其后的列。
2. 选择**列 > 新建列**。
3. 更改列名。

   默认情况下，新列名为**第 1 列**、**第 2 列**，依此类推。
4. 选择所有列的数据类型、建模类型和格式。请参见第 253 页的 “关于 “列信息”窗口”。
5. （可选）选择所有列的初始数据值。请参见 “设置 JMP 列属性”一章中第 262 页的 “初始化列中数据”。
6. 指定新列要放置到的位置。若不选择任何选项，该列将添加到最后一列之后。
7. （可选）对于多个列，指定是否应将这些列分组。请参见第 187 页的“组合数据表中的列”。
8. 要添加更多列，请执行以下步骤：

点击**应用**添加新列。

点击**下一个**，更改列属性，然后点击**应用**。

点击**下一个**添加另一列。

9. 点击**确定**保存所做更改。

图 4.2 “新建列”窗口

提示：要在创建列后更改建模类型，请点击列面板中的建模类型图标，另选一个类型。

使用 “添加多列”添加一个或多个列

1. 双击数据网格中的上部小三角区域。“添加多列”窗口随即显示。

图 4.3 添加多列

2. 更改每列的前缀。

默认情况下，新列名为**第 1 列**、**第 2 列**，依此类推。

3. 输入要添加的列数。
4. 指定是否应对列分组。请参见第 187 页的 “组合数据表中的列”。
5. 选择所有列的数据类型。请参见第 253 页的 “关于 “列信息” 窗口”。
6. 指定新列要放置到的位置。
7. （可选）选择所有列的初始数据值。请参见 “设置 JMP 列属性” 一章中第 262 页的 “初始化列中数据”。
8. 点击**确定**保存所做更改。

使用顺序数据填充数据表列

要使用重复的数据序列或一系列延续值填充列，请执行以下步骤：

1. 在列中创建数据序列。

图 4.4 数据序列示例

	第 1 列
1	2
2	4
3	6
4	8

2. 突出显示包含序列数据的单元格。单元格可位于不同列中。
3. 右击选定的单元格并选择**填充**下的选项。

填充选项

重复序列至表末 指定对选定内容之下的单元格使用选定单元格进行重复填充。

继续序列至表末 指定对选定内容之下的单元格依照在选定单元格中发现的模式继续填充。例如，若选定单元格包含数字 1 和 2，那么其余单元格将使用 3、4、5、6 等继续填充。若选定单元格包含数字 2 和 4，那么其余单元格将使用 6、 8、 10、 12 等继续填充。

重复序列至 JMP 将在选定单元格中依照发现的模式重复至您指定的行号。

继续序列至 JMP 将在选定单元格中依照发现的模式延续至您指定的行号。

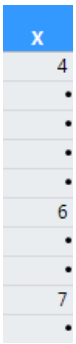
使用数据表列中之前的值替换缺失值

若某列包含缺失值，您可以使用上一个单元格中的值替换那些值。例如，若选定“4, ., ., ., 6, ., ., 7, .”，则这些值将改为“4, 4, 4, 4, 4, 6, 6, 6, 7, 7”。

1. 选择包含缺失值和替换值的单元格。

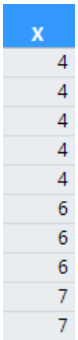
   您选择的第一个单元格必须为缺失值的前一个单元格。在此之后，您可以选择非邻近单元格。

图 4.5 缺失值



2. 右击并选择**填充** > **使用之前的值替换缺失值**。

图 4.6 替换值



将单元格公式输入数据表列

在数值数据表列中，您可以输入前面带有等号 (=) 的单元格表达式。JMP 对该表达式求值，并将新数字存储为该单元格的值。不同于列公式，单元格表达式不会被存储。单元格表达式可以包含运算符、常数以及全局和列变量。

输入表达式

1. 点击要输入表达式的单元格。

2. 键入等号 (=)，然后键入表达式 （表 4.1）。
3. 按 Enter。

表 4.1 表单元格中的表达式示例

示例表达式	单元格值
=sqrt(2)	1.41
=456+890	1346
=height+weight	针对与输入表达式的单元格同处一行的 height 和 weight 列中的单元格值求和。
=height[1]	显示在 height 列的第 1 行中找到的值。

选择数据表中的行

选择一整行

- 点击包含行号的空白区域。

选择特定行号

- 选择**行 > 行选择 > 转至行**，并键入所需行号。

选择多行

- 对于连续选择：
  - 在行号上方点击并拖动光标。
  - 按 Shift 键并点击所需范围内的第一行和最后一行。
  - 按 Shift 键并按上箭头键或下箭头键。
- 对于不连续选择：
  - 按 Ctrl 键并点击每行。

选择或取消选择所有行

- 要选择所有行，请选择**行 > 行选择 > 选择所有行**。

  或
- 按 Shift 键并点击数据网格左上角的下三角区域可选择所有行。在同一区域中再次点击可取消选择所有行（图 4.7）。
- 要清除数据表中的所有突出显示区域，请按 Esc 键。

图 4.7 下三角区域

	姓名	年龄	性别	身高	体重
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	79
9	BARBARA	13	F	60	112
10	ALICE	13	F	61	107

选择随机行

1. 选择**行 > 行选择 > 随机选择**。
2. 您可以随机选择特定数目的行，或是占总行数一定比例的行：
   - 输入一个整数以选择该数目的行。
   - 输入 0 到 1 之间的值，选择该比例的行。

   例如，输入 10 可选择 10 行。输入 0.1 可选择 10% 的行。

反转行选择

- 选择**行 > 行选择 > 反转行选择**。

选择占优行

1. 选择**行 > 行选择 > 选择占优**。
2. 选择您要将其值用于确定占优的列。
3. 为每列选择占优的高值或低值。
4. 点击**确定**。

注意以下有关占优值和行的事项：

- 基于您的指定，若一个值高于或低于另一个值，则前者占优。
- “选择占优”选项将选择未被任何其他行占优的每一行。仅在一行的**所有**值占优另一行的值时，该行才占优另一行。
- 行的结果集被称为 Pareto 边界。

在新列中保存当前行选择

1. 选择**行 > 行选择 > 在列中指明选择项**。
2. 键入列名。
3. 标记选定和取消选定的行。

4. 点击**确定**。

提示：若您在创建新列后重复该过程并指示相同的列名，则原始列会被覆盖。您在数据表中对“已选定”或“未选定”值进行的更改将会丢失。通过清除“在列中指明选择项”窗口中的相应框，可避免覆盖这些值。例如，要避免覆盖“未选定”值，确保清除“未选定”框。

选择已排除、已隐藏或已添加标签的行

1. 选择**行 > 行选择**。
2. 从以下选项中进行选择：
   - 选择排除行
   - 选择隐藏行
   - 选择已添加标签行

注意：有关已排除、已隐藏或已添加标签的行，请参见第 202 页的“为行和列指定特征”。

选择包含重复值的行

1. 选择**行 > 行选择 > 选择重复行**。

   除第一行之外的所有重复行都被选定。若未选定任何列，则会为所有列选定重复行。

   若选定了列，则会在这些列的行中找到重复值。

   该匹配区分大小写。

查找行中的匹配单元格

使用“选择匹配单元格”选项查找具有匹配值的单元格。

- 选择**行 > 行选择 > 选择匹配单元格**

  或

- 右击突出显示的行号之一，然后选择**选择匹配单元格**。

选择包含特定值的行

JMP 可以搜索特定值（或文本字符串）并突出显示数据表中包含该特定值的所有单元格。

1. 选择**行 > 行选择 > 选择符合条件的行**。

图 4.8 指定选择行的条件

2. 从列列表中，突出显示要选择行的列的名称。
3. 使用下拉菜单从列表中选择一个条件 （等于、不等于或其他条件）。请参见图 4.8。
4. 键入搜索值。
5. 点击**确定**。

您还可以指定以下可选功能：

- 要比较两列的值，请点击**比较列**复选框。从列列表中选择要比较的列。
- 要使搜索区分大小写，请点击**区分大小写**旁边的框。
- 可以将“选择行”窗口中的 Where 子句保存到剪贴板、数据表或脚本窗口。选择**保持对话框打开**，选择“行”和其他选项，然后从“行选择”红色小三角菜单中选择一个“保存脚本”选项。将 JSL 添加到重新运行该查询的脚本：

```
dt = Open("$SAMPLE_DATA/Big Class.jmp");
Current Data Table) << Row Selection(
   Select where:age ==15),
   Dialog( Edit( Source Column( :age ) == "" ), Keep dialog open )
);
```

若当前在数据表中已选定行，则可以指定以下功能：

- 点击**当前选择**下的选项，指示 JMP 如何处理这一当前选择：
  - **清除**可取消突出显示当前选定的行，并选择包含指定值的所有行。
  - **扩展**可使当前选定行保持选定状态，同时还可选定包含指定值的行。
  - **限制**可选择在当前选定的数组中包含指定值的行。
- 点击**添加条件**可向列表添加条件。

- 要在搜索中添加更多条件，请重复之前的步骤。在**选择行**区域中点击所需项，指定是否希望 JMP 有条件地选择行：满足**所有**条件，还是满足**任意**条件。

定位下一选定行

通过选择**行 > 下一选定行**，您可以定位并闪现当前行之后的下一选定行。

每次选择**行 > 下一选定行**时，都会找到并闪现下一选定行。找到最后一个选定行后，会发出蜂鸣音信号。

若在大型数据集内间断选定了多行，而您想在数据表中通览这些选定行，此时最好使用该功能。

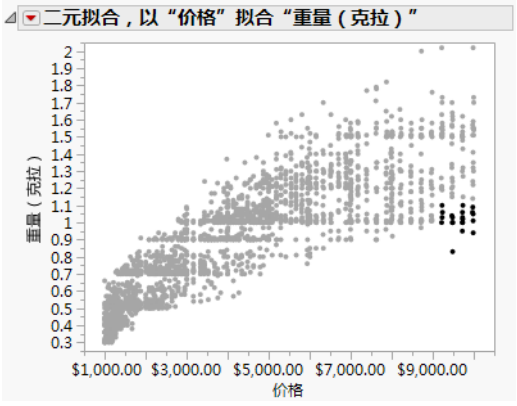
示例：找到下一选定行

1. 选择**帮助 > 样本数据库**并打开 Diamonds Data.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**重量 （克拉）**，然后点击 **Y，响应**。
4. 选择**价格**，然后点击 **X，因子**。
5. 点击**确定**。
6. 选择**工具 > 套索**。

   若看不到菜单栏，则悬停在标题栏下方的蓝条上即可显示菜单栏。
7. 用套索套住图底部 10,000 美元价格附近的某些点 （图 4.9）。
8. 在数据表中，选择**行 > 下一选定行** （也可以按 F7 键）。

您可以轻松浏览选定行以查看每行的数据。

图 4.9 选定的点



选择数据表中的列

有多种选择列的方式：

- 在数据表本身中选择列。请参见第 178 页的“如何选择列”。
- 在有许多列的数据表中，在“列查看器”中按照特性、属性和统计量选择列。请参见第 179 页的“在 JMP 列查看器中选择列”。
- 查找下一个或上一个选定的列。请参见第 185 页的“定位数据表中的下一个选定列和上一个选定列”。

如何选择列

选择用于在数据表中选择列的若干方法之一。

选择一整列

- 在数据网格中，点击列名周围的空白区域。

  或

- 在列面板中，点击列名。

选择特定列号

1. 选择**列** > **列选择** > **转至**。
2. 输入列号或列名，然后点击**确定**。

选择多列

- 对于连续选择：
  - 在列名上方点击并拖动光标。
  - 按 Shift 键并点击所需范围内的第一列和最后一列。
  - 按 Shift 键并按左箭头键或右箭头键。
- 对于彼此不相邻的多个列：
  - 按 Ctrl 键并点击每列。

选择或取消选择所有列

- 按 Shift 键并点击数据网格左上角的上三角区域可选择所有列。在同一区域中再次点击可取消选择所有列。

图 4.10 上三角区域

	姓名	年龄	性别	身高	体重
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	79
9	BARBARA	13	F	60	112
10	ALICE	13	F	61	107

提示：要清除数据表中的所有突出显示区域，请按 Esc 键。

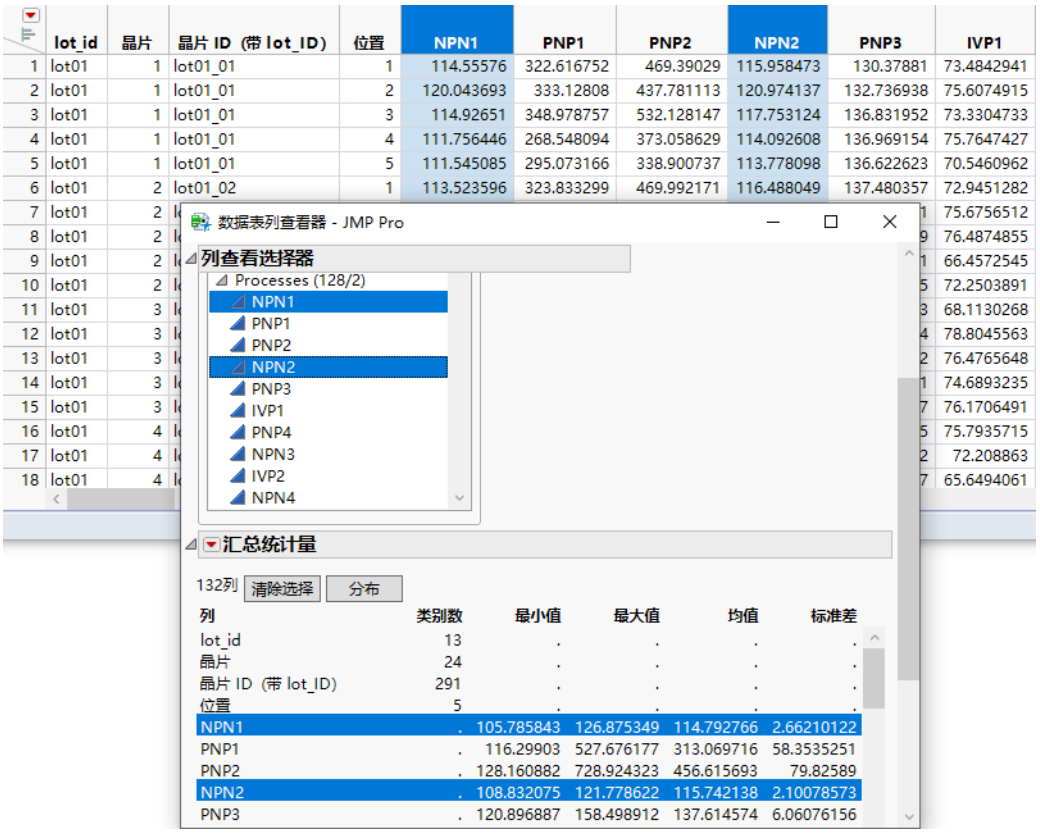
反转列选择

选择**列 > 列选择 > 反转列选择**。选择当前取消选定的所有列，并且取消选定当前选定的所有列。

在 JMP 列查看器中选择列

“列”菜单中的“列查看器”选项可帮助您按照特性、属性和统计量选择列，特别是在有许多列的数据表中。您可以查看这些列的汇总统计量和属性、查看汇总统计量中的四分位数、对数据取子集，以及执行其他操作。而且“列查看器”窗口中的列还链接至数据表列。

图 4.11 “列查看器”中的链接列



	lot_id	晶片	晶片 ID (带 lot_ID)	位置	NPN1	PNP1	PNP2	NPN2	PNP3	IVP1
1	lot01	1	lot01_01	1	114.55576	322.616752	469.39029	115.958473	130.37881	73.4842941
2	lot01	1	lot01_01	2	120.043693	333.12808	437.781113	120.974137	132.736938	75.6074915
3	lot01	1	lot01_01	3	114.92651	348.978757	532.128147	117.753124	136.831952	73.3304733
4	lot01	1	lot01_01	4	111.756446	268.548094	373.058629	114.092608	136.969154	75.7647427
5	lot01	1	lot01_01	5	111.545085	295.073166	338.900737	113.778098	136.622623	70.5460962
6	lot01	2	lot01_02	1	113.523596	323.833299	469.992171	116.488049	137.480357	72.9451282
7	lot01	2								75.6756512
8	lot01	2								76.4874855
9	lot01	2								66.4572545
10	lot01	2								72.2503891
11	lot01	3								68.1130268
12	lot01	3								78.8045563
13	lot01	3								76.4765648
14	lot01	3								74.6893235
15	lot01	3								76.1706491
16	lot01	4								75.7935715
17	lot01	4								72.208863
18	lot01	4								65.6494061

列	类别数	最小值	最大值	均值	标准差
lot_id	13	.	.	.	.
晶片	24	.	.	.	.
晶片 ID (带 lot_ID)	291	.	.	.	.
位置	5	.	.	.	.
NPN1	.	105.785843	126.875349	114.792766	2.66210122
PNP1	.	116.29903	527.676177	313.069716	58.3535251
PNP2	.	128.160882	728.924323	456.615693	79.82589
NPN2	.	108.832075	121.778622	115.742138	2.10078573
PNP3	.	120.896887	158.498912	137.614574	6.06076156

使用“列查看器”可以快速查看数据表特征。例如，“汇总统计量”报表显示哪些列包含缺失值。您可以在报表中选择这些列，然后从数据表中排除它们。

图 4.12 标识缺失值

汇总统计量

39列 清除选择 分布

列	数目	缺失值个数	类别数
我来自大家庭	448	0	2
刷牙	448	0	4
牙线	448	0	4
起床后刷牙	448	0	2
饭后刷牙	448	0	2
睡前刷牙	448	0	2
其他时间刷牙	448	0	.
刷牙其他	8	440	7
刷牙分隔	448	0	12
起床后使用牙线	448	0	.
饭后使用牙线	448	0	.
睡前使用牙线	448	0	.
其他时间使用牙线	448	0	.
牙线其他	27	421	24
牙线分隔	431	17	10
洗牙频率	448	0	4
补牙颗数	448	0	4
牙育成本	448	0	.
牙线成本	448	0	.

“汇总统计量”报表显示以下信息：

- 总行数（**数目**）
- 带缺失值的行数（**缺失值个数**）
- 类别数（**类别数**）
- 对于连续数据，**最小值**、**最大值**、**均值**和**标准差**

其他选项包括：

清除选择 取消选择数据表和“列查看器”中的列。该选项可确保在您开始选择列之前未选定任何列。

子集 基于选定列创建链接的子集数据表。

显示汇总 为选定列创建链接的“汇总统计量”报表。右击可选择按列排序或创建数据表之类的选项。选择**显示四分位数**可包括下四分位数、上四分位数和四分位间距。要基于报表中的所有列创建链接上的数据表，请点击“汇总统计量”红色小三角并选择“数据表视图”。

查找带属性的列 在“带属性的列”报表中显示列属性列表。选择要查找的属性，然后点击**确定**基于所有列创建链接报表。或者，可以首先在“选择列”列表中选择列，然后仅显示这些列的属性列表。

注意：

- 每次点击**显示汇总**或**查找带属性的列**按钮时，就会在窗口中添加新的报表。要删除某个报表，请从该报表的红色小三角菜单中选择**删除**。
- 从自“列查看器”创建的数据表视图中删除行是无法恢复的操作。

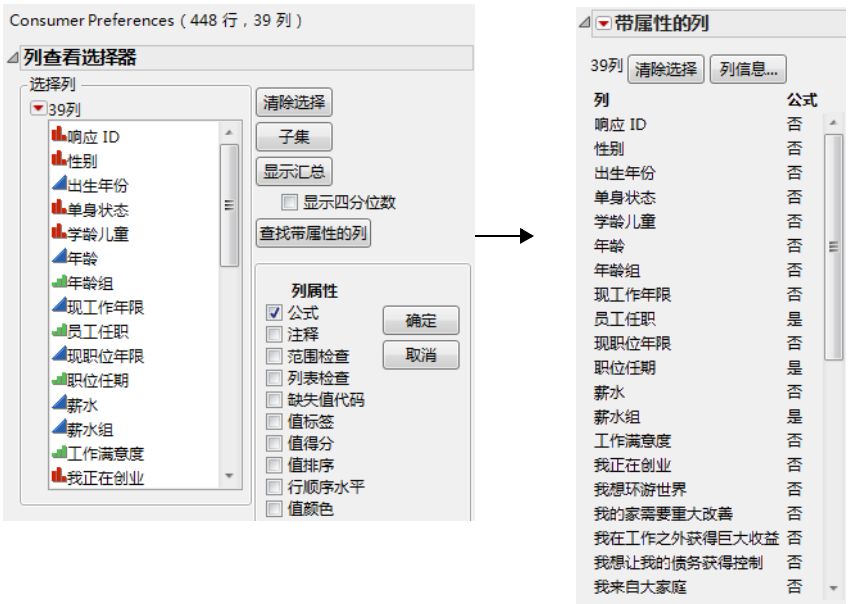
关于查找带特定属性的列的示例

该示例显示如何查找带有“公式”属性的列，然后同时查看所有公式。

1. 选择**帮助** > **样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**列** > **列查看器**打开 “数据表列查看器”窗口。
3. 选择**查找带属性的列**，再选择**公式**，然后点击**确定**。

   “带属性的列”报表随即显示。若干列包括 “公式”属性。由于该列表过长，您希望所有公式列一起显示。

图 4.13 选择 “公式”列属性



4. 右击该报表，依次选择**按列排序**、**公式**，然后点击**确定**。

   带有 “公式”属性的列将显示在报表顶部。

图 4.14 按列排序

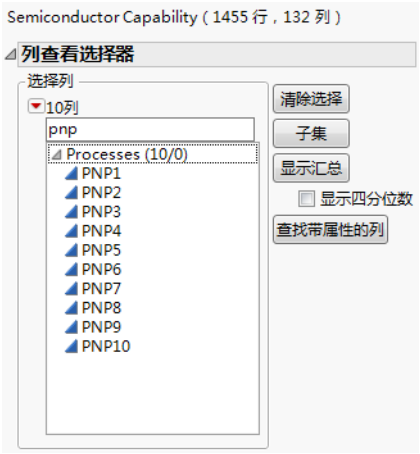
5. 选择**员工任职**、**职位任期**和**薪水组**列，然后选择**列信息**。
   选定列的公式显示在数据表的 “列设置” 窗口中。

关于显示汇总统计量的示例

该示例显示如何查找带有低标准差的列。若您想要从分析中删除或排除数据，这会很有帮助。

1. 选择**帮助** > **样本数据库**并打开 Semiconductor Capability.jmp。
2. 选择**列** > **列查看器**打开 “数据表列查看器” 窗口。
3. 在框中键入 PNP 并按 Enter 键。
   分组过程随即显示。
4. 展开过程分组以显示 PNP 列。

图 4.15 按名称过滤列



5. 点击**显示汇总**以添加 “汇总统计量”报表。
   各行显示各列的最小值、最大值、均值和标准差。

图 4.16 选定列的汇总统计量

汇总统计量

10列 清除选择 分布

列	最小值	最大值	均值	标准差
PNP1	116.29903	527.676177	313.069716	58.3535251
PNP2	128.160882	728.924323	456.615693	79.82589
PNP3	120.896887	158.498912	137.614574	6.06076156
PNP4	149.453273	373.172883	256.375597	32.6073816
PNP5	-55.762191	-40.722372	-47.178172	2.047573
PNP6	0	0	0	0
PNP7	-16.888557	-14.961357	-15.89982	0.28816317
PNP8	-44.913142	-34.771668	-39.21972	1.49081788
PNP9	-79.665058	-66.929598	-73.35004	1.94997812
PNP10	-0.6442798	-0.623852	-0.6335016	0.00299937

6. 在报表中右击并选择**按列排序**。
7. 选择**标准差**和**升序**，然后点击**确定**。
   请注意， PNP6 没有标准差，因为最小值、最大值和均值均为 0。

图 4.17 经过排序的 “标准差” 列

汇总统计量

10列 清除选择 分布

列	最小值	最大值	均值	标准差
PNP6	0	0	0	0
PNP10	-0.6442798	-0.623852	-0.6335016	0.00299937
PNP7	-16.888557	-14.961357	-15.89982	0.28816317
PNP8	-44.913142	-34.771668	-39.21972	1.49081788
PNP9	-79.665058	-66.929598	-73.35004	1.94997812
PNP5	-55.762191	-40.722372	-47.178172	2.047573
PNP3	120.896887	158.498912	137.614574	6.06076156
PNP4	149.453273	373.172883	256.375597	32.6073816
PNP1	116.29903	527.676177	313.069716	58.3535251
PNP2	128.160882	728.924323	456.615693	79.82589

8. 在 “汇总统计量” 报表中，选择 PNP6 所对应的行，然后显示数据表。
9. 查看数据表并按下 Delete 键删除选定列。

   将从数据表中立即删除该列。
10. 要关闭 “列查看器”，请点击窗口右上角 (Windows) 或窗口左上角 (macOS) 的 X 按钮。

定位数据表中的下一个选定列和上一个选定列

您可以通过选择**列** > **列选择** > **下一个选定列**查找当前列之后的下一个选定列。类似的，您可以通过选择**列** > **列选择** > **上一个选定列**查找当前列之前的上一个选定列。

每次您选择其中一个选项时，下一个或上一个选定列会出现和闪现。该选项仅在列被选定时才可用。

您可能需要使用该功能查看大型数据表中间断选定的列。

调整列和行的大小

调整行大小

通过拖动行的一个边框可调整行的大小。所有行会被调整为相同高度。每个单元格内出现的图形会根据行的高度收缩。

调整列大小

要调整列的大小，请将列边框向右或向左拖动。

要将多列调整为相同宽度，请执行以下步骤：

1. 选择要调整大小的列。
2. 按 Alt 键 (Windows) 或 Option 键 (macOS) 并拖动列边框之一。所有列会被调整为相同宽度。

在数据表中组织数据

从许多方法中选择一种来重新排列数据表中的数据。

- “删除数据表中的行”
- “删除数据表中的列”
- “重新排列数据表中的列”
- “组合数据表中的列”
- “移动数据表中的值”
- “为数据表中的单元格着色”
- “编辑或删除数据表中的单元格”
- “编辑数据表中的列名”
- “隐藏和排除数据表中的行”
- “排除数据表中的行”
- “隐藏数据表中的行”
- “删除数据表中的列”
- “隐藏数据表中的列”
- “查看数据表中缺失数据的模式”
- “查找和替换数据表中的单元格值”
- “编辑数据表中的单元格”
- “行列的右击菜单”
- “比较数据表”

删除数据表中的行

要删除 JMP 数据表中的行，请执行以下步骤：

1. 突出显示要删除的行。
2. 按 Delete 键，或右击行号并选择**删除行**。

警告：在尝试删除数千行时，若用户的计算机内存不足，无法保存数据实现撤销操作，系统会显示一条警示。可以选择更少的行进行删除，或从表面板红色小三角菜单中选择**禁用撤销**。该选项将删除撤销历史中的所有操作，且不记录将来的操作。选择**禁用撤销**选项后，其仅在数据表打开时有效；该设置不会与数据表一起保存。

删除数据表中的列

要删除 JMP 数据表中的列，请执行以下步骤：

1. 突出显示要删除的列。
2. 按 Delete 键，或右击并选择**删除列**。

重新排列数据表中的列

您可以按照数据表列的名称、数据类型或建模类型对这些列重新排列或排序，也可以反转当前顺序。要对列重新排序，请选择**列 > 列重新排序**，然后选择以下选项之一：

移动选定列 将选定列移至数据表中的特定位置。在 “移动选定列”窗口中指定放置选定列的位置：

至最前 将选定列移至数据表中最左侧的位置。

至最后 将选定列移至数据表中最右侧的位置。

之后 将选定列移至您标识的列之后。

原始顺序 使列恢复最后保存数据表时列所处的顺序。

按名称重新排序 按列名的字母顺序从左到右排列各列。

按数据类型重新排序 按数据类型 （表达式、行状态、字符、数值）以字母顺序从左到右排列各列。

按建模类型重新排序 按建模类型（无、连续型、有序型、名义型、多重响应、非结构化文本、向量）以字母顺序从左到右排列各列。行状态列不具有建模类型，所以显示在最后。

倒序 反转数据表列的顺序。

组合数据表中的列

将列组合在一个标题下，以便管理大量的列并有助于指定分析角色。组合的列以分级显示格式显示在列面板中。

组合或取消组合列

1. 在数据网格中，选择要组合的列。
2. 从主菜单中，选择**列 > 组合列**或**列 > 取消组合列**。

或

1. 从“列”面板中，选择要组合的列。
2. 右击选定列并选择**组合列**或**取消组合列**。

注意：将为从以下命令生成的数据表自动保留组合的列：**子集**、**排序**、**汇总**、**连接**、**堆叠**和**拆分**。对于**堆叠**命令，若堆叠组中的所有列属于同一列组，该组的名称将用作列名。

移动数据表中的值

要移动数据表中的值，请选择这些值，点击并暂停，然后将这些值拖放至新位置。

提示：点击并拖动选定项时若不暂停，则会进一步扩展选择。

拖放值时，请注意以下事项：

- 单元格保留其所有特征和列属性。
- 移动单元格后，最初选定的单元格中将显示缺失值。
- 选定的单元格和目标单元格必须具有相同的数据类型。
- 若将一组单元格拖至表中的空白区域，则自动创建新列。
- 新列具有原始列的显示格式和建模类型。

指定行要移至的位置

1. 突出显示要移动的行。
2. 选择**行 > 移动行**。
3. 在 “移动行”窗口中指定要将行移至的位置：
   - 移至表的开头（**起始处**）
   - 移至表的末尾（**末尾处**）
   - 移至特定行号之后（**该行后:**）

将内容移入另一个 JMP 窗口

在 Windows 上，您可以将选定内容拖至最小化窗口上。该最小化窗口将移至最前，您可以将选定内容粘贴到该窗口中。在 JMP 中同样可以执行该操作。例如，您可以将选定内容拖至“主窗口”按钮 上（该按钮位于多数窗口的右下角）。随后在“窗口列表”中，可以将内容拖至要移入的窗口上。该窗口将移至最前，您可以在其中放置内容。

提示：若看不到 JMP 主窗口按钮，请选择**视图 > 状态栏**。

例如，您可以将选定的列、行或单元格从一个数据表拖入另一个数据表；将选定文本从一个脚本窗口拖入另一个脚本窗口；或是将选定内容从报表拖入记录。

为数据表中的单元格着色

您可以选择一个或多个数据表单元格并更改颜色以突出显示特定值。右击一个或多个单元格，选择**为单元格着色**，然后选择所需颜色。要删除单元格中的颜色，请选择单元格，右击并选择**列 > 清除颜色**。

编辑或删除数据表中的单元格

要编辑或删除单元格中的内容，请执行以下步骤：

1. 点击包含要编辑或删除的值的单元格。
2. 按 Delete 键。
3. 要编辑该值，请再次点击该单元格，然后编辑该单元格的值。

编辑数据表中的列名

要编辑单个列名，请选择列并开始键入。您还可以在“列信息”窗口中编辑列名，或选择标题并按 Enter 键。

要一次编辑所有列名，请点击**列 > 列名**并选择以下选项之一：

上移 将第一行数据上移以替换列名。

上移和追加 将第一行数据上移并将条目追加到现有列名。

下移 将列名下移，成为一行数据。各列随后命名为“第 1 列”、“第 2 列”，依此类推。

重新编码列名称 打开“重新编码”窗口以便对列名重新编码。请参见第 230 页的“对列中数据重新编码”。

隐藏和排除数据表中的行

针对您不想包括在分析或图计算中的行以及您不想在图中显示的行应用“隐藏和排除”行状态。这些行不包括在随后的分析中。

注意：

- “图形”菜单下的图（“刻画器”除外）将立即重新计算以反映隐藏和排除的行。
- 对于许多平台，报表和图不会立即更新以反映排除和隐藏的观测。您需要重新运行分析以重新计算分析结果和相关图。相关的例外情况，请参见第 190 页的“随隐藏和排除操作立即更新的平台”。
- 排除和隐藏行时，将在自动更新的所有打开的报表和图中排除和隐藏这些行。
- 排除和隐藏行时，带删除线的圆圈（ 代表“排除”）和眼罩图标（ 代表“隐藏”）将显示在行号旁边。
- 行一直保持隐藏和排除状态，直到您突出显示这些行并重新选择**隐藏和排除**

隐藏和排除一个或多个行

1. 选择要隐藏和排除的行。
2. 执行以下操作之一：
   - 右击行号旁边的突出显示区域并选择**隐藏和排除**。

– 从**行**菜单中，选择**隐藏和排除**。

随隐藏和排除操作立即更新的平台

对行应用“隐藏和排除”时，对于“分析”菜单下的多数平台，直接对应于行的点或图元素通常会在图中立即隐藏。不过，随同的报表中显示的计算通常**不**会更新。对于以下平台，应用“隐藏和排除”将立即更新图**和**计算：

- 制表
- 控制图生成器

排除数据表中的行

针对您不想包括在后续计算中的行应用 “排除”行状态。将重新计算统计量而不含排除的行。

要排除一个或多个行，请执行以下步骤：

1. 选择要排除的行。
2. 执行以下操作之一：
   – 右击行号旁边的突出显示区域并选择**排除/撤销排除**。
   – 从**行**菜单中，选择**排除/撤销排除**。

注意：

- 多数情况下，当通过某个点或单个图元素表示某行时，排除该行将**不**在图中隐藏该行。
- 在“图形生成器”中，用一组行表示某个图元素时，将仅使用未排除的行重新构造该图，并且该图不显示对应于排除行的图元素。排除行被视为就像是被隐藏**和**排除一样。
- “图形”菜单下的图（“刻画器”、“平行图”和“散点图矩阵”除外）将立即重新计算以反映排除的行。
- 对于许多平台，报表和图不会立即更新以反映排除的观测。您需要重新运行分析以重新计算图和分析结果。
- “排除”行时，将从自动更新的所有打开的图和报表中排除这些行。
- 带删除线的圆圈 ( ) 将显示在排除行的行号旁边。
- 行一直保持排除状态，直到您突出显示这些行并重新选择**排除/撤销排除**。

隐藏数据表中的行

针对您不想包括在图中的行应用 “隐藏”行状态。

注意：并非所有图都隐藏对应于隐藏行的元素。当用于构造图元素的计算使用来自隐藏行的数据时往往就会发生这种情况。一般的规则是：若行和图元素之间存在一对一对应关系，则隐藏对应于隐藏行的点或元素。

要隐藏一个或多个行，请执行以下步骤：

1. 选择要隐藏的行。
2. 执行以下操作之一：
   - 右击行号旁边的选定区域并选择**隐藏/撤销隐藏**。
   - 从**行**菜单中，选择**隐藏/撤销隐藏**。

注意：

- “隐藏”行时，将从自动更新的所有打开的图中隐藏这些行。
- 隐藏行并**不**会将其从计算和分析中排除。
- 若每个隐藏行都对应于某个点或单个图元素，这些图将在您对行应用“隐藏”行状态时立即更新。
- 根据所用的平台，将使用多个观测计算图元素，但它们不与行直接对应。在这些情况下，图元素通常不会更新以反映隐藏的观测，除非同时排除这些行。
- 对于图元素对应于多组行的某些图，当您“隐藏”行时，该图将会更新以隐藏对应于这些行的图元素。请参见第 191 页的“应用“隐藏”后其分组元素跟着更新的 JMP 图”。
- 眼罩图标 ( ) 将显示在隐藏行的行号旁边。
- 行一直保持隐藏状态，直到您选择这些行并重新选择**隐藏/撤销隐藏**。

应用“隐藏”后其分组元素跟着更新的 JMP 图

对行应用“隐藏”时，在行与图元素之间存在一对一对应关系的图中，点或图元素通常会立即隐藏。不过，对于以下平台和选项，应用“隐藏”将立即更新与各组行对应的图元素：

- 控制图生成器。请参见《质量和过程方法》。
- 图形生成器。条形图、热图和饼图均更新以反映隐藏行。请参见《基本绘图》。
- 气泡图。请参见《基本绘图》。

删除数据表中的列

当数据表包含您不想在分析中考虑的列时，您可以排除这些列，令其不显示在启动窗口的选择列表中。请注意以下事项：

- 排除的列在数据网格中仍然可见。
- 带删除线的圆圈 ( ) 显示在列面板中的列名右侧。

从启动窗口中排除列

1. 选择要排除的一个或多个列。
2. 执行以下操作之一：
   - 右击列面板中突出显示的列名，然后选择**排除/撤销排除**。

– 从**列**菜单中，选择**排除/撤销排除**。

撤销排除列

1. 选择列面板中的列。
2. 执行以下操作之一：
   – 右击列面板中突出显示的列名，然后选择**排除/撤销排除**。
   – 从**列**菜单中，选择**排除/撤销排除**。

隐藏数据表中的列

若数据表中包含您不想在数据网格中显示的列，您可以隐藏这些列。请注意以下事项：

- 隐藏列显示在启动窗口中，可用于分析。
- 眼罩图标 ( ) 显示在列面板中的列名右侧。

隐藏列

1. 选择要隐藏的一个或多个列。
2. 执行以下操作之一：
   – 右击列面板中突出显示的列名，然后选择**隐藏/撤销隐藏**。
   – 从**列**菜单中，选择**隐藏/撤销隐藏**。

撤销隐藏列

1. 选择列面板中的列。
2. 执行以下操作之一：
   – 右击列面板中突出显示的列名，然后选择**隐藏/撤销隐藏**。
   – 从**列**菜单中，选择**隐藏/撤销隐藏**。

查看数据表中缺失数据的模式

若数据表包含缺失数据，最好确定缺失数据是否存在某种模式。该模式可能帮助您发现数据的某种规律。

要查看缺失数据的模式，请执行以下步骤：

1. 在数据表打开时，选择**表 > 缺失数据模式**。
2. 选择您要查找缺失数据模式的列。
3. 点击**添加列**。
4. 选择**计数缺失值代码**将缺失值代码作为缺失值计数。

5. 选择**添加值颜色属性**对包含 0 或 1 （缺失值指示符）的列一致着色。

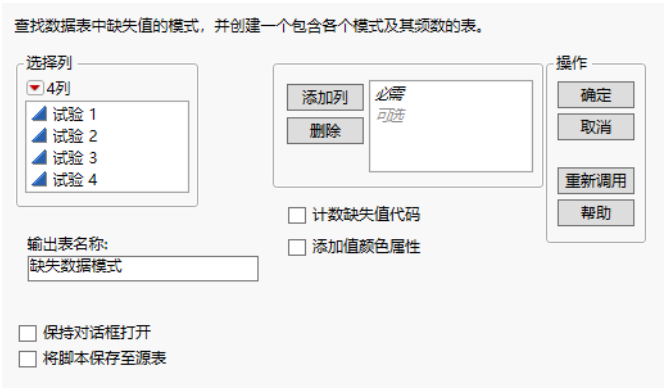
   例如，非缺失始终着蓝色，缺失始终着红色。这对于在各列之间有相似值的数据表尤其有帮助。

6. 点击**确定**。

查看缺失数据模式的示例

1. 选择**帮助** > **样本数据库**并打开 Missing Data Pattern.jmp。
2. 选择**表** > **缺失数据模式**。

图 4.18 “缺失数据模式”窗口

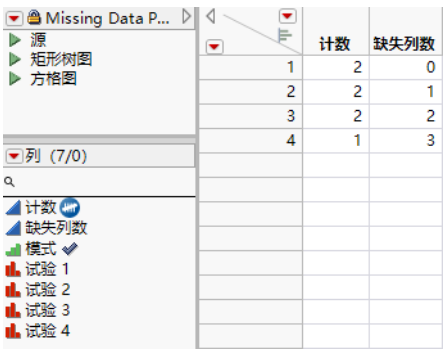


3. 突出显示所有列。

   注意：有关红色小三角菜单中选项的详细信息，请参见 “开始使用 JMP”一章中第 50 页的 ““列过滤器”菜单”。

4. 点击**添加列**。
5. 点击**确定**。

图 4.19 “缺失数据模式”表



提示：要快速创建数据的矩形树图或方格图，请点击表面板的“矩形树图”或“方格图”脚本旁边的绿色小三角。

图 4.19 显示以下模式：

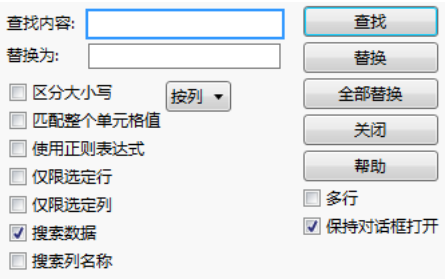
- 第 1 行显示**试验 1**、**试验 2**、**试验 3** 和**试验 4** 中的所有行都没有缺失值的情况一共有两次。
- 第 2 行显示源表中有两行在**试验 4** 列中各有一个缺失值。
- 第 3 行显示源表中有两行在**试验 3** 列和**试验 4** 列中有缺失值。
- 第 4 行显示源表中有一行有三个缺失值发生在**试验 2**、**试验 3** 和**试验 4** 列。

计数列被指定了频数角色。若现在使用“缺失数据模式”数据表运行分析，JMP 会自动将“**计数**”用作频数。请参见“设置 JMP 列属性”一章中第 288 页的“为列指定预选分析角色”。

查找和替换数据表中的单元格值

通过选择**编辑** > **搜索** > **查找**选项，您可以查找和替换单元格值。

图 4.20 “查找”窗口



搜索值时适用以下规则：

- 要查找隐藏列中的值，请撤销隐藏该列。
- 无法修改在锁定列中找到的值。
- **撤销**命令仅适用于**替换**。无法撤销**全部替换**。
- 若数据表包含值标签，使用**搜索**命令将搜索实际值，而无法搜索标签。请参见“设置 JMP 列属性”一章中第 266 页的“值标签”。
- 若数据表包含带格式的值（如日期、时间或持续时间），使用搜索命令将搜索带格式的值而不是实际值。

“查找”窗口选项

使用以下选项细化搜索：

区分大小写 执行区分大小写的搜索，有助于查找专有名词或其他大写字词。

匹配整个单元格值 检测空格，这有助于您搜索字符列中的一连串字词，或查找不需要首尾空格的字符串。

提示：要查找缺失字符值，请将**查找内容**框留空，并选中**匹配整个单元格值**。要查找缺失数值，请在**查找**框中插入一个句点，并选中**匹配整个单元格值**。

使用正则表达式 假定查找字符串是一个正则表达式而不是在**查找内容**框中输入的文字字符串。正则表达式遵循标准的语义。

仅限选定行 将搜索限定为选定行。

仅限选定列 将搜索限定为选定列。

搜索数据 仅搜索数据单元格 （不搜索列名）。

搜索列名 仅搜索列名 （不搜索数据单元格）。

按列 从上到下逐列搜索表列，直到搜索到最右侧列的最后一个单元格或是您停止了搜索。

按行 从左到右逐行搜索数据表，直到最后一行的最右侧单元格或是您停止了搜索。

多行 将 “查找”框和 “替换”框从 1 行增大至 3 行。可使用 Enter 键在字段中插入回车符。

提示：您也可以通过点击并拖动 “查找”框和 “替换”框增大框大小。若使用复制和粘贴，这两个框将重新调整为 1 行大小，但所有文本仍显示在其中。

保持对话框打开 搜索期间保持 “查找”窗口打开。

搜索操作

本节介绍您可能会执行的一些常见搜索。

从搜索数据表中的值入手。搜索将从第一列中的第一个单元格开始，搜索每个单元格，直至找到搜索值或到达表的末尾。

替换当前突出显示的单元格值

在**替换为**框中输入一个值，然后点击**替换**。或者，若 “搜索”窗口已关闭，请选择**编辑** > **搜索** > **替换**。若替换值是缺失值，当前突出显示的单元格内容将变为缺失值。

替换所有出现的指定值

在**替换为**框中输入一个值，然后点击**全部替换**。或者，若 “搜索”窗口已关闭，请选择**编辑** > **搜索** > **全部替换**。

替换值并搜索下一个值

在**替换为**框中输入一个值，然后点击**替换**。或者，若 “搜索”窗口已关闭，请选择**编辑** > **搜索** > **替换并查找下一个**。或者，按 Ctrl+L。

将选定值用作查找内容值

在数据表中，选择一个值。选择**编辑** > **搜索** > **使用选择项进行查找**。下一步，选择**编辑** > **搜索** > **查找**。在数据表中选定的值已经输入**查找内容**字段。

将选定值用作替换为值

在数据表中，选择一个值。选择**编辑** > **搜索** > **使用选择项进行替换**来填充**替换为**字段。

在 “搜索”窗口关闭时查找下一个值

选择**编辑** > **搜索** > **查找下一个**。或者，按 Ctrl+G 或 F3（在 Windows 中）。

查找缺失值：

- 要查找缺失字符值，请将**查找内容**字段留空，并选择**匹配整个单元格值**。
- 要查找缺失的数值，请在**查找内容**文本框中键入一个句点。

编辑数据表中的单元格

使用“行编辑器”可以浏览或编辑单元格，一次处理一行。使用以下方法之一打开“行编辑器”：

- 选择**行** > **行编辑器**。
- 在数据表中，双击行号区域。所用的行将成为“行编辑器”中显示的第一行。
- 在报表窗口中，右击某个图形并选择**行编辑器**。

图 4.21 行编辑器

请注意以下几点：

- 若打开了某个报表窗口，而您希望编辑的数据自动体现在该窗口中，请务必打开“自动重新计算”功能。请参见“JMP 报表”一章中第 394 页的“自动重新计算”。
- 若数据表包含值标签，“行编辑器”会显示这些标签；当单元格突出显示进行编辑时，它将显示实际值。请参见“设置 JMP 列属性”一章中第 266 页的“值标签”。

行编辑器按钮

点击箭头按钮可浏览选定行或整个数据集（若未选定任何行）。

显示上一行。

显示上一选定行。

使行在图形中闪烁。

显示下一选定行。

显示下一行。

搜索行。请参见第 185 页的 “调整列和行的大小”。

在数据表末尾创建新行。

注意：当您在“行编辑器”中更改字段时，使用“行编辑器”对行所做的更改会写入数据表。您仍需要将更改保存至数据表。

行编辑器选项

“行编辑器”中的红色小三角菜单包含以下选项：

下一选定行 显示有关当前行之后的选定行的信息。

上一选定行 显示有关当前行之前的选定行的信息。

下一行 显示有关当前行之后的行的信息，不论是否选定了该行。

上一行 显示有关当前行之前的行的信息，不论是否选定了该行。

保存 保存数据表以及您通过 “行编辑器”对其所做的任何更改。

新建行 在数据表中创建新行。

查找 显示好像您在选定**行 > 行选择 > 选择符合条件的行**时出现的窗口。选择**当前选择**菜单中的选项之一，然后突出显示要选择其行的列。键入您希望 JMP 搜索的值。请参见第 185 页的 “调整列和行的大小”。

闪烁 使当前行突出显示，并快速闪烁。

注意：无法编辑锁定列中的文本或锁定的数据表。请参见“设置 JMP 列属性”一章中第 253 页的“锁定”和第 244 页的“锁定数据表”。

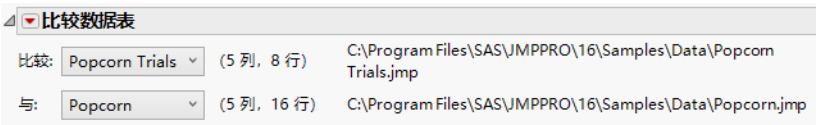
行列的右击菜单

在行号区域或列名区域中的列顶部右击时，将出现弹出式菜单。使用这些菜单可以快速访问行菜单和列菜单中的选定命令。有关这些选项的详细信息，请参见“开始使用 JMP”一章中第 44 页的“列的右击选项”和“开始使用 JMP”一章中第 45 页的“行的右击选项”。

比较数据表

JMP 可以比较两个打开的数据表，并报告数据、脚本、表变量、列名、列属性和列特性之间的差异。每个数据表中的列数和行数都显示在“比较数据表”窗口的顶部。在本例中，对样本数据表 Popcorn Trials.jmp 和 Popcorn.jmp 进行了比较。

图 4.22 关于数据表的基本信息



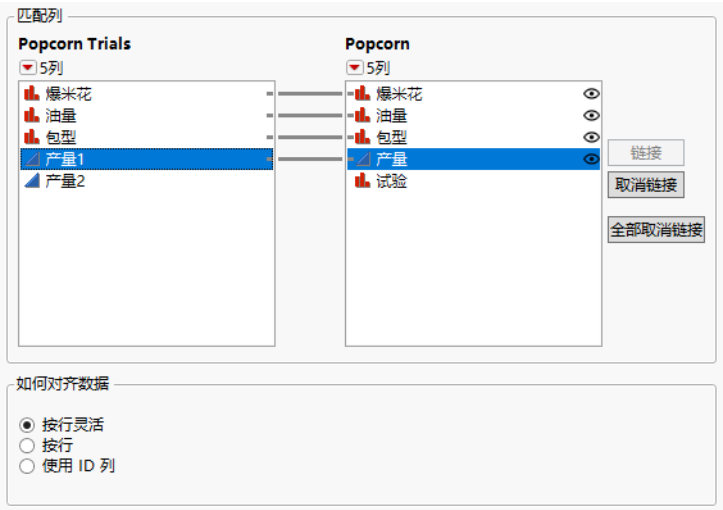
指定匹配列

同名的列将自动匹配。在每个匹配的列之间绘制连线。您也可以手动链接两列。

1. 选择**帮助** > **样本数据库**，然后打开 Popcorn.jmp 和 Popcorn Trials.jmp。
2. 显示 Popcorn Trials.jmp 并选择**表** > **比较数据表**。
3. 从 “与” 列表中，选择 Popcorn.jmp。

   此时应该自动从 “比较” 列表中选择 Popcorn Trials.jmp。
4. 在 “匹配列” 窗格中，选择**产量**和**产量 1**，然后点击**链接**。

图 4.23 手动已链接列



同名列将自动链接。

注意：若点击“比较数据”图标 ，则不会比较已链接的列。这对列的值将显示在结果中，但不会进行比较。

5. 点击**比较**。

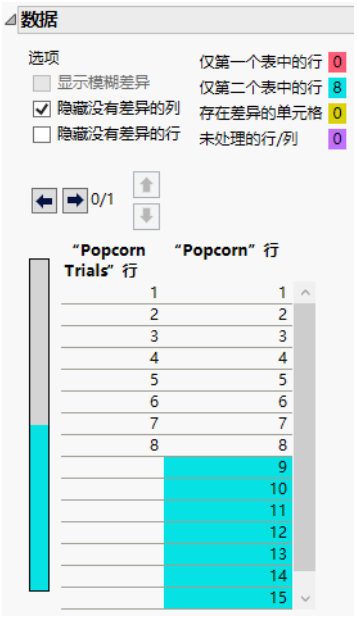
   结果显示在 “数据”报表中。

图 4.24 比较列

前八行相同，所以不显示。其余行 （显示为蓝色）仅显示在第二个表 Popcorn.jmp 中。

6. 在 “数据”报表中，取消选择**隐藏没有差异的行**。

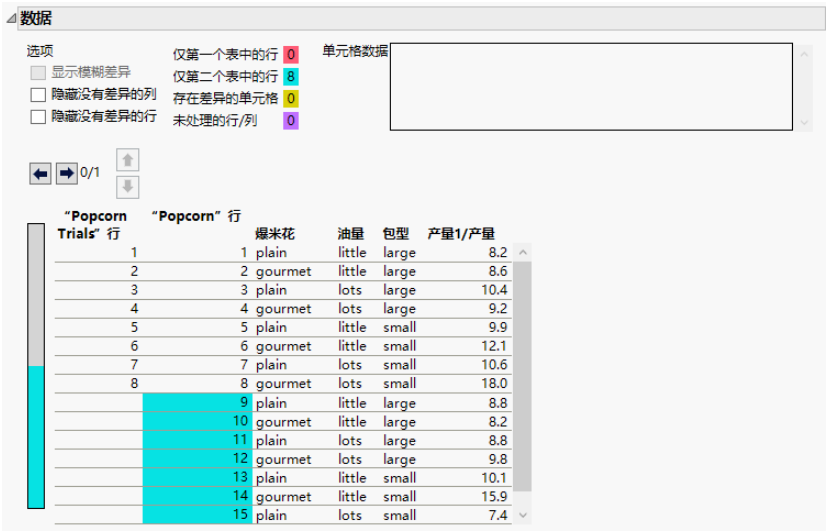
图 4.25 显示没有差异的行



显示前八行是因为它们在两个数据表中是相同的。

7. 取消选择**隐藏没有差异的列**。

图 4.26 显示没有差异的列



同时存在于两个数据表中并且匹配的列将显示出来。您可以选择此选项为匹配的数据提供更多上下文。

注意：若单元格中的数据未完全显示，请选择文本，然后在“单元格数据”框中查看数据。

数据表比较选项

对齐数据选项

按行灵活 搜索要对齐的公共行。若您认为数据表较小且它们几乎相同，请考虑使用该选项。

按行 逐一比较各行。若您已经知道各行应该逐行排列，请考虑该选项。该比较运行起来会快很多。不过，若选择了默认的“按行灵活”，而且比较的时间太长，则可能需要选择该选项。

使用 ID 列 使用选定的 ID 列比较各行。该数据表的各行通过该 ID 列的值唯一标识。若数据表很大、排序不同或有缺失行，则考虑该选项。您可以选择不止一列。

模糊比较选项

忽略缺失值 忽略缺失数据。

允许相对误差 支持您指定数值数据的相对误差比率。若数值在您指定的相对误差率之内，则认为数值相等。相对误差比率越小，比较的精确度越高。

忽略大小写 比较文本时忽略大小写。

忽略空格 比较文本时忽略空白。

数据选项

显示模糊差异 显示大致相同的数值以及字符串数据中的差异。使用 “相对误差”字段中的值来删除不显著的差异。

隐藏没有差异的列 显示或隐藏所有匹配列。

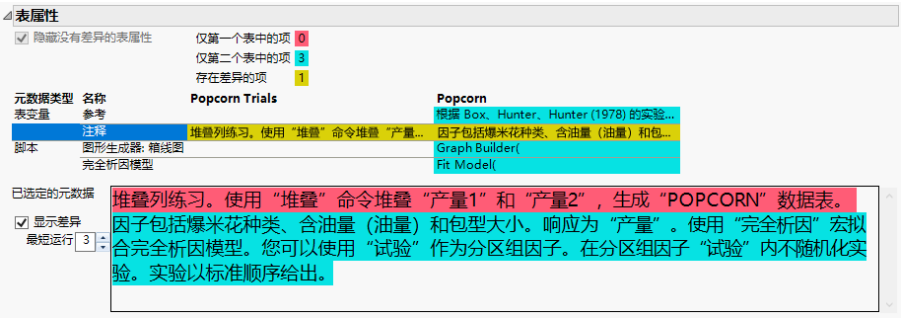
隐藏没有差异的行 显示或隐藏包含匹配数据的行。

比较表属性

点击红色小三角并选择**比较表属性**可查看表脚本和变量中的差异。图 4.27 显示表变量和脚本是不同的。要查看完整的变量或脚本，请选择行并查看选定的元数据。

在本例中，两个注释都不同，引用变量和脚本仅存在于 Popcorn.jmp 中。已选定“注释”变量，以便您可以在“选定元数据”框中的两个数据表中看到该变量的内容。红色着色指示该文本仅存在于 Popcorn.jmp 中。蓝色着色指示该文本仅存在于 Popcorn Trials.jmp 中。

图 4.27 不同表变量



取消选择**显示差异**可查看每个数据表的名称和每个“注释”变量的完整内容，而不是只显示差异。

最短运行是在将字符声明为**公共子段**之前两个文件之间必须相同的最小连续字符数。公共子段没有背景色，因为它在两个文件中都存在。最短运行设置为 3，以防止太短的子段显示为公共，这通常没有任何帮助。例如，最短运行为 1 意味着两个文件中的任何单个字符都可以匹配。这导致许多非常短的公共文本段和差异段，这通常并不是良好的阅读体验。

比较列特性和属性

点击红色小三角并选择**比较列特性和属性**可查看列注释、值颜色和类似项中的差异。

图 4.28 显示 Popcorn.jmp 和 Popcorn Trials.jmp 中的列注释有所不同。已选定 popcorn 列，以便您可以在“已选定的元数据”框中查看完整的注释以及两个注释之间的差异。

图 4.28 比较列特性和属性

为行和列指定特征

本节说明如何排除、隐藏、标记行和列，或为其添加标签或着色，以便定制散点图和其他图形中数据点的外观。

- “如何查找行列选项”

- “为行和列添加标签”
- “为行指定颜色或标记”
- “在图形中创建颜色主题”
- “删除图形中的定制颜色主题”
- “选择匹配行状态单元格”
- “删除行特征”
- “锁定列”

如何查找行列选项

可以从以下位置访问行操作菜单：

- 主菜单中的**行**菜单
- 右击某行
- 行面板中的红色小三角
- 数据网格左上角的左侧红色小三角

与此类似，可以从以下位置访问列操作菜单：

- 主菜单中的**列**菜单
- 右击某列
- 列面板中的红色小三角
- 数据网格左上角的右侧红色小三角

为行和列添加标签

将箭头光标置于图中某个点上方时，将显示该点的标签。默认情况下，行号即用作标签。您可以定制标签：

- 您可以将标签改为显示列值而不是行号。
- 您可以让标签始终显示，而不仅是在将光标置于数据点上方时才显示。
- 标签或黄色标签图标 ( ) 显示在列面板中的列名旁边，指示图中各点由列值来标识。若多列都具有标签，列值在图中显示时将以逗号分隔。
- 数据会一直保持加标签状态，直到您再次选择**添加标签/撤销标签**。

将标签改为显示列值

1. 突出显示列值要在图中显示为标签的一个或多个列。
2. 从菜单中选择**列 > 添加标签 / 撤销标签**，或右击并选择**添加标签 / 撤销标签**。

使标签始终显示 （而不仅是在将光标置于数据点上方时才显示）

1. 突出显示您希望在图中始终显示其标签的一个或多个行。
2. 从菜单中选择**行 > 添加标签 / 撤销标签**。

禁用行或列标签

1. 突出显示您不想再添加标签的已添加标签的行或列。
2. 从**行**菜单或**列**菜单中选择**添加标签 / 撤销标签**。您还可以右击列或行并选择**添加标签 / 撤销标签**。

提示：悬停在数据上方时，标签中会显示照片。例如，SAS Offices.jmp 样本数据表包含添加了标签的“表达式”列。该列存储了每个办公室的照片。当您悬停在地图中数据点的上方时，即显示该行的照片。请参见“设置 JMP 列属性”一章中第 282 页的“表达式角色”。

为行指定颜色或标记

为行指定颜色或标记可区分数据表中的行。

- 若为行指定某种颜色，表示该行中的值的数据点将在图中显示为指定的颜色。
- 若为行指定某个标记，数据点将在图中替换为该标记。
- 您还可以基于列值指定颜色或标记。

为行指定颜色

为选定行指定颜色意味着图中的点将以您选择的颜色显示。在数据网格中，指定给行的有效颜色显示在行号旁边。

为行指定颜色

1. 突出显示要指定颜色的一个或多个行。
2. 右击突出显示的行并选择**行 > 着色**。
3. 选择一种可用颜色。

提示：要为选定行清除指定的颜色，请将颜色指定为黑色。

为行添加标记

要用标记替换图中的标准点，请使用 JMP 标记选项板。在数据表中，这些标记还显示在行号旁边。

1. 突出显示要将标记应用到的一个或多个行。
2. 右击选定行并选择**标记**，然后选择标记形状。

选择**其他**可创建定制标记。您可以键入字母字符、数字和其他键盘符号。

提示：要恢复默认标记，请选择初始的点标记。

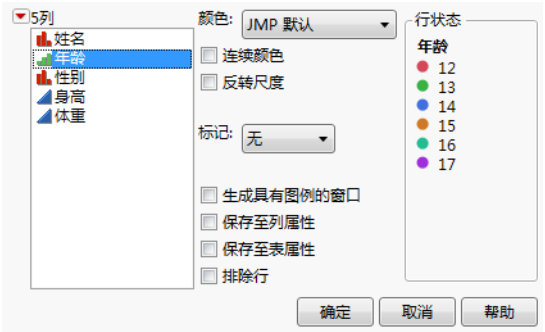
基于列值为行指定颜色或标记

您可以按照特定列中包含的值为数据表行指定颜色或标记。例如，在名为**性别**的列中，您可以为值为 F 的所有行指定红色圆圈标记。值为 M 的所有行可具有绿色加号标记。这些颜色和标记替换图中的默认黑点，并显示在数据表中其行号旁边。

基于列值为行指定颜色或标记

1. 选择**行 > 按列设定颜色或标记**。
2. 选择要设定颜色和 / 或标记的列。

图 4.29 按列设定颜色或标记



3. 选择要应用的**颜色**和**标记**方案。

   所做选择的预览显示在 “行状态”之下。
4. （可选）选择其他任何选项。请参见第 205 页的 “‘按列设定颜色或标记’选项”。
5. 点击**确定**。
6. （可选）要根据行状态对所有行着色，在数据网格的行号区域中右击，然后选择**按行状态为行着色**。

   之后，将使用为行指定的颜色对行着色。

“按列设定颜色或标记”选项

颜色 选择一种颜色主题，为数据表中的行指定不同颜色。颜色指定基于选定列的值。

连续尺度 基于突出显示列中的值依据色彩序列指定颜色。

反转尺度 基于突出显示列中的值依据反转的色彩序列指定颜色。

标记 基于您突出显示的列中包含的值为数据表中的每行指定不同标记。

生成具有图例的窗口 在新特征中加入图例，方便您辨认哪些颜色和标记对应哪些行。

保存至列属性 将颜色和标记信息另存为列属性。基于颜色主题对数据表的选定列中的行着色。

保存至表属性 将颜色和标记信息另存为表属性。

排除行 为排除的行指定颜色或标记。

在图形中创建颜色主题

JMP 包括若干颜色主题，用来区分图形中一定范围内的值。您还可以基于现有颜色主题自行创建颜色主题，或创建定制主题。

注意：选择默认颜色主题时，颜色不会应用到打开的报表。您需要重新运行现有报表，才能使用默认颜色主题设置这些报表的格式。

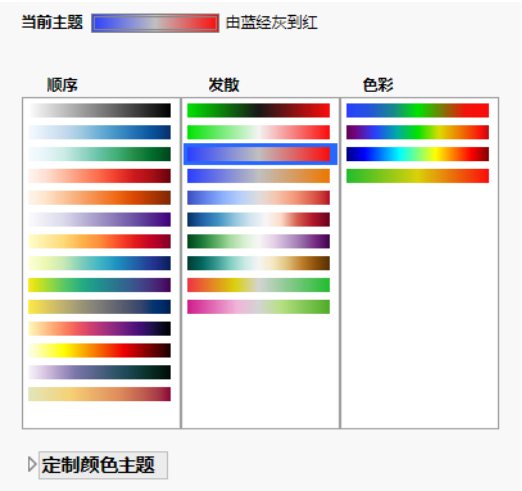
有关删除定制颜色主题的详细信息，请参见第 210 页的“删除图形中的定制颜色主题”。

创建颜色主题

1. 选择**文件 > 首选项 > 图形**。
2. 要创建新的“连续颜色主题”或“分类颜色主题”，请点击相应的颜色主题。

   若是创建新的连续颜色主题，“连续颜色主题”窗口随即显示。

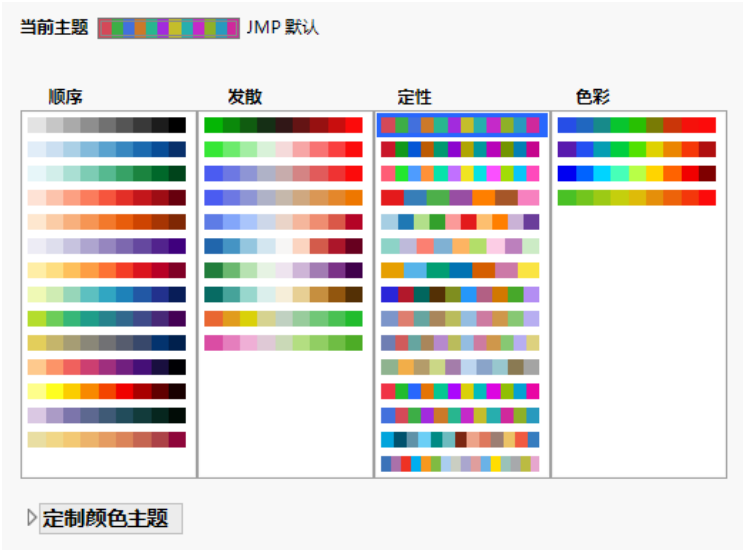
图 4.30 “连续颜色主题”窗口



顺序颜色“Cividis”是一种专为具有颜色限制的观看而设计的 Viridis 形式。

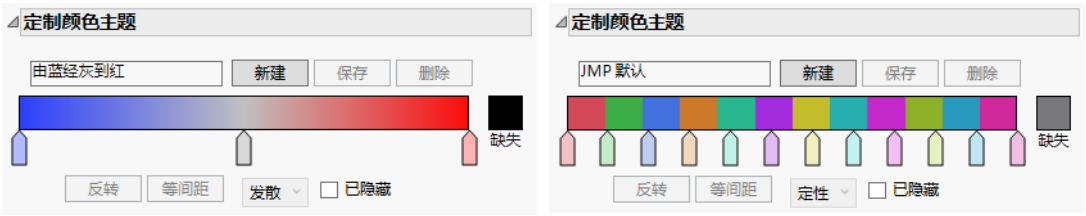
若已选择创建新的分类颜色主题，“分类颜色主题”窗口随即显示。

图 4.31 “分类颜色主题”窗口



3. （可选）要使主题基于现有主题，请从可用主题中选择一种颜色主题。
4. 点击 “定制颜色主题”展开按钮，显示 “定制颜色主题”面板。
   图 4.32 同时显示了分别适用于连续主题和分类主题的颜色主题面板。

图 4.32 “定制颜色主题”面板



5. 点击**新建**创建新主题。
   将基于选定的颜色主题创建新的颜色主题。同时为该主题指定临时名称。
6. 在临时标签位置键入新名称。在 Windows 上，不要按 Enter 键。按 Enter 键将关闭该窗口。
7. 要修改颜色主题，请执行以下任意操作：
   - 要修改缺失值颜色，请点击缺失值颜色以打开调色板。选择所需的缺失值颜色并点击**确定**。
   - 要修改连续颜色的梯度，请向左或向右移动滑块。
   - 要向梯度添加更多颜色，请点击颜色条选择一种颜色。新的滑块将显示在颜色条下方。
   - 要更改滑块的颜色，请点击该滑块以显示“颜色”窗口，从中选择另一种颜色。

- 要反转梯度上颜色的顺序，请点击**反转**。
- 要使颜色在梯度上平均分布，请点击**等间距**。
- 要在“顺序”窗格中列出定制主题，请从列表中选择**顺序**。
- 要在“发散”窗格中列出定制主题，请从列表中选择**发散**。
- 要在“色彩”窗格中列出定制主题，请从列表中选择**色彩**。
- 若不想在颜色主题列表中显示某个主题，请选择**隐藏**。
- 要从颜色主题中删除某种颜色，请点击该颜色的滑块，并将滑块拖到颜色条的上方或下方。
- 要放弃更改，请点击**取消**。

8. 点击**保存**保存定制颜色主题。

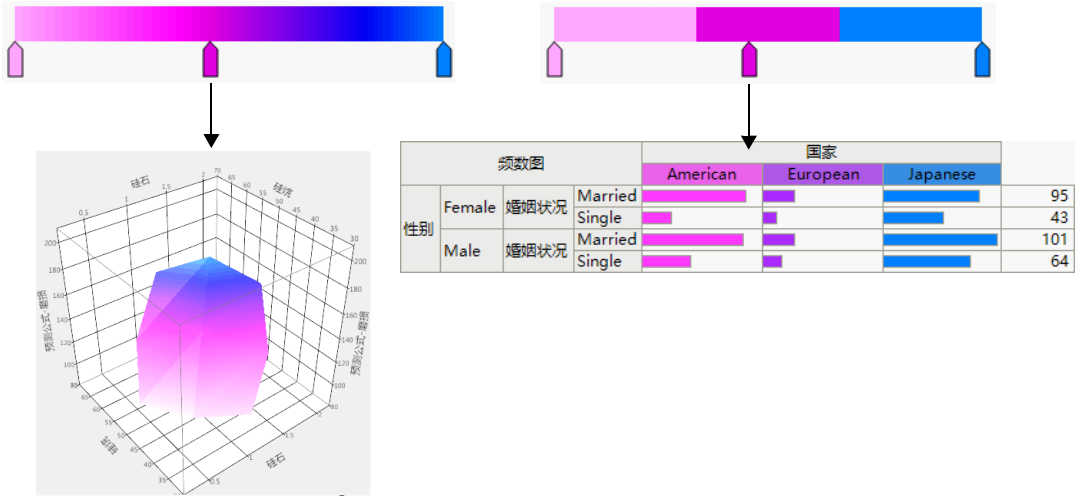
   新的定制颜色主题追加至选定窗格的内容。

9. 点击**确定**关闭颜色主题窗口。

连续和分类颜色主题

下图显示 JMP 中两种颜色主题（连续和分类）的示例。为连续数据选定颜色主题时，颜色是渐变色（如左图所示）。为分类数据选定同样的颜色主题时，颜色将由差别明显的色块构成。（如右图所示）。

图 4.33 连续和分类颜色主题的示例



定制颜色主题

定制颜色主题的应用方式与内置颜色主题相同：

- 您可以从“图形”首选项的**连续颜色主题**和**分类颜色主题**下拉菜单中，将定制颜色主题选作默认主题。连续数据只可使用连续颜色主题。所有颜色主题都可用于分类数据。
- 您可以将定制颜色主题应用到标记和数据表行之类的组件。请参见第 205 页的“基于列值为行指定颜色或标记”。
- 在某些报表中（如矩形树图和曲面图），您可以选择特定的定制颜色主题。请参见《基本绘图》。

在多台计算机上使用定制颜色主题

在 Windows 中，您创建的颜色主题在名为 JMP.PFS 的 JMP 首选项文件中定义。若在多台计算机上使用 JMP（例如，在家用电脑和办公用电脑上），则可以将颜色主题定义从一个 JMP 首选项文件复制到另一个文件。随后便可在两台计算机上使用定制颜色。

在首选项文件中，定制颜色主题的代码如下所示：

```
Add Color Theme(
{"Pink to Blue", 2051, {{255, 168, 255}, {255, 0, 255}, {0, 128, 255}}}
),
```

在本例中，颜色主题名为“Pink to Blue”。每个颜色滑块的红/绿/蓝 (RGB) 值都放在括号中。第一个滑块定义 RGB 值 255、168、255。第二组和第三组括号分别定义第二个和第三个滑块的颜色。

注意：第二个值 (2051) 是颜色主题的类别，例如“分类”或“连续”。您可以省略类别，但是 JMP 总是在首选项文件中或是在运行 `Show Preferences()` 时包含信息。

在文本编辑器（如 Microsoft 记事本）中，将该颜色主题添加至其他计算机上的首选项文件。首选项文件位于 Users 文件夹的 JMP 或 JMPPro 文件夹中。

C:/Users/<user_name>/AppData/Roaming/SAS/JMP/16

C:/Users/<user_name>/AppData/Roaming/SAS/JMPPro/16

注意：要查看之前的文件夹，您必须配置“Windows 资源管理器”显示隐藏的文件和文件夹。有关详细信息，请参考 Windows 帮助。

将颜色主题传输至其他 Windows 计算机

1. 在包含定制 JMP 首选项的计算机上，选择**文件 > 新建 > 脚本**。
   随即显示“脚本”窗口。
2. 键入以下 JSL 函数：
   ```
   Show Preferences()
   ```
3. 点击**运行脚本** 。
   定制的首选项将写入日志。

4. 选择**视图 > 日志**（或显示打开的日志）。

   随即显示您创建的定制颜色主题，例如：

   ```
   Add Color Theme(
   {"Pink to Blue", 2051, {{255, 168, 255}, {255, 0, 255}, {0, 128, 255}}}
   ),
   ```

   该定义可能位于在日志中显示的其他定制首选项的中间。

5. 将日志另存为 Log.jsl，并在要更新首选项的计算机上打开该文件。
6. 在要更新首选项的计算机上，关闭 JMP。
7. 制作 JMP.PFS 的备份副本，然后在文本编辑器中打开原始的 JMP.PFS。
8. 将定制颜色定义从 Log.jsl 复制并粘贴到 JMP.PFS。该定义跟在 `Preferences(` 之后，如下例所示：

   ```
   Preferences(
   Add Color Theme(
   {"Pink to Blue", 2051, {{255, 168, 255}, {255, 0, 255}, {0, 128, 255}}}
   ),
   );
   ```

   注意：务必加入结束圆括号和逗号。无需缩进代码。您可以将代码置于任何有效的位置。将代码粘贴到 `Preferences(` 之后可帮助避免删除任何必要的圆括号或逗号。

9. 保存该文件。

   若打开 JMP 后发现新的颜色定义未显示在首选项中，请删除更新的首选项文件，并将该定义文件添加至原始的首选项文件。确保将定义复制并粘贴到正确的位置。

删除图形中的定制颜色主题

1. 选择**文件 > 首选项 > 图形**。
2. 要删除颜色主题，请选择 “连续颜色主题”或 “分类颜色主题”。

   随即显示相应的 “颜色主题”窗口。

3. 点击 “定制颜色主题”展开按钮，显示 “定制颜色主题”面板。
4. 从相应的窗格中，选择要删除的定制颜色。

   注意：您只能删除定制颜色主题。

5. 点击**删除**。
6. 点击**确定**保存所做更改并关闭 “颜色主题”窗口。

选择匹配行状态单元格

您可以选择行状态列中的某个单元格，然后选择包含选定单元格的特性的所有单元格。该选项支持您选择包含行状态特性（如：排除、隐藏、颜色或标记）的任意组合的单元格。例如，若某个单元格有红色标记且已排除，在您选择匹配单元格时，带红色标记的排除行中的所有单元格都将被选定。

1. 右击您要选择其行状态的单元格。
2. 选择**选择匹配单元格**。

   具有匹配行状态的单元格将被选定。

删除行特征

要清除数据表中的所有行状态，请选择**行 > 清除行状态**。要仅清除选定行中的行状态，请选择**行 > 清除选定行状态**。

所有行变为已包括、可见、未添加标签的行，并在图中显示为黑点。**清除行状态**命令不影响保存在行状态列中的行状态。

锁定列

您可以将列锁在原位，这样在您水平滚动时，该列仍保持可见。突出显示要锁定的列，然后选择**列 > 滚动锁定/撤销锁定**。请注意以下几点：

- 无法滚动锁定隐藏的列。
- 锁定的列的名称在列面板中以斜体显示。
- 滚动锁定的列移至数据网格的左侧。一旦撤销锁定，这些列不会移回它们在数据表中的原始位置，而是保留在左侧。
- 列保持滚动锁定，直到您将其突出显示并再次选择**滚动锁定/撤销锁定**。

重构数据

JMP 提供了几种重构数据表中数据的方法，例如从多个列创建列、创建指示符列和重新编码数据。

- “将数据表列拆分为多列”
- “通过文本匹配创建新列”
- “生成指示符列”
- “组合数据表中的列”
- “压缩数据表中的选定列”
- “在数据表列中生成分箱公式”

- “将标签转换为代码 / 将代码转换为标签”
- “生成新公式列”
- “变换 JMP 平台中的列”
- “对列中数据重新编码”

将数据表列拆分为多列

使用**生成文本列**选项可将具有分隔字段的字符列处理成多列。突出显示数据表中的一列并选择**列 > 实用工具 > 生成文本列**。所有行中分隔字段的最大数目决定了创建的新列数。

注意：生成文本列区分大小写。

生成文本列窗口具有以下选项：

分隔符 指定文本 （如逗号），以指示如何在新列中组织源列中的数据。例如，若原始单元格读取为 “NY, NJ, PA” 且分隔符为逗号，则三个新列将包含 “NY”、“NJ” 和 “PA”。

TAB 将制表符指定为分隔符。

NEWLINE 将换行符 （\r、 \n 和 \r\n）指定为分隔符。

生成指示符列 生成以源列中非重复字段命名的新列，其单元格值为 0 或 1。

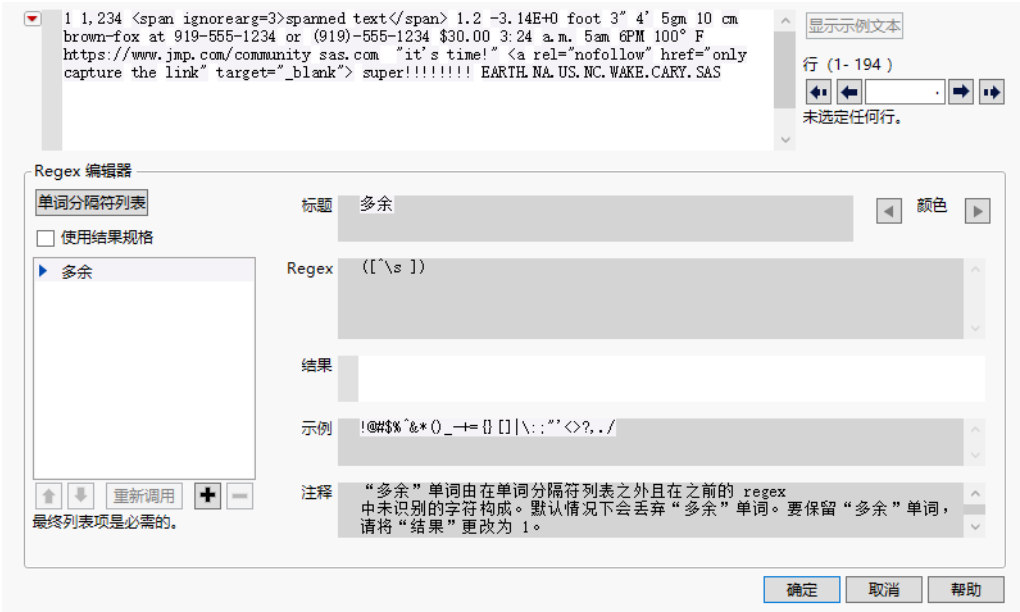
包括缺失值 允许将任意空行计为一个类别。一个名为**缺失值**的额外列被添加至数据表。值 1 指示一个空行。

通过文本匹配创建新列

“通过文本匹配新建列”选项允许您使用正则表达式（用于标识文本中的模式）匹配列中的非结构化文本，然后从匹配的文本创建新列。该功能是使用 “文本资源管理器” 平台编写正则表达式之后将结果保存到列的一种替代方法。

1. 选择**帮助 > 样本数据库**，然后打开 Pet Survey.jmp。
2. 选择**调查回答**列。
3. 选择**列 > 实用工具 > 通过文本匹配新建列**。

图 4.34 正则表达式编辑器



4. 在 “多余” 框下，点击 “添加” 按钮 ✚。
5. 从列表中选择**空**并点击**确定**。
6. 在 “标题” 旁边，输入 Cat。
7. 在 “正则表达式” 旁边，输入 (cat(?:s?))([\s\.])。

   该表达式可以找到 “cat”，也可以后跟 “s”，后面有空格或句点。

   注意：若正则表达式有错误，则 Cat 的左侧会显示一个 X 图标 ✘。

8. 再次点击 “添加” 按钮 ✚。
9. 从列表中选择**空**并点击**确定**。
10. 在 “标题” 旁边，输入 Dog。
11. 在 “正则表达式” 旁边，输入 (dog(?:s?))([\s\.])。
12. （可选）要保存脚本，以便在关闭并重新打开 “正则表达式编辑器” 后不必重新填充它，请从红色小三角菜单中选择一个 “保存脚本” 选项。
13. 点击**确定**。

图 4.35 包含新列的数据表

	调查回答	调查回答 2	调查回答 3
1	The tabby cat d...	cat cat	
2	In the winter I ...		dog
3	My dogs are no...		dogs
4	The dog chases...		dog
5	My dog barks a...		dog
6	My dogs are so ...		dogs dog
7	I have two dogs...		dogs
8	My cat slid back...	cat	
9	The dogs bark li...		dogs

通过“正则表达式编辑器”中的“使用结果规格”选项，可以控制新列中出现的匹配文本：

- 若不选择“使用结果规格”（默认设置），则新列中的结果是编辑器右侧的正则表达式所生成的任何结果。
- 若选择“使用结果规格”（默认设置），则新列中的结果是编辑器右侧的“结果”表达式所生成的任何结果。若正则表达式为 Air(port)，则在原始列中找到 port 一词时，新列将包含该词。

有关“正则表达式编辑器”的详细信息，请参见《基本分析》。

要了解有关正则表达式的更多信息，请访问 Regular-Expressions.info。

生成指示符列

基于每一个非重复类别将一个分类列转换为多个指示符列。这些列按照列中的每个水平来命名。例如，基于 Big Class.jmp 中的**性别**生成指示符列时，指示符列将按照 F 和 M 来命名。

突出显示数据表中的一列并选择**列 > 实用工具 > 生成指示符列**。您随后可以指明是否要在指示符列名之前附加原始列名（如 sex_F 和 sex_M 所示）并包括缺失值。

创建多个值为 0 或 1 的列。值为 1 指示原始列包含特定类别。

若给定列具有“多重响应”建模类型或“多重响应”列属性，则根据这组响应来确定类别。

组合数据表中的列

组合列选项与生成文本列相反。您可以用分隔字段将一组列组合为一个字符列，而不是生成多个列。

要组合指示符列，请执行以下步骤：

1. 选择**帮助 > 样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**起床后使用牙线**、**饭后使用牙线**和**睡前使用牙线**列。
3. 选择**列 > 实用工具 > 组合列**。
4. 键入 “组合牙线”作为列名，并将默认分隔符保留为逗号。

5. 选择**选定列是指示符列**并点击**确定**。

图 4.36 “组合牙线”列

组合牙线	起床后使用牙线	饭后使用牙线	睡前使用牙线
起床后使用牙线	1	0	0
饭后使用牙线	0	1	0
饭后使用牙线	0	1	0
饭后使用牙线	0	1	0
饭后使用牙线	0	1	0
睡前使用牙线	0	0	1
饭后使用牙线	0	1	0
睡前使用牙线	0	0	1
	0	0	0
饭后使用牙线	0	1	0
睡前使用牙线	0	0	1
饭后使用牙线,睡前使用牙线	0	1	1
起床后使用牙线,睡前使用牙线	1	0	1
起床后使用牙线,饭后使用牙线,睡前使用牙线	1	1	1
起床后使用牙线,睡前使用牙线	1	0	1
起床后使用牙线,饭后使用牙线	1	1	0

选定的列在**组合牙线**列中表示，每个字段用逗号分隔。仅值为 1 的列表示在每个给定行的组合列中。

注意：值标签在数据表中显示标签而不是值。为组合列中的值的每个实例都显示一个标签。您可以通过双击单元格中的标签显示原始值。要避免使用值标签，请在组合列时选择**无值标签**。

压缩数据表中的选定列

JMP 支持您压缩数据表中的列，以便尽量减小文件大小，从而减少分析数据所需的内存空间。在数值列包含许多小的整数或是有任何列包含不足 255 个唯一值的情况下，该功能很有帮助。例如，压缩具有 389 列和 85,000 行的数据表中的列可能将文件大小从 250MB 缩小至 33MB（具体取决于数据类型）。

压缩列时，JMP 会根据数据类型来验证是否能将数据存储为更紧凑的格式：

- 在唯一值不足 255 个的字符列中，在适用时会向列添加“列表检查”属性（图 4.37）。选择“允许 16 位列表检查压缩”首选项时，“列表检查”属性也会添加到含有 255 个以上唯一值的字符列中。

  “列表检查”属性会将选定列中的值限制为有效值。当选定列中的值个数太大时，不会应用“列表检查”属性。例如，若值个数几乎与行数相同时，则数据表不会将“列表检查”属性添加至列。

- 对于数值列，仅压缩“最佳”、“固定小数位数”或“数据”格式的列。数据都尽可能压缩为单字节、双字节或 4 字节的整数（图 4.38）。有关短整数的详细信息，请参见“设置 JMP 列属性”一章中第 256 页的“短整型格式”。

若具有非整数值的数值列中的唯一值不足 255 个，也可以对这样的列进行压缩。在这种情况下，将向列添加 “列表检查” 属性。

注意：

- 要自动压缩唯一值小于或等于 65,535 的列，请在 “常规” 组中选择**允许 16 位列表检查压缩**首选项。“列表检查” 属性也会添加到该列中。
- 在具有 “列表检查” 属性的列中，您只能输入列表中的值。若不这样做，JMP 会在您试图输入新值时警告单元格包含无效数据。请参见第 265 页的 “列表检查”。

图 4.37 添加至已压缩字符列的 “列表检查” 属性

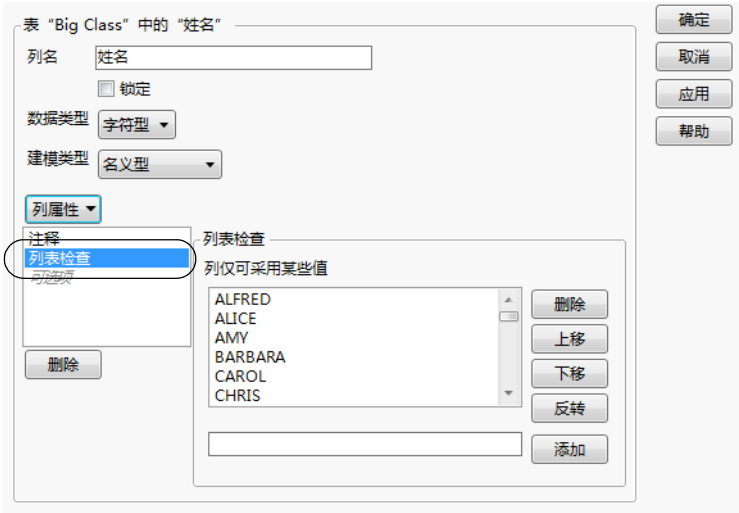


图 4.38 显示压缩前后数值列的 “列信息” 窗口

要压缩列，请选择一个或多个列，然后选择**列 > 实用工具 > 压缩选定列**。（若不知道可以压缩哪些列，则可以选择所有列。）

将尽可能压缩一列或多列。日志会显示已压缩哪些列以及压缩这些列的方式。（选择**视图 > 日志**可显示日志。）

注意：要手动压缩某个数值列，请将 “表” 首选项设置为允许短型数值数据，然后将该列的数据类型改为**单字节整数**、**双字节整数**或 **4 字节整数**。有关该首选项的详细信息，请参见 “JMP 首选项” 一章中第 594 页的 “数据表的首选项”。

在数据表列中生成分箱公式

您可以使用“生成分箱公式”选项将数据分布到箱中。选择要划分到多个箱中的一个或多个列，然后选择**列 > 实用工具 > 生成分箱公式**。将显示箱的直方图和“值标签”图例，以及包含用于定义割点的选项的红色小三角菜单。有几种方法可以添加割点、删除割点和调整箱大小。

- 在直方图中右击可添加或删除割点。
- 点击并拖动直方图中的垂直线条可调整箱大小。
- 编辑割点值标签旁边的框可编辑切割点值并调整箱大小。
- 使用“值标签”图例上方的添加 和删除 按钮添加和删除行，这等同于添加和删除割点。
- 使用撤销 和恢复 按钮可撤销或恢复针对箱设置执行的操作。

有一个选项可以将割点的公式列添加到数据表中。

用于生成分箱公式的右击选项

在“生成分箱公式”窗口中右击直方图以选择以下选项：

添加割点 将割点线添加到位于所点击位置的直方图中。

删除割点 删除割点线。仅当右击现有割点后，该选项才可用。

显示计数 显示或隐藏直条上方的计数。

显示百分比 显示或隐藏直条上方的百分比。

子集 基于当前选择创建子集数据表。

填充模式 指定直方图中直条的填充模式。

用于割点的红色小三角选项

“割点”红色小三角包含以下选项：

使用等宽箱填充 指定开始值（剩余偏差）和箱宽。JMP 将填充该表，直到它超过可能的最大值。例如，若输入“剩余偏差 =50”和“宽度 =10”，并且最大数据值为 95，则将得到以下箱：50、60、70、80、90、100。

使用箱计数填充 指定开始值、结束值和箱计数。JMP 用箱计数等分开始值和结束值之间的间隔。若更改箱计数，则报告的箱宽将在窗口中自动更新。

使用百分位数填充 指定 1 到 50 之间的百分比。JMP 用该分位数除以数据范围。例如，若将百分位数指定为 20，则会得到五个等权重箱。

使用均值 / 标准差填充 指定要设置在数据均值加上均值的第一、第二或第三标准差（基于您在窗口中的选择）的箱割点。

使用 Jenks 自然断点填充 指定箱计数并创建箱割点，以使每个箱内的方差最小化，使各箱之间的方差最大化。

箱标签 指定是否用值标签代替数据值进行显示。

使用值标签 在定义箱的新公式列中显示标签而不是值。

使用范围标签 在标签中包含每个范围的下限值和上限值。

无标签 将下边缘值用作标签。

字符 应用 “字符”数据类型。公式将生成标签。

请参见 “设置 JMP 列属性”一章中第 266 页的 “值标签”。

提示：大多数平台都建议使用值标签，其中很多平台都不支持范围标签。在 “分类”平台中，您必须使用值标签。在某些轴上，您可能会发现范围标签可以更清晰地标识值，或者您可以尝试调整轴的偏移和宽度。

箱标签样式 指定一个预设值，用于定义 “值标签”或 “字符”输出在表中的外观。

低–高 通过每个箱的最低点和最高点定义“值标签”。格式为“低–高”。

低至高 通过每个箱的最低点和最高点定义“值标签”。格式为“低至高”。

低, 高 通过每个箱的最低点和最高点定义“值标签”。格式为“低, 高”。

低–高-1（整数） 通过每个箱的最低点和最高点减 1 来定义“值标签”。格式为“低–高-1”。

低至高-1（整数） 通过每个箱的最低点和最高点减 1 来定义“值标签”。格式为“低至高-1”。

[低, 高)（数学） 指定范围中包含下限值，而不包含上限值。例如，若您的范围为 [50,60)，则 50 这个值将包含在箱内，而 60 这个值将不含在箱内。

下限 通过每个箱的最低点定义“值标签”。

中点值 通过每个箱的中点定义“值标签”。

整数序列 通过一个连续整数序列定义“值标签”。在“整数序列标签”窗口中定义起始整数。

大写字母 按字母顺序将“值标签”定义为大写字母。

小写字母 按字母顺序将“值标签”定义为小写字母。

定制 支持您输入定制“值标签”。

列格式 指定 “值标签”的格式。默认情况下，应用列的格式。不过，您可能希望覆盖该设置（例如，将格式更改为 “美元”）。若列格式类似于 “最佳”之类的格式，但您确实希望您的箱不带小数，则该设置最为有用。您只需指定格式，而不必清除所有值标签。

直方图 指定计数或百分比是否显示在直条上方。

显示点 显示或隐藏直条下方的观测数据点。

重新调整轴 将坐标轴重置为原始尺度。

重新调用 使用您最近执行的分箱操作填充 “生成分箱公式”窗口。

使所有列类似于 X （仅当选定多列时才显示。）将对第一列 (X) 所做的选择应用到对话框中的其余列。

生成公式列 创建公式列并关闭该窗口。

提示：一旦创建公式列，您就可以通过右击列标题并选择**编辑分箱公式**来编辑该公式。这将重新打开 “生成分箱公式”窗口。要更新分箱公式列，请编辑割点并点击**更新列。**

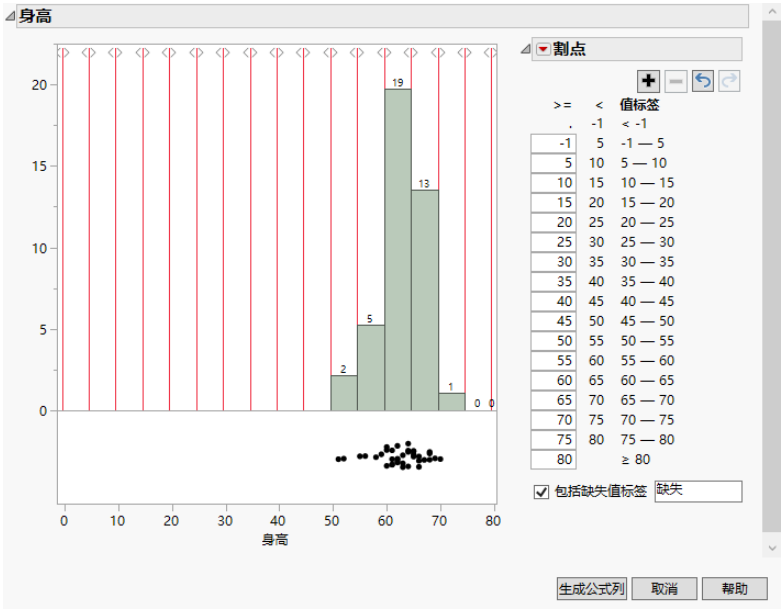
关于生成分箱公式的示例

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**身高**列。
3. 选择**列 > 实用工具 > 生成分箱公式**。

   您希望值范围显示为 X-X，所以请在 “箱标签样式”中将范围设置为**低 - 高**。
4. 点击 “割点”红色小三角并选择**使用等宽箱填充**。
5. 在出现的窗口中，将**剩余偏差**改为 -0.5。

   提示：对于整数数据，将剩余偏差设置为 -0.5 有助于区分边缘处的值。在该示例中，一个箱的取值范围介于 59.5 到 64.5 之间，所以很明显 59 和 65 都不包含在该箱内。
6. 保持**宽度**设置为 5。
7. 点击**确定**。
8. 保留 “箱标签”设置为**使用值标签**，这样您就可以看到该箱的值范围。

图 4.39 已完成的分箱窗口

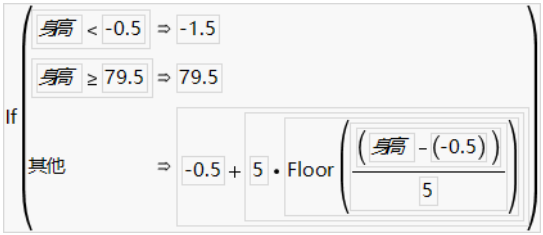


9. 点击**生成公式列**。

   名为 **“身高”分箱**的列添加至 Big Class.jmp 数据表。

10. 要查看如何计算公式，请右击 **“身高”分箱**列并选择**公式**。

图 4.40 公式



将标签转换为代码 / 将代码转换为标签

假定您有包含诸如“不满意”、“有些不满意”、“有些满意”和“满意”之类的字符串的列。您可以将这些值重新编码为“1”、“2”、“3”和“4”。原始字符串放置在该列的“值标签”列属性中。若该列具有“列表检查”属性，该属性将被删除。列数据类型改为单字节整数。将字符串列转换为短数值代码可节省内存，加快处理速度。

您还可以将数值代码列转换为标签。该列必须具有值标签。列数据类型改为“字符”，“值标签”列属性将被删除。这实际上就是将标签转换为代码的相反操作。

将标签转换为代码

1. 选择字符列，然后选择**列** > **实用工具** > **标签转换为代码**。

图 4.41 将标签转换为代码

值分配有默认代码。

2. 按需更改这些代码并点击**确定**。

该列将被重新编码，值标签添加到该列。

将代码转换为标签

1. 选择包含值标签的数值列，然后选择**列** > **实用工具** > **代码转换为标签**。
2. 按需更改这些标签并点击**确定**。

该列将使用原始值标签重新编码。

生成新公式列

要对数据执行进一步的分析，可以使用您现有数据表中的“新建公式列”菜单选项。公式列使用公式或计算来定义列值。

右击数据表中的某个列标题，并选择**新建公式列**。从“变换”、“字符”、“合并”、“配对”、“聚合”、“分布”、“日期时间”、“随机”、“行”或“向量”中进行选择，以计算列值。一个新的公式列将添加至数据表。有关这些选项的说明，请参见第 222 页的“变换 JMP 平台中的列”。

注意：在“新建公式列”菜单和启动窗口的右击列菜单中存在相同的选项。但是，在启动窗口中执行这些任务将会产生临时列，而“新建公式列”将会向原始数据表中添加一个新列。

右击菜单中的选项将取决于选定列的数据类型和选定的列数。若选定列是“字符”列，则会显示字符和行选项。请参见第 228 页的“‘字符’菜单”和第 228 页的“‘行’菜单”。

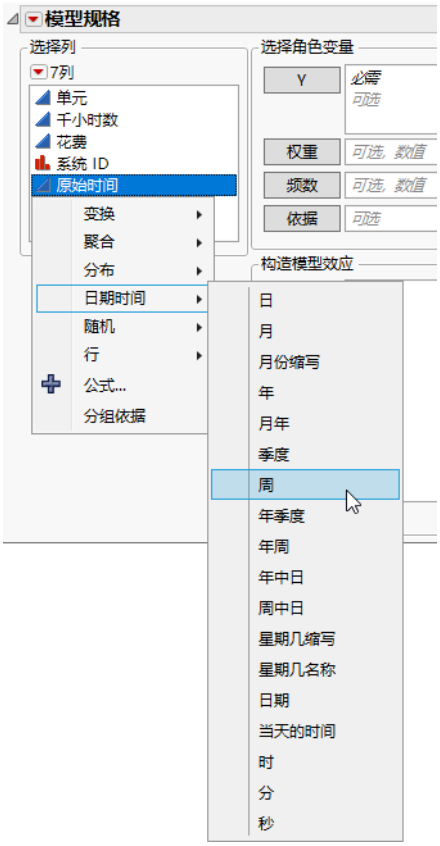
变换 JMP 平台中的列

JMP 中的每个启动窗口都允许您创建一个或多个用于执行分析的临时变换列。这些变换列不是源数据表的组成部分，只能用在当前启动窗口的上下文中。变换列使用公式或计算来定义列值。关闭启动窗口将删除所有变换列。

启动窗口的“选择列”窗格中列出的每一列都包含一个图标（表示该列的建模类型，即：连续型、有序型、名义型）和列名。右击某一列名可使用“变换”、“字符”、“合并”、“配对”、“聚合”、“分布”、“日期时间”、“随机”、“行”、“向量”或“公式”创建一个变换列来计算该列值。

右击选项取决于选定列的数据类型和选定的列数。

图 4.42 “变换列”菜单的示例



依据 ... 分组 对于有序型和名义型数据，指定用于分组数据的列。会为指定列的每个水平计算单独的分析。

注意：

- 变换列仅可用于当前启动窗口。要使变换列可在当前启动窗口外部使用，请右击该变换列并选择**添加至数据表**。变换列将添加至源数据表。
- 您可以将变换列粘贴到启动窗口的“角色”框中。例如，您可以从脚本复制变换列。在相应的启动窗口中右击“为选定列指定角色”框并选择“粘贴”。该方法等价于右击“选择列”列表中的列，选择变换并将变换列添加到某个角色。
- 若变换函数是可逆的，则以下平台将使用预测刻画器的原始尺度并保存预测值和公式。
  - 以 X 拟合 Y
  - 标准最小二乘法
  - 分割
  - 提升树
  - Bootstrap 森林法
- 通过编写 JSL 脚本自行定义变换。请参见《Scripting Guide》。

“变换”菜单

从“变换”菜单中选择一个函数，创建包含基于选定函数计算的变换列。请参见《拟合线性模型》。

注意：您可以通过对多列应用一元函数来生成多个变换列。

表 4.2 “变换”菜单选项说明

舍入	对日期值进行舍入。例如，在星期图表中，“舍入”可将日期值转换为这些日期值在每个星期中出现的第一个日期。
尺度偏移	支持您指定一般线性变换。例如，您可以通过将该值乘以 1.8 且应用偏移量 32 将摄氏温度转换为华氏温度。
定制分箱	打开分箱窗口，该窗口支持您将数据分发到各箱中。
平方根	计算选定列的值的平方根。
平方	计算选定列值的平方值。
对数	对选定列应用自然对数变换。
Log x+1	计算 Log(列+1)。
指数	对选定列应用指数变换。
Log10	对选定列应用以 10 为底的对数变换。

表 4.2 “变换”菜单选项说明 (续)

p 的 10 次方	计算 10 的选定列值次幂。
立方根	计算选定列值的立方根值。
立方	计算选定列值的立方值。
倒数	计算选定列值的倒数（1/列）。
绝对值	计算选定列值的绝对值。
求反	计算选定列值的负数。
Arrhenius	对变量 T（摄氏度温度）应用 Arrhenius 变换： $X = \frac{11604.5181215503}{T + 273.15}$ 这是与活化能相乘的 Arrhenius 关系的成分。
Arrhenius 逆	对变量 X 应用 Arrhenius 变换的逆变换： $T = \frac{11604.5181215503}{X} - 273.15$
Logit	对选定列计算 Logistic 函数的反函数（其中，p 介于 0 到 1 之间）： $\text{Logit}(p) = \log\left(\frac{p}{1-p}\right)$
Logistic	计算选定列的 Logistic（亦称 Squish 和 Logist）函数（其中，结果介于 0 到 1 之间）： $\text{Logistic}(x) = \frac{1}{(1+e^{-x})}$
Logit 百分比	以百分比形式计算选定列的 Logit（其中，pct 是介于 0 到 100 之间的百分比）： $\text{LogitPct}(pct) = \log\left(\frac{\left(\frac{pct}{100}\right)}{1-\left(\frac{pct}{100}\right)}\right)$

表 4.2 “变换”菜单选项说明 (续)

Logistic 百分比	以百分比形式计算选定列的 Logistic（或 Logist，其中，结果介于 0 到 100 之间）： $\text{LogisticPct}(x) = \frac{100}{(1+e^{-x})}$

“合并”菜单

选择多个列以访问合并菜单。“合并”菜单用于创建包含基于选定函数计算的变换列。

菜单中包含以下函数：

总和 计算第一列与第二列之和 (A + B)。

差值 计算第一列和第二列之间的差值 (A - B)。

差值 （倒序） 计算第二列和第一列之间的差值 (B - A)。

乘积 计算第一列与第二列的乘积 (A X B)。

比率 计算第一列与第二列之比 (A / B)。

比率 （倒序） 计算第二列与第一列之比 (B / A)。

最小值 返回选定列的最小值。

最大值 返回选定列的最大值。

平均值 返回选定多列的平均值。

标准差 计算选定列中的值的标准差。

中位数 计算选定列的中位数值。

分位数 计算选定列指定百分比的分位数。

几何均值 返回数据的积的第 n 个根。

“配对”菜单

从“配对”菜单中选择一个函数以创建两个或更多列的配对结果。仅当选定列数为偶数并且大于等于 4 时，该菜单才显示。

总和 计算第一列与第二列之和 (A + B)。

差值 计算第一列和第二列之间的差值 (A - B)。

差值 （倒序） 计算第二列和第一列之间的差值 (B - A)。

乘积 计算第一列与第二列的乘积 (A X B)。

比率 计算第一列与第二列之比 (A / B)。

比率 （倒序） 计算第二列与第一列之比 (B / A)。

最小值 返回选定列的最小值。

最大值 返回选定列的最大值。

平均值 返回选定多列的平均值。

几何均值 返回数据的积的第 *n* 个根。

“聚合”菜单

从“聚合”菜单中选择一个函数，创建包含基于选定列（指定“分组依据”列时为列的某个部分）计算的统计量的变换列。

注意：分组依据选项对这些函数很有用。

菜单中包含以下函数：

均值 返回选定列的平均值。

总和 计算选定列中的值的总和。

计数 计算选定列中的值个数。

中位数 计算选定列的中位数值。

分位数 计算选定列指定百分比的分位数。

最小值 返回选定列的最小值。

最大值 返回选定列的最大值。

标准差 计算选定列中的值的标准差。

“分布”菜单

从“分布”菜单中选择一个函数，创建包含基于选定列计算的统计量的变换列。请参见“帮助”菜单中的“脚本索引”。

菜单中包含以下函数：

中心化 从选定列的所有行的每个值中减去列均值。

标准化 计算列值减去均值 （除以选定列的所有行的标准差后的均值）后的结果。

范围 0 至 1 将数据按比例增大或减小，使得最小值大于等于 0，且最大值小于等于 1。

Box-Cox 使用 Box-Cox 方程变换数据。请参见 《拟合线性模型》。

Johnson 正规化 使用一个 Johnson 方程变换数据。新列名指示 “Johnson Su” 、“Johnson Sb”或 “无”，这取决于用于计算新数据的方程。

信息性缺失 创建两列。**信息性**列使用列均值替换缺失值。**缺失**列使用 1 指示缺失值，使用 0 指示其他值。

秩 返回秩，最低为 1，行顺序可打破并列关系。

秩 （倒序） 返回秩，最高为 1，行顺序可打破并列关系。

累积概率 计算累积概率：Col Rank(col) / (Col Number(col) +1)。

正态分位数 计算正态分布的分位数。

SHASH 计算 sinh-arcsinh (SHASH) 分布在 **x** 处计算的累积分布函数。

“随机”菜单

对于数值列，从“随机”菜单中选择一个函数以创建包含随机值的列。

菜单中包含以下函数：

随机均匀 生成均匀介于 0 到 1 之间的随机数。

随机正态 生成近似均值为 0、标准差为 1 的正态分布的随机数（若未使用参数），否则生成您输入的参数作为均值和标准差的近似正态分布的随机数。

无放回抽样 每次求值时都随机排列这些值。第一个值的结果影响第二个值的结果。

有放回抽样 每次求值时都随机排列这些值。第一个值的结果不影响第二个值的结果。

“日期时间”菜单

对于包含日期或时间值的列值，请从“日期时间”菜单中选择一个函数，以创建包含基于选定列计算的值的变换列。

菜单中包含以下函数：

日 为选定列中的日期返回月份中的日。

月 为选定列中的日期返回月编号。

月份缩写 为选定列中的日期返回缩写月份。

年 为选定列中的日期返回年份。

月年 为选定列中的日期返回月份编号和年份。

季度 为选定列中的日期返回年中的季度 （1、 2、 3 或 4）。

周 为选定列中的日期返回周在年中的编号。

年季度 为选定列中的日期返回年和年中的季度 （1、 2、 3 或 4）。

年周 返回表示年中 ISO-8601 周格式的字符串（例如，将为 June 12, 2013 返回“2013W24”）。

年中日 为选定列中的日期返回年中的日。

周中日 为选定列中的日期返回星期几。

星期几缩写 为选定列中的日期返回星期几的缩写形式。

星期几名称 为选定列中的日期返回完整的星期几。

日期 为选定列中的日期返回月、日、年。

当天的时间 为选定列中的日期返回时间。

时 返回选定列中的日期的小时部分。

分 返回选定列中的日期的分钟部分。

秒 返回选定列中的日期的秒部分。

“字符”菜单

从“字符”菜单中选择一个函数，创建包含由所选“字符”函数形成的字符串的变换列。

菜单中包含以下函数：

长度 计算选定的一个或多个列中每个字符串的字符数。

拼接 将选定的一个或多个列中的字符串拼接为新字符串。

用空格拼接 将选定的一个或多个列中的字符串拼接为一个新字符串，其中用空格字符分隔各个子字符串。

用逗号拼接 将选定的一个或多个列中的字符串拼接为一个新字符串，其中用逗号字符分隔各个子字符串。

首字母大写 将字符串转换为标题大小写 （例如，“Sheila Smith”）。

大写 将字符串转换为大写 （例如，“SHEILA SMITH”）。

小写 将字符串转换为小写 （例如，“sheila smith”）。

第一个单词 从选定的一个或多个列的字符串中提取第一个单词。

最后一个单词 从选定的一个或多个列的字符串中提取最后一个单词。

秩 返回秩，最低为 1，行顺序可打破并列关系。

秩 （倒序） 返回秩，最高为 1，行顺序可打破并列关系。

“行”菜单

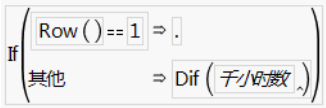
从“行”菜单中选择一个函数，创建包含由所选“行”函数确定的计算的变换列。

除了附录中说明的函数，以下函数也包含在菜单中：

行 返回当前行号。

已选定 返回选定的索引。

差值 使用下面的公式计算选定列中每个值的差值：

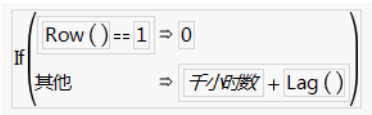


注意：“差值”函数还支持 “分组依据”选项。

滞后 为选定列返回上一行中的值。

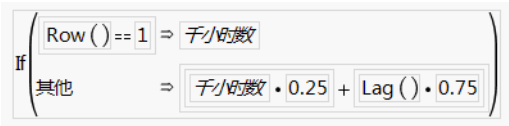
多重滞后 为选定列返回之前多行中的值。

累积和 使用下面的公式计算选定列中每个值的累积和：



注意：“累积和”函数还支持 “分组依据”选项。

移动平均值 计算选定列中每个值的指数加权移动平均值 (EWMA) （使用介于 0 到 1.0 之间的平滑参数）。以下示例使用的平滑参数是 0.25：



权重 确定如何对值加权。递增权重是一个斜坡或三角。指数移动平均值是 EWMA 或 EMA。

之前的项 通过在平均值中包括当前项之前的指定项数（在当前项之外），控制范围（或窗口）的大小。-1 表示之前的所有项。

之后的项 通过在平均值中包括当前项之后的指定项数（在当前项之外），控制范围（或窗口）的大小。-1 表示随后所有项。

为不完全窗口报告缺失值 控制如何处理缺失值。默认情况下，忽略缺失值。

注意：JMP 会计算按需输入的公式；复杂公式可能需要较长的处理时间。

“向量”菜单

从“向量”菜单中选择一种变换，以便从向量数据创建变换列。

目标列 将向量展开到各个数字列。

总和 返回向量中参数或值的总和。

最小值 返回矩阵各值中的最小值。

最大值 返回矩阵各值中的最大值。

最小值位置 返回最小值的第一个位置。

最大值位置 返回最大值的第一个位置。

变换列选项

创建变换列后，您可以执行以下操作：

重命名 重命名变换列。

添加至数据表 将变换列作为公式列添加至数据表。

对列中数据重新编码

使用重新编码工具可以同时更改一列中的所有值。例如，假设您想要比较计算机公司和制药公司的销售额。您当前的公司标签是“Computer”和“Pharmaceutical”。您想将它们更改为“Technical”和“Drug”。对 32 行中的所有数据逐步更改所有值将会十分繁琐低效，而且容易出错，特别是您的数据行数过多的情况下。重新编码会是一个更好的选择。

注意：

- 要对多个列中的相似值重新编码，请使用**列 > 标准化特性**中的“重新编码”选项。请参见“设置 JMP 列属性”一章中第 287 页的“标准化列特性”。
- 要仅对列标题重新编码，请选择**列 > 列名 > 重新编码列名称**。

要对数据重新编码，请执行以下步骤：

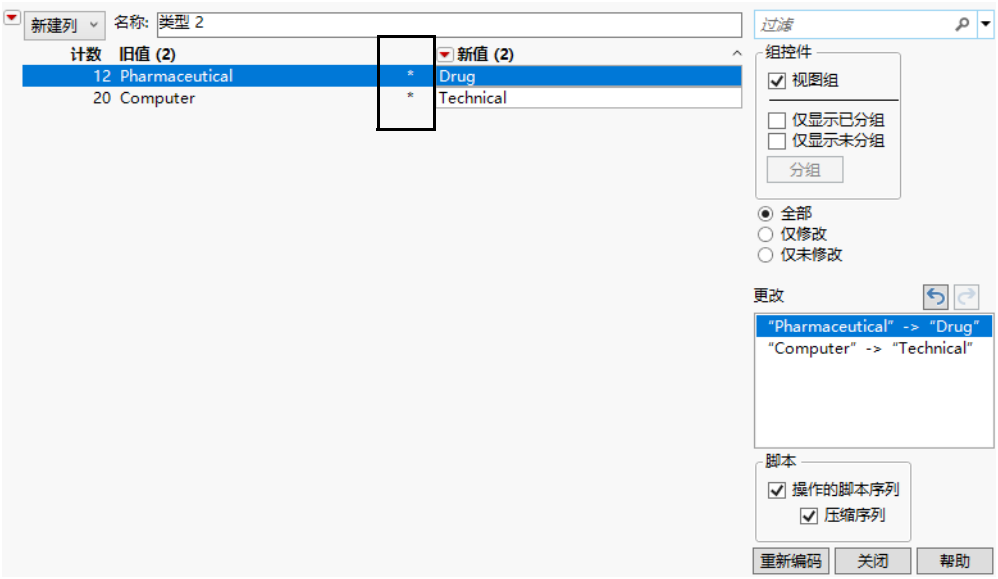
1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 通过点击列标题一次，选择**类型**列。
3. 选择**列 > 重新编码**。

   请注意，默认选定 “新建列”以便将重新编码的值放入新列。新列名为**类型 2**。
4. 在”重新编码 “窗口的**新值**框中输入所需的值。对于本例，在 “Computer”行中输入“Technical”，在 “Pharmaceutical”行中输入 “Drug”。

   请注意，重新编码的值显示在 “更改”框中。

图 4.43 重新编码窗口

星号指示哪些值已更改。



5. 点击**重新编码**。

   新值出现在数据表中名为**类型 2** 的新列下。

注意：

- 默认情况下，重新编码的数据显示在新列中。要始终对您选定的列重新编码，请选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)，选择**重新编码**，然后从“提交样式”列表中选择**原位**。
- 默认情况下，“重新编码”窗口中的数据按新值的升序排序。这将同时应用到“重新编码”和“重新编码列名称”。要更改数据排序方式，请选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)，选择**重新编码**并更改“排序样式”首选项。若指定，选择**旧值**将按列名在数据表中显示的自然顺序对列名进行排序，按值顺序列属性中的顺序对数据进行排序。选择**计数**可按计数的降序进行排序。
- 若数据表包含值标签，则标签显示在“重新编码”窗口中名为“新标签”的列中。编辑标签也会修改“值标签”列属性。您还可以选择**使用值标签替换值**以修改“新值”列中的数据。
- 对列重新编码时将更新以下列属性：值标签、值得分、值顺序、值颜色、超级类别、编码和缺失值代码。
- 若在数据类型为“数值型”的列中输入非数值型值，系统将提示您将数据类型转换为“字符型”。点击**是**可转换列并显示新值。点击**否**可保留列为“数值型”并显示缺失值。
- 在“重新编码”窗口中选择行也会在数据表中选择这些行以进行快速编辑。

重新编码选项

“重新编码”红色小三角旁边的列表中提供了以下选项：

新建列 为更改的数据创建新列并保留原始列。

公式列 使用以公式而不是值表示的更改创建新列。更改数据表原始列中的值将导致公式列自动更新该值。

原位 将任何更改应用到原始数据列。

在“重新编码”窗口还包括其他可用的选项：

过滤 搜索特定值。点击向下箭头选择搜索选项。有关每个选项的详细信息，请参见第 235 页的 “重新编码中的搜索过滤器选项”。请注意，您可以键入正则表达式来定制过滤器。

查看组 取消选择可查看未分组值。默认选定该选项。

仅显示已分组 显示已分组的重新编码值。

仅显示未分组 显示未分组的值。

分组 选定多个值时将变为可用。点击**分组**可使突出显示的值分到同一组中。若您在分组前编辑过值，则编辑过的值会成为“新值”列中的组代表。否则，出现次数最多的值会成为组代表。

全部 显示所有数据值。

仅修改 仅显示已修改的重新编码值。

仅未修改 仅显示已修改的重新编码值。

更改 显示在“重新编码”窗口中所做的更改的列表。

撤销上次对该窗口所做的更改。

重新调用上一次对窗口进行的更改。

脚本

要保存脚本以重新创建重新编码的数据表，请从红色小三角菜单中选择“脚本”，然后选择一个选项。

“重新编码”窗口包含以下脚本选项：

操作的脚本序列 尝试将您的操作保存到脚本并忽略数据。若使用红色小三角选项将所有字符串转换为大写，则生成的脚本包含将输入字符串转换为大写而不包含数据的代码。

压缩序列 通过删除不必要的操作来压缩顺序操作的脚本。考虑该选项可加快包含多个编辑的重新编码。

多重响应

删除空项 在“多重响应”列中，对数据表中的列重新编码时排除该列中的空项。

删除重复项 在“多重响应”列中，排除重复项。例如，若原始数据为 a、b、b、c，并且没有对这些特定项进行更改，则数据表中的重新编码列将包含 a、 b、 c。

对项排序 在“多重响应”列中，按字母顺序对各项排序。例如，若原始数据为 b、 a、 c，并且没有对这些特定项进行更改，则数据表中的重新编码列将包含 a、 b、 c。

用于重新编码的红色小三角选项

红色小三角菜单包含用于“重新编码”窗口的选项。

转换为首字母大写 将每个字词的首字母转换为大写字母，其余字母转换为小写字母。

转换为大写 将所有值转换为大写。

转换为小写 将所有值转换为小写。

去除首尾空格 删除首尾的空格字符。例如，若在名字 John 前后导入了多余空格，该命令将删除这些空格。

删除空格 删除首尾和内部的空格字符。

压缩空格 删除首尾空格字符并删除内部的重复空格字符。即：若存在多个空格字符，**压缩空格**命令会将两个空格替换为一个空格。

删除标点 从字词的开头、中间和结尾删除特殊字符 （如引号和 & 号）。

替换字符串 将指定字符的所有出现情况替换为新字符串或空白。请注意，您可以键入正则表达式来定制替换内容。

第一个单词 基于字符串的第一个单词对值分组。例如，若值为“John Smith”和“John Adams”，该命令会将它们分组到“John”下。

最后一个单词 基于字符串的最后一个单词对值分组。

分组类似值 允许您定制数据的分组方式。从分组选项列表中进行选择。请参见第 235 页的“用于重新编码的分组类似值选项”。

高级

除第一个单词以外的全部 排除第一个字词后，基于剩余值拆分值。

除最后一个单词以外的全部 排除最后一个单词后，基于剩余值拆分值。

从表应用映射 支持您使用映射数据表中的列来修改目标表中的值。请参见第 240 页的“示例：应用映射列中的值”。

提取段 支持您提取特定字符串。请参见第 237 页的“示例：依据逗号拆分”。

选择最近 支持您将映射数据表列中的值映射到要重新编码的列。

解析为数字 将数据转换为数字。

标签转换为代码 将标签值转换为代码。

转换为字符型 在数值列中，将列转换为字符列。

添加值标签 支持您向每个值分配值标签。

使用值标签替换掉值 用指定的值标签替换 “值标签”列属性中指定的值。仅当列具有 “值标签”属性时，该选项才可用。

重新调用 重新调用先前在 “重新编码”窗口进行的更改。

重新开始 将窗口返回默认状态。

脚本 查看选项以导入、合并或保存 “重新编码”脚本。

- **从文件导入**将导入 JSL 脚本以对先前重新编码的数据进行重新编码。在不同数据上运行相同脚本，可通过相同的方式对数据进行重新编码。
- **从数据表导入**将导入保存至数据表的 JSL 脚本。
- **从重新编码结果列导入**将从您选择的列（通常为您保存之前的重新编码结果的列）导入数据。
- **保存到文件**将 “重新编码”更改保存到 JSL 脚本。选择 “保存”后，系统会提示您命名并保存文件。
- **保存到数据表**将 “重新编码”脚本保存到当前数据表。

  要修改已保存脚本中的重新编码，请在数据表中右击该脚本并选择**使用重新编码编辑**。进行更改后，再次从红色小三角菜单中选择**脚本 > 保存到数据表**以将更改保存到新脚本中。
- **与数据表脚本合并**将在 “重新编码”窗口中进行的更改合并至保存到数据表的当前 “重新编码”脚本中。若有多个脚本，系统会提示您选择将哪个脚本与重新编码数据合并。
- **保存到脚本窗口**将 “重新编码”脚本追加至脚本窗口。

JMP PRO **发布至公式存储库** 将重新编码脚本发布至 “公式存储库”，“公式存储库”支持您在其他脚本中重复使用这些代码。

用于重新编码的右击选项

前六个选项可用于分组值：

分组至新值 （仅在选择多个值时可用）支持您为两个或多个选定值指定新值。

分组至 （仅在选择多个值时可用）右击选定值可选择不同的分组值或组代表。该列表显示前 8 个可能的组代表。

交换新值 当突出显示两个值时，选择**交换新值**可使第一个值的新值采纳第二个值的新值，反之亦然。

成为代表 右击组中的单个值并选择**成为代表**可使选定的值成为新值。

从组中删除 值分组后，右击一个或多个值以将其从该组中删除。

分组类似值 右击单个字符串可查找类似值。“分组选项”窗口随即显示。请参见 “用于重新编码的分组类似值选项”。

打开所有组 打开各组，以便可以看到每个组中的所有值。

关闭所有组 折叠各组内的值。

为所有数据提供了以下选项：

选择行 选择数据表中的选定行。在“重新编码”窗口中选择某行时，将在数据表中自动选定相应的行。但是，若在数据表中选择行，在“重新编码”中并不会选定这些行。若要仅选择“重新编码”窗口中显示的行，请右击该行，然后选择该选项。

制成数据表 创建“重新编码”窗口中显示的结果数据表。

复制选定的列行 仅从您右击的列复制数据。您可以随后将该数据粘贴到数据表中。

复制选定的表行 复制所有列的数据。您可以随后将该数据粘贴到数据表中。

用于重新编码的分组类似值选项

选择以下分组类似值命令可增加分组准确度：

忽略大小写 将忽略项的大小写。

忽略非打印字符 将忽略非打印字符。某些数据可能包含仅计算机能够读取的非打印字符（如文件分隔符）。

忽略空格 将忽略空格。

忽略标点符号 将忽略标点符号。

允许进行字符编辑 当分组类似值时允许用新值替换字符。

最大差异比 根据最大比例差异对值进行分组。例如，键入“.25”，可对至多存在 25% 差异的值进行分组。

最大字符差异 根据不相邻字符差异的最大数对值进行分组。例如，键入“5”，可对存在 5 个或少于 5 个字符差异的值进行分组。

最小源字符串长度 防止字符串太短时被匹配。尝试使用该选项以及“差异比”和“最大字符差异”选项，以指定适用于数据的值。

重新编码中的搜索过滤器选项

点击搜索框旁边的下箭头按钮以优化搜索。

包含词条 返回包含一部分搜索条件的项。搜索“ease oom”返回如“Release Zoom”这样的消息。

包含短语 返回包含完全搜索条件的项。搜索“text box”返回包含“text”后面直接跟着“box”的条目（例如，“Context Box”和“Text Box”）。

以短语开头 返回以搜索条件开始的项。

以短语结尾 返回以搜索条件结束的项。

整条短语 返回包含整个字符串的项。搜索 “text box”返回仅包含 “text box”的条目。

正则表达式 允许您在搜索框中使用通配符 (*) 和句点 (.)。搜索“get.*name”查找包含“get”后面跟着一个或多个单词的项。它返回 “Get Color Theme Names”、“Get Name Info”、“Get Effect Names”等。

反转结果 返回不匹配搜索条件的项。

匹配全部项 返回同时包含两个字符串的项。搜索 “t test”返回包含任一搜索字符串或两者的元素：“Pat Test”、“Shortest Edit Script”和 “Paired t test”。

忽略大小写 忽略搜索条件中的大小写。

全字匹配 基于“匹配全部项”设置返回包含字符串中每个单词的项。若您搜索“data filter”，并且选择了 “匹配全部项”，则返回同时包含 “data”和 “filter”的条目。

按类似值分组的示例

您可以根据类似值之间存在差异的字符数对其进行分组。

1. 选择**帮助 > 样本数据库**，然后打开 Candy Bars.jmp。
2. 选择**名称**列。
3. 选择**列 > 重新编码**。
4. 点击红色小三角并选择**分组类似值**。
5. 确保已选定 “允许进行字符编辑”。
6. 在 “最大差异比”旁边，键入 “.5”。

   这允许 JMP 将存在 50% 或少于 50% 差异的值进行分组。换句话说，就是至少共享相同字符的 50%（或一半）的值。“差异比”是由比较每个值的总字符数和两个给定值之间的唯一字符总数决定的。
7. 在 “最大字符差异”旁边，键入 “6”。

   将对不超过六个字符的值进行分组。
8. 点击**确定**。

图 4.44 对类似值分组

显示在新值列中的值代表重新编码数据表中的分组值。

9. （可选）若要更改新值以表示组中不同的旧值，请右击“Almond Roca”，然后选择**成为代表**。

图 4.45 成为代表

10. （可选）要用重新编码的数据替换数据表中的原始数据，请从红色小三角菜单中选择**原位**，然后点击**重新编码**。

示例：依据逗号拆分

Aircraft Incidents.jmp 样本数据表包含用逗号分隔城市和州名称的一列。您可以依据逗号拆分该列中的值，只用州名对该列重新编码。

1. 选择**帮助** > **样本数据库**，然后打开 Aircraft Incidents.jmp。
2. 选择**位置**列并选择**列** > **重新编码**。

图 4.46 原始数据

旧值 (1537)	新值 (1537)
Abilene, TX	Abilene, TX
Adelanto, CA	Adelanto, CA
Afton, VA	Afton, VA
Aieia, HI	Aieia, HI
Aiken, TX	Aiken, TX
Ainsworth, NE	Ainsworth, NE
Akiachak, AK	Akiachak, AK
Akron, CO	Akron, CO
Alamogordo, NM	Alamogordo, NM
Alamosa, CO	Alamosa, CO
Albion, IL	Albion, IL
Albuquerque, NM	Albuquerque, NM
Alcova, WY	Alcova, WY
Alexander City, AL	Alexander City, AL
ALEXANDER, AK	ALEXANDER, AK
Alice, ND	Alice, ND
Allamuchy, NJ	Allamuchy, NJ
Almond, WI	Almond, WI
Alton, IL	Alton, IL
Alturas, CA	Alturas, CA
Altus, OK	Altus, OK
Alvarado, TX	Alvarado, TX
Amargosa Valley, NV	Amargosa Valley, NV
Amarillo, TX	Amarillo, TX
AMHERST, NH	AMHERST, NH

3. 点击红色小三角并选择**高级 > 提取段**。

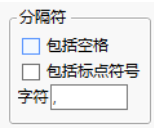
   在“预览”窗格中，注意数据包含城市，后跟逗号和缩写的州名称。您将提取每个字符串的最后两个字符。

4. 取消选择窗口顶部的**包括空格**和**包括标点符号**。

   提取过程中将不考虑空格和标点符号。

5. 在“字符”框中键入一个逗号。

图 4.47 指定分隔符



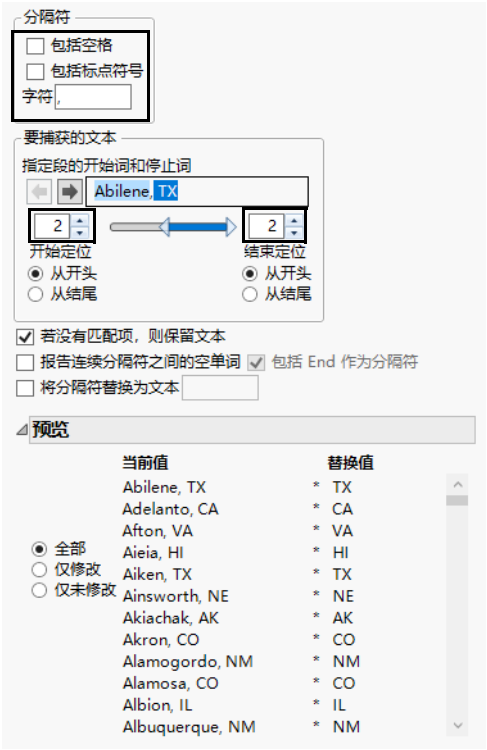
   逗号用于提取最后两个字符。

6. 在“提取段”窗口中，在“开始定位”和“结束定位”上方键入“2”。

   在第二个单词处开始并结束段提取。

7. 确保为“开始定位”和“结束定位”选项选择了“从开头”。

图 4.48 “提取段”窗口



提示：在“要捕获的文本”部分中，点击左箭头或右箭头滚动浏览各值。这样您就可以预览设置。

8. 在图 4.48 所示的 “预览”窗格中，请注意替换值仅包含州的缩写。
9. 在“预览”窗格中向下滚动。请注意， Atlantic Ocean 的当前值和替换值是相同的。数据不能按逗号拆分。

图 4.49 不具有州的数据

当前值	替换值
ATHENS, GA	* GA
Atlanta, GA	* GA
Atlantic Ocean	Atlantic Ocean

10. 点击**确定**。

含有匹配的州名的条目将被分组。

图 4.50 重新编码的城市，州名

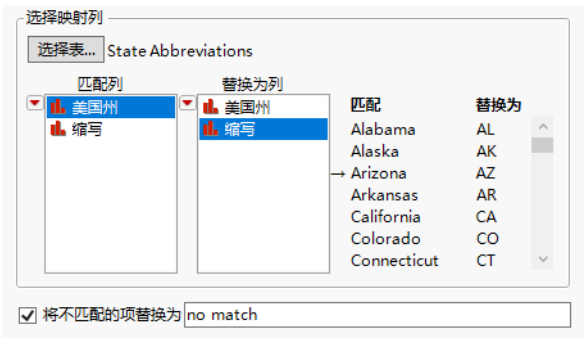
示例：应用映射列中的值

映射数据表中的列包含可用于对另一个数据表中的列重新编码的值。在本例中，State Capitals.jmp 中的美国各州将替换为 State Abbreviations.jmp（映射数据表）中定义的州缩写。

1. 选择**帮助** > **样本数据库**，然后打开 State Abbreviations.jmp 和 State Capitals.jmp。
2. 在 State Capitals.jmp 中，选择**美国州**列。
3. 选择**列** > **重新编码**。
4. 点击红色小三角并选择**高级** > **从表应用映射**。
5. 点击**选择表**，选择 State Abbreviations.jmp 并点击**确定**。
6. 在 “匹配列”框中，选择**美国州**。
7. 在 “替换为列”框中，选择**缩写**。

   “匹配”列和 “替换为”列显示映射的结果。

图 4.51 “匹配”列和 “替换为”列



8. 选择**将不匹配的项替换为**框并键入 “no match”。

   “预览”部分显示如何映射值。向下滚动可以看到 “no match”的州拼写有误，因此它们不会映射到**缩写**列中的任何州。

图 4.52 不匹配的列的预览

当前值		替换值
Kentucky	*	KY
Louisiana	*	LA
Maine	*	ME
Maryland	*	MD
Massachusets	*	no match
Michigan	*	MI
Minnesota	*	MN
Missisippi	*	no match

9. 点击**确定**，然后点击**重新编码**。

   包含缩写州名的新列出现在 State Capitals.jmp 中名为**美国州 2** 的新列中。

图 4.53 缩写州名

美国州	美国州 2
Alabama	AL
Alaska	AK
Arizona	AZ
Arkansas	AR
California	CA
Colorado	CO
Connecticut	CT
Delaware	DE
Florida	FL
Georgia	GA
Hawaii	HI
Idaho	ID
Illinois	IL
Indiana	IN
Iowa	IA
Kansas	KS
Kentucky	KY
Louisiana	LA
Maine	ME
Maryland	MD
Massachusets	no match
Michigan	MI
Minnesota	MN
Missisippi	no match

要修复不匹配的条目，请参见第 241 页的 “示例：改正拼写错误单词”。

示例：改正拼写错误单词

您可以使用数据表作为字典来改正拼写错误的单词。State Abbreviations.jmp 包含与美国各州名称和缩写对应的列。State Capitals.jmp 包含一些拼写错误的州名以模仿实际数据。本例演示如何改正拼写错误的州名。

1. 选择**帮助** > **样本数据库**，然后打开 State Abbreviations.jmp 和 State Capitals.jmp。
2. 在 State Capitals.jmp 中，选择**美国州**列。
3. 选择**列** > **重新编码**。

4. 点击红色小三角并选择**高级** > **选择最近**。
5. 点击**选择表**，选择 State Abbreviations.jmp 并点击**确定**。
6. 在 “匹配列” 框中，选择**美国州**。
7. 在 “预览” 窗格中，向下滚动并注意 “Massachusets” 将替换为 “Massachusetts”，“Missisippi” 将替换为 “Mississippi”。

图 4.54 替换拼写错误的州名

当前值		替换值
Kentucky		Kentucky
Louisiana		Louisiana
Maine		Maine
Maryland		Maryland
Massachusets	*	Massachusetts
Michigan		Michigan
Minnesota		Minnesota
Missisippi	*	Mississippi

8. 点击**确定**，然后点击**重新编码**。
   包含更新州名的新列将添加到数据表中。

示例：解析为数字

“解析为数字” 选项用于将字符型数据转换为数值型数据。该选项等效于 JSL Num() 字符函数。支持多种格式，如日期和时间、货币以及科学符号。若某种格式不受支持，则会显示缺失值。

1. 打开数据表并选择包含数值和字符值的列。

图 4.55 包含数值型和字符型数据的列

	x
1	a
2	4,033.67
3	$8000.56
4	5/4/2019
5	10/21/2017 5:10:33
6	12Jan2018
7	1.07E-6

2. 选择**列** > **重新编码**。
3. 从红色小三角菜单中选择**解析为数字**。
   将显示新值和旧值。

图 4.56 新值预览

计数	旧值 (7)		新值 (7)
1	12Jan2018	*	.
1	a	*	.
1	1.07E-6	*	1.07e-6
1	4,033.67	*	4033.67
1	$8000.56	*	8000.56
1	10/21/2017 5:10:33	*	3.591e+9
1	5/4/2019	*	3.64e+9

4. 点击**重新编码**。

   包含重编码值的新列将添加到数据表中。该列的数据类型为 “数值”。

   “a”无法被解析为数字，因此该值在新列中显示为缺失值。

图 4.57 解析为数字

	X	X 2
1	a	•
2	4,033.67	4033.67
3	$8000.56	8000.56
4	5/4/2019	3.6398e+9
5	10/21/2017 5:10:33	3.5914e+9
6	12Jan2018	•
7	1.07E-6	1.07e-6

示例：将重新编码的值应用到不同列

对列重新编码时，您可以将重新编码的值保存在脚本中。导入或运行该脚本时，您最初重新编码的列将被再次重新编码。

假定您想要对包含相同值的不同列进行重新编码。将重新编码的值保存在某一公式列中，然后编辑该公式。

1. 在“重新编码”窗口的“新值”列中输入重新编码的值。
2. 从红色小三角菜单旁边的列表中选择**公式列**。
3. 点击**重新编码**。
4. 右击公式列，然后选择**公式**。
5. 在公式中，选择已重新编码的列。

   在本例中，**性别**列已重新编码。

图 4.58 重新编码列

```
Recode(
	性别,
	{Map Value(Original, {"F" ⇒ "FEMALE", "M" ⇒ "MALE"}, Unmatched(Working))}
)
```

6. 在 “列”列表中，选择将替换**性别**的列。

在本例中，您想要将 sex （性别）列替换为 gender （性别），所以选择 gender （性别）。

图 4.59 选择 “gender （性别）” 列

编辑数据表

JMP 提供许多方法来更改数据表的基本属性。

- “重命名数据表”
- “锁定数据表”
- “压缩数据表”
- “复制表脚本”
- “使用数据表变量”
- “在数据表中创建和保存脚本”

重命名数据表

数据表的名称显示在其窗口顶部、表面板中，以及所有相关的分析报表中。您可以采用以下任意方式更改数据表的名称：

- 选择**文件 > 另存为**，另存为新名称。
- 在表面板中，双击表名，键入新名称，然后按 Enter 键。
- 在窗口中，选择**窗口 > 设置标题**。

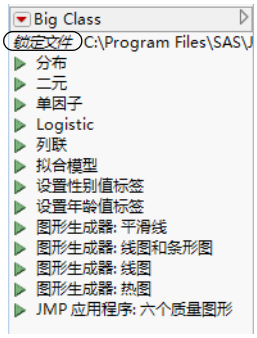
锁定数据表

锁定数据表可防止添加或编辑数据和列属性。您仍可以指定行状态、运行分析，以及执行其他操作。要锁定数据表，请点击表面板中的表名旁边的红色小三角，然后选择**锁定数据表**。

该数据表名称旁边将出现一个锁状图标 。要对该文件解锁，请再次选择**锁定数据表**。

若在 JMP 之外将数据表设置为只读（例如，通过在 Windows 上更改其属性），该数据表会包含一条注释，通知您它已锁定（图 4.60）。这种类型的锁定允许用户编辑数据表，但不能保存更改。

图 4.60 只读文件



压缩数据表

压缩数据表可缩减存储文件的大小。您仍可以运行分析、指定特征，以及执行其他操作。要压缩数据表，请点击表面板中表名旁边的红色小三角，然后选择**保存时压缩文件**并保存该数据表。

保存数据表后，数据表名称旁边将显示一个压缩图标 。要对文件解压缩，请再次选择**保存时压缩文件**。

此外，您还可以配置 JMP 在保存时始终使用 GZ 压缩，方法是选择**首选项 > 常规 > 将数据表列保存为 GZ 压缩文件**。

注意：保存时压缩文件选项只会缩小文件大小。该命令不影响分析数据所需的内存大小。要减小用于分析的文件大小和内存，请使用**列 > 实用工具 > 压缩选定列**。请参见第 215 页的“压缩数据表中的选定列”。

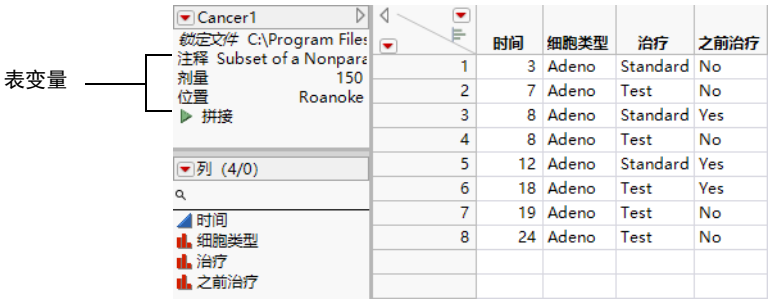
复制表脚本

通过复制表脚本，可以将脚本粘贴到另一个脚本中以重新创建表

使用数据表变量

数据表变量可以包含文本信息（例如，数据的源信息），或者包含可供列公式或 JSL 脚本使用的值。数据表变量名显示在数据网格左侧的表面板中。

图 4.61 表面板中的表变量



表变量的用法

在以下情况下使用表变量：

- 用于记录表
- 在公式中使用
- 在 JSL 脚本中使用

使用表变量记录表

表变量主要用于记录表。随 JMP 安装的许多样本数据表中都包含一个名为“注释”的表变量。该变量提供有关数据的详细信息（例如，数据的来源）。图 4.61 中的示例显示了一个数据表，其中包含**注释**，作为其表变量之一。在您在 JMP 中使用“实验设计”命令创建设计表时，JMP 也会自动创建表变量。设计表具有名为**设计**的表变量，变量值为设计类型的名称。

在公式中引用表变量

表变量还可以合并到使用“公式编辑器”生成的公式中。这些公式通过引用表变量来计算列值。有关构造使用表变量的公式的详细信息，请参见“在 JMP 中创建公式”一章中第 344 页的“在公式中引用数据表值”。

在 JSL 脚本中使用数据表变量

您还可以将表变量合并到 JSL 脚本中。请参见《Scripting Guide》。

数据表变量操作

添加新数据表变量

1. 在表面板中，点击数据表名称左侧的红色小三角。
2. 选择**新建表变量**。
3. 在标记为**名称**和**值**的框中分别为变量指定名称和值。
4. 点击**确定**。
   表变量显示在表面板中。

查看或编辑表变量

1. 双击现有表变量的内容。
2. 编辑内容。

编辑表变量名

1. 双击表变量名。
2. 编辑名称。

删除表变量

选择一个或多个表变量并按 Delete 键，或右击选定的变量并选择删除。

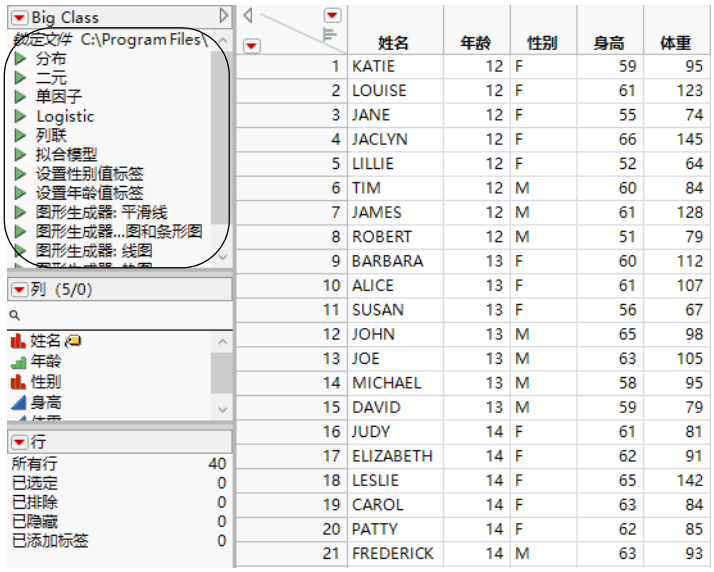
拼接带表变量的数据表

请参见“重塑数据”一章中第 311 页的“拼接数据表和表变量的示例”。

在数据表中创建和保存脚本

要自动完成各种分析和任务，您可以创建 JSL 脚本并将其保存至数据表（图 4.62）。有关编写数据表脚本的详细信息，请参见《Scripting Guide》。

图 4.62 随数据表保存的脚本



	姓名	年龄	性别	身高	体重
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	79
9	BARBARA	13	F	60	112
10	ALICE	13	F	61	107
11	SUSAN	13	F	56	67
12	JOHN	13	M	65	98
13	JOE	13	M	63	105
14	MICHAEL	13	M	58	95
15	DAVID	13	M	59	79
16	JUDY	14	F	61	81
17	ELIZABETH	14	F	62	91
18	LESLIE	14	F	65	142
19	CAROL	14	F	63	84
20	PATTY	14	F	62	85
21	FREDERICK	14	M	63	93

将报表脚本保存至数据表

运行某个分析后，在报表窗口中，您可以向数据表添加脚本。该脚本可生成能够重现分析的 JSL。

将脚本保存至数据表

从报表窗口中，点击平台对应的红色小三角，然后选择**保存脚本 > 至数据表**。

将报表脚本保存至数据表的示例

首先，需要创建分析，然后保存该脚本。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**体重**，然后点击 **Y，响应**。
4. 选择**身高**，然后点击 **X，因子**。
5. 点击**确定**。
6. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**拟合线**。
7. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**保存脚本 > 至数据表**。
8. 在“脚本另存为”窗口中，输入脚本的名称。
9. 要替换同名的现有脚本，请从“名称”列表中选择它，然后选择**替换现有脚本**。

10. 要保存与现有脚本同名的脚本并使用相同名称，请选择**附加唯一后缀**。

    将向脚本名称末尾添加一个编号。

11. 点击**确定**。

该脚本添加至表面板的底部。

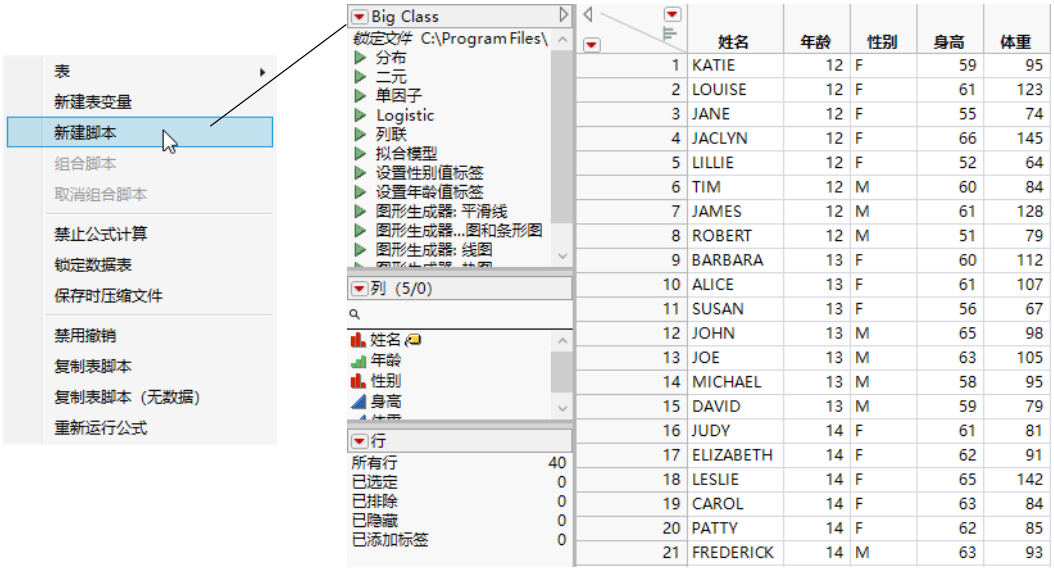
提示：若要在每次打开数据表时都自动运行特定的脚本，请将该脚本命名为 OnOpen。只能将数据表中保存的一个脚本设置为自动运行。若在“拟合模型”脚本中将脚本命名为 Model（或 model），当您选择**分析 > 拟合模型**时将根据脚本自动填充启动窗口。

为数据表编写 JSL 脚本

使用 JSL 向数据表添加脚本

1. 点击表面板中数据表名称左侧的红色小三角。

图 4.63 创建脚本



2. 选择**新建脚本**。
3. 通过在**名称**旁边的框中键入名称来命名脚本。
4. 通过在**脚本**旁边的框中输入 JSL 代码来添加脚本。
5. 执行以下操作之一：
   - 若要对脚本运行 JSL 调试程序以便检查错误，请点击**调试脚本**。
   - 若完成对脚本的编辑，请点击**确定**。脚本将出现在表面板中，窗口随即关闭。

- 若未完成对脚本的编辑，但您想要保存该脚本，请点击**保存**。脚本将显示在表面板中，且窗口保持打开，以供进一步编辑。
- 若要运行脚本，请点击**运行**。

运行、编辑、删除或复制脚本

要运行、编辑、删除或复制一个保存到数据表中的脚本，请执行下列操作步骤：

1. 在表面板中，右击脚本名称旁边的绿色小三角，或右击脚本名称。
2. 选择以下命令之一：
   - 运行脚本
   - 编辑
   - 删除
   - 复制

一旦复制脚本，随后便可将其粘贴至脚本窗口或粘贴至另一数据表的表面板。

第 5 章

设置 JMP 列属性

“列信息”窗口

使用“列信息”窗口可对数据表中的列设置特定属性。下面举例说明了您可以对列执行的操作：

- 更改数据类型和建模类型
- 更改数值格式
- 添加公式
- 对值或缺失值指定限制
- 对分类值或行数据排序
- 保存规格限、控制限或响应限
- 输入 Sigma 的已知值

您还可以在各列之间对特性和属性进行标准化、向列分配预选的分析角色，以及压缩数据表中的列。

图 5.1 “列信息”窗口

目录

关于“列信息”窗口 253
关于建模类型 255
短整型格式 256
数值格式 256
在行状态列中存储信息 261
初始化列中数据 262
JMP 中的列属性 263
基本列属性 264
验证列值的属性 264
将信息附加至列值的属性 266
控制列显示的属性 270
在建模和实验设计中使用的属性 271
与控制图和能力相关的属性 272
控制列在平台中的使用方式的属性 273
SAS 属性 280
链接数据表与虚拟连接属性 280
其他列属性 282
由 JMP 分配并控制的属性 284
在各列之间标准化特性和属性 286
对列值重新编码 286
标准化列特性 287
标准化列属性 287
删除列属性 287
关于公式标准化的示例 287
为列指定预选分析角色 288

关于 “列信息” 窗口

使用 “列信息” 窗口可指定列的所有特性和属性。您可以通过以下任意一种方式访问 “列信息” 窗口：

- 选择**列** > **新建列**。
- 点击现有的列标题并选择**列** > **列信息**。
- 右击现有的列标题并选择**列信息**。
- 直接在列名上方双击。
- 在列面板中，右击一列并选择**列信息**。

图 5.2 “列信息” 窗口

“列信息” 窗口包含以下信息：

列名 键入或编辑列的名称。

锁定 锁定列，以防编辑任何列值。锁定某列后，锁状图标 ( ) 会显示在数据表的列面板中的列名旁边。若向某列添加公式，该列将自动锁定。

数据类型 选择或更改列的数据类型，以便确定以下项：

- 列值在数据网格中的格式
- 列值在内部保存的方式
- 列值是否可用在计算中

请参见第 255 页的 “关于建模类型”。

从以下数据类型中选择：

数值 这些列仅包含数字（带或不带小数点）。

字符 这些列包含所有字符，包括数字。在字符列中，数字被视为字符，并被视为离散值而不是连续值。字符值的最大字段宽度为 32,766 个字节。

行状态 这些列包含行状态信息，用来指示行的状态是否为已排除、已隐藏、已添加标签、已着色还是已标记。请参见第 261 页的 “在行状态列中存储信息”。

表达式 这些列包含从简单条目到复杂语句序列的 JSL 表达式。尤其是表达式可以对应图片、矩阵或表达式。您可以将表达式（如图像）从桌面拖放至表达式列的单元格中。请参见第 282 页的 “表达式角色”。

注意：

– 若您将列的 “数据类型” 从 “字符型” 更改为 “数值型”，则任何字符值将变为缺失数据值且无法恢复。

– 若在具有 “数值型” 数据类型的列中输入非数值数据，将显示警告窗口，提示您将数据类型更改为 “字符型”。点击**重试**可返回到数据表以重新输入值。点击**更改**可更改数据类型。点击**恢复**可取消编辑并返回单元格的值。

– 还可能提供短整型格式。请参见第 256 页的 “短整型格式”。

建模类型 （仅限数值型或字符型数据类型）选择或更改列的建模类型，指示 JMP 在分析期间如何处理列值。您可以更改建模类型，以不同方式处理变量。请参见第 255 页的 “关于建模类型”。

从以下建模类型中选择：

连续型 仅限数值型数据类型。连续型值被视为连续测量值。JMP 平台在计算中直接使用数值。

有序型 数值或字符数据类型。JMP 平台将有序型值视为具有顺序的离散分类值。在 “图形生成器” 中，若有超过 50 个离散值，JMP 将该列视为连续列。

– 若该列具有数值型数据类型，则值将按照其数值大小来排序。

– 若该列具有字符型数据类型并且 “值排序” 列属性已保存到该列中，则值将按照 “值排序” 属性来排序。否则，值将按照其字母数字数据值顺序来排序。不过，在值具有明显顺序的特殊情况下，值将适当地自动排序。请参见第 267 页的 “值顺序”。

名义型 数值或字符数据类型。JMP 平台将所有值视作没有隐式顺序的离散值。

多重响应 仅限字符型数据类型。单个单元格中的多个非重复条目必须用逗号分隔。支持多重响应列的 JMP 平台将逗号分隔列表中的每个条目视为一个单独的数据值。若条目使用逗号之外的字符分隔，请改为使用 “多重响应” 列属性。请参见第 276 页的 “多重响应”。

非结构化文本 仅限字符型数据类型。这些值通常唯一，因此不适合分类分析。“文本分析器” 平台非常适合分析非结构化文本值。

向量 仅限表达式数据类型。单元格中的条目为列或行向量。支持向量建模类型的 JMP 平台可识别这些向量并在计算中正确处理它们。

无 任何数据类型。当某列无法用其他建模类型适当表示时，即可将 “无” 用作建模类型。例如，可以为图片列或 ID 值列分配 “无” 建模类型。JMP 启动窗口不允许您向建模类型为 “无” 的列分配角色。

提示：您必须使用 “列信息” 窗口将列的建模类型更改为 “多重响应”、“非结构化文本” 或 “向量”。

格式 （仅限数值型数据类型）选择或更改数值列的显示格式。请参见第 256 页的 “数值格式”。

初始化数据 （仅在创建新列时出现）指定您希望在列中显示的初始数据值的类型。请参见第 262 页的 “初始化列中数据”。

列属性 （包含不同属性的列表）为列指定属性。请参见第 263 页的 “JMP 中的列属性”。

提示：点击**下一个**按钮可继续添加列。

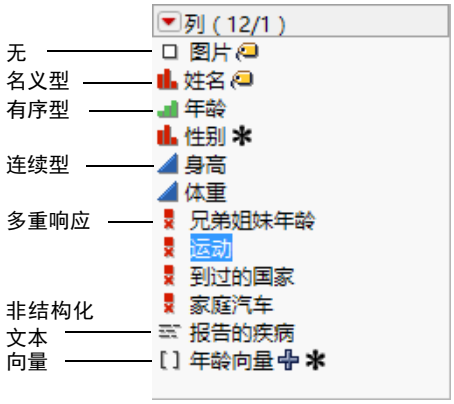
关于建模类型

JMP 数据表中的列可以包含不同类型的信息。不过，单个列中的所有信息必须具有相同的数据类型和建模类型。

- 导入数据时，JMP 会猜测要使用的数据类型和建模类型。因此，您应该确认 JMP 的猜测是否正确。
- 手动插入数据时，应指定数据类型和建模类型。

图 5.3 演示了用来标识不同建模类型的图标。

图 5.3 列面板中的建模类型图标



点击某个图标可将建模类型更改为**连续型**、**名义型**、**有序型**或**无**。

请参见第 254 页的 “建模类型”。

提示：仅当数据类型为数值型时，才可以选择连续型。若连续型选项在菜单上灰显，而您想要将列设置为连续型，则必须首先在“列信息”窗口中更改列的数据类型。

短整型格式

对数据使用正确的短整型格式时，看不到数字的显示有何不同差异。但是，数字占用的磁盘和内存空间都较少。必须在首选项中激活短整型格式，该格式才会显示在“列信息”窗口中。

在 “列信息”窗口中提供短整型格式

1. 选择**文件 > 首选项**，然后点击**表**。
2. 选择**允许短数值数据格式**选项。
3. 点击**确定**返回数据表。

以短整型格式存储数值数据：

1. 在要设置为短整型值的列名上方双击。
   “列信息”窗口随即出现。
2. 从数据类型菜单中，选择**单字节整数**、**双字节整数**或 **4 字节整数**。

JMP 将值存储为您选定范围内的整数。以下是数字的示例：

- 对于单字节整数，您可以输入的数字范围介于 -126 到 127 之间。
- 对于双字节整数，您可以输入的数字范围介于 -32,766 到 32,767 之间。
- 对于 4 字节整数，您可以输入的数字范围介于 -2,147,483,646 到 2,147,483,647 之间。

数值格式

对于数值列，“列信息”窗口中会显示“格式”菜单。指定格式，指示 JMP 如何在列中显示数字。对于所有格式选项，您都可以指定希望列中单元格容纳的字符总数。请参见第 256 页的“指定宽度”。

有关格式选项的说明，请参见第 257 页的“数值格式选项”。

提示：要向表示千位或千位以上的值添加逗号，请选择**使用千位分隔符**选项。必须在“宽度”框中为每个逗号占一个位置，否则这些逗号可能不显示。该选项可用于最佳、固定小数位数、百分比和货币格式。

指定宽度

在宽度字段中指定某个数字时，务必考虑到所有可能的字符。字符包括：数字、小数点、逗号和货币符号。

数值格式选项

从以下数值格式选项中进行选择：

最佳 允许 JMP 考虑每个单元格值的精度，并选择最佳方法来显示该值。默认情况下，列的物理宽度为 12 个字符。

固定小数位数 将列中所有值显示为舍入到您指定的小数位数的值。

- 要仅查看整数，请将小数位数设置为零。
- 若小数点后的整数个数小于您指定的小数位数，会添加零以填满小数位数。例如，若值为 1.23 而您在 “小数位数” 框中键入了 **5**，则 JMP 会显示具有五位小数的数字：1.23000。

百分比 将数值乘以 100，再在该数值后显示百分比符号。

P 值 显示概率值。宽度默认值为 12。若数字小于 0.0001，则该数字显示为 <.0001。该格式多用在 JSL 脚本中，数据表列很少需要该格式。

科学记数法 以科学记数法显示数字。若输入数字 123456，则该数字显示为 1.23456e+5。选择 “小数位数” 显示小数点并输入点数。

工程 与科学记数法类似，但指数始终是 3 的倍数，且 “小数位数” 字段表示有效位数。

工程 SI 与 “工程” 相同，但指数用 SI 符号替换。

精度 将数字舍入到给定的有效位数。在 “小数位数” 字段中指定有效位数。

货币 使用两位小数、千位分隔符和货币符号对数值作货币格式化。其货币符号在用户计算机的语言 / 区域设置中被指定。**货币**格式的默认宽度为 15。若数字需要更宽的字段宽度，**最佳**格式便成为默认的格式。一旦指定，货币符号即显示在列和包含该列的图形中。

日期 将列中所有值显示为日期。请参见第 258 页的 “日期格式”。

时间 将列中所有值显示为特定的时刻，如 12/2/03 2:23 PM。请参见第 259 页的 “时间格式”。

持续时间 将列中所有值显示为持续时间，如小时、分和秒。

:day:hr:m, :day:hr:m:s 显示持续时间，如 52:03:01:30 或 52 天 3 小时 1 分 30 秒。

hr:m, hr:m:s, min:s 显示持续时间，如 17:37 或 17 小时 37 分。

地理 显示地图的纬度和经度数字格式。纬度和经度选项如下：

DDD 度

DMM 度、分

DMS 度、分、秒

在每种格式中，最后一个经纬度单位可具有小数部分。可以使用带符号的经纬度单位或方向后缀来指定方向。要显示带符号的经纬度单位，如 *-59°00′00″*，请取消选择**方向指示符**。要显示方向后缀，如 *59°00′00″ S*，请选择**方向指示符**。

要将空格用作经纬度单位分隔符，请取消选择**经纬度单位符号**。要使用度、分、秒符号，请选择**经纬度单位符号**。

定制 支持您定义数值列的定制格式，包括定制列宽和小数位数。选择**定制**，点击**设置定制格式**，然后在 “公式编辑器” 窗口中定义格式。例如，若您不想在数字旁边显示百分比符号，请选择该选项并将数字乘以 100。这样可以替代选择 “百分比” 格式。或者，若您的数据以米为单位，而您想要以千米为单位显示数据，则可以添加公式 `Char(:value / 1000) || " km"`。您可能还想要应用 “设置日期或时间格式” 列表中未提供的日期/时间格式，如 D/M/Y。

格式模式 支持您定制数值的日期时间格式。详细信息，请参见第 260 页的 “使用格式模式定制日期时间格式”。

日期格式

选择日期格式时，您还可以指定输入格式。日期格式指示日期如何显示在数据表单元格中，输入格式指示如何输入日期。

若为已包含数据的数值列指定日期格式，则数值型值会被视为从 1904 年 1 月 1 日开始计算得出的秒数。例如，若您有一个单元格值为 1,234,567,890 的数值列，且您将格式更改为**日期** > m/d/y，则单元格值将显示为 02/13/1943。

表 5.1 中的示例使用日期 2004 年 12 月 31 日。

表 5.1 日期格式

格式	显示为
m/d/y	12/31/2004
mmddyyyy	12312004
m/y	12/2004
yyyyQq	2004Q4
d/m/y	31/12/2004
ddmmyyyy	31122004
ddMonyyyy	31Dec2004
Monddyyyy	Dec312004
y/m/d	2004/12/31

表 5.1 日期格式 (续)

格式	显示为
yyyymmdd	20041231
yyyy-mm-dd	2004-12-31
完整日期	Friday, December 31, 2004
缩写日期	Dec 31, 2004
语言/区域日期	因本地操作系统设置而异。以下是一个示例：在美国，本地操作系统设置为 mm/dd/yyyy (12/31/2004)。

注意：要更改日期显示在图形中的方式而不更改日期显示在数据表中的方式，请参见 “JMP 报表” 一章中第 431 页的 “更改轴的数值格式”。

时间格式

选择时间格式时，您还可以指定输入格式。时间格式指示时间如何显示在数据表单元格中，输入格式指示如何输入时间。

- 您可以在以下日期格式的预先计划日期的午夜之后添加小时数、分钟数和秒数：
  - m/d/y
  - d/m/y
  - y/m/d
  - ddMonyyyy
  - Monddyyyy
  - 语言/区域日期

  例如， 2004 年 12 月 31 日具有数值 3,187,296,600，该值表示 12/31/2004 12:10 AM。
- :day:hr:m 和 :day:hr:m:s 显示从 1904 年 1 月 1 日开始计算的天数、小时数、分钟数和秒数。例如，2004 年 12 月 31 日的结果分别为 :36890:00:10: 和 :36890:00:10:00。
- h:m:s 和 h:m 显示日期字段中的日期的小时、分钟和秒部分。例如，2004 年 12 月 31 日 12:10 AM 的结果分别为 12:10:00 AM 和 12:10 AM。
- yyyy-mm-ddThh:mm 和 yyyy-mm-ddThh:mm:ss 显示年、月、日和时间。例如，2004-12-31T12:10:00。T 是一个表示其自身的文字值。

注意：要更改时间显示在图形中的方式而不更改时间显示在数据表中的方式，请参见 “JMP 报表” 一章中第 431 页的 “更改轴的数值格式”。

使用格式模式定制日期时间格式

格式模式是用于定义日期时间格式的字符串，例如 <YYYY></><MM></><DD> <hh><:><mm><:><ss><ampm>。尖括号中的模式部分称为**字段描述符**。字段描述符表示一个值（例如 <YYYY>，这是一个四位数的年份）或其他日期时间文本（例如，</>，它是一个特定于语言/区域的日期分隔符）。

格式模式支持您生成 JMP 中未提供的格式。这些格式可用于格式化和输入数据。您可以自行键入模式，将内置格式用作起点，或者从列表中选择字段描述符。

可以在任何可以选择内置日期时间格式的位置使用格式模式。

生成格式模式

1. 在数据表中创建数值列。
2. 右击该列并选择**列信息**。
3. 从格式列表中选择**格式模式**。
4. 点击**设置格式模式。**
5. 要将内置格式用作起点，请选择**内置**，选择日期时间格式，然后编辑该格式。
6. 要编辑该格式，请在 “格式模式” 字段中键入字段描述符，或从 “字段描述符” 列表中选择某个格式模式。
7. 点击**确定**。

生成输入格式模式

1. 在现有数值列中，右击并选择**列信息。**
2. 从格式列表中选择**格式模式**，然后取消选择**使用输出格式。**
3. 点击**设置格式模式**。
4. 按照之前任务所述生成模式。

注意：某些格式不能用于输入。例如，当一个长度可变的数字字段描述符后面紧跟另一个数字时，某些日期是不明确的。格式 <YYYY><M><D> 无法处理输入，因为 <M> 是单数位月份；2020111 之类的日期是不明确的。不清楚日期是 1 月 11 日还是 11 月 1 日。

国际格式

若导入或输入的数据包含特定于国家/地区标准的格式，您可能需要确保正确解释数字格式。在 Windows 上，访问 “控制面板” 中的区域和语言选项，然后选择应针对其设置数字格式的国家/地区。在 macOS上，从 Apple 菜单中选择**语言和区域 > 格式**，然后选择正确的国家/地区。在更高版本的 macOS 上，该选项可能会在 “系统首选项” > “语言和区域” 下显示。

在行状态列中存储信息

类似于为行指定行状态，您可以创建一个仅包含行状态信息的列。行状态列存储有关行是否已排除、已隐藏、已添加标签、已着色、已标记或已选定的信息。要将某列指定为行状态列，请在 “列信息” 窗口中的数据类型旁边，选择**行状态**。

图 5.4 行中的行状态和行状态列

	姓名	年龄	性别	身高	体重	行状态
1	KATIE	12	F	59	95	
2	LOUISE	12	F	61	123	
3	JANE	12	F	55	74	
4	JACLYN	12	F	66	145	
5	LILLIE	12	F	52	64	
6	TIM	12	M	60	84	
7	JAMES	12	M	61	128	
8	ROBERT	12	M	51	79	

由于行状态列存储着行状态，所以您将来可以再次应用这些状态。通过自当前行状态复制行状态或使用列公式来填充行状态列。

创建行状态列

1. 选择**列 > 新建列**。
2. 在数据类型旁边，选择**行状态**。
3. 点击**确定**。

使用新的行状态信息填充单元格，或复制行中现有的行状态信息。

使用新的行状态信息填充单元格

1. 要仅填充行状态列中的特定行，请突出显示这些行。或者，要填充列中所有行，请突出显示行状态列。
2. 右击并选择**行状态单元格**。
3. 选择要应用的行状态。

复制现有的行状态信息

1. 要仅填充行状态列中的特定行，请突出显示这些行。或者，要填充列中所有行，请突出显示行状态列。
2. 点击列面板中列名旁边的星形图标 (★)。
3. 选择以下选项之一：

从行状态复制 使用行中的行状态替换列中的行状态。

从行状态添加 将行中的行状态添加至行状态列。

复制至行状态 使用列中的行状态替换行中的行状态。

添加至行状态 将行状态列中的行状态添加至行中的行状态。

永久选择单元格

您可以像保存其他行状态特征（隐藏、排除、着色等）一样，保存行状态列中的选择。这将使单元格“永久”突出显示。

永久选择单元格

1. 右击某个单元格并选择**行状态单元格 > 选择 / 撤销选择**。
2. 为要选择的所有单元格重复该操作。
3. 要去除突出显示，右击该单元格并选择**行状态单元格 > 选择 / 撤销选择**。

初始化列中数据

第一次向数据表添加新列时，“列信息”窗口中会显示 “初始化数据”菜单。指定要显示在新列中的初始数据值的类型。初始化数据不可用于 “行状态”数据类型。

选择以下选项之一：

缺失 / 空 在列中放置缺失值，黑点 (•) 表示缺失数值数据，空格表示缺失字符数据。

常数 在该列的所有行中放置相同的数字或字符。在出现的框中输入数字或字符。输入任意数量的字符。

今天 为列中的每行放置今天的时间戳。该选项仅与**日期**或**时间**格式相关。

序列数据 基于指定的参数插入顺序数据。请参见第 263 页的 “数值或字符序列数据”。

随机 将随机生成的数据值插入列。选择用于生成随机值的方法：

随机整数 输入最小和最大整数值。生成该范围内的整数值，以便每个值都以基本相同的频率出现。

随机均匀 输入连续值的范围。生成该范围内的随机均匀值。

随机正态 输入正态分布的均值和标准差。生成来自该分布的随机正态值。

随机指示符 输入至多三个值和相应的期望比例。比例之和应等于 1。生成的值具有的分布将尽可能地接近指定的比例。

注意: 该方法以您指定的精确比例生成值，除非行数乘以其中一个比例所得的结果为小数值。

假定有 n 行，三个比例分别为 p_1、p_2 和 p_3。下面说明这些值是如何生成的：

- 每行都分配有一个 0 到 1 之间的随机均匀值。
- 这些行按照其随机均匀值排序。
- 向对应于最小的舍入 (np_1) 随机均匀值的行分配第一个值，向对应于第二小的舍入 (np_2) 随机均匀值的行分配第二个值，向其余行分配第三个值。

数值或字符序列数据

为数值数据插入顺序数据

1. 在**数据类型**旁边，确保已选定**数值型**。
2. 在**初始化数据**旁边，选择**序列数据**。
3. 在**自**框和**至**框中，指定起点和终点。
4. 在**步长**框中，指定序列。
5. （可选）在**每个值的重复次数**框中，输入您希望每个数值重复的次数。
6. 点击**确定**。

例如，若您希望列中包含 2 到 60 之间的偶数，则在**自**框中键入 **2**，在**至**框中键入 **60**，并在**步长**框中键入 **2**。

为字符数据插入顺序数据

1. 在**数据类型**旁边，确保已选定**字符型**。
2. 在**初始化数据**旁边，选择**序列数据**。
3. 在**添加**旁边的框中，输入字符数据并点击**添加**。
4. （可选）在**每个值的重复次数**框中，输入您希望每个字符值重复的次数。
5. 点击**确定**。

JMP 中的列属性

列属性是列的附加信息。一旦输入该信息，它将作为数据表的一部分被保存。您可以选择列属性并将其应用于列。某些列属性是 JMP 为了方便您创建的。通常情况下，您可以修改这些列属性。

列属性在“列信息”窗口的“列属性”菜单中列出。“列属性”菜单下的区域显示已保存至列的列属性。要查看或编辑属性，从该列表中进行选择。

在数据表的列面板中，在包含属性（不包括“注释”属性）的每个列名旁会显示一个图标。“注释”属性不会用图标表示。图标包含以下各项：

指示应用了范围或列表检查属性

指示列包含公式

指示列包含除“注释”、“范围检查”或“列表检查”列属性外的属性

有关更多列属性图标的信息，请参见“开始使用 JMP”一章中第 40 页的“表示列特征和属性的图标”。

将列属性分配给一列或多列

1. 选择要指定属性的一列或多列。

2. 执行以下操作之一：
   - 右击标题区域，选择**列属性**，然后选择属性。
   - 右击标题区域，选择**列信息**，然后从“列属性”菜单中选择属性。
   - 选择**列 > 列信息**，然后从“列属性”菜单中选择属性。
3. 在显示的列属性面板中，指定相应的值并选择合适的选项。

   点击**应用**添加列属性，或点击**确定**添加列属性并关闭列属性窗口。

一个列可能已包含您要应用到其他列的属性。使用“标准化特性”命令可将该属性应用于其他列。请参见第 286 页的“在各列之间标准化特性和属性”。

以下部分说明可添加至列的属性。

基本列属性

应用的基本列属性通常包含 “公式”和 “注释”。

公式

在列中插入公式可为该列计算值。添加公式之后，该列将被锁定，这样将无法手动编辑其数据值（防止公式失效）。

- 点击**编辑公式**可创建公式。有关创建公式的详细信息，请参见第 339 页的“在 JMP 中创建公式”一章。
- 若不希望 JMP 对公式求值，请点击**禁止计算**。
- 若不希望 JMP 对公式中的错误发出警示，请点击**忽略错误**。
- 创建公式之后：
  - 在“列信息”窗口中，公式直观地显示在右侧。不过，若公式过长，则可能仅显示部分公式。点击并拖动公式框的边界可调整其大小。
  - 从数据表中，通过点击列面板中列名旁边的 ( ) 可编辑公式。

提示：要在创建公式时不使用“列信息”窗口，请右击该列并选择**公式**。

注意

向选定列添加注释。

验证列值的属性

以下属性可帮助验证列中的值：“范围检查”、“列表检查”和“缺失值代码”。

范围检查

范围检查会对列中数据进行验证。请对列进行设置，使其仅接受指定范围内的数字。

选择要用于设置范围的公式。*x* 表示在列中输入的值，*a* 表示范围的开头，*b* 表示范围的结尾。

- a = 列接受的最低值
- b = 列接受的最高值
- 对于单边范围检查，请将 a 或 b 留空。
- 从数据表中，通过点击列面板中列名旁边的 ( ) 可修改范围检查。

列表检查

列表检查会对列中数据进行验证。对列进行设置，使其仅接受您指定的单个值。当您想要指定如何对图形中的数据排序时，列表检查很有用。

- 您可以使用多达 127 个值。
- 使用按钮可添加新值、更改或反转值的顺序，以及删除值。
- 对列设置列表检查后，光标置于单元格上方时将改为 。您试图输入验证列表中不包含的值时，将出现一条警告消息。
- 要查看可接受值的菜单，请右击某个单元格并选择**列表检查值**。您可以从菜单中选择单元格值，而不是在单元格中输入值。
- 从数据表中，通过点击列面板中列名旁边的 可修改列表检查。

缺失值代码

使用缺失值代码可指定应视为缺失值的列值。例如，有时使用值 99 作为占位符来表示缺失值，也有可能使用多个值来表示不同类型的缺失值。

注意：“缺失值代码”列属性为扩展属性。若您计划导出至 SAS 9.4 服务器或另存为 SAS 9.4 文件格式，则应在“SAS 集成”首选项页中启用扩展属性。请参见“JMP 首选项”一章中第 619 页的““SAS 集成”首选项”。或者，您应在“另存为”窗口中选择**在 SAS 9.4 扩展属性中存储表和列属性**。若未选择扩展属性，JMP 将导出缺失值代码或将其另存为缺失值。仅 SAS 9.4 支持扩展属性。早期版本的 SAS 将忽略任何扩展属性。

删除单个列的缺失值代码

1. 在数据表的列面板中该列的旁边，点击 > **缺失值代码**。
2. “缺失值代码”在列表中突出显示。点击**删除**。
3. 点击**确定**。

跨多列删除缺失值代码

1. 选择包含您要删除的缺失值代码的列。
2. 选择**列 > 标准化特性**。
3. 在 “删除属性”下，点击**列属性 > 缺失值代码**。
4. 点击**确定**。

将信息附加至列值的属性

以下列属性可将报表和图中使用的信息附加至某一列的值中：

- 值标签
- 值得分
- 值顺序
- 行顺序水平
- 值颜色
- 颜色梯度
- 短名称
- SPSS 名称
- SPSS 标签
- SAS 名称
- SAS 标签
- 长名称
- 问题

值标签

使用值标签可在数据表中显示标签而不是值。每次出现该值时都显示对应的标签。您可以通过双击单元格中的标签显示原始值。

- 在“值”框中输入要为其指定标签的值。
- 在标签框中输入要显示的标签。
- 要使用范围，请点击**允许范围**，然后指定下限值和上限值。若选定允许范围，则值标签可以是非整数。“图形生成器”可以使用值标签表示 x 和 y 轴。

  提示：要为缺失值指定标签，请为下限输入一个句点 (.)，并将上限留空。要为所有其他值指定标签，请为下限输入三个句点 (.)，并将上限留空。

- 添加、更改或删除标签。

提示：

- 要在数据表中禁用值标签，而不删除已经设置的值标签，请在“列属性”窗口中取消选择**使用值标签**。或者，要快速显示或隐藏数据表中的值标签，请右击一列，然后选定或取消选定**使用值标签**。
- 若数据表包含值标签，使用**搜索**命令将搜索实际值，而不会搜索标签。
- 若数据表包含值标签，**行编辑器**会显示该标签；但突出显示某个单元格对其进行编辑时，该单元格显示的是实际值。
- 若复制和粘贴具有值标签的单元格，粘贴的是实际值。
- 在公式中，使用值标签引用某列时，悬停在值标签上方将能查看实际数据值。

值得分

使用值得分可指示分类数据列的值得分对。值为数据值，得分是一个数字。该属性将数据值与得分建立关联（例如，列的数据值可以是“不满意”、“满意”、“非常满意”）。用户可以为“不满意”分配 0 分，为“满意”分配 50 分，为“非常满意”分配 100 分。这些得分随后将用于计算均值之类的计算目的。相关示例，请参见《消费者研究》。

使用值得分

1. 在数据表中，右击列并选择**列信息**。
2. 从**列属性**列表中，选择**值得分**。
3. 在值框中输入您要分配标签的值 （例如，“满意”）。
4. 在得分框中输入您要显示的标签 （例如，“50”）。
5. 点击**添加**、**更改**或**删除**。
6. 点击**确定**。

值顺序

“值顺序”向随后要在报表和图中使用的列值分配排序。报表和图的默认排序采用字母数字数据值排序，但以下情况除外，在以下情况下值将适当地自动排序：

- 一月、二月、三月、四月、五月、六月、七月、八月、九月、十月、十一月、十二月
- 一月、二月、三月、四月、五月、六月、七月、八月、九月、十月、十一月、十二月
- 星期日、星期一、星期二、星期三、星期四、星期五、星期六
- 极低、低、中低、中、中高、高、极高
- 完全不同意、不同意、中性、无关紧要、同意、完全同意
- 令人失望、不可接受、很差、差、较差、可接受、平均、好、更好、很好、非常好、最佳

提示：您可以将“值排序”匿名化，以便将“值排序”属性重新编号。为此，突出显示具有“值顺序”列属性的列并选择**表 > 匿名**。

当您使用“实验设计”平台构造设计时，“值顺序”列属性将自动分配给分类因子。有关分配和编辑“值顺序”列属性的示例，请参见《实验设计指南》。

注意：比较字符数据时，选择**对字符使用语言/区域比较**以便在确定顺序时考虑语言/区域信息。

定制顺序

默认情况下选择“定制顺序”。该选项允许您以用户定义的顺序对数据值排序。最初，字符数据按字母顺序排序，数值数据按数值顺序排序。您可以使用箭头移动各值并用加号添加值。

排序顺序

取消选择**定制顺序**可将值移至“排序顺序”框。然后您需要指示排序项的规则。

未定制顺序的项都受排好的顺序的影响。将各项向下拖至“排序顺序”区域，以便将其从定制顺序中移除，转而对这些项应用排序规则。撤销选择**定制顺序**将排序规则应用到所有项。

当您选择或撤销选择这些复选框时即应用排序顺序规则。

行顺序水平 按列值在数据表中出现的次数对列值排序，而不是按值排序。该选项不适用于对数据表列排序。

注意：若同时选定 “行顺序水平”和 “常用顺序”，则应用 “行顺序水平”

常用顺序 若所有值都具有预定义的排序顺序（例如，周中各天），则其余项将按该基准排序。常用顺序是通常的 ASCII 排序。该选项仅适用于字符数据。

数值顺序 通过将数据拆分为字符串和数值段，对字符列值进行排序。

以下是数值顺序规则：

- 字符串被分成多个数值段（不包括 +/-，但包括中间的点）和非数值字符。
- 各段在两个字符串之间进行比较，只要数字不以前导零开始，就按大小进行比较。

  带前导零的数字在不带前导零的任何数字之前排序。

  若两个数字都有前导零，则将数字逐位进行比较。例如，“0CH2”的顺序排在“0 CH2”之前。
- 若两个字符串的段类型不同，则数值段在字符串段之前排序。
- 语言 / 区域设置比较会影响符号；语言 / 区域设置排序中会忽略连字符、小写或大写等字符。（请注意，在 ASCII 排序中，小写的顺序排在大写之前。每种语言的规则有所不同。目标是以操作系统对文件名排序的方式进行排序。）

  要使用 ASCII 排序，请在 “值顺序”列属性中取消选择**数值排序**。

– 默认情况下，使用语言/区域比较来比较字符串段。这种比较是语言感知型的比较，该比较由操作系统提供，因此在 Windows 和 macOS 上的行为有细微差别。

值颜色

使用值颜色可以为名义型列或有序型列的值指定某种颜色或颜色主题范围。该列的值在所有适用的图形（如马赛克图和带有颜色编码图例的图）中显示时都带有指定的颜色。您还可以为数据表列中的值着色。

- 要更改特定值的颜色，请右击彩色圆圈并选择一种颜色。
- 要使用某种颜色主题，请点击值颜色列表下方的**颜色主题**，然后选择所需颜色。
- 要创建定制颜色主题，请参见第 269 页的“创建定制颜色主题”。
- 若还要对数据表中的单元格着色，请选择**按值为单元格着色**。
- 选择“宏”菜单中的选项。请参见第 269 页的“宏选项”。

创建定制颜色主题

创建定制颜色主题

1. 点击列属性中的值颜色列表下方的**颜色主题**。
2. 打开“定制颜色主题”分级显示项。
3. 点击**新建**。
4. 使用滑块创建颜色。
5. 为颜色命名并点击**保存**。
6. 点击**确定**。

下次使用值颜色属性时可以在颜色主题菜单中访问新的定制颜色。该颜色还会保存在您的首选项中。有关创建颜色主题的详细信息，请参见“输入和编辑数据”一章中第 206 页的“在图形中创建颜色主题”。

宏选项

宏菜单包含以下选项：

在端点之间渐变 设置顶部值和底部值的颜色。 JMP 对整个范围内的值应用颜色梯度。使用该命令可以在端点间为其他水平设置所有颜色。

在选定点之间渐变 设置顶部值和底部值的颜色，以便 JMP 为您在“值颜色”列表中突出显示的一系列的值应用颜色梯度。

反转颜色 自上而下或自下而上反转值的颜色。

恢复以前的颜色 将颜色重新设置为其最初的颜色值。

颜色梯度

选择颜色梯度，对图中的连续列着色。“图形生成器”、“气泡图”、“矩形树图”和“方格图”平台中支持颜色梯度。

- 若还要对数据表中的单元格着色，请选择值颜色列表上方的**按值为单元格着色**。
- 从菜单中选择一种颜色梯度。
- 输入最小值、最大值和中心值：
  - 最小值体现梯度最左侧的颜色。
  - 最大值体现梯度最右侧的颜色。
  - 中心值体现梯度中间的颜色。

注意：要在“图形生成器”中查看颜色梯度，您必须将列指定给颜色区域。要在“气泡图”和“矩形树图”中查看颜色梯度，您必须将列指定给颜色角色。

短名称、SPSS 名称和 SAS 名称

若选定“显示报表中使用的列”红色小三角选项，则这些列属性将在“分类”结构化报表中插入指定的短名称。

“SAS 名称”也是您在导入 SAS 数据时添加的列属性。它包含 SAS 数据集中变量的名称。若将数据导回 SAS，修改它将影响 SAS 数据集。

“SPSS 标签”、“长名称”、“SAS 标签”和“问题”

若选定“显示报表中使用的列”红色小三角选项，则这些列属性将在“分类”结构化报表中插入指定的长名称。

“SAS 标签”也是导入的 SAS 数据的一个列属性。它包含 SAS 数据集中列的说明。若将数据导回 SAS，修改它将影响 SAS 数据集。

控制列显示的属性

“轴”和“单位”列属性控制列值的显示方式。

轴

使用“轴”属性可以更改列的默认轴设置。当该列显示在分析中时，JMP 会自动使用您的设置。有关属性的详细信息，请参见“JMP 报表”一章中第 429 页的“定制图形中的轴和轴标签”。

添加“轴”列属性时，您可以保存所有列属性或仅保存您更改的列属性。例如，您可能希望轴从 0 开始，而不关心轴的上限。设置最小轴值并从“保存至列”列表中选择**仅限更改的属性**。

从图形内部设置列的默认轴属性

1. 创建图形。
2. 按照您的需要来更改轴设置。
3. 右击该轴并选择**保存至列属性**。

单位

使用“单位”属性可以指定收集列数据所用的测量单位。单位会在数据表中列名后的圆括号中显示，也会在图中出现的列名中显示。例如，您可能希望有一列指示以月份测量年龄值，或指示以千美元来表示货币值。

在建模和实验设计中使用的属性

当您使用“实验设计”平台构造设计时，列属性会保存到生成的设计表。但是，其中的某些列属性在常规建模情形中也会有用。要在更一般的情况下使用这些属性，您可以自行指定它们。

注意：《实验设计指南》详细说明了与实验设计关联的属性并提供了示例。此处仅提供这些属性的简要说明。

在建模和实验设计中使用以下列属性：

- 响应限
- 检测限值
- 设计角色
- 编码
- 混料
- 因子更改

响应限

“响应限”列属性为响应定义了意愿函数。“预测刻画器”和“等高线刻画器”使用意愿函数查找最优设置。请参见《实验设计指南》。

JMP PRO 检测限值

“检测限值”列属性定义无法测量的响应的界限。您可以在“广义回归”平台中使用这些限值来指定删失响应。请参见《拟合线性模型》。

设计角色

“设计角色”列属性指示如何在设计实验和用于拟合数据的模型中使用列。例如，列可以表示连续因子、分类因子和分区组因子等。请参见《实验设计指南》。

编码

“编码”列属性将线性变换应用于数值列中的数据。基于您指定的界限，数据变换为 –1 和 +1。以后，每当在“拟合模型”平台中将列输入为模型效应时，JMP 都会使用变换后的数据值。请参见《实验设计指南》。

混料

当数据表中的某个列表示混料的成分时，“混料”列属性很有用。混料成分之和被限制为一个常数。“混料”列属性将列标识为混料成分，并为该列定义编码。请参见《实验设计指南》。

因子更改

“因子更改”列属性指示在设计的实验中更改因子设置的难度。它用于创建和分析裂区、裂-裂区和双因子裂区设计。请参见《实验设计指南》。

与控制图和能力相关的属性

“规格限”、“控制限”、“Sigma”和“过程能力分布”属性与控制图和能力相关。

规格限

使用 “分布”和 “能力”平台执行能力分析时会用到规格限。请注意，在旧的数据表中，列属性可能命名为 “能力分析”。

您可以指定“下规格限”、“上规格限”或“目标”的任意组合。**显示为图形参考线**选项将您指定的规格限和目标显示为相应图中的参考线。

提示：要一次性向若干列添加规格限，请参见《质量和过程方法》。

控制限

“控制限”列属性允许您指定列的控制限，而不是让 JMP 根据列中的值计算控制限。这在将过程数据与历史控制限进行对比时很有用。

“控制限”列属性仅能分配给数值列。要分配“控制限”列属性，选择控制图类型，然后输入请求的值。若缺少以上任何条目，JMP 会在控制图中将它替换为计算值。

Sigma

使用 Sigma 属性可以输入已知的 Sigma 值。该值用于控制图之类的应用或需要使用 Sigma 值完成计算的任何应用。若未提供 Sigma 值，将从样本计算 Sigma。

过程能力分布

“过程能力分布”列属性指定列的过程分布。“过程能力分布”列属性仅用在“过程能力”平台中。另见第 274 页的“分布和过程能力分布”。

质量控制警告脚本

通过编写脚本并将其存储为名为**质量控制警告脚本**的列属性或数据表属性，您可以生成自动警告。

只要检验失败，以下质量控制警告脚本就会自动在日志中写入一条消息：

```
Write(
        "Out of Control for test ",
        qc_test,
        " in column ",
        qc_col,
        " in sample ",
        qc_sample
    )
```

控制列在平台中的使用方式的属性

以下属性控制列在平台中的使用方式：

- 分布
- 时间频率
- 地图角色
- 超类别
- 多重响应
- 目标水平
- 收益矩阵
- 信息性缺失

分布

对于包含连续数值数据的列，请使用“分布”属性选择要对列拟合的分布类型。该分布用在“分布”平台中，在特定条件下也用在“过程能力”平台中。请参见第 274 页的“分布和过程能力分布”。

为列获取“分布”报表时（通过选择**分析 > 分布**），JMP 会使用指定的分布自动估计拟合。表示拟合分布的曲线叠加在直方图上。

若同时设置“分布”属性和“规格限”属性，则“分布”平台将基于在“分布”列属性中指定的分布生成“能力分析”报表。

注意：在较旧的数据表中，可能已将“能力分析”列属性分配给某列以指定规格限。我们建议您改用“规格限”属性。

分布和过程能力分布

若您使用“过程能力”平台分析的列不包含“过程能力分布”属性，则在“分布”列属性中指定的分布将在“过程能力”平台中生成非正态拟合。若“过程能力”平台支持此分布，则使用它。但若不支持此分布，则平台将使用 Johnson 拟合。

若同时为给定的列保存了“分布”和“过程能力分布”列属性，则在“分布”平台中会使用“分布”列属性中指定的分布，在“过程能力”平台中会使用“过程能力分布”列属性中指定的分布。

时间频率

使用“时间序列”平台时，您可以为数据指定“时间频率”属性。“时间频率”属性指定报告数据的频率（如每年、每季度、每月等）。指定时间频率允许 JMP 将闰年和闰日等因素考虑在内。若未指定频率，数据将被视为等间距数值数据。

地图角色

若您创建的数据表中包含边界数据（如国家/地区、州、省或县），要在“图形生成器”中查看相应的地图，请使用“地图角色”属性。

请注意以下几点：

- 若定制边界文件位于默认定制地图目录中，则您只需要在 -Name **文件中指定地图角色**属性。
- 若定制的边界文件位于其他位置，则必须在 -Name 文件和所分析的数据表中指定“地图角色”属性。
- 包含地图角色属性的列必须包含相同的边界名称，但列名可以不同。

有关使用“地图角色”属性的示例，请参见《基本绘图》。

将 “地图角色”属性添加至 -Name 数据表

1. 右击包含边界的列，然后选择**列属性 > 地图角色**。
2. 选择**形状名称定义**。
3. 点击**确定**。
4. 保存数据表。

将 “地图角色”属性添加至所分析的数据表

注意：仅当您的定制边界文件未包含在默认定制地图目录中时才执行以下步骤。

1. 右击包含边界的列，然后选择**列属性 > 地图角色**。
2. 选择**形状名称使用**。
3. 在**地图名称数据表**旁边，点击 浏览至 -Name 地图数据表。您可以输入相对或绝对路径。
   若地图数据表位于同一文件夹中，只需输入文件名。路径包含空格时无需使用引号。
4. 在**形状定义列**旁边，输入地图数据表中值与选定列中的值匹配的列的名称。
5. 点击**确定**。
6. 保存数据表。

在“图形生成器”中生成图形并为形状区域指定修改的列后，边界即显示在图形上。

超类别

当数据集包含评级（例如，五分制评级）时，您可能想知道在这些评级的子集中响应的百分比。添加“超类别”列属性可将特定类别分组到一个类别。

仅“分类”平台中支持“超类别”。

将 “超类别”属性添加至数据列

1. 右击您要对其类别进行分组的列。
2. 选择**列属性 > 超类别**。
   列属性窗口显示 “超类别”选项 （图 5.5）。
3. 选择 “列的类别”列表中您想要分组的类别。
4. 在 “超类别名称”旁输入一个描述性名称。
   将名称留空， JMP 会根据您选择的类别命名这些超类别。
5. 点击**添加**创建超类别。
6. 点击**超类别**红色小三角并从以下选项中进行选择：

   选项 > 隐藏 隐藏报表和图形中选定超类别的数据。

   全部添加 从列中的所有类别创建一个超类别。

   添加均值和添加标准差 计算值得分的统计量。请参见《消费者研究》。
7. 点击**确定**将属性添加至列。

图 5.5 超类别配置示例

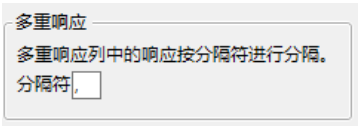
多重响应

多重响应一词是指列中单元格包含不止一个响应值的情况。例如，Consumer Preferences.jmp 样本数据表的**刷牙分隔**列中的许多单元格都包含多个值。例如，第 6 行包含“Wake, After Meal, Before Sleep”。

若想要指定逗号之外的分隔符，需添加“多重响应”列属性。也可以在“列信息”窗口中将该列的建模类型改为“多重响应”。有关“多重响应”建模类型的详细信息，请参见第 255 页的“关于建模类型”。

JMP 为保存在 JMP 12 或更低版本中的数据表自动分配“多重响应”建模类型。该列必须包含“多重响应”列属性且分隔符必须是逗号，这样自动分配才能生效。您可能会选择删除“多重响应”列属性，但 JMP 并不会这样做。

图 5.6 “多重响应配置”窗口



注意：您可以在“分类”平台中使用“多重响应”属性。请参见《消费者研究》。您还可以在“数据过滤器”中使用该属性。请参见“JMP 报表”一章中第 396 页的“数据过滤器”。若分隔符为逗号，请考虑改用“多重响应”建模类型。

目标水平

使用“目标水平”列属性指定 Logistic 回归平台的响应列的关注级别。该水平在 Logistic 回归模型中用作正响应水平，它影响统计、概率曲线和其他拟合测度。该列属性是为二值响应设计的，但它可以用于任何分类列。

注意：指定二值响应列时，“目标水平”列属性由以下平台使用：“以 X 拟合 Y (Logistic)”和“拟合模型”的“名义型 Logistic”特质。

收益矩阵

使用“收益矩阵”列属性向预测模型的名义型或有序型响应变量的水平分配权重。对于名义型响应，您可以使用概率阈值指定收益矩阵条目。

注意：以下平台使用“收益矩阵”列属性：“模型比较”、“分割”、“偏最小二乘”和“拟合模型”。“拟合模型”中使用“收益矩阵”列属性的特质包括“广义回归”、“名义型 Logistic”和“有序型 Logistic”。

收益矩阵

选择**列属性 > 收益矩阵**后会显示一个矩阵模板，选定列中的每个值都对应一行和一列。实际水平显示为行，预测水平显示为列。正确的决策是位于对角线上的决策，此处的预测水平等于实际水平。

- 对于对角线条目，输入反映正确决策的收益或权重的值。
- 对于非对角线条目，输入反映错误决策的收益（或损失）或权重的值。
- 对于未作出预测的情形，使用“未决策”列指示关联的收益或损失。

指定收益矩阵的概率阈值

若响应为二值响应，收益矩阵模板下方还会出现其他选项。这些选项支持您指定概率阈值，而不是直接在收益矩阵中输入权重。

指定“目标水平”和“概率阈值”。然后点击“设置”更新收益矩阵。

目标水平 为其建模概率的水平。

概率阈值 目标水平概率的阈值。若某观测落在目标水平中的概率超过概率阈值，该观测将被分类到该水平中。

设置 在收益矩阵模板中输入反映您所指定的“目标”和“概率阈值”的值。请参见第 277 页的“概率阈值计算”。

概率阈值计算

用 *t* 表示阈值概率。点击“设置”时，收益矩阵中的条目按如下方式指定：

- 0 表示每个对角线条目，反映正确决策无损失
- -1 表示实际值非目标水平时所预测的目标水平
- *-t/(1 - t)* 表示实际值为目标水平时所预测的非目标水平

这意味着用于分类到两个水平中的收益计算如下：

$$\text{目标水平收益} = \left(\frac{-t}{1-t}\right)\text{概率}[\text{非目标水平}]$$

$$\text{非目标水平收益} = -\text{概率}[\text{目标水平}]$$

“最佳收益预测”是具有这两值中较大收益的水平。从上面这两个收益等式可以推断出：只要概率[目标水平] 至少为 *t*，观测即分配给目标水平。

收益矩阵和预测模型

对于具有“收益矩阵”列属性的名义型或有序型列，多数建模平台都支持您保存反映收益矩阵条目的公式列。拟合模型，然后选择“保存预测公式”或“保存概率公式”选项。除了将常用预测公式保存到数据表，JMP 还会保存以下相应的常用公式列：

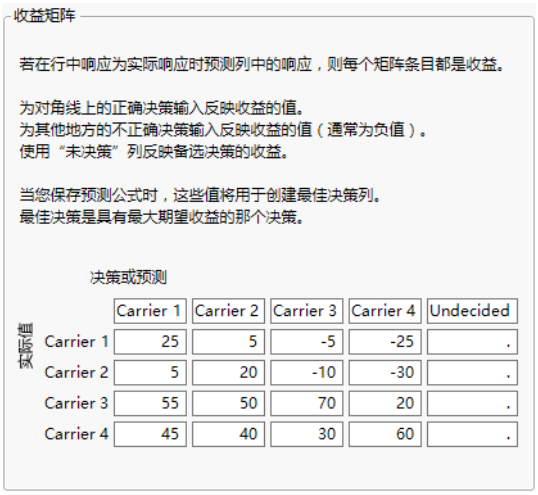
- **<水平> 的收益**：对于每个响应水平，会有一列给出用于将每个观测分类到该水平中的期望收益。
- **<列名> 的最佳收益预测**：对于每个观测，给出具有最高期望收益的响应的水平。
- **<列名> 的期望收益**：对于每个观测，给出“最佳收益预测”列定义的分类的期望收益。
- **<列名> 的实际收益**：对于每个观测，给出用于将该观测分类到“最佳收益预测”列所指定的水平中的实际收益。

请参见第 278 页的“两个以上水平的收益矩阵的示例”。有关在建模中使用收益矩阵的示例，请参见《预测和专业建模》。

两个以上水平的收益矩阵的示例

下例显示了 Travel Costs.jmp 样本数据表中的**航空公司**列的收益矩阵。

图 5.7 “收益矩阵”窗口的示例


收益矩阵

若在行中响应为实际响应时预测列中的响应，则每个矩阵条目都是收益。

为对角线上的正确决策输入反映收益的值。
为其他地方的不正确决策输入反映收益的值（通常为负值）。
使用“未决策”列反映备选决策的收益。

当您保存预测公式时，这些值将用于创建最佳决策列。
最佳决策是具有最大期望收益的那个决策。

决策或预测

实际值	Carrier 1	Carrier 2	Carrier 3	Carrier 4	Undecided
Carrier 1	25	5	-5	-25	.
Carrier 2	5	20	-10	-30	.
Carrier 3	55	50	70	20	.
Carrier 4	45	40	30	60	.

要查看如何在该收益矩阵中赋值，请考虑一个旅行社的情形：该旅行社通过四个航空公司（Carrier 1 到 Carrier 4）来为其客户提供服务。对于售出的每张机票，该旅行社都根据客户选择的航空公司来赚取利润。若旅行社推荐或**预测**了某家航空公司，它需为预订机票支付一小笔预订费。若客户决定乘坐预测的航空公司的航班，旅行社的收益中将扣除这笔预订费。但是，

若客户决定乘坐另一家航空公司的航班，旅行社就会损失预订费，还必须支付另一笔预订费。旅行社的收益因为预测不准而降低。

假定 Carrier 1 到 Carrier 4 的预订费分别为 $15、$20、$30 和 $50，售票收益分别为 $40、$40、$100 和 $110。

若旅行社向客户推荐 Carrier 1 且客户随后决定购票，则旅行社在支付了 $15 预订费后又收到了 $40，纯利为 $25。若旅行社预测客户选择 Carrier 4 但客户改选了 Carrier 1，则旅行社会损失向 Carrier 4 支付的 $50 预订费，还必须向 Carrier 1 支付 $15 的预订费。这样旅行社的净损失为 $40 - $50 - $15 = -$25。

示例：使用概率阈值定义收益矩阵

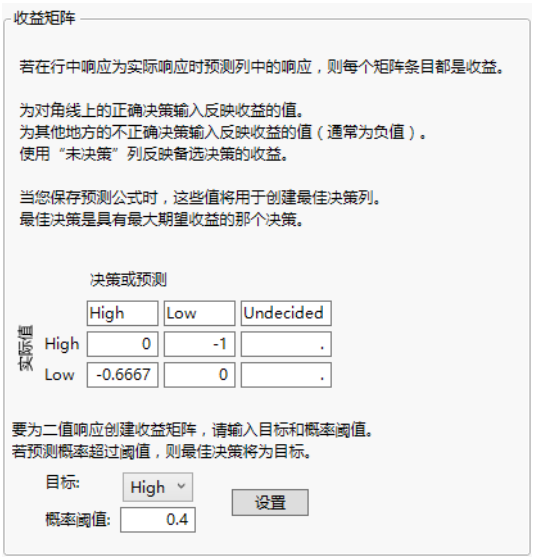
样本数据表 Liver Cancer.jmp 包含 136 位患者的疾病**严重性**等级。您关注的是如何使用从 BMI 到**黄疸**这些列中给出的预测变量对“严重性”建模。通常的模型预测公式会将患者分类到最可能的**严重性**水平中。不过，与将实际上严重性“低”的患者归入严重性“高”的患者的错误相比，将实际上严重性“高”的患者归入严重性“低”的患者的错误为代价高的错误。因此，对于患者病情严重性实际很高的情况，您想要为将这些患者误分类为“低”的情况分配更高的成本。

您可以通过设置概率阈值来分配这一较高成本。借助专家的输入，您确定以下策略为良好策略：将处于“高”水平的预测概率超过 0.4 的所有患者分类到**严重性**“高”的水平中。

1. 选择**帮助** > **样本数据库**，然后打开 Liver Cancer.jmp。
2. 选择 “严重性”列并选择**列** > **列信息**。
3. 选择 “收益矩阵”列属性。
4. 将 “目标”改为 High。
5. 输入 0.4 作为**概率阈值**。
6. 点击**设置**。

   收益矩阵随之更新以显示相应的权重。

图 5.8 显示与概率阈值对应的权重的收益矩阵



收益矩阵显示：将严重性 “高”的患者误分类为严重性 “低”的患者的损失为 -1 ；而将严重性 “低”的患者误分类为严重性 “高”的患者的损失较小，为 -0.6667。

信息性缺失

“信息性缺失”列属性指示大多数拟合平台为包含缺失值的列使用编码系统。对于连续列，编码系统包含两列。第一列为原始值列，其缺失值用非缺失值的均值替换；第二列为指示符列，它指示哪些行缺失。对于分类列，缺失值被视为列的非重复水平。

SAS 属性

在您导入 SAS 数据集时，会将 SAS 标签、SAS 名称、SAS 格式和 SAS 输入格式分配给列。若将数据导回 SAS，您可以更改列属性中的值以控制在 SAS 数据集中创建的列的标签、名称、格式或输入格式。

链接数据表与虚拟连接属性

虚拟连接将一个主数据表链接到一个或多个辅助数据表。该功能支持主数据表访问辅助数据表中的数据，而不必物理连接这些表。请参见“重塑数据”一章中第 324 页的“虚拟连接数据表”。

可通过 “链接 ID”和 “链接引用”列属性进行链接。

注意：

- “链接 ID”列与“链接引用”列的数据类型必须匹配。
- “链接 ID”与“链接引用”对于“行状态”和“表达式”数据类型无效。

- 请考虑在添加“链接引用”列属性之前添加“链接 ID”列属性，以便主表中的引用列可链接到某列。

链接 ID

“链接 ID”列属性将辅助数据表中的某列标记为 ID 列。也就是说，该数据表的各行通过该 ID 列的值唯一标识。包含“链接 ID”列属性的数据表称为**引用**的数据表。

要添加“链接 ID”列属性，请先选择该列，再选择**列 > 列信息**，然后从“列属性”列表中选择**链接 ID**。确保已选中“链接 ID”复选框，然后点击**确定**。选中的复选框指示该列是数据表的 ID 列。

链接引用

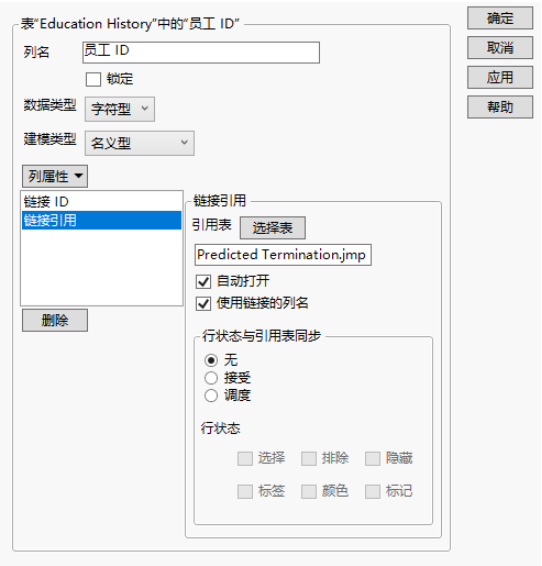
“链接引用”列属性将主数据表中的某列映射到引用数据表中的 ID 列。包含链接引用的列称为**引用**列。引用列可通过 ID 列查找辅助数据表中的数据。

要添加“链接引用”列属性，请先选择该列，再选择**列 > 列信息**，然后从“列属性”列表中选择**链接引用**。输入主表的路径，然后点击**确定**。若您输入的路径正确，引用列会在您点击“确定”后显示在主表的列面板中。

注意：

- 该路径区分大小写。
- 考虑将数据表放入同一文件夹。然后您可以省略“链接引用”属性中的目录名称，仅包含数据表名称。若不这样做，则包含数据表的完整路径（如 c:/users/marie/My Data.jmp）。
- 您还可以通过在“链接引用”路径中使用 JMP 路径变量来提高概率。图 5.9 显示了一个示例。在路径中输入一个正斜杠，将数据表与 Windows 和 macOS 用户共享。有关路径变量的详细信息，请参见《Scripting Guide》。
- 您可以通过右击某列添加“链接引用”属性来避免输入该路径。请参见“重塑数据”一章中第 325 页的“虚拟连接列的示例”。
- 选择“行状态同步”选项以同步包含虚拟连接列的数据表中的行状态。请参见第 328 页的“示例：虚拟连接的列和显示行状态”。
- 选择“自动打开”以便在打开主数据表时自动打开已链接的表。请参见“重塑数据”一章中第 335 页的“更新数据表”。
- 您可以指定要用在虚拟连接列中的链接列名。该选项支持您指定较短的列名（例如，**奶酪**而不是**奶酪[选择]**）。在“链接引用”列属性中，选择**使用链接的列名**。

图 5.9 “链接引用”列属性定义



其他列属性

共有三个附加属性：“表达式角色”、“事件处理程序”和“其他”。

表达式角色

导入包含图像数据的数据列时，数据类型设置为“表达式”以便将数据显示为图片。或者您可以在“列信息”窗口中分配“表达式”数据类型。

要定制图像在数据表单元格中的显示方式，或是将矩阵和表达式显示为图像，请添加“表达式角色”列属性。默认设置为“图片”。若“表达式角色”设置为“图片”且表达式为图片，则表达式的求值结果为图片且图片显示在数据表中。否则，表达式显示为 JSL 表达式。更改该设置可更改表达式的使用方式。

注意：若表达式列包含的图像大小超过 32KB，当它保存至 Microsoft Excel 时会被截断。当导回 JMP 时，这些表达式列不会恢复。

从以下选项中进行选择：

图片 若表达式包含图片数据，列会求值并显示图片。

矩阵 若表达式包含矩阵，列会求值并显示矩阵。否则，数据会显示为表达式。

表达式 若表达式仅仅是表达式，使用该选项可将其显示在列中。

您想要使用可能包含图片数据的表达式，但想阻止 JMP 显示该图像，这时可以使用 “表达式”。

图片表达式 要定制图像在数据表单元格中的显示方式，请在从列表中选择 “图片”后选择以下选项之一。

最大宽度和高度 指定数据表单元格中的图像的最大宽度和高度。

拉伸 放大小图像以填充数据表单元格。

保持纵横比 保持图像在数据表单元格中的宽高比。默认选定该选项。

为数据表中的小图像添加装饰框 在数据表中的小图像四周显示斜切边缘。

将图像拖到表达式列中

1. 在数据表中创建新列。
2. 右击该列，选择**列信息**，然后选择**表达式**数据类型。
3. 右击该列，选择**列信息**，然后选择**表达式角色**。
4. 按如上所述选择选项。
5. 导航至您要包含在数据表中的图像所在的网站。您还可以使用您计算机上的图像。
6. 选择图像并将其拖到 “表达式”列的空单元格中。

   调整单元格大小使图像变大。

   SAS Offices.jmp 和 Big Class Families.jmp 样本数据表显示了表达式列的示例。

提示：

- 您还可以导入包含图像列的数据以创建表达式列。
- 有关您可以对表达式列中的图像执行的操作的详细信息，请参见第 423 页的“将图像用作标记”和“JMP 报表”一章中第 487 页的“将数据表中的图像添加到悬停标签”。

使用 “汇总”添加表达式列

1. 选择**帮助** > **样本数据库**，然后打开 CrimeData.jmp。
2. 选择**表** > **汇总**。
3. 选择**州**并点击**分组**。
4. 选择**总犯罪率**并点击**统计量** > **直方图**。
5. 点击**确定**。

   一个新数据表随即显示，其中包含带直方图图像的新表达式列。调整行大小使图像变大。

   有关 “汇总”的详细信息，请参见第 8 章，“汇总数据”。
6. 右击**直方图 （总犯罪率）**列并选择**列信息**。

   请注意，列的数据类型设置为 “表达式”。
7. 从 “列属性”菜单中选择**表达式角色**。

   “表达式角色”自动设置为 “图片”。在很少见的情况下，需要将图片显示为 “表达式”，此时将设置更改为 “表达式”。

8. 点击**确定**。

“表达式角色”分配给 “表达式”列。

事件处理程序

使用“事件处理程序”列属性可在文本列中定义超链接。这些超链接可用于打开网页和其他文档。

向 URL 列添加“事件处理程序”以便在浏览器中打开 URL。您还可以修改针对该列属性的占位符脚本以打开不同类型的资源。例如，在地理数据数据表中，您可以编辑“点击”脚本以便在用户点击数据表中的超链接时显示 Google 地图。SAS Offices.jmp 样本数据表提供了相关示例。

有关编辑脚本的详细信息，请参见《Scripting Guide》。

其他

使用“其他”列属性可创建您自己的列属性并为其指定您选择的任何名称。该属性随后可用于 JSL 编程。

1. 右击该列并选择**列属性** > **其他**。
2. 为新属性输入名称。
3. 为该属性输入值。

由 JMP 分配并控制的属性

为了控制平台之间信息流动的方式，JMP 将分配一些您无法控制的属性。这些属性不会显示在“列属性”菜单中。

响应概率

在“刻画器”报表中，“响应概率”属性使分类响应变量的所有水平显示在单独一行中。当某些概率公式保存至数据表时，JMP 会自动分配“响应概率”属性。

按以下步骤保存该属性：

1. 使用“拟合模型”平台拟合 Logistic 回归模型。
2. 点击报表红色小三角并选择**保存概率公式**。

JMP 会为新的概率列自动指定 “响应概率”属性。

有关预测刻画器的详细信息，请参见《刻画器指南》。

展开公式

在“刻画器”平台和“非线性”平台中，若模型的成分列可能包含公式，JMP 将替换内部公式，只要它引用其他列。添加“展开公式”属性以防止成分列展开。分配值 0。

请参见《刻画器指南》。

预测

当您使用模型拟合连续响应时，JMP 将自动分配“预测”列属性，并保存预测公式或预测值。“预测”列属性会标识用于创建预测公式的平台。该平台在“模型比较”平台中列为“创建方式”。

信息性缺失项

在默认情况下或若您请求信息性缺失拟合时，某些 JMP 平台中将分类变量的缺失水平视为信息性缺失。在这些情况下，JMP 将“信息缺失项”列属性自动分配至包含分类变量的预测公式列。该列属性确保在涉及预测公式的图和分析中将缺失值类别视为分类变量的非重复水平。尤其是将缺失值当作一个水平显示的刻画器图。

JMP PRO 列 ID

JMP 会向用于拆分初始 Bootstrap 结果数据表的列自动分配“列 ID”列属性。该列属性协助拆分操作，并确保拆分 Bootstrap 结果表中的列名有意义。

约束状态

选择“保存约束”时，每个线性约束的系数会显示在数据表的列中。每个约束列都分配有“约束状态”列属性。该属性指定定义约束的不等式方向。从设计平台选择“加载约束”时，“约束状态”列属性会告知 JMP 不等式的方向。

每区组试验次数

“每区组试验次数”属性指示每个区组中允许的最大试验次数。“评估设计”平台和“扩充设计”平台使用该属性来指示因子的分区组结构。

自由度

许多建模平台都向公式列添加了“自由度”列属性，用来计算预测值的标准误差。在同时提供预测公式列和标准误差公式列时，“预测刻画器”使用该列属性的值来绘制置信区间。

警告：我们不建议您删除或更改该属性的值。这样做可能对“预测刻画器”中的置信区间产生负面影响。

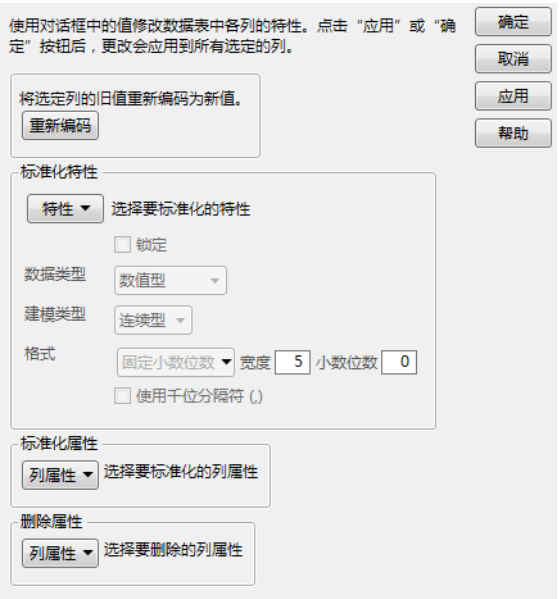
在各列之间标准化特性和属性

某列可能包含您希望其他列也具有的特性（数据类型、建模类型、数值格式等）或属性（公式、注释、列表和范围检查等）。您可以使用现有列在各列之间对这些特性和属性进行**标准化**。这包括添加和删除特性和属性。

将现有列的特性和属性应用到多个列

1. 选择包含所需特性或属性的一个或多个列。
2. 选择**列 > 标准化特性**。

图 5.10 在各列之间标准化特性



对列值重新编码

若需要以相同方式对多个列中的相似值重新编码，可以使用“重新编码”选项。点击**重新编码**后，将基于所有选定列的集合生成显示的值。例如，假定您有两列用户响应。一列包含值“同意”和“不同意”。另一列包含值“同意”、“不同意”、“不确定”和“完全不同意”。您想要将两列中的所有值简化为 A、D、U 和 SD。

注意：若想仅对单个列中的值重新编码，还可以使用“列”>“重新编码”选项。请参见“输入和编辑数据”一章中第 230 页的“对列中数据重新编码”。

标准化列特性

默认情况下，“标准化特性”面板中的项灰显。要访问某项，请点击**特性**按钮并选择要在各列之间重复的项。

注意：输入格式项仅适用于“日期”、“时间”和“持续时间”格式。

要更改任意特性的值，请使用“标准化特性”面板中的菜单。

标准化列属性

在各列之间标准化属性

1. 选择要标准化的列。其中一列必须包含您要添加至其他列的属性。
2. 选择**列 > 标准化特性**。
3. 在 “标准化属性”区域中选择**列属性**。
4. 选择要标准化的属性。若需要，修改属性设置。
5. 点击**应用**在选定的列之间标准化属性。更改会在数据表、打开的报表和打开的图形中显示。
   或点击**确定**在选定的列之间标准化属性并关闭列属性窗口。

删除列属性

删除多列之间相同的属性

1. 选择要删除的特性或属性所在的列。
2. 选择**列 > 标准化特性**。
3. 在 “删除属性”下方，点击**列属性**，然后选择要删除的属性。
4. 点击**确定**。

注意：确保您点击的是“确定”而不是“移除”。点击“移除”只会从列表中移除该属性，而不会删除它。

关于公式标准化的示例

若对某列应用了公式，又想将同一公式应用到数据表中的其他列，请使用**替换列引用**选项。

注意：该选项取决于原始公式中所引用的列的位置。例如，若原始公式基于前一列，则对其他列应用的任何其他公式都基于它们的前一列。

例如，Blood Pressure.jmp 样本数据表包含在三天之内一天三次对五个被测者测量的血压值。您需要查找每一血压 (BP) 列的对数。

1. 选择**帮助** > **样本数据库**，然后打开 Blood Pressure.jmp。

创建九个新列，每个现有的 BP 列对应一个新列。

2. 选择**列** > **新建列**。
3. 添加九列。
4. 点击**确定**。

应用您的原始公式：

5. 右击**列 12 1** 并选择**公式**。
6. 选择 BP 8M。
7. 选择**超越** > Log。
8. 点击**确定**。

   列 12 1 现在包含 BP 8M 列的对数。您希望剩余的空列包含其余 BP 列的对数。
9. 在数据表中，选择您创建的所有新列，包括含有原始公式的列 （列 12 1 一直到 12 9）。
10. 选择**列** > **标准化特性**。
11. 在 “标准化属性”面板中，点击**列属性**并选择**公式**。
12. 选中**替换列引用**旁边的复选框。
13. 保持选定 BP 8M。

    您希望在后续的列公式中取代这一列引用。
14. 点击**确定**。

    现在所有新列都会按照它们的显示顺序依次填充各个 BP 列的对数。**列 12 1** 包含 BP 8M 的对数，**列 12 2** 包含 BP 12M 的对数，依此类推。

请参见“在 JMP 中创建公式”一章中第 355 页的“复制和重复公式”。

为列指定预选分析角色

您可以为选定的列指定一个分析角色，如 *x*、*y*、权重或频数，并将该角色随数据表保存。完成指定后运行分析时，JMP 会使用该预选角色自动填充窗口中的角色框。之后每次运行分析时就不必再指定这些角色。例如，您可能希望在每次分析该数据表时都有一个名为**身高**的列来充当 *x* 角色。为使 *x* 角色生效，需要为该列指定预选的 *x* 角色。

选择频数时，该列中的值是 JMP 用作观测频数的值。若 *n* 是给定行的频数变量的值，那么该行在计算中使用 *n* 次。若该值小于 1 或缺失，JMP 则不会用它计算任何分析。

选择权重时，该列中的值为数据表中的每个观测提供权重。该变量不一定是一个整数，但只有在该值大于零时才会包括在分析中。

JMP PRO 选择“验证”后，该列将在支持“验证”列角色的平台中指定为“验证”列。

为列指定预选角色

1. 突出显示该列。
2. 选择**列** > **预选择角色**。
3. 选择某个角色：**无角色**、 **X**、 **Y**、**权重**、**频数**或**验证**。

选择适当的角色后，列面板中的图标会表明所指定的角色。点击图标可访问角色列表，供您选择不同的角色。请参见“开始使用 JMP”一章中第 40 页的“表示列特征和属性的图标”。

第 6 章

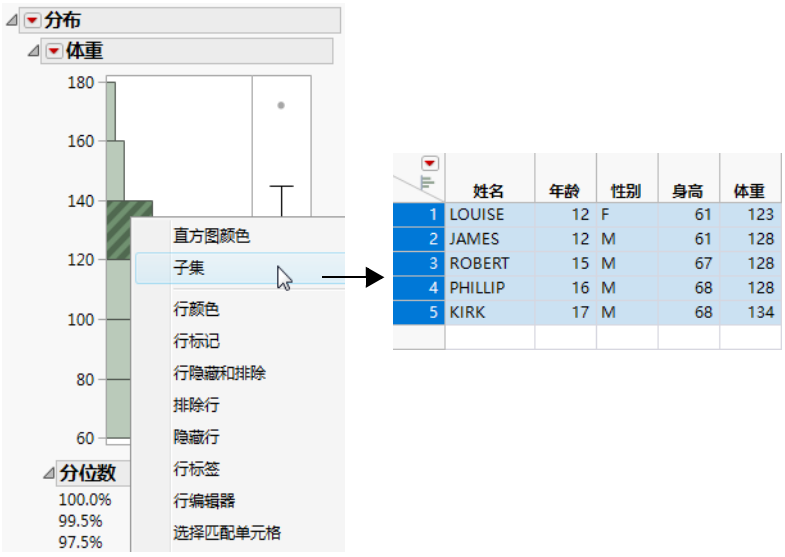
重塑数据

创建子集、数据排序等

本章介绍您可以对 JMP 数据执行的以下任务：

- 基于行列子集创建新数据表
- 按任意数量的列排序
- 将多列堆叠为一列
- 将一列拆分为两列或多列
- 转置行和列
- 多表两端拼接
- 并排连接两个表
- 虚拟连接表而不实际连接它们
- 使用另一表中的值更新某一表中的列
- 数据匿名化

图 6.1 从报表创建子集数据表



目录

创建子集数据表 293
分层的子集 294
从报表创建子集数据表 295
数据表排序 296
堆叠列 298
示例：将多列堆叠为一列 300
示例：将多列堆叠为多列 301
拆分列 302
拆分列的示例 304
转置行和列 306
行和列转置的示例 308
拼接数据表 309
拼接数据表的示例 310
拼接数据表和表变量的示例 311
连接数据表 312
连接数据表的示例 314
使用 JMP 查询生成器查询和连接数据表 321
关于指向查询中数据表的链接 324
虚拟连接数据表 324
虚拟连接列的示例 325
示例：虚拟连接的列和显示行状态 328
示例：合并虚拟连接的数据表中的引用数据 330
示例：单个列的链接 ID 和链接引用 332
更新数据表 335
更新数据表的示例 337
匿名数据 338

创建子集数据表

您可以将活动数据表中的所有行列的一个子集，仅突出显示的行和列，或者随机选定的行生成新的数据表。

创建子集

1. 选择**表** > **子集**。

图 6.2 “子集”窗口

2. 指定要生成子集的内容。选择以下内容的任意组合：
   - 取子集依据（选定列中的水平）
   - 行（所有行、选定行或随机行）
   - 列（所有列或选定列）

   在图 6.2 中，**性别**列将是子集。
3. 使用附加选项进一步定制您的子集表。
4. 点击**确定**创建子集表。

子集选项

取子集依据 按列的水平取子集。选择**取子集依据**，然后选择您要为该子集分类依据的列。

请考虑可能会创建许多新数据表这一事实。针对您在子集窗口中指定的列的每个水平都会显示一个新数据表。

所有行 创建包含活动表中所有行的子集表。

选定行 创建仅包含活动表中的选定行的子集表。默认选定该选项。

随机 - 抽样率 创建子集表，其中包含从活动数据表中随机抽取的一定比例的数据。在该文本框中输入所需的抽样率。例如，若要在新表中包含随机抽取的 50% 的数据，请在该文本框中输入 0.5。

随机 - 样本大小 创建子集表，其中的数据是活动数据表的随机样本。在该文本框中输入样本大小。例如，若要在新表中包含 16 个随机行，请在该文本框中输入 16。

若选择的随机样本是整个源表，结果将为随机排列行顺序的数据表。若指定对列分层，将生成对每组中的每行随机排列的数据表。请参见第 294 页的 “分层的子集”。

所有列 创建包含活动表中所有列的子集表。默认选定该选项。

选定列 创建仅包含活动表中的选定列的子集表。

保留取子集依据列 在输出数据表中保留取子集所依据的列。

输出表名称 指定子集表的名称。

链接至原始数据表 将子集表链接到原始表。在一个表中更改值后，另一个表会随之更新。

复制公式 在输出列中包含原始表中的公式。包括公式计算所需的所有列。默认选定该选项。

禁止公式计算 创建新表时阻止 JMP 计算列公式。默认选定该选项。

保存默认选项 保存您的当前设置。

注意：保存默认选项只保存针对 “选定行”、“选定列”、“链接至原始数据表”、“复制公式”和 “禁止公式计算”的设置。

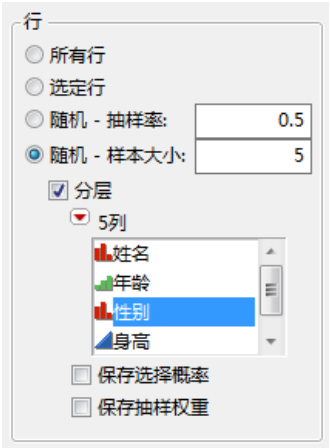
保持对话框打开 点击 “确定”后保持 “子集”窗口打开。

将脚本保存至源表 将脚本保存到原始数据表，这允许您使用相同的设置再次对数据取子集。

分层的子集

若指定了样本大小并添加了分层列，样本大小代表的是每层的大小，而不是整个子集的大小。

图 6.3 分层的子集



对于使用指定样本大小的分层随机样本，可以保存两列：选择概率和抽样权重。选中相应的复选框以保存这两列。

从报表创建子集数据表

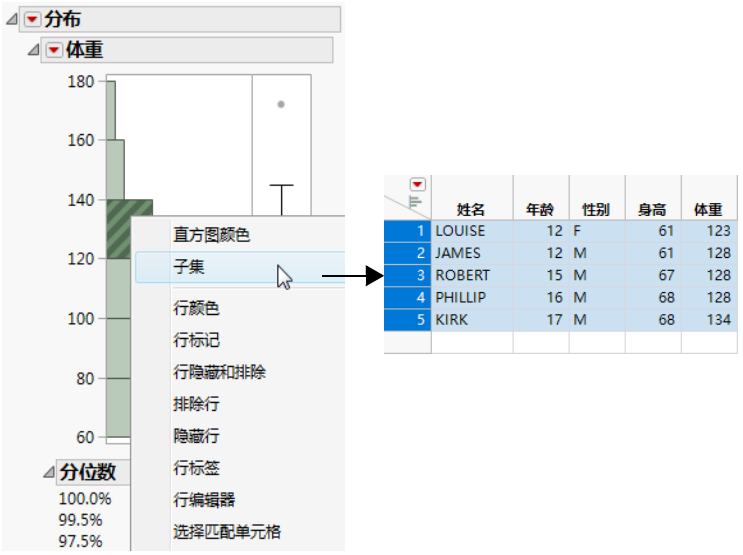
直方图和 Pareto 图生成数据表的链接的子集。

使用直方图

一旦生成包含直方图的输出（通过选择**分析 > 分布**），即可使用该直方图来创建新的数据表。新的数据表将包含直方图的突出显示直条中的数据。

要创建子集，请双击突出显示的直条。或者，右击直方图中的任意位置，然后从菜单中选择**子集**。子集表随即显示。

图 6.4 从直方图创建的子集



注意：子集数据表包含数据的视图而非实际数据。若您需要副本，必须明确保存该表。请注意，保存的数据表的副本不再链接到源表。

使用 Pareto 图

一旦生成包含 Pareto 图的输出（通过选择**分析 > 质量和过程 > Pareto 图**），即可使用该 Pareto 图来创建新的数据表。新的数据表将包含 Pareto 图的突出显示直条中的数据。要创建子集，请双击突出显示的直条。

数据表排序

您可以按列的升序或降序对 JMP 数据表排序。默认情况下，按升序对列排序。您可以创建包含排序值的新表，也可以用排序后的表替换原始表。

若列包含值标签，排序将基于实际数据值而不是值标签。（请参见“设置 JMP 列属性”一章中第 266 页的“值标签”。）但是，值标签显示在经过排序的数据表中。

若排序的列使用“值顺序”属性或“列表检查”属性，该列将依据该顺序排序。

对数据表排序的示例

1. 选择**帮助 > 样本数据库**，然后打开 Popcorn.jmp。
2. 选择**表 > 排序**。

图 6.5 “排序”窗口

3. 突出显示要作为排序依据的列的名称。对于本示例，请选择**爆米花**和**产量**。
4. 点击**依据**将这些列添加至排序列表。

   各列添加到列表的先后顺序决定排序的先后顺序。列表中的第一列是主排序字段。之后的每个变量都在排序列表中的前一个变量范围内进行排序。您可以在“分组依据”列表中拖放以更改排序顺序。
5. 使用附加选项进一步定制您的排序。对于本示例，请突出显示**产量**并点击降序按钮 。
6. 在**输出表名称**旁边的框中输入排序后的新表的名称。对于本示例，请输入 sorted popcorn。

图 6.6 完成的排序窗口

7. 点击**确定**。

图 6.7 按升序和降序排序

	爆米花	油量	包型	产量	试验
1	gourmet	lots	small	18.0	1
2	gourmet	lots	small	16.0	2
3	gourmet	little	small	15.9	2
4	gourmet	little	small	12.1	1
5	gourmet	lots	large	9.8	2
6	gourmet	lots	large	9.2	1
7	gourmet	little	large	8.6	1
8	gourmet	little	large	8.2	2
9	plain	lots	small	10.6	1
10	plain	lots	large	10.4	1
11	plain	little	small	10.1	2
12	plain	little	small	9.9	1
13	plain	little	large	8.8	2
14	plain	lots	large	8.8	2
15	plain	little	large	8.2	1
16	plain	lots	small	7.4	2

排序选项

“选择列”过滤器菜单 包含用于搜索和过滤列的选项。请参见 “开始使用 JMP”一章中第 50 页的 “‘列过滤器’菜单”。

替换表 使用排序后的表替换原始数据表，而非使用排序后的值创建新表。若存在从原始表生成的任何打开的报表窗口，该选项将不可用。

输出表名称 （可选）指定排序后的表的名称。

保持对话框打开 点击**确定**后保持 “排序”窗口打开。

将脚本保存至源表 将脚本保存到原始数据表，这允许您使用相同的设置对数据再次排序。

依据 添加排序要依据的列。各列添加到列表的先后顺序决定排序的先后顺序。列表中的第一列是主排序字段。之后的每个变量都在排序列表中的前一个变量范围内进行排序。

删除 删除所有突出显示的列。

升序按钮和降序按钮 ( ) 您可以更改分组变量的值的升序或降序列表顺序。在 “依据”变量列表中，选择一个变量，然后点击相应的升序或降序按钮。该变量旁边的图标随之更改以指示排序顺序。

保存默认选项 将当前设置另存为新的默认选项。

堆叠列

可通过将两列或多列堆叠为一个新列并保持其他列中的值来重新排列数据表。或者，也可以将一组列堆叠为多个组。第 299 页的 “堆叠选项” 对您可用来堆叠列的各种方法进行了说明。

要堆叠列，请执行以下步骤：

1. 选择**表** > **堆叠**。

图 6.8 “堆叠”窗口

2. 突出显示要堆叠的列的名称，然后点击**堆叠列**。
3. 使用附加选项进一步定制您的堆叠。
4. 点击**确定**。

堆叠选项

“选择列”过滤器菜单 包含用于搜索和过滤列的选项。请参见 “开始使用 JMP”一章中第 50 页的 ““列过滤器”菜单”。

多序列堆叠 将选定的列堆叠为两个或更多列。通过在序列数框中输入数字，指定要将选定列堆叠到几列中。该框将在选中**多序列堆叠**旁边的框时出现。

若序列由相邻的列构成，请选择**邻近**选项。

请参见第 301 页的 “示例：将多列堆叠为多列”。

注意：您向右框添加列时采用的顺序决定它们所属的组。

按行堆叠 按行堆叠列。取消选择该选项可以将一个列堆叠在另一个列下面。默认选定该选项。

剔除缺失行 从新表中剔除缺失数据。若同时选中了按行堆叠，则只会剔除**所有**数据都缺失的行。

非堆叠列 在新数据表中包括或删除非堆叠列。选择以下选项之一：

全部保留 在新表中包括原始表中的所有非堆叠列。默认选定该选项。

全部删除 忽略新表中的非堆叠列。

选择 选择要在新表中包括或删除的非堆叠列。

保持对话框打开 点击**确定**后保持 “堆叠”窗口打开。

将脚本保存至源表 将脚本保存到原始数据表，这允许您使用相同的设置再次堆叠数据。

堆叠列 添加要堆叠的列。

删除 删除所有突出显示的列。

输出表名称 （可选）指定新表的名称。

堆叠数据列 将名称分配给将包含堆叠列数据的列。默认名称为**数据**。若不希望该列出现在新表中，请将该框保留为空。

源标签列 将名称分配给将包含原始表的列名称的列。默认名称为**标签**。若不希望该列出现在新表中，请将该框保留为空。

复制公式 在输出列中包含原始表中的公式。默认选定该选项。

禁止公式计算 创建新表时阻止 JMP 计算列公式。默认选定该选项。

移动列 指定堆叠列移至的位置。“至最后”是默认值。

示例：将多列堆叠为一列

研究人员的数据表中有两列代表产量，现在他们希望将这两列堆叠为一列。（这一新列默认为**数据**。）

1. 选择**帮助** > **样本数据库**，然后打开 Popcorn Trials.jmp。
2. 选择**表** > **堆叠**。
3. 选择**产量 1** 和**产量 2**，然后点击**堆叠列**。
4. 点击**确定**。

图 6.9 堆叠的数据表

	爆米花	油量	包型	标签	数据
1	plain	little	large	产量1	8.2
2	plain	little	large	产量2	8.8
3	gourmet	little	large	产量1	8.6
4	gourmet	little	large	产量2	8.2
5	plain	lots	large	产量1	10.4
6	plain	lots	large	产量2	8.8
7	gourmet	lots	large	产量1	9.2
8	gourmet	lots	large	产量2	9.8
9	plain	little	small	产量1	9.9
10	plain	little	small	产量2	10.1
11	gourmet	little	small	产量1	12.1
12	gourmet	little	small	产量2	15.9
13	plain	lots	small	产量1	10.6
14	plain	lots	small	产量2	7.4
15	gourmet	lots	small	产量1	18.0
16	gourmet	lots	small	产量2	16.0

标签列代表标识数据来源的源标签列。该列的值是堆叠值源自的原始表中的列名。

示例：将多列堆叠为多列

假定研究人员具有血压记录数据。血压记录包含星期一、星期三和星期五这三天的数据。分别是在这三天中的 8am、12pm 和 6pm 记录的。

1. 选择**帮助** > **样本数据库**，然后打开 Blood Pressure.jmp。

   每个 BP （血压）列都根据日期和时间来描述。 BP 8M 列对应于星期一 8am 的记录。 BP 12W 列对应于星期三 12pm 的记录，依此类推。研究人员想要将所有血压列堆叠为三列，分别对应于**星期一**、**星期三**和**星期五**这三天。
2. 选择**表** > **堆叠**。
3. 选择所有 BP 记录并点击**堆叠列**。

   列的顺序反映应对序列中的列进行分组的方式。
4. 选择**多序列堆叠**。
5. 在序列数旁边，键入 3。
6. 由于您要垂直堆叠列，请选择**连续**。
7. 将堆叠数据列的名称从**数据**重命名为 BP （代表血压）。
8. 将源标签列的名称从**标签**重命名为**日期**。

图 6.10 已完成堆叠的窗口

9. 点击**确定**。

图 6.11 堆叠的数据表

	被测者	剂量	日期	BP	日期 2	BP 2	日期 3	BP 3
1	1	A	BP 8M	183	BP 8W	174	BP 8F	171
2	1	A	BP 12M	174	BP 12W	178	BP 12F	178
3	1	A	BP 6M	180	BP 6W	181	BP 6F	171
4	2	A	BP 8M	173	BP 8W	170	BP 8F	175
5	2	A	BP 12M	181	BP 12W	179	BP 12F	185
6	2	A	BP 6M	181	BP 6W	176	BP 6F	188
7	3	A	BP 8M	181	BP 8W	188	BP 8F	183
8	3	A	BP 12M	189	BP 12W	175	BP 12F	183
9	3	A	BP 6M	177	BP 6W	182	BP 6F	180
10	4	A	BP 8M	181	BP 8W	176	BP 8F	183
11	4	A	BP 12M	177	BP 12W	173	BP 12F	187
12	4	A	BP 6M	182	BP 6W	184	BP 6F	183
13	5	A	BP 8M	184	BP 8W	172	BP 8F	170

在堆叠的数据表中，注意以下事项：

- 第一个**日期**列表示星期一。
- **日期 2** 列表示星期三。
- **日期 3** 列表示星期五。

拆分列

您可以通过将一列拆分为若干新列，从活动表创建新数据表。该列将根据另一列（称为**拆分依据**列）中的值进行拆分，您还可以根据一个或多个分组变量的值拆分列。

注意：若针对包含缺失值的分类列进行拆分，则对于数值列，缺失类别的列名称为“.”。

要拆分列，请执行以下步骤：

1. 选择**表** > **拆分**。

图 6.12 “拆分”窗口

2. 突出显示要拆分的一列或多列的名称，然后点击**拆分列**。
3. 突出显示要用作拆分列依据的值所属的列。
4. 点击**拆分**。
5. 使用附加选项进一步定制您的拆分。
6. 点击**拆分**。

拆分选项

“选择列”过滤器菜单 包含用于搜索和过滤列的选项。请参见 “开始使用 JMP”一章中第 50 页的 ““列过滤器”菜单”。

全部保留 在新表中包括所有列。

全部删除 在新表中仅包括在拆分中使用的列。默认选定该选项。

选择 选择要在新表中保留的列。

保持对话框打开 点击**确定**后保持 “拆分”窗口打开。

将脚本保存至源表 将脚本保存到原始数据表，这允许您使用相同的设置再次拆分数据。

拆分依据 添加要用作新列名称和拆分列依据的值所属的列。

拆分列 添加要拆分的一个或多个列。

分组 若希望按选定变量的各组拆分数据，请指定“分组”变量。每个组在输出表中生成一行。您还必须指定必需变量 “拆分依据”和 “拆分列”。

注意：若要作为分组依据的变量包含不等组或处于随机顺序，则指定它作为“分组”变量确保您的数据正确重构且将所有缺失值分配到合适的位置。

按列属性排序 按 “值顺序”列属性对输出列的顺序进行排序。若列没有 “列顺序”属性且数据具有隐含顺序 （如天数或月数），则应用隐含顺序。

输出表名称 （可选）指定新表的名称。

复制公式 （若数据表中有公式，则显示。）在输出列中包含原始表中的公式。

禁止公式计算 （若数据表中有公式，则显示。）创建新表时阻止 JMP 计算列公式。

拆分列的示例

本节包含两个使用拆分命令的示例：

- 在第一个示例中，按一列拆分另一列。请参见第 304 页的“拆分列：基本示例”。
- 第二个示例使用分组变量。请参见第 305 页的“拆分列：对行分组的示例”。

拆分列：基本示例

在 Popcorn.jmp 数据表中，**试验**列中的数据显示有两个试验：1 和 2。在该示例中，将**产量**列拆分为两个新列：一列对应于**试验 1**，另一列对应于**试验 2**。

1. 选择**帮助 > 样本数据库**，然后打开 Popcorn.jmp。
2. 选择**表 > 拆分**。
3. 选择**产量**列并点击**拆分列**。
4. 选择**试验**列并点击**拆分依据**。
5. 在**其余列**下，选择**全部保留**。

   默认设置为全部删除，这将省略不在拆分依据、拆分列或分组字段中的所有列。选择**全部保留**可在新表中包括这些列。
6. （可选）在**输出表名称**字段中键入 Yield column split by Trial column。
7. 点击**确定**。

   随即创建新的数据表 （图 6.13）。请注意以下事项：
   - **产量**和**试验**列不再显示。
   - 数据表包含两个新列，它们按照原始**试验**列中的唯一值（1 和 2）命名。
   - 原始**产量**列中的值现在拆分为名为 1 和 2 的两个新列。
   - **试验**和**产量**列之外的列均与原始表保持完全相同。

8. （可选）将新列重命名为有意义的名称。例如，将 1 重命名为**产量 （试验 1）**，将 2 重命名为**产量 （试验 2）**。

图 6.13 按试验列拆分产量列所创建的新表

	爆米花	油量	包型	1	2
1	plain	little	large	8.2	8.8
2	gourmet	little	large	8.6	8.2
3	plain	lots	large	10.4	8.8
4	gourmet	lots	large	9.2	9.8
5	plain	little	small	9.9	10.1
6	gourmet	little	small	12.1	15.9
7	plain	lots	small	10.6	7.4
8	gourmet	lots	small	18.0	16.0

拆分列：对行分组的示例

Drug Measurements.jmp 样本数据表包含对 12 个不同的试验对象试验的三种不同药物（a、b 和 c）的测量值。您希望将测量值拆分为不同的列，每种药物类型对应一列。您还希望按试验对象对测量值分组。

1. 选择**帮助 > 样本数据库**，然后打开 Drug Measurements.jmp。
2. 选择**表 > 拆分**。
3. 选择**药物类型**并点击**拆分依据**。
4. 选择**测量值**并点击**拆分列**。

   请注意**试验对象**变量包含不等组。多数试验对象都接受了全部三种药物，但试验对象 2 只接受了一种药物，试验对象 7 和 12 只接受了两种药物。在这种情况下，为确保正确的测量值与正确的试验对象关联，请将**试验对象**指定为分组变量。
5. 选择**试验对象**并点击**分组**。
6. 点击**确定**。

图 6.14 按分组变量拆分的 Drug Measurements.jmp

	试验对象	a	b	c
1	1	91	99	91
2	2	•	96	•
3	3	95	99	93
4	4	97	93	96
5	5	95	100	94
6	6	92	96	95
7	7	93	96	•
8	8	100	94	92
9	9	98	99	97
10	10	91	96	99
11	11	98	100	93
12	12	•	95	95

您可以看到为试验对象 2、7 和 12 显示了适当的缺失值。

转置行和列

您可以创建新的 JMP 表，它是活动数据表的转置版本。活动表的列是新表的行，活动表的行是新表的列。

要转置列，请执行以下步骤：

- 选择要转置的列。
- 指定一个“标签”列，新列自该列获取列名（可选）。
- 指定“依据”列，指示 JMP 在组内转置数据（可选）。

注意：要转置的列必须具有相同的数据类型。同样，若列包含值标签，转置将使用实际数据值而不是值标签。（请参见“设置 JMP 列属性”一章中第 266 页的“值标签”。）

要转置行和列，请执行以下步骤：

1. 打开包含要转置的行和列的数据表。
2. 选择**表 > 转置**。

图 6.15 “转置”窗口

3. 在左侧的**选择列**框中突出显示要转置的列名。
4. 点击**转置列**。
5. （可选）使用附加选项进一步定制您的转置表。
6. 点击**确定**。

转置选项

“选择列”过滤器菜单 包含用于搜索和过滤列的选项。请参见 “开始使用 JMP”一章中第 50 页的 ““列过滤器”菜单”。

仅转置选定行 仅转置当前在活动表中突出显示的行。

输出表名称 （可选）指定子集表的名称。

添加列名称标签 （仅在已指定标签列时适用。）指定 “标签”列的备择名称。若不指定，默认列名为**标签**。

保持对话框打开 点击**确定**后保持 “转置”窗口打开。

将脚本保存至源表 将脚本保存到原始数据表，这允许您使用相同的设置再次转置数据。

转置列 添加要转置的列。

标签 将原始表中的列数据用作新表中的列名。按以下步骤执行：

1. 在左侧的“选择列”框中突出显示一列。
2. 点击**标签**。列名显示在 “标签”框中。

默认列名称为**标签**。您可以使用**添加列名称标签**选项为列指定替换名称。将为标签列中的每个不重复值仅创建一列。因此，若标签列中有重复值，JMP 将使用最后一个重复行中的值为重复值仅创建一列。

依据 基于您在 “依据”框中指定的列将转置的列放入各组。按以下步骤执行：

1. 在选择列框中突出显示所需列名，您要将这些列中的值视为分组。
2. 点击**依据**。

表 6.1 说明了适用于转置的规则。

表 6.1 适用于转置的规则

若	那么
原始表只有列没有行	新表包含一列，列出这些列名。
原始表只有一列，且该列分配至标签	其值变为转置表中的列名。
原始表有多列且包含一个标签列	JMP 会在“转置”窗口显示时将标签列自动插入标签框。若不希望显示该列，可以将它删除。
原始表中没有标签列	转置表中的列名为**第 1 行**、**第 2 行**、…、**第 n 行**，其中 *n* 是原始表中的行数。

行和列转置的示例

本节包含三个示例：简单的转置示例、使用标签选项的示例，以及使用依据组的示例。

简单的转置示例

1. 选择**帮助** > **样本数据库**，然后打开 Materials1.jmp。
2. 选择**表** > **转置**。
3. 选择**塑料**、**锡**和**金**，然后点击**转置列**。
4. 点击**确定**。

图 6.16 简单转置表

	标签	第 1 行	第 2 行
1	塑料	1	4
2	锡	2	5
3	金	3	6

图 6.16 中的原始表具有两行以及名为**塑料**、**锡**和**金**的三个连续列。原始表中三列的每一列在转置后的表中各自占有一行，并且原始表的**第 1 行**与**第 2 行**被用来对转置后表中的列命名。名为**标签**的附加列的值即原始表中的列名（**塑料**、**锡**和**金**）。

使用“标签”选项的示例

1. 选择**帮助** > **样本数据库**，然后打开 Materials2.jmp。
2. 选择**表** > **转置**。
3. 选择**塑料**、**锡**和**金**，然后点击**转置列**。
4. 选择**物品**并点击**标签**。
5. 点击**确定**。

图 6.17 带标签的转置

	标签	nails	hooks
1	塑料	1	4
2	锡	2	5
3	金	3	6

原始表的**物品**列中的值将用作转置表中的列标签。

使用“依据”组的示例

1. 选择**帮助** > **样本数据库**，然后打开 Animals Subset.jmp。
2. 选择**表** > **转置**。
3. 选择**研究课题**和**英里数**，并点击**转置列**。

4. 选择**季节**并点击**标签**。
5. 选择**物种**并点击**分组依据**。
6. 点击**确定**。

图 6.18 使用 “依据” 组的转置

	物种	标签	fall	winter	spring	summer
1	COYOTE	研究课题	1	1	1	1
2	COYOTE	英里数	4	2	7	8
3	FOX	研究课题	3	3	3	3
4	FOX	英里数	4	3	6	2

转置表包含按组转置的值。

拼接数据表

在 JMP 中拼接数据表时，需要合并两个或更多数据表中的行。您可以创建新的数据表**或**向第一个数据表追加行。列名在要拼接的数据表中可能是相同的。若是这样，则新数据表中的列按拼接顺序列出所有数据表的值。若两个原始数据表包含名称不同的列，这些列将包括在新数据表中，并显示缺失值。

要拼接有相同列名的两个数据表，请执行以下步骤：

1. 选择**表** > **拼接**。

图 6.19 “拼接” 窗口

2. 突出显示要合并的数据表的名称，然后点击**添加**。

   可以根据您的选择拼接多个数据表，也可以多次添加同一数据表。新数据表中的行数是所有数据表中的行数之和。

3. （可选）点击**保存和计算公式**选项可请求 JMP 包括所有公式。

   若不选择该选项，新数据表中不包括任何公式。

注意：具有相同名称的列可以具有不同的公式。包含该列的公式的第一个数据表中的公式在拼接的数据表中保存。当拼接两个以上的表且第二个或第三个表具有相关列的公式时可能出现此情形。

4. （可选）点击**创建源列**选项可向新数据表添加名为**源表**的列。

   该列在相应行中标识源数据表的名称。

5. （可选）选择**追加至第一个表**选项可将行追加至**要拼接的数据表**字段中列出的第一个数据表。该选项是创建新数据表的备择选项。

6. （可选）在**输出表名称**字段中输入新数据表的名称。

   若不输入名称，JMP 会将该数据表命名为**无标题 #**（例如**无标题 1**）。若已选定**追加至第一个表**选项，则**输出表名称**字段不可用。

7. 点击**确定**。

拼接数据表的示例

假定您要将两个数据表（Trial1 和 Trial2）拼接到一个新数据表中。

1. 选择**帮助 > 样本数据库**，然后打开 Trial1.jmp 和 Trial2.jmp。
2. 从 Trial1.jmp 表中，选择**表 > 拼接**。
3. 在**已打开的数据表**列表中，选择 Trial2 并点击**添加**。
4. 点击**确定**。

数据表将合并为一个新的拼接表，其中包含第一个数据表中的所有行，后接第二个数据表中的所有行。

图 6.20 两个数据表的拼接结果

	爆米花	油量	包型	产量
1	plain	little	large	8.2
2	gourmet	little	large	8.6
3	plain	lots	large	10.4
4	gourmet	lots	large	9.2
5	plain	little	small	9.9
6	gourmet	little	small	12.1
7	plain	lots	small	10.6
8	gourmet	lots	small	18.0
9	plain	little	large	8.8
10	gourmet	little	large	8.2
11	plain	lots	large	8.8
12	gourmet	lots	large	9.8
13	plain	little	small	10.1
14	gourmet	little	small	15.9
15	plain	lots	small	7.4
16	gourmet	lots	small	16.0

原始数据表中的每个列名都可以在拼接的数据表中找到有一列与其列名匹配。但是，若列名不完全匹配，则不会合并这些列名。例如，若**产量**列被命名为**产量1** 和**产量2**，那么会在拼接数据表中为每列单独创建一列。

拼接包含表变量的两个或更多数据表时，将为每个表变量创建单独的列。这将确保在拼接数据表时不会丢失重要的差别。

注意：不会为以名称**注释**开头的表变量创建列。

拼接数据表和表变量的示例

例如，假定在两个不同的医院做了两项癌症试验。一项试验数据包含在 Cancer1.jmp 数据表中，另一项试验数据包含在 Cancer2.jmp 数据表中。

要将数据和变量合并为一个表，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Cancer1.jmp 和 Cancer2.jmp。
   请注意有两个不同的表变量：**剂量**和**位置**。在拼接表中，将为这两个表变量创建对应的列。
2. 从 Cancer1.jmp 数据表中，选择**表** > **拼接**。
3. 选择 Cancer2 并点击**添加**。
4. 点击**确定**。

图 6.21 拼接的数据和变量

	时间	细胞类型	治疗	之前治疗	剂量	位置
1	3	Adeno	Standard	No	150	Roanoke
2	7	Adeno	Test	No	150	Roanoke
3	8	Adeno	Standard	Yes	150	Roanoke
4	8	Adeno	Test	No	150	Roanoke
5	12	Adeno	Standard	Yes	150	Roanoke
6	18	Adeno	Test	Yes	150	Roanoke
7	19	Adeno	Test	No	150	Roanoke
8	24	Adeno	Test	No	150	Roanoke
9	3	Adeno	Standard	No	250	Asheville
10	7	Adeno	Test	No	250	Asheville
11	8	Adeno	Standard	Yes	250	Asheville
12	8	Adeno	Test	No	250	Asheville
13	12	Adeno	Standard	Yes	250	Asheville
14	18	Adeno	Test	Yes	250	Asheville
15	19	Adeno	Test	No	250	Asheville
16	24	Adeno	Test	No	250	Asheville

数据和变量已拼接起来。变量在拼接表中显示为列。每个数据表中的注释作为表变量添加至新数据表。

连接数据表

您可以通过选择**表 > 连接**将两个数据表合并为一个新表。有关如何连接两个数据表的全面说明，请参见第 312 页的“要将两个数据表连接为一个新数据表，请执行以下步骤：”。可以采用三种不同的方式连接表：

- 根据行号合并表。请参见第 314 页的“按行号连接的示例”。
- 在笛卡尔方式中，构成的新表包含两个原始表中行的所有可能组合。请参见第 316 页的“笛卡尔连接的示例”。
- 通过匹配两个数据表或单个数据表中存在的一个或多个列中的值。请参见第 317 页的“按匹配列连接的示例”。

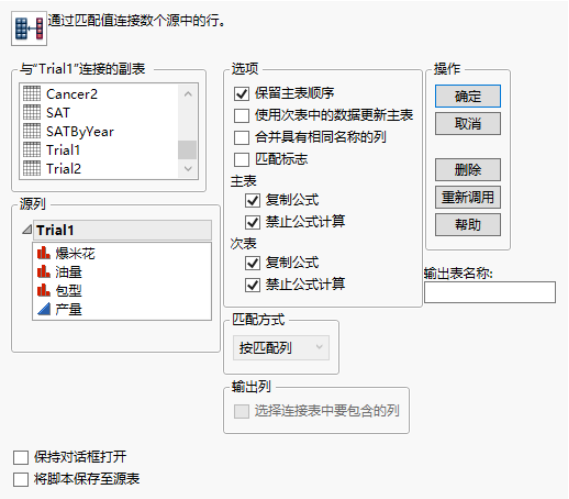
注意：“表”菜单中的“JMP 查询生成器”选项提供一个选项来在执行简单连接前查询数据。请参见“重塑数据”一章中第 321 页的“使用 JMP 查询生成器查询和连接数据表”。

要将两个数据表连接为一个新数据表，请执行以下步骤：

1. 打开要连接的两个数据表。
2. 选择**表 > 连接**。

   在出现的窗口中，所有打开的表的名称显示在**与 “...” 连接的副表**下。

图 6.22 “连接”窗口



3. 在**与 “...” 连接的副表**框中，选择要与活动表连接的表。
4. 从匹配方式区域中，选择指定如何连接表的选项。
5. 在**输出表名称**旁边的文本框中输入新表的名称。
6. （可选）使用附加选项进一步定制连接过程。

7. 点击**确定**创建连接数据表。

连接选项

保持对话框打开 点击**确定**后保持 “连接”窗口打开。

将脚本保存至源表 将脚本保存到原始数据表，这允许您使用相同的设置再次连接数据。

保留主表顺序 在连接表中保持原始数据表的顺序，而不是按匹配列排序。默认选定该选项。

使用次表中的数据更新主表 原始表中的数据将使用具有相同名称的列中的数据进行更新。结果显示在新数据表中。

请注意以下几点：

- JMP 不会使用缺失值替换数据。
- 输出表使用与原始表相同的列。因此，当您使用**使用次表中的数据更新主表**时，**选择连接表中要包含的列**不适用。
- 仅当按行号或按匹配列连接表时，**使用次表中的数据更新主表**选项才可用。

合并具有相同名称的列 第二个表中的列数据替换原始表中同名列的数据。请注意，第一个表中的缺失值将被第二个表中的非缺失值替换。

若您当前是按列进行匹配的，则在您选择**合并具有相同名称的列**时，匹配标志会被选中。新连接的表格将包含一个名为**匹配标志**的名义型列：

- 若该列中显示 Main，则数据源自第一个（活动）表。
- 若该列中显示 With，则数据源自第二个表。
- 若该列中显示 Both，则数据同时可在第一个表和第二个表中找到。

匹配标志 确定按列匹配时是否创建了**匹配标志**列。当您选择**合并具有相同名称的列**时自动选定该选项。

复制公式（主表和次表） 在输出列中包括主表和 / 或次表中的公式。默认选定该选项。

禁止公式计算（主表和次表） 在创建新表期间阻止 JMP 对列公式求值。默认选定该选项。

按匹配列 要连接行，请选择两个表中的值和数据类型都匹配的列。按以下步骤执行：

1. 在**源列**区域的每个列表中突出显示一个列名。顶部列表中第一个突出显示的列与底部列表中第一个突出显示的列相配对，依此方式匹配两个列表中的第二列以及之后所有列。仅当所有配对列的值和数据类型都匹配的情况下，才会连接行。
2. 点击**匹配**。选定的一对列显示在**匹配列**框中。匹配的列不一定同名，也不一定在两个表中处于相同的相对列位置。
3. （可选）若要仅包括首次匹配，请选中两个表中与**删除重复项**关联的框。这样只会将首次匹配写入新表。若为一个表指定该选项，则第一个匹配值将与另一个表中的所有匹配项相连接。若两个表中均不选中与**删除重复项**关联的框，则在每组匹配列值内执行笛卡尔连接。

4. （可选）若要包括数据表中的所有行，则在没有匹配值的情况下，也要选中与**包含不匹配项**关联的框。您可以为要连接的任一个表或同时为两个表指定该选项。

按行号 并排连接两个表。

笛卡尔连接 使用笛卡尔方式连接两个表，这种方式将构成一个新表，其中包含两个原始表中行的所有可能组合。JMP 将第一个表中的数据与第二个表中的数据交叉，以显示每组数据中值的所有组合。

选择连接表中要包含的列 从要包括在输出表中的任意一个表选择列的子集。按以下步骤执行：

1. 在源列区域中，突出显示每个表中要包括在新表中的列。
2. 在输出列区域中点击**选择**。

输出表名称 指定已连接的表的名称。

连接数据表的示例

以下各节提供使用连接命令的示例。

按行号连接的示例

按行号连接表可并排连接两个表。新表包含两个表中的所有列，除非您指定仅包含特定列。

连接行数不等的表

若要连接的两个表的行数不等，新表将包含在两个表中找到的行的值。

1. 选择**帮助 > 样本数据库**，然后打开 Species1.jmp 和 Species2.jmp。
   请注意 Species1.jmp 表包含两行， Species2.jmp 表包含四行。
2. 从 Species1.jmp 表中，选择**表 > 连接**。
3. 在**与 “...” 连接的副表**框中，选择 Species2。
4. 从匹配方式区域中，选择**按行号**。
5. 点击**确定**。

图 6.23 按行号连接的表

	物种	季节
1	FOX	spring
2	COYOTE	summer
3		fall
4		winter

若一个包含两行的表与包含四行的表连接，则新表包含四行。

连接具有相同名称的列

若两个表包含的列名相同，新表中的这些列的名称将显示为“**表名**中的**列名**”。例如，假定您要将 Trial1.jmp 和 Trial2.jmp 数据表（如图 6.24 所示）中的八行合并为一个表。您希望合并后的新表包含两个表中的所有列。

1. 选择**帮助** > **样本数据库**，然后打开 Trial1.jmp 和 Trial2.jmp。
2. 从 Trial1.jmp 数据表中，选择**表** > **连接**。
3. 在**与 “...” 连接的副表**框中，选择 Trial2。
4. 从匹配方式菜单中，选择**按行号**。
5. 点击**确定**。

图 6.24 原始表和连接表

	“Trial1” 中的 “爆米花”	“Trial1” 中的 “油量”	“Trial1” 中的 “包型”	“Trial1” 中的 “产量”	“Trial2” 中的 “爆米花”	“Trial2” 中的 “油量”	“Trial2” 中的 “包型”	“Trial2” 中的 “产量”
1	plain	little	large	8.2	plain	little	large	8.8
2	gourmet	little	large	8.6	gourmet	little	large	8.2
3	plain	lots	large	10.4	plain	lots	large	8.8
4	gourmet	lots	large	9.2	gourmet	lots	large	9.8
5	plain	little	small	9.9	plain	little	small	10.1
6	gourmet	little	small	12.1	gourmet	little	small	15.9
7	plain	lots	small	10.6	plain	lots	small	7.4
8	gourmet	lots	small	18.0	gourmet	lots	small	16.0

两个原始表中的列名可能相同。输出列名然后按源表名进行限定。例如，新表中的列名显示为**表名称**的<变量名称>。

仅连接指定列

假定不想在连接表中包括原始数据表中的所有列。

1. 选择**帮助** > **样本数据库**，然后打开 Trial1.jmp 和 Trial2.jmp。
2. 从 Trial1.jmp 数据表中，选择**表** > **连接**。
3. 在**与 “...” 连接的副表**框中，选择 Trial2。
4. 从匹配方式菜单中，选择**按行号**。
5. 点击**选择连接表中要包含的列**，指定要包括的列的子集。
6. 在 “源列” 列表中，从 Trial1 列表中选择**爆米花**和**产量**，然后从 Trial2 列表中选择**产量**。由于两个表中的**爆米花**列中存在相同的数据，您只需选择一列。
7. 点击**选择**。
8. 点击**确定**。

图 6.25 仅连接指定列

	爆米花	“Trial1” 中的 “产量”	“Trial2” 中的 “产量”
1	plain	8.2	8.8
2	gourmet	8.6	8.2
3	plain	10.4	8.8
4	gourmet	9.2	9.8
5	plain	9.9	10.1
6	gourmet	12.1	15.9
7	plain	10.6	7.4
8	gourmet	18.0	16.0

笛卡尔连接的示例

执行笛卡尔连接时，JMP 以笛卡尔方式连接两个表：由此创建的新表将包含两个原始表中所有可能的行组合。这将在输出表中创建多种情况，每种列值组合对应一种情况。

笛卡尔连接的简单示例

1. 选择**帮助** > **样本数据库**，然后打开 Species1.jmp 和 Species2.jmp。
2. 从 Species1.jmp 表中，选择**表** > **连接**。
3. 在**与 “...” 连接的副表**框中，选择 Species2。
4. 从匹配方式菜单中，选择**笛卡尔连接**。
5. 点击**确定**。

图 6.26 使用 “笛卡尔连接” 连接表

	物种	季节
1	FOX	spring
2	FOX	summer
3	FOX	fall
4	FOX	winter
5	COYOTE	spring
6	COYOTE	summer
7	COYOTE	fall
8	COYOTE	winter

Species1.jmp 中的数据与 Species2.jmp 中的数据交叉以生成连接表，其中显示每个集合中的值的所有组合。

笛卡尔连接的复杂示例

在该示例中，使用**表** > **连接**命令两次：

- 第一次连接使用**笛卡尔**选项将 Oil Amount.jmp 表与 Batch.jmp 表合并。
- 第二次连接将生成的表（**笛卡尔油量 + 包型**）与 Popcorn Type.jmp 表合并，并生成连接所有表的最终表。

1. 选择**帮助** > **样本数据库**，然后打开 Oil Amount.jmp、Batch.jmp 和 Popcorn Type.jmp。

2. 从 Oil Amount.jmp 表中，选择**表 > 连接**。
3. 在**与 “...” 连接的副表**框中，选择 Batch。
4. 从匹配方式菜单中，选择**笛卡尔连接**。
5. 在**输出表名称**下，键入 Oil Amount and Batch。
6. 点击**确定**。

图 6.27 Oil Amount and Batch 连接表

	油量	包型
1	little	small
2	little	large
3	lots	small
4	lots	large

连接表包含 Oil Amount.jmp 和 Batch.jmp 表中的所有列。添加 Popcorn Type.jmp 列：

7. 从您刚创建的 Oil Amount and Batch 表中，选择**表 > 连接**。
8. 在**与 “...” 连接的副表**框中，选择 Popcorn Type。
9. 从匹配方式菜单中，选择**笛卡尔连接**。
10. 点击**确定**。

图 6.28 与 Popcorn Type 连接的 Oil Amount and Batch

	油量	包型	爆米花
1	little	small	gourmet
2	little	small	plain
3	little	large	gourmet
4	little	large	plain
5	lots	small	gourmet
6	lots	small	plain
7	lots	large	gourmet
8	lots	large	plain

最终表包含三个原始表中的所有列。请记住笛卡尔连接生成的行数是原始表中行数的乘积。

按匹配列连接的示例

当您通过匹配列来连接数据表时，JMP 通过匹配列值来对齐两个表中的行。

警告：若匹配的列未生成唯一行，请小心操作。

注意：

- 用于匹配的列必须具有相同的数据类型（数字、字符或行状态）。
- 您可以将数据表连接到其自身，以便从数据表中删除重复的值或行。
- 您可以使用虚拟连接作为创建新数据表的替代方法。请参见第 324 页的“虚拟连接数据表”。

使用唯一标识符连接表

您有两个关于热狗信息的文件。第二个文件包含**大小**这一列，该列未包含在第一个文件中。使用“使用匹配列来连接”来组合数据表。

1. 选择**帮助 > 样本数据库**，然后打开 Hot Dogs.jmp 和 Hot Dogs2.jmp。

   产品名列是数据表中各行的唯一标识符。每个表有 54 个唯一产品名。
2. 从 Hot Dogs 数据表中，选择**表 > 连接**。
3. 在 “与 ‘Hot Dogs’ 连接的副表”列表中，选择 Hot Dogs2。
4. 在 “匹配方式”区域中选择**按匹配列**。
5. 从 Hot Dogs 和 Hot Dogs2 列表中，选择**产品名称**。
6. 点击**匹配**。
7. 在 “选项”中，选择**合并具有相同名称的列**。使用合并列时，将自动选择 “匹配标志”。
8. 选择窗口左下角的**保持对话框打开**，以便在连接的数据表不符合预期的情况下轻松返回到“连接”窗口。
9. 点击 “应用”。

图 6.29 按匹配列连接的表

	匹配标志	产品名称	原料类型	口味	美元/盎司	美元/磅（蛋白质）	卡路里	钠	蛋白质脂肪比	种类	大小
1	二者	Happy Hill Sup...	Beef	Bland	0.11	14.23	186	495	1	Beef	Jumbo
2	二者	Georgies Skinle...	Beef	Bland	0.17	21.70	181	477	2	Beef	Jumbo
3	二者	Special Market'...	Beef	Bland	0.11	14.49	176	425	1	Beef	Jumbo
4	二者	Spike's Beef	Beef	Medium	0.15	20.49	149	322	1	Beef	Regular
5	二者	Hungry Hugh's ...	Beef	Medium	0.10	14.47	184	482	1	Beef	Regular
6	二者	Great Dinner Be...	Beef	Medium	0.11	15.45	190	587	1	Beef	Regular
7	二者	RJB Kosher Beef	Beef	Medium	0.21	25.25	158	370	2	Beef	Regular
8	二者	Wonder Kosher...	Beef	Medium	0.20	24.02	139	322	2	Beef	Regular
9	二者	Happy Fats Jum...	Beef	Medium	0.14	18.86	175	479	1	Beef	Regular
10	二者	Midwest Beef	Beef	Medium	0.14	18.86	148	375	1	Beef	Regular
11	二者	General Kosher ...	Beef	Medium	0.23	30.65	152	330	1	Beef	Regular

生成的表现在包含 Hot Dogs2 表中所含的**大小**列。“匹配标志”列指示所有行的数据都来自这两个表。

连接行数和列名均不同的表

假定 Sarah 和 Joe 要做一个爆米花实验。他们要用不同的油量制作不同类型的爆米花（gourmet 和 plain）。他们记录了产出的爆米花的量（产量）。Sarah 在名为 Trial1.jmp 的文件中给出了第一项试验的数据。Joe 在名为 Little.jmp 的文件中给出了第二项试验的数据。您希望将两个表合并为一个表。

1. 选择**帮助 > 样本数据库**，然后打开 Trial1.jmp 和 Little.jmp。
2. 从 Trial1.jmp 表中，选择**表 > 连接**。
3. 在**与 “...” 连接的副表**框中，选择 Little。

您可以看到有三列（**爆米花**、**油量 / 油**和**包型**）在两个表中包含相同的值。将这些列标识为匹配列。此外，由于 Sarah 和 Joe 为**油**和**油量**列指定了不同的名称，您可以向 JMP 表明**油量**和**油**匹配。

4. 取消选择**保留主表顺序**。

   连接的表将按照匹配列而非 Trial1.jmp 中的数据顺序进行排序。

5. 在 “匹配方式”区域中选择**按匹配列**。
6. 从 Trial1 列表中，选择**爆米花**、**油量**和**包型**。
7. 从 Little 列表中，选择**爆米花**、**油**和**包型**。
8. 点击**匹配**。

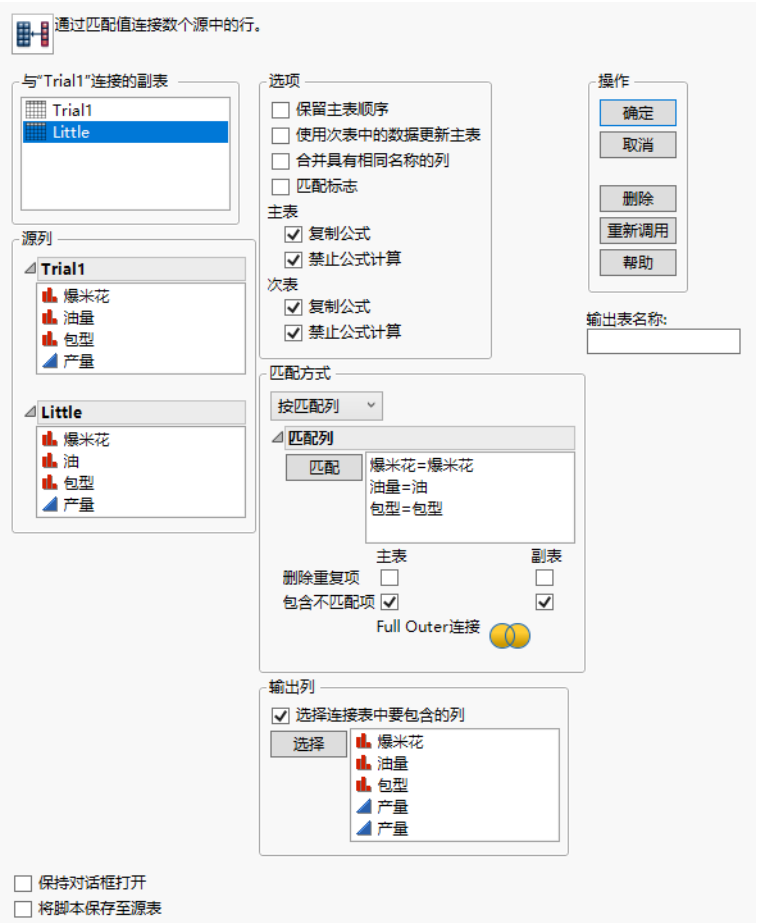
   查看这两个数据表，您可以看到其中具有不同的行数。 Trial1.jmp 具有八个实验条件值，Little.jmp 仅具有其中四个实验条件值。Sarah 完成了她的实验，但 Joe 只是部分完成了他的实验。您希望连接表包含 Trial1.jmp 中的所有行，即使 Little.jmp 表中的行中包含缺失值。

9. 为这两个表选中**包含不匹配项**框。

   在连接的表中，您只想有一列对应**爆米花**、一列对应**油**、一列对应**包型**。但是，您希望有两个**产量**列：一个代表 Trial1.jmp 的**产量**，另一个代表 Little.jmp 的**产量**。

10. 选中**选择连接表中要包含的列**旁边的框。
11. 从 Trial1 列表中选择所有列。
12. 点击**选择**。
13. 从 Little 列表中，选择**产量**。
14. 点击**选择**。

图 6.30 完成的连接窗口



15. 点击**确定**。

图 6.31 连接的 Trial1.jmp 和 Little.jmp

	爆米花	油量	包型	"Trial1" 中的 "产量"	"Little" 中的 "产量"
1	gourmet	little	large	8.6	8.2
2	gourmet	little	small	12.1	15.9
3	gourmet	lots	large	9.2	•
4	gourmet	lots	small	18.0	•
5	plain	little	large	8.2	8.8
6	plain	little	small	9.9	10.1
7	plain	lots	large	10.4	•
8	plain	lots	small	10.6	•

连接表按匹配列排序。请注意，来自 Little.jmp 表的**产量**列（“Little”中的“产量”）具有缺失值，指示没有与 Trial1.jmp 表匹配的值。

将数据表连接到其自身 （以删除重复条目）

1. 选择**帮助** > **样本数据库**，然后打开 Coffee Shop Purchases.jmp。
   您可以看到有些客户在同一日期饮用了同一种饮料。您想要使用**连接**合并这些重复行。
2. 选择**表** > **连接**。
3. 在**与 “...” 连接的副表**框中，选择 Coffee Shop Purchases。
4. 在 “匹配方式”区域中选择**按匹配列**。
5. 从两个 Coffee Shop Purchases 列表中，选择全部三个列：**日期**、**客户**和**饮料**。
6. 点击**匹配**。
7. 为两个表 （主表和副表）同时选择**删除重复项**框。
8. 为**输出表名称**键入 Coffee Shop Purchases Final。
9. 点击**确定**。

图 6.32 原始数据表和连接的 Coffee Shop Purchases 数据表

	日期	客户	饮料
1	01/05/2013	Jeff	Espresso
2	01/05/2013	Mike	Chai tea
3	01/05/2013	Sergio	Latte
4	01/05/2013	Mike	Espresso
5	01/05/2013	Mike	Espresso
6	01/05/2013	Lisa	Espresso
7	01/05/2013	Lisa	Espresso
8	02/13/2013	Lisa	Chai tea
9	02/13/2013	Lisa	Chai tea
10	02/13/2013	Mike	Chai tea
11	02/13/2013	Sergio	Espresso
12	02/13/2013	Sergio	Espresso

	日期	客户	饮料
1	01/05/2013	Jeff	Espresso
2	01/05/2013	Mike	Chai tea
3	01/05/2013	Sergio	Latte
4	01/05/2013	Mike	Espresso
5	01/05/2013	Lisa	Espresso
6	02/13/2013	Lisa	Chai tea
7	02/13/2013	Mike	Chai tea
8	02/13/2013	Sergio	Espresso

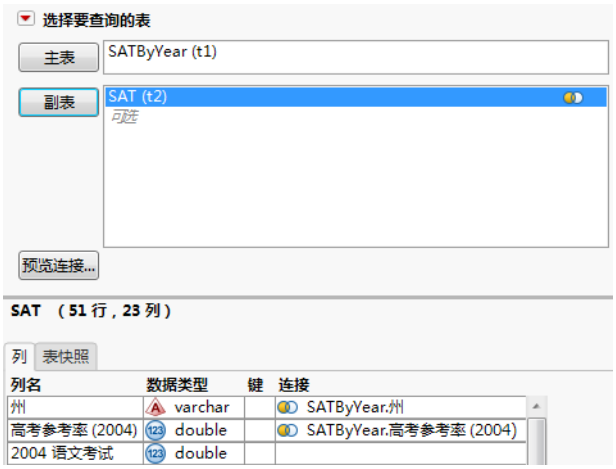
使用 JMP 查询生成器查询和连接数据表

使用 “表”菜单中的 “JMP 查询生成器”选项，您可以查询数据表并将选定的数据保存到一个新数据表。该功能类似于使用 “连接”命令，但是您可以在保存数据前执行查询。例如，您可以查询 SAT 数据并在数据表中仅保存 2004 年测验得分的数据。您还可以包括一个提示，允许用户运行该查询并选择 2004 年测验得分的一个子集。

1. 选择**帮助** > **样本数据库**，然后打开 SAT.jmp 和 SATByYear.jmp。
2. 显示 SATByYear.jmp。
3. 选择**表** > **JMP 查询生成器**。
   将当前数据表 SATByYear.jmp 选定为主表。

4. 在 “可用表” 列表中选择 SAT.jmp，然后点击**副表**。
5. 选择 “副表” 按钮旁边的 SAT.jmp，然后查看 “列” 选项卡。“连接” 列显示这两个列具有相同的名称且已连接。

图 6.33 连接的列



6. 点击**生成查询**。

添加列

1. 在 “可用列” 列表中，选择 t1. 州、t1. 高考参考率 (2004)、t2.2004 语文考试和 t2.2004 数学考试。
2. 点击**添加**。

   这些列被添加到 “已包括的列” 选项卡。
3. 选择**仅非重复行**以避免保存重复行。
4. 在 “查询预览” 选项卡上，确保选择**自动更新预览**以便您可以看到选定的列。

添加过滤器

1. 在 “已包括的列” 列表中，选择 t2.2004 语文考试和 t2.2004 数学考试，然后点击**将选定项添加至过滤器** 。

   将这些列添加到 “过滤器” 列表。

图 6.34 选定的过滤器

2. 点击每个过滤器旁边的红色小三角菜单，然后选择**运行时提示**。接受默认提示消息。

运行查询

1. 点击**运行查询**。

   系统提示您指明在数据表中保存哪些数学和词语得分。若要保存所有数据，请不要更改提示窗口中的值。

2. 点击**确定**以创建数据表。

图 6.35 查询的数据

	州	高考参考率 (2004)	2004 语文考试	2004 数学考试
1	Alabama	0.1	560	553
2	Alaska	0.53	518	514
3	Arizona	0.32	523	524
4	Arkansas	0.06	569	555
5	California	0.49	501	519
6	Colorado	0.27	554	553
7	Connecticut	0.85	515	515
8	District of Colu...	0.77	489	476
9	Delaware	0.73	500	499
10	Florida	0.67	499	499
11	Georgia	0.73	494	493

有关也在数据库的查询生成器中显示的功能的详细信息，请参见“导入数据”一章中第 104 页的“在“查询生成器”中生成 SQL 查询”。

提示：

- 要连接不同源（如一个数据库和 Microsoft Excel）的数据，请使用查询生成器将数据库数据导入数据表，将 Excel 数据导入数据表，然后使用“表”菜单中的“JMP 查询生成器”来查询和连接这些表。
- 打开一个查询时，查询中的数据表作为隐藏文件打开，您可以从“JMP 主窗口”打开该文件。生成查询时还可以从表面板中打开隐藏文件。
- 若您尝试查询的 Microsoft Excel 文件包含的列数超过 255，则仅导入 255 列。应改为选择**文件 > 打开**，在“Excel 导入向导”中打开该 Excel 文件，然后将该文件另存为数据表。

关于指向查询中数据表的链接

当您查询数据表时，得到的数据表包含用于重新运行查询、更新数据和编辑查询的脚本。这些脚本中的数据表具有**绝对路径**。例如，这些脚本的以下部分定义在何处存储两个 JMP 样本数据表：

```
JMP Tables(
     ["SAT" => "C:\Program Files\SAS\JMPPRO\16\Samples\Data\SAT.jmp",
     "SATByYear" => "C:\Program
  Files\SAS\JMPPRO\16\Samples\Data\SATByYear.jmp"]
  ),
```

但是，将查询保存为 .jmpquery 文件时，该文件包含到原始数据表的**相对路径**（若可以创建这些路径）。以下示例显示存储在 FuelData 子文件夹中的数据表，该文件夹的路径是相对 .jmpquery 文件而言：

```
JMP Tables( ["Cars" => "FuelData\Cars.jmp",
  "Trains" => "FuelData\Trains.jmp"] )
```

在 .jmpquery 文件中，若无法创建相对路径，若可能，绝对路径与路径变量替代项一起使用。当您运行查询时，若找不到数据表，系统将提示您选择数据表。JMP 然后检测查询中的其他缺失数据表是否在选定的文件夹中。

您还可以使用路径变量来定位数据表。例如，您可能编写一个脚本，用于在 $DOCUMENTS 文件夹中选择一个数据表。或者，您可以在 JSL 脚本中定义一个路径变量，然后从该脚本运行查询。有关路径变量的详细信息，请参见《Scripting Guide》。

虚拟连接数据表

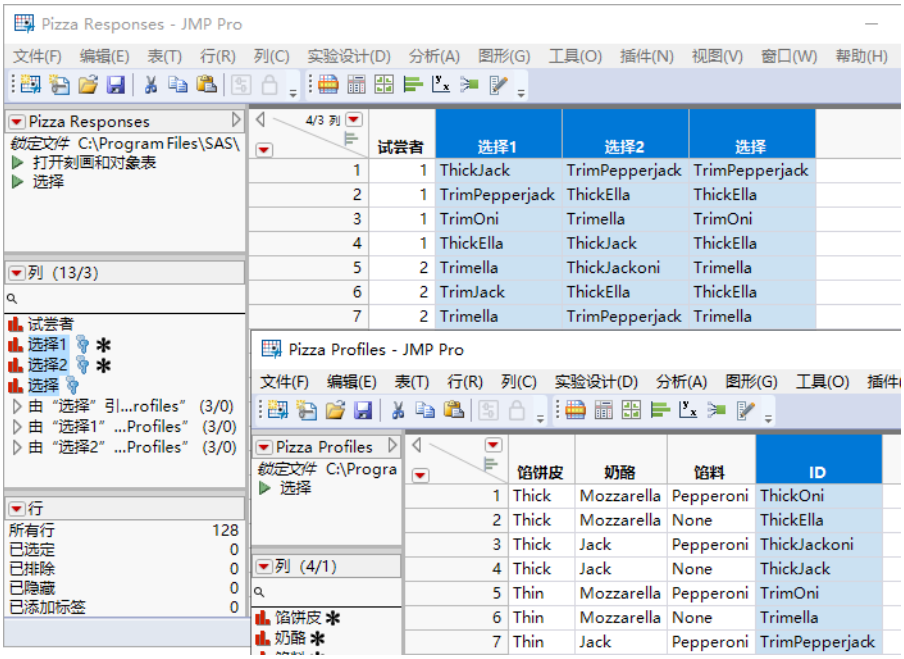
虚拟连接将一个主数据表链接到一个或多个辅助数据表。该功能允许主数据表访问辅助数据表中的数据而不必实际连接这些表。虚拟连接表节省了内存空间，因为相同数据不在引用它们的每个表中进行复制。更新链接的数据更简单；链接的数据可以在源表中独立更新而不必在引用表中进行更新。

可通过 “链接 ID”和 “链接引用”列属性进行链接。

- “链接 ID”列属性将**辅助**数据表中的某列标记为 ID 列。也就是说，该数据表的各行通过该 ID 列的值唯一标识。包含“链接 ID”列属性的数据表称为**引用**的数据表。
- “链接引用”列属性将**主**数据表中的某列链接到引用数据表中的“链接 ID”列。该列属性指定被引用数据表的路径名称。包含链接引用的列称为**引用**列。引用列可通过“链接 ID”列查找辅助数据表中的数据。

图 6.36 显示虚拟连接的数据表的一个示例。Pizza Profiles.jmp 是被引用的数据表，因为 ID 列包含一个“链接 ID”列属性。该 ID 列包含对应于引用数据表 Pizza Responses.jmp 中数据的唯一值。

图 6.36 虚拟连接的数据表



注意：一个列可以有一个“链接引用”列属性，但是不一定要链接引用的数据表中的列。在“列”列表中该列旁边的虚拟连接图标为灰色 。若引用的数据表未打开或引用列被排除，就会出现这样的情况。有关其他虚拟连接图标的详细信息，请参见“开始使用 JMP”一章中第 41 页的“虚拟连接图标”。

设置链接列属性后，来自被引用数据表中的所有列成为主数据表的一部分。因此，若主数据表有一个引用列，则被引用表的列自动显示在启动窗口的“选择列”列表中。

虚拟连接列的示例

假定披萨实验中的响应者选择了自己喜爱的馅饼皮、馅料和奶酪。要查看响应者喜欢哪种奶酪，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Pizza Profiles.jmp 和 Pizza Responses.jmp。
2. 首先，右击 Pizza Profiles.jmp 中的 ID 列并选择**链接 ID**。

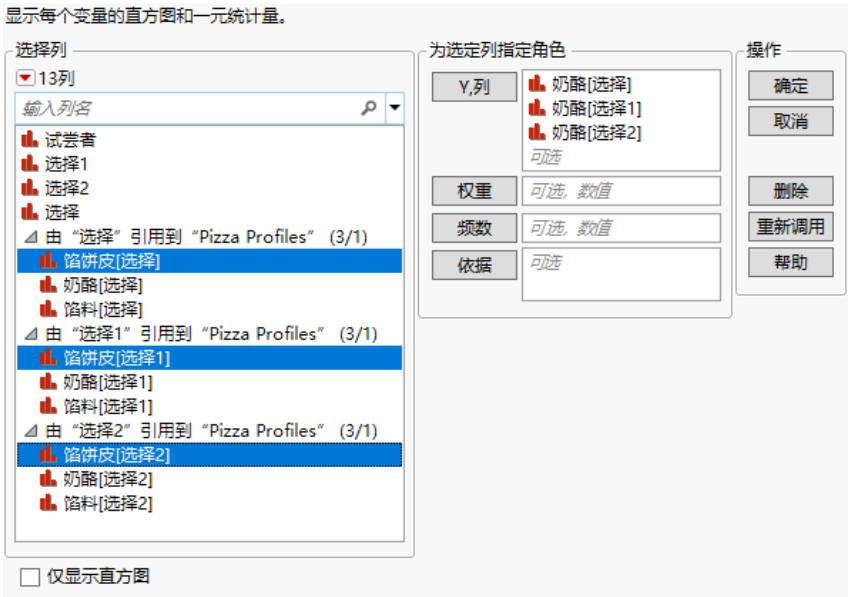
   该列包含对应于 Pizza Responses.jmp 中值的唯一值。例如，Pizza Profiles.jmp 中的一行指示具有厚馅饼皮、Mozzarella 奶酪和意大利辣香肠馅料的披萨昵称为 ThickOni。当 Pizza Responses.jmp 中的所有响应都为“ThickOni”时，表明试尝者选择了具有这些特性的披萨。
3. 在 Pizza Responses.jmp 中，选择以“选择”开头的三列。
4. 右击并选择**链接引用** > Pizza Profiles.jmp。

选定的列链接到 Pizza Profiles.jmp 中的 ID 列并显示在主数据表的“列”列表中（第 325 页上的图 6.36）。

注意：再看一下某个列的“链接引用”属性，我们注意到 Pizza Profiles.jmp 数据表已选定。有关更改链接的信息，请参见“设置 JMP 列属性”一章中第 280 页的“链接数据表与虚拟连接属性”。

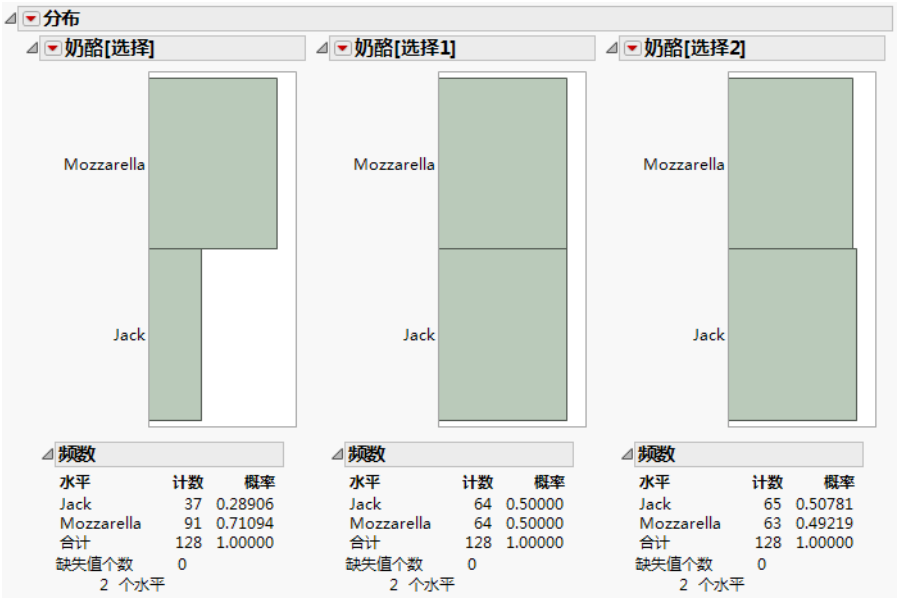
5. 在 Pizza Responses.jmp 中，选择**分析 > 分布**。
6. 向下滚动以选择被引用列组中的三个“奶酪”列，点击 **Y，列**，然后点击**确定**。

图 6.37 “分布”启动窗口



更多响应者选择了**选择**列中的“Mozzarella”。在**选择 1** 和**选择 2** 中，Mozzarella 与 Jack 的比例几乎是相等的。

图 6.38 披萨响应的分布



注意：

- 有关另一虚拟连接的列的示例，请参见 Movie Rentals.jmp 样本数据表。该数据已与 Movie Customers.jmp 和 Movie Inventory.jmp 连接，因此您不需要指定“链接引用”。
- 一个数据表只能有一个“链接 ID”列属性。该“链接 ID”列不能有重复值。
- 一个数据表可以有多个引用列。这些列可以引用不同或相同的数据表。
- 一个被引用的数据表也可以有一个引用列（引用其他数据表的列）。
- 若从列中删除了“链接引用”列属性或被引用的数据表已关闭，则从主数据表中删除相关的被引用列。
- 链接的列的数据类型必须匹配。
- “表达式”和“行状态”数据类型不支持“链接引用”或“链接 ID”列属性。
- 保存主数据表时，保存了“链接引用”列属性，但是不保存引用的列。打开这些数据表时，它们重新链接。
- 重命名被引用的数据表时，自动更新相关的“链接引用”列属性。
- 当您从表面板红色小三角菜单中选择“复制表脚本”时，不包括引用的列。
- 选择“行状态同步”选项以同步包含虚拟连接列的数据表中的行状态。请参见第 328 页的“示例：虚拟连接的列和显示行状态”。
- 列可以同时具有“链接 ID”和“链接引用”列属性。

提示：

- 除了右击一个列以添加“链接 ID”和“链接引用”列属性外，您还可以通过“列信息”窗口添加这些属性。请参见“设置 JMP 列属性”一章中第 280 页的“链接数据表与虚拟连接属性”。
- 您可以指定在虚拟连接的列（显示在“列”列表中）中使用链接的列名。该选项支持您指定较短的列名（例如，**奶酪**而不是**奶酪[选择]**）。在“链接引用”列属性中，选择**使用链接的列名**。您还可以按常规方式手动重命名该列。“表”首选项称为“虚拟连接使用链接的列名”。默认情况下取消选定。
- 为了与另一用户共享数据，您可能要合并数据以便永久连接数据表。该选项很有用，因为您只需提供一个数据表而非提供主数据表和辅助数据表。要将连接的列合并到具有链接引用的表，请从数据表左窗格中的红色小三角菜单选择**合并引用数据**。这会将数据从辅助数据表复制到主数据表。当您保存主数据表时，也会保存实际数据。“链接引用”列属性将从主数据表中的引用列中删除。请参见第 330 页的“示例：合并虚拟连接的数据表中的引用数据”。

示例：虚拟连接的列和显示行状态

当行状态在具有虚拟连接列的一个数据表中更改后，这些行状态可以自动显示在其他链接数据表的虚拟连接列中。假定主数据表和辅助数据表都有一个对象 ID 列。行状态在对象 ID 列上包含“链接 ID”属性的数据表中发生更改。然后在包含虚拟连接的“链接引用”列的数据表中更新这些行状态。您可以指定“链接引用”列将行状态发送（**调度**）到其他数据表还是接收（**接受**）行状态。

1. 选择**帮助 > 样本数据库**，然后打开 Nic Demographics.jmp、Nic Labs.jmp 和 Nic Adverse Events.jmp。
2. 查看 Nic Labs.jmp。

   请注意，在“列”列表中，蓝色的“虚拟连接”图标 显示在**唯一对象标识符**旁边。蓝色指示该列具有“链接引用”列属性且引用的数据表 (Nic Demographics.jmp) 已打开。“链接引用”列属性将当前数据表中的**唯一对象标识符**列链接到引用的数据表中的**唯一对象标识符**列。若该图标为灰色 ，则指示引用的数据表未打开或这些列可能链接得不正确。

   按同样方式设置 Nic Adverse Events.jmp。

图 6.39 唯一对象标识符上的 “链接引用” 列属性

3. 在 Nic Labs.jmp 中，选择**唯一对象标识符**列并选择**列 > 列信息**。
4. 选择**链接引用**列属性。

   请注意，选择了 “接受” 。这意味着当引用的数据表 (Nic Demographics.jmp) 中的行包含某个行状态时，该行状态自动显示在当前数据表中。

   还要注意，“选择”、“排除” 和 “隐藏” 行状态都已选定。在 Nic Demographics.jmp 中选定这些行状态时，这些行状态将显示在 Nic Lib.jmp 和 Nic Adverse Events.jmp 中。
5. 在 Nic Demographics.jmp 中，右击第一行并选择**隐藏和排除**。

   第一行包含唯一对象标识符 101001 的数据。
6. 查看 Nic Labs.jmp，看看唯一对象标识符 101001 的隐藏和排除行有哪些。

警告：

- 在具有多个列的多个表中调度行状态可能很复杂，要格外小心。
- 当没有为同一 ID 值分配多个行状态时，最好使用调度行状态。若 “链接引用” 列调度颜色行状态，则当存在多个具有相同 ID 值但颜色行状态不同的行时，将很快产生歧义。例如，数据表有两行，ID 值为 001。您将一行着红色，另一行着绿色，导致 JMP 无从知晓如何在链接的数据表中设置行状态。

注意：

- 表 A 中的多个列可以链接到表 B。表 B 本身可以有链接到表 C 的多个列。所有行的行状态显示在包含链接的列的数据表中。例如，若第 1 行在表 A 中排除且在表 B 中隐藏，则主数据表将该行显示为“已排除和隐藏”。
- 若数据表正在接受特定的行状态，请避免手动在数据表中应用该行状态。让调度列来控制该行状态。

示例：合并虚拟连接的数据表中的引用数据

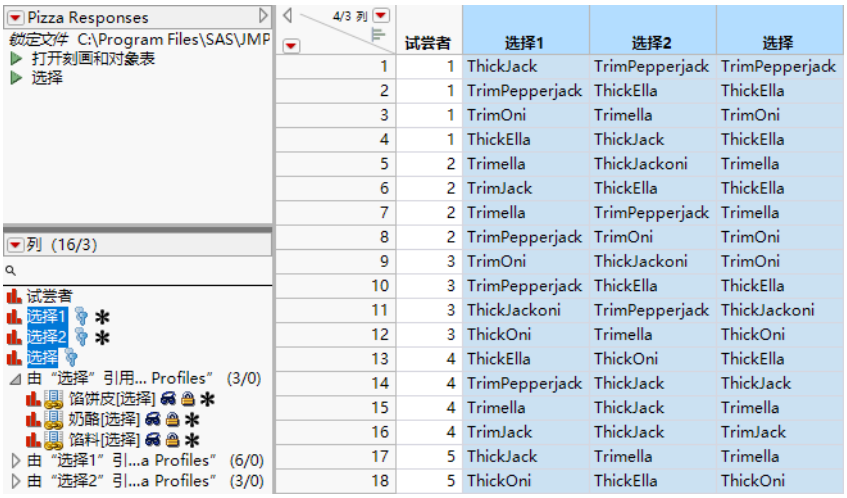
为了与另一用户共享虚拟连接的数据表中的数据，您可能要合并数据以便永久连接数据表。该选项很有用，因为您只需提供一个数据表而非提供主数据表和辅助数据表。

当您合并引用数据时，这会将数据从辅助数据表复制到主数据表。当您保存主数据表时，也会保存实际数据。

1. 选择**帮助** > **样本数据库**，然后打开 Pizza Profiles.jmp 和 Pizza Responses.jmp。
2. 右击 Pizza Profiles.jmp 中的 ID 列并选择**链接 ID**。
3. 选择 Pizza Responses.jmp 中的**选择 1**、**选择 2** 和**选择**列，右击，然后选择**链接引用** > Pizza Profiles.jmp。

   选定的列链接到 Pizza Profiles.jmp 中的 ID 列。

图 6.40 虚拟连接的数据表中的选定列



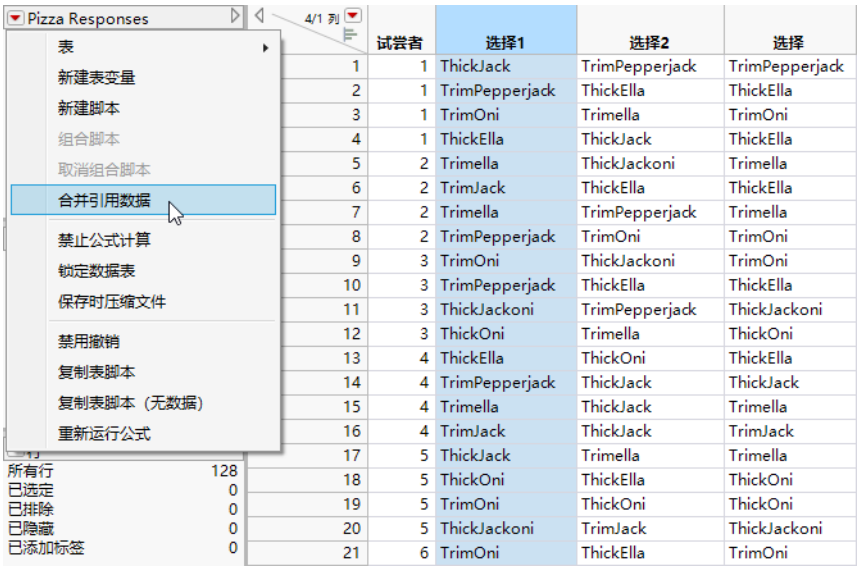
	试尝者	选择1	选择2	选择
1	1	ThickJack	TrimPepperjack	TrimPepperjack
2	1	TrimPepperjack	ThickElla	ThickElla
3	1	TrimOni	Trimella	TrimOni
4	1	ThickElla	ThickJack	ThickElla
5	2	Trimella	ThickJackoni	Trimella
6	2	TrimJack	ThickElla	ThickElla
7	2	Trimella	TrimPepperjack	Trimella
8	2	TrimPepperjack	TrimOni	TrimOni
9	3	TrimOni	ThickJackoni	TrimOni
10	3	TrimPepperjack	ThickElla	ThickElla
11	3	ThickJackoni	TrimPepperjack	ThickJackoni
12	3	ThickOni	Trimella	ThickOni
13	4	ThickElla	ThickOni	ThickElla
14	4	TrimPepperjack	ThickJack	ThickJack
15	4	Trimella	ThickJack	Trimella
16	4	TrimJack	ThickJack	TrimJack
17	5	ThickJack	Trimella	Trimella
18	5	ThickOni	ThickElla	ThickOni

4. 在 Pizza Responses.jmp 中，选择**选择 1** 列 （若尚未选定）。

   注意：若不选择任何列，则所有列都将包含在合并数据中。

5. 点击数据表名称旁边的红色小三角并选择**合并引用数据**。

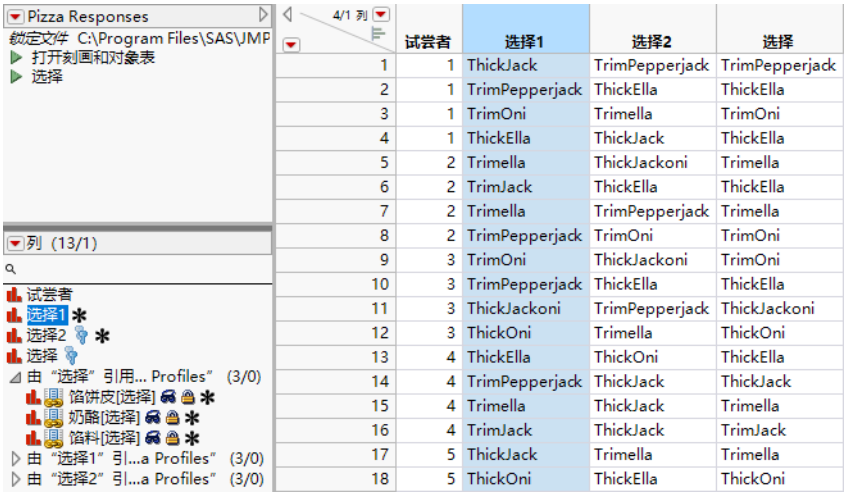
图 6.41 合并引用数据



	试尝者	选择1	选择2	选择
1	1	ThickJack	TrimPepperjack	TrimPepperjack
2	1	TrimPepperjack	ThickElla	ThickElla
3	1	TrimOni	Trimella	TrimOni
4	1	ThickElla	ThickJack	ThickElla
5	2	Trimella	ThickJackoni	Trimella
6	2	TrimJack	ThickElla	ThickElla
7	2	Trimella	TrimPepperjack	Trimella
8	2	TrimPepperjack	TrimOni	TrimOni
9	3	TrimOni	ThickJackoni	TrimOni
10	3	TrimPepperjack	ThickElla	ThickElla
11	3	ThickJackoni	TrimPepperjack	ThickJackoni
12	3	ThickOni	Trimella	ThickOni
13	4	ThickElla	ThickOni	ThickElla
14	4	TrimPepperjack	ThickJack	ThickJack
15	4	Trimella	ThickJack	Trimella
16	4	TrimJack	ThickJack	TrimJack
17	5	ThickJack	Trimella	Trimella
18	5	ThickOni	ThickElla	ThickOni
19	5	TrimOni	ThickOni	ThickOni
20	5	ThickJackoni	TrimJack	ThickJackoni
21	6	TrimOni	ThickElla	TrimOni

将有一条消息确认选定列已合并。请注意，**选择 1** 的蓝色虚拟连接图标不再出现在“列”列表中。“链接引用”列属性已删除。

图 6.42 合并的数据



	试尝者	选择1	选择2	选择
1	1	ThickJack	TrimPepperjack	TrimPepperjack
2	1	TrimPepperjack	ThickElla	ThickElla
3	1	TrimOni	Trimella	TrimOni
4	1	ThickElla	ThickJack	ThickElla
5	2	Trimella	ThickJackoni	Trimella
6	2	TrimJack	ThickElla	ThickElla
7	2	Trimella	TrimPepperjack	Trimella
8	2	TrimPepperjack	TrimOni	TrimOni
9	3	TrimOni	ThickJackoni	TrimOni
10	3	TrimPepperjack	ThickElla	ThickElla
11	3	ThickJackoni	TrimPepperjack	ThickJackoni
12	3	ThickOni	Trimella	ThickOni
13	4	ThickElla	ThickOni	ThickElla
14	4	TrimPepperjack	ThickJack	ThickJack
15	4	Trimella	ThickJack	Trimella
16	4	TrimJack	ThickJack	TrimJack
17	5	ThickJack	Trimella	Trimella
18	5	ThickOni	ThickElla	ThickOni

若悬停在列面板中的某一链接列上，则工具提示将指示它链接到哪个数据表。

图 6.43 包含链接列的数据表

示例：单个列的链接 ID 和链接引用

假设多个数据表中都提供员工数据，并且可由同时具有“链接 ID”和“链接引用”属性的一列链接在一起。要查看如何设置数据表的示例，请执行以下步骤：

1. 选择**帮助 > 样本数据库**，然后打开 Employee Master.jmp。

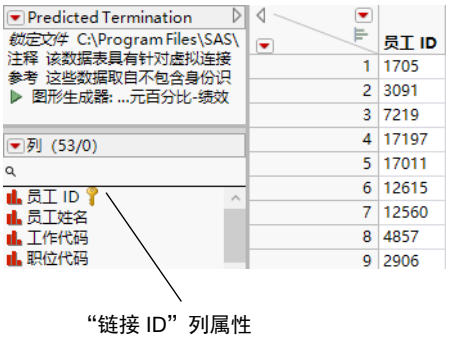
   注意：在“员工 ID”列的“链接引用”列属性中，已选定“自动打开”选项以便自动打开另外两个表：Education History.jmp 和 Predicted Termination.jmp。它们显示在主窗口的“窗口列表”中。

   - Employee Master.jmp 包含有关每位员工的信息。
   - Education History.jmp 包含有关每位员工的教育程度的信息。
   - Predicted Termination.jmp 包含有关每位员工的工作表现信息。

2. 从主窗口的“窗口列表”中打开 Predicted Termination.jmp。

   请注意，“员工 ID”列具有“链接 ID”列属性。

图 6.44 “链接 ID”列属性



3. 在 Employee Master.jmp 中，依次选择**员工 ID** 列和**列 > 列信息**。

图 6.45 Employee Master.jmp 中的 “链接 ID” 和 “链接引用” 列属性

请注意，该列具有 “链接 ID” 列属性和对 Education History.jmp 的 “链接引用”。

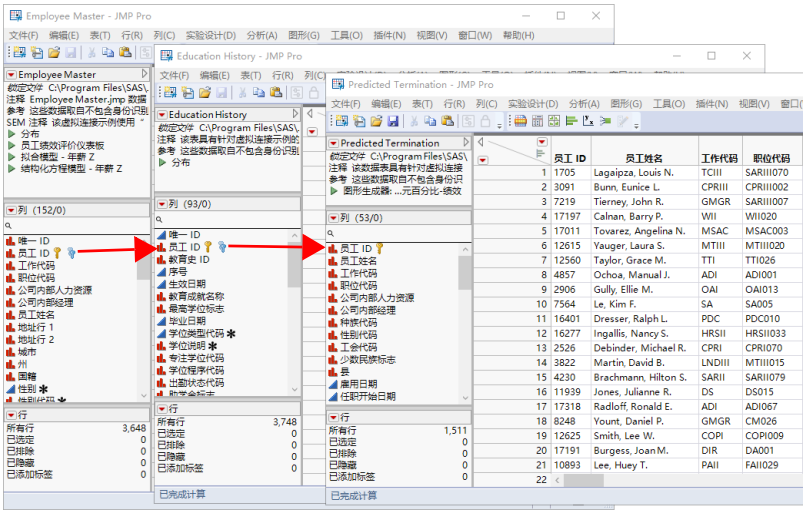
在数据表的 “列” 列表中，“链接引用” 图标 和 “链接 ID” 图标 显示在**员工 ID** 列旁边。

4. 从主窗口的 “窗口列表” 中打开 Education History.jmp。
5. 依次选择**员工 ID** 列和**列 > 列信息**。

图 6.46 Education History.jmp 中的 “链接 ID”和 “链接引用”列属性

如图 6.46 中所示，员工 ID 具有 “链接 ID”列属性和对 Predicted Termination.jmp 的 “链接引用”。

图 6.47 链接的数据表



Employee Master.jmp 的员工 ID 列引用 Education History.jmp，该表随后又引用 Predicted Termination.jmp。所有数据都可以通过主数据表 Employee Master.jmp 中的虚拟连接获得。

注意：

- 列的“链接引用”属性中的“自动打开”指定在打开主数据表时是否自动打开链接表。
- 若在保存数据表时撤销隐藏引用的列，则在打开表时这些引用的列将自动链接并撤销隐藏。
- 关闭数据表时，最好先关闭主表 (Employee Master.jmp)，因为该表与其他所有表之间存在链接。以不同的顺序关闭它们可能会显示一个提示，指示要关闭的表有其他打开的窗口。您可能需要取消并考虑以不同的顺序保存和关闭表。

更新数据表

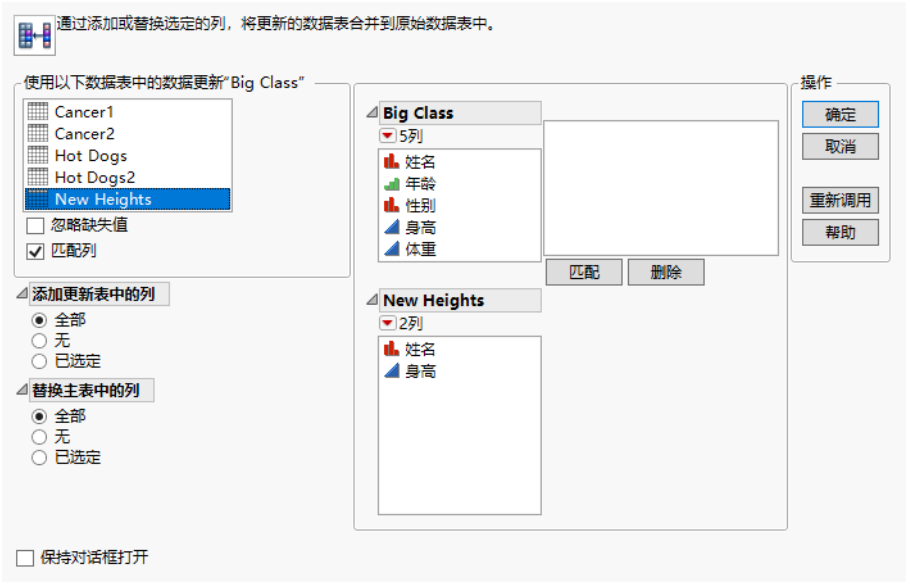
若具有两个数据表并且要使用新表中的数据更新原始表，可以选择**表 > 更新**。更新命令是原位连接的特例。该命令是选中更新选项的连接，但不会生成新表。

更新表之前，确保包含要替换的值的列名与包含用于替换的数据的列名相同。

使用另一个打开表中的值替换活动表中的值

1. 点击要更新的原始表（这是要替换值的表），以便激活该表。
2. 选择**表 > 更新**。

图 6.48 更新表



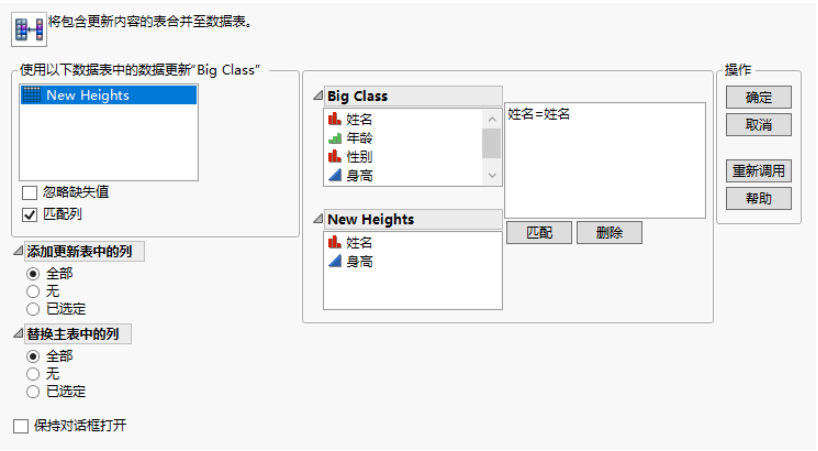
3. 突出显示包含要转移到原始表的数据的新表。
4. （可选）若不想让 JMP 将原始表中的值替换为新表中的任何缺失值，请选中**忽略缺失值**旁边的框。原始表中的原始值若对应新表中的缺失值，则保留原始值。

5. 若两个表具有一个或多个列，其中的值能够唯一描述各行，JMP 会将这些列用作**匹配列值**。也就是说，JMP 会更新匹配列值吻合的行。JMP 使用这些列来保持数据的排序顺序。若您的表不含匹配列值，则可以继续根据更新值的行顺序合并更新值。要继续处理包含匹配列值的表，请参见第 336 页的 “使用匹配列更新表”。
6. 从“添加更新表中的列”区域中，选择一个选项。使用这些选项，您可以将新表中的列（尚不存在）添加至原始表。
   - 选择**全部**可将新表中的所有列添加至原始表。
   - 选择**已选定**仅将新表中选定的列添加至原始表。
   - 若不想将新表中任何不存在的列添加至原始表，请选择**无**。
7. （可选）从“替换主表中的列”区域中，选择一个选项。使用这些选项，您可以替换同名列中的值。
   - 选择**全部**可将新表中具有相同名称的所有列添加至原始表。
   - 选择**已选定**仅将新表中选定的具有相同名称的列替换到原始表中。
   - 若不想将新表中具有相同名称的所有不存在的列添加至原始表，请选择**无**。
8. 点击**确定**。

使用匹配列更新表

1. 按照上一节第 335 页的“使用另一个打开表中的值替换活动表中的值”介绍的前三个步骤操作。
2. 选择**匹配列**。

图 6.49 点击 “匹配列”时将出现该窗口



3. 突出显示要匹配的两个列名 （在各自的表中）。
4. 点击**匹配**。

5. （可选）重复执行以匹配更多列。
6. 点击**确定**。

注意：表中正在更新（且尚未分配匹配项）、具有不同列名的列被追加为单独的列。为避免该问题，请在“添加更新表中的列”区域中选择“无”选项。

更新数据表的示例

假定研究人员具有一个包含学生身高测量值的数据表。研究人员收到一个更新表，其中包含学生最近的身高测量值。研究人员不想通过在数据表中滚动浏览来查找身高变动的学生，然后再复制和粘贴新值。使用**更新**命令，研究人员可以使用新的身高值快速更新原始数据表。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp 和 New Heights.jmp。

   Big Class.jmp 表包含原始数据， New Heights.jmp 表包含更新数据。
2. 从 Big Class.jmp 表中，选择**表** > **更新**。
3. 在**使用以下数据表中的数据更新** “...”框中，选择 New Heights。
4. 选择**匹配列**。
5. 在 Big Class 列表和 New Heights 列表中，选择**姓名**。
6. 点击**匹配**。

   这会指示 JMP 使用**姓名**作为匹配列值，因为该列中的值能唯一描述各行。
7. 点击**确定**。

图 6.50 更新的 Big Class.jmp 表

	姓名	年龄	性别	身高	体重
1	KATIE	12	F	62	95
2	LOUISE	12	F	61	123
3	JANE	12	F	58	74
4	JACLYN	12	F	68	145
5	LILLIE	12	F	52	64
6	TIM	12	M	64	84
7	JAMES	12	M	63	128
8	ROBERT	12	M	70	79
9	BARBARA	13	F	62	112
10	ALICE	13	F	61	107
11	SUSAN	13	F	58	67
12	JOHN	13	M	68	98
13	JOE	13	M	64	105
14	MICHAEL	13	M	60	95

您可以看到更新表中的身高值不再与原始表中的值一致。这些值已使用New Heights.jmp 表中的较新值进行了更新。

匿名数据

您可以使用“匿名数据”功能创建新的数据表，在该表中某些唯一标识符已被删除。

- 列标题、字符数据和值标签已修改。
- 名义型列中的数据已修改。
- 有序型列中已修改的数据采用与原始数据表中相同的顺序显示。
- 连续列中的数据未修改。
- 修改了列属性中的列名。

注意：表脚本和某些表变量无法进行匿名化处理。若您的数据表包含不能匿名的内容，则在日志中出现警告消息。

“注释”列属性已删除。其他列属性已根据数据表中的更改进行了相应修改。例如，将“值顺序”属性重命名。更新了除预测公式之外的公式。某些列属性可能无法再对该列产生预期的影响。

要重命名数据，请执行以下操作之一：

- 要重命名特定列中的数据，请选择相应列，然后选择**表 > 匿名**。
- 要重命名整个数据表中的数据，请选择**表 > 匿名**。

数据将出现在新的数据表中。

免责声明：“匿名数据”实用工具仅用于协助用户按照上文所述的方法从数据表中删除某些唯一标识符。“匿名数据”实用工具可能无法从数据表中删除所有敏感信息或个人的可识别信息。按照法律或政策规定需要对数据进行完全匿名化、伪匿名化或分离识别时（如数据已经或可能透露给其他各方时），用户不应完全依赖“匿名数据”实用工具删除此类信息。SAS 不代表并明确否认，使用“匿名数据”实用工具会自动致使用户遵守任何国家/地区、州/省、地方或国际上关于敏感或个人可识别信息的隐私、分离识别或匿名化的法律、法规或政策。

第 7 章

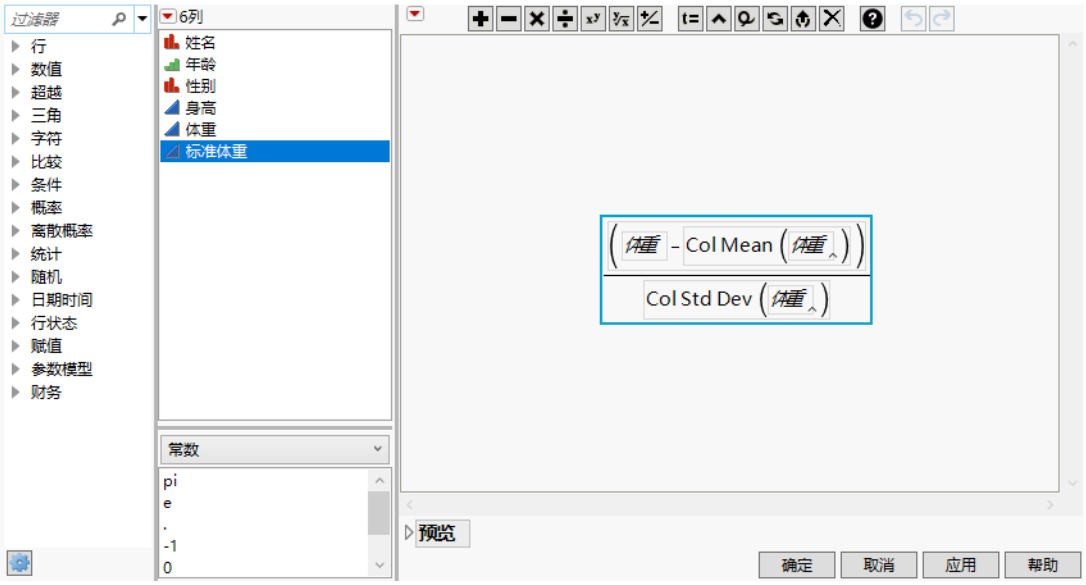
在 JMP 中创建公式

公式编辑器

使用 JMP 公式编辑器可以创建列，该列中的值用公式计算得出并且该公式存储为列信息的一部分。公式既可以通过简单地指定数值型、字符型或行状态常数来构成，也可以包含基于条件子句的复杂计算。“公式编辑器”窗口操作起来就像是一个带有按钮、显示屏和函数列表的计算器。

有关“公式编辑器”中可用的函数的详细信息，请参见第 631 页的“在公式中引用 JMP 函数”一章。

图 7.1 公式编辑器



目录

公式概述 342
生成公式 342
在公式中引用数据表值 344
在公式中使用局部变量 346
在公式中合并参数 347
在公式中插入常数 348
将运算符添加至 JMP 公式 349
“公式编辑器”的小键盘参考 350
在公式中使用函数 351
查找“公式编辑器”中经常使用的函数 353
复制和重复公式 355
对公式中的表达式排序 357
按优先顺序生成公式 358
JMP 公式编辑器选项 359
计算 JMP 函数的导数 359
简化复杂公式 359
计算公式 360
忽略公式中的错误 361
查看用 JSL 显示的公式 361
编辑公式 361
更正公式中的错误 362
在公式中选择表达式 362
删除函数公式 362
剪切、复制和粘贴公式 363
点击并拖动公式 363
在 JMP 公式编辑器中定制公式 363
隐藏和显示分级显示的公式 364
更改字体大小公式 364
指定矩阵大小公式 364
更改 JMP 公式的方向 364
打开和关闭参数公式 365
示例：在 JMP 公式编辑器中生成公式 365
使用 JMP 公式编辑器中的基本功能 365
使用局部变量公式 366
使用 Munger 函数公式 367
使用 Match 条件函数公式 368
使用删除表达式按钮公式 369

JMP 公式编辑器的键盘快捷方式 370
JMP 公式编辑器的术语表 371

公式概述

公式是数据表不可或缺的部分，原因如下：

- 在您保存数据表时，公式存储为列信息的一部分。
- 您可以通过打开“公式编辑器”随时检查或更改公式。
- 公式的值可以链接到或依赖于其他列中的值。每当您编辑公式链接到的列中的值时，公式值将随之自动重新计算。
- 公式值在数据表中被锁定，因此可防止手动编辑。

本章介绍“公式编辑器”并显示如何生成公式。有关每个函数的详细信息，请参见第 631 页的“在公式中引用 JMP 函数”一章。

生成公式

公式是存储在列中的表达式，用来执行运算以便在该列中插入值。公式可以执行数学运算，如加法和乘法；也可以通过引用同一数据表中的其他区域来比较列值或连接值。公式可由任意 JMP 脚本语言 (JSL) 命令构成。一旦在列中插入公式，该列即被锁定：只有更改或删除公式才能编辑该列的值。

可采用三个基本步骤生成公式：

1. 通过右击要应用公式的列名并选择**公式**，打开“公式编辑器”。

   或

   双击要应用公式的列名，从**列属性**菜单中选择**公式**，然后点击**编辑公式**。

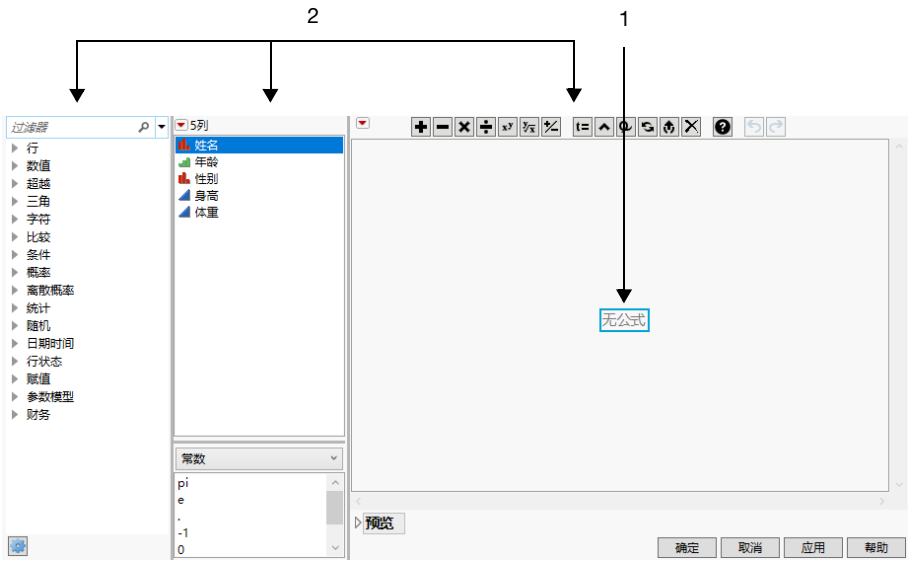
   在“公式编辑器”中，选定了空公式（以蓝色突出显示）。您可以开始键入或选择列或函数。

   – 键入时，会显示一个小的文本编辑器窗口，该窗口支持您编辑该公式。若需要更多空间，您可以点击**最大化编辑器**按钮 。默认在最大化的编辑器中打开长公式。
   – 选择某列或函数时，该项将添加至选定的蓝框。
   – 您还可以将“列”列表中的列拖入选定框。

   注意：元素周围出现蓝色外框时，表示该元素已被选中。相对于所点击位置的最小嵌套框内的所有项都将被选中。后续操作应用于这些组合元素。

2. 添加表达式、函数和项。然后它们就会出现在突出显示的蓝框内。本章中的以下各节提供有关如何添加常数、元素、运算符和函数的详细说明。

图 7.2 生成公式



有关如何使用“公式编辑器”的示例，请参见第 365 页的“使用 JMP 公式编辑器中的基本功能”。

注意：

- 在公式中，使用值标签引用某列时，悬停在值标签上方将能查看实际数据值。
- “公式编辑器”识别“已排除”状态并从计算中移除这些行。以下是使用 Col Maximum() 函数的一个示例：

  ```
  Col Maximum( :height, :sex, Excluded( Row State() ) )
  ```

- JMP 自动删除公式中的尾部空格。
- 您还可以使用“公式编辑器”定义定制格式。例如，您可能希望使用计算 Char(:height * 2.54) || " cm" 将某个数字从英寸转换为厘米。您应该意识到：您在双击数据表单元格时看到的底层数据仍然是 :height 的值。
- 要将光标从公式中的一个字段移至另一个字段，请选择该字段，然后按箭头键。
- 对于“列均值”之类的聚合函数，若您将公式拆分为两列，公式的计算速度可能更快。一列应包含表示聚合函数目标的表达式。另一列应包含引用新目标列的聚合函数本身。下面是原始公式的一个示例：

  ```
  :Y - Col Mean( If( :Status == "nonsmoker", :Y, . ), :Cycle )
  ```

  In a new Y2 column, create this formula:

  ```
  If( :Status == "nonsmoker",
     :Y,
     .
  )
  ```

在另一列中，创建下面的公式：

```
:Y - Col Mean( :Y2, :Cycle )
```

在公式中引用数据表值

您可以创建引用数据表中其他部分（如其他列和表变量）中的值的公式。

当公式使用其他列中的值时，该公式所在列中的值依赖于其他列中的值。只要公式引用的列发生更改，依赖的列也会随之更改。若删除引用的列，包含该公式的列中将显示空项。

若创建引用表变量中的值的公式，这些表变量必须在数据表中已经存在。表变量是可用于整个表的字符串。表变量名显示在数据表左侧的表面板中。请参见“输入和编辑数据”一章中第 246 页的“使用数据表变量”。

提示：要从“公式编辑器”中的值创建表变量，请右击该值并选择**从值创建表变量**。

下例显示如何重新创建 Companies.jmp 样本数据表中的公式列。

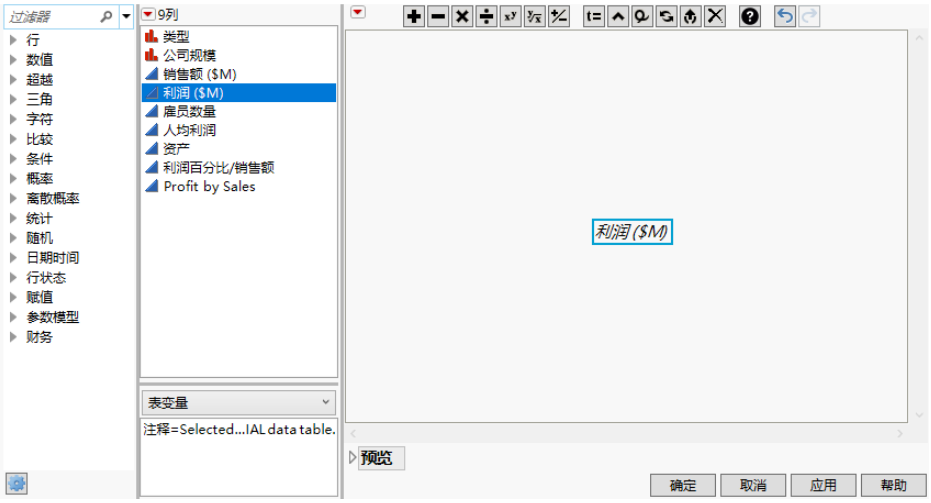
1. 选择**帮助 > 样本数据库**，然后打开 Companies.jmp。
2. 右击最后一个空列并选择**新建列**。

   “列信息”窗口随即出现。
3. 在 “列名”旁边键入 “Profit by Sales”。

   您正在重新创建现有 %profit/sales 列中的公式。
4. 在 “列属性”列表中，选择 “公式”。

   “公式编辑器”随即打开。编辑窗口中该框周围的蓝色外框指示该框已被选定。
5. 当该框处于选定状态时，从 “列”列表中选择**利润 ($M)**。

图 7.3 添加 “利润 ($M)” 列



6. 当该框处于选定状态时，点击 “除号” 按钮 ÷。

图 7.4 添加除号

7. 显示该框处于选定状态时，从 “列” 列表中选择**销售额 ($M)**。

图 7.5 添加 “销售额 ($M)” 列

8. 通过点击公式编辑区域中的外框将其选定。

图 7.6 选择公式

9. 点击乘号按钮 ×。

图 7.7 添加乘号

10. 在新框处于选定状态时，键入 “100” 并按 Enter 键。

图 7.8 将公式乘以 100

11. 点击**确定**。

    该列将用计算所得的数字填充。

12. 右击最后一列，然后选择 “列信息”。
13. 在 “列信息” 窗口中，从 “格式” 列表中选择**固定小数位数**，在 “宽度” 旁边键入 “10”，在 “小数” 旁边键入 “2”。

    由此指定列宽和小数位数。

14. 点击**确定**。

    新公式列匹配 %profit/sales 公式列。

提示：

- 按 Alt（macOS 上的 Option）并点击某列可替换该列的空和非空选定字段。
- “列” 列表的右击菜单包含两个替换选项：**替换所有出现的选定子表达式**可使用当前选定的列替换出现的所有选定表达式。**将选定子表达式替换为列**可针对支持替换的函数（例如，Sum 或 Plus）用多个列替换选定表达式。
- 通过编写 JSL 脚本自行定义函数并变换列。请参见《Scripting Guide》。

在公式中使用局部变量

您可以在表达式中创建和使用临时数值变量。您可以使用普通的局部变量，也可以使用参数，参数是特殊类型的局部变量。局部变量仅对定义这些变量的公式计算存在。它们在公式中显示为粗斜体项。

局部变量最常与赋值函数一同使用，赋值函数可以将表达式指定给复杂方程中使用的局部变量。请参见 “在公式中引用 JMP 函数” 一章中第 675 页的 “赋值函数”。这种方法有时可以简化方程构造，并提高计算效率。

要生成引用局部变量中的值的公式，请执行以下步骤：

第 1 步：创建局部变量

1. 右击数据表中的列名并选择**公式**，打开 “公式编辑器”。
2. 通过点击公式编辑区域中的框将其选中。（出现蓝色外框时即被选中。）
3. 在 “公式编辑器” 的中部窗格中，从列表中选择**局部变量**。

4. 点击**新建局部**。
5. 为该局部变量输入名称。默认情况下，局部变量的名称为 *t0*、*t1*，依此类推，并且包含缺失值。
6. 指定一个起始值，然后点击**确定**。
7. （可选）要复制、编辑或删除某个局部变量，请右击（在 macOS 上按 Ctrl 的同时点击）其名称并选择**复制**、**编辑**或**删除**。

第 2 步：在公式中插入局部变量

1. 通过点击公式编辑区域中的项将其选中。（出现蓝色外框时即被选中。）
2. 在局部变量列表中点击局部变量名称。局部变量在公式中显示为粗斜体项。

注意：

- 双击或按 Alt 并点击局部变量可替换所选定的公式部分。
- 另一种创建局部变量的方式是使用“公式编辑器”小键盘上的“制作临时变量” t= 按钮。该按钮可自动创建和显示局部变量并在其后放置一个分号。请参见第 349 页的“将运算符添加至 JMP 公式”。
- 您还可以从表达式创建局部变量。右击“公式编辑器”中的表达式并选择**从表达式创建局部变量**。
- 按“Alt+Shift+点击”，可以原地而不是在新窗口中编辑选定变量。

有关在公式中引用局部变量的示例，请参见第 366 页的“使用局部变量公式”。

在公式中合并参数

参数是特殊类型的局部变量；这类变量在某些平台中（如在“非线性”拟合平台中）被视为模型参数。参数可以像普通局部变量那样用在公式中。

查看参数示例

1. 在安装 JMP 时安装的 Sample Data 文件夹中，打开 Nonlinear Examples 文件夹，然后打开 US Population.jmp。
2. 右击列名 x- **公式** 并选择**公式**。
3. 在 “公式编辑器”的中部窗格中，从列表中选择**参数**。

注意：

- 完成非线性拟合或点击非线性控制面板中的“重置”按钮后，参数的值为非线性平台计算的最近值。
- 每次拟合算法执行一步运算，更新的参数值都会显示在“非线性”报表中。
- 将带有参数的公式粘贴到列中时，将为该列自动创建参数，除非该列包含同名的参数。
- 双击或按 Alt 并点击参数可替换所选定的公式部分。

- 按“Alt+Shift+点击”可以原地而不是在新窗口中编辑选定参数。

生成引用参数中的值的公式

第 1 步：创建参数

1. 右击数据表中的列名并选择**公式**，打开“公式编辑器”。
2. 确保公式在公式编辑区域中处于选定状态。（出现蓝色外框时即被选中。）
3. 在 “公式编辑器”的中部窗格中，从列表中选择**参数**。
4. 点击**新建参数**。
5. 输入参数的名称。默认情况下，参数的名称为 *b0*、 *b1*，依此类推，并且包含缺失值。
6. 指定一个起始值。在用于非线性平台的模型中使用参数时，输入该值非常重要。完成非线性拟合后，参数的值为非线性平台计算的最近值。
7. （可选）要同时添加若干参数 （例如为分类变量的每个水平都添加一个参数），请选择**扩展到类别，选择列**。然后选择要扩展参数的列。
8. 点击**确定**。
9. （可选）要复制、编辑或删除某个参数，请右击 （在 macOS 上按 Ctrl 的同时点击）其名称并选择**复制**、**编辑**或**删除**。

第 2 步：在公式中插入参数

1. 通过点击公式编辑区域中的项将其选中。（出现蓝色外框时即被选中。）
2. 在 “参数”框中点击参数名称。参数在公式中显示为粗体。

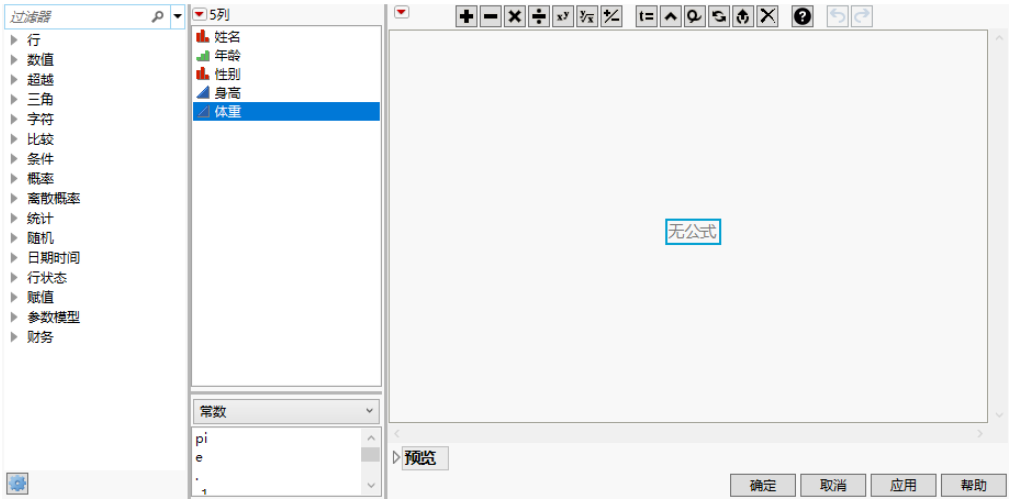
提示：按“Alt+Shift+点击”可以编辑选定参数。

在公式中插入常数

公式既可以通过简单地指定数值型、字符型或行状态常数来构成，也可以包含基于条件子句的复杂计算。常数包括常用的数值项，如 e、pi、-1、0、1 和 2。有两种方法可以将常数值添加至公式：

- 使用键盘手动输入常数
- 从中部窗格中的列表中选择这些常数，如图 7.9 所示。

图 7.9 选定了常数的公式元素浏览器



要向公式添加常数，请执行以下步骤：

1. 右击数据表中的列名并选择**公式**，打开“公式编辑器”。
2. 通过点击公式编辑区域中的框将其选中。（出现蓝色外框时即被选中。）
3. 键入某个数字或在中间窗格中选择**常数**，如图 7.9 所示。然后在出现的列表中点击某个值：0、 1、 2、 -1、 pi、 e。该值显示在红色外框中。
4. 使用小键盘和函数完成公式的其余部分。（请参见第 344 页的 “在公式中引用数据表值”、第 349 页的 “将运算符添加至 JMP 公式”和第 351 页的 “在公式中使用函数”。）

注意：双击或按 Alt 并点击常数可替换所选定的公式部分。

将运算符添加至 JMP 公式

您可以使用小键盘 （其中包含有助于生成公式的按钮）向公式添加运算符。小键盘包括常用运算符 （亦称函数）。

要使用窗口顶部的小键盘运算符生成公式，请执行以下步骤：

1. 右击数据表中的列名并选择**公式**，打开“公式编辑器”。
2. 通过点击公式编辑区域中的框将其选中。（出现蓝色外框时即被选中。）运算符针对突出显示的区域执行运算。
3. 选择要在公式中使用的列或变量。
4. 点击小键盘按钮。

“公式编辑器”的小键盘参考

表 7.1 对小键盘按钮进行了说明。

表 7.1 “公式编辑器”中的小键盘按钮

名称	说明
算术按钮	像通常在小型计算器上操作一样，提供加减乘除运算符。
指数	将给定值按指定幂次乘方。默认情况下指数为 2。
方根	计算被开方数的指定方根。它具有隐含指数 2（平方根），但并不显示出来。
一元符号函数	反转参数的符号。将一元符号函数应用到变量表达式或用它来输入负常数。
局部变量（制作临时变量）	创建和显示局部变量，并为其指定选定表达式的值。局部变量在表达式中的默认名称为 *t0*，之后紧随一个分号。有关创建和插入局部变量的详细信息，请参见第 346 页的“在公式中使用局部变量”。相关示例，请参见第 366 页的“使用局部变量公式”。
插入	插入新子句或函数参数。首先选择新元素要紧随其后的现有子句或参数，然后点击该按钮。新子句随即出现并处于选定状态。您还可以通过键入逗号来插入新子句或参数。 按 Alt 和插入图标 可在选定字段之前插入新字段
Delete	删除元素的值或删除子句。删除按钮的功能与键盘上的 Delete 键相同。
交换项	检查选定表达式的中心运算符，并交换该运算符两侧的表达式。
删除表达式（剥除）	删除包含第一个参数的最外层表达式。您可以重复该过程，以逐项删除公式。相关示例，请参见第 369 页的“使用删除表达式按钮公式”。

表 7.1 “公式编辑器”中的小键盘按钮 (续)

名称	说明
清除	清除整个公式
 帮助	打开“脚本索引”并定位到公式选定部分中的 JSL 函数项。仅当公式可视区域中的选定框是已知函数时，该按钮才适用。
 撤销	恢复最后一次更改。
 重新运行	重新运行最后恢复的更改。

在公式中使用函数

提示：悬停在函数上方可以了解有关该函数的更多信息。

您可以向公式添加许多类型的函数。所有这些函数都组织到“函数”列表中。浏览器将函数集合分组到按主题组织的列表中。该列表包含常用于公式的函数。使用“函数”列表可以指定要对公式中的元素执行的计算类型。

要创建包含函数的公式，请执行以下步骤：

1. 右击数据表中的列名并选择**公式**，打开“公式编辑器”。
2. 通过点击公式编辑区域中的表达式将其选中。（出现蓝色外框时即被选中。）函数针对突出显示的区域执行操作。
3. 在 “函数”列表中，选择一组要查看的函数。请参见表 7.2。

   属于该组的函数随后显示在菜单下的列表中。下面的列表简要说明了各个函数组。
4. 点击函数列表中的任意函数以将其应用到选定项。点击某些项时会出现一个子菜单，您应该从中做出选择。
5. 通过突出显示公式项并点击公式元素浏览器、小键盘或“函数”列表中的项，继续生成公式。

提示：

- 按 Alt 并选择函数可使用您从“函数”列表中选择的函数或您从小键盘中选择的按钮替换公式中的选定函数。
- 按 Alt（macOS 上的 Option）并点击某列或其他元素可替换该项，即便该项非空。

- 多数函数通过在公式编辑区域的框中插入的灰色文字提供有关适当参数的提示。若可以添加更多参数，函数还会在参数区域中显示一个小的加字符号。
- 要了解函数的作用，可右击“函数”列表或公式中的函数，然后选择**在脚本索引中显示**。
- 通过编写 JSL 脚本自行定义函数。请参见《Scripting Guide》。

表 7.2 函数组

分组	包括的函数
行	显示一个函数列表，其中包含 Lag、Dif、Subscript、Row 和 NRow 之类的各种函数。请参见“在公式中引用 JMP 函数”附录中第 633 页的“行函数”。
数值	显示一个函数列表，其中包含公式中的常用项。请参见“在公式中引用 JMP 函数”附录中第 635 页的“数值函数”。
超越	显示一个函数列表，其中包含自然对数、常用对数、指数、方根、阶乘、组合、Beta 和 Gamma 之类的函数。请参见“在公式中引用 JMP 函数”附录中第 635 页的“超越函数”。
三角	显示一个函数列表，其中包含标准的三角函数：正弦、余弦、正切、反函数和双曲函数。请参见“在公式中引用 JMP 函数”附录中第 639 页的“三角函数”。
字符	显示一个函数列表，这些函数针对字符参数操作（如去除首尾空格、查找字符串长度、在数字和字符之间转换）。请参见“在公式中引用 JMP 函数”附录中第 639 页的“字符函数”。
比较	显示一个函数列表，其中包含标准逻辑比较函数，如小于、小于等于、不等于，等等。请参见“在公式中引用 JMP 函数”附录中第 644 页的“比较函数”。
条件	显示一个函数列表，其中包含类似编程的函数，如 If、Match 和 Choose。请参见“在公式中引用 JMP 函数”附录中第 645 页的“条件函数”。
概率	显示一个函数列表，这些函数计算标准统计分布（如正态、Student *t*、卡方和 *F* 分布）的概率和分位数。请参见“在公式中引用 JMP 函数”附录中第 650 页的“概率函数”。
离散概率	显示一个函数列表，这些函数计算离散概率，如 Poisson、Gamma Poisson 和超几何分布的概率。请参见“在公式中引用 JMP 函数”附录中第 657 页的“离散概率函数”。
统计	显示一个函数列表，这些函数计算标准统计量，如均值或标准差。请参见“在公式中引用 JMP 函数”附录中第 658 页的“统计函数”。

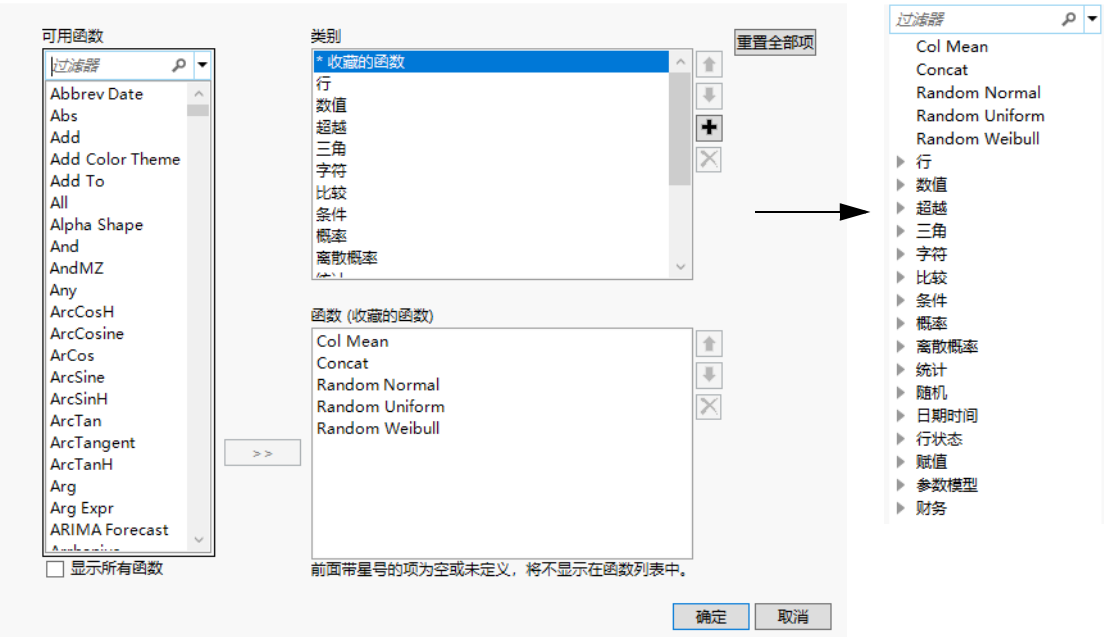
表 7.2 函数组 (续)

分组	包括的函数
随机	显示一个函数列表，这些函数基于预定义分布（如均匀分布、正态分布、Cauchy 分布等）生成随机数。还有一个函数可以随机化表行的顺序。请参见“在公式中引用 JMP 函数”附录中第 662 页的“随机函数”。
日期时间	显示一个函数列表，这些函数需要数据类型为**日期**的参数，日期被解释为自 1904 年 1 月 1 日起的秒数。日期时间函数返回日、星期或月份之类的值。这些函数还可以计算日期并查找日期间隔。请参见“在公式中引用 JMP 函数”附录中第 667 页的“日期时间函数”。
行状态	显示一个函数列表，这些函数可以指定或检测颜色、标记、标签、已隐藏、已排除或已选定这些行状态。请参见“在公式中引用 JMP 函数”附录中第 670 页的“行状态函数”。
赋值	显示一个函数列表，这些函数将赋值运算符右侧的值放入该运算符左侧的变量。请参见“在公式中引用 JMP 函数”附录中第 675 页的“赋值函数”。
参数模型	支持您为选定的列构造线性、交互作用或完全平方模型。选定模型类型之后，会出现一个窗口，您可以从中选择构造模型所基于的变量。按 Ctrl 键并点击可选择不止一项。请参见“在公式中引用 JMP 函数”附录中第 675 页的“参数模型函数”。
财务	支持您创建公式，用于计算本金偿还、利率、收益率等。请参见“在公式中引用 JMP 函数”附录中第 676 页的“财务函数”。

查找“公式编辑器”中经常使用的函数

您可以定制“函数”列表，以便于查找您通常使用的函数。函数显示在列表顶部。将频繁使用的函数显示在列表顶部、自行创建函数类别、删除函数、删除类别、在列表中上下移动函数。

图 7.10 定制函数列表



为喜好的函数快速添加书签

1. 右击“函数”列表中的所需函数。
2. 选择**添加为收藏项**。

为收藏项添加书签并指定位置

1. 从公式编辑器区域上方的红色小三角菜单中选择**定制函数列表**。
2. 选择 “可用函数”列表中的函数。
3. 点击右箭头按钮将其放入 “函数”列表。
4. 点击上下箭头重新排列收藏项。

重新排列或删除收藏项

1. 从公式编辑器区域上方的红色小三角菜单中选择**定制函数列表**。
2. 选择 “类别”列表中的**收藏的函数**。
3. 选择您要重新排列的收藏函数，然后点击上下按钮。
4. 选择您要删除的收藏函数，然后点击 X 按钮。

将函数添加至类别

若某个函数在公式中但不在函数列表中，则可以通过将其添加到类别来将其添加到列表中。

1. 选择要添加到函数列表的类别中的函数。
2. 右击该函数并选择**添加至类别**。
3. 选择类别。
4. 点击**确定**。

重新排列、添加或删除类别

1. 从公式编辑器区域上方的红色小三角菜单中选择**定制函数列表**。
2. 在“类别”列表中，选择某一类别，然后重新排列、添加或删除该类别。

重新排列或删除函数

1. 从公式编辑器区域上方的红色小三角菜单中选择**定制函数列表**。
2. 在“类别”列表中，选择包含您要重新排列或删除的函数的类别。
3. 点击上下按钮或 X 按钮重新排列或删除选定的函数。

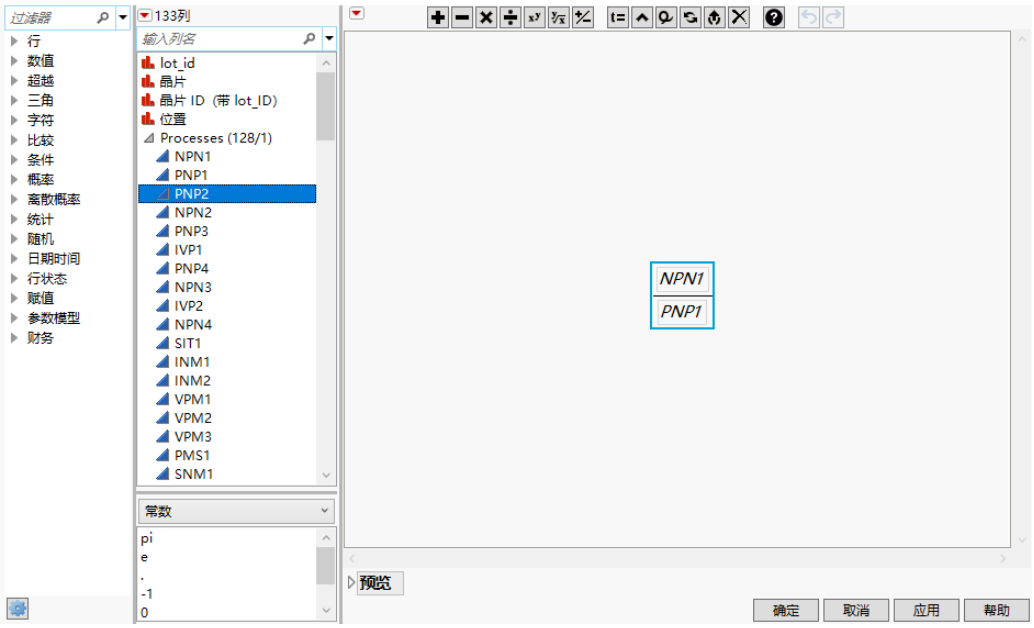
注意：删除函数只会从“函数”列表中删除它。您仍可以在公式中键入该函数。

复制和重复公式

“列”菜单中的“标准化特性”选项提供了一种方法，可以在复制列公式的同时递增公式中的每个列引用。该选项可以替代手动创建新的列公式（其中的列在每个后续列中递增 1）的方法。例如，若列 B 包含公式 Ax2，则标准化列 C 将包含公式 Bx2，并且标准化列 D 将包含公式 Cx2。

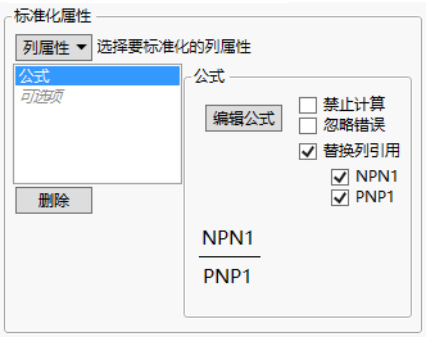
1. 选择**帮助 > 样本数据库**并打开 Semiconductor Capability.jmp。
2. 选择**列 > 新建列**。
3. 在**要添加的列数**旁边键入 3，然后点击**确定**。
4. 选择第一个新列，再选择**列**，然后选择**公式**。
5. 创建公式 NPN1/PNP1 并点击**确定**。

图 7.11 创建公式 NPN1/PNP1



6. 选择三个新列，然后选择**列 > 标准化特性**。
7. 在 “标准化属性”部分中，从列表中选择**公式**。
8. 选择**替换列引用**并保持选定 NPN1 和 PNP1。

图 7.12 “标准化属性”选项



注意：在本例中，我们想要同时更改 NPN1 和 PNP1。您可以取消选择您不想在后续公式中更改的列。

9. 点击**确定**。

   相对于第一个新列中的公式的公式将复制到第二个和第三个选定列中。

第二个和第三个选定列中的公式每次递增一列。

- 第一列中的原始公式：NPN1/PNP1
- 第二列中的相对公式：PNP1/PNP2

  在数据表中， PNP2 紧随 PNP1 之后。
- 第三列中的相对公式：PNP2/NPN2

  在数据表中， NPN2 紧随 PNP2 之后。

图 7.13 生成的数据表

NPN1	PNP1	PNP2	NPN2	X	Y	Z
114.55576	322.616752	469.39029	115.958473	0.3550831117	0.687310238	4.0479171331
120.043693	333.12808	437.781113	120.974137	0.3603529704	0.7609466694	3.618799212
114.92651	348.978757	532.128147	117.753124	0.3293223673	0.6558171354	4.5190151087
111.756446	268.548094	373.058629	114.092608	0.4161505811	0.7198549341	3.2697879024
111.545085	295.073166	338.900737	113.778098	0.3780251749	0.8706772615	2.9786113754
113.523596	323.833299	469.992171	116.488049	0.3505618367	0.6890184949	4.0346814413

注意：Semiconductor Capability.jmp 中的某些列已被删除以显示图 7.13 中的新列 X、Y 和 Z。

更多示例，请参见“设置 JMP 列属性”一章中第 287 页的“关于公式标准化的示例”。

对公式中的表达式排序

生成公式时，请记住所有函数都遵循下表所示的优先顺序，其中第一级的优先级最高。优先级较高的表达式先于优先级较低的表达式计算。若表达式中的各个运算符具有相同的优先级，则从左到右计算该表达式。必要时可以用圆括号越过其他优先规则，因为圆括号内的所有表达式始终优先计算。公式项不具有优先顺序，因为无法进一步计算公式项。表 7.3 显示了前六级优先顺序。

表 7.3 公式中运算符的优先顺序

第 1 级	圆括号
第 2 级	“函数”列表中的函数、And、Or、Not
第 3 级	*、÷、Modulo
第 4 级	+、-
第 5 级	比较：<、≤、=、≠、≥、>、≤ *x* <、< *x*、≤、< *x* <
第 6 级	逻辑运算符 &、| 和 !

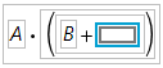
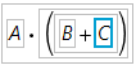
注意：当某个函数以表达式作为参数时，比起括在该函数之外的圆括号中的情况，该参数具有更高的优先级。

按优先顺序生成公式

生成公式时最好以充当参数的任意表达式开头。这是因为函数具有较高的优先级，并且始终与相应的参数组合在一起。同时还建议尽量遵循优先级从高到低的顺序来创建表达式。若需要圆括号，务必在输入要括起来的表达式之前先输入左圆括号。

例如，假定有一个包含 A、B 和 C 这三列的数据表，请使用以下步骤编写表达式 $A(B + C)$。请注意，该表达式不同于 $A \times B + C$，后者计算为 $(A \times B) + C$。

要输入表达式，请执行以下步骤：

1. 在“列”列表中点击列 A。
2. 点击“公式编辑器”小键盘中的乘号按钮 。
3. 输入左圆括号。
4. 在“列”列表中点击列 B。
5. 点击“公式编辑器”小键盘中的加号按钮 。 
6. 在“列”列表中点击列 C。 

由于优先顺序决定每个函数影响的参数，所以它也会影响表达式的分组。选择公式中的函数，检查优先顺序规则是如何应用的。

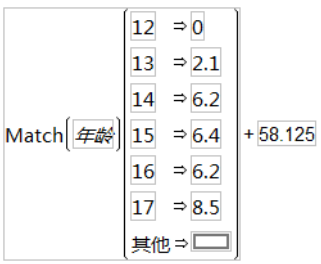
构建公式以提高计算效率

通常，没有必要在构建公式时考虑高效计算。无论公式结构如何，多数公式的计算都几乎是在瞬间完成的。这是因为在计算列值时，统计函数和常数表达式只计算一次。

不过，在创建条件表达式时，请记住 Match 与等效的条件函数 If 相比，计算速度更快，且占用的内存空间要少。（请注意，Match 忽略尾随空格，但 If 不忽略。）

例如，使用 Big Class.jmp，您可以根据儿童的年龄预测其身高，如图 7.14 所示。58.125 英寸是基准身高，在此基础上根据**年龄**变量的值来增加数量。

图 7.14 Match 条件语句比 If 函数的计算速度更快



“Match”条件语句比 If 函数的计算速度更快，这是因为**年龄**变量对数据表中的每一行仅计算一次。If 条件必须在每个 If 子句中为每行都计算年龄变量，直到子句计算结果为真。

JMP 公式编辑器选项

“公式编辑器”提供用于创建和管理公式的稳健选项。

- “计算 JMP 函数的导数”
- “简化复杂公式”
- “计算公式”
- “忽略公式中的错误”
- “查看用 JSL 显示的公式”

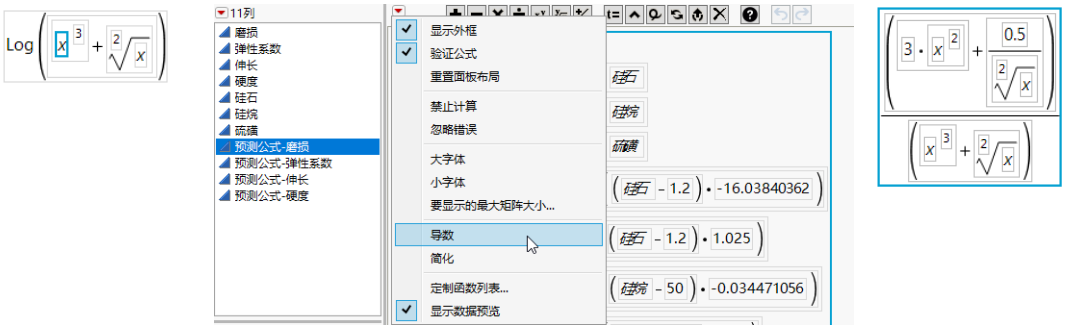
计算 JMP 函数的导数

JMP 公式编辑器可以查找并显示函数的导数。找到的导数是相对于您突出显示的函数参数（单个变量名）。因此，为了相对于 *x* 求导，*x* 必须是表达式中的一个参数。小键盘旁边的红色小三角菜单包含“导数”命令。

要计算导数，请执行以下步骤：

1. 输入一个函数。
2. 突出显示某个变量。
3. 从菜单中选择**导数**。

图 7.15 使用 “导数”选项



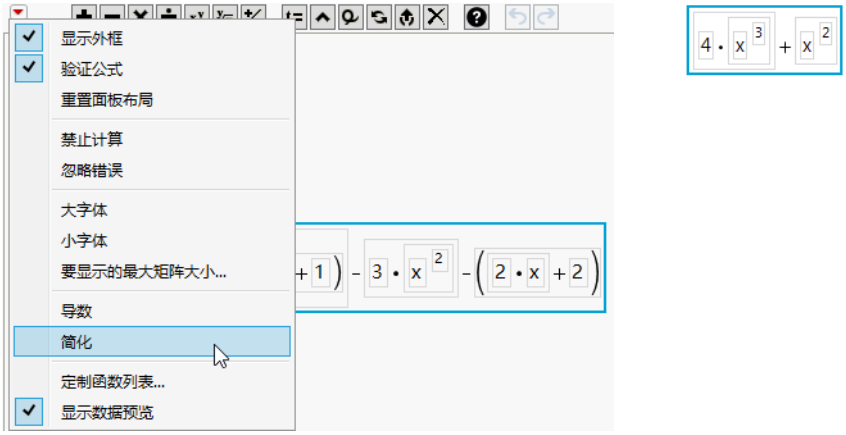
简化复杂公式

当“公式编辑器”包含复杂公式时，JMP 可使用多种代数规则简化复杂公式。它可以执行查找常数表达式、将乘法分配到加法上、组合公式项等操作。

要简化复杂公式，请执行以下步骤：

1. 选择部分或整个公式。
2. 点击小键盘旁边的红色小三角。
3. 从列表中选择**简化**，如图 7.16 所示。

图 7.16 选择 “简化”（左侧）将生成简化的公式 （右侧）



计算公式

默认情况下， JMP 会计算您创建的每个公式。您可以禁止这种计算，也可以在完成公式创建之前使用计算。

禁止计算

禁止计算对于生成复杂公式而言是一种很有用的公式编写模式。您可以关闭计算功能，然后编写公式的各个部分，再计算该公式只是为了检验该公式。其特别之处在于，您可以关闭“公式编辑器”，以后某个时间再打开它继续生成公式，而 JMP 并不会计算该公式。

要禁止公式计算，请点击小键盘旁边的红色小三角，然后选择**禁止计算**。禁止计算时，公式图标 灰显。

请注意以下事项：

- 若该图标显示在“公式编辑器”中红色小三角的右侧，则表示对该公式禁止公式计算。
- 若该图标显示在列面板中列名的旁边，则表示该列中的值是公式计算的结果。禁止公式计算时，该公式图标 灰显。请参见“开始使用 JMP”一章中第 40 页的“表示列特征和属性的图标”。
- 若要为所有公式关闭公式计算功能，请点击数据表名称旁边的红色小三角并选择**禁止公式计算**。要重新计算公式，请点击数据表名称旁边的红色小三角并选择**重新运行公式**。您还可以为包含待执行公式的每列点击“公式编辑器”中的“**应用**”按钮。

- 禁止公式计算时，若编辑公式并点击**应用**，系统会提示您是否要计算该公式。要允许公式计算一次，请点击**此时计算公式**。

忽略公式中的错误

一旦构建了公式并点击**确定**，JMP 会检查公式可能存在的错误，并在发现任何错误时向您发出警示。每个错误都会显示一条错误消息，并询问您是否忽略之后的错误。

有时您希望在编写公式的过程中不显示错误消息。例如，您可能需要某些行的计算值，但不希望看到导致错误的各行的错误消息。

要忽略错误，请执行以下步骤：

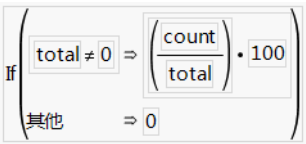
1. 创建一个公式。
2. 点击小键盘旁边的红色小三角并选择**忽略错误**。

查看用 JSL 显示的公式

您可以选择在**文本模式**中输入或更改公式的任何部分。文本模式以 JMP 脚本语言 (JSL) 显示公式。双击公式时，整个公式（或其任何公式项）将显示在“脚本编辑器”中。若公式较长，该公式会在最大化的“脚本编辑器”窗口中打开。否则，您可以点击**最大化编辑器**按钮 将该窗口最大化。

公式中的任何元素都可以显示在“脚本编辑器”中，随后接受编辑。编辑公式脚本命令后，在该公式外点击（或点击最大化窗口下方的**确定**）可查看其带格式的形式并保存更改。要以文本形式输入 If 语句，请为语句中的每个 If/Then 子句添加成对的参数，并按需为 else 子句添加最后一个参数。在文本形式中，图 7.17 中的 If 语句如下所示：`If(:total!=0, (:count/:total)*100, 0)`。

图 7.17 公式模式中的 If 语句



编辑公式

使用小键盘编辑公式；使用箭头选择表达式；剪切、复制和粘贴；删除函数和拖动表达式。

- “更正公式中的错误”
- “在公式中选择表达式”
- “删除函数公式”
- “剪切、复制和粘贴公式”

- “点击并拖动公式”

更正公式中的错误

若在 JMP 公式编辑器中输入公式时出错，请按 Ctrl+Z。这将反转上一个（可撤销的）命令的效果。

有助于修改公式的其他命令包括：

- 点击“公式编辑器”小键盘上的删除按钮 ( ) 可删除选定的表达式。
- 使用剪切、复制和粘贴键盘快捷命令，或右击公式中突出显示的部分并从菜单中选择**剪切**、**复制**或**粘贴**。
- 拖动公式各部分可将它们重新排列。

在公式中选择表达式

使用键盘箭头键可选择要在 JMP 公式编辑器中编辑的表达式。您还可以在未使用圆括号或外框选项时，使用箭头键查看公式的优先顺序。（请参见第 364 页的“隐藏和显示分级显示的公式”。）

点击表达式中的某个运算符（+、–、*、÷）可选择该运算符及其运算对象。项周围将显示蓝色外框。一旦选定某个运算符：

- 使用左右箭头键可以将选定项在表达式中具有相同优先级的其他关联运算符之间移动。
- 使用上箭头可通过向选定内容添加优先级更高的操作数和运算符来扩展当前选择。
- 使用下箭头可通过从选定内容中删除操作数和运算符来缩减当前选择。

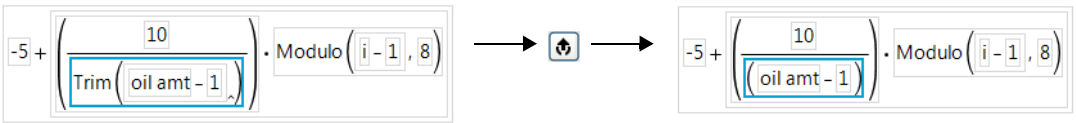
删除函数公式

删除函数也会随之删除其参数；从函数中删除某个必需参数时或在函数中缺失某项时会删除函数本身。在 JMP 公式编辑器中，您可以剥除一个函数，将该函数从其参数中删除。

要从单个参数中剥除函数，请执行以下步骤：

1. 选择该函数。
2. 点击“公式编辑器”小键盘中的剥除按钮 。或者，使用手形工具将参数拖至其函数上方。

图 7.18 剥除参数



3. 完成公式更改。

4. 点击**应用**，新值将自动填充数据表列。
5. 一旦创建公式，就可以更改公式所引用的列中的值。JMP 自动重新计算公式列中的所有受影响值。

剪切、复制和粘贴公式

您可以剪切或复制任意表达式或整个公式，并将其粘贴至 JMP 公式编辑器中的另一个公式。使用剪切、复制和粘贴快捷命令，或右击公式中突出显示的部分并从菜单中选择**剪切**、**复制**或**粘贴**。剪切、复制和粘贴公式时适用以下规则：

- 在粘贴至另一个公式显示区时，公式以带格式的形式显示。
- 公式在剪贴板上保存为 JSL 语句。因此，若将其复制到其他应用程序，它会显示为 JMP 脚本语言 (JSL) 语句。

注意：按 Shift 键并点击红色小三角可显示一个名为**复制为 SAS 公式**的命令。

点击并拖动公式

在 JMP 公式编辑器中，您可以将公式的任意可选部分拖至可选的任何其他位置。

要在公式中点击并拖动，请执行以下步骤：

1. 将箭头光标置于表达式中。
2. 点击该表达式。
3. 在公式中拖动。目标表达式具有虚线外框。

图 7.19 拖动表达式

4. 将选定表达式拖至新的目标位置。选定的表达式将移至新位置，并替换该位置上现有的表达式。

在 JMP 公式编辑器中定制公式

您可以采用多种方式来定制 JMP 公式编辑器中的公式。

- “隐藏和显示分级显示的公式”
- “更改字体大小公式”
- “指定矩阵大小公式”

- “更改 JMP 公式的方向”
- “打开和关闭参数公式”

隐藏和显示分级显示的公式

默认情况下，JMP 会为公式中的特定项分级显示。这称为**加框**。当您想要选择和修改公式的特定部分或需要确定计算的先后顺序时，加框很有帮助。

要启用或禁用加框，请执行以下步骤：

1. 生成公式。
2. 点击小键盘旁边的红色小三角并选择**显示外框**。当**显示外框**旁边的菜单上出现一个复选标记时，外框即显示在公式中。未出现复选标记则不会显示外框。

更改字体大小公式

要递增或递减用来显示格式化公式的字体，请执行以下步骤：

1. 点击小键盘旁边的红色小三角。
2. 选择**大字体**或**小字体**。
3. 重复该过程，进一步增大或减小字体。

指定矩阵大小公式

通过从红色小三角菜单中选择**要显示的最大矩阵大小**，您可以设置矩阵的最大宽度和高度。若宽度或高度大于指定大小，则显示矩阵维而不是数据。您仍可以通过在文本编辑器窗口中编辑该公式来查看数据。

更改 JMP 公式的方向

默认情况下，JMP 会确定公式大小，并以最适合的方向（水平或垂直）将其显示在“公式编辑器”中。不过，若创建较长的公式，则最好用不同的布局来显示公式。

要更改公式方向，请执行以下步骤：

1. 生成公式。
2. 突出显示某个参数或公式。
3. 右击已突出显示的项。
4. 选择**方向**。
5. 从 “方向”选项中选择：最佳、水平或垂直。

打开和关闭参数公式

若公式太大，无法在屏幕中完整显示，您可以关闭公式和参数。

要关闭参数，请执行以下步骤：

1. 生成公式。
2. 突出显示某个参数或公式。
3. 右击已突出显示的项。
4. 从出现的菜单中选择**关闭**或**关闭参数**。

示例：在 JMP 公式编辑器中生成公式

要进一步了解如何生成公式，请查看以下示例和教程。

- “使用 JMP 公式编辑器中的基本功能”
- “使用局部变量公式”
- “使用 Munger 函数公式”
- “使用 Match 条件函数公式”
- “使用删除表达式按钮公式”

使用 JMP 公式编辑器中的基本功能

下面的示例使用 Big Class.jmp 样本数据表，为您演示如何使用“公式编辑器”的基本功能。您可以找到安装 JMP 时所安装的 Sample Data 文件夹中的 Big Class.jmp。

Big Class.jmp 包含名为**体重**的一列。假定您想要一个新列，该列计算标准化的体重值。

要使用公式来创建该列以获取该列的值，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**列** > **新建列**。
3. 在 “列名”旁边的框中键入新名称**标准体重**。
4. 从列属性菜单中选择**公式**。
5. 通过点击公式编辑区域中的空公式元素将其选中。

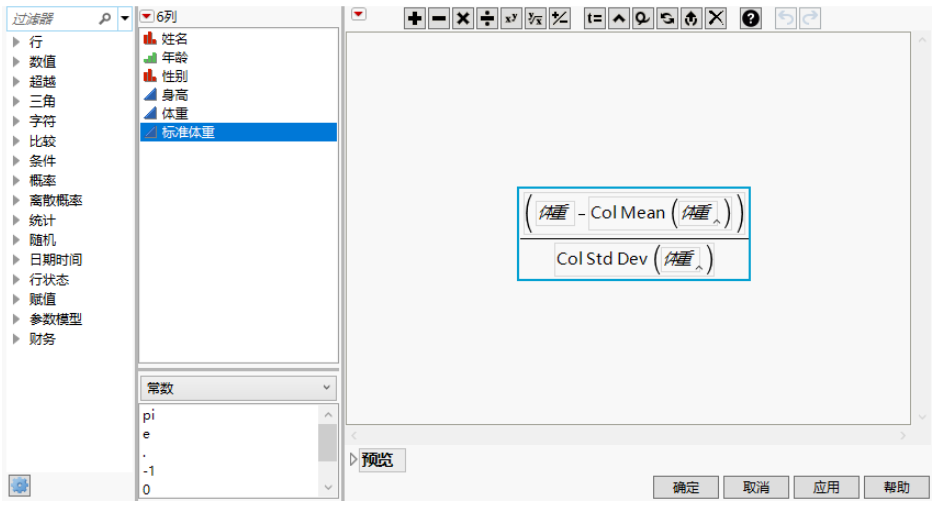
   创建公式时若选中 “显示外框”，选定的公式部分将显示在细蓝线框内。请参见第 364 页的“隐藏和显示分级显示的公式” 。相对于所点击位置的最小嵌套框内的所有项都将被选中，后续操作应用于这些组合元素。

接下来，输入对**体重**值进行标准化的公式。

1. 在选定初始缺失项时，点击“列”列表中的**体重**。

2. 点击 “公式编辑器” 小键盘中的减号按钮 ➖。
   减号之后将出现新的缺失项，如图 7.20 所示。
3. 再次点击**体重**。
4. 点击 “函数” 列表，然后选择**统计**。
5. 从统计列表中选择 Col Mean。
6. 选择整个表达式。
   蓝框现在应该包含整个公式。
7. 点击小键盘中的除号按钮 ➗。
   这将为整个表达式生成一个选定的缺失分母。
8. 再次点击列选择器列表中的**体重**。
   该列将被选定到分母中。
9. 从 “统计” 列表中选择 Col Std Dev。

图 7.20 生成公式



10. 通过点击**确定**关闭 “公式编辑器”。
11. 通过点击**确定**关闭 “列信息” 窗口。

在数据表中，新的**标准体重**列将填充计算所得的值。若**体重**值更改，计算所得的**标准体重**值也会自动重新计算。

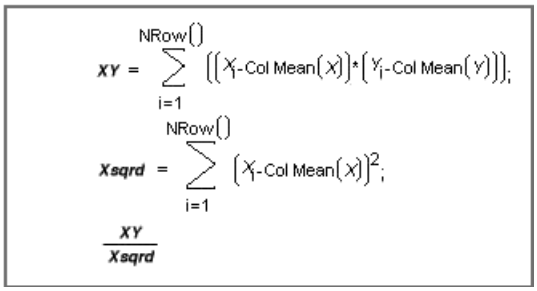
使用局部变量公式

假定您想要使用图 7.21 所示的标准公式计算 y 对于 x 的简单线性回归中的斜率。一种计算方法是创建两个局部变量，并将它们命名为 XY 和 Xsqrd（请参见第 346 页的 “在公式中使用局部

变量”。）然后分别将它们指定给斜率公式的分子和分母计算。用分号分隔各个指定，如图 7.21 所示（“公式编辑器”中的语句实际上是 JSL 编程语句，所以必须用分号分隔公式中的多个语句。）斜率计算简化为 XY 除以 Xsqrd。

图 7.21 标准斜率公式中的局部变量

$$\frac{\sum(x-\bar{x})(y-\bar{y})}{\sum(x-\bar{x})^2}$$



注意：您还可以使用屏幕上小键盘上的 t= 按钮创建局部变量。点击该按钮可在表达式中创建具有默认名称的局部变量，并在变量后放置一个分号。默认名称为 *t1*，后续的局部变量名为 *t2*、*t3*，依此类推。通过双击并编辑这些默认名称可以更改这些名称。不过，重命名每个变量实例时务须谨慎，避免出错。

使用 Munger 函数公式

下面的示例显示 Munger 函数的用法。在这些示例中，假定有一个包含名称的字符列，“Veronica Layman”是其中一个值。要简化这些示例，文字名称“Veronica Layman”只是搜索字符串而不是列名。

有关如何将字符函数（如 Munger）并入公式的说明，请参见“在公式中引用 JMP 函数”附录中第 639 页的“字符函数”。

插入字符

该 Munger 示例在名和姓之间查找空格，并插入中间名首字母“J.”。公式 Munger("Veronica Layman", 1, " ", " J. ") 插入中间名首字母 J. 并生成结果 Veronica J. Layman。

Munger 函数要求用双引号引用文字字符串，包括由空格构成的字符串以及包含前导或尾部空格的字符串。

删除字符

要删除字符串中的一个或多个字符，请执行以下步骤：

1. 将要删除的字符指定为 Munger 函数中的查找字符串。
2. 输入一个空的替换字符串：只有两个引号，中间没有任何内容。

例如，函数 Munger("Veronica Layman", 1, "onic", "") 从 Veronica 中删除“onic”，并生成结果 Vera Layman。

注意：括在双引号内的空（无值）字符串的“替换”字段不同于不含值的“替换”字段。若整体删除替换字符串，Munger 将在“公式编辑器”窗口中显示参数名称 (“Replace”)，并且在求值时就好像该可选函数并不存在。生成的数据类型还可能从字符型改为数值型，具体取决于“查找/长度”参数的值。

查找值的位置（索引）

当查找/长度字段包含字符时，Munger 会像索引函数那样操作，并返回搜索字符串的第一个实例（若存在）的数值位置。例如，Munger("Veronica Layman", 1, " ") 会搜索单个空格并在第 9 个位置找到该空格。若找不到搜索字符串，Munger 将返回零。Munger 的这种用法可以生成与 Contains 函数相同的结果，如“在公式中引用 JMP 函数”附录中第 639 页的“字符函数”所示。

查找子字符串

Munger 可以提取子字符串。例如，要只提取名字，Munger("Veronica Layman", 1, 8,) 将从第一个位置开始，一直读取到第八个位置。剩余字符将被忽略，因为未定义替换参数。这将生成“Veronica”，并生成与 Substring 相同的结果，如“在公式中引用 JMP 函数”附录中第 639 页的“字符函数”所示。

另一种查找子字符串的方式是使用一个起始值、任意的负查找值以及一个**无替换**参数。Munger("Veronica Layman", 9, –1,) 返回“Layman”。

使用 Match 条件函数公式

该示例可引导您熟悉如何使用 Match 条件函数。

假定您需要对 Hot Dogs.jmp 样本数据表中的名义型变量**原料类型**应用 Match 条件函数。

1. 在“公式编辑器”中，从“列”列表中选择**原料类型**。
2. 从“函数”列表中选择**条件**。
3. 选择 Match。
4. 选择**根据数据添加匹配参数**。

   值将自动填充。

注意：不必逐一完成第 2 步到第 4 步，而是可以按 Shift 键，从“函数”列表中选择**条件**，然后选择 Match。

图 7.22 自动填充 Match 条件语句

若不希望系统为您填充值，请从 Match 列表中选择**请不要添加**而不是**根据数据添加匹配参数**。

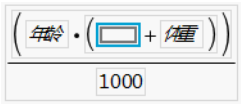
使用删除表达式按钮公式

以下步骤显示了反复点击删除表达式 （或剥除）按钮 的结果：

1. 从某个公式入手。

2. 选择一个公式元素。

3. 点击删除表达式按钮 。



4. 再次点击删除表达式。

5. 再次点击删除表达式。

6. 再次点击删除表达式。

7. 再次点击删除表达式。

8. 再次点击删除表达式。

9. 再次点击删除表达式。

JMP 公式编辑器的键盘快捷方式

表 7.4 说明了可以用在 JMP 公式编辑器中的键盘快捷方式。

表 7.4 “公式编辑器”中的键盘快捷方式

插入项	使用键盘快捷方式
缺失元素 (•)	Delete
下标	[
指数	^（按 Shift 键并键入 6。）
一对圆括号 ()	(
一对引号 " "	"
选定字段之前的新参数	Alt +
*（乘号）	小键盘或键盘上的 *
+	小键盘或键盘上的 +
−	小键盘或键盘上的 -
÷	小键盘或键盘上的 /
+/−	按 Shift+-（减号）
和	&
非	!
或	|
乘方	^
新参数	,

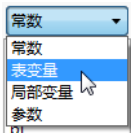
表 7.4 “公式编辑器”中的键盘快捷方式 (续)

插入项	使用键盘快捷方式
x<y	<
x==y	=
x>y	>

JMP 公式编辑器的术语表

以下项用在 “公式编辑器”中：

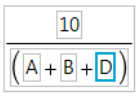
元素 元素浏览器列表中显示的常数、表变量、局部变量或参数的名称。



参数 函数运算针对的任意元素或整个表达式 （包括数学运算对象） 。参数始终与函数组合在一起。要查找充当函数参数的表达式，请在公式中选择该函数。

公式项 表达式不可细分的部分，如常数和变量。

表达式 公式中可选作单一单位的任何部分，包括公式项、缺失项、与函数参数组合在一起的函数，以及整个公式。



子句 条件函数中的完整片段。

函数 对一个或多个参数执行特定操作的数学或逻辑运算。函数包括“函数”列表中的大多数项以及所有小键盘运算符。函数始终对选定的表达式执行运算，并且参数始终与函数组合在一起。要查找充当函数参数的表达式，请在公式中选择该函数。

加框的分组还显示如何应用优先顺序规则，并显示在删除函数时随之删除的参数。请参见第 357 页的“对公式中的表达式排序”。

缺失项 代表表达式的任何空占位符，用一个空框来表示。

缺失值 已排除数据或空数据，由表示数值数据的缺失值标记 (•) 或表示字符数据的空字符串构成。

	姓名	年龄	性别	身高	体重
1	KATIE	•	F	59	95
2		12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84

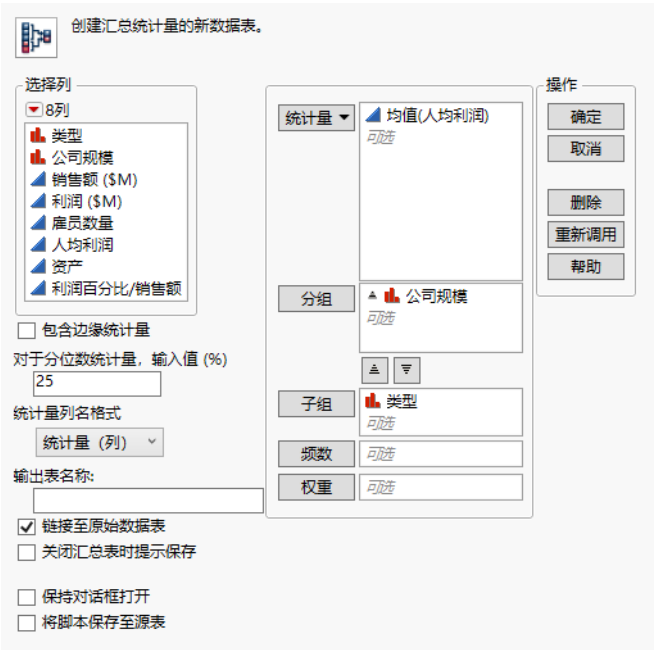
第 8 章

汇总数据

表汇总命令

本章说明如何创建汇总数据表，该表中包括均值和中位数、标准差、最小值和最大值之类的汇总统计量。

图 8.1 Companies.jmp 的汇总表



	公司规模	行数	均值(人均利润, Computer)	均值(人均利润, Pharmaceutical)
1	big	9	4530.478	17140.699
2	medium	7	-3462.506	24035.115
3	small	16	7998.815	38337.191

目录

汇总列 375
汇总启动窗口 375
创建汇总表 375
向现有汇总表添加统计量列 378
解释汇总统计量 379
示例：创建带分组的汇总表 381

汇总列

“表 > 汇总”命令计算各种汇总统计量，包括均值和中位数、标准差、最小值和最大值等。

汇总表具有以下特征：

- 您指定的分组变量的每个水平都对应一行。若未指定分组变量，整个数据表对应一行。
- 若有多个分组变量，所有分组变量的每种水平组合在表中都有对应的行。
- 每个分组变量都有自己的一列，此外表中还在名为**行数**的列中包含频数统计，即每个分组水平的计数。
- 汇总表可链接至其源表。选择汇总表中的行时，相应的行将在其源表中突出显示。
- 若源表中的列包含值标签，值标签将显示在新表中。
- 关闭汇总表时并不会将它保存，只有您选择**文件** > **另存为**并为汇总表指定名称和位置后才会保存它。

汇总启动窗口

统计量 定义源表中数值列的汇总统计量（如均值、标准差和中位数）。请参见第 376 页的“添加汇总统计量”。

分组 按组汇总统计量。请参见第 376 页的“使用一个或多个分组列”。

子组 按子组汇总统计量。请参见第 378 页的“通过添加子组变量创建汇总统计量的双向表”。

频数 标识一个列，该列的值为每行分配频数。

权重 标识其值用于为每行分配权重的列。

创建汇总表

要创建汇总表，请执行以下步骤：

1. 打开一个数据表。
2. 选择**表** > **汇总**。
3. 突出显示要汇总的列。

注意：有关红色小三角菜单中选项的详细信息，请参见“开始使用 JMP”一章中第 50 页的“‘列过滤器’菜单”。

4. 添加汇总统计量、组、子组、频数变量、权重变量，然后选择所需的任意选项。
   - 第 376 页的“添加汇总统计量”
   - 第 376 页的“使用一个或多个分组列”
   - 第 377 页的“使用分位数统计量”

- 第 377 页的“更改统计量列名的格式”
- 第 378 页的“链接至原始数据表”
- 第 378 页的“关闭链接的汇总表时提示保存”
- 第 378 页的“保持“汇总”窗口打开”
- 第 378 页的“通过添加子组变量创建汇总统计量的双向表”

5. 通过在**输出表名称**旁边的框中键入名称，为汇总表命名。
6. 点击**确定**。

注意：

- 当汇总表链接至原始数据表时，您无法编辑汇总表中的数据，因为这将修改和破坏原始数据。要编辑该数据，需创建汇总表的子集，或先保存再关闭该表，然后重新打开它。
- 选择具有“单位”列属性的列时，该值将包含在列标题中。

添加汇总统计量

您可以添加一些列来显示源表中的任意数值列的汇总统计量（如均值、标准差、中位数等）。

1. 在“汇总”窗口中，突出显示要在统计量计算中使用的列。
2. 点击**统计量**按钮。
3. 从“统计量”列表中选择一个标准的一元描述性统计量。第 379 页的“解释汇总统计量”中对统计量进行了说明。

使用一个或多个分组列

若希望按组汇总统计量，请突出显示要作为分组变量的列，然后点击**分组**将该变量移入分组变量列表。相关示例，请参见第 381 页的“示例：创建带分组的汇总表”。若仅添加分组变量，汇总表将显示每个组的计数。

更改分组变量的顺序

要更改分组变量的顺序（升序或降序），请在分组变量列表中选择某个变量，然后点击升序或降序按钮 ( )。该变量旁边的图标随之更改以指示排序顺序。

您还可以使用“值顺序”列属性更改分组变量的顺序。请参见“设置 JMP 列属性”一章中第 267 页的“值顺序”。

包含边缘统计量

要向输出列添加分组变量的边缘统计量，请点击**包含边缘统计量**旁边的框。除了为每个分组变量添加边缘统计量，JMP 还会在表末尾添加汇总第一个分组变量各水平的行。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**表** > **汇总**。

3. 选择**利润 ($M)** 并点击**统计量**。
4. 选择**均值**。
5. 选择**类型**和**公司规模**，并点击**分组**。
6. 选择**包含边缘统计量**。
7. 点击**确定** （或**创建**）。请参见图 8.2 中的左图。

图 8.2 包含和不含边缘统计量的汇总表

	类型	公司规模	行数	均值(利润 ($M))
1	Computer	big	4	1089.93
2	Computer	medium	2	-85.75
3	Computer	small	14	44.94
4	Computer		20	240.87
5	Pharmaceutical	big	5	894.42
6	Pharmaceutical	medium	5	698.98
7	Pharmaceutical	small	2	156.95
8	Pharmaceutical		12	690.08
9			32	409.32

	类型	公司规模	行数	均值(利润 ($M))
1	Computer	big	4	1089.93
2	Computer	medium	2	-85.75
3	Computer	small	14	44.94
4	Pharmaceutical	big	5	894.42
5	Pharmaceutical	medium	5	698.98
6	Pharmaceutical	small	2	156.95

将包含边缘统计量的汇总表（左图）与不含边缘统计量的汇总表（右图）加以比较。您可以看到左图不仅添加了边缘统计量，还有一行显示总共有 32 个 Computer 和 Pharmaceutical 公司。

使用分位数统计量

要添加特定的分位数统计量，请执行以下步骤：

1. 在**对于分位数统计量，输入值 (%)** 下面的框内，键入第一个分位数的所需分位数值 (%)（例如，25）。
2. 选择适用的列并点击**统计量**。
3. 选择**分位数**。
4. （可选）为任何其他分位数重复该过程。

更改统计量列名的格式

要更改汇总表中统计量列名的格式，请从**统计量列名格式**菜单中选择一种格式。表 8.1 显示可用选项。假定您要创建公司利润平均值的汇总表。原始列名为**利润 ($M)**。

表 8.1 统计量列名格式选项和示例

选项	示例
统计量（列）	均值（利润 ($M)）
列	利润 ($M)

表 8.1 统计量列名格式选项和示例 (续)

选项	示例
列的统计量	“利润 ($M)”的均值
列统计量	“利润 ($M)”均值
统计量	均值

链接至原始数据表

您可以选择是否将汇总表链接至原始数据表。默认情况下，**链接至原始数据表**选项处于选定状态。若想要编辑汇总表中的数据，请取消选择**链接至原始数据表**选项。当汇总表链接至原始数据表时，您无法编辑汇总表中的数据，因为这将修改和破坏原始数据。要编辑该数据，需创建汇总表的子集，或先保存再关闭该表，然后重新打开它。

在链接表中，若将汇总表中的列拖到原始数据表中新列的列标题，值将展开，就像按分组列匹配的那样。

关闭链接的汇总表时提示保存

要在关闭一个未保存的链接汇总表时提示保存它，请选择**关闭汇总表时提示保存**。当您在“汇总”窗口中选择该选项时，它在“表”首选项中被选定。若您以后不想系统提示保存链接的汇总表，请取消选择该首选项。

保持“汇总”窗口打开

若选择**保持对话框打开**选项，“汇总”窗口将在您点击**应用**后保持打开。请注意，一旦选择该选项，确定按钮将被创建按钮取代。

通过添加子组变量创建汇总统计量的双向表

1. 突出显示要作为嵌套变量的列。这些列是您要的“子组变量”。
2. 点击**子组**将变量移入子组列表。
3. 突出显示要按子组汇总统计量的列。
4. 在统计量列表中，选择所需的特定统计量。
5. 点击**确定**。

有关统计量的类型的详细信息，请参见第 379 页的“解释汇总统计量”。

向现有汇总表添加统计量列

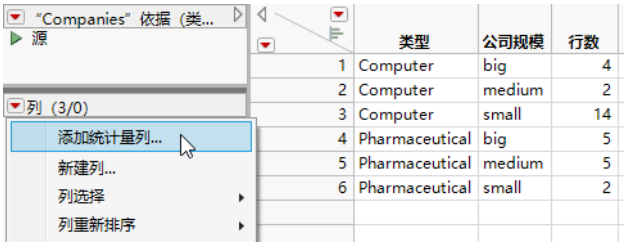
创建汇总表后，您可以再增加一些列。这些列是源表中的任意数值列的描述性汇总统计量。这需要从现有汇总表中，点击数据网格中的上部红色小三角，然后选择**添加统计量列**。

向现有表添加统计量列的示例

假定已经创建了一个汇总表，您又想向现有汇总表添加更多统计量。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**表** > **汇总**。
3. 选择**类型**和**公司规模**，并点击**分组**。
4. 点击**确定**。
5. 点击列面板中的红色小三角，选择**添加统计量列**。

图 8.3 自数据表内部创建汇总统计量列



出现经过修改的“汇总”窗口。

6. 选择所需列，点击**统计量**，并选择所需的特定统计量。对于本示例，请选择**人均利润**并点击**统计量**，然后选择**均值**。
7. 点击**确定**。

图 8.4 包含汇总统计量列的汇总表的示例

	类型	公司规模	行数	均值(人均利润)
1	Computer	big	4	4530.478
2	Computer	medium	2	-3462.506
3	Computer	small	14	7998.815
4	Pharmaceutical	big	5	17140.699
5	Pharmaceutical	medium	5	24035.115
6	Pharmaceutical	small	2	38337.191

汇总统计量列

“均值（人均利润）”列添加至现有的汇总表。

解释汇总统计量

您可以通过点击统计量按钮并从菜单中选择某个选项，为源表中的任意数值列添加描述性汇总统计量列。

统计量菜单提供数值列的以下汇总统计量：

数目 非缺失值的个数。

均值 列值的算术平均值。它是非缺失值的总和 （并且在已定义的情况下，乘以权重变量）除以权重和。

标准差 样本标准差，它针对非缺失值计算，使用指定的权重变量来计算。它是样本方差的平方根。计算公式如下：

$$s = \sqrt{s^2} \text{ 其中}$$

$$s^2 = \sum_{i=1}^{N} \frac{w_i(y_i - \bar{y}_w)^2}{N-1}$$

$$\bar{y}_w = \text{加权均值}$$

最小值 列中最小的非缺失值。

最大值 列中最大的非缺失值。

极差 最大值和最小值的差值。

占合计的百分比 每组占合计数的百分比。或为每组的列的非缺失值占合计数的百分比 （若按此方式指定）。

缺失值个数 缺失值的个数。

类别数 不同类别的个数。

总和 列中所有值的总和。

权重和 列中所有**权重**值的总和。（请参见“设置 JMP 列属性”一章中第 263 页的“JMP 中的列属性”。）或者，若没有为任何列指定权重角色，**权重和**为非缺失值的总数。

方差 为非缺失值计算的样本方差。它是均值偏差的平方和除以非缺失值个数减 1 得出的值。

标准误差 均值的标准误差。它是标准差除以**数目**的平方根所得的值。若为列指定了权重角色，则分母是权重和的平方根。

CV （变异系数） 一种散度测度，它是标准差除以均值再乘以 100。

中位数 第 50 百分位数，在该值处有一半的数据低于第 50 分位数，一半数据高于或等于第 50 分位数 （中位数）。

绝对偏差中位数 数据中位数的绝对偏差的中位数。

几何均值 数据的积的第 n 个根。例如，几何均值通常用于计算利率。当数据包含偏斜分布中的较大值时，该统计量也很有用。

注意：负值导致缺失数字，零值 （无负值）导致零。

四分位间距 第三和第一四分位数之间的差值。

分位数 参数的特定百分比小于或等于的值。例如，75% 的数据小于第 75 分位数。汇总窗口提供一个编辑框，供您输入所需的分位数百分比。

直方图 生成不同组的直方图。直方图的图像保存在数据表的 “表达式”列中。

示例：创建带分组的汇总表

假定研究人员正在使用 Companies.jmp，该表按**类型**和**规模**对公司分组。通过从安装 JMP 时安装的 Sample Data 文件夹打开 Companies.jmp，继续以下示例中的操作。

假定研究人员要执行以下任务：

- 创建一个表，其中显示小型、中型和大型计算机公司和医药公司的人均利润。换言之，创建一个表，在其中为各种规模的公司创建对应的行，为各种类型的公司的人均利润创建对应的列。
- 创建该表，在单元格中包含子组（由行列交叉点定义）的均值。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**表** > **汇总**。
3. 选择**公司规模**并点击**分组**。

   研究人员之所以将**公司规模**选作分组变量，是因为他想要该列中的值成为新表中的行。
4. 选择**人均利润**并点击**统计量**。
5. 选择**均值**。
6. 选择**类型**并点击**子组**。

   这将指示 JMP 为子组变量（**类型**）的每个水平（computer、 pharmaceutical）的人均利润（**均值 （人均利润）**）创建一列。

图 8.5 子组的汇总统计量

	公司规模	行数	均值(人均利润, Computer)	均值(人均利润, Pharmaceutical)
1	big	9	4530.478	17140.699
2	medium	7	-3462.506	24035.115
3	small	16	7998.815	38337.191

注意：汇总表以“组”列名称命名。若列名称包含文件名中不允许的字符，用特殊字符替代连字符。

第 9 章

JMP 报表

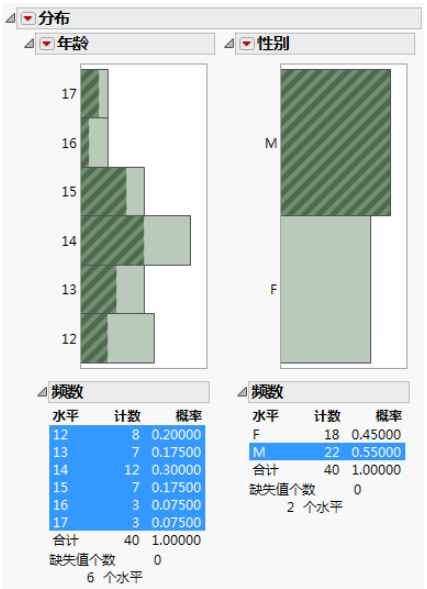
导航和定制报表窗口

大多数 JMP 平台在报表窗口中显示您的分析结果。报表窗口包含通常链接到数据表的图形和统计报表。当您在图形或报表中选择数据时，该数据在数据表中突出显示。图 9.1 显示报表窗口和数据表中选定数据的一个示例。

本章说明如何在报表窗口中定制颜色、线、数据点和其他图形元素。

图 9.1 JMP 报表窗口和数据表的示例

	姓名	年龄	性别	身高	体重
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	79
9	BARBARA	13	F	60	112
10	ALICE	13	F	61	107
11	SUSAN	13	F	56	67
12	JOHN	13	M	65	98
13	JOE	13	M	63	105
14	MICHAEL	13	M	58	95
15	DAVID	13	M	59	79
16	JUDY	14	F	61	81
17	ELIZABETH	14	F	62	91
18	LESLIE	14	F	65	142
19	CAROL	14	F	63	84
20	PATTY	14	F	62	85
21	FREDERICK	14	M	63	93
22	ALFRED	14	M	64	99



目录

导航 JMP 报表 386
使用手形工具 387
JMP 报表的显示选项 387
显示和隐藏 JMP 报表部分 389
合并多个报表 389
重命名报表中的标题 389
增大报表中的字体 390
调整报表窗口的大小 390
JMP 报表中的红色小三角选项 390
报表中的“本地数据过滤器” 391
报表中的“重新运行”菜单 393
报表中的“保存脚本”菜单 393
数据过滤器 396
过滤器列的类型 397
过滤模式 398
“数据过滤器”控制面板 399
“数据过滤器”的红色小三角选项 401
数据过滤器中的动画 405
针对数据过滤器中的变量的红色小三角选项 406
重新格式化报表中的表 409
修改 JMP 报表中的显示框属性 410
将 JMP 报表中的表转换为矩阵 411
在报表中使用条件格式 411
选择图形中的点 416
在图形中选择行 417
在 JMP 图形中选择矩形点区域 417
在 JMP 图形中选择点的不规则形状区域 418
在图形中使用标记 419
更改标记的形状 419
更改标记的颜色 420
更改标记的大小 420
处理拥挤的标记 420
为标记添加轮廓 421
更改标记的选择模式 421
指定标记的透明度 422
排除和隐藏标记 423
将图像用作标记 423

基于值更改标记的形状或颜色 424
更改图形外观 426
调整图形大小 426
放大图形 427
更改图形中的线条粗细 428
更改图形中的背景颜色 428
更改直方图直条的颜色 428
显示图形中的坐标和临时参考线 428
在图形中滚动和调整轴尺度 429
定制图形中的轴和轴标签 429
更改报表中的值顺序 439
更改图形中选定对象的模式和格式 440
将图形添加至数据表 440
定制图形中的标记、线条、文本等元素 441
将图像添加到 JMP 图形或报表 446
将背景图像粘贴到 JMP 图形中 446
在 JMP 报表结尾粘贴图像 446
将图像拖放到 JMP 图形中 447
向 JMP 图形添加地理图像和边界 449
从图像中提取数据 449
向 JMP 报表和图形添加注解和形状 450
向 JMP 报表或图形添加注解 450
向 JMP 报表添加形状 451
定制图形中的悬停标签 454
向悬停标签添加图形或图像 455
向图形中的悬停标签添加富文本 469
更新悬停标签中的名称和值 472
编辑图形中的悬停标签 482
将数据表中的图像添加到悬停标签 487
固定图形中的悬停标签 488
悬停标签的右击选项 489

导航 JMP 报表

JMP 报表显示在带有滚动条和调整大小选项的标准窗口中。报表中也有另外一些如图 9.2 所示的特殊按钮和菜单，以及以下各节介绍的按钮和菜单。

图 9.2 报表窗口的基本要素

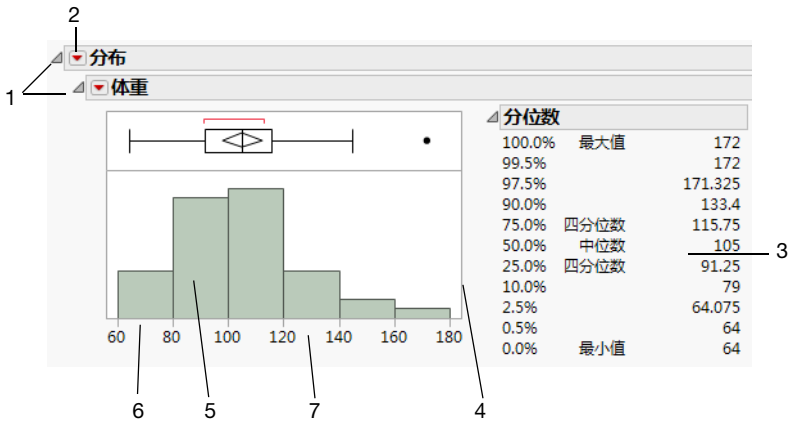


表 9.1 报表窗口操作

数字	操作
1	点击展开图标可隐藏或显示报表的各个部分。
2	点击红色小三角可访问报表选项。
3	在表中右击可访问格式选项。
4	点击并拖动边框可调整图形大小。
5	在图形中的任意位置右击可访问格式选项。
6	在轴内部右击可访问格式选项。
7	光标置于轴上时将变为手形。点击并拖动光标可沿轴滚动或重新调整轴尺度。请参见第 429 页的“在图形中滚动和调整轴尺度”。

提示：考虑在“常规”首选项中设置**自动保存超时**值以在指定的分钟数过后自动保存打开的报表。该自动保存值还应用于数据表、记录、脚本、项目和报表。

使用手形工具

使用“工具 > 抓手”选项选择手形工具。在报表中可配合手形工具（亦称抓手工具）使用多项功能。以下是关于手形工具如何在图形中操作的一些示例：

- 在直方图上，对于连续变量，使用手形工具可改变直条个数或移动直条的边界。
- 在所有报表中，使用手形工具可点击并拖动列以便重新排列。
- 使用手形工具可更改轴值的显示范围。请参见第 429 页的“在图形中滚动和调整轴尺度”。

JMP 报表的显示选项

右击展开图标 ◿ 可显示一个菜单，从中可以重新排列报表，并支持您控制报表分级显示级别。生成的菜单具有以下报表格式选项：

关闭 关闭（隐藏）报表的这一部分。通过点击展开图标也可以完成该操作。

水平 若可用，该选项会使报表分级显示项在垂直和水平布局之间切换。

打开所有子级项 打开选定该命令时所处级别之下的所有分级显示级别，包括当前所处级别。

关闭所有子级项 关闭选定该命令时所处级别之下的所有分级显示级别，包括当前所处级别。

打开所有同类项 打开与分析窗口中当前的报表类型相同的所有报表。若同时分析若干变量，您经常希望同时打开许多同类型的报表。在对单个分析选定多个选项时，您可能也希望同时打开同类型的所有报表。

关闭所有同类项 关闭与分析窗口中当前的报表类型相同的所有报表。

关闭无子级项 关闭不具有子级的所有报表部分。通常在报表分级显示项的顶级使用该命令。这是一种快速查看报表的嵌套结构概览的方式。

分级显示关闭方向 指定当点击展开图标关闭时分级显示框的方向。可用选项为**自动**、**水平**或**垂直**。默认方向为**自动**。

追加项 显示一个子菜单，其中列出向报表追加结构项的方式。这些项包括文本、分级显示项标题栏、针对其他 JMP 文件和窗口的引用、所有打开的 JMP 文件的列表、URL 和脚本。

添加文本项 打开一个窗口，允许您最多添加六个文本项。您可以选择是文本项与项目符号一起出现还是最初显示为隐藏状态。**注意：**您必须恰好在隐藏的文本框上点击才能使文本项可见。

添加分级显示项 打开一个窗口，允许您添加带标题的分级显示框。**注意：**追加的分级显示框包含一个红色小三角菜单，其中列出了追加菜单项。

添加窗口引用 打开一个窗口，允许您选择一个打开的 JMP 窗口，将其作为指向选定的分级显示框的链接添加。

添加文件引用 显示“打开数据文件”窗口，允许您选择一个文件，将其作为指向选定的分级显示框的链接添加。

添加文件目录 显示**选择目录**窗口，该窗口支持您选择一个目录，将其作为指向选定的分级显示框的链接添加。**注意：**指向选定目录中每个文件的链接已追加至分级显示框。

添加所有打开的文件 对于在 JMP 中打开的每个文件，添加一个新的分级显示框。它包含指向文件的链接。

添加 URL 引用 打开“创建至 URL 的链接”窗口，允许您添加**链接名称**和 URL。URL 引用链接已添加至选定分级显示框。

添加脚本按钮 打开“添加脚本按钮”窗口，允许您添加指向选定的分级显示框的命名链接。点击添加的链接将运行指定的脚本。

调整为合适大小 调整分级显示框的大小以适应内部图表或报表的宽度。

显示属性 支持您修改报表中显示框的格式。属性包括刻度线之类的图形坐标轴元素。

提示：通过使用选择工具选择一个或多个显示框并悬停在显示框上，直到出现弹出式属性编辑器，可以直接修改报表中的某些属性。

编辑 显示包含以下选项的子菜单，这些选项将影响使用它们的分级显示级别的所有报表：

选择 突出显示该分级显示级别的所有报表。

撤销选择 撤销选择该分级显示级别的所有选定报表。

记录 在标题为“记录”的单独窗口中复制报表，这样您便可以编辑它或向它追加其他报表。请参见“保存和共享您的数据”一章中第 530 页的“创建记录”。

复制图片 将报表复制到剪贴板。您可以随后打开另一个应用程序并将报表粘贴至该应用程序。

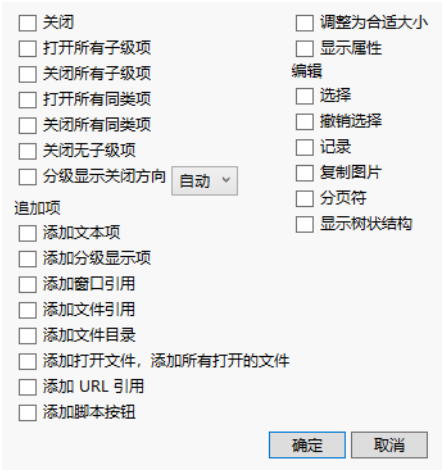
分页符 插入分页符以便打印。

显示树状结构 打开一个窗口，其中显示构成报表的显示框。主要供操作或读取报表部分的 JSL 编程人员使用。请参见《Scripting Guide》。

恢复窗口大小 （仅当窗口大小已从其原始大小修改后才显示。）将选定窗口恢复为其原始大小。

另一种访问这些选项的方法是按 Alt 键并右击展开图标 ◿。这将显示一个窗口，如图 9.3 所示，其中包含命令和选项的复选框，供您同时选择多个操作。您还可以对红色小三角菜单下的菜单执行相同的操作。

图 9.3 分级显示框菜单项窗口



显示和隐藏 JMP 报表部分

JMP 报表按分级显示项来组织。分级显示项的每个级别都有一个三角形的展开图标 ◢。点击展开图标可打开和关闭报表的这一部分。

您还可以更改在报表中显示哪些列。右击该报表，然后从“列”列表中选择列。要一次选择多个列，请在右击前按 Alt。

合并多个报表

假定您要执行多个分析并想要在一个窗口中显示所有结果（和数据表）。您可以采用若干方式选择并合并这些报表和数据表。请参见“使用仪表板组合和演示报表”一章中第 553 页的“示例：组合窗口以创建仪表板”。

重命名报表中的标题

要重命名报表中的标题，请双击以下任意标题：

- 红色小三角菜单旁边的标题
- 展开图标旁边的标题
- 列标题

增大报表中的字体

在 Windows 上，通过选择**窗口 > 字体大小**更改 JMP 在报表和数据表中使用的字体大小。然后选择子菜单项之一：

增大字体 增大字体。再次选择可再次增大字体。

减小字体 减小字体。再次选择可再次减小字体。

在 macOS 上，选择**视图 > 放大文本**或**视图 > 缩小文本**。

调整报表窗口的大小

报表窗口自动根据窗口内容调整大小。在 Windows 上，在手动调整窗口大小后您可以自动调整它的大小。按 Ctrl 键并点击窗口的右下角。

JMP 报表中的红色小三角选项

点击报表中的红色小三角菜单可访问适用于该特定报表的选项列表。除了点击红色小三角菜单，您还可以：

- 同时选择多个操作。按 Alt 键并点击红色小三角菜单。出现一个面板，其中包含带有复选框的所有命令和选项。
- 对报表窗口中所有相似的报表应用某个命令。按 Ctrl 键并点击红色小三角菜单。例如，在单因子分析中，若按 Ctrl 键，点击该图标，然后选择**均值/方差分析/合并的 t 检验**，则将对活动报表窗口中的所有单因子分析执行方差分析。

以下 JMP 文档对适用于 “分析”和 “图形”菜单中各个平台的红色小三角选项进行了介绍：

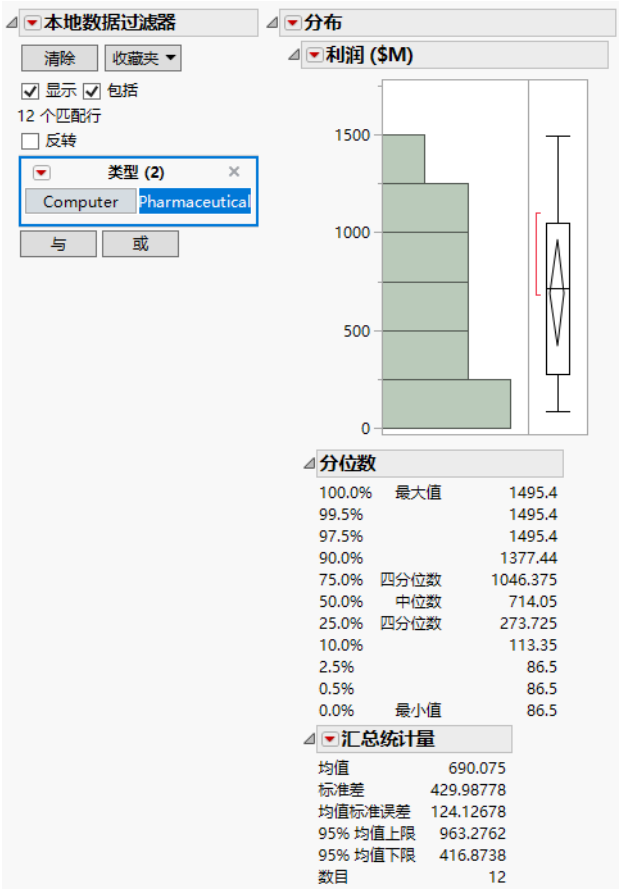
- 《基本分析》
- 《消费者研究》
- 《基本绘图》
- 《拟合线性模型》
- 《多元方法》
- 《刻画器指南》
- 《质量和过程方法》
- 《可靠性和生存方法》
- 《预测和专业建模》

报表中的“本地数据过滤器”

“本地数据过滤器”将过滤器的作用域限定为您所在平台的输出。例如，假定您有关于 30 个不同的计算机和制药公司的利润的数据。您想要按公司类型检查利润。您可以运行某个分布，然后使用“本地数据过滤器”将“分布”平台中的结果范围缩小到公司的类型。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**利润 ($M)** 并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “分布”红色小三角并选择**本地数据过滤器**。
6. 点击**类型**并点击**添加** ✚。
7. 在“类型”框中，点击 Pharmaceutical 仅查看其利润，然后点击 Computer 仅查看其利润。

图 9.4 显示医药公司利润的本地数据过滤器



注意：变量旁边的红色小三角菜单包含反转选择和在复选框列表中显示值等选项。详细信息，请参见第 406 页的“针对数据过滤器中的变量的红色小三角选项”。

本地过滤

使用“本地数据过滤器”时，它仅在平台内过滤而不会像“数据过滤器”那样在数据表内过滤。例如，若您在“本地数据过滤器”中排除了行，这些行不会在数据表中排除。而且，若数据表中有行状态，它们也不受“本地数据过滤器”影响。

保存过滤器

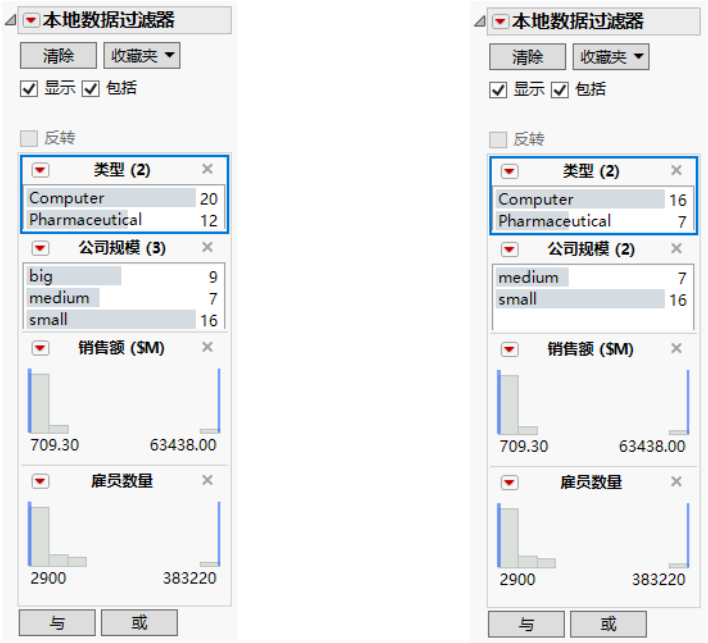
您可以编写“本地数据过滤器”的脚本并进而保存该脚本，这样您就可以快速地再次访问您的过滤器设置。要保存过滤器，点击“本地数据过滤器”旁边的红色小三角并选择“保存脚本”选项之一。

计算排除行数目

每个变量水平的计数（包括排除行）默认都显示在“本地数据过滤器”中。要从计数中忽略排除行，请点击“本地数据过滤器”红色小三角并取消选择**计算排除行数目**。

图 9.5 显示了计数与忽略排除行之间的差异。计算排除行数目时，“本地数据过滤器”显示包含值“Computer”的 20 行。此外，还有 9 个“大”值。忽略排除行时，只有 16 行包含“Computer”。

图 9.5 计算排除行数目 （左）和忽略排除行 （右）



注意：当“显示选项”设置为“区组显示”（第 391 页上的图 9.4）时不显示计数。

有关使用“本地数据过滤器”进行更高级的脚本编写的详细信息，请参见《Scripting Guide》。

使用依据组和子集报表

若分析使用依据组或使用同一表的链接子集将报表合并在一起，则针对父表的本地数据过滤器可以在所有不同的依据组和子集报表之间共享。

在仪表板和应用程序中使用“本地数据过滤器”

您可以将“本地数据过滤器”合并到仪表板或应用程序中：

- 有关使用仪表板的示例，请参见“使用仪表板组合和演示报表”一章中第 557 页的“示例：创建 JMP 查询仪表板和插件”。
- 有关使用应用程序的示例，请参见《Scripting Guide》。

放大时间序列数据

过滤完时间序列数据后，可以放大“本地数据过滤器”直方图以查看更小的时间增量。假定在直方图上选择了一年的数据。要查看该年内时间的更小增量，请从变量旁边的红色小三角菜单中选择**缩放到所选区域**。从“本地数据过滤器”红色小三角菜单中选择**重置缩放**以重置直方图。

报表中的“重新运行”菜单

在执行分析后，您可能要使用不同选项或变量再次运行分析。若数据表中的数据已更改，您可能还要再次执行分析。“重新运行”红色小三角菜单包含以下选项：

列切换器 支持您以交互方式互换图形上的列。请参见第 394 页的“列切换器”。

重新运行分析 基于以前的启动窗口设置重新分析。该分析显示在新报表窗口中。若数据已更改，该选项很有用。

重新启动分析 打开启动窗口并重新调用用于创建报表的设置。

自动重新计算 在数据表值更改时自动更新分析和图形。请参见第 394 页的“自动重新计算”。

报表中的“保存脚本”菜单

完成分析并接收报表后，您可能希望将用于创建报表的过程另存为 JSL 脚本。通过从“脚本”红色小三角菜单选择一个选项来做到这一点。在 JMP 中大多数“保存脚本”选项是相同的。一些平台添加了在特定平台章节中介绍过的额外选项。以下“保存脚本”菜单选项对于所有平台是相同的。

至剪贴板 将脚本保存至您计算机的剪贴板。然后可以将该脚本复制到某个脚本或其他文档中。

至数据表 将脚本保存至用于生成报表的数据表。这允许您从数据表再次运行脚本来重新创建结果。选择该选项可将脚本保存至数据表。

至记录 在记录窗口中保存一个链接来运行分析的脚本。若已打开一个记录窗口，则将脚本添加到该记录窗口。脚本包含到数据表的路径。请注意若找不到数据表，将不运行脚本。

至脚本窗口 打开脚本编辑器窗口并向其中添加脚本。若已打开一个脚本窗口，则将脚本添加到该打开窗口的底部。

至报表 将脚本添加至报表窗口的顶部。

编写所有“依据”组的脚本

若您在启动窗口中指定了“依据”变量，则使用“保存‘依据’组脚本”菜单可以让您将“依据”变量所有水平的脚本保存到数据表或脚本窗口中。

在“保存脚本”菜单中，**至数据表（所有对象）**和**至脚本窗口（所有对象）**保存“依据”变量的所有水平的脚本。

自动重新计算

“自动重新计算”功能可将您对数据表所做的更改立即体现在相应的报表窗口中。您可以对数据表执行以下任意更改：

- 排除或撤销排除数据表行
- 删除或添加数据表行

这一强大功能可将这些更改立即体现在报表窗口相应的分析、统计量和图形中。

要为报表窗口打开“自动重新计算”功能，请点击平台红色小三角并选择**重新运行 > 自动重新计算**。要关闭该功能，请取消选择同一选项。您还可以使用 JSL 打开自动重新计算功能。

支持自动重新计算的平台在“平台”首选项中包含一个首选项。

注意：对于某些平台，“自动重新计算”功能不适用，因此不受支持。这些平台包括：实验设计、刻画器、选择、分割、非线性、神经、神经网络、偏最小二乘、拟合模型（REML、GLM、对数方差）、高斯过程、项目分析、Cox 比例风险、响应筛选和控制图（运行图除外）。

列切换器

在报表中，使用“列切换器”可以快速分析不同的变量，无需重新创建分析。要激活“列切换器”，请从报表窗口中，点击红色小三角并选择**重新运行 > 列切换器**。

若有多个列，请使用按钮动画播放列切换或手动单步切换各列。移动滑块控件可更改动画播放速度。

注意：您无法在报表内复制或移动列切换器。此外，列切换器无法保存至 JMP 记录。

列切换器的示例

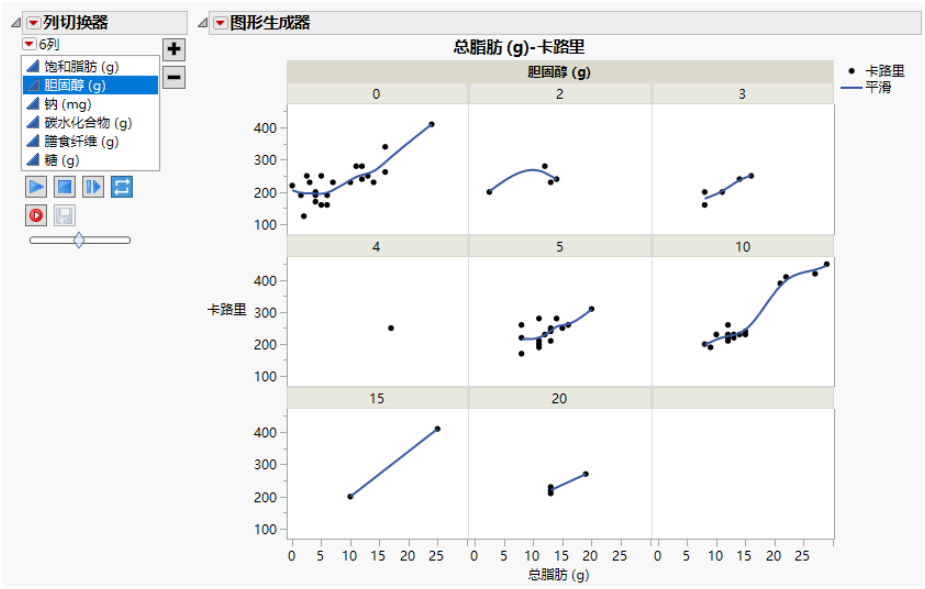
您具有关于糖果条的营养信息数据。您想要检查不同因素，查看哪些因素最能预测卡路里水平。

1. 选择**帮助** > **样本数据库**，然后打开 Candy Bars.jmp。
2. 选择**图形** > **图形生成器**。
3. 点击**总脂肪 (g)** 并拖至 X 区域。
4. 点击**卡路里**并拖至 Y 区域。
5. 点击**胆固醇 (g)** 并拖至**分组**区域。
6. 点击 “图形生成器”红色小三角并选择**显示控制面板**。
7. 点击 “图形生成器”红色小三角并选择**重新运行** > **列切换器**。
8. 选择**胆固醇 (g)**。

   这会选择要切换的初始列。
9. 选择**饱和脂肪 (g)**、**胆固醇 (g)**、**钠 (mg)**、**碳水化合物 (g)**、**膳食纤维 (g)** 和 **糖 (g)**。

   这会选择替换列。
10. 点击**确定**。

图 9.6 “图形生成器”窗口中的列切换器



11. 点击**播放**按钮在不同因素之间循环播放。使用滑块可控制动画播放的速度。或者，您可以单步播放各个因素。

    点击 “循环”按钮可指定动画到达 “列”列表中最后一列时的行为。

您可以看到对于碳水化合物的各个水平，卡路里和脂肪之间的关系相对较强。因此，**碳水化合物 (g)** 看起来最能预测卡路里水平。

注意：

- 要在切换列时保持轴设置不变，请从“列切换器”红色小三角菜单中选择**保留轴设置**。这在定制轴（例如，通过添加参考线）时非常有用。
- 若报表有多个列切换器，则可以使用“列切换器”红色小三角菜单中的“布局”选项调整列切换器的布局。
- 要添加新列，请执行以下选项之一：
  - 在“列切换器”中，点击加号按钮，选择列，然后点击**确定**。
  - 在“图形生成器”中，将列从“列”列表拖入“列切换器”列表。
  - 从数据表中，将列从“列”面板拖入“列切换器”列表。
- 要删除列，请点击减号按钮，选择要删除的列，然后点击**确定**。或者，您可以选择列并按 Delete 键。
- 要恢复更改，请从“列”红色小三角菜单中选择**重置**。
- “列切换器”中的“记录”按钮和“保存”按钮记录动画，然后将其另存为动画 GIF。该功能仅在 Windows 上可用。

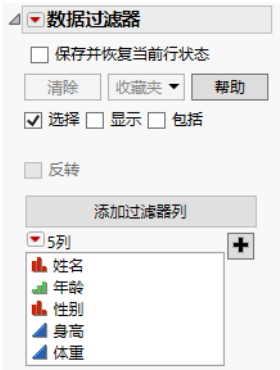
数据过滤器

JMP 中的“数据过滤器”为您提供了多种标识数据子集的方法。您可采用交互方式选择复杂的数据子集、在图形中隐藏这些子集，或将其排除不进行分析。

1. 选择**行 > 数据过滤器**。

提示：除了主数据过滤器，您还可以在平台报表内启动本地数据过滤器。从“报表”工具栏中点击“本地数据过滤器”图标 ，或从报表的红色小三角菜单中选择**本地数据过滤器**。详细信息，请参见第 391 页的“报表中的“本地数据过滤器””。

图 9.7 初始“数据过滤器”窗口



2. 选择要用作过滤器的列，然后点击**添加** ✚。

注意：

- 要在关闭“数据过滤器”窗口后恢复当前行状态，请选择**保存并恢复当前行状态**选项。
- 若您的列列表较长，则可以对列表中的列执行排序、显示、隐藏或搜索操作。使用“添加过滤器列”红色小三角菜单中的选项。
- 默认情况下，“数据过滤器”窗口连接到数据表。您可以暂时或永久断开该窗口的连接：
  - 通过从“数据过滤器”红色小三角菜单中取消选择**使用浮动窗口**选项，可以暂时断开其连接。
  - 通过选择**文件 > 首选项 > 数据过滤器**并取消选择**使用浮动窗口**永久断开其连接。

过滤器列的类型

共有四种类型的过滤器列：

连续列 过滤建模类型设置为连续型的数值列。连续过滤器列通过一个横跨数据范围的直方图来表示。可以调整直方图上表示最小和最大过滤值的垂直线。相关选项，请参见第 407 页的“针对所有类型变量的选项”和第 407 页的“用于连续变量的选项”。

分类列 过滤名义型和有序型列。对于每个分类列，“数据过滤器”都会生成一组不同的类别。这些类别可采用不同形式显示。相关选项，请参见第 407 页的“针对所有类型变量的选项”和第 407 页的“针对名义型或有序型变量的选项”。

注意：对于带有值标签的分类列，若想要包括数据中未显示的响应，请从**文件 > 首选项 > 平台 > 分类**中选择**包括不在数据中的响应**选项。

多重响应列 过滤分配了“多重响应”列属性或“多重响应”建模类型的字符列。列中每个数据单元格通常都包含多个类别，类别之间用逗号之类的常用分隔符分隔。由于每个数据单元格都可能包含多个类别，所以多重响应列具有更多的过滤选项。相关选项，请参见第 407 页

的 “针对所有类型变量的选项”和第 408 页的 “具有 “多重响应”建模类型或列属性的变量的选项”。

非结构化文本 过滤建模类型设置为 “非结构化文本”的文本列。在过滤文本编辑框中键入单词或短语，将其添加到过滤文本列表中。随后可以根据这些单词和短语过滤各行。相关选项，请参见第 407 页的 “针对所有类型变量的选项”和第 408 页的 “针对非结构化文本的选项”。

过滤模式

共有三种过滤模式：**选择**、**显示**和**包括**。您可以使用 “数据过滤器”中相应的复选框设置和清除这些模式。

选择 （不可用于 “本地数据过滤器”。）在数据表中突出显示选定行。

使用**数据过滤器选择检查**首选项，您可以禁用自动选择该选项。另见第 399 页的 “在数据过滤器中做出选择后更改数据表中的行状态”。

显示 对未选定的行显示 “隐藏”图标 。

使用**数据过滤器选择检查**首选项，您可以启用自动选择该选项。有关行状态的详细信息，请参见 “设置 JMP 列属性”一章中第 261 页的 “在行状态列中存储信息”。

包括 对未选定的行显示 “排除”图标 。

使用**数据过滤器包括检查**首选项，您可以启用自动选择该选项。有关行状态的详细信息，请参见 “设置 JMP 列属性”一章中第 261 页的 “在行状态列中存储信息”。

过滤时有两个附加选项：**自动清除**和**有条件**。“数据过滤器”的红色小三角菜单中提供上述选项。请参见第 401 页的 “ “数据过滤器”的红色小三角选项”。

“数据过滤器”控制面板

一旦在初始窗口中添加了列，“数据过滤器”控制面板随即显示。

图 9.8 “数据过滤器”控制面板

“数据过滤器”中的主要控件如下所示：

清除 清除在“数据过滤器”窗口中对变量所做的所有选择。

收藏夹 将您当前的数据过滤器条件另存为收藏项。一旦创建了收藏项，选择该项会将当前条件重置为该收藏项中的条件。您也可以删除收藏项。要在当前会话一结束就保留收藏夹，请通过从“数据过滤器”红色小三角菜单中选择某一“保存脚本”选项来保存数据过滤器脚本。

选择、显示和包括 请参见第 398 页的 “过滤模式”。

反转 反转数据表中行的当前选择状态。

注意：反转的只是数据表中的行，而不是数据过滤器中的选择项。要反转数据过滤器中的选择项，请从列的红色小三角菜单中选择**反转选择**。

与 使用“与”按钮可打开**添加到选定组**列表。**与**运算符用于限定选择。您可以随时在过滤过程中添加变量。

或 使用“或”按钮可打开**添加新的“或”组列表**。**或**运算符用于扩展选择。您可以随时在过滤过程中添加变量。

在数据过滤器中做出选择后更改数据表中的行状态

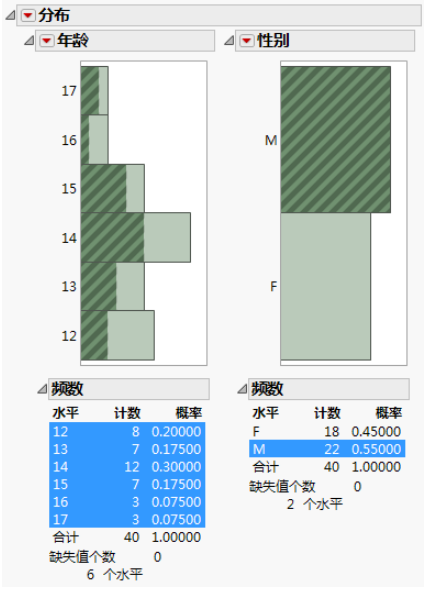
若您已在“数据过滤器”中做出选择，随后在数据表中更改行状态或选择图形中的点，则“数据过滤器”中的选择可能与数据表中的选择不符。“数据过滤器”包含一条警告消息：“您的

选择已在另一窗口中更改”。出现**重置选择**按钮。点击**重置选择**按钮可改回数据表选择，以反映数据过滤器中的选择。

修改选择的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **分布**。
3. 选择**年龄**和**性别**，并点击 **Y，列**。
4. 点击**确定**。
5. 选择**行** > **数据过滤器**。
6. 选择**性别**并点击**添加**。
7. 在 “数据过滤器”控制面板中，通过点击 M 分区选择所有男性。

图 9.9 包含男性的行在数据表和直方图中突出显示



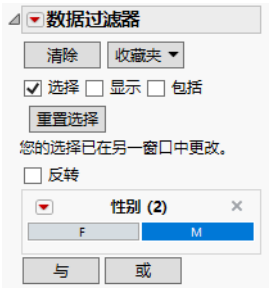
	姓名	年龄	性别	身高	体重
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	79
9	BARBARA	13	F	60	112
10	ALICE	13	F	61	107
11	SUSAN	13	F	56	67
12	JOHN	13	M	65	98
13	JOE	13	M	63	105
14	MICHAEL	13	M	58	95
15	DAVID	13	M	59	79
16	JUDY	14	F	61	81
17	ELIZABETH	14	F	62	91
18	LESLIE	14	F	65	142
19	CAROL	14	F	63	84
20	PATTY	14	F	62	85
21	FREDERICK	14	M	63	93
22	ALFRED	14	M	64	99

您可以看到包含男性的所有行在数据表和直方图中均突出显示。现在，您决定只想查看年龄为 12 岁的学生。

8. 在年龄直方图中，选择表示 12 岁的直条。

   现在的选择与数据过滤器中的选择不符。“数据过滤器”窗口中会显示一条警告消息及**重置选择**按钮。

图 9.10 数据过滤器警告消息和重置按钮



“数据过滤器”的红色小三角选项

“数据过滤器”旁边的红色小三角菜单包含以下选项：

自动清除 若您在 “数据过滤器”中选择了多个名义型或有序型列，则该选项将在进行新的选择之前清除所有其他选择。以 Big Class.jmp 为例，假定您的数据过滤器中有**性别**（名义型）和**年龄**（有序型）这两列。若为性别选定了男性 (M)，然后点击某个年龄组，比如 12 岁年龄组，您之前做出的男性选择将自动清除。这意味着选择 12 岁并不以选择男性为前提条件。反之，若禁用 “自动清除”，您则可以同时选择男性和 12 岁。默认情况下关闭 “自动清除”。

条件 限制为选定的过滤器列显示的类别。请参见第 402 页的 “条件数据过滤器”。

按 “与”分组 允许您控制多组列过滤器的 “与”和 “或”行为。请参见第 403 页的 “数据过滤器中的按 “与”分组”。

显示直方图和直条 在数据过滤器中显示或隐藏直方图和直条。

显示计数 在分类列数据过滤器中显示或隐藏计数。

使用浮动窗口 令 “数据过滤器”窗口停留在其关联的数据表的最上方。若不希望 “数据过滤器”窗口停留在最上方，请取消选择该选项。

重新开始 关闭当前数据过滤器会话并显示最初的 “数据过滤器”窗口。

显示控制项 显示或隐藏“数据过滤器”窗口中除“清除”和“反转”之外的所有按钮和复选框。

显示模式 在 “数据过滤器”窗口中显示或隐藏过滤模式复选框。

动画 按顺序突出显示数据表中单个变量的值。请参见第 405 页的 “数据过滤器中的动画”。

子集 创建仅包含以下对象的新数据表：

- 数据过滤器所标识的行。
- 在活动数据表中选定的列。若未选定任何列，则包括所有列。

该选项类似于 “表 > 子集”命令，只不过没有生成子集选项。

保存 Where 子句 基于所做的值选择生成 Where 子句。

提示：选择**至公式列**创建一个指示符列。在许多平台中，您可以将指示符列用作“图形生成器”中的“依据”角色、组或 ID 等。

保存脚本 提供用于保存脚本的选项。请参见第 393 页的“报表中的“保存脚本”菜单”。

复制本地数据过滤器 （仅可用于“本地数据过滤器”。）将本地数据过滤器的脚本复制到剪贴板。您可以随后使用报表的红色小三角菜单中的“粘贴本地数据过滤器”选项，将该本地数据过滤器应用于新报表。

删除本地数据过滤器 （仅可用于“本地数据过滤器”。）从报表中删除“本地数据过滤器”。

条件数据过滤器

对于包含层次的过滤器列，您可以使用条件选项来过滤列列表中的显示内容。例如，您可以按地区过滤，以便在列表中仅显示选定地区的州。

注意：若有多个“或”组，则条件过滤器不可用。

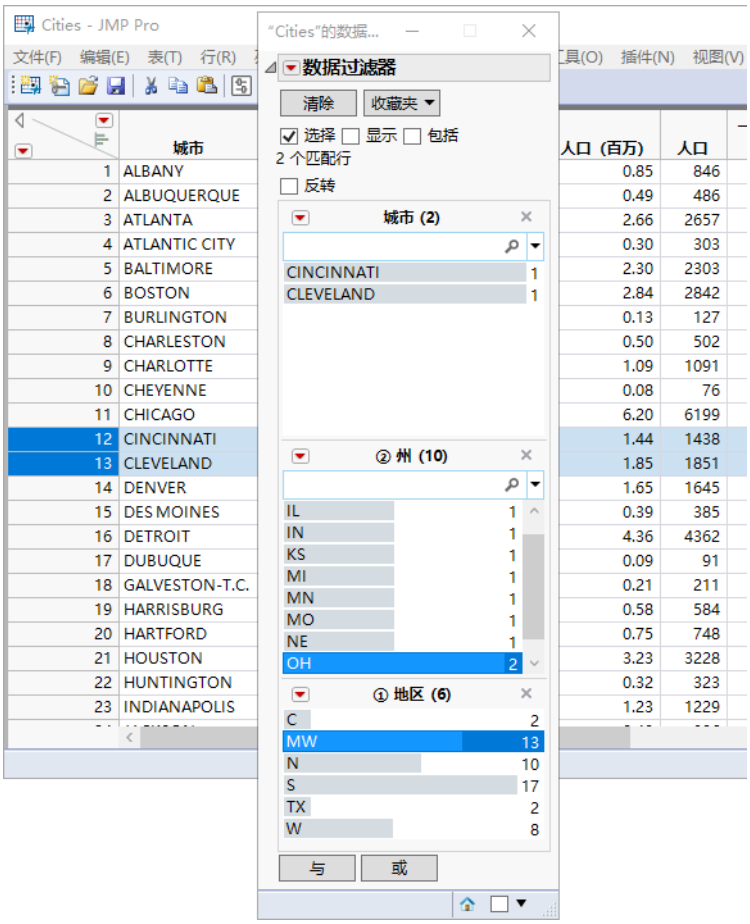
以下示例说明“条件”选项如何帮助清晰地显示子类别，筛除了不满足条件的多余类别。

1. 选择**帮助** > **样本数据库**，然后打开 Cities.jmp。
2. 选择**行** > **数据过滤器**。
3. 在“数据过滤器”窗口中，选择**城市**、**州**和**地区**，然后点击**添加**。

   “数据过滤器”窗口出现，为每个变量显示一个列表。
4. 点击“数据过滤器”红色小三角并选择**有条件**。
5. 在“地区”列表中选择 MW。
6. 从“州”列表中选择 OH。

   位于中西部地区俄亥俄州的城市将在数据表中处于选定状态。在“数据过滤器”窗口中，只有中西部地区的州才显示在“州”列表中，而且只有俄亥俄州的城市才显示在“城市”列表中。

图 9.11 使用 “条件” 选项



列名前方画圈的数字指示列值的选择顺序。在图 9.11 中，先选择的是**地区**，所以其列名前有一个画圈的“1”。其次选择了“州”，所以其列名前有一个画圈的“2”。

当您对以升序排列的层次过滤器进行重新排列时，过滤器编号将更改，以与层次中的升序位置匹配。

要清除所做选择并重置层次顺序，请点击**清除**。

数据过滤器中的按“与”分组

添加过滤器列时，点击 AND 并添加另一个过滤器列，即可创建一个过滤器组。在该组内，在数据表中选择满足所有条件的列。

您点击“或”、添加一个或多个过滤器时，创建了第二组过滤器。当取消选择“按‘与’分组”时，这是默认行为。

若选择“按‘与’分组”，则反转该行为。

下例显示选择“按‘与’分组”时发生的行为。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**行** > **数据过滤器**。
3. 点击“数据过滤器”红色小三角并选择**按“与”分组**。
4. 在“数据过滤器”窗口中，选择**年龄**列，然后点击**添加**。
5. 点击**或**并添加**性别**列。

   性别将添加到**年龄**所在的组。
6. 从**年龄**过滤器中选择 14，然后从**性别**过滤器中选择 M。

   年龄和**性别**同在一组中。在数据表中，满足**年龄**或**性别**条件的行将被选定。也就是说，数据表中年龄为 14 岁的所有人和所有男性都被选定。
7. 点击**与**并添加**体重**列。

   将添加一个新组。
8. 将**左侧体重**滑块移到大约 112。

   已经添加另一个过滤器组。第二个过滤器组将选择内容限定为体重大于 112 的那些行。

   这两个过滤器组一起选择满足第一组中的条件“与”第二组中的条件的那些行，即 14 岁或任何男性但体重必须大于 112 的学生。

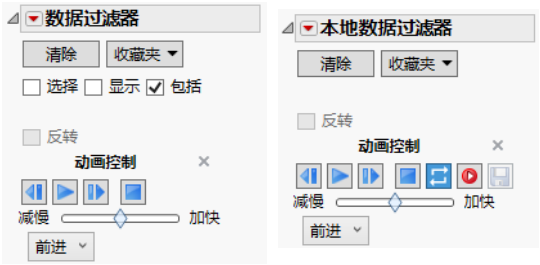
图 9.12 具有按“与”分组条件的数据过滤器

数据过滤器中的动画

动画功能可按顺序突出显示单个变量的值。变量的值将在数据表中突出显示。但是，若首先创建了图，而后使用数据过滤器对变量进行动画播放以查看该变量在图中的行为，此时模式更为重要。

要使用动画功能，请点击“数据过滤器”红色小三角并选择**动画**。然后选择要动画播放的变量。变量四周突出显示的框指示已选定该变量进行动画播放。

图 9.13 “数据过滤器”和“本地数据过滤器”中的“动画控制”面板



“动画控制”面板包含以下控件：

- 中间按钮 ( ) 用于启动和暂停动画播放。启动动画循环后，该按钮将变为暂停按钮 ( )。默认情况下，动画从最顶层变量的第一个值开始播放。
- 后退箭头 ( ) 用于将动画后移一个循环。多次点击则后移多个循环。
- 前进箭头 ( ) 用于将动画前移一个循环。多次点击则前移多个循环。
- 方块按钮 ( ) 用于在“数据过滤器”窗口中隐藏“动画控制”部分。再次从“数据过滤器”标题栏上的菜单中选择**动画**可查看“动画控制”。
- 循环按钮 ( ) 支持循环，默认情况下处于启用状态。若禁用循环按钮，动画将在值序列中循环一次，然后停止。
- 录制按钮 ( ) 开始录制动画。开始录制后，该按钮变为停止录制按钮 ( )。
- 保存按钮 ( ) 用于保存录制的动画。

使用滑块可调节动画播放的速度（减慢到加快）。

“动画”下拉菜单包含以下选项：

前进 从前往后突出显示值。

后退 从后往前突出显示值。

循环 向前突出显示，而后往后突出显示，如此反复播放。

在数据过滤器中保存 WHERE 子句

一旦在数据过滤器中过滤了变量值，该信息可表示为 JMP WHERE 子句。WHERE 子句用在 JSL（JMP 脚本语言）程序中，用来标识要处理或分析的特定数据行。数据过滤器基于所选的值生成 WHERE 子句。

保存 Where 子句菜单中包括如下选项：

至剪贴板 基于过滤条件创建 WHERE 子句，并将其放入剪贴板。

至行状态列 在包含相当于过滤条件的公式的数据表中创建行状态列。

至公式列 在包含相当于过滤条件的公式的数据表中创建公式列。

至数据表 基于过滤条件创建 WHERE 子句，并将其作为 JSL 命令，保存在当前数据表中名为**过滤器**的表属性中。

至脚本窗口 基于过滤条件创建 WHERE 子句，并将其追加至当前脚本文本窗口，或在不存在脚本时创建新脚本。

至记录 基于过滤条件创建 WHERE 子句，并将其追加至当前记录，或在不存在记录时创建新记录。

保存 WHERE 子句的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**行** > **数据过滤器**。
3. 选择**年龄**、**性别**和**身高**并点击**添加**。

选择年龄为 12 岁和 14 岁并且身高在 56 英寸到 60 英寸之间的所有女性：

4. 点击 12 区块、 14 区块以及 F 区块。
5. 点击 51 并键入 56。
6. 点击 70 并键入 60。
7. 点击 “数据过滤器”红色小三角并选择**保存 Where 子句** > **至脚本窗口**。

根据该示例创建的 Where 子句将显示在脚本窗口中：

```
Select Where(
(:age == 12 | :age == 14) & :sex == "F" & (:height >= 56 & :height <= 60)
);
```

针对数据过滤器中的变量的红色小三角选项

将列添加到数据过滤器后，将出现一个红色小三角菜单。针对变量的某些红色小三角选项可能因变量类型而异。

针对所有类型变量的选项

任何类型变量旁边的红色小三角菜单都包含以下通用选项：

清除选择 清除仅针对该变量有效的任何选择。

反转选择 仅针对该变量取消选择任何选定值，并选择以前未选定的所有值。

建模类型 显示列过滤器的建模类型子菜单。默认情况下，列过滤器类型设置为列的建模类型。通过在子菜单中选择不同的建模类型，可以更改过滤器类型。

用于连续变量的选项

对于连续变量，值显示在直方图中，垂直线位于最小值和最大值处。通过点击垂直线并拖动，可以调整选定的范围。更新的范围以蓝色突出显示。

以下选项仅可用于“连续变量”：

选择缺失值 选择包含缺失值的行。

包括最小值 包括或排除选定范围内的最小值。排除最小值后，直方图中的垂直实线将变为垂直虚线。

包括最大值 包括或排除选定范围内的最大值。排除最大值后，直方图中的垂直实线将变为垂直虚线。

设置箱个数 设置直方图的箱个数。

缩放到所选区域 仅在直方图中显示选定的范围。选择菜单项后，该菜单项将改为“重置缩放”。“重置缩放”可重置显示。

针对名义型或有序型变量的选项

对于名义型和有序型变量，值显示在分区中、列表中或菜单中。若变量包含少量类别，则值显示在分区中。若变量包含大量类别，则值显示在列表或菜单中。不过，您可以更改这些默认设置。

以下选项仅可用于名义型或有序型变量：

显示选项 更改显示的外观。可包括以下选项：

- **区块显示** 将每个水平显示为一个区组。
- **列表显示** 将各个水平显示为列表成员，后跟其频数。
- **单一类别显示** 在菜单中显示各个水平，后跟其频数。
- **复选框显示** 在每个值旁边添加一个复选框。要将其设置为默认选项，请选择**文件 > 首选项 > 数据过滤器**中的**复选框显示**选项。
- **单选框显示** 在每个值旁边添加一个单选按钮。

按计数排序 按计数的降序对值排序。

查找 提供一个文本框，您可在其中输入选定列的搜索字符串。按 Enter 键执行搜索。

具有“多重响应”建模类型或列属性的变量的选项

为具有“多重响应”建模类型或列属性的变量提供了以下选项：

显示选项 更改显示的外观。可包括以下选项：

列表显示 将各个水平显示为列表成员，后跟其频数。

单一类别显示 在菜单中显示各个水平，后跟其频数。

复选框显示 在每个值旁边添加一个复选框。要将其设置为默认选项，请选择文件 > 首选项 > 数据过滤器中的**复选框显示**选项。

单选框显示 在每个值旁边添加一个单选按钮。

不匹配 仅选择包含与任何选中项都不匹配的值的行。

任意匹配 选择包含与任何选中值都匹配的值的所有行。默认选定该选项。

全部匹配 仅选择包含所有选中值的值所在的那些行。

精确匹配 仅选择具有与选中值精确匹配的值的那些行。

仅匹配 仅选择具有与选中值精确匹配的内容的那些行。

匹配至少 选择至少 *n* 个选定值。

匹配至多 选择至多 *n* 个选定值。

两者之间匹配 在 *n* 个和 *m* 个选定值之间进行选择。

查找 提供一个文本框，您可在其中输入选定列的搜索字符串。按 Enter 键执行搜索。

注意：有关“多重响应”属性的详细信息，请参见“设置 JMP 列属性”一章中第 276 页的“多重响应”。

针对非结构化文本的选项

以下选项可用于“非结构化文本”建模类型的变量：

区分大小写 若选中该框，则基于在 “过滤器文本”编辑框中键入的单词或短语的过滤器将区分大小写。

正则表达式 认定在过滤器文本编辑框中键入的字符串是正则表达式，而不是文字字符串。

显示选项 更改显示的外观。可包括以下选项：

列表显示 将各个单词或短语显示为列表成员，后跟其频数。

单一类别显示 在菜单中显示每个单词或短语，后跟其频数。

复选框显示 在每个单词或短语旁边添加一个复选框。要将其设置为默认选项，请选择文件 > 首选项 > 数据过滤器中的**复选框显示**选项。

单选框显示 在每个单词或短语旁边添加一个单选按钮。

添加缺失值 为不包含任何文本的条目添加过滤器。

清除过滤器文本列表 删除已添加到过滤器文本列表中的任何单词或短语。

显示过滤器文本编辑框 显示或隐藏过滤器文本编辑框。

按计数排序 按计数的降序排列单词或短语。

不匹配 仅选择包含与所选单词或短语不匹配的文本的行。

任意匹配 选择包含与任何选定单词或短语匹配的文本的所有行。默认选定该选项。

全部匹配 只选择文本中包含所有选定单词或短语的行。

重新格式化报表中的表

更改报表表格式的方式有很多。右击报表表可访问以下格式选项：

表样式 向表添加边框、阴影和分隔线。从以下选项中进行选择：

标题加下划线 显示“表标题加下划线”的首选项设置。

标题着色 显示“表标题着色”的首选项设置。

标题列边框 显示标题中列之间的分隔线。

列边框 在列外部添加边框，在列间添加分隔线。

行边框 在行外部添加边框，在列间添加分隔线。

隔行着色 隔行着色。

单元格着色 对表主体着色。当与隔行着色一起使用时，较深的阴影将隔行使用。

注意：在 JMP 首选项中选择之前的选项可更改所有报表表的格式。在 Windows 上，“报表表”选项位于“文件 > 首选项 > 样式”中。在 macOS 上，在 **JMP > 首选项 > 样式**中选择“报表表”选项。

列 显示或隐藏表中的列。

注意：名称以波浪号 (~) 开头的列 （如 ~ **偏倚**）不适用于您运行的分析，在表中也不显示，即便其名称旁具有复选标记也是如此。

按列排序 按选定列的降序或升序对列排序。也可以选择 “数值顺序”对已排序的字符串列进行数值排序。执行该操作后，当您右击该列时即可使用 “数值顺序”选项。有关“数值顺序”选项使用的排序规则的详细信息，请参见 “设置 JMP 列属性”一章中第 268 页的 “数值顺序”。

制成数据表 从报表创建 JMP 数据表。数据表脚本重新创建报表和 “制成数据表”生成的数据表。

制作合并数据表 在报表中搜索与选定表类似的其他表，并将它们合并为一个数据表。数据表包含标识原始数据表的列。数据表脚本重新创建报表和“制作合并数据表”生成的数据表。

制成矩阵 从报表创建 JMP 矩阵。请参见第 411 页的 “将 JMP 报表中的表转换为矩阵”。

选择符合条件的行 （仅可用于分类列）使用选定列或复杂表达式创建 Where 子句。

显示属性 支持您修改报表中显示框的格式。

格式化列 更改表中一个或多个列的数值格式。

复制列 将右击列的内容复制到剪贴板以粘贴至另一个窗口或应用程序中。

复制表 将右击表复制到剪贴板以粘贴至另一个窗口或应用程序。

模拟 为报表中的选定统计量列提供模拟结果。

JMP PRO **Bootstrap** 对统计量的抽样分布作近似。请参见 《基本分析》。

修改 JMP 报表中的显示框属性

您可以更改报表中每个显示框的属性，例如，字体、文本颜色、填充和对齐方式。当您将报表另存为脚本、图形或记录时，您的更改也会被保存。

要修改显示框属性，请执行以下步骤：

1. 右击报表旁边的展开图标，然后选择**显示属性**。
   随即显示 “属性”窗格。
2. 修改报表属性。
3. 点击报表以查看您的更改。

提示：

- 在“属性”窗格中，使用箭头按钮可在显示树中的显示框之间进行导航。
- 打开“属性”窗格后，选择特定的报表并更改其属性。

修改数值列的格式

在“显示属性”窗格中，您可以修改报表的外观，包括修改选定数值列的格式。例如，您可能希望为所有数值列指定两个小数位。当您将报表另存为脚本、图形或记录时，您的更改也会被保存。

要修改数值格式，请执行以下步骤：

1. 右击报表旁边的展开图标，然后选择**显示属性**。
   随即显示 “属性”窗格。
2. 按 Shift 键并选择想要更改格式的列。
3. 在 “属性”窗格的 “格式”列表中，选择 “格式”属性。

提示：

- 要选择特定类型的所有显示框（例如，所有 NumberColBoxes），请在报表中选择这些框之一。从“属性”窗格的“选择”列表中，选择**类型为“NumberColBox”的框**。
- 要选择具有特定格式的所有列，请在报表中选择这些框之一。从“属性”窗格的“选择”列表中，选择**具有匹配格式的列**。

将 JMP 报表中的表转换为矩阵

您可以从报表表创建 JMP 矩阵。例如，使用 JMP 脚本语言 (JSL) 时，您可能希望访问存储到 JSL 变量中的报表表。或者，您可能希望将报表表的值存储到表属性中，作为一个表属性或 JSL 赋值，JSL 赋值存储在数据表中，可通过脚本或“公式编辑器”访问。

将矩阵形式的表存储到全局变量、表属性或作为赋值的表属性中：

1. 右击报表中的任意位置。
2. 选择**制成矩阵**。
3. 在出现的窗口中，指示 JMP 您希望如何存储该表。
4. （可选）通过在名称旁边的框中输入新名称以重命名变量或属性。

在报表中使用条件格式

注意：您必须在“报表”首选项中启用**显示条件格式**才可使条件格式显示在 JMP 报表中。

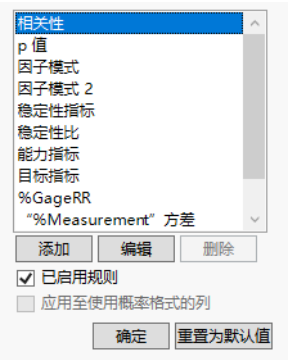
要将报表配置为使用条件格式，您必须首先设置报表首选项才可启用显示条件格式。请参见“JMP 首选项”一章中第 587 页的“JMP 报表的首选项”。

要配置条件格式，请执行以下步骤：

1. 打开 JMP“首选项”窗口。
2. 选择 “报表”首选项组。
3. 点击**管理规则**。

   “条件格式规则”窗口随即显示。

图 9.14 “条件格式规则”窗口



所有规则都默认启用。“p 值”规则应用于包含概率值的列。

从该窗口中，您可以添加、编辑或删除规则。

禁用或启用规则

禁用或启用规则：

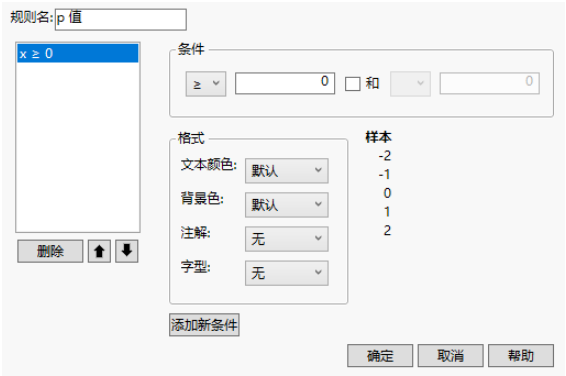
1. 打开 JMP“首选项”窗口。
2. 选择 “报表”首选项组。
3. 点击**管理规则**。
   “条件格式规则”窗口随即显示。
4. 从列表中选择规则。
5. 若**已启用规则**选中，则选择以禁用。
6. 若**已启用规则**未选中，则选择以启用。

添加规则

添加规则：

1. 打开 JMP“首选项”窗口。
2. 选择 “报表”首选项组。
3. 点击**管理规则**。
   “条件格式规则”窗口随即显示。
4. 点击**添加**。
   “条件格式规则”窗口随即显示。占位符规则显示在左侧。

图 9.15 添加条件格式规则



5. 按照第 414 页的 “添加条件” 所述添加和设置条件格式。
6. 点击**确定**保存新规则并返回至上一个 “条件格式规则” 窗口。
7. 选择**已启用规则**启用规则。

编辑规则

更改规则的条件格式：

1. 在 “条件格式规则” 窗口中，从列表中选择规则。
2. 点击**编辑**。

   “条件格式规则” 窗口随即出现，显示选定规则的当前条件。

图 9.16 “条件格式规则” 窗口

“p 值”的默认规则包括两个条件。列表的顺序指示应用规则的顺序。例如，*p* 值为 0.04 将不调用第一个规则 (x < 0.01)，但会调用第二个规则 (x < 0.05)。

注意：您无法编辑规则内的现有条件。要编辑现有条件，您必须删除该条件然后再添加。请参见第 416 页的“编辑现有条件”。

添加条件

将条件添加至规则：

1. 打开规则以查看“条件格式规则”窗口。
2. 在“条件”区域中，从列表中选择一个关系，然后在文本框中输入值。

   或

   若您需要条件包括值范围：

   – 在关系下拉列表中选择 > 或 ≥，并在文本框中输入值。
   – 选中复选框，选择 < 或 ≤ 关系，然后输入值。
3. 使用第 415 页的“设置条件格式”中所述的过程设置条件格式。

提示：当您点击**添加新条件**时，条件及其格式设置会立即添加至条件列表底部，“条件”和“格式”区域恢复为其默认设置。请在点击**添加新条件**前先验证您的设置。

4. 验证您的设置后，点击**添加新条件**将条件添加至规则。
5. 使用 ⬆ 和 ⬆ 按钮确定条件在列表中的位置。

图 9.17 添加具有格式规则的条件的示例

6. 点击**确定**保存您的更改并返回至上一个“条件格式规则”窗口。

删除条件

从规则中删除条件：

1. 在“条件格式规则”窗口中，选择条件并点击**删除**。
2. 点击**确定**保存您的更改并返回至上一个 “条件格式规则”窗口。
3. 点击**确定**返回到 “首选项”窗口。

设置条件格式

设置符合条件的值的格式：

1. 在输入条件后，从下拉列表中选择文本颜色：

   默认 值不带颜色显示。

   纯色 值以指定的颜色显示。

   变暗 值按指定的透明度变暗显示。

   颜色范围 对于有范围的条件，值在指定的颜色范围内以颜色梯度形式显示。

   变暗范围 对于有范围的条件，值按指定的透明度范围以变暗的梯度形式显示。
2. 要指定文本颜色，点击色卡并选择一个颜色。
3. 要指定透明度，输入百分比值 （例如，“60%”）。

   “样本”区域显示文本列的外观预览。
4. 从下拉列表中选择背景色：

   默认 值不带颜色背景显示。

   纯色 值以指定的颜色背景显示。

   颜色范围 对于有范围的条件，值在指定的颜色范围内以背景颜色梯度形式显示。
5. 要指定背景色，点击颜色样本并选择一个颜色。

   “样本”区域显示文本列的外观预览。
6. 要让条件值受到注意，请从下拉列表中选择注解：

   无 值不使用注解。

   方块 值显示在表中时周围加框。

   圆圈 值显示在表中时周围加椭圆。

   ***（星号）** 星号 (*) 显示在表中值的右侧。

   “样本”区域显示文本列的外观预览。
7. 要设置值文本样式，从下拉列表中选择**字型**：

   无 不对值格式使用样式。

   无格式 不对值格式使用样式。

   粗体 值在表中显示为粗体。

斜体 值在表中显示为斜体。

粗斜体 值在表中显示为粗斜体。

有关条件格式的示例，请参见图 9.17。

注意：配置格式后，请记住点击**添加新条件**将新条件添加至规则。

编辑现有条件

更改现有条件的格式：

1. 在“条件格式规则”窗口中，选择条件并点击**删除**。
2. 使用第 414 页的 “添加条件”中所述的过程重新创建条件。
3. 使用第 415 页的 “设置条件格式”中所述的过程设置条件格式。

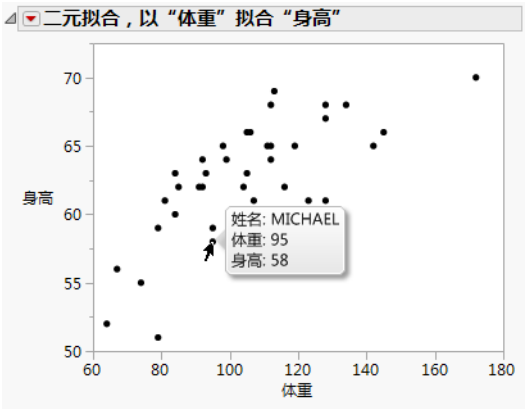
提示：当您点击**添加新条件**时，条件及其格式设置会立即添加至条件列表底部，“条件”和“格式”区域恢复为其默认设置。请在点击**添加新条件**前先验证您的设置。

4. 验证您的设置后，点击**添加新条件**将条件添加至规则。
5. 使用向上 和向下 按钮确定条件在列表中的位置。
6. 点击**确定**保存您的更改并返回至上一个 “条件格式规则”窗口。

选择图形中的点

要选择图形中的点，请点击该点。这将选择该点以及当前数据表中的相应行。要选择多个点，请按 Shift 键的同时选择各点。将光标置于点上方时（点击或不点击）将出现点的标签。

图 9.18 数据点的标签



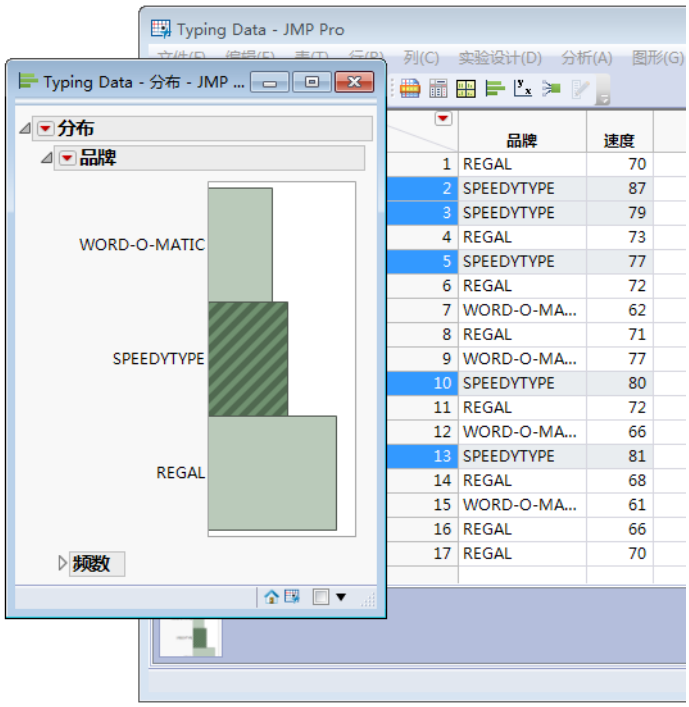
提示：

- 您可以将标签内容粘贴到另一文档。右击该标签，然后选择**复制**。将该文本粘贴到其他文档。
- 若您在“表达式”角色列中存储该图像，可以在标签中显示一个照片。请参见“设置 JMP 列属性”一章中第 282 页的“表达式角色”。

在图形中选择行

代表同一数据表的所有图形都彼此链接并链接至相应的数据表。点击图中的点或图形中的直条时，相应行在数据表中突出显示。图 9.19 中的示例显示了突出显示 SPEEDYTYPE 直条的直方图，并且相应行在表中突出显示。您还可以通过在按 Shift 键的同时做出选择来扩展直方图中直条的选择。

图 9.19 在直方图中突出显示行



在 JMP 图形中选择矩形点区域

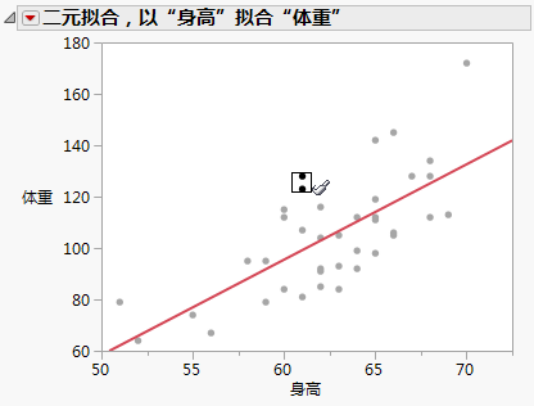
您可以使用鼠标选择矩形区域内的所有点。点击并拖动箭头可突出显示点。或者，您可以使用笔刷工具。将笔刷移至图形上方时，将选中矩形内的所有点。在数据表中标记为隐藏的任何点不会被选中。请参见“输入和编辑数据”一章中第 190 页的“隐藏数据表中的行”。

要使用笔刷工具选择点，请执行以下步骤：

1. 点击工具栏中的笔刷工具 。
2. 在图中点击并按住笔刷工具。随即出现一个矩形。
3. 将该矩形移至点的上方。在经过这些点时，这些点将突出显示并在图中和活动数据表中同时突出显示。

   要在将笔刷工具移过各点时保持选中所有点，请在点击图中某个点之前按 Shift 键。选中的点也会在数据表中选定。

图 9.20 使用笔刷工具



4. 松开鼠标。矩形和数据表内的点仍然保持选中。
   - 要更改选择矩形的大小，请在点击图中某个点之前按 Alt 键。拖动光标以调整选择框的大小。该形状就像一个切片工具，可以贯穿并突出显示任一个轴上的点切片。**注意：**会记住选择框的大小以便您下次使用笔刷工具时使用。
   - 若您按 Ctrl 键并在选定点上点击笔刷工具，则选择矩形内的点会取消选择。选择矩形外的点会保持选中。

在 JMP 图形中选择点的不规则形状区域

您可以使用套索工具选择不规则形状区域内的点。在数据表中标记为隐藏的任何点不会被选中。请参见“输入和编辑数据”一章中第 190 页的“隐藏数据表中的行”。

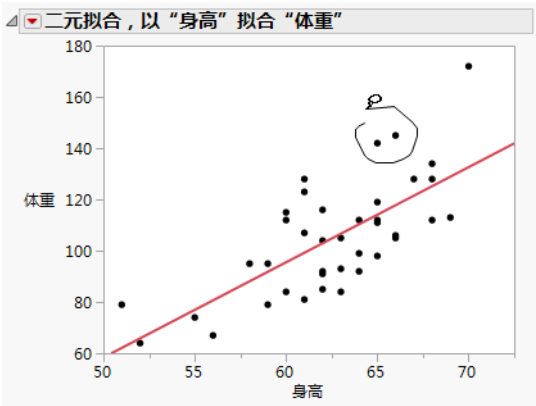
要选择不规则形状区域内的点，请执行以下步骤：

1. 点击工具栏中的套索工具 。
2. 在图中点击并按住套索工具。

   注意：要在将套索拖过几组点的同时保持选中所有点，请在图中点击之前按 Shift 键。

3. 将套索拖过任意多组数据点。

图 9.21 使用套索工具



4. 松开鼠标。 JMP 自动关闭套索并突出显示封闭区域内的点。

在图形中使用标记

标记是图形上表示数据的点。一旦更改了标记的默认设置，这些标记也会显示在数据表中行的旁边。

- “更改标记的形状”
- “更改标记的颜色”
- “更改标记的大小”
- “处理拥挤的标记”
- “为标记添加轮廓”
- “更改标记的选择模式”
- “指定标记的透明度”
- “排除和隐藏标记”
- “将图像用作标记”
- “基于值更改标记的形状或颜色”

更改标记的形状

您可以从 JMP 标记选项板中指定某个字符来替代散点图中的标准点。这些标记还会显示在数据表中的行号旁边。

1. 突出显示要更改形状的一个或多个标记。
2. 右击图形中的任意位置。在直方图中，右击右侧的箱线图区域。
3. 选择**行标记**。

4. 从出现的选项中选择一个标记形状，或点击**其他**输入要用作标记的字符。

更改标记的颜色

您可以为突出显示的行指定任意颜色。指定颜色时，散点图中的点将以您从调色板中选择的颜色显示。指定给行的有效颜色显示在数据网格中的行号旁边。

更改图形上的标记 （或点）的颜色：

1. 突出显示要更改颜色的一个或多个标记。
2. 右击图形中的任意位置。在直方图中，右击右侧的箱线图区域。
3. 选择**行颜色**。
4. 选择一种颜色，或点击**定制**应用定制颜色。

更改标记的大小

增大或减小图形上标记的大小：

1. 右击图形中的任意位置。按 Ctrl 键的同时右击，传播该命令并将其应用到位于同一窗口中的所有同类型图。在直方图中，右击右侧的箱线图区域。
2. 选择**标记大小**。
3. 选择所列的标记大小之一。

   JMP 首选项的 “图形” 页面中已选择默认值 “首选大小”。

注意：绘制速度若设置为 “快速”，则应用最小的标记大小 （点）。请参见第 420 页的 “处理拥挤的标记”。

处理拥挤的标记

图形上的大量标记可能在图形上显示过于拥挤。您可能需要改变透明度以使显示更为清晰。更改透明度可能还会影响标记绘制模式，这是 JMP 在刷新报表窗口时使用的模式。在图上绘制标记时，JMP 会使用以下两种速度之一：正常或快速。

更改标记绘制速度

1. 右击图形中的任意位置。在直方图中，右击右侧的箱线图区域。
2. 选择**标记绘制模式**，然后选择**正常**或**快速**。

正态 若 JMP 处于正常绘制模式下而图形上的标记数大于指定的阈值，JMP 将自动切换到快速模式。有关设置标记阈值的详细信息，请参见 “JMP 首选项” 一章中第 587 页的 “JMP 报表的首选项”。

快速 若将标记绘制速度设置为“快速”，则显示大量标记的图形的显示速度会更快。请注意，若绘制速度设置为“快速”，则应用最小的标记大小（点），而且标记透明度设置也恢复为默认的不透明。

轮廓 请参见第 421 页的“为标记添加轮廓”。

填充空心标记

在 JMP 首选项中，选择**图形**，然后选择**填充空心标记**。所有图形上的空心标记都填充了不透明背景。

为标记添加轮廓

您可以为图中的标记添加黑色轮廓或外框。带轮廓的标记具有“中”、“大”、“特大”、“超大”、“最大”多种标记大小。请参见第 420 页的“更改标记的大小”。

添加轮廓

1. 右击某个图形。
2. 选择**标记绘制模式**。
3. 选择**轮廓**。

要有效使用轮廓，标记颜色最好不要使用黑色。

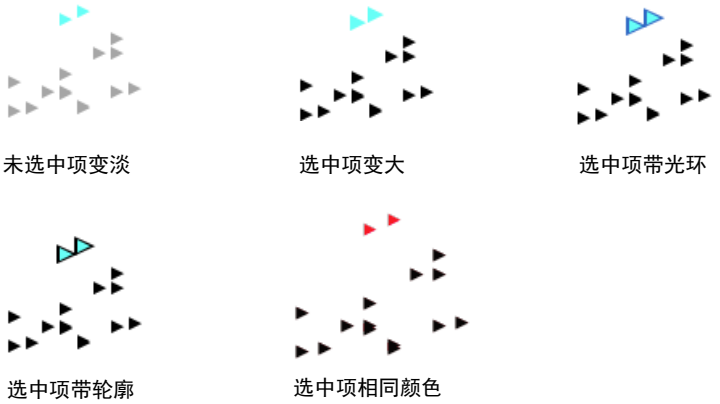
更改标记颜色

1. 突出显示要更改颜色的标记。
2. 右击图形中的任意位置。
3. 选择**行颜色**。
4. 从出现的选项中选择一种标记颜色。

更改标记的选择模式

选择图形上的标记时，只有选定的标记才突出显示。您可以更改标记在当前图形中突出显示的方式。选项应用于下图中最靠上的两个小三角。

图 9.22 突出显示的小三角标记的示例



要更改在当前图形中突出显示的标记，请执行以下步骤：

1. 右击图形中的任意位置并选择**标记选择模式**。
2. 选择以下选项之一：

首选模式 在 JMP 首选项的“图形”页面中选择的“标记选择模式”。默认值为“未选中项变淡”。

未选中项变淡 仅突出显示选定的标记。其他所有内容都按照“首选项”>“图形”中的**未选定标记变淡百分比**所指定的百分比变淡。

选中项变大 选定的标记比取消选定的标记大。

选中项带光环 选定的标记的边缘显示蓝色轮廓线。

选中项带轮廓 选定的标记加黑色外框。

选中项相同颜色 使用在“图形”首选项中选择的“标记选择颜色”对选中标记着色。

指定标记的透明度

您可以更改图形上标记（点）的透明度。例如，这将帮助您控制重叠点的可见性。

注意：绘制速度若设置为“快速”，则应用默认标记透明度。请参见第 420 页的“处理拥挤的标记”。

1. 右击图形中的任意位置并选择**透明度**。在直方图中，右击右侧的箱线图区域并选择**透明度**。
2. 输入您希望标记 （点）在图形中具有的透明度，然后点击**确定**。

   值为 1 表示完全不透明，值为 0 表示不可见。介于 1 和 0 之间的值表示半透明。

排除和隐藏标记

使用“行 > 隐藏和排除”命令可禁止显示并从统计分析中排除突出显示的行。数据保持隐藏和排除，直到您再次选择“行 > 隐藏和排除”为止。

使用“排除/撤销排除”命令，您可以从统计分析中排除突出显示的行。数据一直保持已排除状态，直到您为这些突出显示的行选择“行 > 排除/撤销排除”。

注意：已排除数据不会在图中自动隐藏，即使在文本报表和图形的计算中已经排除这些数据。

使用“隐藏/撤销隐藏”命令，您可以禁止（隐藏）突出显示点在散点图中的外观。例如，您可以从分析中排除点，然后在散点图中隐藏这些点。数据一直保持隐藏状态，直到您为突出显示的隐藏行选择“行 > 隐藏/撤销隐藏”。

注意：隐藏的点不会从影响文本报表和图形的统计计算中自动排除，即便它们不显示在图中。要从分析中排除隐藏的观测，您必须突出显示这些观测，然后选择“行 > 排除/撤销排除”特征。

要从分析中排除或隐藏标记 （点），请执行以下步骤：

1. 突出显示要排除或隐藏的标记。
2. 右击图形中的任意位置。
3. 选择**排除行**或**隐藏行**。

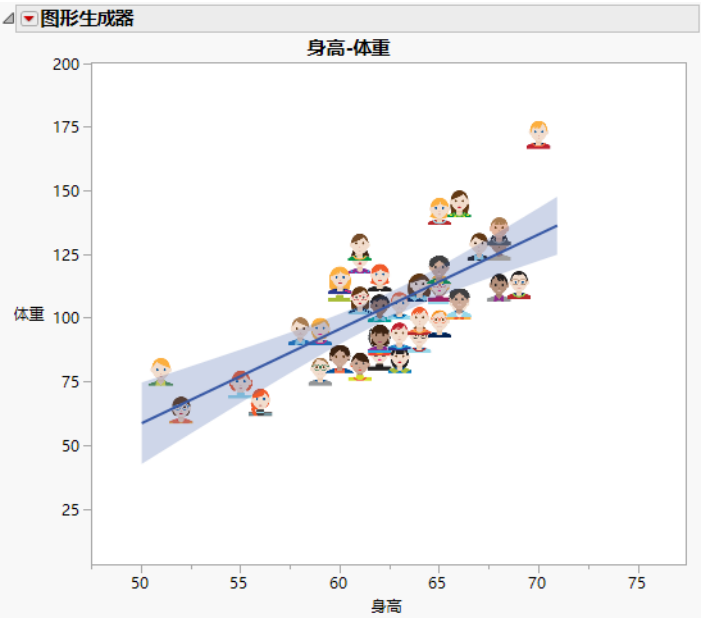
将图像用作标记

将图像放入表达式列，以便将这些图像用作图形上的标记。之后指出要将哪一列用作标记。该功能使得用户不再需要先查看标记颜色再参考图例了解其含义，这对于包含许多水平的数据尤其有帮助。

若将字符或数值列用作标记，则数据值用作标记。

1. 在您的数据表中创建新列。
2. 右击该列，选择**列信息**，然后选择**表达式**数据类型。
3. 点击**确定**。
4. 将您计算机中的图形拖入列中每个单元格。您还可以导入包含图像列的数据。
5. 选择该列，选择**列**菜单，然后选择**用作标记**。
6. 创建使用标记的图形。图 9.23 是在 “图形生成器”中创建的。
7. （可选）要调整标记大小，请右击该图形，然后选择**图形 > 标记大小**（或是在某些平台中右击并选择**标记大小**）。本例中选择了中等大小的标记。
8. 点击 “图形生成器”红色小三角并取消选择**显示图例**。

图 9.23 用作标记的图片



注意：

- 在 Big Class Families.jmp 样本数据表中，**图片**列用作图形中的标记。
- 该功能在“气泡图”中不受支持。

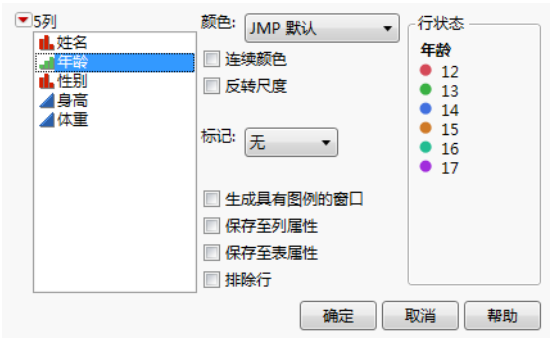
基于值更改标记的形状或颜色

在某些图中，您可以通过添加**行图例**基于点的值更改标记的形状或颜色。之所以称为行图例，是因为 JMP 会使用行颜色或行标记设置自动插入图例。以这种方式指定标记或颜色时，将为图形中的所有点指定特征，无论您选定了哪些点。所有以前的标记和颜色设置都将被覆盖。

要基于列值添加形状或颜色，请执行以下步骤：

1. 右击图形中的任意位置。在直方图中，右击右侧的箱线图区域。
2. 选择**行图例**。
3. 在出现的窗口中，突出显示要对列值进行着色或标记的列。图例预览显示在右侧。

图 9.24 添加行图例



4. 使用以下选项细化行图例：

颜色 支持您从多种预定义颜色方案中进行选择。

提示：若在数据表中已经应用了某种颜色方案，则选择“无”保持该方案。应该对颜色进行一致应用并唯一分组。

连续尺度 分配与值的升序或降序对应的色谱上的颜色。请在突出显示的列包含连续值时使用该选项。

反转尺度 反转颜色的尺度。

标记 支持您在多种标记方案中进行选择。

提示：若在数据表中已经应用了某种标记方案，则选择“无”保持该方案。应该对标记进行一致应用并唯一分组。

生成具有图例的窗口 创建单独的图例窗口，为您指明哪些颜色和形状与哪些值对应。

保存至列属性 添加存储选定的颜色主题的列属性。

保存至表属性 添加保留选定的颜色和标记配置的表属性。

排除行 为排除的行指定颜色或标记。

多数图例只有一列。但是，以下平台在包含 20 个以上的水平时会包含多列图例：

- 复发
- 单因子（对于 CDF 图和全部三个“密度”红色小三角命令）
- 拟合模型（对于“标准最小二乘法”中的回归图）
- 生存

删除行图例

右击行图例并选择**删除**可删除行图例。

更改图形外观

有多种方式可以设置报表格式以满足您的需要。随后的各节详细说明如何对输出报表的图形部分进行更改。

提示：若您有触摸屏，则可以在 JMP 中平移和缩放大多数图形和轴。图形和轴会更改尺度，若收缩则会缩小，若拉伸则会放大。

- “调整图形大小”
- “放大图形”
- “更改图形中的线条粗细”
- “更改图形中的背景颜色”
- “更改直方图直条的颜色”
- “显示图形中的坐标和临时参考线”
- “在图形中滚动和调整轴尺度”
- “定制图形中的轴和轴标签”
- “更改报表中的值顺序”
- “更改图形中选定对象的模式和格式”
- “将图形添加至数据表”
- “定制图形中的标记、线条、文本等元素”

调整图形大小

主要使用两种方法来调整 JMP 中图形的大小：使用点击并拖动的方法或根据像素大小调整图形大小。

注意：您还可以使用“文件 > 首选项 > 图形”中的“图形高度”选项更改图形的默认大小。

使用点击并拖动

使用点击并拖动方法调整图形的大小：

1. 将光标放在图框架的右侧边缘、底部边缘或右下角。光标变为小的双箭头。
2. 点击并拖动可更改图的大小。调整大小时，该框架内所有图的高度和宽度将独立于同一报表窗口内的其他框架进行调整。表 9.2 说明如何调整图。

表 9.2 调整大小操作

操作	说明
调整图框架但保留比例（纵横比）	按 Shift 键，然后点击并拖动框架的一角。
以 8 像素为增量调整图大小	按 Alt 键的同时点击并拖动框架的一角。
同时调整同类型的所有图	按 Ctrl 键的同时点击并拖动其中一个图的一角。若为一个散点图执行该操作，该操作将传播至窗口中的所有散点图，这些图将一起调整大小。其他任何类型的图都保持不变。

以像素为单位指定图形大小

将图形大小调整为特定的像素大小：

1. 右击图形。
2. 选择**大小 / 尺度** > **框架大小**。
3. 为框架的高度和宽度输入像素数。

注意：有关“大小/尺度”菜单中其他选项的详细信息，请参见第 429 页的“在图形中滚动和调整轴尺度”。

放大图形

放大镜 支持您自动放大图的任意区域。点击放大镜时，所点击的点或区域变为新数据视图的中心。新视图的尺度放大，您可以更近距离地观察感兴趣的点或模式。您可以执行以下任意操作：

- 点击并拖动放大镜可聚焦图中特定区域。
- 在三元图上，拖动放大镜可缩放三角轴。
- 反复放大可更近距离地观察数据。
- 按 Ctrl 键并点击放大镜可恢复最后一次缩放之前的状态。
- 双击或按 Alt 键并点击放大镜可恢复原始图。

在 Windows 上，也可以通过按 Ctrl 键并使用鼠标滚轮进行缩放。

在 macOS 上，按 Option 并点击该区域可进行缩放。

注意：通过从轴拖动到图中，可以放大并仍看到中心点。对于垂直轴，点击某个位置并向右拖到图中。这将放大尺度，调整尺度的两个端点，但将点击的值保持在大约相同的位置。向左拖动会执行缩小操作。

更改图形中的线条粗细

在图形中拟合线条后，或生成已存在线条的图形后，您可以调整线条粗细：

1. 右击图形中的任意位置。
2. 选择**定制**。
3. 选择**平滑线**（例如，在图形包含平滑线时）。
4. 更改线条粗细。该值以像素为单位。
5. 点击**确定**。

更改图形中的背景颜色

要更改图形中的背景颜色，请执行以下步骤：

1. 右击图形中的任意位置。（若仅更改箱线图的颜色，请右击箱线图区域。）
2. 选择**背景颜色**。
3. 选择预定义颜色之一，或自行创建颜色。
4. 点击**确定**。

更改直方图直条的颜色

要更改直方图直条的颜色，请执行以下步骤：

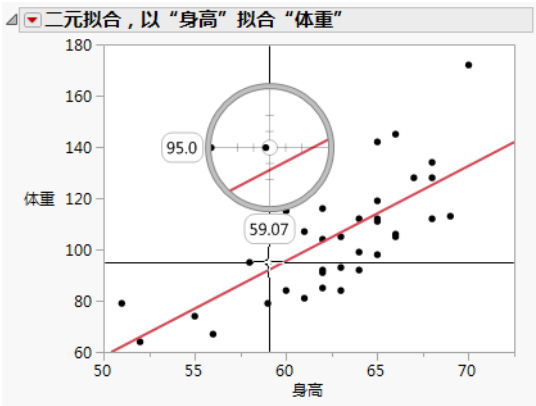
1. 右击直方图中的任意位置并选择**直方图颜色**。
2. 点击一种颜色，或点击**其他**自行创建颜色。

请参见您操作系统的文档，了解有关如何自行创建颜色的详细信息。

显示图形中的坐标和临时参考线

您可以测量图形中的点和距离，或在图形上轻松查找点和距离的精确值或坐标。这需要点击十字准线工具 ，然后在图形中的任意位置点击并按住。在图中拖动十字准线时，只要十字准线与垂直轴和水平轴相交，就会显示坐标值。

图 9.25 使用十字准线工具



在拟合线或曲线上，十字准线标识任意预测值的响应值。在三元图上，该工具显示三角形十字准线。

在图形中滚动和调整轴尺度

使用手形工具（也称为抓手工具）( ) 可以更改图的坐标轴和视图：

- 在 *y* 轴上，拖动 可滚动 *y* 轴；拖动 或 可调整 *y* 轴的尺度。
- 在 *x* 轴上，拖动 可滚动 *x* 轴；拖动 或 可调整 *x* 轴的尺度。

提示：当您拖动轴以更改其尺度时，JMP 会根据新的轴宽度自动更新主刻度和副刻度增量。要防止出现这种情况（保留原始增量），请在拖动时按 Shift 键。

您还可以右击图形，然后选择**大小/尺度**（或**图形 > 大小/尺度**）。选择以下选项之一：

- 要调整 X 轴的尺度，请选择 **X 轴**。要调整 Y 轴的尺度，请选择 **Y 轴**或**右侧 Y 轴**。有关该窗口的详细信息，请参见第 429 页的“定制图形中的轴和轴标签”。
- 有关“框架大小”选项的详细信息，请参见第 427 页的“以像素为单位指定图形大小”。
- 若 *x* 轴和 *y* 轴的测量单位相同，而您希望无论方向如何都要精确表示图形上的距离，请选择**调整为相同大小**。

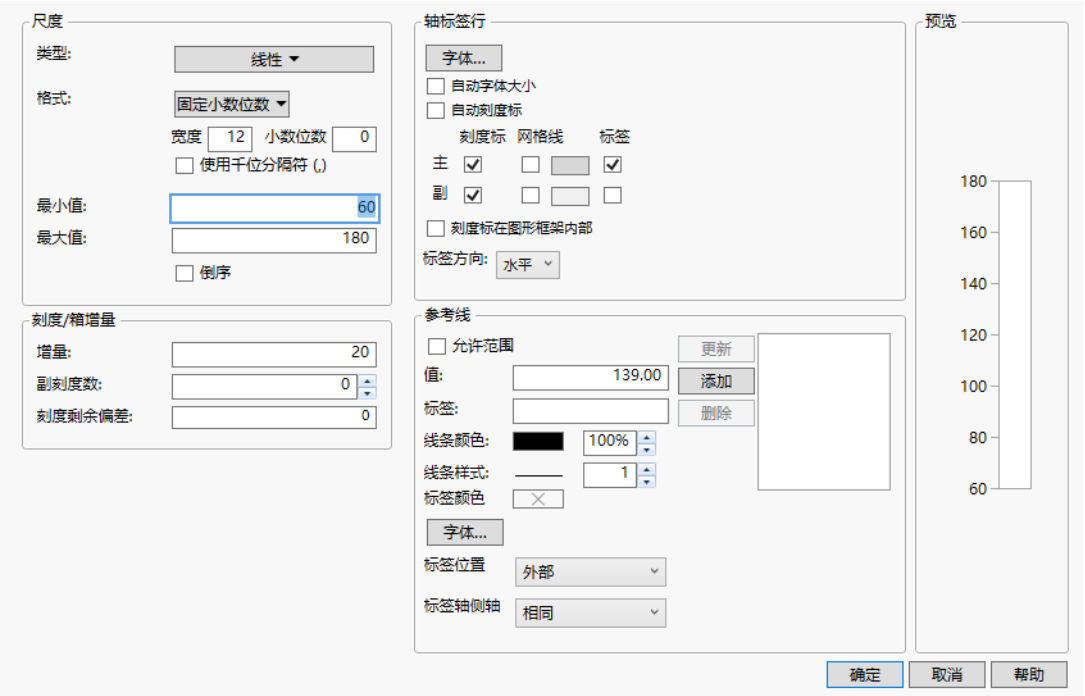
定制图形中的轴和轴标签

双击数值轴可使用“轴设置”窗口定制该轴。或者，右击轴区域并选择**轴设置**也可以访问该窗口。

注意：在“图形生成器”和“分布”平台中，您还可以使用“轴设置”窗口定制名义型轴。

窗口中的定制功能取决于轴的数据类型和 JMP 创建图或图表所用的特定平台。图 9.26 显示了数值（连续）轴的典型“轴设置”窗口。

图 9.26 数值（连续）轴的“轴设置”窗口



“轴设置”窗口包含以下面板：

- 第 430 页的“尺度”
- 第 432 页的“刻度/箱增量”
- 第 434 页的“轴标签行”
- 第 437 页的“参考线”
- “预览”面板显示您当前的选择将如何在轴上显示。

尺度

在“尺度”面板中，您可以执行以下操作：

- 第 431 页的“更改轴尺度类型”
- 第 431 页的“更改轴的数值格式”
- 第 432 页的“设定最小和最大坐标轴值”

更改轴尺度类型

查看具有数值轴的图形时，您可以将该轴的尺度更改为以下任意一种类型：

- 线性
- 对数
- 乘方
- 测地线
- 测地线 US
- 概率尺度（正态、Weibull、Fréchet、Logistic、指数、Gamma、Beta 和混合）

注意：

- 特定平台可能使用其他不能更改的固定尺度类型。例如，“破坏性退化”、“退化”和“以 X 拟合 Y”散点图中的 X 轴使用定制尺度。不要通过“轴设置”窗口更改这些图形中的尺度。
- 若选定尺度类型为“对数”，请输入要使用的底。
- 若选定尺度类型为“乘方”，请输入要使用的乘方。
- 乘方尺度的计算方式与对数尺度类似，但它不是将尺度转换为 log(x)，而是计算为 x^p，其中 p 是一个可定制的轴选项。例如，乘方 = 2 将创建一个 x^2 轴。

要为变量设置默认尺度类型（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“轴”。

更改轴的数值格式

对于包含数值轴区域的图和图表，您可以更改轴的格式。有关数值格式的详细信息，请参见“设置 JMP 列属性”一章中第 256 页的“数值格式”。

注意：

- 若您选择了日期、时间或持续时间，您需要指定增量的格式。请参见第 438 页的“添加和删除轴标签”。您还可以指定标签行嵌套。请参见第 434 页的“标签行嵌套”。
- 若您选择了固定小数位数，请输入您想要 JMP 在“小数位数”框中显示的小数位数。
- 若您选择了精度，请选择是否保留尾随零和所有整数位。
- 要向表示千位或千位以上的值添加逗号，请选择**使用千位分隔符**选项。必须在“宽度”框中为每个逗号占一个位置，否则这些逗号可能不显示。该选项可用于最佳、固定小数位数、百分比和货币格式。
- 对于数值轴，您可以使用宽度框调整刻度标签的宽度。

注意：更改轴的数值格式时，并不会更改相应数据表中值的数值格式显示方式。要更改日期或时间在数据表中的显示方式，请参见“设置 JMP 列属性”一章中第 256 页的“数值格式”。

从日期增量下拉菜单中选择一个日期间隔会将 JMP 日期（秒数）划分为适当的单位。这会为图指定您希望数据具有的尺度。日期轴必须是具有 JMP 日期值的列，并以“列信息”窗口中的日期格式显示在“轴设置”窗口中。不过，您可以使用“轴设置”窗口设置日期在图中的外观格式。

设定最小和最大坐标轴值

对于包含数值轴区域的图和图表，您可以更改希望图形显示的最小值和最大值。

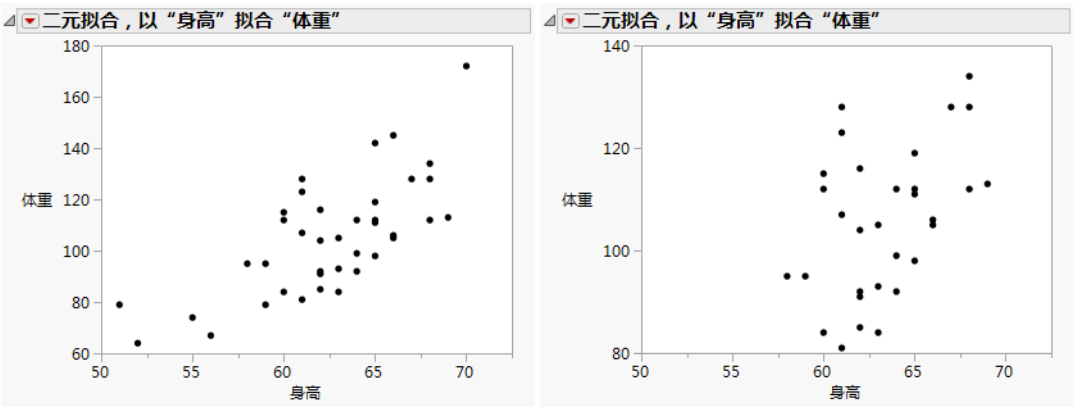
请注意以下几点：

- 要恢复数值轴的默认最小和最大轴设置，请右击数值轴并选择**恢复轴**。
- 选择**倒序**可通过颠倒最小值和最大值反转坐标轴。请注意，三元图不支持该选项。

提示：要为变量设置默认最小和最大坐标轴值（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“轴”。

图 9.27 右侧的示例是放大显示的点群，显示左侧图中介于刻度 80 到 140 之间的点。放大图是通过重新指定最大和最小坐标轴值来实现的。

图 9.27 调整轴尺度以放大图中部分



刻度 / 箱增量

在“刻度 / 箱增量”面板中，您可以执行以下操作：

- 第 433 页的“更改轴增量”
- 第 433 页的“添加副刻度标”
- 第 433 页的“更改刻度偏差”
- 第 434 页的“标签行嵌套”

更改轴增量

查看图形时，您可以更改轴增量。

若轴“格式”设置为日期、时间或持续时间，格式菜单将在增量旁边显示（图 9.28）。选择增量的格式。

图 9.28 选择日期和时间增量的格式

提示：要为变量设置默认轴增量（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“轴”。

添加副刻度标

您可以向数值轴添加刻度标或者更改在数值轴上显示的副刻度数。在“轴标签行”面板中，务必选中**副刻度标**框，这样刻度标便出现在轴上。

提示：要为变量设置默认副刻度标（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“轴”。

更改刻度偏差

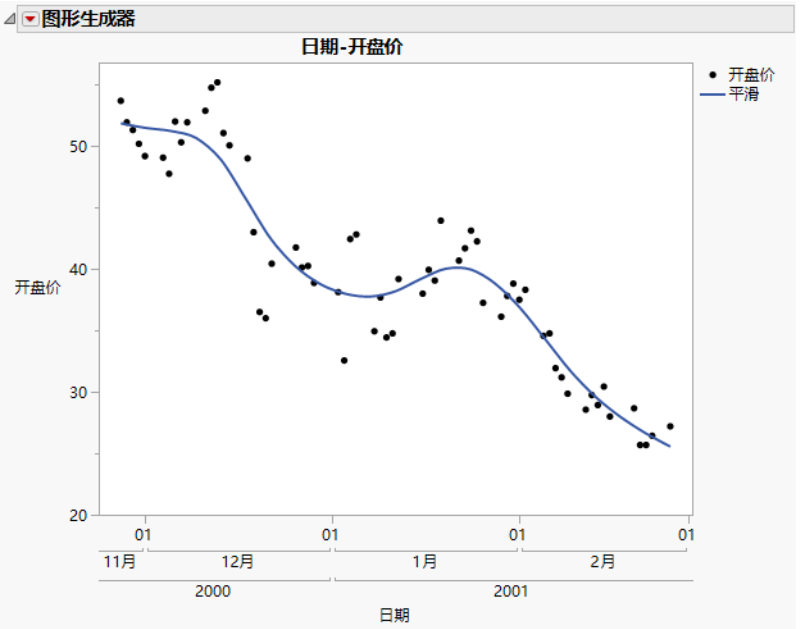
要更改刻度标的起始点，请在刻度偏差框中输入数字。例如，若“刻度偏差”当前设置为 0，将其设置为 1 后，轴上所有的值都将增加 1。

标签行嵌套

只有当您选择了日期、时间或持续时间作为轴格式时，该选项才会出现。通过标签行嵌套，您可以根据格式将一个日期轴分离成多行。例如，您可以将“年”放在最外层的行上，“年”里面是“月”，“月”里面是“天”：

1. 选择**帮助** > **样本数据库**，然后打开 Stock Prices.jmp。
2. 选择**图形** > **图形生成器**。
3. 将**日期**拖至 X 区域。
4. 将**开盘价**拖至 Y 区域。
5. 双击 X 轴。
6. 对于**格式**，选择**日期** > ddMonyyyy。
7. 将**标签行嵌套**的值增加至 3。
8. 点击**确定**。
9. 点击**完成**以查看最终的图形。

图 9.29 标签行嵌套示例



轴标签行

在“轴标签行”面板中，您可以执行以下操作：

- 第 435 页的“更改轴标签字体”

- 第 435 页的“添加刻度标、网格线或标签”
- 第 437 页的“更改刻度标签的方向”

注意：在“图形生成器”中，若您在轴上嵌套了标签，则选项卡会出现在“轴标签行”面板中。您可以在其各自的选项卡中单独修改标签的每一行。

更改轴标签字体

您可以针对任意轴类型修改轴标签字体。您的更改仅应用于活动图形。要设置默认轴标签字体，请参见“JMP 首选项”一章中第 612 页的“‘字体’首选项”。

要更改当前字体类型和大小，请双击某个轴，选择**轴设置**，然后点击**字体**。

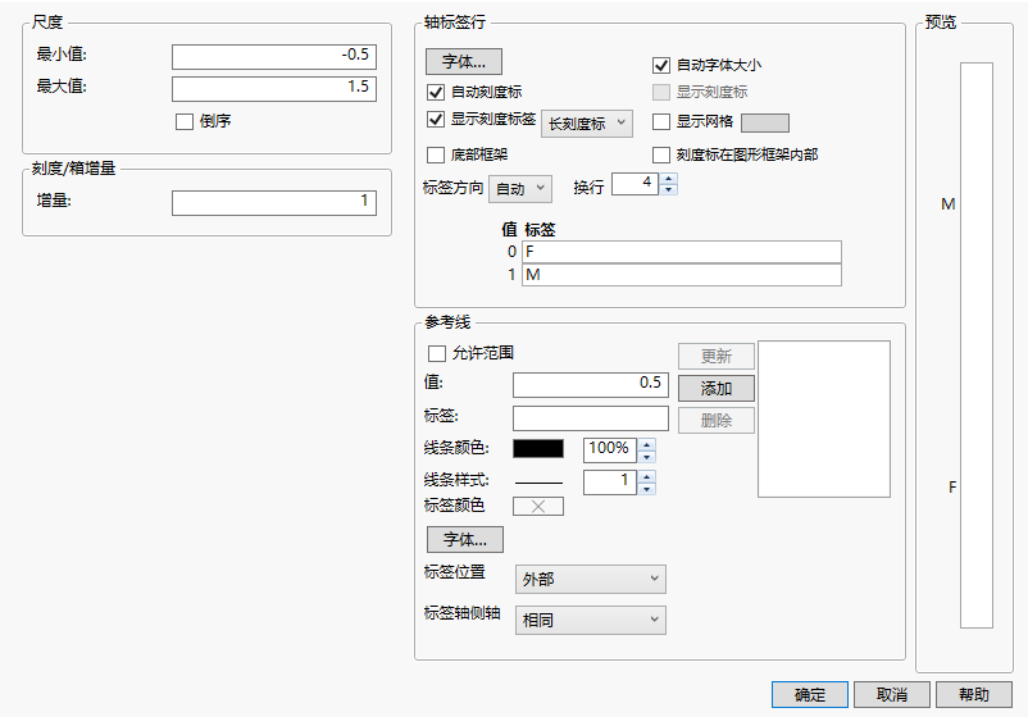
注意：

- 若所有的标签无法适合默认字体大小，请选择**自动字体大小**使 JMP 尝试缩小字体大小（缩小至某个最小值）。
- 要更改轴标签字体颜色，右击该轴并选择**显示属性**。在右侧的“编辑”选项中，更改文本颜色。

添加刻度标、网格线或标签

这些选项可能会有所不同，具体取决于坐标轴是连续轴还是分类轴。图 9.30 显示分类轴的轴设置。第 430 页上的图 9.26 显示连续轴的轴设置。

图 9.30 分类轴的轴标签行



连续轴的选项：

- 仅当一个或多个标签隐藏时（空间不足所致），才选择**自动刻度标**打开刻度标。
- 通过选择相应的复选框，添加主副刻度标、网格线或标签。
- 点击颜色框可更改网格线的颜色。
- 选择**刻度标在图形框架内部**可将刻度标移入图形。

分类轴的选项：

- 若所有标签都可见，则默认不显示刻度标。要添加刻度标，请选择**自动刻度标**。选择其中一个选项可更改刻度标样式。
- 仅当一个或多个标签隐藏时（空间不足所致），选择**自动刻度标**打开刻度标。
- 选择**显示网格**添加网格线。
- 点击颜色框可更改网格线的颜色。
- 选择**刻度标在图形框架内部**可将刻度标移入图形。
- 选择**底部框架**可在标签周围添加框架。
- 更改文本换行的行数。
- 在标签框中输入不同的值可修改标签。

提示：

- 要为变量设置默认刻度标（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“控制列显示的属性”。
- 有关更多的轴定制选项，请从轴右击菜单中选择**显示属性**。

更改刻度标签的方向

要更改刻度标签的方向，请从标签方向列表中选择一项。

自动 刻度标签根据标签的刻度增量和长度自动调整为容易辨读的方向。

水平 以底部为基线。

垂直 以右侧为基线垂直显示。

竖直 垂直轴水平显示，水平轴垂直显示。

注意：平行和竖直对成对轴很有用，例如在多元图中。

平行 垂直轴垂直显示，水平轴水平显示。

倾斜 以 45 度角倾斜。

参考线

向图形添加参考线：

1. 双击某一数值轴。或者，右击某一数值轴并选择**轴设置**。
2. 在值文本框中，输入您要与参考线对应的值。这是在图形上放置线条的位置。

   注意: 若您通过双击图形的方式打开“轴设置”窗口，则默认值与您在图形上所点击的点对应。
3. 为该线条输入标签。
4. 您可以通过选择以下其中一个选项进一步定制参考线：

   允许范围 支持您输入定义参考线的起始和结束的最小值和最大值。参考线出现在一定范围的数据上。

   线条颜色 更改线条颜色。您还可以指定不透明度。

   线条样式 指定线条样式（当未选择“允许范围”时）。您还可以指定线条粗细。

   字体 支持您调整参考线标签的字体。

   标签位置 支持您选择参考线标签的位置。选项包括图形框架外部或内部。

   标签轴侧轴 支持您为参考线标签选择图形一侧。
5. 点击**添加**。该值移入 “添加”按钮右侧的框，指示它将被放在图形上。您的更改会出现在“预览”窗口中。
6. 要添加更多线条，请重复上述步骤。

7. 要修改现有参考线，请进行更改，然后点击**更新**。

提示：要为变量设置默认参考线（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“控制列显示的属性”。

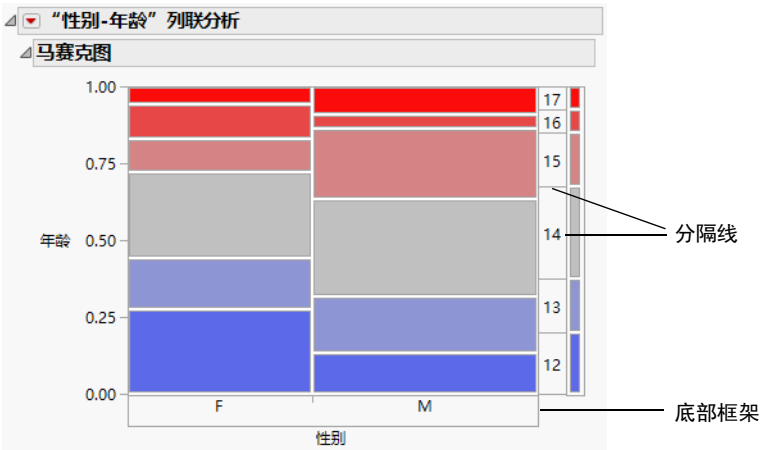
延伸分类轴的分隔线和框架

若名义型或有序型（分类）变量具有许多水平，则延伸刻度标签之间的垂直分隔线很有用。

要将分隔线延长到 *x* 轴标签，请执行以下步骤：

1. 双击某个名义型或有序型轴以显示轴设置。
2. 选择**显示刻度标** > **显示刻度标的短分隔线** > **长分隔线**以添加分隔线，或者选择**底部框架**在轴区域周围添加框架。

图 9.31 分隔线



添加和删除轴标签

您可以在数值轴中添加或删除标签。添加轴标签：

1. 右击某一数值轴并选择**添加轴标签**。
2. 输入轴标签的名称。轴区域放大以容纳您输入的标签行数。

可多次使用该命令添加多个标签。将标签添加至轴之后若要对其进行编辑，请点击该标签，标签将变为一个编辑框。

要删除轴标签，请右击某个数值轴并选择**删除轴标签**。随即删除最后一个添加的标签。

旋转轴标签和刻度标

您可以针对任意轴类型修改轴标签。

要旋转轴标签，请执行以下步骤：

1. 双击某个轴以显示轴设置。
2. 从“标签方向”列表中，选择要旋转文本的方向。

提示：要为变量设置默认轴标签位置（这样设置可避免每次运行分析时都进行该更改），请参见“设置 JMP 列属性”一章中第 270 页的“控制列显示的属性”。

通过右击刻度标签并选择**旋转刻度标签**，可以在名义型轴上旋转刻度标。

复制和粘贴图形内容

通过添加拟合线之类的元素定制图形之后，您可以将一个图形中的内容复制并粘贴至另一个兼容的图形：

1. 右击经过定制的图形。
2. 选择**编辑 > 复制框架内容**。
3. 右击要将设置复制到的图形。
4. 选择**编辑 > 粘贴框架内容**。

提示：若您遇到粘贴直方图直条方面的问题，请参见《Scripting Guide》。

复制并粘贴轴设置

定制轴之后（如第 429 页的“定制图形中的轴和轴标签”所述），您可以将新设置复制并粘贴至另一个轴：

1. 右击经过定制的轴。
2. 选择**编辑 > 复制轴设置**。
3. 右击要将设置复制到的轴。
4. 选择**编辑 > 粘贴轴设置**。

提示：要将轴设置传播到多个图形，请在按 Ctrl 键的同时右击该轴并选择**编辑 > 粘贴轴设置**。

更改报表中的值顺序

JMP 报表中的数据可能并未按照您希望的顺序显示。要为数据指定具体顺序，使其按该顺序显示在报表中，请在运行分析前先为列指定“值顺序”属性，如“设置 JMP 列属性”一章中第 267 页的“值顺序”中所述。

若值包括以下任意内容，它们会自动在报表中以适当的顺序显示：

- 一月、二月、三月、四月、五月、六月、七月、八月、九月、十月、十一月、十二月

- 一月、二月、三月、四月、五月、六月、七月、八月、九月、十月、十一月、十二月
- 星期日、星期一、星期二、星期三、星期四、星期五、星期六
- 极低、低、中低、中、中高、高、极高
- 完全不同意、不同意、中性、无关紧要、同意、完全同意
- 令人失望、不可接受、很差、差、较差、可接受、平均、好、更好、很好、非常好、最佳

更改图形中选定对象的模式和格式

当您选择某些对象（例如，直条和地图）时，这些对象上会显示模式。您可以更改图例中特定对象的模式。

右击图例中的项，然后选择**填充模式**以选择新的模式。

要删除模式，请再次右击该项，然后选择**恢复**。

在所有图形中修改选定对象的格式

选择**文件 > 首选项 > 图形** (Windows) 或 **JMP > 首选项 > 图形** (macOS) 以更改“填充区域”首选项。

- 要更改选定对象的突出显示，选择以下任意一个“填充选择模式”首选项：

  选中项带图案 条纹图案（默认设置）

  选中项颜色变深 纯暗色

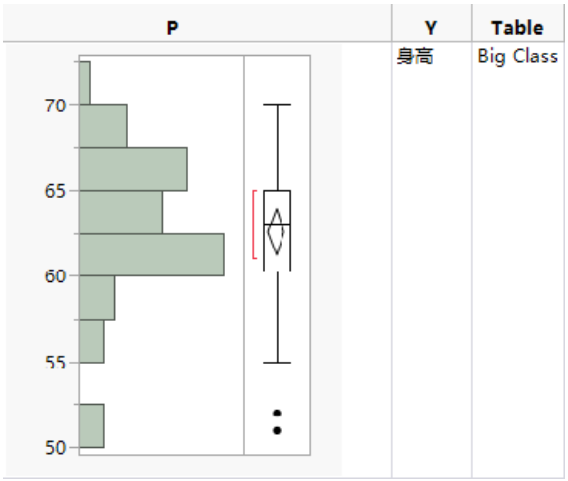
  选中项带轮廓 轮廓

  选中项相同颜色 首选项中的“填充选择颜色”设置选定对象的格式。
- 要在“填充选择模式”为“选中项相同颜色”时更改图形对象中选定行的颜色，请点击“填充选择颜色”栏。默认颜色为红色。

将图形添加至数据表

您可以将图形添加至新的数据表。右击图形并选择**编辑 > 生成像这样的图形表**。图形和变量将出现在新的数据表中。然后调整行和列的大小以放大图形。

图 9.32 添加至数据表的直方图和 Y 变量



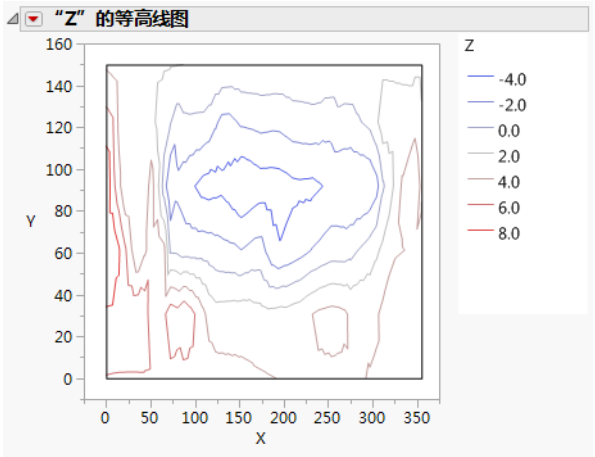
定制图形中的标记、线条、文本等元素

图形包含标记、线条、文本和其他可以定制的图形元素。例如，在散点图矩阵图中，您可能想要使用粉色实心标记突出显示其中一个二元图形中的数据点。在等高线图中，您可以增加等高线的宽度或透明度。

定制等高线图的示例

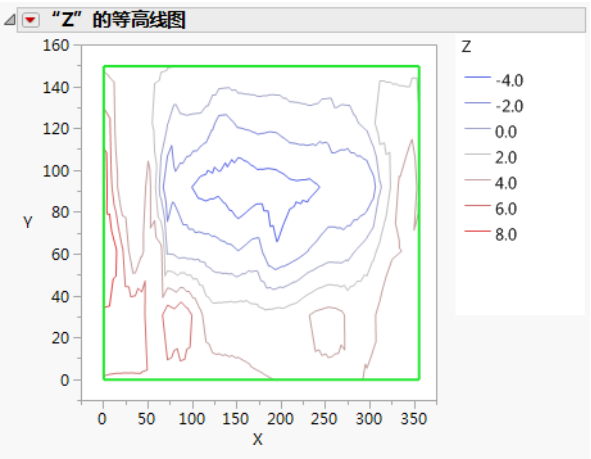
1. 选择**帮助** > **样本数据库**，然后打开 Little Pond.jmp。
2. 在表面板中，点击 “等高线图”脚本旁边的绿色小三角。
3. 点击 “‘Z’的等高线图”旁边的红色小三角，然后取消选择**填充区域**。

图 9.33 定制之前的等高线图



4. 右击该图并选择**定制**。
   随即显示“定制图形”窗口。
5. 点击**形状**。
6. 将**线条粗细**改为 2.0。
7. 点击**线条颜色**旁边的颜色，然后选择绿色。
8. 点击**确定**。

图 9.34 经过定制的等高线图



每个图形可定制的图形元素各有不同。例如，在“控制图”图形中，有三种线元素（下限、上限和连接线）。在其他图形中，线元素可能名为线条或定制。

要在当前图中定制图形元素，请执行以下步骤：

1. 右击图形并选择**定制**。
2. 选择要更改的元素，然后修改属性。
   您的更改立即显示在图形中。
3. 点击**确定**将更改保存到当前图形。

注意：“定制”设置不会覆盖图例中显示的设置。例如，若将条形图中的填充区域设置为红色，并且图例为蓝色，则图例将保持蓝色。

每个元素可以包含线条颜色、标记样式和大小、文本样式等属性。以下是一些常见属性：

线条颜色 更改线条的颜色。点击可在窗口中选择任意颜色。右击可从更多颜色中进行选择。

线条样式 更改线条的样式。点击可选择五种不同样式之一。

线条粗细 更改线条的粗细。在框中点击并以磅值为单位输入所需的线条粗细。

标记 更改标记形状。有关标记的详细信息，请参见第 419 页的 “在图形中使用标记”。

标记大小 更改标记的大小。

箭头 在线条的开头和 / 或结尾添加箭头，或不添加任何箭头。

拟合线 更改线条的颜色、粗细或样式。

文本颜色 更改文本的颜色。

字体 更改文本的字体、样式和磅值。

文本样式 将文本对齐方式改为居中、左对齐或右对齐。**填充**选项可将选定颜色应用到背景。

填充颜色 更改箱线图和拟合置信区域之类的对象的颜色。

填充模式 更改填充对象的模式。

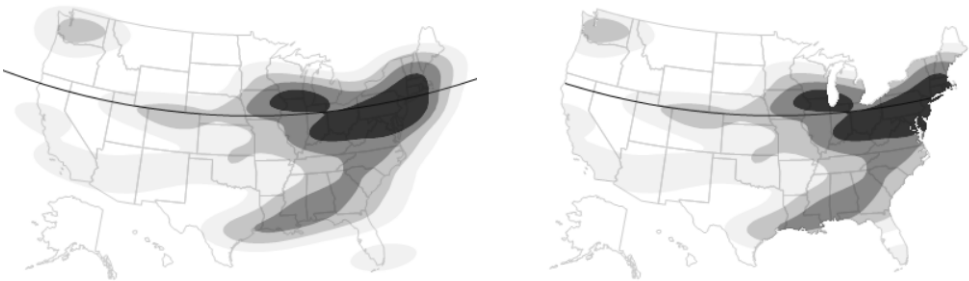
透明度 更改标记或标签的透明度。输入在图形上绘制标记 （点）的透明度水平。透明度介于 0（清晰）到 1（不透明）之间。有关更改标记透明度的详细信息，请参见第 422 页的 “指定标记的透明度”。

剪切形状 根据所选的剪切形状沿边界裁剪对象 （如等高线）。构成该对象的显示段随后会与地图边界相吻合。选择子边界，仅在一部分边界上显示对象。例如，您可能希望只在 “纽约” 上方显示这些对象。

关于剪切形状选项

可以剪切图形元素（参考线、网格线和等高线之类的显示段）以便与地图边界吻合。“定制图形” 窗口提供 “剪切形状” 选项，从中可选择围绕其剪切形状的边界。图 9.35 显示围绕美国地图剪切的参考线。

图 9.35 未剪切 （左）和已剪切 （右）的参考线



提示：

- 请参见《基本绘图》，了解关于如何创建图 9.35 中所示地图的示例。
- “定制” 窗口的 “剪切形状” 选项也同样设置为剪切等高线以便与地图边界吻合。请参见第 443 页的 “剪切形状”。
- 编写脚本可提供更多选项，如：在矩阵或字符串中指定剪切路径。请参见《Scripting Guide》。

创建图形元素的脚本

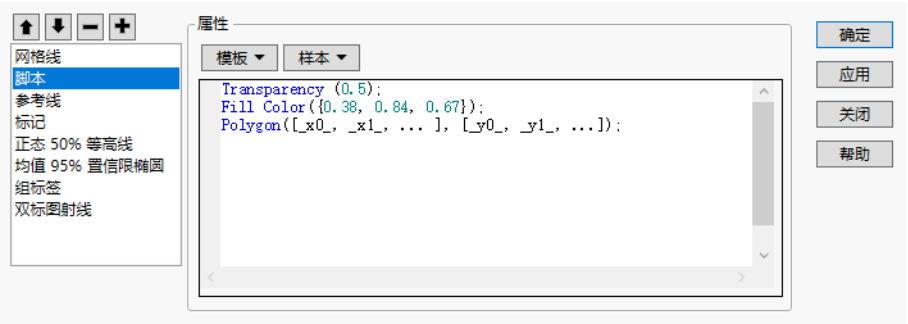
除了定制图形元素，您还可以编写用来添加元素的 JSL 脚本。显示图形时即运行这些脚本。您可以从头编写脚本，也可以从点击**添加新脚本按钮** ✚ 可访问的以下命令和脚本列表中选择：

- 使用模板列表可插入单个 JSL 命令。例如，使用多边形选项可插入多边形命令。下划线之间的文本是点值的占位符，可用您自己的值替换。

  `Polygon([_x0_, _x1_, ...], [_y0_, _y1_, ...]);`

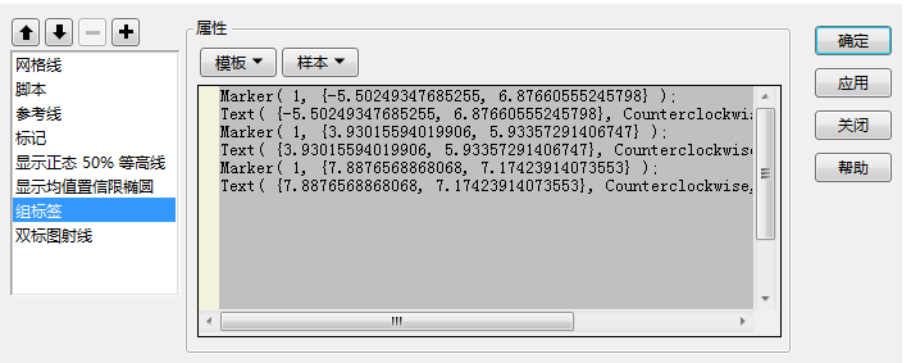
- 使用样本列表可插入创建气泡图和正弦波之类元素的脚本。在该列表中，多边形样本脚本显示了透明度、填充颜色和多边形等值的示例，您可以用自己的值来替换这些值。

图 9.36 多边形样本脚本



在某些图形中，您可以查看创建图形元素的 JSL。图 9.37 显示了判别分析图的“组标签”脚本。该脚本定义组标签的标记和文本。无法修改或删除这些内置脚本中包含的命令。您可以从“模板”列表或“样本”列表中插入命令，但无法在窗口中点击和键入新命令。

图 9.37 创建 “组标签”的 JSL 示例



有关 JSL 的信息，请参见《Scripting Guide》和《JSL Syntax Reference》。

要创建图形脚本，请执行以下步骤：

1. 右击图形并选择**定制**。

2. 点击**添加新脚本**按钮 (+) 创建新脚本。

   默认名称 “脚本” 随即突出显示。

3. 在默认名称突出显示时，键入更具描述性的名称，然后按 Enter 键。（若已经移动光标，名称不再突出显示，请双击 “脚本”，然后输入新名称。）

4. 执行以下一项或多项操作：
   - 在 “属性” 窗口中输入 JSL。
   - 从模板列表中选择一个或多个 JSL 命令，然后修改占位符文本。例如，将画笔颜色选项从 “蓝色” 改为 “红色”。
   - 从样本列表中选择一个或多个样本脚本，并根据需要修改脚本。

5. （可选）点击**应用**可使用所做更改更新图形而不会关闭该窗口。与其他属性更改不同，在您点击**应用**或**确定**之前，脚本更改不会生效。

6. 点击**确定**保存所做更改。

   出现以下之一：
   - 您创建的元素出现在图形上。
   - 若脚本包含错误，则出现错误消息。选择**视图 > 日志**可查阅相关错误，据此修改脚本。

要删除创建的脚本，请选择该脚本，然后选择**删除选定的对象**按钮 (−)。

更改图形元素的绘制顺序

图形元素按照其在列表中所处的顺序绘制。首先绘制列表中的第一个元素，所以该元素显示在其他所有图形元素**背后**。若一个元素遮挡另一个元素，您可以重新排列各元素的顺序。

重新排列图形元素

1. 右击图形并选择**定制**。
2. 选择要移动的元素。
3. 按需点击一次或多次点击**在绘制顺序中上移**按钮 (↑) 或**在绘制顺序中下移** (↓) 按钮，直到元素符合您需要的绘制顺序。
4. 点击**确定**。

保存定制内容

图形定制内容应用于当前图形，在重新运行分析时也会用到图形定制内容。要在以后使用图形定制内容重新创建图形，请从红色小三角菜单中选择**保存脚本**，然后选择其中一个 “保存” 选项。例如，您可以将脚本保存至数据表，这样每次运行脚本时就会应用定制属性。请参见 “保存和共享您的数据” 一章中第 518 页的 “将 JMP 分析另存为脚本”。

复制定制内容

您可以将定制内容从一个图形复制到另一个图形。您创建或修改的所有对象 （如着色的文本或线条）会粘贴至另一个图形。

要将线条从一个图形复制到另一个图形，请参见第 439 页的“复制和粘贴图形内容”。

复制其他对象

1. 在包含定制元素的图形中右击，然后选择**编辑 > 复制定制内容**。
2. 在目标图形中右击，然后选择**编辑 > 粘贴定制内容**。
   对象显示在当前图形中，并添加到定制元素列表中。

注意：

- 当您按 Ctrl 键，点击并粘贴时，定制内容会粘贴到所有相似的图形中。
- 复制定制内容功能仅复制您添加或修改的元素，不会复制图形中的其他内容。

将图像添加到 JMP 图形或报表

要使用徽标和其他图形类型增强分析效果，请粘贴图形或将其拖放到图或报表中。支持 .bmp、.emf、.jpeg、.png 和 .rtf 图形格式。

- “将背景图像粘贴到 JMP 图形中”
- “在 JMP 报表结尾粘贴图像”
- “将图像拖放到 JMP 图形中”
- “向 JMP 图形添加地理图像和边界”
- “从图像中提取数据”

将背景图像粘贴到 JMP 图形中

1. 使用图形程序打开该图形文件，然后选择该图形。将其复制到计算机的剪贴板中。
2. 在图形内右击。
3. 选择**编辑 > 粘贴背景图像**。图像将插入您在图形中右击的位置。

在 JMP 报表结尾粘贴图像

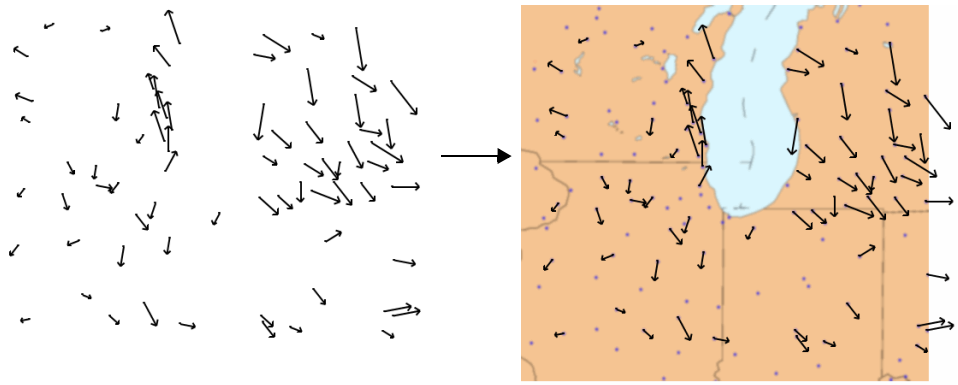
您还可以将图形添加到报表窗口末尾。将该图形从图形程序复制到您计算机的剪贴板，显示报表窗口，然后选择**编辑 > 粘贴**。

将图像拖放到 JMP 图形中

要添加您创建的图像，需要生成报表，然后将该图像从您计算机的文件系统拖放至报表中的图形。添加图像后，JMP 会提供若干选项，如调整图像大小、设置图像格式和旋转图像。右击该图像并选择**图像**可查看这些选项。

以下示例显示芝加哥地区风速的二元图。左侧图包含用于指示风向和风速的箭头。在图中放入了一个地图图像，该图像的大小经过调整，各标记已能与提供这些风数据的观测站对齐（每个点表示一个观测站）。

图 9.38 定制地图图像的示例



图像处理

JMP 包括各种图像处理选项。图像处理包括可以锐化、模糊、调整大小和增强的滤镜。JMP 还允许翻转、旋转和锁定图像。本节介绍了处理选项。

注意：

- 对图像反复更改将降低图像质量。若对调整图像大小、调整图像尺度或对图像应用过滤器后的效果不满意，请删除图像，然后重新开始。
- 在操作系统级别支持所有 JMP 图像过滤器，在 Windows 和 macOS 上处理图像的方式可能有所不同。

锁定

将图像锁定在原位，以防移动。

大小/尺度

- **填充图形：**按比例调整图像的大小以适合放置在图形中。
- **指定大小：**根据您输入的值调整图像的大小。（这些值的单位与图形轴单位相同。）

- **裁剪**：根据您输入的值裁剪图像。（这些值的单位与图形轴单位相同。）例如，左边缘可能定位在 50。若您在“左”旁边输入 60，则介于 50 到 60 之间的图像部分将从图像中删除。

翻转

- **垂直翻转**：将图像上下翻转。
- **水平翻转**：将图像左右翻转。
- **水平和垂直翻转**：同时将图像水平和垂直翻转。

旋转

按指定角度旋转图像。输入负值可逆时针旋转图像。

透明度

更改图像透明度水平。输入 0 到 1 之间的值（1 为完全不透明）。

过滤

提供在许多图形编辑程序中都提供的滤镜，用来更改图像的外观。反复选择过滤器可增强其对图像的效果。

- **对比度**：优化浅色和深色。更大的值会加亮图像。
- **去斑**：删除与周围像素不融合的像素。例如，被白色像素包围的黑色像素将转换为白色像素。
- **边缘**：除了对象外框，其他内容均变暗。
- **增强**：降低噪声图像中像素之间的对比度。
- **Gamma**：平衡图像的亮度和红绿蓝 (RGB) 比率。较大的值可创建更亮的图像。
- **高斯模糊**：按指定半径模糊处理像素。较大的半径可创建更平滑的图像。（在 JSL 中，您还可以指定 sigma 值。较大的 sigma 值可创建更平滑的图像。）
- **中间值**：用周围像素的中间值替换每个像素的颜色值。
- **负片**：将每个像素转换为其互补色（如，粉色转换为绿色，白色转换为黑色）。
- **标准化**：删除一定百分比的高低颜色值。颜色值随后延伸以填充其余图像。该过程增加了颜色的强度。
- **降噪**：查找最小和最大颜色值，替换为与周围像素更协调一致的值。较大的值可创建更平滑的图像。
- **锐化**：使像素边缘更为明显。

删除

从报表中删除图像。

向 JMP 图形添加地理图像和边界

向图形添加地图图像和行政边界可为地理空间数据提供可视环境。例如，您可以向图形添加显示美国图像的地图。另一个选择是显示每个州的边界（在数据包含美国的纬度和经度时）。

数据应包含纬度和经度坐标。否则，地图在数据上下文中没有意义。根据地图类型，x 和 y 轴还具有相应的范围要求。以下各节对这些要求进行了说明。

查看地图时有两种工具尤为有用：

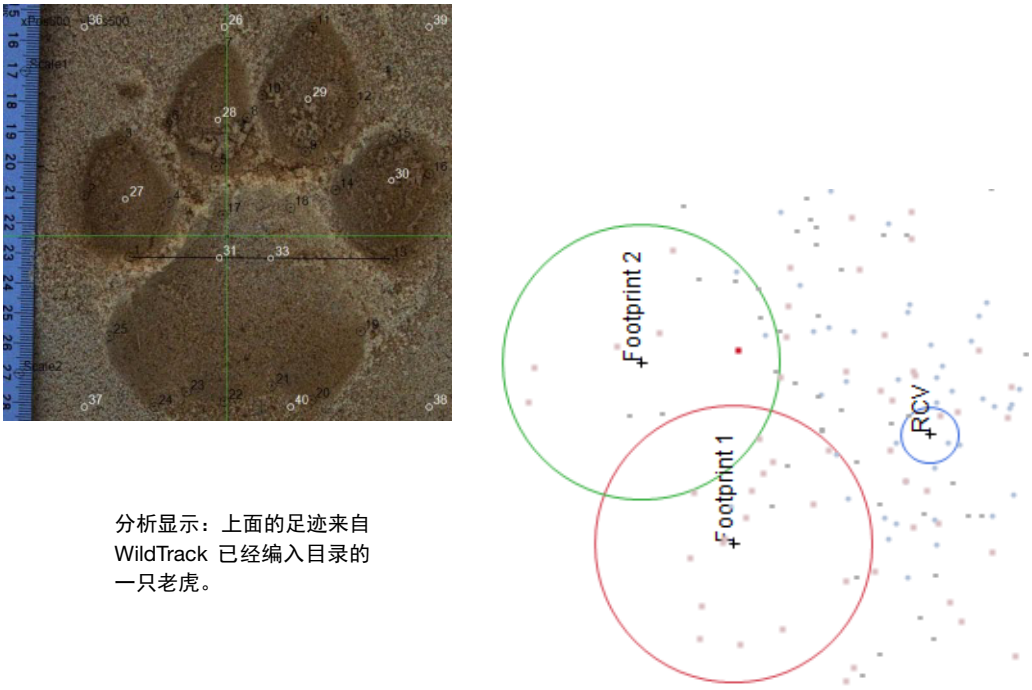
- 抓手工具 ( ) 支持您在地图中水平和垂直滚动。
- 放大镜工具 ( ) 支持您缩放地图。

有关添加地图的详细信息，请参见《基本绘图》。

从图像中提取数据

JMP 支持将图像中的数据提取到数据表中，而后分析该信息。WildTrack.org 的研究人员在 JMP 中分析足迹的数字照片，以便跟踪濒危物种。他们将足迹图像拖放到 JMP 报表中，并通过绘制数据点来捕获足迹的大小和形状。JMP 脚本语言 (JSL) 脚本将这些测量值提取到数据表中。此时，研究人员可以分析数据并确定足迹是否来源于新的动物。该方法可帮助他们跟踪地球上特定地区濒危物种的总量。

图 9.39 提取和分析数据的示例



分析显示：上面的足迹来自 WildTrack 已经编入目录的一只老虎。

由于用户的需求大有不同，关于如何执行该功能的详细信息已超出《使用 JMP》的范围。有关编写 JSL 脚本的详细信息，请参见《Scripting Guide》和《JSL Syntax Reference》。

向 JMP 报表和图形添加注解和形状

您可以使用工具栏中的图形工具向报表或图形添加文本注释、线条、多边形、椭圆和矩形。

向 JMP 报表或图形添加注解

您可以使用注解工具 向 JMP 报表或图形添加可编辑的文本注释。要添加注解，请执行以下步骤：

1. 从工具菜单或工具栏选择注解工具 。
2. 在报表或图形中，点击要添加注解的位置。或者，点击并拖动以调整注解的大小。随即显示可编辑的白色文本框。
3. 输入文本。
4. 在注解外点击。
5. （可选）要添加标签线，请右击注解并选择**标签线**。

   您还可以编辑以下任意选项：

   背景颜色 选择注解的背景颜色。

   文本颜色 选择文本颜色。文本颜色还定义注解外框的颜色。

   字体 更改字体类型、样式和大小。要更改默认字体，请参见“JMP 首选项”一章中第 612 页的““字体”首选项”。

   标签线 在注解和注解定位到的点或文本之间绘制一条线。

   填充 去除注解的背景颜色，让它看起来是透明的。使用透明注释可以方便地将标题和脚注放到图形上。

   可编辑 通过双击使注解可编辑。

   固定大小 避免在向注解框添加文本时调整注解框的大小。

   重新定位 （仅当移动注解时才可用。）注解被定位到报表中的某个位置，但可以四处移动而不受定位位置的影响。注解的行为取决于定位位置。例如，定位在图形中的注解随轴移动，而定位在分级显示项中的注解在打开或关闭分级显示项时随之显示或隐藏。若要重新定位注解，请禁用标签线，将该注解移至新位置并右击将其重新定位。

   删除 删除整个注解。

   注意：若要添加多个注解，请在首次选择注解工具时按 Shift 键。这将导致后续的点击操作都用来添加注解，而您不必在每次添加注解之前都从工具栏中选择注解工具。

一旦添加了注解，即可执行以下操作：

表 9.3 使用注解

操作	说明
添加至或编辑注解	在文本框中点击。
移动注解	在注解框内点击并拖动注解。移动注解时，注解将变为选定状态，此时会通过四周带控点的双蓝线来指示。
调整注解大小	将光标置于选定注释的控点上（显示在边缘的中间位置和各角处）。光标显示为单个十字箭头；拖动它可调整注解的大小。
删除注解	通过点击注释控点突出显示注解，然后按 Delete 键或 Backspace 键。
移动注解的标签线	1. 右击注解并取消选择**标签线**。 2. 将注解拖至想要标签线开始的位置 （附着点）。 3. 右击注解并选择**标签线**。 标签线现在从您将注解移至的位置开始。 4. 将注解移至您希望其出现的位置。
令标签线随注解移动	按 Ctrl 键并拖动注解，使标签线端点随注解一起移动。

向 JMP 报表添加形状

您可以使用绘图工具 向 JMP 报表添加可编辑的线条、多边形和简单形状（椭圆或矩形）。以下各节介绍如何使用这些工具。

注意：每种图形工具都能记住最近所选的选项。若您需要具有相同特征的许多注解或其他图形，该功能非常适用。例如，假定您需要许多一端带箭头的粗绿线。按您想要的方式创建一条线，设置选项，后续线条都将按照这些选项显示。这些选项会一直保持到您对其进行更改为止。

添加线条

向报表窗口添加线条注解：

1. 点击工具选项板中的线条工具 。
2. 在您要插入线条的位置点击并拖动。该线条显示为选定状态，两端都带有控点。
3. 点击并拖动线条可移动它。
4. 点击控点并拖动可旋转线条。
5. 右击线条可显示用于定制线条外观的选项：

   箭头指向和箭头起始 将箭头置于线条的任一端（或两端）。

粗线 将线条的粗细更改为粗。线条默认为细。

虚线 将线条更改为虚线。线条默认为实线。

颜色 显示 JMP 调色板以更改形状的颜色。

重新定位 线条注解被定位到报表中的某个位置，但可以四处移动而不受定位位置的影响。该行为取决于定位位置。例如，定位在图形中的线条注解随轴移动，而定位在分级显示项中的线条注解在打开或关闭分级显示项时随之显示或隐藏。若要重新定位线条注解，请将该线条移至新位置并右击将其重新定位。仅当注解位于图形外部时可用。

删除 从报表上面删除形状。也可以通过选择形状并按 Delete 键或 Backspace 键来删除形状。

添加多边形或 Bezier 曲线

向报表窗口添加多边形 （或 Bezier 曲线）注解：

1. 点击工具选项板中的多边形工具 。
2. 点击可为多边形的第一条边创建起始点。
3. 在要结束第一条边并开始邻边的位置再次点击。一个带控点的方形选择框显示在多边形区域周围。
4. 第三次点击即可完成第二条边。
5. 继续执行该过程，直到如您所需创建了多边形。每次完成某一边，选择框就会随之调整，以包含多边形各边。
6. 双击可释放多边形工具。

一旦添加了多边形，即可执行以下操作：

表 9.4 使用多边形

操作	说明
选择或取消选择多边形	点击完成的多边形的边缘。
调整多边形的大小	选择它并拖动选择框的一个控点。
移动多边形	在框的控点之间点击并拖动选择框。
更改多边形的边数	点击并拖动这些边以形成新形状。

右击多边形可显示用于定制其外观的选项：

填充 填充或清空形状区域。

浮雕 在形状周围显示粗着色线。若形状已填充，图的下边缘则显示为凸起，呈现三维外观。

平滑 平滑多边形的顶点，以生成 Bezier 曲线。经过平滑处理的图按照与多边形相同的方式重新调整形状和大小，并且可以填充并呈现浮雕效果。

闭合 打开或关闭多边形的最后一条线段。

颜色 显示 JMP 调色板以更改形状各边的颜色，并且在应用**填充**选项时更改其填充色。

重新定位 多边形注解被定位到报表中的某个位置，但可以四处移动而不受定位位置的影响。该行为取决于定位位置。例如，定位在图形中的多边形注解随轴移动，而定位在分级显示项中的多边形注解在打开或关闭分级显示项时随之显示或隐藏。若要重新定位多边形注解，请将该多边形移至新位置并右击将其重新定位。

删除 从报表中删除形状。也可以通过选择形状并按 Delete 键或 Backspace 键来删除形状。

添加椭圆或矩形

向报表窗口添加椭圆或矩形注解：

1. 点击工具选项板中的简单形状工具 。
2. 在您要插入形状的位置点击并拖动。随即显示一个由选择框包围的椭圆。
3. （可选）要将椭圆变为矩形，请右击该椭圆并选择**形状 > 矩形**。
4. 双击可释放简单形状工具。

一旦添加了形状，即可执行以下操作：

- 选择形状并拖动选择框的某一控点可重新调整形状或大小。
- 点击并拖动选择框的边缘（在控点之间）可移动选择框。
- 选择形状，然后右击该形状可显示用来定制其外观的选项：

  填充 填充或清空形状区域。

  浮雕 在形状周围显示粗着色线。若形状已填充，图的下边缘则显示为凸起，呈现三维外观。

  形状 显示一个子菜单，选择其中的选项可将形状变换为椭圆或矩形。

  颜色 显示 JMP 调色板以更改形状各边的颜色，并且在应用填充选项时更改其填充色。

  重新定位 形状注解被定位到报表中的某个位置，但可以四处移动而不受定位位置的影响。该行为取决于定位位置。例如，定位在图形中的形状注解随轴移动，而定位在分级显示项中的形状注解在打开或关闭分级显示项时随之显示或隐藏。若要重新定位形状注解，请将该形状移至新位置并右击将其重新定位。

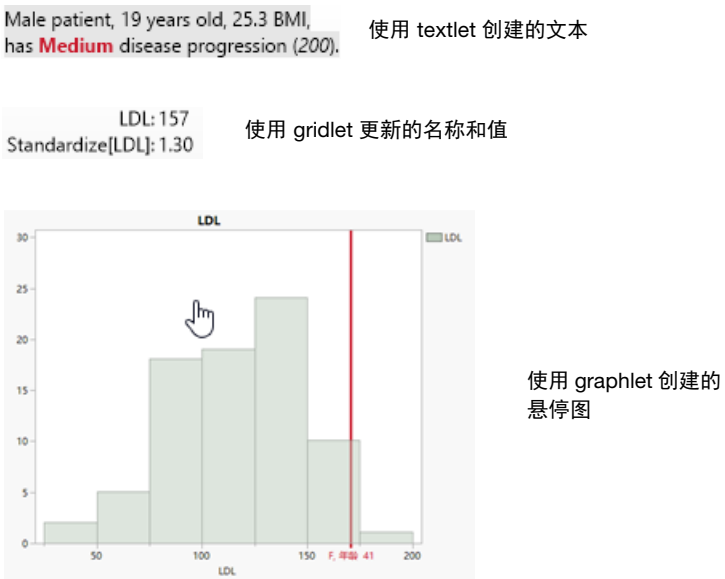
  删除 从报表中删除形状。也可以通过选择形状并按 Delete 键或 Backspace 键来删除形状。

定制图形中的悬停标签

当您悬停在图形中的可视元素上时，通常会在悬停标签中看到数据点的基本描述。通过以下任意方式定制显示在悬停标签中的内容：

使用 graphlet 添加交互式图形或图像。	请参见第 455 页的“向悬停标签添加图形或图像”。
使用 textlet 添加格式化文本。	请参见第 469 页的“向图形中的悬停标签添加富文本”
使用 gridlet 添加、删除或更新名称和值。	请参见第 472 页的“更新悬停标签中的名称和值”。
了解有关使用“悬停标签编辑器”的更多信息。	请参见第 482 页的“编辑图形中的悬停标签”。
更新悬停标签中的名称和值，或向图形添加标签。	请参见第 487 页的“将数据表中的图像添加到悬停标签”。
从数据表添加图形。	请参见第 487 页的“将数据表中的图像添加到悬停标签”。
将悬停标签固定在图形上。	请参见第 488 页的“固定图形中的悬停标签”。
设置悬停标签的格式并访问其他选项。	请参见第 489 页的“悬停标签的右击选项”。

图 9.40 悬停标签中图形、文本、名称和值的示例



向悬停标签添加图形或图像

可以使用 graphlet 将图形或图像的缩略图添加到悬停标签中。

- 预设 graphlet：预设 graphlet 可预先定义和编辑。它们很容易添加，只需右击图形并选择**悬停标签 > 图形类型**，也可以从“悬停标签编辑器”添加它们。

  注意：基于未连接到数据表的合成数据 （例如模拟中使用的数据）的图形不支持预设图。

- 粘贴 graphlet：在“图形生成器”中创建图形。通过将脚本保存到剪贴板来复制脚本，然后将其作为 graphlet 粘贴到基线图中。

添加悬停图后，将光标放在基线图中的某个元素上。随即为该元素显示悬停图的缩略图预览。例如，对于聚合数据，您可以看到底层数据分布的悬停图。

一些 JMP 平台具有内置的悬停标签图，如主成分、过程筛选、模型驱动的多元控制图和函数数据分析器。要将这些图形添加到可视化视图中，您不需要做任何事情。当您悬停在其上方时，图形就会自动显示在悬停标签中。

您还可以使用 JSL 定制悬停标签图。有关详细信息，请参见《Scripting Guide》。

悬停图操作

- 点击悬停图可在新窗口中启动它，或点击某个图像打开相关链接。
- 按 Ctrl 键并点击悬停标签图可将基线图替换为悬停标签图。
- 按 Alt 键并点击悬停标签图可替换上次启动的窗口的内容，而不是打开新窗口。

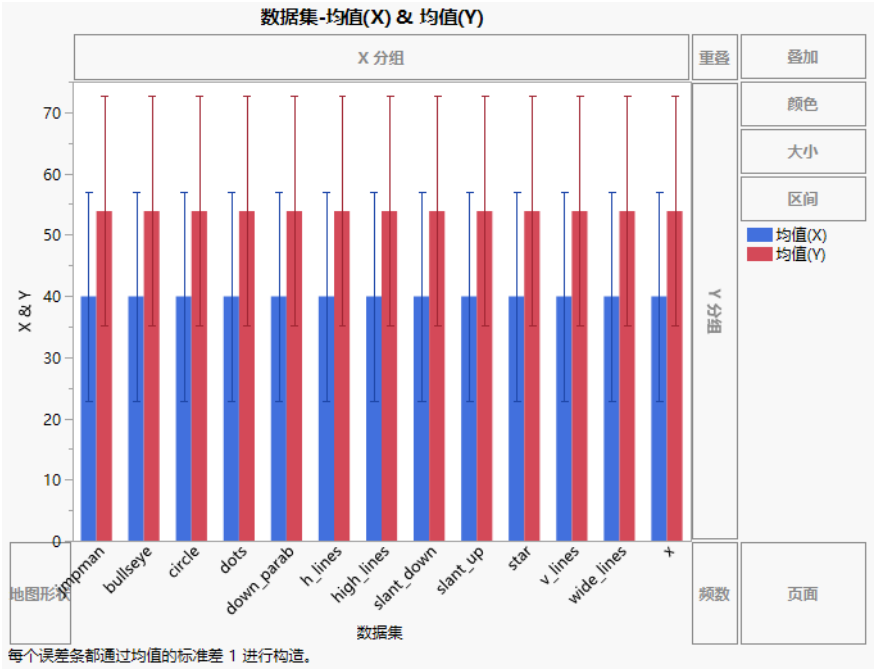
示例：向悬停标签添加预设图

在本例中，创建一个图，将预设图添加到悬停标签，然后定制悬停标签图。

创建原始图形

1. 选择**帮助** > **样本数据库**，然后打开 JMP Man Dozen.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择 X 和 Y 并将其拖至 Y 区域。
4. 选择**数据集**并将其拖至 X 区域。
5. 点击 “条形图” 元素 。
6. 对于 “误差区间”，选择**标准差**。

图 9.41 JMP Man Dozen.jmp 的原始图形



您可以看到这些观测的均值和标准差是相同的，所以该图可提供的信息不多。使用预设图可以更深入地了解数据。

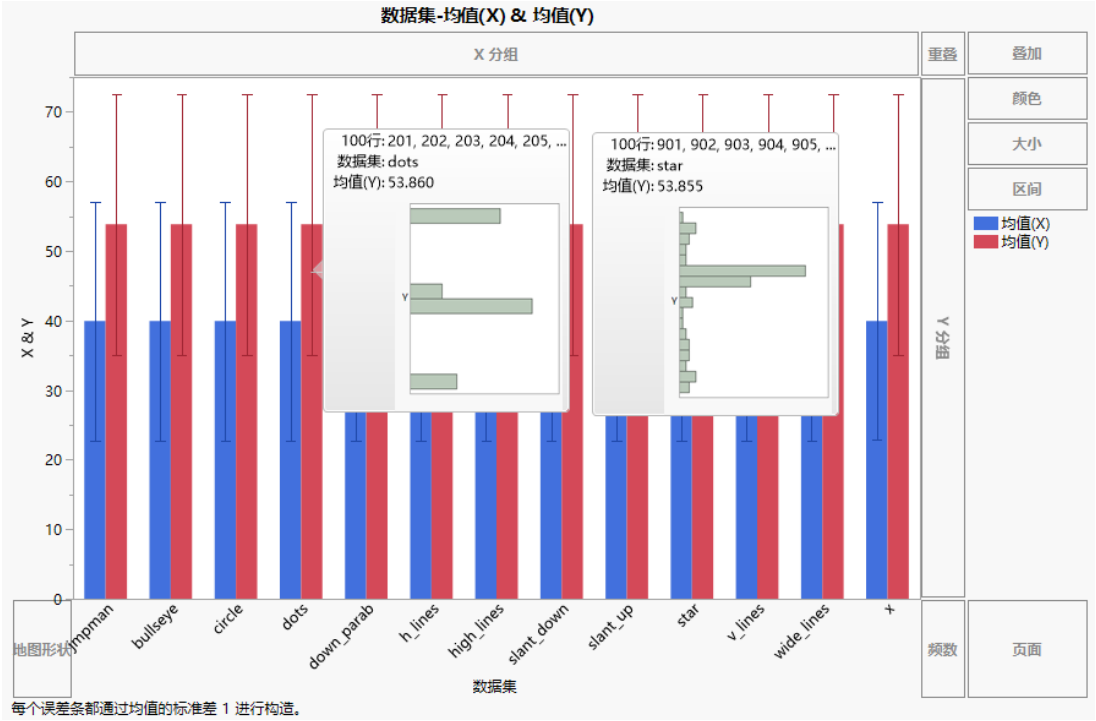
添加预设图

1. 在图形中，右击并选择**悬停标签** > **直方图**。

   这将向悬停标签添加预设直方图。
2. 指向某些直条。在本例中，指向**点**和**星形**对应的红条。

提示：要使悬停标签保持不变以便与其他悬停标签进行比较，请点击右上角的固定图标 。请参见第 488 页的 “固定图形中的悬停标签”。

图 9.42 示例：预设直方图



您可以看到测量值的分布明显不同。但您想查看另一种图形，散点图。

3. 在图形中，右击并选择**悬停标签** > **点**。

   这会将预设图从直方图改为散点图。

4. 悬停在 down_parab 对应的蓝条和 down_parab 对应的红条上方。请注意，轴上的 Y 和 X 变量是相反的。

   您希望对所有图形使用相同的 X 和 Y 角色分配 （将 X 变量分配给 X 轴， Y 变量分配给 Y 轴）。您还需要将平滑线更改为拟合线，并将带有摘要统计信息的标题框添加到散点图。

   您可以在“图形生成器”中定制图形，然后将该图形复制为模板并将其应用于该图形中的所有悬停标签。

定制图形

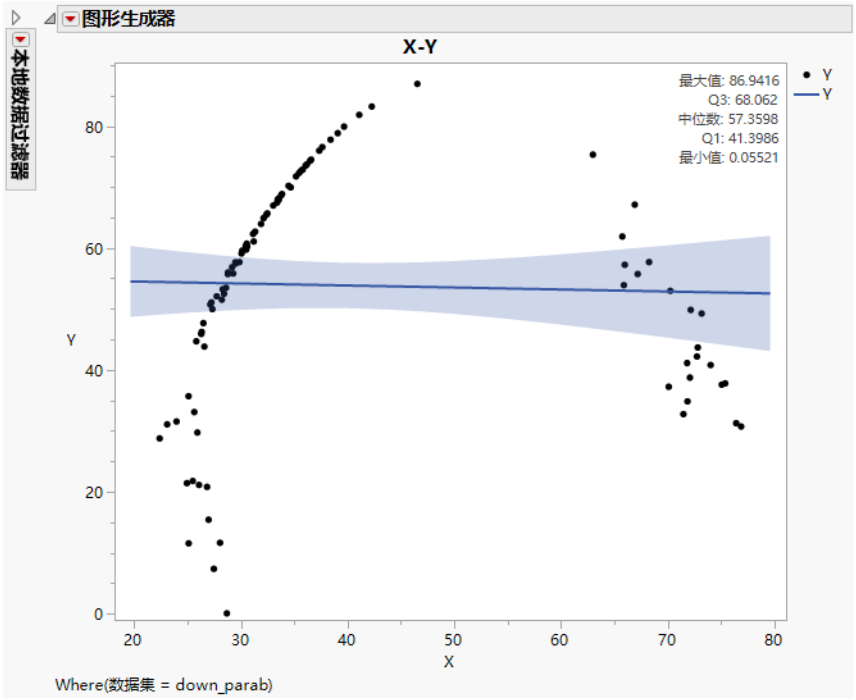
1. 悬停在 down_parab 对应的蓝条上方，并点击该图形以便在新窗口中打开它。

   该图形在 “图形生成器”中打开。

注意：请注意，启动一个预设图时，您会在左侧看到一个 “本地数据过滤器”。您可以使用 “本地数据过滤器”展开并修改该图形。详细信息，请参见第 391 页的 “报表中的 “本地数据过滤器” ”。

2. 点击 “图形生成器”红色小三角并选择**显示控制面板**。
3. 点击 “拟合线”元素 。
4. 点击标题框图标 。
5. 在 “标题框”下，将 “汇总统计量”改为 **5 数汇总**。
6. 点击**完成**。

图 9.43 down_parab 的定制图形



7. 点击 “图形生成器”红色小三角菜单并选择**保存脚本 > 至剪贴板**。
8. 关闭该 “图形生成器”窗口。
9. 在原始图形中，右击并选择**悬停标签 > 粘贴 Graphlet**。

   这会将定制图形作为模板应用到所有悬停标签图。悬停在一些直条上方以查看更改。

提示：要更好地比较原始图形和从悬停标签启动的图形，请使用项目。请参见第 10 章，“保存和共享您的数据”。

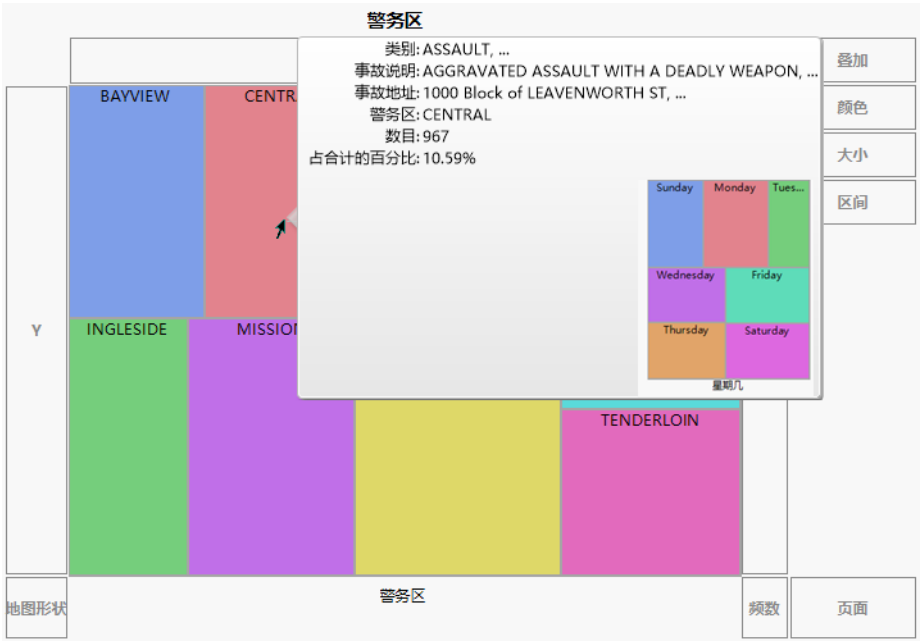
示例：在悬停标签中指定下钻列层次

在使用分类列的预设图中，您可以指定过滤要依据的列。

创建原始图形

1. 选择**帮助** > **样本数据库**，然后打开 San Francisco Crime.jmp。
   这些数据显示旧金山警务区的犯罪统计数据。
2. 选择**图形** > **图形生成器**。
3. 选择**警务区**并将其拖入 X 区域。
4. 点击“矩形树图”元素 。
   您可以看到旧金山不同警务区的矩形树图。
5. 右击该图形并选择**悬停标签** > **矩形树图**。
6. 将光标置于“中央”上方以查看悬停标签图。

图 9.44 “中央”对应的初始悬停标签图



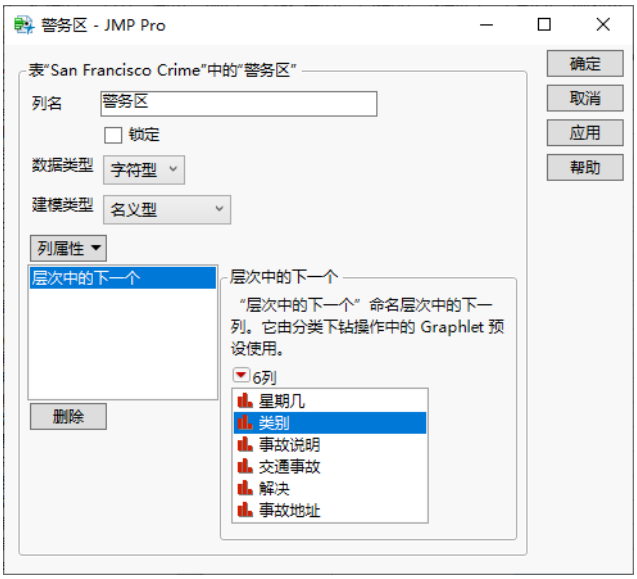
悬停标签图显示了每个警务区各周中星期几所发生的事件数。而在本例中，您想要按类别查看每个警务区发生的事件数。

指定下钻列层次

1. 在 San Francisco Crime.jmp 数据表中，右击**警务区**列标题并选择**列属性** > **层次中的下一个**。

2. 在“层次中的下一个”下方，点击**类别**。

图 9.45 为“层次中的下一个”选定的类别



3. 点击**确定**。
4. 添加在启动第一个悬停标签图时显示的第二级层次：

在 San Francisco Crime.jmp 数据表中，右击**类别**列。

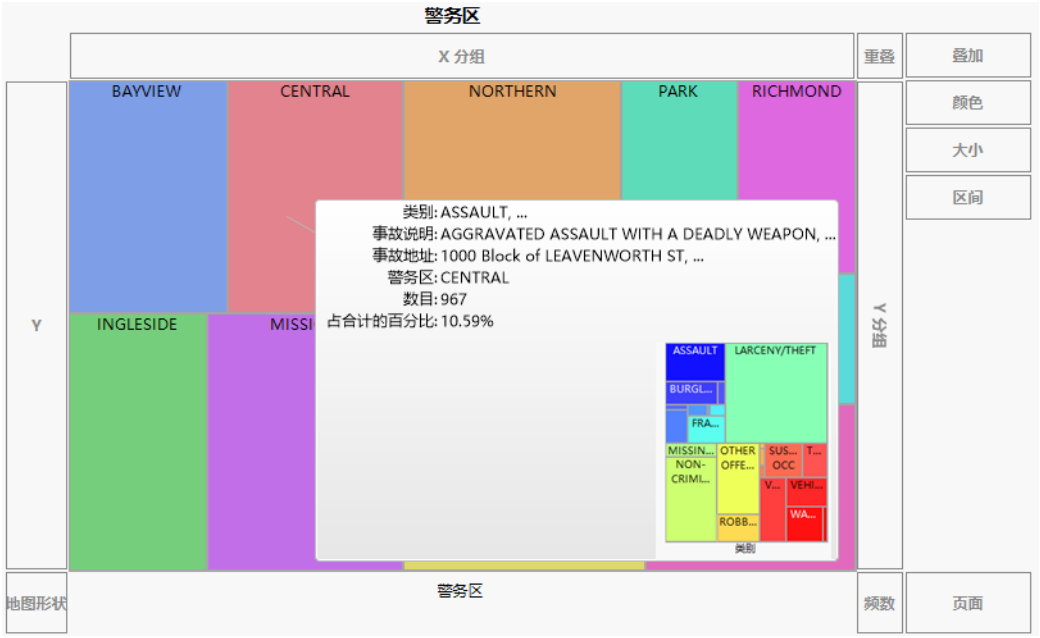
选择**列属性 > 层次中的下一个**。

在“层次中的下一个”下方，点击**事件说明**。

点击**确定**。

5. 在“图形生成器”中，再次将光标置于“中央”上方，以查看更新的悬停标签图。

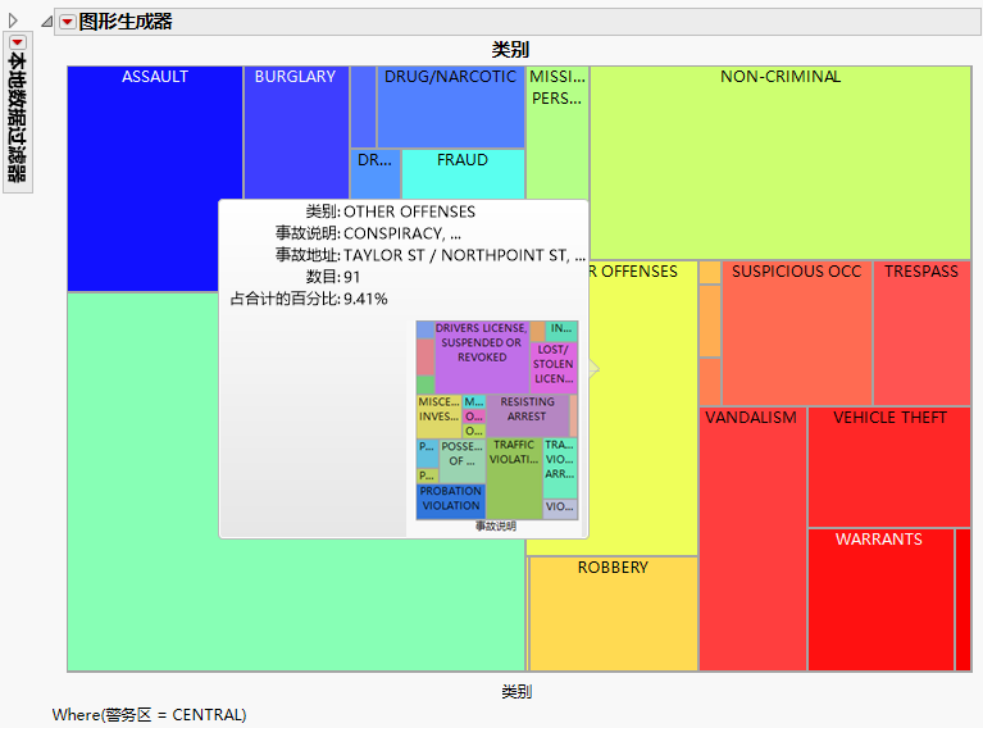
图 9.46 “中央”对应的更新的悬停标签图



悬停标签图显示了按类别划分的每个警务区发生的事件数。

6. 点击悬停标签图以启动它。
7. 将光标置于 “其他犯罪”上方。

图 9.47 “其他犯罪”类别的第二级悬停标签图



悬停标签图的第二级按事件描述显示“其他罪行”类别中的事件数。

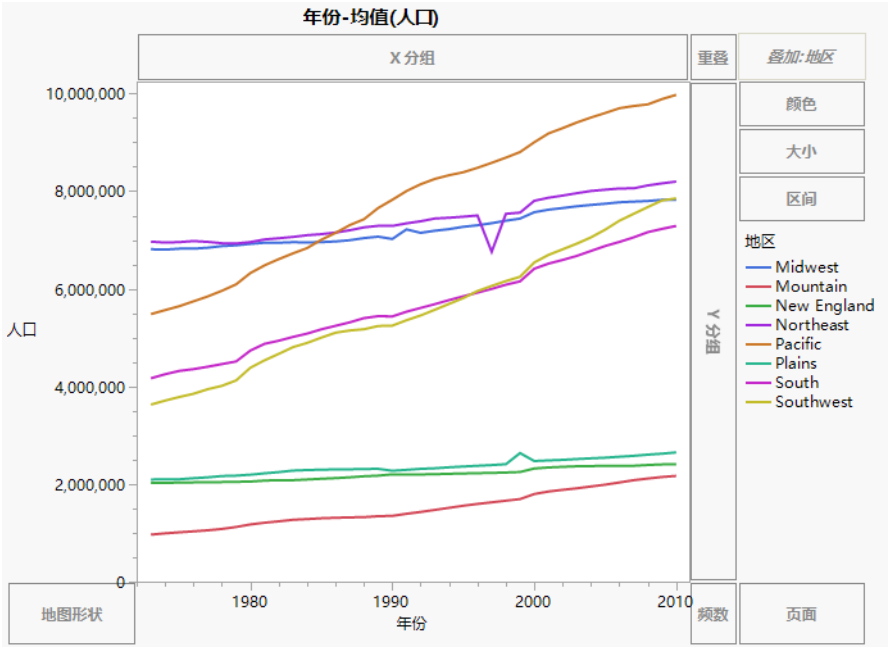
示例：从悬停标签图中删除过滤器列

在某些情况下，悬停标签图所过滤的列可能是要删除的列。例如，过滤器列可能太细化以致于没有用处。您可以使用“悬停标签编辑器”删除它。

创建原始图形

1. 选择**帮助** > **样本数据库**，然后打开 CrimeData.jmp。
   这些数据显示了美国各个地区和各州的犯罪率。
2. 选择**图形** > **图形生成器**。
3. 选择**人口**并将其拖至 Y 区域。
4. 选择**年份**并将其拖至 X 区域。
5. 选择**地区**并将其拖至**叠加**区域。
6. 点击 “线图”元素 。

图 9.48 美国各地区每年人口犯罪率均值



该图显示了不同年份受不同犯罪率影响的每个美国地区的人口均值。在悬停标签图中，您希望看到一个更加细化的视图，其中显示的是美国各州而不是地区的影响。

7. 请让该窗口保持打开状态。

复制和粘贴图形

1. 点击“图形生成器”红色小三角并选择**重新运行 > 重新运行分析**。

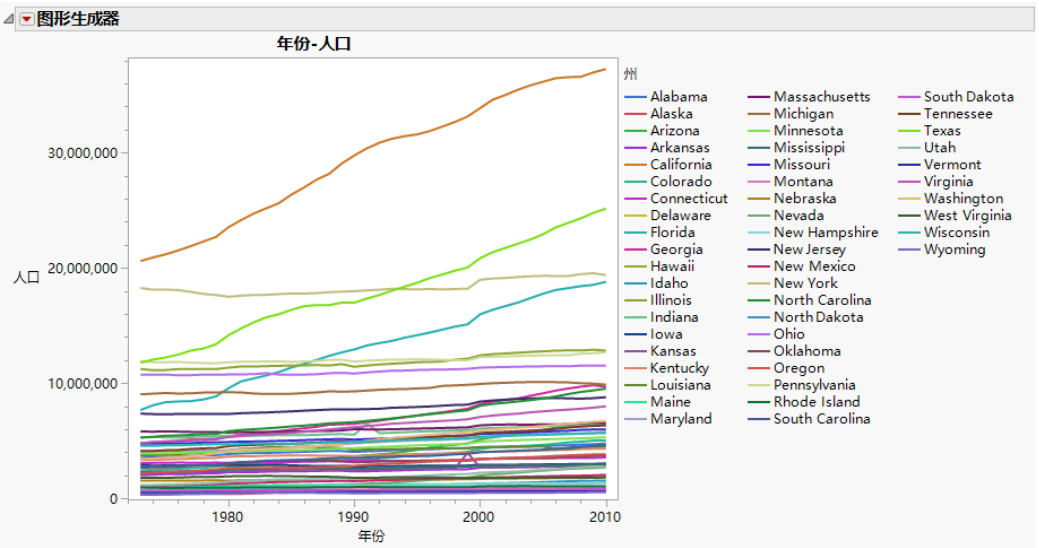
   随即显示原始图形的副本。

2. 在该图形的副本中，选择**州**并将其拖至**叠加**区域。

   地区将被**州**取代。

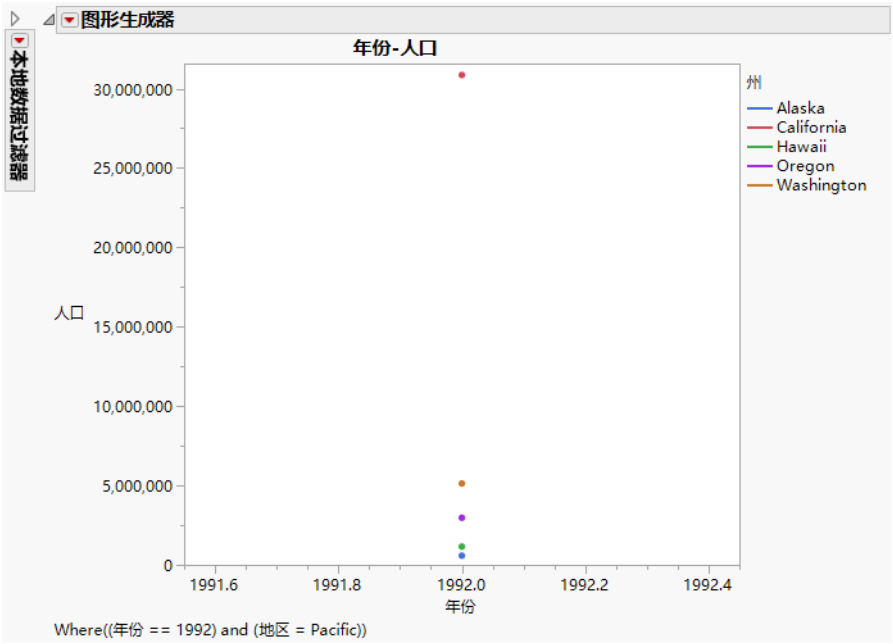
3. 点击**完成**。

图 9.49 美国各州每年人口犯罪率均值



4. 点击 “图形生成器”红色小三角并选择**保存脚本 > 至剪贴板**。
5. 关闭该图形。
6. 在原始图形中，右击并选择**悬停标签 > 粘贴 Graphlet**。
7. 悬停在 “Pacific”线条上方，然后点击以启动悬停标签图。

图 9.50 按州和年份显示太平洋地区犯罪率的悬停标签图



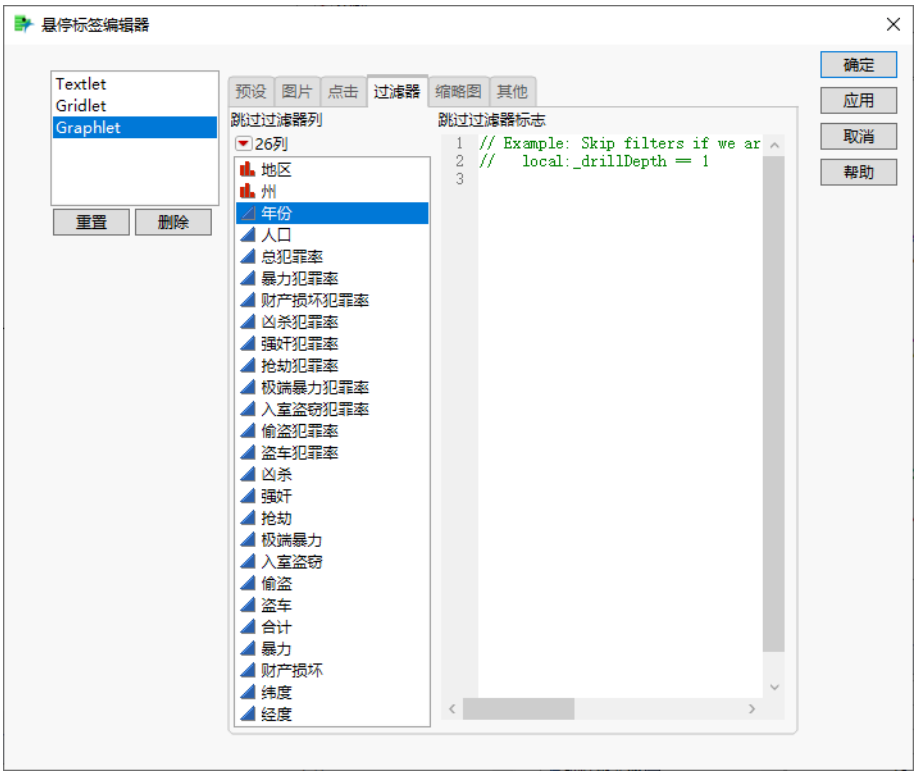
因为我们仍按年过滤，所以这个视图太细化，不太有用。您想要删除年份过滤器，以便图形只按地区过滤。

8. 关闭太平洋地区的悬停标签图。

删除过滤器列

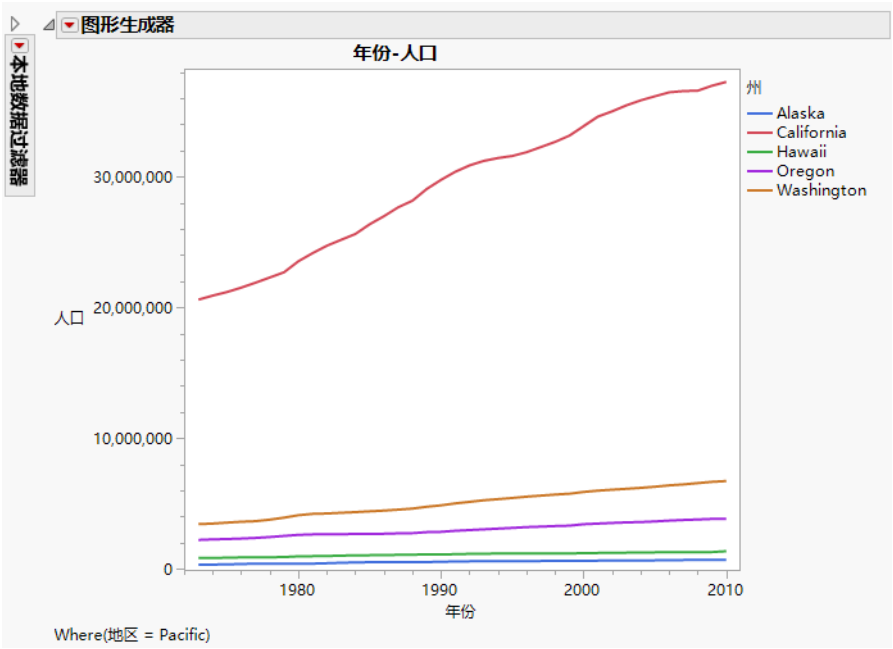
1. 右击原始图形并选择**悬停 标签** > **悬停标签编辑器**。
2. 在 Graphlet 窗口中，点击**过滤器**选项卡。
3. 选择**年份**，即您不想再依据其过滤的列。

图 9.51 忽略 “悬停标签编辑器”中的过滤器



4. 点击**确定**。
5. 悬停在 “Pacific”线条上方，然后点击以启动悬停标签图。

图 9.52 按州显示太平洋地区犯罪率的悬停标签图



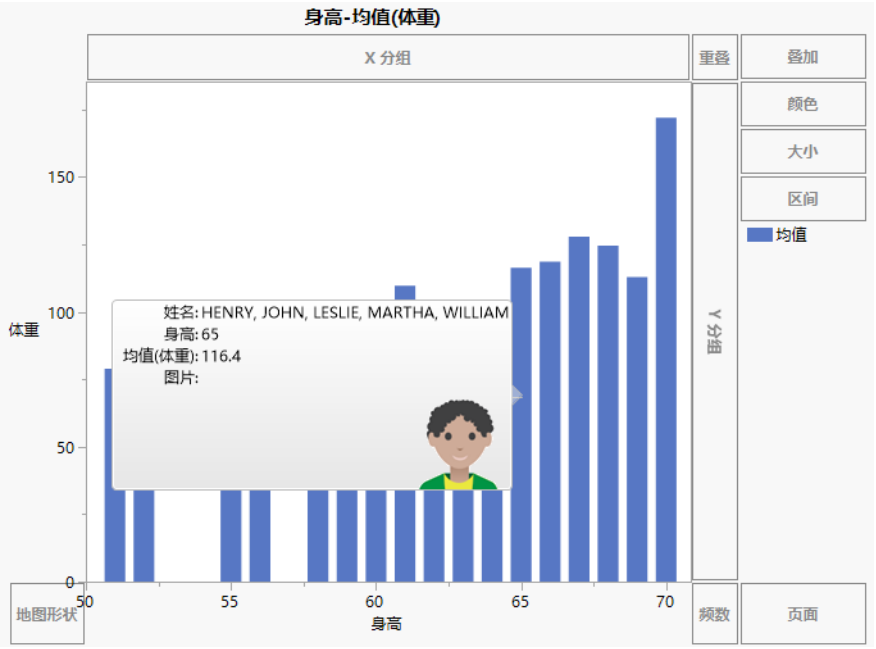
您可以看到，悬停标签图不再按年份过滤，而是仅按地区过滤，从而创建了更有用的可视化视图。

示例：在悬停标签中显示多个图像

对于带标签的列，悬停标签可以包含来自第一个带标签图像列的图像和来自带标签文本列的最多四个值。要显示更多带标签的图像或文本观测，请使用“标签查看器”。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class Families.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**体重**并将它拖至 Y 区域。
4. 选择**身高**并将它拖至 X 区域。
5. 点击“条形图”元素 。
6. 悬停在身高为 65 英寸的直条上并点击“固定” 。

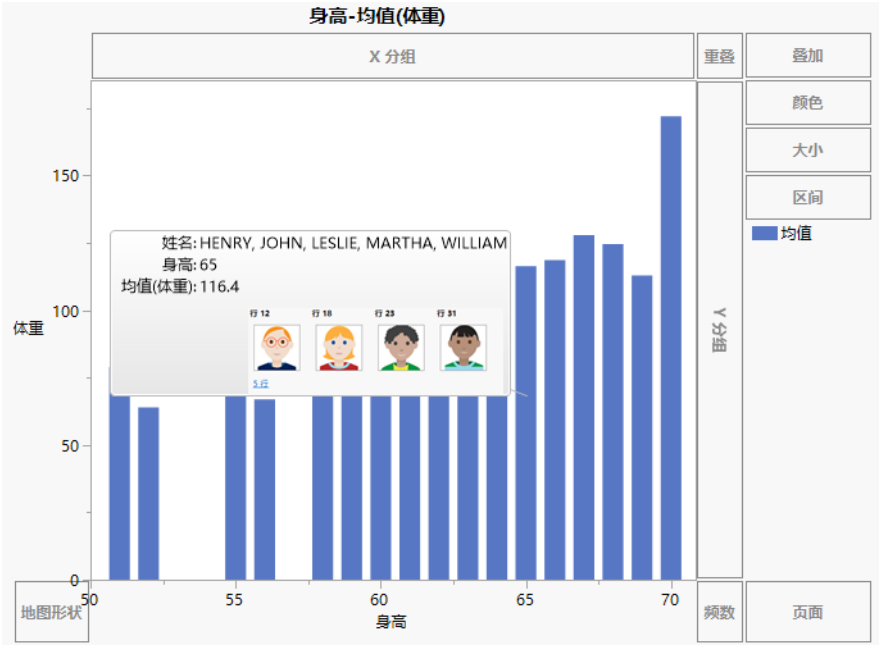
图 9.53 仅带有初始图像的悬停标签



尽管该直条中包含五个姓名，但只有第一个图像 (Henry) 显示在悬停标签中。您想在悬停标签中看到更多图像。

7. 在图中右击并选择**悬停标签 > 标签查看器**。

图 9.54 带有多个图像的悬停标签



现在最多有四个图像出现在悬停标签中。

8. 要查看其他图像，请在悬停标签中点击缩略图。
9. 在 “标签查看器” 中，点击前进箭头 以查看第五个图像。

向图形中的悬停标签添加富文本

textlet 是将格式化文本添加到悬停标签的悬停标签扩展。您可以使用 textlet 来描述解释性段落中的数据，或者通过加粗或用不同颜色绘制来强调重要信息。

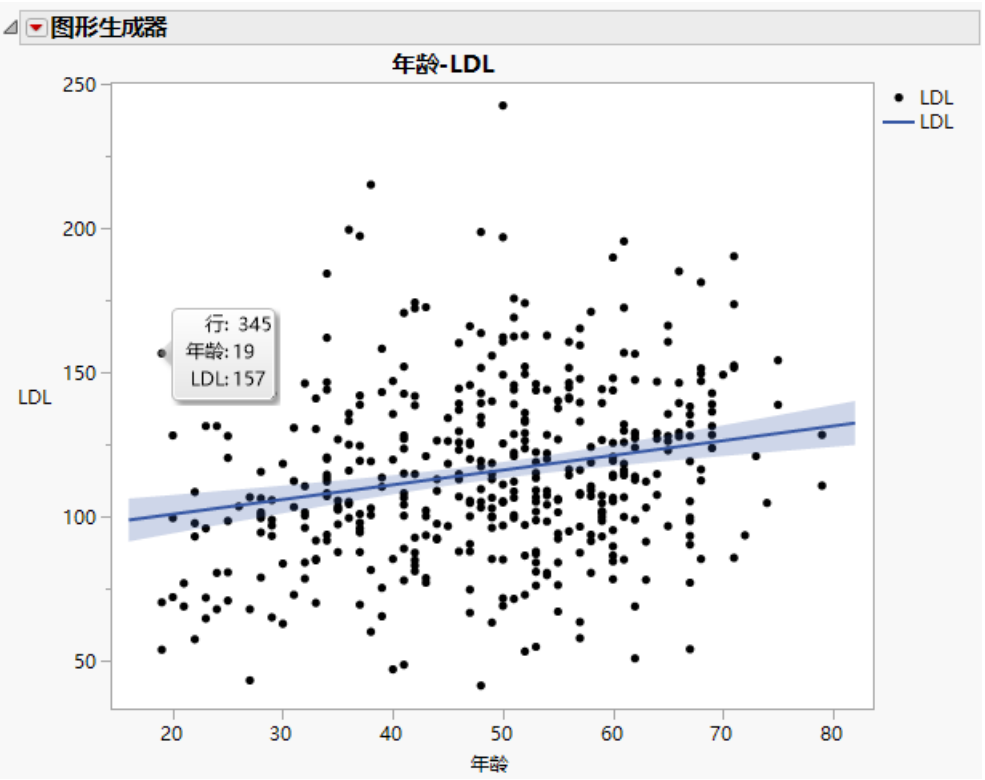
示例：向悬停标签添加文本

创建原始图形

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
   这些数据显示了跟踪疾病进展的糖尿病患者的测量结果。
2. 选择**图形** > **图形生成器**。
3. 选择 LDL 并将其拖至 Y 区域。
4. 选择**年龄**并将其拖至 X 区域。
5. 点击 “拟合线” 元素 。
6. 点击**完成**。

7. 悬停在第 345 行中的点上。

图 9.55 第 345 行内容的初始悬停标签



对于第 345 行中的数据点，您可以看到行号、人员的年龄以及他们的 LDL 测量值。更改悬停标签的显示内容：

- 在一句话中添加患者疾病进展的程度。
- 根据病情的严重程度，为疾病进展的级别分配不同的颜色。

添加文本

1. 将光标置于第 345 行的悬停标签上方，然后点击“固定” 📌。
   通过固定悬停标签，您可以在应用更改时查看更改。
2. 在图形中右击并点击**悬停标签 > 悬停标签编辑器**。
3. 在左侧点击 Textlet。
4. 在 “JSL 变量”框中，使用 JSL 定义变量。复制并粘贴以下 JSL 脚本：

```
// gets the value for the Y Ordinal column for the underlying visual element
local:YOrdinal = As Column( "Y Ordinal" )[local:_firstRow];
// maps the value to HTML color names that are used to style the text
```

```
local:color = Match( local:YOrdinal,
"Low", "Medium Dark Green",
"Medium", "Medium Dark Blue",
"High", "Medium Dark Red");
```

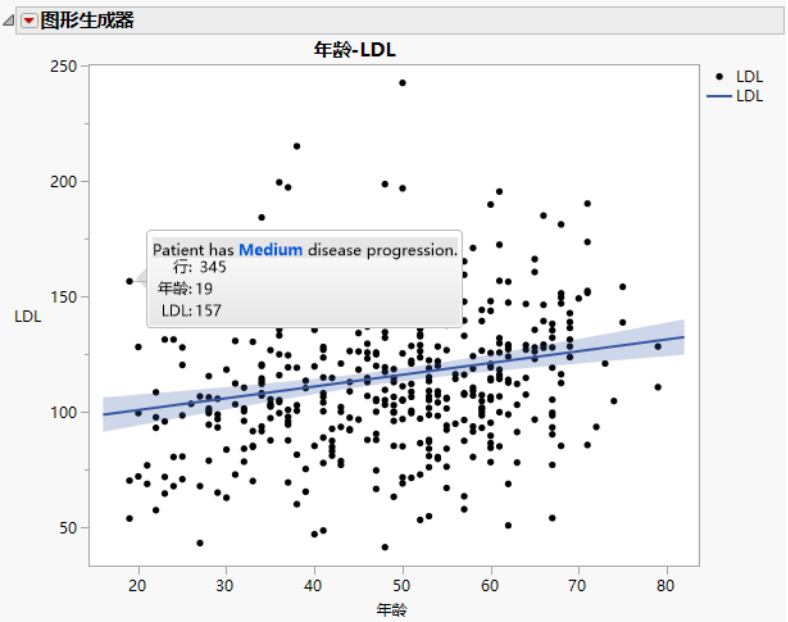
5. 在 HTML 标记下，使用 HTML 和您刚刚定义的 JSL 变量编写文本。复制并粘贴以下代码：

```
Patient has <font color='{local:color}'><b>{local:YOrdinal}</b></font>
  disease progression.
```

图 9.56 用于 Textlet 的 JSL 变量和 HTML 标记

6. 点击**确定**。

图 9.57 包含富文本的更新的悬停标签



您可以看到新句子出现在悬停标签上，表明疾病进展的程度。由于该患者的疾病进展为中度，其颜色编码为蓝色。

更新悬停标签中的名称和值

悬停在图形中的某个元素（如点、直条等）上方时，您可以看到该元素的悬停标签中的内容。默认情况下，悬停标签包含网格中的名称和值对。该值可以来自数据表，也可以是任意计算或 JSL 表达式。您可以使用 gridlet 定制这些名称或值。例如，以下是您可以执行的一些操作：

- 删除某个名称及其关联值
- 添加某个名称及其关联值
- 更改某个名称
- 更改名称样式（例如，粗体）或值的样式（例如，小数位数）

提示：定制悬停标签网格条目的另一种方法是使用标签。请参见第 487 页的“将数据表中的图像添加到悬停标签”。

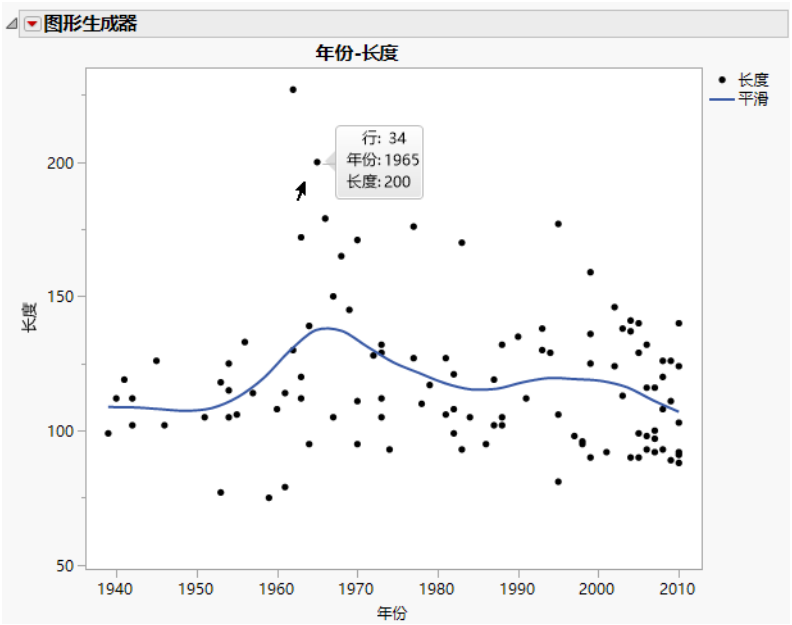
示例：更新悬停标签中的名称和值

在本例中，您将创建一个图形并使用 gridlet 更新悬停标签中的内容。

创建原始图形

1. 选择**帮助** > **样本数据库**，然后打开 Movie Inventory.jmp。
   这些数据显示了电影租赁数据。
2. 选择**图形** > **图形生成器**。
3. 选择**长度**并将其拖至 Y 区域。
4. 选择**年份**并将其拖至 X 区域。
5. 点击**完成**。
6. 悬停在第 34 行中的点上 （图 9.58）。

图 9.58 第 34 行的初始悬停标签内容



对于第 34 行中的数据点，您可以看到行号、制作电影的年份和电影的长度（以分钟为单位）。更改悬停标签的显示内容：

- 删除行号和长度条目。
- 添加电影名称和类型。
- 创建一个新条目，以小时和分钟为单位显示电影的长度。
- 将类型名称字体加粗。

更新名称和值

1. 悬停在第 34 行的悬停标签上方，然后点击“固定” 📌。

通过固定悬停标签，您可以在应用更改时查看更改。

2. 在悬停标签中右击并点击**编辑**。
3. 在缩略图中的左侧，点击**行**。
4. 在 “删除”选项卡中，点击**添加**。
   “行”即显示在要删除的条目下方。
5. 点击**应用**。
   在图形窗口中，您可以看到**行**已从悬停标签中删除。

图 9.59 已删除 “行”的悬停标签

6. 在缩略图的左侧，点击**长度** （您要删除的另一个条目）。
7. 在 “删除”选项卡中，点击**添加**。
   “长度”即显示在要删除的条目下方。
8. 点击**应用**。
   在图形窗口中，您可以看到**长度**已从悬停标签中删除，只留下**年**。
9. 添加电影名称和类型：

点击**追加**选项卡。类型应设置为**名称**。

在“目标”下，键入**名称**（您要添加的名称）。

在“值”下，复制和粘贴以下 JSL 表达式：`:Name[local:_firstRow]`

“名称”是列引用。括号指示获取给定索引的相应列值。`local:_firstRow` 是索引，它是一个悬停标签执行上下文 (HLEC) 变量。有关详细信息，请参见 《Scripting Guide》。

点击**添加**，然后点击**应用**。

在缩略图和图形悬停标签中，您可以看到已添加电影的名称。

图 9.60 已添加 “名称”的悬停标签

在“目标”下，键入**类型**（您要添加的另一个名称）。

在“值”下，复制和粘贴以下 JSL 表达式：`:Genre[local:_firstRow]`

点击**添加**，然后点击**应用**。

在缩略图和图形悬停标签中，您可以看到已添加电影的类型。

10. 将 “长度”数据从分钟转换为小时和分钟：

在**追加**选项卡上的“目标”框中，键入**持续时间**（您要添加的新悬停标签的名称）。

在“值”下，复制和粘贴以下 JSL 表达式：`local:_Length * 60`

local:_Length 是一个 HLEC 变量，变量值是网格条目 “长度”。“长度”包含在图形中，所以我们可以在悬停标签上下文中访问该值。有关详细信息，请参见《Scripting Guide》。

* 60 可将持续时间从分钟转换为秒。要应用“小时 : 分钟”格式，持续时间需以秒为单位。

点击**添加**，然后点击**应用**。

在缩略图和图形悬停标签中，您可以看到已添加 “持续时间”标签。但需要正确设置值的格式。

图 9.61 已添加 “持续时间”的悬停标签

点击**重新格式化**选项卡。

在“目标”中，键入**持续时间**（您要重新格式化的悬停标签条目的名称）。

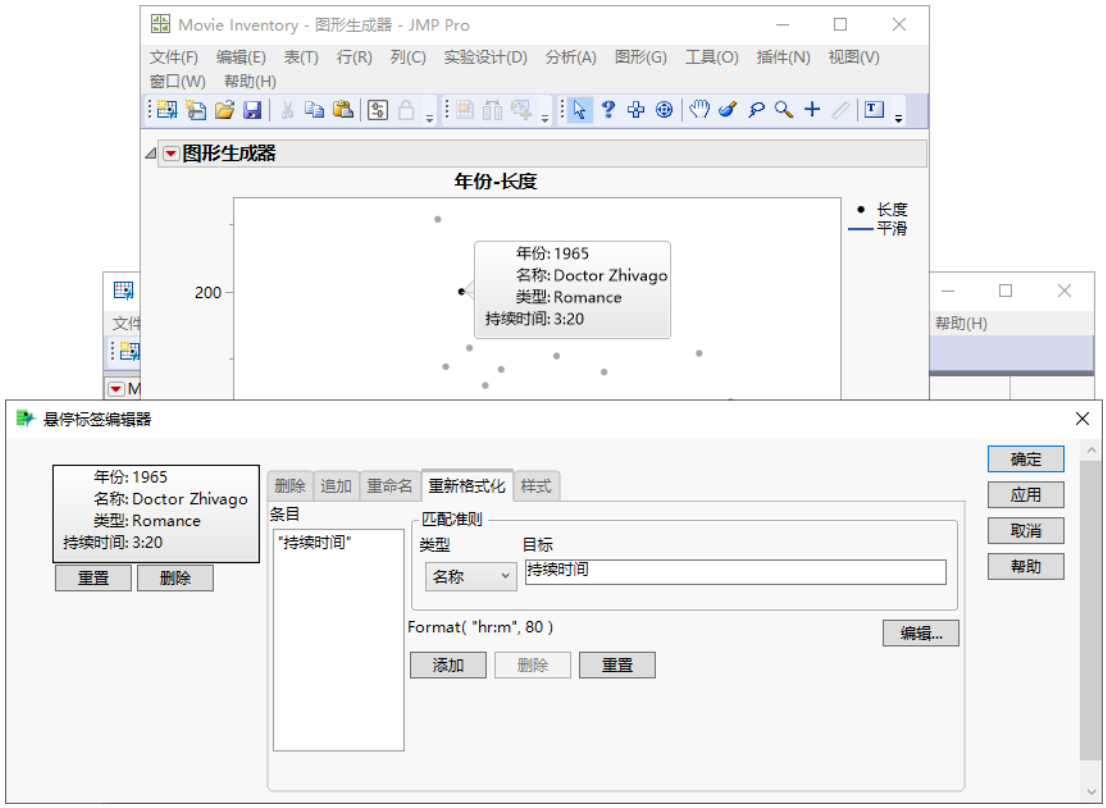
点击**编辑**。

在“格式”旁边，选择**持续时间** > hr:m，然后点击**确定**。

点击**添加**，然后点击**应用**。

在悬停标签上，您可以看到 “持续时间”现在以小时和分钟为单位。

图 9.62 “持续时间”以小时和分钟为单位的悬停标签



11. 将 “类型”值字体加粗：

点击**样式**选项卡。

在“目标”下，键入**类型**（您要加粗的悬停标签条目的名称）。

请注意，“范围”设置为**值**。这会将“类型”值字体加粗而不是将名称“类型”字体加粗。

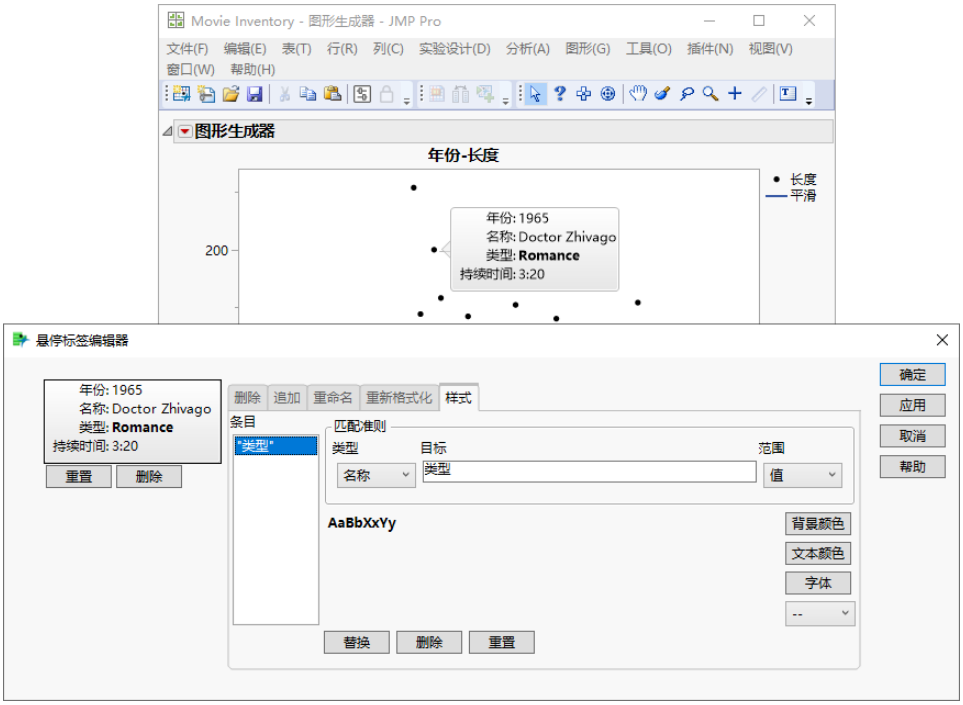
点击**字体**。

在“字体”样式下，点击**粗体**，然后点击**确定**。

点击**添加**，然后点击**应用**。

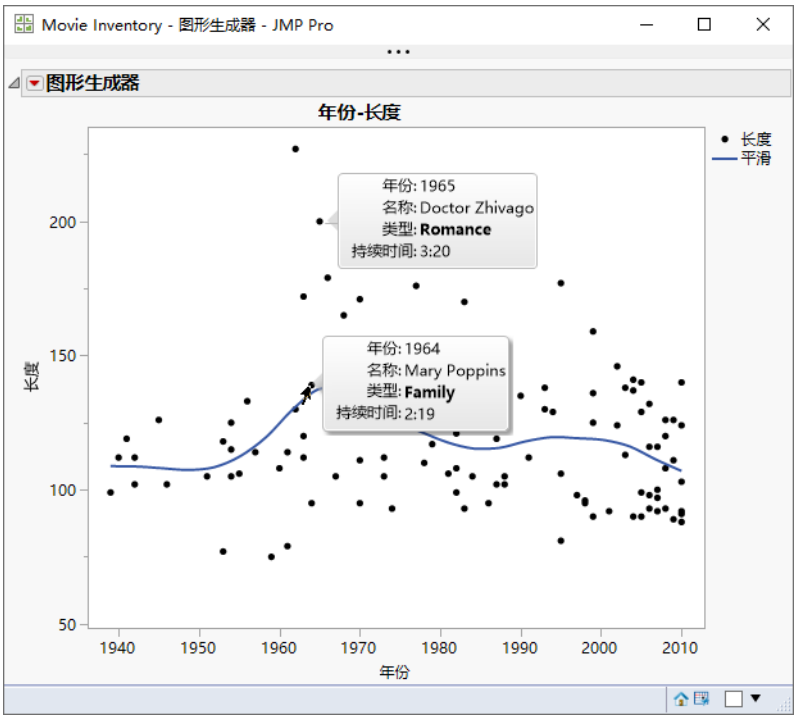
在悬停标签上，您可以看到 “类型”值 Romance 现在以粗体显示。

图 9.63 “类型”值以粗体显示的悬停标签



悬停在其他标签上方可看到它们均已更新。

图 9.64 更新的悬停标签



使用数据表中的标签

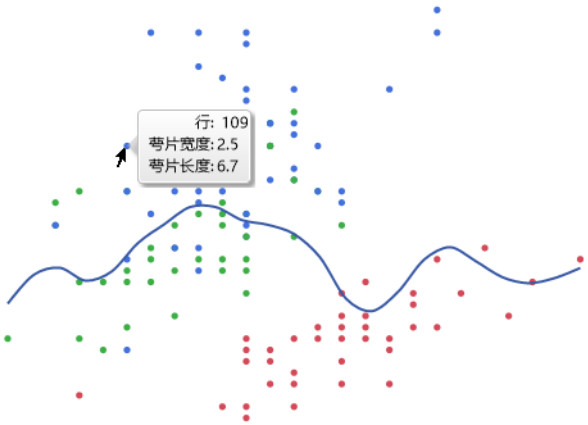
除了使用 gridlet，还可以通过向数据表中的列添加标签来对悬停标签执行一些基本定制。

- 若悬停标签包含“行”条目，则可以通过添加标签将其替换为列中的值。
- 一旦添加标签，即可将其保留在图形上。您也可以在图形上移动标签的位置。可以添加标签线以将标签连接到其对应的点。

示例：在数据表中使用标签

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**萼片长度**并将其拖至 Y 区域。
4. 选择**萼片宽度**并将其拖至 X 区域。
5. 点击**完成**。
6. 悬停在第 109 行中的点上。

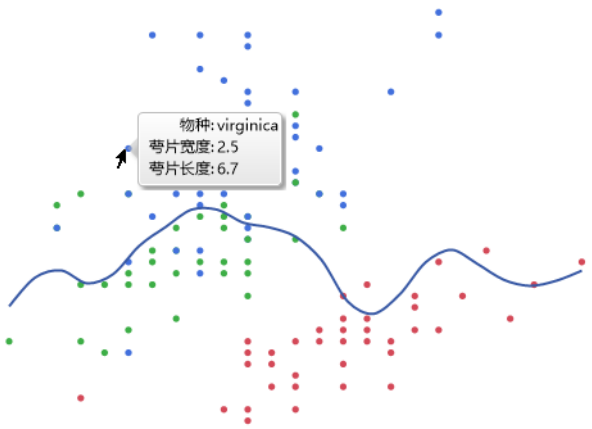
图 9.65 第 109 行的悬停标签



您可以看到悬停标签中出现一个“行”条目。您要将其替换为名为“物种”的数据表中的某列及其列值。

7. 在该数据表中，右击“物种”列并选择**添加标签 / 撤销标签**。
   列面板中的列名旁边将出现一个标签图标 。
8. 在“图形生成器”中，悬停在同一个数据点上方。

图 9.66 显示“物种”列的悬停标签



您可以看到“行”条目被“物种”列取代。

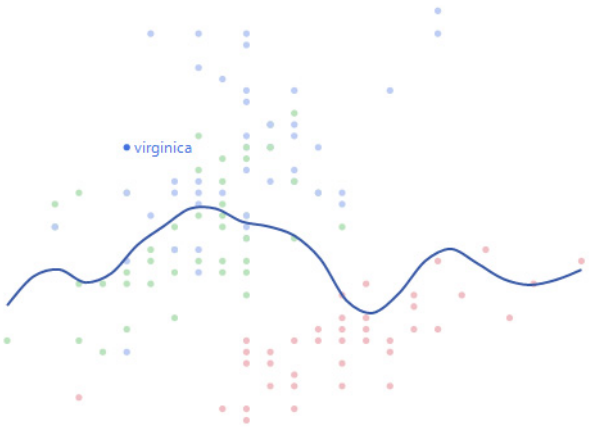
注意：若指定多个带标签的列，它们的值将按字母顺序显示在悬停标签中。

现在，您要让一些标签保留在图形上。

9. 点击同一个数据点使其突出显示。
10. 在任意位置右击并选择**行 > 行标签**。

注意：在直方图中，右击右侧的箱线图区域。

图 9.67 “物种”标签显示在图形上



您可以看到包含该数据点的 “物种”值的标签显示在图形上。

11. 点击并拖动标签使其移动位置。

将添加一条标签线，将该标签连接到该点。

图 9.68 重新放置 “物种” 标签

在数据表中，标签图标 显示在行号旁边。

数据会一直保留标签，直到您突出显示该列或行并再次选择**添加标签/撤销标签**。

编辑图形中的悬停标签

使用 “悬停标签编辑器” 在悬停标签中定制和添加图形、图像、格式化文本、名称和值。

以交互方式编辑悬停标签

1. 在图形中，悬停在数据上方以显示悬停标签。
2. 在悬停标签中，右击并选择**编辑**。

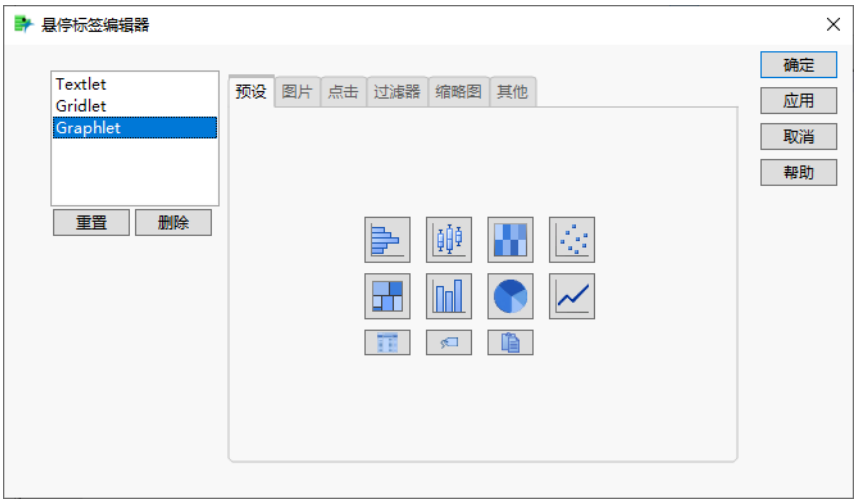
在 “悬停标签编辑器” 中，悬停标签的缩略图显示在左侧。点击缩略图中的某个区域可在右侧编辑它。所做的更改将应用于图形中的所有悬停标签。

图 9.69 以交互方式编辑悬停标签

在本例中，点击“事件地址”数据会显示“删除”选项卡，这样您就可以从图形中的所有悬停标签中删除“事件地址”条目。您还可以对其进行重命名、重新格式化或更改字体样式。

通常情况下，要添加或编辑悬停标签，请启动“悬停标签编辑器”。右击某个图形并选择**悬停标签 > 悬停标签编辑器**。

图 9.70 悬停标签编辑器



在左侧，您可以选择 Textlet、Gridlet 或 Graphlet，具体取决于您要在悬停标签中执行的操作：

Textlet 使用 JSL 将格式化文本添加到悬停标签。有关示例，请参见第 469 页的“示例：向悬停标签添加文本”。有关该窗口的详细信息，请参见《Scripting Guide》。

Gridlet 为悬停标签中显示的名称和值删除、添加、重命名、重新格式化或选择样式。相关示例，请参见第 472 页的“示例：更新悬停标签中的名称和值”。有关该窗口的详细信息，请参见第 485 页的“Gridlet 选项卡”。

Graphlet 向悬停标签添加交互式图形或图像。相关示例，请参见第 456 页的“示例：向悬停标签添加预设图”。有关该窗口的详细信息，请参见第 486 页的“Graphlet 选项卡”。

Textlet 选项卡

标记 定义用于在悬停标签上创建富文本的 HTML 标记。

预设 预先配置的 textlet 的示例。

- Showcase textlet 是一个演示如何基于所对应数据向 textlet 内容添加颜色和样式的示例。可参照第 469 页的“向图形中的悬停标签添加富文本”中的示例。
- Tabulate textlet 是一个演示如何将所对应数据的汇总统计信息添加到 textlet 内容的示例。
- Context textlet 是一个演示如何在 textlet 中显示求值上下文变量的示例（用于调试目的）。求值上下文变量定义执行 graphlet 和 textlet 脚本的环境。有关详细信息，请参见《Scripting Guide》。

JSL 变量 （可选）支持您将表达式的结果分配给本地 JSL 变量。这些变量在 HTML 标记字段中用于创建动态、数据驱动的富文本。

HTML 标记 基于 HTML 标记的子集指定富文本段落模板。标记可以包含用分隔符括起来的 JSL 变量引用。默认分隔符是大括号 ({})。表 9.5 说明了支持的 HTML 标记：

表 9.5 Textlet 中支持的 HTML 标记

HTML 标记	说明
	粗体文本
	斜体文本
<u> </u>	加下划线的文本
 	具有以下任何属性的样式化字体： • color=color：带引号的前景颜色值。可以是颜色名称或十六进制 RGB 颜色值。 • face=family：已安装字体系列的带引号名称。 • size=size：带引号的字体大小。
<background> </background>	背景颜色值 • color=color：带引号的颜色名称或十六进制 RGB 颜色值。

其他 指定悬停标签上文本的分隔符和宽度。

分隔符 选择分隔符，用于指示要在“标记”区域中替换的变量。例如，若脚本包含 C++ 代码或其他使用花括号的代码，则可以对 JMP 变量使用方括号分隔符。

宽度 指定悬停标签的首选宽度（以像素为单位）。

注意：对 textlet 配置进行更改后，可以点击“应用”查看固定的悬停标签上的更改。若您对更新的悬停标签满意，点击**确定**保存更改。

Gridlet 选项卡

Gridlet 选项卡包括“删除”、“追加”（添加）、“重命名”、“重新格式化”和“样式”。在所有这些选项卡中，指定“匹配准则”，该准则标识您要修改的悬停标签条目。

类型和目标

匹配准则 指定您要修改的悬停标签条目：

挑选要选择的条目类型和目标条目。

注意：并非所有选项都显示在所有选项卡中。

名称 输入要修改的条目的确切名称。例如，若条目名称为“均值 （身高）”，则输入该目标值：Mean(height)。

图 9.71 悬停标签示例

索引 在网格中输入要修改的条目的索引。例如，若悬停标签中的第三个条目是“均值 （身高）”，请输入 3 作为“目标”值。

列 输入要修改的条目的列名。例如，若条目为“均值 （身高）”，则输入该目标值：height。

Regex 输入与要修改的名称匹配的正则表达式。例如，若条目为“均值 （身高）”，则输入该目标值：^.*\(height\)$。（自动添加 Pat Regex、括号和引号。）

全部 更改所有悬停标签条目的样式。

Gridlet 按钮

添加 在“条目”下添加指定的“目标”值。点击**应用**将您的更改应用到悬停标签。

删除 删除“条目”下的指定“目标”值。点击**应用**将您的更改应用到悬停标签。

替换 替换“条目”下选定的“目标”值。点击某个条目，输入新目标值，然后点击**替换**。点击**应用**将您的更改应用到悬停标签。

Gridlet 选项卡

删除 指定要从中悬停标签中删除的条目。

注意：若使用的是 JSL，“删除”对应 Expunge 命令。

追加 指定要添加至悬停标签的条目。

值 指定将值添加至悬停标签所用的 JSL。相关示例，请参见第 473 页的“更新名称和值”。

点击 向条目添加一个操作，该操作将显示为一个链接。指定点击链接时要执行的脚本。例如，脚本可以创建可视化视图、打开网页或启动平台。以下是复制和粘贴脚本可使用的几种方法：

创建要链接的可视化视图：在“图形生成器”中，创建所需的可视化视图。点击“图形生成器”红色小三角菜单并选择**保存脚本 > 至剪贴板**。将脚本粘贴到“点击”下。

链接到预设图：在“图形生成器”中，从具有预设悬停标签图对应的图形中，右击 > **悬停标签 > 悬停标签编辑器**。在 Graphlet 下的“图片”选项卡中，复制脚本。将脚本粘贴到“点击”下。

点击**应用**将您的更改应用到悬停标签。

注意：若使用的是 JSL，“追加”对应 Annex 命令。

重命名 指定要在悬停标签中重命名的条目。输入新名称以替换当前 “目标”值。

重新格式化 为悬停标签中的条目值指定其他格式。例如，可以更改日期或持续时间格式，或更改小数位数。点击**编辑**以选择其他格式。有关“编辑”选项的详细信息，请参见第 257 页的 “数值格式选项”。

样式 为悬停标签中的条目指定背景颜色、文本颜色或字体详细信息 （如粗体等）。

范围 指定是否应对条目名称、值或这两者进行样式更新。例如，若要对所有匹配准则应用样式更改，您可以轻松地将所有名称左对齐，或将所有值右对齐。

背景颜色 指定悬停标签中“目标”值、名称或这两者的背景颜色。

文本颜色 指定悬停标签中“目标”值、名称或这两者的文本颜色。

字体 为悬停标签中的“目标”值、名称或这两者指定字体详细信息，例如字体、字体样式和大小。

对齐 指定如何在悬停标签中对齐“目标”值、名称或这两者的文本。

Graphlet 选项卡

预设 点击预设图可将其添加至悬停标签。有关预设图的详细信息，请参见第 455 页的 “向悬停标签添加图形或图像”。

图片 添加 JSL 脚本可定义要添加到悬停标签的图形或图像的类型。

点击 添加当用户点击 “图片” 脚本定义的缩略图 （或悬停图）时执行的 JSL 脚本。若此处未定义任何内容，则会重新使用 “图片” 脚本，以便在新窗口中启动 “图片” 下定义的图像或图形。

过滤器 从悬停图中删除过滤列。点击要删除的列，或在 “跳过过滤器标志” 下添加 “1”。请参见第 462 页的 “示例：从悬停标签图中删除过滤器列”。

缩略图 指定图形缩略图的大小以及是否包含轴值。

其他 使用 JSL 指定以下信息：

标题 设置启动悬停图时创建的新窗口的标题。

重新应用 定义是否将当前悬停标签图定制设置应用于点击悬停图时启动的新窗口。

将数据表中的图像添加到悬停标签

可以使用 graphlet 将图像添加到悬停标签，但也可以从数据表中添加图像。

提示：有关在悬停标签中使用 graphlet 添加 JMP 图形或图像的详细信息，请参见第 454 页的 “定制图形中的悬停标签”。

1. 在您的数据表中创建新列。
2. 右击该列，选择**列信息**，然后选择**表达式**数据类型。
3. 点击**确定**。
4. 将您计算机中的图形拖入列中每个单元格。您还可以导入包含图像列的数据。
5. 右击数据表中的列，然后选择**添加标签 / 撤销标签**以应用标签。
6. 使用您的数据表创建图形 （例如，在 “图形生成器” 或 “分布” 中）。
7. 悬停在数据点或直方图直条上方。

   该图像将显示在标签上方。要查看示例，请运行 SAS Offices.jmp 样本数据表中的 “图形生成器” 脚本。

图 9.72 悬停标签上的图像

提示：

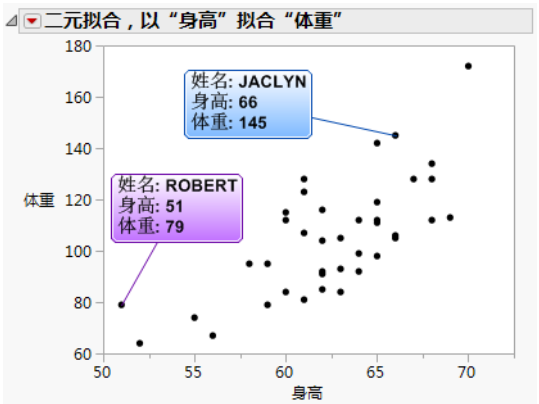
- 表达式列中的图像也可用作标记。请参见第 423 页的“将图像用作标记”。
- 您可以固定悬停标签以便该标签始终显示在图形上。请参见第 488 页的“固定图形中的悬停标签”。

固定图形中的悬停标签

当您悬停在图形中的数据上方时，将显示一个包含数据值的悬停标签。您可以固定悬停标签以便它始终作为标签显示在图形上。固定悬停标签在比较多个标签或要显示标签信息时非常有用。当您保存图形时，保存了该标签。在您打印图形时也会打印该标签。

注意：在您重新执行分析时，悬停标签保持固定状态。但是，若更改了数据，您可能得重新定位标签。

图 9.73 固定的悬停标签



要固定悬停标签，请悬停在数据点上方直到显示该标签。点击该标签右上角的固定图标。然后您可以拖动该标签将其重新定位。

悬停标签的右击选项

背景颜色 将选定颜色应用到标签背景。

框架颜色 将选定颜色应用到标签边框和标签线。

文本颜色 将选定颜色应用到文本。

填充 将背景颜色和框架颜色应用到标签。默认选定该选项。

使用渐变 在垂直方向从浅到深对背景颜色着色。默认选定该选项。

字体 将选定的字体系列应用到文本。默认字体由您的 JMP “字体”首选项决定。

若具有固定标签的脚本引用了计算机上不存在的字体，将不运行该脚本。

添加文本 将文本添加到标签底部。

替换文本 更改工具提示上的文本。若重新执行分析，工具提示文本保持更新。

标签线 在标签和数据点之间画一条线。将框架颜色应用到该线条。

复制 将悬停标签内容复制到剪贴板。

子集 打开包含与悬停标签关联的子集数据的数据表。

编辑 打开“悬停标签编辑器”，从中可以进一步编辑悬停标签的内容。请参见第 482 页的“编辑图形中的悬停标签”。

关闭 删除标签。您还可以通过点击标签右上角的 X 来删除标签。

注意：要返回到默认标签格式，请删除标签后重新添加它。

第 10 章

保存和共享您的数据

在 JMP 外部利用 JMP 数据

本章包含以下主题：

- 将数据表另存为文本文件、SAS 数据集、Excel 文件和其他格式
- 将报表另存为交互式 HTML 和 PowerPoint
- 通过电子邮件发送报表和数据表
- 将分析另存为脚本
- 将部分报表另存为图形
- 打印报表
- 复制和粘贴报表
- 创建 JMP 记录或项目
- 保存 JMP 会话
- 保存日志窗口

所选的方法取决于您将来想要如何与分析交互：

- 若您想以交互方式展示您的数据，则可以选择创建 JMP 记录。
- 若您想将数据表、链接报表和其他项发送给某人，则可以选择创建 JMP 项目。
- 若您想以后继续探索报表，则可以选择创建 JMP 报表。
- 若您想将报表发送给非 JMP 用户，可能要将报表转换为交互式 HTML 或 Web 报表。

目录

保存和共享数据表 494
将数据表另存为 Microsoft Excel 文件 494
将 JMP 数据另存为 CSV 文件 496
将 JMP 数据另存为文本文件 496
将 JMP 数据另存为 JSON 文件 497
将 JMP 数据另存为 SAS 传输文件 498
将 JMP 数据另存为 Google Sheet 499
将 JMP 数据另存为 SAS 数据集 500
将数据表保存至数据库 501
使用数据表替换数据库 502
保存和共享报表 502
将报表共享为交互式 HTML 504
将报表发布至 JMP Live 或 JMP Public 507
将报表另存为 PDF 文件 516
将报表另存为 PowerPoint 演示文稿 517
设置导出图形的图形 DPI 518
将 JMP 分析另存为脚本 518
以图形格式保存报表部分 519
打印报表 520
复制和粘贴报表 520
使用项目 521
创建新项目 522
在项目中打开文件 522
在项目中重新排列文件 522
保存项目 524
项目工作区 524
项目书签 525
项目内容 525
项目中最近使用的文件 526
项目日志 526
将文件移动到项目中 527
编写项目 On Open 脚本 527
创建新项目的示例 528
创建记录 530
进行演示制作记录的示例 531
新建记录 532
管理共享记录中的链接 533

向记录添加分级显示项并对其进行编辑 534
记录中的右击选项 535
控制记录中分级显示级别的显示 535
向记录添加图形或图像 536
定制记录中的项 536
防止修改记录 536
向记录追加报表 537
以其他格式保存记录 537
删除记录中的项 537
保存 JMP 会话 538
退出 JMP 时保存会话 538
手动保存会话 538
保存日志窗口 539

保存和共享数据表

选择**文件 > 导出**或**文件 > 另存为**以其他格式（如 Microsoft Excel 或 CSV）保存数据表。

注意：要将数据表保存为记录，请选择**编辑 > 记录**，然后选择**文件 > 另存为**。

JMP 采用表 10.1 中所列的格式保存数据表。

表 10.1 支持的用于保存数据表的文件类型

文件类型	可用的操作系统
JMP 数据表 (.jmp)	全部
Excel 工作簿 (.xls, .xlsx)	全部
文本导出文件 (.txt)	全部
逗号分隔的文件 (.csv)	Windows
制表符分隔的文件 (.tsv)	Windows
SAS 数据集 (.sas7bdat)	全部
具有扩展属性的 SAS 数据集 (.sas7bxat) **注意：**JMP 可以将列属性和表属性另存为 SAS 9.4 .sas7bxat 文件中的扩展属性。选择**文件 > 导出**，选择 SAS 文件类型，然后选中保存扩展属性所对应的复选框。JMP 使用 .sas7bxat 扩展名命名文件。“SAS 集成”首选项也包括该选项，它默认禁用。	Windows
SAS 传输文件 (.xpt,.stx)	全部
JSON 数据文件 (.json)	全部

您还可以保存具有相应 ODBC 驱动程序的类型的文件。请参见第 501 页的“将数据表保存至数据库”。

注意：

- 数据表名称的最大长度取决于用户计算机的操作系统。详细信息，请参见您的操作系统文档。
- 检测到崩溃时，JMP 会尽量保存所有修改的文件。

将数据表另存为 Microsoft Excel 文件

有两个将数据表另存为 Microsoft Excel 文件的选项：

- 以 Microsoft Excel 格式（.xls 或 .xlsx）保存一个数据表。单个表中的数据保存在以该数据表命名的工作簿中。
- 从多个表生成 .xlsx 格式的工作簿。每个表的数据显示在工作簿中的不同工作表中。

保存 .xlsx 文件将保留数据表单元格颜色、隐藏数据和排除的数据。不包括公式和大多数格式。

Excel 文件名的最大长度取决于用户计算机的操作系统。

将一个数据表另存为 Microsoft Excel 工作簿

要将一个数据表另存为 Microsoft Excel 工作簿，请执行以下步骤：

Windows

1. 选择**文件** > **导出**。
2. 选择 Microsoft Excel 并点击**下一步**。
3. 在 “文件名” 框中输入文件的名称。
4. （可选）要将文件另存为 .xls 文件，请选择**使用 Excel 97-2003 格式 (*.xls)**。
5. （可选）要避免保存后在默认电子表格中打开该文件，请在 “导出” 窗口中取消选择**保存后打开文件**。
6. 点击**保存**。

macOS

1. 选择**文件** > **导出**。
2. 选择 Microsoft Excel，然后选择格式。
3. 点击**下一步**。
4. 在**另存为**框中输入文件的名称。
5. （可选）要避免保存后在默认电子表格中打开该文件，请取消选择**保存后打开文件**。
6. 点击**导出**。

注意：在 Windows 上，当列数或行数超过 Excel 支持的数目时，数据将被拆分成多个工作表。

将多个数据表另存为 Microsoft Excel 工作簿

要在新的或现有 Microsoft Excel 工作簿中保存多个数据表，请执行以下步骤：

1. 打开数据表。
2. (Windows) 选择**文件** > **导出** > Microsoft Excel，选择**导出多个数据表**，然后点击**下一步**。
3. (macOS) 选择**文件** > **导出**，从 Excel 列表中选择**导出多个数据表**，然后点击**下一步**。
4. 在 “JMP 数据表” 列中，仅选择要保存的数据表。

5. 在 “工作表名称”列中，根据需要指定工作表选项卡的名称。
6. 在 “工作簿名称”框中输入工作簿的名称。

   默认情况下该工作簿以当前数据表的名称命名。
7. 点击**选择**并浏览以选择要在其中保存文件的目录。

   您可以覆盖现有工作簿。工作簿中已存在的工作表被替换。现有工作簿中不存在的工作表显示在工作表的末尾。
8. 点击**保存** （或在 macOS 上点击**导出**）。

   该文件被另存为 .xlsx 文件。

将 JMP 数据另存为 CSV 文件

在 Windows 上，JMP 可以将 JMP 数据表中的数据转换为逗号分隔值 (CSV) 文件。要将数据表另存为 CSV 文件，请执行以下步骤：

1. 当 JMP 中已打开特定的数据表时，请选择**文件 > 另存为**。
2. 从**另存为类型**列表中，选择 CSV （逗号分隔） (*.csv)。
3. 输入文件名。
4. 选择 CSV 文件的位置。
5. 点击**保存**。

注意：将数据表导出到 CSV 文件时使用逗号和回车换行符 (CRLF) 作为文件分隔符。若您需要使用其他分隔符，可使用另一扩展名，在以后使用 CSV 扩展名重命名该文件。

将 JMP 数据另存为文本文件

JMP 可以将 JMP 数据表中的数据转换为包含行和列的标准文本格式。

文本格式设置选项

将列名导出至文本文件 要将列标题保存在文本的第一行中并保存数据中的标签或标题信息，请选择**导出表标题**。

为所有列名添加引号，为所有字符值添加引号，为所有数值添加引号 选择一个添加引号的选项 （即使不需要引号）。导入 CSV 文件的一些程序可能需要引号来正确显示文本。始终将引号添加到包含特殊字符 （例如字段分隔符，通常为逗号或制表符） 、换行符 （通常为回车）或引号的字段。

字段结尾 选择标记字段 （或单元格）结尾的字符旁边的单选按钮。若未列出所需字符，请选择**其他**并输入一个字符。

行结尾 选择标记行结尾的字符旁边的单选按钮。若未列出所需字符，请选择**其他**并输入一个字符。

注意：

- 若在导入文本数据时发现双引号，分隔符规则会更改，JMP 会查找结束双引号。其他文本分隔符（包括引号中嵌入的空格）将被忽略，并视为文本字符串的一部分。
- 当您导入文本数据时，为包含图像数据的列分配“表达式”数据类型。

Windows

1. 选择**文件** > **导出**。
2. 选择**文本**，然后点击**下一步**。
3. 在 “文件名”框中输入文件名。
4. 点击**选项**按钮并指定行结尾字符和字段结尾字符，然后选择是否将列标题导出为文本。
5. 点击**确定**。
6. （可选）要避免保存后在默认文本编辑器中打开该文件，请取消选择**保存后打开文件**。
7. 点击**保存**。
8. 点击**是**。

macOS

1. 选择**文件** > **导出**。
2. 选择**文本**，然后点击**下一步**。
3. 在另存为框中输入文件的名称。
4. 指定行结尾字符和字段结尾字符，然后选择是否将列标题导出为文本。
5. （可选）要避免保存后在默认文本编辑器中打开该文件，请取消选择**保存后打开文件**。
6. 点击**导出**。

将 JMP 数据另存为 JSON 文件

您可以将数据表另存为 JSON 文件，该文件由名称和值对组成。JSON 文件经常用于在应用程序之间交换数据。

要将数据表另存为 JSON 文件，请按以下步骤执行：

Windows

1. 选择**文件** > **导出**。
2. 选择 **JSON 数据**并点击**下一步**。
3. 在 “文件名”框中输入文件名。
4. 点击**保存**。

macOS

1. 选择**文件** > **导出**。
2. 选择 **JSON 数据**并点击**下一步**。
3. 在另存为框中输入文件的名称。
4. 点击**导出**。

将 JMP 数据另存为 SAS 传输文件

您可以采用 SAS 传输文件格式保存 JMP 数据表。JMP 将文件名和列名中的空格替换为下划线、将列标题转换为大写 SAS 变量，并执行其他更改以符合传输文件的规范。

将数据表另存为传输文件时，您可以选择将其追加至现有的传输文件。在 JMP 中重新打开该传输文件时，两个数据表将分别在不同的窗口中打开。

注意：

- SAS 传输文件名的最大长度为 8 个字符。JMP 会警告超过该限值的字符将被省略。
- “表达式”列中的图像不能超过 200 个字符，否则将不受支持。

Windows

1. 选择**文件** > **导出**。
2. 选择 **SAS 传输文件**。
3. 点击**下一步**。
4. 在 “文件名”框中输入文件名。
5. 点击**保存**。
6. 点击**确定**。

   出现关于其他 SAS 格式修改的警示。
7. 在每个警示中点击**确定**。
8. （可选）将数据表追加至现有的 SAS 传输文件。选择**文件** > **导出**，选中**追加**复选框，然后点击**下一步**。选择要追加数据表的文件，然后点击**保存**。

macOS

1. 选择**文件** > **导出**。
2. 选择 **SAS 传输文件**。
3. 点击**下一步**。
4. 在**另存为**框中输入该文件的名称，然后点击**导出**。

   出现关于 SAS 格式修改的警示。

5. 在每个警示中点击**确定**。
6. （可选）将数据表追加至现有的 SAS 传输文件。选择**文件 > 导出**，选中**追加**复选框，然后点击**下一步**。选择要追加数据表的文件，然后点击**追加**。

将 JMP 数据另存为 Google Sheet

Google Sheet 是一种保存在 Google 帐户中的电子表格（由类似选项卡的工作表构成）。将 JMP 数据表导出到 Google Sheets 可允许您与其他人共享数据并在 JMP 之外编辑数据。

您可以将 JMP 数据表导出到现有的 Google 电子表格或创建新的电子表格。

要将 JMP 数据表导出为 Google 电子表格，请执行以下步骤：

1. 若您要导出到现有的电子表格，请首先执行这些步骤。如若不然，转至第 2 步。

从浏览器中登录 https://sheets.google.com。

打开现有电子表格。

从浏览器地址栏中复制 URL。

2. 在 JMP 中，打开您要导出数据的数据表。
3. 选择**文件 > 导出 > Google Sheets**。
4. 在 “电子表格类型”下，选择**现有**或**新建电子表格**。
5. 键入您的 Google 帐户名称。（没必要键入 @gmail.com。）
6. （仅限现有电子表格）在 “电子表格名称”旁边，粘贴您在第 1 步中复制的 Google 电子表格的 URL。
7. （仅限新电子表格）输入新电子表格的名称。
8. 输入工作表的名称 （选项卡名称）。
9. 点击**保存**。
10. 若您已从 JMP 导出了数据表，则该数据将显示在 Google Sheets 中。转到您的 Google 电子表格以查看指定工作表中的数据。

    或

    若您是第一次使用指定的 Google 帐户导出数据，系统将提示您为 JMP 提供对您的 Google 帐户的访问权限。

点击**允许**。

转到您的 Google 电子表格以查看指定工作表中的数据。

根据数据表的大小，保存操作可能需要一些时间。

关于导出 Google Sheets

- Google Sheets 不支持公式和“列表检查”列属性之类的 JMP 功能。

- 若电子表格为空，请在 JMP 日志中查找错误消息。在 Windows 上，选择**视图 > 日志**。在 macOS 上，选择**窗口 > 日志**。
- 若 Google Sheet 只有一个工作表，则不能用新工作表替换该工作表。添加另一个具有唯一名称的工作表，然后删除原始工作表。在“导出至 Google Sheets”窗口中，选择**现有的电子表格**，然后再次导出数据表。您指定的工作表名称将显示在工作表中。
- 有关安全、国家限制等方面的详细信息，请参见“导入数据”一章中第 156 页的“关于导入 Google Sheets”。
- 若在 Windows 上的 Internet Explorer 中选择了**在兼容性视图中显示 Intranet 站点**，则无法导出Google Sheet。要避免该问题，请执行以下步骤：
  1. 在 Internet Explorer 中，选择“工具”按钮 ，然后选择**兼容性视图设置**。
  2. 取消选择**在兼容性视图中显示 Intranet 站点**。

将 JMP 数据另存为 SAS 数据集

您可以将数据表另存为 SAS 版本 7 或更高版本的数据集。JMP 将列写为 SAS 变量，将行写为 SAS 观测，并执行其他更改以符合 SAS 数据集规范。

将数据导出至 SAS 数据集时，JMP 日期列变为具有适当 SAS 格式的 SAS 日期值。

在 Windows 上，JMP 可以将列属性和表属性另存为 SAS 9.4 .sas7bxat 文件中的扩展属性。该选项仅可以在“将 JMP 另存为”窗口中和“SAS 集成”首选项中可用。

Windows

1. 选择**文件 > 导出**。
2. 选择 **SAS 数据集**并点击**下一步**。
3. 在 “文件名”旁边输入文件名。
4. （可选）要保存 SAS 变量名或 SAS 格式，请选中**保留 SAS 变量名**或**保留 SAS 格式**复选框。
5. （可选）要将 JMP 元数据存储为 SAS 9.4 扩展属性，请选择**在 SAS 9.4 扩展属性中存储表和列属性**。 JMP 自动分配 .sas7bxat 文件扩展名。
6. 点击**保存**。
7. 点击**是**。
8. 若出现其他警示，请在每个警示中点击**确定**。

macOS

1. 选择**文件 > 导出**。
2. 选择 **SAS** 并点击**下一步**。
3. 输入文件名并选择导出选项。
   – 要避免导出行状态为“已排除”的行，请选择**接受排除行**。其他选项在上一节中已介绍。

4. 点击**导出**。

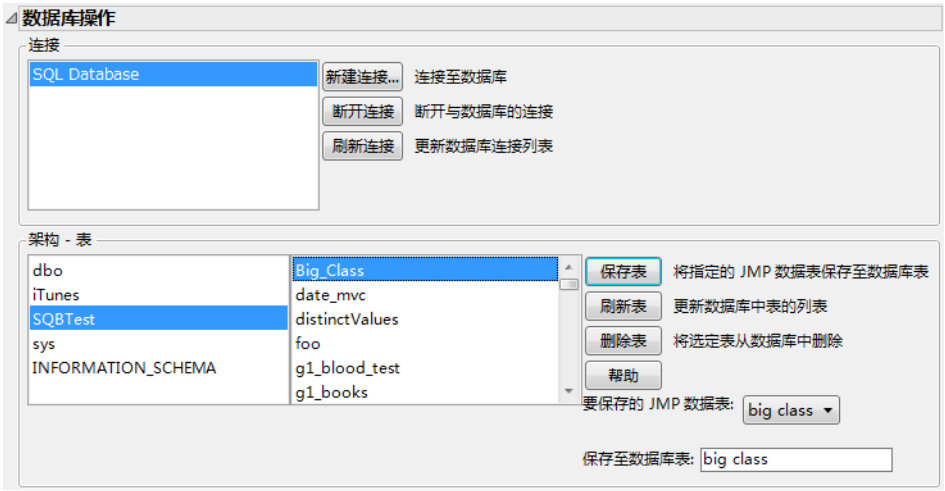
将数据表保存至数据库

您可以将数据表保存至系统上装有兼容的 ODBC（开放式数据库连接）驱动程序的任意数据库。

使用 ODBC 将数据表导出到数据库时，JMP 将数据作为字符（CHAR 或 NCHAR）或数值（FLOAT 或 DOUBLE PRECISION）数据类型导出。若列定义为 JMP 日期或时间，则列导出为字符数据。接收数据库负责将数据类型更改为正确的日期格式。

1. 首先，请连接至数据库。请参见第 501 页的“添加新的数据库连接”。
2. 选择**文件 > 数据库 > 保存表**。

图 10.1 “数据库保存表”窗口



3. 在“连接”框中，突出显示您要在其中保存文件的数据库的名称（图 10.1）。“连接”框包含您系统连接的数据库的列表。
4. 从**要保存的 JMP 数据表**菜单中，选择要保存至数据库的已打开的 JMP 数据表。
5. 在**保存至数据库表**框中，输入在数据库中保存该表时希望它具有的名称。
6. 点击**保存表**。

添加新的数据库连接

1. 点击**连接**。
2. 选择所需的数据源并点击**确定**。或者，要创建新的源，请点击**新建**按钮 (Windows) 或**选择 DSN** 按钮 (macOS)。

   根据所选的数据源（以及您计算机上安装的数据库驱动程序），系统可能为您提供多个窗口。请使用这些窗口来创建数据库源。

3. 选择要将文件保存至的数据库。

使用数据表替换数据库

要使用数据表替换数据库，请执行以下步骤：

1. 首先，请连接至数据库。请参见第 501 页的“添加新的数据库连接”。
2. 选择**文件 > 数据库 > 保存表**。
3. 在“连接”框中，选择要替换文件的数据库的名称。“连接”框包含您系统连接的数据库的列表。
4. 在“架构 - 表”框中，选择要替换的数据库表。

注意：记住该数据库表的名称。替换数据库表也要使用该名称。

5. 点击**删除表**。
   将从数据库中删除该数据库表。
6. 从**要保存的 JMP 数据表**列表中，选择您要保存至数据库的 JMP 数据表。
7. 在**保存至数据库表**框中，输入已删除的数据库表的名称。
8. 点击**保存表**。
   该数据表将保存至数据库。
9. 点击**断开连接**。
10. 关闭“数据库保存表”窗口。

保存和共享报表

要保存报表，请选择**文件 > 导出**。

- 在 Windows 上，您可以将报表另存为表 10.2 中所列的任意文件类型。
- 在 macOS 上，报表将被另存为 JMP 报表 (.jrp)。请注意以下几点：
  - 可以通过选择**编辑 > 记录**，然后选择**文件 > 保存**，将报表另存为记录。
  - 若您要将报表另存为文本、图像、 HTML、PowerPoint 或 RTF 文件，请选择**文件 > 导出**。支持的图像文件包括 .png、.emf、.jpg、.gif、.tiff、.svg 和 .eps。

表 10.2 Windows 支持的报表文件类型

文件类型	说明
Microsoft Word (.doc)	文字处理格式；图片、文本和表的组合。
增强元文件 (.emf)	可以包含向量和位图组件。

表 10.2 Windows 支持的报表文件类型 (续)

文件类型	说明
封装的 PostScript 文件 (.eps)	可显示图像位图预览的线条图（或向量图像）。
CompuServe 图形交换格式 (.gif)	压缩的位图图片。
Hypertext 标记语言 (.htm, .html)	浏览器格式；标记的文本和指向单独图片文件的引用。通过选择以下格式之一将图片保存在 HTML 文件中：PNG、JPEG、SVG 或 GIF。
HTML 或数据交互式 HTML（.htm、.html）	以交互式网页格式保存数据、报表和图形，让非 JMP 用户也能探索数据。请参见第 504 页的“将报表共享为交互式 HTML”。
联合图像专家小组（.jpg、.jpeg）	压缩的位图图片；照片的标准。请参见第 518 页的“设置导出图形的图形 DPI”。
JMP 记录 (.jrn)	复制到名为“记录”的单独窗口中的分析报表。您可以编辑它并向其追加其他报表。请参见第 530 页的“创建记录”。 在 macOS 上，选择**编辑 > 记录**，然后选择**文件 > 保存**。在 Windows 上，应选择**文件 > 保存**，然后选择记录格式。
JMP 报表 (.jrp)	在 JMP 中最初创建的分析报表。可以重新打开该报表，继续执行分析。包含 JSL 和嵌入数据或对数据的引用以重现生成实时报表。若您想再次操作该报表（可能是新数据），请选择该格式。
可移植文档格式 (.pdf)	用于共享文档的格式，与创建文档的操作系统或应用程序无关。
可移植网络图形 (.png)	压缩的位图图片；GIF 的后继格式。请参见第 518 页的“设置导出图形的图形 DPI”。
PowerPoint 演示文稿 (.pptx)	Microsoft PowerPoint 格式；图片、文本和表的组合。请参见第 517 页的“将报表另存为 PowerPoint 演示文稿”。
富文本格式 (.rtf)	文字处理格式；图片、文本和表的组合。通过选择以下格式之一将图片保存在 RTF 文件中：PNG、JPEG 或 EMF。
可扩展矢量图形 (.svg)	存储为文本的图片；最适用于二维图形。

表 10.2 Windows 支持的报表文件类型 (续)

文件类型	说明
标记图像文件格式 (.tif, .tiff)	光栅文件格式。请参见第 518 页的“设置导出图形的图形 DPI”。
文本格式 (.txt)	纯文本格式；无图片。

将报表共享为交互式 HTML

交互式 HTML 支持您共享包含动态图形的 JMP 报表，这样即便是非 JMP 用户也能探索数据。JMP 报表另存为可以与其他用户共享（例如，在共享网络驱动器上，通过电子邮件或在网站上）的交互式网页。然后用户可以如同在 JMP 中一样探索数据。

交互式 HTML 包含数据

当您将报表导出或发布为交互式 HTML 时，您的数据嵌入在 HTML 中。内容未经加密，因为 Web 浏览器无法读取加密数据。为了避免共享敏感数据，通过改为选择**文件 > 导出 > HTML** 将结果导出为非交互式网页。

以交互方式支持哪些功能

交互式 HTML 提供 JMP 中的一部分功能：

- 若完全支持您的报表中的功能，网页创建时没有警告。
- 若您的报表包含不支持的功能，您会在导出或发布窗口中看到一条消息指出部分实现交互式 HTML。详细信息，请参见 JMP 日志（**视图 > 日志**）。
- 部分实现或不支持的功能在网页中会静态显示。若您将鼠标悬停在网页中不支持的功能上，工具提示会显示该功能还无法交互。

有关了解交互式 HTML 输出的信息，请访问 https://www.jmp.com/interactive。

将单个报表导出为交互式 HTML

要将单个 JMP 报表导出为交互式 HTML，使用“导出”作为“数据交互式 HTML”选项创建单个网页。

1. 在 JMP 中，创建该报表并使它成为活动窗口。

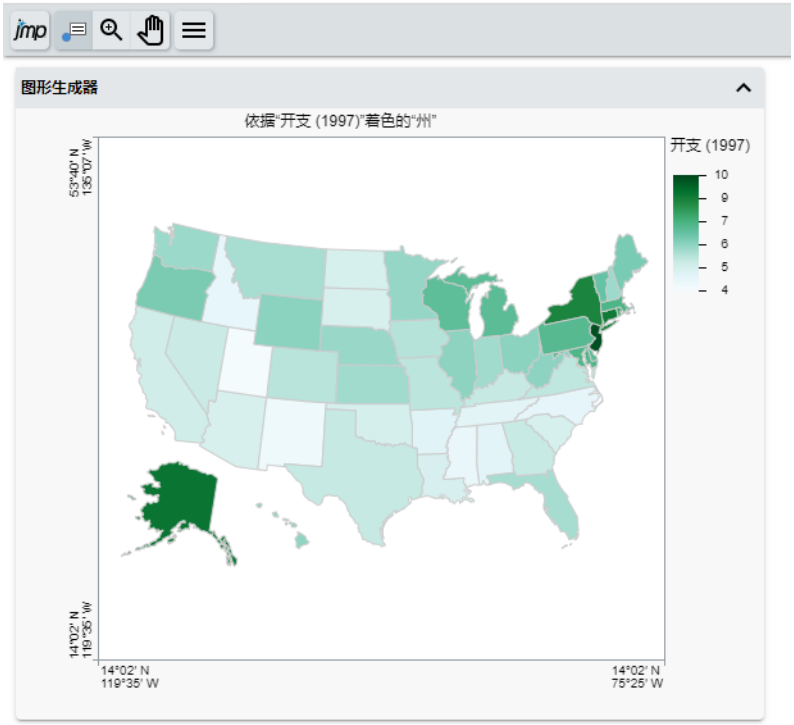
   注意：若报表包含“本地数据过滤器”，则它在 Web 报表中是静态的，用户无法更改选择。要使“本地数据过滤器”可交互，取消选择**包括**模式。（在“图形生成器”中，要查看**包括**模式，从“本地数据过滤器”红色小三角菜单中选择**显示模式**。）

2. 选择**文件 > 导出**，选择**数据交互式 HTML**，然后点击**下一步**。
3. 为文件命名。

4. （可选）要在导出 HTML 文件后在默认浏览器中打开它，选择**保存后打开文件**。
5. 点击**保存**。

   输出内容保存在选定文件夹中。

图 10.2 单个交互式 HTML 报表的网页



将多个报表发布为交互式 HTML

要将多个 JMP 报表发布为交互式 HTML，使用“发布至文件”选项，它会创建包含报表的索引页面。

提示：当您发布多个 JMP 报表时，请指定保存索引页面和报表文件的位置。您可以选择共享网络文件夹并将该位置提供给其他用户，或选择您的计算机上的文件夹并在共享前压缩文件。若您共享的对象不是 JMP 用户，这将会特别有帮助。

1. 在 JMP 中，创建报表。

   注意：若报表包含“本地数据过滤器”，则它在 Web 报表中是静态的，用户无法更改选择。要使“本地数据过滤器”可交互，取消选择**包括**模式。（在“图形生成器”中，要查看**包括**模式，从“本地数据过滤器”红色小三角菜单中选择**显示模式**。）

2. 从报表窗口中，选择**文件 > 发布 > 发布至文件**。

3. 选择要发布的报表。
4. （可选）更改父级文件夹所在的位置或更改将包含报表的子文件夹的名称。
5. 点击**下一步**。
6. 输入索引页面的标题。您还可以更新报表标题。
7. （可选）更改其他选项。请参见“交互式 HTML 报表选项”。
8. 点击**发布**。

图 10.3 多个交互式 HTML 报表的索引页面

9. 点击缩略图以打开报表。

有关使用交互式报表的详细信息，请从 HTML 报表中点击 ☰ > **帮助**。这将打开帮助，网址：https://www.jmp.com/interactive。

交互式 HTML 报表选项

标题 为索引页面（仅多个报表）或报表添加标题。

说明 （可选）将说明添加至索引页面或报表。说明将显示在标题下方。

定制 （仅为多个报表显示）更改网页的外观。您可以更改样式、主题、字体、徽标以及是否显示日期或时间。请参见“定制索引页面选项”。

发布数据 为交互式报表选择该选项。若您取消选择该选项，报表是静态的。

注意：要避免共享敏感数据，请将结果另存为非交互式网页。（选择**文件 > 导出 > HTML**。）

添加图像 将图像添加至网页底部。

打开发布的 Web 报表 您点击 “发布”后，在浏览器中打开网页。

运行后关闭报表 您发布 Web 报表后，关闭 JMP 中的报表。

“删除”图标 （仅为多个报表显示）删除报表。

“箭头”图标 （仅为多个报表显示）更改报表的顺序。

定制索引页面选项

样式格式 确定报表的布局。

大列表 在具有大缩略图的列中显示报表。

小列表 在具有小缩略图的列中显示报表。

网格 在行中显示报表。

定制 CSS 允许您指定一个 CSS 文件来格式化网页。将该 CSS 文件复制到名为 _css 的子文件夹中。指向 CSS 文件的链接是相对的，以便您可以将报表和支持文件发送到另一用户并维护 CSS 格式。

颜色主题 指定网页、标题和边框的颜色。默认网页有一个白色背景、橙色标题和蓝色边框。

更改字体 更改报表中应用的字体。

更改徽标 指定要与报表一起显示的图像。点击图像旁边的向上或向下箭头可以将它移到报表之上或之下。

显示日期 / 时间 显示或隐藏生成网页的日期和时间。

将报表发布至 JMP Live 或 JMP Public

JMP Live 是您的公司购买的用于私下共享 JMP 内容的网站。JMP Live 可由 SAS 或您的公司托管。JMP Public 是一个使用与 JMP Live 相同技术的网站，任何人都可在其中与公众共享 JMP 内容。

JMP Live 和 JMP Public 是用于共享 JMP 数据、可视化视图和仪表板的平台。在 JMP 中准备您的报告，并将其作为帖子发布，以便与您的同事（甚至那些可能没有 JMP 的同事）协作。

JMP Live 和 JMP Public 通过提供 JMP 的可视交互功能来探索您的数据，并允许您执行以下操作：

- 将您的报表存储在一个位置。
- 针对与需要查看报表的人员共享各个报表进行控制。
- 在易于访问的位置为您的团队维护最新的报表。
- 将报表功能扩展到其他位置：在网页、博客、社交媒体帖子等中嵌入报表并与之交互。

注册并登录到 SAS 配置文件中（仅限 JMP Public）

在将帖子发布到 JMP Public 之前，必须使用 SAS 配置文件注册并登录。

注意：若您未登录到 JMP Public，则只能看到特色帖子。

1. 在浏览器中，打开 https://public.jmp.com。

   支持的浏览器包括 Chrome、 Firefox、 Safari 和 Edge （基于 Chromium）。
2. 在右上部，点击**登录**。
3. 若您已有 SAS 配置文件，请使用这些凭证登录，然后点击**提交**。
4. 若您没有 SAS 配置文件：

点击**注册**。完成填写表格并点击**创建配置文件**。

按照电子邮件中的说明设置密码并激活您的配置文件。

添加 JMP Live 服务器

在发布到 JMP Live 之前，必须添加 JMP Live 服务器。

1. 在 JMP 中，选择**文件 > 发布 > 管理连接**。
2. 点击**添加新项**。
3. 输入连接的标识名称，例如 JMP Live。
4. 输入服务器 URL，其格式类似于：https://jmpliveserver.com:3501（其中 3501 是默认的 Web 服务器端口）。
5. （可选）要使用 JSL 创建 JMP Live 报表，请输入 API 密钥。
6. 点击**下一步**。
7. 使用您的用户名和密码登录 JMP Live 服务器。
8. 一旦成功创建连接，请点击**保存**。

   服务器名称显示在 “选择连接”下方。
9. 点击**退出**。

创建并发布 JMP 报表

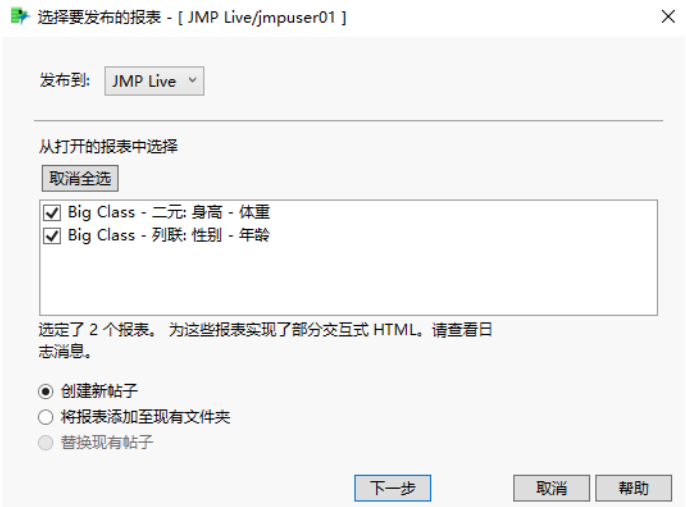
1. 在 JMP 中，创建报表。

   您可以创建一个报表（在 JMP Live 或 JMP Public 中显示为该报表自己的帖子），也可以创建多个报表 （显示在一个文件夹中）。
2. （若报表不包含 “本地数据过滤器”，请跳过。）若报表包含 “本地数据过滤器”，请在发布前添加列，否则 “本地数据过滤器”将是静态的而不是活动的。要添加列，请在 “本地数据过滤器”中选择这些列，然后点击 “添加”图标。
3. 选择**文件 > 发布 > 发布至 JMP Live** 或**发布至 JMP Public**。

– （仅限 JMP Live）若您尚未添加 JMP Live 服务器，请现在添加一个服务器。请参见第 508 页的“添加 JMP Live 服务器”。
– 您可能需要登录 JMP Live 或 JMP Public。

注意：要注册 SAS 登录信息，请访问 https://www.jmp.com 并点击页面顶部搜索框旁边的**登录**。

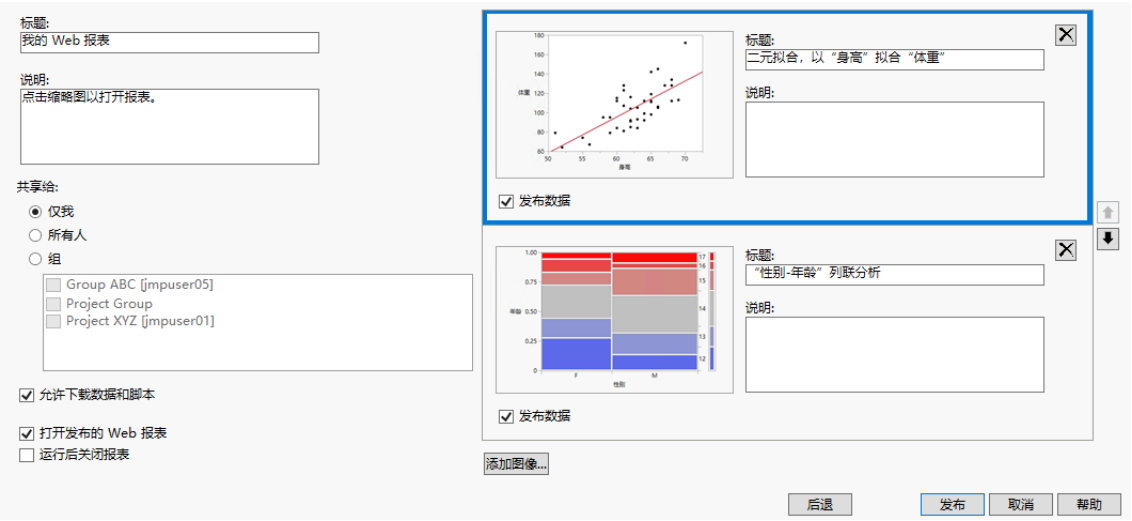
图 10.4 “选择报表”窗口



4. （仅限 JMP Live）在**发布到**旁边，选择一个 JMP Live 服务器。
5. 选择要发布的已打开报表。
6. 选择您的操作，然后点击**下一步**：
   – 第 510 页的“创建新帖子”
   – 第 511 页的“将报表添加至现有文件夹”
   – 第 513 页的“替换帖子中的现有报表”（若存在具有确切名称的帖子）

创建新帖子

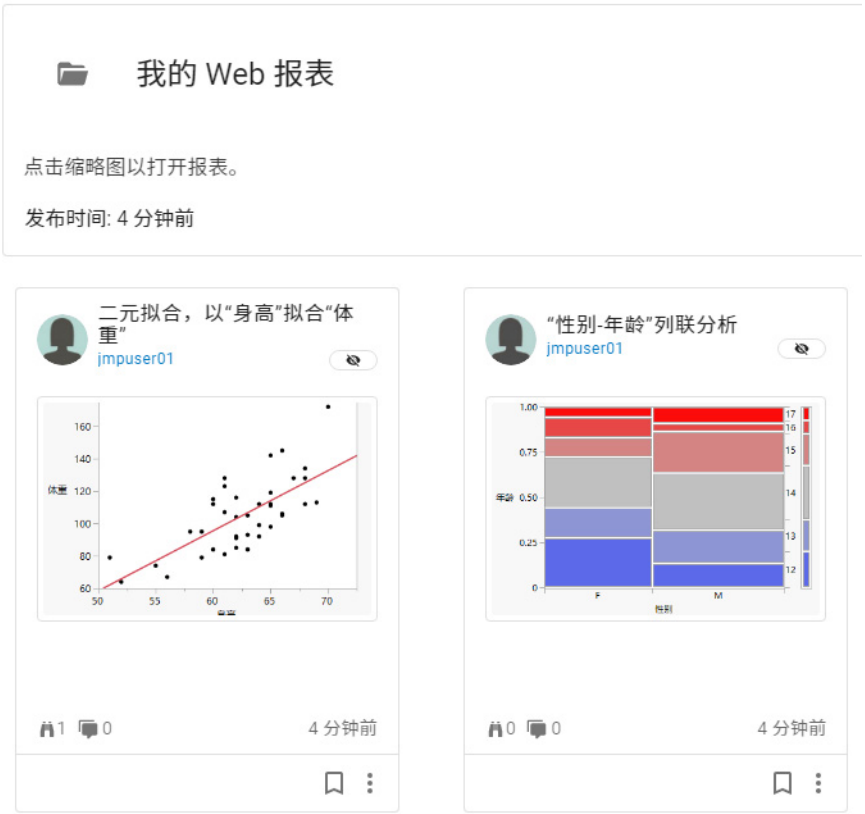
图 10.5 创建包含多个报表的新帖子



注意：若要发布单个报表，图 10.5 中的窗口和选项略有不同。

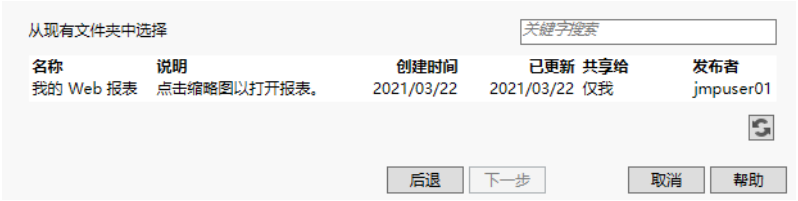
1. （仅限多个帖子）请输入文件夹的标题和可选说明。
2. （可选）更新每个报表的标题并添加说明。
3. 选择如何共享帖子：
   - **仅我** 保持您的帖子私有且仅对您可见。
   - **所有人** 在 JMP Public 或 JMP Live 上与所有人共享您的帖子。
   - **组** 与您选择的组共享您的帖子。您可以发布到的组显示在该选项下面。
4. 更改其他选项。请参见第 515 页的 “帖子选项”。
5. 点击**发布**。
6. 点击**确定**取消显示说明性窗口。

图 10.6 包含报表的 JMP Live 文件夹



将报表添加至现有文件夹

图 10.7 选择现有文件夹



注意：您只能看到您有权更新的文件夹。这些是您拥有的文件夹或您所属组中的文件夹。

1. 选择文件夹执行以下操作之一：
   - 点击列表中的某个文件夹，然后点击**下一步**。

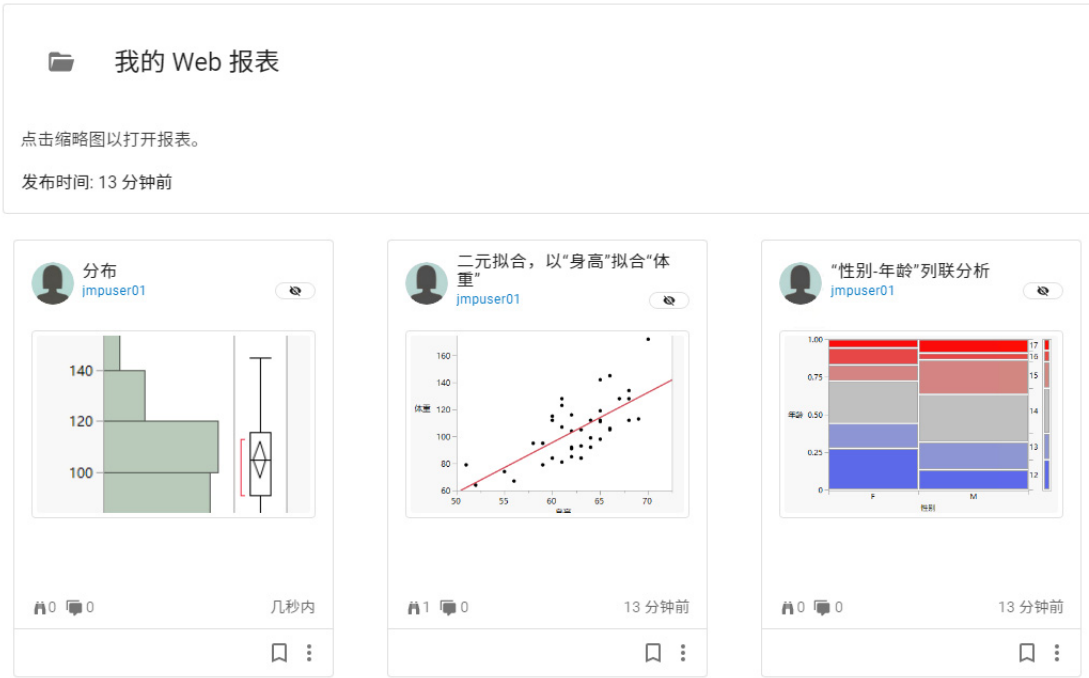
– 搜索文件夹 — 在“关键字搜索”框中，输入出现在文件夹名称或说明中的单词，然后点击某个文件夹。点击**下一步**。

图 10.8 将报表添加至现有文件夹

注意：若要发布多个报表，图 10.8 中的窗口和选项略有不同。

2. （可选）更新每个报表的标题并添加说明。
3. 更改其他选项。请参见第 515 页的 “帖子选项”。
4. 点击**发布**。
5. 点击**确定**取消显示说明性窗口。

图 10.9 添加至现有 JMP Live 文件夹的报表



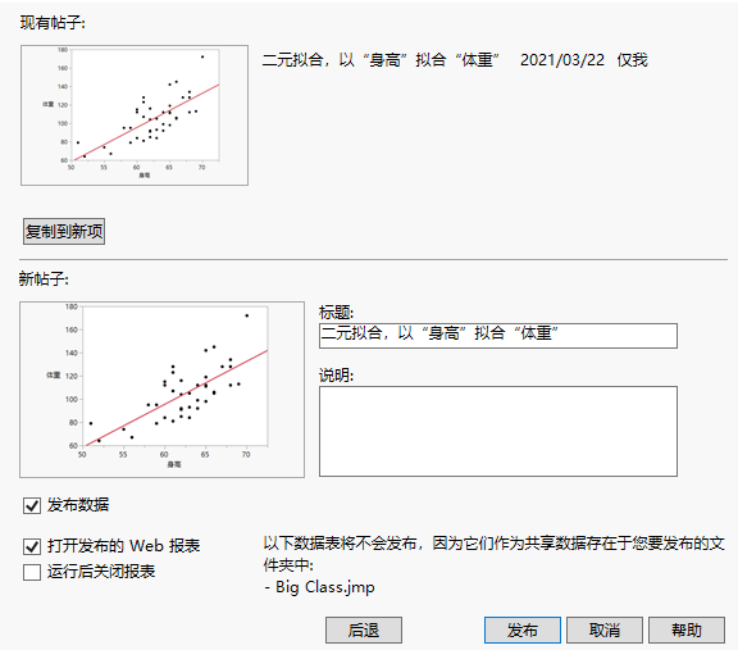
替换帖子中的现有报表

图 10.10 选择现有帖子

注意：您只能看到您有权更新的帖子。

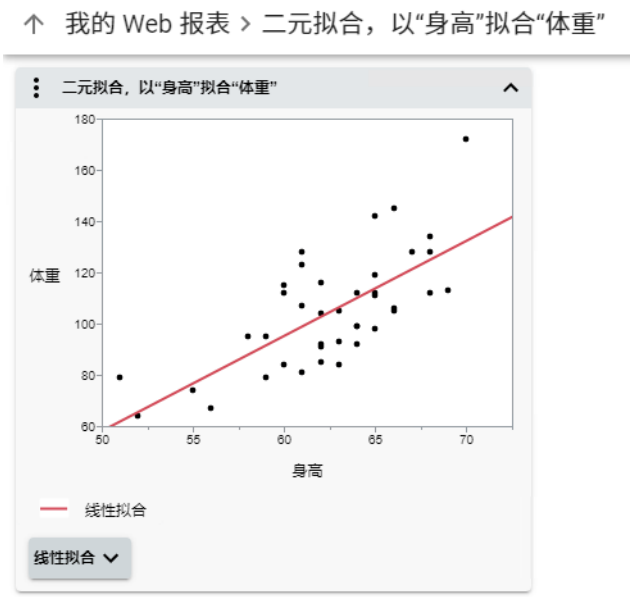
1. 选择某个帖子。执行以下操作之一：
   - 点击列表中的某个帖子，然后点击**下一步**。
   - 搜索帖子 — 在“关键字搜索”框中，输入出现在帖子名称或说明中的单词，然后点击某个帖子。点击**下一步**。

图 10.11 替换现有帖子



2. 选择以下选项之一：
   - 要使用现有帖子中的相同标题和说明（若存在），请点击**复制到新项**。
   - 要使用更新的标题和说明，请输入新的标题和说明。
3. 更改其他选项。请参见第 515 页的 “帖子选项”。
4. 点击**发布**。
5. 点击**确定**取消显示说明性窗口。

图 10.12 在 JMP Live 中替换的报表



帖子选项

标题 为网页添加标题。还可以更改报表标题。

说明 （可选）向网页添加显示在标题下的说明。您还可以添加显示在报表标题下的报表说明。

发布数据 为交互式报表选择该选项。若您取消选择该选项，报表是静态的。选中该选项后，将发布报表背后的数据和报表的 JSL 脚本。

> **注意：**要避免共享敏感数据，请将结果另存为非交互式网页。（对于单个报表，选择**文件 > 导出 > HTML**。对于多个报表，选择**文件 > 发布 > 发布至文件**，然后取消选择**发布数据**。）

启用警告 （仅适用于支持控制图的平台。）当帖子含有控制图警告时通知相关方。相关方包括帖子的发布者以及与其共享帖子并启用警告通知的所有组。

添加图像 将图像添加为仅图像报表。

允许下载数据和脚本 允许下载与报表关联的数据表和脚本。若不选择该选项，则在 JMP Live 中，数据表名称旁边没有下载选项，也没有用于重新创建报表的 JSL 脚本。请参见第 516 页的 “关于数据可访问性的注释”。

打开发布的 Web 报表 您点击“发布”后，在浏览器中打开网页。

运行后关闭报表 您发布 Web 报表后，关闭 JMP 中的报表。

“删除”图标 删除报表。

“箭头”图标 更改报表的顺序。

JMP Live 和 JMP Public 帮助

有关使用 JMP Live 或 JMP Public 的帮助信息，请点击 > **帮助**。这将打开以下帮助页之一：

- JMP Live：https://www.jmp.com/jmplive
- JMP Live（管理员）：https://www.jmp.com/jmpliveadmin
- JMP Public：https://www.jmp.com/jmppublic

关于数据可访问性的注释

已发布报表中表示的数据将以 HTML5 文件以及 JMP 数据表的格式在线提供。

禁用“允许下载数据”选项可防止任何有权访问 JMP Live 页面的用户下载 JMP 数据表格式的数据，但不会阻止用户获取包含 XML 格式数据的 HTML 文件。若启用“允许下载数据”选项，则允许有权访问 JMP Live 页面的用户以两种方式中的任意一种获取数据。JMP 格式只对有权访问 JMP Live 页面的用户有用（这些用户还需拥有单独的 JMP 软件许可证或知道如何读取 JMP 文件的第三方软件），但是 JMP Live 页面的用户只需一个 Web 浏览器即可查看 HTML 文件中的数据。

将报表另存为 PDF 文件

通过选择“文件”>“另存为”>“PDF 文件 (*.pdf)”(Windows) 或“文件”>“导出”>“图像”>“PDF”(macOS)，您可以将报表另存为 PDF 文件。

Windows

- 导出文件时，可以控制页眉、页脚和页边距等分页功能。
  - 选择**文件 > 导出**，再选择 **PDF**，然后点击**下一步**。
  - 为文件命名，点击**保存**，然后在“页面设置”窗口中更改设置。
- 您还可以通过选择**文件 > 页面设置**来控制分页。请注意，您可以点击**设置为默认值**以将这些设置应用到将来的页面设置中。
- 您还可以通过选择**文件 > 打印预览**来控制分页。

macOS

- 选择**文件 > 导出 > 图像 > PDF**。得到的 PDF 文件包含不带页眉或页脚的单个 PDF 图像。
- 您还可以选择**文件 > 打印**，然后从 PDF 窗口选择**另存为 PDF**。得到的 PDF 文件根据需要分为多个页面（包含页眉和页脚），并使用您在“打印”和“页面设置”窗口中选择的页面大小、方向、缩放比例和页边距。

Windows 和 macOS

通过 JSL，可以在 Windows 和 macOS 上控制分页。有关**设置页面设置**、**设置打印页眉**、**设置打印页脚**和**保存 PDF** 的详细信息，请参见《JSL Syntax Reference》。

将报表另存为 PowerPoint 演示文稿

通过将 JMP 结果另存为 Microsoft PowerPoint 演示文稿 (.pptx) 创建一个演示文稿。另存为 .pptx 文件后，在 PowerPoint 中重新排列 JMP 内容并编辑文本。JMP 报表中的各个部分通过不同方式导入到 PowerPoint。报表标题导入为可编辑的文本框，图形导入为图像。特定图形元素（如图例）可导入为单独的图像。图像在 PowerPoint 中打开后将调整大小以适合幻灯片格式。通常情况下，JMP 报表中每一个分级显示项都会在单独的 PowerPoint 幻灯片中显示。若报表中的分级显示项在报表另存为 .pptx 文件时处于关闭状态，则该分级显示项将从 PowerPoint 中排除。较大的内容可在多张 PowerPoint 幻灯片中显示。从 PowerPoint 文件中删除不想要的内容。

注意：在 Windows 系统中，要打开在 JMP 中创建的 pptx 文件，PowerPoint 要求的版本最低为 PowerPoint 2007。在 macOS 上，PowerPoint 要求的版本最低为 PowerPoint 2011。

1. 在 JMP 中，创建报表。
2. 选择**文件 > 导出 > Microsoft PowerPoint** 并点击**下一步**。
3. 选择图形文件格式。

   在 Windows 系统中， EMF 是默认格式。在 macOS 上， PDF 是默认格式。
4. （可选）要避免保存后在默认演示文稿程序中打开该文件，请取消选择**保存后打开文件**。
5. 为文件命名并保存该文件 （在 macOS 上为导出该文件）。

提示：

- 在 macOS 上不支持 Windows 上生成的本地 EMF 图形。在 Windows 上也不支持 macOS 上生成的本地 PDF 图形。为实现跨平台兼容性，请通过选择**文件 > 首选项 > 第三方数据** (Windows) 或 **JMP > 首选项 > 第三方数据** (macOS) 更改默认图形文件格式。 然后，将 **PowerPoint 图像格式**更改为 PNG 或 JPEG。
- 有时需要使用 PowerPoint 自动调整列宽度。PowerPoint 中对象的大小通常与 JMP 中的不同，这可能会导致表列的大小出现异常。
- 拖动 PowerPoint 中的列分隔符以调整列宽度，以便使布局更加美观。您还可以双击一个列边框来减小列中的空白区域。
- 默认 PowerPoint 模板安装在 JMP 目录的 pptx 文件夹中。您可以保存该 PowerPoint 模板的副本，在导出时若想始终使用备择模板，可以使用自己的模板替换默认模板。例如，您可能要创建具有德语默认字体的德语 PowerPoint 模板。请参见“自定义 PowerPoint 模板”。

自定义 PowerPoint 模板

PowerPoint 演示文稿使用默认的 JMP 模板。您可以通过覆盖默认模板文件来使用自定义模板。为此，请打开以下位置之一中的 JMPExportTemplate.pptx：

C:/Program Files/SAS/JMP/16/pptx/

C:/Program Files/SAS/JMPPro/16/pptx/

然后，选择您自己的 PowerPoint 模板，然后选择**文件 > 保存**。下次您将 JMP 内容另存为 PowerPoint 时，它将用您的自定义模板打开。另一种方法是：首先在 PowerPoint 中创建自定义模板，然后将其另存为 JMPExportTemplate.pptx。

注意：

- 无法编辑 macOS 上的自定义 PowerPoint 模板。
- 为了以后使用 JMP 默认模板，在覆盖文件之前请在您的机器上生成副本。

另外，您还可以通过 JSL 脚本使用自定义模板。有关详细信息，请参见《JSL Syntax Reference》。

设置导出图形的图形 DPI

在 Windows 上，有多种设置导出图形的分辨率（或 DPI）的方法：

- 要更改所有图形的默认 DPI，请在“Windows 特定”首选项中设置该 DPI。“默认”选项使用操作系统的默认设置。
- 以下脚本支持您将默认 DPI 设置为不同的值。若想要不同的 DPI，请更改该数字，然后运行该脚本。

  ```
  Pref( Save Image DPI( 600 ) );
  ```

  要在运行前面的脚本后验证默认 DPI，请运行以下脚本：

  ```
  Get Preferences( Save Image DPI );
  ```

  您运行该脚本后，“Windows 特定”首选项和“将报表另存为”窗口显示 DPI 为“默认”。该设置一直应用到您进行修改为止。在“将报表另存为”窗口中，选择**始终使用该设置**以在多个 JMP 会话中保存您的选择。
- 针对特定图形，在“将报表另存为”窗口中设置 DPI。

在 macOS 上，您无法更改默认 DPI，即 72。

将 JMP 分析另存为脚本

完成分析并接收报表后，您可能希望将用于生成报表的过程另存为 JSL 脚本。您可以将脚本保存在数据表内、报表内或单独的脚本窗口内。要进行保存，请点击报表标题中的红色小三角，然后选择您偏好的“保存脚本”选项。请参见第 393 页的“报表中的“保存脚本”菜单”。

JMP 非英文版本既可使用英文保存脚本，也可使用本地语言保存脚本。请参见“JMP 首选项”一章中第 583 页的“常规 JMP 首选项”。

以图形格式保存报表部分

您可以将分析报表窗口的某个部分另存为图形。您可以采用 .png、.jpg、.gif、.eps、.emf、.pdf、.tiff 和 .svg 等格式保存选定内容。在 macOS 上，您可以采用 .png、.jpg、gif、.svg、.eps、.pdf 和 .pptx 格式保存选定内容。

要保存选定内容，请执行以下步骤：

1. 点击选择工具 ( )。
2. 突出显示要保存的区域。
3. 在 Windows 上，选择**编辑 > 将选定内容另存为**。在 macOS 上，按 Control 并选择**文件 > 将选定内容另存为**。

   随即显示“将选定内容另存为”窗口（在 macOS 上显示“保存”窗口）。
4. 输入图形的文件名。

   在 macOS 上，更改文件扩展名可以将文件以 .png （默认格式）之外的格式保存。

   注意：若您使用“将选定内容另存为”并从中创建一个 PDF 文件，该 PDF 文件有页眉和页脚。使用 JSL，您可以在 Windows 和 macOS 上控制分页。请参见《JSL Syntax Reference》。
5. 在 Windows 上更改选项：
   - 选择要以哪种图形文件格式类型保存选定内容。请参见第 502 页的“保存和共享报表”。
   - 选择“图像 DPI 设置”。对于必须拉伸或打印的图像，最好取 300 或更高值。
   - 要保留您为当前会话选定的 DPI 设置，请选择**始终使用该设置**。
   - 要避免保存后在默认图形程序中打开该文件，请取消选择**保存后打开文件**。
   - 选择**下次调用该窗口时选择该过滤条件**保存当前会话的文件类型选择。

   注意：若看不到“选择该过滤器”选项，请选择用来保存图形文件的文件夹位置。
6. 点击**保存**。

提示：

- 您可以在“Windows 特定”首选项中将默认 DPI 设置选择值从 96 更改为 300 DPI。以下脚本支持您将默认 DPI 设置为不同的值。若想要不同的 DPI，请更改该数字，然后运行该脚本。

  ```
  Pref( Save Image DPI( 600 ) );
  ```
- 在 macOS 上，您无法更改默认 DPI，即 72。
- 当您将包含位图图像的图形或报表导出到 .svg 文件时，图像将嵌入 .svg 文件中。导出的内容更容易在其他文档中再次使用。

打印报表

要打印活动窗口中的报表，请选择**文件 > 打印**。该命令显示标准的打印窗口。窗口外观取决于您的操作系统和打印机驱动程序。

要插入分页符以进行打印，请执行以下步骤：

1. 右击报表窗口中的展开图标 。
2. 选择**编辑 > 分页符**。

复制和粘贴报表

当您需要在另一程序中使用 JMP 报表或数据表时，可以将报表或表的各部分粘贴或拖到另一程序中。例如，您可能要在 Microsoft Word 文件中向同事发送一个 JMP 报表的副本。

1. 点击选择工具 。
2. 点击并拖动（或按 Shift 键并点击）以选择报表窗口或数据表中的项。在报表窗口边缘附近点击可选择整个报表。
3. 点击选定项并将其从 JMP 拖至其他程序。或者，在 JMP 中复制选定项并将其粘贴至其他程序。将某个元素粘贴到其他应用程序中时，所用的格式取决于粘贴到的应用程序。若您使用应用程序的“选择性粘贴”命令，可以选择要使用的格式，如文本 (.rtf)、无格式文本 (.txt) 或位图 (.bmp)。

注意：

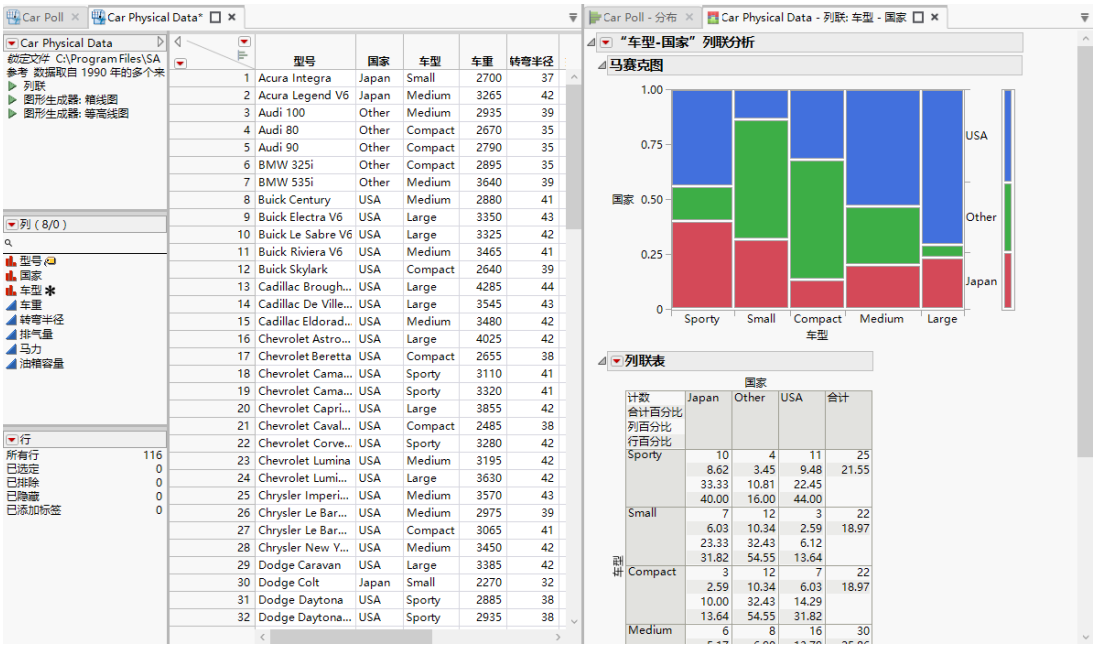
- 要将活动报表窗口中的所有文本（无图形）复制为无格式文本，请选择**编辑 > 复制为文本**。在 macOS 上，在 Microsoft Word 中选择**粘贴 > 特殊**，或在其他 macOS 应用中选择**粘贴并匹配样式**。
- 要复制图形，其中包括标签和轴，请右击该图形并选择**编辑 > 复制图形**。
- 要复制图形之外的任何对象，请右击并选择**编辑 > 复制图片**。例如，要复制报表，请右击包含报表标题的标题栏并选择上述选项。
- 要将文本框的文本复制到剪贴板，请右击该文本，然后选择**复制文本**。
- 要将一列文本复制到剪贴板，请右击该列，然后选择**复制列**。
- 要将整个表复制到剪贴板，请右击该表，然后选择**复制表**。
- “报表”首选项**报表 PNG 图像的透明背景**确定保存为 PNG 图像的报表是否具有透明背景。默认情况下不选择该首选项。

使用项目

使用 JMP 项目，您可以执行以下操作：

- 通过单个选项卡式 JMP 窗口更有效地探索您的数据
- 快速保存和重新打开一组相关的文件和报表
- 通过将您的表和脚本嵌入独立的项目文件中来轻松地共享您的工作

图 10.13 带数据表和报表的项目文件



本节包括下列信息：

- “创建新项目”
- “在项目中打开文件”
- “在项目中重新排列文件”
- “保存项目”
- “项目工作区”
- “项目书签”
- “项目内容”
- “项目中最近使用的文件”
- “项目日志”

- “将文件移动到项目中”
- “编写项目 On Open 脚本”
- “创建新项目的示例”

创建新项目

要创建新的空 JMP 项目，选择**文件 > 新建 > 项目** (Windows) 或**文件 > 新建 > 新建项目** (macOS)。

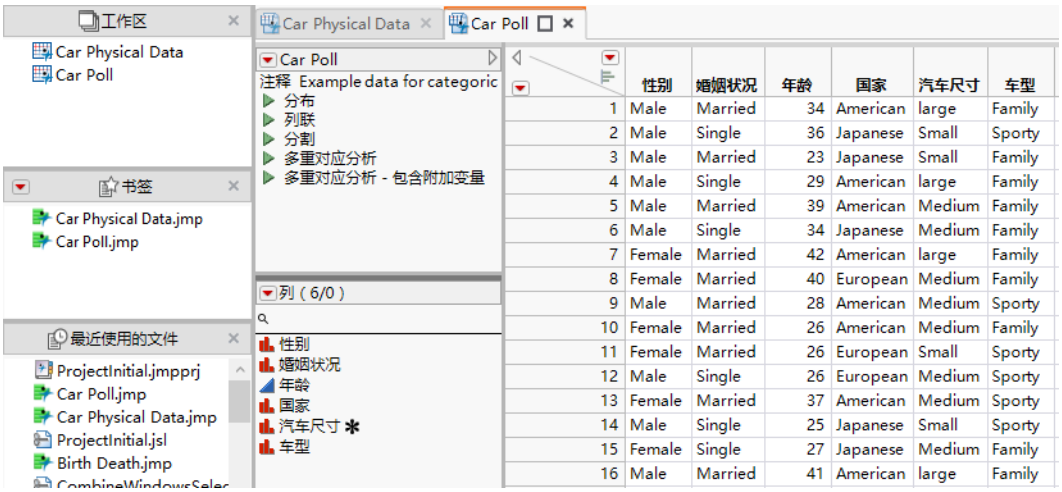
在项目中打开文件

在 JMP 项目中，您可以打开数据表，然后对数据运行分析。每个数据表和分析报表会在新的选项卡中打开。

1. 从项目窗口中，选择**文件 > 打开**，然后导航至您想打开的数据表。
2. 从数据表中，运行分析。

若您的计算机上数据表发生更新，则项目中任何关联的报表会在您重新打开项目时更新。

图 10.14 初始项目



提示：您可以在项目中使用 JMP 键盘快捷方式，例如 Ctrl+S 可以保存文件或 Ctrl+W 可以关闭活动的窗口窗格。有关完整列表，请选择**帮助 > 快速参考卡**。

在项目中重新排列文件

在 JMP 项目中，数据表和报表显示在各自的选项卡中。通过将选项卡拖到停放区中可以重新排列它们。

提示：要撤销或恢复停放项，选择**项目 > 撤销布局**或**项目 > 恢复布局**。要将所有表和报表返回到各自的选项卡，选择**项目 > 重置布局**。

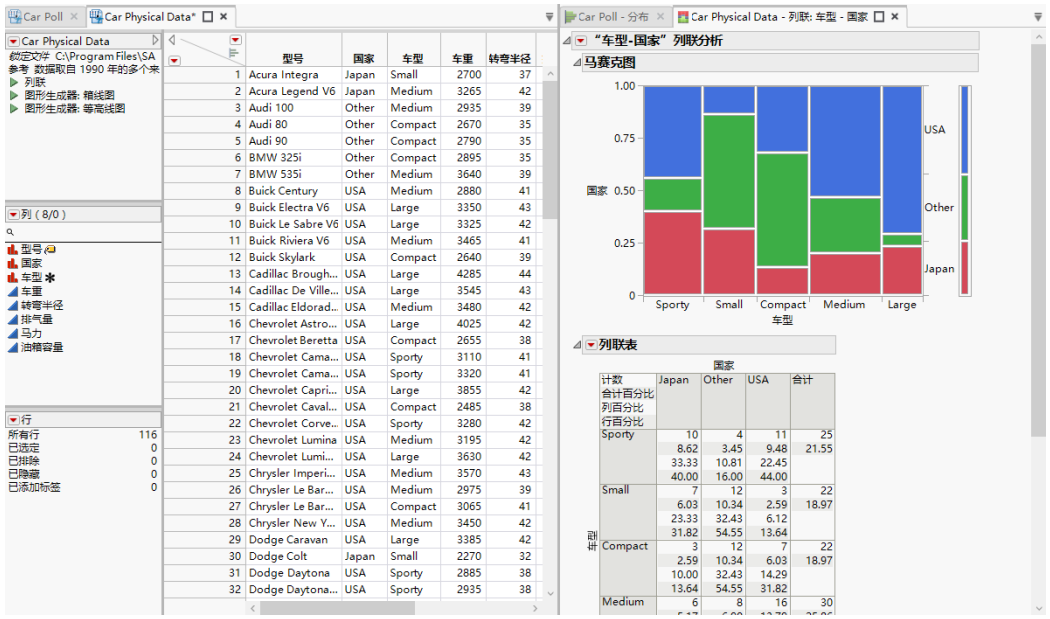
重新排列项目文件的示例

1. 选择**文件 > 新建 > 项目** (Windows) 或**文件 > 新建 > 新建项目** (macOS)。
2. 选择**帮助 > 样本数据库**，然后打开 Car Physical Data.jmp 和 Car Poll.jmp。
3. 在 Car Poll.jmp 数据表中，运行**分布**脚本。
4. 在 Car Physical Data.jmp 数据表中，运行**列联**脚本。
5. 将 Car Poll - **分布**报表选项卡向右拖动并放置在**停放右侧**区中。

   “分布”报表窗格显示在项目窗口的右侧。当您在选项卡之间进行切换时，该报表是停放状态并保持可见。
6. 将 Car Physical Data - **列联选项卡**向 Car Poll - **分布**报表的中间拖动并放置在**“停放”选项卡**区中。

   提示：您可以调整数据表窗口的大小以完全显示两个报表。

图 10.15 报表和数据表分组为选项卡



“分布”和“列联”报表现在以选项卡形式显示在一起，两个数据表也是如此。

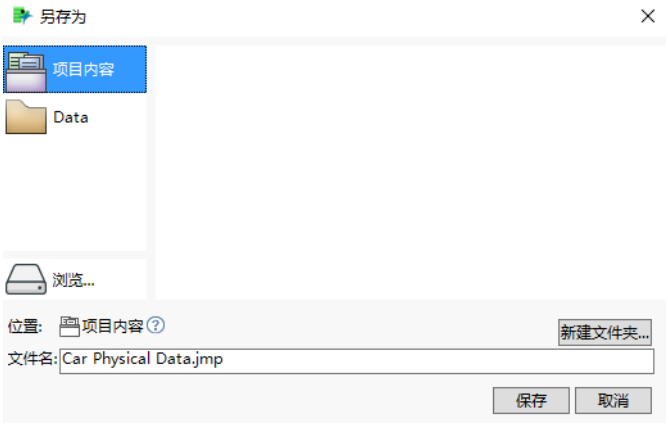
保存项目

若您想共享、分发或存档您的项目，通过将所有数据表和脚本保存至项目内容以创建一个独立的项目。这样会将所有项目文件和文件夹嵌入到您可以与其他 JMP 用户共享的单个项目文件。

1. （可选）要将项目另存为一个独立的项目，通过选择**文件** > **另存为**并选择**项目内容**以将每个数据表和脚本保存在项目中。您可以点击**新建文件夹**在项目内容中创建一个文件夹。

注意：您不需要保存报表，因为它们会自动保存至项目。

图 10.16 将数据表保存至项目内容



2. 选择**文件** > **保存项目**以保存项目文件。

图 10.17 保存项目文件

项目工作区

在 JMP 项目中，您可以在“工作区”工具窗格中查看所有当前打开的文件。报表显示在相应的数据表下方。活动的数据表显示为粗体。

提示：要隐藏或显示“工作区”工具窗格，选择**项目** > **显示工作区**。要指定创建项目时默认显示的工具窗格，选择**文件** > **首选项** > **项目**，然后选择初始工具窗格。

在“工作区”工具窗格中，您可以执行以下操作：

- 双击文件使其成为活动状态。

- 右击文件以将其关闭，隐藏（删除选项卡并灰显文件名）、添加书签以及进行其他操作。

项目书签

在 JMP 项目中，您可以在“书签”工具窗格中为文件或文件夹创建快捷方式。您还可以通过创建书签组以组织添加书签的文件和文件夹。

提示：要隐藏或显示“书签”工具窗格，选择**项目 > 显示书签**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

在项目中对打开的文件添加书签

- 右击选项卡并选择**添加书签**。
- 要对所有打开的文件添加书签，右击选项卡并选择**全部添加书签**。

对计算机上的文件或文件夹添加书签

将计算机中的文件或文件夹拖到“书签”工具窗格，或点击“书签”红色小三角菜单并选择**添加文件**或**添加文件夹**。

注意：若您从计算机上添加书签的文件夹中添加或删除文件，则项目中文件夹内容会自动更新。

打开添加了书签的文件

在“书签”工具窗格中，双击文件。

移除书签

在“书签”工具窗格中，右击文件并选择**移除书签**。

创建书签组

1. 点击“书签”红色小三角菜单并选择**新建组**。
2. 命名组并点击**确定**。
3. 将新的或现有的书签拖到组中，或在“书签”工具窗格中，右击组并选择**添加文件**或**添加文件夹**。

项目内容

在一个独立的 JMP 项目中，您保存至项目内容的任何文件会显示在“内容”工具窗格中。

提示：要隐藏或显示“内容”工具窗格，选择**项目 > 显示内容**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

从项目内容打开文件

在“内容”工具窗格中，双击文件。

在项目内容中创建文件夹

1. 点击“内容”红色小三角菜单并选择**新建文件夹**。
2. 命名组并点击**确定**。

将文件移入文件夹

在“内容”工具窗格中，将文件拖到文件夹中。

将计算机中的文件或文件夹复制到项目内容中

1. 点击“内容”红色小三角菜单并选择**将文件复制到项目中**或**将文件夹复制到项目中**。
2. 导航到文件并点击**打开**，或导航到文件夹并点击**选择**。

在项目内容中重命名文件

在“内容”工具窗格中，右击文件并选择**重命名**。

在项目内容中删除文件

在“内容”工具窗格中，右击文件并选择**删除**。

项目中最近使用的文件

在 JMP 项目中，您可以从“最近使用的文件”工具窗格中打开文件。

提示：要隐藏或显示“最近使用的文件”工具窗格，选择**项目 > 显示最近使用的文件**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。

打开一个文件

在“最近使用的文件”工具窗格中，双击文件或将其拖到项目中。

注意：保存至项目内容的打开文件不会显示在“最近使用的文件”工具窗格中。

项目日志

在 JMP 项目中，您可以在“日志”工具窗格中查看日志消息。

提示：

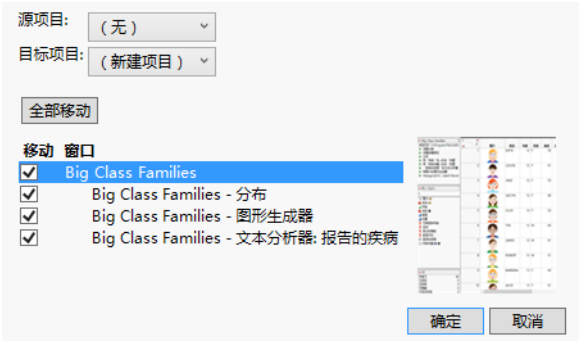
- 要隐藏或显示“日志”工具窗格，选择**项目 > 显示日志**。要指定创建项目时默认显示的工具窗格，选择**文件 > 首选项 > 项目**，然后选择初始工具窗格。
- 默认情况下，项目中出现的日志消息仅显示在项目日志中，不显示在主 JMP 日志中。要更改该设置，选择**文件 > 首选项 > 项目**，然后将**使用项目日志窗格**更新为**条件打开**或**从不**。

将文件移动到项目中

在 JMP 中，您可以将文件移入项目，移出项目或在项目之间移动。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class Families.jmp。
2. 运行 “分布”、“文本分析器”和 “图形生成器”脚本。
3. 从任意窗口中，选择**窗口 > 移至项目 / 从项目移动**。
4. 将 “源项目”保留为 （**无**）以查看在任何项目中均未打开的文件。
5. 将 “目标项目”保留为 （**新建项目**）以将选定文件移动到新创建的项目中。
6. 选中 Big Class Families.jmp 旁边的复选框。与数据表关联的图形也被一起选中。

图 10.18 将窗口移至项目 / 从项目移动窗口



7. 点击**确定**。使用选定表和报表创建了新项目。

编写项目 On Open 脚本

在“项目 On Open 脚本”框中，您可以添加 JSL 脚本，每次特定项目打开时将运行该脚本。例如，您可以创建一个脚本，它创建或修改报表，为用户提示信息，或将使用信息写入日志窗口。

1. 选择**项目 > 项目设置**。
2. 粘贴 JSL 脚本并点击**确定**。

提示：

- 请参见《Scripting Guide》，它解释了当**任何**项目（而非特定项目）打开时如何创建启动脚本。

- 要将现有项目用作新项目的模板，请在**文件 > 首选项 > 项目**下将现有项目指定为新项目模板。

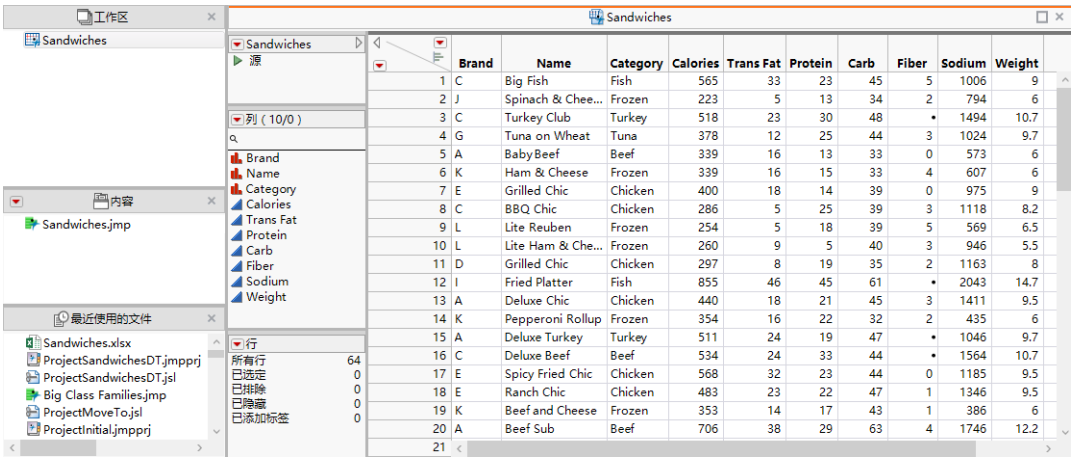
创建新项目的示例

在该示例中，您创建项目，导入数据，生成分析并在项目窗口中停放报表，创建子集表，然后保存、关闭和重新打开项目。

1. 选择**文件 > 新建 > 项目** (Windows) 或**文件 > 新建 > 新建项目** (macOS)。
2. 从项目窗口中，选择**文件 > 打开**。
3. 打开 sandwiches.xlsx 文件，默认情况下它位于此处：
   C:/Program Files/SAS/JMP/16/Samples/Import Data

   提示：在底部，您可能需要将**所有 JMP 文件**更改为 Excel 文件。
4. 点击**导入**。
5. 选择**文件 > 保存**。
6. 确保已选择**项目内容**。
7. 将文件名更改为 Sandwiches.jmp 并点击**保存**。

图 10.19 导入了三明治数据的项目

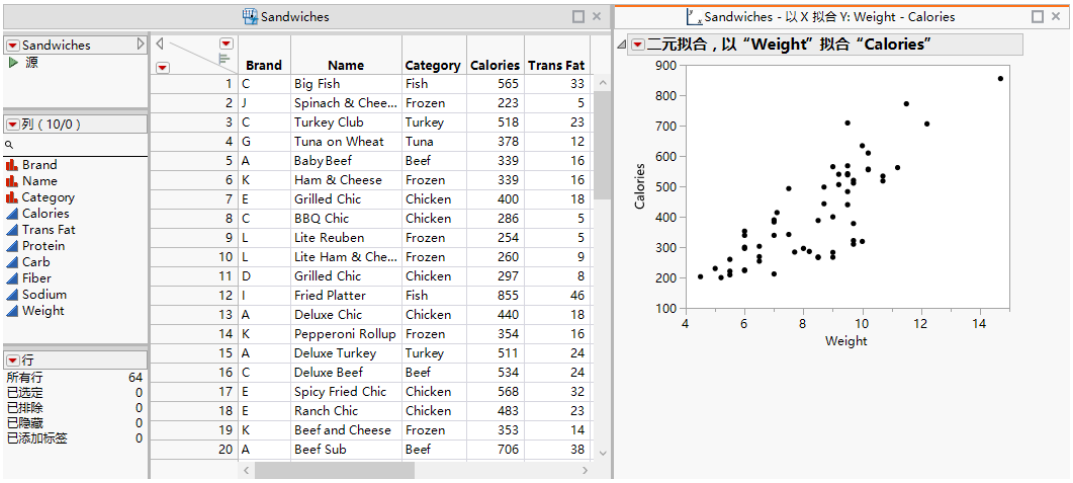


	Brand	Name	Category	Calories	Trans Fat	Protein	Carb	Fiber	Sodium	Weight
1	C	Big Fish	Fish	565	33	23	45	5	1006	9
2	J	Spinach & Chee...	Frozen	223	5	13	34	2	794	6
3	C	Turkey Club	Turkey	518	23	30	48	•	1494	10.7
4	G	Tuna on Wheat	Tuna	378	12	25	44	3	1024	9.7
5	A	Baby Beef	Beef	339	16	13	33	0	573	6
6	K	Ham & Cheese	Frozen	339	16	15	33	4	607	6
7	E	Grilled Chic	Chicken	400	18	14	39	0	975	9
8	C	BBQ Chic	Chicken	286	5	25	39	3	1118	8.2
9	L	Lite Reuben	Frozen	254	5	18	39	5	569	6.5
10	L	Lite Ham & Che...	Frozen	260	9	5	40	3	946	5.5
11	D	Grilled Chic	Chicken	297	8	19	35	2	1163	8
12	I	Fried Platter	Fish	855	46	45	61	•	2043	14.7
13	A	Deluxe Chic	Chicken	440	18	21	45	3	1411	9.5
14	K	Pepperoni Rollup	Frozen	354	16	22	32	2	435	6
15	A	Deluxe Turkey	Turkey	511	24	19	47	•	1046	9.7
16	C	Deluxe Beef	Beef	534	24	33	44	•	1564	10.7
17	E	Spicy Fried Chic	Chicken	568	32	23	44	0	1185	9.5
18	E	Ranch Chic	Chicken	483	23	22	47	1	1346	9.5
19	K	Beef and Cheese	Frozen	353	14	17	43	1	386	6
20	A	Beef Sub	Beef	706	38	29	63	4	1746	12.2
21										

8. 选择**分析 > 以 X 拟合 Y**。
9. 选择 Calories 并点击 **Y，响应**。
10. 选择 Weight 并点击 **X，因子**。
11. 点击**确定**。
12. 将 Sandwiches - 以 X 拟合 Y 选项卡向右拖动并放置在**停放右侧**区中。

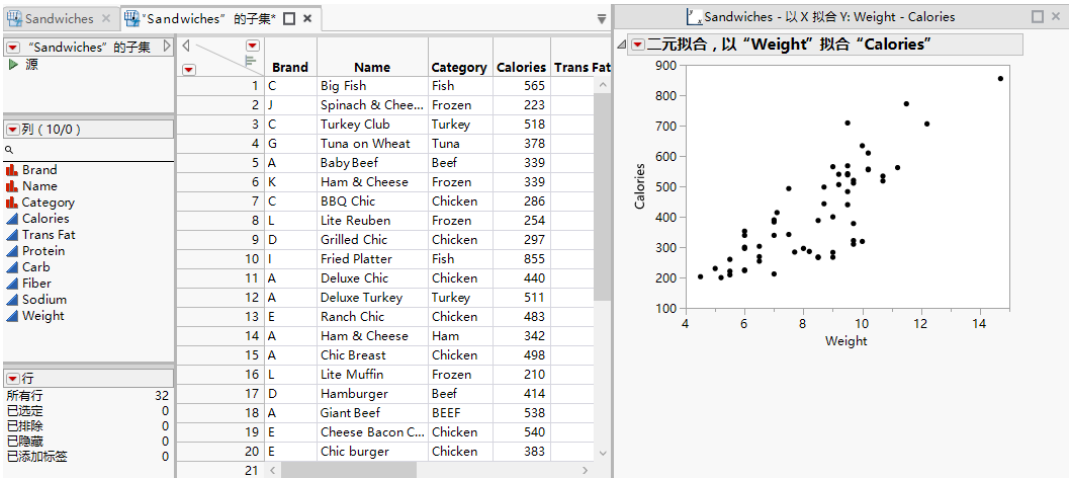
提示：要显示整个报表，您可以将数据表和报表之间的线向左拖动。

图 10.20 停放在右侧的 “以 X 拟合 Y”（二元）报表



13. 选择**表 > 子集**。
14. 在 “行”下方，选择**随机 : 抽样率 0.5**。
15. 点击**确定**。

图 10.21 包含未保存的子集表的项目



16. 选择**文件 > 保存项目**。
17. 导航至您想保存项目的文件夹，命名项目文件，然后点击**保存**。
18. 关闭项目。

19. 选择**文件** > **打开**并打开您的项目文件。

请注意，您未保存的子集表已自动保存至项目文件。

创建记录

当您想存储结果并展示它们时可以创建 JMP 记录。记录日志的其中一个好处就是您可以捕获分析而不依赖于外部文件。另一好处是可以将特定报表或图形复制并粘贴到记录。这使您可以创建定制报表来展示给其他用户。

您使用记录可以创建两种类型的演示：

- 静态演示嵌入一个或多个 JMP 报表的输出，固定在一个特定时刻。输出类似于当您将输出另存为 PDF 文件时得到的样式。没有数据表，无法对点进行选择、着色、隐藏、排除等。仅可以进行外观更改（字体、轴、大小）。好处是不依赖于外部文件。
- 动态演示从包含文本和按钮（或链接）的分级显示项进行构建，可帮助您组织、隐藏或显示一组数据表和报表。这种记录是动态的，因为您可以打开报表和表并与它们进行交互，从而选择点、对数据取子集、生成更多图形或重新运行分析。

图 10.22 显示您可以向记录添加的一些项。

图 10.22 记录的示例

注意：

- 创建完记录后，选择**编辑** > **锁定记录**。要解锁记录以进行编辑，请选择**编辑**，然后取消选择**锁定记录**。

图 10.23 锁定记录

- 您可以将记录保存为其他格式，例如 Microsoft Word、HTML、PDF 等。请参见第 537 页的“以其他格式保存记录”。
- 记录支持 GZ 压缩以减小文件大小。请参见“JMP 首选项”一章中第 583 页的“常规 JMP 首选项”。在记录中嵌入大型数据表时，此功能特别有用。

- 考虑在“常规”首选项中设置**自动保存超时**值以在指定的分钟数过后自动保存打开的报表。该自动保存值还应用于数据表、脚本、项目和报表。

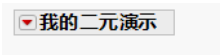
进行演示制作记录的示例

多数用户使用 PowerPoint 之类的演示应用程序提供演示。有了 JMP 记录，您不再需要使用演示应用程序：可以将所有项目符号与活动链接和按钮组合起来，帮助自动执行您要演示的分析。报表中的交互功能也得以保留。

请按照下面的示例操作，使用记录创建演示。

1. 选择**文件 > 新建 > 记录**。
2. 在记录中右击。通过选择**添加分级显示项**开始设置分级显示项。
3. 输入演示的标题 “我的二元演示”。

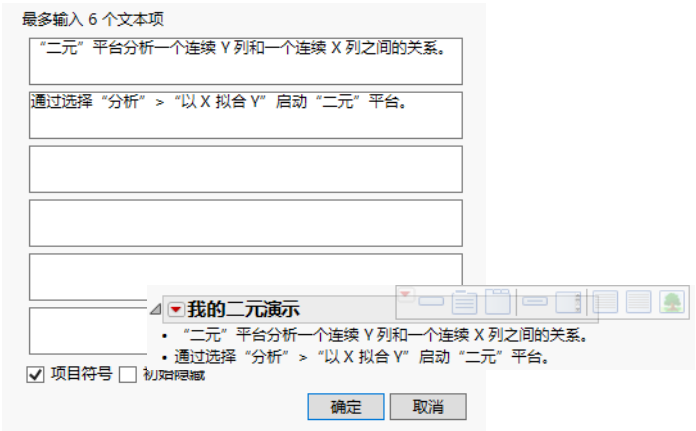
图 10.24 添加分级显示项



演示文稿应有分级显示项节点。当您给出演示文稿时，按顺序嵌套、打开和关闭这些节点。

4. 要在分级显示项中添加要点，请点击“我的二元演示”标题栏中的红色小三角并选择**添加文本项**。
5. 输入图 10.25 所示的文本。

图 10.25 添加文本项



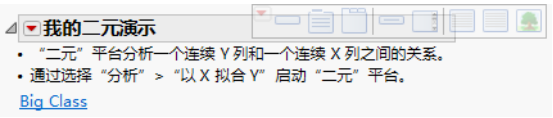
6. 选中**项目符号**复选框并点击**确定**。

提示：记录顶部会出现一条水平闪烁线。使用箭头键将这条线移动到要插入对象的位置。

现在将窗口引用添加为链接。这些链接支持您在演示过程中通过单击来打开文件。

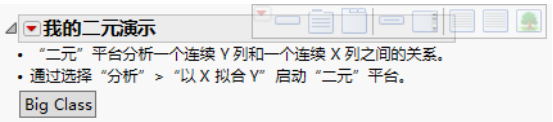
7. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
8. 点击 “我的二元演示”标题栏中的红色小三角，并选择**添加窗口引用**。
9. 选择 Big Class 并点击**确定**。

图 10.26 添加窗口引用



10. 点击窗口引用以验证链接。
11. 通过右击链接并选择**样式 > 正常**，将窗口引用显示为一个按钮。

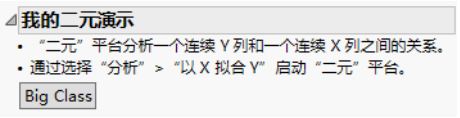
图 10.27 将链接更改为按钮



12. 选择**编辑 > 锁定记录**查看最终记录。

    锁定记录时，记录工具栏和插入光标不再显示。当您从平台中选择 “保存脚本” > “至记录”时，记录不会追加到打开的锁定记录。

图 10.28 最终记录



提示：也可以通过以下方式将对象添加到记录中：从记录工具栏中拖动对象，或将光标放在需要对象的位置再双击记录工具栏中的对象。将光标置于工具栏上方时，工具栏即可用。

新建记录

1. 关闭所有打开的记录，然后执行以下操作之一：
   - 要创建空记录，请选择**文件 > 新建 > 记录**。或者，从“JMP 起始页”窗口中，选择“记录”部分中的**新建**。
   - 要基于整个报表或数据表创建记录，请选择**编辑 > 记录**。

- 要基于特定报表或报表中的文本创建记录，请点击并拖动选择 ( ) 工具以选择报表中或数据表中的相邻项。要选择不连续的项，请按 Shift 键并使用选择 ( ) 工具点击这些项。然后选择**编辑 > 记录**。
- 要基于数据表中的特定文本创建记录，请选择该文本，然后选择**编辑 > 记录**。
- 要基于多数图形创建记录，请右击图形，然后选择**编辑 > 记录**。（不适用于曲面图和三维散点图。）

提示：若您喜欢使用键盘快捷方式，请按 Ctrl+J (Windows) 或 Command+J (macOS) 创建新记录。

2. 选择**文件 > 保存**，以 .jrn 格式保存该记录。

管理共享记录中的链接

在计算机上插入指向数据表的链接时，指向该数据表的整个路径都嵌入在该记录中。要共享该记录或将其移至另一个计算机，请将链接的文件存储在记录所在的同一目录中或存储在共享网络驱动器上。因为文件位于同一目录，因此不必要提供完全路径名。无论在何处打开该记录，链接都不受影响。

提示：

- 压缩记录和支持文件，以便另一用户可以将它们提取到某个目录。若文件仍在该目录中，则链接有效。
- 若您创建到某个样本数据表的链接，请考虑在路径中使用 $SAMPLE_DATA 变量。同样，您可以使用 $DESKTOP（若文件在您的桌面上）或 $DOCUMENTS（针对您的个人文档文件夹）。
- 您可以在一个分级显示项中同时编辑几个链接。还可以通过一个选项来替换链接中的字符串。右击该分级显示项旁边的展开图标，然后选择**追加项 > 编辑链接**。

在记录中创建可迁移链接

1. 将数据表和您要引用的其他文件放置在同一文件夹中。
2. 在该文件夹中保存记录。
3. 在记录底部右击，选择**添加文件引用**，选择该文件，然后点击**打开**。
4. 在记录中右击该链接，然后选择**编辑 > 设置脚本**。

   注意提供完整文件路径。

   ```
   Open("C:\Users\shlori\journals\San Francisco Traffic.jmp")
   ```

5. 删除字符串中除文件名之外的内容。

   ```
   Open("San Francisco Traffic.jmp")
   ```

将数据表嵌入记录

为避免链接到某个外部数据表，可以将该数据表复制到一个脚本，然后将其放入记录。这样做的好处是其他用户始终可以访问您的数据，您不需要编辑链接。缺点是若您要编辑数据，必须编辑脚本。

1. 在数据表中，点击数据表名称旁边的红色小三角并选择**复制表脚本**。

图 10.29 复制表脚本

2. 在记录中，右击并选择**添加脚本按钮**。
3. 在 “名称” 框中输入脚本名称。
4. 将表脚本粘贴到 “脚本” 框中，然后点击**确定**。
5. 右击该链接并取消选择**下划线样式**以将该链接显示为按钮。

向记录添加分级显示项并对其进行编辑

您可以将文本和链接组合到某个分级显示项中，然后点击灰色展开图标展开和折叠该组。

- 要添加分级显示项，请在记录中右击，然后选择**追加项 > 添加分级显示项**。您还可以向已创建的分级显示级别添加嵌套分级显示级别。（点击已记录的图形的红色小三角菜单时，**添加分级显示项**选项不可用。）点击该分级显示项的红色小三角并选择**添加分级显示项**。
- 要编辑分级显示项标题，请双击该标题，输入新名称，然后按 Enter 键。
- 要在某个分级显示项内添加文本或链接，请点击该分级显示项的红色小三角并选择第 535 页的 “记录中的右击选项” 中所述的选项。请注意，点击已编写脚本的项的红色小三角时，这些选项不可用。
- 要在某个分级显示项外添加文本或链接，请右击记录，然后选择第 535 页的 “记录中的右击选项” 中所述的选项之一。
- 要编辑某个链接，请右击该链接，选择**编辑 > 设置脚本**，编辑路径，然后点击**确定**。

- 要编辑记录中显示的链接的名称，请右击该链接，选择**设置按钮名称**，输入新名称，然后点击**确定**。

记录中的右击选项

添加分级显示项 将文本和链接组合到可折叠和可展开的列表中。

添加文本项 输入最多六段文本，同时可选择添加项目符号或隐藏文本。要取消隐藏，请右击隐藏的文本并取消选择**隐藏**。

注意：要在分级显示级别之外添加文本，您还可以在记录末尾双击，并在出现的框中输入文本。在框外点击可退出编辑模式。

添加窗口引用 创建指向已在 JMP 中打开的窗口的链接。

添加文件引用 创建指向计算机上的文件的链接。

添加文件目录 创建指向选定目录中所有文件 （包括非 JMP 文件）的链接。

添加打开文件 创建分级显示级别和指向 JMP 中的打开文件 （日志除外）的链接。例如，指向打开的 Big Class.jmp 数据表和图表的链接显示在名为 Big Class 的可折叠标题之下。记录、脚本和教程之类的文件分组到名为**其他文件**的可折叠标题之下。

添加 URL 引用 创建指向通过 Internet 协议（如 HTTP、FTP 或 FILE://）传递的文件的链接。

添加脚本按钮 创建指向 JSL 脚本的链接。右击该链接并取消选择**下划线样式**以将该链接显示为按钮。

添加脚本框 添加在其中键入脚本的框。

锁定 隐藏记录工具栏和插入光标。若从平台选择“保存脚本”>“至记录”，则结果不会追加到锁定的记录。

编辑链接 允许您重命名文件链接 （例如，为了删除某个硬编码的路径）。还提供测试链接的选项。若该路径有效，点击“测试链接”时什么也不会发生。若该地址无效，将在日志中显示一条错误消息。要更新无效链接，请点击**替换**。

调整为合适大小 调整窗口大小以确保按钮名称完整显示。

控制记录中分级显示级别的显示

您可以即时控制要显示的分级显示级别（例如，关闭选定级别之下的所有子级别）。点击该级别的红色小三角，选择**分级显示框**，然后选择“打开”或“关闭”命令之一。有关“打开”和“关闭”命令的详细信息，请参见“JMP 报表”一章中第 387 页的“JMP 报表的显示选项”。

提示：在 Windows 中，通过按 Alt 键并右击分级显示级别的灰色展开图标，可以显示“分级显示框”选项。

向记录添加图形或图像

在记录中支持 PNG、GIF、JPG、BMP 和 TIF 文件。执行以下操作之一将某图形添加到记录：

- 从 JMP 内部复制图形或从另一个应用程序中复制图像，然后选择**编辑 > 粘贴**。JMP 将图像放在记录的末尾。
- 将图形或图像从另一个窗口拖放到记录中。（请参见“JMP 报表”一章中第 446 页的“将背景图像粘贴到 JMP 图形中”。）蓝线指示可以放置图像的位置。

提示：考虑使用外部程序调整图形大小。然后您可以将该图形以 100% 分辨率插入记录，而不必拖动图形来调整其大小。

定制记录中的项

- 要点击记录项并将其拖至其他位置，请选择选择工具 ( )，选择该项，然后将该项拖至新位置。会出现一条线来指示可以放置该项的位置。
- 要调整文本换行，请右击文本并选择**设置换行**。输入每行所需的像素数，然后点击**确定**。要对所有文本项应用换行，请选择**设置换行**，选择**将该数值扩展到其他文本框**，然后点击**确定**。
- 要修改图坐标轴，请双击或拖动该轴。请参见“JMP 报表”一章中第 429 页的“定制图形中的轴和轴标签”。
- 要在报表中的任意位置添加文本或形状，请使用绘图工具（注解、线条、多边形和简单形状）。有关绘图工具的详细信息，请参见“JMP 报表”一章中第 450 页的“向 JMP 报表和图形添加注解和形状”。
- 要调整图形大小，请执行以下操作之一：
  - 点击并拖动边缘。
  - 右击图形，然后选择**大小/尺度 > 框架大小**。请参见“JMP 报表”一章中第 426 页的“调整图形大小”。
- 为了修改数字列的格式，您编辑显示框的“属性”窗格。请参见“JMP 报表”一章中第 410 页的“修改数值列的格式”。

防止修改记录

打开某个记录后，若随后又对另一个报表或数据表进行记录，则第二个报表或数据表将添加至已打开记录的末尾。

要防止修改记录，请右击记录底部的空白区域，然后选择**锁定**。其他报表不会添加至该记录（如下一节所述）。JMP 会在您再次选择**编辑 > 记录**命令时，转而查找打开的未锁定记录或创建新记录。

要解除对记录的锁定，请右击记录底部的空白区域，然后取消选择**锁定**。

向记录追加报表

要向已记录的报表追加其他报表，请打开该记录，然后选择**编辑** > **记录**。若选择了分析窗口的某个区域，选择**编辑** > **记录**仅保存选定区域而不是整个窗口。

注意：

- 记录窗口具备与报表窗口相同的功能：您可以点击图标、点击并拖动，以及通过右击访问菜单。
- 报表记录后，记录的副本不再连接到数据表。要维护该连接，请将报表保存为 JMP 报表并在记录中插入一个指向它的链接。

以其他格式保存记录

您可以采用其他格式（如 HTML 或 PDF）保存记录。将记录另存为 HTML 格式后，在 Web 浏览器中查看该文件时，按钮和链接均有效。

注意：在打印机设置中定义的页眉和页脚显示在 PDF 文件中。不过，您可以在 JSL 脚本中设置左、中、右页眉和页脚。请在 JMP 脚本索引中搜索**设置打印页脚**消息。（在 JMP 中选择**帮助** > **脚本索引**。）

Windows

1. 选择**文件** > **导出**。
2. 选择格式。请参见第 502 页的 “保存和共享报表”。
3. 点击**下一步**。
4. 命名并保存文件。

macOS

1. 选择**文件** > **导出**。
2. 选择格式。请参见第 502 页的 “保存和共享报表”。
3. 点击**下一步**。
4. 命名并导出该文件。

删除记录中的项

1. 点击选择工具 ( )。
2. 选择要删除的项并按 Delete 键。

保存 JMP 会话

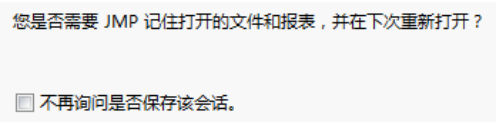
每次使用 JMP 都称为一次**会话**。**已保存的会话**是一个 JSL 脚本，使用该脚本可重新打开文档并重新运行分析，将 JMP 状态恢复到保存会话脚本时的状态。已保存的会话可帮助您返回以前状态，而不必手动重新打开文件并重新运行分析。

若您是高级用户，务必要了解有哪些会话信息会保留在已保存的会话中。已经保存的任何文档（如数据表、脚本和记录）将重新打开。支持脚本保存的 JMP 窗口将重新运行（相当于点击报表中的红色小三角菜单并选择**重新运行 > 重新运行分析**）。不会保存运行脚本的连带效果，比如全局变量值或定制窗口；也不会保存高度交互的分析的状态。

退出 JMP 时保存会话

已保存会话最常见的用法是在 JMP 退出时保存状态，以便 JMP 重新启动时恢复退出时的状态。默认情况下，JMP 会询问您是否要在每次退出该程序时保存会话状态（图 10.30）。若保存会话状态，在退出 JMP 之后又返回到 JMP 时，不必打开之前使用的文件即可恢复到退出前的状态。

图 10.30 退出时保存会话



若要在退出时始终保存会话，请选择**不再询问是否保存该会话**并点击**是**或**否**。该选项也可以阻止 JMP 在退出时保存会话。

您可以随后在“首选项”中更改该选项：

1. 选择**文件 > 首选项**。
2. 选择**常规**图标。
3. 在**退出时保存会话**选项旁边，选择希望 JMP 在何时保存会话。选项包括：“始终”、“从不”和 “提示”（询问您是否要保存会话）。

手动保存会话

您还可以将会话保存到所选的位置并继续工作，这样您可以随时恢复已保存的状态。若想加强对会话保存和恢复的控制，手动保存会话十分有用。当您要维护多个独立会话状态时，该选项特别有用。每个状态具有不同的文件和分析集。

创建 JMP 会话的脚本

1. 选择**文件 > 保存会话脚本**。

2. 在窗口中输入脚本的名称并点击**保存**。

手动保存会话的另一种方式是创建每个会话的记录。记录可以是记事本样式或项目样式的文件。使用记录可以将针对文件的引用收集到项目中、开发演示启动平台、记录项目，以及将许多脚本存储在一个位置。请参见第 530 页的“创建记录”。

创建 JMP 会话的记录

1. 打开要包括在记录中的文件。
2. 选择**文件 > 新建 > 记录**。或者，要将打开的文件追加到现有记录，请打开该记录。
3. 在空记录中右击并选择**添加所有打开的文件**。

保存日志窗口

选择**视图 > 日志**可显示一个窗口，用于监视 JMP 活动，例如执行过程中的 JSL 语句和脚本错误；此外，运行公式、矩阵或其他操作时，也可能会向日志窗口写入相关信息。

保存日志

1. 打开“日志”窗口
2. 选择**文件 > 另存为**。
3. （仅限 Windows）要在向日志添加文本时自动打开日志，请选择**文件 > 首选项 > Windows 特定**。在**打开 JMP 日志窗口**列表中，选择**每当添加文本时**。

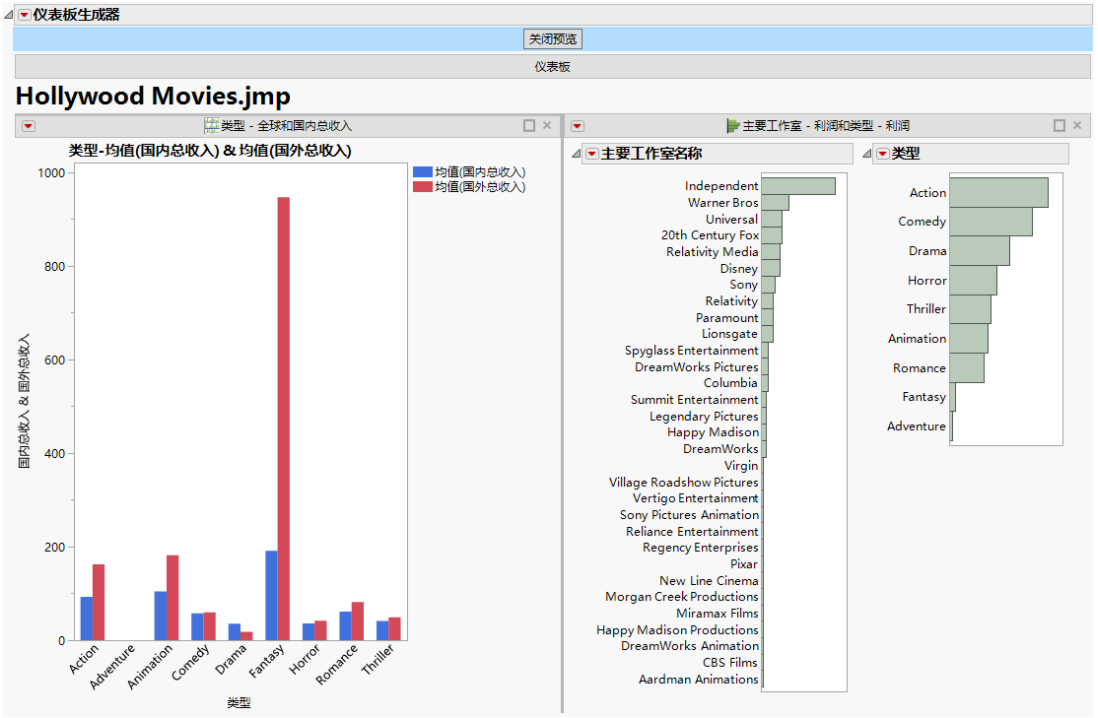
第 11 章

使用仪表板组合和演示报表

扩展 JMP

JMP 是一个可扩展程序，它支持您扩展其功能以创建您自己的 JMP 组件。利用“仪表板生成器”，您可以将报表组合到仪表板中，仪表板是支持您定期运行和演示报表的可视工具。无需具备 JSL 知识。从模板入手，点击报表组件，插入图形和图片，便可生成仪表板。您还可以在“仪表板生成器”外组合窗口来创建仪表板。

图 11.1 显示两个报表的仪表板



目录

通过创建仪表板来组合 JMP 报表 543
示例：创建包含两个 JMP 报表的仪表板 543
使用“仪表板生成器” 544
“仪表板生成器”中的红色小三角选项 545
从样本仪表板入手 546
编辑仪表板 546
查看运行的仪表板 549
示例：组合窗口以创建仪表板 553
示例：向仪表板添加“选择内容过滤器” 555
示例：创建 JMP 查询仪表板和插件 557
示例：从两个数据表创建仪表板 559

通过创建仪表板来组合 JMP 报表

利用“仪表板生成器”，您可以通过组合报表来创建您的仪表板，而不必编写 JSL 脚本。类似“应用程序生成器”，“仪表板生成器”具有拖放界面，但提供预先配置的显示框。在“仪表板生成器”中，显示框不要求像在“应用程序生成器”中那样进行定制。您可以将报表和图形放入选项卡、嵌入数据库查询、将仪表板另存为插件，还可以执行其他操作。

您还可以通过使用“窗口”>“组合窗口”选项组合若干打开的窗口来快速创建仪表板。例如，您可以组合两个报表和一个数据表，将其排列在仪表板上。相关示例，请参见“示例：组合窗口以创建仪表板”。

示例：创建包含两个 JMP 报表的仪表板

假定您创建了两个报表，并且想要在第二天针对更新的数据集再次运行这些报表。本例显示如何基于报表创建仪表板。

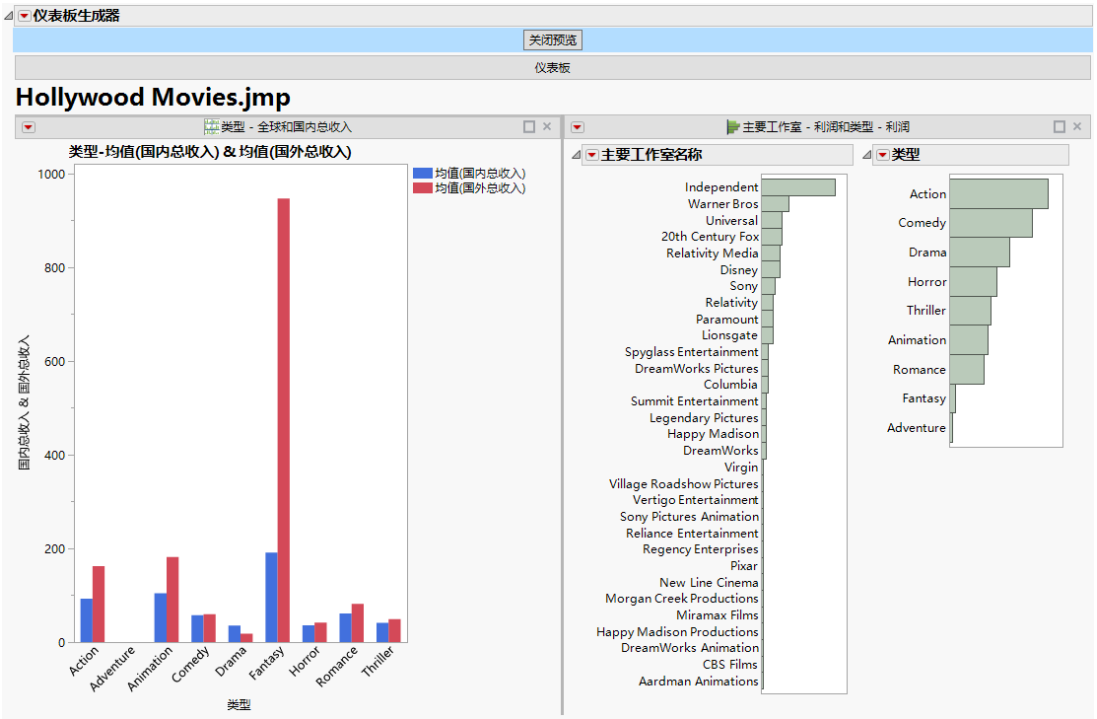
1. 选择**帮助** > **样本数据库**，然后打开 Hollywood Movies.jmp。
2. 要创建报表，运行名为 “分布 : 主要工作室 - 利润和类型 - 利润” 和 “图形生成器 : 类型 - 全球和国内总收入” 的表脚本。
3. 从任意窗口中，选择**文件** > **新建** > **仪表板**。

   随即出现常见布局的模板。
4. 选择 2x1 **仪表板**模板。

   工作区中随即出现可容纳两个报表的框区。
5. 在 “报表” 列表中，双击报表缩略图将其放入仪表板 （图 11.3）。
6. 点击 “仪表板生成器” 红色小三角并选择**预览模式**。

   仪表板预览随即出现。请注意，各图形不仅彼此链接还与数据表链接。它们也具有与 “分布” 平台和 “图形生成器” 平台相同的红色小三角选项。

图 11.2 包含两个 JMP 报表的仪表板



7. 点击**关闭预览**。
8. 在“仪表板生成器”中，选择**文件 > 保存**，然后将该文件另存为 .jmpapp 文件。

   装有 JMP 的用户可以打开该数据表，通过双击该文件即可显示该仪表板。请注意，.jmpapp 文件无法编辑。
9. 返回“仪表板生成器”窗口，选择**文件 > 另存为**，然后将该文件另存为 .jmpappsource 文件。

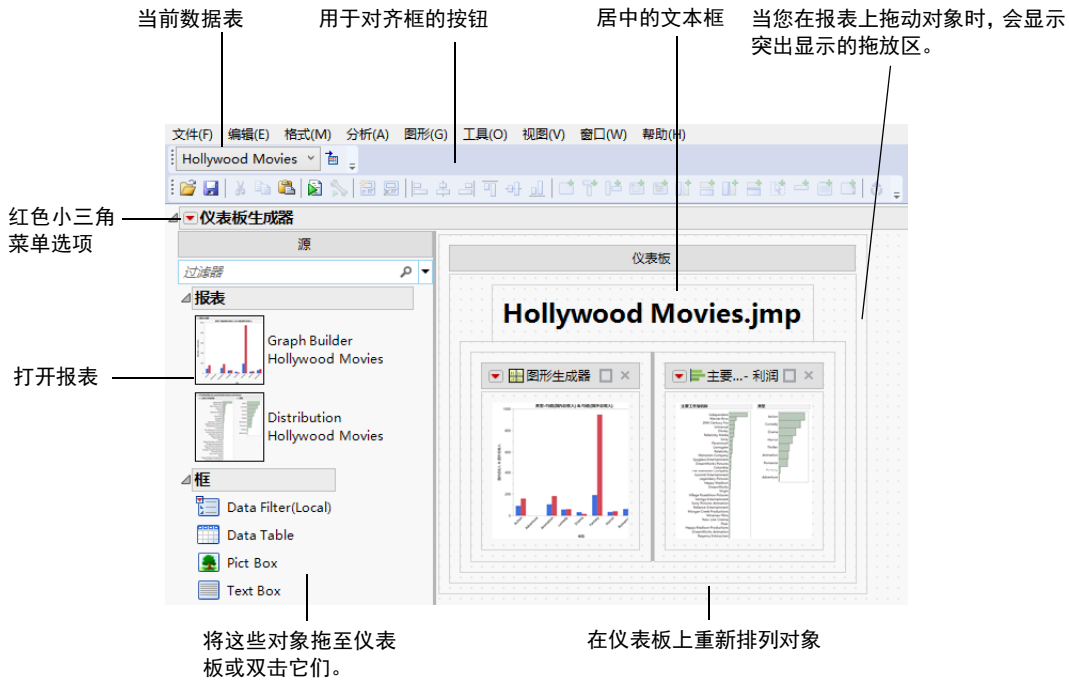
   该仪表板保存在文件中，您将来可对其进行编辑。

注意：您可能还希望将该仪表板另存为一个插件。用户随后可通过选择主“插件”菜单中的菜单项（而不是打开 .jmpapp 文件）来查看该仪表板。请参见第 557 页的“示例：创建 JMP 查询仪表板和插件”。

使用“仪表板生成器”

图 11.3 显示了开发中的仪表板。左窗格中的两个报表是从 Hollywood Movies.jmp 样本数据表生成的。这些报表添加到右窗格中。“Hollywood Movies.jmp”显示在添加到仪表板中的文本内，该文本框使用工具栏按钮居中显示。

图 11.3 “仪表板生成器”窗口



“仪表板生成器”中的红色小三角选项

“仪表板生成器”红色小三角菜单提供运行仪表板、预览仪表板和将仪表板另存为插件之类的选项。

运行仪表板 启动仪表板。

预览模式 显示仪表板的预览。点击仪表板上的**关闭预览**可返回仪表板模式。

重新开始 从工作区中删除对象并选择新的模板。

显示网格 在工作区上显示网格虚线。默认选定该选项。

显示属性 显示或隐藏用于编辑显示框的“对象”和“属性”面板。有关编辑属性的详细信息，请参见《Scripting Guide》。

自动滚动 随着您在工作区边缘附近拖动对象而自动水平或垂直滚动。默认选定该选项。

仪表板模式 取消选择该选项可在“应用程序生成器”中显示仪表板，您可在其中完全定制应用程序并编写脚本。请参见《Scripting Guide》。选择**仪表板模式**可返回仪表板。

“源”面板 显示或隐藏“源”面板（显示报表组件的左列）并对图标进行分类。默认选定“显示源”。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

至数据表 将脚本保存至用于生成报表的数据表。这允许您从数据表再次运行脚本来重新创建结果。

至记录 在记录窗口中保存一个链接来运行分析的脚本。若已打开一个记录窗口，则将脚本添加到该记录窗口。脚本包含到数据表的路径。请注意若找不到数据表，将不运行脚本。

至脚本窗口 打开脚本编辑器窗口并向其中添加脚本。若已打开一个脚本窗口，则将脚本添加到该打开窗口的底部。

至插件 支持 JMP 用户安装该应用程序并从 JMP 菜单启动它。请参见《Scripting Guide》。

至模板 将应用程序另存为模板，您可自该模板创建新的仪表板。根据您的操作系统和 JMP 版本，模板另存在以下位置：

- Windows (JMP)："/C:/Users/<用户名>/AppData/Roaming/SAS/JMP/16/DashboardTemplates"
- Windows (JMP Pro)："/C:/Users/<用户名>/AppData/Roaming/SAS/JMPPro/16/DashboardTemplates"
- macOS："/Users/<用户名>/Library/Application Support/JMP/16/DashboardTemplates"

从样本仪表板入手

打开“仪表板生成器”时会显示仪表板的示例。您可以修改这些样本而不是从空的仪表板开始操作。仪表板也安装在 JMP Samples/Dashboards 文件夹中。

以下样本仪表板可用：

Instant Dashboard.jmpappsource 组合两个多元报表（“主成分 / 因子分析”和“具有全部主成分的 T 方”）。

Six Quality Graphs Dashboard.jmpappsource 创建三个控制图、一个“分布”报表和一个“能力分析”报表。

编辑仪表板

第 543 页的“示例：创建包含两个 JMP 报表的仪表板”显示如何基于两个报表创建仪表板。这一部分显示如何组织这些报表并预览最终仪表板。

在仪表板编辑模式中重新排列报表

为帮助您在编辑仪表板时重新排列报表，围绕报表的边框包含拖放区。将报表拖至框内不同区域时，框的一部分会突出显示。突出显示的部分显示您可以放置该框的位置。

图 11.4 显示如何创建选项卡式报表。要将“分布”报表放入某个选项卡，请将该报表拖过另一个报表，直到条形图报表的右上角突出显示。“分布”报表将显示在另一个报表顶部的选项卡中。

要删除选项卡式报表，请点击该报表的中部并将其拖到该选项卡之外。

图 11.4 在仪表板编辑模式中处理选项卡式报表

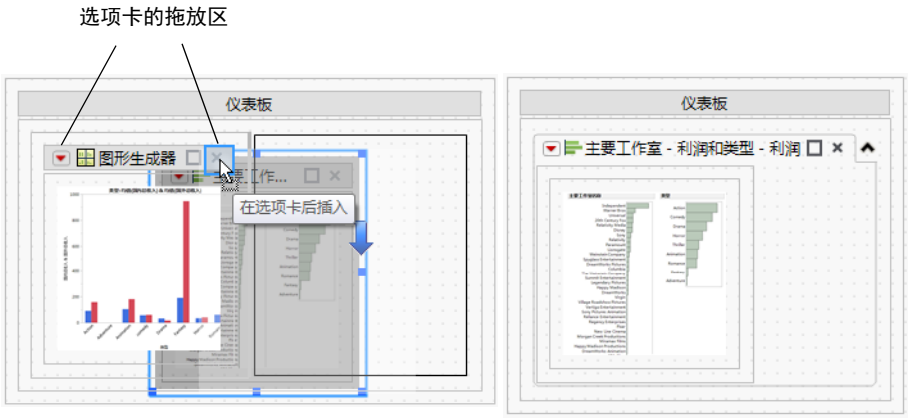


图 11.5 显示了将数据表放入报表底部边框内的示例。

要在报表**下方**而不是其内部显示数据表，请将“数据表”框拖到报表下方的仪表板上。该框将与报表的底部边框内部对齐。

图 11.5 在报表内显示数据表

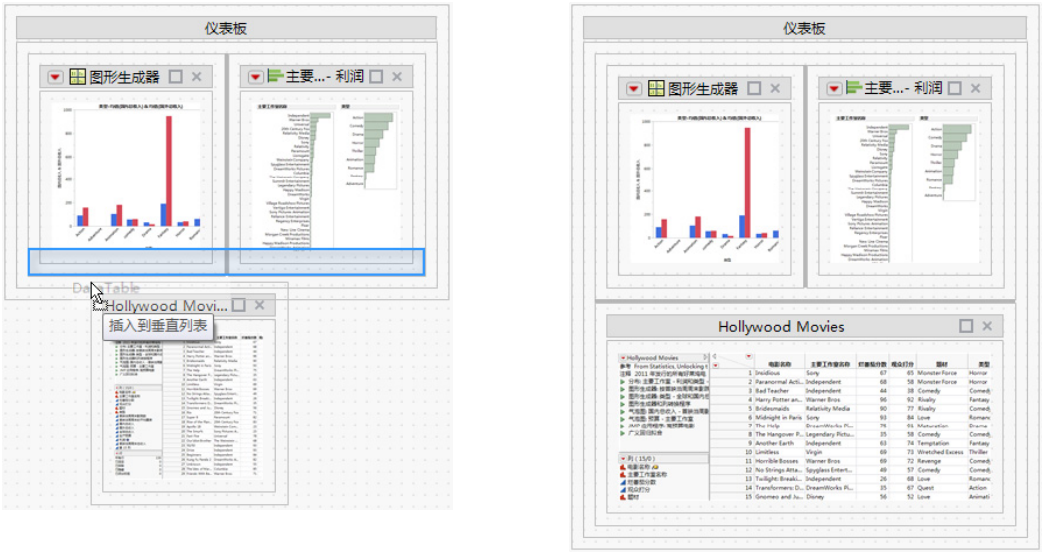
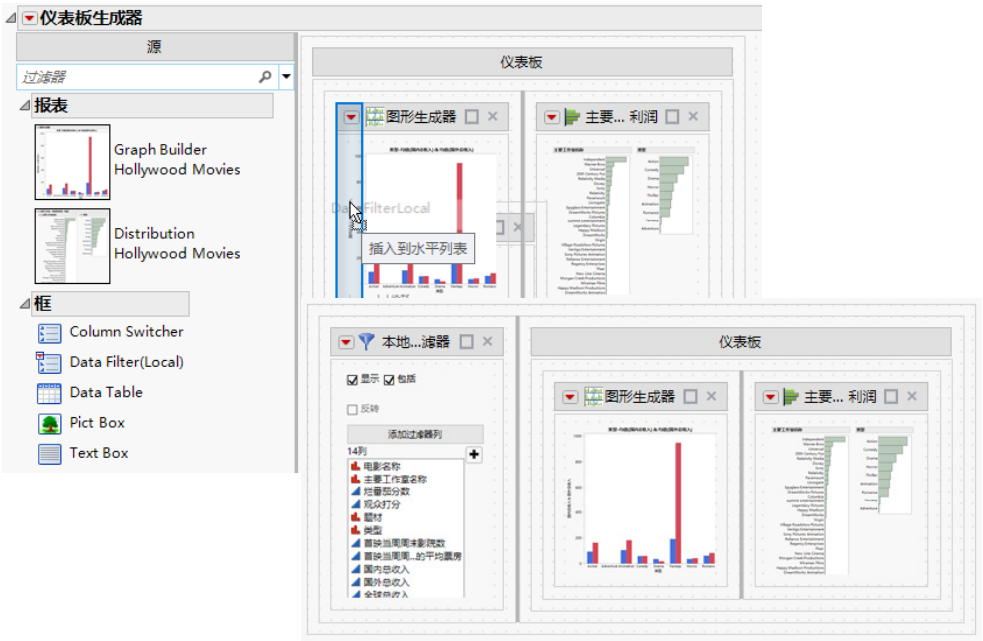


图 11.6 显示了一个数据过滤器，它显示在报表左侧。

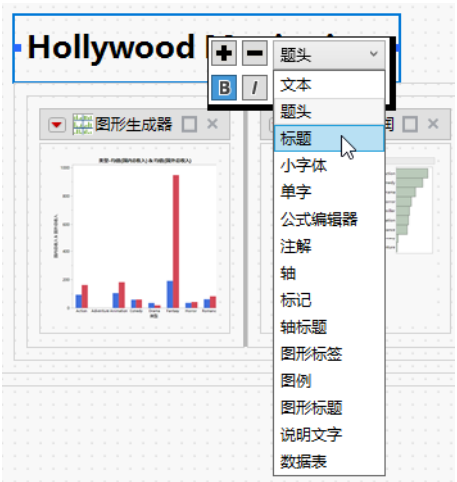
图 11.6 向报表左侧添加数据过滤器



设置仪表板中文本的格式

编辑仪表板时，您可以设置文本框中文本的格式。将文本框拖至仪表板并点击文本框中央。格式设置选项随即显示。

图 11.7 文本框的格式设置选项



预览仪表板

编辑仪表板时，您可以通过从“仪表板生成器”红色小三角菜单中选择**预览模式**来查看最终仪表板的预览。预览仪表板在检验交互式元素（如列切换器、数据过滤器和选项卡式报表）时尤其有帮助。

注意：若您在预览模式中调整窗格大小，您的更改不会保存。

提示：

- 列切换器最初显示表中的所有列，但不会主动更改仪表板或应用程序中的报表。
- 您可以切换到预览模式以执行以下操作：
  - 选择要在所有报表中切换的初始列。
  - 删除不希望显示在列切换器中的任何列，包括报表中使用的任何其他列。
- 运行仪表板或应用程序时，列切换器会自动连接到仪表板中使用相同数据表的所有平台。

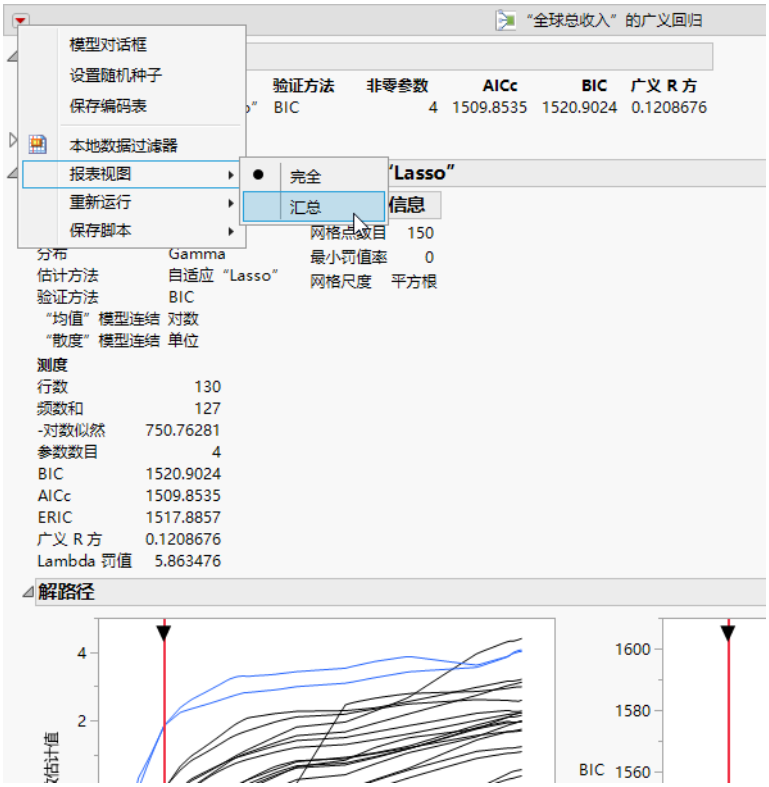
查看运行的仪表板

查看运行的仪表板时，报表中会提供与您在 JMP 平台中看到的相同的红色小三角选项。“仪表板生成器”还提供若干其他选项。

查看报表汇总

包含图形和统计量的报表可能会增长得很大。要在运行报表时仅查看图形，请选择报表顶部的红色小三角，然后选择**报表视图 > 汇总**。图 11.8 中显示的报表最初显示的是参数估计值。在“汇总”视图中只显示了图形。

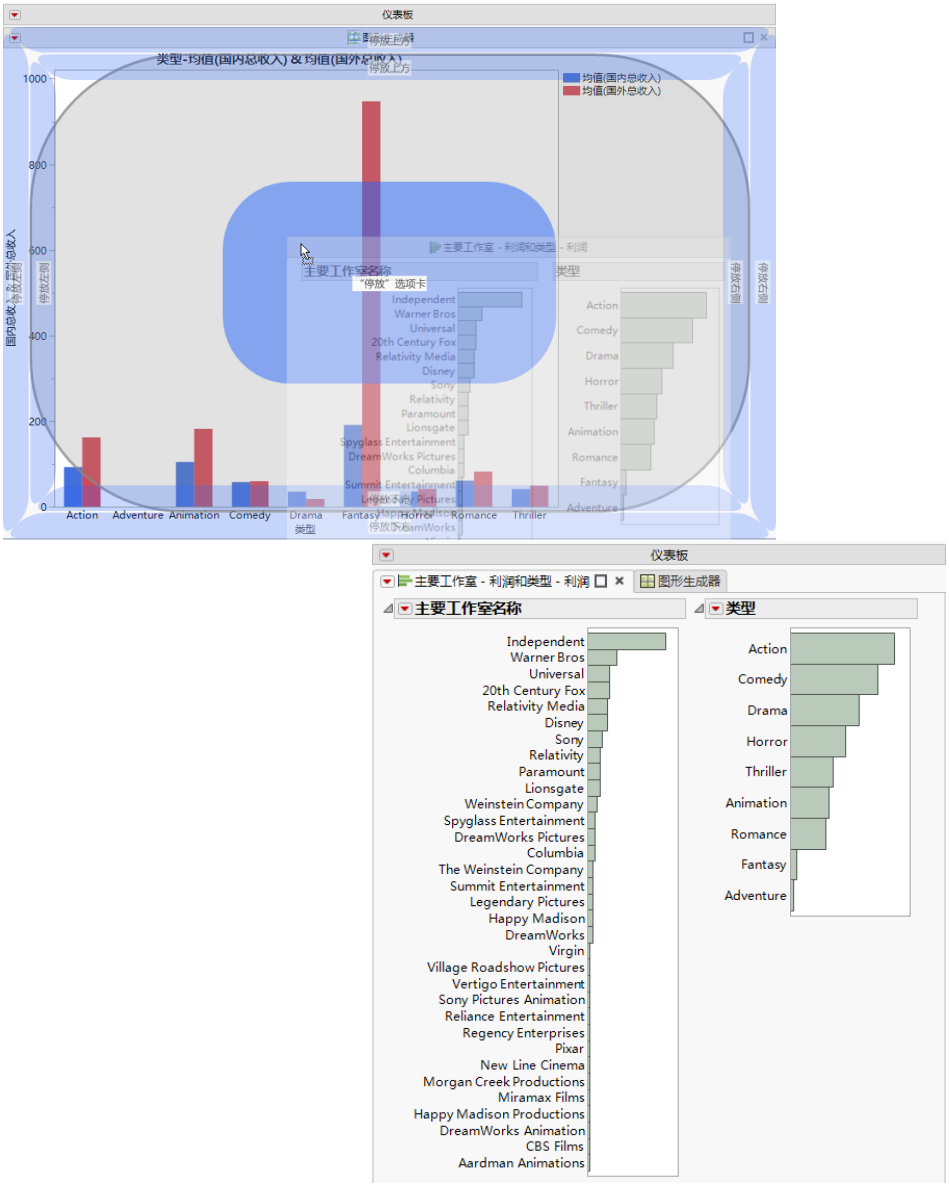
图 11.8 查看报表汇总



在运行的仪表板中重新排列报表

通过将报表拖入突出显示的拖放区，您可以重新排列运行的仪表板中的报表。拖放区指示您可以放置报表的位置。图 11.9 显示将报表移入选项卡的示例。“分布”报表将放在“停放选项卡”拖放区内，以便放入第一个选项卡。

图 11.9 在运行的仪表板中创建选项卡式报表

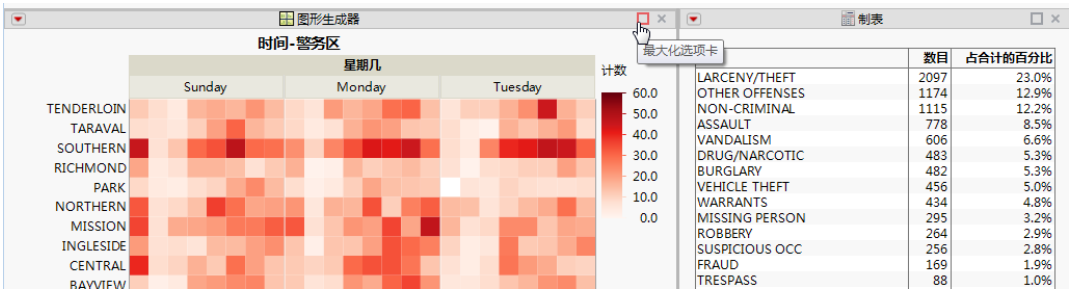


要删除选项卡式报表，请点击该报表的中部并将其拖到该选项卡之外。

在运行的仪表板中将报表最大化

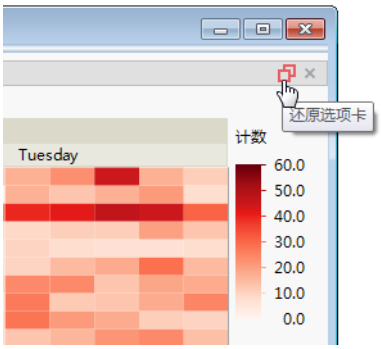
要在运行仪表板时仅查看特定报表，请选择该报表的**最大化选项卡**框。

图 11.10 将报表最大化



要将报表还原为其原始大小，请选择**还原选项卡**框。

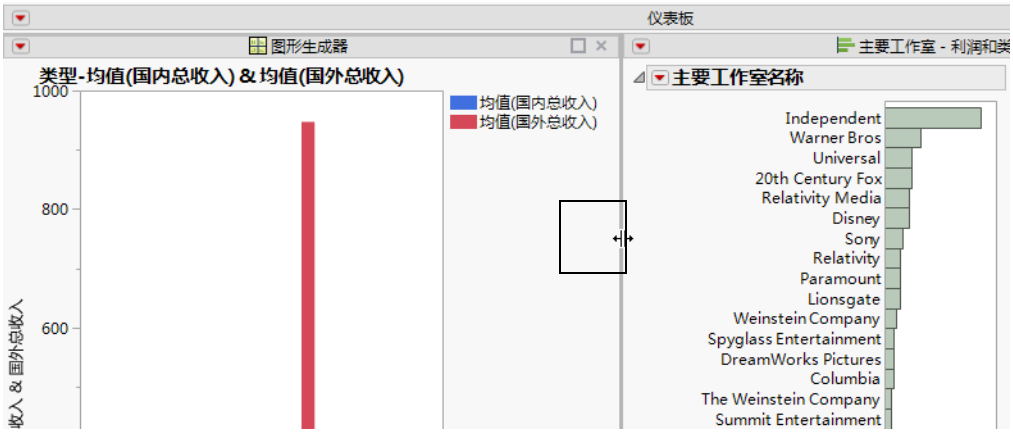
图 11.11 恢复最大化报表



在运行的仪表板中调整报表大小

显示在仪表板中的报表之间的灰色粗线称为**拆分线**。拖动拆分线以调整窗格大小。若窗格太宽，无法显示封闭的报表，该窗格将关闭。从仪表板的外边缘拖动拆分线可再次查看所有报表。图 11.12 显示了折叠和展开的报表。

图 11.12 在运行的仪表板中调整窗格大小



您还可以拖动图形的边缘。图形将加宽到窗口和其他报表所允许的宽度。

与在 JMP 之外工作的用户共享报表

通过将仪表板另存为交互式 HTML，您可与在 JMP 之外工作的用户共享报表。首先，点击“仪表板生成器”红色小三角并选择**运行仪表板**以运行该仪表板。在运行的仪表板中，选择**文件 > 另存为**，然后从该列表中选择**数据交互式 HTML**。报表将保存在同样包含数据的网页中。将该网页通过电子邮件发送给用户，或在网址上发布该网页。然后用户可以如同在 JMP 中一样探索数据。请参见“将报表共享为交互式 HTML”。

编辑 JMP 平台脚本

仪表板包含 JSL 脚本，您可以在“仪表板生成器”中编辑该脚本。例如，您可以编辑“多元”平台的脚本以显示散点图矩阵或使用不同列。

在仪表板右击平台对象并选择**编辑平台脚本**。该脚本将在脚本编辑器窗口中打开。

编辑脚本并点击“确定”后，工作区上的对象将更新。

示例：组合窗口以创建仪表板

您可以通过在 JMP 中合并若干打开的窗口来快速创建仪表板。组合窗口提供了一些选项，不仅支持查看汇总的统计量，还包含一个选择内容过滤器。

1. 选择**帮助 > 样本数据库**，然后打开 Birth Death.jmp。
2. 运行 “分布”脚本和 “二元”表脚本。
3. 从其中一个报表窗口中，选择**窗口 > 组合窗口**。

   “组合窗口”窗口随即出现。

提示：在 Windows 系统中，您还可以从 JMP 窗口右下角的 “排列菜单” 选项中选择 “组合窗口”。

4. 选择**汇总报表视图**显示图形并省略统计报表
5. 在 “组合” 列中，选择 **Birth Death - 二元 : 出生率 - 死亡率**和 **Birth Death - 分布**。
6. 在 “过滤依据” 列中，选择 **Birth Death - 分布**。

图 11.13 “组合窗口” 选项

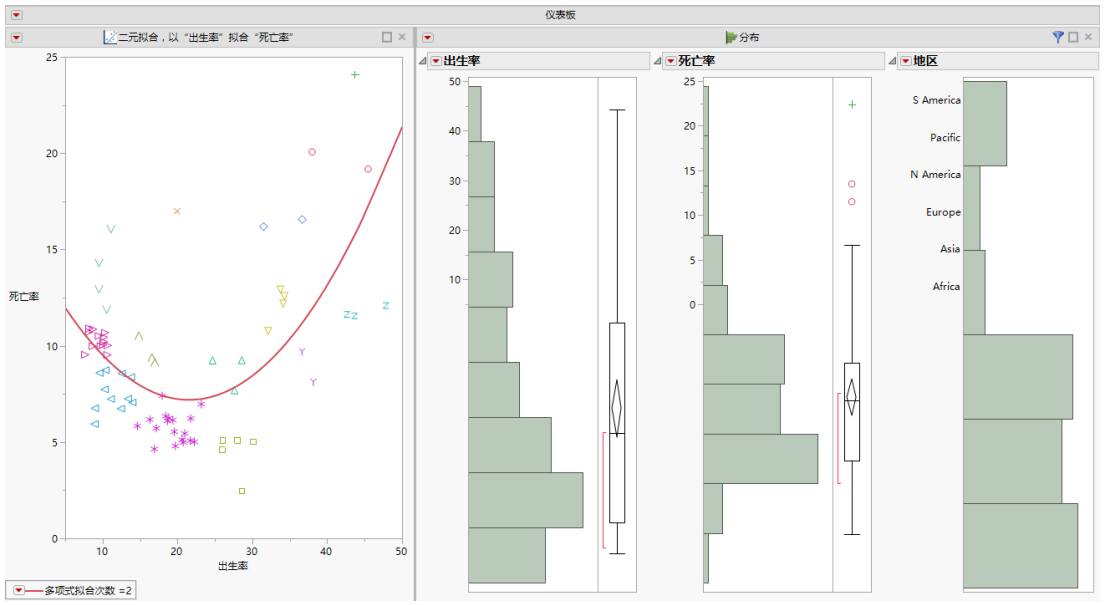
7. 点击**确定**。

   两个报表组合到一个窗口中。请注意 “分布” 报表顶部的过滤器图标 。选择其中一个直方图中的某个直条后，“二元” 图形中的对应数据会被选定。

注意：

- 要在 Windows 上组合报表，您还可以从 JMP 窗口右下角的 “排列菜单” 选项中选择 “组合窗口”。
- 在 “组合窗口” 窗口中，选择**汇总报表视图**仅查看报表中的图形并省略统计学内容。

图 11.14 组合窗口



示例：向仪表板添加“选择内容过滤器”

在包含若干报表的仪表板中，您可以选择某个报表中的数据，然后仅查看同一窗口中包含的其他报表中的这一数据。本例显示如何在编辑仪表板时向主报表添加选择内容过滤器。

打开报表并创建仪表板

1. 选择**帮助 > 样本数据库**，然后打开 Hollywood Movies.jmp。
2. 要创建报表，运行名为 “分布 : 主要工作室 - 利润和类型 - 利润”、“图形生成器 : 类型 - 全球和国内总收入”以及 “气泡图 : 国内总收入 - 首映当周影院数”的表脚本。
3. 从任意窗口中，选择**文件 > 新建 > 仪表板**。

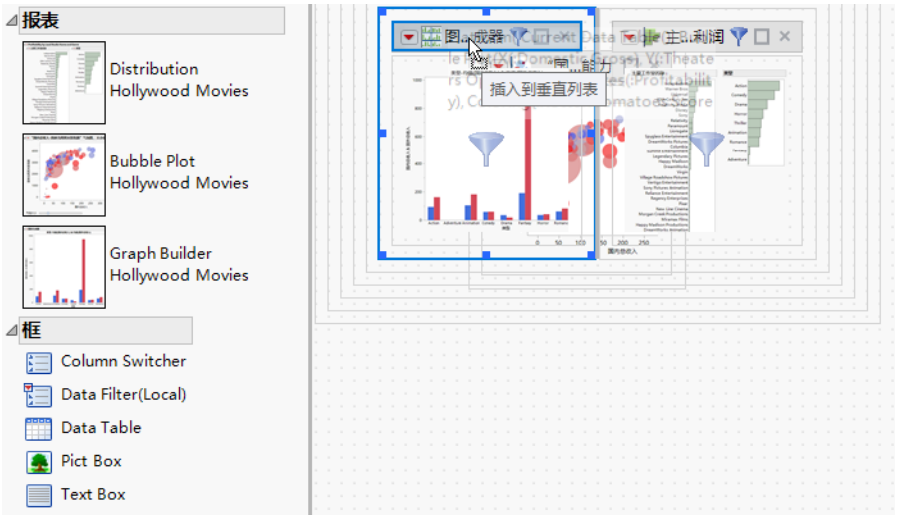
   随即出现常见布局的模板。
4. 选择 **2x1 仪表板**模板。

   工作区中随即出现可容纳两个报表的框区。
5. 在 “报表”列表中，双击 “图形生成器和分布”报表缩略图将其放入仪表板。

添加和使用选择内容过滤器

1. 右击左报表中央（图形生成器）并选择**用作选择内容过滤器**。
2. 要添加另一个报表作为第二个选择内容过滤器，请点击“气泡图”缩略图，并将其拖入“图形生成器”图形周围的蓝线内。

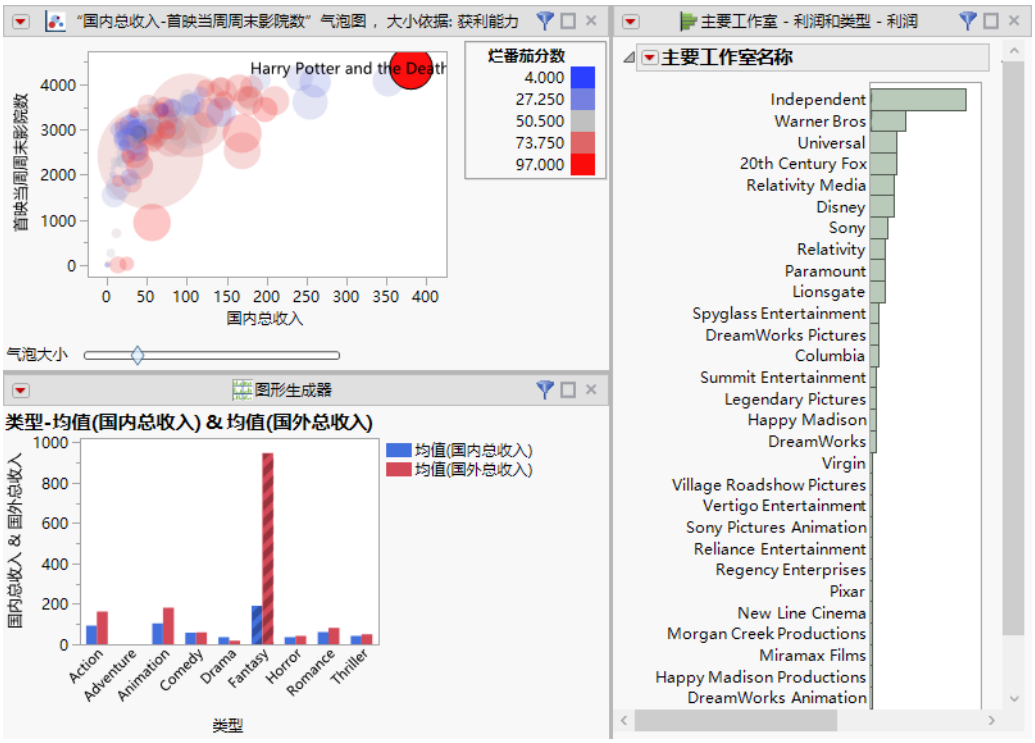
图 11.15 将气泡图拖入图形生成器图形



过滤器图标将显示在这两个缩略图上。

3. 点击 “仪表板生成器”红色小三角并选择**预览模式**。
4. 选择左报表中 “Fantasy”类型的直条之一。

图 11.16 过滤数据



在右报表中，您可以看到 Warner Bros 和独立制片公司制作了大多数“科幻”影片。在底部报表中，您可以看到 Harry Potter 是最为赚钱的 “科幻”影片。

示例：创建 JMP 查询仪表板和插件

假定您使用“JMP 查询生成器”将两个数据表的部分内容导入 JMP。仪表板显示数据报表。您想要创建一个用于查询数据并在仪表板上显示更新报表的插件。

生成查询

1. 选择**帮助 > 样本数据库**，然后打开 Hollywood Movies.jmp。
2. 选择**表 > JMP 查询生成器**。
3. 点击**生成查询**。
4. 在 “可用列”列表中，双击 **t1. 电影名称**、 **t1. 烂番茄分数**、 **t1. 观众打分**和 **t1. 首映当周周末影院数**，将其添加至 “已包括的列”选项卡。
5. 点击**运行查询**。

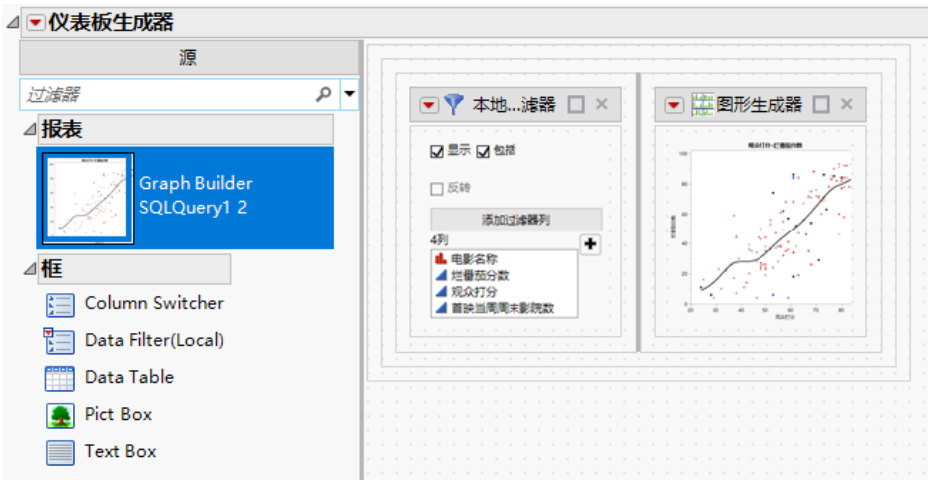
   查询的数据将出现在新的数据表中。请注意 “源”表脚本。打开仪表板时， JMP 运行该“源”脚本以查询 Hollywood Movies.jmp 中的数据。

6. 在新数据表中，运行**图形生成器 : 按首映当周周末影院数划分的观众打分**表脚本以创建报表。

创建仪表板

1. 从报表窗口中，选择**文件 > 新建 > 仪表板**。
2. 选择**空白仪表板**模板。
3. 双击左窗格中的 Data Filter(Local) 框将其添加至仪表板。
4. 双击左窗格中的 “Graph Builder” 报表，将其放在本地数据过滤器旁边。

图 11.17 Hollywood Movies 仪表板



创建和运行插件

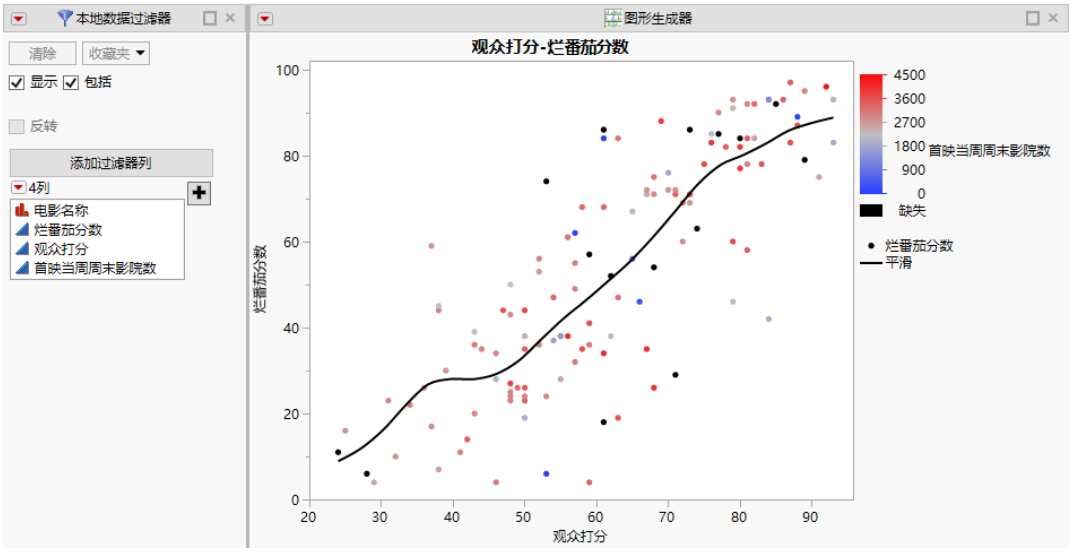
1. 点击 “仪表板生成器” 红色小三角并选择**保存脚本 > 至插件**。

   “插件生成器” 随即打开。
2. 在 “插件” 名称旁边，键入 Hollywood Movies Dashboard （插件文件名）。
3. 点击**菜单项**选项卡并在**菜单项名称** （插件菜单项的名称）旁边键入 Hollywood Movies Dashboard。
4. 点击**保存**并将插件保存到您的桌面。

   插件将被保存并安装到您的 “插件” 菜单中。
5. 点击 “插件生成器” 窗口中的**关闭**。
6. 从 JMP 主菜单中，选择**插件**并选择 Hollywood Movies Dashboard。

   随之查询 Hollywood Movies.jmp 数据表，并且该仪表板打开以显示本地数据过滤器和更新的报表。

图 11.18 Hollywood Movies 插件



示例：从两个数据表创建仪表板

可以从多个数据表创建仪表板中的图形或报表。用于创建仪表板上第一个图形的数据表被默认视为当前数据表。这样即可支持用户灵活地使用任何当前数据表创建图形或报表。仪表板编制者还可以对路径进行硬编码。指定名称还赋予了仪表板编制者对于所用的数据表的控制权。

1. 选择**帮助 > 样本数据库**，打开 Big Class.jmp 并运行“二元”脚本。
2. 选择**帮助 > 样本数据库**，打开 Hollywood Movies.jmp 并运行 “分布”脚本。
3. 从任意 JMP 窗口中，选择**文件 > 新建 > 仪表板**。
4. 在 “仪表板生成器”中，选择 **2 x 1 仪表板**模板。
5. 双击 “分布”图形首先将其添加到仪表板中。

   默认情况下，该图形使用当前数据表。在本例中，存储在 Samples/Data 文件夹中的 Hollywood Movies.jmp 是当前数据表。
6. 双击 “二元”图形将其添加到仪表板中。

   系统提示您：是确定第一个图形使用当前数据表，还是希望对仪表板中的数据表的路径进行硬编码 （按名称引用表）。
7. 选择**按名称引用表**确保指定的数据表始终用于创建该图形。
8. 点击**确定**。

   在本例中， Samples/Data 文件夹中的 Hollywood Movies.jmp 将始终用作第一个图形的当前数据表，不论当前打开的是哪一个数据表。

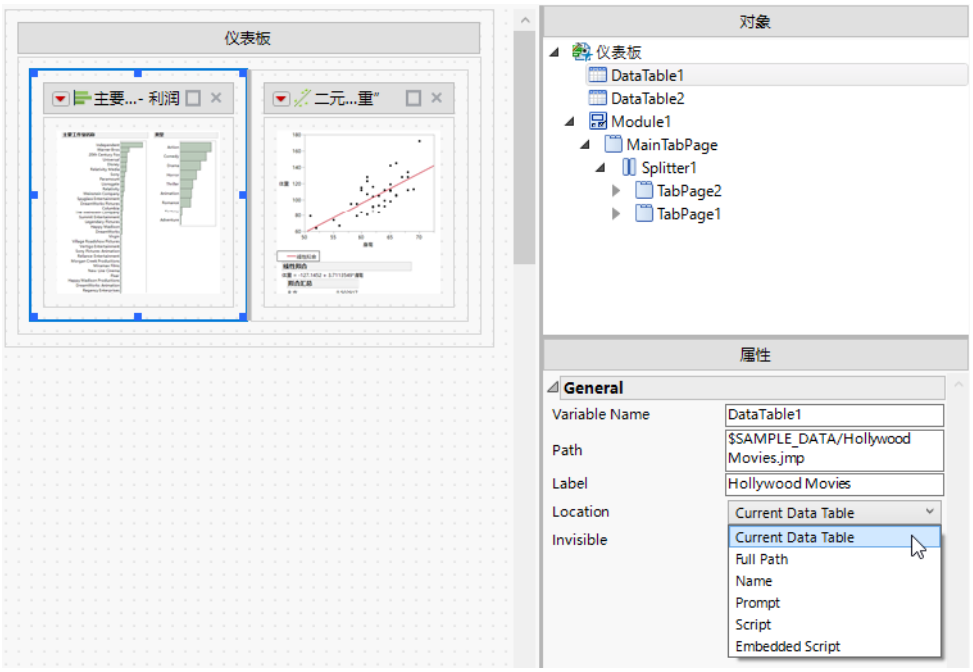
提示：有时，运行仪表板时找不到指定的数据表。尤其是在仪表板移至另一台计算机时，这种情况尤其常见。您可以在运行仪表板之前打开该数据表，或是运行仪表板并在提示时选择该数据表。

更改数据表的位置

指定数据表（当前数据表或名称）的位置后，您可以在仪表板的“对象”窗格和“属性”窗格中更改设置。

1. 点击“仪表板生成器”红色小三角并选择**显示属性**。

图 11.19 “对象”窗格和 “属性”窗格



‘

2. 在 “对象”窗格中，选择 DataTable1。
3. 在 “属性”窗格中，按图 11.19 所示更改位置。

   例如，若将 “名称”位置改为 “当前数据表”，仪表板上的选定图形将基于当前数据表。

注意：“当前数据表”的“位置”值仅在您运行仪表板时才适用。编辑仪表板时，“仪表板生成器”使用完整路径查找该数据表。

第 12 章

对菜单和工具栏进行个性化设置

定制 JMP

本章介绍如何定制 JMP 菜单和工具栏，以便仅显示所需的命令。例如，若不使用 SAS，则可以删除“文件”菜单中的 SAS 选项。或者，也可以为“运行脚本”命令指定一个快捷键。

定制 JMP 还支持您为具有特殊关注点的组设置 JMP。若某个组不设计实验，则可以从 JMP 菜单中删除“实验设计”。

注意：升级到 JMP 16 订阅时，将保留对首选项的更改。

目录

在 Windows 上对 JMP 工具栏和菜单进行个性化设置 563
对工具栏进行个性化设置 564
对主菜单栏进行个性化设置 567
创建菜单项和工具栏按钮 567
设置定制工具栏和按钮的用户权限 571
重新排列工具栏 571
复制和粘贴菜单、菜单项、工具栏和按钮 572
重新排列定制菜单、菜单项和按钮 573
删除定制项 575
显示或隐藏定制项 576
导入定制设置 576
从 JMP 中删除定制设置 577
在 macOS 上对 JMP 工具栏进行个性化设置 578
在 macOS 上对 JMP 菜单进行个性化设置 579
为插件定制 JMP 菜单和工具栏 580

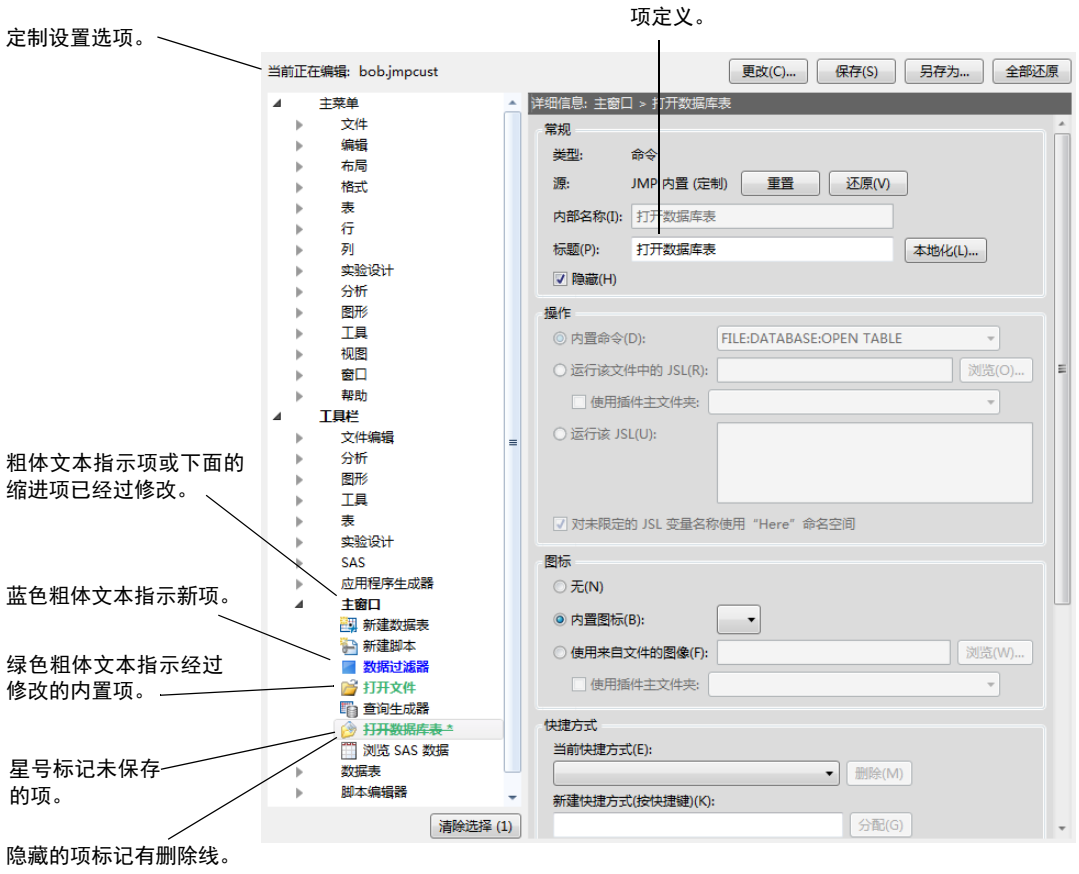
在 Windows 上对 JMP 工具栏和菜单进行个性化设置

工具栏包含可执行命令的按钮以及帮助您组织这些按钮的分隔线。在 Windows 中，有多种方式可以对工具栏进行个性化。例如，您可以创建、重命名、隐藏和修改工具栏或按钮。还可以重新排列或删除所创建的工具栏和按钮。

菜单包含命令、子菜单和分隔符。和工具栏、按钮一样，您可以创建、重命名、隐藏和修改菜单；还可以重新排列或删除所创建的菜单和菜单项。

在“菜单编辑器”中对工具栏和菜单进行个性化。修改的项的名称经过格式设置，可指示创建、修改或隐藏了哪些项以及哪些项未保存。

图 12.1 “菜单编辑器”选项



定制设置选项

要修改的定制设置

当前用户 只有您才能看到更改。定制文件名为 usercust.jmpcust。手动修改该文件可能引起意外结果。

所有用户 在计算机上打开 JMP 的所有用户都能看到更改。定制文件名为 admincust.jmpcust。

JMP 插件 那些使用选定的 JMP 插件并将 JMP **插件定制设置**选作起始设置的用户才能看到更改。若修改已禁用插件的定制设置，除非您启用该插件，否则看不到这些更改。

其他文件 将选定的 .jmpcust 文件选作定制设置后，更改随即显示。需要在文本编辑器中创建该文件。

新建 选择这一新定制设置后，更改随即显示。（执行更改之后，点击**保存**可在选定文件夹中命名这个新文件。）

起始设置

JMP 内置项 选择的定制设置基于默认 JMP 工具栏和菜单。（应用于所有定制设置，且无法取消选择。）

所有用户的定制设置 选择的定制设置基于所有用户定制设置。（可用于其他文件或新建定制设置。无法从当前用户定制设置取消选择。）

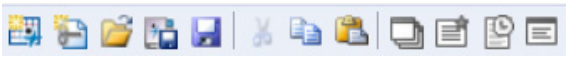
JMP 插件定制设置 选择的定制设置基于 JMP 插件定制设置。插件定制设置通常随插件安装。（可用于除所有用户之外的所有定制设置。）

当前用户的定制设置 选择的定制设置基于当前用户定制设置。（仅可用于其他文件和新建定制设置。）

对工具栏进行个性化设置

JMP 提供可供您向其中添加新按钮的工具栏，您也可以自行创建工具栏。默认情况下，JMP 窗口类型确定显示哪些栏。例如，“文件编辑”工具栏会在所有窗口中显示。在数据表中，“数据表”工具栏也会显示。

项目 显示项目图标。



文件编辑 显示用于 “文件”菜单和 “编辑”菜单中的命令的图标。

分析 显示用于 “分析”菜单中的常用命令的图标。

图形 显示用于 “图形”菜单中的命令的图标。

工具 显示您可以点击并用作光标的工具的图标。在某些窗口中，工具栏会隐藏。取代它的是一条蓝线。悬停在蓝线上方时可显示工具栏。

数据表列表 显示打开的数据表的列表。您可以在该列表中选择某个数据表，使其成为当前表。请注意，当前表不一定是最前面的窗口。要将表 （或任何窗口）移至最前，请从 “窗口”菜单的列表中选择其名称。

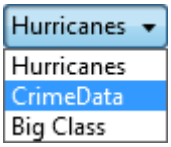


表 显示用于 “表”菜单中的命令的图标。

实验设计 显示用于 “实验设计”菜单中的命令的图标。

SAS 显示分别具有以下功能的图标：访问和浏览 SAS 数据和文件夹、打开 “SAS 查询生成器”、创建新的 SAS 程序、向 SAS 提交数据、打开 SAS 日志窗口和输出窗口，以及查看服务器连接。

应用程序生成器 显示 “应用程序生成器”中的命令图标。

主窗口 显示“文件”菜单中常用的图标。

数据表 显示在数据表中执行常见任务的图标，如打开 “数据过滤器” 、“拟合模型”和 “重新编码”窗口。

脚本编辑器 显示运行、调试和重新格式化脚本的图标。

报表 显示打开 “本地数据过滤器”、“列切换器”以及将脚本保存到数据表的图标。

为默认工具栏中未包含的常用命令创建您自己的工具栏。可以将命令指定给按钮，并且可以添加其他属性，如快捷键、图标和 JMP 脚本语言 (JSL) 脚本。

若要基于现有工具栏创建工具栏，请复制现有工具栏，然后根据需要更改设置。请参见第 572 页的“复制和粘贴菜单、菜单项、工具栏和按钮”。

第 1 步：创建工具栏

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “工具栏”列表显示在左侧。
2. （可选）更改定制设置，控制可以查看定制内容的用户。请参见第 564 页的“定制设置选项”。
3. 在 “菜单编辑器”仍保持打开期间，右击工具栏之一，如**文件编辑**。

   选择任意工具栏，因为所有新工具栏都显示在 “工具栏”列表的末尾。
4. 选择**新建工具栏**。

   该工具栏插入列表的末尾。该工具栏包含无标题按钮，因为所有工具栏都必须至少包含一个按钮。

第 2 步：指定标题和内部名称

工具栏具有若干基本属性：

- 标题显示在**视图 > 工具栏**列表中，可通过该列表显示或隐藏工具栏。为每个工具栏指定唯一名称。JMP 将在关闭并重新打开“菜单编辑器”后合并同名的工具栏。
- 内部名称，不区分大小写。JMP 根据项的内部名称而不是项标题来确定项的位置。

要指定这些工具栏属性，请执行以下步骤：

1. 在“菜单编辑器”仍保持打开期间，选择工具栏。

   “常规”属性显示在右侧。
2. 在**内部名称**旁边输入内部名称。
3. 在**标题**框中，为工具栏或按钮输入唯一名称。
4. （可选）要指定标题的译文，请执行以下操作：
   - 为要定制的项点击**本地化**按钮。
   - 选择语言。

– 在**文本**旁边输入译文。
– 点击**确定**。

5. （可选）点击**隐藏**可在创建工具栏后保持工具栏处于隐藏状态。
6. 按第 567 页的 “创建菜单项和工具栏按钮”中所述定制无标题按钮。
7. 点击**保存**保存所做更改。

对主菜单栏进行个性化设置

将您自己的菜单项添加至 JMP 的主菜单栏中。这些菜单显示在**文件**、**编辑**和**表**之类的菜单的前面或后面。菜单中自动插入无标题命令，因为所有菜单都必须至少包含一个命令。

若要基于现有菜单创建主菜单，请复制现有菜单，然后根据需要更改设置。请参见第 572 页的 “复制和粘贴菜单、菜单项、工具栏和按钮”。

创建主菜单

1. 选择**视图 > 定制 > 菜单和工具栏**。
   “主菜单”列表显示在左侧。
2. （可选）更改定制设置，控制可以查看定制内容的用户。请参见第 564 页的“定制设置选项”。
3. 右击要在其旁边添加新菜单的菜单。
4. 选择**之前插入**或**之后插入**。
   将向列表中添加无标题菜单和菜单项。
5. 完成第 566 页的 “第 2 步：指定标题和内部名称”以重命名主菜单。
6. 按第 567 页的 “创建菜单项和工具栏按钮”中所述定制菜单项。
7. 点击**保存**保存所做更改。

创建菜单项和工具栏按钮

菜单包含以下菜单项：

- 命令是您点击以执行命令的项（如数据表和脚本）。
- 子菜单是您点击以显示更多菜单选项的菜单项。例如，“新建”就是“文件”顶级菜单内的子菜单。“新建”之所以是一个子菜单，还因为它包含命令（如“数据表”和“脚本”）。
- 分隔符 () 是对命令和子菜单进行划分或分组的线条。

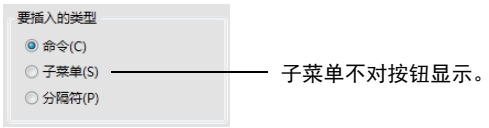
工具栏上的按钮被视为一种命令，所以您还可以通过完成以下步骤来创建新的工具栏按钮。

若要基于现有项创建菜单项或按钮，请复制现有项，然后根据需要更改设置。请参见第 572 页的 “复制和粘贴菜单、菜单项、工具栏和按钮”。

第 1 步：创建新的菜单项或按钮

1. 选择**视图 > 定制 > 菜单和工具栏**。
   “菜单编辑器”随即显示。
2. （可选）更改定制设置，控制可以查看定制内容的用户。请参见第 564 页的“定制设置选项”。
3. 在要添加菜单项或按钮的位置右击。
   随即显示可能位置的列表。
4. 选择**之前插入**或**之后插入**。
   随即显示 “指定类型”窗口。

图 12.2 “指定类型”窗口



5. 执行以下操作之一：
   - 要创建按钮，请选择**命令**。
   - 要在菜单内创建菜单，请选择**子菜单**。（不可用于按钮。）
   - 要在菜单项或工具栏按钮之间创建分隔线，请选择**分隔符**。
6. 点击**确定**。
   随即添加新的无标题按钮、子菜单或分隔符。

第 2 步：指定标题、工具提示和内部名称

按钮具有若干基本属性：

- 标题显示在“视图 > 工具栏”列表中，可通过该列表显示或隐藏工具栏。为每个工具栏指定唯一名称。JMP 将在关闭并重新打开“菜单编辑器”后合并同名的工具栏。
- 将光标置于菜单项或按钮上方时，即显示工具提示。
- 内部名称，不区分大小写。JMP 根据项的内部名称而不是项标题来确定项的位置。

子菜单项包含的许多“常规”属性与按钮相同，但子菜单项不包括工具提示。

要指定这些属性，请执行以下步骤：

1. 在“菜单编辑器”仍保持打开期间，选择菜单项或按钮。
   “常规”属性显示在右侧。
2. 在**内部名称**旁边输入内部名称。
3. 在标题框中，为菜单项或按钮输入唯一名称。
4. 在提示框中输入按钮说明。

5. （可选）要指定标题或提示的译文，请执行以下操作：
   - 为要定制的项点击**本地化**按钮。
   - 选择语言。
   - 在文本旁边输入译文。
   - 点击**确定**。
6. 按以下过程所述定制无标题按钮或菜单项。

第 3 步：分配功能

点击菜单项或按钮可执行预定义命令或运行 JSL 脚本。脚本可存储在单独的文件中，您也可以在“菜单编辑器”的运行该 JSL 区域中输入 JSL。若用户有权访问网络之类的中心位置，通常您希望从该位置运行脚本。这样还可以简化授权用户访问更新脚本的过程。若用户不具有访问权限，则可以将 JSL 存储在菜单项或按钮定义中。

要将菜单项或按钮关联到插件 JSL 脚本，您有两种选择：

- 选择**运行该文件中的 JSL**，然后浏览查找外部文件。
- 键入指向 JSL 脚本的相对路径，然后从**使用插件主文件夹**列表中选择该插件。例如，以下命令从选定的插件运行 mds_application.jsl 脚本：

  ```
  $ADDIN_HOME(com.jmp.mdswithr)/mds_application.jsl
  ```

此外，若在内部或外部 JSL 脚本中包含命名空间，则可以选择**对未限定的 JSL 变量名称使用“Here”命名空间**。有关命名空间的详细信息，请参见《Scripting Guide》。

要分配该功能，请执行以下步骤：

1. 在“菜单编辑器”仍保持打开期间，选择要修改的菜单项或按钮。

   “操作”属性显示在右侧。
2. 选择要执行的操作。
3. 要运行内部 JSL 脚本，请删除**运行该 JSL** 区域中的占位符文本 print(“Not implemented.”);，然后输入该 JSL。
4. （可选）若内部或外部 JSL 脚本包含命名空间，请选择**对未限定的 JSL 变量名称使用“Here”命名空间**。
5. 点击**保存**保存所做更改。

第 4 步：在菜单项或按钮上显示图标（可选）

新菜单项的旁边没有图标。新按钮在工具栏上显示为蓝色方块。您可以为菜单项或按钮分配描述性的图标。图形可位于您的计算机上或位于插件文件夹中。支持 .ico、.png、.jpg 和 .bmp 图形。

向按钮分配图标尤为重要。若不分配，用户只能通过查看按钮的工具提示才能了解该按钮的作用。

要在菜单项或按钮上显示图标，请执行以下步骤：

1. 在“菜单编辑器”仍保持打开期间，选择菜单项或按钮。

   “图标”属性显示在右侧。

2. 选择以下选项之一：

   无 选择该选项可显示一个蓝色方块而不是图标。

   内置图标 选择该选项可显示 JMP 提供的图标，然后从列表中选择所需图标。

   使用来自文件的图像 选择该选项可显示您创建的图像。点击**浏览**，选择所需图形，然后点击**确定**。

   使用插件主文件夹 选择该选项可显示菜单项左侧的插件中的图标。选择该复选框后，选择插件，点击**浏览**，然后选择该图标。指向该图像的路径显示在**使用来自文件的图像**字段中。图像格式可以是 ICO、PNG、JPG 或 BMP 图形。

3. 点击**保存**保存所做更改。

第 5 步：分配快捷键（可选）

使用快捷键可执行某项操作，这样您就不必再选择菜单项或点击工具栏上的按钮。若按钮工具栏未显示，而您仍想执行相应操作时，快捷键选项也会很有帮助。

快捷方式以 Ctrl、Ctrl+Shift、Ctrl+Alt 开头，以键盘上的数字、字母或符号结尾。也同样支持功能键（如 F1）。JMP 中已经分配了许多 Ctrl + **字母**和功能快捷方式，但您仍可以根据需要为按钮重新分配快捷方式。

快捷键显示在菜单中每个菜单项的旁边。

要分配快捷键，请执行以下步骤：

1. 在“菜单编辑器”仍保持打开期间，选择菜单项或按钮。

   “快捷方式”属性显示在右侧。

2. 执行以下操作之一：
   - 要分配新的快捷方式，请在**新建快捷方式**区域中点击，然后按快捷键。若按键组合随后显示在**当前分配至**框中，请按另一组快捷键组合。（不必在按快捷键之前删除键击。）
   - 要更改以前指定的快捷方式，请点击**删除**，在**新建快捷方式**区域中点击，然后按快捷键。

3. 点击**分配**。

   快捷方式出现在**当前快捷方式**列表中。

4. 点击**保存**保存所做更改。

要删除快捷方式，请选择该快捷方式，然后点击**删除**。

设置定制工具栏和按钮的用户权限

使用“JMP 菜单编辑器”可指定哪些用户可以看到定制的工具栏和按钮。您所做的更改保存在名为**定制设置**的扩展名为 .jmpcust 的文本文件中。默认情况下，只有当前用户才能看到您定制的内容；登录到该计算机并打开 JMP 的其他用户看不到您个人的工具栏和菜单。

所有定制设置都包含内置 JMP 按钮和菜单。您还可以将定制设置基于其他定制设置。例如，您可以创建新的定制设置，用来显示 JMP 插件菜单和工具栏以及您所做的修改。

修改**当前用户**定制设置时，JMP 将创建该设置的备份文件。该文件位于 Windows 上的 Users 文件夹的 JMP 或 JMPPro 文件夹中。

C:/Users/< 用户名 >/AppData/Roaming/SAS/JMP/16/

修改**所有用户**定制设置时，JMP 将创建该设置的备份文件。该文件位于 Windows 上的 All Users 文件夹的 JMP 或 JMPPro 文件夹中。

C:/Users/All Users/SAS/JMP/16/

尝试浏览查找 All Users 文件夹时，该文件夹并不显示，所以您必须在“Windows 资源管理器”的地址字段中键入路径。

注意：要在“Windows 资源管理器”中查找定制设置文件，请在“Windows 资源管理器”的“文件夹选项”中显示所有隐藏文件。详细信息，请参见您的操作系统文档。

要更改定制设置，请执行以下步骤：

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “菜单编辑器”随即显示。

2. 点击**更改**。

   尝试更改定制设置时，若未保存对选定设置的更改，系统会提示您保存更改。点击**保存更改**。

3. 选择要修改的定制设置区域中的定制设置。（默认情况下，更改仅适用于**当前用户**定制设置。）。第 564 页的 “定制设置选项”说明了各种选项。

4. （可选）要更改选定的定制设置所基于的设置，请在起始设置区域中选择该设置。（不适用于**所有用户**定制设置。）第 564 页的 “定制设置选项”说明了各种选项。

5. 点击**确定**。

重新排列工具栏

在 Windows 上，有两种重新排列工具栏的方式：

- 要为当前用户快速重新排列工具栏，请将该工具栏拖至相邻工具栏的上方、下方、左侧或右侧。

- 要指定新位置而不是拖动工具栏，请右击该工具栏，然后从位置菜单中选择一个位置。该方法支持您将工具栏移至窗口的顶部或底部。您还可以将工具栏竖立在窗口的左侧或右侧。

  在某些窗口中，工具栏默认处于隐藏状态。指定新位置会使工具栏显示在该类型的所有窗口中。在 Windows 上，您可以更改工具栏首选项，以便始终显示工具栏。请参见“JMP 首选项”一章中第 608 页的 “Windows 首选项”

拖动工具栏

1. 指向工具栏的左角，直到出现移动 光标。
2. 将工具栏拖至新位置。若工具栏因太宽而无法完全显示，请点击箭头以显示所有按钮，如图 12.3 所示。

图 12.3 展开工具栏

对工具栏定位

1. 右击要移动的工具栏。
2. 选择**位置**，然后选择新位置。

注意：将某个工具栏移至左侧、右侧或底部位置后，您可以将其他工具栏添加至同一位置。例如，您可以将**文件编辑**工具栏重新定位在 JMP 窗口的右侧。要在右侧添加其他工具栏，请右击该栏，然后选择要添加的工具栏。

复制和粘贴菜单、菜单项、工具栏和按钮

复制现有项是在菜单或工具栏中创建新项的快捷方式。该选项使您能够尽量少地更改新项中的设置。

复制和粘贴菜单和菜单项

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “菜单编辑器”随即显示。
2. （可选）更改定制设置，控制可以查看定制内容的用户。请参见第 564 页的“定制设置选项”。
3. 右击要复制的菜单或菜单项，然后选择**复制**。
4. 右击复制的项要在其之前、之后或其中显示的菜单或菜单项，然后选择**粘贴**。

   出现以下之一：
   - 随即显示可能位置的列表。
   - 菜单或菜单项显示在选定项之下。请跳至第 7 步。

5. 通过执行以下操作之一选择菜单的位置：
   - 要将项粘贴至选定菜单**之前**，请选择**在“...”之前粘贴**。
   - 要将菜单粘贴至选定菜单**之后**，请选择**在“...”之后粘贴**。
   - 要将菜单粘贴至选定菜单**内部**，请选择**在“...”中粘贴**。（仅当将菜单选作新位置时才可用。）
   - 要取消操作，请选择**取消**。

   该项将按照您的指定显示出来。
6. 修改该项。
7. 点击**保存**保存所做更改。

有关修改项的详细信息，请参见第 567 页的“对主菜单栏进行个性化设置”和第 567 页的“创建菜单项和工具栏按钮”。

复制和粘贴工具栏和按钮

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “菜单编辑器”随即显示。
2. （可选）更改定制设置，控制可以查看定制内容的用户。请参见第 564 页的“定制设置选项”。
3. 右击要复制的工具栏或按钮，然后选择**复制**。
4. 执行以下操作之一：
   - 要粘贴工具栏，请右击**工具栏**并选择**粘贴**。该工具栏将显示在工具栏列表的末尾。
   - 要粘贴按钮，请右击要将新按钮置于其前或其后的按钮，然后选择**粘贴**。若选择了工具栏上的第一个按钮，请选择该按钮的位置。否则，该按钮将粘贴至选定按钮之下。
5. 修改该项。
6. 点击**保存**保存所做更改。

有关修改项的详细信息，请参见第 564 页的“对工具栏进行个性化设置”和第 567 页的“创建菜单项和工具栏按钮”。

重新排列定制菜单、菜单项和按钮

在 Windows 上，您可以重新排列所创建的菜单、菜单项和按钮的顺序。例如，在**文件 > 新建**菜单下，您可以将定制菜单移至**数据表**命令之前。

在项的“常规”属性中，“源”确定您是否能够移动该项。若“源”为**定制项**，则可以移动项。无法移动内置项。在其他定制设置中定义的项只能在该定制设置内移动。请参见第 564 页的“定制设置选项”。

注意：右击菜单提供用于剪切和粘贴项的选项。要剪切某项，还可以按 Ctrl+X 或按 Shift+Delete。如需粘贴，请按 Ctrl+V 或按 Shift+Insert。

重新排列定制菜单和菜单项

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “主菜单”列表显示在左侧。
2. 请执行以下操作：
   - 选择要移动的项。
   - 验证“源”是否为**定制项**。
   - 若“源”为**所有用户**或 **JMP 插件**，请选择指定的定制设置并验证“源”是否为**定制项**。
3. 右击要移动的定制菜单或菜单项，然后选择**剪切**。
4. 右击剪切项要放到的新位置，然后选择**粘贴**。

   出现以下之一：
   - 出现一个列表，其中包含剪切项的可能位置。选择某个菜单（如**文件**或**新建**）或是菜单中的第一项（如**文件 > 新建**菜单中的**数据表**）时，就会显示该列表。
   - 剪切菜单项显示在选定菜单项之下。请跳至第 6 步。
5. 通过执行以下操作之一选择剪切项的位置：
   - 要将剪切项粘贴至选定项**之前**，请选择**在“...”之前粘贴**。
   - 要将剪切项粘贴至选定项**之后**，请选择**在“...”之后粘贴**。
   - 要将剪切项粘贴到选定菜单**内部**，请选择**在“...”中粘贴**。（仅当将菜单选作新位置时才可用。）
   - 要取消移动，请选择**取消**。

   该项将按照您的指定显示出来。
6. 点击**保存**保存所做更改。

重新排列定制按钮

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “工具栏”列表显示在左侧。
2. 请执行以下操作：
   - 选择要移动的定制按钮。
   - 验证“源”是否为**定制项**。
   - 若“源”为**所有用户**或 **JMP 插件**，请选择指定的定制设置并验证“源”是否为**定制项**。
3. 右击要移动的定制按钮，然后选择**剪切**。
4. 右击剪切按钮的新位置。

   出现以下之一：
   - 出现一个列表，其中包含剪切按钮的可能位置。当您选择工具栏上的第一个按钮（如**文件编辑**工具栏中的**新建数据表**）时，就会出现该列表。

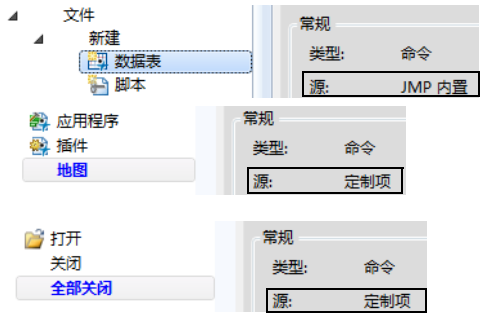
- 该按钮粘贴至工具栏的末尾。当您选择该工具栏的名称时就会执行粘贴操作。例如，当您选择**文件编辑**并粘贴时，该按钮将出现在最后一个按钮**运行脚本**之后。请跳至第 6 步。
- 剪切按钮出现在选定按钮之后。请跳至第 6 步。

5. 通过执行以下操作之一选择剪切按钮的位置：
   - 要将剪切按钮粘贴至选定按钮**之前**，请选择**在“...”之前粘贴**。
   - 要将剪切按钮粘贴至选定按钮**之后**，请选择**在“...”之后粘贴**。
6. 点击**保存**保存所做更改。

删除定制项

若不想暂时显示或隐藏工具栏、按钮、菜单和菜单项，您可以删除所创建的项，将其从 JMP 中永久移除。“源”确定您是否能够删除该项并确定包含该项的定制设置。图 12.4 显示定制“源”类型的示例。

图 12.4 定制 “源”的示例



不支持空菜单。删除菜单中唯一的项时，将删除整个菜单而不仅是选定项。

注意：确保您确实要删除定制项。这些项将立即删除而不进行确认。

要删除定制项，请执行以下步骤：

1. 选择**视图 > 定制 > 菜单和工具栏**。
   “菜单编辑器”随即显示。
2. 选择要删除的项。
3. 验证是否能删除该项，并根据需要更改定制设置。
4. 右击选定项并选择**删除**。
   该项立即删除。
5. 点击**保存**。
   保存所做更改。

显示或隐藏定制项

在“菜单编辑器”中，您可以显示或隐藏特定定制设置中的工具栏、按钮、菜单和菜单项。项的“源”确定显示或隐藏该项的位置。例如：

- 当“源”为**所有用户**时，将定制设置改为**所有用户**可对所有用户隐藏该项。
- 当“源”为 **JMP 插件**时，将定制设置改为 **JMP 插件**，后接该插件的名称。该项将在包含选定项的 JMP 插件的所有设置中显示或隐藏。

若不更改定制设置，则该项仅在当前选定的设置中显示或隐藏。有关定制设置的详细信息，请参见第 564 页的“定制设置选项”。

注意：要对当前用户快速隐藏或显示工具栏，请从**视图 > 工具栏**列表中选择或取消选择这些工具栏。

要显示和隐藏定制集中的项，请执行以下步骤：

1. 选择**视图 > 定制 > 菜单和工具栏**。
   “菜单编辑器”随即显示。
2. 选择要修改的定制设置。
3. 选择要显示或隐藏的项。
4. 在“常规”属性中选择或取消选择**已隐藏**。有关“常规”属性的详细信息，请参见第 566 页的“第 2 步：指定标题和内部名称”。
5. 点击**保存**保存所做更改。

导入定制设置

对工具栏和菜单的更改存储在定制设置中或纯文本文件中。您需要在 JMP 中打开定制设置以导入已修改的工具栏和菜单。定制设置随后显示在“菜单编辑器”中。有关菜单定制文件的详细信息，请参见第 564 页的“定制设置选项”。

要导入定制设置，请执行以下步骤：

1. 选择**文件 > 打开**。
2. 在“文件名”列表中，选择**所有 JMP 文件**（或 **JMP 菜单文件**），然后选择定制设置。
3. 点击**打开**。
   随即显示一条确认消息，指出已导入定制设置。“菜单编辑器”也随之打开。
4. 显示“菜单编辑器”并点击**确定**保存更改，或点击**取消**放弃更改。

从 JMP 中删除定制设置

修改项时，通过点击“菜单编辑器”的**取消**按钮并点击**是**确认取消，可以放弃未保存的更改。保存定制设置后，您还可以删除所有定制设置并恢复为原始菜单和工具栏。

删除当前用户的所有定制设置

1. 选择**视图 > 定制 > 恢复为出厂默认值**。
2. 点击**是**删除当前用户的定制设置。

   若 JMP 找到 JMP 以前版本的定制设置，则会出现一条确认消息。执行以下操作之一：
   - 选择**否**可避免向当前安装的 JMP 添加这些定制设置。
   - 选择**是**可向当前安装的 JMP 添加这些定制设置。

删除其他定制设置中的定制内容

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “菜单编辑器”随即显示。
2. 点击**更改**。
3. 选择定制设置并点击**确定**。
4. 点击**全部还原**。

   随即显示确认窗口。
5. 点击**是**删除定制设置。

删除选定项的定制设置

1. 选择**视图 > 定制 > 菜单和工具栏**。

   “菜单编辑器”随即显示。
2. （可选）选择包含该项的定制设置。
3. 选择要修改的项。
4. 执行以下操作之一：
   - 要恢复**未保存**项的**以前**属性，请点击**重置**。
   - 要恢复**内置**项的**原始**属性，请点击**全部还原**，然后点击**是**进行确认。
5. 点击**保存**保存所做更改。

在 macOS 上对 JMP 工具栏进行个性化设置

在 macOS 上，您可以设置工具栏仅显示您需要的图标。图标的显示取决于窗口类型。例如，脚本窗口会显示用于重新格式化、加密和运行脚本等选项的图标。表图标仅可用于数据表工具栏。

您可以指定在工具栏上是显示图标、图标和图标名称，还是仅显示名称。还可以指定显示小图标。

添加、删除或重新排列工具栏图标

1. 打开要定制工具栏的窗口的类型。
2. 选择**视图 > 定制工具栏**。

   出现一个窗口，其中显示与当前窗口类型相关的图标。
3. 请执行以下任意操作：
   - 要向工具栏添加图标，请将图标从窗口拖到工具栏上。
   - 要移动工具栏上的图标，请将图标拖至工具栏上的新位置。
   - 要从工具栏中删除图标，请将图标从工具栏拖到窗口中。

   所做更改应用于当前窗口以及同类型的所有打开窗口。

图 12.5 在 macOS 上添加、重新排列和删除工具栏按钮

4. 点击**完成**。

   若工具栏因太宽而无法完全显示，请点击工具栏上的右箭头进行显示，然后选择其他图标。或者，您可以点击并拖动窗口，直到显示整个工具栏。

注意：要显示最初的一组工具栏图标，请将工具栏定制窗口底部的默认设置拖至工具栏。

更改工具栏图标的外观

1. 打开要定制工具栏的窗口的类型。
2. 选择**视图 > 定制工具栏**。

随即显示包含工具栏图标的窗口。

3. 在该窗口底部，从“显示”列表中选择以下选项之一：
   - 要仅显示图标，请选择**仅图标**。
   - 要显示图标及其名称，请选择**图标和文本**。
   - 要仅显示图标名称，请选择**仅文本**。
4. 要显示小图标或标准图标，请选择或取消选择**使用小图标**。

   所做更改应用于当前窗口以及同类型的所有打开窗口。
5. 点击**完成**。

在 macOS 上对 JMP 菜单进行个性化设置

定制 JMP 菜单可快速访问文件或修改现有菜单项。选择菜单项可以从您的计算机运行 JSL 脚本，或执行您指定的 JSL 命令。例如，您可以创建菜单项以打开常用的脚本。该选项是在“主窗口”中将脚本添加为收藏项的替代方法。

功能包括添加和重命名菜单项、隐藏菜单项、插入菜单项分隔符，以及将菜单项与 JSL 关联。

JMP 定制设置存储在 <username>/Library/Application Support/JMP/16/usercust.jmpcust 中。

注意：您无法删除默认的 JMP 菜单或删除子菜单（打开该菜单可显示其他菜单项）。默认和自定义子菜单也无法重命名。

添加菜单项的示例

在本例中，创建一个可运行脚本的菜单项。

1. 选择**视图 > 定制菜单**。
2. 在 “JMP 菜单编辑器”中，依次选择**文件**和**打开**。
3. 按 Ctrl 键，然后选择**之后插入新菜单项**。
4. 在 “菜单标签”框中键入**现在是什么时间？**。
5. 选择**运行该文件中的 JSL:**，然后选择 Library/Application Support/JMP/16/Samples/Scripts/timeAnnouncer.jsl。
6. 点击**添加**并关闭 “菜单编辑器”窗口。
7. 选择**文件 > 现在是什么时间？**。

   随即显示 “调度程序”窗口，并报出当前时间。

添加子菜单项的示例

子菜单可打开以显示子菜单项。在本例中，创建可打开数据表的子菜单项。

1. 选择**视图 > 定制菜单**。

2. 在 “JMP 菜单编辑器”中，依次选择**文件**和**打开**。
3. 按 Ctrl 键并选择**之后插入新子菜单**。
4. 键入 “打开我的数据表”，然后选择 **JSL 命令**。

注意：选择 **JSL 脚本**可从您的计算机中选择一个 JSL 脚本。

5. 点击**添加**。
6. 选择**打开我的数据表**，删除 “菜单标签”框中的**子项占位符**，然后键入 “打开 Napoleons March.jmp”。
7. 在 “运行该 JSL”框中，键入 `Open("$SAMPLE_DATA/Napoleons March.jmp");`。
8. 点击**保存更改**。
9. 选择**文件**菜单以查看新的菜单项。

提示：若新的菜单项出现在菜单中，但未出现在菜单编辑器中，请关闭菜单编辑器窗口，然后重新打开它。

删除自定义菜单或子菜单项

- 通过选择一项并从选项列表中选择**删除选择**可删除一个自定义菜单项。
- 通过从选项列表中选择**全部还原**可删除全部自定义菜单项。
- 要删除子菜单中的项，您必须删除整个子菜单。系统不允许存在空菜单。选择子菜单，然后从选项列表中选择**删除选择**。

为插件定制 JMP 菜单和工具栏

要向 JMP 添加新功能，您可以编写脚本并编译文件以创建插件。编译脚本的一种方式就是使用《Scripting Guide》中描述的插件生成器。

“插件生成器”提供的菜单编辑器支持您在“插件”菜单中创建菜单项。您还可以使用“JMP 菜单编辑器”创建插件的菜单项以及添加工具栏按钮。

第 13 章

JMP 首选项

“首选项”窗口

通过 JMP 首选项，您可以指定常规和特定设置并保存这些设置。

更改首选项

1. 选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)。随即显示图 13.1 中的窗口。其中显示您所选的前一首选项页。
2. 点击一个类别并进行选择。点击**应用**查看结果，然后点击**确定**。

注意：升级到 JMP 16 订阅时，将保留对首选项的更改。在其他 JMP 版本中，您必须在每次升级后定制首选项。

要搜索首选项，请执行以下操作：

1. 选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)。
2. 在顶部的 “过滤器”框中，输入您的文本。

   匹配的文本将突出显示。
3. 在 “首选项组”下，点击某个类别以查看其结果。

搜索项可显示在以下位置：

- 首选项本身
- 首选项的工具提示
- 首选项的选项列表

将首选项重置为默认设置

1. 选择**文件** > **首选项** (Windows) 或 **JMP** > **首选项** (macOS)。
2. 点击**全部重置**，然后点击**确定**。

目录

JMP 首选项概述 583
常规 JMP 首选项 583
JMP 项目的首选项 586
JMP 报表的首选项 587
JMP 图形的首选项 589
JMP 图形和报表的样式首选项 592
数据表的首选项 594
“JMP 数据过滤器”的首选项 598
JMP 平台首选项 599
“打印”首选项 601
用于导入和导出文本文件的首选项 602
第三方数据的首选项 606
Windows 首选项 608
Mac 操作系统首选项 610
“字体”首选项 612
“通信”首选项 614
“Internet 选项”的首选项 614
“文件位置”的首选项 615
“JSL 脚本编辑器”的首选项 617
“SAS 集成”首选项 619
用于更新 JMP 的首选项 622
“JSL 调试器”的首选项 623
“菜单”首选项 623
“查询生成器”的首选项 624
用于对列重新编码的首选项 625
“列切换器”的首选项 627
“日志”首选项 628

JMP 首选项概述

使用“文件”菜单（macOS 上的“JMP”菜单）中的“首选项”命令可显示“首选项”窗口。请参见图 13.1。本章将对每一个类别进行说明。

在 Windows 上，您对首选项所做的更改存储在特定于版本的 JMP.PFS 文件中，该文件位于下面的一个文件夹中：

C:/Users/< 用户名 >/AppData/Roaming/SAS/JMP/16/

C:/Users/< 用户名 >/AppData/Roaming/SAS/JMPPro/16/

在 macOS 上：首选项存储在位于 ~/Library/Preferences/ 中的 com.sas.jmp.plist 内。

当您打开程序时，JMP 会搜索该文件并将您的更改视为出厂默认设置。并且所有后续更改都将会保存在该文件中。

常规 JMP 首选项

大多数常规首选项定制启动时的 JMP 会话。其他设置首选项则针对文件处理以及 JMP 工作区的一般外观。

图 13.1 “常规”首选项

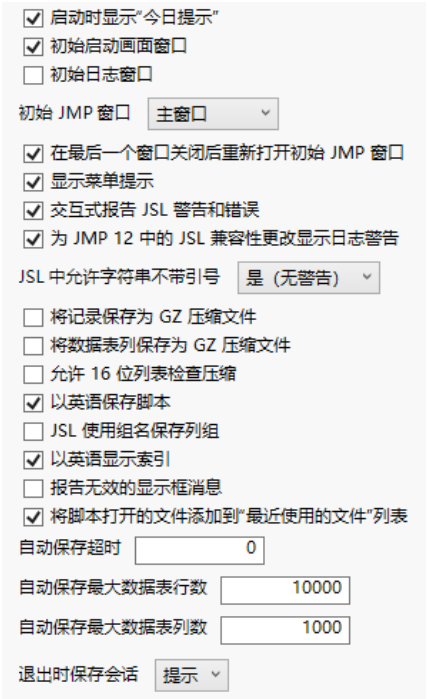


表 13.1 “常规”页面中的首选项

首选项	说明
启动时显示“今日提示”	选择该选项可在启动时显示“今日提示”窗口。默认选择该选项。
初始启动画面窗口	选择该选项可在启动时显示初始启动画面窗口。默认选择该选项。
初始日志窗口	（仅限 Windows）选择该选项可在 JMP 打开时打开日志。
初始 JMP 起始页窗口	（仅限 macOS）选择该选项可在您每次启动 JMP 时显示“JMP 起始页”窗口。默认选择该选项。
初始 JMP 窗口	（仅限 Windows）在所列的窗口中选择其中一个作为默认窗口。当 JMP 启动时，该窗口将出现。“主窗口”是默认窗口。
主窗口	（仅限 macOS）选择该选项可在 macOS 上启动 JMP 时显示“JMP 主窗口”。默认选择该选项。
在最后一个窗口关闭后重新打开初始 JMP 窗口	（仅限 Windows）选择该选项可在关闭最后一个 JMP 窗口时打开您设置为初始 JMP 窗口的任意窗口。若该选项关闭，则关闭最后一个 JMP 窗口将退出 JMP。默认选择该选项。
显示菜单提示	（仅限 Windows）选择该选项可查看主菜单选项和红色小三角菜单选项的工具提示。默认选择该选项。
交互式报告 JSL 警告和错误	在新窗口中显示脚本错误，而不是仅在日志中显示。默认选择该选项。
为 JMP 12 中的 JSL 兼容性更改显示日志警告	显示该选项可将有关 JSL 兼容性的警告打印至日志中。默认选择该选项。
JSL 中允许字符串不带引号	选择该选项可在平台消息中的字符串值不带引号时将警告打印至日志中。带引号的字符串值优先于文字名称。 **是 （无警告）** 允许不带引号的字符串值，并且不将警告打印至日志中。默认选定该选项。 **是 （带警告）** 允许不带引号的字符串值并将警告打印至日志中。 **否** 忽略不带引号的字符串值，并且不将警告打印至日志中。
将记录保存为 GZ 压缩文件	选择该选项可以压缩格式保存 JMP 记录，从而节省磁盘空间。默认取消选择该选项。
将数据表列保存为 GZ 压缩文件	选择该选项可允许 JMP 使用 GZ 压缩格式保存数据表。为了向后兼容，JMP 10 可读取压缩文件但不会对其保存。默认取消选择该选项。

表 13.1 “常规”页面中的首选项 (续)

首选项	说明
允许 16 位列表检查压缩	压缩唯一值小于或等于 65,535 的列。将“列表检查”属性添加至列。默认取消选择该选项。
以英语保存脚本	JMP 非英文版本既可使用英文保存脚本，也可使用本地语言保存脚本。在这些脚本中用户指定的值（如列名和文本字符串）的显示方式与这些值在数据表中的显示方式相同。不过，命令单词（如 `Distribution` 和 `Set Value`）以英语而非本地语言显示，以便脚本以英语显示的 JMP 上运行。字体支持所需的字符时，JMP 在以英语显示的 JMP 中正常显示非罗马字符（如日语变量名称）。 默认选定该选项。取消选择该选项将以本地语言保存脚本。请注意，**仅**在以同种语言运行 JMP 时才正确运行这些脚本。
JSL 使用组名保存列组	在从平台保存的 JSL 脚本中指定列组名，而不是列出所有列名。
以英语显示索引	以英语或当前语言/区域的语言显示 JMP 索引部分。默认选定该选项。取消选择它时，以当前语言/区域的语言显示以下项： • 对象的消息列表 • 显示框的消息列表。 • JSL 函数的类别列表。 **注意：**每个索引项的说明始终以当前语言/区域的语言显示。示例始终以英语显示。
报告无效的显示框消息	将有关无效显示框消息的信息发送到日志。默认选择该选项。 该选项在脚本开发过程中很有用，但是会生成现有脚本不需要的日志消息。
将脚本打开的文件添加到“最近使用的文件”列表	（仅限 Windows）选择该选项可在主窗口“最近使用的文件”窗格和**文件 > 最近使用的文件**列表中包含 JSL Open() 命令打开的文件。默认选择该选项。
自动保存超时	按指定时间间隔自动保存已打开的数据表、记录、报表、项目和脚本（无标题文件除外）的内容。默认设置为 0 分钟，即指示不自动保存。
自动保存最大数据表行数	若行数未超过指定值，则自动保存数据表。10,000 是默认设置。
自动保存最大数据表列数	若列数未超过指定值，则自动保存数据表。1,000 是默认设置。

表 13.1 “常规”页面中的首选项 (续)

首选项	说明
退出时保存会话	该选项允许您在退出 JMP 时保存 JMP 窗口的状态。打开 JMP 时将恢复保存的状态，包括所有打开的文件和窗口。默认设置为“提示”。

JMP 项目的首选项

项目首选项支持您控制如何创建新的空项目、显示哪些工具窗口等。

图 13.2 项目首选项

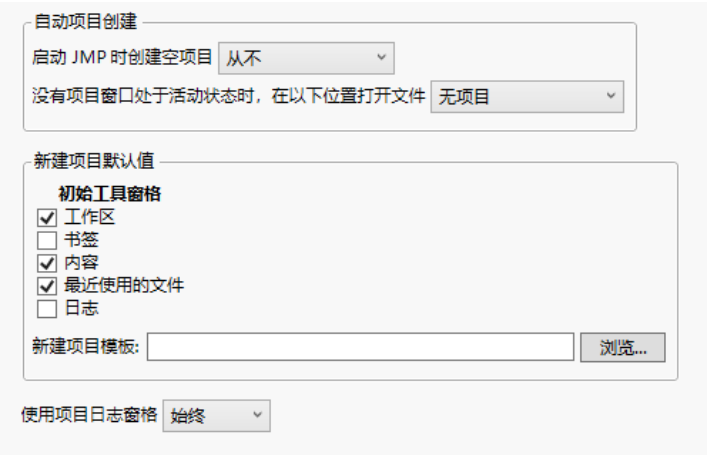


表 13.2 “项目”页上的首选项

首选项	说明
启动 JMP 时创建空项目	在启动 JMP 时创建一个空项目：“从不”、“始终”或“若未打开其他项目”。“从不”是默认设置。
没有项目窗口处于活动状态时，在以下位置打开文件	指定如何自动创建新项目。 **无项目** 不要将文件放入项目。这是默认设置。 **打开项目或无项目** 将文件放到打开的项目（若找到）中或者没有项目。 **打开项目或新建项目** 将文件放到打开的项目（若找到）或新项目中。 **新项目** 将文件放入新项目。

表 13.2 “项目”页上的首选项 (续)

首选项	说明
初始工具窗格	指定项目中最初显示的工具窗格。默认情况下，将显示“工作区”、“内容”和“最近使用的文件”等工具。
新建项目模板	指定要用作新项目模板的项目。新项目具有与模板相同的窗口布局、链接文档、图形、工具窗口和其他项目元素。
使用项目日志窗格	在要查看项目日志而不是使用 JMP 日志时指定。“始终”是默认设置。

JMP 报表的首选项

“报表”首选项用于定制报表的外观。

图 13.3 “报表”首选项

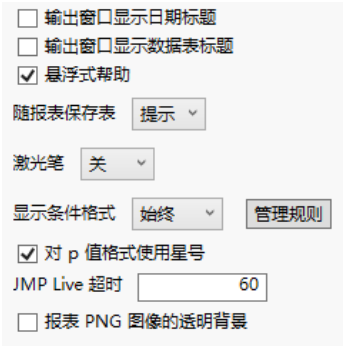


表 13.3 “报表”页面中的首选项

首选项	说明
输出窗口显示日期标题	选择该选项可在报表窗口中显示执行分析的日期和时间。默认取消选择该选项。
输出窗口显示数据表标题	选择该选项可在报表顶部显示数据表的名称和注释（若存在）。默认取消选择该选项。

表 13.3 “报表”页面中的首选项 (续)

首选项	说明
悬浮式帮助	在某些数值输出中，当您将鼠标悬停在结果上时 JMP 提供工具提示样式的帮助。 选择该选项可在可用区域查看工具提示。默认选择该选项。 **注意：**若取消选择该选项，则不再显示悬停标签图。
随报表保存表	从菜单中选择一个选项以确定如何将表保存到报表： **嵌入** 保存报表时，将表嵌入报表。若要与其他人共享报表，请选择该选项。每次打开报表时，将重新打开存储在报表中的数据表（未更改）。若您更改了表，更改内容不会保存到报表中，除非您重新保存报表。 **分隔** 保存报表时，在报表中引用表。 **提示** 保存报表时，系统提示您指定如何保存报表：在报表中嵌入表还是在报表中添加对表的引用。这是默认设置。
激光笔	JMP 具有内置的激光笔，使用它可以很直观地突出报表的某些部分。默认取消选择该选项。要启用它，请从列表中为激光笔选择一种颜色。
显示条件格式	根据条件设置表示值的文本颜色的格式。可用于相关性值、*p* 值和因子模式值。 从菜单中选择以下选项之一： **始终** 条件格式始终应用于相关性值、 *p* 值和因子模式值。这是默认设置。 **仅屏幕** 条件格式始终应用于相关性值、*p* 值和因子模式值。但是，打印报表时不显示条件格式。 **从不** 条件格式从不应用于相关性值、 *p* 值和因子模式值。
管理规则	创建可以应用到数值列或矩阵的定制条件格式规则。请参见“JMP 报表”一章中第 411 页的“在报表中使用条件格式”。 **注意：**验证**对 P 值格式使用星号**首选项，以确保对所有 *p* 值条件使用或不使用星号 (*)。
对 P 值格式使用星号	选择该选项以在显著的 *p* 值旁边显示星号 (*)。默认选择该选项。 **注意：**验证 P 值的所有条件格式规则以确保星号 (*) 未用于任何定义的条件。
JMP Live 超时	指定发布到 JMP Live 的超时值。默认时间为 60 秒。

表 13.3 “报表”页面中的首选项 (续)

首选项	说明
报表 PNG 图像的透明背景	选择该选项可在将报表另存为 PNG 文件时显示透明背景。默认取消选择该选项。

JMP 图形的首选项

“图形”首选项用于定制图形的外观。有关配置图形外观的详细信息，请参见第 592 页的“JMP 图形和报表的样式首选项”。

图 13.4 “图形”首选项

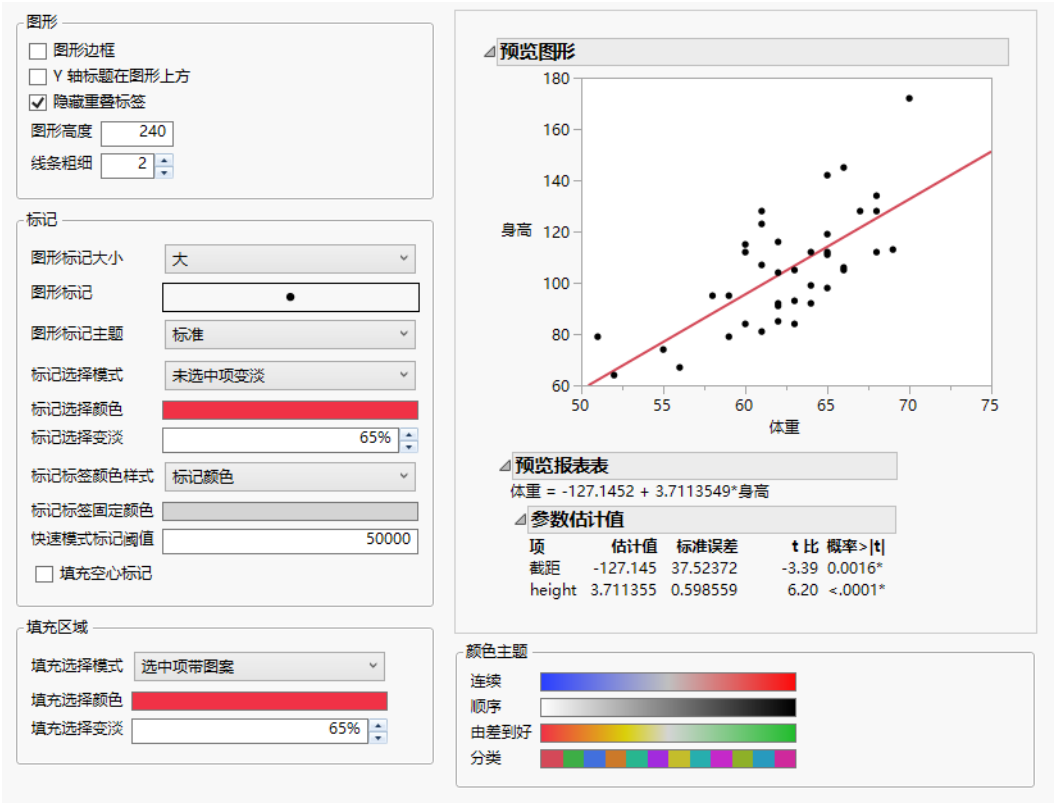


表 13.4 “图形”页面中的首选项

首选项	说明
图形边框	在图形区域的周围绘制边框线。默认取消选择该选项。
Y 轴标题在图形上方	在行上方而不是行旁边显示 Y 轴标题。默认取消选择该选项。
隐藏重叠标签	选择该选项可隐藏重叠的图标签。默认选择该选项。
图形高度	设置图形的默认高度（以像素为单位）。默认设置为 240 像素。某些平台会在该值的基础上乘以特定的量。 **注意：**对于“图形生成器”窗口，指定“图形高度”将按比例扩大窗口，但不应用确切值。
线条粗细	设置与内容相关的线条默认粗细（以像素为单位）。默认设置为两个像素。 **注意：**该首选项不应用于网格线。
图形标记大小	选择图形中标记的默认大小：点、小、中、大、特大、超大或最大。在图形上将标记大小设置为“首选”时应用该首选项。默认设置为“大”。 当图形中的标记数超过“快速标记阈值”时，图形的“标记绘制模式”将设置为“快速”，即：绘制点大小的标记。
图形标记	为图形中的标记选择默认标记形状。默认设置为小点。
图形标记主题	选择按行或按列标记时图形中标记的默认主题。默认设置为“标准”。
标记选择模式	为所选标记选择默认格式。默认设置为“未选中项变淡”。请参见“JMP 报表”一章中第 421 页的“更改标记的选择模式”。
标记选择颜色	当“填充选择模式”为“选中项相同颜色”时，该设置将指定的颜色应用于所选标记。默认颜色为红色。
标记选择变淡	当“标记选择模式”为“未选中项变淡”时，该设置以指定的量（以百分比表示）使未选中标记变淡。默认设置为“65%”。

表 13.4 “图形”页面中的首选项 (续)

首选项	说明
标记标签颜色样式	当您为数据表中的列添加标签时，选择图形中显示的标记标签的颜色。 **标记颜色** 标记标签的颜色与标记相同。默认选定该选项。 **标记颜色变淡** 标记标签根据“标记选择变淡”首选项中的设置变淡。 **固定颜色** 标记标签颜色与指定颜色相同。“标记标签固定颜色”首选项决定该颜色。
标记标签固定颜色	为固定颜色的标记标签选择颜色，您可在“标记标签颜色样式”首选项中选择该选项。默认颜色为灰色。
快速模式标记阈值	当 JMP 刷新报表窗口时，可以采用两种不同的速度在图上绘制标记或点：正常和快速。若 JMP 处于正常绘制模式下，而图形上的标记数大于指定的阈值，JMP 将自动切换到快速模式。 输入区分正常模式和快速模式的标记数量。当图形中的标记数超过“快速标记阈值”时，图形的“标记绘制模式”将设置为“快速”，即：绘制点大小的标记。默认设置为“50,000”。
填充空心标记	选择该选项可将不透明背景应用于标记。默认取消选择该选项。
填充选择模式	为图形对象中的选定行选择默认突出显示： **选中项带图案** 条纹图案 （默认设置） **选中项变暗** 纯暗色 **选中项带轮廓** 轮廓 **选中项相同颜色** 首选项中的“填充选择颜色”设置选定对象的格式。 **未选中项变淡** 未选定行变淡，选定行保持原始颜色。
填充选择颜色	当“填充选择模式”为“选中项相同颜色”时，为图形对象中的选定行选择默认颜色。默认颜色为红色。
连续颜色主题	为连续数据选择默认颜色主题。默认主题为“由蓝经灰到红”。请参见“输入和编辑数据”一章中第 206 页的“在图形中创建颜色主题”。

表 13.4 “图形”页面中的首选项 (续)

首选项	说明
顺序颜色主题	为不具备有意义的中点的顺序数据选择默认颜色主题。默认主题为“由白到黑”。
由差到好颜色主题	为从正值数据到负值数据的发散数据选择默认颜色主题。默认主题为“停止指示灯由差到好”。
分类颜色主题	为分类数据选择默认颜色主题。默认主题为“JMP 默认”。请参见“输入和编辑数据”一章中第 206 页的“在图形中创建颜色主题”。

JMP 图形和报表的样式首选项

“样式”页支持您定制颜色、刻度标、网格线和报表表。

有关配置图形外观的详细信息，请参见第 589 页的“JMP 图形的首选项”。

图 13.5 “样式”首选项

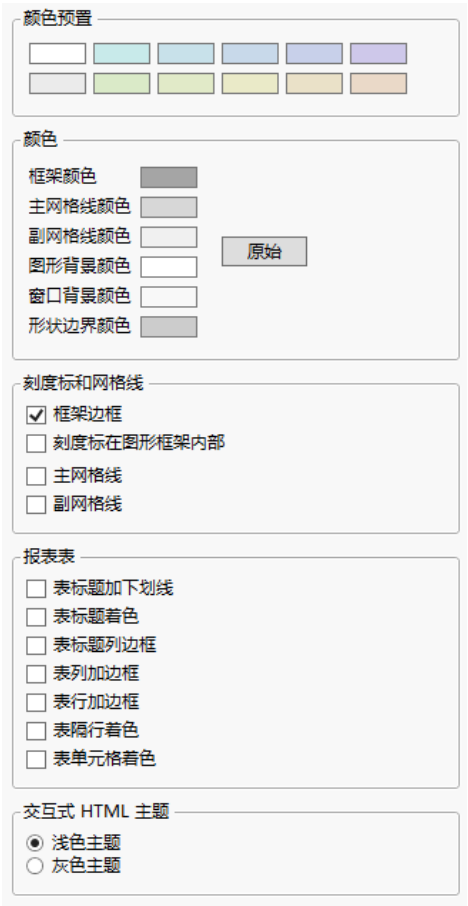


表 13.5 “样式”页面中的首选项

首选项	说明
颜色预置	“颜色预置”提供一组预定义的颜色，可用于定制报表、图形、图表等的外观。选定颜色的淡色会应用于各个元素。 **注意：**选定颜色也应用于窗口。
框架颜色	定义应用于图形框架和刻度标的颜色。点击可更改默认颜色。
主网格线颜色	定义主网格线颜色。点击可更改默认颜色。
副网格线颜色	定义副网格线颜色。点击可更改默认颜色。
图形背景颜色	点击颜色框可为所有图形选择背景颜色。点击可更改默认颜色。

表 13.5 “样式”页面中的首选项 (续)

首选项	说明
窗口背景颜色	点击颜色框可为所有报表和数据表选择背景颜色。点击可更改默认颜色。
形状边界颜色	点击颜色框可选择形状文件边界的颜色。
原始	点击可恢复默认颜色设置。
框架边框	仅在轴上显示边框。默认选择该选项。
刻度标在图形框架内部	在图形框架内显示坐标轴刻度标。默认取消选择该选项。
主网格线	在图形上显示主网格线。默认取消选择该选项。
副网格线	在图形上显示副网格线。默认取消选择该选项。
表标题加下划线	在表标题下方显示边框。默认取消选择该选项。
表标题着色	显示着色的表标题。默认选择该选项。
表标题列边框	显示表标题中列之间的分隔线。默认选择该选项。
表列加边框	在表列之间显示边框。默认取消选择该选项。
表行加边框	在表行之间显示边框。默认取消选择该选项。
表隔行着色	显示隔行着色的表。默认取消选择该选项。
表单元格着色	显示着色的表单元格。默认取消选择该选项。
交互式 HTML 主题	指定交互式 HTML 页的颜色主题。默认主题为“浅色主题”。

数据表的首选项

“表”首选项用于定制 JMP 数据表，包括公式处理、外观以及与 SAS 数据集的兼容性。

图 13.6 “表”首选项

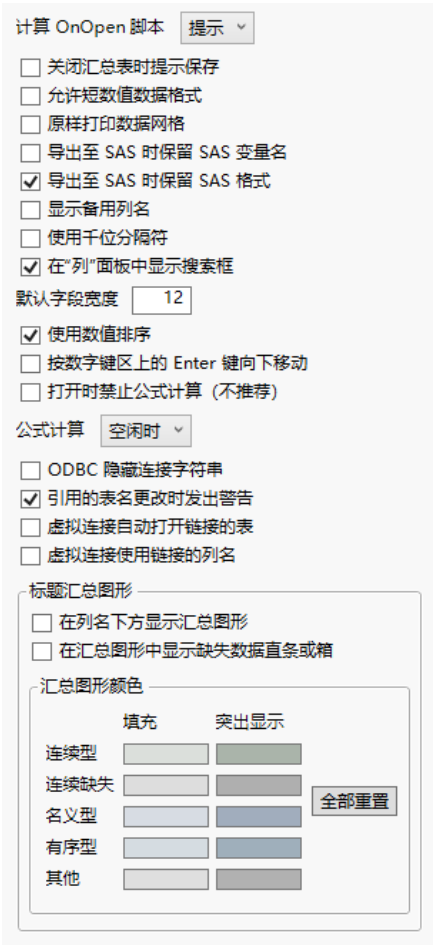


表 13.6 “表”页面中的首选项

首选项	说明
计算 OnOpen 脚本	若将脚本保存到数据表并将它命名为 OnOpen 或 On Open，则在每次打开数据表时自动运行该脚本。用于执行其他程序的 On Open 脚本从不运行。 默认选定“提示”。每次您在当前 JMP 会话中打开数据表时，系统将记住您的选择。 **始终** 允许在无提示的情况下运行 On Open 脚本。 **从不** 防止在打开数据表时自动运行 On Open 脚本。

表 13.6 “表”页面中的首选项 (续)

首选项	说明
关闭汇总表时提示保存	在关闭未保存的链接汇总表时提示保存该表。请注意，在“汇总”窗口中选择该选项后，在“表”首选项中会也选定该选项。若不想系统提示您将来保存所有链接的汇总表，请取消选择该首选项。默认取消选择该选项。
允许短数值数据格式	JMP 可采用 8 位（一个字节）的形式存储数值数据。该选项使您在选择“列 > 列信息”并为列分配数据类型时可以使用短整数格式。为数据使用正确的短整数格式时，数值显示方式没有变化，但是数据表占用的磁盘空间变少了。请参见“设置 JMP 列属性”一章中第 256 页的“短整型格式”。
原样打印数据网格	选择该选项可按屏幕上显示的样式打印 JMP 数据表。 清除该选项将调整列宽度以适应内容宽度。默认取消选择该选项。
导出至 SAS 时保留 SAS 变量名	选择该选项可在将 JMP 数据表导出到 SAS 数据集时使用与 SAS 兼容的变量名。默认取消选择该选项。
导出至 SAS 时保留 SAS 格式	选择该选项可在将 JMP 数据表导出到 SAS 数据集时使用与 SAS 兼容的格式。默认选择该选项。
显示备用列名	选择该选项可显示列的名称-标签对。若定义了两个列属性，且“列名”与其中一个属性相同，则列具有名称-标签对。 JMP 可识别以下名称-标签对： • SAS 名称-标签对，其中属性为“SAS 名称”和“SAS 标签” • SPSS 名称-标签对，其中属性为“SPSS 名称”和“SPSS 标签” • 短-长名称-标签对，其中属性为“短名称”和“长名称” 若选定该选项，则名称和标签都将出现在对话框中。
使用千位分隔符	选择该选项可使用适合语言/区域的千位分隔符显示数值。默认取消选择该选项。
在“列”面板中显示搜索框	在列名上方的“列”面板中显示搜索框。默认选择该选项。
默认字段宽度	更改在数值列中显示的位数。默认设置为“12”。
使用数值排序	指定包含数字的文本按数字顺序排序。默认选择该选项。

表 13.6 “表”页面中的首选项 (续)

首选项	说明
按数字键区上的 Enter 键向下移动	（仅限 Windows）选择数据表单元格并在键盘上按 Enter 键时，将向下选择下一个表单元格。按 Tab 键或键盘右侧的数字键盘上的 Enter 键可向右选择下一个表单元格。 选择该选项可更改数字键盘上 Enter 键的行为，以便向下而非向右选择下一个表单元格。默认取消选择该选项。
打开时禁止公式计算	选择该选项可防止在打开数据表时计算列公式。 取消选择该选项将允许在打开数据表时计算列公式。默认取消选择该选项。 不建议使用该选项，因为通常您需要知道公式是否计算。
公式计算	指定何时计算公式。 **空闲时** 当 JMP 不执行操作时计算公式。不显示进度条。默认选择该选项。 **立即** 立即计算公式并显示进度条。
ODBC 隐藏连接字符串	选择该选项可使“打开数据库”命令隐藏 ODBC 连接设置（即用户 ID 和密码）。有关更多信息，请参见《Scripting Guide》。默认取消选择该选项。
引用的表名更改时发出警告	在虚拟链接（引用）的表名更改时显示警告。默认选择该选项。
虚拟连接自动打开链接的表	自动打开虚拟连接的数据表。指向数据表的链接位于“JMP 主窗口”中。默认取消选择该选项。
虚拟连接使用链接的列名	指定在虚拟连接的列（显示在“列”列表中）中使用链接的列名。该选项支持您指定较短的列名（例如，**奶酪**而不是**奶酪[选择]**）。默认取消选择该选项。
标题汇总图形	**在列名下方显示汇总图形** 在列标题中显示汇总图形（一种分布）。当行数低于阈值 （如 300 万行）时应用。默认取消选择该选项。 **在汇总图形中显示缺失数据直条或箱** 指定缺失的数据条或箱最初是否显示在汇总图形中。要在单个图形中显示缺失值直条，请右击该图形并选择**缺失值直条**。默认不显示缺失值直条。 **汇总图形颜色** 指定填充和突出显示的颜色（选择直条时显示的颜色），包括按数据类型填充和突出显示。

“JMP 数据过滤器”的首选项

“数据过滤器”首选项定制数据过滤器的设置。

图 13.7 “数据过滤器”首选项

表 13.7 “数据过滤器”页上的首选项

首选项	说明
模式	指定过滤模式。 **选择** 在数据表中突出显示选定行。默认选择该选项。 **显示** 对未选定的行显示“隐藏”图标。默认取消选择该选项。 **包括** 对未选定的行显示“排除”图标。默认取消选择该选项。

表 13.7 “数据过滤器”页上的首选项 (续)

首选项	说明
选项	指定数据过滤器选项。 **自动清除** 若您在“数据过滤器”中选择了多个名义型或有序型列，则该选项将在进行新的选择之前清除所有其他选择。默认取消选择该选项。 **按“与”分组** 允许您控制多组列过滤器的“与”和“或”行为。默认取消选择该选项。 **条件** 限制为选定的过滤器列显示的类别。默认取消选择该选项。 **使用浮动窗口** 令“数据过滤器”窗口停留在其关联的数据表的最上方。默认选择该选项。 **选择缺失值** 突出显示数据表中的缺失值。默认取消选择该选项。
显示	指定分类列的显示选项。 **复选框显示** 在每个水平旁边显示复选框。默认取消选择该选项。 **显示直方图和直条** 在数据过滤器中显示直方图和直条。默认选择该选项。

JMP 平台首选项

每个分析报表具有默认显示的各种绘图和表选项。但是，还可能有每次运行特殊分析时想要查看的其他选项。例如，二元分析默认显示散点图，但是您还可能每次想要查看线性拟合。通过选择“首选项”窗口左侧面板中的平台选项，您可以设置分析的默认选项。（通过使用平台，如“二元”平台，可运行分析。因此，该类别的名称为“平台”。）

在“平台”列表中突出显示分析名称。它的可用选项显示在选项框中，其中已选中默认值。

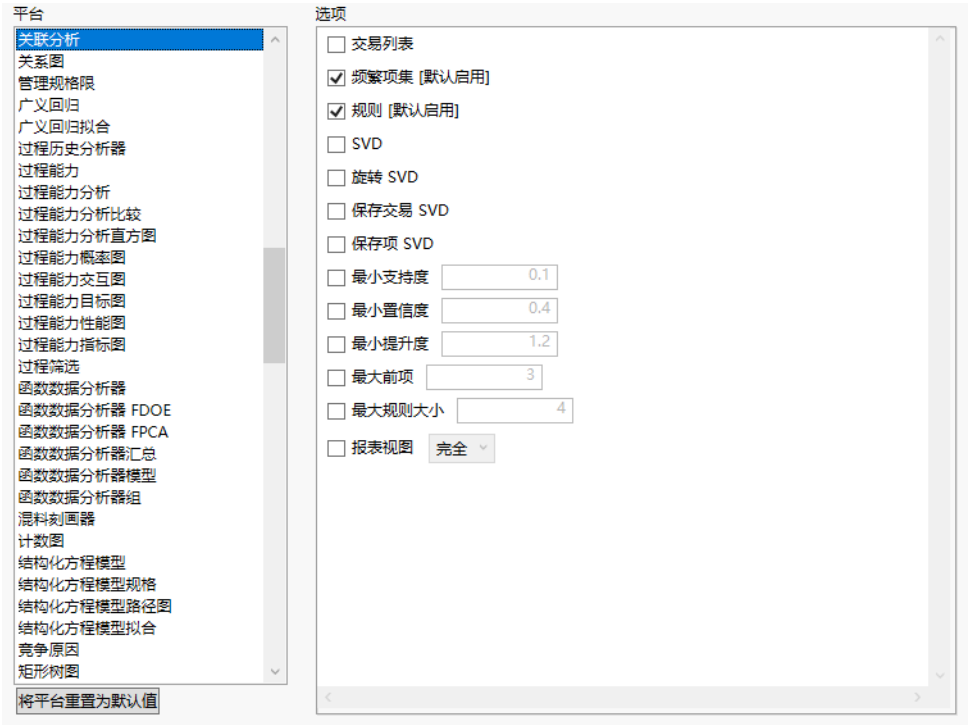
某些平台首选项包括一个“设置”复选框，用来控制在报表中是否显示某个选项。

- 要防止某个选项出现在报表中，请在该选项旁边选择**设置**但不选择该选项。
- 要确保某个选项出现在报表中，请选择**设置**并选择该选项。

点击**将平台重置为默认值**可将选中的平台选项还原为默认设置。

点击**重置为默认值**可将所有平台还原为默认设置。

图 13.8 “平台”首选项


平台
关联分析
关系图
管理规格限
广义回归
广义回归拟合
过程历史分析器
过程能力
过程能力分析
过程能力分析比较
过程能力分析直方图
过程能力概率图
过程能力交互图
过程能力目标图
过程能力性能图
过程能力指标图
过程筛选
函数数据分析器
函数数据分析器 FDOE
函数数据分析器 FPCA
函数数据分析器汇总
函数数据分析器模型
函数数据分析器组
混料刻画器
计数图
结构化方程模型
结构化方程模型规格
结构化方程模型路径图
结构化方程模型拟合
竞争原因
矩形树图
将平台重置为默认值
选项
交易列表
频繁项集 [默认启用]
规则 [默认启用]
SVD
旋转 SVD
保存交易 SVD
保存项 SVD
最小支持度 0.1
最小置信度 0.4
最小提升度 1.2
最大前项 3
最大规则大小 4
报表视图 完全

“打印”首选项

在“打印”页面中，您可以配置默认打印设置，如页边距、页眉和页脚。

图 13.9 “打印”首选项

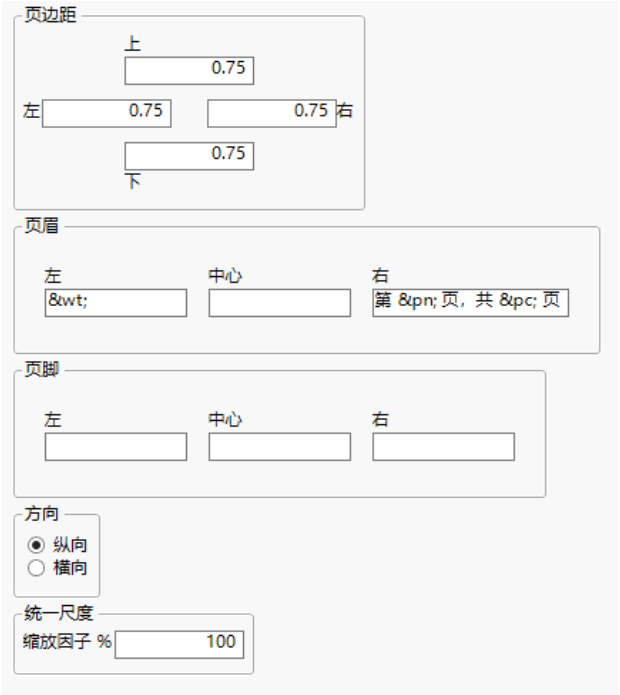


表 13.8 “打印”页上的首选项

首选项	说明
上（页边距）	设置打印页的上边距。默认设置为 0.75 英寸。
左（页边距）	设置打印页的左边距。默认设置为 0.75 英寸。
右（页边距）	设置打印页的右边距。默认设置为 0.75 英寸。
下（页边距）	设置打印页的下边距。默认设置为 0.75 英寸。
左（页眉）	设置页面的左侧页眉信息。默认设置为 &wt;（即标题）。
中心（页眉）	设置页面的中心页眉信息。默认设置为空。
右（页眉）	设置页面的右侧页眉信息。默认设置为第 &pn; 页，共 &pc; 页（即，页码及总页数）。

表 13.8 “打印”页上的首选项 (续)

首选项	说明
左（页脚）	设置页面的左侧页脚信息。默认设置为空。
中心（页脚）	设置页面的中心页脚信息。默认设置为空。
右（页脚）	设置页面的右侧页脚信息。默认设置为空。
方向	设置页面的打印方向为“纵向”或“横向”。默认设置为“纵向”。
缩放因子	设置页面的打印缩放因子（百分比形式）。默认设置为“100%”。

注意：用于在页眉或页脚中打印当前日期的代码是 &d。

用于导入和导出文本文件的首选项

“文本数据文件”首选项用于定制导入和导出文本文件。

图 13.10 “文本数据文件”首选项

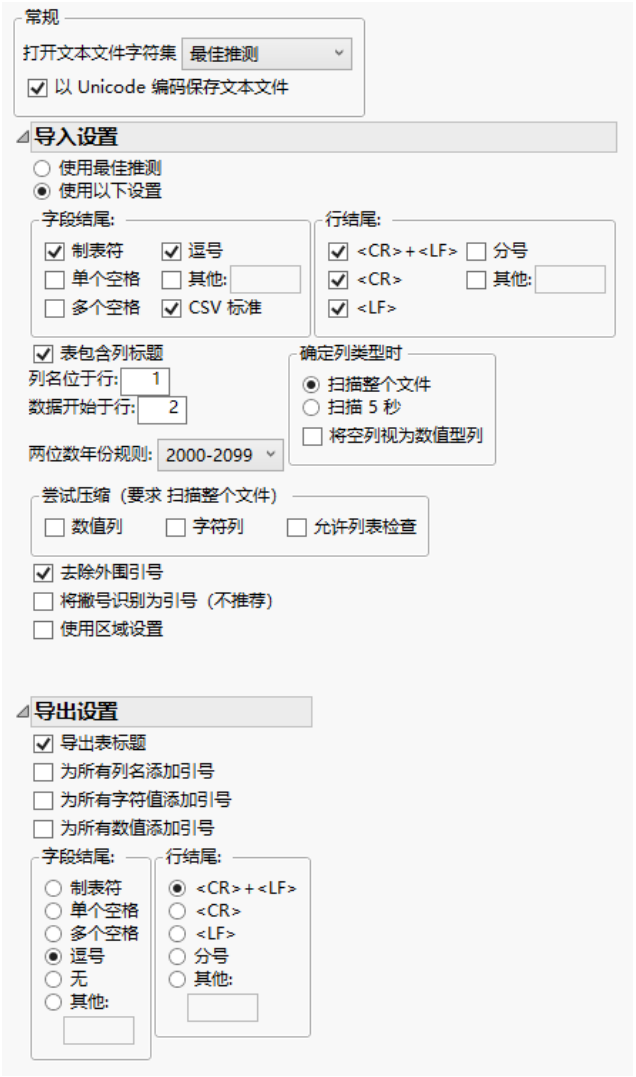


表 13.9 文本文件的“导入设置”首选项

首选项	说明
打开文本文件字符集	选择菜单中的一个选项以确定 JMP 打开文件时使用的字符编码。默认设置为“最佳推测”。请注意：Windows-1252 在某些系统上被视为 ANSII，并且不支持 UTF-8-BOM。

表 13.9 文本文件的“导入设置”首选项 (续)

首选项	说明
以 Unicode 编码保存文本文件	JMP 使用 Unicode 字符集，该字符集支持特殊字符，如 é 和 ½。它将不含特殊的 Unicode 字符的文件自动以纯文本格式保存。默认选择该选项。 取消选中该复选框可将您的所有文件以纯文本格式保存。
导入设置	选择 JMP 用于打开文本文件的策略。默认选择**使用以下设置**。在这种情况下，您需要确保设置反映您的文本文件情况。 若您选择**使用最佳推测**，JMP 将在文本文件中收集有关制表符、逗号、空格和其他几个字符的统计信息，并使用基于规则的系统确定可能的文件格式。这些规则尝试指定合理的字段宽度和每行合理的字段数。若您的数据格式与规则设计来推测的格式有很大区别，则 JMP 推测不正确。在这种情况下，请使用向导或在这些首选项设置中明确描述您的数据。
字段结尾	选择一个或多个字符，用作导入文本数据时指示字段结尾的分隔符。默认选定制表符、逗号和 CSV 标准。 选择**其他**选项并输入一个字符可指定未列出的分隔符。
行结尾	选择一个或多个字符，用作指示行结尾的分隔符。默认选定 <CR>+<LF>、<CR> 和 <LF>。 选择**其他**选项并输入一个字符可指定未列出的分隔符。 请注意，若在导入文本数据时遇到双引号，则分隔符规则会更改以查找结束双引号。其他文本分隔符（包括引号中嵌入的空格）将被忽略，并视为文本字符串的一部分。
表包含列标题	若您的文本文件包含列名，则选择该选项。若选择该选项，请在**列名位于行**旁边的字段中输入列名所在的行号。默认选择该选项。
列名位于行	若选择**表包含列标题**选项，请在该字段中输入列名所在的行号。“第一行”是默认设置。
数据开始于行	输入数据在您的文本文件中开始的行号。“第二行”是默认设置。
确定列类型时	设置 JMP 扫描文本文件以确定列的数据类型的时长。默认选定**扫描整个文件**。请注意，对于大文件，**扫描整个文件**选项可能导致导入文本文件变慢。考虑选择**扫描 5 秒**。 文本文件包含缺失数据的列时，选择**将空列视为数值列**可将这些列作为数值列而非字符列导入。点、Unicode 点、NaN 或空字符串为可能的缺失值指示符。默认取消选择该选项。

表 13.9 文本文件的“导入设置”首选项 (续)

首选项	说明
两位数年份规则	选择要用于导入日期的规则，该日期使用两位数而非四位数的年份。“2000-2099”是默认设置。 有关这些规则的详细信息，请参见“导入数据”一章中第 82 页的“两位数年份规则”。
尝试压缩	选择用于压缩文本文件的选项。提供了以下选项，所有选项在默认情况下均取消选定： • 数值列 • 字符列 • 允许列表检查 **注意：**该功能需要对整个文件进行扫描。
去除外围引号	选择该选项可删除文本文件中括起数据的引号。默认选择该选项。
将撇号识别为引号	选择该选项可将撇号视为引号并忽略它们。默认取消选择该选项。 **注意：**不推荐使用该选项，除非您的数据来自非标准源，该源使用撇号而非引号括起数据字段。
使用区域设置	选择该选项可在导入文本文件时使用操作系统的区域设置。 • 若取消选定该选项（默认设置），用句点表示小数点和用逗号表示值分隔符的文件将正确导入。 • 若该文件用逗号表示小数点并使用其他某种值分隔符（并且区域设置用逗号表示小数点），则选择该选项可正确导入文本。您必须在“文本数据文件”导入首选项中指定值分隔符。

表 13.10 文本文件的“导出设置”首选项

首选项	说明
导出表标题	选择该选项可在将数据表保存为文本文件时包括列名。默认选择该选项。
为所有列名添加引号	选择该选项可在列名周围插入引号。用于将数据导出到要求比 CSV 更严格的程序中。默认取消选择该选项。
为所有字符值添加引号	选择该选项可在字符值周围插入引号。用于将数据导出到要求比 CSV 更严格的程序中。默认取消选择该选项。

表 13.10 文本文件的“导出设置”首选项 (续)

首选项	说明
为所有数值添加引号	选择该选项可在数值周围插入引号。用于将数据导出到要求比 CSV 更严格的程序中。默认取消选择该选项。
字段结尾	选择一个或多个字符，用作导出文本数据时指示字段结尾的分隔符。逗号是默认设置。 选择**其他**选项并输入一个字符可指定未列出的分隔符。
行结尾	选择一个或多个字符，用作指示行结尾的分隔符。<CR>+<LF> 是默认设置。 选择**其他**选项并输入一个字符可指定未列出的分隔符。

第三方数据的首选项

这些首选项指定第三方数据（例如 Microsoft Excel 和 SPSS 文件）的设置。

图 13.11 “第三方数据”首选项

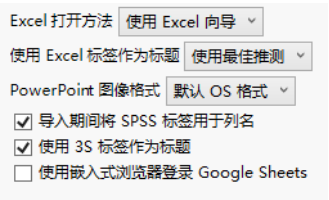


表 13.11 “第三方数据”页上的首选项

首选项	说明
Excel 打开方法	选择以下其中一个选项可确定 JMP 导入 Microsoft Excel 工作表的方式。 **Excel 向导** 可让您在导入数据前预览 Microsoft Excel 工作表并修改设置。在 Windows 上，该选项是默认设置。在 macOS 上，所有 xlsx 文件都会在 “Excel 向导”中打开。 **打开所有工作表** 打开工作簿中的所有工作表。在 macOS 上，该首选项仅适用于 .xls 文件。 **选择单个工作表** 可让您选择要从工作簿打开的工作表。在 macOS 上，该首选项仅适用于 .xls 文件。当您在 macOS 上打开 Excel (.xls) 文件时也提供该选项。 在 Windows 上，您还可以选择**文件 > 打开**，选择工作簿，点击**打开**按钮箭头，然后选择**打开选定的工作表**或**使用 Excel 向导**。
使用 Excel 标签作为标题	选择该选项可允许 JMP 在导入 Excel 文件时将 Excel 标签名称用作列标题。可用设置包括： • 使用最佳推测（默认） • 始终 • 从不
PowerPoint 图像格式	为导出到 Microsoft PowerPoint 的图像选择默认文件格式。 Windows 不支持在 macOS 上生成的本机 PDF 图形。macOS 不支持在 Windows 上生成的本机 EMF 图形。为了实现跨平台兼容，请指定 PNG 或 JPEG 图形格式。通过编写脚本可提供其他文件格式。请参见《JSL Syntax Reference》。
导入期间将 SPSS 标签用于列名	选择该选项可允许 JMP 在导入 SPSS 文件期间将 SPSS 标签名称用作列标题。
使用 3S 标签作为标题	选择该选项可允许 JMP 将 3S 标签名称用作列标题。
使用嵌入式浏览器登录 Google Sheets	选择该选项可使用嵌入式 JMP 浏览器登录到 Google Sheets，而不是默认的系统浏览器。

Windows 首选项

Windows 特定的首选项用于定制针对 Windows 计算机的设置，包括自动隐藏菜单和工具栏，以及选择默认图形格式。

图 13.12 “Windows 特定”首选项

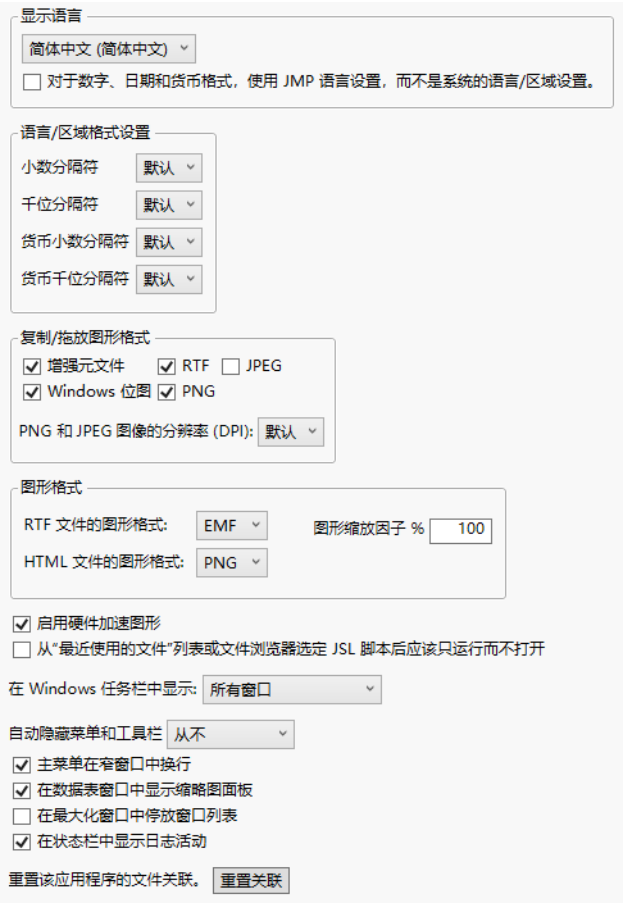


表 13.12 “Windows 特定”页上的首选项

首选项	说明
显示语言	选择要运行 JMP 所用的语言。 操作系统的语言/区域设置通常确定数字、日期和货币格式的设置。选择语言菜单下面的选项可以改用该语言来确定这些格式。 在操作系统提供的窗口中显示的文本（例如，**文件 > 打开**）并不反映语言设置中的变化。 **注意：** • 更改显示语言后，关闭并重新启动 JMP 可使语言设置完全生效。 • 您在 JMP 安装程序中选择的语言确定哪些语言可用。要令某种语言可用，重新运行 JMP 安装程序，选择“修改”，然后选择该语言。
对于数字、日期和货币格式，使用 JMP 语言设置，而不是系统的语言/区域设置	使用 JMP 的显示语言来指定数字、日期和货币格式。
语言/区域格式设置	指定用于数字和货币值的小数和千位分隔符。“默认”设置使用系统语言/区域和“显示语言”首选项中的设置。若指定了非默认设置，则该设置将覆盖“显示语言”首选项和系统语言/区域。
复制/拖放图形格式	选择将图形从 JMP 复制并粘贴（或拖放）到其他应用程序时要使用的一个或多个图形格式。默认选定增强元文件、Windows 位图、RTF 和 PNG。PNG 和 JPEG 图像的分辨率是“默认值”。

PNG 和 JPEG 图像的分辨率 (DPI)	指定复制图形格式时要使用的 DPI。选择默认值 (96) 或 300。300 适用于必须拉伸、嵌入商业出版物或打印的图像。但是，该设置占用更多的内存，对于大图像生成速度较慢。
	以下脚本支持您设置默认 DPI。根据需要将该数字改为不同的 DPI，然后运行该脚本。
	`Pref( Save Image DPI( 600 ) );`
	要在运行前面的脚本后验证默认 DPI，请运行以下脚本：
	`Get Preferences( Save Image DPI );`
	有关指定更高 DPI 的详细信息，请参见“保存和共享您的数据”一章中第 518 页的“设置导出图形的图形 DPI”。

表 13.12 “Windows 特定”页上的首选项

首选项	说明
图形格式	选择将 JMP 报表保存为 RTF 文件（EMF、PNG 或 JPEG）或 HTML 文件（PNG、JPEG、GIF 和 SVG）时要用于图形的格式。 在“图形缩放因子 %”框中，输入要在其他应用程序中显示图形的百分比。该功能可能不适用于某些版本的所选应用程序。“100%”是默认设置。
启用硬件加速图形	更好地利用计算机的图形处理单元 (GPU) 以提升视频性能。若您的计算机在绘制 JMP 窗口时速度较慢，请考虑选择该选项。这是一项试用功能。
从“最近使用的文件”列表或文件浏览器选定 JSL 脚本后应该只运行而不打开	选择该选项可在打开时强制运行所有脚本。若选择该选项，将不打开脚本的脚本窗口。默认取消选择该选项。
在 Windows 任务栏中显示	选择要在 Windows 任务栏中显示的 JMP 窗口。默认选择“所有窗口”。还可以选择显示 JMP 主窗口和数据表窗口。
自动隐藏菜单和工具栏	选择用于确定何时隐藏菜单和工具栏的规则。菜单和工具栏总是隐藏在报表和布局窗口中。对于其他窗口，将根据窗口的高度显示菜单和工具栏。默认设置为“从不”。
主菜单在窄窗口中换行	选择该选项可在窗口比菜单窄时用额外数行显示菜单。默认选择该选项。
在数据表窗口中显示缩略图面板	选择该选项可在数据表底部显示报表的缩略图面板。默认选择该选项。
在最大化窗口中停放“窗口列表”	选择该选项可在将 JMP 窗口最大化时自动停放窗口列表。默认取消选择该选项。
在状态栏中显示日志活动	选择该选项可在“日志”窗口中检测到活动时出现的状态栏中显示一个图标。随后可点击该图标将日志窗口调至最前。默认选择该选项。
重置该应用程序的文件关联	点击该按钮可将所有 JMP 文件类型与您正在运行的 JMP 版本关联。

Mac 操作系统首选项

“Mac 操作系统设置”首选项用于定制针对 macOS 机器的设置，包括图形格式和文件关联。

图 13.13 “Mac 操作系统设置”首选项

表 13.13 “Mac 操作系统设置”页上的首选项

首选项	说明
剪贴板和拖放操作的备用图像格式	选择将图形从 JMP 复制并粘贴（或拖放）到其他应用程序时要使用的一个或多个图形格式。默认设置为“TIFF”。
剪贴板图像缩放因子	输入要在其他应用程序中显示图形的百分比。默认设置为“200%”。
RTF 图像格式	选择将 JMP 报表保存为 RTF 文件时要用于图形的格式。默认设置为“PNG”。
RTF 图像缩放因子	输入要在 RTF 文档中显示图形的百分比。默认设置为“100%”。

表 13.13 “Mac 操作系统设置”页上的首选项 (续)

首选项	说明
HTML 图像格式	选择将 JMP 报表保存为 HTML 文件时要用于图形的格式。默认设置为“PNG”。
语言/区域格式设置	指定用于数字和货币值的小数和千位分隔符。默认设置使用系统语言/区域。若指定了非默认设置，则该设置将覆盖系统语言/区域。

“字体”首选项

“字体”首选项用于定制报表、数据表和脚本的外观，包括字体、文本大小和字体样式。

图 13.14 “字体”首选项

表 13.14 “字体”页上的首选项

首选项	说明
文本	设置 JMP 分析报表的文本部分的字体。默认设置为“Segoe UI，9 点”。
题头	设置分析报表和数据表中列标题的字体。默认设置为“Segoe UI，9 点粗体”。
标题	设置所有标题栏中显示的标题的字体。默认设置为“Segoe UI，11 点，粗体”。
小字体	设置小文本的字体，在数据网格的左上角使用该字体来显示列数和行数。默认设置为“Segoe UI，7 点”。
单字	设置 JMP 脚本语言 (JSL) 编辑器和编辑器框中用于脚本命令的字体。默认设置为“Consolas，10 点”。
公式编辑器	设置输入“公式编辑器”中的表达式的字体。默认设置为“Segoe UI，11 点”。
注解	设置在注解中使用的默认字体。默认设置为“Segoe UI，9 点”。
轴	设置用于轴刻度标签的字体。默认设置为“Segoe UI，9 点”。
标记	设置用于字母数字标记的字体。默认设置为“Segoe UI，9 点”。
轴标题	设置轴标题的字体。默认设置为“Segoe UI，9 点”。
图形标签	设置图形中数据标签（例如标记标签或直条标签）的字体。默认设置为“Segoe UI，9 点”。
图例	设置图例标题、值和其他图例文本的字体。默认设置为“Segoe UI，9 点”。
图形标题	设置主要图形标题但不包括次要标题或脚注的字体。默认设置为“Segoe UI，11 点，粗体”。
说明文字	设置说明文字元素以及任何类似说明文字的文本（例如，拟合统计量行和箱线图 5 值汇总）的字体。默认设置为“Segoe UI，8 点”。
数据表	设置用于在数据表中显示数据的字体。默认设置为“Segoe UI，9 点”。
字体系列	选择非等宽字体。所选的字体将立即应用到除单字（用于脚本）之外的所有设置。
使用希腊字母	选择该选项可使用希腊字母而非拼出希腊字母（例如，π 而不是 pi）。默认选择该选项。

表 13.14 “字体”页上的首选项 (续)

首选项	说明
使用数学符号	选择该选项可使用数学符号而非数学符号的简单文本表示法（例如，使用 ± 而非 +/-）。默认选定该选项。

“通信”首选项

（仅限 Windows）“通信”首选项用于定制从外部源读取数据的设置。仅当使用设备通信时，才需要指定这些设置。

图 13.15 “通信”首选项

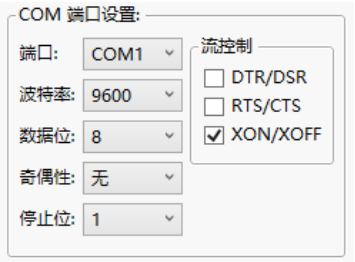


表 13.15 “通信”页面中的首选项

首选项	说明
端口	选择数据源使用的端口。
波特率	设置数据源的波特率。
数据位	设置数据位数（7 或 8）。
奇偶性	设置奇偶校验位。“无”是默认设置。
停止位	设置停止位（1 或 2）。
流控制	设置流控制。“XON/XOFF”是默认设置。

“Internet 选项”的首选项

“Internet 选项”指定用于从 Internet 打开文件的设置。

图 13.16 Internet 选项

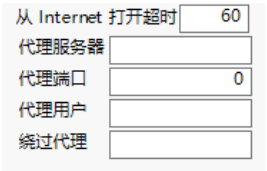


表 13.16 “Internet 选项”页上的首选项

首选项	说明
从 Internet 打开超时	当您使用“从 Internet 打开”打开网页时，JMP 会等待指定的秒数，之后才会因错停止导入。默认设置为“60 秒”。
代理服务器	指定代理主机。`http`（或 `https`）需要包括在规范中。若未使用协议，则假定采用 `http`。
代理端口	指定用于连接代理主机的端口。若未指定，则端口为 443（对于 `http`），对于其他所有 (`https`) 端口则为 443。默认设置为“0”。
代理用户	为代理指定可选用户。并非所有代理设置都需要用户。
绕过代理	指定要绕过的主机的可选逗号分隔列表（即，不通过代理）。用户可能希望过滤出位于公司防火墙（不需要代理）后面的主机。

“文件位置”的首选项

（仅限 Windows）“文件位置”首选项设置 JMP 系统文件的默认位置。

图 13.17 “文件位置”首选项

表 13.17 “文件位置”页面中的首选项

首选项	说明
数据文件目录	更改 JMP 在其中查找数据文件（例如，数据表）的文件夹。
许可文件路径	更改 JMP 在其中查找 JMP 许可文件的文件夹。
首选项文件目录	更改 JMP 在其中查找首选项信息的文件夹。
另存为目录	更改在您选择“文件 > 另存为”时 JMP 在其中保存数据文件的文件夹。
在显示“文件打开”窗口时，系统总是转至该目录	选择该选项后，当您选择“文件 > 打开”时，JMP 始终使用指定路径。

“JSL 脚本编辑器”的首选项

“脚本编辑器”首选项用于定制“JSL 脚本编辑器”和脚本编辑器框的外观，如制表符宽度、语法着色和工具提示。

图 13.18 “脚本编辑器”首选项

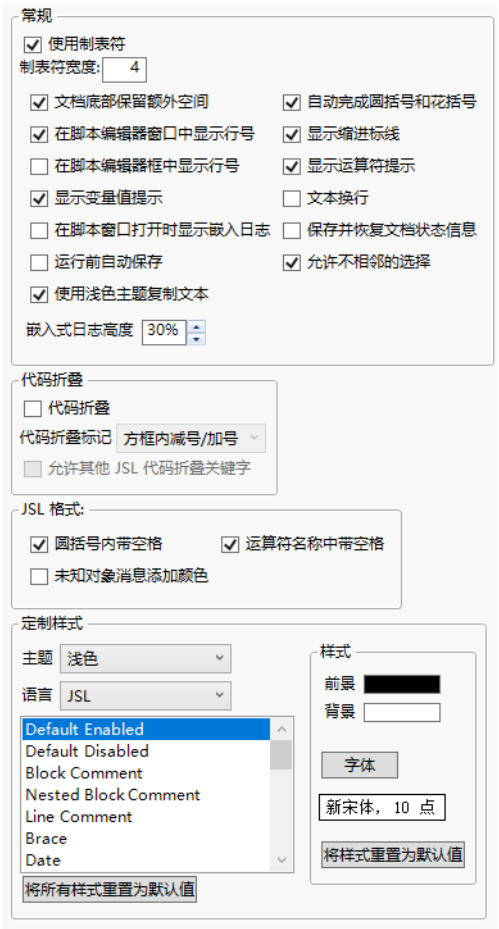


表 13.18 “脚本编辑器”页面中的首选项

首选项	说明
使用制表符	选择该选项可在脚本中使用制表符。默认选定该选项。 清除该选项将用空格替换您键入的任何制表符。

表 13.18 “脚本编辑器”页面中的首选项 (续)

首选项	说明
制表符宽度	输入一个制表符应缩进多少个空格。若禁用制表符，则用指定的空格数替换您键入的任何制表符。默认设置为“4”。
文档底部保留额外空间	选择该选项可允许从脚本的最后空白行向上滚动。在 Windows 上默认选择该选项，在 macOS 上则默认取消选择该选项。
自动完成圆括号和花括号	选择该选项可允许脚本编辑器在您键入圆括号、方括号和花括号的左半部分时自动添加相应的右半部分。默认选择该选项。
在脚本编辑器窗口中显示行号	选择该选项可在脚本编辑器的左侧显示行号。默认选择该选项。
显示缩进标线	选择该选项可查看标记缩进的模糊竖线。默认选择该选项。
在脚本编辑器框中显示行号	选择该选项可在脚本编辑器框（例如“脚本索引”）的左侧显示行号。默认关闭该选项。
显示运算符提示	选择该选项可查看 JSL 运算符的工具提示。默认选择该选项。
显示变量值提示	选择该选项可查看变量值的工具提示。默认选择该选项。
文本换行	选择该选项可始终在脚本编辑器中对文本换行。默认关闭该选项。
在脚本窗口打开时显示嵌入日志	选择该选项可在编辑或运行脚本时在脚本编写窗口中显示嵌入日志窗口。默认关闭该选项。
保存并恢复文档状态信息	保存折叠和展开代码的状态，并在脚本重新打开时恢复该状态。
运行前自动保存	运行脚本前自动保存该脚本。默认关闭该选项。
允许不相邻的选择	支持您选择不相邻的文本。在 Windows 上按 Ctrl 或在 macOS 上按 Command，然后用光标拖过该文本。继续为您想要选择的其他任何文本执行该操作。默认选择该选项。
使用浅色主题复制文本	确定是否将代码的背景色粘贴到其他文档。默认选择该选项。这意味着，当在“脚本编辑器”首选项中选定“浅色主题”时，代码将与浅色背景一起粘贴到另一个文档中。
嵌入式日志高度	指定脚本窗口中嵌入日志的高度。默认设置为“30%”。
代码折叠	选择该选项可在脚本编辑器中使用代码折叠标记，用于标记用户定义函数和 `Expr()` 块的开头和结尾。您可以展开和折叠这些标记的代码块。默认关闭该选项。 您还可以使用 JSL 代码折叠标记菜单选择标记的外观。请参见《Scripting Guide》。

表 13.18 “脚本编辑器”页面中的首选项 (续)

首选项	说明
允许其他 JSL 代码折叠关键字	选择该选项可在脚本编辑器中使用折叠标记的其他关键字。请参见《Scripting Guide》。
圆括号内带空格	选择该选项可使脚本编辑器在圆括号、方括号和花括号与自动设置格式的脚本内容之间添加空格。默认启用该选项。
运算符名称中带空格	选择该选项可使脚本编辑器在运算符名称中的文字之间添加空格。例如，启用该选项时，使用 `New Window` 而非 `NewWindow`。默认选择该选项。
未知对象消息添加颜色	选择该选项可始终为未知对象消息添加颜色。未知对象消息将以“消息未知颜色”指定的颜色显示。默认关闭该选项。
定制样式	为 JSL 脚本、SAS 脚本、“文本分析器”定制窗口、文本文件等中显示的代码元素选择颜色和字体。“将所有样式重置为默认值”可令列表中的所有项恢复初始颜色。“将样式重置为默认值”可令右侧的选定项恢复初始颜色和字体。 请注意，“重置字体”首选项不影响该页上选定的字体。

“SAS 集成”首选项

“SAS 集成”首选项用于定制使用 SAS 服务器的默认设置。有关使用 SAS 集成功能的详细信息，请参见“导入数据”一章中第 124 页的“从 SAS 导入数据”。

注意：当您从“SAS 服务器版本”列表中选择 SAS 9.3 或 9.4 后，将显示“SAS 环境”选项。

图 13.19 “SAS 集成”首选项

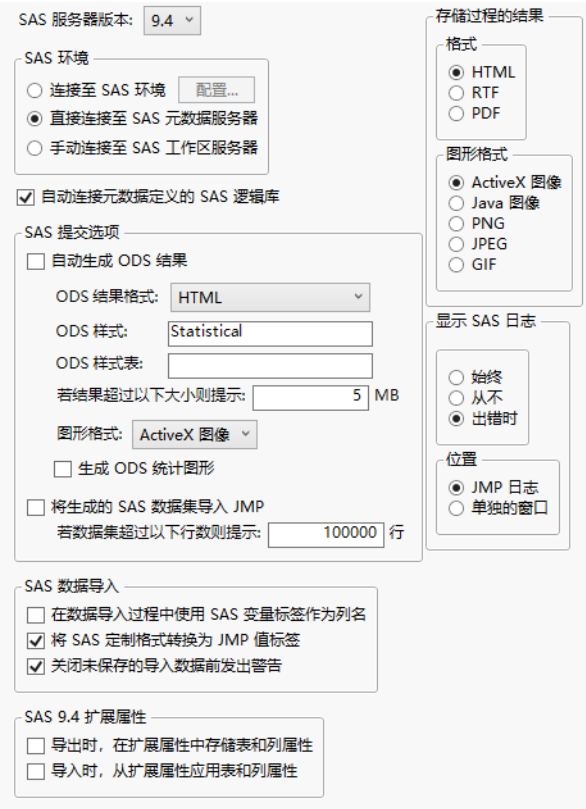


表 13.19 “SAS 集成”页面中的首选项

首选项	说明
SAS 服务器版本	选择 SAS 服务器的默认版本。也可以在“服务器连接”窗口中更改该信息。（请参见“导入数据”一章中第 128 页的“从 JMP 连接至 SAS 元数据服务器”。）连接到 Windows 上的本地 SAS 时，该选项不适用。默认设置为“9.4”。
连接至 SAS 环境	选择该选项可始终连接至 SAS 环境，点击**配置**可配置位置的 URL。默认取消选择该选项。
我要连接至 SAS 元数据服务器	选择该选项可始终连接至 SAS 元数据服务器。默认选定该选项。 若您没有可用的 SAS 元数据服务器则清除该选项，您将直接连接至 SAS 工作区服务器。
手动连接至 SAS 工作区服务器	选择该选项可手动连接至 SAS 工作区服务器。默认取消选择该选项。

表 13.19 “SAS 集成”页面中的首选项 (续)

首选项	说明
自动连接元数据定义的 SAS 逻辑库	选择该选项可在您连接至 SAS 工作区服务器时自动在所有 JMP 会话中连接至元数据定义的 SAS 逻辑库。默认选定该选项。 工作区服务器包含大量元数据定义的 SAS 逻辑库时，取消选择该选项可提高连接到该服务器的速度。
自动生成 ODS 结果	选择该选项可生成输出交付系统 (ODS) 结果。默认取消选择该选项。
ODS 结果格式	选择 ODS 报表的格式。默认设置为“HTML”。
ODS 样式	输入 ODS 报表的样式名称。默认设置为“统计”。
ODS 样式表	输入 ODS 报表的样式表名称。
若结果超过以下大小则提示: __ MB	输入触发提示您继续或取消操作的 MB 数。“5”是默认设置。
图形格式	选择 ODS 报表的图形格式。“ActiveX 图像”是默认设置。
生成 ODS 统计图形	选择该选项可在 ODS 报表中包括统计图形。
将生成的 SAS 数据集导入 JMP	选择该选项可自动将生成的任何数据集导入 JMP。
若数据集超过以下行数则提示: __ 行	输入触发提示您继续或取消操作的行数。默认设置为“100,000”。
在数据导入过程中使用 SAS 变量标签作为列名	在将 SAS 数据集导入 JMP 数据表时，选择该选项可使用 SAS 数据集中的列标签作为 JMP 数据表列名。默认取消选择该选项。
将 SAS 定制格式转换为 JMP 值标签	选择该选项可在将 SAS 数据集导入 JMP 数据表时，使用 SAS 列格式中的信息设置 JMP 值标签。默认选择该选项。
关闭未保存的导入数据前发出警告	选择该选项可触发一个提示，要求您保存在关闭前已导入 JMP 但是尚未保存的任何 SAS 数据集。默认选择该选项。
导出时，在扩展属性中存储表和列属性	选中该选项可允许在 SAS 9.4 数据导出过程中支持扩展属性。默认取消选择该选项。
导入时，从扩展属性应用表和列属性	选中该选项可允许在 SAS 9.4 数据导入过程中支持扩展属性。默认取消选择该选项。
存储过程的结果：格式	选择一种报表格式：HTML（默认）、RTF 或 PDF。
存储过程的结果：图形格式	选择一种图形格式：ActiveX 图像（默认，仅在 Windows 上运行的 SAS 服务器上可用）、Java 图像、PNG、JPEG 或 GIF。

表 13.19 “SAS 集成”页面中的首选项 (续)

首选项	说明
显示 SAS 日志	选择“始终”、“从不”或“出错时”（默认）以设置何时显示 SAS 日志
位置	选择 SAS 日志信息的位置：JMP 日志窗口内还是单独的 SAS 日志窗口。默认设置为“SAS 日志窗口”。

用于更新 JMP 的首选项

“JMP 更新”首选项显示您当前的 JMP 版本、上次更新的时间、JMP 是否自动检查更新以及检查频率。

图 13.20 “JMP 更新”首选项

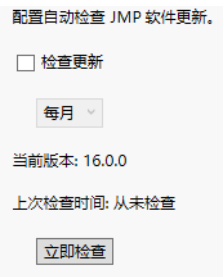


表 13.20 “JMP 更新”页面中的首选项

首选项	说明
检查更新	选择该选项可使 JMP 自动检查更新。 若选择该选项，请指定执行检查的频率：每天、每周或每月。
当前版本	您当前正在运行的 JMP 版本。
上次检查时间	上次检查更新的时间。
立即检查	点击该按钮可立即执行更新检查。

“JSL 调试器”的首选项

“JSL 调试器”首选项允许您显示或隐藏行号、在脚本中创建中断以及显示警告。

图 13.21 “JSL 调试器”首选项

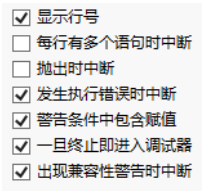


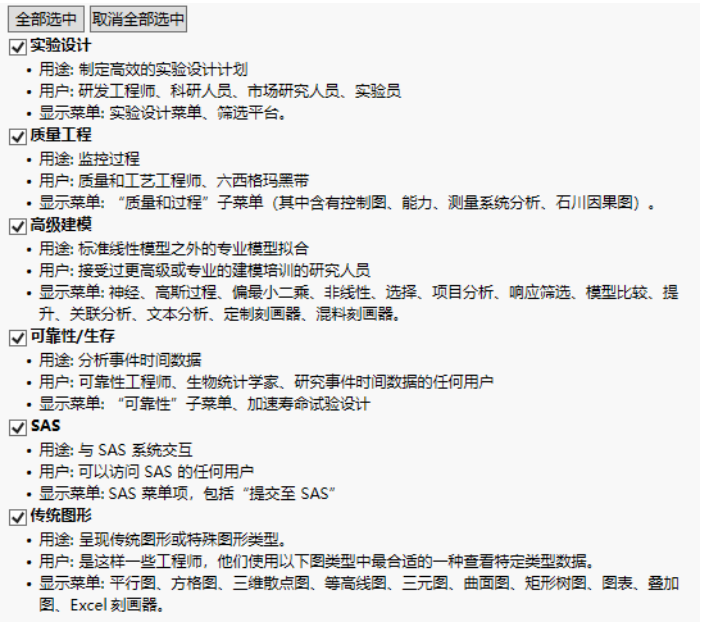
表 13.21 “JSL 调试器”页面中的首选项

首选项	说明
显示行号	在调试器中显示或隐藏脚本的行号。默认选择该选项。
每行有多个语句时中断	在单行的每个表达式之间停止执行脚本。默认取消选择该选项。
抛出时中断	当 Try() 函数出现异常错误时中断。例如，Try() 函数中的异常错误被 Throw() 函数抛出。调试器将在遇到 Throw() 时中断而非继续执行表达式的其余部分。这使您可以识别脚本中出现问题的位置，然后返回进行调试。默认取消选择该选项。
发生执行错误时中断	发生错误时停止执行脚本而非关闭调试器。默认选定该选项。
警告条件中包含赋值	选择该选项可在您为断点条件输入赋值表达式时显示一条警告。例如，若您在断点条件中指定 x = 1 而非 x == 1，系统将提示您验证指定。默认选定该选项。
一旦终止即进入调试器	选择该选项可在 JMP 终止时进入调试器。默认选定该选项。
出现兼容性警告时中断	遇到 JSL 兼容性问题时停止执行脚本。默认选定该选项。

“菜单”首选项

“菜单”首选项基于您使用 JMP 的方式显示和隐藏菜单。这使您浏览更少的菜单项并简化 JMP 界面。例如，若您从不设计实验，可以取消选择**实验设计**。其他菜单按关注区域分组（如质量工程、高级建模、可靠性和生存以及 SAS 选项）。

图 13.22 “菜单”首选项



“查询生成器”的首选项

“查询生成器”首选项用于定制您在 “查询生成器”中执行的 SQL 查询。

图 13.23 “查询生成器”首选项

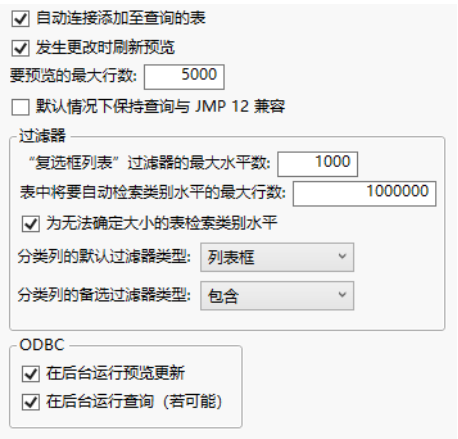


表 13.22 “查询生成器”页面中的首选项

首选项	说明
自动连接添加至查询的表	根据键和常用列名在“查询生成器”中自动连接数据库表。默认选定该选项。取消选择该选项可避免使用大型数据库出现内存问题。
发生更改时刷新预览	发生更改时，更新“查询预览”选项卡上的数据视图。默认选择该选项。
要预览的最大行数	限制在“查询预览”选项卡上显示的行数。默认数字为 5000。
默认情况下保持查询与 JMP 12 兼容	保持查询与 JMP 12 兼容。在 JMP 13 或更高版本中打开标记为 JMP 12 兼容的查询时，无论该首选项如何设置都会隐藏引起兼容性问题的功能。
“复选框列表”过滤器的最大水平数	确定分类变量的过滤器是否添加为“复选框列表”过滤器。默认情况下，若您添加包含 1,000 个以上水平的分类变量时，该过滤器将添加为“包含”过滤器。
表中将要自动检索类别水平的最大行数	指定要检索过滤列中的唯一类别水平的最大行数。最小值为 -1（无限制）。最大值为 10 亿。默认情况下，若分类列超过一百万行，JMP 不会自动检索唯一类别水平。
为无法确定大小的表检索类别水平	确定在无法确定行数时是否为过滤列检索类别水平。默认选定该选项。若取消选择该选项，该过滤器将添加为“包含”过滤器。
分类列的默认过滤器类型	将“列表框”指定为分类列的默认过滤器类型。
分类列的备选过滤器类型	指定在无法确定行数或是取消了查询时，由“包含”过滤器充当分类列的默认过滤器类型。
在后台运行预览更新	更新数据视图“查询预览”选项卡而不显示状态，以便在查询运行时继续工作。默认选择该选项。
在后台运行查询（若可能）	运行数据库查询而不显示状态，以便在查询运行时继续使用“查询生成器”。默认选择该选项。

用于对列重新编码的首选项

“重新编码”首选项定制列的重新编码。

图 13.24 “重新编码”首选项

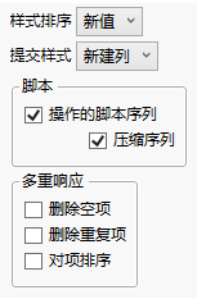


表 13.23 “重新编码”页上的首选项

首选项	说明
样式排序	指定默认情况下如何对值进行排序。 **新值** 按新值升序排序。 **旧值** 按旧值升序排序。 **计数** 按降序排序，最常用的值位于顶部。
提交样式	定义用于对列重新编码的默认行为。 **新建列** 将值放入新列是默认设置。 **公式列** 将值放入公式列。若您想查看如何替换这些值，请选择该选项。 **原位** “原位”替换重新编码列。
操作的脚本序列	将您的所有操作保存在脚本中。默认选择该选项。若取消选择该选项，则脚本中仅包含原始值和替换值。
压缩序列	组合表达式以简化保存的脚本。当您选择**操作的脚本序列**时默认选定该选项。
删除空项	在“多重响应”列中，对数据表中的列重新编码时排除该列中的空项。默认取消选择该选项。
删除重复项	在“多重响应”列中，排除重复项。例如，若原始数据为 a、b、b、c，并且没有对这些特定项进行更改，则数据表中的重新编码列将包含 a、b、c。默认取消选择该选项。

表 13.23 “重新编码”页上的首选项 (续)

首选项	说明
对项排序	在“多重响应”列中，按字母顺序对各项排序。例如，若原始数据为 b、a、c，并且没有对这些特定项进行更改，则数据表中的重新编码列将包含 a、b、c。默认取消选择该选项。

“列切换器”的首选项

“列切换器”首选项定制用于列切换器实用工具的设置。

图 13.25 “列切换器”首选项

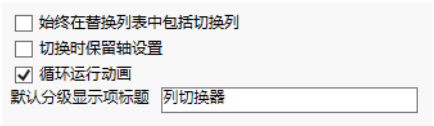


表 13.24 “列切换器”页上的首选项

首选项	说明
始终在替换列表中包括切换列	若选择该选项，则当前列将自动包含在“列切换器”的列列表中。若取消选择该选项（默认设置），在指定的“列切换器”不包括当前列时会显示警告。
切换时保留轴设置	指定在“列切换器”中切换列时，轴设置应保持不变。默认取消选择该选项。
循环运行动画	指定动画循环结束时的默认行为。若选择该选项（默认设置），则动画将在到达循环末尾时从头重新开始。若取消选择该选项，动画将在循环结束时停止。
默认分级显示项标题	指定“列切换器”的标题。默认设置为“列切换器”。

“日志”首选项

“日志”首选项指定在日志中记录哪些交互式 JMP 操作、日志模式以及如何定制 “增强日志”的外观和行为。日志的预览将显示在首选项选项的右侧。

图 13.26 “日志”首选项

表 13.25 “日志”页上的首选项

首选项组	说明
操作录制	包含用于控制日志中记录的操作类型的选项。您可以选择记录针对数据表的操作、平台的启动，以及关闭报表时每个报表的脚本。“操作录制”首选项同时适用于日志的两种模式。默认选定所有选项。
模式	允许您在“增强日志”和“文本日志”（默认设置）之间切换。

表 13.25 “日志”页上的首选项 (续)

首选项组	说明
表	包含用于控制“增强日志”内容外观的选项。默认情况下选定“表标题加下划线”、“表标题着色”、“表标题列边框”和“表行加边框”。
过滤器	包含用于控制“增强日志”中显示的输出类型的选项。默认选定所有选项。
列	包含用于控制“增强日志”中显示哪些信息列的选项。可以显示或隐藏以下信息列：消息的源类型、消息的结果类型和消息的时间戳。默认选定所有选项。
按窗口着色	指定“增强日志”中的行是否由创建日志输出行的 JMP 窗口着色。默认选择该选项。“JMP 默认”是默认颜色主题。

附录 A

在公式中引用 JMP 函数

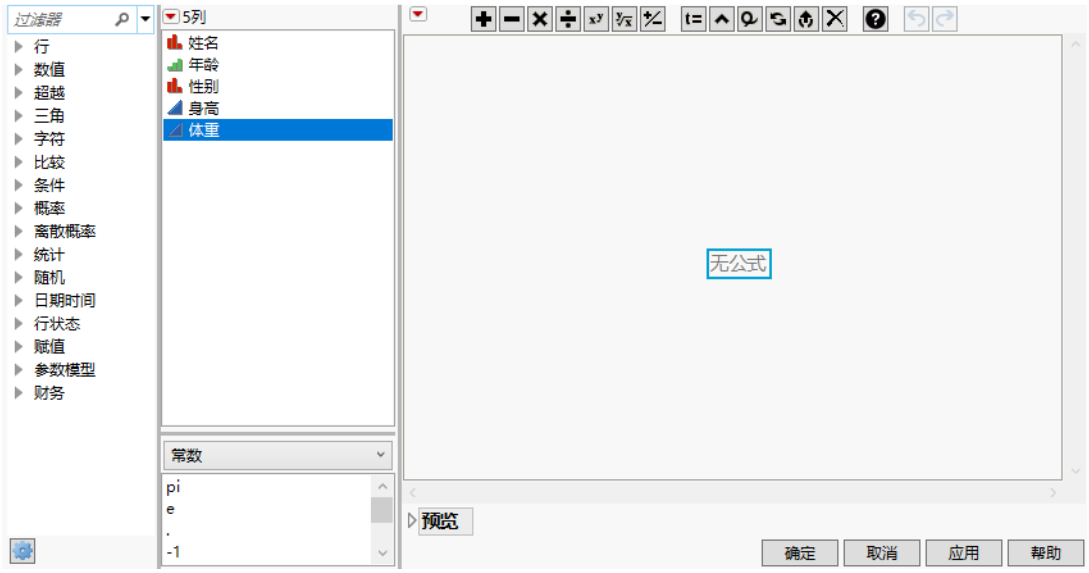
公式编辑器中函数的说明

您可以将函数添加到公式。所有这些函数都在“公式编辑器”中的“函数”列表中提供。本章将介绍“公式编辑器”中的函数。

以下资源中可找到有关函数的详细信息：

- “脚本索引”介绍所有函数及其参数，说明函数的工作原理，并提供联机帮助的链接。在 JMP 中，选择**帮助 > 脚本索引**可查看该交互式资源。
- 《JSL Syntax Reference》还提供 JMP 函数的参数，而不只是提供“公式编辑器”中可用的那些函数参数。在 JMP 中，选择**帮助 > JMP 帮助**并在左列中找到《JSL Syntax Reference》。

图 A.1 “公式编辑器”中的函数



有关如何创建包含函数的公式的说明，请参见“在 JMP 中创建公式”一章中第 342 页的“生成公式”。

目录

行函数 633
数值函数 635
超越函数 635
三角函数 639
字符函数 639
比较函数 644
条件函数 645
概率函数 650
离散概率函数 657
统计函数 658
随机函数 662
日期时间函数 667
行状态函数 670
赋值函数 675
参数模型函数 675
财务函数 676

行函数

通过将行函数添加到公式，您可以引用特定行或特定行中的单元格。您还可以基于等差序列插入值。有关语法的详细信息，请参见《Scripting Guide》。

Col Stored Value

返回未应用列属性的列值。若未指定行，则使用当前行。

假定将“缺失值代码”列属性分配给 x1 列，将“999”视为缺失值。另一列包含计算均值的公式。要使用值“999”而不是缺失值计算均值，请在公式中使用 Col Stored Value：

```
Mean( Col Stored Value( :x1 ), :x2, :x3 )
```

Count

创建从 from 值开始到 to 值结束的值的列表。steps 数指定列表中 from 和 to 值之间（包括端点）的值数。由 count 函数的前三个参数确定的每个值连续出现使用 times 参数指定的次数。达到 to 值时，Count 从 from 值重新开始。

您还可以使用键盘上的插入按钮 添加 times 参数。该参数默认为 1，但是按您指定的次数重复计数过程，如图 A.2 中数据表的 Count4 列所示。要将任意参数添加到 Count 函数，请突出显示位于您要输入参数前面的参数。键入一个逗号或使用“公式编辑器”小键盘上的插入按钮 。

下面数据表中的各列从以下公式得到：

- Count (1, 9, 2) 得到 Count 1
- Count (1, 9, 3) 得到 Count 2
- Count (1, 9, 9) 得到 Count 3
- Count (1, 9, 3, 3) 得到 Count 4

图 A.2 Count 函数的示例

	count1	count2	count3	count4
1	1	1	1	1
2	9	5	2	1
3	1	9	3	1
4	9	1	4	5
5	1	5	5	5

Count 函数对生成一列网格值很有用。例如，以下公式创建一个 NRow() 增量的正方形网格。NRow() 是给出数据表中总行数的 Row 函数。NRow() 还创建范围从 –5 到 5 的轴：

```
Count(-5, 5, Root( NRow() ) );
Count(-5, 5, Root( NRow() ), Root( NRow() ) );
```

Dif

返回当前行中第一个参数的值与当前行减去第二个参数所定义的行中值的差值。默认 Dif 为 1，您可以将其更改为任意数字。请注意，Dif(X, n) 所得的结果与 $X_{Row()}-X_{Row()-n}$ 或 $X_{Row()}-Lag(X, n)$ 所得的结果相同。

Dim

返回由当前数据表、指定数据表或矩阵的维度构成的行向量。维度是行数和列数并以该顺序列出。

Lag

返回当前行减去第二个参数所定义的行中第一个参数的值。默认 Lag 为 1，您可以将其更改为任意数字。为标识行号小于 1 的任意 Lag 返回的值为缺失。请注意，Lag(X, n) 所得的结果与下标表示法 $X_{Row()-n}$ 所得的结果相同。

NRow

返回活动数据表中的总行数。

Row

针对该行对表达式求值时返回当前行号。可以在任何表达式中使用 Row()，包括列名下标。除非特别指定，否则列名的默认下标为 Row()。

Subscript

使您可以使用非当前行的行中的列值。从列表中选择 Subscript 后，将数值表达式输入下标参数。针对不存在的行号求值的下标将生成缺失值。无下标的列名指代当前行。要删除某个下标，请选择该下标并删除它。然后删除缺失框。

公式 $Count_{Row()} - Count_{Row()-1}$ 使用下标计算 Count 列中每对值之间的差值，其中 Row() 是下面所述的行号。该结果与 Dif() 函数给出的结果相同。Row() 为 1 时，计算结果为缺失值。

下面的公式计算名为 Fib 的列，该列包含 Fibonacci 序列的项（每个值是计算列中的前面两个值之和）。

$$\text{If}\begin{cases} Row() <= 2 \Rightarrow 1 \\ \text{其他} \Rightarrow Fib_{Row()-1} + Fib_{Row()-2} \end{cases}$$

它显示如何使用下标来执行递归计算。递归公式包含计算列的名称，已加下标以便它只引用以前求值的行（第 1 行到第 (*i*–1) 行）。所示的 Fibonacci 序列的计算包含条件表达式和比较。请参见第 645 页的“条件函数”和第 644 页的“比较函数”这两节。

Sequence

生成数据表中各行的数字等差序列，其中开始值、结束限值和增量指定为参数。

数值函数

可以创建一个包含公式中常用的算术运算符的公式。有关语法的详细信息，请参见《Scripting Guide》。

Abs

返回与参数值同等大小的正数。例如，|5| 和 |–5| 的结果均为 5。

Ceiling

返回大于等于其参数的最小整数。例如，Ceiling(2.3) 的结果为 3，而 Ceiling(–2.3) 的结果为 –2。

Floor

返回小于等于其参数的最大整数。例如，Floor(2.7) 的结果为 2，而 Floor(–0.5) 的结果为 –1。

Integrate

使用 Gander 和 Gautschi 在 2000 年提出的自适应求积法，根据标量值对表达式求积分。例如，Integrate(Exp(-x), x, 0, .) 的计算结果为 0.99999999986846。

Modulo

返回第一个参数除以第二个参数的余数。例如，Modulo(6, 5) 的结果为 1。

Round

将第一个参数舍入到第二个参数所指定的小数位数。例如，Round(3.554, 2) 舍入到 3.55，Round(3.555, 2) 舍入到 3.56。

超越函数

您可以创建支持超越函数的公式，这些函数包括以任意数为底的对数函数、用于组合计算的函数，Beta 函数和若干 gamma 函数等。有关语法的详细信息，请参见《Scripting Guide》。

Exp

对 *e* 进行指定的幂计算。例如，Exp(1) = e。

ExpM1

当 x 非常小时，返回计算更准确的 Exp(x) - 1。

Ln

计算 x 的自然对数。

Log 和 Log10

计算自然对数（底数为 e）。要更改默认底数，请突出显示该参数并键入一个逗号或点击“公式编辑器”小键盘上的插入键。将显示可编辑的底数。Log 参数可以为任意数值表达式。表达式 Log(e) 的求值结果为 1，Log(32,2) 的求值结果为 5。Log10 函数仅计算底数为 10 的对数。

Log1P

当 x 非常小时，返回计算更准确的 Log(1+x)。

Squash

计算函数 $1 / (1 + e^x)$，其中 x 为任意数值列、变量或表达式。

Logist

亦称 Squish 或 Logistic，是函数 $1 / (1+e^{-x})$ 的高效计算，其中 x 为任意数值列、变量或表达式。

Root (平方根)

根据指数的指定计算参数的根。Root 初始显示指数 2。要更改指数，请突出显示指数参数并输入所需的值。

Factorial

返回 1 到您指定的参数之间所有数的乘积。例如，Factorial(5) 求值结果为 120。

NChooseK

返回从 n 个元素中一次选取 k 个元素（n 个中选择 k 个）的所有组合数，按使用阶乘的标准方式计算：$n! / (k!(n – k)!)$。例如，NChooseK(5,2) 求值结果为 10。

Beta

添加双参数 Beta 函数，用 Gamma 函数表示为：

$$B(m, n) = \frac{\Gamma(m)\Gamma(n)}{\Gamma(m+n)}$$

Gamma

添加 Gamma 函数（记为 $\Gamma(i)$），定义为：

$$\Gamma(i) = \int_0^{\infty} (x^{i-1})(e^{-x})dx$$

单参数的 Gamma 与 Gamma(x, infinity) 相同。可选的第二个参数将整数上限从无限大更改为您输入的值。其他相关 gamma 函数关系为：

- 对于任意 $\alpha > 1$，$\Gamma(\alpha) = (\alpha-1) \bullet \Gamma(\alpha-1)$
- 对于任意正整数 n，$\Gamma(n) = (n-1)!$
- $\Gamma(0.5) = \pi$ 的平方根

LGamma

是 gamma 函数求值结果的自然对数。对 Gamma 函数进行 Log（自然对数）可以得到相同的结果。但是，LGamma 函数的计算效率比 Log（自然对数）和 Gamma 函数结合使用的效率高。使用 LGamma 函数实施 NChooseK。结果并不总是精确的整数。若结果接近某个整数，使用 Floor 函数将其舍入。

Digamma

Gamma 函数的对数导数。

Trigamma

Digamma 函数的导数或 Gamma 函数的对数二阶导数。

Arrhenius

计算 Arrhenius 关系的非特定成分，然后其与 Arrhenius 方程的活化能相乘。

$$\frac{11604.5181215503}{T + 273.15}$$

Arrhenius Inv

Arrhenius 函数的反函数：

$$\left(\frac{11604.5181215503}{y}\right) - 273.15$$

Logit

使用以下公式对参数应用 logit 变换：

$$\text{logit}(p) = \log\left(\frac{p}{(1-p)}\right)$$

Logit Percent

以百分比形式计算参数的 logit。

$$\text{LogitPct}(pct) = \text{对数}\left(\frac{\left(\frac{pct}{100}\right)}{1-\left(\frac{pct}{100}\right)}\right)$$

Logist Percent

以百分比形式计算参数的 Logistic。

$$\text{LogisticPct}(x) = \frac{100}{(1+e^{-x})}$$

Scheffe Cubic

在拟合某些模型时使用。Scheffe Cubic (X1, X2) 等效于 X1*X2*(X1-X2)。

SbInv

Johnson Sb 逆变换。若参数正常，结果为 Johnson Sb。

SbTrans

从双界变量到标准正态 (0, 1) 分布的 Johnson-Sb 变换。

SHASHInv

返回标准正态变量到 sinh-arcsinh (SHASH) 分布变量的变换。

SHASHTrans

返回 sinh-arcsinh (SHASH) 分布变量到标准正态分布变量的变换。

SlInv

Johnson Sl 逆变换。若参数正常，结果为 Johnson Sl。

SlTrans

从双界变量到标准正态 (0, 1) 分布的 Johnson Sl 变换。

SuInv

Johnson Su 逆变换。若参数正常，结果为 Johnson Su。

SuTrans

从双界变量到标准正态 (0, 1) 分布的 Johnson Su 变换。

三角函数

您可以创建支持三角函数（例如正弦和余弦）的公式。JMP 的三角函数需要所有角度参数以弧度为单位。有关语法的详细信息，请参见《Scripting Guide》。

Sine、Cosine、Tangent

Sine 和 Cosine 函数计算以弧度给出的相应参数的正弦和余弦值。例如，表达式 Sine(0) 的求值结果为 0，Cosine(0) 的求值结果为 1。tangent 函数计算以弧度给出的参数的正切值。表达式 Tan(Pi()/4) 的求值结果为 1。

ArcSine、ArcCosine、ArcTangent

ArcSine 和 ArcCosine 函数返回相应参数的反正弦和反余弦值。返回值以弧度为单位。例如，表达式 ArcSine(1) 和 ArcCosine(0) 的求值结果均为 1.57080。ArcTangent 函数返回参数的反正切值。返回值以弧度为单位。表达式 ArcTangent(1) 的求值结果为 0.78540 (=3.14159/4)。

SinH、CosH、TanH

SinH 和 CosH 函数返回相应参数的双曲正弦和双曲余弦值。表达式 SinH(1) 的求值结果为 1.175201，CosH(0) 的求值结果为 1.0。TanH 函数返回参数的双曲正切值。表达式 TanH(1) 的求值结果为 0.761594。

ArcSinH、ArcCosH、ArcTanH

ArcSinH 和 ArcCosH 函数返回相应参数的反双曲正弦和反双曲余弦值。表达式 ArcSinH(1) 的求值结果为 0.881374，ArcCosH(1) 的求值结果为 0。ArcTanH 函数返回参数的反双曲正切值。表达式 ArcTanH(0.5) 的求值结果为 0.549306。

字符函数

您可以创建一个公式，该公式接受字符参数或返回字符串，并将值的数据类型从数值转换为字符或从字符转换为数值。创建这些公式时，请注意：

- 字符函数可以生成字符或数值数据。若您计算不同于指定类型的数据类型，则自动更改计算列的数据类型以匹配结果。

- 作为文字字符串的参数必须用引号括起来。

有关语法的详细信息，请参见《Scripting Guide》。

Char

生成对应于数值参数中的各数字的字符串。例如，Char(1.123) 的求值结果为 1.123。请参见《Scripting Guide》。

Collapse Whitespace

去除首尾空格，并用一个空格替换内部的多个空格。即：若存在多个空格字符，Collapse Whitespace 命令会将两个空格替换为一个空格。

Concat ||

拼接字符串，通过将函数的第二个字符参数追加到第一个字符参数来生成一个新字符串。例如， "Dr." || " " || name 生成一个新字符串，它包含头衔 Dr. 后跟一个空格以及 name 字符串的内容。

Contains

返回第二个参数的第一个实例在第一个参数中的数值位置（若第二个参数存在）。第二个参数可以包含一个或多个字符。若第二个参数不存在，则 Contains 返回零。例如，Contains("Veronica Layman", "ron") 的求值结果为 3。Contains("Lillie Layman", "L") 的求值结果为 1。Contains("Lillie Layman", "Veronica") 的求值结果为 0。第三个参数**剩余偏差**是可选的。“剩余偏差”是一个用于指定搜索起始位置的数值。Contains("Lillie Layman", "L", 5) 的求值结果为 8，因为搜索从第五个位置开始，跳过了第一个“L”。若 offset 为负数，则 Contains 从距离字符串末尾 offset 个字符处向后搜索。

Munger

通过插入或删除字符从现有字符串计算新字符串。它还可以生成子字符串、计算索引和执行其他任务，具体取决于您指定参数的方式。Munger 函数将大写字母和小写字母作为不同字符处理。

Text 是字符表达式。Munger 将其他三个参数应用于该字符串以计算结果。

Offset 是数值表达式，指示要在字符串中搜索的开始位置。若 Offset 大于 find 参数的第一个实例的位置，则忽略第一个实例。

Find/Length 是字符或数值表达式。使用字符串作为搜索条件，或使用正整数返回从 Offset 位置开始的一定数目的连续字符。若指定负整数作为 Length 值，Munger 返回从 Offset 到字符串末尾的所有字符。

Replace（可选参数）可以为字符串或不指定。若它为字符串且 Find/Offset 值为数值，则 Munger 使用 Replace **字符串**替换搜索条件以生成结果。若 Find/Offset 值为数值且未指定字符串，Munger 将计算子字符串。若 Find/Length 值是字符串，则 Munger 始终返回数值偏移量，而忽略 Replace 值（若存在）。要插入 Replace 参数，请点击 Munger 函数中的任意参数，然

后点击插入按钮。按 Delete 键或点击“公式编辑器”小键盘上的删除按钮 (⌫) 可删除 Replace 参数。

Lowercase、Uppercase

Lowercase 函数将参数中找到的任何大写字符转换为相应的小写字符。例如，Lowercase("VERONICA LAYMAN") 的求值结果为 veronica layman。Uppercase 函数将参数中找到的任何小写字符转换为相应的大写字符。例如，Uppercase("Veronica Layman") 的求值结果为 VERONICA LAYMAN。

Length

计算参数的长度。例如，Length("Veronica") 的求值结果为 8。若参数为

- 字符串，则 length 返回字符数；
- 列表，则 length 返回列表中的项数；
- blob（二进制对象），则返回字节数。

Num

当字符串仅包含数字时，生成对应于字符串参数的数值。若字符串包含非数值，结果将为缺失值。例如，Num(“1.123”) 的求值结果为 1.123。

Substr

提取作为第一个参数的一部分的字符。从第二个参数指定的位置开始，基于第三个参数中指定的字符数结束。第一个参数可以为字符列或文字值。起始参数和长度参数可以为求值结果为数值的数值表达式。例如，要仅显示名字，Substr("Veronica Layman", 10, 6) 读取位置 10 到 15 的值，这将得到 Layman。

若 start 为负数，则 Substr 从距离字符串末尾 start 个字符处向后搜索。若 length 为负数或不存在，则 Substr 返回从 start 开始到文本字符串结束之间的字符串。

Substr 也可以结合列表使用。

Titlecase

将字符串转换为首字母大写，即：每个单词的首字符大写，后续字符均小写。例如，Titlecase (“Veronica Layman”) 的求值结果为 Veronica layman。

Trim

根据变元生成新字符串，删除所有首尾空格。第二个参数确定是从该字符串的**左**、**右**还是**两端**删除空格。若未使用第二个参数，则将同时删除两端的空格。例如，Trim("john ") 的求值结果为 john。Trim(" john ", both) 的求值结果也是 john。

Word

从字符串提取第 *n* 个字词。除非指定可选的 delimiters 参数，否则用一个或多个空格定义每个字词的开始位置和结束位置。例如，Word(2, "Veronica Layman") 返回字词 Layman。

要插入 delimiters 参数，请点击 Word 函数中的任意参数，然后点击“公式编辑器”小键盘上的插入按钮 。按 Delete 键或点击“公式编辑器”小键盘上的删除按钮 ( ) 可删除 delimiters 参数。若未指定分隔符，则将空格作为分隔符。若将分隔符定义为空字符串，则每个字符都被视为一个单字。

大多数特殊字符均作为单个分隔符。您可以输入任意字符或字符集作为字词分隔符。例如，要提取以下示例中的姓氏，请将逗号和空格一起作为分隔符并请求获取第一个字词。Word(1, "Layman, Veronica", ", ") 将返回字词 Layman。

Words

根据可选的第二个参数中所列的分隔符从 text 提取字词。默认分隔符为空格。例如，Words("the quick brown fox") 返回 {"the","quick","brown","fox"}。

若包括第二个参数，则提取该参数中的所有字符作为分隔符。例如，Words("Doe, Jane P.", ".") 返回 {"Doe","Jane","P"}。

要插入 delimiters 参数，请点击 Words 函数中的任意参数，然后点击“公式编辑器”小键盘上的插入按钮 。按 Delete 键或点击“公式编辑器”小键盘上的删除按钮 ( ) 可删除 delimiters 参数。若未指定分隔符，则将空格作为分隔符。若将分隔符定义为空字符串，则每个字符都被视为一个单字。

Left、Right

分别返回包含字符串 text 最左侧或最右侧的 *n* 个字符的子字符串。这两个函数也适用于列表。

Starts With、Ends With

若 whole 分别以 part 开头或结尾，则返回 1。否则，返回 0。这两个函数也适用于列表。

Item

由于处理字词分隔符的方式不同，它不同于 Word 函数。若多次找到分隔符或您输入包含多个字符的分隔符，Word 函数将它们视为单个分隔符。Item 函数使用每个分隔符定义新字词位置。为了进行比较，假定姓名采用姓氏, 名字的形式。分隔符为逗号后跟一个空格，例如：

Item(2, "Layman, Veronica", ", ")

Word(2, "Layman, Veronica", ", ")

Item 函数返回一个缺失值，因为它单独处理逗号和空格，发现在它们之间没有任何内容。Word 函数则将逗号和空格作为单个分隔符，找到 Veronica 作为第二个字词。

若未指定分隔符，则将空格作为分隔符。若将分隔符定义为空字符串，则每个字符都被视为单独的项。

Hex 和 Hex to Number

在 16 进制数与 16 进制数和数字之间转换。

Hex 返回参数的十六进制表示。若参数为字符（加引号），则结果为包含字符值的十六进制代码长度 2 倍的字符串。例如，Hex("A") 返回字符串 41。

若参数为数值且指定了“integer”，则 Hex 函数返回所返回整数的 8 位十六进制字符表示。例如，Hex(12, “integer”) 返回字符串 0000000C。

Hex to Number 将十六进制转换为数字。

请参见《Scripting Guide》。

Repeat

创建一个将第一个参数重复第二个参数所指定的次数所得的字符串。第一个参数可以为字符文字、字符变量或字符表达式。例如，Repeat(“Katie”, 3) 得到 KatieKatieKatie。

在 JSL 脚本中使用 Repeat 来重复矩阵时，将应用第三个参数。第一个参数为矩阵时，第二个参数按行重复，第三个参数按列重复。

Insert

Insert 将新项插入**列表**或**表达式**中的指定位置。若未指定位置，则在末尾插入。

Reverse

Reverse 函数颠倒字符串中的字符顺序。

Substitute

第一个参数为字符串，第二个参数为模式，第三个参数为替换字符串。Substitute 函数查找字符串中模式的所有匹配项并用替换字符串来替换它们。

Regex

第一个参数是 Regex 在其中搜索模式匹配项的源字符串。第二个参数是使用正则表达式表示的模式。“公式编辑器”提示您输入这两个必需的参数。

提示：有关使用正则表达式的详细信息，请在 Internet 上搜索**正则表达式教程**。

默认情况下，Regex 执行区分大小写的搜索并返回源字符串中与指定模式匹配的项（若匹配失败则返回缺失值）。可以添加两个可选的参数。您可以键入第三个参数 format，它指定要返回的字符串。若选择，可以使用正则表达式来指定返回的字符串中的替换文本。若指定第三个参数，则还可以指定 IGNORECASE 以便 Regex 在源字符串中搜索匹配项时忽略大写字母。

表 A.1 Regex 示例

Regex 函数示例	返回的字符串
`Regex( "@ q3 #", "([a-z])([0-9])" )`	q3 该函数区分大小写，因此 q3 是匹配项而 Q3 不是。
`Regex( "@ Q3 #", "([a-z])([0-9])", "\0",IGNORECASE)`	Q3 尽管 \0 是默认参数，但在该示例中是必需的，以便可以指定 IGNORECASE。
`Regex( "@ Q3 #", "([a-z])([0-9])", "\2\1",IGNORECASE)`	3Q

有关详细信息以及您可以运行的示例，请选择**帮助** > **脚本索引**然后搜索 Regex。

比较函数

您可以创建一个公式，该公式使用比较函数来比较两个参数的值。根据数值大小或字符顺序，每个比较关系的求值结果为**真**或**假**。真关系用 1 表示，假关系用 0 表示。

在条件表达式中包含比较时，比较很有用，但是若比较中的项都不缺失比较也可以独立作为数值表达式。关系符号的参数可以为任意两个表达式。但是，比较函数中的两个参数的数据类型必须相同。还请注意：

- 若在条件中使用单个“=”，JMP 将显示错误，因为它需要使用“==”。
- 比较字符串时，“公式编辑器”使用 International Utilities 程序包。该程序包包含每个国际字符集的不同排名并考虑了可区分的标记。
- 不应使用比较运算符来专门与缺失值比较。反而应使用 Is Missing 函数来检测缺失值。

有关语法的详细信息，请参见《JSL Syntax Reference》。

< 小于

> 大于

<= 小于等于

>= 大于等于

== 等于

!= 不等于

a<b<=c *b* 大于 *a* 且小于等于 *c*

a<=b<c *b* 大于等于 *a* 且小于 *c*

Is Missing 若当前行的参数值缺失，则返回 1，否则返回 0。

条件函数

可以在公式中包括条件表达式（简称为**条件**）。使用这些表达式，您可以生成一系列与结果表达式配对的子句。构造一系列子句可以让您根据条件将值赋给计算列中的单元格。条件遵循以下规则：

- 没有子句为真时，“公式编辑器”对伴随 else 子句的结果表达式求值。
- 条件表达式中所有结果表达式的求值结果必须为同一数据类型。
- 缺失项匹配任意数据类型。
- 从定义上来说，求值结果为零的表达式为假。
- 若表达式的求值结果为缺失值，则不会执行任何子句，只会返回缺失值。其他所有数值表达式都为真。

有关语法的详细信息，请参见《Scripting Guide》和《JSL Syntax Reference》。

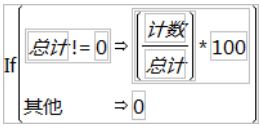
使用“公式编辑器”面板上的插入和删除子句按钮来展开表达式。为了提高效率，最先列出了最常求值的子句和结果对。

注意：Interpolate、Step、For 和 While 最常与其他命令一起使用来生成 JSL 脚本。您可以使用“公式编辑器”在该列中创建和执行脚本，但不推荐这样做，因为这会导致相关性问题并造成混淆。通常将脚本存储为 .jsl 文件，可以作为表属性与数据表一起保存。有关表属性的详细信息，请参见“开始使用 JMP”一章中第 37 页的“数据表中的表面板”。有关脚本命令的文档，请参见《JSL Syntax Reference》。

If

显示包含缺失表达式和缺失 then **子句**的单个 If 条件。突出显示 expr 或 then **子句**，然后输入值。例如，要计算合计不为 0 时计数占合计的百分比，请输入图 A.3 中的条件表达式（使用名为**计数**和**合计**的列）。

图 A.3 条件表达式



要向 If 条件中添加新条件，请突出显示 then **子句**，然后点击“公式编辑器”小键盘上的插入按钮 ( )。初始时它将现有 else 条件更改为 expr 子句。再次点击插入按钮可添加 else **子句**。突出显示 then 或 else 并重复点击插入按钮可将 else 更改为 expr 或添加新的 expr 子句。

要删除子句，请选择它上面的 then 子句，然后按 Delete 键或点击“公式编辑器”小键盘上的删除按钮 ( )。

从定义上来说，求值结果为零的表达式为假。若表达式的求值结果为缺失值，则不会执行任何子句，只会返回缺失值。其他所有数值表达式都为真。

Match

将表达式与子句列表进行比较，并为遇到的第一个匹配子句返回结果表达式的值。匹配表达式只提供一次，然后为每个子句指定匹配值。

在“公式编辑器”中选择 Match 后，将显示一个列表和两个选项：

- 选择**根据数据添加匹配参数**，将自动添加与您数据中所有水平对应的子句。或者，按 Shift 键，选择**条件**，然后选择 Match。在图 A.4 中，左边的示例显示自动添加的子句。
- 选择**请不要添加**，以便您可以单独添加每个子句。在图 A.4 中，右边的示例显示一个空子句，您可以使用缺失表达式填充它。

图 A.4 使用 Match 函数的示例

自动添加的 Match 参数

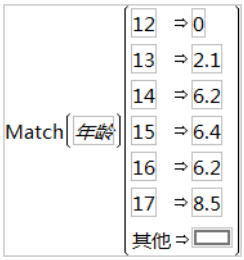
空 Match 参数

在自动填充的参数中，您应突出显示 then 子句，然后输入表达式。在空参数中，您突出显示 expr、value 或 then 子句，然后输入表达式。（或者，若您突出显示一个表达式，然后点击 Match，则“公式编辑器”创建新的 Match 条件，其中原始突出显示的表达式作为 expr 且不为 value 和 else 子句赋值。）另外，请记住：

- Match 与等效的 If 相比，求值速度更快，占用的内存更少，因为对于数据表中的每行仅对变量求一次值。而 If 条件遇到每行的每个 If 子句时都必须对变量求值，直到子句求值结果为真。有关 Match 和 If 条件的比较，请参见第 644 页的“比较函数”。
- 使用 If 和 Match，“公式编辑器”从序列顶部向下搜索第一个真子句，并对相应的结果表达式求值。后续真子句将忽略。

在以下示例中，根据**年龄**变量的值来赋每个值。

图 A.5 使用 Match 函数的示例



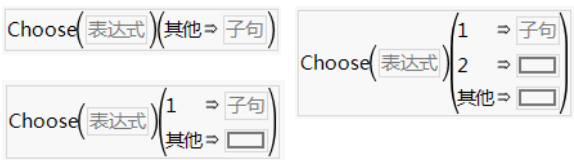
注意：Match 忽略尾部空格而 If 不忽略。

尽管 Match 对于任意缺失值返回“缺失”，您也可以专门匹配缺失值。

Choose

Choose 是 Match 的特殊形式，其中条件的参数为从 1 开始的一系列整数。**子句**的值替换匹配条件。图 A.6 中显示了 Choose 条件的示例。使用 Choose，“公式编辑器”直接转到正确的选择子句并对结果表达式求值。

图 A.6 Choose 条件的示例



突出显示一个表达式并点击 Choose 时，“公式编辑器”创建包含一个**子句**的新条件表达式。使用小键盘上的插入 ( ) 和删除 ( ) 按钮可添加新子句或删除不想要的子句，如前面 If 条件中所述。

Choose 对 choose 表达式求值并立即转到相应的结果表达式以生成返回值。使用 Choose，您只提供一次生成从 1 开始的一系列整数的选择表达式，然后对序列中的每个整数给出选择。

IfMax

对每对参数的第一个参数求值并返回与表达式最大值关联的结果表达式（每对参数的第二个参数）的求值结果。若多个表达式都为最大值，则返回第一个最大值。若所有表达式都缺失且未指定最终结果，则返回缺失值。若所有表达式都缺失但指定了最终结果，则返回最终结果。检验表达式的求值结果必须为数值，但结果表达式可以是任何值。

IfMin

对每对参数的第一个参数求值并返回与表达式最小值关联的结果表达式（每对参数的第二个参数）的求值结果。若多个表达式都为最小值，则返回第一个最小值。若所有表达式都缺失且未

指定最终结果，则返回缺失值。若所有表达式都缺失但指定了最终结果，则返回最终结果。检验表达式的求值结果必须为数值，但结果表达式可以是任何值。

And &

两个参数均为真时求值结果为 1。否则，求值结果为 0（图 A.9）。仅当两个比较结果都为真时，图 A.7 中的公式将“组 1”标记为驾驶员。

图 A.7 创建 And 函数

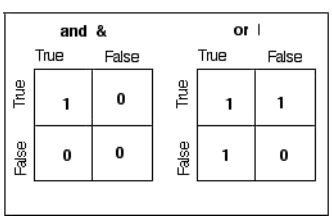
Or |

任一参数为真时求值结果为 1。若两个参数均为假，则 Or 表达式的求值结果为 0（图 A.9）。图 A.8 中的公式将所有大于 13 岁的男性参与者分配给“组 1”。

图 A.8 创建 Or 函数

图 A.9 中左边的真值表显示当两个参数具有求值结果为真或假的非缺失值时 And (&) 和 Or (|) 函数的结果。右边的表显示左边或右边的表达式（称它们为 a 和 b）或两者具有缺失值时的结果。

图 A.9 And 和 Or 表达式的求值



a	b	a|b	a & b
false	•	•	false
true	•	true	•
•	false	•	false
•	true	true	•
•	•	•	•

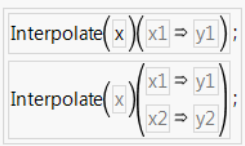
Not !

参数为假时求值结果为 1。否则，Not 的求值结果为 0。应用 Not 函数时，请根据需要使用括号以避免歧义。例如，`!(weight==64)` 可以为真或假（1 或 0），但是 `(!weight)==64` 始终为假 (0)，因为 Not 只能返回 0 或 1。诸如 `!(weight==64)` 的表达式还可以输入为 `weight != 64`。

Interpolate

在与您指定的参数对应的两个点 $(x1, y1)$ 和 $(x2, y2)$ 之间线性插入 y 值。可以使用插入键插入更多对 x, y 参数。Interpolate 查找与 x 值对应的 x, y 值对（点）并完成插值。

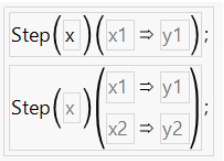
图 A.10 Interpolate 示例



Step

Step 与 Interpolate 类似，不同的是它返回与最大 x 值对应的 y 值，它们分别小于等于 x 和 y 参数。即它从 step 函数而非两个点之间的线性拟合查找对应指定 x 的 y。与 Interpolate 类似，您可以具有所需的任意多对 x 和 y 参数。

图 A.11 Step 示例



For

若 while 条件为真，则重复 body 参数中的语句。init 和 next 控制迭代。

While

重复检验 expr 条件并执行 body，直到 expr 不再为真。

Break、Continue

Break 完全停止执行循环，而继续执行循环后面的语句。Continue 结束循环的当前迭代并在下一迭代开始循环。

均用于 For、While 和 For Each Row 循环中。

Stop

立即停止正在运行的脚本。

Return

从用户定义的函数返回表达式值。

概率函数

您可以为统计分布生成用来计算概率和分位数的公式，这些分布包括 beta、卡方、F、gamma、正态、Student t 和 Weibull 分布。有关分布参数化和函数参数的详细信息，请参见《JSL Syntax Reference》。

Beta Density

有关 Beta Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Beta Distribution

有关 Beta Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Beta Quantile

有关 Beta Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Cauchy Density

有关 Cauchy Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Cauchy Distribution

有关 Cauchy Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Cauchy Quantile

有关 Cauchy Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Density

有关 ChiSquare Density 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Distribution

有关 ChiSquare Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Log CDistribution

有关 ChiSquare Log CDistribution 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Log Density

有关 ChiSquare Log Density 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Log Distribution

有关 ChiSquare Log Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Noncentrality

有关 ChiSquare Noncentrality 函数的语法的信息，请参见《JSL Syntax Reference》。

ChiSquare Quantile

有关 ChiSquare Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Dunnett P Value

有关 Dunnett P Value 函数的语法的信息，请参见《JSL Syntax Reference》。

Dunnett Quantile

有关 Dunnett Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Exp Density

有关 Exp Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Exp Distribution

有关 Exp Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Exp Quantile

有关 Exp Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

F Density

有关 F Density 函数的语法的信息，请参见《JSL Syntax Reference》。

F Distribution

有关 F Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

F Log CDistribution

有关 F Log CDistribution 函数的语法的信息，请参见《JSL Syntax Reference》。

F Log Density

有关 F Log Density 函数的语法的信息，请参见《JSL Syntax Reference》。

F Log Distribution

有关 F Log Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

F Noncentrality

有关 F Noncentrality 函数的语法的信息，请参见《JSL Syntax Reference》。

F Power

有关 F Power 函数的语法的信息，请参见《JSL Syntax Reference》。

F Quantile

有关 F Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

F Sample Size

有关 F Sample Size 函数的语法的信息，请参见《JSL Syntax Reference》。

Frechet Density

有关 Frechet Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Frechet Distribution

有关 Frechet Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Frechet Quantile

有关 Frechet Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Density

有关 Gamma Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Distribution

有关 Gamma Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Log CDistribution

有关 Gamma Log CDistribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Log Density

有关 Gamma Log Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Log Distribution

有关 Gamma Log Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Quantile

有关 Gamma Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

GenGamma Density

有关 GenGamma Density 函数的语法的信息，请参见《JSL Syntax Reference》。

GenGamma Distribution

有关 GenGamma Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

GenGamma Quantile

有关 GenGamma Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

GLog Density

有关 GLog Density 函数的语法的信息，请参见《JSL Syntax Reference》。

GLog Distribution

有关 GLog Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

GLog Quantile

有关 GLog Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Sb Density

有关 Johnson Sb Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Sb Distribution

有关 Johnson Sb Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Sb Quantile

有关 Johnson Sb Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Sl Density

有关 Johnson Sl Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Sl Distribution

有关 Johnson Sl Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Sl Quantile

有关 Johnson Sl Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Su Density

有关 Johnson Su Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Su Distribution

有关 Johnson Su Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Johnson Su Quantile

有关 Johnson Su Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

LEV Density

有关 Largest Extreme Value (LEV) Density 函数的语法的信息，请参见《JSL Syntax Reference》。

LEV Distribution

有关 Largest Extreme Value (LEV) Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

LEV Quantile

有关 Largest Extreme Value (LEV) Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

LogGenGamma Density

有关 LogGenGamma Density 函数的语法的信息，请参见《JSL Syntax Reference》。

LogGenGamma Distribution

有关 LogGenGamma Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

LogGenGammaQuantile

有关 LogGenGamma Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Logistic Density

有关 Logistic Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Logistic Distribution

有关 Logistic Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Logistic Quantile

有关 Logistic Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Loglogistic Density

有关 Loglogistic Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Loglogistic Distribution

有关 Loglogistic Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Loglogistic Quantile

有关 Loglogistic Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Lognormal Density

有关 Lognormal Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Lognormal Distribution

有关 Lognormal Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Lognormal Quantile

有关 Lognormal Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Biv Distribution

有关 Normal Biv Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Density

有关 Normal Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Distribution

有关 Normal Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Log CDistribution

有关 Normal Log CDistribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Log Density

有关 Normal Log Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Log Distribution

有关 Normal Log Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Mixture Density

有关 Normal Mixture Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Mixture Distribution

有关 Normal Mixture Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Mixture Quantile

有关 Normal Mixture Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Normal Quantile

有关 Normal Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

SEV Density

有关 Smallest Extreme Value (SEV) Density 函数的语法的信息，请参见《JSL Syntax Reference》。

SEV Distribution

有关 Smallest Extreme Value (SEV) Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

SEV Quantile

有关 Smallest Extreme Value (SEV) Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

SHASH Density

有关 SHASH Density 函数的语法的信息，请参见《JSL Syntax Reference》。

SHASH Distribution

有关 SHASH Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

SHASH Quantile

有关 SHASH Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

t Density

有关 t Density 函数的语法的信息，请参见《JSL Syntax Reference》。

t Distribution

有关 t Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

t Log CDistribution

有关 t Log CDistribution 函数的语法的信息，请参见《JSL Syntax Reference》。

t Log Density

有关 t Log Density 函数的语法的信息，请参见《JSL Syntax Reference》。

t Log Distribution

有关 t Log Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

t Noncentrality

有关 t Noncentrality 函数的语法的信息，请参见《JSL Syntax Reference》。

t Quantile

有关 t Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Tukey HSD P Value

有关 Tukey HSD P Value 函数的语法的信息，请参见《JSL Syntax Reference》。

Tukey HSD Quantile

有关 Tukey HSD Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Weibull Density

有关 Weibull Density 函数的语法的信息，请参见《JSL Syntax Reference》。

Weibull Distribution

有关 Weibull Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Weibull Quantile

有关 Weibull Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

离散概率函数

Beta Binomial Distribution

有关 Beta Binomial Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Beta Binomial Probability

有关 Beta Binomial Probability 函数的语法的信息，请参见《JSL Syntax Reference》。

Beta Binomial Quantile

有关 Beta Binomial Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Binomial Distribution

有关 Binomial Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Binomial Probability

有关 Binomial Probability 函数的语法的信息，请参见《JSL Syntax Reference》。

Binomial Quantile

有关 Binomial Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Poisson Distribution

有关 Gamma Poisson Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Poisson Probability

有关 Gamma Poisson Probability 函数的语法的信息，请参见《JSL Syntax Reference》。

Gamma Poisson Quantile

有关 Gamma Poisson Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

Hypergeometric Distribution

有关 Hypergeometric Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Hypergeometric Probability

有关 Hypergeometric Probability 函数的语法的信息，请参见《JSL Syntax Reference》。

Neg Binomial Distribution

有关 Neg Binomial Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Neg Binomial Probability

有关 Neg Binomial Probability 函数的语法的信息，请参见《JSL Syntax Reference》。

Poisson Distribution

有关 Poisson Distribution 函数的语法的信息，请参见《JSL Syntax Reference》。

Poisson Probability

有关 Poisson Probability 函数的语法的信息，请参见《JSL Syntax Reference》。

Poisson Quantile

有关 Poisson Quantile 函数的语法的信息，请参见《JSL Syntax Reference》。

统计函数

在公式中可以使用两种**统计**函数：

- 函数名称具有前缀 Col。这些函数计算一列数字或涉及列的表达式的统计值。

- Mean、Std Dev、Number、Sum、Quantile、Maximum、Minimum 和 N Missing 函数。这些函数针对一些列或参数求值。针对一系列参数的每行计算统计值。您可以点击“公式编辑器”小键盘上的插入键 ( ) 或键入逗号来将参数添加到接受多个参数的函数。有多个相邻参数时，选择函数和第一个参数，然后按 Shift 键并点击组中的最后一个参数。这些函数将自动随完整列表一起显示。

注意：统计函数中忽略缺失值。

有关语法的详细信息，请参见《JSL Syntax Reference》。

Col Cumulative Sum

返回当前行的累积和。Col Cumulative Sum 支持 By 列，无需对这些列进行排序。

Col Mean

计算参数所标识的数值的均值（或算术平均值）。公式 Col Mean(age) 计算 age 列中所有非缺失值的平均值。

Col Moving Average

返回基于当前行的给定区间的移动平均值。忽略缺失值。

Col Std Dev

测量参数所标识的分布针对均值的散布程度。在正态分布中，大约分布的 68% 位于均值两侧的一个标准差内。大约分布的 95% 位于均值两侧的两个标准差内。大约分布的 99% 位于均值两侧的三个标准差内。

Col Number

对指定列中的非缺失值数进行计数。当单元格没有分配值或为无效运算的结果（如被零除）时，会出现缺失数值。缺失值在电子表格上显示为缺失值标记 (•)。缺失字符值为空字符串。在行状态列的公式中，排除的行状态特征将视为缺失值。

Col N Missing

对指定列中的缺失值数进行计数。当单元格没有分配值或为无效运算的结果（如被零除）时，会出现缺失数值。缺失值在数据网格中显示为缺失值字符 (•)。缺失字符值为空字符串。

Col Sum

计算数值参数中值的和。忽略缺失值。

Col Minimum 和 Col Maximum

获取数值参数的最小值。Col Minimum 忽略缺失值。Col Maximum 获取数值列参数的最大值并忽略缺失值。

Col Quantile

计算值的特定百分比小于等于该值的值。例如，计算为 50% 分位数（也称为中位数）的值大于等于 50% 的数据。一半数据值小于第 50 分位数。

Col Quantile 函数的分位数参数表示除以 100 的分位数百分比。25% 分位数也称为下四分位数，对应于 $p = 0.25$，75% 分位数称为上四分位数，对应于 $p = 0.75$。

“公式编辑器”通过按升序排列值计算一列包含 n 个非缺失值的分位数。排序的列值下标 y_1、y_2,…,y_n 表示按升序排列的秩。

使用公式 $p(n + 1)$ 计算第 p 分位数值，其中 p 为百分比值，n 为非缺失值的总数。若 $p(n+1)$ 为整数，则分位数值为 $y_{p(n+1)}$。若 $p(n + 1)$ 不是整数，则通过以下方式将该值插值：结果的整数部分赋给 i 并将小数部分赋给 f，然后应用公式 $(1 - f)y_i + (f)y_{i+1}$。

例如，假定列包含值 2, 4, 6, 8, 10, 12, 14, 16, 18 和 20。50% 分位数的计算公式为：`0.5(10 + 1) = 5.5`。

由于结果为小数，则将 50% 分位数值插值为：

$$(1 – 0.5) \times 10 + (0.5) \times 12 = (0.5)10 + (0.5)12 = 6 + 5 = 11$$

以下是示例 ColQuantile 公式：

- ColQuantile(age, 1)　计算最大年龄。
- ColQuantile(age, 0.75)　计算上四分位数年龄。
- ColQuantile(age, 0.5)　计算中位数年龄。
- ColQuantile(age, 0.25)　计算下四分位数年龄。
- ColQuantile(age, 0)　计算最小年龄。

ColQuantile 参数可以是求值结果为 0 到 1（含端点）之间的值的任意表达式。例如，图 A.12 中的第一个公式计算每行按升序排列的年龄的分位数值。该列然后包含计算列中按升序排序的 age 插值。第二个公式列出按降序排列的年龄插值。

图 A.12 Quantile 函数的示例

Col Rank

从表示最低值的 1 到表示最高值的非缺失列数，对每行的值进行排名。可通过报告可能排名的平均值，按照结值最初出现的顺序分配排名，为两个值指定可能的最低排名或以任意方式指定，来打破并列关系。（JMP 12 使用任意方式打破并列关系。）分布的“正态分位数”输出使用平均值方式打破并列关系。

Col Standardize

对数值表达式执行常规标准化。对于每个行 i，Col Standardize(height) 为 ($Height_{Row()}$–Col Mean(Height))/Col Std Dev(Height)。

Mean

计算指定的参数列表中非缺失值的算术平均数。参数可以是常数、数字或表达式。Mean 函数最初显示带一个参数。您可以使用“公式编辑器”小键盘上的插入按钮 ( ) 或通过键入逗号添加参数。

Std Dev

计算指定的参数列表中非缺失值的标准差。参数可以是常数、数字或表达式。Std Dev 函数最初显示带一个参数。您可以使用“公式编辑器”小键盘上的插入按钮 ( ) 或通过键入逗号添加参数。

Number

计算指定的参数列表中非缺失值的数目。

Sum

计算指定的参数列表中非缺失值的总和。

Quantile

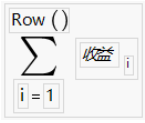
为所有非缺失值（在您指定的参数剩余列表中）计算第一个参数指定的分位数。

Summation (Σ)

按求和索引所指定的那样对列中值的显式范围进行求值。此行为不同于所有其他统计函数（Product 除外），它**始终**对**每行**求值。Summation 函数使用图 A.13 中所示的求和符号。 要求和，请用包含索引变量 i 的表达式或您分配的索引变量替换缺失的 body 项。Summation 对 $i = 1$、$i = 2$ 直到 i = NRow() 重复对表达式求值并将非缺失结果相加以确定最终结果。

您可以使用对公式合适的任意表达式替换 NRow()（活动电子表格中的行数）和索引常数 i。例如，图 A.13 中的求和公式计算第 1 行到当前行号每行的所有收入值的总和，使用收入列的累积总计填充计算列。

图 A.13 Summation 函数的示例



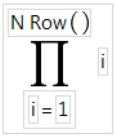
Product (Π)

按求和索引所指定的那样对列中值的显式范围进行求值，与所有其他统计函数不同（Summation 除外），它**始终**对**每行**求值。Product 使用图 A.14 中的公式所示的符号。 要计算乘积，请用包含索引变量 *j* 的表达式替换缺失的 body 项。Product 对 $i = 1$、$i = 2$ 直到 $i = n$ 重复对表达式求值并将非缺失结果相乘以确定最终结果。

您可以使用对公式合适的任意表达式替换 NRow()（活动电子表格中的行数）和索引常数 *i*。

例如，图 A.14 中的 product 示例计算 *i*!（每个行号的阶乘）。

图 A.14 Product 函数的示例



Minimum 和 Maximum

分别从指定的非缺失参数的列表返回最小值和最大值。

N Missing

计算指定的参数列表中缺失值的数目。

SSQ

返回所有元素的平方和。

随机函数

您可以创建公式，这些公式通过在指定的分布约束内高效“掷骰子”来生成随机数。每次在“公式编辑器”窗口中点击**应用**时，这些函数生成一组新的随机数。

注意：使用 Mersenne-Twister 方法生成随机数。此方法具有长度为 2^{19937}-1 的期间。有关生成程序的详细信息，请参见 Matsumoto 和 Nishimura (1998)。这些新的生成程序已被验证通过了 Marshalled (1996) 中所述的所有 DIEHARD 检验。

有关函数参数的详细信息，请参见《JSL Syntax Reference》。有关分布参数化的详细信息，请参见《JSL Syntax Reference》。

Random ChiSquare

从具有指定自由度的卡方分布生成一列随机数。可选的非中心参数必须大于或等于 0。非中心参数默认为 0。有关卡方分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random F

从具有指定分子自由度和分母自由度的 F 分布生成一列随机数。可选的非中心参数必须大于或等于 0。非中心参数默认为 0。有关 F 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Index

返回 k×1 的矩阵，矩阵中的元素是 1 到 n 之间不重复的随机整数。

Random Uniform

从均匀分布生成一列随机数。若未指定参数，则均匀分布的最小值为 0，最大值为 1。这意味着介于 0 到 1 之间的任意数与生成的任何其他数的可能性相同。结果近似均匀分布。可以使用可选参数为均匀分布的最小值和最大值指定不同的值。

Random Normal

从具有指定均值和标准差的正态分布生成一列随机数。若未指定参数，则正态分布的均值为 0，标准差为 1。有关正态分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Normal Mixture

返回具有指定参数的正态混合分布的随机数。有关 Random Normal Mixture 函数的语法的信息，请参见《JSL Syntax Reference》。

Random Exp

从 lambda=1 的指数分布生成一列随机数。您可以调整指数函数的尺度以使用不同的 lambda。例如，`Random Exp()*.1` 生成 lambda=0.1 的指数分布。有关指数分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Gamma

从具有指定形状和尺度参数的 gamma 分布生成一列随机数。尺度参数的默认值为 1。当尺度参数为 1 时，gamma 分布等同于指数分布。有关 gamma 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random GenGamma

从具有指定位置、尺度和形状参数的扩展广义 gamma 分布生成一列随机数。若未指定参数，则广义 gamma 分布的位置为 0，尺度为 1，形状为 0。有关广义 gamma 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random LogGenGamma

从具有指定位置、尺度和形状参数的对数广义 gamma 分布生成一列随机数。若未指定参数，则扩展广义 gamma 分布的位置为 0，尺度为 1，形状为 0。有关对数广义 gamma 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Beta

从具有两个指定形状参数及可选阈值和尺度参数的 beta 分布生成一列随机数。默认阈值为 0，默认尺度为 1。有关 beta 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Cauchy

从位置参数为 0 且尺度参数为 1 的 Cauchy 分布生成一列随机数。可以使用公式 `alpha+beta*Random Cauchy()` 生成位置参数为 alpha、尺度参数为 beta 的 Cauchy 变元。有关 Cauchy 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Category

基于一组概率和结果表达式生成一列随机类别分配（例如，`Random Category(.2, "A", .3, "B", .4, "C", "D");`）。生成一个随机均匀数并与概率进行比较，以确定返回哪个结果表达式。

Random Johnson Su

从具有两个指定形状参数、一个位置参数和一个尺度参数的 Johnson Su 分布生成一列随机数。有关 Johnson Su 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Johnson Sb

从具有两个指定形状参数、一个位置参数和一个尺度参数的 Johnson SB 分布生成一列随机数。有关 Johnson Sb 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Johnson Sl

从具有两个指定形状参数、一个位置参数和一个可选偏度参数的 Johnson SI 分布生成一列随机数。对于正偏，偏度参数可以是 +1；对于负偏，偏度参数可以是 -1。默认值为 +1。有关 Johnson Sl 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Seed State

从 BLOB 对象检索或恢复随机种子状态，或将随机种子状态检索或恢复为 BLOB 对象。

Random Triangular

从具有指定最小值、最大值和众数参数的三角形分布生成一列随机数。若仅指定一个参数，则该参数表示众数，并且三角形分布介于 0 和 1 之间。若指定两个参数，则它们分别表示众数和最大值，并且三角形分布介于 0 和最大值之间。若指定三个参数，则它们分别表示最小值、众数和最大值，并且三角形分布介于最小值和最大值之间。

Random Integer

从整数均匀分布生成一列随机数。若指定一个参数，则整数介于 1 和该参数之间。若指定两个参数，则整数介于这两个参数的值之间，并且包含这两个参数的值。

Random Binomial

从具有指定参数的二项分布生成一列随机数。第一个参数为 n，即二项实验中的试验次数。第二个参数为 p，即发生关注事件的概率。n 为 1 时，二项函数生成 Bernoulli 试验分布。有关二项分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Negative Binomial

从具有指定参数的负二项分布生成一列随机数。第一个参数是 r，即关注的成功次数。第二个参数是 p，即成功概率。关注的随机变量是第 r 次成功前的失败次数。二项变元的试验次数是固定的而成功次数是变量，负二项变元则针对固定的成功次数和随机的试验次数。有关负二项分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Beta Binomial

从具有指定参数的 beta 二项分布生成一列随机数。第一个参数是 n，即试验次数。第二个参数是 p，即每次试验的成功概率。第三个参数是可选的，表示过度离散参数。默认过度离散参数为 0。有关 beta 二项分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Frechet

从具有指定位置和尺度参数的 Fréchet 分布生成一列随机数。若未指定参数，则 Fréchet 分布的位置为 0 且 sigma 为 1。有关 Fréchet 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Geometric

从具有指定概率参数的几何分布生成一列随机数。该参数是特定事件在任意一次试验中出现的概率。关注的随机变量是第一次发生特定事件之前的试验次数。

Random Poisson

从具有指定形状参数的 Poisson 分布生成一列随机数。有关 Poisson 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Gamma Poisson

从具有指定形状和过度离散参数的 gamma Poisson 分布生成一列随机数。若未指定第二个参数，gamma Poisson 分布的过度离散值为 1。有关 gamma Poisson 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Weibull

从具有指定形状和尺度参数的 Weibull 分布生成一列随机数。若未指定第二个参数，则 Weibull 分布的尺度为 1。有关 Weibull 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Logistic

从具有指定位置和尺度参数的 logistic 分布生成一列随机数。若未指定参数，则 logistic 分布的位置为 0 且尺度为 1。有关 logistic 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Loglogistic

从具有指定位置和尺度参数的对数 Logistic 分布生成一列随机数。若未指定参数，则对数 Logistic 分布的位置为 0 且尺度为 1。有关 loglogistic 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Lognormal

从具有指定位置和尺度参数的对数正态分布生成一列随机数。若未指定参数，则对数正态分布的位置为 0 且尺度为 1。有关对数正态分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random GLog

从具有指定位置、尺度和形状参数的广义对数分布生成一列随机数。当形状参数为 0 时，广义对数分布等价于对数正态分布。有关广义对数分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Reset

使用指定的种子重新开始随机数序列。

Random LEV

从具有指定位置和尺度参数的最大极值 (LEV) 分布生成一列随机数。若未指定参数，则 LEV 分布的位置为 0 且尺度为 1。有关最大极值分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random SEV

从具有指定位置和尺度参数的最小极值 (SEV) 分布生成一列随机数。若未指定参数，则 SEV 分布的位置为 0 且尺度为 1。有关最小极值分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random SHASH

从具有两个指定形状参数、一个位置参数和一个尺度参数的 sinh-arcsinh (SHASH) 分布生成一列随机数。有关 SHASH 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Random Shuffle

从矩阵参数中的数字生成一列随机排列的数字。

Random t

从具有指定自由度的 t 分布生成一列随机数。可选的非中心参数可为负或正。非中心参数默认为 0。有关 t 分布的参数化的详细信息，请参见《JSL Syntax Reference》。

Col Shuffle

随机选择当前数据表的一个行号。每个行号只能选择一次。Col Shuffle 用作下标时，它返回从作为参数用的列中随机选择的值。原始列的每个值作为 Col Shuffle 结果只能被分配一次。有关 Col Shuffle 函数的语法的详细信息，请参见《JSL Syntax Reference》。

Resample Freq

生成有放回的频数统计的随机选择，适用于 Bootstrapping。例如，它支持第二个 Freq Column 参数，以便执行 Bootstrap 样本，这些样本与在第二个参数中指定的预先存在的频数列有关。`Resample Freq()` 生成 100% 再抽样。`ResampleFreq(rate)` 生成 `rate` 频数样本。`Resample(rate, column)` 生成由该比率乘以指定列之和所得到的样本。有关 Resample Freq 函数的语法的详细信息，请参见《JSL Syntax Reference》。

日期时间函数

JMP 使用**自 1904 年 1 月 1 日以来的秒数**的 macOS 标准储存数值列中的日期和时间。列具有日期值时，可以通过双击列名并从**格式**菜单中选择**日期**或**时间**，将日期格式分配给该列。请参见“设置 JMP 列属性”一章中第 257 页的“数值格式选项”。

有关语法的详细信息，请参见《JSL Syntax Reference》。

In Minutes、In Hours、In Days、In Weeks、In Years

从函数名称的单位转换为参数的等效秒数。参数必须为数字或数值表达式。例如，In Minutes(2) 得到 120，In Years(1) 得到 31,557,600（60 秒 * 60 分钟 * 24 小时 * 365.25 天）。

Date DMY、Date MDY

接受日、月和年的数值表达式并返回相关的 JMP 日期。例如，Date DMY (20, 3, 1991) 和 Date MDY(3, 20, 1991) 的求值结果为 2,752,272,000。

Today

返回 1904 年 1 月 1 日到当前日期之间的秒数。例如，在 1991 年 3 月 20 日（星期三）的午夜，Today 函数返回 2752272000（2,752,272,000 秒），并继续计数。若您当天稍后对 Today 函数求值，结果将反映增加的秒数。

Day, Month, Year

分别返回月份中的日、月份（1 到 12 之间的数）和四位数的年份。这些函数的参数解释为 JMP 日期。例如，在 1991 年 3 月 20 日：

- Day(2752272000) 返回数字 20。
- Month(2752272000) 返回数字 3。
- Year(2752272000) 返回数字 1991。

Quarter

将 datetime 值的每年季度返回为 1-4 的整数。

Hour、Minute、Second

分别返回日期-时间值的小时、分钟和秒。这些函数的参数解释为 JMP 日期。例如，在 1991 年 3 月 20 日：

- Hour(2752572649) 返回数字 11。
- Minute(2752572649) 返回数字 30。
- Second(2752572649) 返回数字 49。

Day of Week、Day of Year、Week of Year、Time of Day

这些函数的参数为 JMP 日期。Day Of Week 返回 1 到 7 之间的数字，其中 1 表示星期日。Day Of Year 返回从年初算起经历的天数。Week Of Year 基于指定的规则返回 1 到 52 之间的数字。规则 1（默认）的周从周日开始计算，每年的第一个周日为第 2 周的开始，第 1 周可能不完整或者为空；规则 2 的第一个周日为第 1 周的开始，前面几天为第 0 周；规则 3 返回 ISO 周数，其中周从周一开始计算，第 1 周为包含该年四天的第一周。使用 ISO 周数，一年的前三天或后四天有可能属于前一年或后一年的周数。Time Of Day 返回 0 到 86399 之间的数字（用秒表示的当天时间）。例如，在 1991 年 3 月 20 日（星期三）：

- Day Of Week(2752272000) 返回数字 4。
- Day Of Year(2752272000) 返回数字 79。
- Week Of Year(2752272000) 返回数字 12。
- Time Of Day(2752272000) 返回数字 0。

Informat

Informat 函数的参数是日期字符串。例如，Informat("03/20/1991") 返回相应的 JMP 日期值 2752272000。JMP 可以读取除 Abbrev Date 和 Long Date 之外的所有日期格式。

Abbrev Date、Long Date、Short Date

这些日期函数的参数是 JMP 日期。它们返回作为参数的格式化表示法的字符串。例如：

- Abbrev Date(2752272000) 返回 Wed, Mar 20,1991。
- Long Date(2752272000) 返回 Wednesday, March 20, 1991。
- Short Date(2752272000) 返回 3/20/91。

Format

Format 函数中的第一个参数是 JMP 日期。该函数用您在第二个参数中指定的日期格式返回日期的字符串表示法，它是一个带引号的字符串。若您将该格式应用到数值列，则 JMP 自动将列的数据类型更改为字符。

您还可以为第一个参数提供列并将其他参数留空。结果为列引用的格式化值。这可以用于在禁用值标签时提取列的值标签。

MDYHMS

MDYHMS 的参数是 JMP 日期。该函数显示所有日期和时间字段，若未提供时间信息，则追加零作为时间字段。当日期列采用公式而未显示全部日期信息时，这很有用。可以使用 MDYHMS 函数查看所有可用的日期和时间信息。

Date Increment

向起始日期时间值添加 1 个或更多的间隔。例如，Date Increment(Today(), "Day", 3) 向当前日期加 3 天。Date Increment(Today(), "Year", 3) 向当前日期加 3 年。

Date Difference

返回两个日期时间值的差值。间隔参数可以为 Second、Minute、Hour、Day、Week、Month、Quarter、Year。下面介绍了定位参数：

Start 用于对间隔开始的次数计数。

Actual 用于对完整间隔计数。

Fractional 用于对小数间隔计数。

例如，以下公式返回 207.890243055556，即两个日期之间间隔的天数：

```
Date Difference(
  01Jan2010:00:00:00,
  27Jul2010:21:21:57,
  "Day",
  "fractional"
);
```

以下公式返回 207，即两个日期之间间隔的整天数：

```
Date Difference(
  01Jan2010:00:00:00,
  27Jul2010:21:21:57,
  "Day",
  "actual"
);
```

以下公式返回 9，即两个时间之间间隔的整小时数：

```
Date Difference(
```

```
    01Jan2010:00:00:00,
    01Jan2010:09:22:57,
    "Hour",
    "actual"
);
```

以下公式返回 1，即两个时间之间新的 1 小时开始的次数：

```
Date Difference(
    31Dec2010:23:59:59,
    01Jan2011:00:59:59,
    "Hour",
    "start"
);
```

行状态函数

数据表中的行可以有六个特征：已选定、已隐藏、已排除、已添加标签、已着色和已添加标记。若您为行指定其中的一个或多个特征，然后创建行状态数据表列，可以接着创建计算并保存行状态条件的公式。（请参见“设置 JMP 列属性”一章中第 263 页的“JMP 中的列属性”和“设置 JMP 列属性”一章中第 261 页的“在行状态列中存储信息”。）该公式像处理字符和数值数据一样来处理行状态数据。

有关语法的详细信息，请参见《JSL Syntax Reference》。

注意：可以为行分配行状态的任意组合；行状态列可以具有多个行状态作为值。

表 A.2 介绍了每个行状态函数所需的参数类型以及返回的值类型。

表 A.2 行状态函数

函数名称	所需的参数类型	函数返回的值类型（您的列数据类型应为该类型）
Row State	无	当前行的行状态
As Row State	数值	当前行的所有行状态
Combine States	多个行状态参数	多个行状态赋值
Excluded State	正整数或零	行状态：已排除或未排除
Hidden State	正整数或零	行状态：已隐藏或未隐藏
Labeled State	正整数或零	行状态：已添加标签或未添加标签

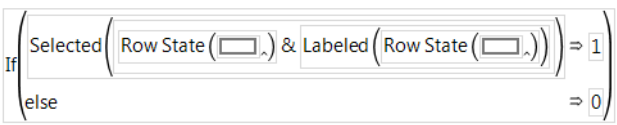
表 A.2 行状态函数 (续)

函数名称	所需的参数类型	函数返回的值类型（您的列数据类型应为该类型）
Color State	整数、颜色名称或 {red, green, blue}	行状态颜色
Marker State	整数或字符	行状态标记
Selected State	正整数或零	行状态：已选定或未选定
Hue State	整数	行状态色调
Shade State	整数 1-5	行状态强度
Excluded	Row State() 或行状态列	整数 0（未排除）或 1（已排除）
Hidden	Row State() 或行状态列	整数 0（未隐藏）或 1（已隐藏）
Labeled	Row State() 或行状态列	整数 0（未添加标签）或 1（已添加标签）
Color Of	Row State() 或行状态列	颜色映射整数
Marker Of	Row State() 或行状态列	标记映射整数
Selected	Row State() 或行状态列	整数 0（未选定）或 1（已选定）

Row State

将当前行的活动行状态条件返回为真或假。您可以使用该函数方便地编写依赖当前行的状态的条件子句。例如，图 A.15 将 1 赋给当前已选定且已添加标签的行，将 0 赋给其他行。

图 A.15 Row State



As Row State

将数值参数转换为行状态或行状态条件的集合。行状态在 JMP 内部作为 16 位数字储存，分配的每位数表示图 A.3 中所示的可能行状态之一。例如，1327 的二进制表示为 0000010100101111。As Row State(1327) 因此将行状态设置为“已选定”、“已排除”、“已隐藏”和“已添加标签”，并具有标记 2 和颜色 10。

表 A.3 作为 16 位数储存的行状态：每位数表示一个行状态

位	行状态
0	未选定 (0) 或已选定 (1)
1	未排除 (0) 或已排除 (1)
2	未隐藏 (0) 或已隐藏 (1)
3	未添加标签 (0) 或已添加标签 (1)
4-7	标记
8-14	颜色

Combine States

使用两个或多个参数生成行状态组合。使用“公式编辑器”小键盘上的插入按钮 (^) 或键入逗号可以将参数添加到 Combine States 函数。当您选择 Combine States 时，当前所选的表达式成为第一个参数。使用求值结果为行状态的表达式**替换**每个参数。该公式：

```
Combine States(
   Selected State( Modulo( Row(), 2 ) ),
   Labeled State( Modulo( Row() + 1, 2 ) ) );
```

交替地为计算的行状态列中的每行添加标签或选定它们。Selected State 和 Labeled State 函数的定义将在本节的后面介绍。使用“公式编辑器”小键盘上的插入 (^) 和 删除 (⌫) 按钮可以添加更多参数或删除不想要的参数。

若在组合中包括冲突的行状态，结果将无法预测。

Excluded State

将数值参数解释为真或假。参数求值结果为真时，Excluded State 函数分配已排除条件作为该行的列值。例如，Excluded State(Modulo(Row(),2)) 分配已排除行状态作为每个奇数编号行的行状态列值。

Hidden State

参数大于零时分配已隐藏行状态条件。若参数为零，则不隐藏该行对应列中的值。

Labeled State

参数大于零时指定已添加标签行状态条件。若参数为零，则该行对应列中的行值不添加标签。

Color State

返回 JMP 颜色映射中与整数参数对应的颜色。JMP 颜色用数字 0-84 表示。0 映射为黑色。

Marker State

返回 JMP 标记映射中与整数参数对应的标记。JMP 标记用数字 0-16 表示。公式 Marker State(Row()) 将当前行号确定的重复序列中的所有行状态标记分配给计算的行状态列。行状态列可以有多个行状态作为值。

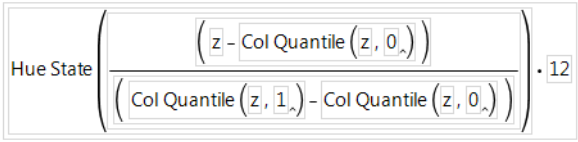
Selected State

参数大于零时指定已选定行状态条件。若参数为零，则该行对应列中的值未选定。

Hue State

返回 JMP 色调映射中与整数参数对应的颜色。JMP 色调用数字 0-11 表示，但是将较大的整数视为**模数** 12。Hue State 函数不映射为黑色、灰色或白色。值为 0 的色调映射为红色，值为 11 的色调映射为品红色。图 A.16 中公式基于 *z* 的值分配颜色扩散中的行状态颜色。Hue State 函数用于行状态数据类型列。

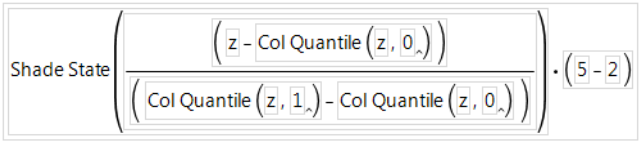
图 A.16 Hue State 函数的示例



Shade State

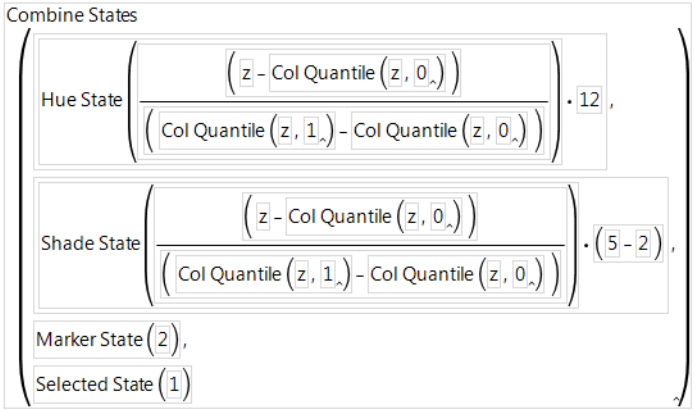
将 5 个阴影级别分配给颜色或色调。值为 –2 的阴影最深，值为 +2 的阴影最浅。值为 0 的阴影为纯色。图 A.17 中的公式基于 *z* 的值分配阴影值。

图 A.17 Shade State 函数的示例



要分配调色板中的所有颜色的所有阴影，您需要一起使用 Hue State 和 Shade State 指定分配。图 A.18 中的公式使用本节开头所述的 Combine States 函数。Combine States 函数中的第一个参数为以前所示的 Hue State 公式，第二个参数为 Shade State 公式。此外，参数为 2 的 Marker State 函数将 X 标记分配给每行，参数为 1 的 Selected State 函数选择每行。

图 A.18 同时使用 Hue State 和 Row State 的 Combine States 示例



Excluded、Hidden、Labeled 和 Selected

接受求值结果为 1 或 0（真或假）的行状态表达式参数（行状态列或行状态常数）。这些特征默认为不活动。通常，Row() 函数为参数，它检测每行的活动行状态条件。例如，在图 A.19 中，只要有一行当下被选定，该公式分配 99，否则分配 0。请注意该公式在具有数值数据类型的列中使用。

图 A.19 使用 Selected 函数的公式示例

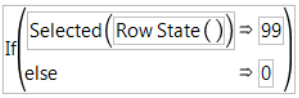


图 A.20 中的示例将行状态条件分配给行状态列。行状态列的公式（在名为 x 的列中）检查以确定活动行状态是否为 Hidden 或 Excluded，若是，则分配 Labeled 行状态。

图 A.20 在行状态列中计算行状态信息

	x
1	
2	
3	
4	
5	

Color Of

接受任意行状态表达式、列或 Row State() 函数作为参数。返回 JMP 颜色映射中与活动颜色状态对应的数字，若未分配颜色，则返回 0。

Marker Of

接受任意行状态表达式、列或 Row State() 函数作为参数。返回 JMP 标记映射中与活动标记对应的数字，若未分配标记，则返回 0。

赋值函数

赋值函数在原位工作。即运算返回的结果（在运算符右侧）储存在运算符左侧的参数中并替换其当前值。

赋值语句通常与其他命令一起使用来生成 JSL 脚本。您可以使用“公式编辑器”在该列中创建和执行脚本，但不推荐这样做，因为这会导致相关性问题并造成混淆。脚本通常存储为 .jsl 文件，并可以与数据表一起保存。请参见“输入和编辑数据”一章中第 247 页的“在数据表中创建和保存脚本”。有关语法的详细信息，请参见《JSL Syntax Reference》。

注意：赋值函数的第一个参数必须可以被赋值。这意味着您不能具有像 3+=4 这样的赋值，因为 3 是无法重新赋值的常数值。您必须首先创建一个值为 3 的变量（表变量或局部变量）。（有关表变量的详细信息，请参见“输入和编辑数据”一章中第 246 页的“使用数据表变量”。有关局部变量的详细信息，请参见“在 JMP 中创建公式”一章中第 344 页的“在公式中引用数据表值”）。然后使用该变量作为赋值函数左侧的参数。

= (assign) 将 *b* 的值放入 *a*。例如 (*a*=*b*)。

+= (add to) 将 *b* 的值加到 *a* 并将结果放入 *a*。例如， *a*+=*b*。

-= (subtract to) 减去 *b* 的值并将结果放入 *a*。例如， *a*–=*b*。

***= (multiply to)** 将 *b* 与 *a* 相乘并将结果放入 *a*。例如， *a**=*b*。

/= (divide to) 用 *b* 去除 *a* 并将结果放入 *a*。例如， *a*/=*b*。

++ (post increment) 在原位将 1 加到 *a* 即 *a*++。例如，若 *a* 的初始值为 4，则表达式 *a*++ 将 a 更改为 5。

-- (post decrement) 在原位从 *a* 减去 1 即 *a*– –。例如，若 *a* 的初始值为 4，则表达式 *a*– – 将 a 更改为 3。

参数模型函数

该类别是创建三个参数模型的快捷方式，这些模型是已选定窗口的列集的线性函数。

Linear Model、Interactions Model、Full Quadratic Model

选择这些函数之一将打开列选择框，您可以在此选择模型中要包括的一列或多列。该函数然后创建并填充所选的模型。

财务函数

支持您创建公式，用于计算本金偿还、利率、收益率等。

Double Declining Balance

返回资产在指定时间段内的折旧值。该函数使用双倍余额递减法或其他折旧系数。

Future Value

返回基于定期的固定付款和固定利率的投资的未来值。

Interest Payment

基于定期的固定付款和固定利率的投资，返回给定期间的利息付款。

Interest Rate

返回年金的每个期间的利率。

Internal Rate of Return

返回 values 参数中一系列现金流的内部回报率。

Modified Internal Rate of Return

返回一系列定期现金流的修改后的内部回报率。包括投资成本和再投资现金获得的利息。

Net Present Value

使用贴现率和一系列未来付款（负值）和收入（正值）返回投资的净现值。

Number of Periods

返回基于定期的固定付款和固定利率的投资的期间数。

Payment

返回基于固定付款和固定利率的还贷额。

Present Value

返回投资的现值。

Principal Payment

基于定期的固定付款和固定利率的投资，返回给定期间内偿还的本金。

Straight Line Depreciation

返回资产在一个期间内的直线折旧值。

Sum Of Years Digits Depreciation

返回资产在指定期间内的年限总额折旧值。

附录 B

JMP 技术详细信息

本章包含适用于 JMP 中的多个平台和 SAS 集成的技术详细信息。

目录

支持多线程的 JMP 功能 681
将 JMP 特性映射为 SAS 扩展属性的约定 682

支持多线程的 JMP 功能

将 JMP 中的一些功能编码为利用计算机上的多个中央处理单元 (CPU)，从而使这些功能大大提高运行速度。该过程称为多线程。

以下功能支持多线程：

- JMP PRO 提升树
- JMP PRO Bootstrap 森林法
- 选择
- JMP PRO 覆盖阵列
- 分布中的能力分析
- 探索离群值
- 探索模式
- 因子分析
- 以 X 拟合寿命
- 拟合模型：参数生存、JMP PRO 混合模型、JMP PRO 广义回归和 JMP PRO 响应筛选
- 函数数据分析器
- 高斯（JMP PRO 仅限在 JMP Pro 中使用快速 GASP 和分类变量。）
- JMP PRO K 最近邻
- 潜在类分析
- 寿命分布
- 多元
- 神经（JMP PRO 一些功能仅在 JMP Pro 中提供。）
- 名义型 Logistic
- 非线性和非线性曲线
- 聚类分析中的正态混合
- 偏最小二乘（JMP PRO 一些功能仅在 JMP Pro 中提供。）
- 分割
- 预测变量筛选
- 主成分
- 过程筛选
- 刻画器（不适用于从其他平台中启动的刻画器。）
- 可靠性预测
- JMP PRO 可修复系统模拟

- 响应筛选
- JMP PRO 稀疏主成分
- JMP PRO 结构化方程模型
- JMP PRO 支持向量机
- 缺失值筛选中的 SVD 补缺
- 文本分析器（JMP PRO SVD 和“潜在类分析”仅适用于 JMP Pro。）
- 时间序列预测
- JMP PRO 提升
- 变量聚类

将 JMP 特性映射为 SAS 扩展属性的约定

SAS 扩展属性是您在 SAS 代码中定义的元数据，用于导入表脚本、标签、长度和类型之类的信息。您将扩展属性与某个数据集或变量关联，并以名称-值对的形式定义它们，如 `_JMP_TABLESCRIPTNAME_2="OnOpen"`。

本节提供有关 SAS 扩展属性和对应的 JMP 特性的信息。

表 B.1 表特性

JMP 表特性	SAS 特性名称	SAS 特性类型和默认值	说明
表变量计数	_JMP_TABLEVARCOUNT	数值，0	表变量的计数
表变量名称 1	_JMP_TABLEVARNAME_1	字符串	第一个表变量的名称
表变量值 1	_JMP_TABLEVARVALUE_1	字符串	第一个表变量的值
表变量名称 2	_JMP_TABLEVARNAME_2	字符串	第二个表变量的名称
表变量值 2	_JMP_TABLEVARVALUE_2	字符串	第二个表变量的值
表变量名称 n	_JMP_TABLEVARNAME_*n*	字符串	第 *n* 个表变量的名称
表变量值 n	_JMP_TABLEVARVALUE_*n*	字符串	第 *n* 个表变量的值
表脚本计数	_JMP_TABLESCRIPTCOUNT	数值	表脚本的计数
表脚本名称 1	_JMP_TABLESCRIPTNAME_1	字符串	第一个表脚本的名称
表脚本值 1	_JMP_TABLESCRIPTVALUE_1	字符串	第一个表脚本的值
表脚本名称 2	_JMP_TABLESCRIPTNAME_2	字符串	第二个表脚本的名称

表 B.1 表特性 (续)

JMP 表特性	SAS 特性名称	SAS 特性类型和默认值	说明
表脚本值 2	_JMP_TABLESCRIPTVALUE_2	字符串	第二个表脚本的值
表脚本名称 n	_JMP_TABLESCRIPTNAME_*n*	字符串	表脚本 *n* 的名称
表脚本值 n	_JMP_TABLESCRIPTVALUE_*n*	字符串	表脚本 *n* 的值
锁定数据表	_JMP_ISLOCKED	布尔值，0	已锁定表
禁止公式计算	_JMP_SUPPRESSEVAL	布尔值，0	已禁止公式计算
列组计数	_JMP_COLGRPCOUNT	数值，0	列组的数目
列组名称 1	_JMP_COLGRPNAME_1	字符串	列组 1 的名称
列组起始列 1	_JMP_COLGRPSTARTCOL_1	字符串	组 1 中第一个列的名称
列组列数 1	_JMP_COLGRPNCOLS_1	数值	组 1 中的列数
列组名称 2	_JMP_COLGRPNAME_2	字符串	列组 2 的名称
列组起始列 2	_JMP_COLGRPSTARTCOL_2	字符串	组 2 中第一个列的名称
列组列数 2	_JMP_COLGRPNCOLS_2	数值	组 2 中的列数
列组名称 n	_JMP_COLGRPNAME_*n*	字符串	列组 *n* 的名称
列组起始列 n	_JMP_COLGRPSTARTCOL_*n*	字符串	组 *n* 中第一个列的名称
列组列数 n	_JMP_COLGRPNCOLS_*n*	数值	组 *n* 中的列数

表 B.2 列特性

JMP 列特性	SAS 特性名称	SAS 特性类型和默认值	说明
列名	_JMP_COLNAME	字符串	JMP 中的原始列名
数据长度	_JMP_DATALENGTH	数值，0	数据的长度。对于数值类型，该值可以是 1、2、4 或 8。对于字符类型，0（或负数）表示可变长度，正值表示固定长度。
行状态列	_JMP_ISROWSTATE	数值，0	0 = 非行状态，1 = 表的默认行状态，2 = 行状态列

表 B.2 列特性 (续)

JMP 列特性	SAS 特性名称	SAS 特性类型和默认值	说明
标签	_JMP_ISLABEL	布尔值，0	标签列1 表示“是”。0 表示“否”。
滚动锁定	_JMP_ISSCROLLLOCK	布尔值，0	已锁定滚动的列。1 表示“是”。0 表示“否”。
隐藏	_JMP_ISHIDDEN	布尔值，0	隐藏的列。1 表示“是”。0 表示“否”
已排除	_JMP_ISEXCLUDED	布尔值，0	已排除的列。1 表示“是”。0 表示“否”
注释	_SAS_NOTES	字符串	有关列的注释
建模类型	_SAS_LEVEL	字符串	有效值为“二元”、“区间”、“有序型”、“名义型”、“一元”。JMP 将“二元”和“一元”映射为“名义型”，将“区间”映射为“连续”。
预选择角色	_SAS_MININGROLE	字符串	有效值为“评估”、“删失”、“分类”、“成本”、“交叉 ID”、“决策”、“频数”、“ID”、“输入”、“关键字”、“标签”、“预测”、“来源”、“拒绝”、“残差”、“分段”、“序列”、“目标”、“文本”、“文本位置”、“时间 ID”、“处理”、“网址”、“权重”。JMP 将“Y”映射为“目标”、将“X”映射为“输入”、将“频数”映射为“频数”、将“权重”映射为“权重”。
分布	_SAS_DISTRIBUTION	字符串	要为该列拟合的分布类型
格式名称	_JMP_FORMATNAME	字符串	列的格式（如“固定小数位数”或“百分比”）
格式宽度	_JMP_FORMATWIDTH	数值	列的宽度
格式小数位数	_JMP_FORMATDECIMALS	数值	小数位数

表 B.2 列特性 (续)

JMP 列特性	SAS 特性名称	SAS 特性类型和默认值	说明
格式代码	_JMP_FORMATCODE	字符串	通常为货币格式的货币代码
使用千位分隔符	_JMP_USETHOUSANDSSEP	布尔值，0	包括千分隔符。1 表示“是”。0 表示“否”。
输入格式	_JMP_INFORMAT	字符串	输入格式（用于日期、时间或持续时间格式的列）
公式	_JMP_FORMULA	字符串	列格式
禁止公式计算	_JMP_SUPPRESSEVAL	布尔值，0	已禁止公式计算。1 表示“是”。0 表示“否”。
锁定	_JMP_ISLOCKED	布尔值，0	已锁定列。1 表示“是”。0 表示“否”。
范围检查代码	_JMP_RANGECHECK_CODE	字符串	标识范围检查类型的代码。示例为 LE、LT、LTLT、!LT 和 !LELE。
范围检查低	_JMP_RANGECHECK_LOW	数值	要进行范围检查的第一个参数
范围检查高	_JMP_RANGECHECK_HIGH	数值	要进行范围检查的第二个可选参数
列表检查	_JMP_LISTCHECK	字符串	列表检查
缺失值代码	_JMP_MISSINGCODES	字符串	要视为缺失值的值
值标签计数	_JMP_VALUELABEL_COUNT	数值	值标签的计数
值标签代码	_JMP_VALUELABELCODE_1	字符串	值标签的代码（如 EQ、LT、LE、LTLT 和 LELT）。EQ 始终用于字符变量。
值标签下限	_JMP_VALUELABELLOW_1	变化的	数值范围的下限，或只有一个边界（若范围在一侧无限制）。将字符串用于字符列。将数值用于数值列。
值标签上限	_JMP_VALUELABELHIGH_1	数值	数值范围的上限。不用于字符值标签。
值标签显示	_JMP_VALUELABELDISPLAY_1	字符串	该值标签的显示字符串
使用值标签	_JMP_USEVALUELABELS	布尔值，0	显示该列时使用值标签。1 表示“是”。0 表示“否”。

表 B.2 列特性 (续)

JMP 列特性	SAS 特性名称	SAS 特性类型和默认值	说明
值得分	_JMP_VALUESCORES	字符串	将数据值与数值得分关联
值顺序	_JMP_VALUEORDER	字符串	列值的顺序
值颜色	_JMP_VALUECOLORS	字符串	将值映射为颜色。
颜色梯度	_JMP_COLORGRADIENT	字符串	将值映射为颜色梯度。
为单元格着色	_JMP_COLORCELLS	布尔值，0	使用指定的值颜色或颜色梯度为该列的单元格着色。1 表示“是”。0 表示“否”。
轴	_JMP_AXIS	字符串	轴定义
编码	_JMP_CODING	字符串	为了建模，将值范围编码为 -1, 1
混料	_JMP_MIXTURE	字符串	参与列的混料，总计为一个值。“混料”列属性具有数值（限值）和字符串值（L 伪成分和 U 伪成分）。
行顺序水平	_JMP_ROWORDER	布尔值，0	按出现次数（而非值）对列排序。1 表示“是”。0 表示“否”。
规格限	_JMP_SPECLIMITS	字符串	目标值、上规格限和下规格限
控制限	_JMP_CONTROLLIMITS	字符串	控制图类型的控制限
响应限	_JMP_RESPONSELIMITS	字符串	响应的可接受范围的边界，由刻画器使用
设计角色	_JMP_DESIGNROLE	字符串	设计的实验中的角色
因子更改	_JMP_FACTORCHANGES	字符串	更改实验设计的因子的难度。有效值为：“容易”、“困难”、“极难”
Sigma	_JMP_SIGMA	字符串	控制图的 Sigma
单位	_JMP_UNITS	字符串	该变量的单位，可以为任意字符串

表 B.2 列特性 (续)

JMP 列特性	SAS 特性名称	SAS 特性类型和默认值	说明
时间频率	_JMP_TIMEFREQ	字符串	“数值”、“每年”、“每季度”、“每月”、“每周”、“每天”、“每小时”、“每分钟”、“每秒”
地图角色	_JMP_MAPROLE	字符串	地图数据集中的角色
多重响应分隔符	_JMP_MRSEP	字符串，","	多重响应列的分隔符。
标签列	_JMP_LABELCOLUMN	布尔值，0	该列是否为标签列？1 表示“是”。0 表示“否”。
表达式列	_JMP_ISEXPRESSIONCOLUMN	布尔值，0	该列是否为表达式列？1 表示“是”。0 表示“否”。

表 B.3 定制列属性

JMP 列特性	SAS 特性名称	SAS 特性类型和默认值	说明
定制属性计数	_JMP_CUSTCOLPROPCOUNT	数值，0	定制列属性的计数
定制属性名称 1	_JMP_CUSTCOLPROPNAME_1	字符串	定制列属性 1 的名称
定制属性值 1	_JMP_CUSTCOLPROPVALUE_1	字符串	定制列属性 1 的值
定制属性名称 2	_JMP_CUSTCOLPROPNAME_2	字符串	定制列属性 2 的名称
定制属性值 2	_JMP_CUSTCOLPROPVALUE_2	字符串	定制列属性 2 的值
定制属性名称 n	_JMP_CUSTCOLPROPNAME_*n*	字符串	定制列属性 *n* 的名称
定制属性值 n	_JMP_CUSTCOLPROPVALUE_*n*	字符串	定制列属性 *n* 的值

附录 C

技术许可声明

- Scintilla - Copyright © 1998-2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library - Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages - Copyright © 2009-2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software - Copyright © 1995-2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 996-2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999-2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998 - 2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball - Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako - Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 2006 –2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright (C) 1999-2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS@AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright(C) January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

版本 16

基本分析

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020–2021. 基本分析® 16. Cary, NC: SAS Institute Inc.

基本分析 ® 16

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

March 2021

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

基本分析

1 了解 JMP 17

文档和其他资源

JMP 文档中的格式规范 19

JMP 帮助 20

JMP 文档库 20

学习 JMP 的其他资源 26

JMP 教程 26

样本数据表 26

学习统计和 JSL 术语 27

学习 JMP 的提示与技巧 27

JMP 工具提示 27

JMP 用户社区 27

免费在线统计思维课程 27

JMP 新用户欢迎套件 28

统计学知识门户 28

JMP 培训 28

用户编写的 JMP 手册 28

“JMP 起始页”窗口 28

JMP 技术支持 28

2 基本分析介绍 29

基本分析方法概述

3 分布 31

使用 “分布”平台

“分布”平台概述 34

“分布”平台的示例 34

启动“分布”平台 36

“分布”报表 37
直方图 38
“频数”报表 41
“分位数”报表 41
“汇总统计量”报表 41
“分布”平台选项 43
用于分类变量的选项 44
分类变量的显示选项 44
分类变量的直方图选项 44
分类变量的保存选项 45
用于连续变量的选项 45
连续变量的显示选项 46
连续变量的直方图选项 46
正态分位数图 47
离群值箱线图 48
分位数箱线图 49
茎叶图 50
CDF 图 50
检验均值 51
检验标准差 52
检验等价性 52
置信区间 53
预测区间 54
容差区间 54
过程能力 54
拟合分布 56
连续变量的保存选项 59
“分布”平台的更多示例 61
在多个直方图中选择数据的示例 61
使用“依据”变量的示例 62
“检验概率”选项的示例 63
预测区间的示例 65
容差区间的示例 66
过程能力示例 67

“分布”平台的统计详细信息 69
标准误差条 69
分位数的统计详细信息 69
汇总统计量 70
正态分位数图 71
Wilcoxon 符号秩检验 71
标准差检验 73
正态分位数 73
保存标准化数据 74
预测区间 74
容差区间 74
连续拟合分布 76
离散拟合分布 82
旧版分布拟合器详细信息 85
“拟合分布”选项（旧版） 85
连续拟合分布的统计详细信息（旧版） 89
离散拟合分布的统计详细信息（旧版） 93
拟合分位数的统计详细信息（旧版） 95
“拟合分布”选项的统计详细信息（旧版） 95

4 “以 X 拟合 Y”介绍 97
检查两个变量之间的关系

“以 X 拟合 Y”平台概述 99
启动“以 X 拟合 Y”平台 99
从“JMP 起始页”窗口启动特定分析 100

5 二元分析 101
检查两个连续变量之间的关系

二元分析的示例 104
启动“二元”平台 104
二元图 105
拟合选项 106
拟合选项 107
拟合选项类别 108
多次拟合同一选项 108

直方图边框 109
拟合均值 110
“拟合均值”报表 110
拟合线和拟合多项式 111
“线性拟合”报表和“多项式拟合”报表 111
特殊拟合 116
“特殊拟合”报表和菜单 117
灵活 117
拟合样条 118
核平滑法 118
拟合每个值 119
拟合正交 120
正交拟合比报表 120
稳健 121
稳健拟合 121
拟合 Cauchy 121
密度椭圆 122
“相关性”报表 122
非参数密度 123
“分位数密度等高线”报表 123
依据...分组 124
拟合菜单 124
拟合菜单选项 125
诊断图 127
“二元”平台的更多示例 128
“特殊拟合”选项的示例 128
“拟合正交”选项的示例 130
“稳健拟合”命令的示例 132
使用密度椭圆的“依据 ... 分组”的示例 134
使用回归线的“依据 ... 分组”的示例 135
使用“依据”变量进行分组的示例 136
“二元”平台的统计详细信息 138
拟合线 138

拟合样条 138
拟合正交 138
“拟合汇总”报表 139
“失拟”报表 140
“参数估计值”报表 140
“平滑拟合”报表 141
“相关性”报表 141

6 单因子分析 143
检查连续 Y 和分类 X 变量之间的关系

单因子分析概述 146
单因子分析示例 146
启动“单因子”平台 148
数据格式 148
单因子图 149
“单因子”平台选项 149
显示选项 152
分位数 153
离群值箱线图 154
“均值/方差分析”和“均值/方差分析/合并的 t” 154
“拟合汇总”报表 155
t 检验报表 155
“方差分析”报表 156
“单因子方差分析均值”报表 156
“区组均值”报表 157
“均值菱形”和“X 轴成比例” 157
均值线、误差条和标准差线 158
均值分析方法 158
位置均值分析 159
尺度均值分析 159
均值分析图表 160
“均值分析”选项 161
比较均值 161
使用比较环 162

每对，Student t 164
所有对，Tukey HSD 164
带最佳组的 Hsu MCB 检验 164
带控制组的 Dunnett 检验 165
每对逐步，Newman-Keuls 166
“比较均值”选项 167
非参数检验 167
Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验报表 168
“Kolmogorov-Smirnov 双样本检验”报表 169
非参数多重比较 170
不等方差 173
“等方差检验”报表 173
等价性检验 175
“等价性检验”报表 175
稳健 176
稳健拟合 176
Cauchy 拟合 177
功效 177
“功效详细信息”窗口和报表 177
正态分位数图 178
CDF 图 178
密度 178
匹配列 179
“单因子”平台的更多示例 179
均值分析图表示例 180
方差均值分析图表的示例 181
“每对，Student t”检验的示例 181
“所有对，Tukey HSD”检验的示例 183
“带最佳组的 Hsu MCB 检验”的示例 185
“带控制组的 Dunnett 检验”的示例 186
“每对逐步，Newman-Keuls”检验的示例 187
示例：比较四个“比较均值”检验 188
非参数 Wilcoxon 检验的示例 189
“不等方差”选项的示例 191

“等价性检验”的示例 192
“稳健拟合”选项的示例 194
“功效”选项的示例 195
正态分位数图的示例 197
CDF 图的示例 198
“密度”选项的示例 199
“匹配列”选项的示例 200
示例：堆叠数据以进行单因子分析 201
“单因子”平台的统计详细信息 208
比较环 208
功效 209
“拟合汇总”报表 210
等方差检验 210
非参数检验统计量 211

7 列联分析 215
检查两个分类变量之间的关系
列联分析示例 218
启动“列联”平台 219
数据格式 219
“列联”报表 220
“列联”平台选项 221
马赛克图 222
弹出菜单 223
列联表 224
列联表的说明 225
检验 226
“检验”报表的说明 226
Fisher 精确检验 226
比例均值分析 227
对应分析 227
了解对应分析图 227
“对应分析”选项 227
“详细信息”报表 228

Cochran-Mantel-Haenszel 检验 228
一致性统计量 229
相对风险 229
双样本比例检验 229
关联测量 230
Cochran Armitage 趋势检验 231
精确检验 231
“列联”平台的更多示例 232
“比例均值分析”的示例 232
“对应分析”的示例 233
“Cochran Mantel Haenszel 检验”的示例 235
“一致性统计量”选项的示例 236
“相对风险”选项的示例 237
“双样本比例检验”的示例 239
“关联测量”选项的示例 240
“Cochran Armitage 趋势检验”的示例 240
“列联”平台的统计详细信息 241
“一致性统计量”选项 241
“优势比”选项 242
检验报表 242
对应分析中的“详细信息”报表 243

8 Logistic 分析 245
检查分类 Y 和连续 X 变量之间的关系

Logistic 回归概述 247
名义型 Logistic 回归 247
有序型 Logistic 回归 247
名义型 Logistic 回归的示例 247
启动“Logistic”平台 249
数据结构 249
Logistic 报表 250
Logistic 图 251
迭代 251
整体模型检验 251

拟合详细信息 252
参数估计值 253
“Logistic”平台选项 253
ROC 曲线 254
保存概率公式 255
逆预测 255
Logistic 回归的其他示例 255
有序型 Logistic 回归的示例 256
Logistic 图的更多示例 257
ROC 曲线的示例 259
使用十字准线工具进行逆预测的示例 260
使用“逆预测”选项的逆预测示例 261
“Logistic”平台的统计详细信息 263

9 制表 265
以交互方式创建汇总表
“制表”平台的示例 267
启动“制表”平台 271
使用对话框 272
添加统计量 273
制表输出 276
分析列 277
分组列 277
列表和行表 277
编辑表 278
“制表”平台选项 279
显示检验生成结果面板 280
列的右击菜单 280
“制表”平台的更多示例 281
创建不同表和重新排列内容的示例 281
将列组合到单个表的示例 284
使用页列的示例 286

10 模拟 289
回答关于参数再抽样的难题
“模拟”平台概述 291
关于使用“模拟”的示例 291
构造方差分量的半参数置信区间 292
执行置换检验 296
探索在广义回归中保留一个因子 299
执行非线性模型的前瞻功效分析 303
启动“模拟”窗口 311
“模拟”窗口 311
“模拟结果”表 312
“模拟结果”报表 312
“模拟功效”报表 312

11 Bootstrapping 315
通过再抽样近似统计量的分布
Bootstrapping 概述 317
支持 Bootstrapping 的平台 317
Bootstrapping 示例 318
“Bootstrapping”窗口选项 320
“堆叠的 Bootstrap 结果”表 322
“未堆叠的 Bootstrap 结果”表 323
Bootstrap 结果的分析 323
更多 Bootstrapping 示例 324
Bootstrapping 的统计详细信息 329
计算分式权重 329
修正偏倚的百分位数区间 329

12 文本分析器 331
探索数据中的非结构化文本
“文本分析器”平台概述 333
文本处理步骤 334
“文本分析器”平台的示例 335
启动“文本分析器”平台 338

在“正则表达式编辑器”中定制 Regex 340
文本分析器报表 345
“汇总计数”报表 345
词条和短语列表 346
“文本分析器”平台选项 349
文本准备选项 349
文本分析选项 354
保存选项 355
报表选项 357
潜在类分析 357
潜在语义分析 (SVD) 358
SVD 报表 359
SVD 报表选项 359
主题分析 361
“主题分析”报表 361
“主题分析”报表选项 362
判别分析 363
“判别分析”报表 363
“判别分析”报表选项 364
词条选择 364
词条选择设置 364
“词条选择”报表 365
“词条选择”报表选项 367
情感分析 367
“情感分析”报表 368
“情感分析”报表选项 370
“文本分析器”平台的其他示例 371

A 参考书目 375

B 技术许可声明 379

第 1 章

了解 JMP

文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 19
JMP 帮助 20
JMP 文档库 20
学习 JMP 的其他资源 26
JMP 教程 26
样本数据表 26
学习统计和 JSL 术语 27
学习 JMP 的提示与技巧 27
JMP 工具提示 27
JMP 用户社区 27
免费在线统计思维课程 27
JMP 新用户欢迎套件 28
统计学知识门户 28
JMP 培训 28
用户编写的 JMP 手册 28
“JMP 起始页”窗口 28
JMP 技术支持 28

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 20 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

基本分析介绍

基本分析方法概述

《基本分析》介绍您在 JMP 中经常执行的初始分析类型：

- “分布”平台使用直方图、其他图形和报表来描述单个变量的分布。知道您的数据是如何分布的之后，您可以选择合适的分析类型以继续操作。请参见第 3 章，“分布”。
- “以 X 拟合 Y”平台根据建模类型在上下文中分析您指定的一对 X 和 Y 变量。请参见第 4 章，“‘以 X 拟合 Y’介绍”。四种分析类型是指：
  - “二元”平台，它分析两个连续 X 变量之间的关系。请参见第 5 章，“二元分析”。
  - “单因子”平台，它分析在分类 X 变量定义的不同分组间连续 Y 变量的分布状况。请参见第 6 章，“单因子分析”。
  - “列联”平台，它分析在分类 X 因子值的条件下分类响应变量 Y 的分布。请参见第 7 章，“列联分析”。
  - “Logistic”平台，它对连续 X 预测变量拟合响应类别 (Y) 的概率。请参见第 8 章，“Logistic 分析”。
- “制表”平台以交互方式构造描述性统计量的表。请参见第 9 章，“制表”。
- “模拟”功能提供参数和非参数模拟能力。请参见第 10 章，“模拟”。
- Bootstrap 分析近似统计量的抽样分布。对数据进行放回再抽样并计算统计量。重复该过程以生成统计量值的分布。请参见第 11 章，“Bootstrapping”。
- “文本分析器”平台支持您对未格式化的文本数据进行分类和分析。继续分析之前，您可以使用正则表达式清理数据。请参见第 12 章，“文本分析器”。

第 3 章

分布

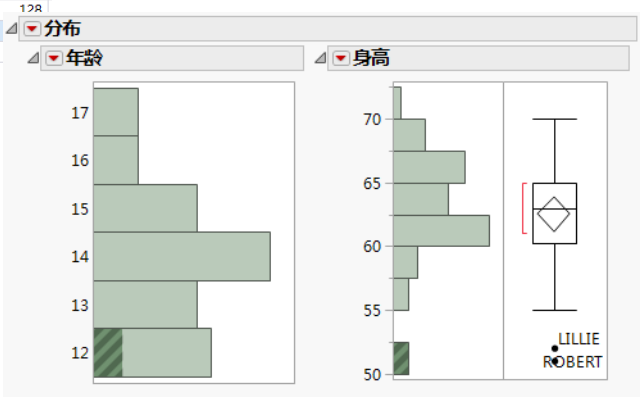
使用“分布”平台

使用“分布”平台可通过直方图、箱线图和汇总统计量来探索单个变量的分布。您可以执行若干类型的假设检验，包括 *t* 检验、*z* 检验、卡方检验和等价性检验。您还可以构造置信度、容差和预测区间，并估计过程能力。**一元**一词就是意味着仅涉及一个变量而不是两个（二元）或许多（多元）变量。但是，您可以在报表内查看若干单个变量的分布。每个变量的报表内容取决于该变量是分类变量（名义型或有序型）还是连续变量。

“分布”报表窗口是交互式窗口。点击某个直方图直条可突出显示其他任何直方图和数据表中相应的数据。

图 3.1 “分布”平台的示例

	姓名	年龄	性别	身高	体重
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	
9	BARBARA	13	F	60	



目录

“分布”平台概述 34
“分布”平台的示例 34
启动“分布”平台 36
“分布”报表 37
直方图 38
“频数”报表 41
“分位数”报表 41
“汇总统计量”报表 41
“分布”平台选项 43
用于分类变量的选项 44
分类变量的显示选项 44
分类变量的直方图选项 44
分类变量的保存选项 45
用于连续变量的选项 45
连续变量的显示选项 46
连续变量的直方图选项 46
正态分位数图 47
离群值箱线图 48
分位数箱线图 49
茎叶图 50
CDF 图 50
检验均值 51
检验标准差 52
检验等价性 52
置信区间 53
预测区间 54
容差区间 54
过程能力 54
拟合分布 56
连续变量的保存选项 59
“分布”平台的更多示例 61
在多个直方图中选择数据的示例 61
使用“依据”变量的示例 62
“检验概率”选项的示例 63
预测区间的示例 65
容差区间的示例 66
过程能力示例 67

“分布”平台的统计详细信息 69
标准误差条 69
分位数的统计详细信息 69
汇总统计量 70
正态分位数图 71
Wilcoxon 符号秩检验 71
标准差检验 73
正态分位数 73
保存标准化数据 74
预测区间 74
容差区间 74
连续拟合分布 76
离散拟合分布 82
旧版分布拟合器详细信息 85
“拟合分布”选项（旧版） 85
连续拟合分布的统计详细信息（旧版） 89
离散拟合分布的统计详细信息（旧版） 93
拟合分位数的统计详细信息（旧版） 95
“拟合分布”选项的统计详细信息（旧版） 95

“分布”平台概述

根据变量的建模类型：名义型、有序型或连续型（名义型和有序型归为分类变量），“分布”平台中对变量的处理是不同的。

分类变量

对于分类变量，显示的初始图形是一个直方图。直方图为名义型或有序型变量的每个水平显示一个直条。您还可以添加分段式的（马赛克）条形图。

“频数”报表显示计数和比例。您可以从红色小三角菜单中的选项添加置信区间并检验概率。

连续变量

对于数值型连续变量，显示的初始图形是一个直方图和一个离群值箱线图。直方图为分到一组的连续变量值显示一个直条。还提供以下选项：

- 正态分位数图
- 分位数箱线图
- 茎叶图
- CDF 图

报表显示选定的分位数和汇总统计量。红色小三角菜单提供针对以下的附加报表选项：

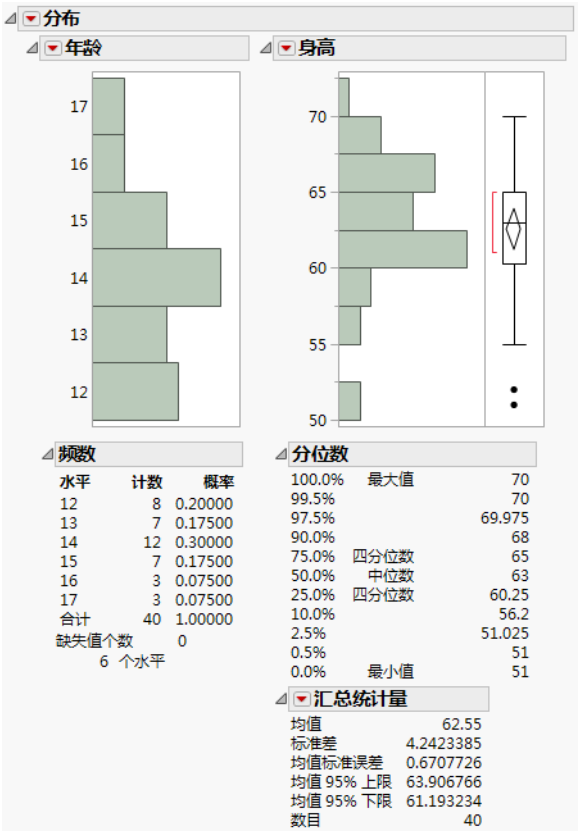
- 将秩、概率得分、正态分位数值等另存为数据表中的新列（“保存”选项）
- 对照指定的常数检验列的均值和标准差（“检验均值”和“检验标准差”选项）
- 拟合各种分布和非参数平滑曲线（“连续拟合”和“离散拟合”选项）
- 为质量控制应用执行过程能力分析
- 置信区间、预测区间和容差区间

“分布”平台的示例

了解如何在 JMP 中创建分布。假定您有 40 个学生的数据，您想要查看年龄和身高在这些学生中的分布情况。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 分布**。
3. 选择**年龄**和**身高**，并点击 **Y，列**。
4. 点击**确定**。

图 3.2 “分布”平台的示例



从直方图中，您会注意到以下内容：

- 年龄不是均匀分布。
- 对于身高，有两个包含极值（可能为离群值）的点。

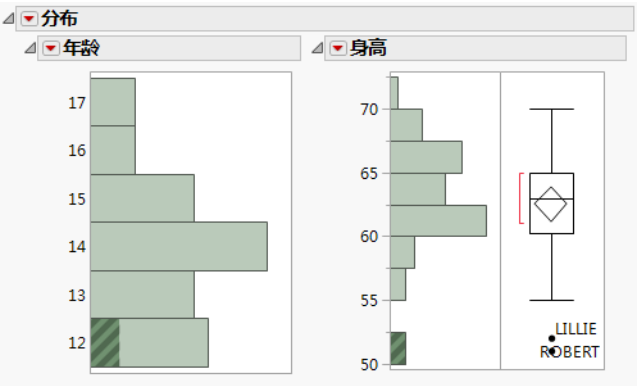
点击身高直方图中标记为 50 的直条，进一步查看潜在离群值。

- 对应年龄在年龄直方图中突出显示。潜在离群值对应的年龄为 12 岁。
- 对应的行在数据表中突出显示。潜在离群值对应的姓名为 Lillie 和 Robert。

向身高直方图中的潜在离群值添加标签。

1. 选择这两个离群值。
2. 右击其中一个离群值并选择**行标签**。
   标签图标将添加至数据表中的对应行。
3. 将箱线图加宽以查看完整标签。

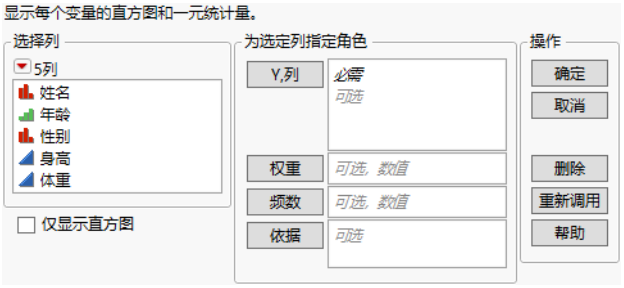
图 3.3 添加了标签的潜在离群值



启动 “分布” 平台

通过选择**分析** > **分布**可启动 “分布” 平台。

图 3.4 “分布” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 指定要分析的变量。将为每个变量显示直方图和关联的报表。

权重 分配一个变量，该变量指定观测在连续 Y 的观测所占的权重。对于分类 Y，“权重” 列将被忽略。基于权重和的所有统计量都受权重影响。

频数 指定一个频数变量。这在数据进行汇总时很有用。在这种情况下，有一列包含 Y 值，另一列包含 Y 值出现的频率。该变量的总和包括在 “汇总统计量” 报表中显示的总计数中（表示为**数目**）。所有其他矩统计量 （均值、标准差等）也都受**频数**变量影响。

依据 为**依据**变量的每个水平生成单独的报表。若指定了多个**依据**变量，将为**依据**变量水平的每种可能组合生成单独的报表。

创建过程能力 （仅当列包含 “规格限” 列属性时才显示。）为包含 “规格限” 列属性的分析列添加 “过程能力” 报表。

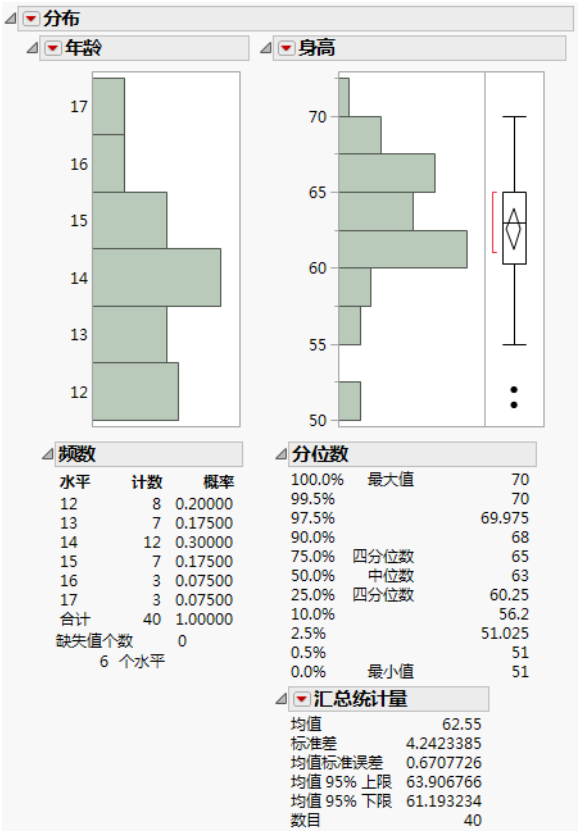
仅显示直方图 从报表窗口中删除直方图之外的所有内容。

有关启动窗口的详细信息，请参见《使用 JMP》。

“分布”报表

初始“分布”报表包含每个变量的直方图和报表。

图 3.5 初始“分布”报表窗口



注意：按照第 34 页的““分布”平台的示例”中的说明生成图 3.5 所示的报表。

红色小三角菜单

“分布”旁边的红色小三角菜单包含影响所有变量的选项。请参见第 43 页的““分布”平台选项”。

每个变量旁边的红色小三角菜单包含仅影响该变量的选项。请参见第 44 页的“用于分类变量的选项”或第 45 页的“用于连续变量的选项”。

提示：若按住 Ctrl 键并选择某个变量选项，该选项将应用于报表中具有相同建模类型的所有变量。

直方图和报表

直方图可直观地显示数据。请参见第 38 页的“直方图”。

分类变量的初始报表包含一个“频数”报表。请参见第 41 页的“‘频数’报表”。

连续变量的初始报表包含一个“分位数”报表和一个“汇总统计量”报表。请参见第 41 页的“‘分位数’报表”和第 41 页的“‘汇总统计量’报表”。

在直方图中替换、添加和删除变量

要替换直方图中的变量，请从关联的数据表的列面板中，将变量拖放到直方图的轴上。

要使用新变量在报表中创建新的直方图，请将该变量拖放到现有直方图之外。可以在现有直方图之前、之间或之后添加新变量和直方图。

要从直方图中删除某个变量，请从红色小三角菜单中选择**删除**。

隐藏和排除的行

若将“隐藏”行状态应用到数据表中的行，相应的点不会显示在显示点的图中。但仍会使用隐藏行构造直方图。

若想要从直方图构造和分析结果中排除行，请应用“排除”行状态。然后从“分布”旁边的红色小三角菜单中选择**重新运行 > 重新运行分析**。数据表中已排除的任何行也会在显示点的图中隐藏。

直方图

直方图可直观地显示数据。对于分类 （名义型或有序型）变量，直方图为有序型或名义型变量的每个水平显示一个直条。对于连续变量，直方图用直条来表示连续变量的分组值。

突出显示数据 点击某个直方图直条或图形中的离群点。对应的行在数据表中突出显示，其他直方图的相应部分也突出显示 （若适用）。请参见第 40 页的“突出显示直条并选择行”。

创建子集 双击某个直方图直条，或右击某个直方图直条并选择**子集**。将创建仅包含选定数据的新数据表。

调整整个直方图的大小 悬停在直方图边框上方，直到看到双向箭头。然后点击并拖动边框。

调整轴的尺度 在轴上点击并拖动重新调节其尺度。

或者，悬停在轴上方直到看到手形工具。然后，双击该轴，并在“轴设置”窗口中设置参数。

调整直方图直条的大小 （仅适用于连续变量。）有多个选项可调整直方图直条的大小。请参见第 39 页的 “调整连续变量对应的直方图直条的大小”。

指定选择内容 在多个直方图中指定选择的数据。请参见第 40 页的“在多个直方图中指定您的选择”。

查看直方图或关联的数据表的其他选项：

- 右击直方图。请参见《使用 JMP》。
- 右击某个轴。您可以添加标签或修改该轴。请参见《使用 JMP》。
- 点击变量旁边的红色小三角，然后选择**直方图选项**。选项根据变量建模类型稍有不同。请参见第 44 页的“用于分类变量的选项”或第 45 页的“用于连续变量的选项”。

调整连续变量对应的直方图直条的大小

使用以下工具或选项调整连续变量对应的直方图直条的大小：

- 抓手（手形）工具
- **设置箱宽度**选项
- **增量**选项

使用抓手工具

使用抓手工具可以快速浏览数据。

1. 选择**工具 > 抓手**。

注意：（仅限 Windows）要查看菜单栏，您可能需要悬停在窗口标题下方的横条上方。您还可以在**文件 > 首选项 > Windows 特定**中更改该设置。

2. 将抓手工具置于直方图中的任意位置。
3. 点击并拖动直方图直条。

将每个直条想象成内装一定数量观测的箱子。对于垂直直方图：

- 将手形工具左移可以增大箱宽度并合并区间。直条数随着直条的增大而减少。
- 将手形工具右移可以减小箱宽度，从而生成更多的直条。
- 将手形工具上下移动可以移动箱在轴上的位置，这将更改每个箱的内容和大小。

注意：若已将直方图的方向改为水平显示，则反转这些方向。将手形工具向下移动可以增大箱宽度，向上移动可以减小箱宽度，向左或向右移动可以移动箱在轴上的位置。

使用“设置箱宽度”选项

使用**设置箱宽度**选项可以为直方图中的所有直条更为精确地设置宽度。要使用“设置箱宽度”选项，请从变量的红色小三角菜单中选择**直方图选项 > 设置箱宽度**。更改箱的宽度值。

使用“增量”选项

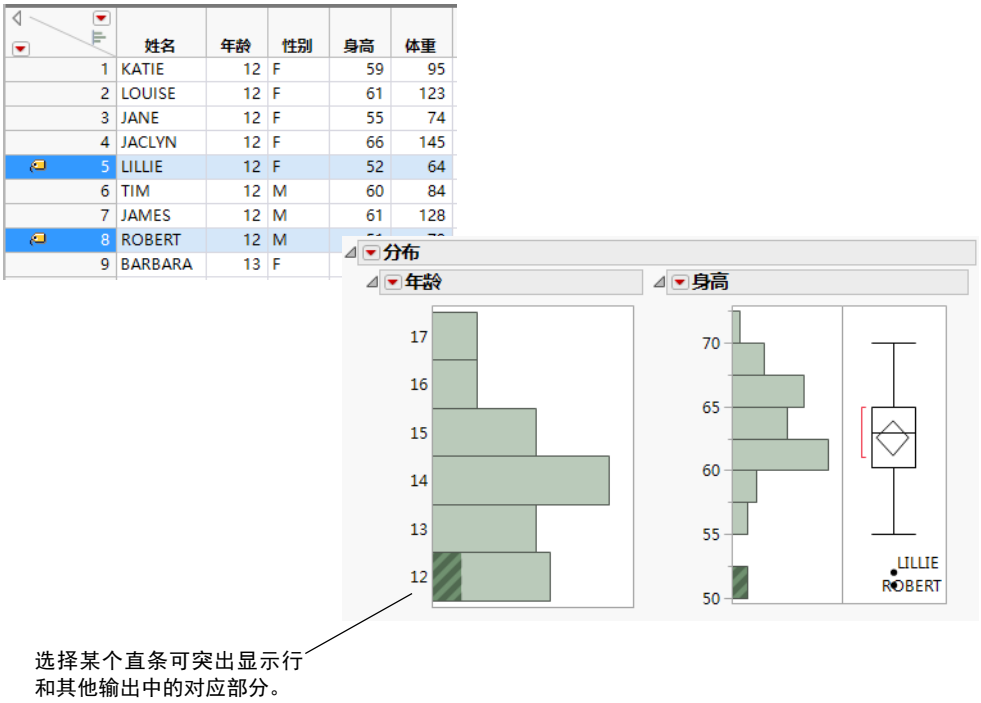
使用**增量**选项也可以精确设置直条宽度。要使用**增量**选项，请双击轴，然后更改“增量”值。

突出显示直条并选择行

点击某个直方图直条可突出显示该直条并选定数据表中相应的行。其他所有图形显示中的相应部分也会突出显示选定内容。图 3.6 显示突出显示**身高**直方图中的某个直条的结果。对应的行在数据表中处于选定状态。

提示：要取消选择特定的直方图直条，请按 Ctrl 键并点击突出显示的直条。

图 3.6 突出显示直条和行



在多个直方图中指定您的选择

在多个直方图中扩大或缩小您的选择：

- 要扩大选择，请按住 Shift 键并选择另一个直条。这相当于使用 or 运算符。
- 要缩小选择，请按住 Ctrl 键和 Alt 键 (Windows)，或 Command 键和 Option 键 (macOS)，同时选择另一个直条。这相当于使用 and 运算符。

有关示例，请参见第 61 页的“在多个直方图中选择数据的示例”。

“频数”报表

对于名义型和有序型变量，“频数”报表列出变量的水平以及关联的频数和概率。

对于分类（名义型或有序型）变量的每个水平，“频数”报表包含下面的列表中说明的信息。分析中省略缺失值。

提示：点击“频数”报表中的某个值，可选择直方图和数据表中的相应数据。

水平 列出响应变量的每个值。

计数 列出响应变量的每个水平对应的行数。若使用 “频数”变量，“计数”将为响应变量每个水平的 “频数”变量的总和。

概率 列出响应变量每个水平出现的概率 （或比例）。通过将计数除以变量总频数 （显示在表的底部）可以得出概率。

概率标准误差 列出概率的标准误差。该列可能隐藏。要显示该列，请在表中右击并选择**列 > 概率标准误差**。

累积概率 包含概率列的累积总和。该列可能隐藏。要显示该列，请在表中右击并选择**列 > 累积概率**。

“分位数”报表

对于连续变量，“分位数”报表列出选定分位数（有时称为**百分位数**）的值。有关统计详细信息，请参见第 69 页的“分位数的统计详细信息”。

“汇总统计量”报表

对于连续变量，“汇总统计量”报表显示均值、标准差和其他汇总统计量。通过从“汇总统计量”旁边的红色小三角菜单中选择**定制汇总统计量**，您可以控制在该报表中显示哪些统计量。

提示：您可以指定每次为连续变量运行“分布”分析时显示在报表中的汇总统计量。选择**文件 > 首选项 > 平台 > 分布汇总统计量**，然后选择您希望显示的汇总统计量。

- “‘汇总统计量’报表的说明” 说明了默认显示的统计量。
- “附加汇总统计量” 说明了您可以使用**定制汇总统计量**窗口向报表添加的其他统计量。

“汇总统计量”报表的说明

均值 估计响应变量所基于的分布的预期值，是该列值的算术平均值。它是非缺失值总和除以非缺失值个数所得的值。

标准差 正态分布主要由均值和标准差来定义。这些参数可在样本变大时很容易地汇总数据：

– 68% 的值位于均值的一个标准差内

- 95% 的值位于均值的两个标准差内
- 99.7% 的值位于均值的三个标准差内

均值标准误差 均值的标准误差，用来估计均值分布的标准差。

均值置信上限和置信下限 均值的 95% 置信限。定义一个极可能包含真正的总体均值的区间。

数目 非缺失值的总数。

附加汇总统计量

权重和 在启动窗口中分配至 “权重”角色的列的总和。“权重和”代替**数目**用作均值计算的分母。

总和 响应值的总和。

方差 样本方差，样本标准差的平方。

偏度 测量偏离或对称程度。

峰度 测量峰尖程度或尾重。有关公式的详细信息，请参见第 71 页的 “峰度”。

变异系数 变异的百分比系数。由标准差除以均值再乘以 100 计算得出。变异系数可用于评估相对变异。例如，用于比较以不同单位或不同量级测量的数据中的变异。

缺失值个数 缺失观测的个数。

零值数 零值的个数。

唯一值数 唯一值的个数。

未校正平方和 未校正的平方和，即值的平方和。

校正平方和 已校正的平方和，即均值偏差的平方和。

自相关性 （仅当未指定“频数”变量时才显示。）一阶自相关性检验行间的残差是否相关。该检验可帮助检测数据中的非随机性。

最小值 代表数据的第 0 百分位数。

最大值 代表数据的第 100 百分位数。

中位数 代表数据的第 50 百分位数。

众数 数据中出现次数最多的值。若有多个众数，则显示最小的众数。

切尾均值 删除最小的 p% 和最大的 p% 数据后计算出的均值。在该窗口底部的**输入切尾均值百分比**文本框中输入 p 的值。若您指定了 “权重”变量，则 “切尾均值”选项不可用。

几何均值 数据的积的第 n 个根。例如，几何均值通常用于计算利率。当数据包含偏斜分布中的较大值时，该统计量也很有用。

注意：负值生成缺失数字，零值 （无负值）生成零。

极差 最大和最小数据之间的差值。

四分位间距 第三和第一四分位数之间的差值。

绝对偏差中位数 （若指定了“权重”变量则不显示。）相对中位数的绝对偏差的中位数。

零比例 等于零的非缺失值的比例。

非零比例 不等于零的非缺失值的比例。

稳健均值 使用 Huber M 估计以不受离群值影响的方式计算的稳健均值。请参见 Huber and Ronchetti (2009)。

稳健标准差 使用 Huber M 估计以不受离群值影响的方式计算的稳健标准差。请参见 Huber and Ronchetti (2009)。

为均值置信区间输入 (1-alpha) 指定均值置信区间的 alpha 水平。

输入切尾均值百分比 指定切尾均值百分比。将从数据两侧切除该百分比的数据。

“汇总统计量”选项

“汇总统计量”旁边的红色小三角菜单包含以下选项：

定制汇总统计量 从列表中选择要显示的统计量。您可以选择或取消选择所有汇总统计量。

显示所有众数 若有多个众数，则显示所有众数。

有关统计详细信息，请参见第 70 页的“汇总统计量”。

“分布”平台选项

“分布”旁边的红色小三角菜单包含影响“分布”平台中的所有报表和图形的选项。

统一尺度 使用相同的最小值、最大值和区间设置所有轴的尺度，以便轻松比较各个分布。

堆叠 将直方图和报表的方向改为水平显示，并将各个分布报表垂直堆叠。取消选择该选项可使报表窗口恢复其原始布局。

排列成行 输入在一行中显示的图的个数。该选项帮助您竖向显示各图，而不是全部显示在一个宽行中。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

用于分类变量的选项

报表窗口中每个变量旁边的红色小三角菜单包含可应用于该变量的附加选项。本节说明可用于分类 （名义型或有序型）变量的选项。

显示选项 请参见第 44 页的 “分类变量的显示选项”。

“直方图”选项 请参见第 44 页的 “分类变量的直方图选项”。

马赛克图 为每个名义型或有序型响应变量显示马赛克条形图。马赛克图是堆叠的条形图，其中每组堆叠条形图的宽度与该组的频数计数成比例。

排序方式 对直方图、马赛克图和 “频数”报表按计数的升序或降序重新排列。要将新顺序另存为列属性，请使用**保存 > 值排序**选项。

检验概率 显示检验假设概率的报表。请参见第 63 页的 ““检验概率”选项的示例”。

置信区间 该菜单包含置信水平。选择所列的值，或选择**其他**输入您自己的值。 JMP 计算得分置信区间。

保存 请参见第 45 页的 “分类变量的保存选项”。

删除 从 “分布”报表中永久删除变量及其所有报表。

分类变量的显示选项

频数 显示或隐藏 “频数”报表。请参见第 41 页的 ““频数”报表”。

水平布局 将直方图和报表的方向改为垂直或水平。

左侧轴 将**计数**、**概率**和**密度**轴移至左侧而不是右侧。

仅当选定**水平布局**时该选项才适用。

分类变量的直方图选项

直方图 显示或隐藏直方图。请参见第 38 页的 “直方图”。

垂直 将直方图的方向从垂直改为水平。

标准误差条 对直方图的每个水平绘制标准误差条。

分离直条 分离直方图直条。

直方图颜色 更改直方图直条的颜色。

计数轴 添加一个轴，用来显示直方图直条所表示的列值的频数。

概率轴 添加一个轴，用来显示直方图直条所表示的列值的比例。

密度轴 添加一个轴，用来显示直方图中直条的长度。

计数轴和概率轴基于以下公式计算：

概率 = (直条宽度)* 密度

计数 = (直条宽度)* 密度 *(合计数)

显示百分比 为每个直方图直条所表示的列值添加百分比标签。

注意：要指定小数位数，右击直方图，然后选择**定制 > 直方图**。

显示计数 为每个直方图直条所表示的列值添加频数标签。

分类变量的保存选项

水平编号 在数据表中创建一个名为**水平 < 列名 >** 的新列。每个观测的水平编号对应于包含该观测的直方图直条。

值排序 （与**排序方式**选项一同使用）在数据表中创建一个新的值排序列属性，用于反映新顺序。

在日志中显示脚本 在日志窗口中显示用于生成当前报表的脚本命令。选择**视图 > 日志**可查看日志窗口。

用于连续变量的选项

报表窗口中每个变量旁边的红色小三角菜单包含可应用于该变量的附加选项。本节说明可用于连续变量的选项。

显示选项 请参见第 46 页的 “连续变量的显示选项”。

“直方图”选项 请参见第 46 页的 “连续变量的直方图选项”。

正态分位数图 帮助您直观显示变量呈正态分布的程度。请参见第 47 页的 “正态分位数图”。

离群值箱线图 显示分布并帮助您标识可能的离群值。请参见第 48 页的 “离群值箱线图”。

分位数箱线图 显示 “分位数”报表中的特定分位数。请参见第 49 页的 “分位数箱线图”。

茎叶图 请参见第 50 页的 “茎叶图”。

CDF 图 创建经验累积分布函数图。请参见第 50 页的 “CDF 图”。

检验均值 执行均值的单样本检验。请参见第 51 页的 “检验均值”。

检验标准差 执行标准差的单样本检验。请参见第 52 页的 “检验标准差”。

检验等价性 评估某个总体均值是否等价于假设值。请参见第 52 页的 “检验等价性”。

置信区间 选择均值和标准差的置信区间。请参见第 53 页的 “置信区间”。

预测区间 为单个观测选择预测区间，或为下一个随机选定的样本的均值和标准差选择预测区间。请参见第 54 页的 “预测区间”。

容差区间 计算一个区间，用来包含至少指定比例的总体。请参见第 54 页的 “容差区间”。

过程能力 测量过程是否符合给定的规格限。请参见第 54 页的 “过程能力”。

连续拟合 拟合连续变量的分布。请参见第 56 页的 “拟合分布”。

离散拟合 （在所有数据值均为整数时可用。）拟合离散变量的分布。请参见第 56 页的 “拟合分布”。

保存 保存有关连续或分类变量的信息。请参见第 54 页的 “预测区间”。

删除 从 “分布”报表中永久删除变量及其所有报表。

连续变量的显示选项

分位数 显示或隐藏 “分位数”报表。请参见第 41 页的 ““分位数”报表”。

设置分位数增量 更改分位数增量或恢复默认分位数增量。

定制分位数 按值或按增量设置定制分位数。您可以指定置信水平并选择是否计算平滑经验似然分位数 （对于大型数据集，这可能会需要一些时间）。

- 有关如何估计加权平均分位数的详细信息，请参见第 69 页的 “分位数的统计详细信息”。
- 有关加权平均分位数的分布自由的置信限的详细信息，请参见 Meeker et al. (2017) 中的第 5.2 节。
- 平滑经验似然分位数基于核密度估计值。有关如何估计这些分位数及其置信限的更多详细信息，请参见 Chen and Hall (1993)。
- 当使用小数频数时，置信区间和平滑经验似然分位数不可用。

汇总统计量 显示或隐藏 “汇总统计量”报表。请参见第 41 页的 ““汇总统计量”报表”。

定制汇总统计量 在 “汇总统计量”报表中添加或删除统计量。请参见第 41 页的 ““汇总统计量”报表”。

水平布局 将直方图和报表的方向改为垂直或水平。

左侧轴 将**计数**、**概率**、**密度**和**正态分位数图**轴移至左侧而不是右侧。

仅当选定**水平布局**时该选项才适用。

连续变量的直方图选项

直方图 显示或隐藏直方图。请参见第 38 页的 “直方图”。

阴影图 用阴影图替代直方图。要了解阴影图，请考虑若直方图的箱宽度改变，那么直方图的外观也随之改变。阴影图叠加箱宽度不同的各个直方图。分布的显著特征在阴影图上的透明度较低。

请注意，以下选项不适用于阴影图：

- 标准误差条
- 显示计数
- 显示百分比

垂直 将直方图的方向从垂直改为水平。

标准误差条 使用标准误差对直方图的每个水平绘制标准误差条。使用手形工具调整直条个数时，标准误差条将自动调整。请参见第 39 页的 “调整连续变量对应的直方图直条的大小” 和第 69 页的 “标准误差条”。

设置箱宽度 更改直方图直条的箱宽度。请参见第 39 页的“调整连续变量对应的直方图直条的大小”。

直方图颜色 更改直方图直条的颜色。

计数轴 添加一个轴，用来显示直方图直条所表示的列值的频数。

注意：若调整直方图直条的大小，计数轴也随之调整大小。

概率轴 添加一个轴，用来显示直方图直条所表示的列值的比例。

注意：若调整直方图直条的大小，概率轴也随之调整大小。

密度轴 密度是直方图中直条的长度。计数和概率都基于以下公式计算：

概率 = (直条宽度)* 密度

计数 = (直条宽度)* 密度 *(合计数)

查看通过 “拟合分布” 选项添加的密度曲线时，密度轴显示曲线的点估计值。

注意：若调整直方图直条的大小，计数轴也随之调整大小。

显示百分比 为每个直方图直条所表示的列值添加比例标签。

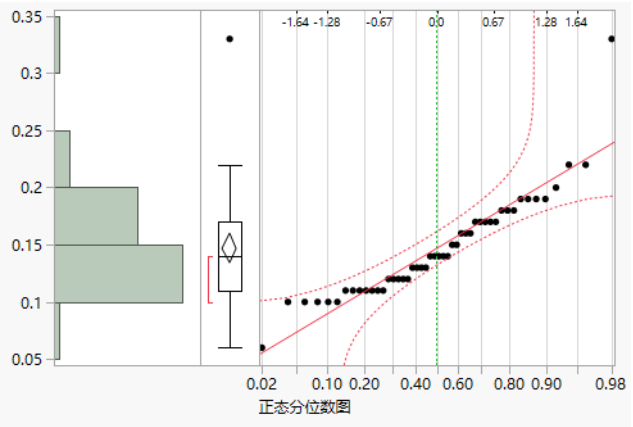
显示计数 为每个直方图直条所表示的列值添加频数标签。

正态分位数图

使用**正态分位数图**选项可直观显示变量呈正态分布的程度。若变量呈正态分布，正态分位数图近似一条对角直线。该类型的图也称为分位数-分位数图或 Q-Q 图。

正态分位数图还显示 Lilliefors 置信界限 (Conover 1980) 以及概率和正态分位数尺度。

图 3.7 正态分位数图



请注意下列信息：

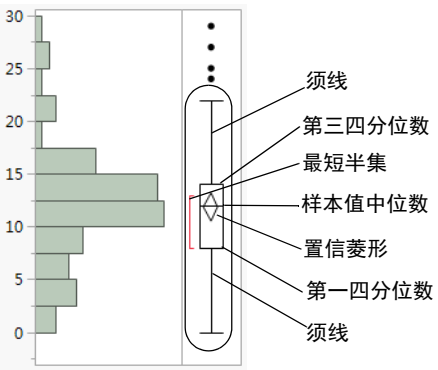
- 垂直轴显示列值。
- 上部水平轴显示正态分位数尺度。
- 下部水平轴显示每个值的经验累积概率。
- 红色虚线显示 Lilliefors 置信界限。

有关统计详细信息，请参见第 71 页的“正态分位数图”。

离群值箱线图

使用离群值箱线图（亦称 Tukey 离群值箱线图）可以查看分布并标识可能的离群值。通常，箱线图显示连续分布的选定分位数。

图 3.8 离群值箱线图



请注意关于离群值箱线图的以下方面：

- 箱中的水平线表示样本值中位数。
- 置信菱形包含均值以及均值的 95% 上限和下限。若横穿菱形中部绘制一条线就会得到均值。菱形的高点和低点分别表示均值的 95% 上限和下限。
- 箱的两端分别表示第 25 分位数和第 75 分位数，也表示为第一和第三**四分位数**。
- 第一和第三四分位数之间的差值称为**四分位间距**。
- 箱的两端延伸出有时称为**须线**的线条。须线从箱的两端延伸到如下距离内的最远数据点：

  第一四分位数 - 1.5*(四分位间距)

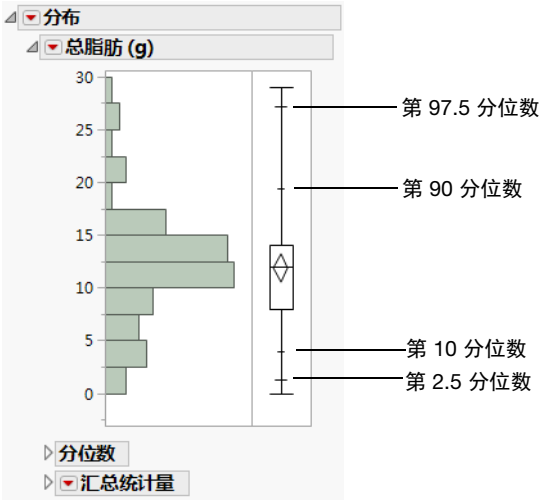
  第三四分位数 + 1.5*(四分位间距)

  若数据点未达到计算范围两端，则由这些数据点的高值和低值（不包括离群值）来确定须线。
- 箱外部的方括号标识**最短半集**，这是最密集的 50% 的观测 (Rousseeuw and Leroy 1987)。
- 要从离群值箱线图中删除对象，请参见第 50 页的“删除离群值或分位数箱线图中的对象”。

分位数箱线图

分位数箱线图显示“分位数”报表中的特定分位数。若分布是对称的，箱线图中的各个分位数彼此之间的距离大致相等。您可以大致查看分布是否对称。例如，若分位数标记在一端聚集，但另一端的间距较大，则分布朝着间距较大的一端偏斜。

图 3.9 分位数箱线图



分位数是指这样的值：第 p 分位数大于 p% 的值。例如，10% 的数据位于第 10 分位数之下，90% 的数据位于第 90 分位数之下。

删除离群值或分位数箱线图中的对象

可以从离群值或分位数箱线图中删除置信菱形和最短半集。可以从单个图形中删除它们，也可以从将来所有图形中删除它们。

要从单个图形中删除它们，请执行以下操作：

1. 右击离群值箱线图并选择**定制**。
2. 点击**箱线图**。
3. 取消选中**置信菱形**或**最短半集**旁边的复选框。

有关“定制图形”窗口的详细信息，请参见《使用 JMP》。

要从将来所有图形中删除它们，请执行以下操作：

1. 选择**文件 > 首选项 > 平台 > 分布**。
2. 撤销选择这些选项：
   - **显示箱线图置信菱形**
   - **显示离群值箱线图最短半集**
3. 点击**确定**。

   现在在分布中添加的任何箱线图都不会包含置信菱形或最短半集。

茎叶图

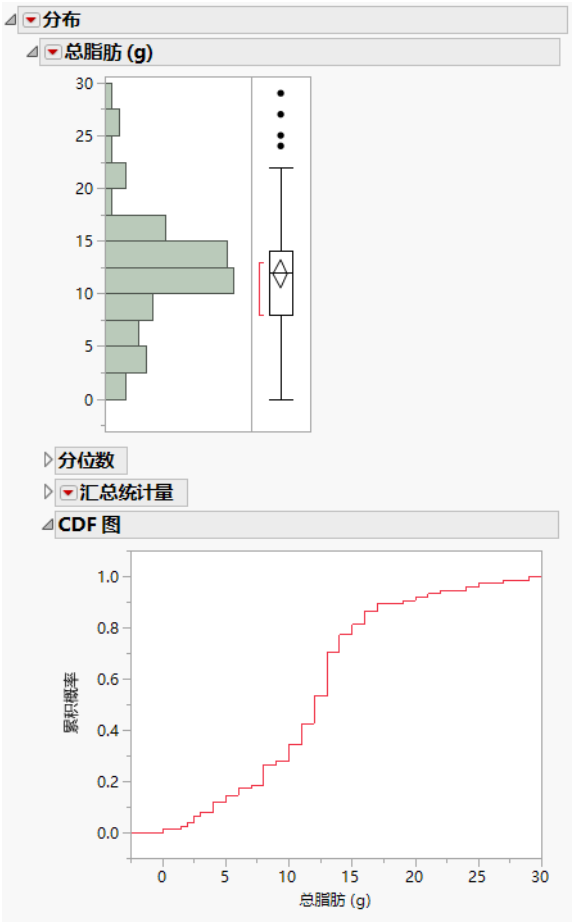
图中每条线都有一个**茎**值，该值是一组列值内的首位数字。**叶**值由值中首位之后的数字按从小到大的顺序构成。您可以看到茎叶连接起来后形成的数据点。在某些情况下，茎叶图中的数字是表中实际数据四舍五入后的数字。茎叶图能积极响应点击和笔刷工具。

注意：茎叶图将小数频数转换为大于等于指定频数的最小整数。

CDF 图

“分布”中的 CDF 图用于创建经验累积分布函数图。使用 CDF 图可以确定等于或小于水平轴上的给定值的数据的百分比。当拟合数据分布时，拟合分布的累积分布函数叠加在 CDF 图上。

图 3.10 CDF 图



例如，在该 CDF 图中，大约 34% 的数据小于总脂肪值 10 克。

检验均值

使用**检验均值**窗口可以为均值指定选项并执行均值的单样本检验。若指定了标准差的值，则执行 *z* 检验。否则会使用样本标准差来执行 *t* 检验。您还可以请求非参数 Wilcoxon 符号秩检验。

反复使用**检验均值**选项可检验不同值。每次检验均值都会出现新的“检验均值”报表。

“检验均值”报表的说明

为“检验均值”计算的统计量

t 检验 （或 z 检验） 列出检验统计量的值，以及双侧和单侧备择项的 *p* 值。

符号秩 （仅当选定 Wilcoxon 符号秩检验时才显示。）列出 Wilcoxon 符号秩检验统计量的值，后跟双侧和单侧备择项的 *p* 值。该检验使用 Pratt 方法处理零值。这是一个非参数检验，其原假设是中位数等于假设值。它假设分布是对称的。请参见第 71 页的 “Wilcoxon 符号秩检验”。

概率值

概率 > |t| 当总体均值等于假设值时，得到绝对 *t* 值大于观测到的 *t* 值的概率。这是观测到的双侧 *t* 检验的显著性的 *p* 值。

概率 >t 当总体均值与假设值没有差别时，得到 *t* 值大于计算的样本 *t* 比的概率。这是右侧检验的 *p* 值。

概率 < t 当总体均值与假设值没有差别时，得到 *t* 值小于计算的样本 *t* 比的概率。这是左侧检验的 *p* 值。

“检验均值”选项的说明

p 值动画 开始 *p* 值的交互式直观表示。支持您更改假设均值，同时观看所做更改会如何影响 *p* 值。

功效动画 开始功效和 beta 的交互式直观表示。您可以更改假设均值和样本均值，同时观看这些更改如何影响功效和 beta。

删除检验 删除均值检验。

检验标准差

使用**检验标准差**选项可执行标准差的单样本检验。反复使用**检验标准差**选项可检验不同值。每次检验标准差都会出现新的 “检验标准差”报表。

检验统计量 提供卡方检验统计量的值。请参见第 73 页的 “标准差检验”。

最小 p 值 当总体标准差与假设值没有差别时，得到更极端的卡方值的概率。请参见第 73 页的 “标准差检验”。

概率 > 卡方 当总体标准差与假设值没有差别时，得到卡方值大于计算的样本卡方的概率。这是观测到的单侧 *t* 检验的显著性的 *p* 值。

概率 < 卡方 当总体标准差与假设值没有差别时，得到卡方值小于计算的样本卡方的概率。这是观测到的单侧 *t* 检验的显著性的 *p* 值。

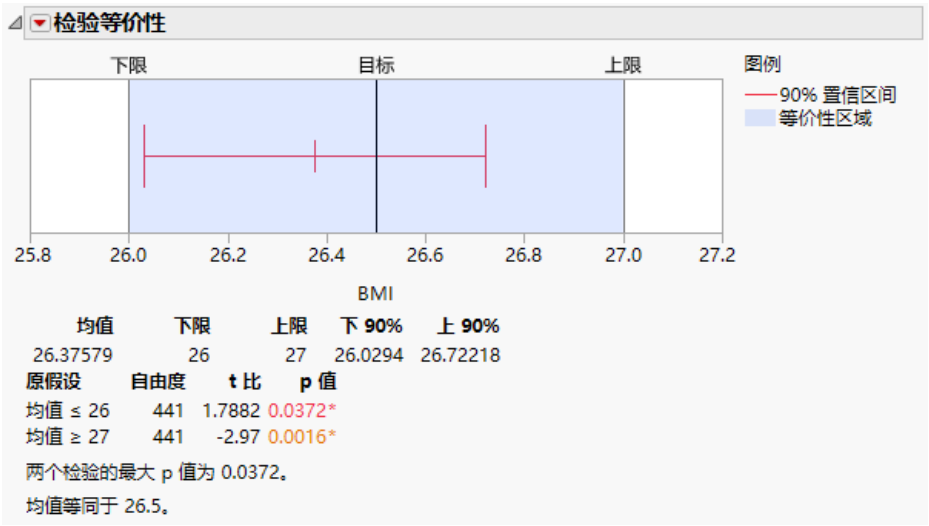
检验等价性

等价性检验评估某个总体均值是否等价于假设值。您必须定义一个阈值差，该阈值差被视为等价于无差值。“检验等价性”选项使用“双单侧检验 (TOST)”方法。针对真正均值与假设值之间的差值超过该阈值的原假设构造双单侧 *t* 检验。若两个原假设都被拒绝，这表示真正差值未超过该阈值。您可以得出结论：均值可被视为实际上等价于假设值。

当您选择“检验等价性”选项时，您需要指定“假设均值”、阈值差（实际视为零的差值）和“置信水平”。“置信水平”为 1 - alpha，其中，alpha 是每个单侧检验的显著性水平。

图 3.11 中的“检验等价性”报表针对 Diabetes.jmp 样本数据表中的变量 BMI。“假设均值”为 26.5，“实际视为零的差值”指定为 0.5。

图 3.11 检验等价性报表



该报表显示以下内容：

- 定义的等价性区域图，其中显示由标记为“下限”和“上限”的垂线定义的目标和边界。
- 计算的均值的置信区间。该置信区间为 1 - 2*alpha 水平区间。
- 一个表，其中显示计算的均值、指定的下限和上限，以及该均值的 (1 - 2*alpha) 水平置信区间。
- 一个显示双单侧检验结果的表。
- 一条注释，其中总结了结果并且表明该均值是否可视为等价于目标值。

置信区间

连续变量的**置信区间**选项显示均值和标准差的置信区间。0.90、0.95 和 0.99 选项计算均值和标准差的双侧置信区间。使用**置信区间** > **其他**选项可选择一个置信水平，并且选择单侧或双侧置信区间。您还可以输入已知的 Sigma 值。若使用已知的 Sigma 值，均值的置信区间将基于 *z* 值而不是 *t* 值。

“置信区间”报表显示均值和标准差参数估计值，以及置信度为 1 - α 时的置信上限和置信下限。

预测区间

预测区间关注的是单个观测，或是下一个随机选定的样本的均值和标准差。该计算假设给定样本是从正态分布中随机选定的。选择单侧或双侧预测区间。

为某个变量选择**预测区间**选项时，将显示“预测区间”窗口。使用该窗口可指定置信水平、将来样本数以及是单侧还是双侧限制。

相关信息

- 有关统计详细信息，请参见第 74 页的“预测区间”。
- 有关示例，请参见第 65 页的“预测区间的示例”。

容差区间

容差区间包含至少指定比例的总体。它是指定总体比例（而不是均值或标准差）的置信区间。Meeker et al. (2017) 与 Tamhane and Dunlop (2000) 的著作中对容差区间进行了完整介绍。

为某个变量选择**容差区间**选项时，将显示“容差区间”窗口。使用该窗口可指定置信水平、要包含的比例、单侧或双侧限制以及方法。有两个可用的方法：“假定服从正态分布”和“非参数”。“假定服从正态分布”选项基于以下假设计算容差区间：样本是从正态分布中随机选定的。“非参数”选项计算分布自由的容差区间。

相关信息

- 有关统计详细信息，请参见第 74 页的“容差区间”。
- 有关示例，请参见第 66 页的“容差区间的示例”。

过程能力

过程能力分析用来测量一个过程相对于给定规格限的表现情况。好的过程不仅稳定，并且能持续生产符合规格限的产品。能力指标测量相对于规格限的过程能力，它由过程中心和变异性来汇总。

规格限

若某列包含“规格限”列属性并且选择了启动窗口上的“创建过程能力”选项，则会自动创建“过程能力”报表。该报表基于正态分布，除非该列还包含“分布”列属性。若该列包含“分布”列属性，则“过程能力”报表基于该列属性中指定的分布。

提示：要一次性向若干列添加规格限，请参见《质量和过程方法》。

若某列不包含规格限，则从分析变量名称旁边的红色小三角中选择**过程能力**，并在“过程能力分析”窗口中设置规格限。

要将报表中的规格限保存为数据表中的列属性，请从“过程能力”红色小三角中选择**将规格限保存为列属性**。在您重复执行过程能力分析时将自动检索保存的规格限。

“过程能力分析”窗口

从分析变量名称旁边的红色小三角中选择“过程能力”选项时，将显示“过程能力分析”窗口。

使用“过程能力分析”窗口指定能力分析选项，包括规格限、分析所基于的分布以及 sigma 估计方法。过程能力要求您选择如何评估 sigma，组内（短期）变异。根据您选择的过程能力选项，将显示不同的子选项。

图 3.12 “过程能力分析”窗口

输入规格限 指定过程能力分析的下规格限、目标和上规格限。以上至少有一个必为非缺失值。若选择“显示限值”选项，则规格限将显示在“分布”平台报表中的直方图上。

过程能力选项 根据您选择的选项，将显示不同的附加选项。选择以下选项之一：

子组大小 = 1 将子组大小设置为 1 并提供更多“移动极差”选项。请参见《质量和过程方法》。

使用子组 ID 列 支持您选择子组 ID 列并提供其他子组和移动极差选项。请参见《质量和过程方法》。

使用常数子组大小 支持您设置常数子组大小并提供其他子组和移动极差选项。请参见《质量和过程方法》。

使用历史 Sigma 为 sigma 分配一个以往接受的值。请参见《质量和过程方法》。

使用非正态分布 支持您选择非正态分布并提供其他非正态分布选项。请参见《质量和过程方法》。

显示组内能力 （仅当在“过程能力选项”中选择“子组大小 = 1”或“使用非正态分布”时才可用。）指定报表中是否显示组内 sigma 的估计值。

指定 Alpha 水平 指定置信限的显著性水平。

“过程能力分析”报表

在“过程能力分析”窗口中点击“确定”后，将显示一个“过程能力”报表，其中包含所选变量的能力报表。有关该报表的详细信息，请参见《质量和过程方法》。

- 有关统计详细信息，请参见《质量和过程方法》。
- 有关示例，请参见第 67 页的“过程能力示例”。
- 有关“过程能力”平台的信息，请参见《质量和过程方法》。

注意：您可以在**文件 > 首选项 > 平台 > 过程能力**中，为分布中的“过程能力”报表内的许多选项设置首选项。

拟合分布

您可以使用“连续拟合”或“离散拟合”子菜单中的选项拟合连续变量的分布。拟合连续变量分布时，曲线将叠加在直方图上，并将“比较分布”报表和“拟合分布”报表添加到报表窗口中。“拟合分布”报表中的红色小三角菜单包含其他选项。请参见第 58 页的“‘拟合分布’选项”。若某列包含“分布”列属性，则默认情况下，将在“分布”报表中拟合该列属性中的分布。

注意：“寿命分布”平台还包含可能使用不同参数化并允许删失观测的分布拟合选项。请参见《可靠性和生存方法》。

连续拟合

“连续拟合”子菜单包含用于拟合连续分布的选项。有关这些分布的参数化的详细信息，请参见第 76 页的“连续拟合分布”。

拟合正态 拟合数据的正态分布。正态分布通常用于针对对称且多数值位于曲线中间的数据建模。

拟合 Cauchy 拟合数据的 Cauchy 分布。Cauchy 分布具有未定义的均值和标准差。尽管大多数数据本身并不服从 Cauchy 分布，但对于包含大量离群值（高达 50%）的数据，它可以用于估计稳健的位置和尺度。

拟合 Student t 拟合数据的 Student *t* 分布。Student *t* 分布是跨越正态分布与 Cauchy 分布之间空间的稳健选项。当 Student *t* 分布中的自由度接近无穷大时，该分布等价于正态分布。当 Student *t* 分布中的自由度等于 1 时，该分布等价于 Cauchy 分布。“分布”平台估计自由度值。

拟合 SHASH 拟合数据的 sinh-arcsinh (SHASH) 分布。SHASH 分布类似于 Johnson 分布，因为它是朝向正态分布的一个变换；但 SHASH 分布是将正态分布作为特例包括在内。该分布可以对称也可以非对称。

拟合指数 （仅当所有观测都非负时才可用。）拟合数据的指数分布。指数分布是向右偏斜的，通常用于对寿命或两个相继事件之间的时间建模。

拟合 Gamma （仅当所有观测都为正时才可用。）对数据拟合 gamma 分布。gamma 分布是用于对正值建模的灵活分布。

拟合对数正态 （仅当所有观测都为正时才可用。）拟合数据的对数正态分布。对数正态分布是向右偏斜的，通常用于对寿命或事件发生前的时间建模。

拟合 Weibull （仅当所有观测都为正时才可用。）拟合数据的 Weibull 分布。Weibull 分布是较为灵活的分布，经常用于对寿命或事件发生前的时间建模。

拟合两正态混合 拟合两正态混合分布。这种灵活分布能够拟合双模态数据。

拟合三正态混合 拟合三正态混合分布。这种灵活分布能够拟合多模态数据。

拟合平滑曲线 使用非参数密度估计 （核密度估计）拟合平滑曲线。通过使用 “非参数密度” 报表中显示的滑块更改带宽来控制平滑量。

拟合 Johnson 对数据拟合 Johnson 分布。拟合和报告三种 Johnson 分布类型（Su、Sb 和 Sl）中最合适的一种。Johnson 系列分布具有很强的数据拟合能力，因为它支持偏度和峰度的每种可能组合。有关 Johnson 分布的选择过程和参数估计的信息，可以在 Slifker 和 Shapiro (1980) 中找到。

拟合 Beta （仅当所有观测都介于 0 和 1 之间时才可用。）对数据拟合 beta 分布。beta 分布适用于对介于 0 和 1 （不包括 0 和 1）之间的数据建模，通常用于对比例或比率建模。

全部拟合 对一个变量拟合所有可用的连续分布。“比较分布” 报表包含有关每个拟合分布的统计量。默认情况下选定最佳拟合分布并显示在直方图上。使用复选框可显示或隐藏拟合报表以及叠加选定分布的曲线。最初，“比较分布” 报表按 AICc 升序排序。

提示：通过双击 “分布” 列中的分布的名称，可以从 “比较分布” 列表中快速删除分布。该操作还将删除相应的 “拟合分布” 报表。

启用旧版拟合器 显示或隐藏 “旧版拟合器” 子菜单。在 JMP 15 中更新了分布拟合的一些功能。该选项支持您使用以往 JMP 版本中的旧功能，这些功能出于兼容目的而得以保留。请参见第 85 页的 “旧版分布拟合器详细信息”。

离散拟合

当所有数据值都是整数时，“离散拟合” 子菜单可用。“离散拟合” 子菜单包含用于拟合离散分布的选项。有关这些分布的参数化的详细信息，请参见第 82 页的 “离散拟合分布”。

拟合 Poisson 拟合数据的 Poisson 分布。Poisson 分布用于对给定时间间隔中的事件数建模，经常表示为计数数据。

拟合负二项 对数据拟合负二项分布。负二项分布用于对指定失败次数之前的成功次数建模。负二项分布也等价于 Gamma Poisson 分布。

拟合零泛滥 Poisson （仅当数据中有零值时才可用。）对数据拟合零泛滥 Poisson 分布。与标准 Poisson 分布相比，零泛滥 Poisson 假设数据中的零值比例更大。

拟合零泛滥负二项 （仅当数据中有零值时才可用。）对数据拟合零泛滥负二项分布。与标准负二项分布相比，零泛滥负二项假设数据中的零值比例更大。

拟合二项 拟合数据的二项分布。二项分布有助于对在 n 次独立试验 （全部具有固定的成功概率 p）中成功的总次数进行建模。样本大小可以指定为所有观测的固定样本大小，也可以指定为数据表中包含每行样本大小的另一列。

注意：指定非常数样本大小时，密度曲线、诊断图和刻画器不可用。

拟合 Beta 二项 对数据拟合 beta 二项分布。beta 二项分布是过度离散版本的二项分布。beta 二项分布对于每个观测都要求样本大小大于 1。样本大小可以指定为所有观测的固定样本大小，也可以指定为数据表中包含每行样本大小的另一列。

注意：指定非常数样本大小时，密度曲线、诊断图和刻画器不可用。

“拟合分布”选项

每个拟合的分布报表都有一个包含更多选项的红色小三角菜单。

密度曲线 使用分布的估计参数在直方图上叠加密度曲线。

诊断图 包含以下选项：

QQ 图 显示或隐藏分位数图。该图显示观测与使用估计参数得到的分位数之间的关系。

PP 图 显示或隐藏分位数-分位数 (PP) 图。该图显示经验累积分布函数 (CDF) 与使用估计参数获得的拟合 CDF 之间的关系。

刻画器 包含以下选项：

“分布”刻画器 显示或隐藏累积分布函数 (CDF) 的预测刻画器。

“分位数”刻画器 显示或隐藏分位数函数的预测刻画器。

保存列 包含以下选项：

保存密度公式 将列保存到数据表中，该数据表包含使用估计的参数值计算的密度公式。

保存分布公式 将列保存到数据表中，该数据表包含使用估计的参数值计算的累积分布函数 (CDF) 公式。

保存模拟公式 将使用估计参数生成模拟值的公式所在的列保存到数据表中。该列可用在模拟实用工具中，作为“换入列”。请参见第 289 页的“模拟”一章。

保存变换 （仅可用于 Johnson 和 SHASH 分布拟合。）将包含变换公式的列保存到数据表中。该公式可用于通过拟合分布对分析列进行正态变换。

拟合优度 （不可用于 Johnson、平滑曲线、正态混合、二项、或 Beta 二项分布。）显示或隐藏“拟合优度检验”报表，其中包含拟合分布的拟合优度检验。对于连续拟合，这是 Anderson-Darling 检验。该检验的 p 值使用参数 bootstrap 进行模拟，这类似于 Stephens

(1974) 第 4.1 节中所述的过程。对于离散拟合，这是 Pearson 卡方检验。对于正态分布，当样本小于等于 2000 且没有固定参数时，还将报告 Shapiro-Wilk 正态性检验。

固定参数 （不可用于 Johnson 分布或平滑曲线拟合。）支持您固定参数并重新估计未固定的参数。同时显示“适当性似然比检验”报表，用于检验新参数，以确定这些参数是否能拟合数据。

过程能力 （不可用于 Cauchy、Student *t* 或离散分布拟合。）支持您使用拟合分布创建过程能力分析，该能力是衡量过程在规格限方面的表现的测度。从“拟合分布”红色小三角菜单中选择 “过程能力”选项时，将显示一个窗口，其中包含以下选项：

输入规格限 支持您手动输入规格限。要使用拟合分布计算规格限，请将此部分留空，改为使用“计算分位数规格限选项”下的选项。

计算分位数规格限选项 支持您根据拟合分布计算规格限。可使用两种方法。

在第一种方法中，输入与拟合分布的分位数相关联的概率以计算规格限。

在第二种方法中，输入用于计算规格限的 K-Sigma 乘数值。该方法提供用于创建双侧或单侧限制的选项。

输入概率或 sigma 乘数的值后，点击**计算规格限**以计算规格限。这些限值将输进“输入规格限”面板。点击**确定**接受这些限值并生成“过程能力”报表。

过程能力选项 包含以下选项：

“移动极差选项”分级显示项包含支持您选择移动极差统计量类型的选项。请参见《质量和过程方法》。

“非正态分布选项”分级显示项包含一些选项，这些选项支持您选择用于非正态过程能力计算的方法。请参见《质量和过程方法》。

有关 “过程能力”选项和报表的详细信息，请参见 《质量和过程方法》。

注意：您可以在**文件 > 首选项 > 平台 > 过程能力**中，为分布中的 “过程能力”报表内的许多选项设置首选项。

删除拟合 从报表窗口中删除分布拟合。

连续变量的保存选项

使用“保存”菜单选项可保存有关连续变量的信息。每个“保存”选项都在当前数据表中生成一个新列。通过在“保存”命令名称之后追加变量名（在下面的定义中表示为**<列名>**）来为新列命名（表 3.1）。

反复选择“保存”选项可在不同环境下（比如在合并直方图直条之前和之后）多次保存相同的信息。若多次使用“保存”选项，列名将被编号（名称1、名称2，依此类推），以确保列名唯一。

表 3.1 “保存”选项的说明

选项	添加至数据表的列	说明
水平编号	水平: <列名>	每个观测的水平编号对应于包含该观测的直方图直条。直方图直条从 1 开始由低到高编号。 **注意：**要保持源信息，向新列添加了值标签，但默认关闭值标签。
水平中点值	中点值: <列名>	水平下界加上水平宽度的一半即得到每个观测的中点值。 **注意：**要保持源信息，向新列添加了值标签，但默认关闭值标签。
秩	秩: <列名>	为每个相应列值提供从 1 开始的排序。重复的响应值将根据其在数据表中出现的顺序来指定连续秩。
平均秩	平均秩: <列名>	若某个值是唯一的，则平均秩与秩相同。若某个值出现 k 次，平均秩即该值的秩总和除以 k 所得的值。
概率得分	概率: <列名>	对于 N 个非缺失值的得分，值的概率得分通过该值的平均秩除以 $N+1$ 计算得出。该列类似于经验累积分布函数。
正态分位数	正态分位数: <列名>	保存正态分位数。请参见第 71 页的“正态分位数图”。
标准化	标准化: <列名>	保存标准化值。请参见第 74 页的“保存标准化数据”。
中心化	中心化: <列名>	保存用于以零点为中心的值。
标准化稳健	稳健标准 <列名>	保存一列，其中包含以稳健均值为中心且使用稳健标准差标准化的响应值。
中心化稳健	中心化稳健 <列名>	保存包含以稳健均值为中心的响应值的列。
在日志中显示脚本	（无）	将脚本打印输出至日志窗口。运行脚本以重新创建分析。

“分布”平台的更多示例

- “在多个直方图中选择数据的示例”
- “使用“依据”变量的示例”
- ““检验概率”选项的示例”
- “预测区间的示例”
- “容差区间的示例”
- “过程能力示例”

在多个直方图中选择数据的示例

本例显示如何一次选择并突出显示多个直方图中的数据。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **分布**。
3. 选择**类型**和**公司规模**，并点击 **Y，列**。
4. 点击**确定**。

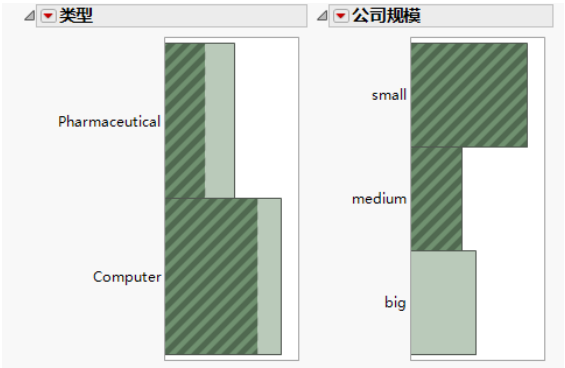
   您想要查看小型公司的类型分布。
5. 点击 small 旁边的直条。

   您可以看到小型计算机公司比小型医药公司多。要扩大选择，请添加中型公司。
6. 按住 Shift 键。在 “公司规模”直方图中，点击 medium 旁边的直条。

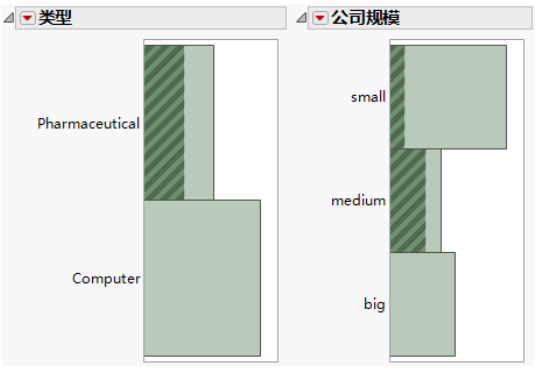
   您可以看到小型和中型公司的类型分布。请参见图 3.13 中的左图。现在缩小选择，假定您只想查看小型和中型医药公司。
7. 按住 Ctrl 键和 Shift 键 （在 Windows 上），或 Command 键和 Shift 键 （在 macOS 上）。在 “类型”直方图中，点击 Computer 直条将其取消选定。

   您可以看到小型和中型公司中有多少是医药公司。请参见图 3.13 中的右图。

图 3.13 在多个直方图中选择数据



使用 Shift 键扩大选择。



使用 Ctrl + Shift (Windows) 或 Command + Shift (macOS) 来缩小选择。

使用“依据”变量的示例

本例显示如何使用“依据”变量为分类变量的每个水平创建单独的分析。

1. 选择**帮助** > **样本数据库**，然后打开 Lipid Data.jmp。
2. 选择**分析** > **分布**。
3. 选择**胆固醇**并点击 **Y，列**。
4. 选择**性别**并点击**依据**。

   这将为 “依据”变量**性别**的每个水平 （female 和 male）生成单独的分析。
5. 点击**确定**。

更改直方图和报表的方向。

6. 点击 “分布”红色小三角并选择**堆叠**。

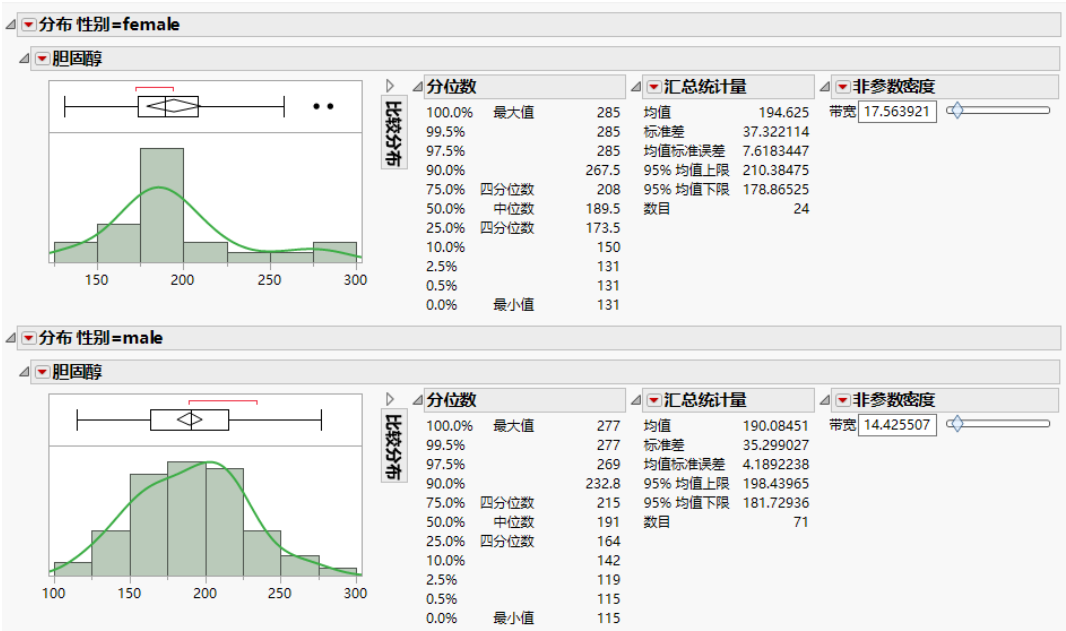
向两个直方图添加平滑曲线。

7. 按住 Ctrl 键。点击 “胆固醇”红色小三角并选择**连续拟合** > **拟合平滑曲线**。

隐藏 “比较分布”报表。

8. 按住 Ctrl 键。点击**比较分布**旁边的灰色展开图标。

图 3.14 按 “性别”分隔分布



“检验概率”选项的示例

这些示例显示如何为具有两个或更多水平的变量创建 “检验概率”报表。

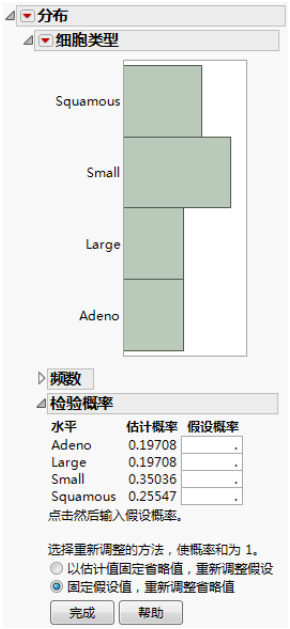
为包含两个以上水平的变量启动检验概率报表：

1. 选择**帮助 > 样本数据库**，然后打开 VA Lung Cancer.jmp。
2. 选择**分析 > 分布**。
3. 选择**细胞类型**并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “单元格类型”红色小三角并选择**检验概率**。
   请参见图 3.15 中的左图。

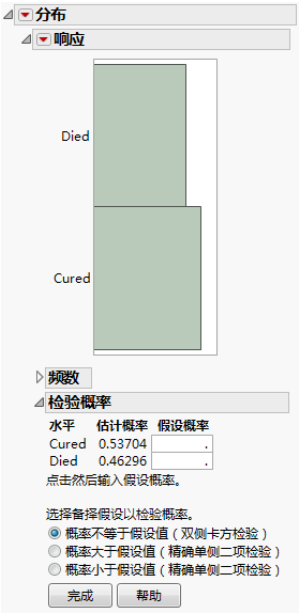
为包含正好两个水平的变量启动检验概率报表：

1. 选择**帮助 > 样本数据库**，然后打开 Penicillin.jmp。
2. 选择**分析 > 分布**。
3. 选择**响应**并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “响应”红色小三角并选择**检验概率**。
   请参见图 3.15 中的右图。

图 3.15 “检验概率”选项的示例



包含两个以上水平的变量的报表选项



包含正好两个水平的变量的报表选项

生成“检验概率”报表的示例

为包含两个以上水平的变量生成检验概率报表：

1. 请参考图 3.15 中的左图。在全部四个“假设概率”字段中均键入 0.25。
2. 点击**固定假设值，重新调整省略值**按钮。
3. 点击**完成**。

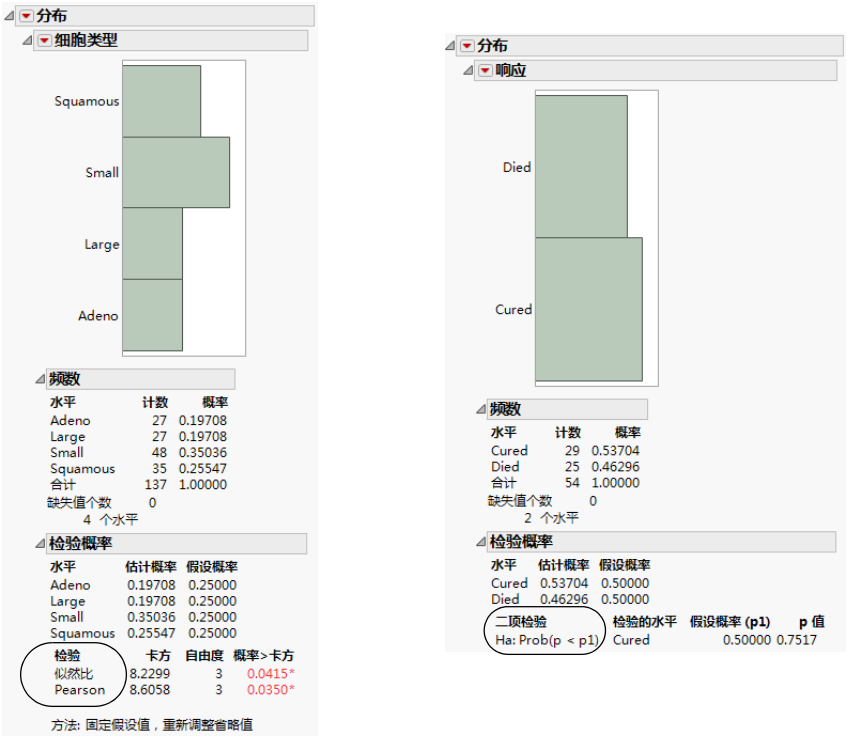
将计算似然比和 Pearson 卡方检验。请参见图 3.16 中的左图。

为包含正好两个水平的变量生成检验概率报表：

1. 请参考图 3.15 中的右图。在两个“假设概率”字段中均键入 0.5。
2. 点击**概率小于假设值**按钮。
3. 点击**完成**。

将为二项检验计算精确概率。请参见图 3.16 中的右图。

图 3.16 “检验概率”报表的示例

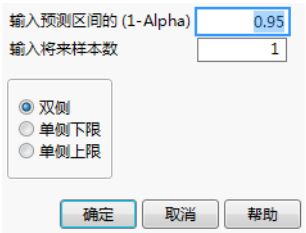


预测区间的示例

本例显示如何将预测区间添加到分布报表中。假定您想要计算臭氧水平的下十个观测的预测区间。

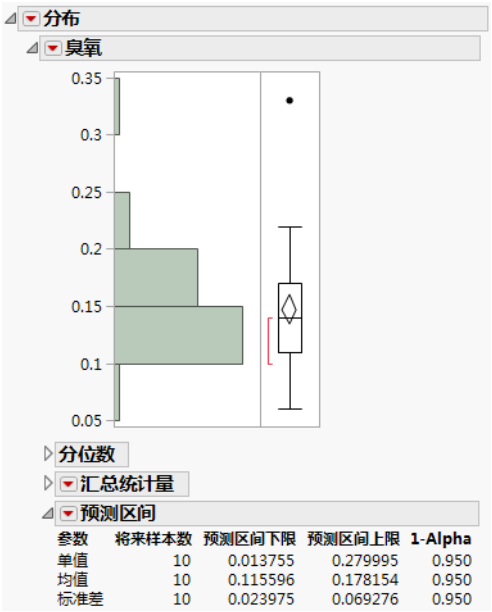
1. 选择**帮助** > **样本数据库**，然后打开 Cities.jmp。
2. 选择**分析** > **分布**。
3. 选择**臭氧**并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “臭氧”红色小三角并选择**预测区间**。

图 3.17 “预测区间”窗口



6. 在“预测区间”窗口中的**输入将来样本数**旁边键入 10。
7. 点击**确定**。

图 3.18 “预测区间”报表的示例



在本例中，您有 95% 的把握确信以下情况：

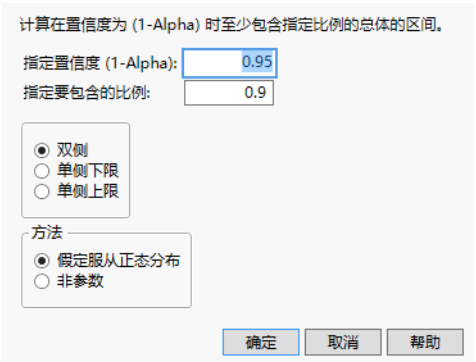
- 下十个观测中的每一个观测都将介于 0.013755 和 0.279995 之间。
- 下十个观测的均值将介于 0.115596 和 0.178154 之间。
- 下十个观测的标准差将介于 0.023975 和 0.069276 之间。

容差区间的示例

本例显示如何将容差区间添加到分布报表中。假定您要估计包含 90% 的臭氧水平测量值的区间。

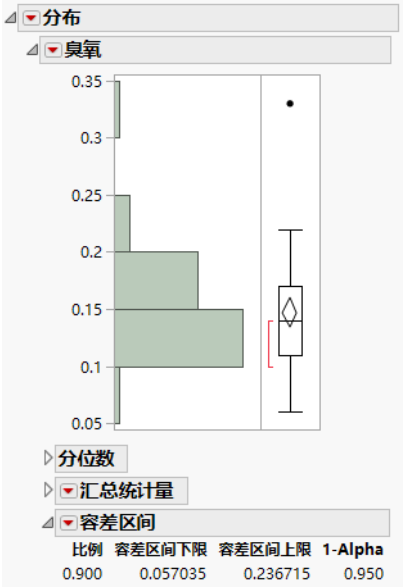
1. 选择**帮助 > 样本数据库**，然后打开 Cities.jmp。
2. 选择**分析 > 分布**。
3. 选择**臭氧**并点击 **Y，列**。
4. 点击**确定**。
5. 点击“臭氧”红色小三角并选择**容差区间**。

图 3.19 “容差区间”窗口



6. 保持默认选择，然后点击**确定**。

图 3.20 “容差区间”报表的示例



在本例中，根据容差区间下限值和容差区间上限值，您有 95% 的把握确信至少 90% 的总体介于 0.057035 和 0.236715 之间。

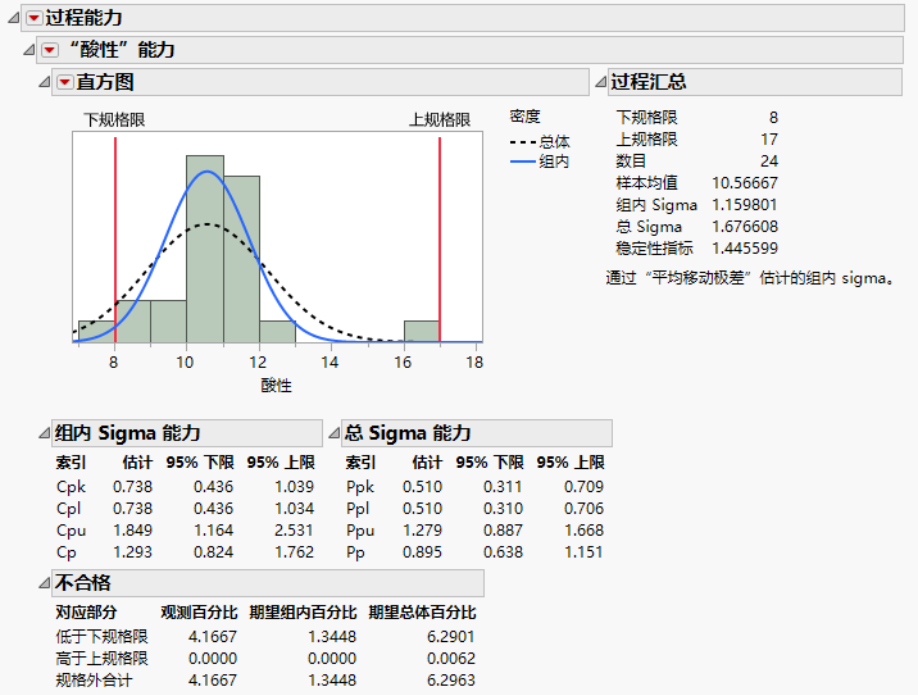
过程能力示例

本例显示如何将过程能力结果添加到分布报表中。假定您想知道腌菜在酸度上有什么特点。下规格限和上规格限分别为 8 和 17。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/Pickles.jmp。

2. 选择**分析 > 分布**。
3. 选择**酸性**并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “酸性”红色小三角并选择**过程能力**。
6. 为 LSL （下规格限）键入 8。
7. 为 USL （上规格限）键入 17。
8. 点击**确定**。

图 3.21 “过程能力”报表示例



过程能力结果添加到报表中。规格限显示在“过程能力”报表中的直方图上，以便可以直观地将数据与限值进行比较。您可以看到，某些酸度水平低于下规格限，有些则非常接近上规格限。Ppk 值为 0.510，指示相对于给定的规格限，过程不符合要求。

“分布”平台的统计详细信息

- “标准误差条”
- “分位数的统计详细信息”
- “汇总统计量”
- “正态分位数图”
- “Wilcoxon 符号秩检验”
- “标准差检验”
- “正态分位数”
- “保存标准化数据”
- “预测区间”
- “容差区间”
- “连续拟合分布”
- “离散拟合分布”

标准误差条

标准误差条是使用标准误差 $\sqrt{np_i(1-p_i)}$ 计算的，其中 $p_i=n_i/n$。

分位数的统计详细信息

本节介绍如何计算分位数。

要计算列中 n 个非缺失值的第 p 分位数，请将 n 个值按升序排列并将这些列值称为 $y_1, y_2, \ldots, y_n$。将第 p 分位数的秩数计算为 $p / 100(n + 1)$。

- 若结果为整数，则第 p 分位数就是该秩的相应值。
- 若结果不为整数，则通过插值来查找第 p 分位数。第 p 分位数（表示为 q_p）定义如下：

  $$q_p = (1-f)y_i + (f)y_{i+1}$$

  其中：
  - n 是变量的非缺失值个数
  - $y_1, y_2, \ldots, y_n$ 代表变量的排序值
  - y_{n+1} 视为 y_n
  - i 是 $(n+1)p$ 的整数部分，f 是其小数部分
  - $(n + 1)p = i + f$

例如，假定某个数据表有 15 行，而您想要查找某一连续列的第 75 分位数和第 90 分位数的值。将该列按升序排列后，包含这些分位数的秩计算如下：

$$\frac{75}{100}(15+1) = 12 \text{ 且 } \frac{90}{100}(15+1) = 14.4$$

值 y_{12} 是第 75 分位数。第 90 分位数通过计算排在第 14 位和第 15 位的值的加权平均值内插得到：$y_{90} = 0.6y_{14} + 0.4y_{15}$。

汇总统计量

本节包含 “分布汇总统计量”报表中的特定统计量的统计详细信息。

均值

均值是非缺失值总和除以非缺失值个数所得的值。若指定了**权重**或**频数**变量，则按如下方法计算均值：

1. 每个列值均乘以其相应的权重或频数。
2. 将这些值相加后再除以权重和或频数和。

标准差

标准差测量分布在均值附近的散布程度。经常表示为 s，并且是样本方差（表示为 s^2）的平方根。

$$s = \sqrt{s^2}$$

其中

$$s^2 = \sum_{i=1}^{N} \frac{w_i(y_i - \bar{y}_w)^2}{N-1}$$

$\bar{y}_w$ = 加权均值

均值标准误差

均值标准误差通过样本标准差 (s) 除以 N 的平方根计算得出。在启动窗口中，若指定了“权重”或“频数”列，则分母为权重和或频数和的平方根。

偏度

偏度基于均值的三阶矩，计算如下：

$$\sum w_i^{\frac{3}{2}} z_i^3 \frac{N}{(N-1)(N-2)} \text{ 其中 } z_i = \frac{x_i - \bar{x}}{s}$$

w_i 是权重项（= 1 即表示各项权重相同）

峰度

峰度基于均值的四阶矩，计算如下：

$$\frac{n(n+1)}{(n-1)(n-2)(n-3)}\sum_{i=1}^{n} w_i^2\left(\frac{x_i-\bar{x}}{s}\right)^4-\frac{3(n-1)^2}{(n-2)(n-3)}$$

其中 w_i 是权重项（= 1 即表示各项权重相同）。使用该公式，正态分布的峰度为 0。该公式经常称为超值峰度。

正态分位数图

每个值的经验累积概率计算如下：

$$\frac{r_i}{N+1}$$

其中，r_i 是第 i 个观测的秩，N 是非缺失（且未排除）观测的个数。

正态分位数值计算如下：

$$\Phi^{-1}\left(\frac{r_i}{N+1}\right)$$

其中 Φ 是正态分布的累积概率分布函数。

这些正态分位数值是对正态分布应有的次序统计量的 Van Der Waerden 近似。

Wilcoxon 符号秩检验

可使用 Wilcoxon 符号秩检验来检验单一总体的中位数，或在配对数据中检验共同的中位数。在配对的情况下，该检验简化为检验由配对差值组成的单一总体的中位数为零。该检验假定该相应总体是对称的。

Wilcoxon 检验支持结值。使用 Pratt 建议的方法，针对差值为 0 调整了检验统计量。请参见 Lehmann and D′Abrera (2006)、Pratt (1959) 和 Cureton (1967)。

检验单一总体的中位数

- 有 N 个观测：
  $X_1, X_2,..., X_N$
- 原假设为：
  H_0：X 的分布围绕 m 对称。
- 观测值与假设值 m 之间的差值计算如下：
  $D_j = X_j - m$

检验包含配对数据的两个总体的中位数是否相等

Wilcoxon 符号秩检验的特例适用于配对数据。

- 有来自两个总体的 N 对观测：
  $X_1, X_2, ..., X_N$ 和 $Y_1, Y_2, ..., Y_N$
- 原假设为：
  H_0: $X - Y$ 的分布围绕 0 对称。
- 观测对之间的差值计算如下：
  $D_j = X_j - Y_j$

Wilcoxon 符号秩检验统计量

检验统计量基于符号秩的总和。符号秩定义如下：

- 差值的绝对值 $|D_j|$ 按最小到最大的顺序排列。
- 秩从值 1 开始，即使有差值为零。
- 若存在绝对差值结值，则为其分配观测秩的平均值，即**中秩**。

用 R_j 来表示差值 D_j 的秩或中秩。D_j 的符号秩定义如下：

- 若差值 D_j 为正数，则符号秩为 R_j。
- 若差值 D_j 为 0，则符号秩为 0。
- 若差值 D_j 为负数，则符号秩为 $-R_j$。

符号秩统计量计算如下：

$$S = \frac{1}{2}\sum_{j=1}^{N} \text{符号秩}$$

定义以下公式：

d_0 是等于 0 的符号秩数目

R^+ 是正符号秩的总和

则以下公式成立：

$$S = R^+ - \frac{1}{4}[N(N+1) - d_0(d_0+1)]$$

Wilcoxon 符号秩检验 P 值

对于 $N \leq 20$，计算精确 p 值。

对于 $N > 20$ 的情况，使用如下定义的统计量的 Student t 近似。请注意它应用了结值校正。请参见 Iman (1974) 和 Lehmann and D’Abrera (2006)。

在原假设下， S 的均值为 0。 S 的方差通过以下公式得出：

$$Var(S) = \frac{1}{24}\left[N(N+1)(2N+1) - d_0(d_0+1)(2d_0+1) - \frac{1}{2}\sum_{i>0} d_i(d_i+1)(d_i-1)\right]$$

Var(S) 的表达式中最后的求和是在校正结值。符号 d_i (i > 0) 表示非零符号秩的第 i 组中的值个数。（若给定的符号秩没有结值，则 $d_i = 1$ 并且被加数为 0。）

下面公式得出的统计量 t 服从自由度为 N - 1 的近似 t 分布：

$$t = \frac{S}{\sqrt{\frac{N \cdot Var(S) - S^2}{N-1}}}$$

标准差检验

以下是计算该检验统计量的公式：

$$\frac{(n-1)s^2}{\sigma^2}$$

当总体服从正态分布时，该检验统计量呈现自由度为 n - 1 的卡方变量的分布。

最小 p 值是双侧检验的 p 值，计算如下：

2*min(p1,p2)

其中，p1 是单尾 p 值下限，p2 是单尾 p 值上限。

正态分位数

正态分位数值计算如下：

$$\Phi^{-1}\left(\frac{r_i}{N+1}\right)$$

其中：

Φ 是正态分布的累积概率分布函数。

r_i 是第 i 个观测的秩。

r 是非缺失观测数。

保存标准化数据

标准化值使用以下公式计算：

$$\frac{X-\overline{X}}{S_X}$$

其中：

X 是原始列

$\overline{X}$ 是列 X 的均值

S_X 是列 X 的标准差

预测区间

预测区间定义如下：

- 对于 m 个将来观测：

$$[\underset{\sim}{y}_m, \tilde{y}_m] = \overline{X} \pm t_{(1-\alpha/2m;n-1)} \times \sqrt{1+\frac{1}{n}} \times s \quad (m \geq 1)$$

- 对于 m 个将来观测的均值：

$$[Y_l, Y_u] = \overline{X} \pm t_{(1-\alpha/2,\, n-1)} \times \sqrt{\frac{1}{m}+\frac{1}{n}} \times s \quad (m \geq 1)$$

- 对于 m 个将来观测的标准差：

$$[s_l, s_u] = \left[s \times \sqrt{\frac{1}{F_{(1-\alpha/2;(n-1,\, m-1))}}},\; s \times \sqrt{F_{(1-\alpha/2;(m-1,\, n-1))}} \right] \quad (m \geq 2)$$

其中， m = 将来观测数， n = 当前分析样本中的点数。

- 通过在分位数函数中使用 1-α 可形成单侧区间。

请参见 Meeker et al.(2017, ch.4)。

容差区间

本节包含单侧和双侧容差区间的统计详细信息。

基于正态分布的区间

单侧区间

单侧区间计算如下：

下限 $=\bar{x} - g's$

上限 $=\bar{x} + g's$

其中

$$g' = t(1-\alpha, n-1, \Phi^{-1}(p)\cdot\sqrt{n})/\sqrt{n}$$

s 是标准差

t 是非中心 t 分布的分位数

Φ^{-1} 是标准正态分位数

双侧区间

双侧区间计算如下：

$$[T_{p_L}, T_{p_U}] = [\bar{x} - g_{(1-\alpha/2;p,n)}s, \bar{x} + g_{(1-\alpha/2;p,n)}s]$$

其中， s 是标准差并且 $g_{(1-\alpha/2;\,p,n)}$ 是常数。

要确定 g，请考虑容差区间覆盖的总体中所占比例。Tamhane and Dunlop (2000) 按如下方式定义该分数：

$$\Phi\left(\frac{\bar{x} + gs - \mu}{\sigma}\right) - \Phi\left(\frac{\bar{x} - gs - \mu}{\sigma}\right)$$

其中， Φ 表示标准正态 cdf （累积分布函数）。

因此， g 可对以下方程求解：

$$P\left\{\Phi\left(\frac{\overline{X} + gs - \mu}{\sigma}\right) - \Phi\left(\frac{\overline{X} - gs - \mu}{\sigma}\right) \geq 1-\gamma\right\} = 1-\alpha$$

其中， $1-\gamma$ 是容差区间中包含的全部将来观测的比例。

有关基于正态分布的容差区间的详细信息，请参见 Meeker et al.(2017) 中的表 J.1a、J.1b、J.6a 和 J.6b。

非参数区间

单侧下限

100(1 - α)% 单侧容差下限 （包含至少比例为 β 的来自样本大小 n 的抽样分布）是次序统计量 $x_{(l)}$。指数 l 计算如下：

$$l = n - \Phi_{bin}^{-1}(1-\alpha, n, \beta)$$

其中， $\Phi^{-1}_{\text{bin}}(1-\alpha, n, \beta)$ 是试验次数为 n 且成功概率为 β 的二项分布的第 $(1-\alpha)$ 分位数。

实际置信水平计算为 $\Phi_{bin}(n\text{-}l, n, \beta)$，其中 $\Phi_{bin}(x, n, \beta)$ 是试验次数为 n 且成功概率为 β 的二项分布随机变量小于等于 x 的概率。

请注意，要计算单侧分布自由的容差区间的下限，样本大小 n 必须至少与 $(\log\alpha)/(\log\beta)$ 一样大。

单侧上限

100(1 - α)% 单侧容差上限 （包含至少比例为 β 的来自样本大小 n 的抽样分布）是次序统计量 $x_{(u)}$。指数 u 的计算方式如下：

$$u = 1 + \Phi_{bin}^{-1}(1-\alpha, n, \beta)$$

其中， $\Phi^{-1}_{bin}(1\text{-}\alpha, n, \beta)$ 是试验次数为 n 且成功概率为 β 的二项分布的第 (1 - α) 分位数。

实际置信水平计算为 $\Phi_{bin}(u\text{-}1, n, \beta)$，其中 $\Phi_{bin}(x, n, \beta)$ 是试验次数为 n 且成功概率为 β 的二项分布随机变量小于等于 x 的概率。

请注意，要计算单侧分布自由的容差区间的上限，样本大小 n 必须至少与 $(\log\alpha)/(\log\beta)$ 一样大。

双侧区间

100(1 - α)% 双侧容差区间 （包含至少比例为 β 的来自样本大小 n 的抽样分布）计算如下：

$$[\tilde{T}_{p_L}, \tilde{T}_{p_U}] = [x_{(l)}, x_{(u)}]$$

其中， $x_{(i)}$ 是第 i 个次序统计量， l 和 u 计算如下：

让 $\nu = n - \Phi^{-1}_{bin}(1\text{-}\alpha, n, \beta)$，其中 $\Phi^{-1}_{bin}(1\text{-}\alpha, n, \beta)$ 是试验次数为 n 且成功概率为 β 的二项分布的第 (1 - α) 个分位数。若 ν 小于 2，则无法计算双侧分布自由的容差区间。若 ν 大于等于 2，则 $l = \text{floor}(\nu/2)$ 且 $u = \text{floor}(n + 1 - \nu/2)$。

实际置信水平计算为 $\Phi_{bin}(u\text{-}l\text{-}1, n, \beta)$，其中 $\Phi_{bin}(x, n, \beta)$ 是试验次数为 n 且成功概率为 β 的二项分布随机变量小于等于 x 的概率。

请注意，要计算双侧分布自由的容差区间，样本大小 n 必须至少与以下等式中的 n 一样大。

$$1 - \alpha = 1 - n\beta^{n-1} + (n-1)\beta^{n}$$

有关分布自由的容差区间的详细信息，请参见 Meeker et al.(2017, sec.5.3)。

连续拟合分布

本节包含 “连续拟合”菜单中的选项的统计详细信息。

拟合正态

“拟合正态”选项估计正态分布的两个参数：

- μ（均值）定义分布在 x 轴上所处的位置

- σ（标准差）定义分布的离散或散布程度

当 μ = 0 且 σ = 1 时出现标准正态分布。

pdf: $\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$ $-\infty < x < \infty;\ -\infty < \mu < \infty;\ 0 < \sigma$

$E(x) = \mu$

$Var(x) = \sigma^2$

拟合 Cauchy

“拟合 Cauchy”选项使用位置 μ 和尺度 σ 拟合 Cauchy 分布。

pdf: $\left\{\pi\sigma\left[1+\left(\frac{x-\mu}{\sigma}\right)^2\right]\right\}^{-1}$ $-\infty < x < \infty;\ -\infty < \mu < \infty;\ 0 < \sigma$

$E(x)$ = 未定义

$Var(x)$ = 未定义

拟合 Student t

“拟合 Student *t*”选项拟合位置为 μ、尺度为 σ 且自由度为 ν 的 Student *t* 分布。

pdf: $\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)}\frac{1}{\sqrt{\nu\pi\sigma^2}}\left[1+\frac{(x-\mu)^2}{\nu\sigma^2}\right]^{-\left(\frac{\nu+1}{2}\right)}$ $-\infty < x < \infty;\ -\infty < \mu < \infty;\ 0 < \sigma;\ 1 \leqslant \nu$

拟合 SHASH

“拟合 SHASH”选项拟合 sinh-arcsinh (SHASH) 分布。SHASH 分布基于变换了的正态分布，它将正态分布作为特例包括在内。该分布可以对称也可以非对称。形状由两个形状参数 γ 和 δ 决定。有关 SHASH 分布的详细信息，请参见 Jones and Pewsey (2009)。

pdf: $f(x) = \frac{\delta\cosh(w)}{\sqrt{\sigma^2+(x-\theta)^2}}\phi[\sinh(w)]$ $-\infty < \gamma, x, \theta < \infty;\ 0 < \delta, \sigma$

其中

$\phi(\cdot)$ 是标准正态 pdf

$w = \gamma + \delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)$

- 当 $\gamma = 0$ 且 $\delta = 1$ 时，SHASH 分布等价于位置为 θ 且尺度为 σ 的正态分布。
- 变换 $\sinh(w)$ 服从 $\mu = 0$ 且 $\sigma = 1$ 的正态分布。

拟合指数

指数分布尤其适用于描述随时间随机出现的事件，如生存数据。指数分布可能还适用于对不重叠事件各次出现之间经过的时间进行建模。不重叠事件的示例包括以下几项：用户计算机查询与服务器响应之间的时间、客户先后到达服务台相隔的时间，或是传入交换机的各次呼叫之间的时间。

“指数”分布是双参数 Weibull 在 $\beta = 1$ 且 $\alpha = \sigma$ 时的特例，也是 Gamma 分布在 $\alpha = 1$ 时的特例。

$$\text{pdf:} \quad \frac{1}{\sigma}\exp\left(-\frac{x}{\sigma}\right) \qquad 0 < \sigma; \;\; 0 \leq x$$

$$E(x) = \sigma$$

$$\text{Var}(x) = \sigma^2$$

Devore (1995) 提出指数分布是**无记忆的**。无记忆意味着若您在 t 小时后检查某个部件，而该部件仍然在运转，此时附加寿命时间的分布（该部件一直存活到 t 时刻情形下的附加寿命的条件概率）与原始分布相同。

拟合 Gamma

“拟合 Gamma”选项估计 gamma 分布参数，$\alpha > 0$ 和 $\sigma > 0$。参数 α（在拟合 gamma 报表中称为 alpha）描述形状或曲率。参数 σ（称为 Sigma）是分布的尺度参数。数据必须大于零。

$$\text{pdf:} \quad \frac{1}{\Gamma(\alpha)\sigma^{\alpha}}x^{\alpha-1}\exp(-x/\sigma) \qquad 0 < x; \;\; 0 < \alpha,\sigma$$

$$E(x) = \alpha\sigma$$

$$\text{Var}(x) = \alpha\sigma^2$$

- **标准** gamma 分布的 $\sigma = 1$。Sigma 称为尺度参数，因为 1 之外的值将沿着水平轴延展或收缩分布。
- 当 $\sigma = 2$ 且 $\alpha = \nu/2$ 时呈卡方 $\chi^2_{(\nu)}$ 分布。
- 当 $\alpha = 1$ 时呈指数分布。

当 $\alpha \leq 1$ 时，标准 gamma 密度函数严格递减。当 $\alpha > 1$，密度函数从 0 开始，增加到最大值，然后递减。

拟合对数正态

“对数正态”选项估计双参数对数正态分布的参数 μ（尺度）和 σ（形状）。当且仅当 $X = \ln(Y)$ 服从正态分布时，变量 Y 服从对数正态分布。数据必须大于零。

$$\text{pdf:} \quad \frac{1}{\sigma\sqrt{2\pi}} \frac{\exp\left[\frac{-(\log(x)-\mu)^2}{2\sigma^2}\right]}{x} \qquad 0 \le x; \quad -\infty < \mu < \infty; \quad 0 < \sigma$$

$$E(x) = \exp(\mu + \sigma^2/2)$$

$$\text{Var}(x) = \exp(2(\mu + \sigma^2)) - \exp(2\mu + \sigma^2)$$

拟合 Weibull

Weibull 分布根据 α（尺度）和 β（形状）的值呈不同形状。它通常为估计寿命（特别是机械设备和生物方面）提供合适的模型。

Weibull 分布的 pdf 定义如下：

$$\text{pdf:} \quad \frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} \exp\left[-\left(\frac{x}{\alpha}\right)^{\beta}\right] \qquad \alpha,\beta > 0; \quad 0 < x$$

$$E(x) = \alpha\Gamma\left(1 + \frac{1}{\beta}\right)$$

$$\text{Var}(x) = \alpha^2\left\{\Gamma\left(1 + \frac{2}{\beta}\right) - \Gamma^2\left(1 + \frac{1}{\beta}\right)\right\}$$

其中， $\Gamma(\cdot)$ 是 Gamma 函数。

拟合两正态混合和拟合三正态混合

“拟合两正态混合”和“拟合三正态混合”选项可拟合混合的两个或三个正态分布。这些灵活的分布能够拟合双模态或多模态数据。将为每个分组估计单独的均值、标准差和占整体的比例。在下面的方程中， k 等于混合中的正态分布数目。

$$\text{pdf:} \quad \sum_{i=1}^{k} \frac{\pi_i}{\sigma_i}\phi\left(\frac{x-\mu_i}{\sigma_i}\right)$$

$$E(x) = \sum_{i=1}^{k} \pi_i\mu_i$$

$$\mathrm{Var}(x) = \sum_{i=1}^{k} \pi_i(\mu_i^2 + \sigma_i^2) - \left(\sum_{i=1}^{k} \pi_i \mu_i \right)^2$$

其中， μ_i、 σ_i 和 π_i 分别是第 i 个分组的均值、标准差和比例； $\phi(\cdot)$ 是标准正态 pdf。

拟合 Johnson

“拟合 Johnson”选项用于选择并拟合 Johnson 分布体系中的最佳拟合分布，Johnson 分布体系包含三个分布，这三个分布全部基于变换正态分布。这三种分布如下：

- Johnson Su，无界。
- Johnson Sb，双侧均有界限。这些界限由可以估计的参数定义。
- Johnson Sl，单侧有界限。该界限由可以估计的参数定义。Johnson Sl 系列包含一系列对数正态分布。

仅报告选定分布的拟合。有关 Johnson 分布的选择过程和参数估计的信息，可以在 Slifker 和 Shapiro (1980) 中找到。参数估计不使用最大似然。

Johnson 分布十分灵活，所以很受欢迎。特别是，Johnson 分布体系以其数据拟合能力闻名，因为它支持偏度和峰度的每种可能组合。不过，SHASH 分布也是非常灵活的，相对于 Johnson 分布，优先推荐使用 SHASH 分布。

若 Z 是标准正态变量，则该体系定义如下：

$$Z = \gamma + \delta f(Y)$$

其中，对于 Johnson Su：

$$f(Y) = \ln\left(Y + \sqrt{1 + Y^2}\right) = \sinh^{-1} Y$$

$$Y = \frac{X - \theta}{\sigma} \qquad -\infty < X < \infty$$

其中，对于 Johnson Sb：

$$f(Y) = \ln\left(\frac{Y}{1 - Y}\right)$$

$$Y = \frac{X - \theta}{\sigma} \qquad \theta < X < \theta + \sigma$$

对于 Johnson Sl，其中 $\sigma = \pm 1$。

$$f(Y) = \ln(Y)$$

$$Y = \frac{X - \theta}{\sigma} \qquad \begin{array}{ll} \theta < X < \infty & \text{若} \quad \sigma = 1 \\ -\infty < X < \theta & \text{若} \quad \sigma = -1 \end{array}$$

Johnson Su

pdf: $\frac{\delta}{\sigma}\left[1+\left(\frac{x-\theta}{\sigma}\right)^2\right]^{-1/2}\phi\left[\gamma+\delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)\right]$ $-\infty < x, \theta, \gamma < \infty;\ \ 0 < \theta,\delta$

Johnson Sb

pdf: $\phi\left[\gamma+\delta\ln\left(\frac{x-\theta}{\sigma-(x-\theta)}\right)\right]\left(\frac{\delta\sigma}{(x-\theta)(\sigma-(x-\theta))}\right)$ $\theta < x < \theta+\sigma;\ \ 0 < \sigma$

Johnson Sl

pdf: $\frac{\delta}{|x-\theta|}\phi\left[\gamma+\delta\ln\left(\frac{x-\theta}{\sigma}\right)\right]$ $\theta < x$ 若 $\sigma = 1;\ \ \theta > x$ 若 $\sigma = -1$

其中 $\phi(\cdot)$ 是标准正态 pdf。

拟合 Beta

beta 分布适用于对限制在 0,1 区间内的随机变量的行为建模。例如，比例总是介于 0 和 1 之间。“拟合 Beta”选项估计两个形状参数：$\alpha > 0$ 和 $\beta > 0$。beta 分布的值仅位于区间 0,1 中。

pdf: $\frac{1}{B(\alpha,\beta)\sigma^{\alpha+\beta-1}}x^{\alpha-1}x^{\beta-1}$ $0 < x < 1;\ \ 0 < \sigma,\alpha,\beta$

$E(x) = \sigma\frac{\alpha}{\alpha+\beta}$

$Var(x) = \frac{\sigma^2\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$

其中 $B(\cdot)$ 是 Beta 函数。

全部拟合

在“比较分布”报表中，“分布”列表按 AICc 升序排序。使用复选框可显示或隐藏拟合报表以及叠加选定分布的曲线。

AICc 和 BIC 的公式定义如下：

$AICc = -2\log L + 2k + \frac{2k(k+1)}{n-(k+1)}$

$BIC = -2\log L + k\ln(n)$

其中：

- logL 是对数似然。
- n 是样本大小。

– k 是参数个数。

“AICc 权重”列显示加总为 1 的标准化 AICc 值。AICc 权重可解释为在其中的一个拟合分布成立的前提下，特定分布为真实分布的概率。因此，AICc 权重最接近 1 的分布是更好的拟合。仅使用非缺失 AICc 值计算 AICc 权重：

AICc 权重 = exp[-0.5(AICc-min(AICc))] / sum(exp[-0.5(AICc-min(AICc))])

其中， min(AICc) 是拟合分布中的最小 AICc 值。

有关“比较分布”报表中的测度的详细信息，请参见《拟合线性模型》。

离散拟合分布

本节包含 “离散拟合”菜单中的选项的统计详细信息。

拟合 Poisson

Poisson 分布只有一个尺度参数 $\lambda > 0$。

pmf: $\frac{e^{-\lambda}\lambda^{x}}{x!}$ $0 \le \lambda < \infty;\ x = 0,1,2,...$

$E(x) = \lambda$

$Var(x) = \lambda$

由于 Poisson 分布是离散分布，叠加的曲线是一个阶梯函数，该函数在每个整数处跳跃。

拟合负二项

负二项分布用于对指定失败次数之前的成功次数建模。以下参数化包含均值参数 λ 和散度参数 σ。

pmf: $\frac{\Gamma[x+(1/\sigma)]}{\Gamma[x+1]\Gamma[1/\sigma]}\left[\frac{(\lambda\sigma)^{x}}{(1+\lambda\sigma)^{x+(1/\sigma)}}\right],\ x = 0, 1, 2, \ldots$

$E(x) = \lambda$

$Var(x) = \lambda + \sigma\lambda^{2}$

其中， $\Gamma(\cdot)$ 是 Gamma 函数。

负二项分布与 Gamma Poisson 分布之间的关系

负二项分布等价于 Gamma Poisson 分布。Gamma 分布对于以下情况很有用：数据是多个 Poisson(μ) 分布的组合，每个 Poisson(μ) 分布都具有不同的 μ。

Gamma Poisson 分布的假设前提是：$x|\mu$ 服从 Poisson 分布，μ 服从 Gamma(α,τ) 分布。Gamma Poisson 包含参数 $\lambda = \alpha\tau$ 和 $\sigma = \tau+1$。参数 σ 是一个离散参数。若 $\sigma > 1$，则出现过度离散，这意味着 x 的变异比 Poisson 分布本身所解释的要大。若 $\sigma = 1$，x 将简化为 Poisson(λ)。

pmf: $\frac{\Gamma\left(x+\frac{\lambda}{\sigma-1}\right)}{\Gamma(x+1)\Gamma\left(\frac{\lambda}{\sigma-1}\right)}\left(\frac{\sigma-1}{\sigma}\right)^{x}\sigma^{-\frac{\lambda}{\sigma-1}}$ $0<\lambda;\ 1\leqslant\sigma;\ x=0,1,2,...$

$E(x)=\lambda$

$Var(x)=\lambda\sigma$

其中，$\Gamma(\cdot)$ 是 Gamma 函数。

Gamma Poisson 等价于 $\sigma_{negbin}=(\sigma_{gp}-1)/\lambda_{gp}$ 的负二项分布。

运行 JMP Samples/Scripts 文件夹中的 demoGammaPoisson.jsl，比较带有参数 λ 和 σ 的 Gamma Poisson 分布与带有参数 λ 的 Poisson 分布。

拟合零泛滥 Poisson

零泛滥 (ZI) Poisson 分布的尺度参数 $\lambda>0$，零泛滥参数为 π。

pmf: $\begin{cases}\pi+(1-\pi)\exp[-\lambda]\ ，对于\quad x=0\\(1-\pi)\frac{\lambda^{x}}{x!}\exp[-\lambda],\ 对于\ x=1,2,\ldots\end{cases}$

$E(x)=(1-\pi)\lambda$

$Var(x)=\lambda(1-\pi)(1+\lambda\pi)$

拟合零泛滥负二项

零泛滥 (ZI) Poisson 负二项分布的尺度参数 $\lambda>0$，散度参数 $\sigma>0$，零泛滥参数为 π。

pmf: $\begin{cases}\pi+(1-\pi)(1+\lambda\sigma)^{-(1/\sigma)},\ 对于\ x=0\\(1-\pi)\frac{\Gamma[x+(1/\sigma)]}{\Gamma[x+1]\Gamma[1/\sigma]}\left[\frac{(\lambda\sigma)^{x}}{(1+\lambda\sigma)^{x+(1/\sigma)}}\right]\ ，对于\ x=1,2,\ldots\end{cases}$

$E(x)=(1-\pi)\lambda$

$Var(x)=\lambda(1-\pi)[1+\lambda(\sigma+\pi)]$

拟合二项

“拟合二项”选项接受两种格式的数据：常数样本大小或包含样本大小的列。

pmf: $\binom{n}{x}p^{x}(1-p)^{n-x}$ $0\leq p\leq 1;\ x=0,1,2,...,n$

$E(x)=np$

$\text{Var}(x) = np(1-p)$

其中， n 是独立试验数。

注意：二项参数的置信区间是得分区间。请参见 Agresti and Coull (1998)。

拟合 Beta 二项

beta 二项分布对于以下情况很有用：数据是多个 Binomial(p) 分布的组合，每个 Binomial(p) 分布都具有不同的 p。从多个生产线组合而来的缺陷总数就是这样的例子，此时缺陷数均值 (p) 在不同生产线之间是不同的。

Beta 二项分布的假设前提是：$x|\pi$ 服从 Binomial(n,π) 分布， π 服从 Beta(α,β) 分布。 beta 二项分布具有参数 $p = \alpha/(\alpha+\beta)$ 和 $\delta = 1/(\alpha+\beta+1)$。参数 δ 是一个离散参数。若 $\delta > 0$，则出现过度离散，这意味着 x 的变异比二项分布本身所解释的要大。若 $\delta < 0$，则出现离散不足。若 $\delta = 0$， x 呈 Binomial(n,p) 分布。仅当 $n \geq 2$ 时才存在 beta 二项分布。

$$\text{pmf:} \quad \binom{n}{x} \frac{\Gamma\left(\frac{1}{\delta}-1\right)\Gamma\left[x+p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-x+(1-p)\left(\frac{1}{\delta}-1\right)\right]}{\Gamma\left[p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[(1-p)\left(\frac{1}{\delta}-1\right)\right]\Gamma\left(n+\frac{1}{\delta}-1\right)}$$

$$0 \leq p \leq 1; \quad \max\left(-\frac{p}{n-p-1}, -\frac{1-p}{n-2+p}\right) \leq \delta \leq 1; \quad x = 0,1,2,\ldots,n$$

$E(x) = np$

$\text{Var}(x) = np(1-p)[1+(n-1)\delta]$

其中， $\Gamma(\cdot)$ 是 Gamma 函数。

请记住 $x|\pi \sim$ Binomial(n,π)，同时 $\pi \sim$ Beta(α,β)。参数 $p = \alpha/(\alpha+\beta)$ 和 $\delta = 1/(\alpha+\beta+1)$ 由该平台估计得出。要得到 α 和 β 的估计值，请使用以下公式：

$$\hat{\alpha} = \hat{p}\left(\frac{1-\hat{\delta}}{\hat{\delta}}\right)$$

$$\hat{\beta} = (1-\hat{p})\left(\frac{1-\hat{\delta}}{\hat{\delta}}\right)$$

若 δ 的估计值为 0，则该公式无效。在这种情况下，beta 二项分布简化为 Binomial(n,p) 分布，而且 $\hat{p}$ 是 p 的估计值。

beta 二项参数的置信区间是边侧似然区间。

运行 JMP Samples/Scripts 文件夹中的 demoBetaBinomial.jsl，比较带有离散参数 δ 的 beta 二项分布与带有参数 p 且 n = 20 的二项分布。

旧版分布拟合器详细信息

在 JMP 15 中更新了分布拟合的一些功能。本节详细介绍了以往 JMP 版本中为兼容目的而保留的旧功能。通过在变量的红色小三角菜单中选择**连续拟合 > 启用旧版拟合器**，可以使用这些功能。

- “拟合分布”选项（旧版）”
- “连续拟合分布的统计详细信息（旧版）”
- “离散拟合分布的统计详细信息（旧版）”
- “拟合分位数的统计详细信息（旧版）”
- “拟合分布”选项的统计详细信息（旧版）”

“拟合分布”选项（旧版）

使用连续拟合或离散拟合选项可以拟合连续变量的分布。

注意：在 JMP 15 中更新了分布拟合的一些功能。本节详细介绍了以往 JMP 版本中为兼容目的而保留的旧功能。通过在变量的红色小三角菜单中选择**连续拟合 > 启用旧版拟合器**，可以使用这些功能。

在直方图上叠加曲线，同时向报表窗口添加“参数估计值”报表。红色小三角菜单包含更多选项。请参见第 86 页的““拟合分布”选项（旧版）”。

注意：“寿命分布”平台还包含可能使用不同参数并允许删失的分布拟合选项。请参见《可靠性和生存方法》。

连续拟合（旧版）

本节介绍 “旧版拟合器”子菜单中与更新的 “连续拟合”选项中的相应分布不同的分布。

- “Weibull”分布、“有阈值的 Weibull”分布和“极值”分布通常可为估计寿命（特别是机械设备和生物学方面）提供合适的模型。
- Gamma 分布受零限制并具有灵活形状。
- Beta 分布适用于对限制在 0,1 区间内的随机变量的行为建模。例如，比例总是介于 0 和 1 之间。
- “平滑曲线”分布使用非参数密度估计（核密度估计）拟合平滑曲线。平滑曲线叠加在直方图上，该图下方会出现一个滑块。通过使用滑块更改核心标准差来控制平滑度。初始**核心标准差**估计值基于数据的标准差计算得出。
- Johnson Su、Johnson Sb 和 Johnson Sl 分布具有很强的数据拟合能力，因为它支持偏度和峰度的每种可能组合。
- “广义对数”(Glog) 分布适用于拟合与正态分布相差甚远且经常具有不恒定方差的数据（如生化数据）。

比较所有分布

全部选项用于拟合变量的所有适用连续分布。“比较分布”报表包含有关每个拟合分布的统计量。使用复选框可显示或隐藏拟合报表以及叠加选定分布的曲线。默认情况下选定最佳拟合分布。

“显示分布”列表按 AICc 升序排序。

若变量包含负值，“显示分布”列表则不包含要求正值数据的那些分布。该命令仅拟合连续分布。具有阈值参数（如 Beta 和 Johnson Sb）的分布不包含在可能的分布列表中。

相关信息

有关统计详细信息，请参见以下各节：

- 第 89 页的“连续拟合分布的统计详细信息（旧版）”
- 第 95 页的“拟合分位数的统计详细信息（旧版）”
- 第 86 页的“‘拟合分布’选项（旧版）”

离散拟合（旧版）

当所有数据值都是整数时，“离散拟合”选项可用。使用 “离散拟合”选项可以拟合离散变量的分布 （如 Poisson 或二项分布）。以下分布可用：

- Poisson
- Gamma Poisson
- 二项
- Beta 二项

相关信息

有关统计详细信息，请参见以下各节：

- 第 93 页的“离散拟合分布的统计详细信息（旧版）”
- 第 95 页的“拟合分位数的统计详细信息（旧版）”
- 第 86 页的“‘拟合分布’选项（旧版）”

“拟合分布”选项（旧版）

每个拟合的分布报表都有一个包含更多选项的红色小三角菜单。

诊断图 创建分位数或概率图。请参见第 87 页的 “诊断图”。

密度曲线 使用分布的估计参数在直方图上叠加密度曲线。

拟合优度 计算拟合分布的拟合优度检验。请参见第 88 页的 “拟合优度”。

固定参数 支持您固定参数并重新估计未固定的参数。同时显示“适当性似然比检验”报表，用于检验新参数，以确定这些参数是否能拟合数据。

分位数 为您指定的特定低概率值返回未统一尺度且未中心化的分位数。

针对 K Sigma 设置规格限 若您不了解某个过程的规格限但想要将其分布用作设置规格限的指导，可以使用该选项。

通常，使用工程考虑因素来生成规格限。若不存在工程考虑因素并且数据来自良好的过程，那么拟合分布的分位数通常用于帮助设置规格限。请参见第 96 页的“针对 K Sigma 设置规格限”。

规格限 基于您指定的规格限和目标，计算广义标准能力指标。请参见第 88 页的“规格限”。

保存拟合分位数 将拟合分位数值另存为当前数据表中的新列。请参见第 95 页的“拟合分位数的统计详细信息（旧版）”。

保存密度公式 在当前数据表中创建新列，在其中包含密度公式计算得出的拟合值。密度公式使用估计参数值。

保存规格限 将规格限另存为列属性。

保存变换 创建新列并保存公式。公式可以使用拟合分布对列进行正态变换。仅当拟合 Johnson 分布、 Glog 分布或 SHASH 分布中某一个分布的情况下，该选项才可用。

删除拟合 从报表窗口中删除分布拟合。

诊断图

诊断图选项用于创建分位数或概率图。根据拟合的分布，该图为四种格式之一。

数据 - 拟合分位数

- 有阈值的 Weibull
- Gamma
- Beta
- Poisson
- GammaPoisson
- 二项
- Beta 二项

数据 - 拟合概率

- 正态
- 正态混合
- 指数

对数尺度的数据 - 拟合概率

- Weibull
- 对数正态
- 极值

标准正态分位数 - 拟合概率

- SHASH
- Johnson Sl
- Johnson Sb
- Johnson Su
- 广义对数

“诊断图”红色小三角菜单提供以下选项：

旋转 反转 x 和 y 轴。

置信限 为“正态分位数”图绘制 Lilliefors 95% 置信限，为所有其他分位数图绘制 95% 等精度置信带（$a = 0.001$ 且 $b = 0.99$）(Meeker and Escobar 1998)。

拟合线 绘制对角参考直线。若变量与选定的分布拟合，则各值大致落在该参考线上。

中位数参考线 在响应的中位数处绘制一条水平线。

拟合优度

拟合优度选项计算拟合分布的拟合优度检验。拟合优度检验不是卡方检验，而是 EDF（经验分布函数）检验。 EDF 检验包含改进的功效并且相对于直方图中点值保持不变，所以优于卡方检验。

- 对于正态分布，当样本小于或等于 2000 时，将报告 Shapiro-Wilk 正态性检验。当样本大于 2000 时，将为样本计算 KSL 检验。
- 对于样本小于等于 30 的离散分布，拟合优度检验按以下方式形成：将两个单侧 Kolmogorov 精确检验合并以形成最接近精确的检验。请参见 Conover (1972)。对于大于 30 的样本，将执行 Pearson 卡方拟合优度检验。

相关信息

- 有关统计详细信息，请参见第 86 页的““拟合分布”选项（旧版）”。

规格限

使用**规格限**选项可打开一个支持您输入规格限和目标的窗口。然后计算广义的标准能力指标。请注意，对于正态分布，3σ 既是从下 0.135 百分位数到中位数（或均值）的距离，也是从中位数（或均值）到上 99.865 百分位数的距离。这些百分位数从拟合分布估计得出，在标准公式中用 3σ 来替换适当的百分位数到中位数的距离。

相关信息

- 有关统计详细信息，请参见第 86 页的““拟合分布”选项（旧版）”。

连续拟合分布的统计详细信息（旧版）

本节包含 “连续拟合”菜单中的选项的统计详细信息。

注意：在 JMP 15 中更新了分布拟合的一些功能。本节详细介绍了以往 JMP 版本中为兼容目的而保留的旧功能。通过在变量的红色小三角菜单中选择**连续拟合 > 启用旧版拟合器**，可以使用这些功能。

正态

有关正态分布拟合的详细信息，请参见第 76 页的“拟合正态”。

对数正态

有关正态分布拟合的详细信息，请参见第 79 页的“拟合对数正态”。

Weibull、有阈值的 Weibull 和极值

Weibull 分布根据 α（尺度）和 β（形状）的值呈不同形状。它通常为估计寿命（特别是机械设备和生物方面）提供合适的模型。

“Weibull”分布和 “有阈值的 Weibull”分布的 pdf 定义如下：

$$\text{pdf:}\quad \frac{\beta}{\alpha}\left(\frac{x-\theta}{\alpha}\right)^{\beta-1}\exp\left[-\left(\frac{x-\theta}{\alpha}\right)^{\beta}\right] \qquad \alpha,\beta>0;\ \ \theta<x$$

$$E(x)=\theta+\alpha\Gamma\left(1+\frac{1}{\beta}\right)$$

$$\text{Var}(x)=\alpha^2\left\{\Gamma\left(1+\frac{2}{\beta}\right)-\Gamma^2\left(1+\frac{1}{\beta}\right)\right\}$$

其中， $\Gamma(\cdot)$ 是 Gamma 函数。

Weibull 选项将阈值参数 (θ) 设置为零。**有阈值的 Weibull** 选项，使用最小观测的值估计阈值参数 (θ)，使用其余观测估计 α 和 β。若了解应该如何设置阈值，请使用**固定参数**选项设置该阈值。请参见第 58 页的“拟合分布”选项”。

注意：“分布”平台对于“有阈值的 Weibull”分布中的阈值参数使用与“寿命分布”平台不同的估计方法。推荐使用“寿命分布”估计方法拟合该分布。请参见《可靠性和生存方法》。

“极值”分布等价于重新参数化为 $\delta = 1/\beta$ 且 $\lambda = \ln(\alpha)$ 的双参数 Weibull (α, β) 分布。

指数

有关指数分布拟合的详细信息，请参见第 78 页的“拟合指数”。

Gamma

Gamma 拟合选项估计 gamma 分布参数，$\alpha > 0$ 和 $\sigma > 0$。参数 α（在拟合 gamma 报表中称为 alpha）描述形状或曲率。参数 σ（称为 Sigma）是分布的尺度参数。第三个参数 θ（称为**阈值**）是下端点参数。除非有负值，否则默认情况下该参数设置为零。您还可以通过使用**固定参数**选项设置其值。请参见第 58 页的“拟合分布”选项”。

$$\text{pdf:} \quad \frac{1}{\Gamma(\alpha)\sigma^{\alpha}}(x-\theta)^{\alpha-1}\exp(-(x-\theta)/\sigma) \qquad 0 \leq x;\ \ 0 < \alpha,\sigma$$

$$E(x) = \alpha\sigma + \theta$$

$$\text{Var}(x) = \alpha\sigma^2$$

- **标准** gamma 分布的 $\sigma = 1$。Sigma 称为尺度参数，因为 1 之外的值将沿着水平轴延展或收缩分布。
- 当 $\sigma = 2$、$\alpha = \nu/2$ 且 $\theta = 0$ 时呈卡方 $\chi^2_{(\nu)}$ 分布。
- 指数分布是在 $\alpha = 1$ 且 $\theta = 0$ 时出现的一系列 gamma 曲线。

当 $\alpha \leq 1$ 时，标准 gamma 密度函数严格递减。当 $\alpha > 1$，密度函数从 0 开始，增加到最大值，然后递减。

Beta

标准 Beta 分布适用于对限制在 0,1 区间内的随机变量的行为建模。例如，比例总是介于 0 和 1 之间。Beta 拟合选项估计两个形状参数：$\alpha > 0$ 和 $\beta > 0$，以及两个阈值参数 θ 和 σ。阈值下限表示为 θ，阈值上限表示为 $\theta + \sigma$。beta 分布仅在区间 $\theta \leq x \leq (\theta + \sigma)$ 内有值。θ 用最小值估计，σ 用极差估计。当 $\theta = 0$ 且 $\sigma = 1$ 时呈标准 beta 分布。

使用**固定参数**选项可将参数设置为固定值。阈值上限必须大于等于最大数据值，阈值下限必须小于等于最小数据值。有关“固定参数”选项的详细信息，请参见第 58 页的“拟合分布”选项”。

$$\text{pdf:} \quad \frac{1}{B(\alpha,\beta)\sigma^{\alpha+\beta-1}}(x-\theta)^{\alpha-1}(\theta+\sigma-x)^{\beta-1} \qquad \theta \leq x \leq \theta+\sigma;\ \ 0 < \sigma,\alpha,\beta$$

$$E(x) = \theta + \sigma\frac{\alpha}{\alpha+\beta}$$

$$\text{Var}(x) = \frac{\sigma^2\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

其中 $B(\cdot)$ 是 Beta 函数。

正态混合

有关正态混合分布拟合的详细信息，请参见第 79 页的“拟合两正态混合和拟合三正态混合”。

平滑曲线

平滑曲线选项使用非参数密度估计（核密度估计）拟合平滑曲线。平滑曲线叠加在直方图上，该图下方会出现一个滑块。通过使用滑块更改核心标准差来控制平滑度。初始**核心标准差**估计值基于数据的标准差计算得出。

SHASH

有关 SHASH 分布拟合的详细信息，请参见第 77 页的“拟合 SHASH”。

Johnson Su、Johnson Sb、Johnson Sl

Johnson 分布体系包含三种分布，这些分布全都基于变换的正态分布。这三种分布如下：

- Johnson Su，无界。
- Johnson Sb，双侧均有界限。这些界限由可以估计的参数定义。
- Johnson Sl，单侧有界限。该界限由可以估计的参数定义。Johnson Sl 系列包含一系列对数正态分布。

S 代表体系，即范围的下标。尽管我们在旧版的拟合器中执行不同的方法，但关于特定 Johnson 体系的选择标准的信息在 Slifker and Shapiro (1980) 中提供。

Johnson 分布十分灵活，所以很受欢迎。特别是，Johnson 分布体系以其数据拟合能力闻名，因为它支持偏度和峰度的每种可能组合。

若 Z 是标准正态变量，则该体系定义如下：

$$Z = \gamma + \delta f(Y)$$

其中，对于 Johnson Su：

$$f(Y) = \ln\left(Y + \sqrt{1 + Y^2}\right) = \sinh^{-1} Y$$

$$Y = \frac{X - \theta}{\sigma} \qquad -\infty < X < \infty$$

其中，对于 Johnson Sb：

$$f(Y) = \ln\left(\frac{Y}{1 - Y}\right)$$

$$Y = \frac{X - \theta}{\sigma} \qquad \theta < X < \theta + \sigma$$

对于 Johnson Sl，其中 $\sigma = \pm 1$。

$$f(Y) = \ln(Y)$$

$$Y = \frac{X-\theta}{\sigma} \qquad \begin{array}{ll} \theta < X < \infty & \text{若} \quad \sigma = 1 \\ -\infty < X < \theta & \text{若} \quad \sigma = -1 \end{array}$$

Johnson Su

$$\text{pdf:} \quad \frac{\delta}{\sigma}\left[1+\left(\frac{x-\theta}{\sigma}\right)^2\right]^{-1/2}\phi\left[\gamma+\delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)\right] \qquad -\infty < x, \theta, \gamma < \infty; \quad 0 < \theta, \delta$$

Johnson Sb

$$\text{pdf:} \quad \phi\left[\gamma+\delta\ln\left(\frac{x-\theta}{\sigma-(x-\theta)}\right)\right]\left(\frac{\delta\sigma}{(x-\theta)(\sigma-(x-\theta))}\right) \qquad \theta < x < \theta+\sigma; \quad 0 < \sigma$$

Johnson Sl

$$\text{pdf:} \quad \frac{\delta}{|x-\theta|}\phi\left[\gamma+\delta\ln\left(\frac{x-\theta}{\sigma}\right)\right] \qquad \theta < x \text{ 若 } \sigma = 1; \quad \theta > x \text{ 若 } \sigma = -1$$

其中 $\phi(\cdot)$ 是标准正态 pdf。

请注意以下几点：

- 不同的机器的参数估计值可能由于操作顺序和机器的精确度而有所不同。
- 参数置信区间在默认报表中处于隐藏状态。参数置信区间对 Johnson 分布的意义不大，因为这些分布是变换了的正态分布。要显示参数置信区间，请在报表中右击，然后选择**列** > **95% 下限**和 **95% 上限**。

广义对数 (Glog)

该分布适用于拟合与正态分布相差甚远且经常具有不恒定方差的数据 （如生化数据）。使用参数 μ （位置）、σ （尺度）和 λ （形状）来描述广义对数分布。

$$\text{pdf:} \quad \phi\left\{\frac{1}{\sigma}\left[\log\left(\frac{x+\sqrt{x^2+\lambda^2}}{2}\right)-\mu\right]\right\}\frac{x+\sqrt{x^2+\lambda^2}}{\sigma(x^2+\lambda^2+x\sqrt{x^2+\lambda^2})}$$

$$0 \le \lambda; \quad 0 < \sigma; \quad -\infty < \mu < \infty$$

广义对数分布是变换了的正态分布，来自以下关系：

若 $z = \frac{1}{\sigma}\left[\log\left(\frac{x+\sqrt{x^2+\lambda^2}}{2}\right)-\mu\right] \sim N(0,1)$，则 $x \sim Glog(\mu,\sigma,\lambda)$。

当 $\lambda = 0$ 时，Glog 简化为 LogNormal ((μ,σ)。

注意：参数置信区间在默认报表中处于隐藏状态。参数置信区间对广义对数分布的意义不大，因为该分布是变换了的正态分布。要显示参数置信区间，请在报表中右击，然后选择**列 > 95% 下限**和 **95% 上限**。

全部

在“比较分布”报表中，“分布”列表按 AICc 升序排序。

按以下方式定义 AICc：

$$AICc = -2\log L + 2\nu + \frac{2\nu(\nu+1)}{n-(\nu+1)}$$

其中：

- logL 是对数似然。
- n 是样本大小。
- ν 是参数个数。

若列包含负值，“分布”列表则不包含要求正值数据的那些分布。仅列出连续分布。具有阈值参数（如 Beta 和 Johnson Sb）的分布不包含在可能的分布列表中。

离散拟合分布的统计详细信息（旧版）

本节包含 “离散拟合”菜单中的选项的统计详细信息。

注意：在 JMP 15 中更新了分布拟合的一些功能。本节详细介绍了以往 JMP 版本中为兼容目的而保留的旧功能。通过在变量的红色小三角菜单中选择**连续拟合 > 启用旧版拟合器**，可以使用这些功能。

Poisson

有关 Poisson 分布拟合的详细信息，请参见第 82 页的“拟合 Poisson”。

Gamma Poisson

该分布对于以下情况很有用：数据是多个 Poisson(μ) 分布的组合，每个 Poisson(μ) 分布都具有不同的 μ。从多个十字路口组合而来的事故总数就是这样的例子，此时事故数均值 (μ) 在不同十字路口之间是不同的。

Gamma Poisson 分布的假设前提是：x|μ 服从 Poisson 分布，μ 服从 Gamma(α,τ) 分布。Gamma Poisson 包含参数 λ = ατ 和 σ = τ+1。参数 σ 是一个离散参数。若 σ > 1，则出现过度离散，这意味着 x 的变异比 Poisson 分布本身所解释的要大。若 σ = 1，x 将简化为 Poisson(λ)。

$$\text{pmf:} \quad \frac{\Gamma\left(x+\frac{\lambda}{\sigma-1}\right)}{\Gamma(x+1)\Gamma\left(\frac{\lambda}{\sigma-1}\right)}\left(\frac{\sigma-1}{\sigma}\right)^{x}\sigma^{-\frac{\lambda}{\sigma-1}} \qquad 0<\lambda;\ \ 1\leqslant\sigma;\ \ x=0,1,2,\ldots$$

$E(x) = \lambda$

$Var(x) = \lambda\sigma$

其中，Γ(·) 是 Gamma 函数。

请记住 $x|\mu \sim$ Poisson(μ)，同时 μ~ Gamma(α,τ)。该平台估计 λ = ατ 和 σ = τ+1。要得到 α 和 τ 的估计值，请使用以下公式：

$$\hat{\tau} = \hat{\sigma} - 1$$

$$\hat{\alpha} = \frac{\hat{\lambda}}{\hat{\tau}}$$

若 σ 的估计值为 1，则该公式无效。在这种情况下，Gamma Poisson 简化为 Poisson(λ)，而且 $\hat{\lambda}$ 是 λ 的估计值。

若 α 的估计值为整数，则 Gamma Poisson 等价于负二项分布，pmf 如下：

$$p(y) = \binom{y+r-1}{y}p^{r}(1-p)^{y} \qquad \text{对于} \quad 0 \leqslant y$$

r = α，$(1\text{-}p)/p = \tau$。

运行 JMP Samples/Scripts 文件夹中的 demoGammaPoisson.jsl，比较带有参数 λ 和 σ 的 Gamma Poisson 分布与带有参数 λ 的 Poisson 分布。

二项

有关二项分布拟合的详细信息，请参见第 83 页的“拟合二项”。

Beta 二项

有关 beta 二项分布拟合的详细信息，请参见第 84 页的“拟合 Beta 二项”。

拟合分位数的统计详细信息（旧版）

注意：在 JMP 15 中更新了分布拟合的一些功能。本节详细介绍了以往 JMP 版本中为兼容目的而保留的旧功能。通过在变量的红色小三角菜单中选择**连续拟合 > 启用旧版拟合器**，可以使用这些功能。

诊断图中的拟合分位数和使用**保存拟合分位数**命令保存的拟合分位数是使用以下方法形成的：

1. 数据经过排序和排名。为结值指定了不同的秩。
2. 计算 $p_{[i]} = rank_{[i]}/(n+1)$。
3. 计算 $quantile_{[i]} = Quantile_d(p_{[i]})$，其中 $Quantile_d$ 是特定拟合分布的分位数函数，并且 $i = 1,2,...,n$。

“拟合分布”选项的统计详细信息（旧版）

本节介绍拟合分布的拟合优度检验以及与拟合分布相关的规格限的统计详细信息。

注意：在 JMP 15 中更新了分布拟合的一些功能。本节详细介绍了以往 JMP 版本中为兼容目的而保留的旧功能。通过在变量的红色小三角菜单中选择**连续拟合 > 启用旧版拟合器**，可以使用这些功能。

拟合优度

表 3.2 JMP 拟合优度检验的说明

分布	参数	拟合优度检验
正态[a]	μ 和 σ 未知	Shapiro-Wilk（适用于 $n \leq 2000$） Kolmogorov-Smirnov-Lillefors（适用于 $n > 2000$）
	μ 和 σ 均已知	Kolmogorov-Smirnov-Lillefors
	μ 或 σ 已知	（无）
对数正态	μ 和 σ 已知或未知	Kolmogorov *D*
Weibull	α 和 β 已知或未知	Cramér-von Mises W^2
有阈值的 Weibull	α、β 和 θ 已知或未知	Cramér-von Mises W^2
极值	α 和 β 已知或未知	Cramér-von Mises W^2
指数	σ 已知或未知	Kolmogorov *D*

表 3.2 JMP 拟合优度检验的说明 (续)

分布	参数	拟合优度检验
Gamma	α 和 σ 已知	Cramér-von Mises W^2
	α 或 σ 未知	（无）
Beta	α 和 β 已知	Kolmogorov *D*
	α 或 β 未知	（无）
二项	ρ 已知或未知，*n* 已知	Kolmogorov *D*（适用于 $n \leq 30$）；Pearson χ^2（适用于 $n > 30$）
Beta 二项	ρ 和 δ 已知或未知	Kolmogorov *D*（适用于 $n \leq 30$）；Pearson χ^2（适用于 $n > 30$）
Poisson	λ 已知或未知	Kolmogorov *D*（适用于 $n \leq 30$）；Pearson χ^2（适用于 $n > 30$）
Gamma Poisson	λ 或 σ 已知或未知	Kolmogorov *D*（适用于 $n \leq 30$）；Pearson χ^2（适用于 $n > 30$）

a. 对于三个 Johnson 分布和广义对数分布，数据将变换为正态分布，然后执行适用的正态分布检验。

针对 K Sigma 设置规格限

键入 K 值，并选择单侧或双侧过程能力分析。与 K 标准差对应的尾概率从正态分布计算得出。概率转换为拟合的特定分布的分位数。生成的分位数用于过程能力分析中的规格限。该选项类似于**分位数**选项，但您提供的是 K 而不是概率。K 对应于规格限远离均值的标准差个数。

例如，对于正态分布，若 K=3，则均值上下 3 个标准差分别对应于第 0.00135 分位数和第 0.99865 分位数。下规格限设置在拟合的分布中的第 0.00135 分位数，上规格限设置在第 0.99865 分位数。基于以上规格限返回过程能力分析。

第 4 章

“以 X 拟合 Y”介绍

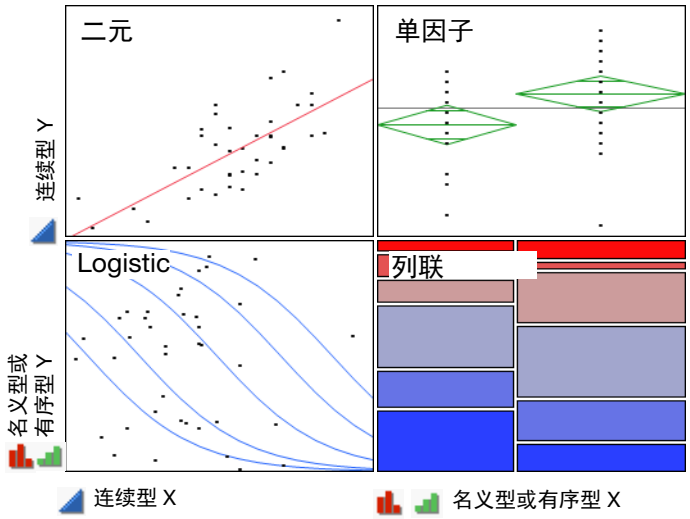
检查两个变量之间的关系

使用“以 X 拟合 Y”平台分析成对变量。您可以使用散点图、线性回归、方差分析、多重比较、Logistic 回归、列联表等多种方式进行分析。具体分析取决于变量的建模类型。

“以 X 拟合 Y”启动四个平台之一：

- 使用“二元”可分析连续型 Y 和连续型 X
- 使用单因子分析（包括方差分析）可分析连续型 Y 和分类型 X
- 使用 Logistic 回归可分析分类型 Y 和连续型 X
- 使用“列联”可分析分类型 Y 和分类型 X

图 4.1 四种类型分析的示例



目录

“以 X 拟合 Y”平台概述 99
启动“以 X 拟合 Y”平台 99
从“JMP 起始页”窗口启动特定分析 100

“以 X 拟合 Y”平台概述

“以 X 拟合 Y”平台是四个特定平台 （或分析类型）的集合。

特定平台	建模类型	说明
二元	连续型 X - 连续型 Y	分析两个连续变量之间的关系。请参见“二元分析”。
单因子	名义型或有序型 X - 连续型 Y	分析在分类 X 变量定义的不同分组间连续型 Y 变量的分布状况。请参见“单因子分析”。
Logistic	连续型 X - 名义型或有序型 Y	拟合响应类别对连续型 X 预测变量的概率。请参见“Logistic 分析”。
列联	名义型或有序型 X - 名义型或有序型 Y	分析在分类 X 因子值的条件下分类响应变量 Y 的分布。请参见“列联分析”。

启动“以 X 拟合 Y”平台

通过选择**分析 > 以 X 拟合 Y** 可以启动以 X 拟合 Y 平台。

图 4.2 “以 X 拟合 Y”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

二元、单因子、 Logistic、列联 该网格显示不同的数据类型组合所生成的分析。一旦指定了列，适用的平台即作为标签显示在该网格的上方。

区组 （可选。仅适用于单因子和列联。）：

– 对于“单因子”平台，指定“区组”变量将标识另一个因子，形成无交互作用的双因子分析。在单因子图中，Y 变量的值以“区组”变量为中心。若按分组单元格划分的每个区组中的计数相等，则可用的“单因子”平台选项将适用于双因子分析设置。若按分组单元格划分的区组中的计数不相等，则“拟合模型”报表将显示在单因子图下方，并且“单因子”平台选项不可用。
– 对于“列联”平台，指定“区组”变量将标识另一个因子并执行 Cochran-Mantel-Haenszel 检验。

有关启动窗口的详细信息，请参见《使用 JMP》。

从“JMP 起始页”窗口启动特定分析

从“JMP 起始页”窗口中，您可以启动特定分析（**二元**、**单因子**、Logistic 或**列联**）。若选择该选项，请为分析指定正确的建模类型（Y 和 X 变量）（表 4.1）。

要从 “JMP 起始页”窗口启动特定分析，请点击**基本**类别，然后选择某一特定分析。

大多数平台启动选项都是相同的。但是，某些 Y 和 X 平台按钮的命名针对您所执行的特定分析进行了特别改动。

表 4.1 平台和按钮

平台或分析	Y 按钮	X 按钮
以 X 拟合 Y	Y，响应	X，因子
二元	Y，响应	X，回归变量
单因子	Y，响应	X，分组
Logistic	Y 轴，分类响应	X 轴，连续回归变量
列联	Y，响应类别	X，分组类别

第 5 章

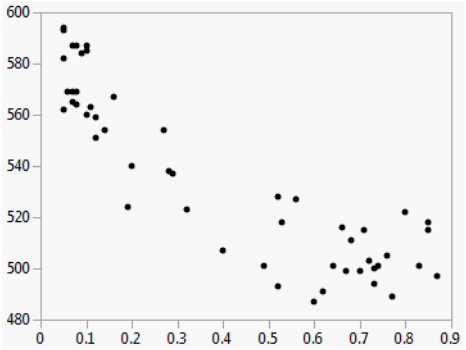
二元分析

检查两个连续变量之间的关系

使用“二元”平台可调查两个连续变量之间的关系。以交互方式向散点图添加简单的线性回归线、多项式回归曲线或平滑线。使用散点图，您可以快速了解两个变量之间的关系。

“二元”平台是“以 X 拟合 Y”平台的**以连续变量拟合连续变量**特质。“二元”一词仅表示涉及两个变量而非一个（一元）或多个变量（多元）。

图 5.1 二元分析的示例



目录

二元分析的示例 104
启动“二元”平台 104
二元图 105
拟合选项 106
拟合选项 107
拟合选项类别 108
多次拟合同一选项 108
直方图边框 109
拟合均值 110
“拟合均值”报表 110
拟合线和拟合多项式 111
“线性拟合”报表和“多项式拟合”报表 111
特殊拟合 116
“特殊拟合”报表和菜单 117
灵活 117
拟合样条 118
核平滑法 118
拟合每个值 119
拟合正交 120
正交拟合比报表 120
稳健 121
稳健拟合 121
拟合 Cauchy 121
密度椭圆 122
“相关性”报表 122
非参数密度 123
“分位数密度等高线”报表 123
依据 ... 分组 124
拟合菜单 124
拟合菜单选项 125
诊断图 127
“二元”平台的更多示例 128
“特殊拟合”选项的示例 128
“拟合正交”选项的示例 130
“稳健拟合”命令的示例 132
使用密度椭圆的“依据 ... 分组”的示例 134
使用回归线的“依据 ... 分组”的示例 135

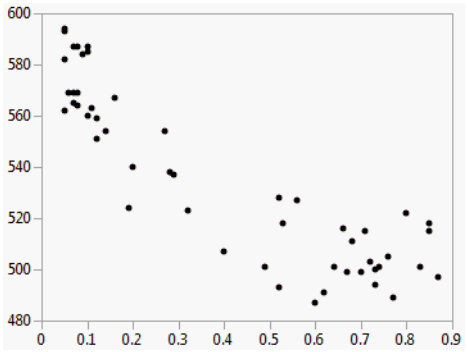
使用“依据”变量进行分组的示例 136
“二元”平台的统计详细信息 138
拟合线 138
拟合样条 138
拟合正交 138
“拟合汇总”报表 139
“失拟”报表 140
“参数估计值”报表 140
“平滑拟合”报表 141
“相关性”报表 141

二元分析的示例

了解如何为两个连续变量构造散点图。该示例使用 SAT.jmp 样本数据表。美国 50 个州以及哥伦比亚特区的学生的 SAT 考试分数分为两个方面：语文和数学。您想要明确参加 SAT 考试的学生的百分比与 2004 年的语文考试分数有何关联。

1. 选择**帮助** > **样本数据库**，然后打开 SAT.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择 **2004 语文考试**并点击 **Y，响应**。
4. 选择**高考参考率 (2004)** 并点击 **X，因子**。
5. 点击**确定**。

图 5.2 以高考参考率拟合 SAT 分数的示例



您可以看到参考人数百分比较低时语文分数较高。

启动“二元”平台

要调查二元分析中两个连续变量之间的关系，请执行以下操作：

1. 选择**分析** > **以 X 拟合 Y**。
2. 为 **Y，响应**输入连续列。
3. 为 **X，因子**输入连续列。

图 5.3 “二元”启动窗口

“二元”一词出现在图的上方，指示您正在执行二元分析。

注意：您还可以从“JMP 起始页”窗口启动二元分析。选择**视图 > JMP 起始页 > 基本 > 二元**。

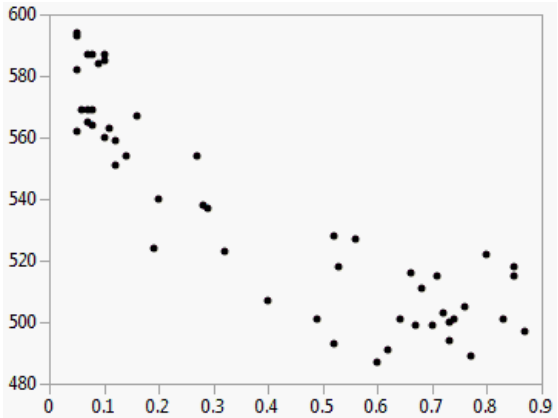
有关该启动窗口的详细信息，请参见第 97 页的““以 X 拟合 Y”介绍”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

二元图

二元报表最初会为每对 *X* 和 *Y* 变量生成一个散点图。

注意：要生成图 5.4 中所示的图，请按第 104 页的“二元分析的示例”中的说明操作。

图 5.4 二元图



通过以下两种方式之一拖放变量，以此替换图中变量：将一个轴上的变量拖放到另一个轴上来交换现有变量；或者，点击相关数据表的列面板中的某个变量并将其拖到轴上。

您可以像与其他 JMP 图交互那样与该图交互（例如，调整图的大小、用箭头工具或笔刷工具突出显示数据点，以及为数据点添加标签）。有关这些特性的详细信息，请参见《使用 JMP》。

您可以在该图上拟合曲线，并使用红色小三角菜单上的拟合选项查看统计报表和附加菜单。请参见第 106 页的“拟合选项”。

拟合选项

“二元拟合”红色小三角菜单包含显示选项、拟合选项和控制选项。

注意：若您指定了多个 Y 或多个 X 变量，将显示“拟合组”菜单。使用菜单选项，您可以排列报表或按 R 方将它们排序。请参见《拟合线性模型》。

显示点 隐藏或显示散点图中的点。复选标记指示显示这些点。

直方图边框 向散点图的 x 和 y 轴附加直方图。复选标记指示已启用直方图边框。请参见第 109 页的“直方图边框”。

注意：若只将“隐藏”行状态应用到数据表中的行，相应的点不会显示在散点图中，但会使用隐藏行构造直方图。若想要从直方图构造和分析结果中排除行，请应用“排除”行状态，然后从“二元”红色小三角菜单中选择**重新运行 > 重新运行分析**。

汇总统计量 显示图的汇总统计量，如：相关性和置信区间、均值和标准差。

依据 ... 分组 支持您选择分类（或分组）变量。为分组变量的每个水平计算单独的分析，并在散点图中叠加回归曲线或椭圆。请参见第 124 页的“依据 ... 分组”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

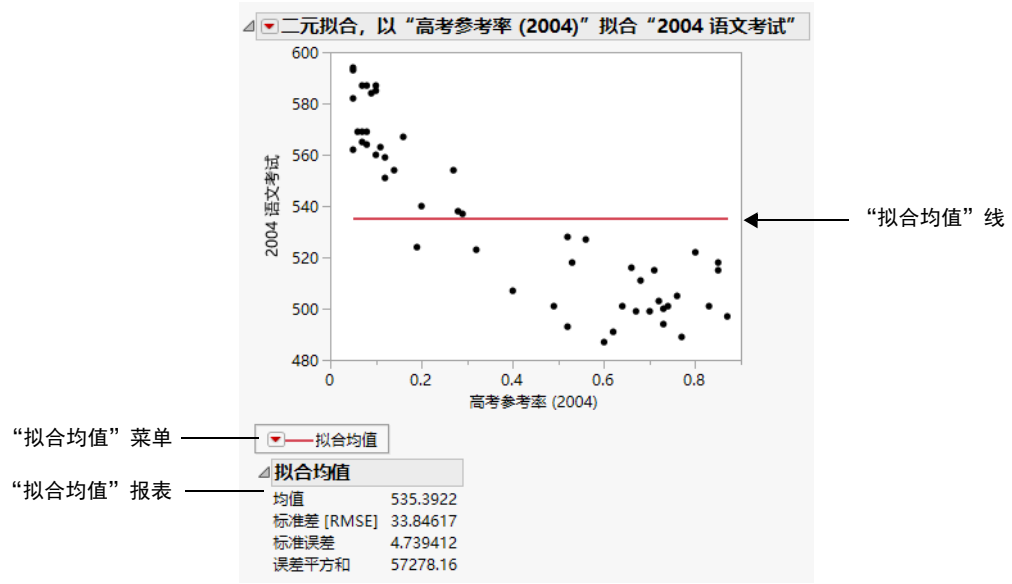
保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

拟合选项

每个拟合选项添加以下项：

- 向散点图添加直线、曲线或分布
- 向报表窗口添加红色小三角菜单
- 向报表窗口添加特定报表

图 5.5 “拟合均值”选项的示例



提供以下拟合选项：

拟合均值 向散点图添加一条水平线，用来表示 Y 响应变量的均值。请参见第 110 页的 “拟合均值”。

拟合线 向散点图添加使用最小二乘回归拟合的直线。请参见第 111 页的 “拟合线和拟合多项式”。

拟合多项式 使用最小二乘回归拟合特定次数的多项式曲线。请参见第 111 页的 “拟合线和拟合多项式”。

特殊拟合 变换 *Y* 和 *X*。变换包括：对数、平方根、平方、倒数和指数。您还可以禁用中心多项式、约束截距和斜率以及拟合多项式模型。请参见第 116 页的 “特殊拟合”。

灵活 提供选项以支持您控制估计回归曲线的平滑度。请参见第 117 页的 “灵活”。

拟合正交 提供用于正交回归拟合的选项，这在假定 X 有变异的情况下会很有用。该选项提供子选项以反映关于 X 和 Y 的方差的各种假设。请参见第 120 页的 “拟合正交”。

稳健 提供选项以降低数据集中的离群值对拟合模型的影响。请参见第 121 页的 “稳健”。

密度椭圆 针对 *X* 和 *Y* 变量的二元正态分布拟合标绘密度椭圆。请参见第 122 页的“密度椭圆”。

非参数密度 基于平滑曲面标绘密度等高线。这些等高线描绘数据点的密度。请参见第 123 页的 “非参数密度”。

注意：您可以使用**删除拟合**选项删除拟合。请参见第 125 页的“拟合菜单选项”。

拟合选项类别

拟合选项包括回归拟合和密度估计这两个类别。

类别	说明	拟合选项
回归拟合	回归方法对观测数据点拟合一条曲线。拟合方法包括最小二乘拟合以及样条拟合、核平滑、正交拟合和稳健拟合。	**拟合均值** **拟合线** **拟合多项式** **特殊拟合** **灵活** **拟合正交** **稳健**
密度估计	密度估计拟合点的二元分布。您可以选择二元正态密度（特有椭圆形等高线）或选择一般的非参数密度。	**密度椭圆** **非参数密度**

多次拟合同一选项

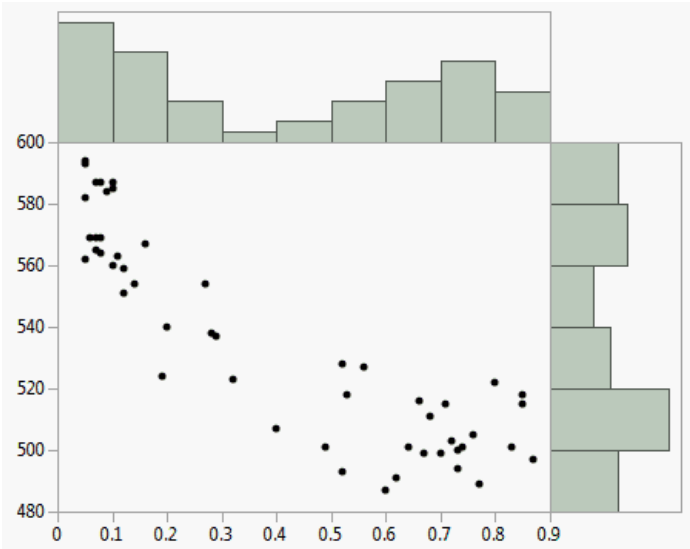
您可以多次选择同一拟合选项，每个新拟合都叠加在散点图上。您可以尝试拟合、排除点后再重新拟合，也可以在同一散点图上比较拟合。

要对报表窗口中的多个分析应用拟合选项，请按住 Ctrl 键并选择某个拟合选项。

直方图边框

直方图边框选项可向散点图的 x 和 y 轴附加直方图。您可以使用直方图直观查看 X 和 Y 变量的边缘分布。

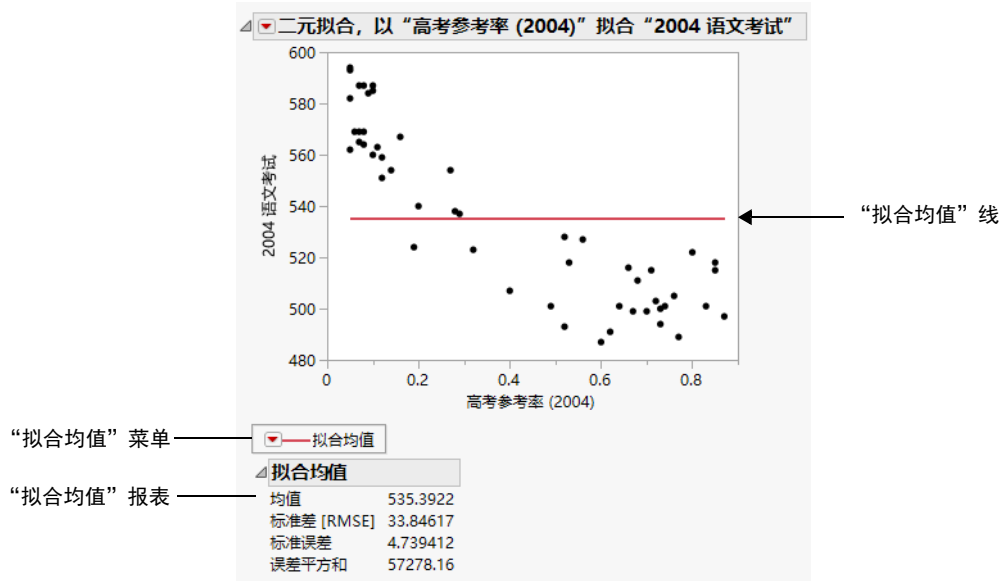
图 5.6 直方图边框的示例



拟合均值

使用**拟合均值**选项，您可以向散点图添加一条水平线，用来表示 Y 响应变量的均值。您可以先拟合均值，然后将均值线用作其他拟合 （如直线、置信曲线、多项式曲线等）的参考。

图 5.7 拟合均值的示例



“拟合均值”报表

“拟合均值”报表显示有关均值拟合的汇总统计量。

均值 响应变量的均值。若模型中没有指定的效应，则为预测响应。

标准差 [RMSE] 响应变量的标准差。均方误差的平方根，亦称均方根误差 （或 RMSE）。

标准误差 响应均值的标准差。由 RMSE 除以数值个数的平方根计算得出。

误差平方和 简单均值模型的误差平方和。在每个模型拟合的方差分析表中显示为**误差**的平方和。

有关“拟合均值”菜单的选项的详细信息，请参见第 124 页的“拟合菜单”。

拟合线和拟合多项式

使用**拟合线**选项，您可以向散点图添加使用最小二乘回归拟合的直线。使用**拟合多项式**选项，您可以使用最小二乘回归拟合特定次数的多项式曲线。

图 5.8 拟合线和拟合多项式的示例

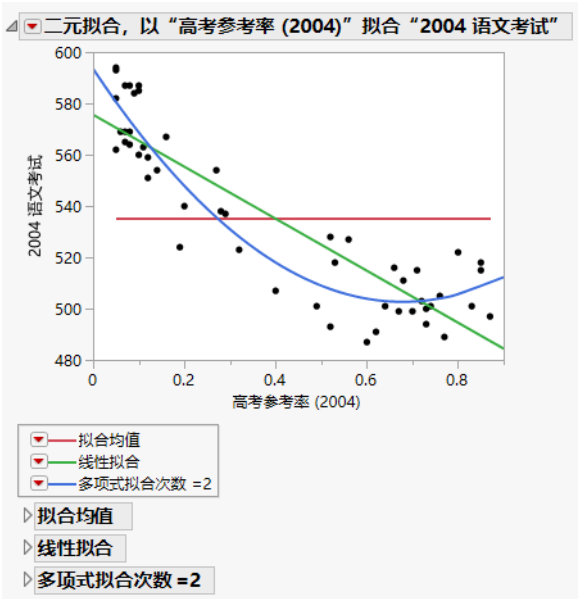


图 5.8 显示了一个示例，其中将线性拟合与均值线和二次多项式拟合进行了比较。

请注意下列信息：

- **拟合线**输出相当于次数为 1 的多项式拟合。
- **拟合均值**输出相当于次数为 0 的多项式拟合。

有关“线性拟合”和“多项式拟合次数”菜单中的选项的详细信息，请参见第 124 页的“拟合菜单”。有关该拟合的统计详细信息，请参见第 138 页的“拟合线”。

“线性拟合”报表和“多项式拟合”报表

“线性拟合”报表和“多项式拟合”报表最先显示的都是拟合方程。

图 5.9 拟合方程的示例

提示：您可以通过点击该方程对其进行编辑。

每个线性和多项式拟合次数报表都至少包含三个报表。第四个报表（失拟）仅在数据中有多个相同的 *X* 值时才显示。

“拟合汇总”报表

“二元”平台中的“拟合汇总”报表显示针对相同数据的线性拟合和二次多项式拟合的响应的数值汇总。您可以比较多个“拟合汇总”报表，通过观察 R 方值是否增大，**均方根误差**是否减小来判断不同模型之间是否有改进。

图 5.10 线性拟合和多项式拟合的 “拟合汇总”报表

“拟合汇总”报表包含以下列：

R 方 测量模型解释的变异的比例。剩余变异不通过模型解释，而是归因于随机误差。若模型完全拟合，则 R 方为 1。

注意：较低的 R 方值表明可能有一些模型中不包含的变量可以解释无法解释的变异。不过，若数据受大量的内在变异影响，即便是有用的回归模型也可能具有较低的 R 方值。阅读您的研究领域中的文献材料，了解有关典型 R 方值的信息。

图 5.10 中的 **R 方**值指示二次多项式拟合相对于线性拟合稍有改进。请参见第 139 页的“ “拟合汇总”报表”。

调整 R 方 通过在 R 方计算中使用自由度，调整 R 方值以使得它可以用来比较具有不同参数个数的模型。请参见第 139 页的 “ “拟合汇总”报表”。

均方根误差 估计随机误差的标准差。它是“方差分析”报表中的**误差**均方的平方根（图 5.12）。

响应均值 提供响应变量的样本均值（算术平均值）。若未指定模型效应，则为预测响应。

观测数 提供用于估计拟合的观测数。若有权重变量，则为权重和。

“失拟”报表

注意：仅当多行具有相同的 *x* 值时才会显示“失拟”报表。

使用“失拟”报表，无论您的模型形式是否正确，您都可以估计误差。当多个观测具有相同的 *x* 值时就会出现这种误差。为这些精确重复测量的误差称为**纯误差**。这是样本误差中无论使用哪种形式的模型都无法解释或预测的部分。不过，若失拟检验中的自由度不高（重复的 *x* 值不多），则失拟检验的用处不大。

图 5.11 线性拟合和多项式拟合的 “失拟”报表的示例

线性拟合

2004 语文考试 = 575.62539 - 101.32814*高考参考率 (2004)

▷ **拟合汇总**

◿ **失拟**

源	自由度	平方和	均方	F 比
失拟	36	9662.983	268.416	1.4850
纯误差	13	2349.750	180.750	概率>F
总误差	49	12012.733		0.2252
				最大 R 方
				0.9590

多项式拟合次数 =2

2004 语文考试 = 562.47113 - 111.07875*高考参考率 (2004) + 196.95725*(高考参考率 (2004)-0.39706)^2

▷ **拟合汇总**

◿ **失拟**

源	自由度	平方和	均方	F 比
失拟	35	4557.2473	130.207	0.7204
纯误差	13	2349.7500	180.750	概率>F
总误差	48	6906.9973		0.7862
				最大 R 方
				0.9590

模型剩余误差与纯误差之间的差值称为**失拟误差**。若回归变量的函数形式有误，则失拟误差可能显著大于纯误差。这种情况下，应该尝试不同类型的模型拟合。“失拟”报表检验失拟误差是否为零。

“失拟”报表包含以下列：

源 变异的三个来源：**失拟**、**纯误差**和**总误差**。

自由度 每种误差来源的**自由度** (DF)。

- **总误差**自由度是“方差分析”表（显示在第 114 页的““方差分析”报表”之下）的**误差**行中的自由度。该值是该表中的**校正总和**自由度与**模型**自由度之间的差值。**误差**自由度又分为失拟自由度和纯误差自由度。
- **纯误差**自由度是各个分组的合并结果，这些分组中有多个行对每个效应都包含相同的值。请参见第 140 页的““失拟”报表”。
- **失拟**自由度是**总误差**自由度与**纯误差**自由度之间的差值。

平方和 每种误差来源的平方和（简写为 SS）。

- **总误差**平方和是相应的“方差分析”表（显示在第 114 页的““方差分析”报表”之下）的**误差**行中的平方和。

- **纯误差**平方和是各个分组的合并结果，这些分组中有多个行对 *x* 变量都包含相同的值。该统计量用来估计模型 *x* 效应无法解释的真正随机误差部分。请参见第 140 页的“ “失拟”报表”。
- **失拟**平方和是**总误差**平方和与**纯误差**平方和之间的差值。若失拟平方和较大，则模型可能不适用于数据。下文所述的 *F* 比检验失拟导致的变异是否足够小，小到为纯误差的一个可忽略比例而被接受。

均方 平方和除以与其相关的自由度。该计算将平方和转换为平均值（均方）。用于统计检验的 *F* 比是均方之比。

F 比 失拟的均方与纯误差的均方之比。它检验失拟误差是否为零这一假设。

概率 > F 当因失拟方差引起的变异与因纯误差方差引起的变异相同时，得到更大的 *F* 值的概率。较高的 *p* 值意味着失拟不显著。

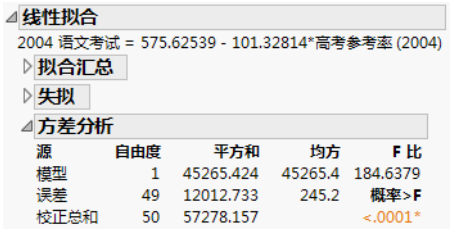
最大 R 方 某个模型仅使用该模型所含变量可以实现的最大 R^2。请参见第 140 页的 “ “失拟”报表”。

“方差分析”报表

回归的方差分析 (ANOVA) 将样本总变异分为多个分量。这些分量用于计算 *F* 比，以便评估模型的有效性。若与 *F* 比关联的概率较小，则认为相对于响应均值本身，模型是更适合数据的统计拟合。

图 5.12 中的“方差分析”报表对线性拟合（**拟合线**）和二次拟合（**拟合多项式**）进行了比较。这两种拟合都是比均值水平线更适合的统计拟合。

图 5.12 线性拟合和多项式拟合的 “方差分析”报表的示例


线性拟合
2004 语文考试 = 575.62539 - 101.32814*高考参考率 (2004)
拟合汇总
失拟
方差分析

源	自由度	平方和	均方	F 比
模型	1	45265.424	45265.4	184.6379
误差	49	12012.733	245.2	概率>F
校正总和	50	57278.157		<.0001*

多项式拟合次数 =2
2004 语文考试 = 562.47113 - 111.07875*高考参考率 (2004) + 196.95725*(高考参考率 (2004)-0.39706)^2
拟合汇总
失拟
方差分析

源	自由度	平方和	均方	F 比
模型	2	50371.160	25185.6	175.0265
误差	48	6906.997	143.9	概率>F
校正总和	50	57278.157		<.0001*

“方差分析”报表包含以下列：

源 变异的三个来源：**模型**、**误差**和**校正总和**。

自由度 每种变异来源的自由度 (DF)：

- 针对计算中使用的每个参数估计值，都会从非缺失值总数 (*N*) 中减去一个自由度。总样本变异的计算使用均值的估计值。因此，需要从总数中减去一个自由度，得到 50。总校正自由度分为**模型**和**误差**自由度。

- 总和中的一个自由度（显示在**模型**行上）用于估计线性拟合的单个回归参数（斜率）。两个自由度用于估计二次多项式拟合的参数（β_1 和 β_2）。
- **误差**自由度是**校正总和**自由度与**模型**自由度之间的差值。

平方和 每种变异来源的平方和 （简写为 SS）：

- 在本例中，每个响应与样本均值之间距离的总平方和（**校正总和**）为 57,278.157，如图 5.12 所示。这是用于与其他所有模型进行比较的基本模型（或简单均值模型）的平方和。
- 对于线性回归，每个点与拟合线之间距离的平方和减至 12,012.733。这是在拟合模型后剩余或未解释的（**误差**）平方和。二次多项式拟合的剩余平方和为 6,906.997，该拟合比线性拟合所解释的变异略多一些。也就是说，该模型可解释更多变异，因为二次多项式的模型平方和比线性拟合的模型平方和要高。**校正总和**平方和减去**误差**平方和之后的值就是模型平方和。

均方 平方和除以与其相关的自由度。用于统计检验的 *F* 比是以下均方的比值：

- 线性拟合的**模型**均方为 45,265.4。该值估计误差方差，但仅是在模型参数为零的假设前提之下。
- **误差**均方为 245.2。该值估计误差方差。

F 比 模型均方除以误差均方。该拟合的假设前提是所有回归参数 （截距除外）都为零。若该假设成立，则误差均方和模型均方都估计误差方差，并且它们的比值服从 *F* 分布。

概率 > F 得到更大的 *F* 值观测到的显著性概率 （*p* 值）（若指定的模型没有总体响应均值的拟合效果好）。观测到的显著性概率为 0.05 或更小通常被视为回归效应的证明。

“参数估计值”报表

对于线性拟合，“参数估计值”报表中的项包括截距和单个 *x* 变量。

对于 *k* 次多项式拟合，“参数估计值”报表包含模型截距的估计值，以及针对 *X* 变量所有 0 到 *k* 次幂的参数估计值。

图 5.13 线性拟合和多项式拟合的 “参数估计值”报表的示例

线性拟合

2004 语文考试 = 575.62539 - 101.32814*高考参考率 (2004)

▷ 拟合汇总
▷ 失拟
▷ 方差分析
◿ 参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	575.62539	3.684288	156.24	<.0001*
高考参考率 (2004)	-101.3281	7.457094	-13.59	<.0001*

多项式拟合次数 =2

2004 语文考试 = 562.47113 - 111.07875*高考参考率 (2004) + 196.95725*(高考参考率 (2004)-0.39706)^2

▷ 拟合汇总
▷ 失拟
▷ 方差分析
◿ 参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	562.47113	3.583843	156.95	<.0001*
高考参考率 (2004)	-111.0788	5.942967	-18.69	<.0001*
(高考参考率 (2004)-0.39706)^2	196.95725	33.06487	5.96	<.0001*

“参数估计值”报表包含以下列：

项 列出所请求的模型中每个参数的名称。截距是所有模型中的常数项。

估计值 列出线性模型的参数估计值。预测公式是这些估计值与其对应变量的值的线性组合。

标准误差 列出参数估计值的标准误差的估计值。这些值用于构成检验和置信区间。

t 比 列出每个参数都为零的假设前提下的检验统计量。这是参数估计值与其标准误差的比值。若假设成立，则该统计量具备 Student *t* 分布。

概率 >|t| 列出从每个 *t* 比计算得出观测到的显著性概率。它是得到 *t* 比的绝对值大于计算值的概率（在原假设成立的前提下）。通常将 0.05（或有时为 0.01）以下的值解释为参数与零之间存在显著差异的证据。

要显示其他统计量，请在报表中右击并选择**列**菜单。默认情况下不显示以下统计量：

95% 下限 参数估计值的 95% 置信区间的下端点。

95% 上限 参数估计值的 95% 置信区间的上端点。

标准 Beta 标准化的参数估计值。适用于比较以不同尺度测量的 *X* 变量的效应。请参见第 140 页的 “‘参数估计值’报表”。

VIF 方差膨胀因子。

设计标准误差 参数估计值的设计标准误差。请参见第 140 页的 “‘参数估计值’报表”。

特殊拟合

使用**特殊拟合**选项，您可以变换 Y 和 X。尽管由于各种原因可以变换数据，但通常这样做是为了使呈现的数据更趋向正态分布，以便随后可以进行适当的检验。您还可以约束截距和斜率、拟合特定次数的多项式，以及使多项式中心化。

注意：有关该选项的示例，请参见第 128 页的 “‘特殊拟合’选项的示例”。

“指定变换或约束”窗口包含以下选项：

Y 或 X 变换 使用以下选项之一变换 Y 或 X 变量：

- 自然对数
- 平方根
- 平方
- 倒数
- 指数

次数 使用该选项可以拟合指定次数的多项式。

中心化多项式 要禁用多项式中心化，请取消选中**中心化多项式**复选框（图 5.19）。请注意，对于 *X* 变量的变换，不执行多项式中心化。中心化多项式可稳定回归系数并降低多重共线性。

约束截距 选中该复选框可将模型截距约束为指定值。

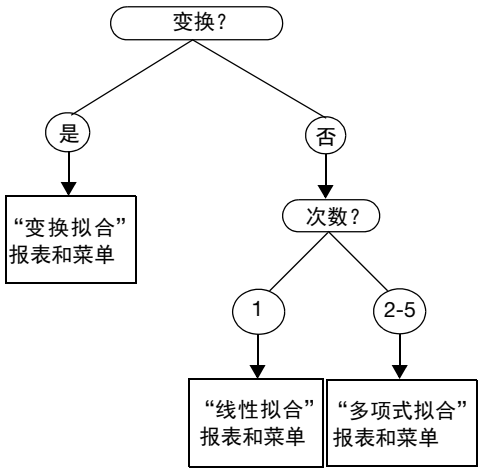
约束斜率 选中该复选框可将模型斜率约束为指定值。

有关该拟合的红色小三角选项的详细信息，请参见第 124 页的“拟合菜单”。

“特殊拟合”报表和菜单

根据您在“特殊拟合”窗口中的选择，您将看到不同的报表和菜单。图 5.14 中的流程图显示根据您的选择您将看到的报表和菜单。

图 5.14 “特殊拟合”流程图的示例



“变换拟合”报表

“变换拟合”报表包含第 111 页的“‘线性拟合’报表和‘多项式拟合’报表”中所述的报表。但是，若变换 *Y*，则显示“基于原始尺度衡量的拟合”报表。该报表显示基于原始 *Y* 变量的拟合测量值，以及变换回原始尺度的拟合模型。

灵活

使用“灵活”菜单中的选项可控制估计回归曲线的平滑度。

- “拟合样条”使用惩罚最小二乘法。使用参数 Lambda 调整平滑度。请参见第 118 页的“拟合样条”。
- “核平滑法”基于局部加权拟合。使用参数 alpha 控制局部行为的影响。请参见第 118 页的“核平滑法”。
- “拟合每个值”计算每个 X 值处的响应均值。请参见第 119 页的“拟合每个值”。

拟合样条

使用**拟合样条**选项，您可以拟合根据 lambda (λ) 值变动平滑度（或灵活度）的平滑样条。lambda 值是样条公式中的调节参数。随着 λ 值减小，样条模型中的误差项权重增大，拟合变得更灵活、更弯曲。随着 λ 值增大，拟合会变得更陡（曲度变小），接近一条直线。

请注意下列信息：

- 平滑样条可帮助您查看 *Y* 在 *X* 中分布的预期值。
- 最接近每段拟合曲线的点对其影响最大。影响会随着您减小 λ 的值而增大，生成高度灵活的曲线。
- 若想要使用菜单中未列出的 lambda 值，请选择**拟合样条 > 其他**。若 *X* 变量的尺度改变，拟合模型也随之改变。为防止出现改变，请选择**标准化 X** 选项。请注意，拟合模型对于原始 *X* 变量或统一尺度的 *X* 变量保持不变。
- 尝试多个 λ 值可能对您有所帮助。您可以使用“平滑样条”报表下方的 Lambda 滑块来试用不同的 λ 值。但是，λ 对于不同的数据尺度并不是一成不变的。例如，以英寸测量的 *X* 的 λ 值不同于以厘米测量的 *X* 的 λ 值。

有关“拟合平滑样条”菜单中的选项的详细信息，请参见第 124 页的“拟合菜单”。有关该拟合的统计详细信息，请参见第 138 页的“拟合样条”。

“平滑样条拟合”报表

“平滑样条拟合”报表包含样条拟合的 **R 方**和**误差平方和**。您可以使用这些值将样条拟合与其他拟合进行比较，或在不同的样条拟合之间进行比较。

R 方 测量平滑样条模型可解释的变异比例。请参见第 141 页的“‘平滑拟合’报表”。

误差平方和 每个点到拟合样条的距离的平方和。这是在拟合样条模型后未解释的误差（**残差**）。

更改 Lambda 支持您通过输入某个数字或移动滑块来更改 λ 值。

核平滑法

使用**核平滑法**选项，可以生成通过在域中抽样点处反复查找局部加权拟合的简单曲线（线性或二次线条）而形成的曲线。众多的局部拟合（总计 128 个）合并起来，生成覆盖整个域的平滑曲线。该方法亦称 Loess 或 Lowess，最初是 Locally Weighted Scatterplot Smoother（局部加权散点图平滑法）的缩写。“核平滑法”选项实施 Cleveland (1979) 方法，其中对接近完美的拟合进行了微调；Cleveland 双权函数中的 6 * q50 参数被 max(6 * q50, 2 * q90) 取代，其中 q50 和 q90 分别是第 50 和第 90 百分位数。

使用该方法可快速查看变量间关系，帮助您确定要执行的分析或拟合的类型。

有关“局部平滑线”菜单的选项的详细信息，请参见第 124 页的“拟合菜单”。

“局部平滑线”报表

“局部平滑线”报表包含平滑法拟合的 **R 方**和**误差平方和**。您可以使用这些值将平滑法拟合与其他拟合进行比较，或在不同的平滑法拟合之间进行比较。

R 方 测量平滑法模型可解释的变异比例。请参见第 141 页的 “‘平滑拟合”报表”。

误差平方和 每个点到拟合平滑线的距离的平方和。这是在拟合平滑法模型后未解释的误差（**残差**）。

局部拟合 (lambda) 为每个局部拟合选择多项式次数。二次多项式可以更平滑地跟踪局部波动。 Lambda 是该方法拟合的特定多项式的次数。 Lambda 可为 0、 1 或 2。

权重函数 指定如何衡量各个局部拟合邻域数据的权重。Loess 使用 Tri-cube 法。权重函数确定每个 x_i 和 y_i 对于线条拟合的影响。影响随着 x_i 与 x 的距离增大而减小，最终变为零。

平滑性 (alpha) 控制各个局部拟合中包含的点数。使用滑块选择值或直接键入一个值。 Alpha 是平滑参数。它可以是任意正数，但典型值为 1/4 到 1。随着 alpha 增大，曲线会变得更平滑。

抽样 Delta 控制拟合过程中使用的抽样量。默认情况下，抽样 Delta 为零，这意味着不跳过任何点。随着抽样 Delta 的增加，拟合过程中将跳过最后一个抽样点 Delta 内的点。可以使用该选项减少数据密集时使用的点数。

稳健性 对各点再加权，以弱化离拟合曲线较远的点。指定重复该过程的次数 （操作数） 。目标是要收敛该曲线，并通过为离群值指定较小的权重自动过滤出离群值。

拟合每个值

拟合每个值选项用于对每个唯一的 *X* 值拟合一个值。拟合值是每个唯一的 *X* 值所对应响应的均值。

有关“拟合每个值”菜单的选项的详细信息，请参见第 124 页的“拟合菜单”。

“拟合每个值”报表

“拟合每个值”报表显示有关模型拟合的汇总统计量。

观测个数 观测总数。

唯一值个数 唯一 X 值的个数。

自由度 纯误差自由度。

平方和 纯误差平方和。

均方 纯误差均方。

拟合正交

拟合正交选项用于拟合解释 *X* 和 *Y* 中的变异性的线性模型。这也称为 Deming 回归。

注意：有关该选项的示例，请参见第 130 页的““拟合正交”选项的示例”。

“拟合正交”选项

选择以下选项之一来指定方差比。

一元方差，主成分 使用从 *X* 和 *Y* 的样本计算得出的一元方差估计值。结果就是标准化的第一主成分。该选项在测量系统应用中并不是一个很好的选择，因为误差方差很少与总体方差成比例。

方差相等 将 1 用作方差比，即假定误差方差是相同的。使用相等方差等价于非标准化的第一主成分线。假定散点图在 *X* 和 *Y* 方向上的尺度是相同的。在显示正态密度椭圆时，您会看到这条线是椭圆的最长轴。

以 X 拟合 Y 使用零作为方差比，指示 *Y* 实际上没有方差。

指定方差比 支持您任意输入所需比率，让您能够使用关于 *X* 中的测量值误差和 *Y* 中的响应误差的已知信息。

有关“正交拟合比”菜单中的选项的详细信息，请参见第 124 页的“拟合菜单”。有关该拟合的统计详细信息，请参见第 138 页的“拟合正交”。

正交拟合比报表

“正交拟合比”报表显示关于正交回归模型的汇总统计量。

变量 用于拟合线条的变量的名称。

均值 每个变量的均值。

标准差 每个变量的标准差。

方差比 用于拟合线条的方差比。

相关性 两个变量之间的相关程度。

截距 拟合线的截距。

斜率 拟合线的斜率。

置信下限 斜率的置信下限。

置信上限 斜率的置信上限。

Alpha 输入计算置信区间时使用的 alpha 水平。

稳健

稳健选项提供两种方法来降低数据集中离群值的影响。离群值可导致不正确的估计值和决策。

注意：有关该选项的示例，请参见第 132 页的““稳健拟合”命令的示例”。

有关“稳健拟合”和“Cauchy 拟合”菜单中的选项的详细信息，请参见第 124 页的“拟合菜单”。

稳健拟合

“稳健拟合”选项可降低响应变量中离群值的影响。需要使用 Huber M 估计方法。Huber M 估计查找可将 Huber 损失函数最小化的参数估计值：

$$l(\underline{e}) = \sum_i \rho(e_i)$$

其中

$$\rho(e) = \begin{cases} \frac{1}{2}e^2 & \text{若 } |e| < k \\ k|e| - \frac{1}{2}k^2 & \text{若 } |e| \geq k \end{cases}$$

e_i 指残差

Huber 损失函数惩罚离群值，并且对于小误差呈二次增长，对于大误差呈线性增长。在 JMP 实现中，$k = 2$。有关稳健拟合的更多详细信息，请参见 Huber (1973) 和 Huber and Ronchetti (2009)。

拟合 Cauchy

Cauchy 选项使用最大似然和 Cauchy 连结函数估计参数。该方法假定误差服从 Cauchy 分布。Cauchy 分布具有比正态分布更肥大的尾部，从而弱化了对离群值的强调。若您的数据中有较大比例的离群值，该选项会很有用。不过，若您的数据接近正态，只含有少数离群值，该选项会导致不正确的推断。

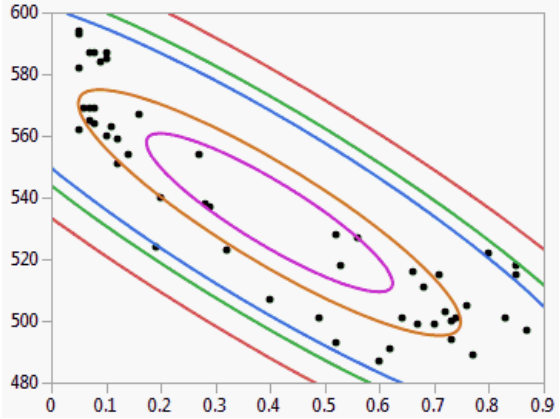
密度椭圆

使用**密度椭圆**选项，您可以绘制包含指定数量的点的一个或多个椭圆。点数由您从**密度椭圆**菜单中选择的概率决定。

注意：有关该选项的示例，请参见第 134 页的“使用密度椭圆的“依据...分组”的示例”。

有关“二元正态椭圆”菜单中的选项的详细信息，请参见第 124 页的“拟合菜单”。

图 5.15 密度椭圆的示例



密度椭圆由 *X* 和 *Y* 变量的二元正态分布拟合计算得出。二元正态密度是 *X* 和 *Y* 变量的均值和标准差以及二者之间相关程度的函数。**其他**选择项支持您指定大于 0 且小于等于 1 的任意概率。

这些椭圆既是密度等高线也是置信曲线。作为置信曲线，这些椭圆显示给定百分比的数据预计出现在哪个区域（假定服从二元正态分布）。

密度椭圆能够很恰当地以图形方式指示两个变量之间的相关程度。当两个变量之间的相关程度趋向于 1 或 –1 时，椭圆将沿对角方向压缩。若两个变量相关程度较弱时，该椭圆将更圆（不沿对角方向延伸）。

“相关性”报表

伴随每个**密度椭圆**拟合的“相关性”报表显示 *X* 和 *Y* 变量的相关系数。

注意：要查看多对变量的椭圆矩阵和相关性，请使用**分析 > 多元方法**菜单中的**多元**平台。

变量 创建椭圆时使用的变量的名称。

均值 *X* 和 *Y* 变量的平均值。

标准差 *X* 和 *Y* 变量的标准差。

“分布”一章中第 41 页的 “‘汇总统计量’报表”一节中对均值和标准差进行了说明。

相关性 Pearson 相关系数。若两个变量之间存在精确线性关系，根据这两个变量是正相关还是负相关，相关性为 1 或 –1。若不存在任何关系，相关性趋向于零。请参见第 141 页的 “‘相关性’报表”。

显著性概率 当 *X* 变量与 *Y* 变量之间不存在线性关系时，得到相关性的绝对值大于计算值的概率。

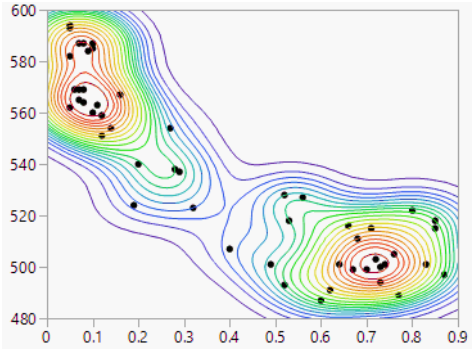
数字 计算中使用的观测数。

非参数密度

当图中显示数以千计的点时，如此多的点可能过于密集而无法显示密度中的模式。使用非参数密度选项可以很容易看清模式。

“非参数密度”估计可描述数据点密度的平滑非参数二元曲面。该图通过添加一组等高线来显示密度（图 5.16）。这些等高线是以 5% 为区间的分位数等高线。这意味着从估计的非参数分布生成的大约 5% 的点位于最低等高线之下，10% 的点位于次低等高线之下，依此类推。最高等高线之下有大约 95% 的点。

图 5.16 非参数密度的示例



要更改非参数密度等高线网格的大小，请按 Shift 键并从“二元”红色小三角菜单中选择**非参数密度**。输入大于默认值 102 个点的值。

有关“分位数密度等高线”菜单中的选项的详细信息，请参见第 124 页的“拟合菜单”。

“分位数密度等高线”报表

“分位数密度等高线”报表显示创建非参数密度时使用的标准差。

依据 ... 分组

使用**依据...分组**选项，您可以选择分类（分组）变量。使用分组变量时，“二元”平台将为分组变量中的每个水平计算单独的分析。散点图为每个分组变量显示拟合模型（直线、曲线、等高线或椭圆）。散点图下方会标识出针对分组变量中每个水平的拟合，并且分别提供弹出菜单。

使用分组变量时，将选中“二元拟合”红色小三角菜单中的**依据...分组**选项。您可以更改分组变量，方法是：首先选择**依据...分组**选项以删除（取消选定）现有变量，然后再次选择**依据...分组**选项并按之前的方法响应其窗口操作。

依据...分组选项支持您依据分组变量生成单独的分析，同时在同一散点图上标绘拟合。这样您就可以直观地比较不同组之间的拟合。

注意：有关该选项的示例，请参见第 134 页的“使用密度椭圆的“依据...分组”的示例”和第 135 页的“使用回归线的“依据...分组”的示例”。

拟合菜单

除报表之外，每个拟合选项还会向报表窗口添加一个拟合菜单。下表显示对应于每个拟合选项的拟合菜单。

拟合选项	拟合菜单
拟合均值	拟合均值
拟合线	线性拟合
拟合多项式	多项式拟合次数=X*
特殊拟合	线性拟合 多项式拟合次数=X* 变换拟合 X* 约束拟合
拟合样条	平滑样条拟合，lambda=X*
核平滑法	局部平滑线
拟合每个值	拟合每个值
拟合正交	正交拟合比=X*
稳健拟合	稳健拟合

拟合选项	拟合菜单
拟合 Cauchy	Cauchy 拟合
密度椭圆	二元正态椭圆 P=X*
非参数密度	分位数密度等高线

*X= 变量字符或数字

拟合菜单选项

“拟合”菜单选项取决于选定的拟合。

- 第 125 页的“适用于大多数拟合的选项”
- 第 125 页的“适用于多个拟合的选项”
- 第 126 页的“适用于二元正态椭圆的选项”
- 第 126 页的“适用于分位数密度等高线的选项”

适用于大多数拟合的选项

拟合线 显示或隐藏描述模型拟合的直线或曲线。对于 “二元正态椭圆”报表，该选项显示或隐藏表示等高线边界的椭圆。不适用于 “分位数密度等高线”。

线条颜色 支持您从调色板中选择要为每个拟合指定的颜色。不适用于“分位数密度等高线”。

线条样式 支持您从线条样式选项板中为每个拟合选择线条样式。不适用于“分位数密度等高线”。

线条粗细 支持您更改拟合线的线条粗细。默认线条粗细是最细线。不适用于 “分位数密度等高线”。

报表 启用或禁用拟合的报表。不修改二元图。

删除拟合 从图形中删除拟合，并删除该拟合的报表。

适用于多个拟合的选项

拟合置信曲线 显示或隐藏预期值 （均值）的置信限。该选项不适用于 “拟合样条” 、“密度椭圆”、“拟合每个值”和 “拟合正交”拟合，所以在这些菜单上灰显。

单值置信曲线 显示或隐藏单个预测值的置信限。置信限反映误差变异和参数估计值的变异。该选项不适用于 “拟合均值”、“拟合样条”、“密度椭圆”、“拟合每个值”和 “拟合正交”拟合，所以在这些菜单上灰显。

保存预测值 在当前数据表中创建名为**预测列名**的新列，其中**列名**是 *Y* 变量的名称。该列包括预测公式和计算的样本预测值。预测公式为您添加至该表的行自动计算值。该选项不适用于“拟合每个值”和“密度椭圆”拟合，所以在这些菜单上灰显。

您可以对每个拟合使用**保存预测值**和**保存残差**选项。若多次使用这些选项或是对分组变量使用这些选项，最好重命名数据表中的结果列以反映每个拟合。

保存残差 在当前数据表中创建名为**残差列名**的新列，其中**列名**是 *Y* 变量的名称。每个值都是实际（观测）值与其预测值之间的差值。不同于**保存预测值**选项，该命令不在新列中创建公式。该选项不适用于“拟合每个值”和“密度椭圆”拟合，所以在这些菜单上灰显。

您可以对每个拟合使用**保存预测值**和**保存残差**选项。若多次使用这些选项或是对分组变量使用这些选项，最好重命名数据表中的结果列以反映每个拟合。

保存学生化残差 在数据表中创建一个新列，在其中包含残差除以残差标准误差之后的结果。

均值置信限公式 在包含均值置信区间公式的数据表中创建新列。

单值置信限公式 在包含单值置信区间公式的数据表中创建新列。

标绘残差（仅限线性拟合、多项式拟合和特殊拟合） 生成五种诊断图：“预测值 - 残差”图、“预测值 - 实际值”图、“行号 - 残差”图、“X- 残差”图和残差的正态分位数图。请参见第 127 页的“诊断图”。

设置 α 水平 支持您设置在计算多个拟合的置信带时使用的 alpha 水平。

着色拟合置信带 绘制与**拟合置信曲线**选项相同的曲线，并对曲线之间的区域着色。

着色单值置信带 绘制与**单值置信曲线**选项相同的曲线，并对曲线之间的区域着色。

保存系数 将样条系数另存为新的数据表，其中的列名为 X、A、B、C 和 D。X 列包含结点。A、B、C 和 D 分别为三次多项式的截距、线性系数、二次系数和三次系数。这些系数从 X 列中的相应值跨越至下一个最高值。

适用于二元正态椭圆的选项

着色等高线 对密度椭圆内部区域着色。

选择内部点 选择椭圆内部的点。

选择外部点 选择椭圆外部的点。

适用于分位数密度等高线的选项

核心控制 为每个变量显示一个滑块，可通过滑块更改定义 *X* 和 *Y* 值范围的标准差，以确定等高线的密度。

5% 等高线 显示或隐藏 5% 等高线。

等高线 显示或隐藏 10% 等高线。

填充等高线 填充等高线之间的区域。

颜色主题 更改等高线的颜色主题。

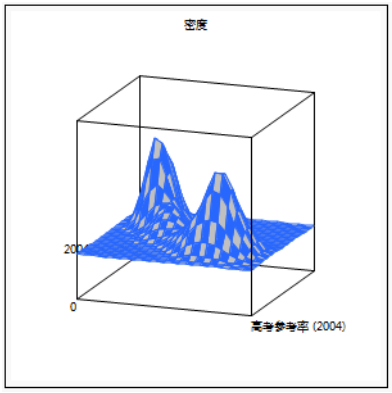
按密度选择点 选择用户指定分位数范围内的点。

按密度分位数显示颜色 根据密度对点着色。

保存密度分位数 创建包含每个点所在的密度分位数的新列。

网格图 一个叠加在两个分析变量的网格之上的三维密度图。

图 5.17 网格图的示例



模态聚类 在当前数据表中创建新列，在其中填充聚类值。

> **注意：**若先保存模态聚类值，再保存密度网格，则网格表中也包含聚类值。对于为图中的点进行着色和标记，聚类值很有用。

保存密度网格 将密度估计值以及与其关联的分位数保存到新的数据表中。该网格数据可用于以其他方式（比如使用“三维散点图”或“等高线图”平台）直观演示密度。

诊断图

标绘残差选项用于创建残差图和其他图，以便诊断模型拟合。提供以下图：

“预测值 - 残差”图 预测值与残差之间的关系图。同时还创建残差直方图。

“预测值 - 实际值”图 预测值与实际值之间的关系图。

“行号 - 残差”图 行号与残差值之间的关系图。

“X- 残差”图 X 变量与残差值之间的关系图。

残差正态分位数图 残差的正态分位数图。

“二元”平台的更多示例

以下各节提供不同二元拟合的示例。

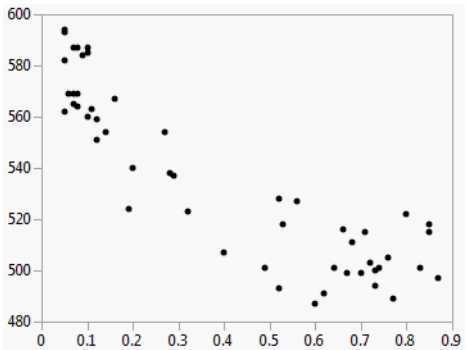
- “‘特殊拟合’选项的示例”
- “‘拟合正交’选项的示例”
- “‘稳健拟合’命令的示例”
- “使用密度椭圆的‘依据 ... 分组’的示例”
- “使用回归线的‘依据 ... 分组’的示例”
- “使用‘依据’变量进行分组的示例”

“特殊拟合”选项的示例

本例显示如何向数据添加变换。先将 Y 变换为对数，再将 X 变换为平方根：

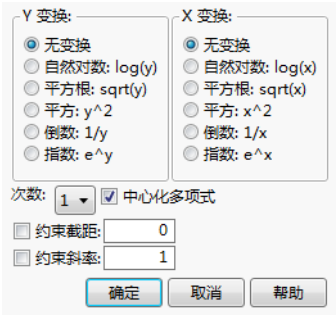
1. 选择**帮助** > **样本数据库**，然后打开 SAT.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择 **2004 语文考试**并点击 **Y，响应**。
4. 选择**高考参考率 (2004)** 并点击 **X，因子**。
5. 点击**确定**。

图 5.18 以高考参考率拟合 SAT 分数的示例



6. 点击“二元拟合”红色小三角并选择**特殊拟合**。随即显示“指定变换或约束”窗口。有关该窗口的说明，请参见第 116 页的“特殊拟合”。

图 5.19 “指定变换或约束”窗口



7. 在“Y 变换”内，选择**自然对数 : log(y)**。
8. 在“X 变换”内，选择**平方根 : sqrt(x)**。
9. 点击**确定**。

图 5.20 “特殊拟合”报表的示例

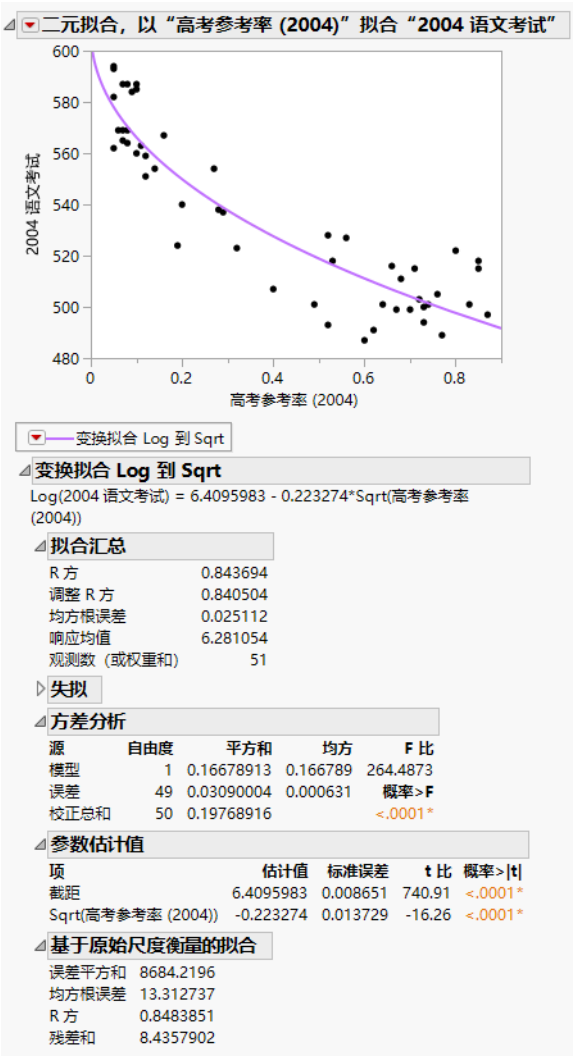


图 5.20 显示以原始尺度标绘的拟合线。该模型看起来能够很好地拟合数据，因为标绘的线条穿过了点簇。

“拟合正交”选项的示例

本例显示如何使用“分布”平台将变量标准化，然后使用标准化的变量拟合正交模型。

使变量标准化

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。

2. 选择**分析 > 分布**。
3. 选择**身高**和**体重**，并点击 **Y，列**。
4. 点击**确定**。
5. 按住 Ctrl 键。点击**身高**红色小三角并选择**保存 > 标准化**。

   按住 Ctrl 键将该操作传播至报表窗口中的所有变量。请注意， Big Class.jmp 样本数据表中添加了两个新列。
6. 关闭 “分布”报表窗口。

使用标准化变量拟合正交模型

1. 从 Big Class.jmp 样本数据表中，选择**分析 > 以 X 拟合 Y**。
2. 选择**标准化 : 体重**并点击 **Y，响应**。
3. 选择**标准化 : 身高**并点击 **X，因子**。
4. 点击**确定**。
5. 点击 “二元拟合，以 ‘标准化 : 身高’ 拟合 ‘标准化 : 体重’” 旁边的红色小三角，然后选择**拟合线**。
6. 点击以 ‘标准化 : 身高’ 拟合 ‘标准化 : 体重’” 旁边的红色小三角，然后选择**拟合正交**。然后选择以下每一项：
   - **方差相等**
   - **以 X 拟合 Y**
   - **指定方差比**，并键入 0.2。
   - **指定方差比**，并键入 5。

图 5.21 正交拟合选项的示例

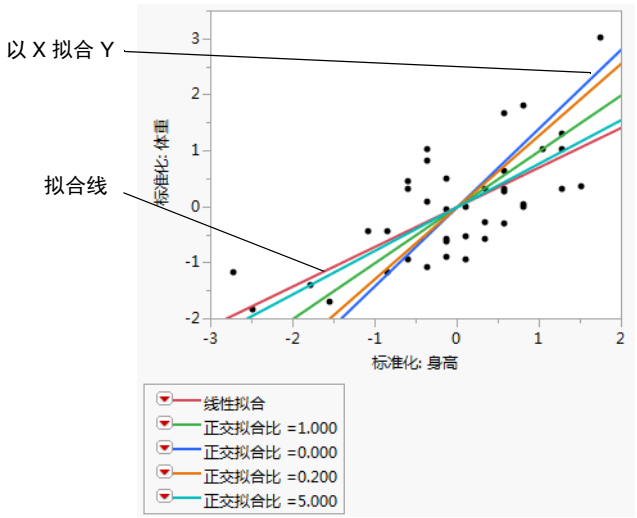


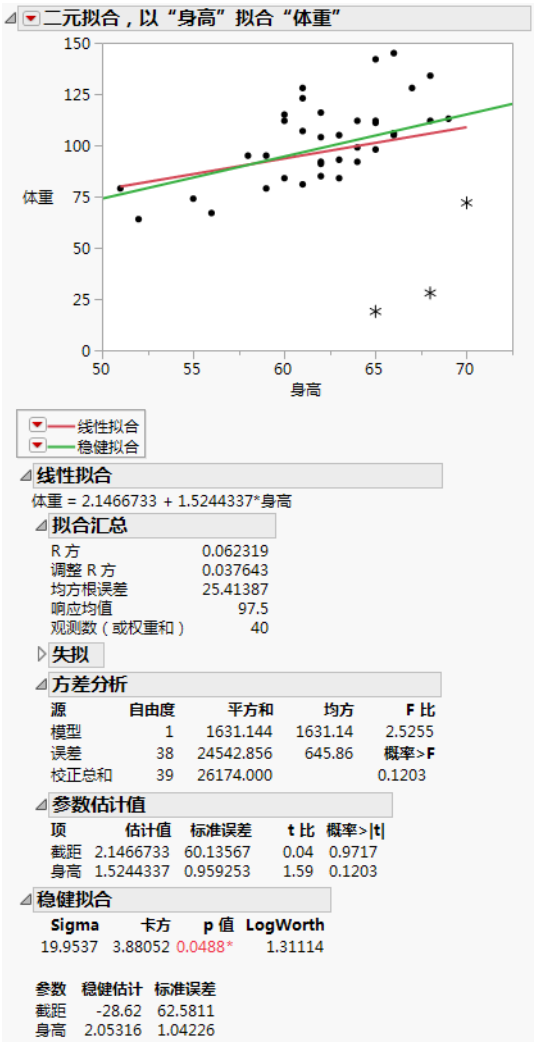
图 5.21 中的散点图显示标准化的身高和体重值，并通过多个线条拟合来演示所选的不同正交方差比的行为。在认为 *X* 变量的方差非常小时，将呈现标准线性回归（**拟合线**）。当 *Y* 变量的变异可忽略时，**以 X 拟合 Y** 是另一个极端。所有其他线条都落在这两个极端之间，并随方差比的变动而偏移。随着方差比增大，*Y* 响应中的变异占主导地位，拟合线的偏斜程度更靠近以 *X* 拟合 *Y*。同样，当您减小方差比时，拟合线的偏斜程度更靠近以 *Y* 拟合 *X*。

“稳健拟合”命令的示例

本例显示如何使用 Huber M 估计方法拟合稳健模型。Weight Measurements.jmp 样本数据表中的数据显示 40 名学生的身高和体重测量值。

1. 选择**帮助 > 样本数据库**，然后打开 Weight Measurements.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**体重**，然后点击 **Y，响应**。
4. 选择**身高**，然后点击 **X，因子**。
5. 点击**确定**。
6. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**拟合线**。
7. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**稳健 > 稳健拟合**。

图 5.22 “稳健拟合”的示例



若您查看标准“方差分析”报表，可能会错误地得出身高和体重不具有线性关系的结论，因为 p 值为 0.1203。但是，当您查看“稳健拟合”报表时，可能得出它们具有线性关系的结论，因为这里的 p 值为 0.0489。一些测量值非常低，可能是用户输入不正确造成的。这些测量值过度影响了分析。

使用密度椭圆的“依据 ... 分组”的示例

本例显示如何使用分组 （依据）变量并向数据中添加密度椭圆。在 Hot Dogs.jmp 样本数据表中，**原料类型**列标识三种不同类型的热狗：beef （牛肉）、 meat （猪肉）或 poultry （禽肉）。您想要根据这三种热狗的成本变量对其进行分组。

1. 选择**帮助** > **样本数据库**，然后打开 Hot Dogs.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**美元 / 盎司**并点击 **Y，响应**。
4. 选择**美元 / 磅 （蛋白质）**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“二元拟合，以 ‘美元 / 磅 （蛋白质）’ 拟合 ‘美元 / 盎司’”旁边的红色小三角，然后选择**分组依据**。
7. 从列表中选择**原料类型**。
8. 点击**确定**。

   若再次查看**依据 ... 分组**选项，您会看到该选项旁边有一个复选标记。
9. 点击“二元拟合，以 ‘美元 / 磅 （蛋白质）’ 拟合 ‘美元 / 盎司’”旁边的红色小三角，然后选择**密度椭圆** > 0.90。

根据**原料类型**对点着色：

10. 右击散点图并选择**行图例**。
11. 选择列列表中的**原料类型**并点击**确定**。

图 5.23 “依据 ... 分组”的示例

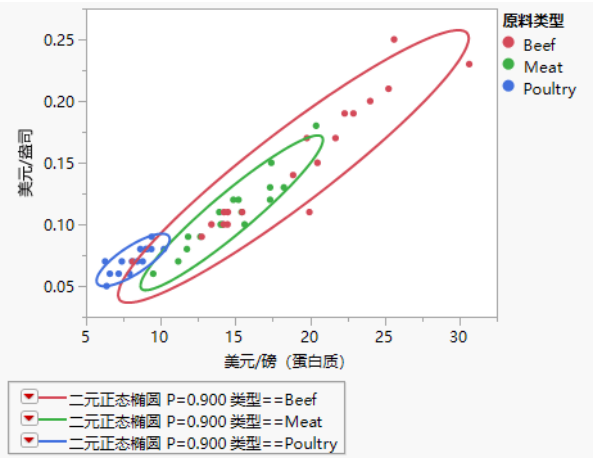


图 5.23 中的椭圆清晰地显示了不同类型的热狗针对成本变量如何聚类。

使用回归线的“依据 ... 分组”的示例

本例显示如何使用分组变量叠加回归线，以比较不同组的斜率。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**，然后点击 **Y，响应**。
4. 选择**身高**，然后点击 **X，因子**。
5. 点击**确定**。

创建图 5.24 中左侧的示例：

6. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**拟合线**。

创建图 5.24 中右侧的示例：

7. 从“线性拟合”菜单中，选择**删除拟合**。
8. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**分组依据**。
9. 从列表中选择**性别**。
10. 点击**确定**。
11. 点击“二元拟合，以‘身高’拟合‘体重’”旁边的红色小三角，然后选择**拟合线**。

图 5.24 针对整个样本和分组样本的回归分析的示例

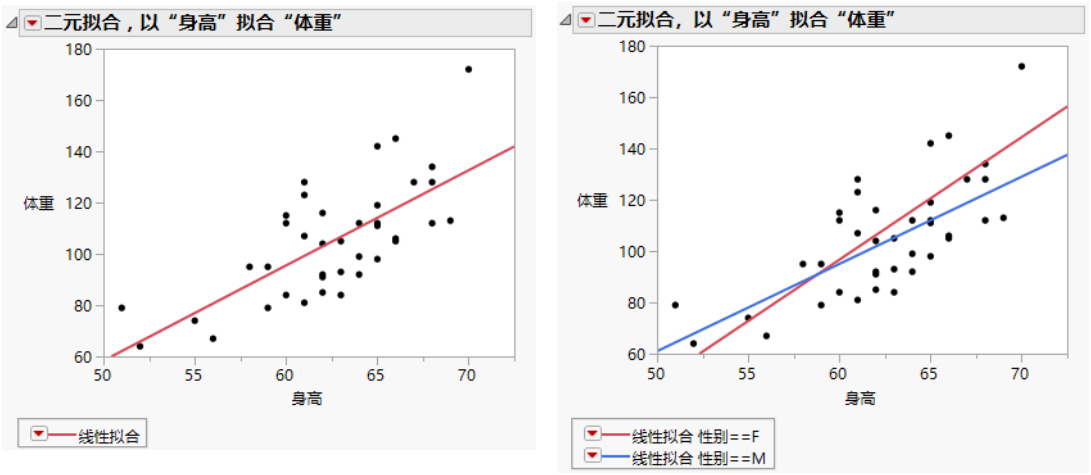


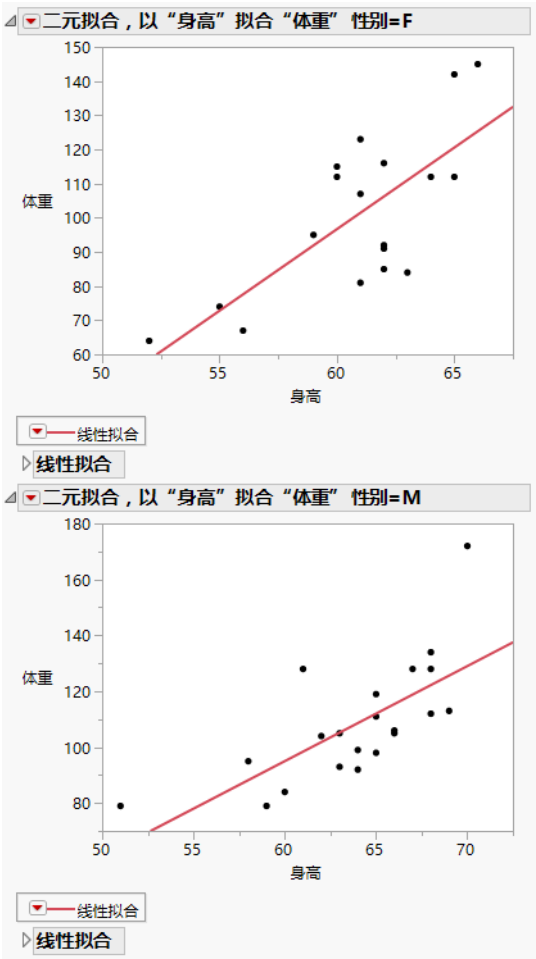
图 5.24 中左侧的散点图中有一条回归线将体重和身高关联起来。右侧的散点图显示分别对应于男性和女性的回归线。

使用“依据”变量进行分组的示例

本例显示另一种通过在启动窗口中指定“依据”变量进行分组的方法。这将为“依据”变量（或“依据”变量的组合）的每个水平生成单独的报表和图形。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**，然后点击 **Y，响应**。
4. 选择**身高**，然后点击 **X，因子**。
5. 选择**性别**并点击**依据**。
6. 点击**确定**。
7. 按 Ctrl 键，点击“二元拟合，以‘身高’拟合‘体重’，性别 =F”旁边的红色小三角菜单，然后从红色小三角菜单中选择**拟合线**。

图 5.25 “依据”变量图的示例



“依据”变量（性别）的每个水平都有单独的分析。所以您看到有针对女性的散点图和针对男性的散点图。

“二元”平台的统计详细信息

- “拟合线”
- “拟合样条”
- “拟合正交”
- ““拟合汇总”报表”
- ““失拟”报表”
- ““参数估计值”报表”
- ““平滑拟合”报表”
- ““相关性”报表”

拟合线

拟合线选项为拟合点的直线查找参数 β_0 和 β_1，以使残差平方和最小化。第 i 行的模型表示为 $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$。

二次多项式是一条抛物线；三次多项式是一条三次曲线。对于 k 次，第 i 个观测的模型定义如下：

$$y_i = \sum_{j=0}^{\kappa} \beta_j x_i^j + \varepsilon_i$$

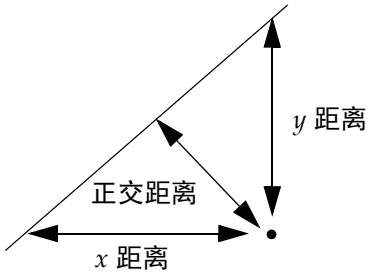
拟合样条

三次样条方法使用一组接合在一起的三次多项式，由此在接合点（结点）处生成连续平滑的曲线。通过组合误差平方和与曲线范围内积分运算得到的曲率的罚值，使所生成的目标函数达到最小化来完成估计。请参见 Reinsch (1967) 的论文或 Eubank (1999) 的著作，了解该方法的说明。

拟合正交

标准最小二乘拟合假设 X 变量是固定的，Y 变量是 X 加上误差所构成的函数。若测量的 X 存在随机变异，则应拟合一条线，将垂直差值的平方和最小化（图 5.26）。 但是，垂直距离取决于 X 和 Y 的尺度，并且如何将垂线缩放是一个统计问题而不是图形问题。

图 5.26 垂直于拟合线的线



该拟合要求指定 *Y* 中的误差与 *X* 中的误差的方差比。这是误差的方差，不是抽样点的方差，所以您必须仔细选择。$(\sigma_y^2)/(\sigma_x^2)$ 之比在标准最小二乘法中是无限的，因为 σ_x^2 为零。若执行的正交拟合具有较大的误差比，拟合线将接近于标准最小二乘拟合线。若指定比率为零，拟合等价于 *X* 对于 *Y* 的回归，而不是 *Y* 对于 *X* 的回归。

该方法最常用于比较在测量同一值时均存在误差的两个测量系统。因此，*Y* 响应误差和 *X* 测量值误差均为同一类型的测量值误差。从何得到测量值误差方差呢？您无法从二元数据得到，因为您无从知晓哪一个测量系生成哪部分的误差。所以，您要么盲目假定 1 之类的比率，要么必须依赖这两个测量系统对同一单位各自的重复测量值。

该方法的一个优点是：计算可以给出 *Y* 和 *X* 的预测值；预测值是拟合线上最接近数据点的点，接近程度与方差比相关。

置信限按照 Tan and Iglewicz (1999) 中所述计算。

“拟合汇总”报表

R 方

使用相应方差分析表中的量，任意连续响应拟合的 R 方计算如下：

$$\frac{\text{模型的平方和}}{\text{校正总和的平方和}}$$

调整 R 方

调整 R 方是均值平方 （而非平方和）之比，按以下公式计算：

$$1 - \frac{\text{误差均方}}{\text{校正总和的均值平方}}$$

误差均方列在“方差分析”报表中（图 5.12）。 您可以按如下方法计算**校正总和**的均方：用**校正总和**的**平方和**除以其相应的自由度。

“失拟”报表

纯误差自由度

对于“纯误差自由度”，请考虑多个观测具有相同身高值的情况。通常，若 g 个组中有多个行对每个效应都包含相同值，则合并自由度（表示为 DF_p）定义如下：

$$DF_p = \sum_{i=1}^{g} (n_i - 1)$$

其中，n_i 是第 i 组中的观测数。

纯误差平方和

对于纯误差平方和，通常，若 g 个组中有多个行都具有相同的 x 值，则合并平方和（表示为 SS_p）定义如下：

$$SS_p = \sum_{i=1}^{g} SS_i$$

其中，SS_i 是第 i 组的已校正均值的平方和。

最大 R 方

由于**纯误差**不随模型形式变化而变化，并且是最小可能方差，**最大 R 方**计算如下：

$$1 - \frac{SS(\text{纯误差})}{SS(\text{整体模型合计})}$$

“参数估计值”报表

标准 Beta

“标准 Beta”计算如下：

$$\hat{\beta}(s_x / s_y)$$

其中，$\hat{\beta}$ 是估计的参数，s_x 和 s_y 分别是 X 和 Y 变量的标准差。

设计标准误差

设计标准误差通过参数估计值的标准误差除以 RMSE 的方式计算。

“平滑拟合”报表

R 方等于 1-(误差平方和/校正总和平方和)，其中校正总和平方和在“拟合线方差分析”报表中提供。

“相关性”报表

Pearson 相关系数表示为 r，计算如下：

$$r_{xy} = \frac{s_{xy}}{\sqrt{s_x^2 s_y^2}} \text{ 其中 } s_{xy} = \frac{\sum w_i (x_i - \overline{x}_i)(y_i - \overline{y}_i)}{df}$$

其中，w_i 是第 i 个观测的权重值（若指定了权重列），或为 1（若未指定权重列）。

第 6 章

单因子分析

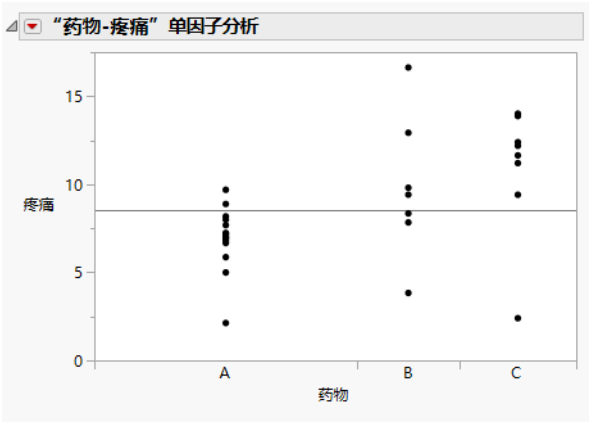
检查连续 Y 和分类 X 变量之间的关系

使用“单因子”平台可探索在分类 X 变量定义的分组间连续 Y 变量的不同分布状况。您可以使用方差分析、非参数检验或多重比较来评估各组间均值的差异。密度图和 CDF 图支持您直观演示按 X 类别划分的响应的分布。例如，您可能要了解不同类别的同类型药物 (X) 如何影响患者的疼痛程度（用数值尺度 Y 表示）。

“单因子”平台是“以 X 拟合 Y”平台中以**名义型或有序型拟合连续型的**特质。分析结果显示在图中，您可以通过交互方式添加其他分析，例如：

- 单因子方差分析，以拟合均值和检验它们是否相等
- 非参数检验
- 方差齐性检验
- 均值的多重比较检验（使用均值比较环）
- 每个组的离群值箱线图叠加
- 单因子布局的功效详细信息

图 6.1 单因子分析



目录

单因子分析概述 146
单因子分析示例 146
启动“单因子”平台 148
数据格式 148
单因子图 149
“单因子”平台选项 149
显示选项 152
分位数 153
离群值箱线图 154
“均值 / 方差分析”和“均值 / 方差分析 / 合并的 t” 154
“拟合汇总”报表 155
t 检验报表 155
“方差分析”报表 156
“单因子方差分析均值”报表 156
“区组均值”报表 157
“均值菱形”和“X 轴成比例” 157
均值线、误差条和标准差线 158
均值分析方法 158
位置均值分析 159
尺度均值分析 159
均值分析图表 160
“均值分析”选项 161
比较均值 161
使用比较环 162
每对，Student t 164
所有对，Tukey HSD 164
带最佳组的 Hsu MCB 检验 164
带控制组的 Dunnett 检验 165
每对逐步，Newman-Keuls 166
“比较均值”选项 167
非参数检验 167
Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验报表 168
“Kolmogorov-Smirnov 双样本检验”报表 169
非参数多重比较 170
不等方差 173
“等方差检验”报表 173
等价性检验 175

“等价性检验”报表 175
稳健 176
稳健拟合 176
Cauchy 拟合 177
功效 177
“功效详细信息”窗口和报表 177
正态分位数图 178
CDF 图 178
密度 178
匹配列 179
“单因子”平台的更多示例 179
均值分析图表示例 180
方差均值分析图表的示例 181
“每对，Student t”检验的示例 181
“所有对，Tukey HSD”检验的示例 183
“带最佳组的 Hsu MCB 检验”的示例 185
“带控制组的 Dunnett 检验”的示例 186
“每对逐步，Newman-Keuls”检验的示例 187
示例：比较四个“比较均值”检验 188
非参数 Wilcoxon 检验的示例 189
“不等方差”选项的示例 191
“等价性检验”的示例 192
“稳健拟合”选项的示例 194
“功效”选项的示例 195
正态分位数图的示例 197
CDF 图的示例 198
“密度”选项的示例 199
“匹配列”选项的示例 200
示例：堆叠数据以进行单因子分析 201
“单因子”平台的统计详细信息 208
比较环 208
功效 209
“拟合汇总”报表 210
等方差检验 210
非参数检验统计量 211

单因子分析概述

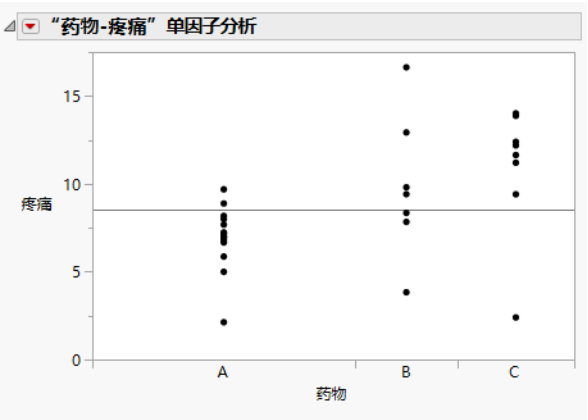
单因子方差分析检验各组均值之间的差异。将响应的总变异性分为两个部分：组内变异性和组间变异性。若组间变异性相对组内变异性而言很大，则认为组均值之间的差异是显著的。

单因子分析示例

了解如何在 JMP 中执行单因子方差分析，以检验组均值中的差异是否显著。在该数据中，对 33 个参与者提供了三个不同类型的镇痛剂（A、B 和 C）。要求这些参与者使用滑尺评价自己的痛苦程度。您要确定 A、 B 和 C 的均值是否具有显著差异。

1. 选择**帮助 > 样本数据库**，然后打开 Analgesics.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**疼痛**并点击 **Y，响应**。
4. 选择**药物**并点击 **X，因子**。
5. 点击**确定**。

图 6.2 单因子分析示例



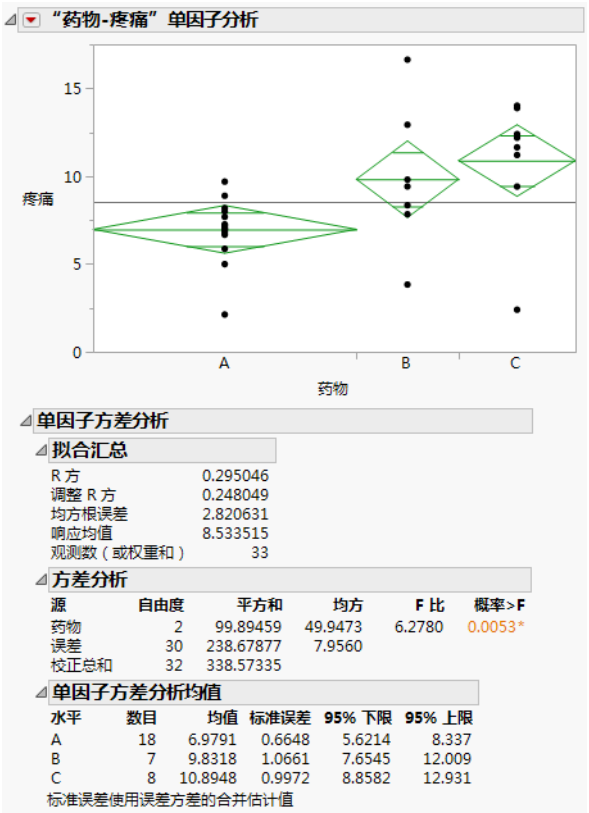
您注意到一种药物 (A) 总是比其他两种药物的得分低，还注意到 *x* 轴的刻度间距并不相等。刻度之间的间距与每种药物的得分（观测）数成比例。

对数据执行方差分析。

6. 点击 “‘药物 - 疼痛’单因子分析”旁边的红色小三角并选择**均值 / 方差分析**。

注意：若 X 因子只有两个水平，则**均值/方差分析**选项显示为**均值/方差分析/合并的 t** 并向报表窗口添加合并的 *t* 检验报表。

图 6.3 “均值 / 方差分析”选项示例



请注意以下事项：

- 显示表示置信区间的均值菱形。
  - 每个菱形中心附近的线表示组均值。您一眼就可以看到每种药物的均值看上去具有显著差异。
  - 每个菱形的垂直跨度表示每个组均值的 95% 置信区间。

  请参见第 157 页的 “‘均值菱形’和‘X 轴成比例’”。
- “拟合汇总”表提供有关分析的所有汇总信息。
- “方差分析”报表显示标准方差分析信息。注意，**概率 > F**（*p* 值）为 0.0053，这支持您的直观结论，即这些药物之间具有显著差异。
- “单因子方差分析均值”报表显示分类因子每个水平的均值、样本大小和标准误差。

启动“单因子”平台

要执行单因子方差分析，请执行以下操作：

1. 选择**分析 > 以 X 拟合 Y**。
2. 为 **Y，响应**输入连续列。
3. 为 **Y，因子**输入名义型或有序型列。

图 6.4 “以 X 拟合 Y”启动窗口

“单因子”一词出现在图的上方，指示您正在执行单因子分析。

注意：您还可以从“JMP 起始页”窗口启动单因子分析。选择**视图 > JMP 起始页 > 基本 > 单因子**。

有关该启动窗口的详细信息，请参见第 97 页的““以 X 拟合 Y”介绍”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

指定“区组”变量将标识另一个因子，形成无交互作用的双因子分析。在单因子图中，Y 变量的值以“区组”变量为中心。若按分组单元格划分的每个区组中的计数相等，则可用的“单因子”平台选项将适用于双因子分析设置。若按分组单元格划分的区组中的计数不相等，则“拟合模型”报表将显示在单因子图下方，并且“单因子”平台选项不可用。

数据格式

“单因子”平台要求每行包含有关 X 变量的同一水平的一个或多个观测的信息。若某行表示不止一个观测，则必须使用“权重”或“频数”变量可指示该行表示的观测数。

若单因子数据采用 JMP 数据表之外的格式，有时这些数据会排列成一行中包含多个观测的信息。要在 JMP 中分析这些数据，您必须导入数据并调整其结构，令 JMP 数据表中的每行都包含一个观测的信息。请参见第 201 页的“示例：堆叠数据以进行单因子分析”。

单因子图

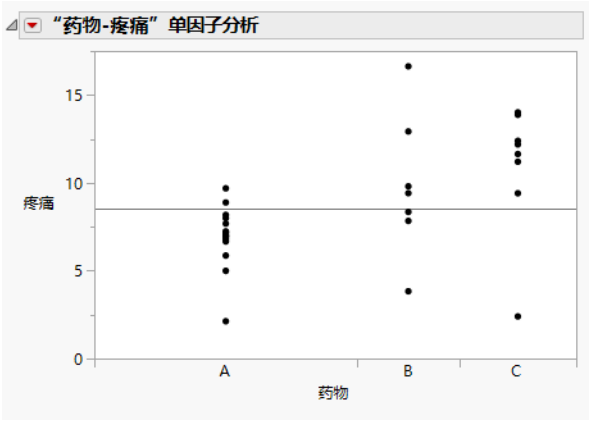
单因子图显示每个 X 因子值的响应点。您可以比较 X 因子各水平的响应分布情况。X 的不同值有时称为水平。若在启动窗口中指定了“区组”变量，则单因子图中 Y 变量的值将以“区组”变量为中心。

通过以下两种方式之一替换图中的变量：将一个轴上的变量拖放到另一个轴上来交换现有变量；或者，点击相关数据表的列面板中的某个变量并将其拖到轴上。

您可以使用“单因子分析”的红色小三角菜单中的选项向报表窗口添加报表、更多图和检验。请参见第 149 页的““单因子”平台选项”。

要生成图 6.5 中所示的图，请按第 146 页的“单因子分析示例”中的说明操作。

图 6.5 单因子图



注意：在数据表中排除的所有行也在单因子图中隐藏。

“单因子”平台选项

“单因子”平台红色小三角菜单包含图形和分析选项。这些选项包括方差分析、非参数检验和多重比较方法，用来评估各组间均值的差异。密度图和 CDF 图支持您直观演示按 X 类别划分的响应的分布。若在启动窗口中指定了“区组”变量，则其中许多选项都不可用。

注意：若您指定了多个 Y 或 X 变量，将显示“拟合组”菜单。使用菜单选项，您可以排列报表或按 R 方将它们排序。请参见《拟合线性模型》。

表 6.1 选项和元素的示例

平台选项	添加到图的对象	添加到报表窗口的报表
分位数	箱线图	“分位数”报表
均值/方差分析	均值菱形	“单因子方差分析”报表
均值和标准差	均值线、误差条和标准差线	“均值与标准差”报表
比较均值	比较环 （“每对逐步，Newman-Keuls”选项除外）	“均值比较”报表

“单因子分析”红色小三角菜单提供以下选项。除非满足特定条件，否则某些选项可能不显示。

分位数 为每个组列出以下分位数：

- 0%（最小值）
- 10%
- 25%
- 50%（中位数）
- 75%
- 90%
- 100%（最大值）

从**显示选项**菜单激活**箱线图**。请参见第 153 页的 “分位数”。

均值 / 方差分析 拟合每个组的均值并执行单因子方差分析以检验均值间是否有差异。请参见第 154 页的 ““均值 / 方差分析”和“均值 / 方差分析 / 合并的 t””。

注意：若 X 因子有两个水平，该菜单选项改为**均值 / 方差分析 / 合并的 t** 并生成一个假定方差相等的合并 *t* 检验报表。

均值和标准差 显示每个组的汇总统计量。均值的标准误差使用各个组标准差而非标准差的合并估计值。

图现在包含均值线、误差条和标准差线。有关这些元素的简单说明，请参见第 152 页的“显示选项”。有关这些元素的详细信息，请参见第 158 页的 “均值线、误差条和标准差线”。

t 检验 （仅当 X 因子有两个水平时才可用。）生成 *t* 检验报表（假定方差不等）。请参见第 155 页的 “t 检验报表”。

均值分析方法 提供五个执行均值分析 (ANOM) 过程的命令。提供用于比较均值、方差和极差的命令。请参见第 158 页的 “均值分析方法”。

比较均值 提供多重比较方法来比较一系列组均值。请参见第 161 页的 “比较均值”。

非参数 提供组位置的非参数比较。请参见第 167 页的 “非参数检验”。

不等方差 执行组方差是否相等的四个检验。还提供 Welch 检验，它是组内方差不相等时用于比较均值的方差分析检验。请参见第 173 页的 “不等方差”。

等价性检验 检验差异是否小于指定的阈值。您可以为检验中的方差假设指定使用合并方差或不等方差。请参见第 175 页的 “等价性检验”。

稳健 提供两种方法用来降低离群值对数据的影响。请参见第 176 页的 “稳健”。

功效 提供有关给定假设检验的统计功效计算和其他详细信息。请参见第 177 页的 “功效”。

“功效详细信息”窗口和报表也在 “拟合模型”平台中显示。有关进一步的讨论和功效计算的示例，请参见 《拟合线性模型》。

设置 α 水平 可以从最常见的 alpha 水平中选择一个选项或使用**其他**选项指定任意水平。更改 alpha 水平将导致以下操作：

- 重新计算置信限
- 调整图上的均值菱形（若显示）
- 修改报表中的置信上限值和置信下限值
- 更改所有 “比较均值”报表的临界值和比较环
- 更改所有 “非参数多重比较”报表的临界值

正态分位数图 提供以下选项来绘制每个组中数据的分位数：

标绘 “分位数-实际值”图 向单因子分析图追加分位数图。分位数图共享响应变量的单因子分析垂直轴。每组的累积概率位于水平轴上。该图显示分类 X 因子每个水平内计算出的分位数。

标绘 “实际值-分位数”图 生成分位数图，在水平轴上显示响应变量，在垂直轴上显示累积概率。该图显示分类 X 因子每个水平内计算出的分位数。

拟合线 显示或隐藏与 X 变量每个水平的数据拟合的参考线。

正态分位数标签 在 “正态分位数”图的右垂直轴内和 “分位数-实际值”图的上轴内显示或隐藏标签。标签代表正态分位数尺度。

CDF 图 绘制 “单因子”报表中所有组的累积分布函数。请参见第 178 页的 “CDF 图”。

密度 比较各个组的密度。请参见第 178 页的 “密度”。

匹配列 指定一个匹配变量来执行匹配模型分析。当您的 “单因子”分析中的数据来自匹配的（配对的）数据，如不同组中的观测来自同一参与者时，可以使用该选项。

图现在包含连接匹配点的匹配线。请参见第 179 页的 “匹配列”。

保存 将以下量作为当前数据表中的新列保存：

保存残差 保存响应变量减去因子变量每个水平内响应变量的均值所得的值。

保存标准化值 保存因子变量每个水平内计算出的响应变量的标准化值。这是除以每个水平内的标准差的中心化响应。

保存正态分位数 保存分类因子变量每个水平内计算出的正态分位数值。

保存预测值 保存因子变量每个水平对应的响应变量的预测均值。

显示选项 在图中添加或删除元素。请参见第 152 页的 “显示选项”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

显示选项

使用 “显示选项”，您可以在单因子数据图中添加或删除元素。某些选项可能不显示，除非它们相关。

所有图形 显示或隐藏所有图形。

点 显示或隐藏图上的数据点。

箱线图 显示或隐藏每个组的离群值箱线图。有关示例，请参见第 205 页的“执行单因子分析”。

均值菱形 在每组均值处各画一条水平线，其宽度与其水平轴宽度成正比。均值菱形顶部和底部的点分别显示每个组的 95% 置信上限和置信下限点。请参见第 157 页的 ““均值菱形”和“X 轴成比例””。

均值线 在每个组的均值位置绘制一条线。请参见第 158 页的“均值线、误差条和标准差线”。

均值置信区间线 在每个组的 95% 置信上限和置信下限位置绘制线。

均值误差条 标识每个组的均值并显示高于和低于均值一个标准误差的误差条。请参见第 158 页的 “均值线、误差条和标准差线”。

总均值 在图上绘制 *Y* 变量的总体均值。

标准差线 显示高于和低于每个组均值一个标准差的线。请参见第 158 页的 “均值线、误差条和标准差线”。

比较环 显示或隐藏比较环。仅当选择一个**比较均值**选项时，该选项才可用。请参见第 208 页的 “比较环”。有关示例，请参见第 205 页的 “执行单因子分析”。

连接均值 使用直线连接组均值。

均值的均值 在组均值的均值位置绘制一条线。

X 轴成比例 与每个水平的样本大小成正比调整在 x 轴上的跨度。请参见第 157 页的 ““均值菱形”和“X 轴成比例””。

散布点 把点散布到整个区间那么宽。

随机散布点 在同一 y 值上叠加的两个点之间增加小的间距。点的水平调整值变动范围为 0.375 到 0.625，服从 $4*(\text{Uniform}(0,1)-0.5)^5$ 分布。

匹配线 （仅当选择**匹配列**选项时才显示。）连接匹配点。

匹配虚线 （仅当选择**匹配列**选项时才显示。）绘制虚线来连接表中缺失单元格的单元格均值。使用双因子方差分析模型获取用作线的端点的这些值。

直方图 在原始图的右侧绘制并排直方图。

稳健均值线 （仅当选择 “稳健” 选项时才显示。）在每个组的稳健均值位置绘制一条线。

图例 为 “正态分位数图”、“CDF 图” 和 “密度” 选项显示图例。

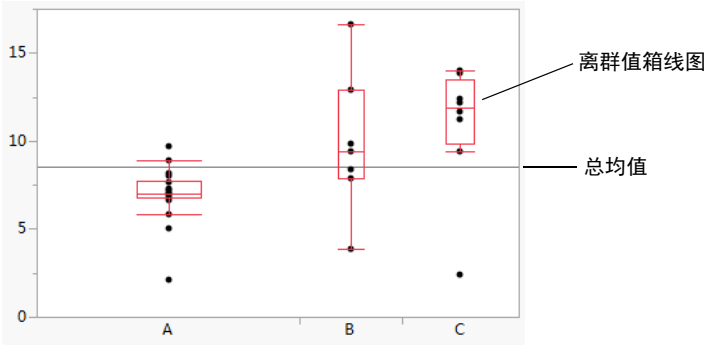
分位数

“分位数” 报表列出 X 因子变量每个水平的所选百分位数。中位数为第 50 百分位数，第 25 和第 75 百分位数称为**四分位数**。

分位数选项将以下元素添加到图中：

- 表示 Y 变量的总体均值的总均值
- 汇总每个因子水平的点分布的离群值箱线图

图 6.6 离群值箱线图和总均值



注意：要隐藏这些元素，请点击 “单因子分析” 旁边的红色小三角，然后选择**显示选项** > **箱线图**或**总均值**。

离群值箱线图

离群值箱线图是数据分布的图形汇总。请注意关于离群值箱线图的以下方面（图 6.7）：

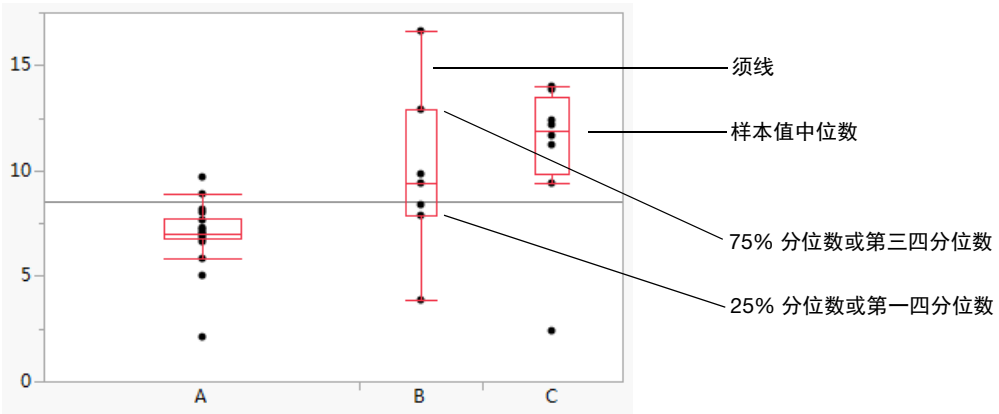
- 箱中的水平线表示样本值中位数。
- 箱的两端表示第 75 和第 25 分位数，也分别称为第三和第一四分位数。
- 第一和第三四分位数之间的差值称为**四分位间距**。
- 每个箱有从两端延伸的线，有时也称为**须线**。须线从箱的两端延伸到如下距离内的最远数据点：

  第三四分位数 + 1.5*(四分位间距)

  第一四分位数 - 1.5*(四分位间距)

  若数据点未达到计算范围两端，则由这些数据点的高值和低值（不包括离群值）来确定须线。

图 6.7 离群值箱线图示例



“均值 / 方差分析”和“均值 / 方差分析 / 合并的 t”

均值 / 方差分析选项执行方差分析。若 X 因子正好包含两个水平，则该选项显示为**均值 / 方差分析 / 合并的 t**。除了其他报表外，还显示假定合并的（或相等）方差的合并的 *t* 检验报表。

将均值菱形添加到单因子图 请参见第 152 页的“显示选项”和第 157 页的“‘均值菱形’和‘X 轴成比例’”。

报表 请参见第 155 页的“‘拟合汇总’报表”、第 156 页的“‘方差分析’报表”、第 156 页的“‘单因子方差分析均值’报表”、第 155 页的“t 检验报表”和第 157 页的“‘区组均值’报表”。

- 仅当选择**均值/方差分析/合并的 t** 选项时才显示 *t* 检验报表。
- 仅当在启动窗口中指定了“区组”变量时才显示“区组均值”报表，并且区组和 X 因子水平的每个组合中的计数相等。

“拟合汇总”报表

“单因子”平台中的 “拟合汇总”报表显示单因子方差分析的汇总信息。

R 方 拟合每个因子水平的均值所能解释变异的比例用 R 方来测量。剩余变异由随机误差造成。若拟合组均值可解释所有变异 （无误差），则 R^2 值为 1。 R^2 为 0 表示拟合作为一个预测模型所起的作用与总体响应均值一样差。请参见第 210 页的 “‘拟合汇总’报表”的统计详细信息。

R^2 也称为**决定系数**。

注意：较低的 R 方值表明可能有一些模型中不包含的变量可以解释无法解释的变异。不过，数据可能受大量的内在变异影响，即便是有用的方差分析模型也可能具有较低的 R 方值。阅读您的研究领域中的文献材料，了解有关典型 R 方值的信息。

调整 R 方 通过在计算中使用自由度，调整 R^2 使该值在带有不同数量参数的各个模型之间更具可比性。请参见第 210 页的 “‘拟合汇总’报表”的统计详细信息。

均方根误差 估计随机误差的标准差。它是在 “方差分析”报表中显示的**误差**的均方平方根。

响应均值 响应变量的总体均值 （算术平均值）。

观测数 （或权重和） 估计拟合中使用的观测数。若使用权重，则它是权重和。请参见第 210 页的 “‘拟合汇总’报表”的统计详细信息。

t 检验报表

有两种 *t* 检验类型：等方差和不等方差。

- 若选择**均值/方差分析/合并的 t** 选项，将显示 “合并的 t 检验”报表。该 *t* 检验假定方差相等。
- 若从红色小三角菜单中选择 **t 检验**选项，将显示 *t* 检验报表。该 *t* 检验假定方差不等。

“t 检验”报表包含以下列：

t 检验图 显示均值差值的抽样分布，假定原假设为真。垂直红线是均值的实际差值。阴影面积对应于 *p* 值。

差值 显示两个 X 水平之间的估计差值。在图中，差值显示为比较两个水平的红线。

差值标准误差 显示差值的标准误差。

差值置信上限 显示差值的置信上限。

差值置信下限 显示差值的置信下限。

置信 显示置信水平 (1-alpha)。若要更改置信水平，请从平台红色小三角菜单的**设置 α 水平**命令选择新的 alpha 水平。

t 比 *t* 统计量的值。

自由度 *t* 检验中使用的自由度。

概率 > |t| 与双侧检验相关的 *p* 值。

概率 >t 与右侧检验相关的 *p* 值。

概率 < t 与左侧检验相关的 *p* 值。

“方差分析”报表

“方差分析”报表将样本的总变异分为两个部分。两个均方之比为 *F* 比。若与 *F* 比相关的概率很小，则模型在统计上比总体响应均值的拟合效果好。

注意：若指定了**区组**列，则“方差分析”报表包含**区组**变量。

源 列出变异的三个来源。这些来源分别是模型源、**误差**和**校正总和**。

自由度 记录每个变异来源的相关自由度 （简写为 DF）：

- **校正总和**的自由度为 N - 1，其中 N 为分析中使用的总观测数。
- 若 X 因子有 k 个水平，则模型有 k - 1 个自由度。

误差自由度为**校正总和**自由度与**模型**自由度的差值 （即 N - k）。

平方和 记录每个变异来源的平方和 （简写为 SS）：

- 每个响应与总体响应均值差值的平方和总和（**校正总和**）。**校正总和**平方和是用于与所有其他模型进行比较的基本模型。
- 每个点与相应组均值的距离的平方和。这是拟合方差分析模型后剩余无法解释的**误差**（残差）平方和。

总平方和减去误差平方和得到归因于模型的平方和。它指示模型可在多大程度上解释总变异。

均方 由平方和除以相关的自由度计算而来：

- **模型**均方仅在组均值相等的假设前提下估计误差的方差。
- **误差**均方独立于模型均方估计误差项的方差，没有任何模型假设条件。

F 比 模型均方除以误差均方。若组均值相等 （两者之间没有真正的差异）的假设为真，则使用误差的均方和模型的均方来估计误差方差。两者之比服从 *F* 分布。若方差分析模型导致总变异显著减小，则 *F* 比高于预期值。

概率 >F 在总体组均值之间没有差异的情况下，得到 *F* 值大于计算出的值的概率。观测到的显著性概率 0.05 或更小值通常被视为组均值之间存在差异的证明。

“单因子方差分析均值”报表

“单因子方差分析均值”报表汇总名义型或有序型因子每个水平的响应信息。

水平 列出 X 变量的水平。

数字 列出每个组的观测数。

均值 列出每个组的均值。

标准误差 列出组均值的标准差估计值。在假定响应方差在每个水平中相同的前提下估计该标准误差。它是“拟合汇总”报表中所示的均方根误差除以用于计算组均值的值个数的平方根所得的值。

95% 下限和 95% 上限 列出组均值的 95% 下限和 95% 上限置信区间。

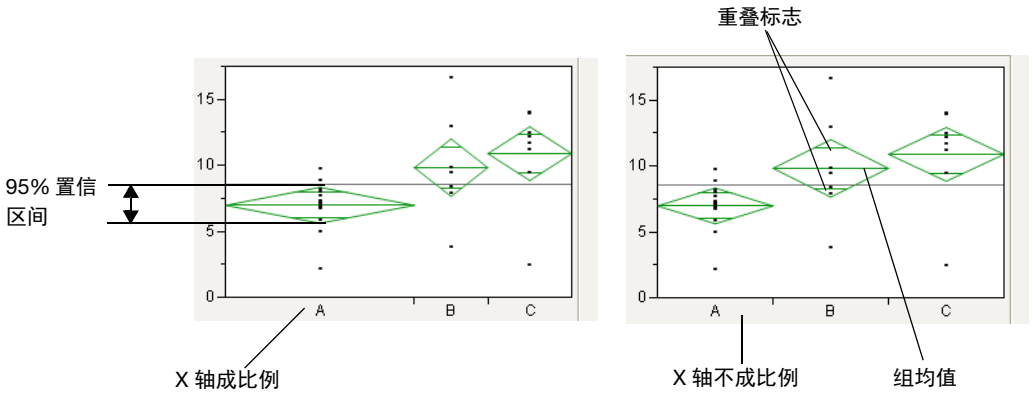
“区组均值”报表

若在启动窗口中指定了“区组”变量，则**均值/方差分析**和**均值/方差分析/合并的 t** 命令可生成“区组均值”报表。该报表显示每个区组的均值和每个区组中的观测数。

“均值菱形”和“X 轴成比例”

均值菱形说明样本均值和置信区间。

图 6.8 “均值菱形”选项和 “X 轴成比例”选项的示例



请注意以下事项：

- 每个菱形的顶部和底部表示每个组的 (1-alpha)x100 置信区间。置信区间的计算假定各个观测的方差相等。因此，菱形的高度与组中观测数的平方根倒数成比例。
- 若选择 **X 轴成比例**选项，则每组沿水平轴的水平延伸程度（菱形的水平大小）与 X 变量每个水平的样本大小成比例。因此，越窄的菱形通常会越高，因为更少的数据点导致了更宽的置信区间。
- 横穿每个菱形中部的均值线表示组均值。
- 重叠标志显示为组均值上方和下方的线。对于具有相等样本大小的组，重叠标志指示两个组均值在给定的置信水平上没有显著差异。重叠标志计算为组均值 $\pm(\sqrt{2})/2 \times$ 置信区间 $/2$。若一个菱形中的重叠标志与另一菱形中的重叠标志相比更接近该菱形的均值，表示这两个组在给定的置信水平上没有差异。

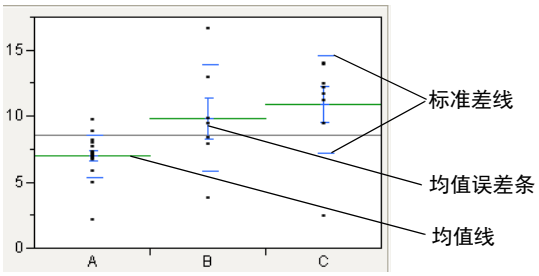
- 从平台菜单中选择**均值/方差分析/合并的 t** 或**均值/方差分析**选项时，自动显示均值菱形。但是，您可以通过从红色小三角菜单中选择**显示选项 > 均值菱形**来随时显示或隐藏它们。

均值线、误差条和标准差线

通过选择**显示选项 > 均值线**可显示均值线。均值线指示 X 变量每个水平的响应均值。

从红色小三角菜单中选择**均值与标准差**选项时，显示均值误差条和标准差线。 要单独启用或禁用每个选项，请选择**显示选项 > 均值误差条**或**标准差线**。

图 6.9 均值线、均值误差条和标准差线



均值分析方法

均值分析 (ANOM) 是用于均值、比例或方差的多重比较方法。在以下情况下您可能要使用这些 ANOM 方法：

- 检验任何组均值在统计上是否与总体（样本）均值有差异
- 检验任何组标准差在统计上是否与均方根误差 (RMSE) 有差异
- 检验任何组极差在统计上是否与极差的总均值有差异

注意：在“列联”平台内，当响应有两个类别时，您可以使用**比例均值分析**。请参见第 215 页的“列联分析”一章。

有关 ANOM 方法的说明以及 JMP 执行 ANOM 的方法，请参见 Nelson et al. (2005) 一书。

注意：若在启动窗口中指定“区组”变量，并且区组和 X 变量水平的每一组合的计数相等，则 ANOM 图是唯一可用的均值分析图。若在启动窗口中指定“区组”变量但计数不等，则没有可用的“均值分析”选项。

位置均值分析

您可以使用以下选项检验各组是否有共同的均值或中心值：

- ANOM
- 变换秩 ANOM

ANOM

使用 ANOM 可比较组均值与总均值。该方法假定您的数据近似服从正态分布。请参见第 180 页的“均值分析图表示例”。

变换秩 ANOM

这是 ANOM 分析的非参数版本。若您的数据明显不服从正态分布并且不能转换为正态分布，则使用该方法。“变换秩 ANOM”比较各组变换秩均值与变换秩总均值。ANOM 检验涉及对变换观测应用常用的 ANOM 过程和临界值。

变换秩

假定有 n 个观测。变换观测计算如下：

- 按最小到最大的顺序排列所有观测，将结值也考虑在内。对于结值观测，为每个观测分配它们所共享的秩区组的平均值。
- 用 $R_1, R_2, \ldots, R_n$ 来表示各个秩。
- 与第 i 个观测对应的变换秩为：

$$\text{变换 } R_i = Normal\ Quantile\left[\left(\frac{R_i}{2n+1}\right) + 0.5\right]$$

ANOM 过程应用到值“变换 R_i”。由于秩服从均匀分布，变换秩服从折叠正态分布。请参见 Nelson et al. (2005) 一书。

尺度均值分析

您可以使用以下选项检验组内的变异齐性：

- 方差 ANOM
- 使用 Levene (ADM) 的方差 ANOM
- 极差 ANOM

方差 ANOM

使用该方法可比较组标准差（或方差）与均方根误差（或均方误差）。该方法假定您的数据近似服从正态分布。若要使用该方法，每个组必须至少有四个观测。有关“方差 ANOM”检验

的详细信息，请参见 Wludyka and Nelson (1997) 以及 Nelson et al. (2005) 一书。有关示例，请参见第 181 页的“方差均值分析图表的示例”。

使用 Levene (ADM) 的方差 ANOM

该方法提供了稳健检验，可用于比较**中位数绝对偏差** (ADM) 的组均值与 ADM 总均值。若您怀疑您的数据不服从正态分布，并且不能转换为正态分布，则使用“使用 Levene (ADM) 的方差 ANOM”。“使用 Levene (ADM) 的方差 ANOM”是“方差 ANOM”分析的非参数版本。有关“使用 Levene (ADM) 的方差 ANOM”检验的详细信息，请参见 Levene (1960) 或 Brown and Forsythe (1974)。

极差 ANOM

使用该检验可比较组极差与组极差均值。这是以极差作为散布测度进行的尺度差异检验。请参见 Wheeler (2003)。

注意：“极差 ANOM”仅可用于平衡设计和特定组大小。请参见第 160 页的“‘极差 ANOM”检验的局限性”。

“极差 ANOM”检验的局限性

与其他 ANOM 决策限不同，“极差 ANOM”图的决策限仅使用表临界值。出于此原因，“极差 ANOM”仅可用于以下情况：

- 大小相同的组
- 具体为以下大小的组：2–10、12、15 和 20
- 2 到 30 之间的组数目
- alpha 水平为 0.10、0.05 和 0.01

均值分析图表

每个“均值分析方法”选项向报表窗口添加一个决策图，该图显示以下信息：

- 上决策限 (UDL)
- 下决策限 (LDL)
- 位于决策限之间的水平（中心）线；中心线位置由图表类型决定：
  - ANOM：总均值
  - 变换秩 ANOM：变换秩的总均值
  - 方差 ANOM：均方根误差（以方差为尺度时为均方误差）
  - 使用 Levene (ADM) 的方差 ANOM：总均值绝对偏差
  - 极差 ANOM：组极差均值

若某个组的标绘统计量落在决策限之外，则该检验指示该组的统计量与所有组的这一统计量的总平均值之间存在统计差异。

“均值分析”选项

每个“均值分析方法”选项向报表窗口添加一个“均值分析”红色小三角菜单。

设置 Alpha 水平 从最常见的 alpha 水平选择一个选项或使用**其他**选项指定任意水平。更改 alpha 水平会修改上决策限和下决策限。

注意：对于极差 ANOM，只可以选择 0.10、 0.05 和 0.01。

显示汇总报表 报表基于 “均值分析”方法：

- 对于 ANOM，创建显示组均值和决策限的报表。
- 对于变换秩 ANOM，创建显示组变换秩均值和决策限的报表。
- 对于方差 ANOM，创建显示组标准差（或方差）和决策限的报表。
- 对于使用 Levene (ADM) 的方差 ANOM，创建显示组 ADM 均值和决策限的报表。
- 对于极差 ANOM，创建显示组极差和决策限的报表。

以方差尺度绘图 （仅可用于**方差 ANOM**。）将垂直轴的尺度从标准差更改为方差。

显示选项 包含以下用于定制显示的选项：

显示决策限 显示或隐藏决策限线。

显示决策限着色 显示或隐藏决策限着色。

显示中心线 显示或隐藏中心线统计量。

点选项：显示针 显示针。这是默认选项。**显示连接的点**可显示连接各组均值的线。**仅显示点**仅显示表示各组均值的点。

比较均值

使用“比较均值”选项可以执行组均值的多重比较。方法包括 Tukey-Kramer 所有对检验和带控制组的 Dunnett 多重比较检验。所有这些方法将合并的方差估计值用于均值。每个“比较均值”选项（每对逐步，Newman-Keuls 方法除外）都在图旁边添加比较环并向报表窗口添加特定报表。有关比较环的详细信息，请参见第 162 页的“使用比较环”。

注意：比较均值的另一个方法是 ANOM。请参见第 158 页的“均值分析方法”。

选项	说明	非参数菜单选项
每对，Student t	使用 Student *t* 检验计算单个配对比较。若执行很多配对检验，则没有推断保护。因此，各个假设检验的 alpha 大小（第一类错误率）高于单个检验的 alpha 大小。请参见第 164 页的“每对，Student t”。	非参数 > 非参数多重比较 > 对每对执行 Wilcoxon 检验
所有对，Tukey HSD	显示针对各均值的所有差异大小的检验。这是 Tukey 或 Tukey-Kramer HSD（真实显著性差异）检验（Tukey 1953；Kramer 1956）。若样本大小相同，该检验为精确 alpha 水平检验；若样本大小不同，则为保守检验 (Hayter 1984)。请参见第 164 页的“所有对，Tukey HSD”。	非参数 >非参数多重比较 >对所有对执行 Steel-Dwass 检验
带最佳组的 Hsu MCB 检验	检验均值是否小于未知最大值或大于未知最小值。这是 Hsu MCB 检验（Hsu 1996；Hsu 1981）。请参见第 164 页的“带最佳组的 Hsu MCB 检验”。	无
带控制组的 Dunnett 检验	检验均值是否与控制组的均值有差异。这是 Dunnett 检验 (Dunnett 1955)。请参见第 165 页的“带控制组的 Dunnett 检验”。	非参数 > 非参数多重比较 > 带控制组的 Steel 检验
每对逐步，Newman-Keuls	使用逐步过程中的学生化范围检验来检验各均值之间是否有差异。这是 Newman-Keuls 或 Student-Newman-Keuls 方法 (Keuls, 1952)。该检验没有 Tukey HSD 检验更保守。请参见第 166 页的“每对逐步，Newman-Keuls”。	无

注意：若指定了**区组**列，则对进行了区组均值调整的数据执行多重比较方法。

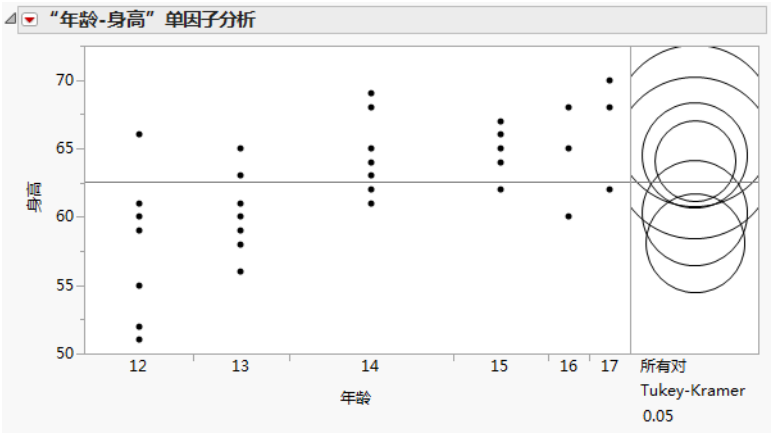
有关显示所有这些检验的示例，请参见第 188 页的“示例：比较四个“比较均值”检验”。

使用比较环

每个多重比较检验（每对逐步，Newman-Keuls 方法除外）都从**比较环**图开始，该图是组均值比较的直观表示。图 6.10 显示“所有对，Tukey HSD”方法的比较环。其他比较检验增大或减小环的半径。

注意：要永久隐藏比较环图，请选择**文件 > 首选项 > 平台 > 单因子**，然后取消选择**比较环**选项。

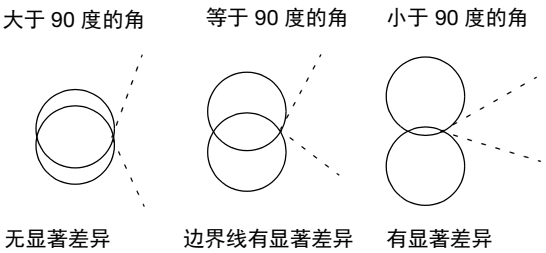
图 6.10 组均值的直观比较



通过检查比较环的交叉，直观比较每对组均值。外角的交叉说明组均值是否有显著差异（图 6.11）。

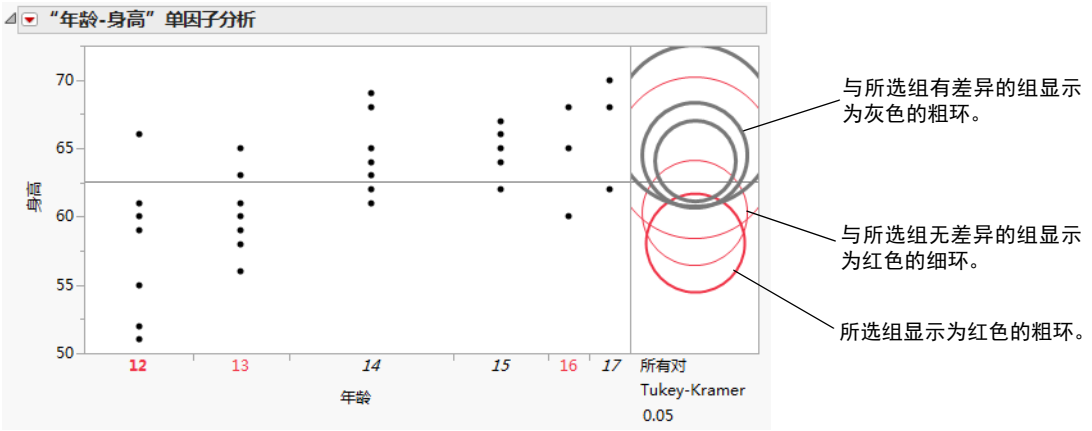
- 有显著差异的均值的环不交叉或轻微交叉，因此交叉的外角小于 90 度。
- 若环以大于 90 度的角交叉或环嵌套，则均值没有显著差异。

图 6.11 交叉角和显著性



若交叉角接近 90 度，您可以通过点击比较环以选择它，验证均值是否有显著差异（图 6.12）。要取消选择环，请在环外的空白处点击。

图 6.12 突出显示比较环



相关信息

- 第 208 页的"比较环"

每对，Student t

每对，Student t 检验显示组水平的每对 Student *t* 检验并且只检验个体比较。有关该检验的示例，请参见第 181 页的"'每对，Student t'检验的示例"。

所有对，Tukey HSD

所有对，Tukey HSD 检验（也称为 Tukey-Kramer）保护配对的所有组合的显著性检验，并且 HSD 区间变得大于 Student *t* 配对 LSD。从图形上看，比较环变大，差异变得不显著。

Q 统计量计算如下：q* = (1/sqrt(2)) * q，其中 q 为学生化范围分布的必需百分位数。有关该检验的示例，请参见第 183 页的"'所有对，Tukey HSD'检验的示例"。

带最佳组的 Hsu MCB 检验

带最佳组的 Hsu MCB 检验确定给定水平的均值是超过其余水平的最大均值还是小于其余水平的最小均值。请参见 Hsu (1996)。有关该检验的示例，请参见第 185 页的"'带最佳组的 Hsu MCB 检验'的示例"。

Hsu MCB 检验的分位数随分类变量的水平而变化。除非样本大小在不同水平之间是相同的，否则比较环方法并不精确。比较环的半径是水平的标准误差与最大分位数值的乘积。使用检验的 *p* 值获取显著性差异的精确评估。请参见第 165 页的"与最大值和最小值的比较"。

注意：不被 MCB 视为最大值或最小值的均值也不包含在可能的最大或最小均值的 Gupta 所选子集 (1965) 中。

置信分位数

该报表提供分类变量各水平的分位数。这些分位数对应于指定的 Alpha 值。

与最大值和最小值的比较

该报表显示单侧 Dunnett 检验的 *p* 值。对于除最佳之外的每个水平，给出的 *p* 值都针对下面的检验：将样本最佳水平的均值与视为控制（可能是最佳）水平的其余每个水平的均值进行比较。通过将第二个样本最佳水平的均值与视为控制的样本最佳水平的均值进行比较，来获取样本最佳水平的 *p* 值。

该报表显示三列。

水平 分类变量的水平。

与最大值比较 p 值 对于分类变量的每个水平，该列提供以下检验的 *p* 值：该水平的均值超过其余水平的最大均值。使用该列中的检验可筛选出均值显著小于或等于（未知）最大真实均值的水平。

与最小值比较 p 值 对于分类变量的每个水平，该列提供以下检验的 *p* 值：该水平的均值小于其余水平的最小均值。使用该列中的检验可筛选出均值显著小于或等于（未知）最大真实均值的水平。

LSD 阈值矩阵

显示的第一个报表针对最大值，第二个报表针对最小值。

对于**最大值**报表，一个列显示行均值减去列均值减去 LSD。若值为正数，则行均值显著大于该列的均值，且该列的均值不是最大值。

对于**最小值**报表，一个列显示行均值减去列均值加上 LSD。若值为负数，则行均值显著小于该列的均值，且该列的均值不是最小值。

带控制组的 Dunnett 检验

带控制组的 Dunnett 检验针对控制组的均值比较一系列均值。它生成的 LSD 介于 Student *t* 和 Tukey-Kramer LSD 之间，因为已调整它们的大小来限制均值的比较，其比较次数介于两者之间。有关该检验的示例，请参见第 186 页的““带控制组的 Dunnett 检验”的示例”。

在 Dunnett 报表中，显示 |*d*| 分位数，可以通过类似 Student *t* 统计量的方式使用它。LSD 阈值矩阵显示差值的绝对值减去 LSD。若值为正数，则它的均值减去控制组均值后大于 LSD，因此具有显著差异。

每对逐步，Newman-Keuls

每对逐步， Newman-Keuls 检验使用迭代逐步过程比较样本均值。在每次迭代中， Tukey HSD 检验用于检验两组均值之间的差异。有关该检验的示例，请参见第 187 页的 “ “每对逐步，Newman-Keuls”检验的示例”。

警告：Newman-Keuls 检验不控制族状误差率。解释该过程的结果时务必要小心。

以下过程用于检验 *J* 组均值：

定义以下公式：

J = 组数 （按组均值的升序排序）

n = 观测数

d = 自由度，计算为 *N* - *J*

i = 比较中涉及的最小组均值的指数

j = 比较中涉及的最大组均值的指数

k = 该过程期间任何比较中的 *j* 的最小值。

在过程开始，设置 *i* = 1， *j* = *J*，且 *k* = 2。

1. 为组 *i* 和 *j* 执行 Tukey HSD 检验，其中，用于查找适合的分位数的组数等于 *j* - i + 1。
   - 若检验显著，则认为组 *i* 和 *j* 有显著差异。将 *j* 减 1。

     若这导致 *j* 小于 *k*，则将 *i* 加 1，设置 *k* = max(*i*, *j*) + 1，设置 *j* = *J*，并且继续第 2 步。

     若这导致 *j* 大于等于 *k*，则继续第 2 步。
   - 若检验不显著，则认为组 *i* 和 *j* 不具有显著差异。将 *i* 加 1，设置 *k* = *j* + 1，设置 *j* = *J*，并且继续第 2 步。
2. 根据 *k* 的值确定过程是继续还是停止。
   - 若 *k* 大于 *i* 且 *k* 小于等于 *J*，则重复第 1 步。
   - 若 *k* 小于等于 *i* 或 *k* 大于 *J*，则停止该过程。其余任何未检验的范围均被视为没有显著差异。

用于 Tukey HSD 的分位数对于每个检验都是不同的，并且基于待检验的排序均值之间的组均值数目。在 Newman-Keuls 报表中，“考虑的最小分位数”（标记为“最小 q*”）是在上述过程中使用的最小学生化范围分位数除以 2 的平方根所得的结果。

检验结果在“连接字母报表”中报告。

有关 Newman-Keuls 检验的详细信息，请参见 Howell (2013)。

注意：当您使用**每对逐步，**Newman-Keuls 检验时，不会将均值环添加到“比较环”图形。这是因为每次比较都具有不同的截止值，该值取决于待检验的两个均值之间的均值数目。因此，每个环的大小将不同。

“比较均值”选项

“均值比较”报表包含一个带有以下选项的红色小三角菜单：

差值矩阵 显示或隐藏所有均值差值的表。

置信分位数 显示或隐藏用于均值比较过程的临界值和显著性水平 (α)。

LSD 阈值矩阵 （不可用于 Newman-Keuls 检验。）显示或隐藏均值减去这些均值的最小显著性差异后的配对差值矩阵。正值表示均值对有显著差异。

连接字母报表 （仅可用于 Student *t*、Tukey HSD 和 Newman-Keuls 检验。）显示或隐藏传统的字母编码报表，在该报表中具有不同字母的均值被认为有显著差异。

差值排序报表 （仅可用于 Student *t* 和 Tukey HSD 检验。）显示或隐藏所有配对正差值、差值的标准误差、置信区间、*p* 值和叠加了置信区间的差值幅度图。当置信区间未完全包含相应差值直条时，它表明均值彼此有显著差异。

详细比较报表 （仅可用于 Student *t* 检验。）显示或隐藏每个比较的详细报表。每个部分显示两个水平的差值、标准误差和置信区间、*t* 比、*p* 值和自由度。说明比较结果的图显示在每个报表的右侧。

非参数检验

当通常假设正态分布的方差分析不适用时，非参数检验很有用。“非参数”选项提供若干方法来检验各组间的均值或中位数相等的假设。非参数多重比较过程也可用于控制配对比较的总误差率。非参数检验使用响应秩（称为秩得分）函数。请参见 Hajek (1969) 和 SAS Institute Inc. (2020a)。

注意：若在启动窗口中指定“区组”变量，并且区组和 X 变量水平的每一组合的计数相等，则 Friedman 秩检验是唯一可用的非参数选项。若在启动窗口中指定“区组”变量但计数不等，则没有可用的非参数选项。

请注意以下几点：

- 对于 Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验，若 X 因子具有两个以上的水平，则执行近似卡方分布的单因子检验。
- 若指定“区组”列，将使用区组均值对中心化的数据值执行非参数检验（Friedman 秩检验除外）。

Wilcoxon 检验 基于 Wilcoxon 秩得分执行检验。Wilcoxon 秩得分是数据的简单秩。Wilcoxon 检验是对服从 Logistic 分布的误差的最强大秩检验。若因子具有两个以上的水平，则执行 Kruskal-Wallis 检验。有关该报表的信息，请参见第 168 页的“Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验报表”。有关示例，请参见第 189 页的“非参数 Wilcoxon 检验的示例”。

Wilcoxon 检验也称为 Mann-Whitney 检验。

中位数检验 基于中位数秩得分执行检验。秩中位数得分是 1 或 0，取决于秩是在中位数秩之上还是之下。中位数检验是对服从双指数分布的误差的最强大秩检验。有关该报表的信息，请参见第 168 页的 “Wilcoxon、中位数、 Van der Waerden 和 Friedman 秩检验报表”。

Van der Waerden 检验 基于 Van der Waerden 秩得分执行检验。Van der Waerden 秩得分是数据的秩除以 1 加上观测值数所得的商，这些观测值通过应用正态分布函数的反函数转换为正态得分。Van der Waerden 检验是服从正态分布的误差的最强大秩检验。有关该报表的信息，请参见第 168 页的“Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验报表”。

Kolmogorov Smirnov 检验 （仅当 X 因子有两个水平时才可用。）基于经验分布函数执行检验，它检验响应的分布在不同的组中是否相同。给出近似检验和精确检验结果。有关该报表的信息，请参见第 169 页的 ““Kolmogorov-Smirnov 双样本检验”报表”。

Friedman 秩检验 （仅当在启动窗口指定了 “区组”变量时才可用。）基于 Friedman 秩得分执行检验。 Friedman 秩得分是分区组变量的每个水平内数据的秩。该检验的参数版本是重复测量方差分析。有关该报表的信息，请参见第 168 页的 “Wilcoxon、中位数、 Van der Waerden 和 Friedman 秩检验报表”。

注意：每个区组内的观测数必须相等。

JMP PRO **精确检验** 提供执行精确 Wilcoxon、中位数、 Van der Waerden 和 Kolmogorov-Smirnov 检验的选项。仅当 X 因子具有两个水平时这些选项才可用。同时提供近似检验和精确检验的结果。

有关该报表的信息，请参见第 169 页的“ 双样本，精确检验”。要查看涉及 Wilcoxon 精确检验的示例，请参见第 189 页的 “非参数 Wilcoxon 检验的示例”。

Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验报表

对于每个检验，该报表都显示描述性统计量和检验结果。检验结果显示在 “单因子检验，近似卡方分布”报表中；若 X 变量恰好有两个水平，则还会显示一个“双样本检验，近似正态分布”报表。描述性统计量如下：

水平 X 的水平。

计数 每个水平的频数。

得分和 每个水平的秩得分和。

期望得分 在分类水平无差别的原假设下的期望得分。

得分均值 每个水平的均值秩得分。

(Mean-Mean0)/Std0 标准化得分。Mean0 是原假设下期望的均值得分。Std0 是原假设下期望的得分和的标准差。原假设是组均值或中位数位于各组的同一位置。

双样本检验，近似正态分布

若 X 恰好有两个水平，则还会显示一个 “双样本检验，近似正态分布”报表。该报表提供以下信息：

S 给出观测数较少的水平的秩得分和。

Z 给出近似正态分布检验的检验统计量。请参见第 212 页的 “双样本正态近似”。

概率 >|Z| 基于标准正态分布，给出正态近似检验的 *p* 值。

单因子检验，近似卡方分布

该报表给出位置卡方检验的结果。请参见 Conover (1999)。

卡方 给出卡方检验统计量的值。请参见第 213 页的 “单因子卡方近似”。

自由度 给出检验的自由度。

概率 > 卡方 提供检验的 *p* 值。 *p* 值基于自由度等于 X 水平数减 1 的卡方分布。

JMP PRO 双样本，精确检验

若数据稀疏、偏斜或结值很多，精确检验可能比基于渐近行为的近似检验更适合。若 X 恰好有两个水平， JMP Pro 将计算精确检验的检验统计量。选择**非参数 > 精确检验**，然后选择您所需的检验。随即显示一个 “双样本 : 精确检验”报表。该报表提供以下信息：

S 给出较小组中观测的秩得分和。若 X 的两个水平有相同的观测数，则 S 值对应于值排序中 X 的最后一个水平。

概率 ≤ S 给出检验的单侧 *p* 值。

概率 ≥ |S-Mean| 给出检验的双侧 *p* 值。

“Kolmogorov-Smirnov 双样本检验”报表

仅当 X 恰好有两个水平时 Kolmogorov-Smirnov 检验才可用。该报表显示描述性统计量和检验结果。描述性统计量如下：

水平 X 的两个水平。

计数 每个水平的频数。

经验分布函数最大值 对于 X 的某个水平，给出两个经验累积分布函数 (EDF) 之间差值最大时该 X 值处的 EDF 的值。对于名为 “合计”的行，给出两个经验累积分布函数 (EDF) 之间差值最大时该 X 值处的合并 EDF （整个数据集的 EDF）的值。

均值偏差最大值 对于每个水平，给出通过以下步骤获得的值：

– 计算给定水平最大值处的 EDF 与合并数据集（合计）最大值处的 EDF 之间的差值。

– 用该水平中的观测数的平方根（作为“计数”提供）乘以该差值。

Kolmogorov-Smirnov 渐近检验

该报表给出该检验的详细信息。

KS Kolmogorov-Smirnov 统计量计算如下：

$$\mathrm{KS} = \max_j \sqrt{\frac{1}{n}\sum_i n_i (F_i(x_j) - F(x_j))^2}$$

该公式使用以下符号：

– x_j, $j = 1, \ldots, n$ 为观测

– n_i 是 X 的第 i 个水平的观测数

– F 是合并累积经验分布函数

– F_i 是 X 的第 i 个水平的累积经验分布函数

即便 X 有两个以上水平，这一版本的 Kolmogorov-Smirnov 统计量也适用。不过请注意，仅当 X 恰好有两个水平时， JMP 才会执行 Kolmogorov-Smirnov 分析。

KSa 计算为 $\mathrm{KS}\sqrt{n}$ 的渐近 Kolmogorov-Smirnov 统计量，其中， n 是观测总数。

D=max|F1-F2| 两个水平的 EDF 之间的最大绝对偏差。这是通常用于比较两个样本的 Kolmogorov-Smirnov 统计量版本。

概率 > D 检验的 p 值。这是在水平之间没有差异的原假设下 D 超过计算值的概率。

D+ = max(F1-F2) 关于第一组的水平超过第二组的水平的备择假设的单侧检验统计量。

概率 > D+ D+ 检验的 p 值。

D-=max(F2-F1) 关于第二组的水平超过第一组的水平的备择假设的单侧检验统计量

概率 > D- D- 检验的 p 值。

JMP PRO Kolmogorov-Smirnov 精确检验

对于 Kolmogorov-Smirnov 精确检验，该报表给出与渐近检验相同的统计量，但计算的 p 值很精确。

非参数多重比较

该选项提供若干方法用于执行非参数多重比较。这些检验基于秩并控制整体实验误差率（对每对执行 Wilcoxon 检验除外）。有关这些检验的详细信息，请参见 Dunn (1964) 和 Hsu (1996)。有关报表的信息，请参见第 171 页的“非参数多重比较过程”。

非参数多重比较过程

对每对执行 Wilcoxon 检验 对每对执行 Wilcoxon 检验。该过程不控制总 alpha 水平。这是“比较均值”菜单中的**每对，** Student t 选项的非参数版本。请参见第 171 页的 “Wilcoxon 每对检验、 Steel-Dwass 所有对检验，以及带控制组的 Steel 检验”。

对所有对执行 Steel-Dwass 检验 对每对执行 Steel-Dwass 检验。这是 “比较均值”菜单中的**所有对，**Tukey HSD 选项的非参数版本。请参见第 171 页的“Wilcoxon 每对检验、Steel-Dwass 所有对检验，以及带控制组的 Steel 检验”。

带控制组的 Steel 检验 将每个水平与控制水平进行比较。这是 “比较均值”菜单中的**带控制组的 Dunnett 检验**选项的非参数版本。请参见第 171 页的“Wilcoxon 每对检验、Steel-Dwass 所有对检验，以及带控制组的 Steel 检验”。

使用联合秩的 Dunn 所有对检验 对每对执行比较，与 “对所有对执行 Steel-Dwass 检验”选项类似。 Dunn 方法对所有数据都计算秩，而不仅仅是要比较的配对。报告的 p 值反映 Bonferroni 调整。它是未调整的 p 值乘以比较次数后的结果。若调整的 p 值超过 1，则将该值报告为 1。请参见第 172 页的 “使用联合秩的 Dunn 所有对检验和使用联合秩的带控制组的 Dunn 检验”。

使用联合秩的带控制组的 Dunn 检验 将每个水平与控制水平进行比较，与“带控制组的 Steel 检验”选项类似。 Dunn 方法对所有数据都计算秩，而不仅仅是要比较的配对。报告的 p 值反映 Bonferroni 调整。它是未调整的 p 值乘以比较次数后的结果。若调整的 p 值超过 1，则将该值报告为 1。请参见第 172 页的 “使用联合秩的 Dunn 所有对检验和使用联合秩的带控制组的 Dunn 检验”。

Wilcoxon 每对检验、Steel-Dwass 所有对检验，以及带控制组的 Steel 检验

这些多重比较过程的报表提供检验结果和置信区间。对于这些检验，在通过仅组合给定比较中使用的两个水平而获得的样本中对这些观测排序。

q* 用于计算置信区间的分位数。

Alpha 计算置信区间时使用的 alpha 水平。通过从 “单因子”菜单中选择 “设置 α 水平”选项，您可以更改该置信水平。

水平 配对比较中使用的 X 变量的第一个水平。

- 水平 配对比较中使用的 X 变量的第二个水平。

均值差得分 第一个水平 （水平）中的观测的秩得分均值减去第二个水平 （- 水平）中的观测的秩得分均值，其中应用了连续校正。

用 n_1 表示第一个水平中的观测数，用 n_2 表示第二个水平中的观测数。在由这两个水平构成的样本中对这些观测排序。对结值秩求平均值。用 ScoreSum_1 表示第一个水平的秩总和，用 ScoreSum_2 表示第二个水平的秩总和。

若得分均值的差值为正，则 “得分均值差”定义如下：

$$\text{均值差得分} = (\text{ScoreSum}_1 - 0.5)/n_1 - (\text{ScoreSum}_2 + 0.5)/n_2$$

若得分均值的差值为负，则 “均值差得分”定义如下：

均值差得分 = $(ScoreSum_1 + 0.5)/n_1 - (ScoreSum_2 - 0.5)/n_2$

差值标准误差 均值差得分的标准误差。

Z 标准化检验统计量，服从在均值无差异这一原假设下的渐近标准正态分布。

p 值 基于 Z 的渐近检验的 *p* 值。

Hodges-Lehmann 位置移动的 Hodges-Lehmann 估计量。构造第一个水平中的观测减去第二个水平中的观测后的所有配对差值。Hodges-Lehmann 估计量是这些差值的中位数。“差值图”条形图显示 Hodges-Lehmann 估计值的大小。

置信下限 Hodges-Lehmann 统计量的置信下限。

注意：若组样本大小因过大而引起内存问题，则不计算。

置信上限 Hodges-Lehmann 统计量的置信上限。

注意：若组样本大小因过大而引起内存问题，则不计算。

使用联合秩的 Dunn 所有对检验和使用联合秩的带控制组的 Dunn 检验

这些比较过程基于整个数据集的观测的秩。对于使用联合秩的带控制组的 Dunn 检验，您必须选择一个控制水平。

水平 配对比较中使用的 X 变量的第一个水平。

- 水平 配对比较中使用的 X 变量的第二个水平。

均值差得分 第一个水平 （水平）中的观测的秩得分均值减去第二个水平 （- 水平）中的观测的秩得分均值，其中应用了连续校正。通过对整个样本中的观测排序获得秩。对结值秩求平均值。连续校正在第 171 页的 “均值差得分”中进行了说明。

差值标准误差 均值差得分的标准误差。

Z 标准化检验统计量，服从在均值无差异这一原假设下的渐近标准正态分布。

p 值 基于 Z 的渐近检验的 *p* 值。

不等方差

JMP 提供组方差是否相等的四个检验。若组间方差不等，则不满足方差分析的通常假设。例如，方差分析 *F* 检验无效，应使用不假定组方差相等的分析。相等方差的前三个检验隐含的概念是对构造用于测量每个组的散布程度的新响应变量执行方差分析。第四个检验为 Bartlett 检验，它类似于正态分布下的似然比检验。在启动窗口中指定“区组”变量时，“不等方差”选项不可用。

注意：用于不等方差的另一检验方法为 ANOMV。请参见第 158 页的“均值分析方法”。

以下相等方差检验可用：

O'Brien 构造因变量，使新变量的组均值等于原始响应的组样本方差。对 O'Brien 变量的方差分析实际是对组样本方差的方差分析 （O'Brien 1979； Olejnik and Algina 1987）。

Brown-Forsythe 显示方差分析的 *F* 检验，其中响应是每个观测与组中位数差值的绝对值 (Brown and Forsythe 1974)。

Levene 显示方差分析的 *F* 检验，其中响应是每个观测与组均值差值的绝对值 (Levene 1960)。散布程度的计算公式为：$z_{ij} = |y_{ij} - \bar{y}_i|$（相对于 SAS 默认计算方法 $z_{ij}^2 = (y_{ij} - \bar{y}_i)^2$）。

Bartlett 比较样本方差的加权算术平均值与样本方差的加权几何平均值。几何平均值始终小于或等于算术平均值，仅在所有样本方差相等时才保持相等性。组方差之间的变异越大，这两个平均值的差异越大。创建这两个平均值的函数，它近似服从 χ^2 分布 （或实际为某个公式下的 *F* 分布）。大值对应于算术或几何比的大值，因此对应于变异范围很宽的组方差。Bartlett 卡方检验统计量除以自由度得到表中所示的 *F* 值。Bartlett 检验对于违反正态分布假定的情况不是很稳健 (Bartlett and Kendall 1946)。

F 检验双侧 （仅当 X 变量具有两个水平时才可用。）若仅检验两个组，则还执行针对不等方差的标准 *F* 检验。*F* 检验是较大方差与较小方差估计值之比。将 *F* 分布的 *p* 值加倍可以使它成为双侧检验。

注意：若指定了**区组**列，则对进行了区组均值调整的数据执行方差检验。

请参见第 191 页的“‘不等方差’选项的示例”。

“等方差检验”报表

“等方差检验”报表显示各组均值与总均值和中位数的差值，并提供检验过程汇总信息。

若等方差检验指示组方差有显著差异，请使用 Welch 检验替代常规的方差分析检验。Welch 统计量基于通常的方差分析 *F* 检验。但是，均值使用组均值方差的倒数进行加权（Welch 1951；Brown and Forsythe 1974； Asiribo and Gurland 1990）。若只有两个水平，则 Welch 方差分析等价于不等方差 *t* 检验。

“等方差检验”报表的说明

水平 列出数据中出现的因子水平。

计数 记录每个水平的频数。

标准差 记录每个因子水平对应响应的标准差。标准差等于 O'Brien 变量的均值。若水平仅在数据中出现一次，则不计算标准差。

相对均值的绝对差值平均值 记录响应和组均值的绝对差值平均值。绝对差值平均值等于 Levene 变量的组均值。

相对中位数的绝对差值平均值 记录响应和组中位数的绝对差值平均值。绝对差值平均值等于 Brown-Forsythe 变量的组均值。

检验 列出所执行检验的名称。

F 比 记录每个检验计算得到的 *F* 统计量。请参见第 210 页的 “等方差检验”。

分子自由度 记录每个检验的分子的自由度。若因子具有 *k* 个水平，则分子具有 *k* - 1 个自由度。在计算 O'Brien、Brown-Forsythe 或 Levene 的检验统计量时，不使用在数据中只出现一次的水平。此情况下的分子自由度是计算中使用的水平数减 1。

分母自由度 记录在每个检验的分母中使用的自由度。对于 O'Brien、Brown-Forsythe 和 Levene，用于计算检验统计量的每个因子水平都要扣除一个自由度。若因子具有 k 个水平，则分母自由度为 *n* - *k*。

p 值 当方差在所有水平上都相等时得到 *F* 比值大于计算出的 F 比值的概率。

注意：若 X 变量的任何水平包含的观测不足 5 个，则会显示一条警告。有关以上样本较小的检验的性能的详细信息，请参见 Brown and Forsythe (1974) 和 Miller (1972)。

“Welch 检验”报表的说明

F 比 显示等方差检验的 *F* 检验统计量。

分子自由度 记录检验的分子的自由度。若因子具有 *k* 个水平，则分子具有 *k* - 1 个自由度。在计算 Welch 方差分析时，不使用在数据中只出现一次的水平。此情况下的分子自由度是计算中使用的水平数减 1。

分母自由度 记录检验的分母的自由度。请参见第 210 页的 “等方差检验”。

概率 >F 当均值在所有水平上都相等时得到 *F* 比值大于计算出的 F 比值的概率。观测到的显著性概率 0.05 或更小值被视为各水平的均值不等的证明。

t 检验 显示 *F* 比和 *t* 检验之间的关系。计算为 *F* 比的平方根。仅当 X 因子具有两个水平时才显示。

等价性检验

若您想要检测有**实际**意义的均值差值，等价性检验很有用。您必须指定一个阈值差，在该条件下较小的差值实际上被视为等价。换言之，若两个组均值的差值未超过该数值，您愿意将其视为等价。

请参见第 192 页的“‘等价性检验’的示例”。

“等价性检验”报表

一旦指定阈值和方差假设，即显示“等价性检验”报表。该报表的完整标题为“合并方差的等价性检验”或“不等方差的等价性检验”，具体取决于指定的方差假设。指定的阈值表示上下限，显示在报表顶部。报表顶部还包含等价性检验的 α 水平。该报表由一个包含等价性检验的表和一个显示这些检验的散点图构成。这些等价性检验和置信区间均基于 Student *t* 临界值。

提示：要更改等价性检验的 α 水平，请在选择“等价性检验”选项之前使用“单因子分析”红色小三角菜单中的**设置 α 水平**选项。

“合并方差的等价性检验”或“不等方差的等价性检验”旁边的红色小三角菜单包含以下选项：

等价性 TOST 检验 显示或隐藏 “等价性 TOST 检验”报表。

等价性检验散点图 显示或隐藏 “等价性检验散点图”报表。

等价性检验配对比较 显示或隐藏所有配对比较的 “实际等价性”报表。

删除 从 “单因子分析”报表窗口中删除 “等价性检验”报表。

等价性 TOST 检验

双单侧检验 (TOST) 方法用于检验均值之间的实际差值 (Schuirmann 1987)。针对实际差值超过阈值的原假设，构造使用指定方差假设的双单侧 *t* 检验。若两个检验都拒绝，则均值差值在统计上未超过任一阈值。因此，这些组被视为实际上等价。若只有一个检验拒绝或两个检验均未拒绝，则各组可能实际上不等价。

对于每个比较，“等价性 TOST 检验”报表都提供以下信息：

差值 均值的估计差值。

t 比下限、 t 比上限 双单侧合并方差显著性检验的 *t* 比的下限和上限。

p 值下限、 p 值上限 与 *t* 比的下限和上限对应的 *p* 值。

最大 p 值 *p* 值下限和上限的最大值。

下限和上限 均值差值的 1–2α 置信区间的限值。

等价性检验散点图

该散点图使用颜色指示哪些均值实际上是等价的，哪些实际上不等价，正如等价性检验确定的那样。该图有时称为**差值图**或均值-均值散点图。

该图在对角线上显示一条参考实线以及一个着色参考带。参考带的宽度是实际差值的两倍宽。线段上的点的坐标是对应组的均值。对角线上有一个隐含的第三轴，其中每条线段都对应于某一配对比较的 1–2α 置信区间。悬停在其中一个点上方可显示工具提示，该工具提示指示要比较的组和估计的差异。若线段完全包含在对角带内，由此判定均值实际上等价。

等价性检验散点图具有以下选项：

显示参考线 显示散点图上点的参考线。若散点图中有很多点，则不推荐。若有很多点，则最好悬停在各点上方以查看标签。

稳健

稳健选项提供两种方法来降低来自数据集中离群值或极端数据点的影响：“稳健拟合”和“Cauchy 拟合”。在启动窗口中指定 “区组”变量时，“稳健”选项不可用。

稳健拟合

“稳健拟合”选项可降低响应变量中离群值的影响。需要使用 Huber M 估计方法。Huber M 估计查找可将 Huber 损失函数最小化的参数估计值：

$$l(\underline{e}) = \sum_i \rho(e_i)$$

其中

$$\rho(e) = \begin{cases} \frac{1}{2}e^2 & \text{若 } |e| < k \\ k|e| - \frac{1}{2}k^2 & \text{若 } |e| \geq k \end{cases}$$

e_i 指残差

Huber 损失函数惩罚离群值，并且对于小误差呈二次增长，对于大误差呈线性增长。有关稳健拟合的更多详细信息，请参见 Huber (1973) 和 Huber and Ronchetti (2009)。请参见第 194 页的 ““稳健拟合”选项的示例”。

Cauchy 拟合

“Cauchy 拟合”选项假定误差服从 Cauchy 分布。Cauchy 分布具有比正态分布更肥大的尾部，从而弱化了对离群值的强调。若您的数据中有较大比例的离群值，该选项会很有用。不过，若您的数据接近正态，只含有少数离群值，该选项会导致不正确的推断。Cauchy 选项使用最大似然和 Cauchy 连结函数估计参数。

功效

功效选项计算有关给定假设检验的统计功效和其他详细信息。请参见第 195 页的 ““功效”选项的示例”。有关统计详细信息，请参见第 209 页的 “功效”。

- LSV（最小显著性值）是一些参数或参数函数的值，其产生某一 *p* 值 alpha。即您要了解在某个 *p* 值 alpha 下多小的效应可被称为显著。LSV 提供了以参数尺度而非概率尺度测量显著性的方法。它显示设计和数据的敏感程度。
- LSN（最小显著性数）是在数据采用相同形式的情况下将生成指定 *p* 值 alpha 的总观测数。LSN 定义为在给定的 alpha、sigma 和 delta 值的情况下（即显著性水平、误差的标准差和效应大小）减少估计值方差以实现显著性结果所需的观测数。若您需要更多数据来实现显著性，LSN 可帮助告知还需要多少数据。LSN 是生成约 50% 功效的总观测数。
- **功效**是当两个组存在真正差异时获得显著性（*p* 值 < alpha）的概率。它是样本大小、效应大小、误差的标准差和显著性水平的函数。功效告知您的实验在给定的 alpha 水平下有多大可能检测到差异（效应大小）。

注意：单因子布局中仅有两个组时，功效工具所计算的 LSV 与多重比较表中所示的最小显著性差异 (LSD) 相同。

“功效详细信息”窗口和报表

“功效详细信息”窗口和报表与“拟合模型”平台所启动的常规拟合平台中的窗口和报表相同。有关功效计算的详细信息，请参见《拟合线性模型》。

对于四个列 Alpha、Sigma、Delta 和个数，请分别填写一个值、两个值或一系列值的初值、终值和增量 （图 6.32）。对您指定值的所有可能组合执行功效计算。

Alpha (α) 显著性水平，介于 0 到 1 之间 （通常为 0.05、 0.01 或 0.10）。最初显示值 0.05。

Sigma (σ) 模型中剩余误差的标准误差。最初在此处提供 RMSE，即误差的均方平方根的估计值。

Delta (δ) 原始效应大小。有关效应大小计算的详细信息，请参见《拟合线性模型》。第一个位置最初设置为假设的平方和除以 *n* 所得值的平方根 （即 $\delta = \sqrt{SS/n}$ ）。

个数 (*n*) 所有组的总样本大小。最初在第一个位置存放实际样本大小。

功效求解 将功效 （显著性结果的概率）作为以下四个值的函数求解：α、 σ、 δ 和 *n*。

最小显著数求解 在给定 α、 σ 和 δ 的情况下求解实现约 50% 的功效所需的观测数。

最小显著值求解 求解生成 α 的 *p* 值的参数或线性检验的值。它是 α、σ、*n* 和估计值的标准误差的函数。仅当 X 因子具有两个水平且通常用于单个参数时，才可使用该功能。

调整功效和置信区间 要追溯了解功效时，可以使用标准误差和检验参数的估计值。

- 调整功效是从非中心参数的更加无偏估计值计算的功效。
- 调整功效的置信区间基于非中心估计值的置信区间。

仅对原始 Delta 计算调整功效和置信限，因为其中存在随机变异。

正态分位数图

您可以创建两种类型的正态分位数图：

- **标绘“分位数-实际值”图**创建响应值与正态分位数值的图。针对 X 变量的每个水平分别计算和标绘分位数。
- **标绘“实际值-分位数”图**创建正态分位数值与响应值的图。针对 X 变量的每个水平分别计算和标绘分位数。

拟合线选项显示或隐藏在分位数图中的拟合线。**正态分位数标签**选项在“正态分位数”图的右垂直轴内和“分位数-实际值”图的上轴内显示或隐藏标签。标签代表正态分位数尺度。

在启动窗口中指定“区组”变量时，“正态分位数图”选项不可用。

请参见第 197 页的“正态分位数图的示例”。

CDF 图

CDF 图显示“单因子”报表中所有组的累积分布函数。若要比较 X 因子各水平上响应的分布，CDF 图很有用。在启动窗口中指定“区组”变量时，“CDF 图”选项不可用。请参见第 198 页的“CDF 图的示例”。

密度

密度选项提供几种方式来直观比较 X 因子各水平上响应的分布和成分。有三种密度选项：

- **比较密度**显示估计每个组的密度的平滑曲线。平滑曲线是每个组的密度估计值。
- **密度成分**显示按每个组的计数加权的密度之和。在每个 X 值上，密度成分图显示每个组对总和的贡献情况。
- **密度比例**显示在每个 X 水平上组的贡献占总和的比例。

在启动窗口中指定“区组”变量时，“密度”选项不可用。

请参见第 199 页的“‘密度’选项的示例”。

匹配列

使用**匹配列**选项可以指定匹配模型分析的匹配 (ID) 变量。**匹配列**选项适用于单因子分析中的数据来自匹配（配对）数据的情况。不同组的观测来自同一参与者时即会出现匹配数据。在启动窗口中指定“区组”变量时，“匹配列”选项不可用。请参见第 200 页的“‘匹配列’选项的示例”。

注意：匹配的特殊情况导致配对 *t* 检验。**配对**平台处理这类数据，但是数据必须使用不同列（而非不同行）中的配对进行整理。

匹配列选项执行两个主要操作：

- 它拟合加法模型（使用迭代比例拟合算法），该模型包括分组变量（“以 X 拟合 Y”分析中的 X 变量）和您选择的匹配变量。若有数百个参与者，则迭代比例拟合算法会带来差异，因为等价的线性模型将运行很慢并且需要大量内存资源。
- 它将匹配各组的点绘制成线。若相同匹配 ID 值具有多个观测值，则根据观测值组的均值绘制线。

匹配列选项自动激活连接匹配点的**匹配线**选项。若要禁用这些线，请选择**显示选项** > **匹配线**。

“匹配拟合”报表显示使用 *F* 检验的效应。这些检验等价于您运行两个模型时（一个具有交互作用项，另一个不具有交互作用项）使用“拟合模型”平台得到的检验。若仅有两个水平，则 *F* 检验等价于配对 *t* 检验。

注意：有关“拟合模型”平台的详细信息，请参见《拟合线性模型》。

“单因子”平台的更多示例

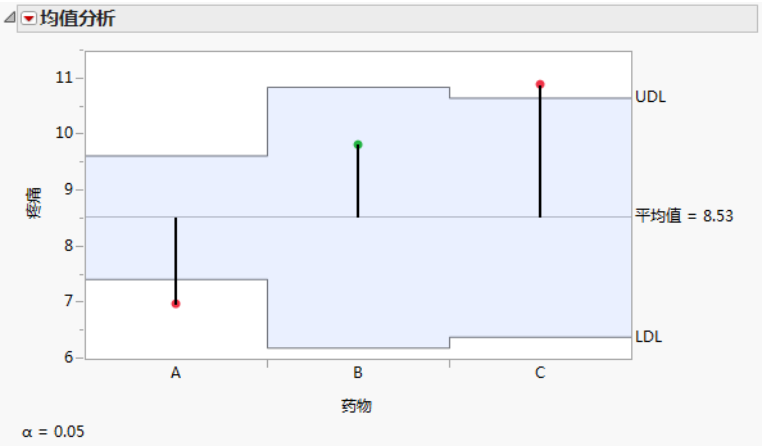
- “均值分析图表示例”
- “方差均值分析图表的示例”
- “‘每对，Student t’检验的示例”
- “‘所有对，Tukey HSD’检验的示例”
- “‘带最佳组的 Hsu MCB 检验’的示例”
- “‘带控制组的 Dunnett 检验’的示例”
- “‘每对逐步，Newman-Keuls’检验的示例”
- “示例：比较四个‘比较均值’检验”
- “非参数 Wilcoxon 检验的示例”
- “‘不等方差’选项的示例”

- “‘等价性检验’的示例”
- “‘稳健拟合’选项的示例”
- “‘功效’选项的示例”
- “正态分位数图的示例”
- “CDF 图的示例”
- “‘密度’选项的示例”
- “‘匹配列’选项的示例”
- “示例：堆叠数据以进行单因子分析”

均值分析图表示例

1. 选择**帮助** > **样本数据库**，然后打开 Analgesics.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**疼痛**并点击 **Y，响应**。
4. 选择**药物**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “均值分析”红色小三角并选择**均值分析方法** > ANOM。

图 6.13 均值分析图表的示例



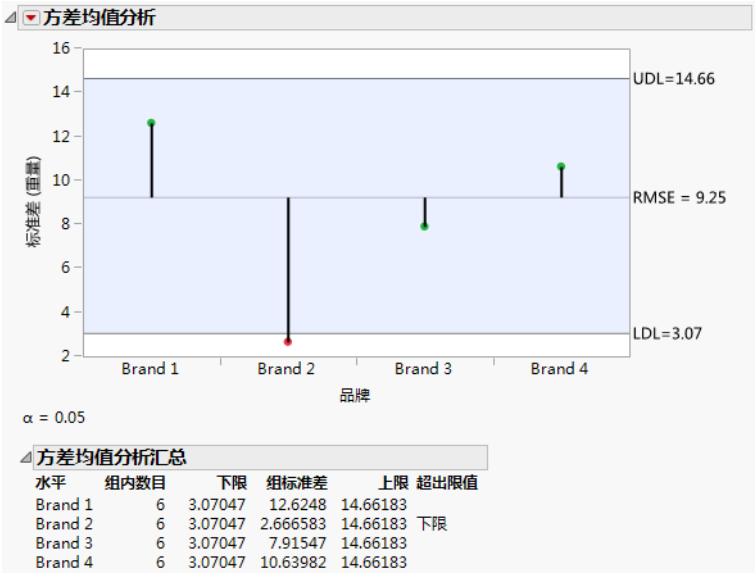
例如，药物 A 和 C 的均值在统计上与总体均值存在差异。药物 A 均值偏低，药物 C 均值偏高。请注意由于样本大小不同，药物类型的决策限将不同。

方差均值分析图表的示例

使用 ANOMV 检验不同品牌弹簧之间的变异性是否存在显著差异。检验四个不同品牌的弹簧，以了解延伸弹簧 0.10 英寸所需的重量。检验每个品牌的 6 个弹簧。检查数据是否服从正态分布，因为 ANOMV 检验对非正态分布不稳健。检查品牌以确定各个品牌间是否存在显著差异。

1. 选择**帮助** > **样本数据库**，然后打开 Spring Data.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**重量**并点击 **Y，响应**。
4. 选择**品牌**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘品牌 - 重量’单因子分析”旁边的红色小三角并选择**均值分析方法** > **方差 ANOM**。
7. 点击“方差均值分析”旁边的红色小三角并选择**显示汇总报表**。

图 6.14 方差均值分析图表的示例



方差均值分析汇总

水平	组内数目	下限	组标准差	上限	超出限值
Brand 1	6	3.07047	12.6248	14.66183	
Brand 2	6	3.07047	2.666583	14.66183	下限
Brand 3	6	3.07047	7.91547	14.66183	
Brand 4	6	3.07047	10.63982	14.66183	

请注意品牌 2 的标准差超过下决策限。因此，品牌 2 的方差明显低于其他品牌的方差。

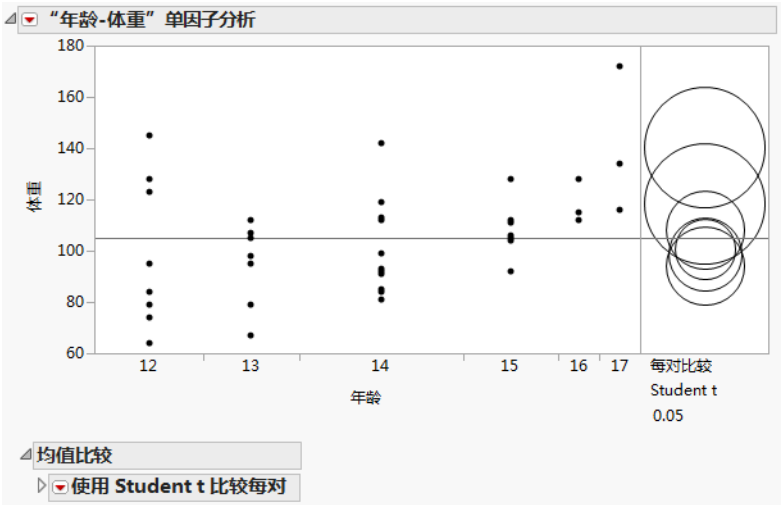
“每对，Student t”检验的示例

本例演示所有可能的 *t* 检验的用法。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**并点击 **Y，响应**。

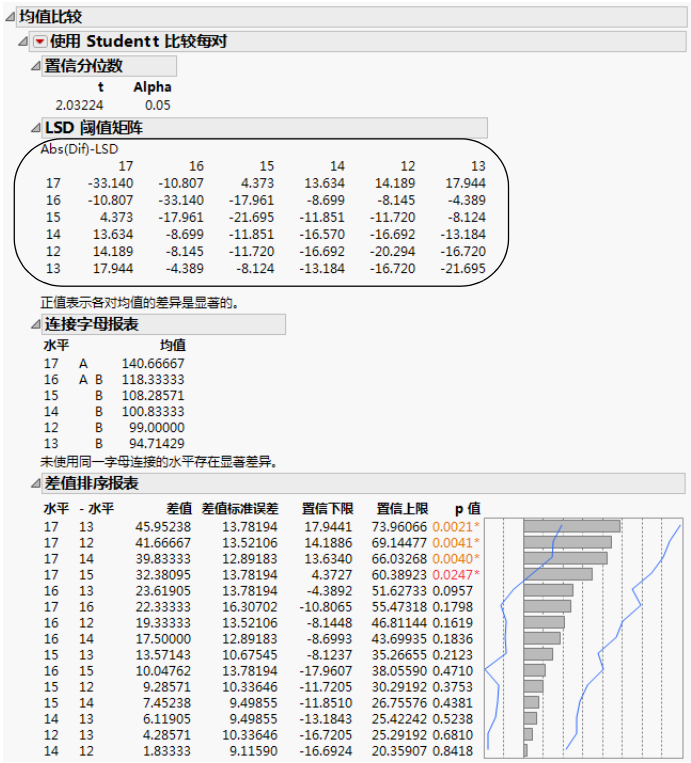
4. 选择**年龄**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘年龄 - 体重’单因子分析”旁边的红色小三角并选择**比较均值 > 每对，** Student t。

图 6.15 “每对， Student t”比较环的示例



该均值比较方法可以看作是检查均值的实际差异是否大于可以称得上显著性的差异。该差异称为 LSD（最小显著性差异）。LSD 项用于 Student *t* 区间和与其他检验的区间有关的上下文中。在比较环图中，环中心之间的距离表示实际差异。LSD 是环以直角交叉时环中心间的距离。

图 6.16 “每对， Student *t*”的均值比较报表示例



均值比较

使用 Studentt 比较每对

置信分位数

t	Alpha
2.03224	0.05

LSD 阈值矩阵

Abs(Dif)-LSD

	17	16	15	14	12	13
17	-33.140	-10.807	4.373	13.634	14.189	17.944
16	-10.807	-33.140	-17.961	-8.699	-8.145	-4.389
15	4.373	-17.961	-21.695	-11.851	-11.720	-8.124
14	13.634	-8.699	-11.851	-16.570	-16.692	-13.184
12	14.189	-8.145	-11.720	-16.692	-20.294	-16.720
13	17.944	-4.389	-8.124	-13.184	-16.720	-21.695

正值表示各对均值的差异是显著的。

连接字母报表

水平		均值
17	A	140.66667
16	A B	118.33333
15	B	108.28571
14	B	100.83333
12	B	99.00000
13	B	94.71429

未使用同一字母连接的水平存在显著差异。

差值排序报表

水平	- 水平	差值	差值标准误差	置信下限	置信上限	p 值
17	13	45.95238	13.78194	17.9441	73.96066	0.0021*
17	12	41.66667	13.52106	14.1886	69.14477	0.0041*
17	14	39.83333	12.89183	13.6340	66.03268	0.0040*
17	15	32.38095	13.78194	4.3727	60.38923	0.0247*
16	13	23.61905	13.78194	-4.3892	51.62733	0.0957
17	16	22.33333	16.30702	-10.8065	55.47318	0.1798
16	12	19.33333	13.52106	-8.1448	46.81144	0.1619
16	14	17.50000	12.89183	-8.6993	43.69935	0.1836
15	13	13.57143	10.67545	-8.1237	35.26655	0.2123
16	15	10.04762	13.78194	-17.9607	38.05590	0.4710
15	12	9.28571	10.33646	-11.7205	30.29192	0.3753
15	14	7.45238	9.49855	-11.8510	26.75576	0.4381
14	13	6.11905	9.49855	-13.1843	25.42242	0.5238
12	13	4.28571	10.33646	-16.7205	25.29192	0.6810
14	12	1.83333	9.11590	-16.6924	20.35907	0.8418

在图 6.16 中，LSD 阈值表显示均值的绝对差值与 LSD（最小显著性差异）之间的差值。若值为正数，则两个均值之间的差值大于 LSD，说明这两个组存在显著差异。

“所有对，Tukey HSD”检验的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**并点击 **Y，响应**。
4. 选择**年龄**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘年龄 - 体重’单因子分析”旁边的红色小三角并选择**比较均值** > **所有对，Tukey HSD**。

图 6.17 “所有对， Tukey HSD”比较环的示例

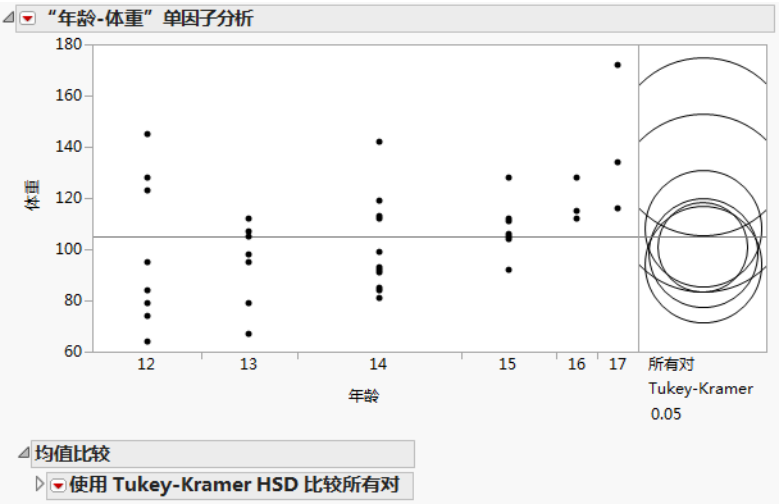


图 6.18 “所有对， Tukey HSD”的均值比较报表示例

均值比较

使用 Tukey-Kramer HSD 比较所有对

置信分位数

q*	Alpha
3.01825	0.05

HSD 阈值矩阵

Abs(Dif)-HSD

	17	16	15	14	12	13
17	-49.219	-26.885	-9.216	0.923	0.857	4.355
16	-26.885	-49.219	-31.550	-21.411	-21.477	-17.978
15	-9.216	-31.550	-32.221	-21.217	-21.912	-18.650
14	0.923	-21.411	-21.217	-24.609	-25.681	-22.550
12	0.857	-21.477	-21.912	-25.681	-30.140	-26.912
13	4.355	-17.978	-18.650	-22.550	-26.912	-32.221

正值表示各对均值的差异是显著的。

连接字母报表

水平		均值
17	A	140.66667
16	A B	118.33333
15	A B	108.28571
14	B	100.83333
12	B	99.00000
13	B	94.71429

未使用同一字母连接的水平存在显著差异。

差值排序报表

水平	- 水平	差值	差值标准误差	置信下限	置信上限	p 值
17	13	45.95238	13.78194	4.3551	87.54968	0.0233*
17	12	41.66667	13.52106	0.8568	82.47657	0.0431*
17	14	39.83333	12.89183	0.9226	78.74404	0.0423*
17	15	32.38095	13.78194	-9.2163	73.97825	0.2029
16	13	23.61905	13.78194	-17.9783	65.21635	0.5325
17	16	22.33333	16.30702	-26.8853	71.55192	0.7443
16	12	19.33333	13.52106	-21.4766	60.14323	0.7091
16	14	17.50000	12.89183	-21.4107	56.41071	0.7512
15	13	13.57143	10.67545	-18.6497	45.79256	0.7982
16	15	10.04762	13.78194	-31.5497	51.64492	0.9769
15	12	9.28571	10.33646	-21.9123	40.48369	0.9442
15	14	7.45238	9.49855	-21.2166	36.12134	0.9683
14	13	6.11905	9.49855	-22.5499	34.78801	0.9867
12	13	4.28571	10.33646	-26.9123	35.48369	0.9983
14	12	1.83333	9.11590	-25.6807	29.34736	1.0000

在图 6.18 中，Tukey-Kramer HSD 阈值矩阵显示均值的实际绝对差值减去 HSD 的值。该值表示该差异会是显著的。具有正值的配对存在显著差异。q*（显示在 HSD 阈值矩阵表上方）是用于缩放 HSD 的分位数。它的计算作用与 Student *t* 相当。

“带最佳组的 Hsu MCB 检验”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**并点击 **Y，响应**。
4. 选择**年龄**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘年龄 - 体重’单因子分析”旁边的红色小三角并选择**比较均值** > **带最佳组的 Hsu MCB 检验**。

图 6.19 “带最佳组的 Hsu MCB 检验”比较环的示例

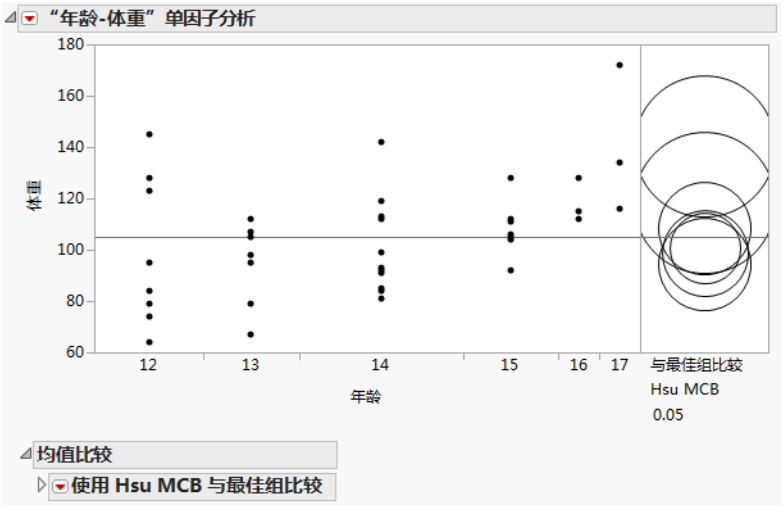
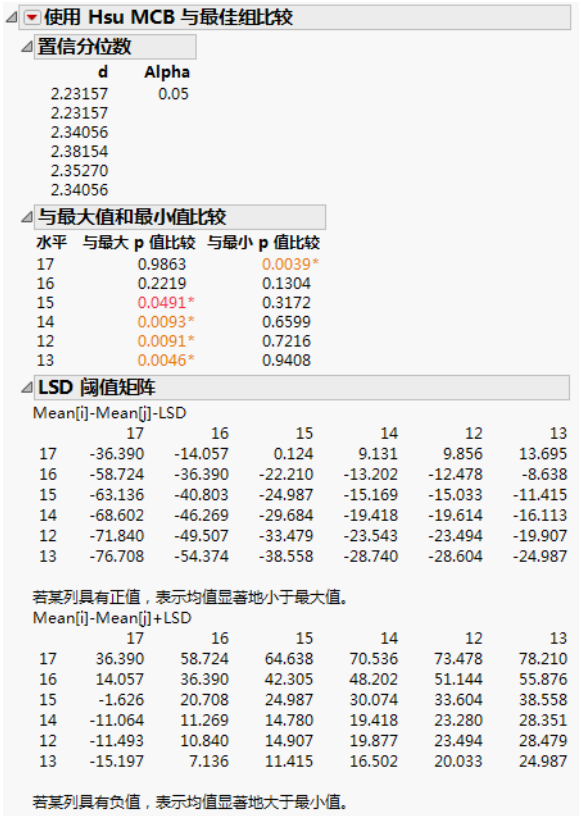


图 6.20 “带最佳组的 Hsu MCB 检验”的均值比较报表示例



使用 Hsu MCB 与最佳组比较

置信分位数

d	Alpha
2.23157	0.05
2.23157	
2.34056	
2.38154	
2.35270	
2.34056	

与最大值和最小值比较

水平	与最大 p 值比较	与最小 p 值比较
17	0.9863	0.0039*
16	0.2219	0.1304
15	0.0491*	0.3172
14	0.0093*	0.6599
12	0.0091*	0.7216
13	0.0046*	0.9408

LSD 阈值矩阵

Mean[i]-Mean[j]-LSD

	17	16	15	14	12	13
17	-36.390	-14.057	0.124	9.131	9.856	13.695
16	-58.724	-36.390	-22.210	-13.202	-12.478	-8.638
15	-63.136	-40.803	-24.987	-15.169	-15.033	-11.415
14	-68.602	-46.269	-29.684	-19.418	-19.614	-16.113
12	-71.840	-49.507	-33.479	-23.543	-23.494	-19.907
13	-76.708	-54.374	-38.558	-28.740	-28.604	-24.987

若某列具有正值，表示均值显著地小于最大值。

Mean[i]-Mean[j]+LSD

	17	16	15	14	12	13
17	36.390	58.724	64.638	70.536	73.478	78.210
16	14.057	36.390	42.305	48.202	51.144	55.876
15	-1.626	20.708	24.987	30.074	33.604	38.558
14	-11.064	11.269	14.780	19.418	23.280	28.351
12	-11.493	10.840	14.907	19.877	23.494	28.479
13	-15.197	7.136	11.415	16.502	20.033	24.987

若某列具有负值，表示均值显著地大于最小值。

“与最大值和最小值的比较”报表将每个水平的均值与其余水平均值的最大值和最小值进行比较。例如，年龄 15 的均值与其余水平均值的最大值显著不同。年龄 17 的均值与其余水平均值的最小值显著不同。最大均值可能出现在年龄 16 或年龄 17 上，因为这两个均值均未与最大均值有显著不同。出于相同原因，最小均值可能对应于除年龄 17 之外的任何年龄。

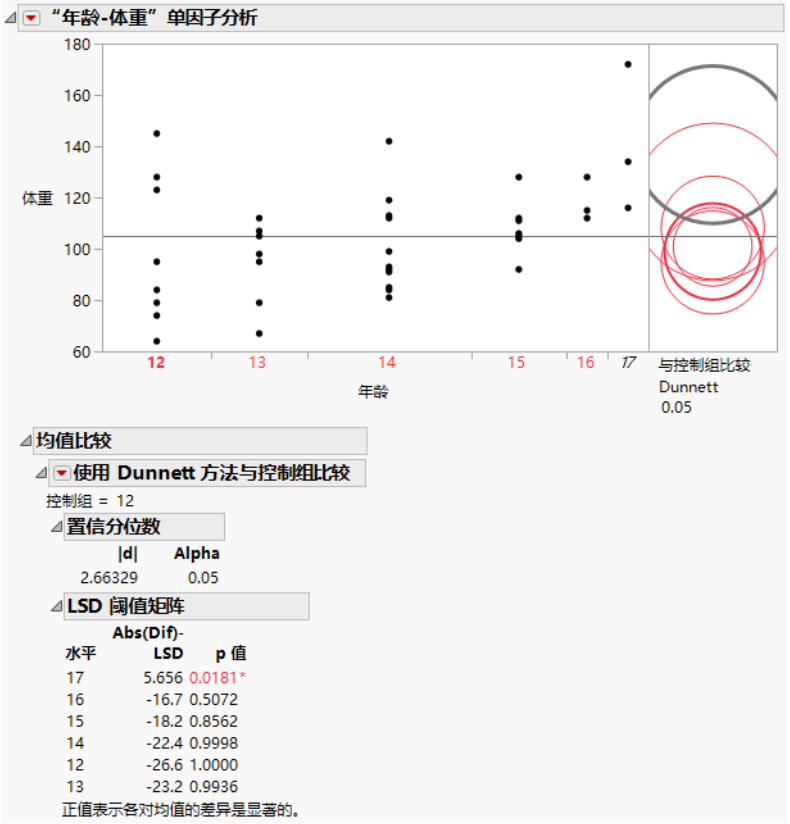
“带控制组的 Dunnett 检验”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**并点击 **Y，响应**。
4. 选择**年龄**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘年龄 - 体重’单因子分析”旁边的红色小三角并选择**比较均值** > **带控制组的 Dunnett 检验**。
7. 选择要用作控制组的组。在该示例中，选择年龄 12 岁。

或者，在选择**比较均值** > **带控制组的 Dunnett 检验**选项前点击某行以在散点图中突出显示它。检验将使用所选行作为控制组。

8. 点击**确定**。

图 6.21 “带控制组的 Dunnett 检验”比较环的示例

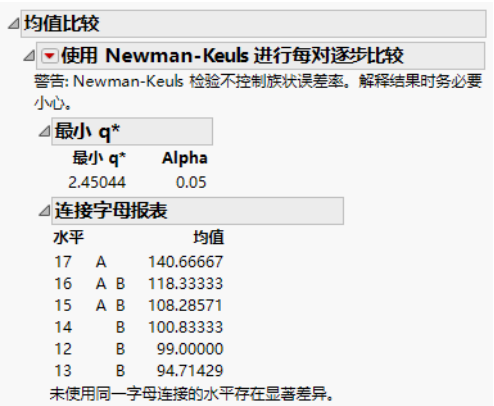


使用比较环，您可以得出结论：水平 17 是唯一与控制水平 12 存在显著差异的水平。

“每对逐步，Newman-Keuls”检验的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**，然后点击 **Y，响应**。
4. 选择**年龄**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘年龄 - 体重’单因子分析”旁边的红色小三角并选择**比较均值** > **拟合每对逐步，Newman-Keuls**。

图 6.22 “每对逐步， Newman-Keuls”的均值比较报表示例



均值比较

使用 Newman-Keuls 进行每对逐步比较

警告: Newman-Keuls 检验不控制族状误差率。解释结果时务必要小心。

最小 q*

最小 q*	Alpha
2.45044	0.05

连接字母报表

水平		均值
17	A	140.66667
16	A B	118.33333
15	A B	108.28571
14	B	100.83333
12	B	99.00000
13	B	94.71429

未使用同一字母连接的水平存在显著差异。

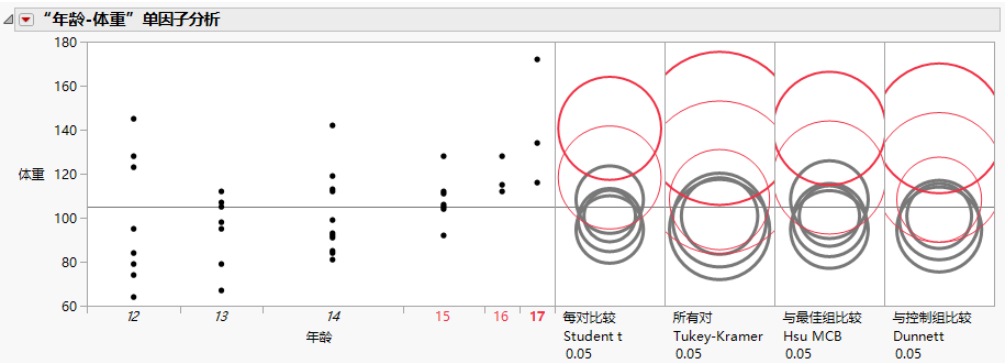
“连接字母报表”显示：水平 17 与除 16 和 15 之外的其他所有水平都显著不同。

示例：比较四个“比较均值”检验

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**体重**并点击 **Y，响应**。
4. 选择**年龄**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘年龄 - 体重’单因子分析”旁边的红色小三角并选择每个**比较均值**选项。对于“带控制组的 Dunnett 检验”选项，将年龄 17 选作控制组。

四个方法都检验各组均值之间的差异。每项检验都用于一个特定的假设，可能会出现不同的结果。

图 6.23 四个多重比较检验的比较环



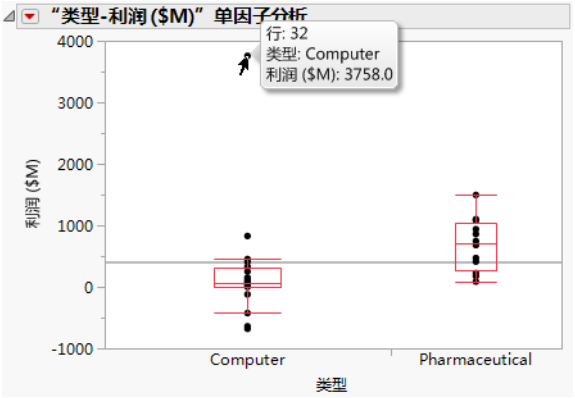
在图 6.23 中，年龄组 17 被突出显示。其他控制环的颜色与年龄组 17 有关。请注意，对于 Student t 和 Hsu 方法，年龄组 15（从顶部数第三个环）显示为灰色。这表明它与年龄组 17 有显著差异。不过，对于 Tukey 和 Dunnett 方法，年龄组 15 为红色，这表明它与年龄组 17 没有显著差异。

非参数 Wilcoxon 检验的示例

使用 Wilcoxon 检验来确定各公司赢利均值是否因公司类型而异。在 Companies.jmp 中，数据包含医药 （12 个公司）和计算机 （20 个公司）这两类公司的各种量度。

1. 选择**帮助** > **样本数据库**，然后打开 Companies.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**利润 ($M)** 并点击 **Y，响应**。
4. 选择**类型**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘类型 - 利润 ($M)’单因子分析”旁边的红色小三角并选择**显示选项** > **箱线图**。

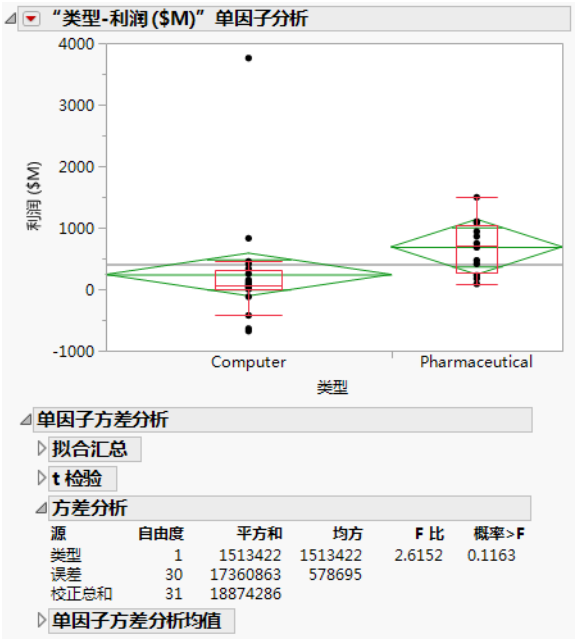
图 6.24 计算机公司的利润分布



箱线图表明分布非正态分布，甚至不对称。第 32 行中的公司有一个非常大的值，该值可能影响参数检验。

7. 点击“‘类型 - 利润 ($M)’单因子分析”旁边的红色小三角并选择**均值 / 方差分析 / 合并的 t**。

图 6.25 公司方差分析



F 检验显示不显著，因为 *p* 值较大 ($p = 0.1163$)。这可能是因为第 32 行中的值较大，导致正态性假设可能不成立。

8. 点击“‘类型 - 利润 ($M)’单因子分析”旁边的红色小三角并选择 **t 检验**。

图 6.26 t 检验结果

双侧检验的概率 > |t| 为 0.0671。*t* 检验不假设方差相等，但不等方差 *t* 检验也是一个参数检验。

9. 点击“‘类型 - 利润 ($M)’单因子分析”旁边的红色小三角并选择**非参数 > Wilcoxon 检验**。

图 6.27 Wilcoxon 检验结果

Wilcoxon / Kruskal-Wallis 检验 (秩和)

水平	计数	得分和	期望得分	得分均值	(Mean-Mean0)/Std0
Computer	20	245.000	330.000	12.2500	-3.289
Pharmaceutical	12	283.000	198.000	23.5833	3.289

双样本检验，近似正态分布

S	Z	概率>|Z|
283	3.28916	0.0010*

单因子检验，近似卡方分布

卡方	自由度	概率>卡方
10.9470	1	0.0009*

Wilcoxon 检验是一个非参数检验。它基于秩，所以不受离群值影响。同时，它不要求正态性。

Wilcoxon 检验统计量的正态和卡方近似均表明显著性的 *p* 值为 0.0010。您可以得出结论：分布位置存在显著性差异，并且利润均值随公司类型而有所不同。

正态和卡方检验均基于检验统计量的渐近分布。若使用 JMP Pro，您可以执行精确检验。

10. JMP PRO 点击“‘类型 - 利润 ($M)’单因子分析”旁边的红色小三角并选择**非参数 > 精确检验 > Wilcoxon 精确检验**。

图 6.28 Wilcoxon 精确检验结果

Wilcoxon / Kruskal-Wallis 检验 (秩和)

水平	计数	得分和	期望得分	得分均值	(Mean-Mean0)/Std0
Computer	20	245.000	330.000	12.2500	-3.289
Pharmaceutical	12	283.000	198.000	23.5833	3.289

双样本检验，近似正态分布

S	Z	概率>|Z|
283	3.28916	0.0010*

单因子检验，近似卡方分布

卡方	自由度	概率>卡方
10.9470	1	0.0009*

双样本:精确检验

S	概率≥S	概率≥|S-Mean|
283	0.0003*	0.0005*

检验统计量的观测值为 S = 283。这是样本大小较小的**类型**（医药）的水平的秩之和。观测到与中秩均值的绝对差值超过 S 减去中秩均值后的绝对值的概率为 0.0005。这是一个针对位置差异的双侧检验，并且支持拒绝关于利润不随公司类型而有所不同的假设。

在本例中，非参数检验比基于正态性的方差分析检验和不等方差 *t* 检验更合适。非参数检验不受第 32 行中的大值影响，并且不要求假设服从正态分布。

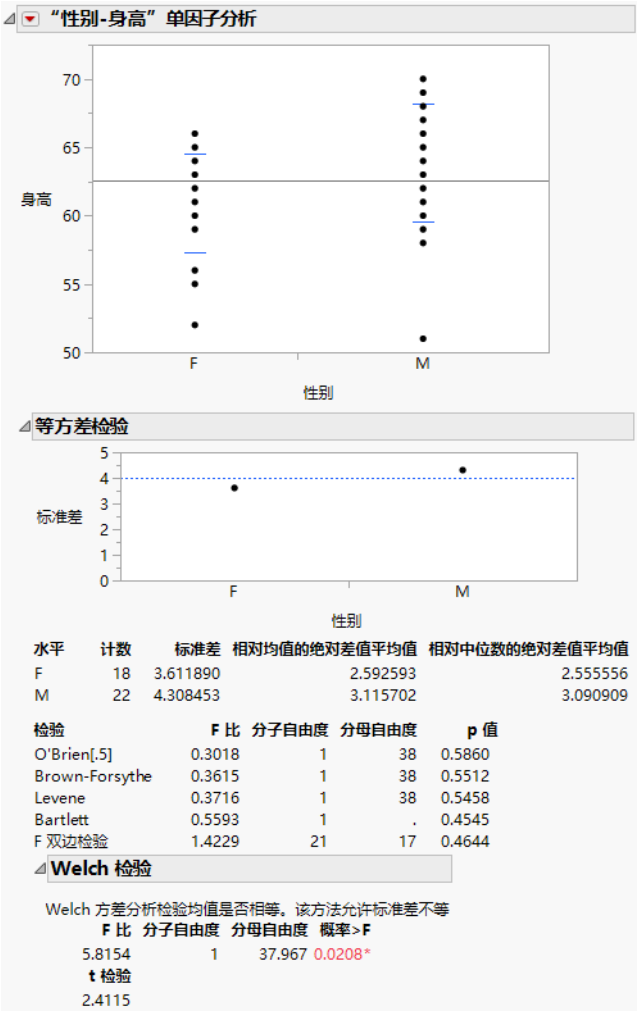
“不等方差”选项的示例

在本例中，假设您要检验两组（男性和女性）的方差是否相等。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**身高**并点击 **Y，响应**。

4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘性别 - 身高’单因子分析”旁边的红色小三角并选择**不等方差**。

图 6.29 “不等方差”报表的示例



由于双侧 *F* 检验的 *p* 值很大，您无法得出两个方差不等的结论。

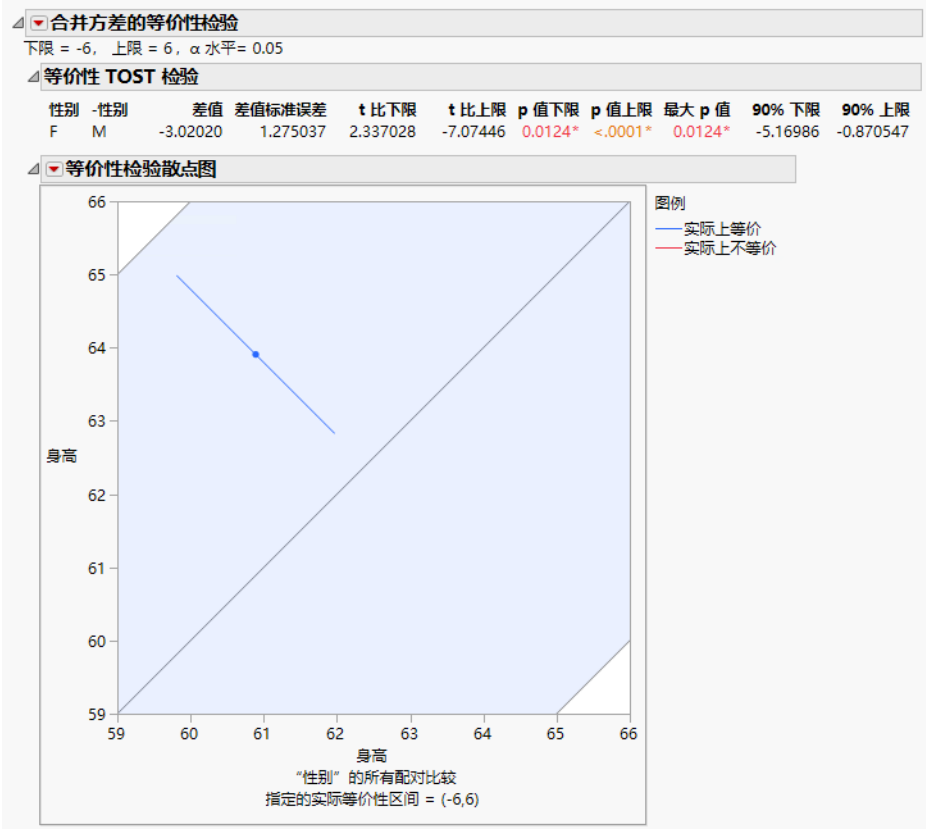
“等价性检验”的示例

在本例中，您调查男性和女性的身高差异是否小于 6 英寸。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。

2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**身高**并点击 **Y，响应**。
4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘性别 - 身高’单因子分析”红色小三角菜单并选择**等价性检验**。
7. 键入 6 作为实际可视为零的差异。
8. 点击**确定**。

图 6.30 “等价性检验”的示例



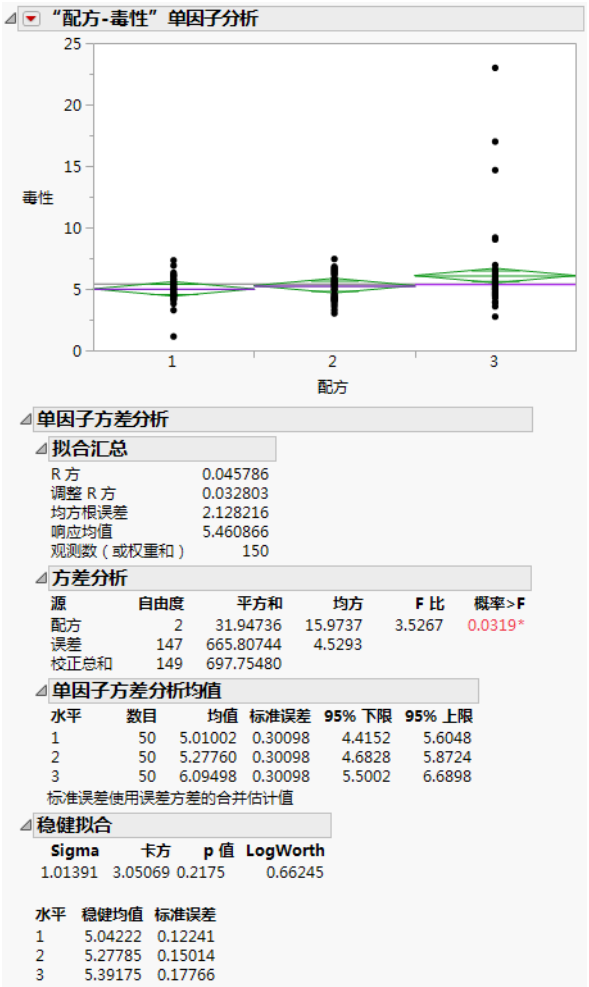
使用双单侧检验，您可以看到两者的 p 值都很小。因此，您可以得出结论：总体均值中的差值位于 6 到 -6 之间的某个位置。从而您可以声明均值实际上是等价的。

“稳健拟合”选项的示例

本例中使用“稳健拟合”，其中三个组中的一个包含离群值。 Drug Toxicity.jmp 样本数据表中的数据显示药物的三种不同配方的毒性水平。

1. 选择**帮助** > **样本数据库**，然后打开 Drug Toxicity.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**毒性**并点击 **Y，响应**。
4. 选择**配方**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘配方 - 毒性’单因子分析”旁边的红色小三角并选择**均值 / 方差分析**。
7. 点击“‘配方 - 毒性’单因子分析”旁边的红色小三角并选择**稳健** > **稳健拟合**。

图 6.31 “稳健拟合”的示例



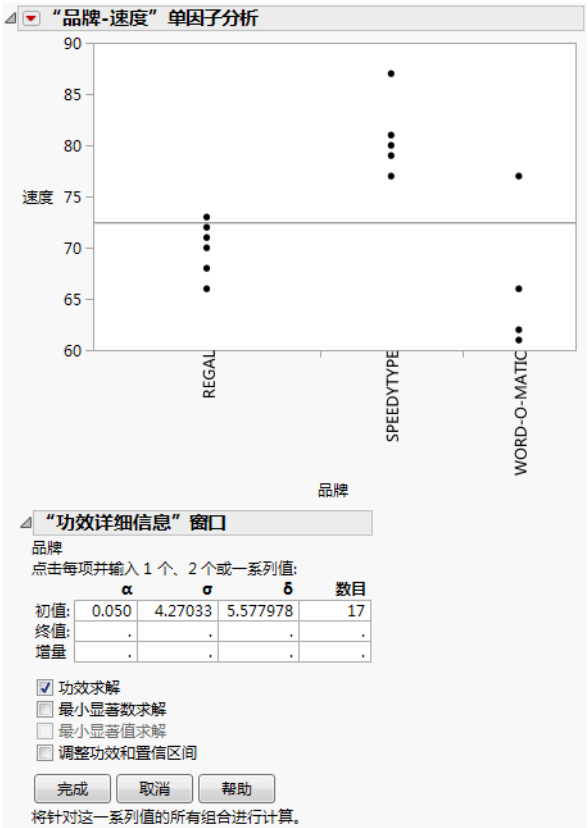
若您查看标准“方差分析”报表，可能得出这三种配方有差异的错误结论，因为 p 值为 0.0319。但是，当您查看“稳健拟合”报表时，不能得出这三种配方具有显著差异的结论，因为 p 值为 0.21755。几个观测的毒性偏高，过度影响了数据。

“功效”选项的示例

1. 选择**帮助** > **样本数据库**，然后打开 Typing Data.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**速度**并点击 **Y，响应**。
4. 选择**品牌**并点击 **X，因子**。
5. 点击**确定**。

6. 点击“‘品牌 - 速度’单因子分析”旁边的红色小三角并选择**功效**。
7. 在“初值”行内，键入 2 作为 Delta 值（第三个框），键入 11 作为“数目”值。
8. 在“终值”行内，键入 6 作为 Delta 值，在“数目”框中键入 17。
9. 在“增量”行内，键入 2 同时作为 Delta 和“数目”的值。
10. 选中**功效求解**复选框。

图 6.32 “功效详细信息”窗口的示例



11. 点击**完成**。

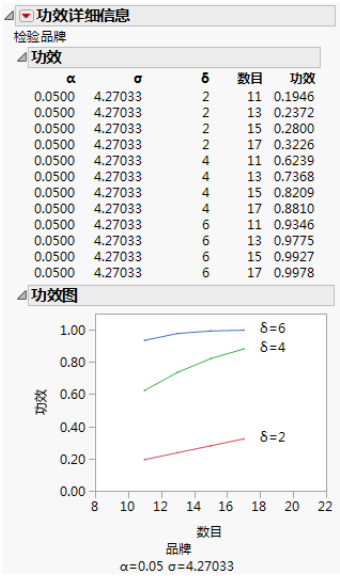
注意：完成按钮始终保持灰显，直到应用了所有必需选项。

为每个 Delta 和“数目”的组合计算功效并显示在“功效”报表中。

对“功效”值绘图：

12. 点击“功效详细信息”红色小三角并选择**功效图**。

图 6.33 “功效”报表的示例



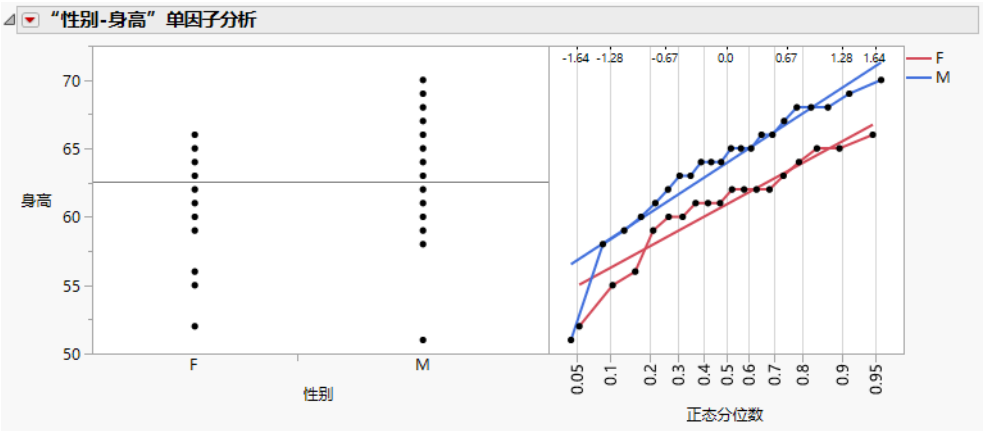
13. 您可能需要点击并在 “功效”轴上垂直拖动才可以查看图中的所有数据。

为每个 Delta 和“数目”的组合绘制功效。如您预期的那样，对于更大的“数目”值（样本大小）和 Delta 值（均值的差异），功效增大。

正态分位数图的示例

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**身高**并点击 **Y，响应**。
4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘性别 - 身高’单因子分析”旁边的红色小三角并选择**正态分位数图 > “分位数 - 实际值”图**。

图 6.34 正态分位数图的示例



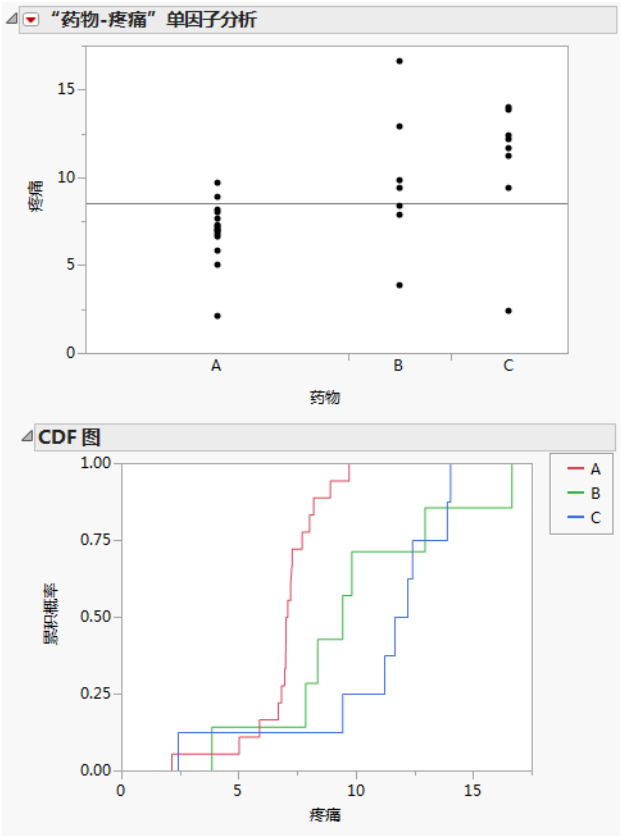
请注意以下几点：

- 默认显示“拟合线”。
- 数据点离拟合线很近，说明服从正态分布。

CDF 图的示例

1. 选择**帮助** > **样本数据库**，然后打开 Analgesics.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**疼痛**并点击 **Y，响应**。
4. 选择**药物**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘药物 - 疼痛’单因子分析”旁边的红色小三角并选择 **CDF 图**。

图 6.35 CDF 图的示例

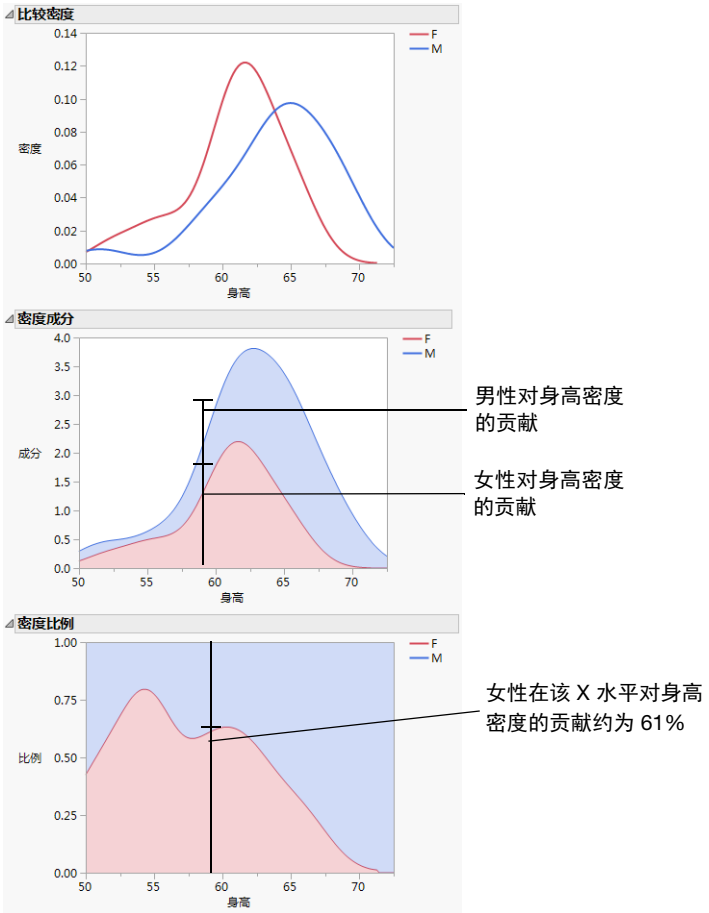


初始单因子分析中的 X 变量的不同水平在 CDF 图中作为不同曲线显示。CDF 图的水平轴使用初始单因子分析中的 *y* 值。

“密度”选项的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**身高**并点击 **Y，响应**。
4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘性别 - 身高’单因子分析”旁边的红色小三角并选择**密度** > **比较密度**、**密度** > **密度成分**和**密度** > **密度比例**。

图 6.36 “密度”选项的示例

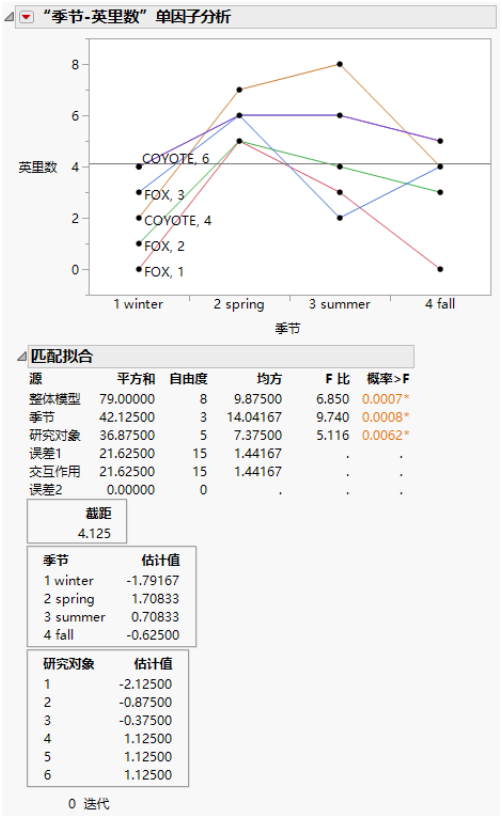


“匹配列”选项的示例

该示例使用 Matching.jmp 样本数据表，其中包含六种动物的数据和它们在不同季节行走的英里数。

1. 选择**帮助 > 样本数据库**，然后打开 Matching.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**英里数**并点击 **Y，响应**。
4. 选择**季节**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘季节 - 英里数’单因子分析”旁边的红色小三角并选择**匹配列**。
7. 选择**研究对象**作为匹配列。
8. 点击**确定**。

图 6.37 “匹配列”报表的示例



该图显示按季节行走的英里数，以研究对象作为匹配变量。图形中每个研究对象的第一个测量值旁边的标签由**物种**和**研究对象**变量决定。

“匹配拟合”报表显示使用 *F* 检验的**季节**和**研究对象**效应。这些检验等价于您运行两个模型时（一个具有交互作用项，另一个不具有交互作用项）使用“拟合模型”平台得到的检验。若仅有两个水平，则 *F* 检验等价于配对 *t* 检验。

注意：有关“拟合模型”平台的详细信息，请参见《拟合线性模型》。

示例：堆叠数据以进行单因子分析

若您的数据采用 JMP 数据表之外的格式，有时这些数据会排列成一行中包含多个观测的信息。要在 JMP 中分析这些数据，您必须导入数据并调整其结构，令 JMP 数据表中的每行都包含一个观测的信息。例如，假定数据包含在电子表格中。在三条生产线上生产的部件的数据排列在三组列中。在 JMP 数据表中，您需要将这三条生产线上的数据堆叠到一组列中，以便每行表示单个部件的数据。

说明和目标

本例使用文件 Fill Weights.xlsx，其中包含从三条不同生产线随机抽样的麦片盒的重量。图 6.38 显示了数据格式。

- “ID”列包含测量的每个麦片盒的标识符。
- “Line”列包含从相应生产线抽样的麦片盒的重量（以盎司为单位）。

图 6.38 数据格式

Weights					
ID	Line A	ID	Line B	ID	Line C
215	12.42	705	13.63	254	11.73
287	12.49	670	12.56	282	11.40
381	12.80	715	12.87	938	12.78
		683	13.09	597	12.19
		514	13.31	179	12.25
		517	12.64		
		946	12.75		

这些麦片盒的目标填充重量为 12.5 盎司。尽管您关注的是这三条生产线是否满足目标要求，但最初您想要看看这三条生产线是否实现了相同的填充率均值。您可以使用“单因子”来检验填充重量均值之间的差值。

要使用 “单因子”平台，您需要执行以下操作：

1. 将数据导入 JMP。请参见第 202 页的“导入数据”。
2. 重塑数据，以便 JMP 数据表中的每行都仅反映一个观测。重塑数据要求您将麦片盒 ID、生产线标识符和重量都堆叠到列中。请参见第 204 页的 “堆叠数据”。

导入数据

本例演示将数据从 Microsoft Excel 导入 JMP 的两种方式。选择其中一种方法或探究两种方法：

- 使用**文件 > 打开**选项通过“Excel 导入向导”从 Microsoft Excel 文件导入数据。请参见第 202 页的“使用“Excel 导入向导”导入数据”。该方法对任何 Excel 文件都很方便。
- 将数据从 Microsoft Excel 复制并粘贴到新的 JMP 数据表中。请参见第 203 页的“复制和粘贴 Excel 中的数据”。您可以对小型数据文件使用该方法。

有关如何从 Microsoft Excel 导入数据的详细信息，请参见《使用 JMP》。

使用“Excel 导入向导”导入数据

1. 选择**帮助 > 样本数据库**，然后打开位于 Samples/Import Data 文件夹中的 Fill Weights.xlsx。
   该文件在 “Excel 导入向导”中打开。
2. 在**列标题起始于行**旁边键入 3。
   在 Excel 文件中，第 1 行包含有关该表的信息，第 2 行为空。列标题信息从第 3 行开始。

3. 为**带列标题的行数**键入 2。

   在 Excel 文件中，第 3 行和第 4 行都包含列标题信息。

4. 点击**导入**。

图 6.39 使用 “Excel 导入向导”创建的 JMP 表

	Weights-ID	Weights-Line A	Weights-ID 2	Weights-Line B	Weights-ID 3	Weights-Line C
1	215	12.42	705	13.63	254	11.73
2	287	12.49	670	12.56	282	11.40
3	381	12.80	715	12.87	938	12.78
4	•	•	683	13.09	597	12.19
5	•	•	514	13.31	179	12.25
6	•	•	517	12.64	•	•
7	•	•	946	12.75	•	•

数据包含在七行中，每行中都显示多个 ID。对于这三条生产线中的每一条，都有一个“ID”列和一个 “重量”列，总共有六列。

请注意，“ID”列名中的 “Weights”部分不是必要的，有误导作用。您可以现在重命名这些列，但是堆叠数据之后再重命名这些列将更为高效。

5. 前进到第 204 页的 “堆叠数据”。

复制和粘贴 Excel 中的数据

1. 在 Microsoft Excel 中打开 Fill Weights.xlsx。
2. 选择表内数据，但排除不必要的 “重量”标题。
3. 右击并选择**复制**。
4. 在 JMP 中，选择**文件 > 新建 > 数据表**。
5. 选择**编辑 > 带列名一起粘贴**。

   若剪贴板上的选择内容中包含列名，则使用**编辑 > 带列名一起粘贴**选项。

图 6.40 使用 “带列名一起粘贴”创建的 JMP 表

	ID	Line A	ID 2	Line B	ID 3	Line C
1	215	12.42	705	13.63	254	11.73
2	287	12.49	670	12.56	282	11.40
3	381	12.80	715	12.87	938	12.78
4	•	•	683	13.09	597	12.19
5	•	•	514	13.31	179	12.25
6	•	•	517	12.64	•	•
7	•	•	946	12.75	•	•

6. 前进到第 204 页的 “堆叠数据”。

堆叠数据

使用 “堆叠”选项在新数据表的每行中放入一个观测。有关 “堆叠”选项的详细信息，请参见《使用 JMP》。

1. 在 JMP 数据表中，选择**表 > 堆叠**。
2. 选择全部六列并点击**堆叠列**。
3. 选择**多序列堆叠**。

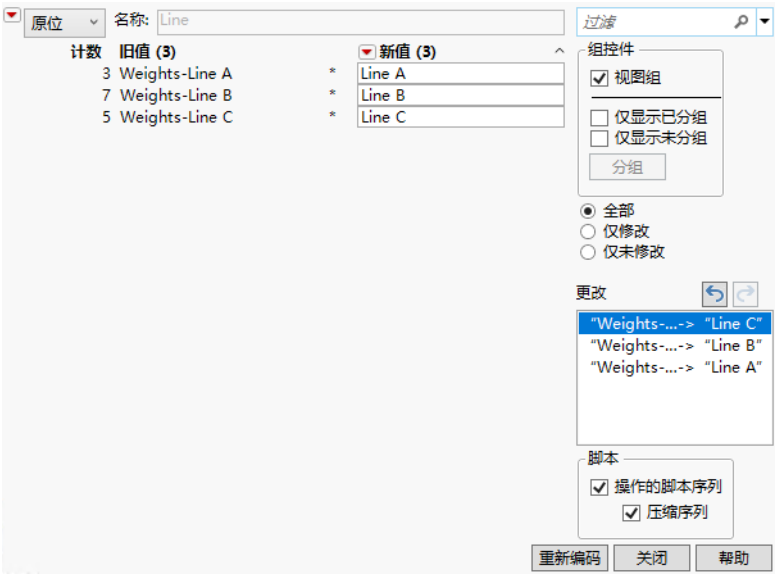
   您需要堆叠两个序列，分别是 “ID”和 “Line”，所以您无需更改默认设置为 2 的 “序列数”。包含序列的列不连续。这些列交替显示（ID、Line A、ID、Line B、ID、Line C）。出于此原因，您未选中 “连续”。
4. 取消选择**按行堆叠**。
5. 选择**剔除缺失行**。
6. 在**输出表名称**旁边输入**堆叠**。
7. 点击**确定**。

   在新数据表中，**数据**和**数据 2** 是包含 “ID”和 “重量”数据的列。
8. 右击**标签**列标题，然后选择**删除列**。

   标签列中的条目曾是导入的数据表中麦片盒 ID 的列标题。不再需要这些条目。
9. 通过双击列标题重命名每一列。
   - “**数据**”改为“**ID**”
   - “**标签 2**”改为“**生产线**”
   - “**数据 2**”改为“**重量**”
10. 在列面板中，点击 ID 左侧的图标并选择**名义型**。

    尽管 ID 被指定为数字，但它是一个标识符，应在建模时被视为名义型。这在本例中不是问题，但最好为列指定适合的建模类型。
11. （仅在使用**文件 > 打开**从 Excel 导入数据时适用。）请执行以下操作：
    1. 点击**生产线**列标题选择该列，然后选择**列 > 重新编码**。
    2. 点击**新建列**并选择**原位**。
    3. 更改**新建值**列中的值，以便与下面的图 6.41 中的值匹配。

图 6.41 对列值重新编码



4. 点击**重新编码**。

新的数据表现在正确调整结构以适应 JMP 分析。每行都包含单个麦片盒的数据。第一列提供麦片盒 ID，第二列提供生产线，第三列提供麦片盒的重量（图 6.42）。

图 6.42 重新编码的数据表

	ID	生产线	重量
1	215	Line A	12.42
2	287	Line A	12.49
3	381	Line A	12.8
4	705	Line B	13.63
5	670	Line B	12.56
6	715	Line B	12.87
7	683	Line B	13.09
8	514	Line B	13.31
9	517	Line B	12.64
10	946	Line B	12.75
11	254	Line C	11.73
12	282	Line C	11.4
13	938	Line C	12.78
14	597	Line C	12.19
15	179	Line C	12.25

执行单因子分析

示例的这一部分包含以下任务：

- 执行单因子方差分析以检验三条生产线中填充重量均值中的差值。
- 获取比较环以探索哪些生产线可能不同。

- 若想要重新称重或进一步检查生产线上的麦片盒，可通过 ID 为点添加标签。

开始之前，验证您使用的是**堆叠**数据表。

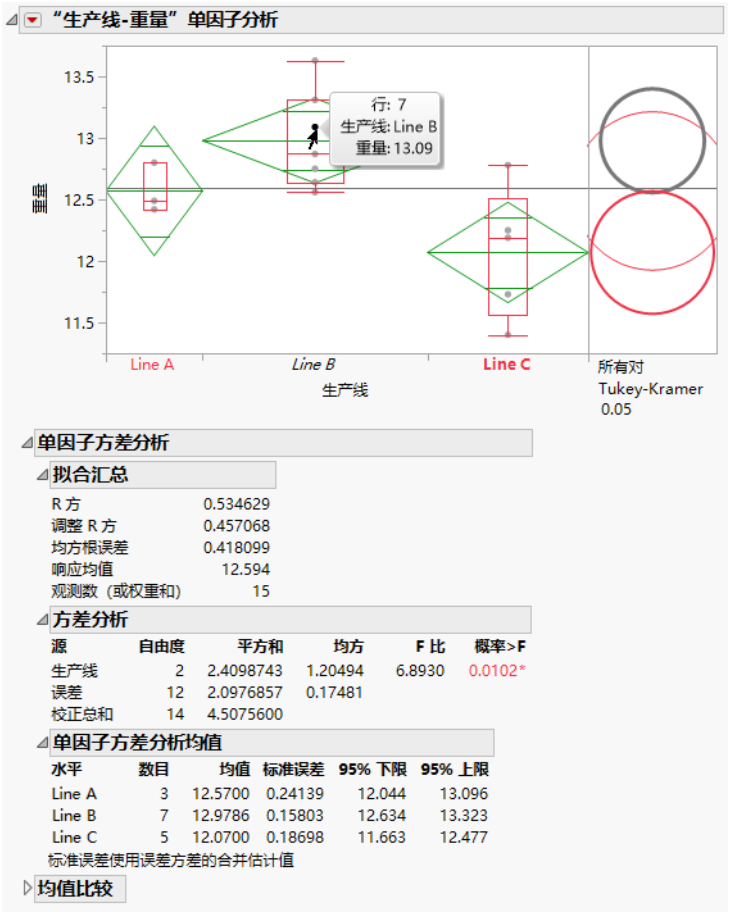
1. 选择**分析 > 以 X 拟合 Y**。
2. 选择**重量**并点击 **Y，响应**。
3. 选择**生产线**并点击 **X，因子**。
4. 点击**确定**。
5. 点击“‘生产线 - 重量’单因子分析”旁边的红色小三角并选择**均值 / 方差分析**。

   图中的均值菱形显示生产线均值的 95% 置信区间。落在均值菱形之外的点可能看起来像离群值。但实际上它们不是。要查看这种情况，请在图中添加箱线图。
6. 点击“‘生产线 - 重量’单因子分析”旁边的红色小三角并选择**显示选项 > 箱线图**。

   所有点都落在箱线图边界内。因此这些点不是离群值。
7. 从数据表的列面板中，右击 ID 并选择**添加标签 / 撤销标签**。
8. 在该图中，悬停在各点上方以查看其 ID 值，以及其**生产线**和**重量**数据 （图 6.43）。
9. 点击“‘生产线 - 重量’单因子分析”旁边的红色小三角并选择**比较均值 > 所有对， Tukey HSD**。

   比较环显示在图右侧的面板中。
10. 点击底部的比较环。

图 6.43 按生产线划分的重量的单因子分析



单因子方差分析

拟合汇总

R 方	0.534629
调整 R 方	0.457068
均方根误差	0.418099
响应均值	12.594
观测数（或权重和）	15

方差分析

源	自由度	平方和	均方	F 比	概率>F
生产线	2	2.4098743	1.20494	6.8930	0.0102*
误差	12	2.0976857	0.17481		
校正总和	14	4.5075600			

单因子方差分析均值

水平	数目	均值	标准误差	95% 下限	95% 上限
Line A	3	12.5700	0.24139	12.044	13.096
Line B	7	12.9786	0.15803	12.634	13.323
Line C	5	12.0700	0.18698	11.663	12.477

标准误差使用误差方差的合并估计值

均值比较

在“方差分析”报表中，*p* 值 0.0102 为均值不全相同提供了证据。在该图中，“Line C”的比较环处于选定状态并显示为红色。由于“Line B”的圆环显示为深灰色，“Line C”的均值在 0.05 显著性水平下不同于“Line B”的均值。“Line A”与“Line B”的均值未显示统计上的显著性差异。

图中所示的均值菱形跨均值的 95% 置信区间。95% 置信区间的数值边界在“单因子方差分析均值”报表中提供。这两者都表明“Line B”和“Line C”的置信区间未包含目标填充重量 12.5：“Line B”看起来填充过度，“Line C”看起来填充不足。对于这两条生产线，必须去了解导致未满足目标填充重量的底层原因。

“单因子”平台的统计详细信息

- “比较环”
- “功效”
- ““拟合汇总”报表”
- “等方差检验”
- “非参数检验统计量”

比较环

比较环是多重比较检验中**最小显著性差异** (LSD) 的图形表示。该最小显著性差异是 Student *t* 统计量与两个均值差值的标准误差的乘积，定义如下：

$$\text{LSD} = t_{\alpha/2}\text{std}(\hat{\mu}_1 - \hat{\mu}_2)$$

根据以下关系计算两个独立均值的差值的标准误差：

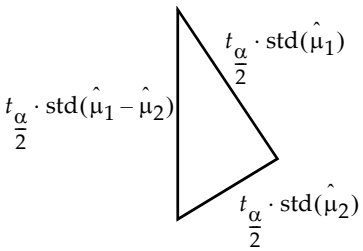
$$[\text{std}(\hat{\mu}_1 - \hat{\mu}_2)]^2 = [\text{std}(\hat{\mu}_1)]^2 + [\text{std}(\hat{\mu}_2)]^2$$

均值不相关时，这些量之间具有以下关系：

$$\text{LSD}^2 = [t_{\alpha/2}\text{std}((\hat{\mu}_1 - \hat{\mu}_2))]^2 = [t_{\alpha/2}\text{std}(\hat{\mu}_1)]^2 + [t_{\alpha/2}\text{std}\hat{\mu}_2]^2$$

这些平方值构成了勾股定理关系，图 6.44 中所示的直角三角形对此进行了说明。

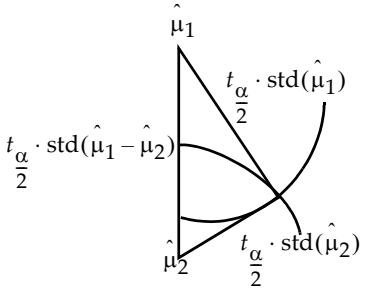
图 6.44 两个均值之间的差值的关系



该直角三角形的斜边是比较均值的量尺。仅当实际差值大于直角三角形的斜边 (LSD) 时，均值存在显著差异。

假定您在边界线上正好有两个均值，其中实际差值与最小显著性差异相同。使用在垂直尺度上测量的均值为顶点绘制三角形。此外，在每个均值周围绘制圆，使每个圆的直径等于该均值的置信区间。

图 6.45 *t* 检验统计量的几何关系



每个圆的半径是三角形相应直角边的长度，即 $t_{\alpha/2}\text{std}(\hat{\mu}_i)$。

两个直角边相交成一直角，这些圆也在同一点相交成同一直角，给出以下关系：

- 若均值差值刚好等于最小显著性差异，则每个均值周围的置信区间圆相交为直角。即切线的角为直角。

现在，考虑均值差值大于或小于最小显著性差异时这些圆应如何相交：

- 若圆相交导致外角大于直角，则均值**不**存在显著差异。若圆相交导致外角小于直角，则均值**存在**显著差异。外角小于 90 度表示均值分开的距离超过最小显著性差异。
- 若圆不相交，则均值存在显著差异。若圆嵌套，则均值没有显著差异（图 6.11）。

同样的图形方法适用于很多多重比较检验，用 Student *t* 的不同概率分位数值替代即可。

功效

要计算功效，可以使用非中心 *F* 分布。公式 (O’Brien and Lohr 1984) 定义如下：

功效 = 概率 ($F > F_{crit}, \nu_1, \nu_2, nc$)

其中：

F 服从非中心 $F(nc, \nu_1, \nu_2)$ 分布，$F_{crit} = F_{(1-\alpha, \nu1, \nu2)}$ 是 *F* 分布的 1 - α 分位数，该分布的自由度为 ν_1 和 ν_2。

$\nu_1 = r - 1$ 是分子自由度。

$\nu_2 = r(n - 1)$ 是分母自由度。

n 是每个组的观测数。

r 是组数。

$nc = n(CSS)/\sigma^2$ 是非中心参数。

$CSS = \sum_{g=1}^{r} (\mu_g - \mu)^2$ 是校正的平方和。

μ_g 是第 g 个组的均值。

μ 是总体均值。

σ^2 根据均方误差 (MSE) 来估计。

“拟合汇总”报表

R 方

使用模型的 “方差分析”报表的量，按以下公式计算任意连续响应拟合的 R^2：

$$\frac{\text{平方和(模型)}}{\text{平方和(校正总和)}}$$

调整 R 方

调整 R 方是均方 （而非平方和）之比，按以下公式计算：

$$1-\frac{\text{均值平方(误差)}}{\text{均值平方(校正总和)}}$$

误差的均方可在“方差分析”报表中找到，**校正总和**的均方计算方法为：**校正总和**平方和除以相应的自由度。请参见第 156 页的““方差分析”报表”。

等方差检验

F 比

O'Brien 检验构造了一个因变量，使得新变量的组均值等于原始响应的组样本方差。O'Brien 变量的计算公式为：

$$r_{ijk}=\frac{(n_{ij}-1.5)n_{ij}(y_{ijk}-\overline{y_{ij}})^2-0.5s_{ij}^2(n_{ij}-1)}{(n_{ij}-1)(n_{ij}-2)}$$

其中 n 表示 y_{ijk} 观测数。

Brown-Forsythe 是来自对 $z_{ij}=\left|y_{ij}-\tilde{y}_i\right|$ 进行方差分析的模型 F 统计量，其中 $\tilde{y}_i$ 是第 i 个水平的响应中位数。

Levene F 是来自对 $z_{ij}=\left|y_{ij}-\overline{y_{i.}}\right|$ 进行方差分析的模型 F 统计量，其中 $\overline{y_{i.}}$ 是第 i 个水平的响应均值。

Bartlett 检验的计算公式如下：

$$T = \frac{v\log\left(\sum_i \frac{v_i}{v} s_i^2\right) - \sum_i v_i \log(s_i^2)}{1 + \left(\frac{\sum_i \frac{1}{v_i} - \frac{1}{v}}{3(k-1)}\right)} \quad \text{其中 } v_i = n_i - 1,\ v = \sum_i v_i$$

n_i 是第 i 个水平的计数，s_i^2 是第 i 个水平的响应样本方差。Bartlett 统计量服从 χ^2 分布。将卡方检验统计量除以自由度得到报告的 F 值。

Welch 检验 F 比

Welch 检验 F 比的计算公式如下：

$$F = \frac{\left[\frac{\sum_i w_i (\bar{y}_i - \tilde{y}_{..})^2}{k-1}\right]}{\left\{1 + \frac{2(k-2)}{k^2-1}\left[\sum_i \frac{\left(1 - \frac{w_i}{u}\right)^2}{n_i - 1}\right]\right\}} \quad \text{其中 } w_i = \frac{n_i}{s_i^2},\ u = \sum_i w_i,\ \tilde{y}_{..} = \sum_i \frac{w_i \bar{y}_{i.}}{u},$$

n_i 是第 i 个水平的计数，$\bar{y}_{i.}$ 是第 i 个水平的响应均值，s_i^2 是第 i 个水平的响应样本方差。

Welch 检验分母自由度

Welch 近似分母自由度定义如下：

$$df = \frac{1}{\left(\frac{3}{k^2-1}\right)\left[\sum_i \frac{\left(1 - \frac{w_i}{u}\right)^2}{n_i - 1}\right]}$$

其中 w_i、n_i 和 u 定义同 F 比公式。

非参数检验统计量

本节提供在 Wilcoxon、中位数、Van der Waerden 和 Friedman 秩检验中使用的检验统计量的公式。

符号

这些检验基于得分并使用以下符号。

$j = 1,\ldots, n$ 整个样本中的观测。

$i = 1,\ldots, k$ X 的水平，其中，k 为水平总数。

$n_1, n_2,\ldots, n_k$ X 的 k 个水平中每个水平的观测数。

R_j 第 j 个观测的中秩。**中秩**在观测无结值时即该观测的秩，在有结值时即该观测的平均秩。

α 用于定义各种检验得分的中秩的函数。

在启动窗口指定了“区组”变量时将使用以下符号。

$b = 1,\ldots, B$ 分区组变量的水平，其中，B 是区组总数。

R_{bi} 区组 b 内第 i 个水平的中秩。

函数 α 按如下方式定义得分：

Wilcoxon 得分

$$\alpha(R_j) = R_j$$

中位数得分

$$\alpha(R_j) = \begin{cases} 1 \text{ 若 } R_j > \text{中位数} \\ 0 \text{ 若 } R_j < \text{中位数} \\ t \text{ 若 } R_j = \text{中位数} \end{cases}$$

用 n_t 表示中位数处的结值观测数，用 n_u 表示大于中位数的观测数。则 t 通过以下方式得出：

$$t = \frac{\text{floor}(n/2) - n_u}{n_t}$$

van der Waerden 得分

$$\alpha(R_j) = \text{标准正态分位数}(R_j/(n+1))$$

Friedman 秩得分

$$\alpha(R_{bi}) = R_{bi}$$

双样本正态近似

仅当 X 恰好有两个水平时，才提供基于正态近似的检验。本节中使用的符号在第 212 页的 “符号”中定义。“双样本正态近似”报表中显示的统计量定义如下。

S 统计量 S 是较小组中观测的值 $\alpha(R_j)$ 的总和。若 X 的两个水平有相同的观测数，则 S 值对应于值排序中 X 的最后一个水平。

Z Z 的值定义如下：

$$Z = (S - E(S)) / \sqrt{Var(S)}$$

注意：Wilcoxon 检验添加连续校正。若 (*S* - *E(S)*) 大于 0，则从分子中减去 0.5。若 (*S* - *E(S)*) 小于 0，则给分子加上 0.5。

***E*(*S*)** 原假设下 S 的期望值。用 n_l 表示水平较小组中的观测数，或表示值排序中的最后一个水平中的观测数（在两个组具有相同的观测数时）：

$$E(S) = \frac{n_l}{n} \sum_{j=1}^{n} \alpha(R_j)$$

***Var*(*S*)** 定义 *ave* 为所有观测的平均得分。那么 S 的方差定义如下：

$$Var(S) = \frac{n_1 n_2}{n(n-1)} \sum_{j=1}^{n} (\alpha(R_j) - ave)^2$$

Friedman 秩检验的双样本正态近似

使用 Friedman 秩检验时，双样本正态近似的计算与上文相同，只不过 S 的方差不同。S 的方差计算如下：

$$Var(S) = \frac{B}{(n-1)} \sum_{j=1}^{n} (\alpha(R_j) - ave)^2$$

单因子卡方近似

注意：基于 Wilcoxon 得分的卡方检验称为 Kruskal-Wallis 检验。

本节中使用的符号在第 212 页的 “符号”中定义。以下数量用于计算卡方统计量：

T_i X 的第 *i* 个水平的得分合计。

***E*(T_i)** 在水平中无差值的原假设下水平 *i* 的总得分的期望值，定义如下：

$$E(T_i) = \frac{n_i}{n} \sum_{j=1}^{n} \alpha(R_j)$$

Var(T) 定义 *ave* 为所有观测的平均得分。那么 T 的方差定义如下：

$$Var(T) = \frac{1}{(n-1)} \sum_{j=1}^{n} (\alpha(R_j) - ave)^2$$

检验统计量的值如下所示。该统计量是 *k - 1* 自由度下的渐近卡方。

$$C = \left(\sum_{i=1}^{k} (T_i - E(T_i))^2 / n_i \right) / Var(T)$$

Friedman 秩检验的单因子卡方近似

Friedman 秩检验的卡方检验统计量计算如下：

$$C = \frac{\sum_{i=1}^{k} (T_i - E(T_i))^2 / n_i}{\frac{1}{(k-1)} \sum_{j=1}^{n} (\alpha(R_j) - ave)^2 / n_i}$$

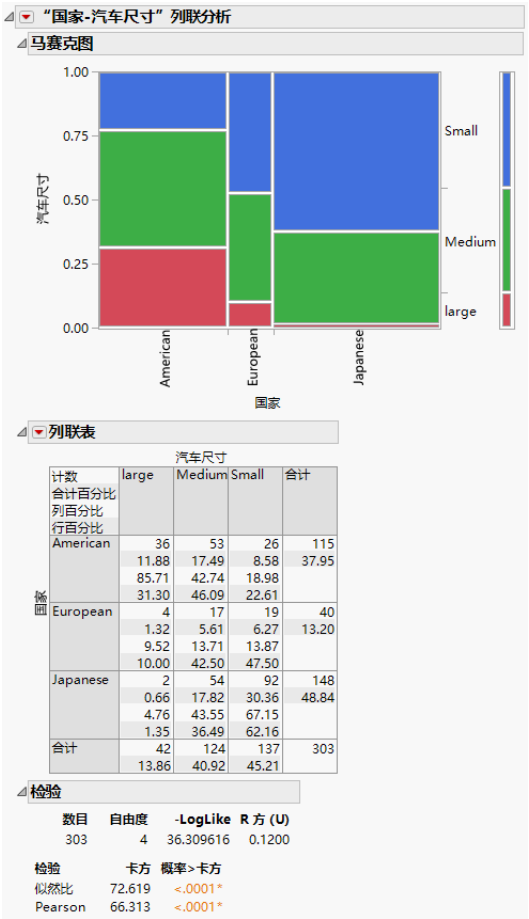
第 7 章

列联分析

检查两个分类变量之间的关系

使用“列联”平台可调查两个分类变量之间的关系。分类变量可以是名义型或有序型。分析结果包括马赛克图、频数统计和比例的列联表以及卡方显著性检验。您可以对数据以交互方式执行其他分析和检验，如比例均值分析、对应分析和关联分析等。“列联”平台是“以 X 拟合 Y”平台的**以分类变量拟合分类变量**特质。

图 7.1 列联分析示例



国家	计数 合计百分比 列百分比 行百分比	large	Medium	Small	合计
American		36	53	26	115
		11.88	17.49	8.58	37.95
		85.71	42.74	18.98	
		31.30	46.09	22.61	
European		4	17	19	40
		1.32	5.61	6.27	13.20
		9.52	13.71	13.87	
		10.00	42.50	47.50	
Japanese		2	54	92	148
		0.66	17.82	30.36	48.84
		4.76	43.55	67.15	
		1.35	36.49	62.16	
合计		42	124	137	303
		13.86	40.92	45.21	

目录

列联分析示例 218
启动“列联”平台 219
数据格式 219
“列联”报表 220
“列联”平台选项 221
马赛克图 222
弹出菜单 223
列联表 224
列联表的说明 225
检验 226
“检验”报表的说明 226
Fisher 精确检验 226
比例均值分析 227
对应分析 227
了解对应分析图 227
“对应分析”选项 227
“详细信息”报表 228
Cochran-Mantel-Haenszel 检验 228
一致性统计量 229
相对风险 229
双样本比例检验 229
关联测量 230
Cochran Armitage 趋势检验 231
精确检验 231
“列联”平台的更多示例 232
“比例均值分析”的示例 232
“对应分析”的示例 233
“Cochran Mantel Haenszel 检验”的示例 235
“一致性统计量”选项的示例 236
“相对风险”选项的示例 237
“双样本比例检验”的示例 239
“关联测量”选项的示例 240
“Cochran Armitage 趋势检验”的示例 240
“列联”平台的统计详细信息 241
“一致性统计量”选项 241
“优势比”选项 242

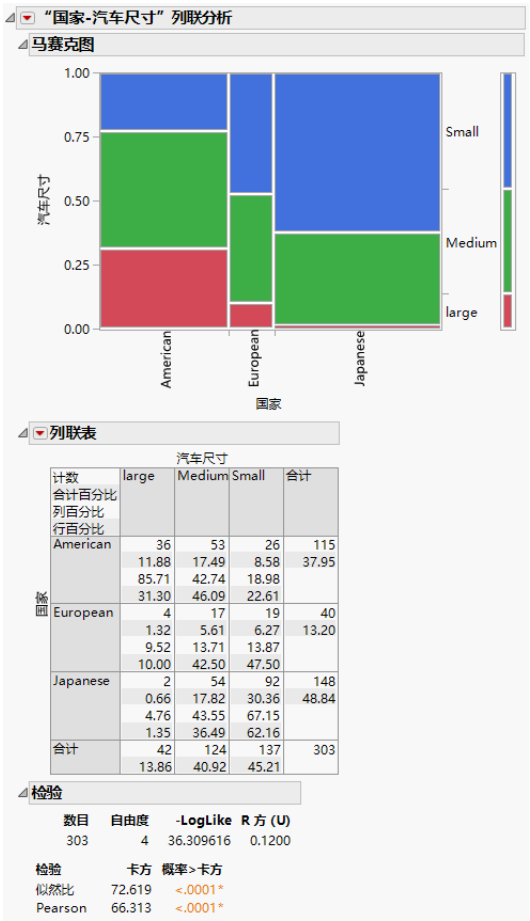
检验报表 242
对应分析中的“详细信息”报表 243

列联分析示例

了解如何检查 JMP 中两个分类变量之间的关系。本例使用从汽车调查收集的数据。数据中包括响应者特性：性别、婚姻状况和年龄。数据中还包括响应者汽车的特性：原产地、大小和类型。检查汽车尺寸 （小型、中型和大型）与汽车原产地之间的关系。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**汽车尺寸**并点击 **Y，响应**。
4. 选择**国家**并点击 **X，因子**。
5. 点击**确定**。

图 7.2 列联分析示例



从马赛克图和图例中可看到：

- 日本汽车很少属于“大型”类别。
- 大多数欧洲汽车属于“小型”和“中型”类别。
- 大多数美国汽车属于“大型”和“中型”类别。

启动“列联”平台

要执行列联分析，请执行以下操作：

1. 选择**分析 > 以 X 拟合 Y**。
2. 为 **Y，响应**输入名义型或有序型列。
3. 为 **Y，因子**输入名义型或有序型列。

图 7.3 “以 X 拟合 Y”启动窗口

“列联”一词出现在图的上方，指示您正在执行列联分析。

注意：您还可以从“JMP 起始页”窗口启动列联分析。选择**视图 > JMP 起始页 > 基本 > 列联**。

有关该启动窗口的详细信息，请参见第 97 页的““以 X 拟合 Y”介绍”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

数据格式

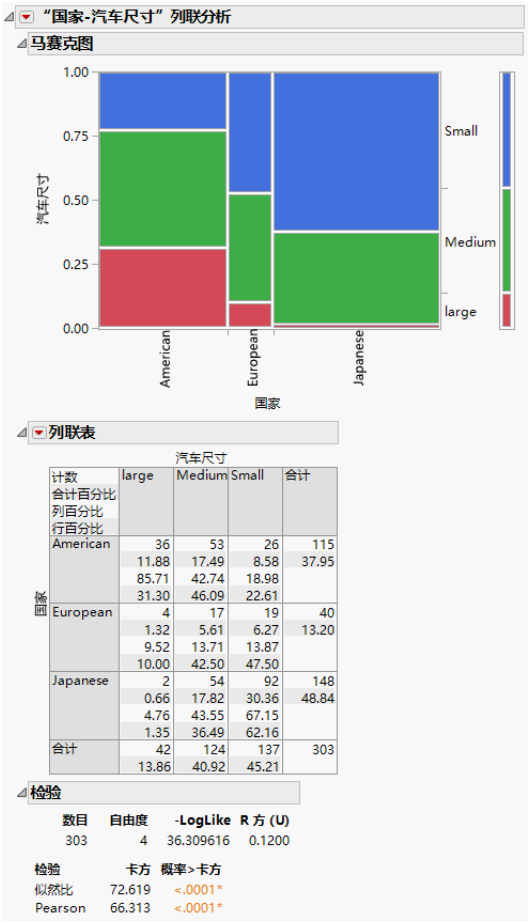
分类数据经常以汇总格式呈现，X 和 Y 变量水平的每种组合在数据表中都只对应一行。在这种情况下，使用“频数”或“权重”变量可指示每行表示的观测数。有关汇总分类数据的示例，请参见第 232 页的““比例均值分析”的示例”。

“列联”报表

“列联”报表最初显示马赛克图、列联表和 “检验”报表。您可以使用红色小三角菜单选项运行其他分析和检验。有关所有这些报表和选项的详细信息，请参见第 221 页的 ““列联”平台选项”。

要生成图 7.4 中所示的图，请按第 218 页的 “列联分析示例”中的说明操作。

图 7.4 “列联”报表示例



注意：在数据表中排除的所有行也在马赛克图中隐藏。

“列联”平台选项

使用“列联分析”红色小三角菜单中的选项执行其他分析。

注意：若您指定了多个 Y 变量，将显示“拟合组”菜单。使用菜单选项，您可以排列报表或按 R 方将它们排序。请参见《拟合线性模型》。

马赛克图 列联表中数据的图形表示法。请参见第 222 页的 “马赛克图”。

列联表 双因子频数表。每个因子水平对应一行，每个响应水平对应一列。请参见第 224 页的 “列联表”。

检验 类似于连续数据的“方差分析”表。这些检验显示响应水平比率在 X 各水平上是相同的。请参见第 226 页的 “检验”。

设置 α 水平 更改置信区间中使用的 alpha 水平。选择一个常见值 (0.10, 0.05, 0.01) 或使用**其他**选项选择特定值。

比例均值分析 （仅当响应恰好具有两个水平时才显示。）比较 X 水平的响应比例与总体响应比例。请参见第 227 页的 “比例均值分析”。

对应分析 显示频数表的哪些行或列具有相似的计数模式。在对应分析图中，列联表的每行和每列对应一个点。请参见第 227 页的 “对应分析”。

Cochran Mantel Haenszel 检验 检验在对第三个分类分区组后两个分类变量之间是否存在关系。请参见第 228 页的 “Cochran-Mantel-Haenszel 检验”。

一致性统计量 （仅当 X 变量和 Y 变量具有相同水平时才显示。）显示 Kappa 统计量 (Agresti 1990)、它的标准误差、置信区间、假设检验和 Bowker 对称性检验 （也称为 McNemar 检验）。请参见第 229 页的 “一致性统计量”。

相对风险 （仅当 X 变量和 Y 变量都有两个水平时才显示。）计算风险比。请参见第 229 页的 “相对风险”。

优势比 （仅当 X 变量和 Y 变量都有两个水平时才显示。）生成优势比的报表。请参见第 242 页的 “‘优势比’选项”。

报表还给出该比率的置信区间。您可以使用**设置 α 水平**选项更改 alpha 水平。

双样本比例检验 （仅当 X 变量和 Y 变量都有两个水平时才显示。）执行双样本比例检验。该检验比较 X 变量两个水平间 Y 变量的比例。请参见第 229 页的 “双样本比例检验”。

关联测量 说明列联表中这些变量之间的关联。请参见第 230 页的 “关联测量”。

Cochran Armitage 趋势检验 （仅当一个变量具有两个水平且另一个变量为有序型时才显示。）对单个变量的各水平间进行二项式比例趋势检验。请参见第 231 页的 “Cochran Armitage 趋势检验”。

JMP PRO **精确检验** 提供以下检验的精确检验：

- Fisher 检验
- Cochran Armitage 趋势检验
- 一致性统计量

请参见第 231 页的 “精确检验”。

显示选项 > 水平马赛克图 水平或垂直旋转马赛克图。

制成数据表 从报表创建 JMP 数据表。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

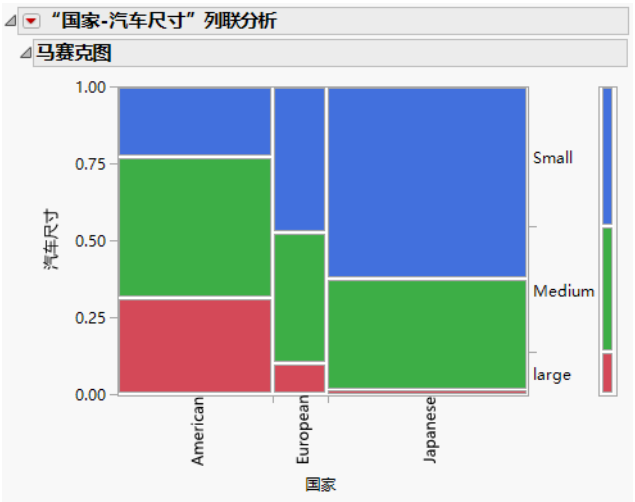
保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

马赛克图

马赛克图是双因子频数表或列联表的图形表示法。马赛克图分为一些矩形，每个矩形的垂直长度与 X 变量每个水平中 Y 变量的比例成比例。马赛克图由 Hartigan and Kleiner (1981) 引入，后来由 Friendly (1994) 进行了改进。

要生成图 7.5 中所示的图，请按第 218 页的“列联分析示例”中的说明操作。

图 7.5 马赛克图的示例



关于马赛克图，请注意以下事项：

- 水平轴上的比例表示 X 变量（即国家）每个水平的观测数。
- 垂直轴右侧上的比例表示组合水平（美国、欧洲和日本）的小型、中型和大型汽车的总体比例。
- 左侧垂直轴的尺度显示响应概率。整个轴等同于概率为 1（代表总样本）。

点击马赛克图中的矩形将突出显示所选内容以及相关数据表中的相应数据。

通过以下两种方式之一拖放变量，以此替换马赛克图中的变量：将一个轴上的变量拖放到另一个轴上来交换现有变量；或者，点击相关数据表的列面板中的某个变量并将其拖到轴上。

弹出菜单

右击马赛克图可更改颜色并为单元格添加标签。

设置颜色　显示当前给水平分配的颜色。请参见第 223 页的 “设置颜色”。

添加单元格标签　指定要在马赛克图中绘制的标签。选择以下选项之一：

无标签　不显示标签，并且删除所有其他选项。

将计数用作标签　在每个单元格中显示观测数。

将百分比用作标签　在每个单元格中显示观测百分比。

将值用作标签　显示与每个单元格对应的 Y 变量的水平。

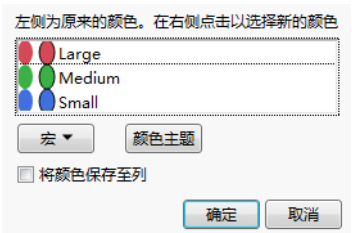
将行用作标签　显示单元格所表示的所有行对应的行标签。

注意：有关右击选项其余部分的说明，请参见《使用 JMP》。

设置颜色

选择**设置颜色**选项后，将显示“选择值的颜色”窗口。

图 7.6　“选择值的颜色”窗口



默认马赛克颜色取决于响应列是有序型还是名义型的，以及是否具有现有的“值颜色”列属性。要更改任意水平的颜色，请点击颜色第二列中的椭圆并选择新颜色。

“选择值的颜色”窗口的说明

宏 计算任意两个水平之间的颜色梯度：

在端点之间渐变 将一个梯度应用到变量的所有水平。

在选定点之间渐变 将一个颜色梯度应用到您选定的水平。可以通过将光标拖动到要选择的水平上，或通过按 Shift 键并点击第一个和最后一个水平来选择水平范围。

反转颜色 反转颜色顺序。

恢复以前的颜色 恢复所做的任何更改。

颜色主题 基于颜色主题更改每个值的颜色。

将颜色保存至列 若更改默认颜色主题，请选中该复选框，则新的列属性 （**值颜色**）将添加到相关数据表中的列。要从数据表编辑该属性，请选择**列 > 列信息**。

列联表

列联表是双因子频数表。每个因子水平对应一行，每个响应水平对应一列。

要生成图 7.7 中所示的图，请按第 218 页的“列联分析示例”中的说明操作。

图 7.7 列联表示例

列联表

		汽车尺寸			
	计数 合计百分比 列百分比 行百分比	large	Medium	Small	合计
原产国	American	36 11.88 85.71 31.30	53 17.49 42.74 46.09	26 8.58 18.98 22.61	115 37.95
	European	4 1.32 9.52 10.00	17 5.61 13.71 42.50	19 6.27 13.87 47.50	40 13.20
	Japanese	2 0.66 4.76 1.35	54 17.82 43.55 36.49	92 30.36 67.15 62.16	148 48.84
	合计	42 13.86	124 40.92	137 45.21	303

请注意有关列联表的以下事项：

- “计数”、“合计百分比”、“列百分比”和“行百分比”对应于具有行标题和列标题的每个单元格（例如“American”和“Large”下的单元格）内的数据。
- 最后一列包含每行的总计数和每行的百分比。
- 最下面一行包含每列的总计数和每列的百分比。

在图 7.7 中，关注美国生产的大型汽车。下表解释您使用列联表得到关于这些汽车的结论。

表 7.1 基于列联表示例的结论

数字	说明	表中的标签
36	美国生产的大型汽车个数	计数
11.88%	所有汽车中美国生产的大型汽车所占的百分比 (36/303)[a]。	合计百分比
85.71%	大型汽车中美国生产的所占的百分比 (36/42)[b]	列百分比
31.30%	美国汽车中大型尺寸所占的百分比 (36/115)[c]。	行百分比
37.95%	所有汽车中美国生产的所占的百分比 (115/303)。	（无）
13.86%	所有汽车中大型尺寸所占的百分比 (42/303)。	（无）

a. 303 是调查中的汽车总数。
b. 42 是调查中的大型汽车总数。
c. 115 是调查中的美国汽车总数。

提示：要显示或隐藏列联表中的数据，请从“列联表”旁边的红色小三角菜单中选择要显示或隐藏的选项。

列联表的说明

计数 单元格频数、边缘总频数和总计 （总样本大小）。

合计百分比 单元格计数和边缘合计占总计的百分比。

行百分比 每个单元格计数占行合计的百分比。

列百分比 每个单元格计数占列合计的百分比。

期望值 在独立性假设下每个单元格的期望频数 (*E*)。它由相应行合计与列合计之积除以总计得到。

偏差 观测到的单元格频数 (*O*) 减去期望的单元格频数 (*E*)。

单元格卡方值 按照以下公式为每个单元格计算的卡方值：$(O - E)^2 / E$。

列累积 累积列合计。

列累积百分比 累积列百分比。

行累积 累积行合计。

行累积百分比 累积行百分比。

检验

“检验”报表显示两个检验的结果，以确定响应水平比率在各 X 水平上是否相同。

要生成图 7.8 中所示的报表，请按第 218 页的“列联分析示例”中的说明操作。

图 7.8 “检验”报表示例

检验

数目	自由度	-LogLike	R 方 (U)
303	4	36.309616	0.1200

检验	卡方	概率>卡方
似然比	72.619	<.0001*
Pearson	66.313	<.0001*

请注意有关卡方统计量的以下事项：

- 两个分类变量均为响应（Y 变量）时，卡方统计量检验它们是否独立。
- 您可能有 X 变量固定的 Y 变量。在这种情况下，卡方统计量检验 Y 变量的分布在每个 X 水平上是否相同。

“检验”报表的说明

数目 总观测数。

自由度 记录与检验有关的自由度。自由度等于 $(c - 1)(r - 1)$，其中 c 为列数， r 为行数。

-LogLike 负对数似然，它测量拟合和不确定性（很像连续响应情况中的平方和）。请参见《拟合线性模型》。

R 方 (U) 总不确定性中归因于模型拟合的部分。

- R^2 为 1 表示因子可完全预测分类响应。
- R^2 为 0 表示使用模型取代固定背景响应率后，预测没有任何改进。

请参见第 242 页的“检验报表”。

检验 列出假设响应率在每个样本类别中相同的两个卡方统计检验。请参见第 242 页的“检验报表”。

概率 > 卡方 列出得到卡方值大于当响应和因子之间不存在任何关系时的计算值的概率。若两个变量都只有两个水平，还会显示单侧检验和双侧检验的 Fisher 精确概率。

Fisher 精确检验

该报表显示 2x2 表的 Fisher 精确检验结果。自动为 2x2 表显示这些结果。有关 Fisher 精确检验以及 r x c 表的检验的详细信息，请参见第 231 页的“精确检验”。

比例均值分析

比例均值分析 (ANOMP) 是一种多重比较方法，用于检验单个组内比例是否与总比例不同。

注意：有关“均值分析方法”的说明，请参见 Nelson et al.(2005) 文档。另见第 232 页的“比例均值分析”的示例”。

若响应有两个水平，您可以使用该选项比较 X 水平的响应比例与总体响应比例。该方法使用正态分布近似代替二项分布。因此，若样本大小过小，将在结果中显示警告。

以下选项出现在红色小三角菜单中：

设置 Alpha 水平 选择分析中使用的 alpha 水平。

显示汇总报表 生成一个报表，该报表显示 X 变量每个水平带决策限的响应比例。该报表指示是否已超过限制。

切换比例响应水平 更改分析中使用的响应类别。

显示选项 显示或隐藏决策限、决策限着色、中心线和点选项。

对应分析

“列联”平台中的“对应分析”选项提供了一种图形方法，用于显示频数表的哪些行或列具有相似计数模式。在对应分析图中，每行和每列对应一个点。当您有很多水平，很难从马赛克图得到有用信息时，可以使用“对应分析”。

注意：另见第 233 页的“对应分析”的示例”。

了解对应分析图

行轮廓可以定义为按行比率的集合，即行中计数除以该行的总计数。若两个行具有很相似的行轮廓，则它们在对应分析图中的点靠得很近。行点之间的平方距离近似与检验每对行齐性的卡方距离成比例。

列轮廓和行轮廓类似，因为问题是对称定义的。行点和列点之间的距离没有意义。但是，行和列与原点的方向有意义，并且其关系帮助解释图。

“对应分析”选项

使用“对应分析”红色小三角菜单中的选项可以向“列联”报表添加三维散点图，并向数据表添加列属性。

三维对应分析 生成三维散点图。

保存值顺序 采用按第一个对应得分系数排序的水平顺序并为 X 和 Y 列生成列属性。

“详细信息”报表

“详细信息”报表包含有关对应分析的统计信息并显示图中使用的值。

奇异值 提供列联表的奇异值分解。有关公式，请参见第 243 页的“对应分析中的“详细信息”报表”。

惯量 列出奇异值的平方，反映典型维度中解释的相对变异。

对应部分 涉及总惯量的惯量部分。

累积 显示惯量的累积部分。若前两个奇异值捕获绝大部分惯量，则二维对应分析图足以显示表中的关系。

X 变量 c1、 c2、 c3 在对应分析图 （图 7.11）上标绘的值。

Y 变量 c1、 c2、 c3 在对应分析图 （图 7.11）上标绘的值。

Cochran-Mantel-Haenszel 检验

Cochran-Mantel-Haenszel 检验评估在对第三个分类分区组后两个分类变量之间是否存在关系。

注意：另见第 235 页的“ “Cochran Mantel Haenszel 检验”的示例”。

得分的相关性 Y 或 X 为有序型或区间型时适用。备择假设是在分区组变量的至少一个水平上 Y 和 X 之间存在线性关联。

按列类别列出的行得分 Y 为有序型或区间型时适用。备择假设是在分区组变量的至少一个水平上 *r* 行的均值得分是不相等的。

按行类别列出的列得分 X 为有序型或区间型时适用。备择假设是在分区组变量的至少一个水平上 *c* 列的均值得分是不相等的。

常规类别关联 检验在分区组变量的至少一个水平上 X 和 Y 之间有某种类型的关联。

一致性统计量

两个变量有相同水平时，可以使用**一致性统计量**选项。该选项显示 Kappa 统计量 (Agresti 1990)、它的标准误差、置信区间、假设检验和 Bowker 对称性检验。

本节中给出的 Kappa 统计量和相关的 *p* 值是近似值。还提供精确一致性统计量。请参见第 231 页的 “精确检验”。

注意：有关统计详细信息，请参见第 241 页的 “‘一致性统计量’选项”。另见第 236 页的 “‘一致性统计量’选项的示例”。

Kappa 显示 Kappa 统计量。

标准误差 显示 Kappa 统计量的标准误差。

95% 下限 显示 Kappa 的置信区间下端点。

95% 上限 显示 Kappa 的置信区间上端点。

概率 >Z 显示 Kappa 单侧检验的 *p* 值。原假设检验 Kappa 是否等于零。

概率 >|Z| 显示 Kappa 双侧检验的 *p* 值。

卡方 显示 Bowker 检验的检验统计量。对于 Bowker 对称性检验，原假设是平方表中的概率满足对称性，即对于所有成对表单元格都满足 $p_{ij}=p_{ji}$。X 和 Y 都有两个水平时，该检验等于 McNemar 检验。

概率 > 卡方 显示 Bowker 检验的 *p* 值。

相对风险

注意：另见第 237 页的 “‘相对风险’选项的示例”。

使用**相对风险**选项计算 2x2 列联表的风险比。报表中还显示置信区间。您可以在 Agresti（1990 第 3.4.2 节）中找到有关该方法的详细信息。

选择**相对风险**选项时，将显示“选择相对风险类别”窗口。您可以选择单个响应和因子组合，或计算响应和因子水平的所有组合的风险比。

双样本比例检验

X 和 Y 变量均有两个水平时，您可以构造两个比例之间差值的置信区间。它还会计算与该置信区间对应的检验。

注意：另见第 239 页的 “‘双样本比例检验’的示例”。

说明 显示正在执行的检验。

比例差值 显示 X 变量的两个水平间的比例差值。

95% 下限 显示差值的置信区间下端点。基于调整的 Wald 置信区间。

95% 上限 显示差值的置信区间上端点。基于调整的 Wald 置信区间。

调整的 Wald 检验 显示双侧检验和单侧检验。

概率 显示检验的 *p* 值。

< 变量 > 响应中所关注的类别 选择要在检验中使用的响应水平。

关联测量

关联测量选项提供关联统计量。

注意：另见第 240 页的“关联测量”选项的示例”。

Gamma 基于一致的和不一致的对个数，忽略同分对。取值范围介于 -1 到 1 之间。

Kendall Tau-b 与 Gamma 类似，使用结值的校正值。取值范围介于 -1 到 1 之间。

Stuart Tau-c 与 Gamma 类似，使用表大小的调整值和结值的校正值。取值范围介于 -1 到 1 之间。

Somers D Tau-b 的非对称修改。

- C|R 指明行变量 X 视为自变量，列变量 Y 视为因变量。
- 同样，R|C 指明列变量 Y 视为自变量，行变量 X 视为因变量。

Somer D 与 Tau-b 的区别在于它仅在自变量有同分对时才使用结值的校正值。取值范围介于 -1 到 1 之间。

Lambda 不对称 对于 C|R 和 R|C 不同。

- 对于 C|R，被解释为在给出行变量 X 的知识后预测列变量 Y 能力可能的改进。
- 对于 R|C，被解释为在给出有关列变量 Y 的知识后预测行变量 X 能力可能的改进。

取值范围介于 0 到 1 之间。

Lambda 对称 粗略被解释为两个 Lambda 不对称测量的平均值。取值范围介于 0 到 1 之间。

不确定性系数

- 对于 C|R，是指列变量 Y 中由行变量 X 解释的不确定性比例。
- 对于 R|C，被解释为行变量 X 中由列变量 Y 解释的不确定性比例。

取值范围介于 0 到 1 之间。

对称式不确定性系数 两个不确定性系数测量的对称形式。取值范围介于 0 到 1 之间。

注意：

- 每个统计量与其标准误差和置信区间一起显示。
- Gamma、Kendall Tau-b、Stuart Tau-c 和 Somer D 是考虑变量 Y 是否随 X 增加而增加的有序型关联的测量。它们将观测对划分为一致或不一致。若具有较大 X 值的观测也具有较大的 Y 值，则对是一致的。若具有较大 X 值的观测具有较小的 Y 值，则对是不一致的。仅当两个变量均为有序型时，这些测量才适用。
- Lambda 和不确定性测量适用于有序型和名义型变量。

有关关联测量统计量的计算详细信息，请参见 SAS Institute Inc.(2020b)。以下参考书目也包含另外一些信息：

- Brown and Benedetti (1977)
- Goodman and Kruskal (1979)
- Kendall and Stuart (1979)
- Snedecor and Cochran (1980)
- Somers (1962)

Cochran Armitage 趋势检验

注意：另见第 240 页的““Cochran Armitage 趋势检验”的示例”。

该 Cochran Armitage 趋势对单个变量的各水平间进行二项式比例趋势检验。仅当一个变量具有两个水平且另一个变量为有序型时该检验才适用。两水平变量表示响应，另一个变量表示具有有序水平的说明变量。原假设为无趋势假设，即二项式比例对于说明变量的所有水平都相同。

该检验中给出的检验统计量和 *p* 值是近似值。还提供精确趋势检验。请参见第 231 页的“精确检验”。

精确检验

JMP PRO 下面的精确检验在“列联”平台中可用：

Fisher 精确检验 执行 *r* x *c* 表的 Fisher 精确检验。该检验检查两个变量之间的关联。Fisher 精确检验假设行合计和列合计是固定的，并且使用超几何分布来计算概率。

该检验不依赖于任何大样本分布假设。这意味着它适用于似然比检验和 Pearson 检验变得不太可靠时的情况 （如小样本或稀疏表）。

该报表包括以下信息：

表概率 (P) 给出所观测表的概率。它不是检验的 *p* 值。

双侧概率 ≤ P 提供双侧检验的 *p* 值。

对于 2x2 表，Fisher 精确检验自动执行，除非有一行或一列包含值全是零（在这种情况下，无法计算该检验）。请参见第 226 页的 “检验”。

Cochran Armitage 精确趋势检验 执行精确的 Cochran Armitage 趋势检验。仅当其中一个变量具有两个水平时该检验才可用。有关趋势检验的详细信息，请参见第 231 页的 “Cochran Armitage 趋势检验”。

精确一致性统计量 执行精确检验以检验变量之间的一致性。它是 Kappa 统计量的精确检验。仅当两个变量具有相同水平时该检验才可用。有关一致性检验的详细信息，请参见第 229 页的 “一致性统计量”。

注意：若总样本量大于 32767，列联表大于 2x2，则不提供精确检验选项。

“列联”平台的更多示例

- “比例均值分析”的示例
- “对应分析”的示例
- “Cochran Mantel Haenszel 检验”的示例
- “一致性统计量”选项的示例
- “相对风险”选项的示例
- “双样本比例检验”的示例
- “关联测量”选项的示例
- “Cochran Armitage 趋势检验”的示例

“比例均值分析”的示例

本例显示如何检查准时按预约时间到达的患者比例。Office Visits.jmp 样本数据表记录某地理区域六个诊所的预约是否准时的情况。对这六个诊所的每个诊所，从一周记录中选择 60 个随机预约。要被视为准时，患者必须在预定时间的 5 分钟内被带到检查室。

1. 选择**帮助** > **样本数据库**，然后打开 Office Visits.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**准时与否**并点击 **Y，响应**。
4. 选择**诊所**并点击 **X，因子**。
5. 选择**频数**并点击**频数**。
6. 点击**确定**。
7. 点击 “‘诊所 - 准时与否’列联分析”旁边的红色小三角并选择**比例均值分析**。
8. 点击 “比例均值分析”旁边的红色小三角并选择**显示汇总报表**和**切换比例响应水平**。

图 7.9 “比例均值分析”的示例

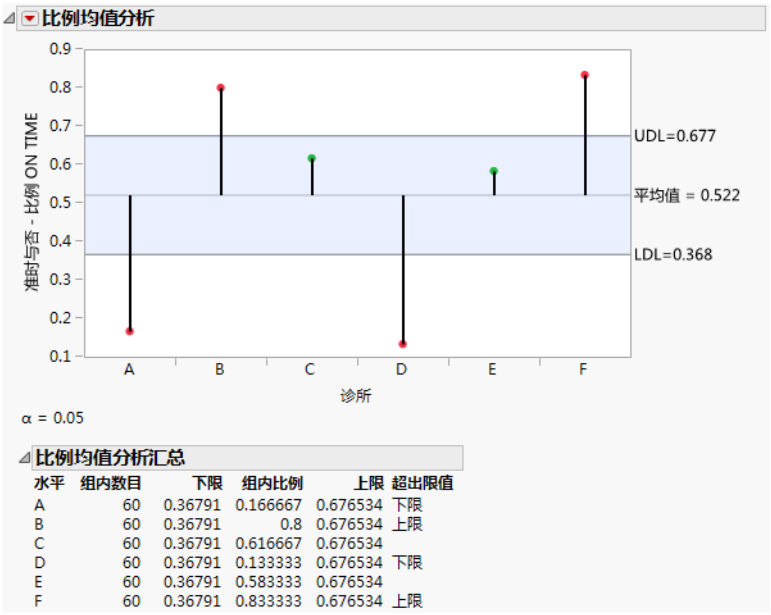


图 7.9 显示每个诊所准时的患者比例。从图 7.9 中可看到：

- 诊所 F 准时到达的比例最高，其次是诊所 B。
- 诊所 D 准时到达的比例最低，其次是诊所 A。
- 诊所 E 和诊所 C 接近平均值，并且没有超过决策限。

“对应分析”的示例

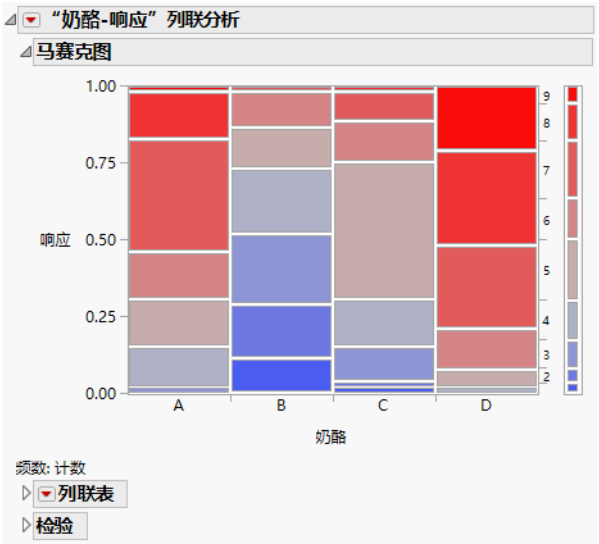
本例使用 Cheese.jmp 样本数据表。该表包含 Newell 奶酪品尝实验的数据；这些是在 McCullagh and Nelder (1989) 中报告的数据。该实验记录了四个不同奶酪添加剂的九个以上不同响应水平的计数。

1. 选择**帮助** > **样本数据库**，然后打开 Cheese.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**响应**并点击 **Y，响应**。

   响应值范围为 1 到 9，其中 1 为最不喜欢， 9 为最喜欢。
4. 选择**奶酪**并点击 **X，因子**。

   A、 B、 C 和 D 表示四种不同的奶酪添加剂。
5. 选择**计数**并点击**频数**。
6. 点击**确定**。

图 7.10 奶酪数据的马赛克图



从马赛克图，您注意到分布看起来不太相同。但是，了解九个水平上的马赛克图的意思比较困难。对应分析可帮助定义这类情况下的关系。

7. 要查看对应分析图，请点击“‘奶酪 - 响应’列联分析”旁边的红色小三角并选择**对应分析**。

图 7.11 对应分析图的示例

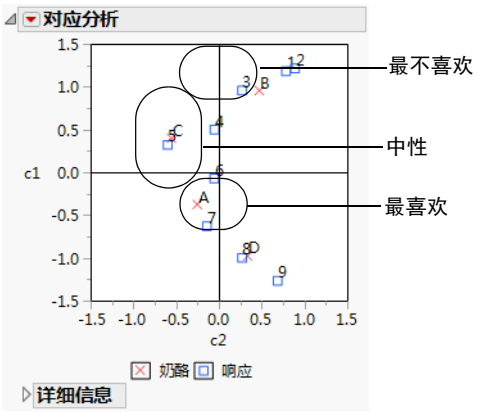
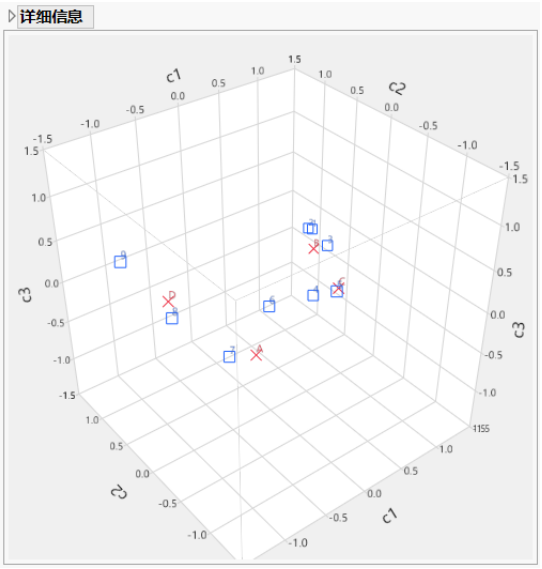


图 7.11 以图形方式显示对应分析结果，其中图轴标签为 c1 和 c2。请注意以下事项：

- c1 似乎对应于一般满意水平。c1 轴上的奶酪按顶部的最不喜欢到底部的最喜欢排列。
- 奶酪 D 是最喜欢的奶酪，具有响应 8 和 9。
- 奶酪 B 是最不喜欢的奶酪，具有响应 1、2 和 3。
- 奶酪 C 和 A 位于中间，具有响应 4、5、6 和 7。

8. 点击“对应分析”旁边的红色小三角并选择**三维对应分析**。

图 7.12 三维散点图的示例



请注意以下事项：

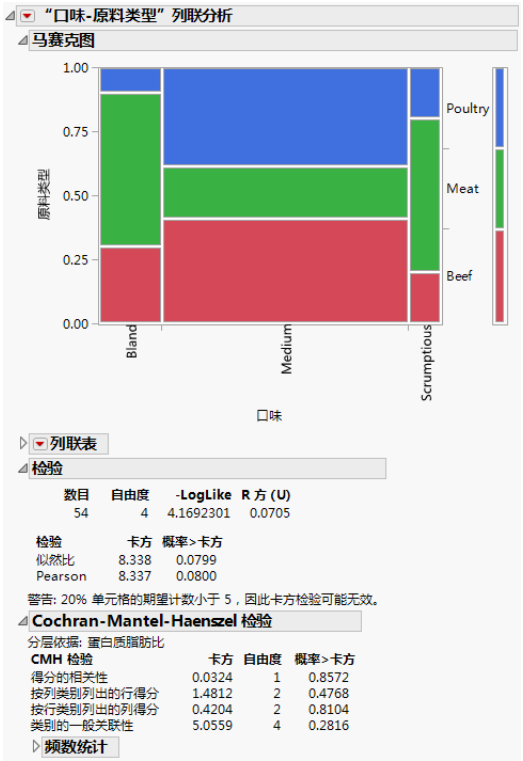
- 在 c1 轴上，响应 1 到 5 显示在 0 的右侧（正数）。响应 6 到 9 显示在 0 的左侧（负数）。
- 在 c2 轴上，A 和 C 显示在 0 的右侧（正数）。B 和 D 显示在 0 的左侧（负数）。
- 您可以得出结论：c1 对应于一般满意度（从最不喜欢到最喜欢）。

“Cochran Mantel Haenszel 检验”的示例

本例检查热狗类型和口味之间的关系。

1. 选择**帮助 > 样本数据库**，然后打开 Hot Dogs.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**原料类型**并点击 **Y，响应**。
4. 选择**口味**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“‘口味 - 原料类型’列联分析”旁边的红色小三角并选择 Cochran Mantel Haenszel。
7. 选择**蛋白质 / 脂肪**作为分组变量并点击**确定**。

图 7.13 “Cochran-Mantel-Haenszel 检验”的示例



请注意以下事项：

- “检验”报表显示接近显著的卡方概率约为 0.0799，指示热狗口味和类型之间的关系存在一定的显著性。
- Cochran Mantel Haenszel 报表显示常规类别关联的 p 值为 0.2816，远远大于 5%。

“一致性统计量”选项的示例

本例检查两个评级员之间的关系。数据给出三个人（评级员）对 50 个部件的三次评级结果。检查评级员 A 和 B 之间的关系。

1. 选择**帮助 > 样本数据库**，然后打开 Attribute Gauge.jmp。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择 A 并点击 **Y，响应**。
4. 选择 B 并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘B-A’列联分析”旁边的红色小三角并选择**一致性统计量**。

图 7.14 “一致性统计量”报表的示例

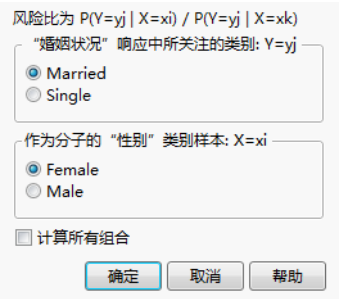
您注意到一致性统计量 0.86 很高（接近 1），且 *p* 值 <.0001，很小。这证实了查看列联表对角线时所看到的高一致性。两个评级员都给出评级 0 或都给出评级 1 时，会出现评级员之间的一致性。

“相对风险”选项的示例

本例检查参与调查的已婚和单身人士的相对概率。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**婚姻状况**并点击 **Y，响应**。
4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘性别 - 婚姻状况’列联分析”旁边的红色小三角并选择**相对风险**。
   将显示 “选择相对风险类别”窗口。

图 7.15 “选择相对风险类别”窗口



请注意有关 “选择相对风险类别”窗口的以下事项：

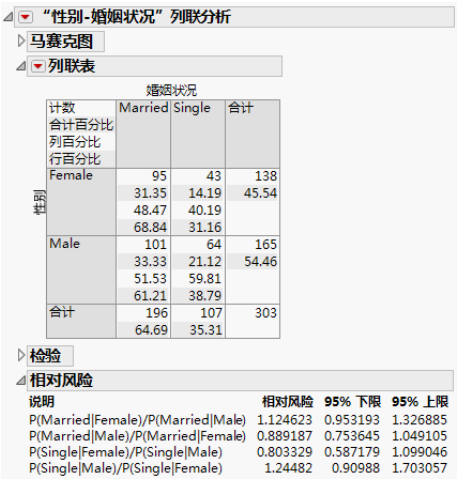
- 若要仅关注单个响应和因子组合，可以在此处选择。例如，若点击图 7.15 所示窗口中的**确定**，则会进行以下计算：

$$\frac{P(Y = \text{Married} \mid X = \text{Female})}{P(Y = \text{Married} \mid X = \text{Male})}$$

- 若要计算响应和因子水平的所有 $(2 \times 2 = 4)$ 组合的风险比，可以选中**计算所有组合**复选框（图 7.16）。

7. 通过选中**计算所有组合**复选框请求所有组合。将所有其他默认选择原样保留。

图 7.16 “风险比”报表的示例



相对风险计算如下：

1. 检查“相对风险”报表中的第一个条目，即 P(Married|Female)/P(Married|Male)。
2. 您可以在列联表中找到这些概率。由于基于性别的两个水平计算概率，它在表的各行上不同，因此使用行百分比来读取概率：

$$P(\text{Married} \mid \text{Female}) = 0.6884$$

$$P(\text{Married} \mid \text{Male}) = 0.6121$$

因此，相对风险计算如下：

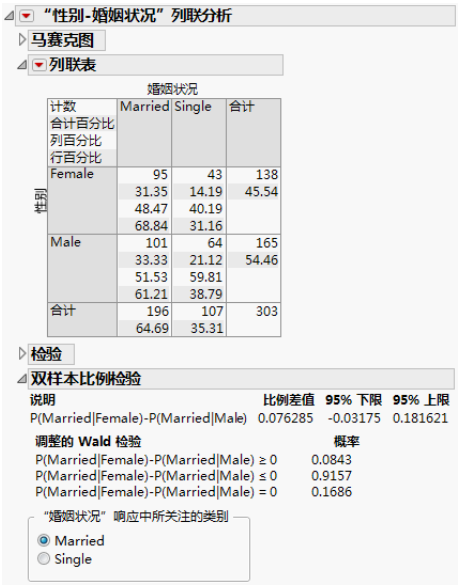
$$P(\text{Married} \mid \text{Female}) / P(\text{Married} \mid \text{Male}) = \frac{0.6884}{0.6121} = 1.1247$$

“双样本比例检验”的示例

本例检查男性和女性已婚的概率。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**婚姻状况**并点击 **Y，响应**。
4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘性别 - 婚姻状况’列联分析”旁边的红色小三角并选择**双样本比例检验**。

图 7.17 “双样本比例检验”报表的示例



在该示例中，您比较男性和女性已婚的概率。查看列联表中的 “行百分比”可获得以下信息：

$$P(\text{Married} \mid \text{Female}) = 0.6884$$

$$P(\text{Married} \mid \text{Male}) = 0.6121$$

这两个数字之间的差值 0.0763 为报表中所示的“比例差值”。双侧置信区间为 [-0.03175, 0.181621]。与置信区间对应的调整 Wald 方法的 *p* 值为 0.1686，该值接近 Pearson 卡方检验的 *p* 值 (0.1665)。通常，Pearson 卡方检验比修改的 Wald 检验更常用于检验两个比例的差值。

“关联测量”选项的示例

本例检查结婚与性别的关系。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**婚姻状况**并点击 **Y，响应**。
4. 选择**性别**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘性别 - 婚姻状况’列联分析”旁边的红色小三角并选择**关联测量**。

图 7.18 “关联测量”报表的示例

“性别-婚姻状况”列联分析

马赛克图

列联表

检验

关联测量

测量	值	标准误差	95% 下限	95% 上限
Gamma	0.1667	0.1184	-0.0654	0.3987
Kendall Tau-b	0.0795	0.0570	-0.0321	0.1911
Stuart Tau-c	0.0757	0.0543	-0.0307	0.1821
Somer D C|R	0.0763	0.0547	-0.0309	0.1835
Somer D R|C	0.0828	0.0593	-0.0335	0.1991
Lambda 不对称 C|R	0.0000	0.0000	0.0000	0.0000
Lambda 不对称 R|C	0.0000	0.0000	0.0000	0.0000
Lambda 对称	0.0000	0.0000	0.0000	0.0000
不确定性系数 C|R	0.0049	0.0070	0.0000	0.0186
不确定性系数 R|C	0.0046	0.0066	0.0000	0.0176
对称式不确定性系数	0.0047	0.0068	0.0000	0.0181

由于您要检查的变量（性别和婚姻状况）是名义型变量，因此使用 Lambda 和“不确定性”测量。所有值都很小，因此看起来在性别和婚姻状况之间存在弱关联。

“Cochran Armitage 趋势检验”的示例

本例调查了购买不同尺寸汽车的男女比例之间是否存在关系。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。

针对该检验的目的，将汽车尺寸更改为有序型变量：

2. 在列面板中，右击汽车尺寸旁边的图标并选择**有序型**。
3. 选择**分析** > **以 X 拟合 Y**。
4. 选择**性别**并点击 **Y，响应**。
5. 选择**汽车尺寸**并点击 **X，因子**。

6. 点击**确定**。
7. 点击“‘汽车尺寸-性别’列联分析”旁边的红色小三角并选择 **Cochran Armitage 趋势检验**。

图 7.19 “Cochran Armitage 趋势检验”报表的示例

双侧 p 值 (0.7094) 很大。因此您无法得出以下结论：在购买不同尺寸汽车的女性和男性比例方面存在关系。

“列联”平台的统计详细信息

- “‘一致性统计量”选项”
- “‘优势比”选项”
- “检验报表”
- “对应分析中的“详细信息”报表”

“一致性统计量”选项

查看作为 n 个对象的两个独立评级的两个响应变量时，当评级员具有完全一致性时，Kappa 系数等于 +1。当观测到的一致性超过机会一致性时，Kappa 系数为正，其量值反映了一致性的强度。尽管不常见，当观测到的一致性小于机会一致性时，Kappa 为负数。Kappa 的最小值介于 -1 到 0 之间，取决于边缘比例。

Kappa 系数计算如下：

$$\hat{\kappa} = \frac{P_0 - P_c}{1 - P_c} \text{ 其中 } P_0 = \sum_i p_{ii}, \; P_c = \sum_i p_{i+}p_{+i}$$

请注意，p_{ij} 是第 (i, j) 个单元格中的对象的比例，所以 $\sum_i \sum_j p_{ij} = 1$。

按以下方式估计简单 kappa 系数的渐近方差：

$$\text{方差} = \frac{A + B - C}{(1 - P_c)^2 n} \text{ 其中 } A = \sum_i p_{ii}[1 - (p_{i+} + p_{+i})(1 - \hat{\kappa})]^2, \; B = (1 - \hat{\kappa})^2 \sum_{i \neq j} \sum p_{ij}(p_{+i} + p_{j+})^2,$$

$$C = [\hat{\kappa} - P_c(1 - \hat{\kappa})]^2$$

请参见 Cohen (1960) 和 Fleiss et al.(1969)。

对于 Bowker 对称性检验，原假设为 2x2 表中的概率满足对称性 ($p_{ij}=p_{ji}$)。

“优势比”选项

优势比的计算方式如下：

$$\frac{p_{11} \times p_{22}}{p_{12} \times p_{21}}$$

其中 p_{ij} 是 2x2 表的第 i 行第 j 列中的计数。

检验报表

R 方 (U)

R 方 (U) 的计算方式如下：

$$\frac{-\text{对数似然(模型)}}{-\text{对数似然(校正总和)}}$$

通过拟合总样本的固定响应率来获得负对数似然合计。

检验

两个卡方检验计算如下：

“似然比卡方”检验统计量的计算方式为：“检验”表中模型的负对数似然的两倍。某些书使用符号 G^2 表示该统计量。两个负对数似然的差值（一个具有**整个总体**的响应概率，一个具有**每个总体**的响应率）的公式定义如下：

$$G^2 = 2\left[\sum_{ij}(-n_{ij})\ln(p_j) - \sum_{ij} -n_{ij}\ln(p_{ij})\right] \text{ 其中 } p_{ij} = \frac{n_{ij}}{N},\ p_j = \frac{N_j}{N}$$

该公式可以简写为：

$$G^2 = 2\sum_i\sum_j n_{ij}\ln\left(\frac{n_{ij}}{e_{ij}}\right)$$

通过对观测到的单元格计数和单元格的期望计数的差值求平方和来计算 Pearson 卡方检验统计量。Pearson 卡方检验揭示在样本很大时频数统计趋向于正态分布这个特性。该卡方统计量的熟悉形式定义如下：

$$\chi^2 = \sum\frac{(O-E)^2}{E}$$

其中 O 是观测到的单元格计数，E 是期望的单元格计数。对所有单元格求和。此处没有进行连续校正，有时在 2x2 表中会进行这样的校正。

对应分析中的“详细信息”报表

列出来自以下奇异值分解 (SVD) 的值：

$$\mathbf{D}_r^{-0.5}(\mathbf{P} - rc')\mathbf{D}_c^{-0.5} = \mathbf{U}Diag(\Lambda)\mathbf{V}'$$

其中：

- **P** 是计数除以总频数所得的值构成的矩阵
- *r* 和 *c* 是 **P** 的行总和与列总和
- **D** 矩阵是 *r* 和 *c* 的值构成的对角矩阵
- Λ 是详细信息报表中报告的奇异值的列向量

有关奇异值分解的详细信息，请参见《多元方法》。

详细信息报表中的行坐标 (*rc*) 和列坐标 (*cc*) 计算如下：

$$rc = \mathbf{D}_r^{-0.5}\mathbf{U}Diag(\Lambda)$$

$$cc = \mathbf{D}_c^{-0.5}\mathbf{V}Diag(\Lambda)$$

第 8 章

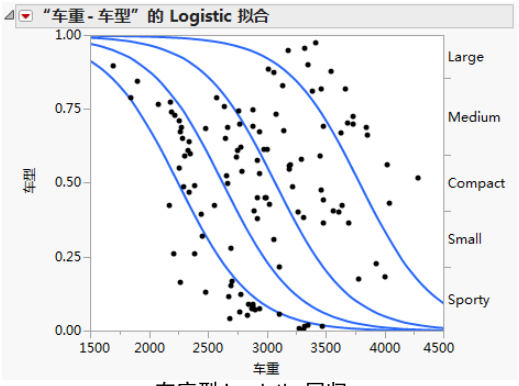
Logistic 分析

检查分类 Y 和连续 X 变量之间的关系

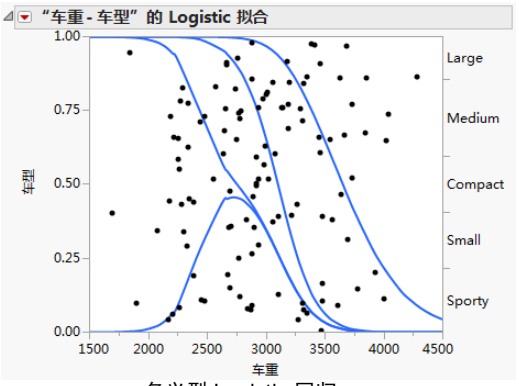
使用 Logistic 平台拟合具有连续 *x* 预测变量的分类响应的 Logistic 回归模型。选项包括 ROC 曲线、提升曲线和优势比估计。拟合的模型估计每个 *x* 值的概率。“Logistic”平台是“以 X 拟合 Y”平台的**名义型**或**有序型 - 连续型**特质。在该平台上名义型响应和有序型响应有区别：

- 名义型 Logistic 回归估计一组曲线来分割各响应的概率。
- 有序型 Logistic 回归为小于或等于给定响应的概率建模。它具有估计单个 Logistic 曲线的效应，曲线进行水平偏移以生成有序类别的概率。该模型更简单，推荐用于有序响应。

图 8.1 Logistic 回归示例



有序型 Logistic 回归



名义型 Logistic 回归

目录

Logistic 回归概述 247
名义型 Logistic 回归 247
有序型 Logistic 回归 247
名义型 Logistic 回归的示例 247
启动“Logistic”平台 249
数据结构 249
Logistic 报表 250
Logistic 图 251
迭代 251
整体模型检验 251
拟合详细信息 252
参数估计值 253
“Logistic”平台选项 253
ROC 曲线 254
保存概率公式 255
逆预测 255
Logistic 回归的其他示例 255
有序型 Logistic 回归的示例 256
Logistic 图的更多示例 257
ROC 曲线的示例 259
使用十字准线工具进行逆预测的示例 260
使用“逆预测”选项的逆预测示例 261
“Logistic”平台的统计详细信息 263

Logistic 回归概述

Logistic 回归历史悠久，具有广泛的应用，如为剂量响应数据和购买选择数据建模。遗憾的是，很多导论性的统计学课程没有涉及这一相当简单的方法。除了有关 Logistic 回归的教科书 (Hosmer and Lemeshow 1989) 外，分类统计学中的很多教科书都涉及到它 (Agresti 1990)。一些分析员将该方法与一个分布不同的函数（正态分布）一起使用。在这种情况下，它被称为 Probit 分析。一些分析员使用判别分析而非 Logistic 回归，因为他们倾向于将连续变量视为 *Y* 而将类别视为 *X*，并进行逆向操作。但是，判别分析假定连续数据是正态分布的随机响应而非固定回归变量。

简单 Logistic 回归是“拟合模型”平台中分类响应的简化常规工具，该工具的图形功能更强。有关复杂 Logistic 回归模型的示例，请参见《拟合线性模型》。

名义型 Logistic 回归

名义型 Logistic 回归估计选择某一响应水平的概率，作为 *x* 因子的一个平滑函数。拟合的概率必须介于 0 到 1 之间，并且对于给定的因子值各响应水平的拟合概率之和为 1。

在 Logistic 概率图中，垂直轴表示概率。对于 *k* 个响应水平，用 *k* - 1 条平滑曲线来分割各响应水平的总概率（等于 1）。Logistic 回归的拟合原则是对发生的响应事件拟合的概率的负自然对数之和最小化（即最大似然）。

有序型 Logistic 回归

Y 为有序型时，将修改后的 Logistic 回归用于拟合。通过曲线对等于或低于每个响应水平的累积概率建模。这些曲线对于每个水平是相同的，只是向右或向左平移了一段距离。

有序型 Logistic 模型对于 *r* - 1 个累积 Logistic 比较中的每个比较拟合不同的截距，但是斜率相同，其中 *r* 是响应水平数。可以逐个检查和检验每个参数估计值，尽管这样做意义不大。

有时有序型模型比名义型模型好，因为它需要估计的参数更少。现实中经常需要拟合具有数百个响应水平的有序型响应。

名义型 Logistic 回归的示例

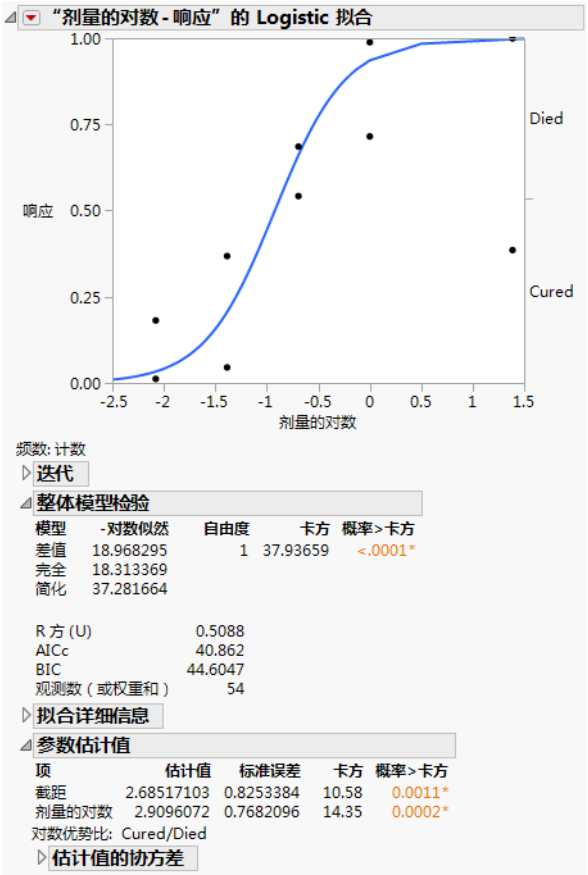
了解如何检查名义型 Y 响应和连续型 X 因子之间的关系。该示例中的数据来自一个涉及 5 个组的实验，其中每个组包含 12 只兔子，这些兔子被注射了链球菌。兔子被确认身体感染这一细菌后，给它们注射不同剂量的青霉素。您要了解剂量的自然对数 （**剂量**）是否影响兔子的治愈。

1. 选择**帮助** > **样本数据库**，然后打开 Penicillin.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**响应**并点击 **Y，响应**。
4. 选择**剂量**并点击 **X，因子**。

请注意， JMP 会为**频数**自动填充**计数**。之前已为**计数**指定了 “频数”角色。

5. 从 “目标水平”列表中，选择 “Cured”。
6. 点击**确定**。

图 8.2 名义型 Logistic 报表的示例



该图显示拟合模型是**剂量的对数**的函数。该拟合模型预测了治愈概率。*p* 值是显著的，指示剂量对兔子是否治愈具有显著影响。

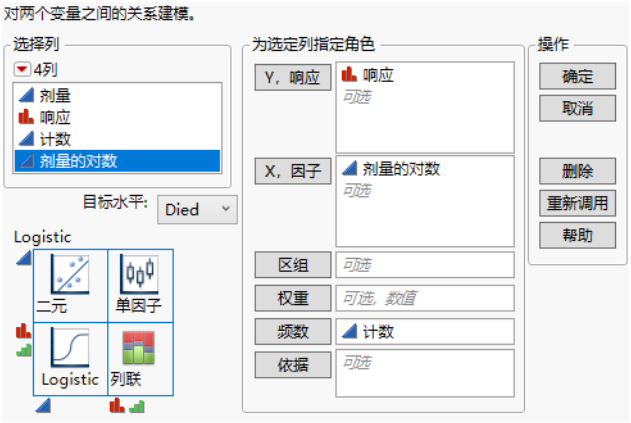
提示：要更改分析的响应水平，请在启动窗口中指定“目标水平”或使用“值顺序”列属性。

启动“Logistic”平台

要执行 Logistic 分析，请执行以下操作：

1. 选择**分析 > 以 X 拟合 Y**。
2. 为 **Y，响应**输入名义型或有序型列。
3. 为 **X，因子**输入连续列。

图 8.3 “以 X 拟合 Y”启动窗口



“Logistic”一词出现在图像的上方，指示您正在执行 Logistic 分析。

注意：您还可以从“JMP 起始页”窗口启动 Logistic 分析。选择**视图 > JMP 起始页 > 基本 > Logistic**。

若响应为二值响应且具有名义建模类型，则“目标水平”菜单显示在启动窗口中。使用该菜单可指定您要建模其概率的响应的水平。

提示：默认值是基于水平顺序的两个水平中的较高者。检查默认值是否为要建模的水平。

有关“以 X 拟合 Y”启动窗口的详细信息，请参见第 97 页的““以 X 拟合 Y”介绍”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

数据结构

您的数据可以包含未汇总或汇总的数据：

未汇总数据 每个观测都对应一行，其中包含其 X 值和其 Y 值。

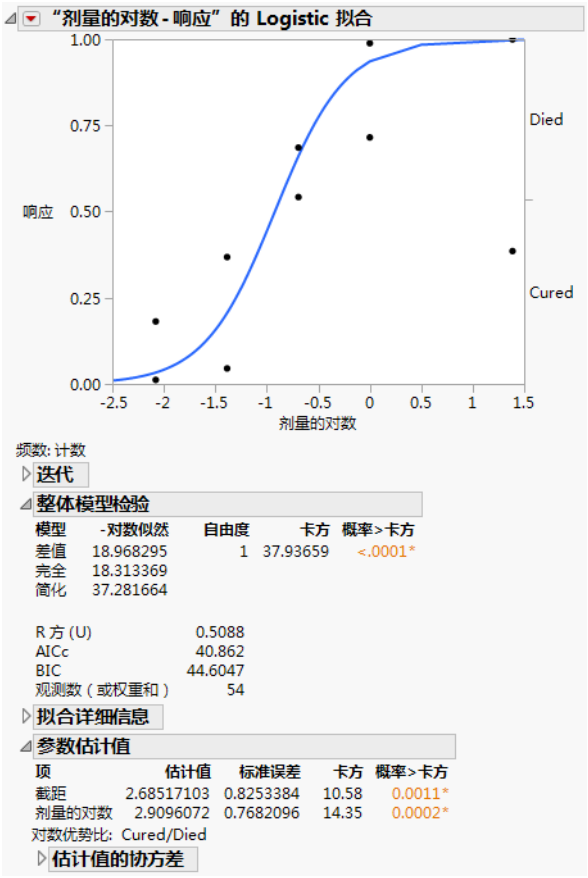
汇总数据 每行都表示具有公共 X 和 Y 值的一组观测。数据表必须包含提供每行计数的一列。在启动窗口中将该列作为频数输入。

Logistic 报表

Logistic 报表窗口包含 Logistic 图、“迭代”报表、“整体模型检验”报表、“拟合详细信息”报表和“参数估计值”报表。

要生成图 8.4 中所示的图，请按第 247 页的“名义型 Logistic 回归的示例”中的说明操作。

图 8.4 Logistic 报表示例



注意：红色小三角菜单提供可以添加到初始报表窗口的更多选项。请参见第 253 页的“‘Logistic’平台选项”。

Logistic 图

Logistic 概率图演示 Logistic 拟合。在水平轴上的每个值处，将垂直方向上的概率尺度划分为每个响应类别的概率。用两个曲线之间的垂直距离表示这些概率大小，且所有响应类别的概率之和为 1。

Logistic 图中的点表示数据表中的观测。每个点的水平位置由其连续因子的值决定。每个点的垂直位置会随机选择在对应于其响应类别值的曲线之间。这些点的随机散布更易于看清点最密集的位置，但是垂直位置与垂直轴上的值不对应。由于使用了固定的随机种子，所以在同一模型的多个拟合中，垂直位置没有差异。

通过点击相关数据表的列面板中的某个变量并将其拖到轴上，可以替换图中的变量。

注意：另见第 257 页的“Logistic 图的更多示例”。

迭代

“迭代”报表显示每次迭代结果和确定 Logistic 模型是否收敛的评估准则。仅对名义型 Logistic 回归显示迭代。

整体模型检验

“整体模型检验”报表显示模型拟合效果是否好于常数响应概率。该报表类似于连续响应模型的“方差分析”报表。它是特定的似然比卡方检验，用于估计分类模型拟合数据的好坏程度。

观测到的概率的自然对数负数和称为负对数似然（**–对数似然**）。分类数据的负对数似然与连续数据中的平方和所起的作用相同：从数据所拟合模型与具有相等概率的模型得到的负对数似然差值加倍后便为卡方统计量。该检验统计量检查 x 变量对响应没有影响的假设是否成立。

R 方 (U)（有时表示为 R^2）的值介于 0 到 1 之间。R^2 值较大表示模型拟合效果好，这在分类模型中很少见。

“整体模型检验”报表包含以下列：

模型 有时称为“源”。

- **简化**模型仅包含截距。
- **完全**模型包含所有效应以及截距。
- **差值**为完全模型和简化模型的对数似然差值。

自由度 记录与模型有关的自由度。

–对数似然 测量样本中的变异，有时称为**不确定性**。

完全（完全模型）是拟合模型后计算出的负对数似然（或不确定性）。拟合过程涉及使用线性模型和 Logistic 响应函数预测响应率。通过拟合过程将该值最小化。

简化（简化模型）是用固定背景比率估计概率情况下的负对数似然（或不确定性）。这是模型没有效果时的背景不确定性。

这两个负对数似然的差值是由于拟合模型导致的减少。该值的两倍即似然比卡方检验统计量。

请参见 《拟合线性模型》。

卡方 似然比卡方检验检查在整个样本上模型的拟合效果不比固定响应率好的假设是否成立。它是差值模型的 – **对数似然**的两倍，即两个负对数似然差值的两倍，一个具有整个总体的响应概率，一个具有每个总体的响应率。请参见第 263 页的 “‘Logistic’平台的统计详细信息”。

概率 > 卡方 该观测到的显著性概率，通常称为卡方检验的 *p* 值。它是得到卡方值大于计算值的概率。若该概率小于 0.05，通常判断模型是显著的。

R 方 (U) 归因于模型拟合的总不确定性的比例，定义为**差值**负对数似然值除以**简化**负对数似然值所得的值。R 方 (U) 值为 1 指示预测的事件发生的概率等于 1：预测概率不存在不确定性。因为预测概率的确定性对于 Logistic 模型很少见，R 方 (U) 通常很小。请参见第 263 页的 “‘Logistic’平台的统计详细信息”。

注意：R 方 (U) 亦称 McFadden 伪 R 方。

AICc 校正的 Akaike 信息准则。请参见 《拟合线性模型》。

BIC Bayes 信息准则。请参见 《拟合线性模型》。

观测数 有时称为 “权重和”。计算中使用的总样本大小。若指定了**权重**变量，则为权重和。

拟合详细信息

Logistic 平台中的 “拟合详细信息”报表包含以下统计量：

测量 包含以下拟合测度：

熵 R 方 比较拟合模型与常数概率模型的对数似然。这与 R 方 (U) 相同。请参见第 263 页的 “‘Logistic’平台的统计详细信息”。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 *L*，并且统一尺度后最大值为 1。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。

-Log p 均值 -log(*p*) 的平均值，其中 *p* 是与发生的事件关联的拟合概率。

RASE 均方根误差，其中差值为响应和 *p*（实际发生事件的拟合概率）之间的差值。

绝对偏差的均值 响应和 p（实际发生事件的拟合概率）的差值绝对值的平均值。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。

数目 观测数。

对于熵 R 方和广义 R 方，值越接近 1 表示拟合效果越好。对于 -Log p 均值、 RASE、绝对偏差的均值和误分类率，值越小表示拟合效果越好。

训练 拟合测量的值。

定义 拟合测量的代数定义。

参数估计值

名义型 Logistic 模型为每个（有 $k-1$ 个）Logistic 比较的截距和斜率拟合参数，其中 k 为响应水平数。“参数估计值”报表列出这些估计值。可以逐个检查和检验每个参数估计值，尽管这样做意义不大。

项 列出 Logistic 模型中的每个参数。除了最后一个水平外，在响应变量每个水平上都有因子的截距和斜率项。

估计值 列出 Logistic 模型给出的参数估计值。

标准误差 列出每个参数估计值的标准误差。它们用于计算将每项与零比较的统计检验。

卡方 列出每个参数均为零假设下的 Wald 检验。Wald 卡方的计算公式为：(估计值/标准误差)^2。

概率 > 卡方 列出卡方检验的观测到的显著性概率。

估计值的协方差

报告参数估计值的方差估计值以及参数估计值之间的协方差估计值。方差估计值的平方根与**标准误差**部分中给出的值相同。

“Logistic”平台选项

“Logistic 拟合”红色小三角菜单包含影响 Logistic 图的选项。若指定了多个 Y 响应变量，则改为显示 “拟合组”红色小三角菜单。请参见 《拟合线性模型》。

优势比 向 “参数估计值”报表添加优势比。请参见 《拟合线性模型》。

该选项仅适用于具有两个水平的响应。

逆预测 从给定的 y 值预测 x 值。请参见第 255 页的 “逆预测”。

该选项仅适用于二水平名义型响应。

Logistic 图 显示或隐藏 Logistic 图。

图选项 包含以下选项：

显示点 在 Logistic 图中显示或隐藏点。

显示比率曲线 仅当对每个 x 值都具有若干点时有用。在这种情况下，您获取每个值处的比率的合理估计值，然后将该比率与拟合的 Logistic 曲线进行比较。为了防止出现过多的退化点（通常在 0 或 1 处），JMP 仅在 x 值处至少有三个点时才显示该比率值。

线条颜色 支持您挑选图曲线的颜色。

ROC 曲线 生成受试者操作特征 (ROC) 曲线。 ROC 曲线是灵敏度 vs. 每个 *x* 值的 （1 – 特异度）的图。请参见第 254 页的 “ROC 曲线”。

提升曲线 生成模型的提升曲线。提升曲线显示与 ROC 曲线相同的信息，但在某种程度上充分反映了开始处排序的丰富度。垂直轴显示总体比例在所选响应水平中的丰富程度与该响应水平作为整体的比率之比。有关提升曲线的详细信息，请参见 《拟合线性模型》。

保存概率公式 创建包含公式的新数据表列。请参见第 255 页的 “保存概率公式”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

ROC 曲线

当您执行具有二值结果的简单 Logistic 回归时，系统提供用于该分析的 ROC 曲线的平台选项。选择 ROC 曲线选项后，您必须指定要用作**正**响应的水平。

注意：另见第 259 页的 “ROC 曲线的示例”。

假定您有作为诊断测量的 *x* 值，并要确定指示以下内容的 *x* 的阈值：

- 若 *x* 值大于阈值，则条件存在。
- 若 *x* 值小于阈值，则条件不存在。

例如，您可以测量血液成分水平作为诊断检验以预测某种癌症。现在考虑当更改阈值时诊断检验的情况，它会导致更多或更少假阳性和假阴性。然后您标绘这些比率。理想情况是在很窄范围的 *x* 准则值能最好地区分真阴性和真阳性。受试者操作特征 (ROC) 曲线显示这一转变以多快速度发生。ROC 曲线的目标是得到最大化曲线下的面积的诊断。

医学中使用了两个标准定义：

- **灵敏度**：给定的 *x* 值（检验或测量）正确预测现有条件的概率。对于给定的 *x* 值，错误预测某一现有条件的概率为 1 – 灵敏度。
- **特异度**：检验正确预测条件不存在的概率。

ROC 曲线是显示每个 *x* 值的（1 – 特异度）对比灵敏度图。ROC 曲线下的面积是用于汇总曲线所含信息的常见指数。

若检验预测效果好，将有一个值，整个异常人群所在的值范围将落在该值之上且所有正常值范围将落在该值之下。它将非常灵敏并穿过网格上的点 (0,1)。ROC 曲线离这个理想点越近，它的判别能力越强。没有预测能力的检验生成一条沿网格对角线的曲线（DeLong 和其他人 1988）。

ROC 曲线是假阳性率和真阳性率之间关系的图形表示法。估计该关系的标准方法是使用曲线下的面积，它显示在报表中的图下方。在图中，黄线是根据 ROC 曲线的正切 45 度角绘制的。它标记了假定假阴性和假阳性具有相似成本时的一个理想截止点。

保存概率公式

保存概率公式选项用于创建新的数据表列。这些数据表列保存以下信息：

- *x* 因子的线性组合（通常称为 logit）的公式
- 响应水平概率的预测公式
- 给出最可能的响应的预测公式

逆预测

逆预测与预测正好相反。它从给定的 *y* 值预测 *x* 值。但是在 Logistic 回归中，不使用 *y* 值，您使用归因于某个 *Y* 水平的概率。该功能仅适用于二水平名义型响应。

“拟合模型”平台还提供一个选项用于执行具有置信限的逆预测。有关“逆预测”的详细信息，请参见《拟合线性模型》。

注意：另见第 260 页的“使用十字准线工具进行逆预测的示例”和第 261 页的“使用“逆预测”选项的逆预测示例”。

Logistic 回归的其他示例

本节包含使用 Logistic 平台中的 Logistic 回归的更多示例。

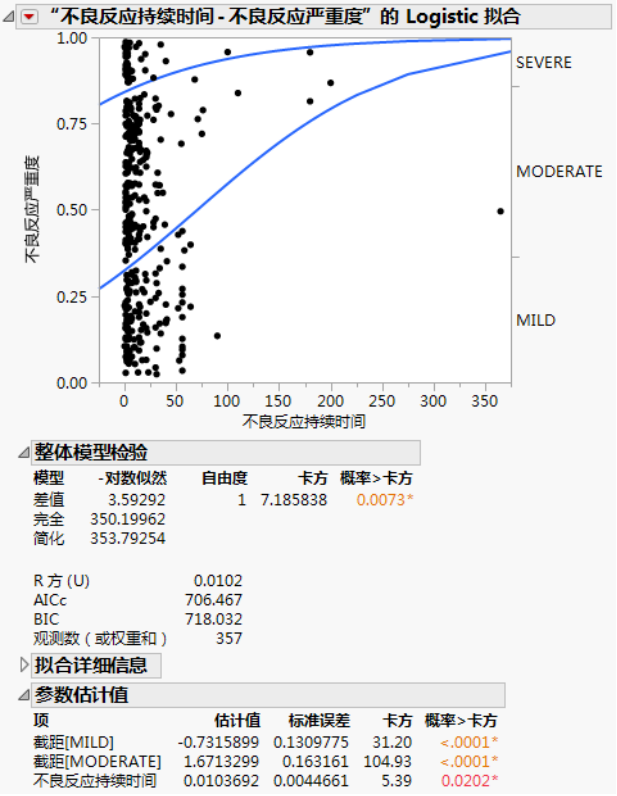
- “有序型 Logistic 回归的示例”
- “Logistic 图的更多示例”
- “ROC 曲线的示例”
- “使用十字准线工具进行逆预测的示例”
- “使用“逆预测”选项的逆预测示例”

有序型 Logistic 回归的示例

本例显示如何检查有序型 Y 响应和连续型 X 因子之间的关系。在本例中，假定您要对不良反应严重度建模，将它表示为治疗持续时间值的函数。

1. 选择**帮助 > 样本数据库**，然后打开 AdverseR.jmp。
2. 右击**不良反应严重度**左侧的图标，将建模类型更改为有序型。
3. 选择**分析 > 以 X 拟合 Y**。
4. 选择**不良反应严重度**并点击 **Y，响应**。
5. 选择**不良反应持续时间**并点击 **X，因子**。
6. 点击**确定**。

图 8.5 有序型 Logistic 报表的示例



您可以使用与名义型报表相同的方式解释该报表。请参见第 250 页的“Logistic 报表”。

在图中，根据数据的 *x* 坐标绘制数据标记。当若干数据点在同一 *y* 位置出现时，则将点随机散布。即在数据点之间显示一些小间隙，以便您可以更清楚地看到每个点。

在有很多点的地方，曲线被分隔开。在点很少甚至没有点的地方，曲线靠得很近。数据以此方式推动曲线，因为最大化准则是模型拟合的概率乘积。该拟合尝试避免被认为是小概率的点，这些点被拟合曲线紧靠。有关计算细节的详细信息，请参见《拟合线性模型》。

有关“整体模型检验”报表和“参数估计值”报表的详细信息，请参见第 250 页的“Logistic 报表”。在“参数估计值”报表中，会为除最后一个响应水平外的每个响应水平估计截距参数，但是只有一个斜率参数。截距参数显示响应水平的间距。它们始终单调递增。

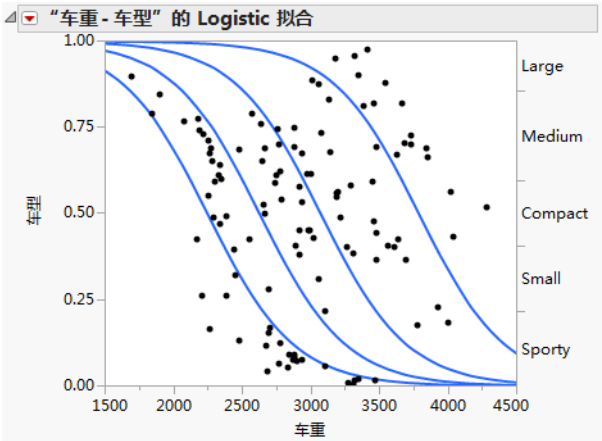
Logistic 图的更多示例

在本例中，假定您要使用 116 辆汽车的车重来预测汽车尺寸（**车型**）。汽车尺寸可以为以下值之一（按最小到最大的顺序排列）：赛车、小型车、紧凑型车、中型车或大型车。

1. 选择**帮助** > **样本数据库**，然后打开 Car Physical Data.jmp。
2. 在**列**面板中，右击**车型**左侧的图标并选择**有序型**。
3. 右击**车型**并选择**列信息**。
4. 从“列属性”菜单中选择**值顺序**。
5. 验证数据是按以下从上到下的顺序排列：赛车、小型车、紧凑型车、中型车、大型车。
6. 点击**确定**。
7. 选择**分析** > **以 X 拟合 Y**。
8. 选择**车型**并点击 **Y，响应**。
9. 选择 Weight 并点击 **X，因子**。
10. 点击**确定**。

    随即显示报表窗口。

图 8.6 “车重 - 车型” Logistic 图的示例



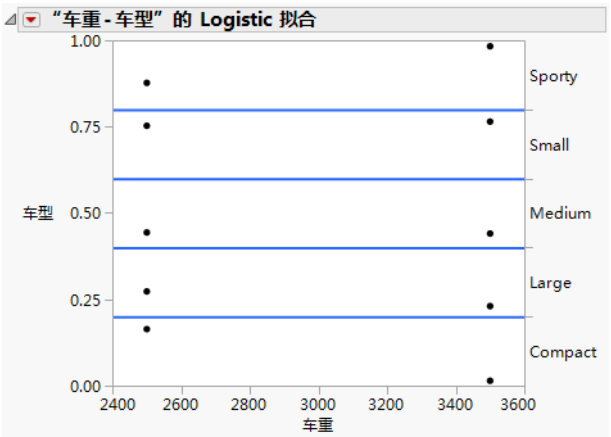
请注意以下事项：

- 第一条（底部）曲线表示给定车重下汽车是赛车的概率。
- 第二条曲线表示汽车是小型车或赛车的概率。仅查看第一条和第二条曲线之间的距离，它对应汽车是小型车的概率。
- 正如您期望的那样，越重的车越有可能是大型车。
- 根据数据的 *x* 坐标绘制数据标记。*y* 位置在对应于该行的响应类别范围内随机散布。

若 *x* 变量对响应没有影响，则拟合的线是水平的且连续因子范围内每个响应的概率为常数。图 8.7 显示一个 Logistic 图，在该图中**车重**对于预测**车型**没有帮助。

图 8.7 显示无 *x*-*y* 关系的样本数据表和 Logistic 图的示例

	车型	车重
1	Large	2500
2	Medium	2500
3	Compact	2500
4	Small	2500
5	Sporty	2500
6	Large	3500
7	Medium	3500
8	Compact	3500
9	Small	3500
10	Sporty	3500

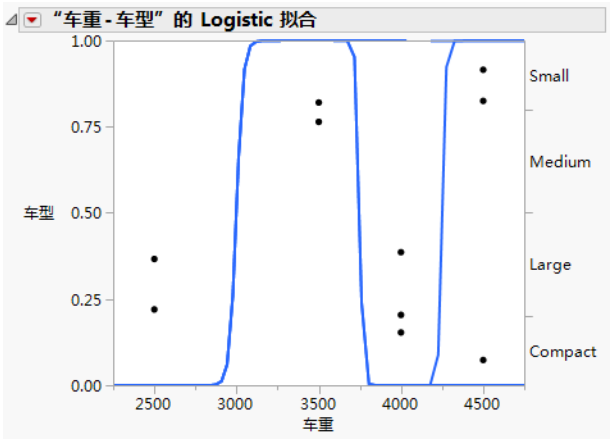


注意：要重新创建图 8.7 和图 8.8 中的图，您必须首先创建此处所示的数据表，然后执行本节开头的步骤 7-10。

若响应完全由因子值来预测，则 Logistic 曲线实际上是垂直的。响应的预测值在每个因子水平上接近确定（概率几乎为 1）。图 8.8 显示一个 Logistic 图，在该图中**车重**几乎可以完全预测**车型**。

图 8.8 显示几乎完全 *x-y* 关系的样本数据表和 Logistic 图的示例

	车型	车重
1	Small	2500
2	Small	2500
3	Compact	3500
4	Compact	3500
5	Medium	4000
6	Medium	4000
7	Medium	4000
8	Large	4500
9	Large	4500
10	Large	4500



在这种情况下，参数估计值变得很大并在回归报表中被标记为**不稳定**。在这些情况下，您可以考虑使用带 Firth 偏倚调整估计值的“广义线性模型”特质。请参见《拟合线性模型》。

ROC 曲线的示例

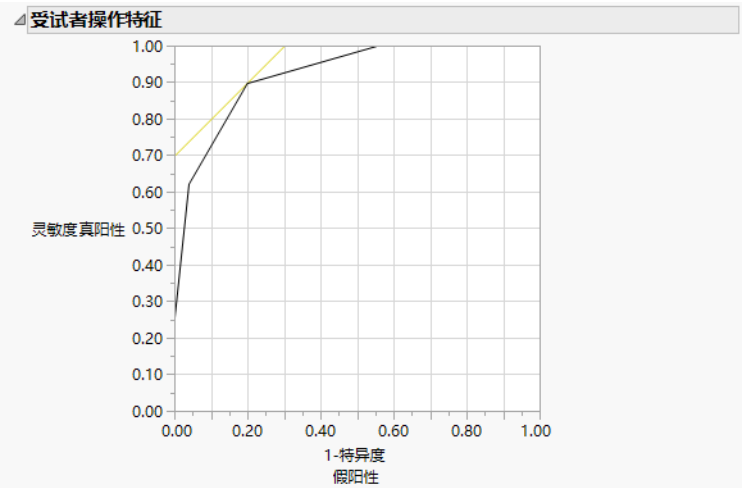
1. 选择**帮助** > **样本数据库**，然后打开 Penicillin.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**响应**并点击 **Y，响应**。
4. 选择**剂量**并点击 **X，因子**。

   请注意， JMP 会为**频数**自动填充**计数**。之前已为**计数**指定了 “频数”角色。
5. 从 “目标水平”列表中，选择 “Cured”。
6. 点击**确定**。
7. 点击 “‘剂量的对数 - 响应’的 Logistic 拟合”旁边的红色小三角并选择 **ROC 曲线**。

注意：该示例显示名义型响应的 ROC 曲线。有关有序型 ROC 曲线的详细信息，请参见《预测和专业建模》。

此处显示**剂量的对数 - 响应**示例的结果。ROC 曲线标绘上文所述的预测响应的概率。请注意在 ROC 表中，具有最高灵敏度 -（1 – 特异度）的行标有星号。

图 8.9 ROC 曲线和表的示例



ROC 表

X	概率	1-特异度	灵敏度	灵敏度-(1-特异度)		真阳性	真阴性	假阳性	假阴性
.	.	0.0000	0.0000	0.0000		0	25	0	29
1.38629	0.9988	0.0000	0.2414	0.2414		7	25	0	22
0.00000	0.9361	0.0400	0.6207	0.5807		18	24	1	11
-0.69315	0.6611	0.2000	0.8966	0.6966	*	26	20	5	3
-1.38629	0.2061	0.5600	1.0000	0.4400		29	11	14	0
-2.07944	0.0334	1.0000	1.0000	0.0000		29	0	25	0
-2.07944	0.0334	1.0000	1.0000	0.0000		29	0	25	0

由于 ROC 曲线完全在对角线上方，因此您得出结论：模型具有良好的预测能力。

使用十字准线工具进行逆预测的示例

本例显示如何在 JMP 中使用十字准线工具以直观方式近似展示逆预测。在注射青霉素的兔子实验中，您要了解导致治愈兔子的概率为 0.5 时对应的青霉素剂量是多少。在该示例中，0.5 的逆预测称为 ED50，即对应 50% 生存率的有效剂量。

确定哪个剂量值对应于治愈和死亡的机会相等。

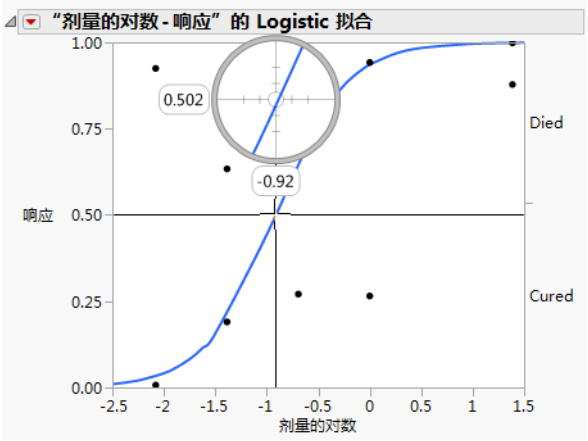
1. 选择**帮助** > **样本数据库**，然后打开 Penicillin.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**响应**并点击 **Y，响应**。
4. 选择**剂量**并点击 **X，因子**。

   请注意， JMP 会为**频数**自动填充**计数**。之前已为**计数**指定了 “频数”角色。
5. 从 “目标水平”列表中，选择 “Cured”。
6. 点击**确定**。

7. 点击十字准线工具。
8. 将水平十字准线放置到大约 0.5 处的垂直 （**响应**）概率轴上。
9. 将十字准线交叉点移到预测线，读取水平轴上显示的**剂量**值。

在本例中，注射**剂量的对数**值约为 -0.9 的兔子治愈和死亡的机会相等。

图 8.10 Logistic 图上十字准线工具的示例



使用“逆预测”选项的逆预测示例

若您的响应正好有两个水平，使用**逆预测**选项可以请求执行精确的逆预测。为您计算对应低响应类别的给定概率的 x 值以及该 x 值的置信区间。

1. 选择**帮助** > **样本数据库**，然后打开 Penicillin.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**响应**并点击 **Y，响应**。
4. 选择**剂量**并点击 **X，因子**。

   请注意， JMP 会为**频数**自动填充**计数**。之前已为**计数**指定了 “频数” 角色。
5. 从 “目标水平” 列表中，选择 “Cured”。
6. 点击**确定**。
7. 点击 “‘剂量的对数 - 响应’的 Logistic 拟合” 旁边的红色小三角并选择**逆预测**（图 8.11）。
8. 键入 0.95 作为**置信水平**的值。
9. 为置信区间选择**双侧**。
10. 请求关注的响应概率。对于该示例键入 0.5 和 0.9，这指示您正在请求计算对应治愈概率为 0.5 和 0.9 的**剂量的对数**值。
11. 点击**确定**。

将显示逆预测图。

图 8.11 “逆预测”窗口

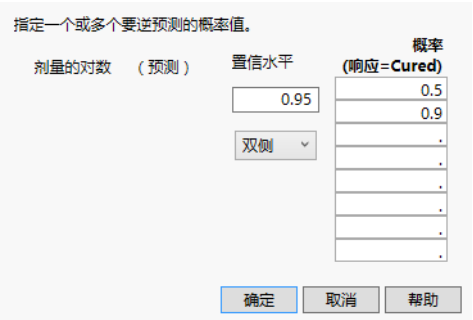
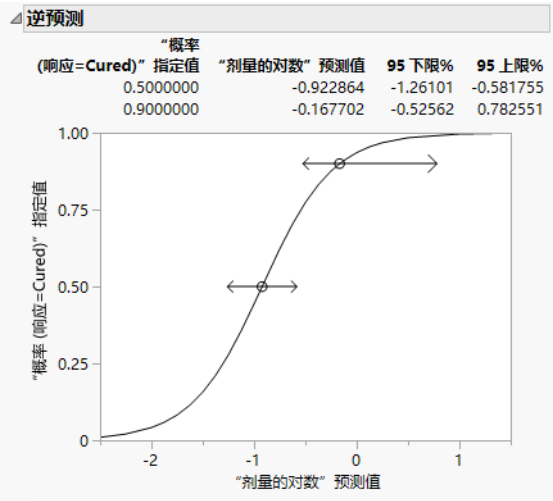


图 8.12 逆预测图的示例



x 值的估计值和置信区间显示在报表和概率图中。例如，导致治愈概率为 90% 的**剂量的对数**值估计介于 -0.526 到 0.783 之间。

“Logistic”平台的统计详细信息

本节包含关于“整体模型检验”报表的统计详细信息。

卡方

卡方统计量有时表示为 G^2，并且定义如下：

$$G^2 = 2(\sum -\ln p(\text{背景}) - \sum -\ln p(\text{模型}))$$

其中求和是对所有观测而非所有单元格执行的。

R 方 (U)

从 1 减去该检验统计量与背景对数似然之比来计算 R^2。更简单点说，R 方 (U) 的计算方式如下：

$$\frac{\text{差值的负对数似然}}{\text{简化模型的负对数似然}}$$

使用“整体模型检验”报表中的量。

注意：R 方 (U) 亦称 McFadden 伪 R 方。

第 9 章

制表

以交互方式创建汇总表

使用“制表”平台可采用交互方式构造描述性统计量的汇总表或数据透视表。“制表”平台提供一种以表格形式显示汇总数据的轻松灵活的方式。表是通过在制表中将数据表列指定为行或列，然后指定所需的汇总统计量来生成的。

图 9.1 制表示例

	性别											
	F						M					
	年龄						年龄					
	12	13	14	15	16	17	12	13	14	15	16	17
	体重	体重	体重	体重	体重	体重	体重	体重	体重	体重	体重	体重
最小值	64	67	81	92	112	116	79	79	92	104	128	134
均值	100.2	95.3	96.6	102.0	113.5	116.0	97.0	94.3	103.9	110.8	128.0	153.0
最大值	145	112	142	112	115	116	128	105	119	128	128	172

		性别							
		Female				Male			
		婚姻状况				婚姻状况			
		Married		Single		Married		Single	
		均值	标准差	均值	标准差	均值	标准差	均值	标准差
国家	汽车尺寸	年龄	年龄	年龄	年龄	年龄	年龄	年龄	年龄
American	Large	33.6	8.107	41.0	.	34.7	3.931	32.0	6.265
	Medium	31.4	5.827	29.0	9.258	31.3	5.413	32.1	11.05
	Small	31.0	5.657	29.0	9.539	31.8	4.813	26.5	6.455
European	Large	34.0	7.071	28.0	.	.	.	26.0	.
	Medium	31.0	5.06	28.7	5.508	32.3	5.62	31.0	10.13
	Small	29.8	6.611	28.0	1.414	33.8	4.381	25.7	2.517
Japanese	Large	25.0	.	.	.	32.0	.	.	.
	Medium	30.5	4.993	28.0	3.071	32.3	3.878	27.4	5.016
	Small	29.6	4.251	31.1	9.562	29.8	5.357	28.7	4.739
国家									
American		31.9	6.452	30.0	9.115	32.6	4.919	31.0	8.179
European		31.0	5.612	28.3	3.559	33.3	4.608	28.4	7.328
Japanese		29.8	4.54	30.1	8.113	30.9	4.822	28.3	4.781

		均值		
类型	公司规模	利润 ($M)	销售额 ($M)	人均利润
Computer	big	1089.9	20597.48	4530.478
	medium	-85.75	3018.85	-3462.51
	small	44.94	1758.06	7998.815
	全部	240.87	5652.02	6159.015
Pharmaceutical	big	894.42	7474.04	17140.70
	medium	698.98	4261.06	24035.11
	small	156.95	1083.75	38337.19
	全部	690.08	5070.25	23546.12
全部	全部	409.32	5433.86	12679.18
类型				
Computer		240.87	5652.02	6159.015
Pharmaceutical		690.08	5070.25	23546.12
全部		409.32	5433.86	12679.18

目录

“制表”平台的示例 267
启动“制表”平台 271
使用对话框 272
添加统计量 273
制表输出 276
分析列 277
分组列 277
列表和行表 277
编辑表 278
“制表”平台选项 279
显示检验生成结果面板 280
列的右击菜单 280
“制表”平台的更多示例 281
创建不同表和重新排列内容的示例 281
将列组合到单个表的示例 284
使用页列的示例 286

“制表”平台的示例

了解如何使用“制表”平台以表格格式汇总数据。在本例中，您具有包含男生和女生的身高测量值的数据。您想创建一个表，它显示男生和女生的身高均值以及所有学生的聚合均值。

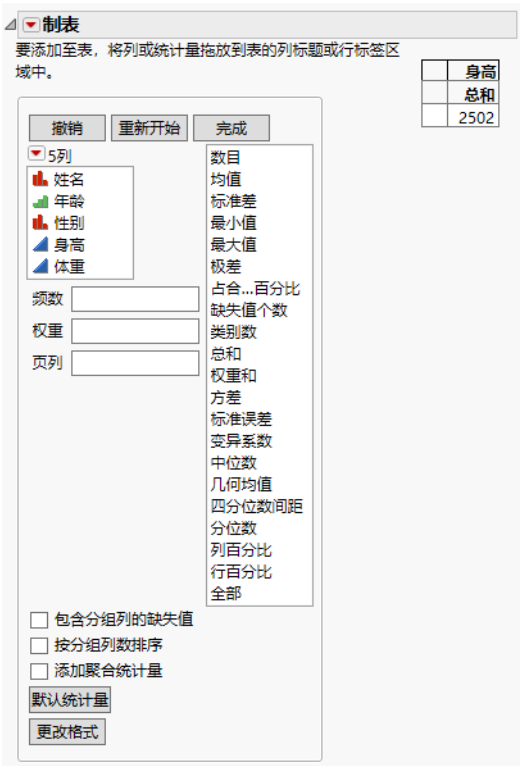
图 9.2 显示身高均值的表

	身高
性别	均值
F	60.9
M	63.9
全部	62.6

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **制表**。

   因为身高是您要检查的变量，您希望它显示在表的顶部。
3. 点击**身高**并将其拖放到 “列的拖放区”。

图 9.3 添加的 “身高”变量



您希望统计量按性别划分，希望性别显示在一侧。

4. 点击**性别**并将它拖放到数字 2502 旁边的空白单元格。

图 9.4 添加的 “性别”变量

您需要它显示均值而非显示总和。

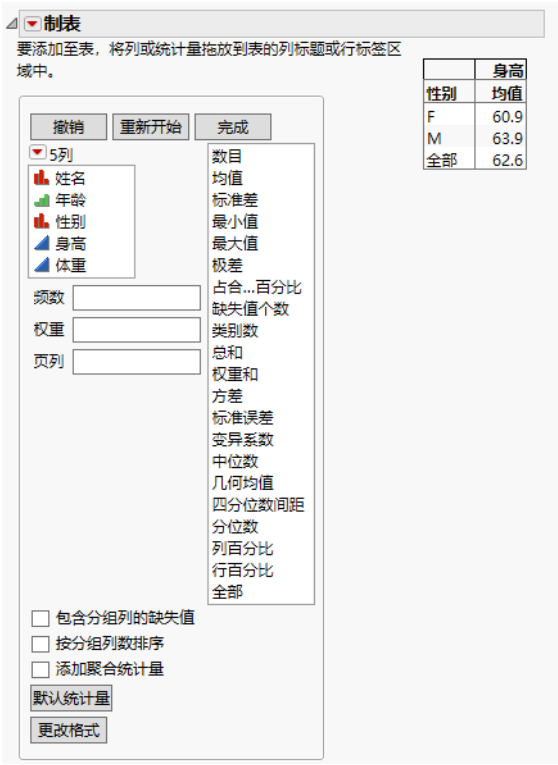
5. 点击**均值**并将它拖到**总和**上方。

图 9.5 添加的 “均值”统计量

您还想查看男生和女生的组合均值。

6. 点击**全部**并将它拖到**性别**上方。或者，您只需要选中**添加聚合统计量**复选框。

图 9.6 添加的 “全部”统计量



7. （可选）点击**完成**。

完成的表显示男生、女生的身高均值以及全体学生的组合身高均值。

启动“制表”平台

要启动“制表”平台，请选择**分析 > 制表**。

图 9.7 “制表”交互表

注意：有关红色小三角选项的详细信息，请参见第 279 页的““制表”平台选项”。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“制表”窗口包含以下选项：

交互表 / 对话框 在这两种模式之间切换。使用交互表模式可拖放项、创建定制表。使用对话框模式可使用固定格式创建简单表。请参见第 272 页的“使用对话框”。

统计量选项 列出标准统计量。将任意统计量从列表拖到表以合并它。请参见第 273 页的“添加统计量”。

列的拖放区 将列或统计量拖放到此处以创建列。

注意：若数据表包含的列名与“统计量”选项中的名称相同，一定要从列列表拖放列名。否则，JMP 可能会替换表中同名的统计量。

行的拖放区 将列或统计量拖放到此处以创建行。

提示：您还可以在列列表中选择一列或多列，选择一个或多个统计量，然后在拖放区上按住 Alt 键的同时点击（在 macOS 上是在按 Option 的同时点击），以在表中创建行或列。

产生的单元格 基于您拖放的列或统计量显示产生的单元格。

频数 标识数据表列，该列的值为每行分配频数。当为汇总数据中的每行指定频数时，该选项很有用。

权重 标识数据表列，该列的变量为数据指定权重 （如重要性或影响）。

页列 为名义型或有序型列的每个类别生成单独的表。请参见第 286 页的“使用页列的示例”。

包含分组列的缺失值 为分组列中的缺失值创建单独的组。未选中时，缺失值不包括在表中。请注意，已经定义为列属性的所有缺失值代码也被考虑在内。

按分组列数排序 将表顺序改为按分组列的值降序排序。

添加聚合统计量 添加所有行和列的聚合统计量。

默认统计量 允许您更改在拖放分析或非分析 （例如分组）列时显示的默认统计量。

更改格式 允许您更改特定统计量的数值显示格式。请参见第 275 页的 “更改数值格式”。

更改图形尺度 （仅当从红色小三角菜单选择**显示图表**时显示。）允许您指定统一定制尺度。

统一图形尺度 （仅当从红色小三角菜单选择**显示图表**时显示。）若取消选择该框，每列的直条使用分别由每个显示列中的数据确定的尺度。

使用对话框

若您不想通过拖放以交互方式生成表，则可以使用“制表”平台中的“对话框”界面创建简单的表。在选择**分析 > 制表**后，从菜单中选择**对话框**（如图 9.8 中所示）。您可以通过从红色小三角菜单中选择**显示控制面板**来更改表，然后将新项拖放到表。

图 9.8 使用对话框

对话框包含以下选项：

包含边缘统计量 聚合某个分组列类别的汇总信息。

对于分位数统计量，输入值 (%) 输入参数的特定百分比小于或等于的值。例如， 75% 的数据小于第 75 分位数。这适用于所有分组列。

统计量 选择一列后，选择适用于该列的标准统计量。请参见第 273 页的 “添加统计量”。

分组 （行标签） 选择要用作行标签的列。

分组 （列标签） 选择要用作列标签的列。

添加统计量

制表支持显示在控制面板中的一组标准统计量。您可以将任意关键字从该列表拖至表中，就像拖动列一样。请注意以下几点：

提示：您可以同时选择一列和一个统计量并将它们拖放到表中。

- 各个单元格关联的统计量是针对由分组列定义的类别中分析列所有观测的值计算得到的。
- 所有请求的统计量都必须位于同一维中，即：要么位于行表中，要么位于列表中。
- 若将连续列拖放到数据区域，该列将被视为分析列。

注意：分析列是您要计算其统计量的数值连续列。请参见第 277 页的“分析列”。

“制表”使用以下关键字：

数目 提供列中非缺失值的个数。这是不存在分析列时默认的统计量。

均值 提供列值的算术平均值。它是非缺失值的总和（并且在已定义的情况下，乘以**权重**变量）除以**权重和**。

标准差 提供为非缺失值计算的样本标准差。它是样本方差的平方根。

最小值 提供列中的最小非缺失值。

最大值 提供列中的最大非缺失值。

极差 提供**最大值**和**最小值**的差值。

占合计的百分比 计算占总体合计的百分比。计算中使用的分母是所有包含的观测的合计，分子是对应类别的合计。若不存在分析列，则“占合计的百分比”为占总计数的百分比。若存在分析列，则“占合计的百分比”为分析列总和占合计的百分比。因此，分母是分析列在所有包含的观测中的总和，分子是分析列针对该类别的总和。通过将关键字拖放到表中，您可以请求不同的百分比。

- 将一个或多个分组列从表中拖到“占合计的百分比”标题将更改分母定义。为此，“制表”将这些分组列的总和用作分母。
- 要获得列合计的百分比，请将行表上的所有分组列拖放至**占合计的百分比**标题（等同于“列百分比”）。同样，要获得行合计的百分比，请将列表上的所有分组列拖放至**占合计的百分比**标题（等同于“行百分比”）。

缺失值个数 提供缺失值的个数。

类别数 提供分析列中非重复类别的数目。

总和 提供列中所有值的总和。当表不存在任何其他统计量时，这是分析列的默认统计量。

权重和 提供列中所有**权重**值的总和。或者，若没有为任何列指定**权重**角色，**权重和**为非缺失值的总数。

方差 提供为非缺失值计算的样本方差。它是均值偏差的平方和除以非缺失值个数减去 1 所得的值。

标准误差 提供均值的标准误差。它是标准差除以**数目**的平方根所得的值。若为列指定了权重角色，则分母是权重和的平方根。

变异系数 （变异系数）提供散度的一种度量，它由标准差除以均值再乘以 100。

中位数 提供第 50 百分位数，在该值处有一半的数据低于第 50 分位数，一半数据高于或等于第 50 分位数 （中位数）。

几何均值 数据的积的第 n 个根。例如，几何均值通常用于计算利率。当数据包含偏斜分布中的较大值时，该统计量也很有用。

注意：负值生成缺失数字，零值 （无负值）生成零。

四分位间距 提供第三四分位数和第一四分位数之间的差值。

分位数 提供参数的特定百分比小于或等于其位置的值。例如，75% 的数据小于第 75 分位数。通过点击**分位数**关键字并将其拖到表，然后在显示的框中输入分位数，可以请求不同的分位数。

列百分比 提供在没有分析列的情况下每个单元格计数占其列合计的百分比。若存在分析列，则 “列百分比” 为列合计占分析列总和的百分比。对于在顶部具有统计量的表，可以将 “列百分比” 添加到具有多个行表 （垂直堆叠）的表。

行百分比 提供在没有分析列的情况下每个单元格计数占其行合计的百分比。若存在分析列，则 “行百分比” 为行合计占分析列总和的百分比。对于在一侧具有统计量的表，可以将 “行百分比” 添加到具有多个列表 （并排排列）的表。

全部 聚合某个分组列类别的汇总信息。

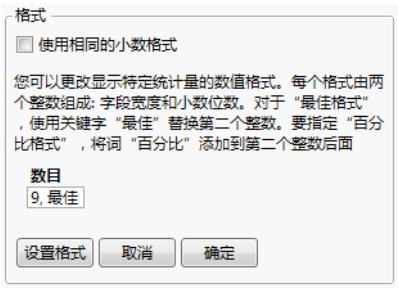
更改数值格式

每个单元格的格式取决于分析列和统计量。对于计数，默认格式没有小数位。对于某些统计量定义的各个单元格， JMP 会尝试使用所请求的分析列和统计量的格式确定合理格式。要覆盖默认格式，请执行以下步骤：

1. 点击 “制表” 窗口底部的**更改格式**按钮。
2. 在显示的面板中，输入字段宽度、逗号，然后输入要在表中显示的小数位数 （图 9.9）。
3. 要以 “百分比” 格式显示单元格值，请在小数位数后添加一个逗号，然后键入**百分比**一词。
4. （可选）若希望 JMP 为您确定要使用的最佳格式，请在文本框中键入**最佳**一词。

   JMP 现在会考虑每个单元格值的精度，并选择最佳方法来显示该值。
5. 点击**确定**执行更改并关闭 “格式” 部分，或点击**设置格式**查看已执行的更改但不关闭 “格式” 部分。

图 9.9 更改数值格式



制表输出

制表输出包含并排拼接的一个或多个列表，以及从上到下拼接的一个或多个行表。输出可能只有一个列表或一个行表。

图 9.10 制表输出

	性别											
	F						M					
	年龄						年龄					
	12	13	14	15	16	17	12	13	14	15	16	17
	体重	体重	体重	体重	体重	体重	体重	体重	体重	体重	体重	体重
最小值	64	67	81	92	112	116	79	79	92	104	128	134
均值	100.2	95.3	96.6	102.0	113.5	116.0	97.0	94.3	103.9	110.8	128.0	153.0
最大值	145	112	142	112	115	116	128	105	119	128	128	172

		性别							
		Female				Male			
		婚姻状况				婚姻状况			
		Married		Single		Married		Single	
		均值	标准差	均值	标准差	均值	标准差	均值	标准差
国家	汽车尺寸	年龄	年龄	年龄	年龄	年龄	年龄	年龄	年龄
American	Large	33.6	8.107	41.0	.	34.7	3.931	32.0	6.265
	Medium	31.4	5.827	29.0	9.258	31.3	5.413	32.1	11.05
	Small	31.0	5.657	29.0	9.539	31.8	4.813	26.5	6.455
European	Large	34.0	7.071	28.0	.	.	.	26.0	.
	Medium	31.0	5.06	28.7	5.508	32.3	5.62	31.0	10.13
	Small	29.8	6.611	28.0	1.414	33.8	4.381	25.7	2.517
Japanese	Large	25.0	.	.	.	32.0	.	.	.
	Medium	30.5	4.993	28.0	3.071	32.3	3.878	27.4	5.016
	Small	29.6	4.251	31.1	9.562	29.8	5.357	28.7	4.739
国家									
American		31.9	6.452	30.0	9.115	32.6	4.919	31.0	8.179
European		31.0	5.612	28.3	3.559	33.3	4.608	28.4	7.328
Japanese		29.8	4.54	30.1	8.113	30.9	4.822	28.3	4.781

		均值		
类型	公司规模	利润 ($M)	销售额 ($M)	人均利润
Computer	big	1089.9	20597.48	4530.478
	medium	-85.75	3018.85	-3462.51
	small	44.94	1758.06	7998.815
	全部	240.87	5652.02	6159.015
Pharmaceutical	big	894.42	7474.04	17140.70
	medium	698.98	4261.06	24035.11
	small	156.95	1083.75	38337.19
	全部	690.08	5070.25	23546.12
全部	全部	409.32	5433.86	12679.18
类型				
Computer		240.87	5652.02	6159.015
Pharmaceutical		690.08	5070.25	23546.12
全部		409.32	5433.86	12679.18

以交互方式创建表是一个迭代过程：

- 从相应的列表中点击项（列或统计量），将它们拖到行或列的拖放区。请参见第 278 页的“编辑表”和第 277 页的“列表和行表”。
- 通过重复拖放过程添加到表。该表会更新以反映最新添加的项。若已存在列标题或行标签，可以决定添加项相对于现有项的位置。

请注意以下点击和拖放事项：

- JMP 使用建模类型确定列的角色。假定连续列为分析列。请参见第 277 页的“分析列”。假定有序型列或名义型列为分组列。请参见第 277 页的“分组列”。
- 当您将多个列拖放到初始表时：
  - 若列共享一组共同值，它们将合并到单个表中。生成包含列名和从这些列收集的类别的交叉表。每个单元格由其中一列和其中一个类别共同定义。
  - 若列不共享共同值，它们将放置到单独的表。

– 通过右击一列并选择**组合表**或**单独的表**，始终可以更改默认操作。请参见第 280 页的“列的右击菜单”。

- 要嵌套列，请使用第一列创建一个表，然后将其他列拖放到第一列。
- 在正常创建的表中，所有分组列放置在一起，所有分析列放置在一起，所有统计量放置在一起。因此，JMP 不会在分析列的列表内散置统计量关键字。JMP 也不会在分组列的列表内插入分析列。
- 您可以将列从数据表的“表”面板拖到“制表”表，而不必使用“制表控制面板”。

注意：当您向打开的数据表添加数据、删除行以及对数据重新编码时，“制表”表随之更新。

分析列

“制表”平台中的分析列是要为其计算统计量的任何数值（连续）列。“制表”会为从分组列生成的每个类别计算分析列的统计量。

注意：所有分析列都必须位于同一维中，即：要么位于行表中，要么位于列表中。

分组列

“制表”平台中的分组列是您要用于将数据分为几个信息类别的列（名义型或有序型）。它们可以具有字符、整数甚至是小数值，但是应限制唯一值个数。

请注意以下几点：

- 若分组列嵌套，“制表”从列值的分层嵌套构造非重复的类别。例如，从性别值为“女性”和“男性”的分组列，婚姻状况值为“已婚”和“单身”的分组列，“制表”构造四个非重复类别：已婚女性、单身女性、已婚男性和单身男性。
- 您可以指定列表和行表的分组列。它们共同生成定义每个表单元格的类别。
- 默认情况下，“制表”不包括具有一个或多个分组列的缺失值的观测。您可以通过选中**包含分组列的缺失值**选项来包含缺失值。
- 要指定应作为缺失值处理的代码或值，请使用“缺失值代码”列属性。您可以通过选中**包含分组列的缺失值**选项来包含缺失值。有关缺失值代码的详细信息，请参见《使用 JMP》。

列表和行表

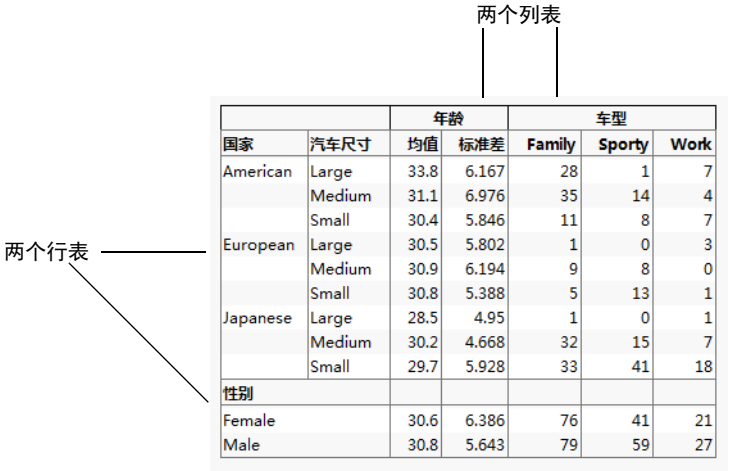
在“制表”平台中，表由其列标题和行标签定义。这些子表被称为**行表**和**列表**（图 9.11）。

“制表”平台中的行表和列表的示例

1. 选择**帮助 > 样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析 > 制表**。

3. 将**汽车尺寸**拖放至 “行的拖放区”。
4. 将**国家**拖放到**汽车尺寸**标题的左侧。
5. 将**均值**拖放到**数目**标题上方。
6. 将**标准差**拖放到**均值**标题下方。
7. 将**年龄**拖放到**均值**标题上方。
8. 将**车型**拖放到表的最右侧。
9. 将**性别**拖放到表下方。

图 9.11 行表和列表



		年龄		车型		
国家	汽车尺寸	均值	标准差	Family	Sporty	Work
American	Large	33.8	6.167	28	1	7
	Medium	31.1	6.976	35	14	4
	Small	30.4	5.846	11	8	7
European	Large	30.5	5.802	1	0	3
	Medium	30.9	6.194	9	8	0
	Small	30.8	5.388	5	13	1
Japanese	Large	28.5	4.95	1	0	1
	Medium	30.2	4.668	32	15	7
	Small	29.7	5.928	33	41	18
性别						
Female		30.6	6.386	76	41	21
Male		30.8	5.643	79	59	27

对于多个列表，一侧的标签由所有列表共享。在本实例中，产地和性别由各个表共享。同样，对于多个行表，顶部的标题由所有行表共享。在本实例中，年龄和车型由各个表共享。

编辑表

有几种方式可以编辑您添加到表中的项。您可以删除项、删除列标签或是编辑统计关键字或标签。

删除项

在将项添加到表后，可以通过以下任一种方式来删除它们：

- 将项拖离表。
- 要删除最后一项，请点击**撤销**。
- 右击某项并选择**删除**。

删除列标签

分组列的列名显示在与该列关联的类别上方。对于某些列，列名可能显得多余。通过右击列名并选择**删除列标签**，可从列表中删除列名。要重新插入列标签，请右击它的关联类别之一并选择**恢复列标签**。

编辑统计关键字和标签

您可以编辑统计关键字或统计标签。例如，您可能要使用“平均值”关键字而非“均值”。右击要编辑的字并选择**更改项标签**。在显示的框中，输入新标签。或者，您可以直接在编辑框中键入。

若您将一个统计量关键字更改为另一个统计量关键字，JMP 假定您实际上要更改统计量，而不仅是更改标签。这视同您从表中删除了该统计量然后添加另一个统计量。

“制表”平台选项

从“制表”旁边的红色小三角菜单可使用以下选项：

显示控制面板 显示控制面板，以便进一步交互。

显示表 以表格形式显示汇总数据。

显示图表 在条形图中显示反映汇总统计量表的汇总数据。使用简单的条形图，您可以直观比较汇总统计量的相对大小。默认情况下，条形图的所有列共享相同的尺度。通过清除**统一图形尺度**复选框，可以使条形图的每个列使用分别由每个显示列中的数据确定的尺度。您可以使用**更改图形尺度**按钮指定统一定制尺度。这些图表基于 0 或以 0 为中心。若数据全部为非负值或全部为非正值，则图表基线位于 0 处。否则，图表以 0 为中心。

显示阴影 有多个行时在表中显示灰色阴影框。

显示工具提示 显示悬停在表格区域上方时出现的提示。

显示检验生成结果面板 显示控制区域，该区域允许您使用来自原始表的随机样本创建检验生成结果。具有大量数据时，该选项特别有用。请参见第 280 页的“显示检验生成结果面板”。

制成数据表 从报表生成数据表。每个行表对应一个数据表，因为不同的行表标签可能不映射为同一结构。

完整路径列名 对所创建数据表中的列名使用分组列的完全限定列名。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

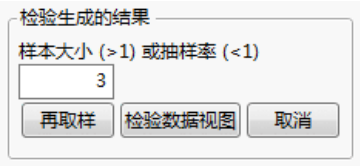
有关“选择列”红色小三角菜单中选项的说明，请参见《使用 JMP》。

显示检验生成结果面板

若您的数据表非常大，可能要使用数据表的一个小子集来尝试不同的表布局，以找到显示汇总信息的最佳方式。在这种情况下，JMP 会生成指定大小的随机子集，并在生成表时使用该子集。要使用检验生成结果功能，请执行以下步骤：

1. 点击“制表”红色小三角并选择**显示检验生成结果面板**。
2. 在**样本大小 (>1) 或抽样率 (<1)** 下的框中输入所需的样本大小，如图 9.12 中所示。样本大小可以为您输入的活动表的比例或活动表的行数。

图 9.12 检验生成结果面板



3. 点击**再取样**。
4. 要查看 JMP 数据表中的抽样数据，请点击**检验数据视图**按钮。当您关闭检验生成结果面板时，“制表”将使用整个数据表来按设计重新生成表。

列的右击菜单

右击表中的某列可看到以下选项：

删除 删除选定的列。

用作分组列 将分析列更改为分组列。

用作分析列 将分组列更改为分析列。

更改项标签 （仅对于单独的列或嵌套列显示。）输入新标签。

组合表（按类别分类的列） （仅对于单独的列或嵌套列显示。）合并单独的列或嵌套列。请参见第 284 页的 “将列组合到单个表的示例”。

嵌套分组列 垂直或水平嵌套分组列。

单独的表 （仅对合并表显示。）为每列创建一个单独的表。

删除列标签 从列表中删除列名。

恢复列标签 将隐藏的列名恢复到列表中。

按计数排序 按每个水平的计数从最多到最少对分组列的水平进行排序。该选项还可用于覆盖特定分组列的 “按分组列数排序”选项的设置。

“制表”平台的更多示例

- “创建不同表和重新排列内容的示例”
- “将列组合到单个表的示例”
- “使用页列的示例”

创建不同表和重新排列内容的示例

本例包含以下步骤：

1. “创建计数表”
2. “创建显示统计量的表”
3. “重新排列表内容”

创建计数表

假定您要创建一个表，该表包含的计数表示在调查中有多少人拥有日本、欧洲和美国产的车。您还想要按汽车尺寸细分的计数。

图 9.13 显示车辆拥有情况的计数的表

国家	汽车尺寸	数目
American	Large	36
	Medium	53
	Small	26
European	Large	4
	Medium	17
	Small	19
Japanese	Large	2
	Medium	54
	Small	92

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **制表**。
3. 点击**国家**并将其拖放到 “行的拖放区”。
4. 点击**汽车尺寸**并将其拖放到**国家**标题的右侧。

图 9.14 已添加到表的 “国家”和 “汽车尺寸”

国家	汽车尺寸	数目
American	Large	36
	Medium	53
	Small	26
European	Large	4
	Medium	17
	Small	19
Japanese	Large	2
	Medium	54
	Small	92

创建显示统计量的表

假定您要查看拥有每种汽车尺寸的人员的年龄均值（平均值）和标准差。

图 9.15 按年龄显示均值和标准差的表

国家	汽车尺寸			
American	Large	年龄	均值	33.8
			标准差	6.2
	Medium	年龄	均值	31.1
			标准差	7.0
	Small	年龄	均值	30.4
			标准差	5.8
European	Large	年龄	均值	30.5
			标准差	5.8
	Medium	年龄	均值	30.9
			标准差	6.2
	Small	年龄	均值	30.8
			标准差	5.4
Japanese	Large	年龄	均值	28.5
			标准差	4.9
	Medium	年龄	均值	30.2
			标准差	4.7
	Small	年龄	均值	29.7
			标准差	5.9

1. 从图 9.14 开始。点击**年龄**并将它拖放到**汽车尺寸**标题右侧。
2. 点击**均值**并将它拖放到 “总和”上方。
3. 点击**标准差**并将它拖放到 “均值”下方。

   将 “标准差”放置在表中 “均值”的下方。将 “标准差”放在 “均值”的上方会将 “标准差”放置在表中 “均值”的上方。

图 9.16 已添加到表的 “年龄”、“均值”和 “标准差”

国家	汽车尺寸			
American	Large	年龄	均值	33.8
			标准差	6.167
	Medium	年龄	均值	31.1
			标准差	6.976
	Small	年龄	均值	30.4
			标准差	5.846
European	Large	年龄	均值	30.5
			标准差	5.802
	Medium	年龄	均值	30.9
			标准差	6.194
	Small	年龄	均值	30.8
			标准差	5.388
Japanese	Large	年龄	均值	28.5
			标准差	4.95
	Medium	年龄	均值	30.2
			标准差	4.668
	Small	年龄	均值	29.7
			标准差	5.928

重新排列表内容

假定您希望将“汽车尺寸”放置在顶部，显示一个交叉表布局。

图 9.17 位于顶部的 “汽车尺寸”

	汽车尺寸					
	Large		Medium		Small	
	均值	标准差	均值	标准差	均值	标准差
国家	年龄	年龄	年龄	年龄	年龄	年龄
American	33.8	6.167	31.1	6.976	30.4	5.846
European	30.5	5.802	30.9	6.194	30.8	5.388
Japanese	28.5	4.95	30.2	4.668	29.7	5.928

重新排列表内容。

1. 从图 9.16 开始。点击**汽车尺寸**标题并将它拖放到表标题的右侧。

图 9.18 移动汽车尺寸

国家	汽车尺寸			
American	Large	年龄	均值	33.8
			标准差	6.167
	Medium	年龄	均值	31.1
			标准差	6.976
	Small	年龄	均值	30.4
			标准差	5.846
European	Large	年龄	均值	30.5
			标准差	5.802
	Medium	年龄	均值	30.9
			标准差	6.194
	Small	年龄	均值	30.8
			标准差	5.388
Japanese	Large	年龄	均值	28.5
			标准差	4.95
	Medium	年龄	均值	30.2
			标准差	4.668
	Small	年龄	均值	29.7
			标准差	5.928

→

			汽车尺寸		
国家			Large	Medium	Small
American	年龄	均值	33.8	31.1	30.4
		标准差	6.167	6.976	5.846
European	年龄	均值	30.5	30.9	30.8
		标准差	5.802	6.194	5.388
Japanese	年龄	均值	28.5	30.2	29.7
		标准差	4.95	4.668	5.928

2. 点击“年龄”并将它拖放到 Large Medium Small 标题下方。
3. 选择“均值”和“标准差”，然后将它们拖放到 Large 标题下方。

现在您的表已清晰显示数据。更容易查看按汽车尺寸和产地细分的车主年龄的均值和标准差。

将列组合到单个表的示例

在本例中，您具有学生的数据，指示儿童是否受欢迎的自我报告因素（成绩、体育、外貌、财力）的重要性。假定您要在具有更多统计量和因子的单个组合表中查看所有这些因子。

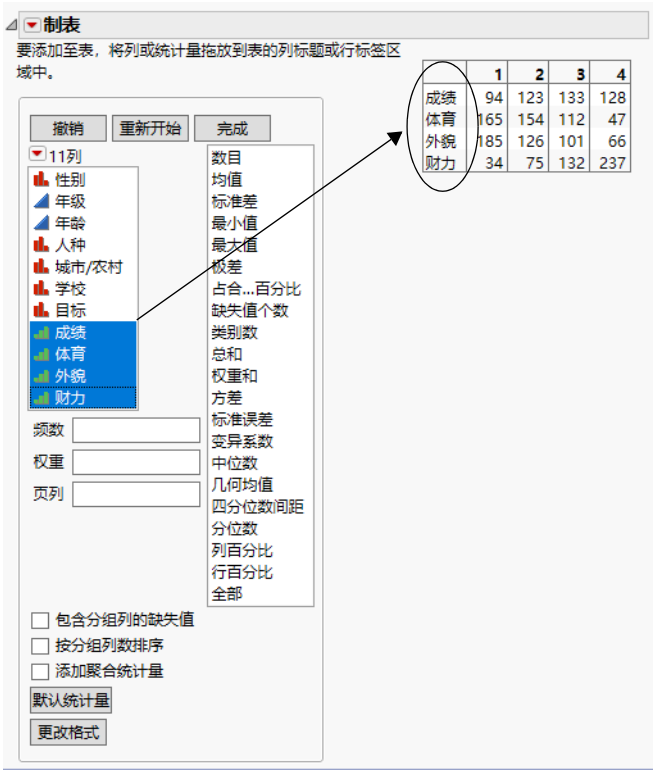
图 9.19 添加人口统计数据

		占合计的百分比														
		城市/农村														
		Rural					Suburban					Urban				
性别		1	2	3	4	全部	1	2	3	4	全部	1	2	3	4	全部
boy	成绩	2.30%	2.93%	3.56%	5.02%	13.81%	2.72%	5.86%	5.02%	5.02%	18.62%	3.14%	3.97%	5.44%	2.51%	15.06%
	体育	6.49%	3.97%	2.72%	0.63%	13.81%	11.30%	4.60%	2.30%	0.42%	18.62%	8.79%	3.97%	1.46%	0.84%	15.06%
	外貌	3.14%	4.60%	2.72%	3.35%	13.81%	3.77%	5.86%	5.44%	3.56%	18.62%	2.30%	5.02%	4.18%	3.56%	15.06%
	财力	1.88%	2.30%	4.81%	4.81%	13.81%	0.84%	2.30%	5.86%	9.62%	18.62%	0.84%	2.09%	3.97%	8.16%	15.06%
girl	成绩	4.39%	4.39%	4.39%	4.18%	17.36%	2.51%	3.35%	2.93%	4.18%	12.97%	4.60%	5.23%	6.49%	5.86%	22.18%
	体育	2.30%	6.69%	5.44%	2.93%	17.36%	1.67%	3.56%	5.65%	2.09%	12.97%	3.97%	9.41%	5.86%	2.93%	22.18%
	外貌	9.62%	3.35%	3.35%	1.05%	17.36%	7.95%	2.93%	1.26%	0.84%	12.97%	11.92%	4.60%	4.18%	1.46%	22.18%
	财力	1.05%	2.93%	4.18%	9.21%	17.36%	0.84%	3.14%	3.14%	5.86%	12.97%	1.67%	2.93%	5.65%	11.92%	22.18%

1. 选择**帮助** > **样本数据库**，然后打开 Children’s Popularity.jmp。
2. 选择**分析** > **制表**。

3. 选择**成绩**、**体育**、**外貌**和**财力**并将它们拖放到 “行的拖放区”。

图 9.20 按类别分类的列



请注意，显示单个组合表。

将每个类别的 1 到 4 评分的百分比制表。

4. 将**性别**拖放到左侧的空标题。
5. 将**占合计的百分比**拖放到编号的标题。
6. 将**全部**拖放到数字 4 旁边。

图 9.21 已添加到表的 “性别”、“占合计的百分比”和 “全部”

		占合计的百分比				
性别		1	2	3	4	全部
boy	成绩	8.16%	12.76%	14.02%	12.55%	47.49%
	体育	26.57%	12.55%	6.49%	1.88%	47.49%
	外貌	9.21%	15.48%	12.34%	10.46%	47.49%
	财力	3.56%	6.69%	14.64%	22.59%	47.49%
girl	成绩	11.51%	12.97%	13.81%	14.23%	52.51%
	体育	7.95%	19.67%	16.95%	7.95%	52.51%
	外貌	29.50%	10.88%	8.79%	3.35%	52.51%
	财力	3.56%	9.00%	12.97%	26.99%	52.51%

通过添加人口统计数据进一步细分制表。

7. 将**城市 / 农村**拖放到**占合计的百分比**标题下方。

图 9.22 已添加到表的 “城市 / 农村”

		占合计的百分比														
		城市/农村														
		Rural					Suburban					Urban				
性别		1	2	3	4	全部	1	2	3	4	全部	1	2	3	4	全部
boy	成绩	2.30%	2.93%	3.56%	5.02%	13.81%	2.72%	5.86%	5.02%	5.02%	18.62%	3.14%	3.97%	5.44%	2.51%	15.06%
	体育	6.49%	3.97%	2.72%	0.63%	13.81%	11.30%	4.60%	2.30%	0.42%	18.62%	8.79%	3.97%	1.46%	0.84%	15.06%
	外貌	3.14%	4.60%	2.72%	3.35%	13.81%	3.77%	5.86%	5.44%	3.56%	18.62%	2.30%	5.02%	4.18%	3.56%	15.06%
	财力	1.88%	2.30%	4.81%	4.81%	13.81%	0.84%	2.30%	5.86%	9.62%	18.62%	0.84%	2.09%	3.97%	8.16%	15.06%
girl	成绩	4.39%	4.39%	4.39%	4.18%	17.36%	2.51%	3.35%	2.93%	4.18%	12.97%	4.60%	5.23%	6.49%	5.86%	22.18%
	体育	2.30%	6.69%	5.44%	2.93%	17.36%	1.67%	3.56%	5.65%	2.09%	12.97%	3.97%	9.41%	5.86%	2.93%	22.18%
	外貌	9.62%	3.35%	3.35%	1.05%	17.36%	7.95%	2.93%	1.26%	0.84%	12.97%	11.92%	4.60%	4.18%	1.46%	22.18%
	财力	1.05%	2.93%	4.18%	9.21%	17.36%	0.84%	3.14%	3.14%	5.86%	12.97%	1.67%	2.93%	5.65%	11.92%	22.18%

您可以看到对于农村、郊区和城市中的男孩，体育是影响受欢迎的最重要因素。对于农村、郊区和城市中的女孩，外貌是影响受欢迎的最重要因素。

使用页列的示例

在本例中，您具有包含男生和女生的身高测量值的数据。您想创建一个表，它显示按学生年龄划分的身高均值。然后您想在不同表中按性别对数据分层。为此，添加分层列作为页列，它为每个组生成页。

图 9.23 按性别划分的学生身高均值

性别 = F

年龄	身高均值
12	58.6
13	59.0
14	62.6
15	63.0
16	62.5
17	62.0

男性

性别 = M

年龄	身高均值
12	57.3
13	61.3
14	65.3
15	65.2
16	68.0
17	69.0

女性

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 制表**。

   因为**身高**是您要检查的变量，您希望它显示在表的顶部。
3. 点击**身高**并将其拖放到 “列的拖放区”。

   您希望统计量按年龄划分，希望年龄显示在一侧。
4. 点击**年龄**并将它拖放到数字 2502 旁边的空白单元格。
5. 点击 “均值”并将其拖入 “总和”所在的单元格。

6. 点击**性别**并将它拖放到**页列**。
7. 从“页列”列表中选择 F，仅为女性显示身高均值。
8. 从“页列”列表中选择 M，仅为男性显示身高均值。您还可以选择**全不选**以显示所有值。

图 9.24 使用页列

第 10 章

JMP PRO 模拟

回答关于参数再抽样的难题

“模拟”平台仅在 JMP Pro 中提供。

“模拟”功能提供强大的参数和非参数模拟能力。使用“模拟”可执行以下操作：

- 扩展 bootstrap 以提供参数 bootstrapping。
- 获取非标准情况下的功效计算。
- 在非标准情况下近似统计量（如，预测值和置信区间）的分布。
- 执行置换检验。
- 探索关于预测变量的假设对于模型的影响。
- 探索与模型相关的各种“假设”情境。
- 评估新增或现有的统计方法。

“模拟”选项在很多报表中都提供，包括支持 Bootstrap 的所有报表。要访问“模拟”选项，请在报表中右击。

图 10.1 使用模拟进行功效分析

目录

“模拟”平台概述 291
关于使用“模拟”的示例 291
构造方差分量的半参数置信区间 292
执行置换检验 296
探索在广义回归中保留一个因子 299
执行非线性模型的前瞻功效分析 303
启动“模拟”窗口 311
“模拟”窗口 311
“模拟结果”表 312
“模拟结果”报表 312
“模拟功效”报表 312

JMP PRO “模拟”平台概述

“模拟”平台可为报表中的某列统计量提供模拟结果。右击报表中某列统计量并选择“模拟”。在“模拟”窗口中，指定模拟基于的数据表列。这是您**换出**的列。该列可在分析中具有任何角色。特别是，它可以是模型中的响应或预测变量。您随后可指定数据表中包含您要用于模拟的公式的列。这是您**换入**的列。它充当您换出的列的替代列。

注意：您的数据表必须包含具有随机成分的列。

该方法的原理如下。基于您换入的公式列中的公式生成模拟值列。用这个新的模拟值列替换您换出的列，重新运行生成包含相关统计量的报表的整个分析。该过程重复 *N* 次，其中，*N* 是您指定的总样本数。

“模拟”分析生成一个输出数据表，其中显示分析汇总。

- 数据表每行都表示一列模拟值的分析结果。
- 模拟中涉及的报表表的每行都对应一列。
- 可利用脚本来加快分析。

提示：“模拟”平台将重新运行在调用“模拟”的平台报表中显示的整个分析。报表中的额外分析可能会导致针对您选定的列的“模拟”运行较慢。若“模拟”时间较长，请先从平台报表中删除额外选项，然后再运行“模拟”。

“模拟”可用于除以下平台之外的所有统计平台：关联分析、关系图、多维尺度化、多重因子分析、可靠性框图、可靠性预测、可修复系统模拟、响应筛选和文本分析器。

JMP PRO 关于使用“模拟”的示例

了解如何在不同情况下获得数据的模拟结果。

- “构造方差分量的半参数置信区间”
- “执行置换检验”
- “探索在广义回归中保留一个因子”
- “执行非线性模型的前瞻功效分析”
- 有关显示如何模拟 Ppk 的置信区间和非正态变量的不合格百分比的示例，请参见《质量和过程方法》。

JMP PRO 构造方差分量的半参数置信区间

在本例中，您的目标是获得难以更改变量的方差分量的半参数置信区间。对于这些数据，您关注的是温度、时间和催化剂剂量对于某一反应的效应。温度是一个极难更改的变量（整区因子），时间是一个难以更改的变量（子区因子），催化剂剂量是一个易于更改的变量。有关整区和子区因子的信息，请参见《实验设计指南》。

之前的研究已经表明整区标准差大约是误差标准差的两倍，而子区误差大约是误差标准差的 1.5 倍。REML 报表中给出的 Wald 区间假定方差分量服从渐近正态分布，这些区间的覆盖性能较差。您使用方差分量模拟分布的百分位数来获取置信区间。

JMP PRO 构造设计

在本节中，为您的裂 - 裂区实验构造一个定制设计。

提示：若您想要跳过本节中的步骤，请选择**帮助 > 样本数据库**，然后打开 Design Experiment/Catalyst Design.jmp。在 Catalyst Design.jmp 数据表中，点击**实验设计模拟**脚本旁边的绿色小三角。然后转到第 294 页的“拟合模型”。

1. 选择**实验设计 > 定制设计**。
2. 在 “因子”分级显示项中的**添加因子数**旁边键入 3。
3. 点击**添加因子 > 连续**。
4. 通过双击将这些因子重命名为**温度**、**时间**和**催化剂**。

   保留这些因子的默认值为 –1 和 1。
5. 对于**温度**，点击**容易**并选择**极难**。

   这样即可将**温度**定义为一个整区因子。
6. 对于**时间**，点击**容易**并为**时间**选择**困难**。

   这样即可将**时间**定义为一个子区因子。
7. 点击**继续**。
8. 在 “模型”分级显示项中，选择**交互作用 > 二阶**。

   这会将所有双因子交互作用添加到该模型。
9. 点击 “定制设计”红色小三角，然后选择**模拟响应**。

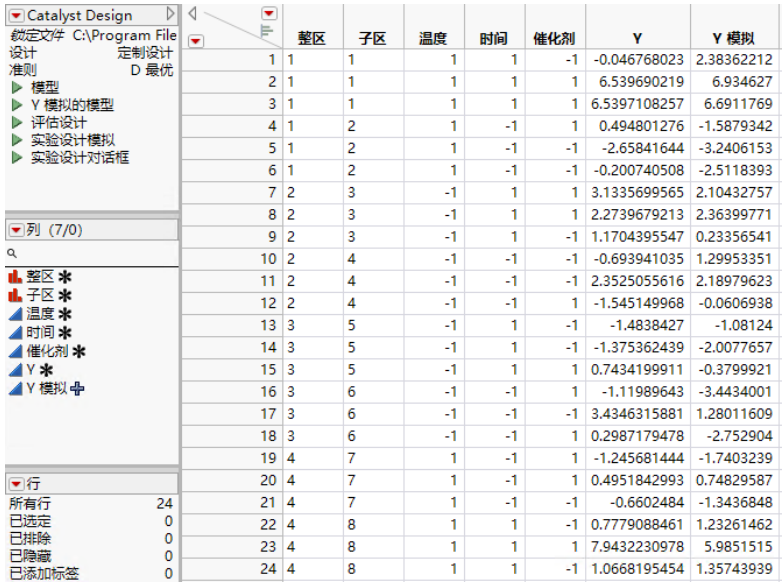
   这将在您选择 “制表”以构造设计表后打开 “模拟响应”窗口。

   注意：在第 10 步中设置 “随机种子”，在第 11 步中设置 “开始数”，这将会重现本例中显示的相同设计。自行构造设计时，这些步骤不是必需的。
10. （可选）点击 “定制设计”红色小三角并选择**设置随机种子**。键入 12345 并点击**确定**。
11. （可选）点击 “定制设计”红色小三角并选择**开始数**。键入 1000 并点击**确定**。

12. 点击**制作设计**。
13. 点击**制表**。

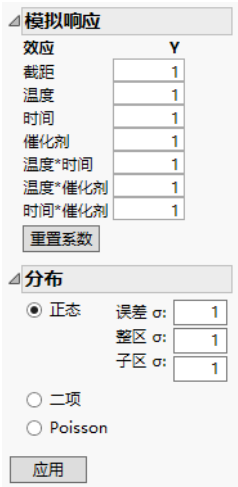
注意：Y 和 Y **模拟**列中的条目可能不同于图 10.2 中所示的条目。

图 10.2 设计表



	整区	子区	温度	时间	催化剂	Y	Y 模拟
1	1	1	1	1	-1	-0.046768023	2.38362212
2	1	1	1	1	1	6.539690219	6.934627
3	1	1	1	1	1	6.5397108257	6.6911769
4	1	2	1	-1	1	0.494801276	-1.5879342
5	1	2	1	-1	-1	-2.65841644	-3.2406153
6	1	2	1	-1	-1	-0.200740508	-2.5118393
7	2	3	-1	1	1	3.1335699565	2.10432757
8	2	3	-1	1	1	2.2739679213	2.36399771
9	2	3	-1	1	-1	1.1704395547	0.23356541
10	2	4	-1	-1	-1	-0.693941035	1.29953351
11	2	4	-1	-1	-1	2.3525055616	2.18979623
12	2	4	-1	-1	1	-1.545149968	-0.0606938
13	3	5	-1	1	-1	-1.4838427	-1.08124
14	3	5	-1	1	-1	-1.375362439	-2.0077657
15	3	5	-1	1	1	0.7434199911	-0.3799921
16	3	6	-1	-1	1	-1.11989643	-3.4434001
17	3	6	-1	-1	-1	3.4346315881	1.28011609
18	3	6	-1	-1	1	0.2987179478	-2.752904
19	4	7	1	-1	1	-1.245681444	-1.7403239
20	4	7	1	-1	1	0.4951842993	0.74829587
21	4	7	1	-1	-1	-0.6602484	-1.3436848
22	4	8	1	1	-1	0.7779088461	1.23261462
23	4	8	1	1	1	7.9432230978	5.9851515
24	4	8	1	1	-1	1.0668195454	1.35743939

图 10.3 “模拟响应”窗口



设计表和“模拟响应”窗口随即显示。请注意，该设计表包含一个**实验设计模拟**脚本。您可以随时运行该脚本以指定不同的参数值。

继续前进到下一节，从中可为整区和子区误差指定标准差，并拟合第一组模拟值的 REML 模型。

JMP PRO 拟合模型

在本节中，您使用 REML 方法拟合模型。假定整区和子区误差服从正态分布。根据其标准差的估计值，若误差标准差大约为 1 个单位，整区标准差则大约为 2 个单位，子区标准差大约为 1.5 个单位。由于您仅关注整区和子区变异，则不必更改分配给 “模拟响应”分级显示项中的 “效应”的值。

1. 在“分布”面板中（图 10.3）的**整区** σ 旁边键入 2。

   请注意，默认选定正态分布。这会导致在公式中加上正态误差。

2. 在**子区** σ 旁边键入 1.5。
3. 点击**应用**。

   在数据表中，**Y 模拟**的公式随之更新以反映您指定的内容。要查看该公式，点击列面板中列名右侧的加号。

4. 在数据表中，点击**模型**脚本旁边的绿色小三角。
5. 点击 Y 按钮旁边的 Y 变量，然后点击**删除**。
6. 点击 **Y 模拟**，然后点击 Y 按钮。

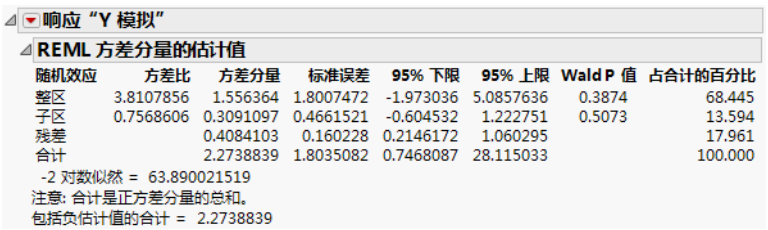
   该操作用包含模拟公式的列替换 Y。

7. 点击**运行**。

   拟合的模型基于一组模拟响应。

注意：由于 **Y 模拟**中的值是随机生成的，您报表中的条目可能不同于图 10.4 中显示的那些条目。

图 10.4 显示 Wald 置信区间的 REML 报表



响应 “Y 模拟”

REML 方差分量的估计值

随机效应	方差比	方差分量	标准误差	95% 下限	95% 上限	Wald P 值	占合计的百分比
整区	3.8107856	1.556364	1.8007472	-1.973036	5.0857636	0.3874	68.445
子区	0.7568606	0.3091097	0.4661521	-0.604532	1.222751	0.5073	13.594
残差		0.4084103	0.160228	0.2146172	1.060295		17.961
合计		2.2738839	1.8035082	0.7468087	28.115033		100.000

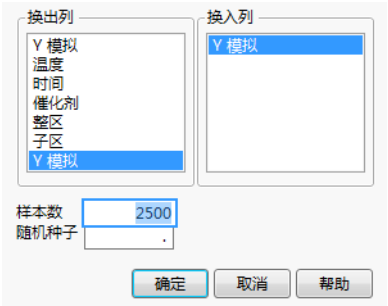
-2 对数似然 = 63.890021519
注意: 合计是正方差分量的总和。
包括负估计值的合计 = 2.2738839

JMP PRO 生成置信区间

在本节中，您将模拟方差分量估计值，并使用这些估计值来构造模拟的百分位数置信区间。

1. 在“REML 方差分量估计值”分级显示项中，右击**方差分量**列并选择**模拟**。

图 10.5 “模拟”窗口



在您的模拟中，用新的 **Y 模拟**列 （用于为每次模拟生成新的模拟值列）替换用于运行模型的列 **Y 模拟**。为 “参数估计值”表中列出的每个效应模拟您右击并显示为已选定的**方差分量**列。

2. 在**样本数**旁边输入 200。
3. （可选）在**随机种子**旁边，输入 456。

   这将重现图 10.6 中显示的值，只有第 1 行中的值除外。
4. 点击**确定**。

   第 1 行中的条目可能不同于图 10.6 中显示的那些条目。

图 10.6 方差分量模拟结果表 （部分视图）

拟合最小二乘法 模拟结果 (方...
随机种子 456
制作合并数据表
分布

列 (6/0)
Y *
SimID•
残差
合计
整区
子区

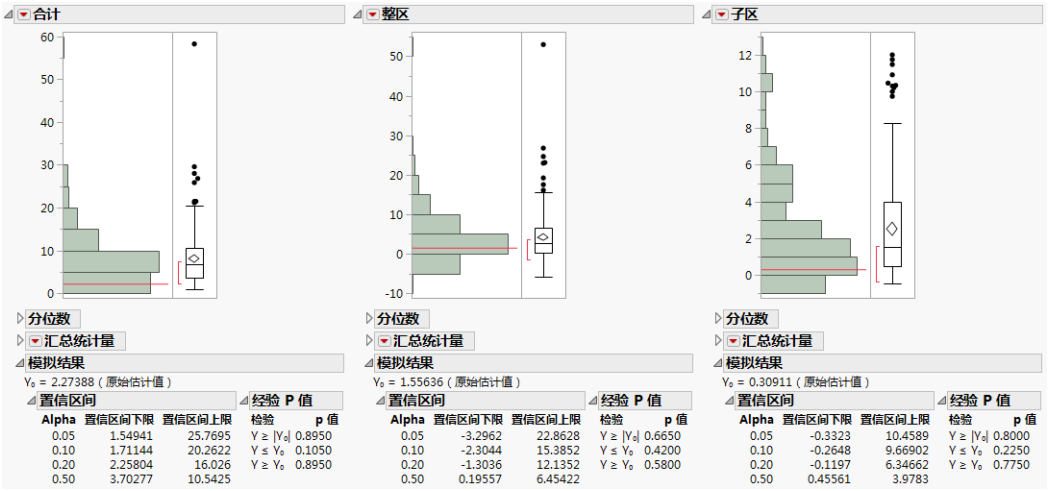
行
所有行 201
已选定 0
已排除 1
已隐藏 0
已添加标签 0

	Y	SimID•	残差	合计	整区	子区
1	Y Simulated	0	0.4084102799	2.2738839302	1.5563639995	0.3091096508
2	Y Simulated	1	1.7118437836	11.825684937	0.1160090855	9.9978320675
3	Y Simulated	2	0.7259348275	8.6529959081	3.3652848741	4.5617762065
4	Y Simulated	3	0.8416572537	26.836360491	22.95631251	3.0383907272
5	Y Simulated	4	0.9872034755	13.143778421	12.156574946	-0.176636748
6	Y Simulated	5	1.3514412755	1.3514412755	-0.060658009	-0.153702674
7	Y Simulated	6	0.9837455205	13.894478258	11.804365807	1.1063669307
8	Y Simulated	7	0.8559296173	3.7015003812	2.6824797817	0.1630909822
9	Y Simulated	8	1.0422128078	28.022776455	26.765100099	0.2154635485
10	Y Simulated	9	1.0302645404	17.850598404	12.75877132	4.0615625437
11	Y Simulated	10	1.2569403028	3.0738531832	-0.433818317	1.8169128804
12	Y Simulated	11	1.6719518802	11.105096973	2.4634884406	6.9696566519
13	Y Simulated	12	1.0326091926	7.2251377019	4.2166345958	1.9758939136
14	Y Simulated	13	0.7050109246	4.6661579836	2.8027318323	1.1584152266
15	Y Simulated	14	0.9281404099	6.53640214	0.3194564512	5.2888052789
16	Y Simulated	15	1.6586718874	3.1414624009	-0.270350632	1.4827905134
17	Y Simulated	16	0.9471174937	1.7108138759	-0.192580268	0.7636963822
18	Y Simulated	17	0.9782751744	2.927052512	0.756815441	1.1919618966
19	Y Simulated	18	1.5524955876	7.4052160139	4.8135329401	1.0391874862

“拟合最小二乘法模拟结果 （方差分量）”数据表的第一行包含**方差分量**的初始值，该行已被排除。其余行包含模拟值。

5. 运行**分布**脚本。

图 10.7 方差分量的分布图 （部分视图）



对于每个方差分量，各置信水平处的置信区间显示在 “模拟结果” 报表中。将每个表的 Alpha=0.05 行中的 95% 区间与 REML 报表中给定的区间进行比较 （图 10.4）：

- 整区方差分量的模拟 95% 置信区间为 -3.296 到 22.863。REML 报表中给出的 Wald 区间为 -1.973 到 5.086。
- 分区方差分量的模拟 95% 置信区间为 -0.332 到 10.459。REML 报表中给出的 Wald 区间为 -0.605 到 1.223。

使用模拟获得的区间要比从您的一组值计算得出的 REML 区间大得多。要获得更为精确的区间，请考虑运行更大量的模拟。

JMP PRO 执行置换检验

在本例中，您使用 “模拟” 执行随机化或置换试验。您正在研究三种药物治疗疼痛的效果，并关注这些药物在效果上是否不同。由于您只有极小的样本并且有些担心违反通常的方差分析假设，所以您可以使用 “模拟” 来执行置换检验。

首先，您构造了一个公式，该公式在三种药物之间随机排列疼痛测量值。在 “无效” 这一原假设下，以上任意分配都与其他任何分配相似。由此判定：以这种方式下获得的 *F* 比近似原假设下的 *F* 比的分布。最后，您将 *F* 比的观测值与模拟获得的零分布进行比较。

JMP PRO 定义模拟公式

1. 选择**帮助 > 样本数据库**，然后打开 Analgesics.jmp。
2. 选择**列 > 新建列**。
3. 为列名键入**随机排列的疼痛**。

4. 从 “列属性”列表中，选择**公式**。
5. 在函数列表中，选择**行** > Col Stored Value。
6. 在 “列”列表中，双击**疼痛**。
7. 点击编辑器面板上方符号列表中的插入键 (^)。
8. 从函数列表中，选择**随机** > Col Shuffle。

图 10.8 完成的公式

该公式随机排列**疼痛**列中的条目。

9. 在 “公式编辑器”窗口中点击**确定**。
10. 在 “列信息”窗口中点击**确定**。

执行置换检验

1. 选择**分析** > **以 X 拟合 Y**。
2. 选择**疼痛**并点击 **Y，响应**。
3. 选择**药物**并点击 **X，因子**。
4. 点击**确定**。
5. 点击 “单因子分析”红色小三角菜单并选择**均值 / 方差分析**。

图 10.9 “方差分析”报表

方差分析

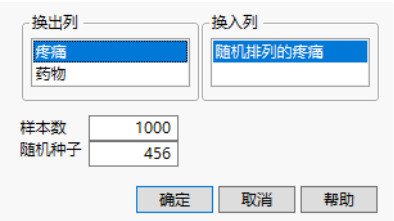
源	自由度	平方和	均方	F 比	概率>F
药物	2	99.89459	49.9473	6.2780	0.0053*
误差	30	238.67877	7.9560		
校正总和	32	338.57335			

请注意， *F* 比为 6.2780。

6. 在 “方差分析”分级显示项中，右击 “F 比”列并选择**模拟**。
7. 在 “换出列”列表中，点击**疼痛**。
8. 在 “换入列”列表中，点击**随机排列的疼痛**。
9. 在**样本数**旁边，输入 1000。
10. （可选）在**随机种子**旁边，输入 456。

这将重现本例中的值。

图 10.10 完成的 “模拟” 窗口

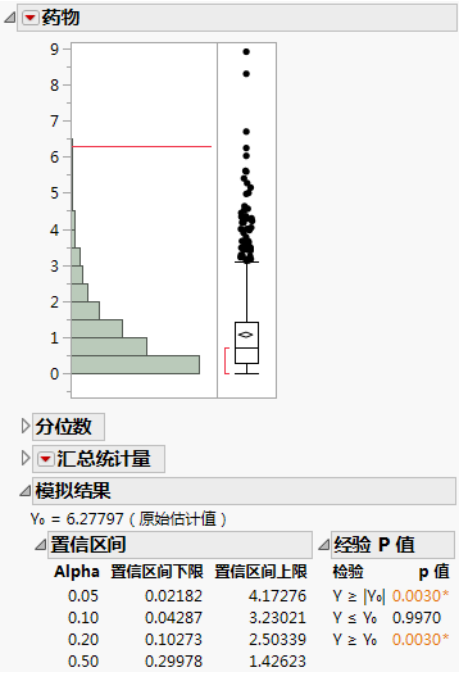


11. 点击**确定**。

    在模拟结果表中，“校正总和” 和 “误差” 列均为空，因为 “方差分析” 表中的 “*F* 比” 值仅应用于**药物**。

12. 在模拟值表中，运行**分布**脚本。

图 10.11 零分布下 *F* 比的模拟分布



观测到的 *F* 比值 6.2780 在直方图中用一条红线表示。该值落在 *F* 比的模拟零分布的上 0.5% 中。这表示：有很明显的证据表明这三种药物在缓解**疼痛**的效果上有差异。

JMP PRO 探索在广义回归中保留一个因子

在本例中，将构造一个广义回归模型并使用活跃参数拟合简化模型。基于该简化模型，使用模拟探索其中一个因子包含在该模型中的可能性。

一家医药制造公司掌握有关某种药片在人体内的溶解率以及可能影响溶解率的各种因子的历史信息。溶解率低于 70 的药片被视为有缺陷。您想要了解哪些因子影响溶解率。

JMP PRO 拟合模型

在本节中，您使用广义回归拟合模型。

提示：若您不想按照本节中的步骤执行，请点击 Tablet Production.jmp 数据表中**广义回归**脚本旁边的绿色小三角以获取该模型。

1. 选择**帮助 > 样本数据库**，然后打开 Tablet Production.jmp。
2. 选择**分析 > 拟合模型**。
3. 点击**溶解**并点击 Y。
4. 从**搅拌时间**一直选择到**喷嘴压力**，然后点击**添加**。
5. 从 “特质”列表中选择**广义回归**。
6. 点击**运行**。
7. 在 “模型启动”面板中，选择**自适应**框。
8. 在 “模型启动”面板中，点击**执行**。

图 10.12 基于自适应 Lasso 的模型

原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	108.6406	24.332975	19.933977	<.0001*	60.948851	156.33236
搅拌时间	0.130278	0.028185	21.365139	<.0001*	0.0750364	0.1855195
筛孔尺寸[3-5]	4.1877616	0.541493	59.8106	<.0001*	3.1264548	5.2490685
筛孔尺寸[4-5]	2.3907729	0.567217	17.765534	<.0001*	1.2790481	3.5024977
硬脂酸镁供应商[Jones Inc-Smith Ind]	0	0	0	1.0000	0	0
乳糖供应商[Bond Inc-James Ind]	0	0	0	1.0000	0	0
糖供应商[Sour-Sweet]	0	0	0	1.0000	0	0
滑石供应商[Rough-Smooth]	0	0	0	1.0000	0	0
混合时间	0.7223423	0.1381848	27.325338	<.0001*	0.4515051	0.9931796
混合速度	0.2130888	0.2389205	0.7954524	0.3725	-0.255187	0.6813644
压缩机[Compress1-Compress2]	-0.538112	0.3993673	1.8155148	0.1778	-1.320857	0.2446338
力量	0	0	0	1.0000	0	0
涂层供应商[Coat-Mac]	0	0	0	1.0000	0	0
涂层供应商[Down-Mac]	0	0	0	1.0000	0	0
涂层粘度	0.1834341	0.0500838	13.414242	0.0002*	0.0852717	0.2815965
进气温度	0	0	0	1.0000	0	0
排气温度	0	0	0	1.0000	0	0
喷涂率	-0.204356	0.0420142	23.658267	<.0001*	-0.286702	-0.12201
喷嘴压力	0	0	0	1.0000	0	0

“正态”分布参数	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
尺度	2.0224259	0.1641142	151.86339	<.0001*	1.700768	2.3440838

您关注的是“使用 ‘AICc’验证的 ‘正态’自适应 ‘Lasso’”报表中显示的参数估计值。基于非零参数估计值，该模型表明**搅拌时间**、**筛孔尺寸**、**混合时间**、**混合速度**、**压缩机**、**涂层粘度**和**喷涂率**均与**溶解**有关。

JMP PRO 简化模型

简化模型之前，确保没有在 Tablet Production.jmp 数据表中选定任何列。选定的列不会在下面的第一步中取消选择。确保未选定任何列可防止意外包含带清零项的列。

若您不想按照本节中的步骤执行，请点击 Tablet Production.jmp 数据表中**广义回归简化模型**脚本旁边的绿色小三角以获取简化模型。

1. 点击“使用 ‘AICc’验证的 ‘正态’自适应 ‘Lasso’”旁边的红色小三角，然后选择**重新启动活跃集 > 使用活跃效应重新启动**。

   这将打开“拟合模型”窗口，其中将“参数估计值”报表中带非零系数估计值的项放入“构造模型效应”列表。响应输入为 Y。已选定 “广义回归”特质。
2. 点击**运行**。
3. 在 “模型启动”面板中，选择**自适应**框。
4. 在 “模型启动”面板中，点击**执行**。

图 10.13 使用自适应 Lasso 的简化模型

原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	95.142391	23.430178	16.489099	<.0001*	49.220085	141.0647
搅拌时间	0.1394103	0.0275609	25.585971	<.0001*	0.0853919	0.1934288
筛孔尺寸[3-5]	4.3323833	0.534237	65.763638	<.0001*	3.285298	5.3794685
筛孔尺寸[4-5]	2.6331283	0.5457852	23.275584	<.0001*	1.563409	3.7028476
混合时间	0.7583048	0.1385246	29.966364	<.0001*	0.4868017	1.029808
混合速度	0.4575802	0.2289168	3.9955744	0.0456*	0.0089116	0.9062488
压缩机[Compress1-Compress2]	-0.877986	0.4127286	4.5252866	0.0334*	-1.686919	-0.069053
涂层粘度	0.198673	0.0486798	16.656386	<.0001*	0.1032624	0.2940836
喷涂率	-0.212753	0.0410929	26.805238	<.0001*	-0.293294	-0.132213

“正态” 分布参数	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
尺度	1.9982636	0.1574951	160.97992	<.0001*	1.689579	2.3069483

请注意，**混合速度**估计值的置信区间（95% 下限）非常接近包含零时的置信区间。下一步，执行模拟研究，以查看**混合速度**包含在以下模型中的频率：若来自溶解分布的其他数据值被观测到的话。

JMP PRO 探索在模型中包含混合速度的效果

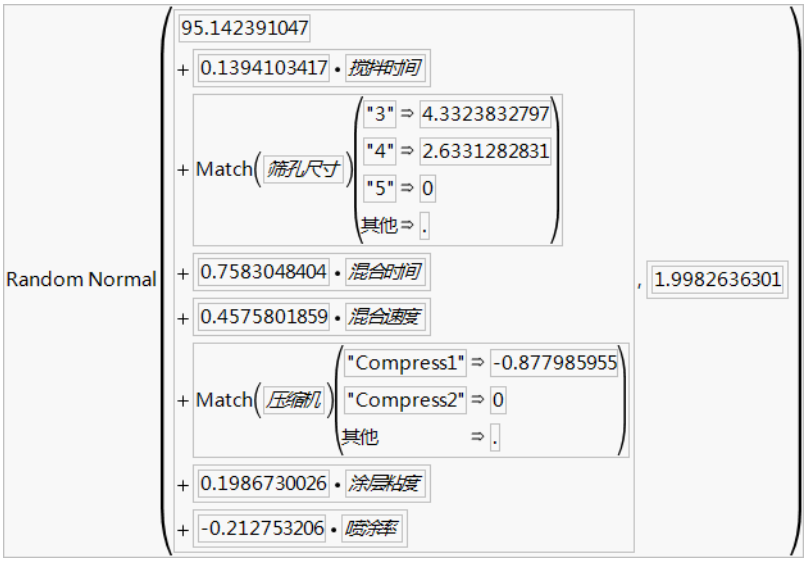
使用以下步骤中的简化模型 （图 10.13）报表。

1. 点击“使用 ‘AICc’验证的 ‘正态’自适应 ‘Lasso’”旁边的红色小三角，然后选择**保存列 > 保存模拟公式**。

   这将会向 Tablet Production.jmp 数据表添加名为 **“溶解”模拟公式**的新列。

2. （可选）在数据表列面板中，点击 **“溶解”模拟公式**右边的加号。

图 10.14 模拟公式



对于每行，该公式都模拟在给定模型和**溶解**分布的前提下可能获取的一个值；估计该分布为标准差约为 1.998 的 “正态”分布。

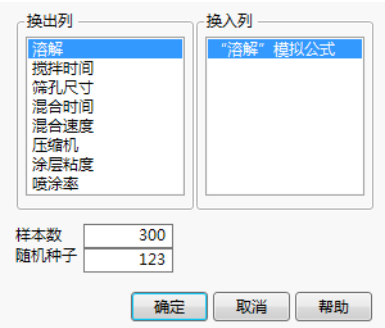
3. 点击**取消**。
4. 返回简化模型报表窗口。在 “原始预测变量的参数估计值”报表中，右击 “估计值”列并选择**模拟**。

   确保在 “换出列”列表中选定了**溶解**。
5. 在**样本数**旁边，输入 300。

   对于该模拟，您要求 JMP 使用 **“溶解”模拟公式**列用模拟值替换 300 个分析中每一个分析中的**溶解**列。
6. （可选）将**随机种子**设置为 123。

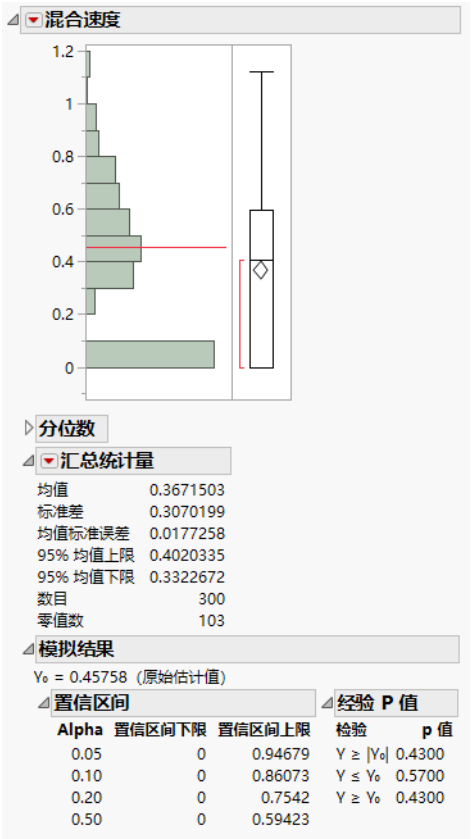
   这将重现本例中的值。

图 10.15 完成的 “模拟” 窗口



7. 点击**确定**。
   表的第一行包含 “估计值” 的初始值，并且已被排除。其余行包含模拟值。
8. 运行**分布**脚本。
9. 按 Ctrl 键，点击**混合速度**红色小三角并选择**显示选项 > 定制汇总统计量**。
10. 选择**零值数**。
11. 点击**确定**。
12. 滚至**混合速度**的 “分布” 报表。

图 10.16 模拟的混合速度系数估计值的直方图



“汇总统计量”报表显示对于 103/300 = 34.3% 的模拟，**混合速度**估计值均为零。

JMP PRO 执行非线性模型的前瞻功效分析

在本例中，您关注的是六个连续因子针对某个部件是通过还是未通过质检的主效应。响应为二项响应，您可以总共运行 60 次。

计划示例

您需要使用广义线性模型（Logit 作为连结函数）对成功概率建模。Logit 连结函数拟合 Logistic 模型：

$$\pi(\mathbf{X}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \ldots + \beta_6 X_6)}}$$

其中，$\pi(\mathbf{X})$ 表示处在给定的设计设置 $\mathbf{X} = (X_1, X_2, \ldots, X_6)$ 时，某部件通过的概率。

用 $L(\mathbf{X})$ 表示线性预测变量：

$$L(\mathbf{X}) = \beta_0 + \beta_1 X_1 + \ldots + \beta_6 X_6$$

接下来，您将探索线性预测变量系数的以下值的功效：

系数	值
β_0	0
β_1	1
β_2	0.9
β_3	0.8
β_4	0.7
β_5	0.6
β_6	0.5

由于线性预测变量的截距为 0，所以当所有因子设置为 0 时，通过的部件的概率等于 50%。在其他所有因子都设置为 0 时，与第 *i* 个因子的水平关联的概率如下所示。

因子	$\mathbf{X_i}$ = 1 时通过的百分比	$\mathbf{X_i}$ = -1 时通过的百分比	差值
X_1	73.11%	26.89%	46.2%
X_2	71.09%	28.91%	42.2%
X_3	69.00%	31.00%	38.0%
X_4	66.82%	33.18%	33.6%
X_5	64.56%	35.43%	29.1%
X_6	62.25%	37.75%	24.5%

例如，在除 X_1 之外的其他所有因子都设置为 0 时，您想要检测的通过率的差值为 46.2%。在除 X_6 之外的其他所有因子都设置为 0 时，出现您要检测的通过率的最小差值，即 24.5%。

JMP PRO 构造设计

在本节中，为您的实验构造一个定制设计。

注意：若您想要跳过本节中的步骤，请选择**帮助 > 样本数据库**，然后打开 Design Experiment/Binomial Experiment.jmp。点击**实验设计模拟**脚本旁边的绿色小三角，然后转到第 306 页的“定义模拟响应”。

1. 选择**实验设计 > 定制设计**。

   注意：尽管定制设计对于非线性情形并不是最佳的，但在本例中，为了简便起见，您可以使用 “定制设计”平台而不是 “非线性设计”平台。要查看解释使用 “非线性设计”平台构造的设计优于正交设计的原因的示例，请参见 《实验设计指南》。

2. 在 “因子”分级显示项中的**添加因子数**旁边键入 6。
3. 点击**添加因子 > 连续**。
4. 点击**继续**。

   您构造的是主效应设计，所以不要对 “模型”分级显示项进行任何更改。

5. 在 “试验次数”下的**用户指定**旁边键入 60。
6. 点击 “定制设计”红色小三角，然后选择**模拟响应**。

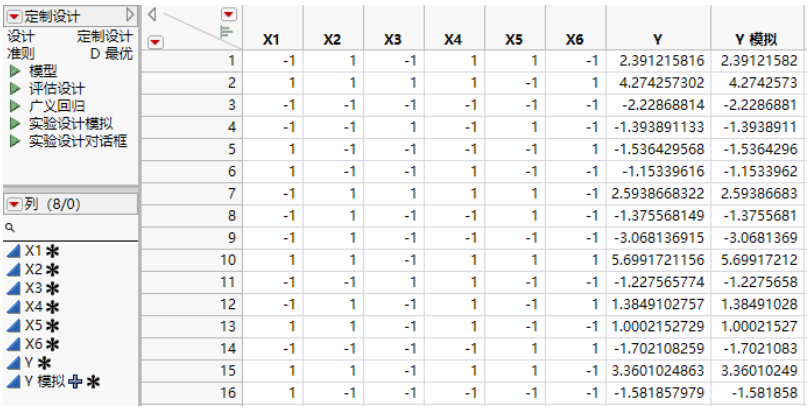
   这将在您选择 “制表”以构造设计表后打开 “模拟响应”窗口。

   注意：在第 7 步中设置 “随机种子”，在第 8 步中设置 “开始数”，这将会重现本例中显示的相同设计。自行构造设计时，这些步骤不是必需的。

7. （可选）点击 “定制设计”红色小三角并选择**设置随机种子**。键入 12345 并点击**确定**。
8. （可选）点击 “定制设计”红色小三角并选择**开始数**。键入 1 并点击**确定**。
9. 点击**制作设计**。
10. 点击**制表**。

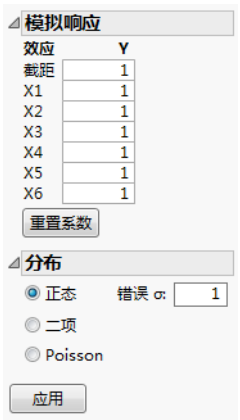
    注意：Y 和 **Y 模拟**列中的条目可能不同于图 10.17 中所示的条目。

图 10.17 设计表的部分视图



	X1	X2	X3	X4	X5	X6	Y	Y 模拟
1	-1	1	-1	1	1	-1	2.391215816	2.39121582
2	1	1	1	1	-1	1	4.274257302	4.2742573
3	-1	-1	-1	-1	-1	-1	-2.22868814	-2.2286881
4	-1	-1	1	-1	1	-1	-1.393891133	-1.3938911
5	1	-1	-1	-1	-1	1	-1.536429568	-1.5364296
6	1	-1	-1	1	-1	-1	-1.15339616	-1.1533962
7	-1	1	1	1	1	-1	2.5938668322	2.59386683
8	-1	1	-1	-1	1	-1	-1.375568149	-1.3755681
9	-1	1	-1	-1	-1	-1	-3.068136915	-3.0681369
10	1	1	-1	1	1	1	5.6991721156	5.69917212
11	-1	-1	1	1	-1	-1	-1.227565774	-1.2275658
12	-1	1	-1	1	-1	1	1.3849102757	1.38491028
13	1	1	-1	1	-1	-1	1.0002152729	1.00021527
14	-1	-1	-1	-1	1	1	-1.702108259	-1.7021083
15	1	1	-1	1	1	-1	3.3601024863	3.36010249
16	1	-1	-1	-1	-1	-1	-1.581857979	-1.581858

图 10.18 “模拟响应”窗口



设计表和 “模拟响应”窗口随即显示。有两列添加到设计表中：

- Y 包含根据“模拟响应”窗口中的指定设置模拟的一组值。
- Y **模拟**包含一个公式，该公式使用在“模拟响应”窗口中为该模型指定的公式计算其值。要查看该公式，点击列面板中列名右侧的加号。

继续前进到下一节，在其中模拟二项响应并拟合这些模拟响应的广义线性模型。

JMP PRO 定义模拟响应

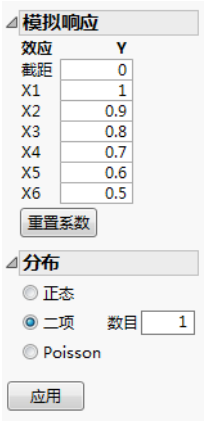
您的计划是要模拟二项响应数据，其成功概率由 Logistic 模型给出。有关“模拟响应”的详细信息，请参见《实验设计指南》。

注意：若想要跳过本节中的步骤，请点击**模拟模型响应**脚本旁边的绿色小三角。然后转到第 308 页的“拟合广义线性模型”。

1. 在“模拟响应”窗口（图 10.18）中，在 Y 下输入以下值：
   - 在“截距”旁边键入 0。
   - 在 X1 旁边，默认输入的是 1。请保留该值。
   - 在 X2 旁边，键入 0.9。
   - 在 X3 旁边，键入 0.8。
   - 在 X4 旁边，键入 0.7。
   - 在 X5 旁边，键入 0.6。
   - 在 X6 旁边，键入 0.5。
2. 在 “分布”分级显示项中，选择**二项**。

   保持**数目**的值设置为 1，指示每次试验只有一个单位。

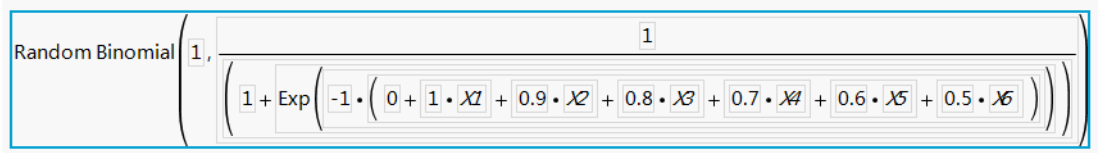
图 10.19 完成的 “模拟响应”窗口



3. 点击**应用**。

   在设计数据表中，Y **模拟**列替换为生成二项值的公式列。称为 Y n **次试验**的列指示每次运行的试验数。
4. （可选）点击列面板中 Y **模拟**右侧的加号。

图 10.20 “Y 模拟”的随机二项公式



5. 点击**取消**。

JMP PRO 拟合广义线性模型

在本节中，使用“广义线性模型”特质拟合 Logistic 模型。

1. 在数据表中，点击**模型**脚本旁边的绿色小三角。
2. 点击 Y 按钮旁边的 Y 变量，然后点击**删除**。
3. 点击 Y **模拟**，然后点击 Y 按钮。

   您要使用包含随机生成的二项值的列来替换 Y。
4. 从 “特质”列表中选择**广义线性模型**。
5. 从 “分布”列表中选择**二项**。

   请注意， Logit 函数显示在 “连结函数”菜单中。
6. 点击**运行**。

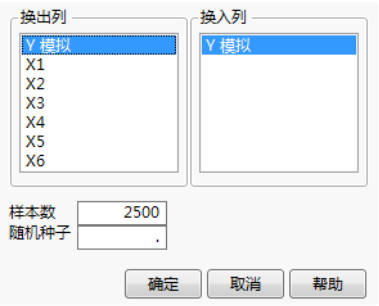
   拟合的模型基于一组模拟二项响应。

JMP PRO 探索功效

在本节中，模拟似然比检验 *p* 值，以探索检测由线性预测变量（带有第 303 页的“计划示例”中给定的系数值）确定的某范围概率值中存在的差异的功效。

1. 在“效应检验”分级显示项中，右击**概率 > 卡方**列，然后选择**模拟**。

图 10.21 “模拟”窗口



确保在 “换出列”列表中选定了 Y **模拟**列。该列包含用于拟合模型的值。当您选择 “换入列”下的列 Y **模拟**后，对于每次模拟，您都在指示 JMP 将 Y **模拟**中的值替换为使用 Y **模拟**列中的公式模拟的一列新值。

您在报表中已选定的列**概率 > 卡方**是关于相关的主效应是否为 0 的似然比检验的 *p* 值。为“效应检验”表中所列的每个效应都模拟 “概率 > 卡方”值。

2. 在**样本数**旁边，键入 500。
3. 点击**确定**。

   随即显示 “广义线性模型模拟结果”数据表。

注意：由于响应值是模拟的，您的模拟 *p* 值可能不同于图 10.22 中显示的结果。

图 10.22 模拟结果表，部分视图

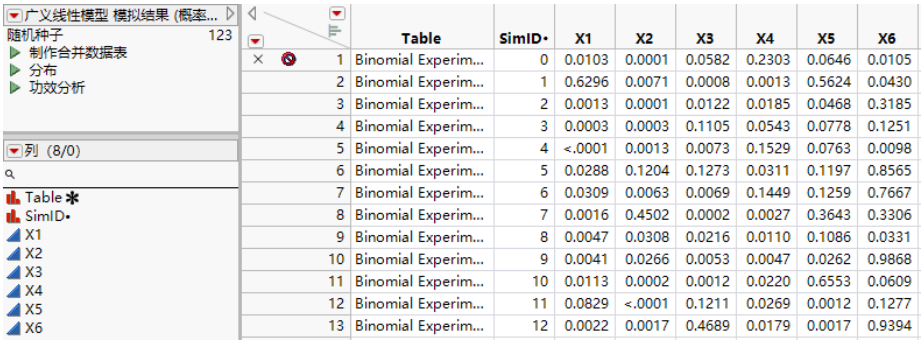


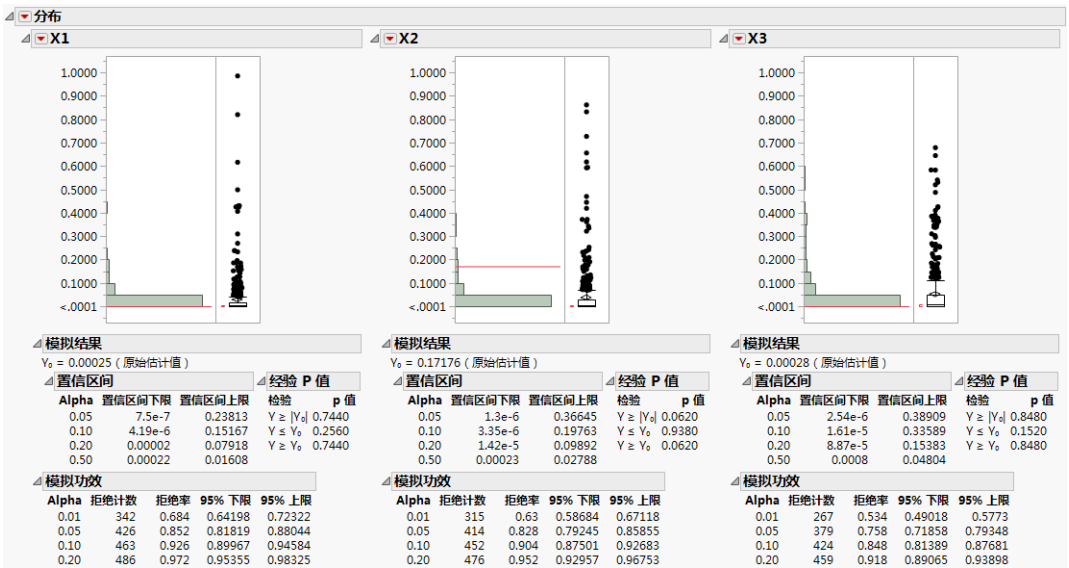
	Table	SimID	X1	X2	X3	X4	X5	X6
1	Binomial Experim...	0	0.0103	0.0001	0.0582	0.2303	0.0646	0.0105
2	Binomial Experim...	1	0.6296	0.0071	0.0008	0.0013	0.5624	0.0430
3	Binomial Experim...	2	0.0013	0.0001	0.0122	0.0185	0.0468	0.3185
4	Binomial Experim...	3	0.0003	0.0003	0.1105	0.0543	0.0778	0.1251
5	Binomial Experim...	4	<.0001	0.0013	0.0073	0.1529	0.0763	0.0098
6	Binomial Experim...	5	0.0288	0.1204	0.1273	0.0311	0.1197	0.8565
7	Binomial Experim...	6	0.0309	0.0063	0.0069	0.1449	0.1259	0.7667
8	Binomial Experim...	7	0.0016	0.4502	0.0002	0.0027	0.3643	0.3306
9	Binomial Experim...	8	0.0047	0.0308	0.0216	0.0110	0.1086	0.0331
10	Binomial Experim...	9	0.0041	0.0266	0.0053	0.0047	0.0262	0.9868
11	Binomial Experim...	10	0.0113	0.0002	0.0012	0.0220	0.6553	0.0609
12	Binomial Experim...	11	0.0829	<.0001	0.1211	0.0269	0.0012	0.1277
13	Binomial Experim...	12	0.0022	0.0017	0.4689	0.0179	0.0017	0.9394

表的第一行包含**概率 > 卡方**的初始值，并且已被排除。其余 500 行包含模拟值。

4. 运行**功效分析**脚本。

注意：由于响应值是模拟的，您的模拟功效结果可能不同于图 10.23 中显示的结果。

图 10.23 前两个效应的分布图



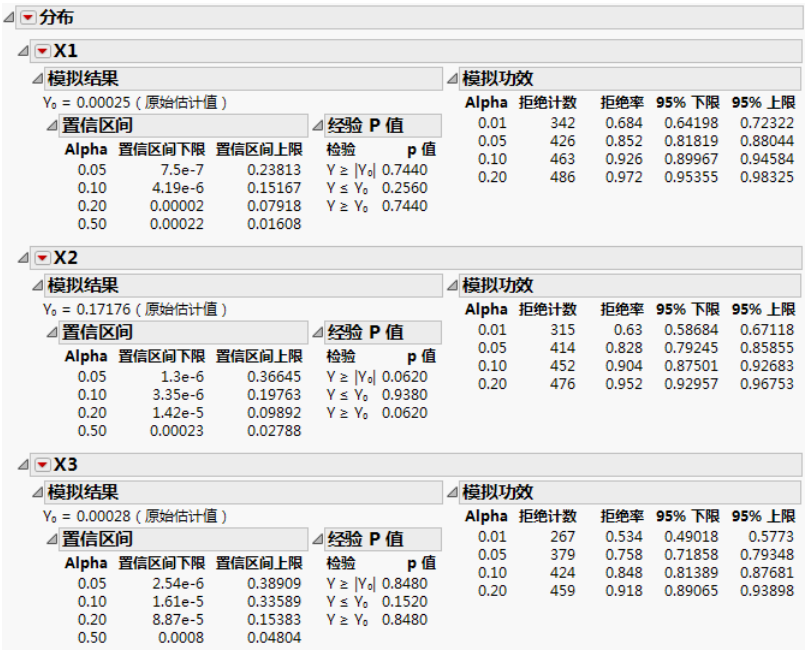
直方图为每个主效应都绘制了 500 次模拟的概率 > 卡方值。“模拟功效”分级显示项显示这 500 次模拟中的模拟 “拒绝率”。

为便于查看，堆叠这些报表并且取消选择这些图。

5. 点击 “分布” 红色小三角并选择**堆叠**。
6. 按 Ctrl 并点击 X1 红色小三角，然后取消选择**离群值箱线图**。
7. 按 Ctrl 并点击 X1 红色小三角，选择**直方图选项**并取消选择**直方图**。

注意：由于响应值是模拟的，您的模拟功效结果可能不同于图 10.24 中显示的结果。

图 10.24 前三个效应的功效结果



在 “模拟功效” 分级显示项中，每行的 “拒绝率” 给出小于相应的 Alpha 的 *p* 值的比例。例如，对于 X3（对应于系数值 0.8，概率差值为 38%），0.05 显著性水平下的模拟功效为 379/500 = 0.758。表 10.1 汇总了所有效应在 0.05 显著性水平下的模拟功效。请注意功效如何随着 “待检差值” 的下降而下降。还请注意，检验 24.5% 这样大的效应 (X6) 的功效仅约为 0.37。

注意：由于响应值是模拟的，您的模拟功效结果可能不同于表 10.1 中显示的结果。

表 10.1 显著性水平 0.05 下的模拟功效

因子	$X_i = 1$ 时通过的百分比	$X_i = -1$ 时通过的百分比	待检差值	Alpha=0.05 时的模拟功效（拒绝率）
X_1	73.11%	26.89%	46.2%	0.852

表 10.1 显著性水平 0.05 下的模拟功效 (续)

因子	$X_i = 1$ 时通过的百分比	$X_i = -1$ 时通过的百分比	待检差值	Alpha=0.05 时的模拟功效（拒绝率）
X_2	71.09%	28.91%	42.2%	0.828
X_3	69.00%	31.00%	38.0%	0.758
X_4	66.82%	33.18%	33.6%	0.654
X_5	64.56%	35.43%	29.1%	0.488
X_6	62.25%	37.75%	24.5%	0.372

JMP PRO 启动“模拟”窗口

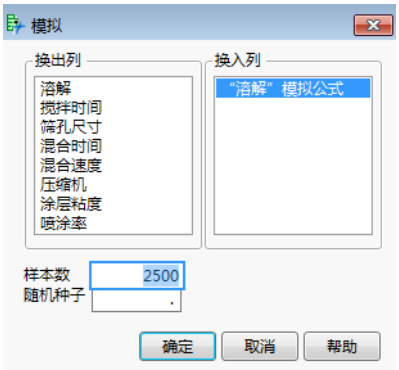
要启动“模拟”平台，右击报表窗口中的某个计算值列，然后选择“模拟”。“模拟”平台在很多报表中都提供，包括支持 bootstrapping 的所有报表。要使用“模拟”平台，数据表必须包含一个公式，公式中包含用于模拟数据的随机成分。

注意：“模拟”平台在使用“依据”变量的报表中不可用。

JMP PRO “模拟”窗口

点击“确定”运行模拟时，将出现一个包含进度条和“提早停止”按钮的窗口。模拟的样本数显示在进度条上方。若点击“提早停止”，计算到这一时点的模拟值将显示在“模拟结果”表中。该窗口还为您显示在任意给定时点正运行哪些分析。

图 10.25 Tablet Production.jmp 的 “模拟”窗口



“模拟”窗口包含以下面板和选项：

换出列 要替换为“换入列”的列。

换入列 用于替换“换出列”的列。使用根据“换入列”中的公式模拟的值反复执行该分析。只有带公式的列列在“换入列”面板中。

样本数 为一组模拟数据重复运行该报表的次数。默认值为 2500。

随机种子 一个用于控制模拟结果的值。随机种子使结果可重现。

JMP PRO “模拟结果”表

模拟结果显示在一个表中。请注意以下事项：

- 该表的第一行包含报表中显示的表项的值。出于此原因，第一行始终被排除。
- 其余行给出模拟结果。其余行数等于您在“模拟”启动窗口中指定的“样本数”。
- 报表中的行通过报表表中包含选定的计算值列的第一列来标识。模拟结果表中会为这个第一列中的每一项显示一列。
- 该表包含用于构造“分布”报表的**分布**脚本。该报表包含模拟结果数据表中每列的直方图、分位数、汇总统计量和模拟结果。除了标准“分布”报表，该报表还包含以下项：
  - 直方图上会显示一条表示原始估计值的红线。
  - “模拟结果”报表，其中包含原始估计值，以及模拟的置信区间和经验 p 值。请参见第 312 页的““模拟结果”报表”。
  - 若模拟结果数据表中的值采用 P 值格式，则还会提供“模拟功效”报表。请参见第 312 页的““模拟功效”报表”。
- 仅当您模拟了 p 值列时，该表才包含**功效分析**脚本。该脚本构造显示 p 值直方图的“分布”报表，并提供“模拟功效”报表。请参见第 312 页的““模拟功效”报表”。

JMP PRO “模拟结果”报表

原始估计值 原始估计值的值也显示在“模拟结果”数据表的第一行中。该估计值标记为 Y_0。

置信区间 以下显著性水平下基于分位数的置信区间的上下限：0.05、0.10、0.20 和 0.50。

经验 p 值 用于将模拟值与原始估计值进行比较的双侧检验和两个单侧检验的经验 p 值。这些 p 值计算为落在报表“检验”列中的指定的范围内的模拟值的比例。

JMP PRO “模拟功效”报表

Alpha 显著性水平：0.01、0.05、0.10 和 0.20。

拒绝计数 在相应的显著性水平下，检验拒绝的模拟数。

拒绝率 在相应的显著性水平下，检验拒绝的模拟比例。

95% 下限和 95% 上限 模拟的拒绝率的 95% 置信区间的上下限。使用 Wilson 得分方法计算该区间。请参见 Wilson (1927)。

提示：增加较窄置信区间的样本数。

第 11 章

JMP PRO Bootstrapping

通过再抽样近似统计量的分布

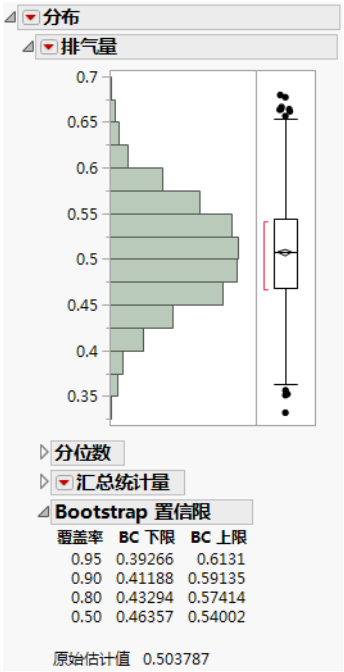
Bootstrapping 仅在 JMP Pro 中提供。

Bootstrapping 是近似统计量抽样分布的再抽样方法。您可以使用 bootstrapping 估计统计量分布及其属性，比如其均值、偏倚、标准误差和置信区间。Bootstrapping 在以下情形下特别有用：

- 统计量的理论分布很复杂或未知。
- 由于假设不成立，无法使用参数方法进行推论。

注意：只有在报表中右击时，Bootstrap 才可用。这不是一个平台命令。

图 11.1 斜率参数的 Bootstrapping 结果



目录

Bootstrapping 概述 317
支持 Bootstrapping 的平台 317
Bootstrapping 示例 318
“Bootstrapping”窗口选项 320
“堆叠的 Bootstrap 结果”表 322
“未堆叠的 Bootstrap 结果”表 323
Bootstrap 结果的分析 323
更多 Bootstrapping 示例 324
Bootstrapping 的统计详细信息 329
计算分式权重 329
修正偏倚的百分位数区间 329

JMP PRO Bootstrapping 概述

Bootstrapping 对您报表中使用的观测反复再抽样，以便构造一个或多个统计量的分布的估计值。假定这些观测均为独立观测。

在简单 bootstrap 中，对 *n* 个观测进行放回再抽样，以生成大小为 *n* 的 bootstrap 样本。请注意，某些观测可能不出现在 bootstrap 样本中，而其他观测则可能多次出现。某个观测出现在 bootstrap 样本中的次数被称为它的 **bootstrap 权重**。对于每次 bootstrap 迭代，生成相关统计量的整个分析将使用以下更改重新运行：

- 包含 *n* 个观测的 bootstrap 样本是数据集
- bootstrap 权重是分析平台中的频数变量

将重复执行该过程以生成一个或多个关注的统计量的值分布。

不过，简单 bootstrap 有时可能有不足之处。例如，假定您的数据集较小或是您在 Logistic 回归设置中可能遇到分离问题。在这些情况下，JMP 支持您使用分式权重执行 Bayes bootstrapping。在使用分式权重时，每个观测都关联一个分式权重。各个分式权重加总为 *n*。通过将分式权重视为分析平台中的频数变量来计算相关统计量。有关分式权重的信息，请参见第 321 页的“分式权重”和第 329 页的“计算分式权重”。

要运行报表中的 bootstrap 分析，请右击包含您要 bootstrap 的统计量的表列，然后选择“Bootstrap”。

注意：只有在报表中右击时，Bootstrap 才可用。这不是一个平台命令。

JMP 在许多统计平台上都提供 bootstrapping。有关完整列表，请参见第 317 页的“支持 Bootstrapping 的平台”。构成样本的观测即用在相关统计量计算中的所有观测。若报表使用频数列，该列中的观测将被视为就像是被重复了“频数”变量所指示的次数一样。若报表使用“权重”变量，则 Bootstrap 会像在报表计算中对待“权重”变量一样对待该变量。

提示：Bootstrap 重新运行在调用 Bootstrap 的平台报表中显示的整个分析。报表中的额外分析可能会导致针对您选定的列的 Bootstrap 运行较慢。若 Bootstrap 运行较慢，请先从平台报表中删除额外选项，然后再运行 Bootstrap。

支持 Bootstrapping 的平台

Bootstrapping 在以下统计平台中提供：

- 提升树
- Bootstrap 森林法
- 分类
- 破坏性退化

- 判别
- 分布
- 拟合曲线
- 以 X 拟合寿命
- 拟合参数生存
- 拟合比例风险
- 以 X 拟合 Y
- 广义线性模型
- 广义回归
- 寿命分布
- Logistic
- 对数线性方差
- 多重对应分析
- 多元
- 神经
- 非线性
- 参数生存
- 偏最小二乘
- 分割
- 主成分
- 比例风险
- 标准最小二乘法
- 生存
- 提升

JMP PRO Bootstrapping 示例

该示例使用 Car Physical Data.jmp 样本数据表。某轮胎制造商要从发动机的排气量（英寸^3）预测发动机的马力。该公司最关注估计两个变量之间关系的斜率。斜率值帮助公司预测在排气量发生变化时马力的相应变化。

在本例中，由于方差齐性的回归假设不成立，所以斜率回归分析中的置信限可能产生错误结果。出于此原因，该公司使用斜率置信区间的 bootstrap 估计值。

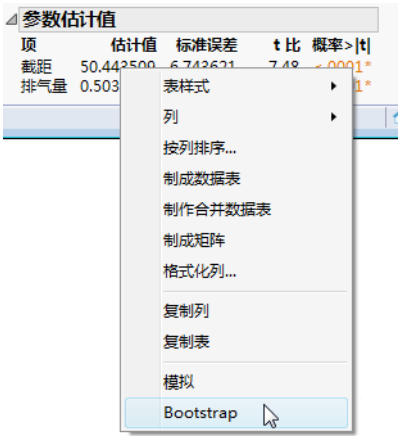
1. 选择**帮助** > **样本数据库**，然后打开 Car Physical Data.jmp。

2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**马力**并点击 **Y，响应**。
4. 选择**排气量**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “二元拟合，以 ‘排气量’ 拟合 ‘马力’” 旁边的红色小三角，然后选择**拟合线**。

   斜率估计值为 0.503787，近似 0.504。
7. （可选）在**参数估计值**报表中右击，然后选择**列 > 95% 下限**。
8. （可选）在**参数估计值**报表中右击，然后选择**列 > 95% 上限**。

   从回归分析得到的斜率置信限分别为 0.4249038 和 0.5826711。
9. 右击**参数估计值**报表中的**估计值**列并选择 **Bootstrap**。

图 11.2 Bootstrap 选项



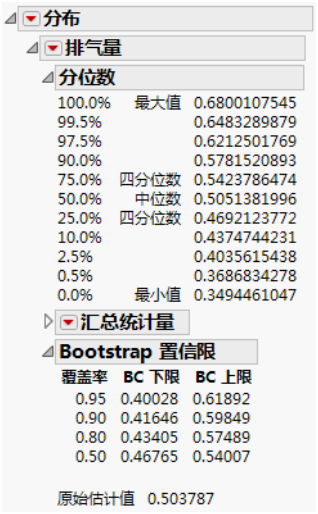
   拆分选定列选项选定时，右击的列是相关的。请参见第 320 页的 “ “Bootstrapping” 窗口选项”。
10. 键入 1000 作为 **Bootstrap 样本数**的值。
11. （可选）要匹配图 11.3 中的结果，请为**随机种子**键入 12345。
12. 点击**确定**。

    Bootstrap 过程运行并生成一个 “Bootstrap 结果” 数据表，其中包含斜率和截距的未堆叠结果。

    接下来，分析 Bootstrap 斜率。
13. 在 “Bootstrap 结果” 表中，运行**分布**脚本。

    “分布” 报表包括 “Bootstrap 置信限” 报表。

图 11.3 Bootstrap 报表



斜率的估计值（第 6 步）为 0.504。根据 Bootstrap 结果中的 95% 覆盖率，该公司可以估计出斜率介于 0.40028 和 0.61892 之间。排气量变化一个单位时，在 95% 的置信区间内，马力变化量介于 0.40028 和 0.61892 之间。该斜率的 bootstrap 置信区间（0.400 到 0.619）比使用第 7 步和第 8 步中的常用回归假设获得的置信区间（0.425 到 0.583）略宽一些。

注意：“Bootstrap 置信限”报表中的“BC 下限”和“BC 上限”列为**修正偏倚的区间**。请参见第 329 页的“修正偏倚的百分位数区间”。

JMP PRO “Bootstrapping”窗口选项

要执行 bootstrap 分析，在报表窗口中右击某表中样本统计量的某个数值列，然后选择 Bootstrap。选定的列将突出显示，并且同时显示“Bootstrapping”窗口。选择选项并在“Bootstrapping”窗口中点击**确定**后，该列中每个统计量的 bootstrap 结果都会显示在默认结果表中。

注意：Bootstrap 选项在使用“依据”变量的报表中不可用。

“Bootstrapping”窗口包含以下选项：

Bootstrap 样本数 设置您要对数据再抽样并计算统计量的次数。值越大，得到的统计量属性的估计值越精确。默认情况下， bootstrap 样本数设置为 2500。

随机种子 设置您可以在随后的 bootstrap 分析运行中重新输入的随机种子，以重复您当前的结果。默认情况下，未设置任何种子。

分式权重 执行 Bayes bootstrap 分析。在每次 bootstrap 迭代中，每个观测都分配有一个如第 329 页的“计算分式权重”中所述计算的权重。加权的观测用于计算相关统计量。默认情况下，未选定“分式权重”选项，执行的是简单 bootstrap 分析。

提示：若分析中使用的观测数很小或您关注 Logistic 回归设置中的分离问题，请使用“分式权重”选项。

假设选定“分式权重”选项。对于每次 Bootstrap 迭代，为报表中使用的每个观测分配一个非零权重。这些权重的和为 *n*，即相关统计量计算中使用的观测数。有关如何计算并使用权重的详细信息，请参见第 329 页的“计算分式权重”。

拆分选定列 将您选定用于 bootstrapping 的列中每个统计量的 bootstrap 结果放入“Bootstrap 结果”表中的**单独**列。“Bootstrap 结果”表中的每行（首行除外）都对应一个 bootstrap 样本。

若取消选定该选项，则会显示一个“堆叠的 Bootstrap 结果”表。对于每次 bootstrap 迭代，该表都包含您选定用于 bootstrapping 的列所在的整个报表表的结果。报表表中每行的结果都显示为“堆叠的 Bootstrap 结果”表中的行。报表表中的每列都定义“堆叠的 Bootstrap 结果”表中的一列。有关示例，请参见第 322 页的“堆叠的 Bootstrap 结果”表”。

若拆分有效，放弃堆叠表 （仅在选定**拆分选定列**选项时适用。）确定 Bootstrap 生成的结果表的数量。

若**未**选定“若拆分有效，放弃堆叠表”选项，则显示以下两个 Bootstrap 表：

- “堆叠的 Bootstrap 结果”表，其中包含您选定用于 bootstrapping 的列所在的表的每行的 bootstrap 结果。该表提供报表中每个统计量的 bootstrap 结果，其中每列都由一个统计量来定义。
- “未堆叠的 Bootstrap 结果”表，通过拆分堆叠表获得。该表仅为原始报表中选定的列提供结果。

若**确实**选定了“若拆分有效，放弃堆叠表”选项，并且**拆分选定列**操作成功，则不显示“堆叠的 Bootstrap 结果”表。

JMP PRO “堆叠的 Bootstrap 结果”表

Bootstrap 分析的初始结果显示在堆叠的结果表中（图 11.4）。若您选择了**若拆分有效，放弃堆叠表**选项，可能不显示该表。图 11.4 显示了一个 bootstrap 表，该表基于针对 Car Physical Data.jmp 在“以 X 拟合 Y”中拟合二元模型获得的“参数估计值”报表。请参见第 317 页的“Bootstrapping 概述”。

图 11.4 “堆叠的 Bootstrap 结果”表

	X	Y	项	~偏倚	估计值	标准误差	t 比	概率>|t|	BootID•
1	排气量	马力	截距		50.443509471	6.7436209999	7.48	<.0001	0
2	排气量	马力	排气量		0.5037874592	0.0398202611	12.65	<.0001	0
3	排气量	马力	截距		45.385604758	5.4724273046	8.29	<.0001	1
4	排气量	马力	排气量		0.5451674092	0.0306151772	17.81	<.0001	1
5	排气量	马力	截距		40.843862813	7.0508187326	5.79	<.0001	2
6	排气量	马力	排气量		0.5854988173	0.0457041828	12.81	<.0001	2
7	排气量	马力	截距		47.765642104	5.4677610087	8.74	<.0001	3
8	排气量	马力	排气量		0.4943459579	0.0305765677	16.17	<.0001	3
9	排气量	马力	截距		59.758060908	7.0436720036	8.48	<.0001	4
10	排气量	马力	排气量		0.4385540785	0.042053201	10.43	<.0001	4
11	排气量	马力	截距		66.341062413	6.4663383596	10.26	<.0001	5
12	排气量	马力	排气量		0.3969640071	0.0393100058	10.10	<.0001	5
13	排气量	马力	截距		41.234989734	5.8543901826	7.04	<.0001	6
14	排气量	马力	排气量		0.5723548142	0.034172131	16.75	<.0001	6
15	排气量	马力	截距		43.876867815	7.8085946052	5.62	<.0001	7
16	排气量	马力	排气量		0.5731377467	0.0439823115	13.03	<.0001	7

请注意有关堆叠的结果表的以下情况：

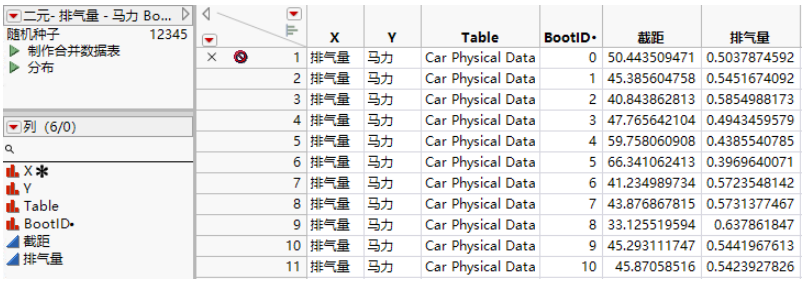
- 对于每个 bootstrap 样本，报表表的第一列中给出的每个值都对应一行。这些值显示在与报表表中第一列同名的列中。在本例中，对于每个 bootstrap 样本，都有一行包含每个**项**：“截距”和“排气量”（显示在**项**列中）的结果。
- 分析中使用的数据表列显示在该表中。在本例中，X 是**排气量**，Y 是**马力**。
- 您执行 bootstrapping 的报表表中的每一列都对应一列。在本例中，这些列包括 **~偏倚**、**估计值**、**标准误差**、**t 比**和**概率>|t|**。请注意，**~偏倚**是“以 X 拟合 Y”报表中的列，除非其中一个参数估计值有偏倚，否则该列将被隐藏。
- BootID•列标识 Bootstrap 样本。BootID•= 0 的行对应于原始估计值。这些行使用 X 标记并具有已排除行状态。在本例中，每个 bootstrap 样本都用于计算两行的结果：**截距**的结果和**排气量**的结果。
- 数据表名称以“堆叠的 Bootstrap 结果”开头。

若您选择了**拆分选定列**选项，也可能会显示未堆叠的结果表。请参见第 323 页的““未堆叠的 Bootstrap 结果”表”。

JMP PRO “未堆叠的 Bootstrap 结果”表

选择**拆分选定列**以创建包含用于您选定的报表列的单独列的 bootstrap 表。每列都对应报表表中的一个“项”。例如，在图 11.5 中，将图 11.4 中的**估计值**列拆分为两个列（**排气量**和**截距**），分别对应**项**的每个水平。

图 11.5 “未堆叠的 Bootstrap 结果”表



	X	Y	Table	BootID	截距	排气量
1	排气量	马力	Car Physical Data	0	50.443509471	0.5037874592
2	排气量	马力	Car Physical Data	1	45.385604758	0.5451674092
3	排气量	马力	Car Physical Data	2	40.843862813	0.5854988173
4	排气量	马力	Car Physical Data	3	47.765642104	0.4943459579
5	排气量	马力	Car Physical Data	4	59.758060908	0.4385540785
6	排气量	马力	Car Physical Data	5	66.341062413	0.3969640071
7	排气量	马力	Car Physical Data	6	41.234989734	0.5723548142
8	排气量	马力	Car Physical Data	7	43.876867815	0.5731377467
9	排气量	马力	Car Physical Data	8	33.125519594	0.637861847
10	排气量	马力	Car Physical Data	9	45.293111747	0.5441967613
11	排气量	马力	Car Physical Data	10	45.87058516	0.5423927826

请注意有关未堆叠的结果表的以下情况：

- 每个 bootstrap 样本都对应一行。
- 分析中使用的数据表列显示在表中。在本例中，X 是**排气量**，Y 是**马力**。
- 报表中进行 bootstrap 的每行都对应一列。
- 若在“Bootstrapping”窗口中指定了“随机种子”，bootstrap 结果表将包含称为“随机种子”的表变量，用来提供其值。
- 未堆叠的 bootstrap 结果表包含一个“源”表脚本和一个“分布”表脚本。“分布”表脚本支持您基于 bootstrap 样本快速获取统计量，包括 bootstrap 置信区间。
- BootID•列标识 Bootstrap 样本。BootID•= 0 的行对应于原始估计值。该行使用 X 标记并具有已排除行状态。在未堆叠的 bootstrap 表中，基于单个 bootstrap 样本计算每行。
- 数据表名称以“Bootstrap 结果 (<列名>)”结尾，其中 <列名> 标识报表中进行 bootstrap 的列。

JMP PRO Bootstrap 结果的分析

使用 “分布”平台分析 bootstrap 结果：

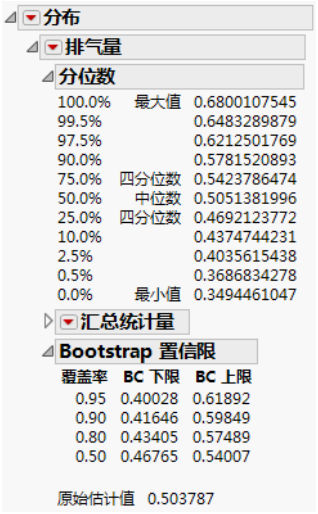
- 若您的分析生成了未堆叠的 bootstrap 结果表，请运行表中的“分布”脚本。
- 若您的分析生成了堆叠的 Bootstrap 结果表，则选择**分析 > 分布**，并将关注的列分配给相应的角色。多数情况下，应该将对应于报表表中第一列的列分配给“依据”角色。

“分布”平台提供您的 bootstrap 结果的汇总统计量。它还为包含 BootID 列的所有表生成“Bootstrap 置信限”报表（图 11.6）。

您可以使用“分布”报表获取两类 bootstrap 置信区间：

- “分位数”报表提供**百分位数区间**。例如，要使用百分位数方法构造 95% 置信区间，请将 2.5% 和 97.5% 分位数用作区间边界。
- “Bootstrap 置信限”报表提供**修正偏倚的百分位数区间**。该报表显示覆盖率为 95%、90%、80% 和 50% 的区间。“BC 下限”列和“BC 上限”列分别显示下端点和上端点。有关计算修正偏倚的百分位数区间的详细信息，请参见第 329 页的“修正偏倚的百分位数区间”。

图 11.6 “Bootstrap 置信限”报表


分布

排气量

分位数

100.0%	最大值	0.6800107545
99.5%		0.6483289879
97.5%		0.6212501769
90.0%		0.5781520893
75.0%	四分位数	0.5423786474
50.0%	中位数	0.5051381996
25.0%	四分位数	0.4692123772
10.0%		0.4374744231
2.5%		0.4035615438
0.5%		0.3686834278
0.0%	最小值	0.3494461047

汇总统计量

Bootstrap 置信限

覆盖率	BC 下限	BC 上限
0.95	0.40028	0.61892
0.90	0.41646	0.59849
0.80	0.43405	0.57489
0.50	0.46765	0.54007

原始估计值 0.503787

“Bootstrap 置信限”报表底部的**原始估计值**是使用原始数据的统计量估计值。

有关解释“Bootstrap 置信限”报表的详细信息，请参见第 317 页的“Bootstrapping 概述”。Efron (1981) 对百分位数区间方法和修正偏倚的百分位数区间方法都进行了说明。

JMP PRO 更多 Bootstrapping 示例

本例演示针对一个小型数据表的“分式权重”(Bayes Bootstrap) 选项的优点。数据中包含一个响应 Y，该响应是针对七种不同土壤类型中的每一种的三个样本测量的结果。一位科学家想要查明 wabash 土壤类型的响应均值的置信区间。

由于每种土壤类型仅有三个观测，简单 bootstrap 有可能从 bootstrap 样本中排除全部三个 wabash 观测。而“分式权重”选项可确保在所有 bootstrap 样本中都能表示每种土壤类型的所有观测。

这位科学家检查了使用两种 bootstrap 方法得到的 wabash 样本均值的分布：

- 第 325 页的“简单 Bootstrap 分析”
- 第 327 页的“Bayes Bootstrap 分析”

简单 Bootstrap 分析

1. 选择**帮助** > **样本数据库**，然后打开 Snapdragon.jmp。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择 **Y** 并点击 **Y，响应**。
4. 选择**土壤**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “‘土壤 -Y’ 单因子分析”旁边的红色小三角并选择**均值 / 方差分析**。
7. 在**单因子方差分析均值**报表中，右击**均值**列并选择 **Bootstrap**。
8. 键入 1000 作为 **Bootstrap 样本数**的值。
9. （可选）要匹配图 11.7 中的结果，请为**随机种子**键入 12345。
10. 点击**确定**。

图 11.7 简单 Bootstrap 的 Bootstrap 结果

	X	Y	BootID	clarion	clinton	compost	knox	o'neill	wabash	webster
1	土壤	Y	0	32.1667	30.3000	29.6667	34.9000	33.8000	35.9667	31.1000
2	土壤	Y	1	32.2333	30.9000	28.0000	35.7000	34.3500	31.9000	31.1000
3	土壤	Y	2	32.4333	30.3000	30.2000	33.9667	31.2000	38.0000	•
4	土壤	Y	3	32.5000	30.7500	29.2000	34.0333	32.8000	38.0000	31.1000
5	土壤	Y	4	31.5000	29.4000	29.9000	35.7500	35.0400	34.9500	•
6	土壤	Y	5	32.5000	30.6000	31.8000	35.7000	34.3000	35.0500	31.5667
7	土壤	Y	6	32.4000	30.1000	29.8500	34.4500	36.0000	38.2000	30.4000
8	土壤	Y	7	31.9000	29.3400	•	34.4000	33.6000	35.9667	31.4500
9	土壤	Y	8	32.1400	31.3500	29.6667	33.1000	35.1000	37.9333	30.6333
10	土壤	Y	9	32.7000	30.8000	30.2000	35.2600	31.2000	34.8500	32.5000
11	土壤	Y	10	32.1000	32.1000	28.0000	33.1000	34.1143	34.0000	31.8000
12	土壤	Y	11	31.5000	30.3000	•	33.1000	34.4000	37.2125	30.6333
13	土壤	Y	12	32.7000	30.4200	31.8000	34.0333	35.1000	35.0500	•
14	土壤	Y	13	32.1667	•	30.1000	34.9000	33.9000	38.2000	31.8000
15	土壤	Y	14	32.3000	•	29.2000	33.9667	34.4000	35.6000	31.9400
16	土壤	Y	15	32.2500	29.1000	29.6667	33.1000	35.1000	31.9000	•
17	土壤	Y	16	31.5000	30.4200	30.0000	•	35.2800	37.8000	31.1000
18	土壤	Y	17	32.4333	29.7000	29.2000	33.1000	34.4000	34.0000	31.1000
19	土壤	Y	18	32.3800	32.1000	29.9000	35.8000	34.6800	35.0500	31.8000
20	土壤	Y	19	32.0600	30.1000	30.2800	•	33.1500	31.9000	31.1000
21	土壤	Y	20	31.9000	29.1000	28.4000	35.9000	34.8000	34.8500	30.8200

图 11.7 中的缺失值表示 bootstrap 迭代，在这些迭代中，给定土壤类型的所有观测都未选定用于 bootstrap 样本。

11. 选择**分析** > **分布**。
12. 选择 **wabash**，然后点击 **Y，列**。
13. 点击**确定**。

图 11.8 简单 Bootstrap 的 wabash 均值分布

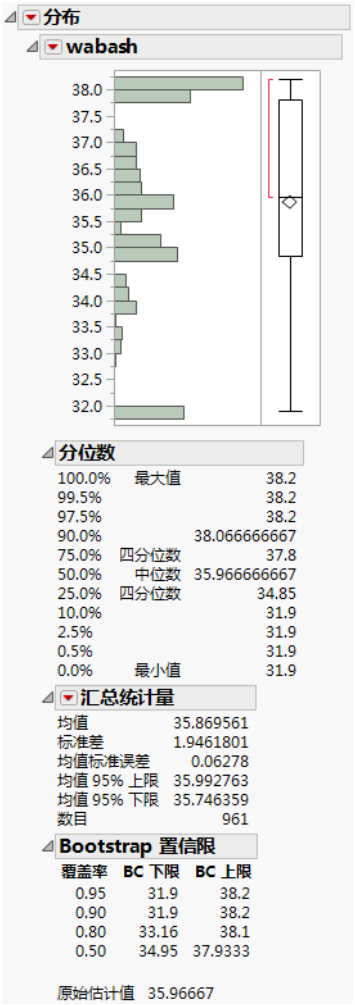


图 11.8 显示来自简单 bootstrap 分析的 wabash 均值的分布。请注意以下事项：

- “汇总统计量”报表指示包含 wabash 的 bootstrap 均值的行数为 N = 961。尽管您执行了 1,000 次迭代，有 39 个 bootstrap 样本未包含 wabash 三个观测中的任何一个观测。
- 样本均值的直方图不平滑，在两个极值处出现了峰值。wabash 的三个值为 38.2、37.8 和 31.9。分布低端出现的峰值来自仅包含值 31.9 的 bootstrap 样本。高端出现的峰值来自包含 38.2 和 37.8 中的一个或两个值的 bootstrap 样本。

下一步，使用“分式权重”(Bayes Bootstrap) 选项避免获取 bootstrap 样本中的缺失值，并且平滑 bootstrapped 均值分布。

Bayes Bootstrap 分析

1. 在“单因子分析”报表中，右击**单因子方差分析均值**报表中的**均值**列，然后选择 Bootstrap。
2. 键入 1000 作为 **Bootstrap 样本数**的值。
3. （可选）要匹配图 11.9 中的结果，请为**随机种子**键入 12345。
4. 选择**分式权重**选项。
5. 点击**确定**。

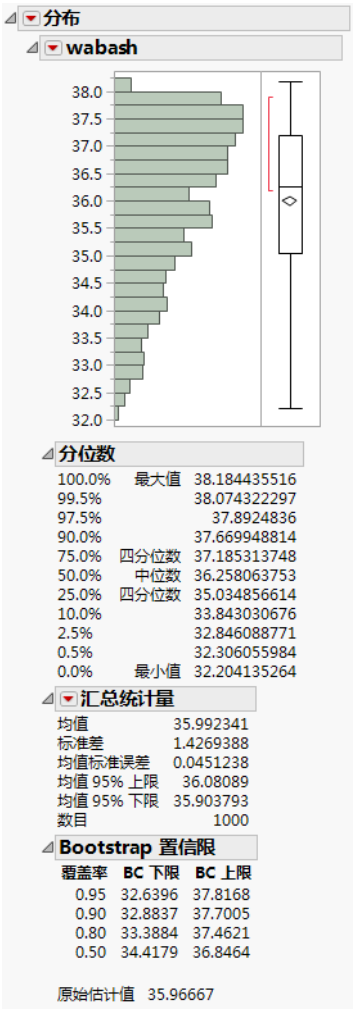
图 11.9 Bayes Bootstrap 的 Bootstrap 结果

	X	Y	BootID	clarion	clinton	compost	knox	o'neill	wabash	webster
1	土壤	Y	0	32.1667	30.3000	29.6667	34.9000	33.8000	35.9667	31.1000
2	土壤	Y	1	31.9365	31.3493	29.3234	35.4497	34.7105	33.7270	31.4071
3	土壤	Y	2	32.4189	30.3474	29.5143	35.6281	32.7006	34.1027	32.1212
4	土壤	Y	3	32.2339	30.2001	31.4102	34.8758	33.6674	38.0389	31.5428
5	土壤	Y	4	32.4054	30.3242	30.6227	33.8495	32.9495	36.8344	31.9607
6	土壤	Y	5	32.2262	30.8672	29.7999	33.6759	32.2022	35.5792	31.9058
7	土壤	Y	6	32.3222	31.9732	28.8823	35.6307	34.8863	35.3014	30.1325
8	土壤	Y	7	31.9948	30.8828	29.2516	35.3424	33.2094	36.3367	30.8386
9	土壤	Y	8	31.6254	29.7677	28.6390	34.4697	33.0662	36.6183	31.4667
10	土壤	Y	9	32.3499	29.9416	29.5732	35.2564	31.9583	35.8246	29.9302
11	土壤	Y	10	32.5228	30.4506	28.4859	34.9088	35.8126	34.6317	31.6100
12	土壤	Y	11	31.9057	30.0711	29.0693	35.4018	34.4654	33.2086	31.0309
13	土壤	Y	12	31.7275	29.6189	29.1609	34.3984	33.6840	35.0815	31.5446
14	土壤	Y	13	32.5893	30.5210	28.6054	33.4594	34.0958	33.7692	31.7129
15	土壤	Y	14	32.2473	30.5199	31.6080	35.5617	34.1706	35.7146	31.8023
16	土壤	Y	15	32.2329	29.7275	30.1770	35.3286	32.7820	37.4866	30.9706
17	土壤	Y	16	31.5831	29.7508	28.7122	33.5304	34.5348	37.1092	31.2752
18	土壤	Y	17	32.3545	31.3237	29.1542	35.5890	32.2606	37.2005	30.8430
19	土壤	Y	18	32.3811	29.6241	30.6138	35.4308	33.2024	33.0787	31.2926
20	土壤	Y	19	31.7488	29.7763	28.7327	34.7007	33.8910	34.0573	29.9064

Bayes Bootstrap 结果表中没有缺失值。每个 bootstrap 样本中都包含 Snapdragon.jmp 数据表中的全部 21 行，这些行的 bootstrap 权重各异。

6. 选择**分析 > 分布**。
7. 选择 wabash，然后点击 **Y，列**。
8. 点击**确定**。

图 11.10 Bayes Bootstrap 的 wabash 均值分布



Bayes Bootstrap 为 wabash 样本均值生成了平滑得多的分布。全部 1,000 个 bootstrap 样本都包含 wabash 的三个观测。对于每次迭代， wabash 样本均值都使用不同的分式权重计算。

“Bootstrap 置信限”报表显示均值的 95% 置信区间介于 32.6396 到 37.8168 之间。

JMP PRO Bootstrapping 的统计详细信息

JMP PRO 计算分式权重

“分式权重”选项基于 Bayes Bootstrap (Rubin 1981)。某个观测出现在给定 bootstrap 样本中的次数被称为它的 **bootstrap 权重**。在简单 bootstrap 中，每个 bootstrap 样本的 bootstrap 权重都使用有放回的简单随机抽样确定。

在 Bayes 方法中，抽样概率被视为未知参数，使用非信息性先验获取其后验分布。通过从这一后验分布抽样来获取概率估计值。这些估计值用于构造 bootstrap 权重：

- 从形状参数等于 $(n - 1)/n$ 且尺度参数等于 1 的 gamma 分布随机生成 n 个值的向量。

注意：Rubin (1981) 将 1 用作 gamma 形状参数。 JMP Pro 中使用的形状参数可确保分式权重的均值和方差等于简单 bootstrap 权重的均值和方差。

- 将 S 计算为 n 个值的总和。
- 通过将 n 个值的向量乘以 N / S 计算分式权重，其中 N 等于行数或频数总和（若指定了“频数”变量）。

注意：若为分析指定了“频数”变量，则逐行将 gamma 分布的形状参数乘以“频数”值。频数变量的值的总和必须大于 1。因此，形状参数等于 $f_i(N - 1)/N$，其中 f_i 是第 i 行的“频数”值，N 等于“频数”值总和。

该过程调整每行的分式权重的尺度，使其 bootstrap 抽样的均值和方差等于简单 bootstrap 权重的均值和方差。每个 bootstrap 样本中的分式 bootstrap 权重为正数，加总为 N，均值为 1。

JMP PRO 修正偏倚的百分位数区间

本节说明当您运行“Bootstrap 结果”表中的“分布”脚本时，如何计算出现在“Bootstrap 置信限”报表中的修正偏倚的 (BC) 置信区间。修正偏倚的百分位数区间改进了百分位数区间在解释 bootstrap 分布的不对称性方面的能力。请参见 Efron (1981)。

符号

- p^* 是相关统计量的估计值小于等于原始估计值的 bootstrap 样本的比例。
- z_0 是标准正态分布的 p^* 分位数。
- z_α 是标准正态分布的 α 分位数。

修正偏倚的置信区间端点

$(1 - \alpha)$ 修正偏倚的置信区间的端点由 bootstrap 分布的分位数提供：

- 下端点是以下分位数：

$$\Phi\left(2z_0 + z_{\frac{\alpha}{2}}\right)$$

- 上端点是以下分位数：

$$\Phi\left(2z_0 + z_{1-\frac{\alpha}{2}}\right)$$

第 12 章

文本分析器

探索数据中的非结构化文本

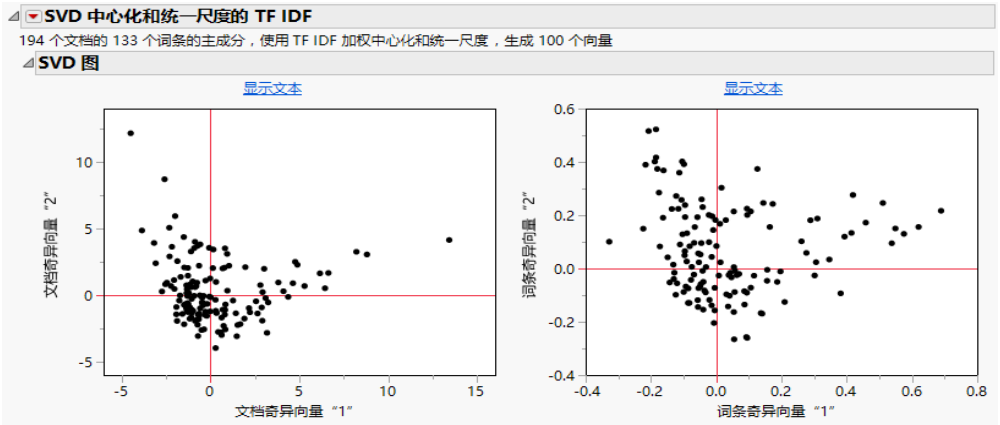
JMP PRO 该平台中的很多功能只能在 JMP Pro 中使用，并采用该图标进行标注。

通过“文本分析器”平台，您可以分析非结构化文本，如调查或事故报告中的备注字段。通过使用工具与文本数据交互，来合并类似词条、重新编码错误指定的词条以及了解文本数据中的基本模式。

JMP PRO 该平台的 JMP Pro 版本包含一些分析工具，它们使用奇异值分解 (SVD) 来将类似文档按主题分组。您可以对文本文档聚类或对处于一个文档集合中的词条聚类。您还可以使用潜在类分析对文档聚类。

JMP PRO 该平台的 JMP Pro 版本还包含用于识别文档中重要词条和情感的工具。词条选择支持您识别最适合解释不同响应的词条。情感分析支持您使用词法分析识别文档中的情感词条，并对文档的正面、负面和整体情感进行评分。情感分析功能包括基本的自然语言处理 (NLP) 支持。

图 12.1 文本分析器中的 SVD 图



目录

“文本分析器”平台概述 333
文本处理步骤 334
“文本分析器”平台的示例 335
启动“文本分析器”平台 338
在“正则表达式编辑器”中定制 Regex 340
文本分析器报表 345
“汇总计数”报表 345
词条和短语列表 346
“文本分析器”平台选项 349
文本准备选项 349
文本分析选项 354
保存选项 355
报表选项 357
潜在类分析 357
潜在语义分析 (SVD) 358
SVD 报表 359
SVD 报表选项 359
主题分析 361
“主题分析”报表 361
“主题分析”报表选项 362
判别分析 363
“判别分析”报表 363
“判别分析”报表选项 364
词条选择 364
词条选择设置 364
“词条选择”报表 365
“词条选择”报表选项 367
情感分析 367
“情感分析”报表 368
“情感分析”报表选项 370
“文本分析器”平台的其他示例 371

“文本分析器”平台概述

通过“文本分析器”平台，您可以探索非结构化文本以更好地了解它的含义。非结构化文本数据很常见。例如，它们可能来自调查中的自由回答字段、产品评论或事故报告。

文本分析通常是一个迭代过程，因此您可能要交替进行词条列表的调理和分析。

审校词条列表

文本分析使用一些独特的术语。一个**词条**或**标记**是最小的文本片段，类似于句子中的单词。但是，您可以使用很多方式来定义词条，包括使用正则表达式；将文本拆分为词条的过程称为**标记化**。

- **短语**是词条的简短集合；平台提供管理短语的选项，短语的指定方式可以与词条相同，也可以有自己独特的方式。
- **文档**是指单词的集合；在 JMP 数据表中，文本列的每行中的非结构化文本对应于一个文档。
- **语料库**是指文档的集合。

通常希望从分析中排除一些常见单词。这些被排除的单词称为**停止词**。平台提供停止词的默认列表，但是您也可以将特定单词添加为停止词。尽管停止词不能是词条，但是可以在短语中使用它们。

您还可以对词条重新编码；这对将同义词合并为一个普通词条很有用。

词干处理是指通过删除不同的结尾部分，使用相同开头（**词干**）合并单词的过程。这样“jump”、“jumped”和“jumping”都将视为词条“jump”。词干处理步骤类似于在 Snowball 字符串处理语言中使用的步骤。对短语进行词干处理时，将短语中的每个单词作为独立的词条来进行词干处理。

分析词条列表

“文本分析器”平台中的文本分析使用**词袋**方法。与短语形成过程不同，词条的顺序这里被忽略。该分析基于词条计数。

当您使用正则表达式、停止词、重新编码和词干处理审校词条列表后，可以对审校后的词条列表进行分析。平台中的分析选项基于**文档词条矩阵** (DTM)。DTM 中的每行对应于一个文档（JMP 数据表的文本列中的一个单元格）。DTM 中的每列对应于审校后的词条列表中的一个词条。该方法实施词袋方法，因为它忽略单词顺序。在它最简单的形式中，DTM 的每个单元格包含该列的词条在该行的文档中的频数（出现次数）。还有其他很多针对 DTM 的权重方案；相关信息请参见第 355 页的“保存选项”。

JMP PRO 平台中可用的分析选项首先对文档词条矩阵执行奇异值分解 (SVD)。这可以大大减少表示数据中的词条信息所需的列数。有关奇异值分解的详细信息，请参见《多元方法》。“层次聚类”选项可用于对词条和文档聚类。使用这些选项，您可以将类似词条或文档分在一组。

“文本分析器”平台工作流

以下是使用 “文本分析器”平台的预期步骤：

1. 指定标记化的方法（内置或定制的正则表达式）。
2. 使用报表指定其他停止词，将短语添加到词条列表，执行词条的重新编码以及指定词干处理规则的例外情况。
3. 指定词干处理的首选项。
4. 使用单词和短语计数、 SVD 和聚类方法来确定重要词条和短语。

注意： JMP PRO SVD 和聚类选项仅在 JMP Pro 中可用。

5. 保存结果以供进一步分析：词条表、 DTM、奇异向量或其他结果。

注意： JMP PRO 保存奇异向量的选项仅在 JMP Pro 中可用。

6. 保存 “短语”、“重新编码”和 “停止词”属性，以在进一步分析类似文本数据中使用。

文本处理步骤

在 “JMP 文本分析器”平台中，分三个阶段处理文本：标记化、短语化和词条化。

标记化阶段

标记化阶段执行以下操作：

1. 将文本转换为小写形式。
2. 应用标记化方法 （“基本单词”或 “Regex”）来将字符分组为标记。
3. 根据指定的重新编码定义对标记重新编码。请注意在词干处理前进行重新编码。

注意：无论在报表窗口中指定的顺序如何，重新编码操作都是在内部一次性处理。

短语化阶段

短语化阶段收集在语料库（文档集合）中出现的短语并允许您指定将这些短语视为词条。短语不能以停止词开头或结尾，但是可以包含停止词。

词条化阶段

词条化阶段使用从以前阶段得到的标记和短语创建词条列表。

对于每个标记，词条化阶段执行以下操作：

1. 检查是否满足在启动窗口中指定的最小和最大长度要求。仅包含数字的标记将从该操作中排除。

2. 检查标记是否可以成为词条；按“基本单词”标记化方法解析的标记必须至少包含一个字母字符或 Unicode 字符。仅包含数字的标记将从该操作中排除。“Regex”标记化方法使用正则表达式来确定哪些字符是标记的一部分。
3. 检查确定该标记不是停止词。
4. 应用词干处理和词干例外情况。

对于您添加的每个短语，词条化阶段执行以下操作：

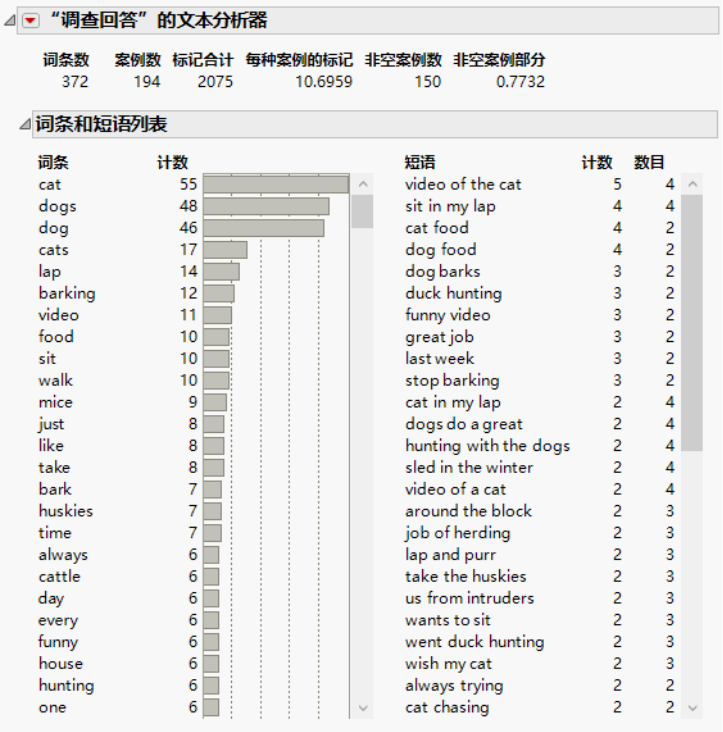
1. 将短语添加到词条列表。短语应将词干处理应用到已在词条列表中进行词干处理的短语中的每个单词。在词条列表中合并具有不同原始标记但有相同词干的短语。
2. 删除在短语中出现的标记词条实例。

“文本分析器”平台的示例

了解如何在 JMP 中探索文本响应。在该示例中，您要探索有关宠物的调查的文本响应。

1. 选择**帮助** > **样本数据库**，然后打开 Pet Survey.jmp。
2. 选择**分析** > **文本分析器**。
3. 选择**调查回答**并点击**文本列**。
4. 从 “语言”列表中，选择**简体中文**。
5. 点击**确定**。

图 12.2 初始文本分析器报表的示例



您一眼可看到 194 个文档中有 372 个独特词条。总共有 2075 个标记化词条。最常见的词条是 “cat”，它出现了 55 次。

6. 点击“‘调查回答’的文本分析器”旁边的红色小三角并选择**词条选项 > 词干处理 > 处理所有词条的词干**。
7. 在 “短语列表”表中，选择 cat food 和 dog food，右击选定内容，然后选择**添加短语**。

   词条 cat food 和 dog food 包含在词条列表中。
8. 在词条列表中向下滚动，找到 cat 和 dog food 条目。

   您可以看到每个短语出现了四次。

图 12.3 修改和滚动后的词条列表

一个词条	计数
cat·food·	4
dog·food·	4
anymor·	3
bath·	3
shake·	3
kid·	3
see·	3
box·	3
stay·	3
roll·	3
fluffi·	3
read·	3
hilari·	3
best·	3
leg·	3

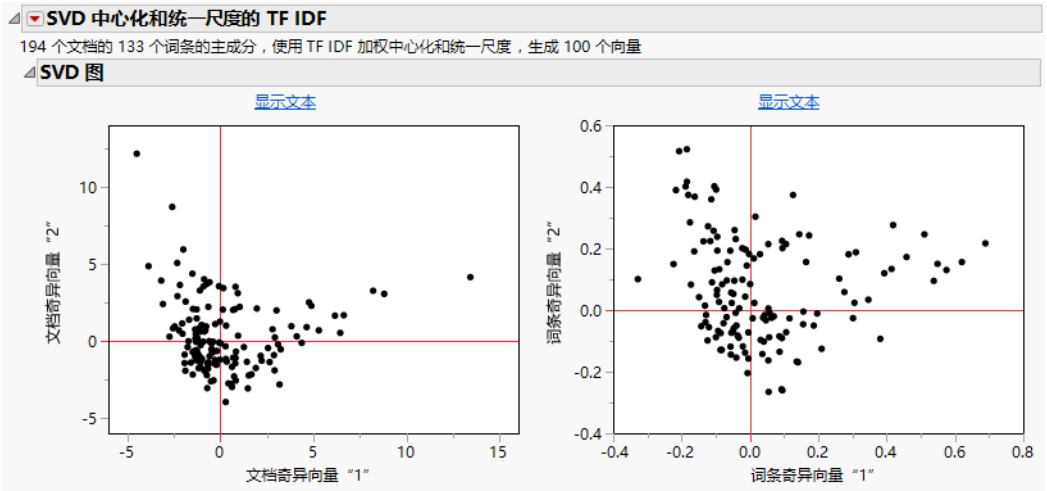
在短语列表中， cat food 和 dog food 是灰色的，因为它们现在在本地被视为该文本分析器报表中的词条。

JMP PRO 本例的其余步骤只能在 JMP Pro 中完成。

9. JMP PRO 点击 “‘调查回答’的文本分析器”旁边的红色小三角并选择**潜在语义分析， SVD**。
10. JMP PRO 点击**确定**以接受默认值。

两个 SVD 图显示在报表中。左侧的图显示文档空间中的前两个奇异向量。右侧的图显示词条空间中的前两个奇异向量。

图 12.4 SVD 图



11. JMP PRO 选择左侧 SVD 图中的三个最右侧的点。

这三个点表示与其他点不聚类在一起的调查回答。为了进一步调查该聚类，您阅读了这些响应的文本。

12. JMP PRO 点击左侧 SVD 图上方的**显示文本**按钮。

图 12.5 选定文档的文本

```
There was this funny video of a cat trying to jump into someones lap, but fell into the pool instead.
[56]

The funny cat video where the cats jumped through the window right into the bathtub was hilarious. [142]

We made this funny video of the cat trying to climb the wall to chase a laser pointer. [153]
```

显示一个窗口，它包含选定点所表示的三个文档的文本。这些调查回答是类似的，它们都具有 “funny”、“cat”和 “video”的某些组合。这些文档与其余文档相比，其第一个奇异向量的正值更大。这些较大值指示在该维上它们与其余文档不同。

奇异向量维的进一步调查可能导致解释这些维表示什么。例如，图的最右侧的很多文档是有关 cat 的响应。在最左边，很多响应是关于 dog 的。因此，第一个奇异向量表示基于响应是关于 cat 还是 dog 的差异。

启动“文本分析器”平台

通过选择**分析 > 文本分析器**来启动 “文本分析器”平台。

图 12.6 “文本分析器”启动窗口

有关 “选择列”红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。“文本分析器”启动窗口包含以下选项：

文本列 分配包含文本数据的列。若您指定多个列，则为每个列创建单独的分析。

JMP PRO **验证** 在 JMP Pro 中，您可以输入一个 “验证”列。若在 “选择列”列表中没有选择任何列的情况下点击 “验证”按钮，您可以向数据表添加一个验证列。有关 “生成验证列”实用工具的详细信息，请参见 《预测和专业建模》。

“验证”列的指定并不影响文档词条矩阵的计算。不过，指定 “验证”列后，只有训练集可用于 “潜在类分析”、“潜在语义分析”、“主题分析”和 “判别分析”选项。“验证”列用作 “词条选择”选项的 “广义回归”验证方法。

ID 分配一个列，用于在“保存用于关联的堆叠 DTM”输出数据表中标识不同响应者。该输出数据表适合进行关联分析。该列还用于在 “潜在类分析”报表中标识不同响应者。

依据 标识用于创建报表的一个列，该报表包括变量的每个水平的单独分析。若指定了多个“依据”变量，将为 “依据”变量水平的每种可能组合生成单独的报表。

注意：若您指定 “依据”变量，则 “定制 Regex”选项和设置适用于 “依据”变量的所有水平。

语言 指定用于文本处理的语言。它影响词干处理和停止词、重新编码以及短语的内置列表。该选项与运行 JMP 的语言无关。除非设置了 “语言”平台首选项，否则根据 JMP “显示语言”首选项设置 “语言”选项。不过，“文本分析器”中的 “语言”选项不支持 “韩语”。若 “JMP 显示语言”为 “韩语”，该选项默认显示 “英语”。

每个短语的最大单词数 指定一个短语作为分析中的短语可以包含的最大单词数。

最大短语数 指定在短语列表中显示的最大短语数。

每个单词的最小字符数 指定一个单词作为分析中的词条必须包含的字符数。

每个单词的最大字符数 指定一个单词作为分析中的词条可以包含的最大字符数（最多 2000）。

词干处理 （仅当 “语言”选项设置为 “英语”、“德语”、“西班牙语”、“法语”或 “意大利语”时才可用。）指定合并具有类似开头字符但是结尾不同的词条的方法。提供了以下选项：

无需词干处理 不合并词条。

要组合的词干 仅处理这些词条的词干：要将两个或更多词条的词干处理为同一词条。

处理所有词条的词干 处理所有词条的词干。

注意：“词干处理”选项的使用还影响已添加到词条列表中的短语。在短语中的词条已进行词干处理后进行短语标识。例如，“dogs bark”和“dog barks”都匹配指定的短语“dog· bark·”。

标记化 （仅当 “语言”选项设置为 “英语”、“德语”、“西班牙语”、“法语”或 “意大利语”时才可用。）指定将文本解析为词条或标记的方法。提供以下标记化选项：

Regex 使用默认的一组内置正则表达式来解析文本。若您要添加、删除或编辑用于解析文本的这组正则表达式，请选择**定制 Regex** 选项。请参见第 340 页的“在“正则表达式编辑器”中定制 Regex”。

基本单词 基于一组通常分隔单词的字符将文本解析为单词。这些字符包括空格、制表符、换行符和大多数标点符号。若您希望将数字解析为供分析的词条，请选择**将数字视为单词**选项。若您不选择该选项，将在标记化步骤中忽略仅包含数字的分隔符之间的文本片段。

提示：您可以在使用 “基本单词”标记化方法的文本分析器报表中使用**显示选项 > 显示分隔符**选项来查看这组默认分隔符。

定制 Regex （仅适用于 Regex 标记化方法。）您可以使用 “文本分析器正则表达式编辑器”窗口修改 Regex 设置。使用该选项来处理非传统单词。示例包括电话号码或由字符和数字组

成的单词。除非默认 Regex 方法无法给出您所需的结果，否则不建议使用“定制 Regex”选项。当您的文本包含默认 Regex 方法无法识别的结构时就会出现这种情况。请参见第 340 页的“在“正则表达式编辑器”中定制 Regex”。

将数字视为单词 （仅适用于“基本单词”标记化方法。）允许数字标记化为分析中的词条。选择该选项时，对于包含数字位的词条，忽略“每个单词的最小字符数”设置。

在启动窗口上点击**确定**后，若您在启动窗口中选择了**定制 Regex**，将显示“文本分析器正则表达式编辑器”窗口。否则，显示文本分析器报表。

注意：对文本输入的处理不区分大小写。在标记化和所有分析步骤之前，所有文本都将在内部转换为小写形式。该转换在文本分析器输出中影响正则表达式的处理和词条的聚合。

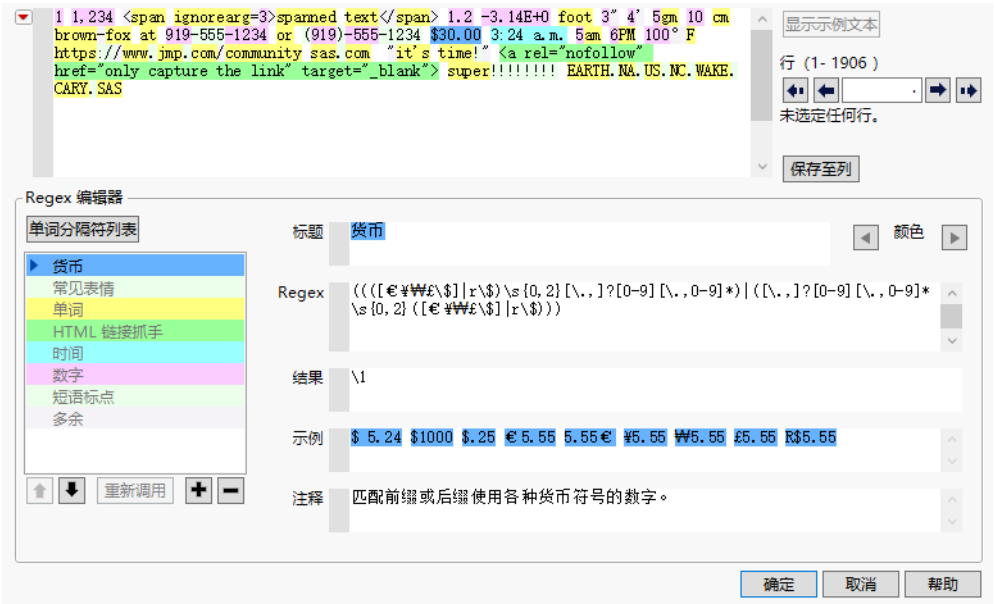
在“正则表达式编辑器”中定制 Regex

当您选择**定制 Regex** 选项时，显示“文本分析器正则表达式编辑器”。在该窗口中，使用各种内置正则表达式（如电话号码、时间或货币值）解析文本文档。您还可以创建自己的正则表达式定义。

注意：仅当您无法从默认 Regex 方法获得所需结果时，才建议使用“定制 Regex”选项。当您的文本包含默认 Regex 方法无法识别的结构时就会出现这种情况。

提示：若在启动窗口中将“日语”、“中文（简体）”或“中文（繁体）”指定为“语言”选项，则 Regex 模式列表会包含指定语言的单个 Regex。若要添加其他 Regex 模式，建议您在单个 Regex 模式后添加它们。应避免在语言特定的 Regex 模式前使用“单词”模式，因为“单词”模式可能将较长的亚洲语言字符合并为单词。

图 12.7 文本分析器正则表达式编辑器



使用脚本编辑器框解析

窗口顶部的脚本编辑器框显示样本文本是如何解析的。“Regex 编辑器”列表中解析正则表达式的结果用颜色突出显示，所用颜色与 “Regex 编辑器”列表中的颜色一致。

- 点击**第一行**、**上一行**、**下一行**和**最后一行**等按钮，以便使用您自己的数据文本填充脚本编辑器框。这使您可以了解给定的文本数据行是如何解析的。也可以在编辑框中输入行号，以使用数据表中特定行的文本填充脚本编辑器框。
- 点击**保存至列**按钮以将新列保存到包含正则表达式标记化结果的数据表。有关指定正则表达式结果的详细信息，请参见第 342 页的“编辑正则表达式”。若通过**列 > 实用工具 > 通过文本匹配新建列**访问“正则表达式编辑器”，则**保存至列**按钮不显示。

注意：保存至列按钮仅使用正则表达式匹配文本。不使用以下设置来修改正则表达式的输出：停止词、重新编码、词干处理、短语或每个单词的最小字符数和最大字符数。

添加正则表达式

要添加准备用在标记化中的正则表达式，请点击列表下方的加号。将显示“Regex 逻辑库选择”窗口。该窗口包含所有内置正则表达式以及所有最近修改的正则表达式（它们是您在“正则表达式编辑器”的以前实例中创建的）。对内置正则表达式做了标记。使用您指定的名称标记在您的逻辑库中保存的定制正则表达式。只有具有给定名称的最新表达式存储在 Regex 逻辑库中。

点击**重新调用**按钮，用“正则表达式编辑器”的最近实例中的正则表达式填充正则表达式列表。重新调用的正则表达式是在点击**保存至列**按钮或**确定**按钮后，在编辑器的上一个实例中显示的正则表达式。

在列表中选择一个或多个正则表达式，然后点击**确定**添加要在标记化中使用的选定正则表达式。使用**删除选定项**按钮从 Regex 逻辑库中删除一个或多个定制正则表达式。每个用户的 Regex 逻辑库作为 JSL 文件存储在名为 TextExplorer 的目录中。该目录的位置基于计算机的操作系统：

- Windows："C:/Users/<用户名>/AppData/Roaming/SAS/JMP/TextExplorer/"
- macOS："/Users/<用户名>/Library/Application Support/JMP/TextExplorer/"

这些文件可以与其他用户共享，但是您不应直接编辑它们。请使用正则表达式编辑器。

编辑正则表达式

通过按“Regex 编辑器”面板中指定的顺序处理正则表达式来标记化词条。要更改正则表达式的顺序，请在列表中选择一个正则表达式并点击列表下的向上或向下箭头按钮。您还可以在正则表达式列表中拖放项来更改执行顺序。蓝色小三角表示当前选择的正则表达式。要删除一个正则表达式并从标记化中排除它，请在列表中选择它并点击列表下的减号。不能删除“多余”正则表达式，它必须在正则表达式序列中显示为最后一个。

当您在列表中选择一个正则表达式时，“Regex 编辑器”面板中的可编辑字段指的是选定的正则表达式。点击并在任意这些字段中键入可以编辑它们。

每个正则表达式具有以下特性：

标题 指定一个名称，用于标识当前窗口中的正则表达式 （以及以后 “Regex 逻辑库”中的正则表达式）。

Regex 指定正则表达式定义。该正则表达式必须至少具有一组括号来指定正则表达式捕获。

结果 指定用什么替换正则表达式所匹配的文本。该值可以是静态文本、空白或正则表达式捕获的值。正则表达式捕获定义为 Regex 定义的结果：

- 要用静态文本替换匹配的文本，请在“结果”字段中指定静态文本。
- 要忽略匹配的文本，请将“结果”字段留空。
- 要保留正则表达式最外层括号产生的文本，请在“结果”字段中使用“\1”（不含引号）。
- 要保留正则表达式的整个结果，请在“结果”字段中使用“\0”（不含引号）。

示例 （可选）使用指示正则表达式行为的颜色指定示例文本字符串。

备注 （可选）指定一个备注以说明正则表达式和它的行为。

颜色 指定颜色以用于标识脚本编辑器框的文本中和 “示例”字段中正则表达式的匹配。使用箭头按钮更改颜色。

注意：若 Regex 字段中的正则表达式定义无效，在正则表达式列表中该正则表达式名称旁边将显示一个红色的 X。

创建定制正则表达式

按以下步骤来创建您自己的定制正则表达式：

1. 点击列表下方的加号。
2. 在 “Regex 逻辑库选择”窗口中，请注意，“空”正则表达式处于选定状态。
3. 点击**确定**。
4. 在 “Regex 编辑器”面板中编辑 Regex 定义。
5. 在 “标题”字段中给您的定制正则表达式指定一个唯一名称。

提示：编辑 Regex 定义字段时，使“日志”窗口打开并可见很有帮助。一些错误消息仅在“日志”窗口中显示。要打开“日志”窗口，请选择**查看 > 日志**。有很多 Internet 资源可用于排查正则表达式问题，如 https://regexr.com/。

单词分隔符列表

使用**单词分隔符列表**按钮可以指定在标记化过程中单词之间出现的字符列表。**字间字符**不能作为单词开头，但是若某个正则表达式允许，它们可以出现在某个单词内部。点击该按钮时，您可以在显示的窗口中的列表中添加或删除字符。默认情况下，列表中的唯一字符为空格字符。在“分隔符”窗口中，点击**重置**按钮可以撤销对分隔符列表的所有修改。对分隔符列表的修改仅应用到当前正则表达式标记化。

以下步骤说明指定的正则表达式和所需的 “多余”正则表达式的处理过程：

1. 比较文本流中的当前字符与分隔符列表。
   - 若该字符在分隔符列表中，则忽略该字符，处理“多余”临时字符串中的所有累积字符，移到下一字符并重复第 1 步。
   - 若该字符不在分隔符列表中，请转到第 2 步。
2. 将以当前字符开头的字符串与每个正则表达式比较 （一次一个，但是不包括 “多余”正则表达式）。
   - 若以当前字符开头的字符串匹配某个正则表达式，则发生以下事件。处理“多余”临时字符串中的所有累积字符。将“结果”字段的值保存为词条。文本流中的当前字符成为跟着匹配字符串的字符。该处理返回到第 1 步。
   - 若以当前字符开头的字符串不匹配任何正则表达式直到“多余”正则表达式，请转到第 3 步。
3. 通过追加当前字符并将当前字符设置为文本流中的下一字符，收集字符到“多余”临时字符串。返回到第 1 步。
   - “多余”临时字符串一次累积一个字符，直到其他正则表达式中的一个产生匹配。

– “多余”正则表达式的默认结果是放弃累积“多余”临时字符串。

提示：

- 若您将“多余”正则表达式的结果设置为 \1，可能要添加更多分隔符，如标点符号。这确保您的结果不包含指定的标点符号。
- 您可能需要考虑采用以下一个或多个操作来捕获相关词条，而不是将“多余”正则表达式的结果更改为 \1。
  – 从“Regex 逻辑库”添加更多正则表达式。
  – 创建定制正则表达式。

该处理遵循上述步骤，直到到达数据表中每行的文本字符串结尾。

将结果保存到数据表中的列

点击**保存至列**按钮以保存到数据表的一个新列，其中包含正则表达式标记化的结果。新列是一个字符列，其名称与在“文本分析器”启动窗口中指定的文本列名称相同；同时追加一个数字到该名称以便列名称是唯一的。您还可以使用“列”>“实用工具”>“通过文本匹配新建列”下的独立“正则表达式”实用工具。请参见《使用 JMP》。

注意：当您将定制正则表达式标记化的结果保存到数据表中的列时，在数据表的每行的原始文本上运行正则表达式处理。不在转换为小写形式的文本字符串版本上运行它。

关闭文本分析器正则表达式编辑器

在“文本分析器正则表达式编辑器”窗口中点击**确定**后，发生以下事件：

1. 将在“文本分析器正则表达式编辑器”窗口中定义的定制正则表达式保存到 Regex 逻辑库。

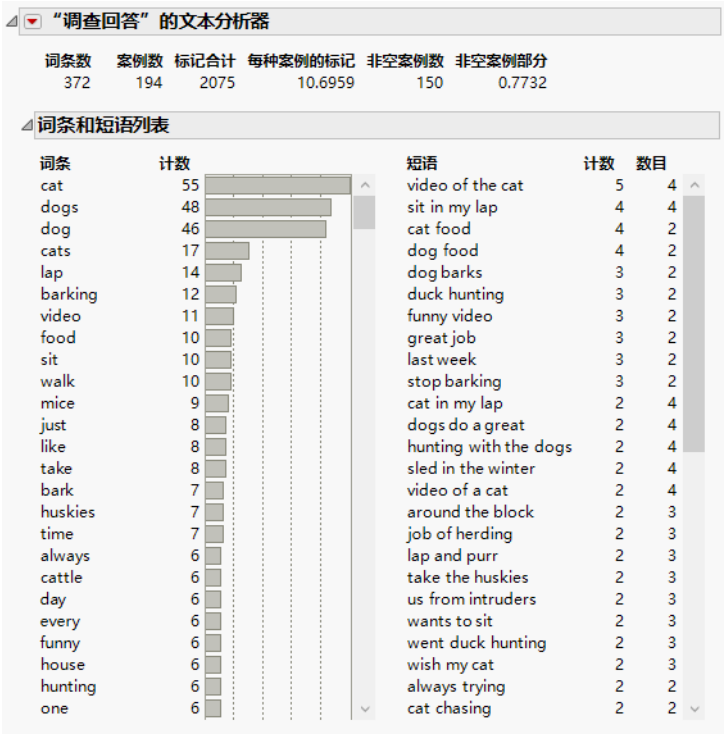
   警告：仅当您点击**确定**并且存在定制正则表达式时，才保存定制 Regex 逻辑库。最新保存的正则表达式将在下次可用。使用唯一名称来在 Regex 逻辑库中保留更多正则表达式。要确保某个正则表达式在以后可用，您可以在“文本分析器报表”窗口中保存一个脚本。

2. 将显示文本分析器报表。该报表显示使用指定的正则表达式设置来标记化文本的结果。

文本分析器报表

“文本分析器报表”窗口包含 “汇总计数”报表和 “词条和短语列表”报表。

图 12.8 文本分析器报表的示例



“调查回答” 的文本分析器

词条数	案例数	标记合计	每种案例的标记	非空案例数	非空案例部分
372	194	2075	10.6959	150	0.7732

词条和短语列表

词条	计数
cat	55
dogs	48
dog	46
cats	17
lap	14
barking	12
video	11
food	10
sit	10
walk	10
mice	9
just	8
like	8
take	8
bark	7
huskies	7
time	7
always	6
cattle	6
day	6
every	6
funny	6
house	6
hunting	6
one	6

短语	计数	数目
video of the cat	5	4
sit in my lap	4	4
cat food	4	2
dog food	4	2
dog barks	3	2
duck hunting	3	2
funny video	3	2
great job	3	2
last week	3	2
stop barking	3	2
cat in my lap	2	4
dogs do a great	2	4
hunting with the dogs	2	4
sled in the winter	2	4
video of a cat	2	4
around the block	2	3
job of herding	2	3
lap and purr	2	3
take the huskies	2	3
us from intruders	2	3
wants to sit	2	3
went duck hunting	2	3
wish my cat	2	3
always trying	2	2
cat chasing	2	2

“汇总计数”报表

“文本分析器报表”窗口中的第一个表包含以下汇总统计量：

词条数 词条列表中的词条数。

案例数 语料库中的文档数。

标记合计 语料库中的词条总数。

每种案例的标记 标记数除以案例数所得的值。

非空案例数 语料库中至少包含一个词条的文档数。

非空案例部分 语料库中至少包含一个词条的文档比例。

词条和短语列表

“词条和短语列表”报表包含标记化后在文本中找到的词条和短语的表。有关“词条和短语列表”报表的示例，请参见图 12.8。词条列表中的“计数”列指示词条在语料库中出现的次数。短语列表中的“计数”列指示短语在语料库中出现的次数，“数目”列指示短语中的单词数。

默认情况下，词条列表按计数的降序排序；计数相同的词条按字母顺序排序。短语列表按计数的降序排序，计数相同的短语按长度 (N) 的降序排序。短语列表中再次出现相同计数时按字母顺序排序。可以使用每个列表中的选项将每个列表的排序顺序更改为字母排序。

短语列表中显示的短语由启动窗口中的**每个短语的最大单词数**和**最大短语数**选项的设置决定。在数据表中仅出现一次的短语不显示在短语列表中。

短语可以指定为各种作用域的词条。根据短语规格的作用域对已指定为词条的短语列表中的短语着色（表 12.1）。 有关指定不同作用域的短语的详细信息，请参见第 351 页的“词条选项管理窗口”。

表 12.1 指定短语的颜色

范围	颜色
内置	红色
用户逻辑库	绿色
项目	蓝色
列属性	橙色
本地	灰色

针对词条和短语的操作

您可以通过以下方式访问“词条列表”和“短语列表”表中的选项：选择项，然后在每个表最左侧的列中右击。可以通过以下方式将每个表保存为一个数据表：在每个表的“计数”列中右击，然后选择“制成数据表”。

词条列表弹出菜单选项

当您在 “词条列表”表的 “词条”列中右击时，将显示一个包含以下选项的弹出菜单：

选择行 在包含选定词条的数据表中选择行。

显示文本 显示包含选定词条的文档。

> **注意：**默认情况下，只显示前 10,000 个文档。若包含选定词条的文档数超过 10,000，将显示一个窗口，您可以在其中增大此限值。

按字母顺序 指定 “词条列表”的排序顺序。若选定该选项，则词条按字母顺序排序。若未选定该选项，词条按计数的降序排序。

数值顺序 （仅当选定 “字母顺序”选项时才可用。）指定 “词条列表”的排序顺序。选择该选项后，各项将拆分为字符串和数值段，然后按数值顺序对数值段进行排序。有关“数值顺序”选项使用的排序规则的详细信息，请参见 《使用 JMP》。

复制 将选定词条放置到剪贴板。

颜色 使您可以将颜色分配给选定词条。

标签 将标签放置在选定词条的词条 SVD 图中相应的点上。

包含短语 在 “短语列表”表中选择包含选定词条的短语。

保存指示符 为在词条列表中选择的每个词条将指示符列保存到数据表。若行中的文档包含词条，则该行的指示符列的值为 1；否则为 0。

保存公式 为在词条列表中选择的每个词条将列公式保存到数据表。若行中的文档包含词条，则该行的列公式计算结果为 1；否则为 0。这对于新文档很有用。

重新编码 使您可以更改一个或多个词条的值。在选择该选项前请在列表中选择词条。在选择该选项后，将显示 “重新编码”窗口。请参见 《使用 JMP》。

添加停止词 将选定词条添加到停止词列表并从词条列表中删除这些词条。该操作还将更新短语列表。

注意：若将词干词添加为停止词，则与该词干对应的所有标记都将添加为停止词。

添加词干例外情况 （仅当 “语言”选项设置为 “英语”、“德语”、“西班牙语”、“法语”或 “意大利语”时才可用。）将选定词条添加到从词干处理中排除的词条列表。

删除短语 （仅当在 “词条列表”中选定指定的短语时才可用。）从一组指定短语中删除选定的短语，并相应更新 “词条计数”。

注意：若已将某个短语添加为 “情感短语”，则 “删除短语”选项也会将该短语从当前 “情感分析”报表的情感词条列表中删除。

JMP PRO **添加情感** （仅当 “情感分析”报表在当前报表窗口中打开时才可用。）将所选词条添加到当前 “情感分析”报表的情感词条列表中。

注意：若将词干词添加为情感词条，则与该词干对应的所有标记都将添加为情感词条。

显示过滤器 显示或隐藏“词条列表”上方的搜索过滤器。请参见第 348 页的“搜索过滤器选项”。

制成数据表 从报表创建 JMP 数据表。

制作合并数据表 在报表中搜索类似您所选表的其他表，然后将其合并到单个 JMP 数据表中。

短语列表弹出菜单选项

当您在 “短语列表” 表的 “短语” 列中右击时，将显示一个包含以下选项的弹出菜单：

选择行 在包含选定短语的数据表中选择行。

显示文本 显示包含选定短语的文档。

保存指示符 为在短语列表中选择的每个短语将指示符列保存到数据表。若行中的文档包含短语，则该行的指示符列的值为 1；否则为 0。

按字母顺序 指定 “短语列表” 的排序顺序。若选定该选项，则词条按字母顺序排序。若未选定该选项，词条按计数的降序排序。

数值顺序 （仅当选定 “字母顺序” 选项时才可用。）指定 “短语列表” 的排序顺序。选择该选项后，各项将拆分为字符串和数值段，然后按数值顺序对数值段进行排序。有关 “数值顺序” 选项使用的排序规则的详细信息，请参见 《使用 JMP》。

复制 将选定的短语放置在剪贴板上。

选择包含项 在短语列表中选择包含选定短语的较大短语。

选择所含项 在 “短语列表” 中选择较小短语，并且在 “词条列表” 中选择包含在选定短语中的词条。

添加短语 将选定短语添加到词条列表并相应更新词条计数。

添加停止词 向停止词列表添加选定短语。该操作还将更新词条列表。

JMP PRO **添加情感短语** （仅当 “情感分析” 报表在当前报表窗口中打开时才可用。）将所选短语添加至 “词条列表” 以及当前 “情感分析” 报表中的情感词条列表。

显示过滤器 显示或隐藏 “短语列表” 上方的搜索过滤器。请参见第 348 页的 “搜索过滤器选项”。

制成数据表 从报表创建 JMP 数据表。

制作合并数据表 在报表中搜索类似您所选表的其他表，然后将其合并到单个 JMP 数据表中。

搜索过滤器选项

点击搜索框旁边的下箭头按钮以优化搜索。

包含词条 返回包含一部分搜索条件的项。搜索 “ease oom” 返回如 “Release Zoom” 这样的消息。

包含短语 返回包含完全搜索条件的项。搜索 “text box” 返回包含 “text” 后面直接跟着 “box” 的条目 （例如，“Context Box” 和 “Text Box”）。

以短语开头 返回以搜索条件开始的项。

以短语结尾 返回以搜索条件结束的项。

整条短语 返回包含整个字符串的项。搜索 “text box” 返回仅包含 “text box” 的条目。

正则表达式 允许您在搜索框中使用通配符 (*) 和句点 (.)。搜索“get.*name”查找包含“get”后面跟着一个或多个单词的项。它返回“Get Color Theme Names”、“Get Name Info”、“Get Effect Names”等。

反转结果 返回不匹配搜索条件的项。

匹配全部项 返回同时包含两个字符串的项。搜索“t test”返回包含任一搜索字符串或两者的元素：“Pat Test”、“Shortest Edit Script”和“Paired t test”。

忽略大小写 忽略搜索条件中的大小写。

全字匹配 基于“匹配全部项”设置返回包含字符串中每个单词的项。若您搜索“data filter”，并且选择了“匹配全部项”，则返回同时包含“data”和“filter”的条目。

“文本分析器”平台选项

本节介绍“文本分析器”平台中提供的选项。

- “文本准备选项”
- “文本分析选项”
- “保存选项”
- “报表选项”

文本准备选项

“文本分析器”红色小三角菜单包含以下文本准备选项：

显示选项 显示控制报表显示的选项子菜单。

显示词云 显示或隐藏“词云”报表。使用“词云”红色小三角菜单可以更改词云的布局和字体。请参见第 351 页的“词云选项”。

可以通过更改宽度来交互式调整词云大小。然后自动确定高度。将词条列表中的行链接到词云中的词条。

显示词条列表 显示或隐藏词条列表。

显示短语列表 显示或隐藏短语列表。

显示词条和短语选项 显示“词条和短语列表”报表中的按钮，它们对应于每个列表的弹出菜单中提供的选项。请参见第 346 页的“词条和短语列表”。

显示汇总计数 显示或隐藏“汇总计数”表。请参见第 345 页的“‘汇总计数’报表”。

显示停止词 显示或隐藏在分析中使用的停止词列表。最初使用内置的停止词列表。要添加一个停止词，请右击词条列表中的该词，然后从弹出菜单中选择**添加停止词**。请参见第 351 页的“词条选项管理窗口”。

显示重新编码 显示或隐藏重新编码的词条列表。请参见第 351 页的“词条选项管理窗口”。

显示指定短语 显示或隐藏已被用户指定视为词条的短语列表。请参见第 351 页的“词条选项管理窗口”。

显示词干例外情况 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”时才可用。）显示或隐藏从词干处理中排除的词条。请参见第 351 页的“词条选项管理窗口”。

显示分隔符 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”并且选定的“标记化”方法为“基本单词”时才可用。）显示或隐藏“基本单词”标记化方法使用的分隔符。要修改使用的一组分隔符，必须在 JSL 中使用 `Add Delimiters()` 或 `Set Delimiters()` 消息。

显示词干报表 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”并且选定的“词干处理”方法为“无需词干处理”时才可用。）显示或隐藏包含词干处理结果的两个表的“词干处理”报表。左侧的表将每个词干映射到相应词条。右侧的表将每个词条映射到相应词干。

显示选定行 打开一个窗口，其中包含在当前选定行中的文档文本。

显示所有表的过滤器 显示或隐藏可用于在报表中搜索表的过滤器。该选项适用于以下表：停止词、指定的短语、词干例外情况、词条列表、短语列表和词干报表。有关过滤器工具的详细信息，请参见第 348 页的“搜索过滤器选项”。

词条选项 显示适用于词条列表的选项的子菜单。

词干处理 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”时才可用。）请参见第 338 页的“启动“文本分析器”平台”中有关词干处理选项的说明。

包括内置停止词 指定在标记化过程中使用的停止词是否包括内置停止词。

包括内置短语 指定在标记化过程中使用的短语是否包括内置短语。

管理停止词 显示一个窗口，您可以在其中添加或删除停止词。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的停止词。请参见第 351 页的“词条选项管理窗口”。

管理重新编码 显示一个窗口，您可以在其中添加或删除重新编码。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的重新编码。请参见第 351 页的“词条选项管理窗口”。

管理短语 显示一个窗口，您可以在其中添加或删除视为词条的短语。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的短语。请参见第 351 页的“词条选项管理窗口”。

管理词干例外情况 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”时才可用。）显示一个窗口，您可以在其中添加或删除词干处理例外情况。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的词干例外情况。请参见第 351 页的“词条选项管理窗口”。

解析选项 显示适用于解析和标记化的选项的子菜单。

标记化 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”时才可用。）请参见第 338 页的“启动“文本分析器”平台”中关于标记化选项的说明。

定制 Regex （仅适用于 Regex 标记化方法。）显示“定制 Regex”窗口。使用该选项可以修改当前“文本分析器”报表的 Regex 设置。

注意：若您在平台启动窗口中指定了“依据”变量，则“定制 Regex”选项自动传播到“依据”变量的所有水平。

将数字视为单词 （仅当“语言”选项设置为“英语”、“德语”、“西班牙语”、“法语”或“意大利语”并且“基本单词”是选定的“标记化”方法时才可用。）允许数字标记化为分析中的词条。请注意该选项受“每个单词的最小字符数”设置的影响。

词云选项

“词云”红色小三角菜单包含以下选项：

布局 指定词条在词云中的排列方式。默认情况下，“布局”设置为 “有序”。

有序 在水平线上将词条按频数从最高到最低顺序呈现。

按字母顺序 在水平线上呈现词条，按字母顺序升序排序。

中心化 呈现云中的词条并按频数确定大小。

着色 指定词云中词条的颜色。默认情况下，“着色”设置为 “无”。

无 按照每个词条在“词条列表”中的着色为其着相同颜色。

均匀颜色 为每个词条着相同颜色。可以在“图例”中更改该颜色。

任意灰色 用不同的灰度指定每个词条的颜色。

任意颜色 用各种颜色指定词条的颜色。可以在“图例”中调整颜色。

按列值 用梯度色标指定每个词条的颜色。该尺度基于“按列对词条评分”选项生成的词条的得分。可以在“图例”中调整颜色和梯度。

字体 指定词云中词条的字体、样式和大小。

显示图例 显示或隐藏词云的图例。

词条选项管理窗口

可以为很多不同作用域指定停止词、重新编码、短语和词干例外情况信息。它们可以存储在以下位置：文本分析器用户逻辑库（“用户”作用域）、当前项目、分析列的列属性（“列”作用域）或平台脚本（“本地”作用域）中。对于文本分析器的特定实例，您可以通过保存文本分析器报表的脚本来保存其本地规格和本地例外情况。

“词条选项”管理窗口是四个类似的窗口，您可以使用它们来管理停止词、重新编码、短语和词干例外情况的集合。图 12.9 显示“管理停止词”窗口。“管理短语”和“管理词干例外情

况”窗口与“管理停止词”窗口相同。“管理重新编码”窗口略有不同。请参见第 353 页的“管理重新编码”。

图 12.9 “管理停止词”窗口

管理停止词

“管理停止词”窗口包含停止词的多个列表，它们表示指定停止词的不同作用域（或位置）。每个列表下面有一个文本编辑框和一个“添加”按钮。使用这些控件可以将定制停止词添加到每个作用域。您可以通过拖动停止词将它们从一个作用域移到另一作用域。可以将项从一个列表复制并粘贴到另一列表。窗口底部的两个按钮用于将选定项从一个作用域移到另一个（左边或右边）作用域。X 按钮用于将选定项从当前作用域中删除。您可以通过双击某个项并更改文本来编辑列表中的现有项。

语言 指定内置停止词列表以及以哪种语言保存用户逻辑库选择内容。若您为“语言”选择“应用项”，则将更改保存到主用户逻辑库。“语言”设置仅适用于“内置”、“用户”和“项目”作用域。

内置（已锁定） 列出指定语言的停止词内置列表。可以通过将某个内置停止词放入“本地例外情况”列表中来排除它。

用户 列出用户逻辑库中指定语言的停止词。

项目 （仅当在项目中启动“文本分析器”时才可用。）列出当前项目中指定语言的停止词。

列 列出文本列的“停止词”列属性中的停止词。

本地 列出本地作用域中的停止词。您可以在通过 JSL 启动“文本分析器”时指定停止词。这些停止词仅用于当前“文本分析器”平台报表。

本地例外情况 列出在当前“文本分析器”平台中未被视为停止词的单词。您可以在通过 JSL 启动“文本分析器”时指定停止词。“本地例外情况”中列出的单词取代所有其他作用域中列出的单词。

导入 使您可以从文本文件导入停止词。将停止词复制到剪贴板。您可以将它们粘贴到除 “内置”之外的任意列表。

导出 使您可以将停止词导出到剪贴板或文本文件。显示一个 “导出”窗口，您可以在其中选择要将停止词导出到的作用域以及导出位置。

用户逻辑库文件位于 TextExplorer 目录中。该目录的位置基于计算机的操作系统：

- Windows："C:/Users/<用户名>/AppData/Roaming/SAS/JMP/TextExplorer/<语言>/"
- macOS："/Users/<用户名>/Library/Application Support/JMP/TextExplorer/<语言>/"

主用户逻辑库文件位于 TextExplorer 目录本身中。这些文件不是语言特定的。

点击确定后，对“用户”、“项目”和“列”列表的更改将分别保存到用户逻辑库、项目和列属性中。仅当保存“文本分析器”报表的脚本时，才保存在“本地”和“本地例外情况”列表中指定的所有内容。

若将停止词保存到用户逻辑库，则该文件名为 stopwords.txt。若保存到列属性，则该属性称为“停止词”。

管理重新编码

“管理重新编码”窗口与“管理停止词”窗口略有不同。此时每个列表下面不是有一个文本编辑框而是有两个文本编辑框。将旧值（在顶部框中指定）重新编码为新值（在底部框中指定）。

若将重新编码保存到用户逻辑库，则该文件名为 recodes.txt。若保存到列属性，则该属性称为“重新编码”。

管理短语

若将短语保存到用户逻辑库，则该文件名为 phrases.txt。若保存到列属性，则该属性称为“短语”。

管理词干例外情况

若将词干例外情况保存到用户逻辑库，则该文件名为 stemExceptions.txt。若保存到列属性，则该属性称为“词干例外情况”。

注意：“管理词干例外情况”窗口中的“本地例外情况”列表列出从词干例外情况列表中排除的词干例外情况。该列表中的单词参与词干处理操作。

JMP PRO 管理否定词条

“管理否定词条”窗口在“情感分析”报表中提供。请参见第 367 页的“情感分析”。

若将否定词条保存到用户逻辑库，则该文件名为 negations.txt。若保存到列属性，则该属性称为“否定词条”。

注意：显示在“本地”列表或“本地例外情况”列表中的词条仅应用于当前“情感分析”报表。

JMP PRO 管理强化词条

“情感分析”报表中提供的“管理强化词条”窗口与“管理停止词”窗口略有不同。除了每个列表下面的文本编辑框之外，还有一个“乘数”控件。“乘数”控件支持您在将某个词条添加到一组强化词条时为该词条指定强化乘数。请参见第 367 页的“情感分析”。

若将强化词条保存到用户逻辑库，则该文件名为 intensifiers.txt。若保存到列属性，则该属性称为“强化词条”。

注意：显示在“本地”列表或“本地例外情况”列表中的词条仅应用于当前“情感分析”报表。

JMP PRO 管理情感词条

“情感分析”报表中提供的“管理情感词条”窗口与“管理停止词”窗口略有不同。除了每个列表下面的文本编辑框之外，还有一个“得分”控件。“得分”控件支持您在将某个词条添加到一组情感词条时为该词条指定情感得分。请参见第 367 页的“情感分析”。

若将情感词条保存到用户逻辑库，则该文件名为 sentiments.txt。若保存到列属性，则该属性称为“情感词条”。

注意：显示在“本地”列表或“本地例外情况”列表中的词条仅应用于当前“情感分析”报表。

JMP PRO 文本分析选项

“文本分析器”红色小三角菜单包含以下分析选项：

潜在类分析 使用稀疏矩阵例程对二进制加权文档词条矩阵执行潜在类分析。请参见第 357 页的“潜在类分析”。

当您从“文本分析器”红色小三角菜单选择“潜在类分析”时，将显示包含以下选项的“规格”窗口：

最大词条数 包括在潜在类分析中的最大词条数。

最小词条频数 一个词条要包括在潜在类分析中必须出现的最小次数。

聚类数 潜在类分析中的聚类数。

潜在语义分析 , SVD 执行文档词条矩阵的偏奇异值分解。请参见第 358 页的“潜在语义分析 (SVD)”。

判别分析 根据文档词条矩阵预测组或类别中每个文档的成员关系。请参见第 363 页的“判别分析”。

词条选择 分析哪些词条最适合解释不同响应。当响应是评级时，“词条选择”也有助于情感分析。请参见第 364 页的 “词条选择”。

情感分析 （仅当“语言”选项设置为“英语”时才可用。）使用词法分析识别文档中的情感词条，并对文档的正面、负面和整体情感进行评分。请参见第 367 页的 “情感分析”。

JMP PRO “奇异值分解规格”窗口

“文本分析器”平台中的分析选项基于文档词条矩阵 (DTM)。通过为词条列表中的每个词条（最大为指定的最大词条数）创建列来生成 DTM。每个文本文档（等价于数据表中的行）对应 DTM 的一行。DTM 单元格中的值取决于用户在“规格”窗口中指定的权重类型。

图 12.10 显示 “奇异值分解规格”窗口。当您从 “文本分析器”红色小三角菜单选择对文档词条矩阵执行奇异值分解的选项时，将显示包含以下选项的 “规格”窗口：

最大词条数 包括在奇异值分解中的最大词条数。

最小词条频数 一个词条必须要包括在奇异值分解中的最小次数。

权重 用于确定进入文档词条矩阵单元格的值的权重方案。在第 356 页的 “ “文档词条矩阵规格”窗口”中介绍了各个权重方案选项。

奇异向量数 奇异值分解中的奇异向量数。默认值为文档数、词条数或 100 中的最小值。

中心化和统一尺度 文档词条矩阵的中心化和统一尺度选项。您可以选择**中心化和统一尺度**、**中心化**和**未中心化**。默认情况下，文档词条矩阵已中心化且统一尺度。

图 12.10 “SVD 规格”窗口

保存选项

“文本分析器”红色小三角菜单包含以下选项，用于将信息保存到数据表、表列和列属性：

保存文档词条矩阵 将文档词条矩阵每个列（最多到指定的最大词条数）都保存到数据表中去。

JMP PRO **保存用于关联的堆叠 DTM** 将文档词条矩阵的堆叠形式保存到 JMP 数据表。堆叠形式适用于 “关联分析”平台中的分析。请参见 《预测和专业建模》。若您在 “文本分析器”启

动窗口中指定 ID 变量，可使用该 ID 变量标识每个词条来自原始文本数据表中的哪一行。堆叠的表还包含一个用于启动 “关联分析”的表脚本。

保存 DTM 公式 将建模类型为 “向量”的公式列保存到数据表中。向量的长度取决于用户指定的“最大词条数”、“最小词条频数”和 “权重”选项。生成的列使用 `Text Score()` JSL 函数。有关该函数的详细信息，请参见 “帮助” > “脚本索引”。

保存词条表 创建一个 JMP 数据表，它包含词条列表中的每个词条、出现次数和包含每个词条的文档数。若您在选择 “保存词条表”后选择了 “按列对词条评分”选项，则会向 “保存词条表”选项创建的表添加包含每个词条的得分的一列。

按列对词条评分 将基于指定列中的值的得分保存到 “保存词条表”选项创建的 JMP 数据表中。每个词条的得分是指定列的均值用该词条在每行中的出现次数加权得到。若您已经选定“保存词条表”选项，则 “按列对词条评分”选项会向 “保存词条表”选项创建的数据表添加包含得分的一列。如若不然，则会为该词条表创建 JMP 数据表。若指定的列非 “连续”列，则创建包含指定列中每个水平的得分的列。

“文档词条矩阵规格”窗口

当您从 “文本分析器”红色小三角菜单选择 “保存文档词条矩阵”和 “保存 DTM 公式”选项时，将显示包含以下选项的 “文档词条矩阵规格”窗口：

最大词条数 包括在文档词条矩阵中的最大词条数。

最小词条频数 一个词条要包括在文档词条矩阵中必须出现的最小次数。

权重 用于确定进入文档词条矩阵单元格的值的权重方案。

为 “权重”提供了以下选项：

二进制 若词条在每个文档中出现，则分配 1；否则分配 0。这是默认权重，除非之前运行过 SVD 分析。

三进制 若词条在每个文档中出现一次以上，则分配 2；若仅出现一次，则分配 1，否则分配 0。

频数 分配每个文档中词条的出现次数计数。

频数对数 分配 $log10(1 + x)$，其中 x 是每个文档中词条的出现次数计数。

TF IDF 分配 $TF * log10(nDoc / nDocTerm)$。**词条频数 - 反转文档频数**的缩写。这是默认权重。公式中的词条定义如下：

TF = 文档中的词条频数

nDoc = 语料库中的文档数

nDocTerm = 包含词条的文档数

注意：若在运行 SVD 分析后选择“保存文档词条矩阵”或“保存 DTM 公式”，“规格”窗口包含来自最近的 SVD 分析的规格。

报表选项

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO 潜在类分析

在“文本分析器”平台中，使用潜在类分析，您可以将语料库中的文档分组为类似文档的聚类。“潜在类分析”报表包含模型规格、模型的 Bayesian 信息准则 (BIC) 值和 “显示文本”按钮。若在“聚类混合概率”表中选择了一个或多个聚类，则点击“显示文本”按钮可打开一个窗口，其中包含最可能属于选定聚类的文档的文本。

“潜在类分析”红色小三角菜单包含以下选项：

显示选项 指定 “潜在类分析”报表的内容。默认情况下，显示除每个聚类的词云之外的所有报表选项。

聚类混合概率 显示或隐藏某观测属于每个聚类的概率的表。

提示：您可以在“按聚类划分的混合概率”表中选择一个或多个行，以选择分配给相应聚类的观测。

按聚类划分的词条概率 显示或隐藏一个词条表，其中包含每个聚类的条件概率估计值（条件概率是指若某文档属于某个特定聚类，则该文档包含词条的概率）。默认情况下，该表中的词条在语料库中按频数的降序排序。

“最具特征的聚类”列显示词条出现在其中比率最高的聚类。

“最有可能的聚类”列显示最可能找到随机选择的、包含某词条的文档的聚类。

按聚类划分的前几位词条 显示或隐藏每个聚类中得分最高的十个词条的表。聚类 c 中词条 t 的得分 $S_{t,c}$ 计算如下：

$$S_{t,c} = 100 \bullet \text{均值}(p_t) \bullet \log 10\left(\frac{p_{t,c}}{\text{均值}(p_t)}\right)$$

其中，均值 (p_t) 是词条 t 按聚类划分的词条概率的均值，$p_{t,c}$ 是聚类 c 中词条 t 的按聚类划分的词条概率。

MDS 图 显示或隐藏一个多维尺度化图，它是聚类的邻近关系的两维表示。有关 MDS 图的详细信息，请参见《多元方法》。点击“显示文本”按钮将打开一个窗口，其中包含选定文档的文本。

按行划分的聚类概率 显示或隐藏“混合概率”表，该表显示每行的聚类成员关系的概率。“最可能的聚类”列指示每行具有最高成员关系概率的聚类。

按聚类划分的词云 显示或隐藏词云矩阵，每个聚类一个词云。

重命名聚类 允许您为一个或多个聚类添加说明性名称。

保存概率 将“混合概率”表中的值保存到数据表中的相应行。

保存概率公式 将每个聚类的公式列以及最可能的聚类的公式列保存至数据表。

保存的得分公式使用 `Text Score()` JSL 函数且将权重参数设置为 “LCA”。

按聚类设定颜色 根据最可能的聚类为数据表中的每行设定颜色。

删除 从“文本分析器”报表中删除“潜在类分析”报表。

有关潜在类分析的详细信息，请参见《多元方法》。

注意：在“文本分析器”平台中使用的 LCA 算法利用文档词条矩阵的稀疏性。因为这个原因，“文本分析器”平台中的 LCA 结果与“潜在类分析”平台中的结果不完全一致。

JMP PRO 潜在语义分析 (SVD)

在“文本分析器”平台中，潜在语义分析侧重于计算文档词条矩阵 (DTM) 的偏奇异值分解 (SVD)。该分解将文本数据简化为可处理的维数来进行分析。潜在语义分析等价于执行主成分分析 (PCA)。

偏奇异值分解使用三个矩阵来近似 DTM：**U**、 **S** 和 **V′**。这三个矩阵的关系定义如下：

$$DTM \approx \mathbf{U} * \mathbf{S} * \mathbf{V}'$$

将 *nDoc* 定义为 DTM 中的文档（行）数，*nTerm* 定义为 DTM 中的词条（列）数，*nVec* 定义为指定的奇异向量数。请注意，*nVec* 必须小于等于 min(*nDoc*, *nTerm*)。由此判定：**U** 是包含 DTM 的左侧奇异向量的 *nDoc* * *nVec* 矩阵。**S** 是维 *nVec* 的对角矩阵。**S** 中的对角线元素是 SVD 中的奇异值。**V′** 是 *nVec* * *nTerm* 矩阵。**V′** 中的行（或 **V** 中的列）是右奇异向量。

右侧奇异向量使用类似含义或主题领域捕获不同词条之间的关联。若三个词条倾向于出现在同一文档中，SVD 可能在 **V′** 中生成一个对这三个词条值都很大的奇异向量。**U** 奇异向量表示投影到这个新词条空间的文档。

潜在语义分析还捕获间接关联。若两个单词从不一起出现在同一文档中，但是它们通常出现在具有第三个单词的文档中，则 SVD 可以捕获一些这样的关联。若两个文档没有相同的单词但是包含在降维空间中有关联的单词，则它们映射到 SVD 输出中的类似向量。

SVD 将文档数据变换为固定维的向量空间，使它适用于所有类型的聚类、分类和回归技术。使用“保存”选项可以将这个向量空间导出到其他 JMP 平台去分析。

DTM 在进行奇异值分解前默认是中心化和统一尺度的，并且被 *nDoc* 减 1 除。该分析等价于 DTM 相关性矩阵的 PCA。

您还可以在 “规格”窗口中指定 “中心化”或 “未中心化”。

- 若您指定“中心化”，则 DTM 在进行奇异值分解前是中心化的，并且被 *nDoc* 减 1 除。该分析等价于 DTM 协方差矩阵的 PCA。
- 若您指定“未中心化”，则 DTM 在进行奇异值分解前被 *nDoc* 除。该分析等价于未统一尺度的 DTM 的 PCA。

SVD 实施利用了 DTM 的稀疏，即使是在 DTM 中心化时。

JMP PRO SVD 报表

“文本分析器”平台中的“潜在语义分析”选项从奇异值分解生成两个 SVD 图和一个奇异值表。

JMP PRO SVD 图

第一个图为每个文档包含一个点。对于给定文档，标绘的点的定义方式如下：前两个奇异向量（**U** 矩阵的前两列）中的文档值乘以对角线奇异值矩阵 (**S**)。该图等同于“主成分”平台中的得分图。该图中的每个点都表示一个文档（数据表行）。您可以通过选择该图中的点来选择数据表中的相应行。

第二个图为每个词条包含一个点。对于给定词条，标绘的点的定义方式如下：前两个奇异向量（**V'** 矩阵的前两行）中的词条值乘以对角线奇异值矩阵 (**S**)。该图等同于“主成分”平台中的“载荷图”。在该图中，这些点对应“词条列表”表中的行。

在每个 SVD 图的上方，您可以点击“显示文本”按钮打开一个窗口，其中包含图中选定点的文本。

JMP PRO 奇异值

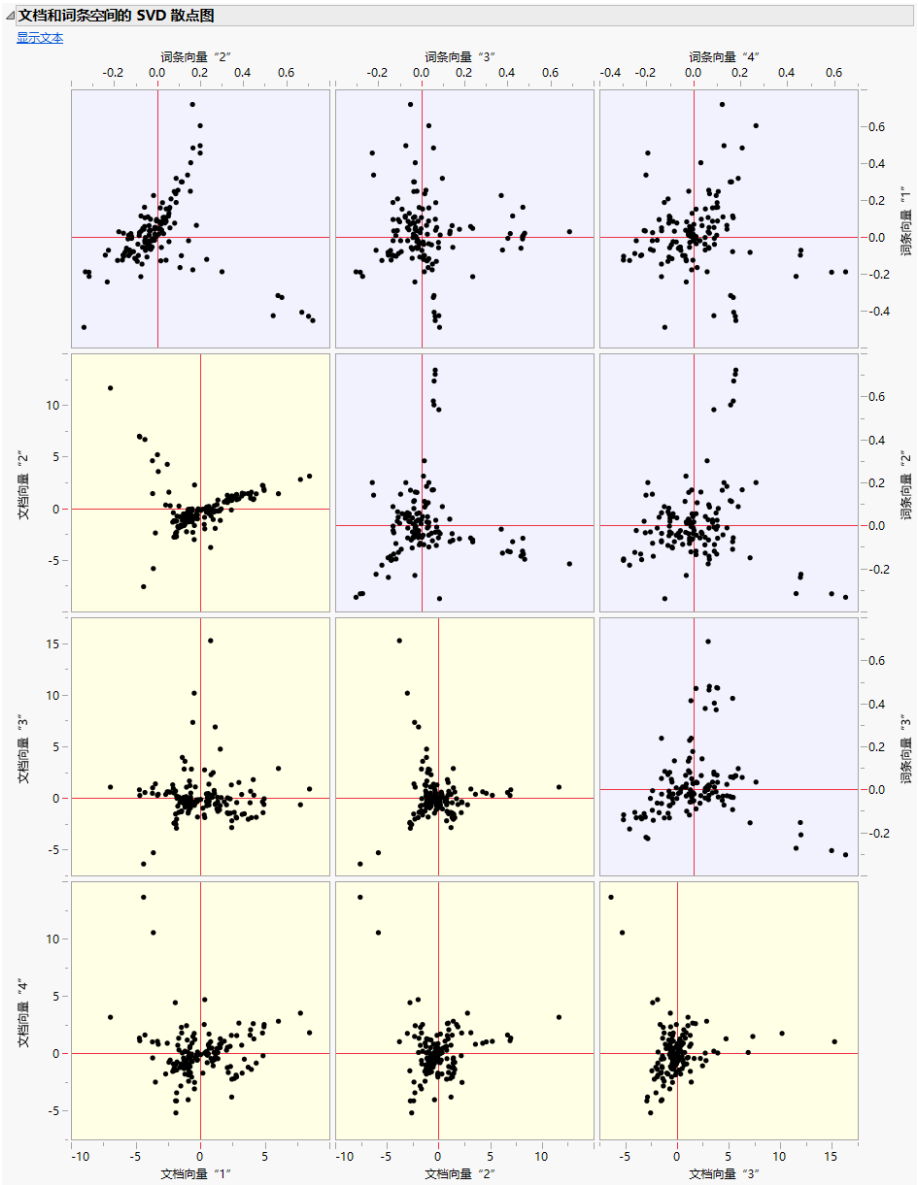
在文档和词条 SVD 图下面，显示一个奇异值的表。这些在文档词条矩阵的奇异值分解中是 **S** 矩阵的对角线元素。“奇异值”表还包含等价的主成分分析的相应特征值列。类似于“主成分”平台，每个特征值（或奇异值）解释的变异的百分比和累积百分比都对应一列。您可以使用“累积百分比”列决定您要保留的来自 DTM 的方差百分比，然后使用相应的奇异向量数。

JMP PRO SVD 报表选项

“文本分析器”平台中的 “SVD”红色小三角菜单包含以下选项：

SVD 散点图矩阵 显示或隐藏词条和文档奇异值分解向量的散点图矩阵。选择该选项时，系统提示您选择散点图矩阵的大小。使用该散点图矩阵不仅可视化奇异值分解的前两个维，还可以可视化更多维。点击“显示文本”按钮将打开一个窗口，其中包含选定文档的文本。

图 12.11 文档和词条空间的 SVD 散点图



主题分析，旋转 SVD 执行文档词条矩阵的可变最大旋转偏奇异值分解以生成称为主题的词条组。您可以多次选择该选项以查找不同的主题数。请参见第 361 页的“主题分析”。

对词条进行聚类 显示或隐藏数据中词条的层次聚类分析。在系统树图右侧，提供了用于设置聚类数和将聚类保存至数据表的选项。对于每个词条，该数据表包含频数、包含词条的文档数以及给它分配的聚类。有关层次聚类和系统树图的详细信息，请参见 《多元方法》。

对文档进行聚类 显示或隐藏数据中文档的层次聚类分析。系统树图的右侧，可使用选项来执行以下操作：设置聚类数、将聚类保存到数据表中的列以及在选定的系统树图分支中显示文档。

选择近邻 查找并选择文档 SVD 图中所选点的最近邻。该算法使用奇异值分解中的 **U** 矩阵查找最近邻。选择该选项时，您必须指定要选择的最近邻数。默认情况下，该选项选择 10 个最近邻。

保存文档奇异向量 将文档奇异值分解中的用户指定的奇异向量数作为列保存到数据表。前两个保存的列表示在文档 SVD 图中绘制的点。请参见第 358 页的 “潜在语义分析 (SVD)”。

保存奇异向量公式 将建模类型为 “向量” 的公式列 （其中包含文档奇异值分解）保存到数据表中。生成的列使用 `Text Score()` JSL 函数。有关该函数的详细信息，请参见 “帮助” > “脚本索引”。

保存词条奇异向量 将来自词条奇异值分解的用户指定的奇异向量数作为列保存到一个新数据表，该表的每行对应一个词条。若 “词条表” 数据表已打开，该选项将这些列保存到该数据表。前两个保存的列表示在词条 SVD 图中绘制的点。请参见第 358 页的 “潜在语义分析 (SVD)”。

删除 从 “文本分析器” 报表窗口中删除 SVD 报表。

JMP PRO 主题分析

“主题分析，旋转 SVD” 选项对文档词条矩阵 (DTM) 的偏奇异值分解 (SVD) 执行最大方差法旋转。您必须指定一些旋转奇异向量，这些向量与您想要从 DTM 保留的主题数相对应。指定一些主题后就会显示 “主题分析” 报表。

主题分析等价于旋转的主成分分析 (PCA)。最大方差法旋转获取一组奇异向量并旋转它们以使它们更直接指向坐标方向（朝向词条）。当每个旋转向量朝向一组词条时，该旋转使向量有助于解释文本。负值表示排斥力。具有负值的词条与具有正值的词条相比，在主题中出现的次数更少。

JMP PRO “主题分析” 报表

“主题分析” 报表显示旋转后的每个主题中具有最大载荷的词条。还有其他显示旋转奇异值分解的成分的报表。

“按主题划分的前几位载荷” 报表显示每个主题的词条表。每个表中的词条是对每个主题而言具有最大绝对值载荷的词条。按载荷绝对值的降序对每个表排序。可以使用这些表来确定对应于每个主题的概念性主题。

“主题分析”报表还包含以下报表：

主题载荷 包含跨各词条的主题的载荷矩阵。该矩阵等价于旋转 PCA 中的因子载荷矩阵。

按主题划分的词云 包含词云矩阵，每个主题一个词云。

主题得分 包含每个主题的文档得分的矩阵。主题中具有更高得分的文档更可能与该主题关联。

主题得分图 包含一个“显示文本”按钮和每个文档的主题得分图。点击“显示文本”按钮将打开一个窗口，其中包含选定文档的文本。

“主题得分图”报表是“主题得分”报表中的矩阵的直观表示。图中的每个面板对应于某一主题或“文档得分”矩阵的某一列。在每个面板内，每个点对应于语料库中的某一文档或“主题得分”矩阵的某一行。

每个主题解释的方差 包含由每个主题解释的方差表。该表还包含每个主题解释的变异的百分比和累积百分比列。

旋转矩阵 包含最大方差法旋转的旋转矩阵。

JMP PRO “主题分析”报表选项

“文本分析器”平台中的“主题分析”红色小三角菜单包含以下选项：

主题散点图矩阵 显示或隐藏旋转奇异值分解向量的散点图矩阵。点击“显示文本”按钮将打开一个窗口，其中包含选定文档的文本。

显示选项 包含用于显示或隐藏“主题分析”报表中显示的内容的选项。请参见第 361 页的“‘主题分析’报表”。

重命名主题 允许您为一个或多个主题添加说明性名称。

保存文档主题向量 将旋转奇异值分解中的用户指定的奇异向量数作为列保存到数据表。

保存主题向量公式 将建模类型为“向量”的公式列（其中包含旋转奇异值分解）保存到数据表中。生成的列使用 `Text Score()` JSL 函数。有关该函数的详细信息，请参见“帮助”>“脚本索引”。

保存词条主题向量 将主题向量作为列保存到使用“保存词条表”选项创建的数据表。

删除 从 SVD 报表中删除“主题分析”报表。

JMP PRO 判别分析

判别分析根据文档词条矩阵 (DTM) 中的列预测组或类别中每个文档的成员关系。具体而言，判别分析预测每个文档归属哪一个响应列类别。选择 “判别分析”选项时，您必须选择一个包含类别或组的响应列。通过 GTM 列预测组成员关系。有关判别分析的详细信息，请参见 《多元方法》。

“文本分析器”平台中的判别分析方法是基于中心化 DTM 的奇异值分解。每一组响应列都有自己的组均值，用来对 DTM 进行中心化。“文本分析器”平台中的判别分析方法的执行速度比“判别分析”平台快，因为它利用了 DTM 的稀疏性。

JMP PRO “判别分析规格”窗口

“文本分析器”平台中的“判别分析”选项基于“文档词条矩阵”(DTM)。通过为词条列表中的每个词条（最大为指定的最大词条数）创建列来生成 DTM。每个文本文档（等价于数据表中的行）对应 DTM 的一行。DTM 单元格中的值取决于用户在“规格”窗口中指定的权重类型。

当您从“文本分析器”红色小三角菜单选择 “判别分析”选项时，将显示包含以下选项的 “规格”窗口：

最大词条数 包括在判别分析中的最大词条数。

最小词条频数 一个词条要包括在判别分析中必须出现的最小次数。

权重 用于确定进入文档词条矩阵单元格的值的权重方案。在第 356 页的 “ “文档词条矩阵规格”窗口”中介绍了各个权重方案选项。

奇异向量数 判别分析中的奇异向量数。默认值为文档数、词条数或 100 中的最小值。

JMP PRO “判别分析”报表

默认情况下，“文本分析器”平台中的“判别分析”报表包含两个打开的报表：“分类汇总”和“判别得分”。其他报表最初是关闭的。

“判别分析”报表包含以下报表：

词条均值 提供用在判别分析中的词条表。这些词条对应 DTM 列。该表包含每个词条在每个组中的均值，以及每个词条的总均值和加权标准差。

到各组的平方距离 提供一个表，其中包含到每个文档的每个组的 Mahalanobis 距离的平方。有关 Mahalanobis 距离的详细信息，请参见 《多元方法》。

各组概率 提供一个表，其中包含某个文档属于每组的概率。

分类汇总 提供汇总判别得分的报表。该报表对应于 “判别分析”平台报表中的 “得分汇总”报表。

判别得分 提供包含每个文档的预测类别和其他支持信息的表。该表对应于“判别分析”平台报表中的“判别得分”表。

JMP PRO “判别分析”报表选项

“判别分析”红色小三角菜单包含以下选项：

典型图 显示或隐藏典型空间中的文档和组均值图。典型空间是最能分隔各组的空间。若响应变量有两个以上水平，您必须指定典型坐标数。若指定了两个以上的典型坐标，该选项生成典型图矩阵。

保存概率 将概率列保存到每个响应水平的数据表以及包含最可能响应的列。“最可能的”响应列包含具有基于模型的最高概率的水平。

每个概率列给出某观测在该响应水平中的成员关系的后验概率。“响应概率”列属性保存至每个概率列。有关“响应概率”列属性的详细信息，请参见《使用 JMP》。

保存概率公式 将公式列保存至数据表以便预测最可能的响应。第一个保存的列包含使用 Text Score() 函数计算每个响应水平的概率的公式。还有一些列包含每个响应水平的概率，以及一个包含预测响应的列。

保存典型得分 将包含每个观测的典型空间得分的列保存至数据表。典型空间是最能分隔各组的空间。第 *k* 个典型得分的列名为**典型** <k>。

删除 从“文本分析器”报表窗口中删除“判别分析”报表。

JMP PRO 词条选择

词条选择用于标识哪些词条最适合解释不同响应。分析中使用“广义回归”平台对文档词条矩阵 (DTM) 执行变量选择，并标识对响应影响最大的词条。词条选择可以用于二元响应，类似于情感分析，以及其他类型的响应。拟合模型对指定的响应列使用适当的响应分布。

提示：有关“词条选择”的示例，请选择**帮助 > 样本数据库**，打开 Chips.jmp 并运行“文本分析器 - 词条选择”表脚本。

JMP PRO 词条选择设置

“设置”报表支持您选择响应列、指定响应的目标水平以及调整模型的设置。指定模型设置后，点击“运行”按钮可运行模型。拟合模型随后显示在“汇总”报表中。请参见第 366 页的“词条选择汇总报表”。

目标水平

选择某个响应列后，“目标水平”分级显示项随即显示。

- 对于名义型响应，选择响应的一个水平作为 logistic 回归模型中的目标水平；Logistic 回归模型中的响应是目标水平对比其他水平所有的组合。
- 对于有序型响应，所有响应水平最初都包含在模型中。使用本地数据过滤器，可以选择要从模型中排除的响应水平；所包含水平的底层数值采用正态响应分布建模。

注意：对于有序型响应，只有当响应列的数据类型为数值时才能拟合词条选择模型。

- 对于连续响应，使用本地数据过滤器直方图选择要从模型中排除的响应值；包含的值采用正态响应分布建模。
- 对于建模类型为“多重响应”的响应列，选择一个或多个响应水平作为二元 logistic 回归模型中的目标水平。若选择多个水平，且某个文档的“响应”列中存在所选的任何水平，则该文档属于目标水平。选择**使用 AND 合并**选项可要求所有选定的水平都显示在文档的“响应”列中，以便该文档包含在目标水平中。

模型设置

默认情况下，“广义回归”模型使用提前停止的“弹性网络”估计方法和 AICc 验证方法。您可以在“模型设置”分级显示项中更改这些设置。请参见《拟合线性模型》。

注意：若在“文本分析器”启动窗口中指定了“验证”列，则“词条选择”报表中的“广义回归”平台将使用“验证”列作为验证方法。

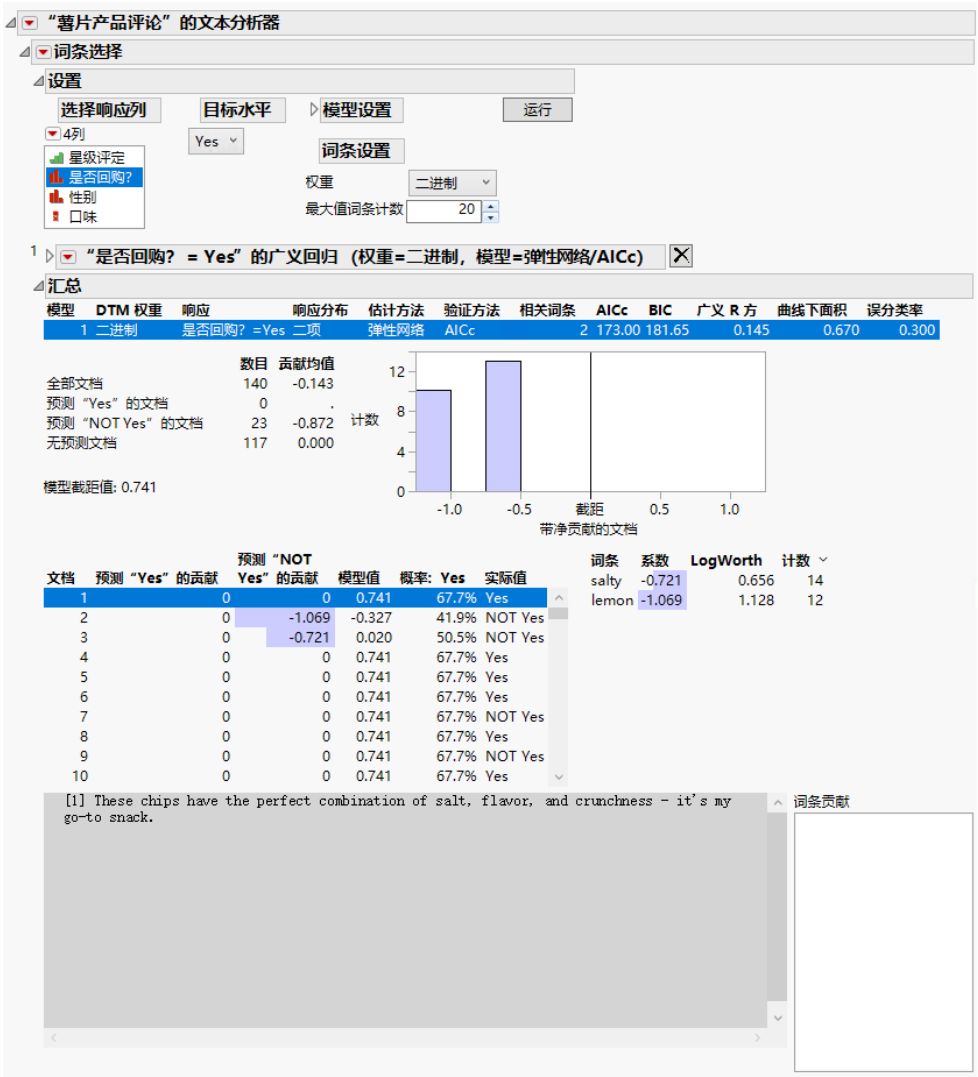
词条设置

“词条设置”定义回归模型中使用的文档词条矩阵 (DTM)。您可以更改加权方法以及 DTM 中包含的最大词条数；每个词条对应于 DTM 的一列。请注意，在语料库中出现次数少于 10 次的词条不包括在模型使用的 DTM 中。有关 DTM 选项的详细信息，请参见第 356 页的““文档词条矩阵规格”窗口”。

JMP PRO “词条选择”报表

运行分析后，“词条选择”报表由三个部分组成。“设置”报表包含用于指定分析的控件。请参见第 364 页的“词条选择设置”。在“设置”报表下面，对于您已经运行的每个分析，都有最初关闭的“广义回归”报表。请参见《拟合线性模型》。该报表的最后一部分是“汇总”报表。

图 12.12 “词条选择”报表



词条选择汇总报表

“汇总”报表包含一个“模型比较”表、一个“汇总”表及直方图、一个“文档得分”表、一个“词条得分”表和一个文本框。

“模型比较”表包含为每个拟合模型都包含一行。“汇总”报表的其余部分显示该表中当前选定模型的结果。

“汇总”表从总体上以及按模型响应的预测值来显示文档的计数和得分均值。“贡献均值”是“文档得分”表中贡献值的平均值。汇总直方图显示文档的总体贡献值的分布。该直方图是交互式的，因此您可以点击一个直条来突出显示“文档得分”表中的相应文档。

“文档得分”表显示每个文档的正贡献值和负贡献值，以及每个文档的预测值和实际值。对于二项响应模型，预测值是文档处于目标水平的概率；对于正态响应模型，预测值是来自每个文档的拟合模型的预测。若选择表中的一行，则相应文档的文本将显示在表下方的文本框中。

“词条得分”表列出了拟合模型所选择的每个词条、模型中该词条的系数、其 LogWorth 以及该词条在语料库中的出现次数。若选择表中的一行，则相应文档的文本将显示在表下方的文本框中。

文本框显示在“文档得分”表中选择的文档的文本或在“词条得分”表中选择的词条的上下文。

JMP PRO “词条选择”报表选项

“词条选择”红色小三角菜单包含以下选项：

保存文档得分 （仅当在“汇总”表中选择了分析时才可用。）将“文档得分”表中的列保存到数据表中的新列。新列包含正贡献和负贡献，以及每个文档的预测值。

保存词条得分 DTM （仅当在“汇总”表中选择了分析时才可用。）将当前所选分析中每个相关词条的列保存到数据表中。这些列包含每个文档的词条得分，该得分使用“词条选择”的“词条设置”中指定的权重。

保存预测公式 （仅当在“汇总”表中选择了分析时才可用。）将包含当前选定分析的预测公式的列保存到数据表中。

显示词条云 在“汇总”报表中显示或隐藏词云。词云显示当前所选分析中的系数词条。单词的大小由其系数的绝对值决定，颜色由其系数的符号决定。

删除 从“文本分析器”报表窗口中删除“词条选择”报表。

JMP PRO 情感分析

情感分析使用词法分析识别文档中的情感词条，并对文档的正面、负面和整体情感进行评分。该分析假设每个文档都是在单个主题上具有二元情感的自由文本。情感分析将基本自然语言处理 (NLP) 并入结果。有关自然语言处理的详细信息，请参见 https://opennlp.apache.org/。若不想使用 NLP，请取消选择“解析文档”选项。

提示：有关“情感分析”的示例，请选择**帮助 > 样本数据库**，打开 Chips.jmp 并运行“文本分析器 - 情感分析”表脚本。

注意：

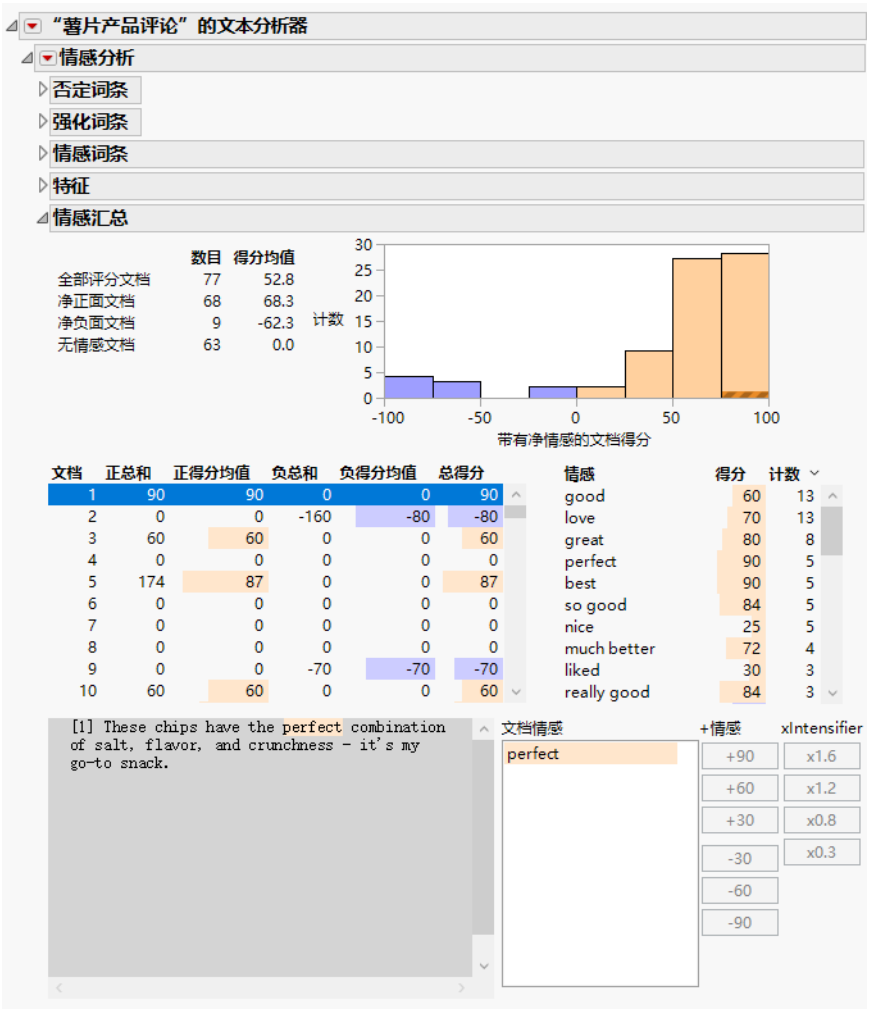
- 在情感分析中，一个词只能在一类词条中起作用：否定词条、强化词条或情感词条。
- 情感分析可以识别一些表情符号，或被视为单个单元的字符序列。您可以在“情感词条”报表或“管理情感词条”窗口中查看内置的表情符号及其默认情感得分。

- 若将某个词指定为否定词条、强化词条或情感词条，而该词已被指定为停止词，则只要“情感分析”报表处于打开状态，该词将被作为停止词暂时删除。这一暂时删除会影响整个“文本分析器”报表。当“情感分析”报表关闭时，它将恢复为停止词。

JMP PRO “情感分析”报表

默认情况下，“文本分析器”平台中的“情感分析”报表包含一个打开的报表：“情感汇总”。其他报表最初是关闭的。

图 12.13 “情感分析”报表



“情感分析”报表包含以下报表：

否定词条

包含当前情感分析中否定词条的列表。右击该列表可查看其他选项的菜单。您可以选择列表中的词条将其删除。

强化词条

包含强化词条及其相应的乘数值的列表。右击该列表可查看其他选项的菜单。您可以选择列表中的词条将其删除。

情感词条

包含情感词条及其相应的得分值的列表。该报表支持您添加新的情感词条。“可能的情感”表包含可以考虑添加为情感词条的词条的计数。要将某个词条添加为情感词条，请在“可能的情感”表中选择该词条，然后点击“+情感”下面的按钮之一。要选择未列出的情感得分值，可以在将其添加到情感词条列表后编辑该得分值。

在“可能的情感”表中选择某个词条时，包含该词条的文档将显示在“情感词条”报表的右侧。这为该词条在语料库中的用法提供了上下文。

特征

包含用于对语料库中的特征进行评分的选项。特征是情感词条所描述的内容。点击**搜索**按钮可生成可能的特征词条的列表。从“可能的特征”表中选择一个或多个词条时，包含这些词条的文档摘录将显示在表右侧的文本框中。点击**对选定特征评分**按钮可更新“情感汇总”报表，以显示对选定特征词条评分的结果。

注意：若选择“解析文档”选项，则当单词出现在与情感相同的伞形子句中时，“特征”报表会对这些词评分。

情感汇总

包含基于当前设置的情感分析结果。该报表包含一个“汇总”表及直方图、一个“文档得分”表、一个“情感词条”表、一个文本框，以及一个支持您添加更多情感和强化词条的控制面板。

汇总表显示了按文档的评分方式细分的文档的计数和得分均值。得分均值由“得分”选项的设置确定。请参见第 370 页的““情感分析”报表选项”。汇总直方图显示文档的总情感得分的分布。该直方图是交互式的，因此您可以点击一个直条来突出显示“文档得分”表中的相应文档。

“文档得分”表显示了正负情感得分总和和均值，以及每个文档的总情感得分。若选择表中的一行，则相应文档的文本将显示在表下方的文本框中。若指定“得分列”，则表中包含评分列中的值。

提示：您可以悬停在“文档得分”表中的单元格上方，以查看用于生成表结果的得分计算。

“情感词条”表列出每个情感词条、其得分值以及该词条在语料库中出现的次数。

提示：对于含有多个单词的情感词条，可以悬停在“得分”列中的单元格上方，以查看用于生成得分的计算。

文本框显示在“文档得分”表中选择的文档的文本或在“情感词条”表中选择的词条的上下文。在“文档得分”表中选择某个文档时，文本框右侧将显示该文档中的情感列表。

提示：当您悬停在文本框中被分类为否定词条、强化词条或情感词条的词条上方时，将出现一个框，其中显示该分类并包含一个“删除”按钮。点击**删除**按钮可从否定词条、强化词条或情感词条列表中快速删除该词条。

在文本框中选择某个词条后，控制面板将激活。要将某个词条添加为情感词条，请在文本框中选择该词条，然后点击“+情感”下面的按钮之一。要将某个词条添加为强化词条，请在文本框中选择该词条，然后点击“×强化”下面的按钮之一。

JMP PRO “情感分析”报表选项

“情感分析”红色小三角菜单包含以下选项：

评分 包含以下用于为文档计算总得分的选项：

统一尺度 将正负短语的得分加总。然后将总和除以文档中的短语数以确定总得分。

最小最大值 总得分计算为最大正分和最小负分之和。

得分列 指定一个数据表列，该列包含可与计算的情感进行比较的已知信息。得分列将添加至“文档得分”表。

提示：您可以通过直观地比较“总得分”列与得分列来评估情感得分。

解析文档 指定是否使用自然语言处理 (NLP) 来解析文档。有关自然语言处理的详细信息，请参见 https://opennlp.apache.org/。

保存文档得分 将“文档得分”表中的列保存到数据表中的新列。新列包含正负情感得分总和和均值，以及每个文档的总情感得分。

按文档保存情感得分计数 在数据表中为每个情感词条保存一列。每列包含每个文档中每个情感词条的出现次数。

显示否定词条 显示或隐藏“否定词条”报表。

显示强化词条 显示或隐藏“强化词条”报表。

显示情感词条 显示或隐藏“情感词条”报表。

显示特征查找工具 显示或隐藏“特征”报表。

显示情感云 在“情感汇总”报表中显示或隐藏情感词条的词云。

包含内置否定词条 指定情感分析中使用的否定词条是否包含内置否定词条。

包含内置强化词条 指定情感分析中使用的强化词条是否包含内置强化词条。

包含内置情感词条 指定情感分析中使用的情感词条是否包含内置情感词条。

管理否定词条 显示一个窗口，您可以在其中添加或删除否定词条。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的否定词条。请参见第 351 页的“词条选项管理窗口”。

管理强化词条 显示一个窗口，您可以在其中添加或删除强化词条。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的强化词条。请参见第 351 页的“词条选项管理窗口”。

管理情感词条 显示一个窗口，您可以在其中添加或删除情感词条。可以在“用户”、“列”和“本地”水平上应用所做更改。您还可以指定本地例外情况，用于排除在任何其他水平上指定的情感词条。请参见第 351 页的“词条选项管理窗口”。

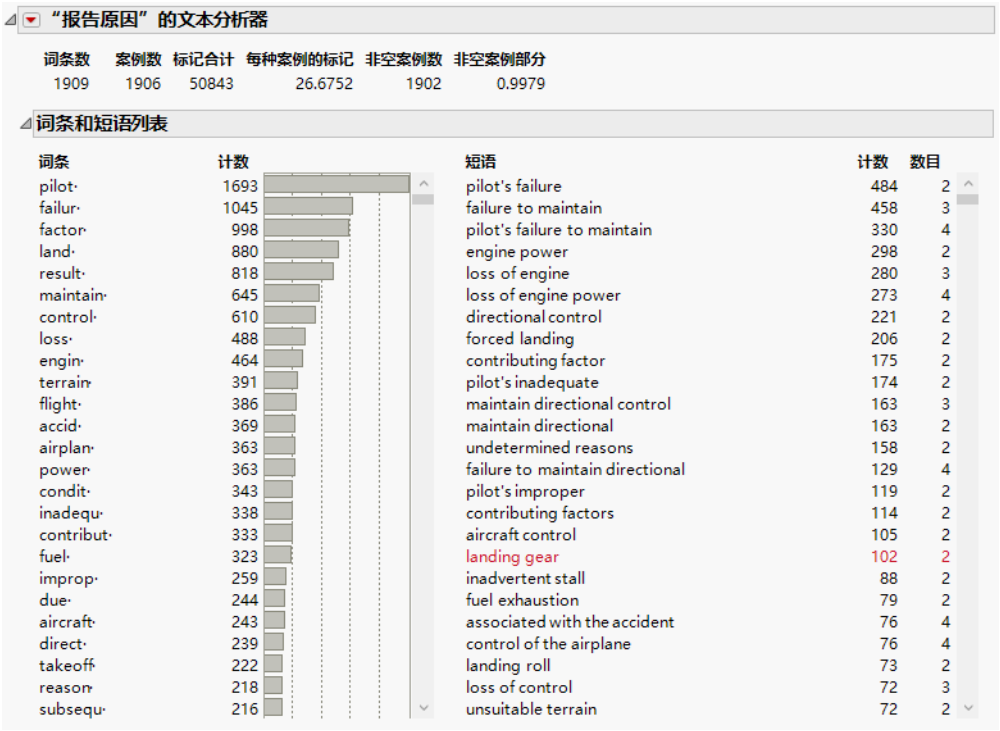
JMP PRO “文本分析器”平台的其他示例

该示例取自美国国家运输安全委员会发布的飞机事故报告，其中列出美国 2001 年发生的事故。您想探索一些文本，这些文本包含了对事故原因结果的描述。您还想在事故报告集合中找到主题。

1. 选择**帮助** > **样本数据库**，然后打开 Aircraft Incidents.jmp。
2. 选择**行** > **按列设定颜色或标记**。
3. 从“列”列表中选择**严重**，然后点击**确定**。

   将包含涉及死亡事故的行设定为红色。
4. 选择**分析** > **文本分析器**。
5. 从“选择列”列表中选择**报告原因**，然后点击**文本列**。
6. 从“语言”列表中，选择**简体中文**。
7. 从“词干处理”列表中，选择**处理所有词条的词干**。
8. 从“标记化”列表中，选择**基本单词**。
9. 点击**确定**。

图 12.14 报告原因的 “文本分析器”报表



从报表中，您看到有几乎 51,000 个标记和大约 1,900 个独特词条。

10. 右击 “词条列表”中的 pilot 并选择**选择行**。

    从数据表中的选定行数，您可以看到某种形式的“pilot”一词出现在超过 1,300 起事故报告中。

11. 右击 pilot，然后选择**添加停止词**。

    由于某种形式的 “pilot”一词出现的频率高于其他词条，这些词条提供不了很多信息来区分文档。所以，以 pilot 为词干的所有词条都添加至停止词列表。

JMP PRO 本例的其余步骤只能在 JMP Pro 中完成。

12. JMP PRO 点击 “报告原因的文本分析器”旁边的红色小三角并选择**潜在语义分析， SVD**。

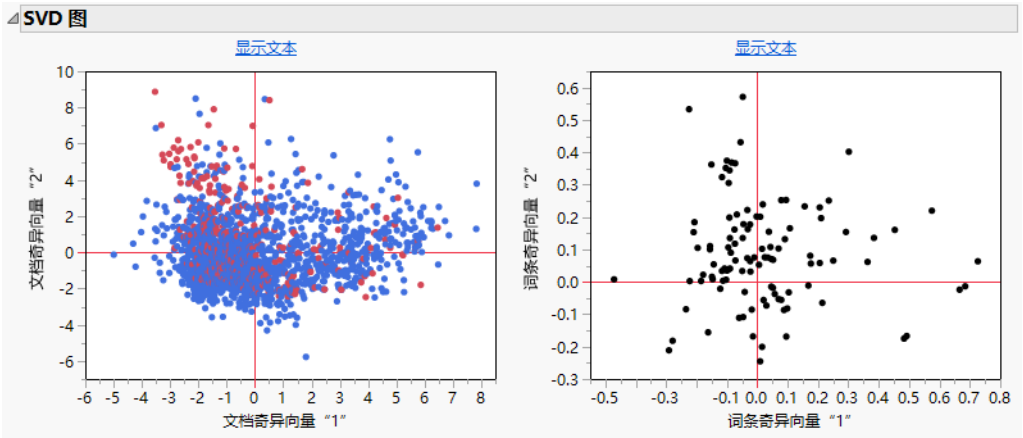
    这是迈向执行 SVD 旋转的主题分析的第一个分析步骤。

13. JMP PRO 在 “规格”窗口中，为 “最小词条频数”键入 50。

    因为大约有 51,000 个标记，该频数等价于至少代表所有词条的 0.1% 的一个词条。

14. JMP PRO 点击**确定**。

图 12.15 报告原因的 SVD 图



严重和非严重事故的文档 SVD 图区别不大。

15. JMP PRO 点击 “SVD 中心化和统一尺度的 TF IDF”旁边的红色小三角并选择**主题分析，旋转 SVD**。

    您想查找构成主题的词条组。

16. JMP PRO 为 “主题数”键入 5。
17. JMP PRO 点击**确定**。

图 12.16 按报告原因主题划分的前几位载荷

按主题划分的前几位载荷

主题 “1”		主题 “2”		主题 “3”		主题 “4”		主题 “5”	
词条	载荷	词条	载荷	词条	载荷	词条	载荷	词条	载荷
power	0.67567	altitud	0.48052	factor	0.5093	control	0.5125	fuel	0.4864
loss	0.66539	low	0.45137	condit	0.4677	direct	0.4957	personnel	0.4630
forc	0.62046	dark	0.42289	unsuit	0.3984	experi	0.4382	mainten	0.4546
engin	0.61866	night	0.40408	accid	0.3909	student	0.4273	result	0.4153
suitabl	0.58926	maintain	0.39283	select	0.3842	lack	0.3625	preflight	0.3930
lack	0.53292	instrument	0.39239	associ	0.3806	maintain	0.3616	exhaust	0.3663
reason	0.47828	clearanc	0.37881	area	0.3628	instructor	0.3559	inspect	0.3640
undetermin	0.46599	airspe	0.33612	compens	0.3352	supervis	0.3282	plan	0.3474
terrain	0.37013	condit	0.33473	failur	-0.3207	power	-0.3269	reason	-0.3468
total	0.31932	stall	0.33010	wind	0.3202	failur	0.3171	undetermin	-0.3438
land	0.29402	flight	0.32838	inadequ	0.3080	reason	-0.3142	improp	0.3389
		continu	0.31729	result	-0.2845	undetermin	-0.3101	inadequ	0.3353
		maneuv	0.30931	terrain	0.2663	crosswind	0.3085	subsequ	0.3246
		weather	0.29484	airspe	-0.2542	aircraft	0.3007	maintain	-0.3117
		adequ	0.28082			factor	0.2787	due	0.2882

具有最高载荷的每个主题的词条支持您解释该主题是否捕获了事故报告中的主题。

例如，主题 1 对于动力、失去和发动机具有高载荷，这表示发动机失去动力的主题是事故原因。这对应于短语 “发动机动力失去”在一组事故报告中出现了 273 次。

根据主题 2 中具有高载荷的单词，可以将它描述为与涉及黑暗或低海拔的事故有关。

在这个文本分析阶段，您对于如何进行分析有很多选择。文本分析是一个迭代过程，因此您可能通过添加停止词或指定短语，使用主题信息进一步审校您的词条列表。您可能保存加权的文档词条矩阵，将 SVD 或旋转 SVD 中的向量保存为数据表中的数值列，然后在其他 JMP 分析平台中使用它们。在其他平台中使用这些列时，还可以包括数据表中的其他列以进行深入分析。

附录 A

参考书目

Agresti, A. (1990).*Categorical Data Analysis*.New York: John Wiley & Sons.

Agresti, A., and Coull, B. A. (1998)."Approximate is Better Than 'Exact' for Interval Estimation of Binomial Proportions."*American Statistician* 52:119–126.

Asiribo, O., and Gurland, J. (1990)."Coping with Variance Heterogeneity."*Communication in Statistics: Theory and Methods* 19:4029–4048.

Bartlett, M. S., and Kendall, D. G. (1946)."The Statistical Analysis of Variance-Heterogeneity and the Logarithmic Transformation."*Supplement to the Journal of the Royal Statistical Society* 8:128–138.

Bissell, A. F. (1990)."How Reliable is Your Capability Index?"*Applied Statistics* 30:331–340.

Brown, M. B., and Benedetti, J. K. (1977)."Sampling Behavior of Tests for Correlation in Two-Way Contingency Tables."*Journal of the American Statistical Association* 72:305–315.

Brown, M. B., and Forsythe, A. B. (1974)."Robust Tests for Equality of Variances."*Journal of the American Statistical Association* 69:364–367.

Chen, S.-X., and Hall, P. (1993)."Empirical Likelihood Confidence Intervals for Quantiles."*The Annals of Statistics* 21:1166–1181.

Chou, Y.-M., Owen, D. B., and Borrego, S. A. (1990)."Lower Confidence Limits on Process Capability Indices."*Journal of Quality Technology* 22:223–229.

Cleveland, W. S. (1979)."Robust Locally Weighted Regression and Smoothing Scatterplots."*Journal of the American Statistical Association* 74:829–836.

Cohen, J. (1960)."A Coefficient of Agreement for Nominal Scales."*Education Psychological Measurement* 20:37–46.

Conover, W. J. (1972)."A Kolmogorov Goodness-of-fit Test for Discontinuous Distributions."*Journal of the American Statistical Association* 67:591–596.

Conover, W. J. (1980).*Practical Nonparametric Statistics*.New York: John Wiley & Sons.

Conover, W. J. (1999).*Practical Nonparametric Statistics*.3rd ed. New York: John Wiley & Sons.

Cureton, E. E. (1967)."The Normal Approximation to the Signed-Rank Sampling Distribution when Zero Differences are Present."*Journal of the American Statistical Association* 62:1068–1069.

DeLong, E. R., DeLong, D. M., and Clarke-Pearson, D. L. (1988)."Comparing the Areas under Two or More Correlated Receiver Operating Characteristic Curves: A Nonparametric Approach."*Biometrics* 44:837–845.

Devore, J. L. (1995).*Probability and Statistics for Engineering and the Sciences*.Pacific Grove, CA: Duxbury Press.

Dunn, O. J. (1964).“Multiple Comparisons Using Rank Sums.”*Technometrics* 6:241–252.

Dunnett, C. W. (1955).“A Multiple Comparisons Procedure for Comparing Several Treatments with a Control.”*Journal of the American Statistical Association* 50:1096–1121.

Efron, B. (1981).“Nonparametric Standard Errors and Confidence Intervals.”*The Canadian Journal of Statistics* 9:139–158.

Eubank, R. L. (1999).*Nonparametric Regression and Spline Smoothing*.2nd ed. Boca Raton, Florida: CRC.

Fleiss, J. L., Cohen, J., and Everitt, B. S. (1969).“Large-Sample Standard Errors of Kappa and Weighted Kappa.”*Psychological Bulletin* 72:323–327.

Friendly, M. (1994).“Mosaic Displays for Multi-Way Contingency Tables.”*Journal of the American Statistical Association* 89:190–200.

Goodman, L. A., and Kruskal, W. H. (1979).*Measures of Association for Cross Classification*.New York: Springer-Verlag.

Gupta, S. S. (1965).“On Some Multiple Decision (Selection and Ranking) Rules.*Technometrics* 7:225–245.

Hajek, J. (1969). *A Course in Nonparametric Statistics*.San Francisco: Holden-Day.

Hartigan, J. A., and Kleiner, B. (1981).“Mosaics for Contingency Tables.”In *Computer Science and Statistics: Proceedings of the Thirteenth Symposium on the Interface*, edited by W. F. Eddy, 268–273.New York: Springer-Verlag.

Hayter, A. J. (1984).“A Proof of the Conjecture That the Tukey-Kramer Method Is Conservative.”*Annals of Mathematical Statistics* 12: 61–75.

Hosmer, D. W., and Lemeshow, S. (1989).*Applied Logistic Regression*.New York: John Wiley & Sons.

Howell, D. C. (2013).*Statistical Methods for Psychology*.8th ed. Belmont, California: Wadsworth.

Hsu, J. (1981).“Simultaneous Confidence Intervals for All Distances from the ‘Best’.”*Annals of Statistics* 9:1026–1034.

Hsu, J. C. (1996).*Multiple Comparisons: Theory and Methods*.London: Chapman and Hall.

Huber, P. J. (1973).“Robust Regression: Asymptotics, Conjecture, and Monte Carlo.”*Annals of Statistics* 1:799–821.

Huber, P. J., and Ronchetti, E. M. (2009).*Robust Statistics*.2nd ed. New York: John Wiley & Sons.

Iman, R. L. (1974).“Use of a t-statistic as an Approximation to the Exact Distribution of Wilcoxon Signed Ranks Test Statistic.”*Communications in Statistics—Simulation and Computation* 3:795–806.

Jones, M. C., and Pewsey, A. (2009).“Sinh-Arcsinh Distributions.”*Biometrika* 96:761–780.

Kendall, M., and Stuart, A. (1979).*The Advanced Theory of Statistics*.4th ed. Vol. 2.New York: Macmillan.

Keuls, M. (1952).“The Use of the ‘Studentized Range’ in Connection with an Analysis of Variance.”*Euphytica* 1.2:112–122.

Kramer, C. Y. (1956).“Extension of Multiple Range Tests to Group Means with Unequal Numbers of Replications.”*Biometrics* 12:307–310.

Lehmann, E. L., and D’Abrera, H. J. M. (2006).*Nonparametrics: Statistical Methods Based on Ranks*.Rev. ed. San Francisco: Holden-Day.

Levene, H. (1960).“Robust Tests for the Equality of Variance.”In *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, edited by I. Olkin, S. G. Ghurye, W. Hoeffding, W. G. Madow, and H. B. Mann.Palo Alto, CA: Stanford University Press.

McCullagh, P., and Nelder, J. A. (1989).*Generalized Linear Models*.London: Chapman and Hall.

Meeker, W. Q., and Escobar, L. A. (1998).*Statistical Methods for Reliability Data*.New York: John Wiley & Sons.

Meeker, W. Q., Hahn, G. J., and Escobar, L. A. (2017).*Statistical Intervals: A Guide for Practitioners and Researchers*.2nd ed. New York: John Wiley & Sons.

Miller, A. J. (1972).“Letter to the Editor.”*Technometrics* 14:507.

Nagelkerke, N. J. D. (1991).“A Note on a General Definition of the Coefficient of Determination.”*Biometrika* 78:691–692.

Nelson, P. R., Wludyka, P. S., and Copeland, K. A. F. (2005).*The Analysis of Means: A Graphical Method for Comparing Means, Rates, and Proportions*.Philadelphia: Society for Industrial and Applied Mathematics.

Neter, J., Wasserman, W., and Kutner, M. H. (1996).*Applied Linear Statistical Models*.4th ed. Boston: Irwin.

O’Brien, R. G. (1979).“A General ANOVA Method for Robust Tests of Additive Models for Variances.”*Journal of the American Statistical Association* 74:877–880.

O'Brien, R., and Lohr, V. (1984).“Power Analysis For Linear Models: The Time Has Come.”*Proceedings of the Ninth Annual SAS User’s Group International Conference*, 840–846.Cary, NC: SAS Institute Inc.

Olejnik, S. F., and Algina, J. (1987).“Type I Error Rates and Power Estimates of Selected Parametric and Nonparametric Tests of Scale.”*Journal of Educational Statistics* 12:45–61.

Pratt, J. W. (1959).“Remarks on Zeros and Ties in the Wilcoxon Signed Rank Procedures.”*Journal of the American Statistical Association* 54:655–667.

Reinsch, C. H. (1967).“Smoothing by Spline Functions.”*Numerische Mathematik* 10:177–183.

Rousseeuw, P. J., and Leroy, A. M. (1987).*Robust Regression and Outlier Detection*.New York: John Wiley & Sons.

Rubin, D. (1981).“The Bayesian Bootstrap.”*The Annals of Statistics* 9:130–134.

SAS Institute Inc.(2020a).“Introduction to Nonparametric Analysis.”In *SAS/STAT 15.2 User’s Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/intronpar.pdf.

SAS Institute Inc.(2020b).“The FREQ Procedure.”In *SAS/STAT 15.2 User’s Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/freq.pdf.

Schuirmann, D. J. (1987).“A Comparison of the Two One-Sided Tests Procedure and the Power Approach for Assessing the Equivalence of Average Bioavailability.”*Journal of Pharmacokinetics and Biopharmaceutics* 15:657–680.

Slifker, J. F., and Shapiro, S. S. (1980).“The Johnson System: Selection and Parameter Estimation.”*Technometrics* 22:239–246.

Snedecor, G. W., and Cochran, W. G. (1980).*Statistical Methods*.7th ed. Ames, Iowa: Iowa State University Press.

Somers, R. H. (1962).“A New Asymmetric Measure of Association for Ordinal Variables.”*American Sociological Review* 27:799–811.

Stephens, M. A. (1974).“EDF Statistics for Goodness of Fit and Some Comparisons.”*Journal of the American Statistical Association* 69:730–737.

Tan, C. Y., and Iglewicz, B. (1999).“Measurement-Methods Comparisons and Linear Statistical Relationship.”*Technometrics* 41:192–201.

Tamhane, A. C., and Dunlop, D. D. (2000).*Statistics and Data Analysis*.Englewood Cliffs, NJ: Prentice-Hall.

Tukey, J. W. (1953).“The Problem of Multiple Comparisons.”In *Multiple Comparisons, 1948–1983*, edited by H. I. Braun, vol. 8 of *The Collected Works of John W. Tukey* (published 1994), 1–300.London: Chapman & Hall.Unpublished manuscript.

Welch, B. L. (1951).“On the Comparison of Several Mean Values: An Alternative Approach.”*Biometrika* 38:330–336.

Wheeler, D. J. (2003).*Range Based Analysis of Means*.Knoxville, TN: SPC Press.

Wilson, E. B. (1927).“Probable Inference, the Law of Succession, and Statistical Inference.”*Journal of the American Statistical Association* 22:209–212.

Wludyka, P. S., and Nelson, P. R. (1997).“An Analysis-of-Means-Type Test for Variances From Normal Populations.”*Technometrics* 39:274–285.

附录 B

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR

ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

基本绘图

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020–2021. JMP® 16 基本绘图. Cary, NC: SAS Institute Inc.

JMP® 16 基本绘图

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

March 2021

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

基本绘图

1 了解 JMP 11

文档和其他资源

JMP 文档中的格式规范 13

JMP 帮助 14

JMP 文档库 14

学习 JMP 的其他资源 20

JMP 教程 20

样本数据表 20

学习统计和 JSL 术语 21

学习 JMP 的提示与技巧 21

JMP 工具提示 21

JMP 用户社区 21

免费在线统计思维课程 21

JMP 新用户欢迎套件 22

统计学知识门户 22

JMP 培训 22

用户编写的 JMP 手册 22

“JMP 起始页”窗口 22

JMP 技术支持 22

2 交互式绘图介绍 23

数据可视化概述

3 图形生成器 25

交互式数据可视化

如何使用“图形生成器” 28

关于“图形生成器”窗口 28

图形生成器中的功能的示例 30

移动或删除区域中的变量 37
使用轴 37
合并公共轴上的变量 38
在 X 和 Y 区域中创建不同的轴 39
创建另一个 Y 轴 40
为字符变量创建嵌套轴 42
对分类变量水平排序 45
元素类型和选项 48
点 51
平滑线 52
拟合线 53
椭圆 54
等高线 55
线 57
条形图 59
面积图 68
箱线图 72
直方图 74
热图 76
饼图 78
矩形树图 80
马赛克图 82
说明文字框 83
公式 85
地图形状 86
平行 87
图形区域 90
“X 分组”或“Y 分组”区域中的分类和连续变量 93
移动 X 分组和 Y 分组标签 93
红色小三角选项 94
“图形生成器”按钮 95
右击菜单 95
图形选项 96
分组区域选项 96

用于“变量”面板中的列的选项 97
用于轴、变量标签或图形标题的选项 97
用于多个图形之间的分割线的选项 98
图例选项 98

4 图形生成器示例 103

交互式数据可视化

有序条形图示例 105
面积图和线图示例 108
地图示例 112
蝴蝶图示例 115
散点图示例 119
堆叠填充的条形图示例 124
叠加直方图和脊线图的示例 125
弹道条形图示例 127
预测图示例 129

5 气泡图 131

使用气泡图查看多维数据中的模式

动态气泡图的示例 133
启动“气泡图”平台 135
指定两个 ID 变量 136
指定“时间”变量 136
与气泡图交互 136
控制动态气泡图的动画播放 136
选择气泡 138
使用笔刷工具 138
“气泡图”平台选项 138
显示角色 140
“气泡图”平台的更多示例 142
仅指定一个“时间”变量的示例 142
仅指定 ID 变量并且拆分气泡的示例 144
静态气泡图的示例 146
包含分类 Y 变量的气泡图示例 150

6 散点图矩阵 153
同时查看多个二元关系
散点图矩阵的示例 155
启动“散点图矩阵”平台 156
更改矩阵格式 158
“散点图矩阵”窗口 158
“散点图矩阵”平台选项 159
使用分组变量的示例 160
创建分组变量 161

7 平行图 163
通过绘制平行坐标来查看多维数据中的模式
平行图示例 165
启动“平行图”平台 166
平行图 167
解释平行图 167
“平行图”平台选项 169
“平行图”平台的更多示例 169
检查鸢尾花测量值 170
检查学生测量值 171

8 方格图 173
查看变量的颜色强度图
方格图的示例 175
启动“方格图”平台 176
方格图 176
“方格图”平台选项 177
方格图的右击菜单 178
“方格图”平台的更多示例 179

9 三维散点图 181
创建数据的旋转三维视图
三维散点图示例 183
启动“三维散点图”平台 184

“三维散点图”报表 185
旋转三维散点图 186
更改轴上的变量 187
调整轴 187
为数据点指定颜色和标记 187
在数据表中指定颜色和标记 188
“三维散点图”平台选项 188
正态等高线椭圆 190
非参数密度等高线 190
弹出菜单 192
“三维散点图”平台的更多示例 194
未分组正态等高线椭圆的示例 194
已分组正态等高线椭圆的示例 195
已分组非参数密度等高线的示例 196

10 等高线图 199
在二维中查看多维关系

等高线图的示例 201
使用“等高线”平台的示例 201
图形生成器中等高线图的示例 202
启动“等高线图”平台 202
等高线图 204
“等高线图”平台选项 204
填充区域 205
等高线规格 206
等高线图保存选项 210
使用公式指定等高线 211
等高线图的更多示例 211

11 三元图 215
查看表示构成或混合数据的图

三元图的示例 217
启动“三元图”平台 219
三元图 220
混料和约束 221

“三元图”平台选项 221
“三元图”平台的更多示例 221
使用混料约束的示例 222
使用等高线函数的示例 222

12 地图 225
添加地图或定制形状以增强数据可视化效果

地图绘制概述 227
在“图形生成器”中创建地图的示例 227
图形生成器 229
地图形状 229
颜色 231
大小 232
定制地图文件 235
背景地图 239
地图中的图像 240
边界 245
创建地图的示例 246
路易安娜行政区域示例 246
飓风跟踪示例 252
办公室温度研究 257
包含剪切元素的边界地图 262

A 传统图形平台 267
矩形树图 267
图表 268
叠加图 268

B 参考书目 269

C 技术许可声明 271

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 13
JMP 帮助 14
JMP 文档库 14
学习 JMP 的其他资源 20
JMP 教程 20
样本数据表 20
学习统计和 JSL 术语 21
学习 JMP 的提示与技巧 21
JMP 工具提示 21
JMP 用户社区 21
免费在线统计思维课程 21
JMP 新用户欢迎套件 22
统计学知识门户 22
JMP 培训 22
用户编写的 JMP 手册 22
“JMP 起始页”窗口 22
JMP 技术支持 22

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 14 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

交互式绘图介绍

数据可视化概述

《基本绘图》介绍您可用来直观演示数据的所有不同的图形和元素：

- “图形生成器”可采用交互方式创建许多不同种类的图形。请参见第 3 章，“图形生成器”和第 4 章，“图形生成器示例”。
- 气泡图用于创建将点表示为圆圈（或气泡）的散点图。气泡图可以是动态的（随时间变化播放动画），也可以是静态的（不能移动的固定气泡）。请参见第 5 章，“气泡图”。
- 散点图矩阵显示有序的二元图形的集合。请参见第 6 章，“散点图矩阵”。
- 平行图用于绘制表示数据表中各行的相连线段。请参见第 7 章，“平行图”。
- 方格图用于绘制方格的矩形阵列，其中每个方格对应一个数据表条目。请参见第 8 章，“方格图”。
- 三维散点图将相关数据表中数值列的值显示在可旋转的三维视图中。请参见第 9 章，“三维散点图”。
- 等高线图用于在直角坐标系中构造某个响应的等高线。请参见第 10 章，“等高线图”。
- 三元图显示由三部分组成的数据的分布和变异性。请参见第 11 章，“三元图”。
- 地图可用在“图形生成器”中，也可把地图作为背景用在其他平台中。请参见第 12 章，“地图”。

第 3 章

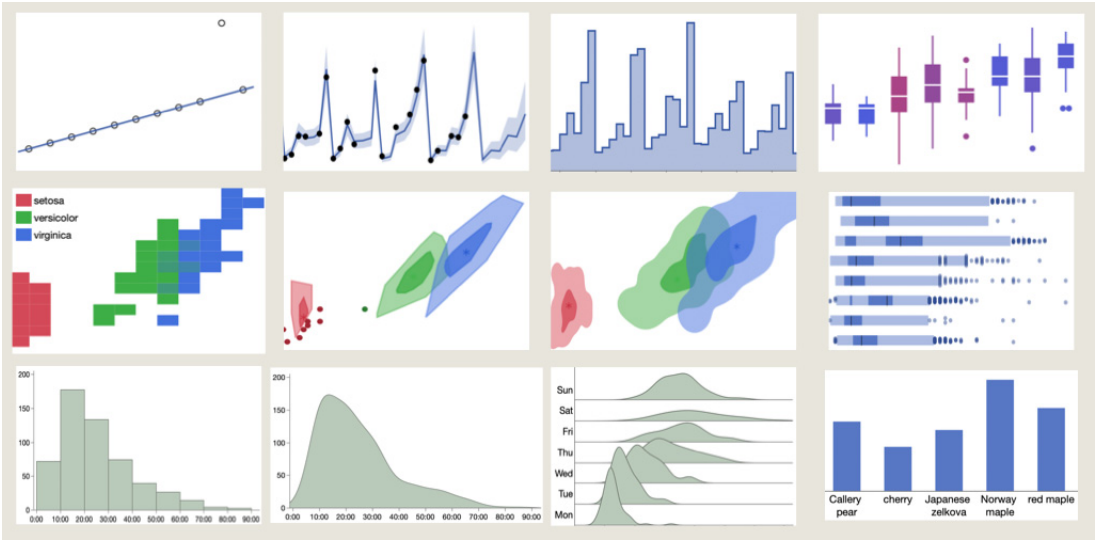
图形生成器

交互式数据可视化

使用“图形生成器”可采用交互方式探索您的数据。您可以快速创建和试验各种图，直到找到所需的图。然后，您可与他人共享结果。

本章为您演示如何使用“图形生成器”。有关使用“图形生成器”的详细示例，请参见第 103 页的“图形生成器示例”。

图 3.1 在 “图形生成器”中创建的图形



目录

如何使用“图形生成器” 28
关于“图形生成器”窗口 28
图形生成器中的功能的示例 30
移动或删除区域中的变量 37
使用轴 37
合并公共轴上的变量 38
在 X 和 Y 区域中创建不同的轴 39
创建另一个 Y 轴 40
为字符变量创建嵌套轴 42
对分类变量水平排序 45
元素类型和选项 48
点 51
平滑线 52
拟合线 53
椭圆 54
等高线 55
线 57
条形图 59
面积图 68
箱线图 72
直方图 74
热图 76
饼图 78
矩形树图 80
马赛克图 82
说明文字框 83
公式 85
地图形状 86
平行 87
图形区域 90
“X 分组”或“Y 分组”区域中的分类和连续变量 93
移动 X 分组和 Y 分组标签 93
红色小三角选项 94
“图形生成器”按钮 95
右击菜单 95
图形选项 96
分组区域选项 96

用于“变量”面板中的列的选项 97
用于轴、变量标签或图形标题的选项 97
用于多个图形之间的分割线的选项 98
图例选项 98

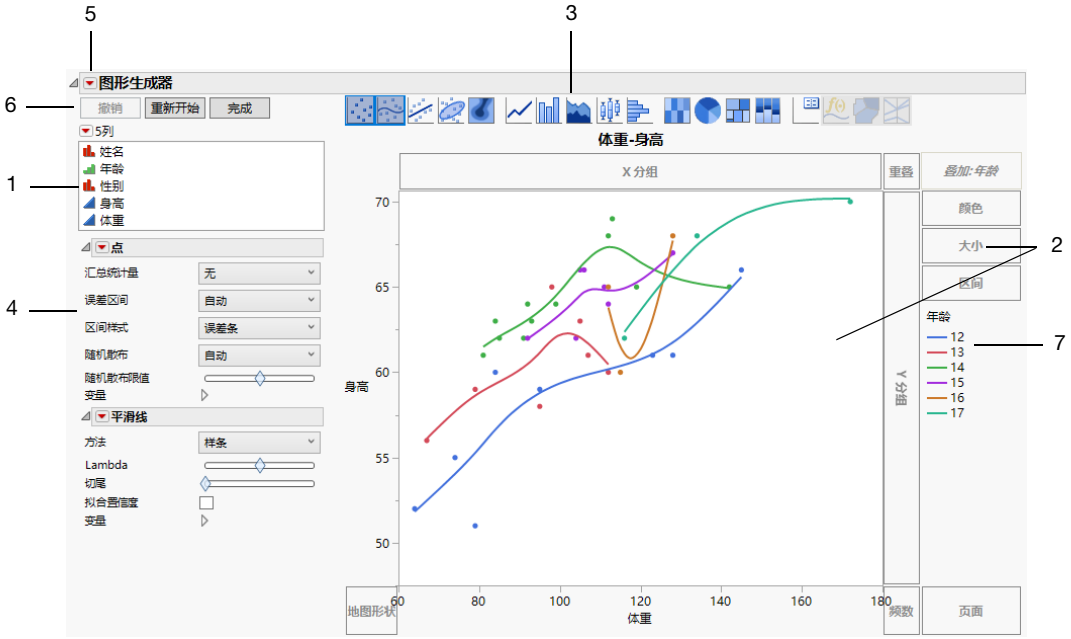
如何使用 “图形生成器”

1. 打开包含您要绘制的数据的数据表。
2. 选择**图形** > **图形生成器**。
3. 将 “变量” 列表中的列拖入各个拖放区。请参见第 37 页的 “移动或删除区域中的变量” 和第 90 页的 “图形区域”。
4. 点击元素类型图标，选择不同类型的图形或元素。请参见第 48 页的 “元素类型和选项”。
5. （可选）定制选定元素类型。请参见第 48 页的 “元素类型和选项”。
6. （可选）定制图例。请参见第 98 页的 “图例选项”。
7. 若对图形满意，则点击**完成**。

关于 “图形生成器” 窗口

要启动 “图形生成器”，请打开数据表，然后选择**图形** > **图形生成器**。“变量” 列表包含您数据表中的列，您可以将这些列拖入拖放区。

图 3.2 Big Class.jmp 的 “图形生成器” 窗口



数字	说明
1	数据表中的列列在“变量”面板中。将“列”列表中的列拖入拖放区，或选择“变量”列表中的列并点击空的拖放区。 您可以创建虚拟列，只需右击“变量”列表中的列名并从提供的选项中进行选择即可。 • 变换列以斜体显示在列表底部。 • 您可以将该列拖入“图形生成器”区域。 • 右击变换列可访问更多选项。 **提示：**有关“变量”红色小三角菜单中的列选项的说明，请参见《使用 JMP》。
2	图形区域和拖放区是“图形生成器”的主体。您的图形显示在中间，您可以将变量拖入不同的拖放区。请参见第 90 页的“图形区域”。 **注意：**所有排除的行在图形中均隐藏。
3	点击元素类型图标可选择您需要的图形类型。您可以同时对数据应用若干类型的元素（使用 Shift 键）。请参见第 48 页的“元素类型和选项”。
4	使用针对您选定的图形类型的特定选项定制图形。请参见第 48 页的“元素类型和选项”。
5	定制“图形生成器”窗口。请参见第 94 页的“红色小三角选项”。
6	撤销操作，重新开始等操作。请参见第 95 页的“ “图形生成器”按钮”。
7	若图形有图例，则会显示在这里。要定制图例，请参见第 98 页的“图例选项”。

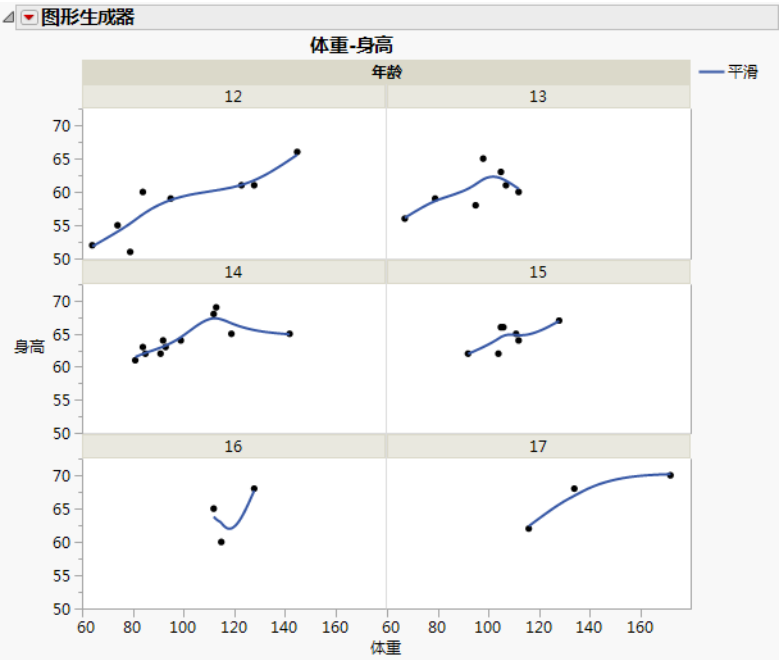
要重新创建和修改图 3.2 中的图形，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**身高**并将其拖至 Y 区域。
4. 选择**体重**并将其拖至 X 区域。
5. 选择**年龄**并将其拖至**叠加**区域。

   随即显示图 3.2 中的图。每个**年龄**值的**身高**和**体重**之间的关系都使用颜色显示在一个图形中。

   现在您想要查看每个**年龄**值的单独图形。
6. 在**叠加**区域中选择**年龄**，然后将其拖至**重叠**区域。

图 3.3 按年龄分组的身高和体重



对于每个**年龄**值，**身高**和**体重**之间的关系都显示在单独的图形中。

图形生成器中的功能的示例

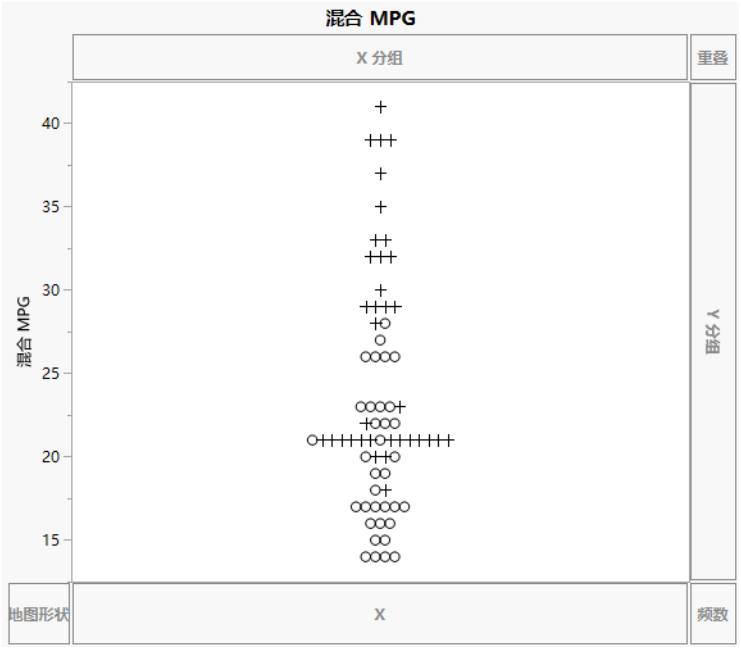
下例使用 “图形生成器”中的一些功能引导您执行操作。本例使用针对混合和非混合动力汽车收集的节能数据。您想要了解每加仑英里数 （MPG，即一种节能指标），并想要查看哪些因子可能影响 MPG。本例为您演示如何标绘若干相关因子：

- 第 30 页的 “了解混合每加仑英里数的概况”
- 第 32 页的 “标绘按发动机和驾驶类型分层的里程数”
- 第 35 页的 “按发动机类型找出高速路上的 MPG 与市区中的 MPG 之间的关系”
- 第 35 页的 “按发动机类型找出高速路上的 MPG 与市区中的 MPG 之间的关系”

了解混合每加仑英里数的概况

1. 选择**帮助** > **样本数据库**，然后打开 Hybrid Fuel Economy.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**混合 MPG** （混合每加仑英里数）并将其拖至 **Y** 区域。

图 3.4 混合 MPG 的初始图形



由于为数据表中的行分配了标记，所以用这些标记来表示各点。每行的**混合 MPG** 值都标绘在垂直轴上其对应值的位置。为避免具有相同的**混合 MPG** 值的观测点的重叠，JMP 自动随机散布各点。

您可以通过标绘直方图更清晰地了解分布。

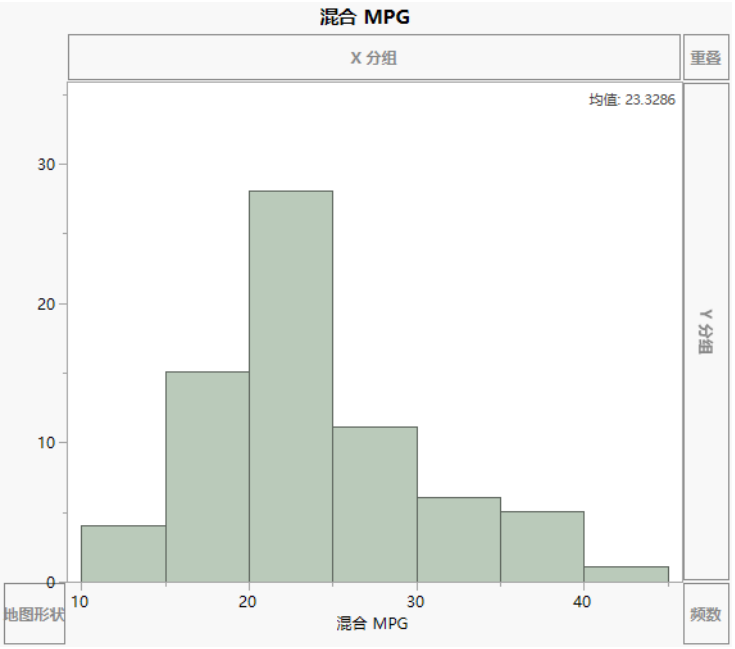
4. 点击直方图图标 。

   您想要竖向直方图，所以您需要将**混合 MPG** 移至 X 区域。

5. 悬停在 Y 轴中的**混合 MPG** 上方。光标变为手形光标。选择**混合 MPG** 并将其拖入 X 区域。

6. 点击说明文字框图标 以便在图形上显示 “混合 MPG” 均值。

图 3.5 混合 MPG 的直方图



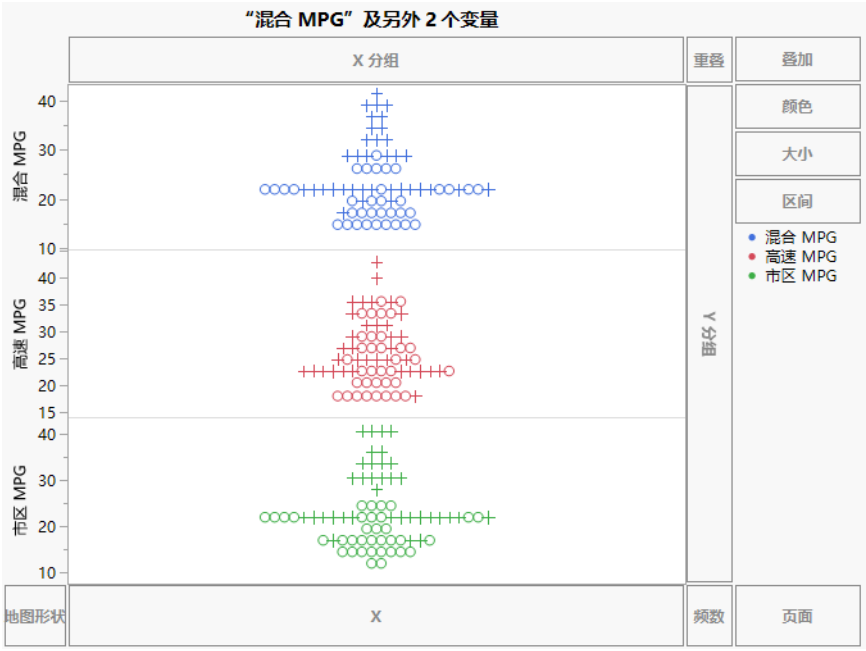
直方图显示“混合 MPG”的范围可较广，从大约 10 到 45。不过，很多车辆的里程数都介于 15 到 30 之间。混合 MPG 均值为 23.3286。

标绘按发动机和驾驶类型分层的里程数

1. 点击**重新开始**。
2. 选择**市区 MPG**、**高速 MPG** 和**混合 MPG**。
3. 悬停在 Y 区域上方，按住 Shift 键并在 Y 区域中点击。

   这将为**市区 MPG**、**高速 MPG** 和**混合 MPG** 创建单独的 Y 轴。

图 3.6 使用单独 Y 区域轴的 MPG 变量

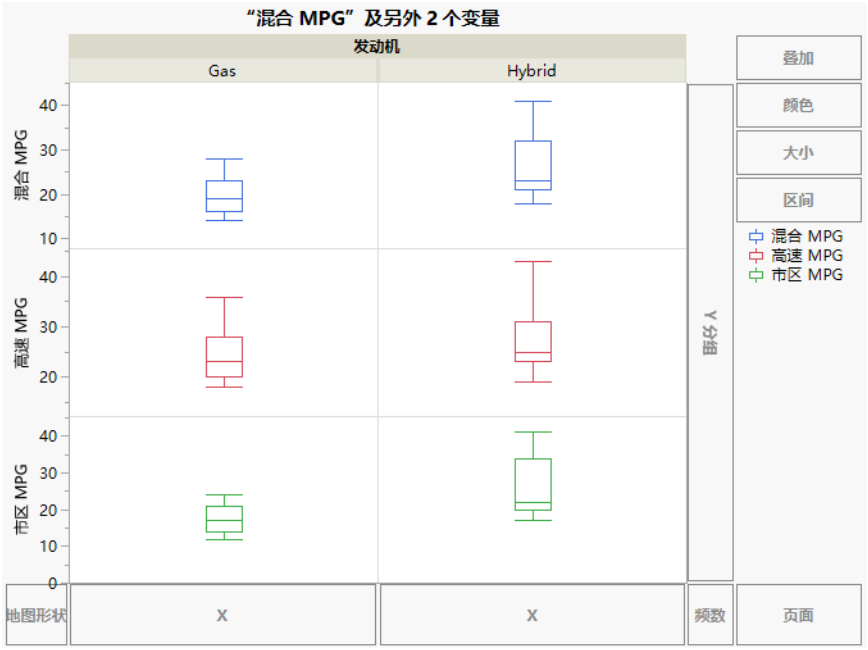


在右侧的图例中，根据驾驶类型对标记着色。

提示：要更改为驾驶类型分配的颜色，请右击图例中的标记。

4. 将**发动机**拖至 “X 分组”区域。
5. 点击箱线图图标 。

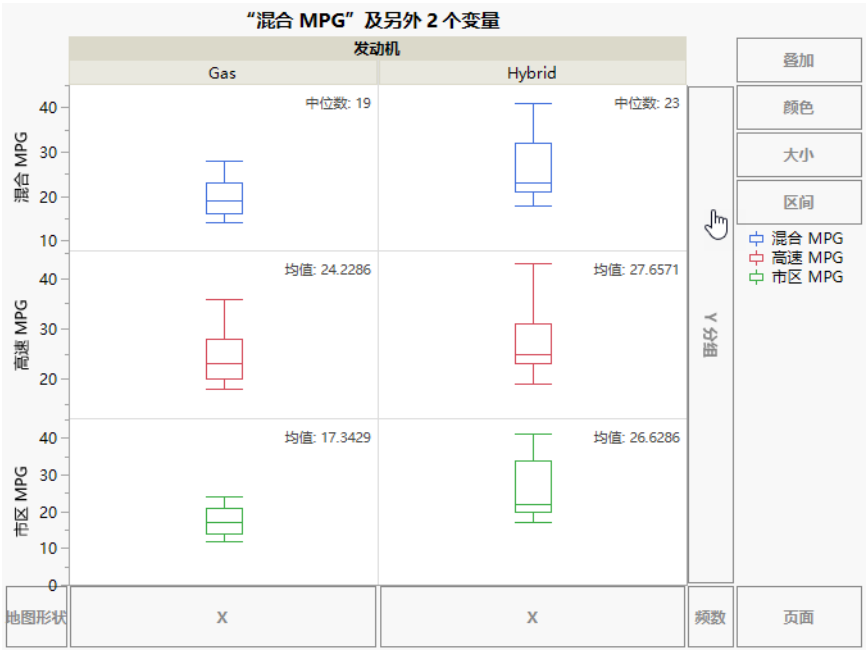
图 3.7 按发动机类型划分 MPG 变量的箱线图



箱线图以极为紧凑的格式显示分布。您可以看到在研究中，比起混合动力汽车，全部三个类型的 MPG 值对于天然气汽车要低很多。从两个**市区** MPG 箱线图的中心中的水平线，您可以估计中位数差值约为每加仑五英里。不过，您可以使用 “说明文字”元素进一步标明。

6. 点击说明文字框图标 。
7. 在图左侧的 “说明文字框”选项中，从**混合** MPG、**高速** MPG 和**市区** MPG 的 “汇总统计量”列表中选择**中位数**。

图 3.8 带说明文字的 MPG 变量的箱线图

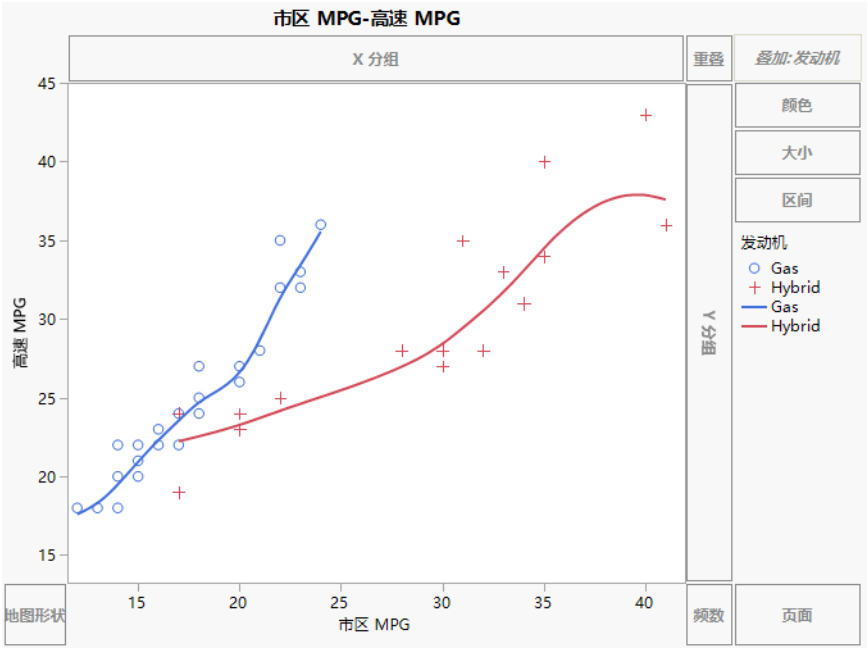


天然气发动机的市区 MPG 中位数为 17，混合发动机的市区 MPG 中位数为 22。

按发动机类型找出高速路上的 MPG 与市区中的 MPG 之间的关系

1. 点击**重新开始**。
2. 选择**市区 MPG** 并将其拖入 X 区域。
3. 选择**高速 MPG** 并将其拖入 Y 区域。
4. 选择**发动机**并将其拖至**叠加**区域。

图 3.9 按发动机类型划分的高速 MPG 和市区 MPG



为每种类型的发动机 （天然气和混合）显示一条平滑线。

提示：要更改线条属性 （颜色、宽度等），请右击图例中的某条线。

5. 点击拟合线图标 。

   对于每种发动机，将删除平滑线，并且添加最小二乘线和预测均值的置信带。

   提示：要添加 y = x 线，请在图形中右击并选择**定制** > + > **模板** > **Y 函数**。将 `_function_of_x_` 替换为 `y = x`，然后点击**确定**。

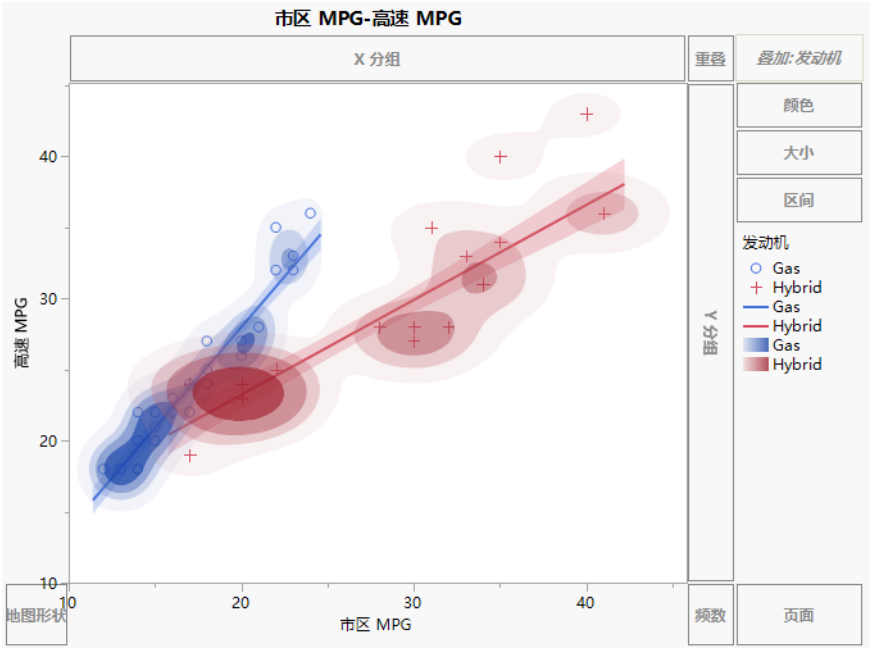
6. 拖动具有最小的**市区 MPG** 值的 “o” 标记周围的矩形，并且检查在数据表的 “行” 面板中选定的行数。

   您会看到该标记表示两行。您可以查看对其他标记也进行了重叠。若观测重叠，也就是在图上由一个点表示若干观测的情况，密度等高线图可帮助您查看点的密度。

7. 将等高线图标 拖入图形。

   将等高线图标拖入图形会保留每种发动机的拟合线。

图 3.10 按发动机类型划分的高速 MPG 和市区 MPG 的密度等高线



等高线指示对于两种发动机，研究中具有较低 MPG 值的车辆要比具有较高 MPG 值的车辆多。

移动或删除区域中的变量

有若干方式可供您将变量移入某个区域：

- 要将某个变量从一个区域移至另一个区域，点击该变量并将其从旧区域拖入新区域。
- 要在区域中替换某个变量，请将新变量拖到旧变量上方。
- 要将变量从某个区域切换到另一个区域，请右击您要移动的区域中的变量并选择**交换**。然后，选择您要交换位置的变量。
- 要合并同一轴上的多个变量，请参见第 38 页的“合并公共轴上的变量”。

要删除某个变量，请将其拖入空区域，或右击该区域内变量的名称并选择**删除**。

使用轴

- “合并公共轴上的变量”
- “在 X 和 Y 区域中创建不同的轴”

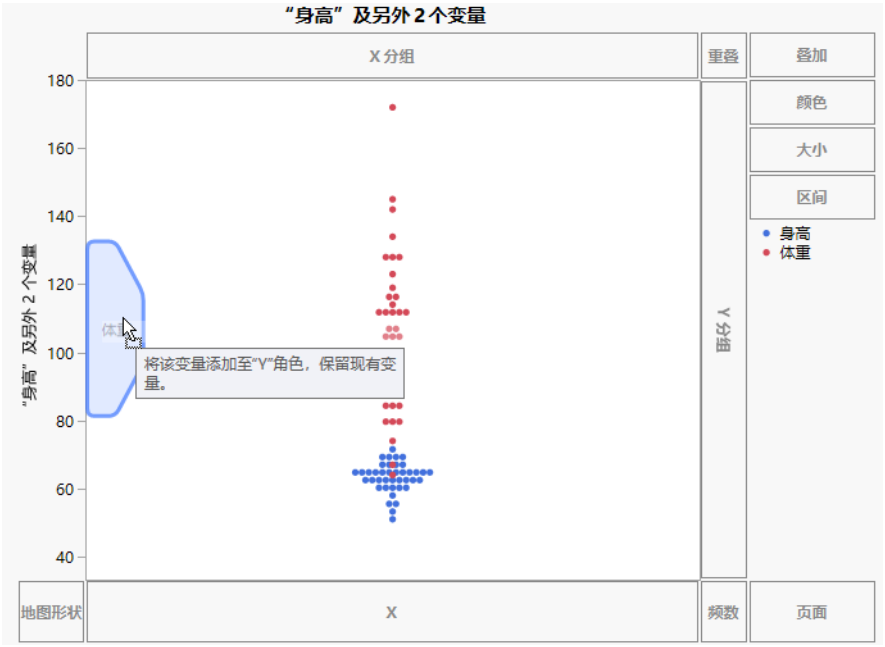
- “创建另一个 Y 轴”
- “为字符变量创建嵌套轴”
- “对分类变量水平排序”

合并公共轴上的变量

通过合并变量可将两个变量置于同一轴上，并为这两个变量创建一个图形。图中的元素由变量着色，并且在图右侧显示一个图例。标绘各点时，将为轴组合定义的每个值或类别随机散布各点。

- 若最初使用空的拖放区，则突出显示列表中的多个变量，然后将其同时拖入该拖放区。
- 若拖放区中已有变量而您想要添加更多变量，请将新变量拖到拖放区中的图形区域内，直到出现一个对称的多边形形状。

图 3.11 与现有变量合并的新变量



基于数据和建模类型的结果

- 若仅合并连续建模类型的变量，则沿着公共轴标绘这些变量的值。
- 若建模类型为分类（名义型或有序型）的变量与连续变量合并，则沿着公共轴标绘所有变量的值。

  在将字符数据类型的变量与数值变量合并之前，必须为字符变量的水平分配数值。基于“值顺序”列属性或默认值排序分配整数值。这些整数值介于 0（值排序中的最高水平）到水平数减 1 之间 （值排序中的最低值）。随后沿着公共轴标绘这些整数值。

- 若仅合并分类建模类型的变量，则为每个变量构造不同的**嵌套**轴。请参见第 42 页的“为字符变量创建嵌套轴”。

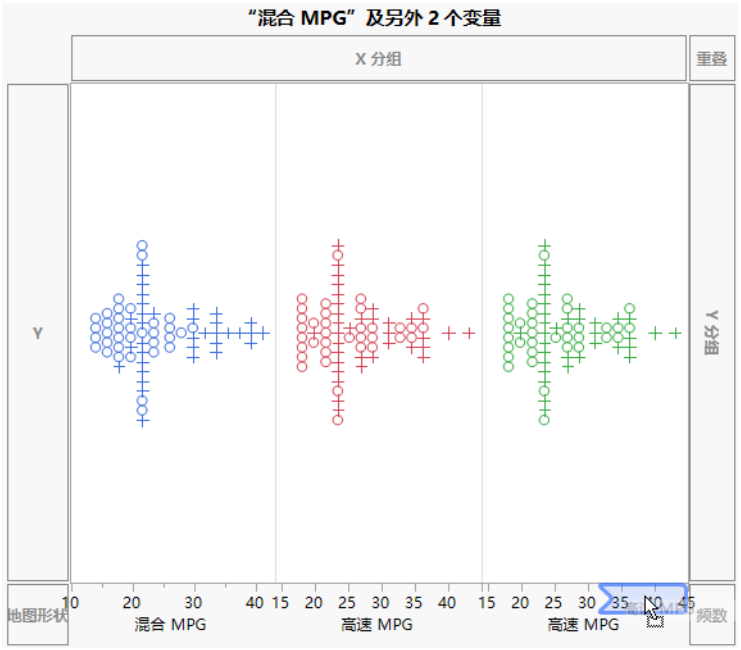
在 X 和 Y 区域中创建不同的轴

要在 X 区域中沿着不同轴快速标绘多个变量，请执行以下操作：

1. 选择“变量”列表中的列。
2. 点击**偏移**，然后点击 X 区域。

若想要进一步控制变量显示的位置，请逐一拖动它们。将其他变量拖至 X 区域中的现有变量之上、之下或之间。

图 3.12 将某个变量拖至现有变量的右侧



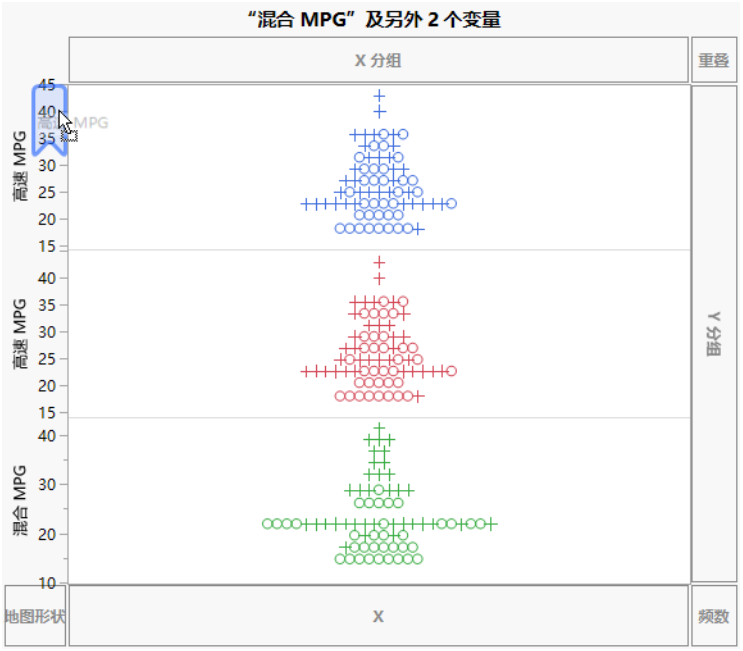
当变量位于合适的目标区域中时，将通过一个多边形形状来指示。

要在 Y 区域中沿着不同轴快速标绘多个变量，请执行以下操作：

1. 选择“变量”列表中的列。
2. 点击**偏移**，然后点击 Y 区域。

若想要进一步控制变量显示的位置，请逐一拖动它们。将其他变量拖至 Y 区域中的现有变量之上、之下或之间。

图 3.13 将某个变量拖至现有变量的上方



当变量位于合适的目标区域中时，将通过一个多边形形状来指示。

创建另一个 Y 轴

若同一轴上有两个或更多 Y 变量，您可以通过创建另一个 Y 轴来反映另一组变量的尺度。这在两个变量测量相同的基础量但具有不同尺度时很有用。通常，在其他任何情况下都不建议使用另一个 Y 轴。请参见 Few (2008)。

要创建另一个 Y 轴，请执行以下操作：

1. 右击 Y 区域中某一个 Y 变量名，然后选择**右移**。
2. 选择您要移至新轴的一个或多个变量。

   新轴将根据指定变量值统一尺度，并且沿着该轴绘制选定的变量。
3. 重复该过程，沿新轴标绘更多变量。

   新轴将调整以适应其他变量的值。

更改另一个 Y 轴的图形

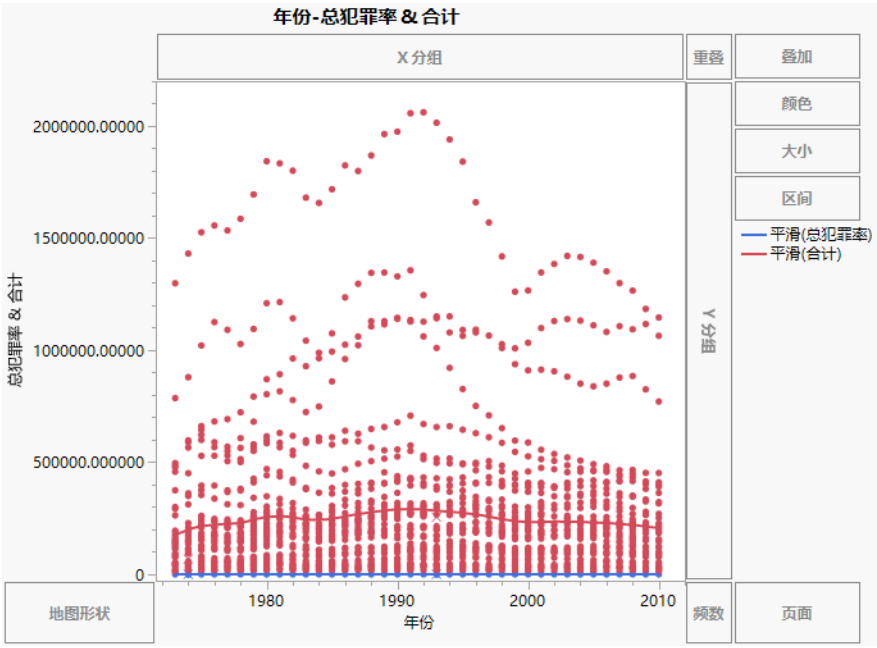
点击某一元素类型时，该类型仅应用到左侧 Y 轴上的变量。要将某个元素应用到右侧 Y 轴上的变量，请右击并更改与右侧轴对应的第二个元素。

创建另一个 Y 轴的示例

CrimeData.jmp 样本数据表包含从 1973 到 2010 年间美国 50 个州每年各类罪行的数据。**合计**列提供事件数总和，**总犯罪率**列提供总体调整后的比率。您想要在单个图上显示这两个变量。

1. 选择**帮助** > **样本数据库**，然后打开 CrimeData.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**总犯罪率**和**合计**，并将其拖入 Y 区域。
4. 选择**年份**并将其拖至 X 区域。

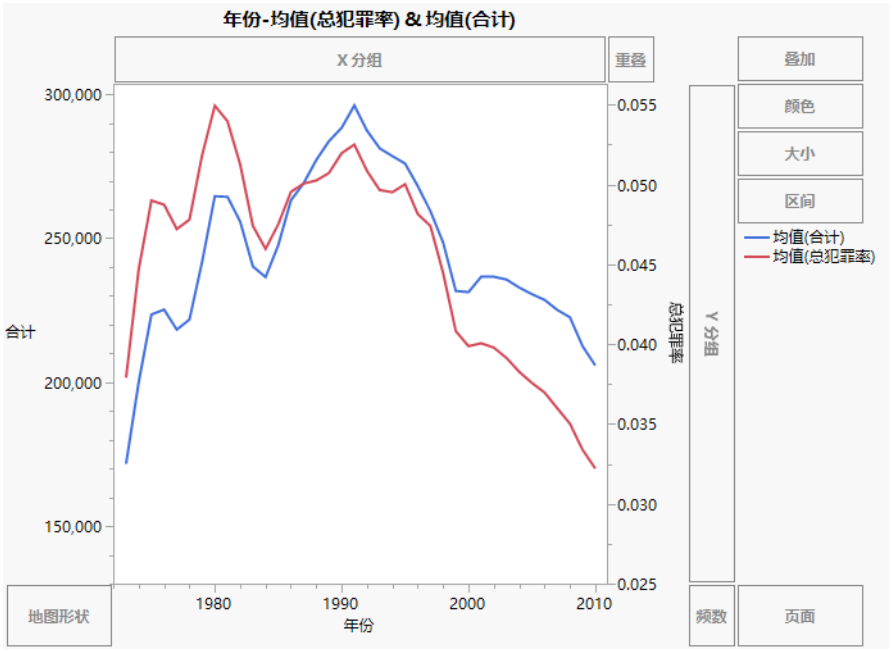
图 3.14 合并之后的总犯罪率与合计



总犯罪率值全部介于 0 和 1 之间，所以几乎看不出。计算每年均值，再使用另一个 Y 轴让这些值可见。

5. 在 “点”的选项面板中，选择**汇总统计量**旁边的**均值**。
6. 点击 “线图”元素 。
7. 右击 Y 区域中的变量名，然后选择**右移** > **总犯罪率**。

图 3.15 添加了另一个 Y 轴



在右侧添加针对**总犯罪率**的轴，并且各轴重新统一尺度以同时显示两组值。您现在可以看到两个测度之间的关系。例如，两个测度都开始在 1991 年下降。

8. （可选）点击**完成**。

为字符变量创建嵌套轴

若合并全部为分类建模类型的变量，则为每个变量构造单独的轴。最外层轴对应于第一个选定的变量，次外层轴对应第二个选定的变量，依此类推。

要以特定顺序嵌套各轴，请逐一将这些变量拖至拖放区。首先，将最内层轴对应的变量拖入拖放区。然后将下一个轴拖至前一个变量的外部并在出现一个梯形时放下该变量。

下例演示了嵌套轴。

在通过更改来改善测量过程后，会执行一个测量系统分析研究，目的是研究重复性和再现性（以**部件**、**操作员**和**仪器**为因子）。三个操作员每人都会测量四种仪器的每一个部件（共八个）。最为关注的是仪器的一致性。测量的数量称为**新建 Y**。

1. 选择**帮助 > 样本数据库**，然后打开 Variability Data/3 Factors Crossed.jmp。
2. 选择**图形 > 图形生成器**。
3. 选择**新建 Y** 并将其拖至 Y 区域。
4. 选择**部件**并将其拖至 X 区域。

该图显示为八个部件的每一个部件的测量值都存在变异。部件间存在预期的系统化差异。例如，部件 7 和 8 的测量值低于部件 1 到 6 的测量值。

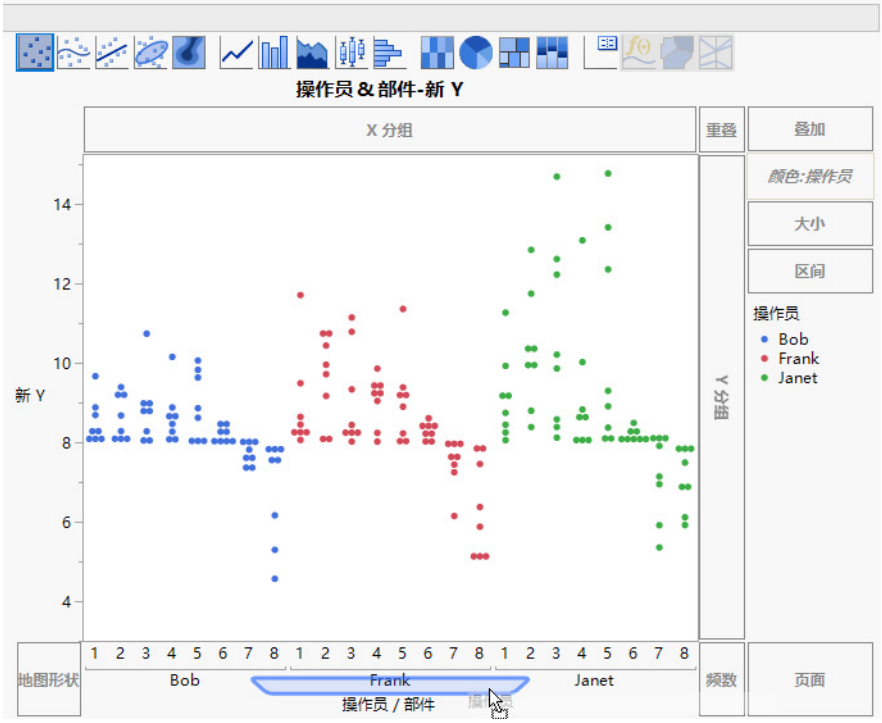
5. 将**操作员**拖至**颜色**区域。

   使用图形右侧的图例按照**操作员**对**新建 Y** 值着色。看起来与其他两个操作员相比， Janet 对多数部件测量的值都可能要高一些。但是**操作员**效应不易可视化，所以您需要创建单独的**操作员**轴。

6. 选择**操作员**并将其拖至 X 区域的**部件**之下。

   随即显示标记**操作员 / 部件**，指示**部件**与顶部轴关联，**操作员**与下面的轴关联。

图 3.16 拖动操作员以添加另一个轴

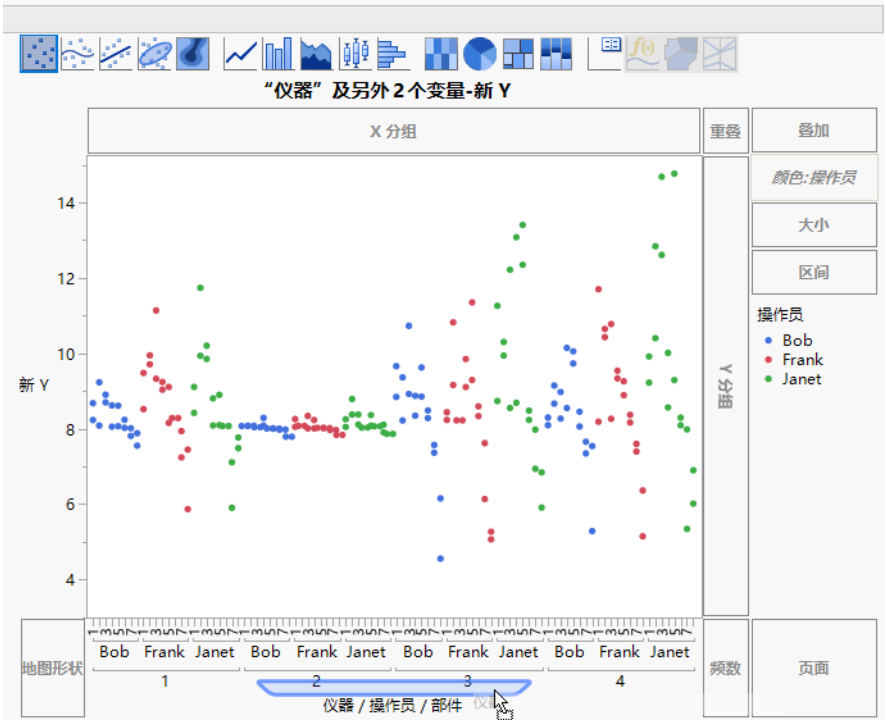


现在很容易看到：与 Bob 和 Frank 相比， Janet 往往会在相同的部件上得到较高的测量值。但**仪器**的影响如何？

7. 选择**仪器**并将其拖至 X 区域中的**操作员 / 部件**下方。

   随即显示标签**仪器 / 操作员 / 部件**，指示已在**操作员**轴的下方添加了第三个轴**仪器**。

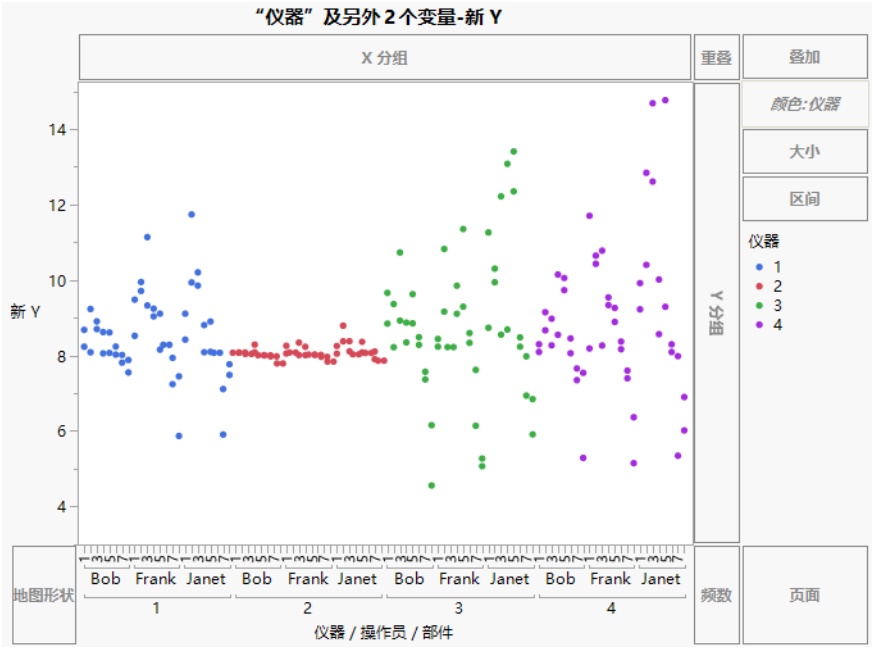
图 3.17 拖动仪器以添加第三个轴



很明显：与另外三个仪器相比，仪器 2 给出的测量值的一致性要高得多。对于仪器 2，操作员之间或操作员本人各次之间之间的变异都较小。

8. 选择**仪器**并将其拖至**颜色**区域。

图 3.18 三个嵌套轴



现在**新建 Y** 值按**仪器**来着色，很容易看出**仪器**的不同。

通过嵌套研究中三个因子的轴，您能够形象了解因这些因子所导致的变异。

对分类变量水平排序

JMP 根据以下规则对分类变量水平排序：

- 数值、名义型数据按数字顺序排序。
  - 比较数值周围的空格：“vt 1”的顺序排在“vt1”之前。
- 只含数字（数值）的字符数据按数字顺序排序。
- 字符数据按字母顺序排序，但以下情况除外：
  - 月份和星期几按时间顺序排序。
  - 等级按照从低到高排序：

    从低到高：极低、低、中低、中、中高、高、极高

    同意：完全不同意、不同意、中立、无关紧要、同意、完全同意

    较差或好：令人失望、不可接受、很差、差、较差、可接受、平均、好、更好、很好、非常好、最佳
  - 含有字符前缀和数值后缀的字符数据首先按前缀排序，之后按后缀排序。例如，批次 1 到 12 按以下方式排序：“lot1”、“lot2”、“lot3”...“lot10”、“lot11”、“lot12”。

– 若数值后缀以“0”开头，则不按后缀进行数值排序。（“lot1”排在“lot02”之后）。

注意：以上规则仅适用于有分类轴的图形。

要更改图形轴上分类变量的水平顺序，请点击并按住轴上的水平以进入编辑模式。在编辑模式下（通过每个标签周围的框来表示），可以拖动一个或多个标签（按 shift 键并点击可进行多选）来重新排列它们。重新排列标签时，“值顺序”列属性将更新。您还可以编辑标签，这样即可更新值标签列属性。按 Esc 键可退出编辑模式。

您还可以使用“值顺序”列属性对轴上的分类（名义型或有序型）变量的水平进行排序，该排序将优先于其他所有排序值。要更改排序，您可以执行以下操作之一：

- 使用“值顺序”列属性确保值按您的预期排序。有关“值顺序”列属性的详细信息，请参见《使用 JMP》。
- 使用图形中已有的数值变量的值。在“图形生成器”中，右击分类轴并选择**排序依据**。
- 使用任意数值变量的值。在“图形生成器”中，拖动拖放区中分类变量旁边的数值变量，直到出现一个梯形，然后放下该变量。轴标签改为由<数值变量>排序的<分类变量>。请参见第 46 页的“使用数值变量对分类变量水平进行排序的示例”。

更改水平或统计量的顺序：

1. 右击轴标签并选择**排序依据**。
2. 选择升序或降序选项之一，或选择**其他**以选择数值列。

   您可以按变量的“次序统计量”排序（这些选项的前缀为变量名），或是按分类变量每个水平中的观测数（计数）排序。
3. 默认的次序统计量为均值。要使用另一个统计量，请右击轴标签并选择**次序统计量**。

注意：若尝试用其他数值变量对某数值变量的值排序，JMP 将合并这些变量。请参见第 38 页的“合并公共轴上的变量”。

使用数值变量对分类变量水平进行排序的示例

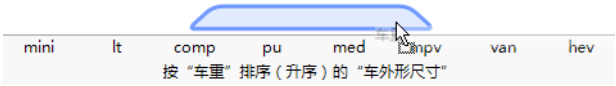
要依据数值变量对名义型或有序型变量排序，请考虑关于车辆类型的数据。您想要查看以有意义的顺序排列的车辆类型。

1. 选择**帮助 > 样本数据库**，然后打开 Cars.jmp。
2. 选择**图形 > 图形生成器**。
3. 选择**车外形尺寸**并将其拖至 X 区域。

   该变量表示车辆类型。八个水平按字母顺序排列在 X 轴上：紧凑型 (comp)、重型 (hev)、轻型 (lt)、中型 (med)、迷你型、多功能 (mpv)、皮卡 (pu) 和**厢式**。由于水平按字母顺序排列，所以没有以有意义的方式排序。例如，重型排在迷你型和轻型之前。您想要按照**车重**对水平排序。

4. 选择**车重**并将其拖至 X 轴的中间。将其放到 X 轴正上方。放下变量之前，将出现一个蓝色四边形。

图 3.19 合并车重和车外形尺寸

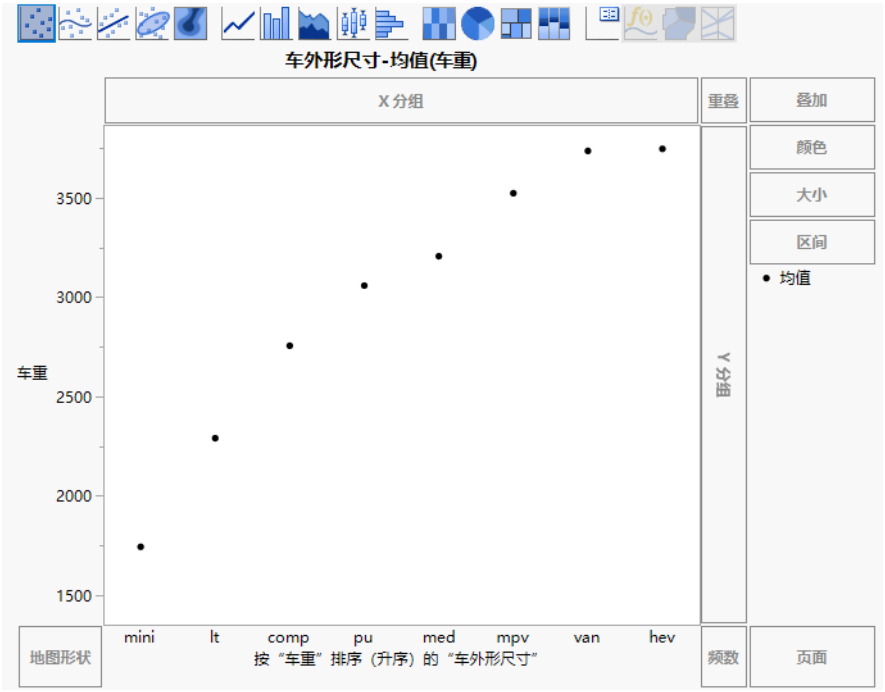


车外形尺寸的各个水平现在依据水平中所有车辆的平均**车重**以升序排列。请注意：**迷你型**和**轻型**现在排在**重型**之前。轴标签随之更新，表明使用了一个排序变量。

下一步，验证**车外形尺寸**是否真正按照**车重**排序。

5. 选择**车重**并将其拖至 Y 区域。
6. 从“汇总统计量”列表中选择**均值**。

图 3.20 按“车重”升序排序的车外形尺寸的示例



您可以看到平均**车重**从左到右递增。

将顺序从升序改为降序。

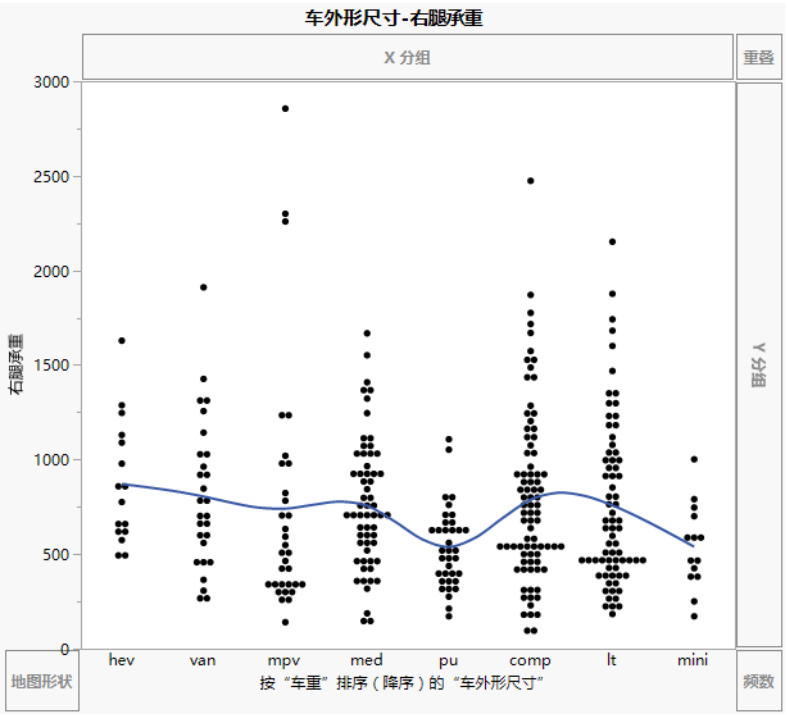
7. 右击 X 区域并选择**排序依据 > 车重，降序**。

现在**车外形尺寸**的水平即按照**车重**均值降序排列。您想要查看右腿承重（**右腿承重**）是否随着车重降低。

8. 右击 Y 区域中的**车重**，然后选择**删除**。
9. 选择**右腿承重**并将其拖至 Y 区域。
10. 从 “汇总统计量” 列表中选择**无**。

    该选择会将所有单个观测的均值替换为点。
11. 点击 “平滑线” 元素 。

图 3.21 按 “车重，降序” 排列的 “右腿承重” 的示例



右腿承重似乎与车重级别不相关。

默认的次序统计量为均值。要使用另一个统计量，右击 X 区域，选择**次序统计量**，并将其改为您需要的统计量。

元素类型和选项

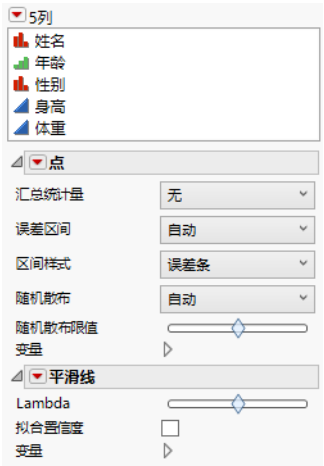
通过点击元素类型图标（如条形图、线图或直方图），可以更改数据的显示方式。

图 3.22 元素类型图标

注意：支持的元素类型因所选的变量类型和区域而异。不适用的元素类型将变暗。

对于添加到图形中的每个元素，左侧的“属性”区域中将显示一个分级显示项。在此可以指定和更改每个元素类型的属性。

图 3.23 点和平滑线的 “属性”区域

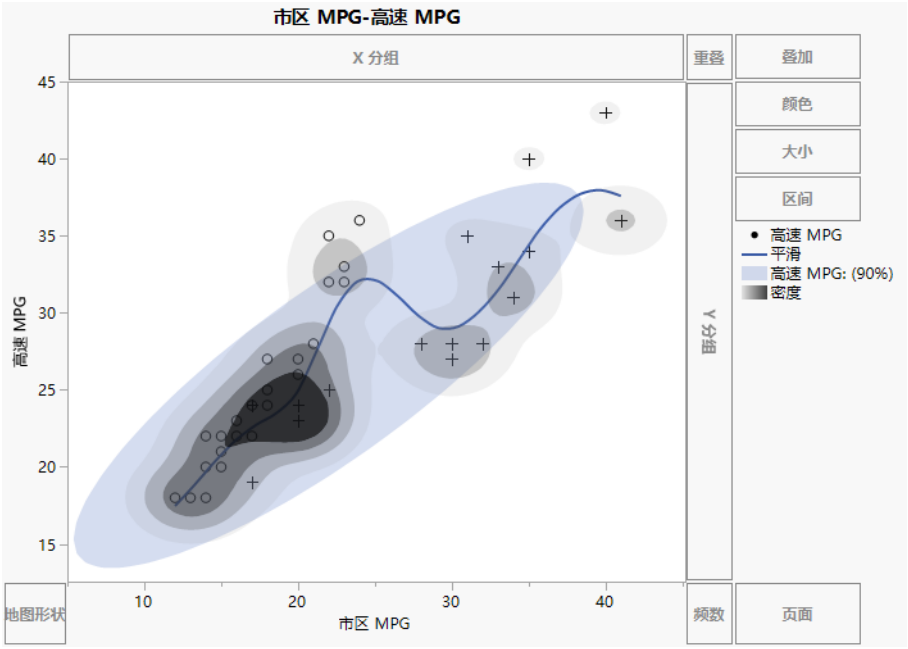


您还可以右击任意图形以更改该元素或是在图形中定制元素。

同时选择多个元素

要选择和叠加多个元素，请按住 Shift 键并点击这些元素。或者，您可以逐一将这些元素拖入图形。

图 3.24 显示点、平滑线、椭圆和等高线元素的图形



点的标签

在多数显示点的图形中，悬停在某个点上方时，会出现一个标签，其中显示关于相应行的信息。悬停在该标签上方并右击可访问更多选项。您可以将该标签固定在图形中、复制内容或关闭以删除该标签。还可以通过点击该标签右上角的大头针图标来固定该标签。请参见《使用 JMP》。

隐藏和排除的行

隐藏和排除是两个不同的行状态：

- 隐藏行的目的是影响可见性，而不是分析结果。
- 排除行的目的是影响分析结果。

对于图形，分析值通常是外观的一部分。因此，排除行通常会影响图形外观和作为图形一部分包含的任何分析结果。隐藏行通常只影响图形的外观。

例如，考虑一个带有点和拟合线的图形：

- 若隐藏行，则这些行将包含在用于获得拟合线的计算中，但不会标绘这些行。
- 若排除行，则这些行将从计算和显示中同时排除。

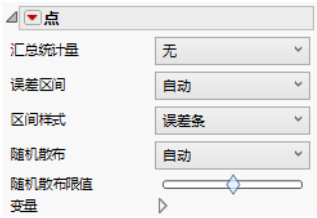
在这两种情况下，出现的点数相同，但拟合线却不同。

在绘制组时，若组中的所有行都被隐藏，则该组的视觉效果也被隐藏。

点

点元素 将数据值显示为点。

图 3.25 点选项



汇总统计量 更改所标绘的统计量。使用响应轴上的变量计算统计量。“无”是默认设置，指示标绘数据值本身。

误差区间 在图形中添加或删除指定的误差区间。

区间样式 绘制选定误差区间的误差条或误差带。

随机散布 随机散布可通过向值添加随机噪声来减少重叠。从以下随机散布类型中选择：

无 未进行任何调整。

自动 涉及分类变量时添加各种类型的随机散布。仅有连续变量时不添加随机散布。

随机均匀 服从均匀分布的随机剩余偏差。

随机正态 服从高斯分布的随机剩余偏差。

堆叠填充 紧密填充标记以保持任何非随机散布维。

居中网格 类似于“填充”，但调整非随机散布维，使其落入网格。

正网格 类似于“居中网格”，但取正方向。

密度随机 对于一维随机散布，该选项将标记随机放置在小提琴图的边界内。对于二维随机散布，该选项在椭圆内随机放置标记。

您还可以通过右击和选择**定制** > **标记**来定制随机散布。

提示：要创建点图，请选择“正网格”选项。通过将 Y 轴值改为 0-1，您可以移动点图使其在底部开始。您还可以将标记大小调整为较大的值，例如 20，以便调整点的分箱。

随机散布限值 控制重叠的散布程度和重叠量。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于“X 分组”、“Y 分组”、“重叠”或“页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对点的红色小三角选项

响应轴 将与响应变量关联的轴改为“X”（水平）、“Y”（垂直）或自动确定响应方向。

设置形状表达式 支持使用 JSL 表达式定义标记形状。

设置形状列 设置要用作标记形状的列。

平滑线

平滑线 元素显示穿过数据的平滑曲线。

提示：要标记平滑线，右击图例中的平滑线元素。

图 3.26 平滑线选项

方法 指定用于生成平滑拟合的方法。选项包括“样条”、“局部核”、“Savitzky-Gloay”、“移动平均值”和“移动框”。

Lambda （可用于“样条”。）支持您调整 lambda 的值。默认值为 0.05。有关 lambda 的详细信息，请参见《基本分析》。

次数 （可用于 Savitzky-Golay。）指定平滑度。

局部宽度 （不可用于“样条”。）支持您根据局部到全局拟合来控制平滑法拟合。使用红色小三角“局部区域”选项执行进一步控制。

切尾 支持您修剪基于不完整或不平衡数据范围的平滑线部分。滑块的尺度为 0 到 1。为 1 时，将修剪所有不完整或不平衡的区域，但对于 1 对应于倒数第二个边结点的样条除外。

拟合置信度 显示或隐藏每个拟合的 Bootstrap 置信区域。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对平滑线的红色小三角选项

响应轴 将与响应变量关联的轴改为“X”（水平）、“Y”（垂直）或“自动”（Y 轴表示平滑线）。

局部区域 指定的选项用于控制用于 “局部宽度”选项的区域。

局部权重 指定权重选项。

适应轴尺度 关于对数和其他轴变换，它是对变换的坐标应用计算的。

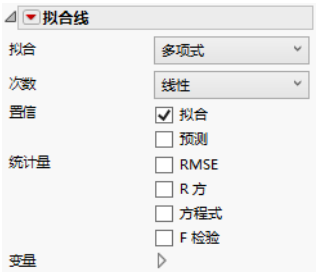
保存公式 将单个变量的拟合公式和上下置信区间保存到数据表中。

拟合线

拟合线元素 显示带拟合置信区间的线性回归线。

提示：要为拟合线添加标签，请右击图例中的线图元素。

图 3.27 拟合线选项



拟合 指定拟合类型：

多项式 简单线性回归。

稳健 Cauchy 假定服从 Cauchy 分布的残差以弱化离群值影响的线性回归。

时间序列 具有最优季节性的等间隔 X 值的平滑法。包含用于预测模型和季节与预测周期数的选项。平滑模型从状态空间平滑模型子集中选择，这些模型由 Hyndman et al.(2008) 定义。请参见《预测和专业建模》。

次数 指定线性回归拟合的多项式次数，可以是一次、二次或三次。

置信 显示或隐藏预测值（拟合）或单值（预测）的置信区间。这两种区间都固定在 95% 置信区间。

统计量 显示图形的各种选择。您可以显示回归线的均方根误差 (RMSE)、 R 方、方程和 F 检验值。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠” 或 “页面” 区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在 “变量” 选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对拟合线的红色小三角选项

响应轴 为用作线性回归线计算中的响应的变量指定轴。由于标准最小二乘通常将 Y （垂直）轴用作响应轴，所以 “自动” 设置默认为 “Y 轴”。

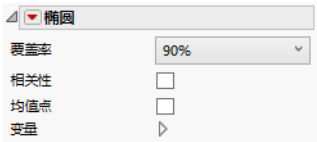
适应轴尺度 关于对数和其他轴变换，它是对变换的坐标应用计算的。

保存公式 将单个变量的拟合公式和上下置信区间保存到数据表中。

椭圆

椭圆 元素显示二元正态密度椭圆。

图 3.28 椭圆选项



覆盖率 指定密度椭圆的覆盖百分比 （99%、 95%、 90% 或 50%）。

相关性 显示图形中 X 和 Y 变量的 Pearson 相关系数。

均值点 在图形上显示均值。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠” 或 “页面” 区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将 “颜色”、 “大小”、 “形状” 或 “频数” 变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在 “变量” 选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见 “图形生成器示例” 一章中第 108 页的 “面积图和线图示例”。

针对椭圆的红色小三角选项

适应轴尺度 关于对数和其他轴变换，它是对变换的坐标应用计算的。

等高线

等高线元素 显示密度区域 （在与 “颜色” 变量一同使用时则显示值等高线）。密度等高线在以下情况下很有用：散点图中有很多点，这些点过于密集而很难辨别密度中的模式。提供多个等高线类型。默认的等高线类型是一个平滑二元非参数密度曲面。它的拟合用以反映数据点的密度。非参数密度曲面估计每个点的二元概率密度函数，提供连续模拟的二元直方图。

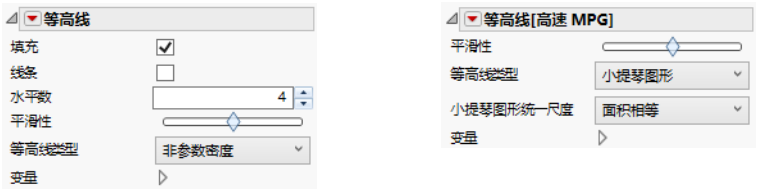
- 对于两个连续变量，默认标绘四条等高线。这些等高线分别为 100%、75%、50% 和 25% 密度等高线。您可以指定要显示的等高线水平数。或者，您可以选择 “袋状图” 或 “高密度区” (HDR) 等高线。
- 对于仅有一个连续变量的情况，显示的是**小提琴**图而不是等高线图。小提琴图通过在公共垂直轴周围标绘对称核密度来演示数据密度。核密度估计每个点的概率密度函数，提供连

续模拟的直方图。小提琴图类似于具有对称核密度（替换了箱和须线）的箱线图。或者，您可以选择“高密度区”(HDR) 等高线。

- 若向等高线图添加“颜色”变量，该图将显示反映“颜色”变量水平的**值等高线**。使用 Delaunay 三角法测量计算值等高线。您可以选择某一选项（变换）来显示已经标准化 X 和 Y 范围的图。

等高线选项

图 3.29 针对等高线图或小提琴图的等高线选项



填充 （不可用于袋状图。）填充等高线图。

线条 （不可用于袋状图。）添加环绕等高线的线条。

水平数 （可用于非参数密度。）对于密度等高线，指定显示的等高线数量。该数量可介于 1 与 1000 之间，默认为 4 条等高线。

边界 （仅当存在 “颜色”变量时才可用。）围绕等高线的外边界添加一条线。

Alpha （仅当存在 “颜色”变量时才可用。）控制值等高线的轮廓。增大 alpha 可以消除一些可能不需要内插的长三角形、细三角形或大三角形。

平滑性 （不可用于袋状图。）平滑等高线图的边界。平滑度值在 -1 与 1 之间进行标准化。该值可解释为平滑内核半径。原始数据先内插成网格，然后应用高斯平滑线。

变换 （仅当存在 “颜色”变量时才可用。）变换三角测量，通过选择 “标准化范围”来使用 X 和 Y 的标准化尺度。这将在计算 Delaunay 三角测量之前使得 X 和 Y 值的尺度统一为 [0,1]。该选项可能更适合 X1 和 X2 的单位极为不同的情况。在这些情况下，若各点在一个维上拉伸（在另一个维上不拉伸），则 Delaunay 三角测量会尝试最小化细长三角（细长三角可能模糊某些功能）。

等高线类型 支持选择等高线类型

非参数密度 （仅可用于二元图。）一个平滑二元非参数密度曲面。

袋状图 绘制袋状图，亦称二元箱线图。袋状图包含两个多边形、一组离群值点和一个中位点。所有计算都基于对数据中各点的 Tukey 深度（二元深度）的首次计算。中位数点是所有点在最大深度时的平均值，绘制为一个星号。内部多边形是**袋**，其中包含至多 50% 的数据点。图中没有显示的是**围栏**，它是相对于中位点膨胀三倍的袋状多边形。外部多边形是围栏内包含的所有点的凸包。这称为**循环**。位于围栏外部的点被指定为离群值，并在图中显示为点。有关袋状图的详细信息，请参见 Rousseeuw (1999)。

HDR 为一元数据绘制最高密度区域矩形，为二元数据绘制等高线。较浅着色区域表示 99% 概率区域；较深着色区域表示 50% 概率区域。（请注意，这些区域可以是不连续的。）50% 概率区域内的密度模式由一条线（一元数据）或一个星号（二元数据）表示。若从图中删除点，则保留的点表示相对于 99% 概率区域的离群值。可以使用“平滑度”选项调整区域的平滑度。

HDR 区域基于非参数密度，该密度是在将点内插成网格后对数据应用高斯核估计得出。有关最高密度区域的详细信息，请参见 Hyndman (1996)。

小提琴图形 （仅可用于一元图。）通过在公共垂直轴周围标绘对称核密度来绘制数据密度的小提琴图。

随机散布 启用或禁用随机散布。随机散布会向数据添加随机噪声，并且可以帮助减少重叠。有关特定随机散布选项的说明，请参见第 51 页的 “点”。

离群值 （可用于袋状图和 HDR 等高线。）绘制离群值。

小提琴图形统一尺度 对于多个小提琴图形，选择一个尺度选项。面积或宽度相等意味着小提琴图形的面积或最大宽度相同。加权面积意味着小提琴图形的面积与每个小提琴图形中的观测数成正比。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对等高线的红色小三角选项

适应轴尺度 关于对数和其他轴变换，它是对变换的坐标应用计算的。

等高线放置 （可用于分组变量。）支持您对齐等高线或调整等高线偏差。

线

线元素 使用线段跨越另一个变量的值或类别来连接标绘在响应轴的汇总统计量值（或原始值）。

有关将线图与面积图相结合以强调某条线的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

注意：

- 若在计算图中点时使用的任何行处于隐藏但未排除状态，则不显示该点，但仍保留该点占据的位置。线段连接邻近的各点。
- 要为某条线添加标签，请右击图例中的线元素。

图 3.30 线选项

行顺序 按照各点的行号的顺序用线段连接各点。

连接 将连接类型改为线条、曲线、步进、居中步进、水平或垂直线。若您选择曲线，还可以更改平滑度。

平滑性 更改“曲线”连接的平滑度。要使用该选项，选择“曲线”作为“连接”类型。

汇总统计量 指定标绘的统计量。为 X 区域中每个非重复变量值处的观测计算统计量（若将 X 指定为“响应轴”，则为 Y 区域中每个非重复变量值处的观测计算统计量）。

堆叠 （可用于多个 Y 变量。）累积或堆叠 Y 值。用于构建面积图。

填充 指定如何填充线条下方或线条之间的区域。

误差区间 在图形中添加或删除指定的误差区间。

区间样式 绘制选定误差区间的误差条或误差带。

缺失因子 当 X 轴上的变量设置缺失所有值时，提供用于连接的选项：

跳过 跳过缺失因子值，将上一个因子连接到下一个因子。

视为缺失 跳过缺失因子值，将上一个因子连接到下一个因子。采用为“缺失值”指定的样式绘制连接。

视为零 将缺失因子值视为零。

缺失值 当响应轴上的变量设置缺失所有值时，提供用于连接的选项：

连接穿过 使用对其余点使用的连接类型连接邻近缺失值的各点。

连接变淡 使用变淡的连接来连接邻近缺失值的各点。

用虚线连接 使用虚线连接来连接邻近缺失值的各点。

无连接 在邻近缺失值的各点之间显示无连接。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对线条的红色小三角选项

响应轴 指定在标绘线段时用作响应的变量的轴。“自动”设置的是 Y 轴。

保存公式 将单个变量的拟合公式和上下置信区间保存到数据表中。

条形图

条形图元素 显示在一个或多个分层变量的不同值或类别之间，一个或多个变量的汇总统计量值的条形图。该图形可显示在 X 和 Y 区域中排列的任何类型的单个变量或多个变量。

- 若添加单个分类变量，则每个直条都显示相应的变量水平中的观测计数。
- 若添加两个变量，每个区域添加一个，则每个直条都表示指定“响应轴”上的变量的指定汇总统计量值。分层变量的每个值或水平都对应一个直条。
- 若在每个区域中添加多个变量，则为分层变量的每个值或水平组合都标绘多个直条。

数据类型为“字符”的响应轴变量

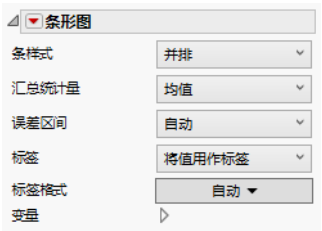
若响应轴上的变量的数据类型为 “字符”，则为该变量分配数值：

- 基于“值顺序”列属性或默认值排序分配整数值。
- 这些整数值介于 0（值排序中的最高水平）到水平数减 1 之间（值排序中的最低值）。
- 这些整数值用于计算汇总统计量。

注意：若在构造图中直条时使用的任何行处于隐藏但未排除状态，则不显示该直条，但仍保留该直条占据的位置。

有关使用条形图元素的图形的示例，请参见“图形生成器示例”一章中第 125 页的“叠加直方图和脊线图的示例”和“图形生成器示例”一章中第 124 页的“堆叠填充的条形图示例”。

图 3.31 条形图选项



条样式 您可以更改描述分层变量水平的直条的外观。许多条样式只有在响应轴上有合并变量或存在叠加变量时才适用。有关条样式说明的信息，请参见表 3.1。要更改条的宽度和间隔，请在图形中右击并选择**定制 > 条形图**。

堆叠填充条样式选项

堆叠填充的主类别 指定主直条数目。

堆叠填充的放置 指定次级直条的堆叠位置。选项包括以下内容：

第一个堆叠 — 从顶行开始，填充它之后再移至下一行。

最小堆叠 — 将每个直条放置在当前最小的堆叠中。

单独堆叠 — 创建一个名为“其他”的堆叠，将所有二级直条都置于其中。

堆叠填充的排序 指定次级直条的放置顺序。可按照大小放置。最大的直条显示在最前；也可以按标签排，即自然排序，通常按字母顺序放置。

堆叠填充的着色 指定如何为次级直条着色。选项包括以下内容：

直条颜色 — 应用与主条相同的颜色。

变淡的直条颜色 — 应用变淡的主条颜色。

灰色 — 应用浅灰色的随机灰度。

堆叠填充的添加标签 控制添加标签的次级直条的百分比。

堆叠填充的主标签 指定应放置主标签的位置，直条内或轴上。

汇总统计量 指定用作条形标签的统计量并控制响应轴尺度。可用统计量取决于所用的条形图类型。

误差区间 若适用，则在图中添加或删除误差区间，否则不可用。要定制误差条，请在图形中右击并选择**定制 > 误差条**。

标签 在条形图中添加或删除标签。您可以按照值、占合计值的百分比或按行来添加标签。若选定**将值用作标签**，则由汇总统计量设置确定值。点击并拖动对标签重新定位。

标签格式 （可用于值和百分比标签。）支持您设置单元格标签的格式。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对条形图的红色小三角选项

响应轴 指定在标绘直条时用作响应的变量的轴。仅当每个轴都包含至少一个建模类型为 “连续”的变量时，该选项才可用。若只有一个变量是 “连续”变量，其轴即为响应轴。“自动”设置的是 Y 轴。

保存公式 将单个变量的拟合公式和上下置信区间保存到数据表中。

叠加 指定多个直条之间的重叠量 （自动、无、一半、完全）。

表 3.1 条样式和说明

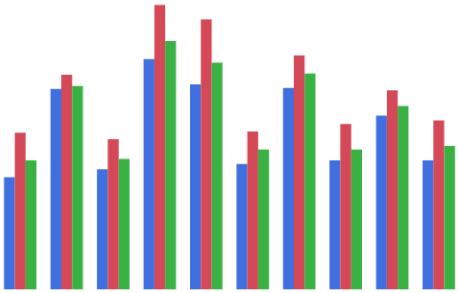
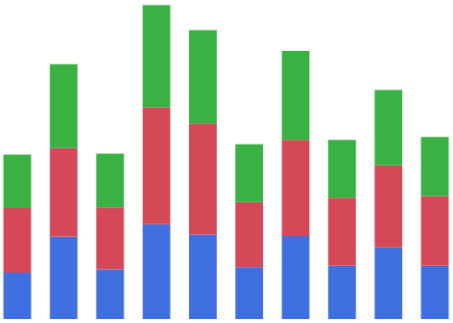
条样式	说明
并排	若响应轴上有两个或更多合并变量，则为每个合并变量都标绘一个直条。对于分层变量的每个水平，各个直条彼此并排显示。对这些直条着色并且显示一个图例。 
堆叠	若响应轴上有两个或更多合并变量，则为分层变量的每个水平都标绘一个直条。每个直条都堆叠每个合并变量的直条。将为对应于每个合并变量的直条部分着色，并显示图例。 

表 3.1 条样式和说明 (续)

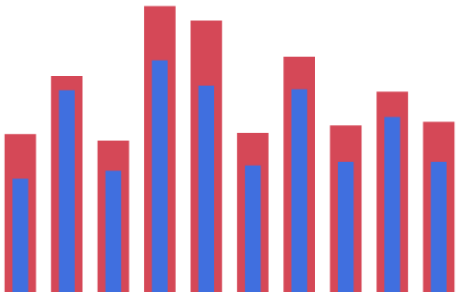
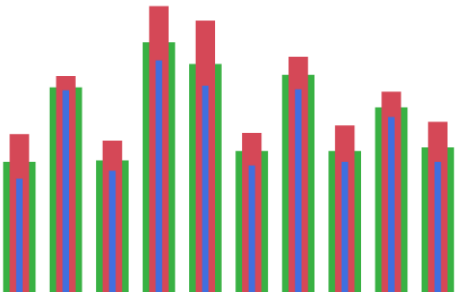
条样式	说明
弹道图	若响应轴上有两个或更多合并变量，则窄条标绘在较宽和较高的直条内。直条的宽度取决于变量的顺序。将为对应于每个合并变量的直条部分着色，并显示图例。 若仅有一个变量，则该样式标绘窄条。 
嵌套	若响应轴上有两个或更多合并变量，则每个变量的直条都彼此嵌套。最窄的直条对应于图例中列出的第一个变量，次窄的直条对应于图例中列出的第二个变量，依此类推。 **提示：**您可以使用 “变量”面板中的箭头更改图例顺序。 

表 3.1 条样式和说明 (续)

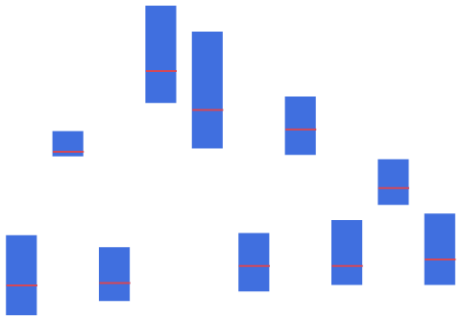
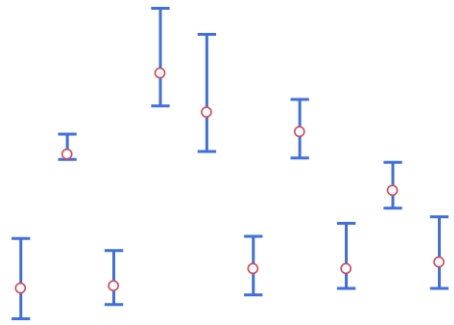
条样式	说明
范围	若响应轴上有两个或更多合并变量，则为分层变量的每个水平都标绘一个矩形和一条线。对于分层变量的每个水平： • 为每个合并变量计算汇总统计量的值。 • 标绘一个矩形，其边界值等于前两个变量的汇总统计量值，如图例所示。 • 在其他变量的汇总统计量值的位置标绘线条。 
区间	若响应轴上有两个或更多合并变量，则为分层变量的每个水平都标绘带边界的线条和一个圆圈。对于分层变量的每个水平： • 为每个合并变量计算汇总统计量的值。 • 标绘带边界的线条，其边界值等于前两个变量的汇总统计量值，如图例所示。 • 在其他变量的汇总统计量的总体值的位置标绘一个圆圈。 

表 3.1 条样式和说明 (续)

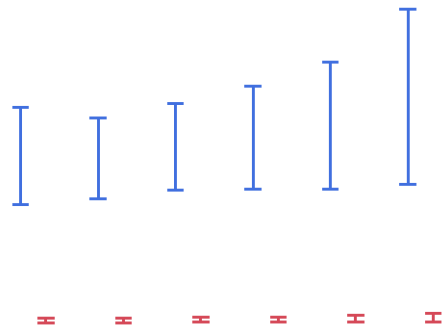
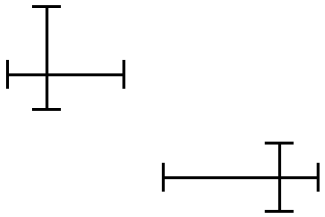
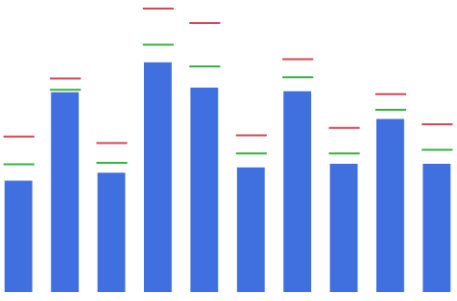
条样式	说明
并排区间	在该样式中，每对变量定义一个区间。 
双向区间	该样式针对每个方向上都有直条的二维条形图。 
单个	对于分层变量的每个水平，标绘一个直条。若响应轴上有两个或更多合并变量，直条表示图例中第一个变量的汇总统计量的值，为其余变量的汇总统计量的值显示线条。 **提示：**您可以使用 “变量”面板中的箭头更改图例顺序。 

表 3.1 条样式和说明 (续)

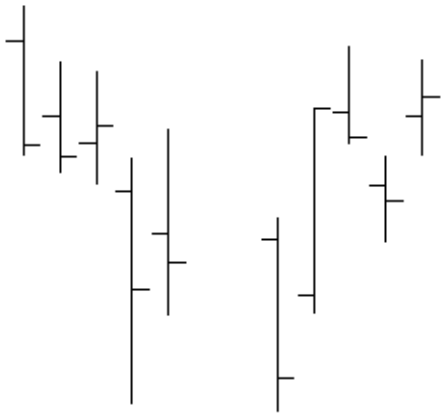
条样式	说明
股票	若响应轴上有两个或更多合并变量，则在分层变量的各个水平处标绘线段。对于分层变量的每个水平： • 为每个合并变量计算汇总统计量的值。 • 连接其中两个变量的汇总统计量值的线段，如图例所示。 • 在其余汇总统计量值的线段的垂直方向上标绘直条。 
箱线图	若响应轴上有两个或更多合并变量，则为分层变量的各个水平显示箱线图。对于分层变量的每个水平： • 为每个合并变量计算汇总统计量的值。 • 为这些值构造离群值箱线图。

表 3.1 条样式和说明 (续)

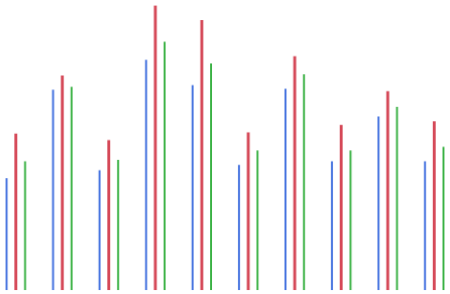
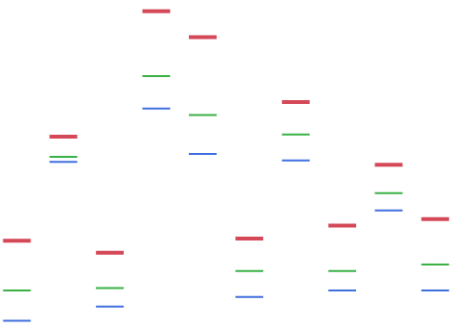
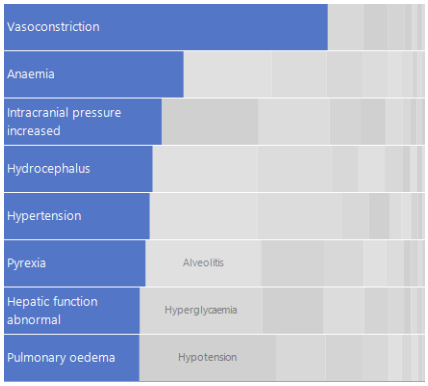
条样式	说明
针	若响应轴上有两个或更多合并变量，则为每个合并变量并排标绘针状直条。对这些直条着色并且显示一个图例。该样式适用于有许多分层变量水平的情况。 
浮动	若响应轴上有两个或更多合并变量，则在分层变量的各个水平处标绘直条。对于分层变量的每个水平： • 为每个合并变量计算汇总统计量的值。 • 在以上每个值处标绘一个直条。 该样式适用于有多个合并变量的情况。 

表 3.1 条样式和说明 (续)

条样式	说明
堆叠填充	该条样式将顶部类别显示为条形图，将其他类别堆叠为接近矩形填充的直条。顶部类别是焦点，其他直条提供上下文。 该样式适用于有许多类别且响应数据有偏斜的情况。 

面积图

注意：也可以使用“线图”元素 创建面积图。

面积图元素 显示的连续区域表示的是某变量在另一个变量的不同值或类别之间的汇总统计量值下方的区域。“面积图”元素可被视为 “条形图”元素的连续表示。

- 两个变量既可以是名义型也可以是有序型。汇总统计量表示这种轴上的变量水平中的观测数，该轴并未指定为“响应轴”。
- 其中一个变量可能是连续型，而另一个变量是名义型或有序型。汇总统计量表示分类变量对应的水平下的连续变量的指定汇总统计量的值。
- 若两个变量均为连续型，则为指定为“响应轴”的轴上的连续变量计算汇总统计量。

注意：若只隐藏行而未排除行，则该图不受任何影响。

有关使用面积图元素的图形的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

图 3.32 面积图选项

面积图样式 当响应轴上有合并变量时，更改面积图的显示方式。有关面积图样式的说明，请参见表 3.2。

行顺序 按照各点的行号的顺序用线段连接各点。

连接 将连接类型改为线条、曲线、步进、居中步进、水平或垂直线。若您选择曲线，还可以更改平滑度。

汇总统计量 指定标绘的统计量。为 X 区域中每个非重复变量值处的观测计算统计量（若将 X 指定为“响应轴”，则为 Y 区域中每个非重复变量值处的观测计算统计量）。

误差区间 在图形中添加或删除指定的误差区间。

区间样式 绘制选定误差区间的误差条或误差带。

缺失因子 当 X 轴上的变量设置缺失所有值时，提供用于连接的选项：

跳过 跳过缺失因子值，将上一个因子连接到下一个因子。

视为缺失 跳过缺失因子值，将上一个因子连接到下一个因子。采用为“缺失值”指定的样式绘制连接。

视为零 将缺失因子值视为零。

缺失值 当响应轴上的变量设置缺失所有值时，提供用于连接的选项：

连接穿过 使用对其余点使用的连接类型连接邻近缺失值的各点。

连接变淡 使用变淡的连接来连接邻近缺失值的各点。

用虚线连接 使用虚线连接来连接邻近缺失值的各点。

无连接 在邻近缺失值的各点之间显示无连接。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对面积图的红色小三角选项

响应轴 指定在标绘面积图时用作响应的变量的轴。仅当两个变量的建模类型均为“连续”时才可用。若只有一个变量是“连续”变量，其轴即为响应轴。“自动”设置的是 Y 轴。

保存公式 将单个变量的拟合公式和上下置信区间保存到数据表中。

表 3.2 面积图样式和说明

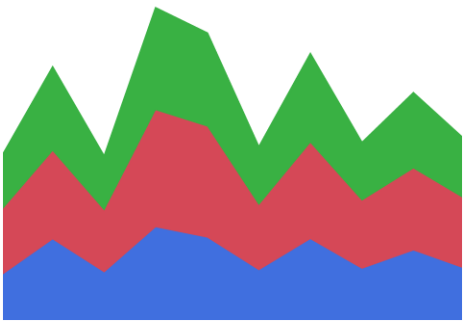
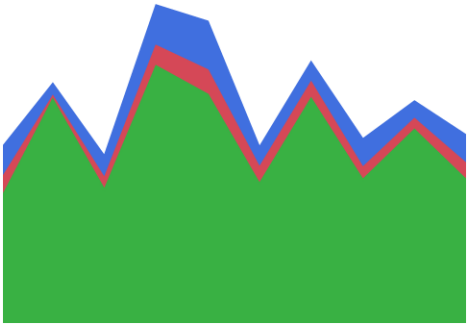
面积图样式	说明
堆叠	显示对应于每个合并变量的汇总统计量的面积图。这些图将被着色，同时在“图例”区域内显示图例。 

表 3.2 面积图样式和说明 (续)

面积图样式	说明
叠加	显示彼此叠加的合并变量的面积图。这些图将被着色，同时在“图例”区域内显示图例。 
范围	显示对应于合并变量的汇总统计量中的差异的面积图。
堆叠极差	显示堆叠变量后与汇总统计量中的差异相对应的区域。

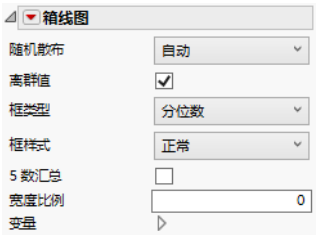
箱线图

箱线图元素 显示离群值或分位数箱线图。箱线图提供值分布的紧凑视图。当变量的数据类型为“数值”时，箱线图元素很有用。有关离群值和分位数箱线图的详细信息，请参见《基本分析》。

- 若其中一个变量是连续型并且另一个变量是名义型或有序型，则针对连续变量的值构造箱线图。分类变量的每个水平都对应一个箱线图。
- 若两个变量均为连续型，则为指定为“响应轴”的轴上的连续变量的值构造箱线图。其他连续变量的每个水平都对应一个箱线图。

有关使用箱线图的图形的示例，请参见第 30 页的“图形生成器中的功能的示例”。

图 3.33 箱线图选项



随机散布 启用或禁用随机散布。随机散布会向数据添加随机噪声，并且可以帮助减少重叠。

离群值 显示或隐藏延伸到须线之外的值。

框类型 指定箱线图为离群值箱线图还是分位数箱线图。

框样式 更改框样式。有关框样式的说明，请参见表 3.3。

5 数汇总 向图形添加中位数、最大值、最小值以及第 1 和第 3 四分位数之类的统计量。

宽度比例 调整箱线图的宽度。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对箱线图的红色小三角选项

置信菱形 显示或隐藏关于均值的置信菱形。菱形覆盖均值的第 95 置信区间。

锯齿状 显示一个箱线图，该图在锯齿跨越的中位数处呈锯齿状：

$$\text{median} \pm (1.58 \times \text{IQR}) / (\sqrt{n})$$

保护 在箱线图须线末端处显示或隐藏垂直线或围线。

最短半集 显示或隐藏一个括号，该括号指示包含 50% 数据的最短区间的位置。

最短半集颜色 支持您为最短半集括号选择颜色。

响应轴 （仅当 X 轴和 Y 轴变量的建模类型均为连续时才可用。）指定在构造箱线图时用作响应的变量的轴。仅当两个变量的建模类型均为连续时才可用。若只有一个变量是连续变量，其轴即为响应轴。“自动”设置的是 Y 轴。

箱放置 （可用于分组变量。）支持您对齐箱线图或调整箱线图偏差。

提示：当您的箱线图有分类分组变量时，右击分类轴并选择**大小依据 > 计数**可绘制与每个类别中的观测数成比例的箱线图。

表 3.3 框样式和说明

框样式	说明
正常	将四分位数定义的框显示为空心，仅在中位数处显示一条直线。

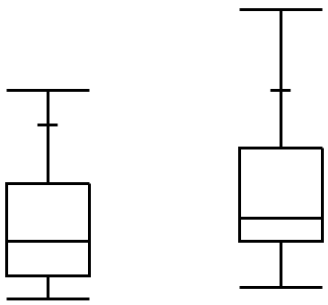
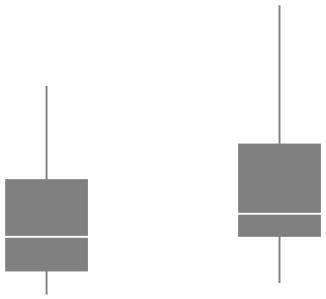
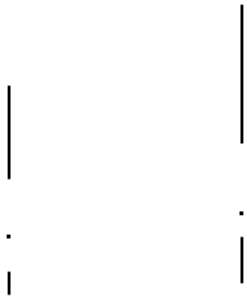


表 3.3 框样式和说明 (续)

框样式	说明
纯色	用纯色填充显示四分位数定义的框。中位数处的直线用空白区域来显示。 
细	不显示四分位数定义的框。因须线很容易识别该框。中位数用一个点来表示。 

直方图

直方图 元素使用分箱显示变量分布。有关直方图的详细信息，请参见《基本分析》。

注意：直方图的直条与包含变量的区域对应的轴上的刻度标对齐。若定制该轴，直方图将调整以便与新刻度标匹配。

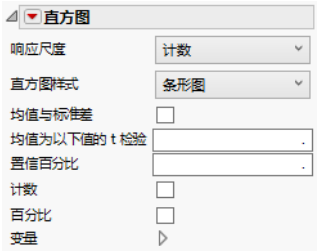
您可以使用 “响应轴”选项将图形的主方向改为 “X”（水平）、“Y”（垂直）或 “自动”。

- 若两个变量均为名义型或有序型，则直方图显示未指定为“响应轴”的轴上的变量水平中的观测数。

- 若其中一个变量是连续型并且另一个变量是名义型或有序型，则该图显示分类变量每个水平下的连续变量的直方图。
- 若为 *X* 和 *Y* 指定了相同的变量，则忽略 *Y* 角色，仅显示一个直方图。
- 若两个变量都是连续变量，则 X 轴上的变量被视为离散变量。您可能需要重新调整 X 轴的尺度以清晰查看图形。可使用包含“等高线”元素的散点图作为备选的可视化视图。

提示：您可以用透明颜色叠加直方图，以便看清各个直方图。为此，需要为相关变量分配 X 或 Y。将叠加变量分配到**叠加**区域。然后，点击**直方图**元素图标。请参见“图形生成器示例”一章中第 125 页的“叠加直方图和脊线图的示例”。

图 3.34 直方图选项



响应尺度 指定响应轴的尺度。选项包括计数、百分比或填充。填充没有响应轴标签。它填充显示空间，并独立缩放多个直方图。

叠加 （可用于分类响应。）调整垂直类别之间的直方图叠加量。

直方图样式 指定直方图样式。

条形图 默认的传统直方图直条，其高度基于落在每个直条或箱中的观测数或百分比。

多边形 连接每个直方图直条的峰值以构造用多边形表示的数据分布。

核密度 具有平滑线控制的密度曲线。

阴影图 叠加箱宽度不同的各个直方图。

均值与标准差 显示 X 或 Y 区域中各变量水平的均值和标准差。

均值为以下值的 t 检验 执行指定均值的 *t* 检验。

置信百分比 指定均值的置信区间。

计数 在直方图直条上显示计数。

百分比 在直方图直条上显示百分比。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。
– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对直方图的红色小三角选项

响应轴 指定在构造直方图时用作响应的变量的轴。仅当两个变量的建模类型均为“连续”时才可用。若只有一个变量是“连续”变量，另一轴即为响应轴。

热图

热图元素 使用颜色强度尺度显示各组观测的计数或平均值。若将单个变量标绘为 Y 或 X，则图显示为直条。若标绘 Y 和 X 这两个变量的交叉表，则图中会显示矩形。

- 对于分类变量，各水平定义各组观测。
- 对于连续变量，“图形生成器”构造非重叠的值区间。这些区间定义各组观测。

注意：若在构造图中直条或矩形时使用的所有行都隐藏并且未排除，则不显示该矩形，但仍保留该矩形占据的位置。

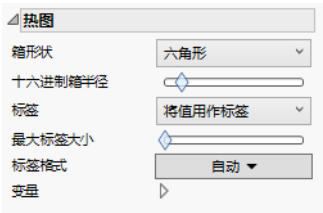
创建热图时，“图例”区域中会显示强度图例。请参见第 99 页的“离散或连续图例项”。强度图例显示计数或平均值：

- 若未分配“颜色”变量，则每个直条或矩形中的计数定义强度图例的尺度。
- 若分配了“颜色”变量，则每个直条或矩形中的观测所对应的这一变量的平均值定义强度图例的尺度。

提示：悬停在某个方格上方可查看标签。点击某个方格可选择相应的行。

要按照两个变量对热图着色或调整大小，请将第一个变量拖入“颜色”或“大小”区域，然后将第二个变量拖入该区域的顶部或底部一角。

图 3.35 热图选项



箱形状： （可用于连续变量）。指定箱的形状。

十六进制箱半径 （可用于六角形箱。）支持您调整箱的半径。

标签 指定热图的标签。您可以按照值、占合计的百分比或按行来添加标签。

最大标签大小 支持您调整标签大小。

单元格分级显示项 （在使用大小变量时可用。）支持您为原始箱大小添加分级显示项。这将突出显示箱的大小尺度。

标签格式 （可用于值和百分比标签。）支持您设置单元格标签的格式。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

使用热图应用背景颜色

若有图元素使用 “重叠”区域、“X 分组”或 “Y 分组”区域，您还可以使用热图控制背景颜色。为此，请执行以下操作：

1. 选择背景颜色对应的变量并将其拖至**颜色**区域。
2. 选择 “热图”元素 。
3. 构造图元素。

在“热图”面板中的“变量”下，取消选中除“颜色”复选框之外的所有复选框。

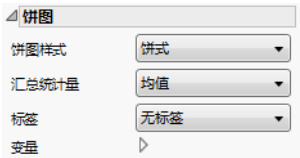
饼图

饼图元素 标绘各组观测的汇总统计量，将这些观测的值表示为饼图上的扇区或圆环的大小。

注意：若在构造饼图扇区时使用的任何行处于隐藏但未排除状态，则不显示该扇区，但仍保留该扇区占据的位置。

- 对于单个变量，扇区按每个类别中的观测数分类。
- 对于两个变量，Y 区域中的变量用于根据选定的汇总统计量调整扇区大小。“图例”区域中出现图例，显示 X 区域中变量的值。

图 3.36 饼图选项



添加“叠加”变量后，将为“叠加”变量的每个水平构造一个环形图。外环表示“值排序”中的最小值，内环表示最大值。

饼图样式 更改饼图的外观。有关饼图样式的说明，请参见表 3.4。

汇总统计量 更改标绘的统计量。

标签 在饼图中添加或删除标签。

标签格式 （可用于值和百分比标签。）支持您设置单元格标签的格式。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

表 3.4 饼图样式说明

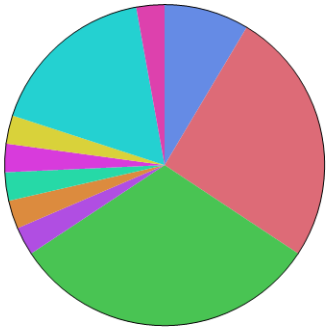
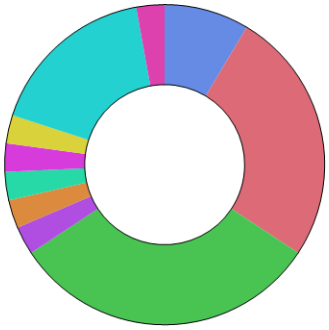
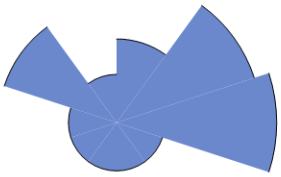
饼图样式	说明
饼式	传统饼图，各个扇区的大小由汇总统计量确定。 
环式	用一个同心环来表示分层变量的每个变量或水平。各个图段的大小由汇总统计量确定。环形图可使用同心环帮助形象演示分层数据。 

表 3.4 饼图样式说明 (续)

饼图样式	说明
鸡冠花式	所有扇区的中心角都相等。每个扇区的大小由汇总统计量确定。亦称**分解饼图**，鸡冠花图可以帮助您查看较小的区域。



矩形树图

矩形树图元素 将变量的水平或值显示为较大矩形显示区中的小矩形。矩形大小表示在一个或多个 X 变量的各个水平或值之间，**大小**区域中变量的汇总统计量的值。

要构造一个矩形树图，使其显示两个或更多 X 变量的各个水平之间的汇总值，请合并 X 区域中的变量。调整各个矩形的大小来表示汇总统计量的值，整体矩形布局中不留任何未使用的空间。

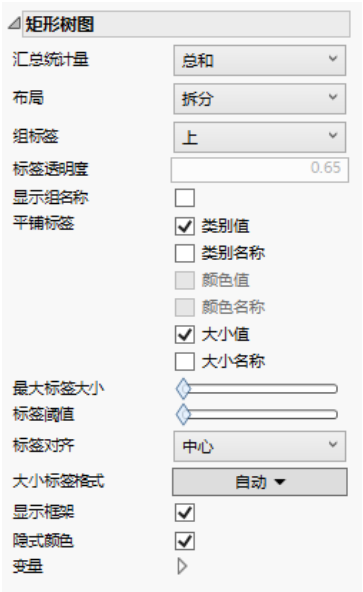
默认情况下，在使用“拆分”布局对矩形树图中的图块进行排序时使用数据表顺序。或者，可以指定一个或两个连续变量作为 Y 变量进行排序：

- 若指定一个 Y 变量，则它会对 X 轴和 Y 轴同时排序。
- 若指定两个 Y 变量，则第一个变量对 X 轴排序，第二个变量对 Y 轴排序。

若指定“方形化”布局，图块的顺序将由大小决定，而忽略排序变量。

注意：若只隐藏行而未排除行，则该图不受任何影响。

图 3.37 矩形树图选项



汇总统计量 指定对应于 Y 区域中的变量并且用于调整矩形大小的汇总统计量。

布局 尽可能按照 X 变量的值排序或矩形大小来排列各个矩形。

拆分 可按照 X 变量的水平或值的值排序来排列各个矩形。“拆分”为默认设置。

方形化 可按照汇总统计量的值（降序排列）排列各个矩形。这会将最大的矩形放在图的左上角，将最小的矩形放在图的右下角。

混合 仅当有两个或更多 X 变量时才适用。对外层变量应用“拆分”，对其他变量应用“方形化”。由此判定：大的分组按照其值排序来排序，而内切矩形按照汇总统计量值来排序。

组标签 （当有多个 X 变量时可用。）显示或隐藏每个类别上方或浮动在每个类别中心的其他组标签。

提示：要过滤组中数据，请点击某个组标签。

标签透明度 （可用于浮动组标签。）指定浮动组标签的透明度。

显示组名称 向组标签添加变量标题。

平铺标签 显示或隐藏以下标签：

类别值 X 变量的值。

类别名称 X 变量的名称。

颜色值 颜色变量的值。

颜色名称 “颜色”变量的名称。

大小值 “大小”变量的值，若没有“大小”变量则为 Y 变量的值（大小值）。

大小名称 “大小”变量的名称，若没有“大小”变量则为 Y 变量的值（大小名称）。

提示：悬停在某个矩形上方可查看详细信息。

最大标签大小 增大或减小标签。

标签阈值 基于矩形大小删除标签。默认情况下，所有标签都显示。将滑块移至右侧可删除逐渐增大的矩形的标签。

标签对齐 将标签置于左、中、右位置。

颜色标签格式 （当使用颜色变量时可用。）支持您设置基于颜色变量的单元格标签的格式。

大小标签格式 （当选定“大小值”时可用。）支持您设置基于大小值变量的单元格标签的格式。

显示框架 显示或隐藏矩形的边框。

提示：使用右击出现的定制菜单中的矩形树图选项调整框架颜色、样式和厚度。

隐式颜色 显示或隐藏颜色。若没有在 “颜色”区域中指定变量，则所有矩形都具有相同的颜色。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

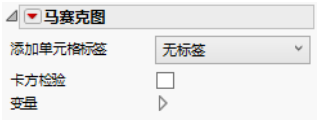
马赛克图

马赛克图元素 使用矩形大小表示某个变量的类别中的观测在另一个变量的不同类别之间的比例。尽管变量可以是分类或连续的，但它们将被视为分类变量。有关马赛克图的详细信息，请参见《基本分析》。

默认“响应轴”为 Y 轴。这意味着，对于 X 变量的每个水平，垂直堆叠直条都显示 Y 变量的每个水平中的值的比例。您可以将图形的主方向改为 X（水平）。

注意：若只隐藏行而未排除行，则该图不受任何影响。

图 3.38 马赛克图选项



添加单元格标签 基于计数、百分比、值或行向单元格添加标签。

标签格式 （可用于百分比或值标签。）支持您设置单元格标签的格式。

卡方检验 执行 Pearson 卡方检验并将结果显示在图形上的标签中。该选项仅在每侧有 2 个或 2 个以上水平时才显示。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

– 显示或隐藏对应于区域中某个变量的元素。

– 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对马赛克图的红色小三角选项

响应轴 将与响应变量关联的轴改为 “X”（水平）、“Y”（垂直）或 “自动”（Y 轴表示马赛克图）。响应轴上的变量用于计算该变量在另一个轴上的每个水平中的观测比例。

说明文字框

说明文字框元素 显示图上的汇总统计量。该汇总统计量对应于您在“说明文字框”选项中指定为“响应轴”的轴所对应的变量。

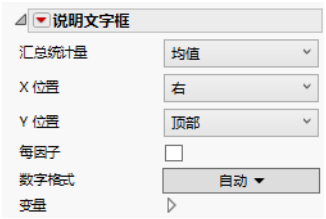
分类变量的汇总统计量

对于分类变量，通过向分类变量的水平分配数值来计算显示的汇总统计量。这些水平按照它们的值排序定义的顺序来排列。为这些水平分配从 0 到水平数减 1 的整数值。

注意：在与 Y 和“颜色”之类的汇总角色关联的区域中使用分类值时，将基于为变量水平分配的数值计算汇总统计量。（整数值从 0 到水平数减 1。）最终结果随后作为一个水平映射回相应的分类值或值范围。这种方法允许在汇总中使用有序型变量，但可能不会为名义型变量生成明确结果。

有关演示“说明文字框”元素的示例，请参见第 30 页的“图形生成器中的功能的示例”。

图 3.39 说明文字框选项



汇总统计量 指定要为“响应轴”设置所定义的变量显示的汇总统计量。默认的汇总统计量为“均值”。有关如何为分类变量计算汇总统计量的详细信息，请参见第 83 页的“分类变量的汇总统计量”。

X 位置 指定说明文字的水平位置。

Y 位置 指定说明文字的垂直位置。

每因子 为每个 X 或每个 Y（若为水平）显示一个说明文字。

数字格式 支持您设置汇总统计量的格式。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于“X 分组”、“Y 分组”、“重叠”或“页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对标题框的红色小三角选项

响应轴 为要显示汇总统计量的变量指定轴。默认指定 Y 轴。请注意，该轴可不同于定义该图所指定的响应轴。

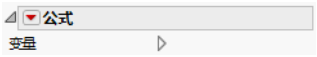
公式

公式元素 显示某个函数、其逆函数或参数曲线的图形。使用列公式来定义一个或多个函数。

- $y = f(x)$：
  - 要标绘单个参数在其单个参数的不同值下的函数，请将包含该公式的列拖至 Y 区域，将包含其单个参数的值的列拖至 X 区域。
  - 确保“响应轴”设置为“自动”或“Y”。
- $y = f^{-1}(x)$：
  - 要标绘某列值的单个参数的逆函数，请将包含该公式的列拖至 X 区域。将包含关注值的列拖至 Y 区域。
  - 将“响应轴”设置为“X”。
- $x = f(t)$ 和 $y = g(t)$：要标绘两个函数定义的参数曲线，请将其中一个拖至 X 区域，将另一个拖至 Y 区域。

注意：由于“公式”元素标绘函数的值，所以即便有隐藏但未排除的行，该图也不会受到任何影响。

图 3.40 公式选项



变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于 “X 分组”、“Y 分组”、“重叠”或 “页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

针对公式的红色小三角选项

响应轴 指定带公式变量的轴。

地图形状

“地图形状”元素 显示“地图形状”区域中的变量所定义的区域。

若数据表中的某列包含地理区域（如国家/地区、州、省、县）的名称，您可以将该列分配至“地图形状”区域。当某个变量放入“地图形状”区域后，“图形生成器”会搜索与该列中的值对应的内部地图。若找到合适的地图，就会在图形区域中绘制该地图。

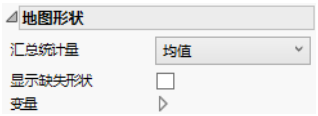
或者，您可以为相关列定义“地图角色”列属性。该属性指示 JMP 如何使用地图形状数据连接该列中的各值。该属性在创建定制地图时尤其有用。请参见“地图”一章中第 235 页的“定制地图文件”。

若“地图形状”区域中有变量，X 和 Y 区域将消失。您可以使用“颜色”和“大小”从其他变量添加信息。

- 要按照汇总统计量的值对地图形状着色，请将相关列拖至**颜色**区域。在“首选项”中选定的分类或连续颜色主题会应用于每个形状。
- 要按照汇总统计量的值调整地图形状的大小，请将相关列拖至**大小**区域。这将按照大小变量的汇总统计量值统一地图形状的尺度，将失真最小化。

请参见第 225 页的“地图”一章。例如，请参见“地图”一章中第 246 页的“创建地图的示例”，或运行以下样本数据表中的关联脚本：PopulationByMSA.jmp 或 SAT.jmp。

图 3.41 地图形状选项



汇总统计量 更改标绘为“颜色”或“大小”变量的统计量。

显示缺失形状 显示或隐藏地图中的缺失数据。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于“X 分组”、“Y 分组”、“重叠”或“页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

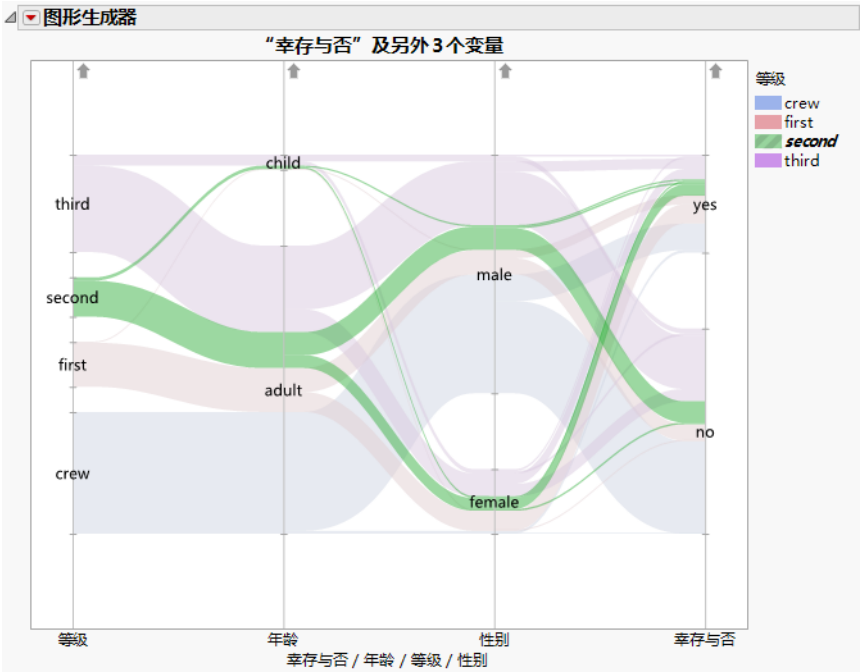
平行

平行元素 连接跨两个或更多变量的某行中的值。将两个或更多变量一同拖至 X 或 Y 区域。变量名显示为其被拖至的区域中的轴标签。

- 每个连续变量的值都沿着与另一轴平行的线（轴）标绘。
- 分类变量的水平用与另一轴平行的线（轴）上的区间表示。区间大小与分类变量每个水平中的观测数成正比。
- 值（对于连续变量）或数据带（对于分类变量）使用连续曲线连接。

若所有变量都是分类变量，则分类变量水平的每个组合都对应一个数据带。数据带随着从左到右移动而拆分。在对应于分类变量的第一个区间中，该分类变量的每个水平都对应一个数据带。在对应于分类变量的最后一个区间中，该分类变量的每个可能组合都对应一个数据带。

图 3.42 使用 Titanic.jmp 的分类带的示例



在图 3.42 中，已选定包含所有二等舱乘客的数据带。平行图显示多数人是成人、男性多于女性，幸存人数比遇难人数略多。

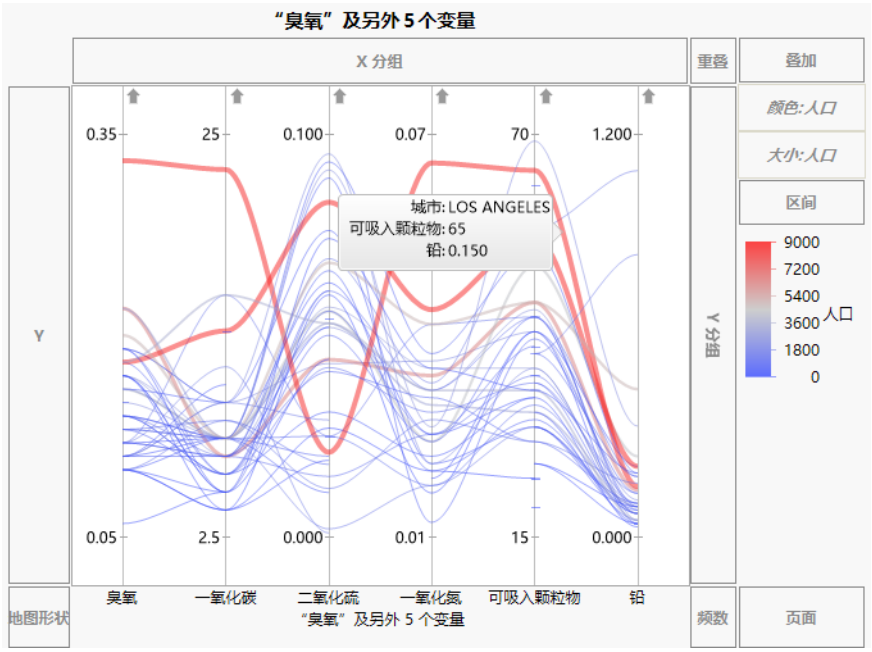
使用 “曲线”选项用直线或曲线连接变量的各个值或水平。

- 要按照变量值对曲线着色，请将相关列拖至**颜色**区域。在“首选项”中选定的分类或连续颜色主题显示在“图例”区域中。
- 要按照变量值调整曲线大小，请将相关列拖至**大小**区域。
- 要更改变量轴的方向，请点击表示变量轴的垂线顶部的箭头。
- 要移动变量及其轴，请点击并拖动该轴。

提示：悬停在曲线上方可看到一个标签，该标签提供关于相应行的信息。

图 3.43 显示 Cities.jmp 数据表中的六个变量的平行图。变量**人口**同时用作“颜色”和“大小”变量。为对应于“Los Angeles”的曲线加注了标签。

图 3.43 Cities.jmp 中污染数据的平行图



提示：

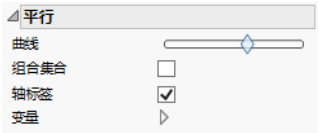
- 可以向平行图中添加线图元素，以显示平行图中的汇总关系。
- 您可以为规格限添加参考线。相关信息，请参见《使用 JMP》。

统一尺度

默认情况下，变量值的尺度要经过调整，以便在同一水平下标绘最小值和最大值。例如，在图 3.43 中，每个变量的值都具有相同的垂直散布。每条垂线都用变量的最小值和最大值来标注。

在图 3.43 中，**一氧化碳**和**可吸入颗粒物**的尺度与其他变量的尺度显著不同。若在极为不同的尺度上测量变量，该统一尺度可支持您清晰地查看差异。

图 3.44 平行选项



曲线 调整连接各点的曲线的曲率。将滑块一直推到左侧可表示直线。将滑块右移时，曲率会增加。

组合集合 （仅当有三个或更多分类因子时才适用。）对于分类变量，若未选定“组合集合”，数据带在从左向右移动时拆分。在对应于分类变量的最后一个区间中，分类变量的每个可能组合都对应一个数据带。选择“组合集合”选项可使数据带不拆分。跟随分类变量的每个区间都为该变量的每个水平显示一个数据带。

轴标签 从显示中删除轴标签。

变量 显示或隐藏变量的图形元素，或重新排列变量的显示顺序。

注意：这些选项不适用于“X 分组”、“Y 分组”、“重叠”或“页面”区域中的变量。

复选框后跟区域指定和变量的名称。使用复选框执行以下操作：

- 显示或隐藏对应于区域中某个变量的元素。
- 添加或删除将“颜色”、“大小”、“形状”或“频数”变量应用到区域中变量的效果。

提示：若您有多个图形，可以使用不同变量为每个图形着色或调整大小。将第二个变量拖到**颜色**或**大小**区域，然后将它放到角落。在“变量”选项中，选择要应用到每个图形的特定颜色或大小变量。

若区域中有多个变量，可使用箭头来重新排列其显示顺序。突出显示某个变量名称并点击箭头来对它调整位置。

有关使用变量的示例，请参见“图形生成器示例”一章中第 108 页的“面积图和线图示例”。

图形区域

“图形生成器”窗口中的主要元素是图形区域。图形区域包含多个拖放区，您可以将变量从“变量”框拖放到拖放区中。

提示：若将变量放入图形中心区域，JMP 将根据变量是连续型、有序型还是名义型来猜测要将变量放入的拖放区。

您可以将变量拖入两种区域：

- **数据区域**包括 X、Y、**地图形状**、**频数**、**颜色**、**大小**和**区间**。X、Y 和**地图形状**区域均与位置相关，并且影响可用的图形元素类型。可在**频数**、**颜色**、**大小**和**区间**区域中修改特定的图形元素。
- **分组区域**可将数据划分为各个子集，为每个可能的子集都重复该图形。分组区域包括 X 分组、Y 分组、**重叠**、**叠加**和**页面**。

“图形生成器”提供以下区域：

表 3.5 “图形生成器”区域

区域	说明
X 和 Y	构造基于您放入这些区域的变量的值、水平或类别的图。将 X 区域用于水平布局，将 Y 区域用于垂直布局。

表 3.5 “图形生成器”区域 (续)

X 分组和 Y 分组	按照名义型或有序型变量的水平对数据分层，并且为每个水平构造一个图。X 分组生成水平显示，Y 分组生成垂直显示。 您放入 X 分组和 Y 分组的变量的类型还决定显示的以下方面： • 添加连续变量可创建非重叠区间子组并为每个区间中的值绘图。 • 添加分类变量可为分类变量的每个水平创建图。 您可以向 X 分组或 Y 分组添加多个变量： • 要将变量添加到分组区域之一中的某个变量**上方**，将其拖至“X 分组”区域中该变量的左侧或“Y 分组”区域中该变量的上方。 • 要将变量添加到分组区域之一中的某个变量**下方**，将其拖至“X 分组”区域中该变量的右侧或“Y 分组”区域中该变量的下方。 要按照两个变量的水平来分层，请将变量同时拖入“X 分组”和“Y 分组”。要更改显示的水平数，请右击轴边框并选择“视图中的水平数”。有关所有选项的说明，请参见第 97 页的“用于轴、变量标签或图形标题的选项”。
地图形状	添加地图。变量必须包含地理位置名称，如：国家、地区、州或省，或者您必须在定制形状文件中提供信息。请注意以下几点： • 若拖放标为**纬度**和**经度**的列，JMP 会自动将其相应分配至 Y 和 X 角色。 • 将变量放入**地图形状**区域后，X 和 Y 区域将消失。 • “地图形状”元素可配合“点”元素使用，共同将点放到形状中。 有关创建地图形状的更多详细信息，请参见第 86 页的“针对公式的红色小三角选项”和“地图”一章中第 229 页的“图形生成器”。

表 3.5 “图形生成器”区域 (续)

重叠	为分类变量的各个水平或为连续变量的区间子组创建多行图。若分层变量有许多水平，则使用“重叠”。要更改显示的水平数，请右击轴边框并选择**视图中的水平数**。有关所有选项的说明，请参见第 96 页的“分组区域选项”。 **注意：**一旦将某个变量放到此处，就不能将任何变量放入 **X 分组**。
叠加	根据“叠加”变量的水平数划分图变量的子组并对其着色。 • 添加名义型或有序型变量可按照变量的水平对图元素分层和着色。 • 添加连续变量可创建区间子组。根据子组对图元素分层和着色。 图右侧会显示一个图例。右击图例项以显示定制选项。
颜色	向点、地图形状和其他对象添加颜色。 • 添加名义型或有序型变量可按照变量的水平对图元素着色。 • 添加连续变量可按照强度尺度对图元素着色。 图右侧会显示一个图例。右击图例可更改颜色或定制强度尺度。 **提示：**一旦在“颜色”区域中添加了变量，就可以使用元素属性面板中的“变量”选项显示或隐藏颜色。
大小	按照汇总统计量或其他大小变量调整图形元素的大小。 图右侧会显示一个图例。右击图例项以显示定制选项。
区间	添加区间变量以绘制定制误差条。 • 单个区间变量被视为 delta，用来计算区间的上限和下限。 • 两个区间变量被视为区间的下限和上限值。
频数	添加频数或权重变量。当所有值都是整数值时，“频数”变量被视为频数，否则将被视为权重。页脚注释指示如何处理变量。“频数”变量影响汇总统计量。

表 3.5 “图形生成器”区域 (续)

页面	将“依据”分组变量放置到“页面”区域可在单独的图形中显示分组的每个水平。默认情况下，“页面”区域限制为 200 页。显示前 200 个类别。在“图形生成器”首选项中调整该限值。请参见《使用 JMP》。 **提示：** • 一旦点击“页面”，就可以选择红色小三角菜单选项“关联页面轴”。使用该选项可关联或取消关联“页面”区域中“依据”分组变量各水平间的图形轴尺度。 • 要使用选择工具复制选定的单个图，首先将图形生成器输出保存到记录中（编辑 > 记录）。

有关图例的详细信息，请参见第 98 页的“图例选项”。

注意：在与 Y 和“颜色”之类的汇总角色关联的区域中使用分类值时，将基于为变量水平分配的数值（从 0 到水平数减 1 的整数值）计算汇总统计量。最终结果随后作为一个水平映射回相应的分类值或值范围。这种方法允许在汇总中使用有序型变量，但可能不会为名义型变量生成明确结果。

“X 分组”或“Y 分组”区域中的分类和连续变量

向 **X 分组**或 **Y 分组**区域添加分类变量时，将为每个变量水平创建一个分区。

向分组区域添加连续变量时，“图形生成器”使用数据的分位数将变量划分为五组。要更改分组数量，请执行以下操作：

1. 右击分组变量标签并选择**水平数**。
2. 键入您想要显示的水平数。
3. 点击**确定**。

移动 X 分组和 Y 分组标签

分组变量标签可重新放置到图形上的其他位置。**X 分组**标签可以放置在图形顶部或底部。**Y 分组**标签可以放置在图形左侧或右侧。

- 要重新确定 **X 分组**标签的位置，请右击 **X 分组**区域中的变量，然后选择 **X 分组边缘** > **顶部**或**底部**。
- 要重新确定 **Y 分组**标签的位置，请右击 **Y 分组**区域中的变量，然后选择 **Y 分组边缘** > **左侧**或**右侧**。

红色小三角选项

“图形生成器”的红色小三角菜单包含以下选项：

显示控制面板 显示或隐藏平台按钮、“选择列”框和拖放区边框。

显示

标题 显示或隐藏图形标题。右击以更改标题的对齐方式或跨度。

副标题 显示或隐藏图形副标题。右击以更改副标题的对齐方式或跨度。

图例 显示或隐藏图例。

页脚 显示或隐藏页脚，页脚中包含缺失地图形状、误差条注释、频数注释和 WHERE 子句之类的说明性消息。

X 轴 显示或隐藏 X 轴。

Y 轴 显示或隐藏 Y 轴。

X 轴标题 显示或隐藏 X 轴标题。

Y 轴标题 显示或隐藏 Y 轴标题。

图例位置 设置图例的位置。默认情况下，图例显示在右侧。将图例置于底部时，图例会放置在图形下方正中央。图例项也随之水平显示，而不是垂直显示。或者，您可以将图例放在图形内部、左侧或右侧。

图例设置 打开一个窗口，您可在其中修改标题、字体和标题位置之类的图例设置。

颜色设置 包含以下选项：

连续颜色主题 选择用于连续变量的颜色主题。

分类颜色主题 选择用于分类变量的颜色主题。

注意：有关颜色主题的详细信息，请参见 《使用 JMP》。

加亮大面积填充 自动为填充大面积区域（如饼图、矩形树图和马赛克图）的元素加亮颜色。

为水平使用行颜色 当每个水平具有唯一颜色时，使用行颜色初始化图例水平。

锁定尺度 防止轴尺度和梯度图例尺度因数据变化或过滤更改而自动调整。

误差条剩余偏差 调整叠加误差条的剩余偏差。

关联页面轴 关联或取消关联 “页面”区域中 “依据”分组变量各水平间的图形轴尺度。

适应窗口大小 指定是否在您调整 JMP 窗口大小时调整图形的大小。默认设置为 “自动”，该设置通常设置为“开”，除非该窗口与其他内容共享或是存在 “页面”变量。要始终使图形适应窗口大小，请保持该设置为 “开”。您还可以保持纵横比。要防止图形调整大小，请将设置改为 “关”。

抽样 使用随机数据样本以加快图形绘制。若样本大小为零或大于等于数据表中的行数，则禁用抽样。

图形间距 设置图形面板间的空白量。

包括缺失类别 支持图形收集并显示分类变量的缺失值。

设置 Alpha 水平 支持您为使用置信区间的 alpha 值的图形和统计量设置 Alpha 水平。

启动分析 启动带有指定角色的图中的变量的 “拟合模型”平台。若只放置了一个变量，该选项将启动 “分布”平台。

制成数据表 创建包含图形结果的新数据表。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

“图形生成器”按钮

“图形生成器”窗口包含以下按钮：

重新调用 用您上次创建的图形填充 “图形生成器”窗口。执行该操作后，该按钮将变成 “撤销”按钮。

对话框 打开 “图形生成器”启动窗口。执行该操作后，该按钮将变成 “重新开始”按钮。您可以在 “图形生成器”启动窗口中添加 “依据”变量。

完成 隐藏按钮、“列”列表和所有拖放区分级显示项。等同于取消选择“显示控制面板”选项。

使用这种适于演示的格式以便粘贴到其他应用程序中。要复制图形，请选择**编辑** > **复制**。要将窗口还原为交互模式，请点击 “图形生成器”红色小三角菜单上的**显示控制面板**。

撤销 撤销上次对该窗口所做的更改。

重新开始 可使该窗口恢复默认状态，从窗口中删除所有数据和图形元素并从拖放区中删除所有变量。

右击菜单

- “图形选项”
- “分组区域选项”
- “用于“变量”面板中的列的选项”

- “用于轴、变量标签或图形标题的选项”
- “用于多个图形之间的分割线的选项”
- “图例选项”

对于多个图形 （使用分组区域或 “重叠” 区域创建），请注意以下事项：

- 通常，您对一个图形元素应用的任何选项都会应用到所有分组和重叠变量应用到的所有图形。
- 要仅对一个图形应用某个选项，在该图形区域中右击并从 “图形” 菜单中选择该选项。
- 要将某个选项同时应用到多个图形，请首先按住 Ctrl 键并右击，然后从 “图形” 菜单中选择该选项。

图形选项

右击图形将显示以下信息：

1. 图中显示的元素类型（如：点、线等）。子菜单提供特定于每个元素类型的选项以及常规选项。或者，您可以在每个元素的属性面板中（该面板位于 “变量” 面板之下）更改元素特定的选项。
2. “添加” 选项，用于选择向图形添加的其他元素。或者，您可以通过拖动所需的元素图标向图形添加更多元素。
3. “行”、“图形”、“定制” 和 “编辑” 菜单。有关这些选项的说明，请参见 《使用 JMP》。

提示：选择**行 > 在列中指明选择项**以创建要命名的新列并且为图形中的选定点添加标签。

分组区域选项

右击 “X 分组”、“Y 分组”、“重叠”、“颜色”、“大小” 和 “叠加” 区域可显示以下选项：

视图中的水平数 更改分组变量的可见水平数。使用箭头按钮可前后滚动。

水平数 更改水平数。请参见第 93 页的 “‘X 分组’ 或 ‘Y 分组’ 区域中的分类和连续变量”。

排序依据 对变量水平排序。请参见第 45 页的 “对分类变量水平排序”。

显示标题 显示或隐藏变量标题。

标题方向 将变量文本的方向改为水平或垂直。

水平方向 将水平值的方向改为水平或垂直。

每行的水平数 更改图形中包括的列数。与**重叠**变量一起使用。

颜色 更改分组区域的背景颜色。

X 或 Y 分组边缘 移动分组变量标签。请参见第 93 页的 “移动 X 分组和 Y 分组标签”。

汇总统计量 基于汇总统计量应用颜色或大小变量。仅可用于汇总的 Y 响应。

交换 交换两个变量的位置。请参见第 37 页的 “移动或删除区域中的变量”。

删除 删除变量。

用于“变量”面板中的列的选项

右击“变量”面板中的某列可显示依赖于该列的建模类型的选项。有关这些菜单的详细信息，请参见《使用 JMP》。

用于轴、变量标签或图形标题的选项

右击轴、变量标签或图形标题可显示特定于 “图形生成器”的选项和常规选项：从右击菜单中选择**显示属性**，通过访问属性面板来进行更多定制操作。以下选项特定于 “图形生成器”：

对齐 更改图形标题对齐方式。

跨度 更改图形标题跨度的宽度。

合并尺度 当同一区域中有多个变量时合并尺度。自动执行以下操作：

- 若一个轴上有 4 个或更多变量，而另一个轴上没有变量，支持平行模式的元素将合并为平行尺度。若所有变量都是连续变量且重叠，则使用“平行合并”。
- 若所有变量都是分类变量，则使用“嵌套”。

此外，考虑以下事项：

- “合并”将创建一个覆盖所有变量的尺度。
- “嵌套”将创建一个嵌套尺度，每个变量一个水平。
- “平行合并”适用于有共享尺度的平行坐标。您可以将该选项与以下元素配合使用：点、等高线（小提琴图形）、直方图、箱线图和平行图。
- “平行独立”适用于有独立尺度的平行坐标。您可以将该选项与以下元素配合使用：点、等高线（小提琴图形）、直方图、箱线图和平行图。

日期 / 时间箱 （仅对日期或时间轴变量显示。）支持您选择一种粒度水平以便对数据分箱。例如，您可以选择在较高水平上按年份分箱，或在较低水平上按秒分箱。

日期 / 时间重叠 （仅对日期或时间轴变量显示。）支持您为日期数据选择周期。

交换 将变量从一个区域切换到另一个区域。选择您想切换位置的变量。

删除 从区域中删除变量。

右移 （仅当 Y 区域中有不止一个变量时才显示。）为您选择的变量创建另一个 Y 轴（在右侧）。

排序依据 （仅当 X 或 Y 区域中有分类变量时才显示。）对分类变量水平排序。请参见第 45 页的 “对分类变量水平排序”。

注意：选择**其他**指示连续变量作为排序变量。

次序统计量 支持您选择要用于排序的统计量。

保存值顺序 （定义排序后即可用。）将定义的顺序保存到 “值顺序”列属性。

注意：可以从 “图形生成器”中编辑值排序。在轴上右击并选择**编辑值顺序**。

大小依据： （可用于分类轴。）支持您通过组内的观测计数或数值列来调整箱线图的大小。

大小统计量 （当“大小依据”选项将数值列指示为大小调整变量时可用。）支持您选择要用于调整大小的统计量。

有关该线以下的常规选项的说明，请参见《使用 JMP》。

提示：对于其他轴定制，例如刻度线粗细或颜色，请从右击菜单中选择**显示属性**。

用于多个图形之间的分割线的选项

仅当图形区域中有两个或更多图形时，该菜单才显示。右击分隔图形的分割线。随即显示用于更改分割线格式（如，颜色、透明度、间距和边框）的选项。

提示：若图形之间的分割线太细，您可以调整其粗细。点击“图形生成器”红色小三角，选择**图形间距**，然后输入较大的值。

图例选项

图例由线条、填充颜色和梯度之类的元素构成。右击图例元素以显示定制属性。这些属性包括颜色和标签选项（若可用）。在“图例设置”窗口中可以找到其他图例设置。

要更改图例设置或恢复初始设置，请右击图例右侧或标题 （若适用） 。要撤销最近对图例进行的更改，请点击 “撤销”。

提示：若同时具有一个叠加变量和一个颜色变量，则颜色变量水平将由图例中的圆形颜色图标显示。

图例设置 更改图形元素 （线、直条等）的特定方面 （如：颜色、填充、透明度和梯度）。您还可以向图例添加标题。请参见第 99 页的 ““图例设置”窗口”。

恢复图例 将图例恢复为其初始设置。

移动图例

要移动图例的位置，请点击并拖至所需位置。该位置可以是对应于可从“图形生成器”红色小三角菜单或图例访问的位置的一个热点，也可以是图形中的任何其他位置。

要使用菜单选项，请点击“图形生成器”红色小三角并选择**图例位置**。要令图例垂直或水平显示，请在“图例设置”窗口中更改“项方向”。

“图例设置”窗口

通过“图例设置”窗口更改图例的各个方面。

图 3.45 使用 Hybrid Fuel Economy.jmp 的 “图例设置”窗口的示例

标题 图例的名称。

复选框 显示或隐藏图例中的项。 JMP 隐藏看起来冗余的项，但您可以通过选定这些项令其显示出来。若图例有大量条目，有些可能隐藏在“图形生成器”窗口中以确保为图形留出足够的空间。不过，所有条目都显示在这里，所以您可以选择希望显示的条目。要选择或取消选择所有条目，请右击并选择**切换复选框**。

上箭头和下箭头 更改图例中项的顺序。

颜色主题 选择不同的颜色主题。有关颜色主题的详细信息，请参见 《使用 JMP》。

标题位置 将图例标题置于图例中项的上部或左侧。

项方向 水平或垂直显示图例。

换行 / 列的项数目 将图例设置为 *n* 项高 （若为垂直）或 *n* 项宽 （若为水平）。

字体 更改字体类型、样式和大小。

预览 显示对图例的更改。

确定 提交对图例的更改。

取消 取消对图例的更改。

帮助 打开联机帮助。

离散或连续图例项

图例可列出离散值或显示连续值的尺度：

- 离散图例可包含分类变量水平或变量列表。
- 连续图例描述一定范围内的数值的颜色强度尺度。

右击以上任一图例中的项可查看以下选项：

线条颜色 （仅限离散图例。）更改与项目关联的元素的颜色。

标记 更改与项目关联的标记的类型。

标记大小 更改与项目关联的标记的大小。

模式或样式 更改与项目关联的元素的填充或模式。

线条粗细 更改与项目关联的线条的粗细。

填充颜色 为项指定颜色。

按主题着色 按颜色主题中的水平对多个等高线着色。该选项为密度等高线设置线条颜色和填充颜色。

透明度 更改标记或标签的透明度。输入在图形上绘制标记 （点）的透明度水平。不透明度的范围介于 0 （清晰）到 1 （不透明）之间。

梯度 （仅限连续图例。）更改梯度设置。请参见第 100 页的“梯度设置”。

密度梯度 （仅限等高线元素。）更改梯度颜色：

渐变为白色 最高密度等高线使用填充颜色，颜色在密度变低时渐变为白色。

渐变为灰色 最高密度等高线使用填充颜色，颜色在密度变低时渐变为灰色。

全色 密度等高线使用预定义或定制的颜色主题和全色控制。

梯度透明度 （仅限等高线元素。）更改梯度的透明度：

无 密度等高线对所有水平都使用一种透明度水平。

线性 密度等高线在密度变低时逐渐变得更透明。

恢复 恢复默认设置

梯度设置

您可以更改“梯度设置”窗口中的梯度图例的各个方面。

图 3.46 “梯度设置”窗口

要设置图例中标签的格式，请使用该菜单、“宽度”框和 “小数位数”框，以及该窗口左上角的**使用千位分隔符 (,)** 复选框。

颜色主题 更改颜色主题或定义定制颜色主题。有关颜色选项的详细信息，请参见 《使用 JMP》。

亮度范围 设置梯度颜色的强度范围。

标签数 指定图例的标签数。值为 0 则提供默认的标签数。

显示缺失值颜色 指定是否在图例中显示缺失值颜色。选择 “自动”仅当有缺失值时才显示颜色，选择 “关闭”从不显示颜色，选择 “打开”则始终显示颜色。

尺度类型 设置梯度颜色的尺度。在 “最大值”选项下面查找关于尺度类型的说明。

- **线性** 尺度在“最小值”与“中心值”之间以及“中心值”与“最大值”之间是分段线性的。这是默认尺度。
- **分位数** 对于图例所表示的变量的分位数，尺度是分段线性的。
- **标准差** 图例范围细分为由标准差增量确定的均值的偏移。尺度在这些偏移设置之间是分段线性的。偏移数由“标签数”确定。
- **对数** 对于图例所表示的值的对数，尺度是线性的。
- **对数剩余偏差** 对于图例所表示的值的以 10 为底的对数的剩余偏差，尺度是线性的。

范围类型 设置图例值的范围。

- **默认** 除非您指定了“最小值”和“最大值”，否则最小值和最大值包含数据范围并且选定适当的取修约值。

精确数据范围 最小值和最大值是数据的精确最小值和最大值，或是您指定为“最小值”和“最大值”的值。

中间 90% 最小值和最大值分别是第 5 分位数和第 95 分位数，或是您指定为“最小值”和“最大值”的值。该选项可防止离群值。

最小值，中心值，最大值 指定用于图例尺度的最小值、中间值和最大值。

注意：这些值可覆盖 “范围类型”值。

水平 将图例的方向改为水平。

反转颜色 在颜色方案中反转颜色。

反转尺度 反转颜色方案的尺度。

离散颜色 将颜色方案从连续梯度改为具有离散颜色的不连续梯度。

显示标签 显示或隐藏图例标签。

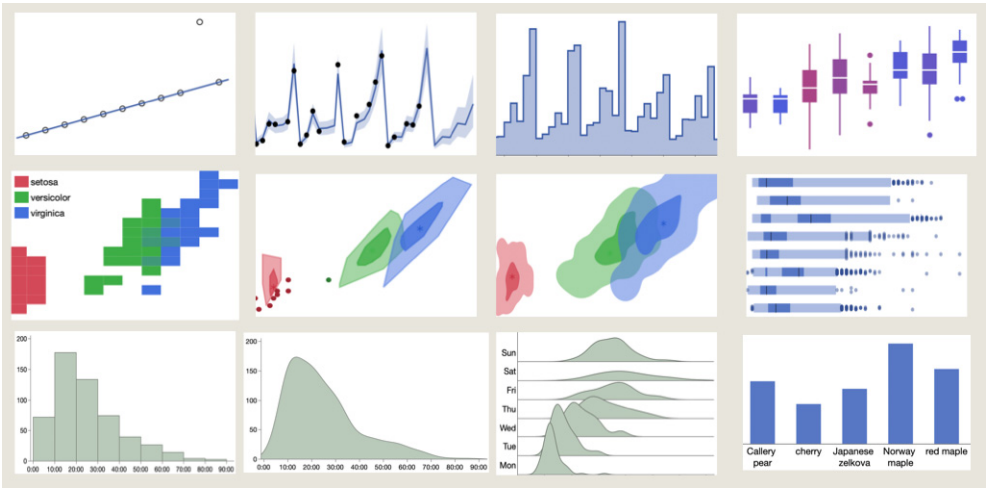
第 4 章

图形生成器示例

交互式数据可视化

“图形生成器”为您提供多种方式来直观演示和探索数据。本章提供了使用条形图、直方图、面积图、地图和其他类型的图进行数据探索的详细示例。

图 4.1 在 “图形生成器”中创建的图形



目录

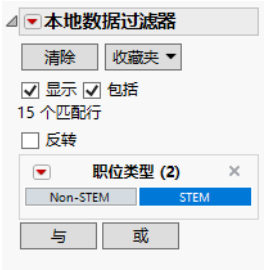
有序条形图示例 105
面积图和线图示例 108
地图示例 112
蝴蝶图示例 115
散点图示例 119
堆叠填充的条形图示例 124
叠加直方图和脊线图的示例 125
弹道条形图示例 127
预测图示例 129

有序条形图示例

数据	本例使用 2012 届大学毕业生的数据。该表包含 15 个研究领域的毕业生学历统计信息。每个学位都按毕业生的职位类型细分。针对学历、学历类型和职位类型（其中的类型为 STEM 或与 STEM 无关）提供了分类列。连续列**数量**包含来自学历和职位类型的每个唯一组合的毕业生人数。 **注意：**这些数据来自美国人口普查局 (2012)。
方法	本例使用颜色、本地数据过滤器和排序等选项。
目标	本例的目标是根据毕业生的学位类型，直观演示从事 STEM 相关工作的毕业生百分比。

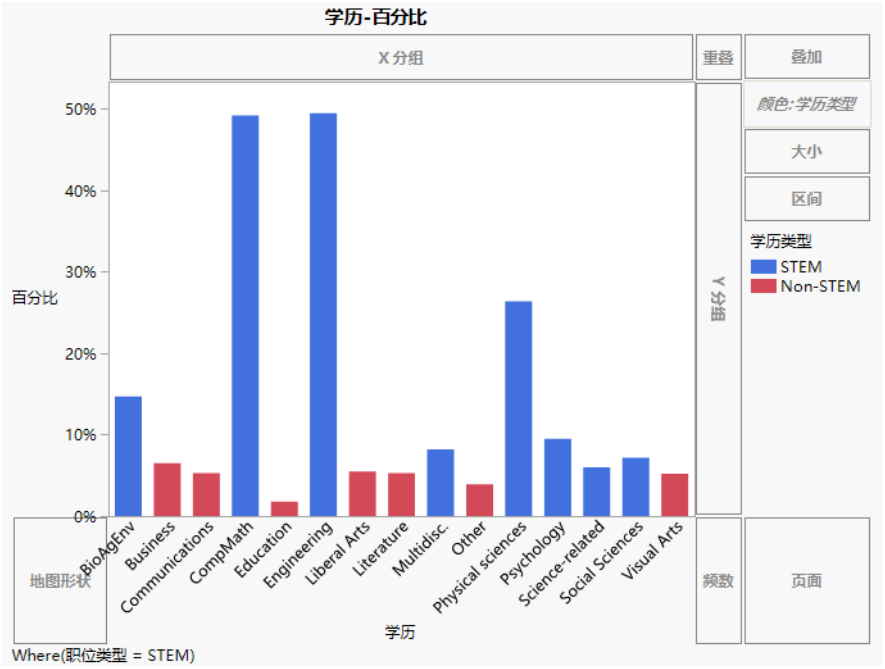
1. 选择**帮助** > **样本数据库**，然后打开 STEM Jobs.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**学历**并将其拖至 X 区域。
4. 选择**百分比**并将其拖至 Y 区域。
5. 点击“条形图”元素 。
6. 点击“图形生成器”红色小三角并选择**本地数据过滤器**。在“本地数据过滤器”中，执行以下步骤：
   a. 点击**职位类型**，然后点击 。
   b. 点击 STEM。

图 4.2 “本地数据过滤器”中的选择内容



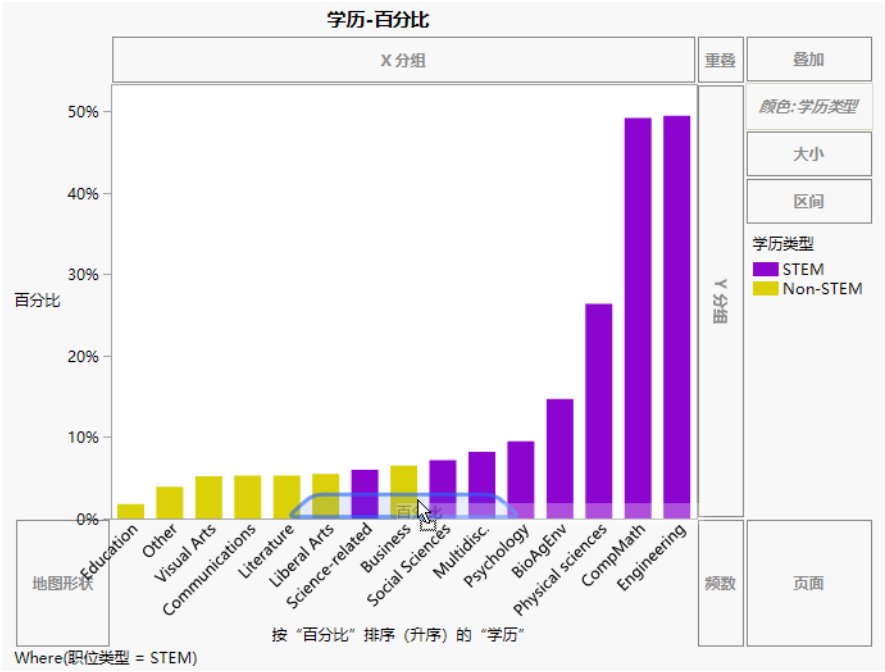
7. 在“图形生成器”列列表中，选择**学历类型**并将其拖至**颜色**区域。

图 4.3 按 “学历类型”着色的条形图



8. 在图例中，根据学历类型更改条形图颜色：
   - 右击 STEM 并将填充颜色改为紫色。
   - 右击 Non-STEM 并将填充颜色改为黄色。
9. 在 “图形生成器”列列表中，选择**百分比**并将其拖到 X 轴上方。请参见图 4.4。

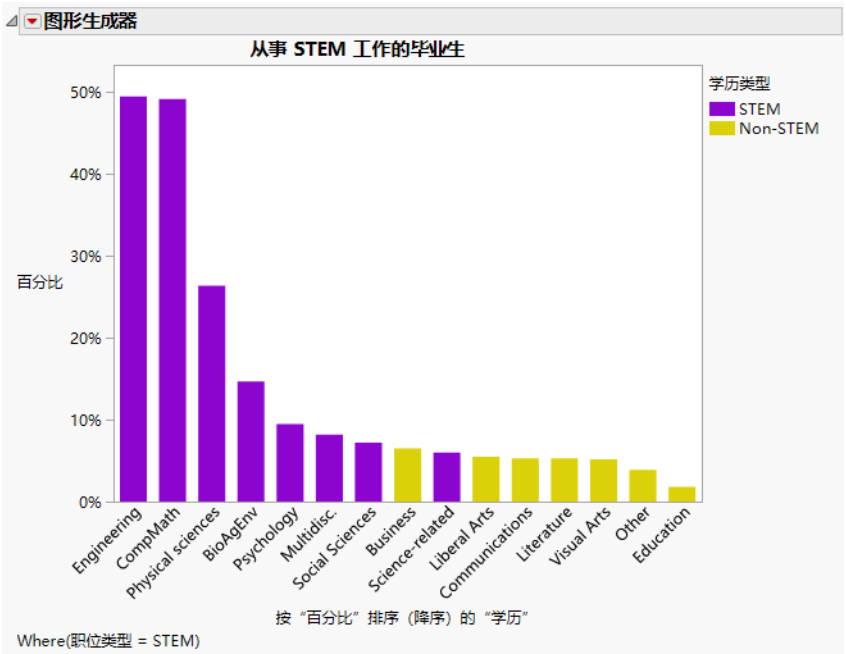
图 4.4 按百分比对学历排序



这将按百分比升序对学历排序。不过，您希望按降序排序。

10. 右击 X 轴并选择**排序依据 > 百分比，降序**。
11. 点击图形标题并键入 “从事 STEM 工作的毕业生”，然后按 Enter 键。
12. 点击**完成**。

图 4.5 完成的条形图



该图按学历类型显示从事 STEM 工作的毕业生百分比。近 50% 的工程和计算数学专业毕业生从事 STEM 工作，而教育专业毕业生不到 2%。

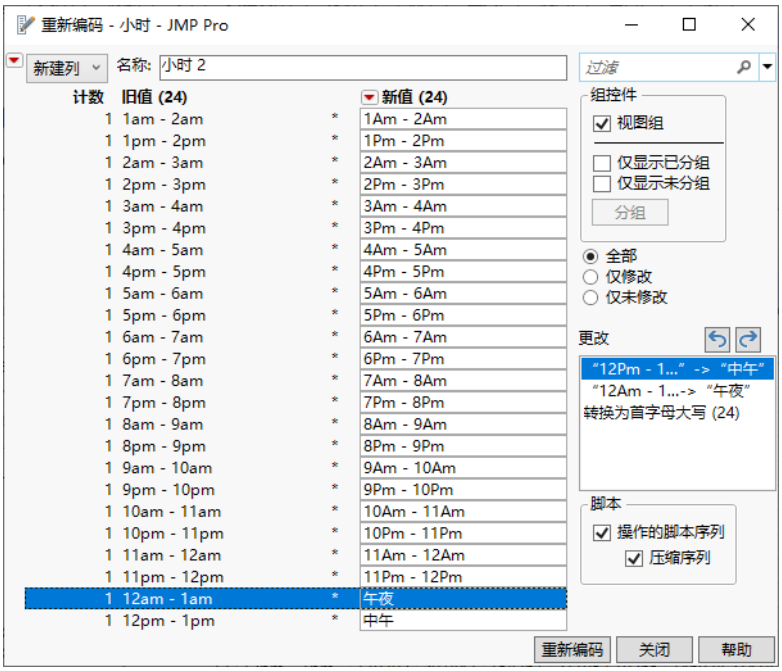
面积图和线图示例

数据	本例使用基于网站流量的数据。它包括每小时测量值和历史范围信息。
方法	本例使用“重新编码”来改进数据标签、多个图形元素（面积图和线图）以及图例定制。其中还演示：在一个轴上绘制多个变量时如何从图形元素中删除变量。
目标	本例的目标是显示整个范围内网站流量的趋势线。

1. 选择**帮助** > **样本数据库**，然后打开 Bands.jmp。
2. 选择**小时**列，然后选择**列** > **重新编码**。
3. 在“重新编码”窗口中进行以下更改：
   a. 点击红色小三角并选择**第一个单词**。
      请注意，“新值”仅包括第一个小时。

b. 在 “新值” 下，点击 12am 并输入**午夜**。点击 12pm 并输入**中午**。

图 4.6 完成的 “重新编码” 窗口



c. 点击**重新编码**。

名为**小时 2** 的新列添加到数据表中。

提示：“重新编码” 支持您将数据重新编码到新列、作为新列中的公式或覆盖当前数据。有关 “重新编码” 的详细信息，请参见 《使用 JMP》手册。

---

4. 选择**图形 > 图形生成器**。
5. 选择**小时 2** 并将其拖至 **X** 区域。

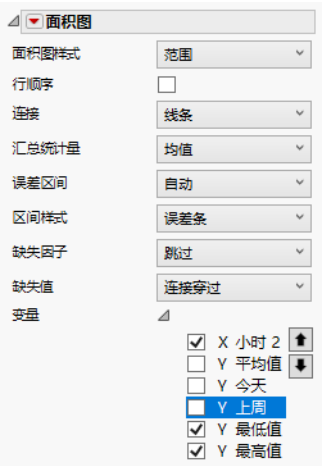
注意：X 轴的顺序符合逻辑，从午夜到晚上 11 点。这是因为 “小时” 列有一个值排序列属性，该属性在您使用 “重新编码” 创建 “小时 2” 列时已复制到该列。

---

6. 选择**平均值**、**今天**、**上周**、**最低值**和**最高值**，并将其拖至 **Y** 区域。
7. 点击 “面积图” 元素 。
8. 在 “面积图” 属性面板中，进行以下选择：
   a. 将 “面积图样式” 更改为**范围**。
   b. 打开 “变量” 并取消选择**平均值**、**今天**和**上周**。

   现在，面积图元素仅应用于**最低值**和**最高值**变量。

图 4.7 完成的 “面积图” 属性面板



9. 按住 Shift 键并点击 “线” 元素 。

提示：按住 Shift 键并点击可将其他元素添加到图形中，而不替换任何现有元素。

10. 在 “线” 属性面板中，打开 “变量” 并取消选择**最低值**和**最高值**。
现在 “线” 元素仅适用于**平均值**、**今天**和**上周** Y 变量。

图 4.8 完成的 “线” 属性面板

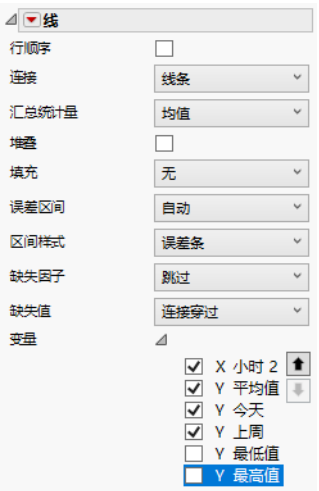
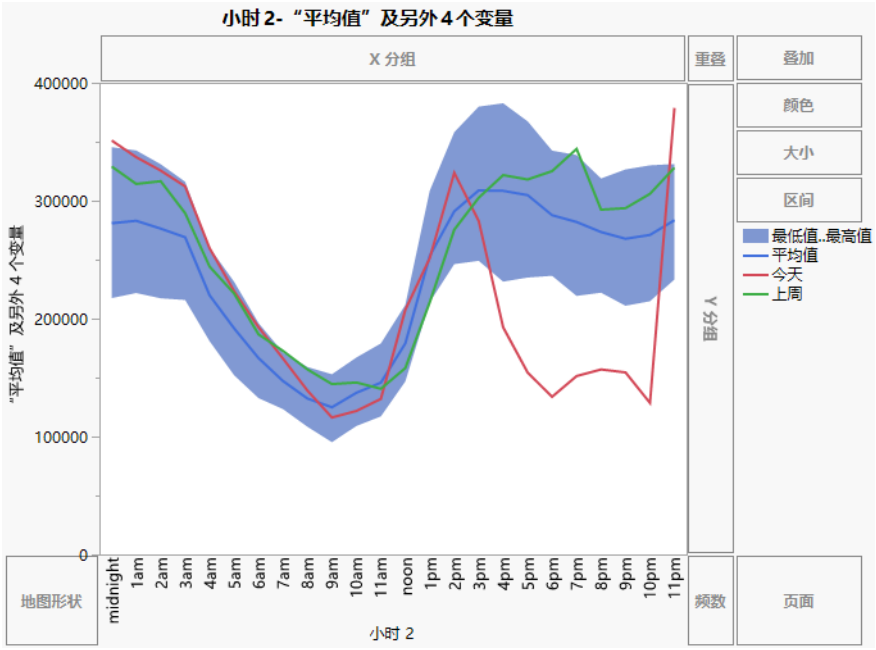
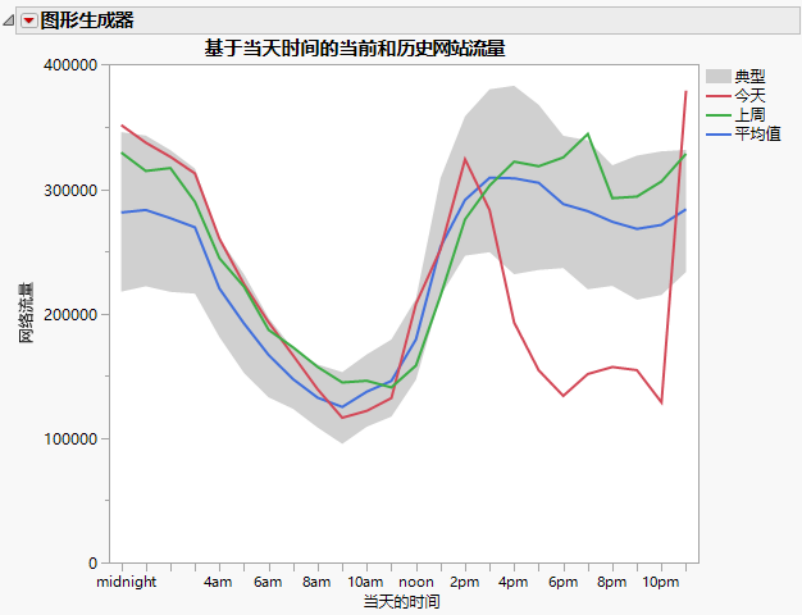


图 4.9 面积和线图



11. 使用图例定制图形元素：
    a. 在右侧的图例中，双击**最低值 .. 最高值**。
    b. 在“图例设置”窗口中，双击**最低值 .. 最高值**并将其重命名为**典型**。
    c. 突出显示**今天**和**上周**，然后点击向上箭头将其上移一个位置，以便顺序改为：**典型**、**今天**、**上周**和**平均值**。点击**确定**。
    d. 右击**典型** > **填充颜色**并选择浅灰。
12. 右击 X 轴 > **轴设置**。对于标签方向，点击**水平**，然后点击**确定**。
13. 更改轴和图形的标题：
    a. 点击 Y 轴标题并输入“网络流量”。
    b. 点击 X 轴标题并输入“当天的时间”。
    c. 点击图形标题并输入“基于当天时间的当前和历史网站流量”。
14. 点击**完成**。

图 4.10 完成的面积和线图



提示：若图例中的某项仍突出显示，请点击该项取消选定。

图形显示下午的网站流量降到了今天异常低的水平。红色趋势线表示今天的流量在下午 1 点达到峰值，然后回落到灰色带之外，灰色带表示该网站在下午 3 点左右的预期每日流量范围。

地图示例

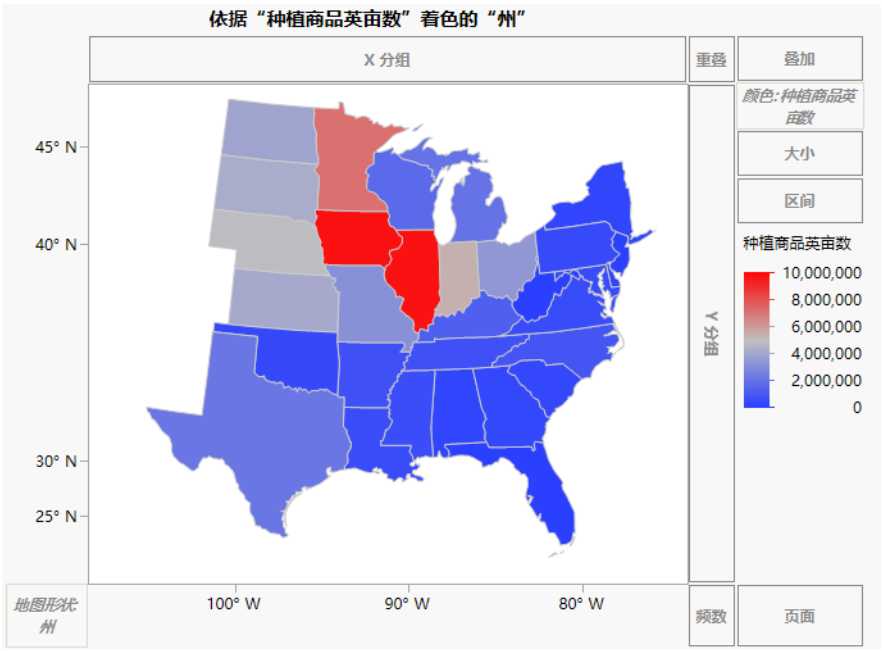
数据	本例使用 2017 年美国各州种植玉米、小麦和大豆的商品英亩数据。 **注意：**这些数据来自美国农业部 (2017)。
方法	本例使用地图形状和图例定制。
目标	本例的目标是显示每个州已种植的商品英亩数的中位数。

1. 选择**帮助** > **样本数据库**，然后打开 Corn Wheat Soybean Production.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**州**并将其拖至**地图形状**区域。

注意：JMP 将美国州名识别为地图形状。详细信息，请参见第 229 页的 “地图形状”。

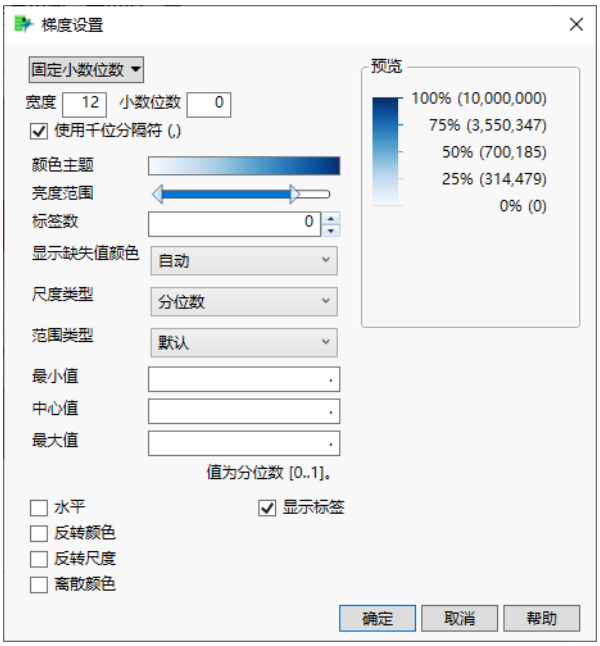
4. 选择**种植商品英亩数**并将其拖至**颜色**区域。
5. 在 “地图形状” 属性面板中，将 “汇总统计量” 改为**中位数**。

图 4.11 显示 “种植商品英亩数” 中位数的地图



6. 在图例中定制图形元素：
   a. 在图例颜色条中，右击并选择**梯度**。
   b. 在 “宽度” 旁边，输入 12。
   c. 在 “颜色主题” 旁边，点击颜色条。在 “顺序” 之下，点击第二个选项 （由白到蓝），然后点击**确定**。
   d. 从 “尺度类型” 列表中，选择**分位数**。

图 4.12 完成的 “梯度设置” 窗口



e. 点击**确定**。

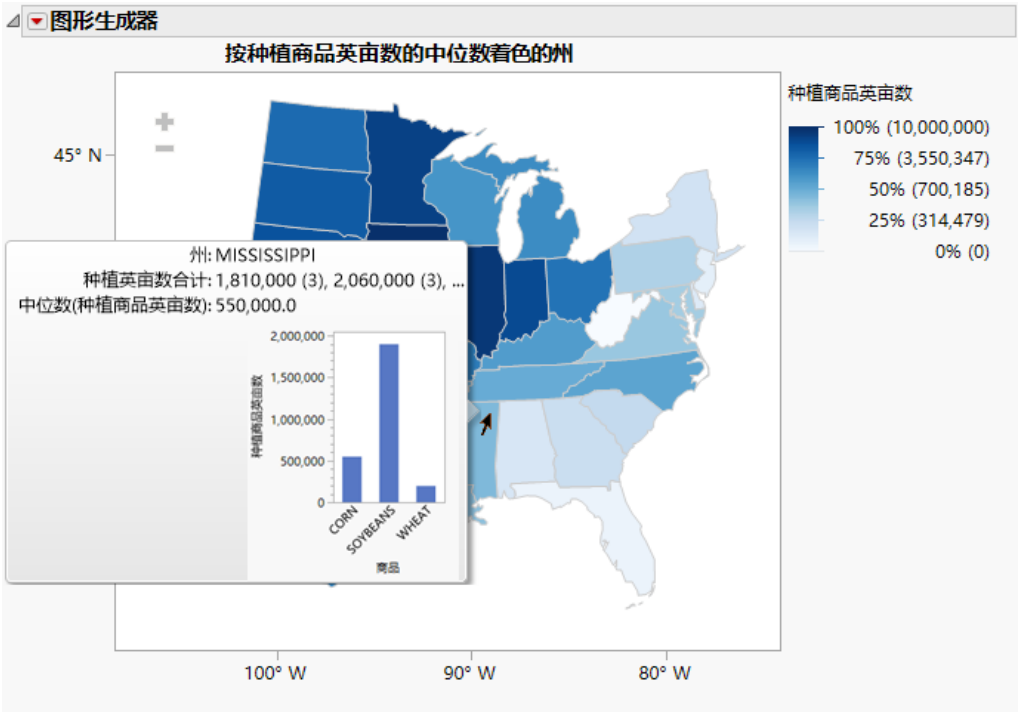
7. 点击图形上方的标题并键入**按种植商品英亩数的中位数着色的州**。
8. 在图形中右击并选择**悬停标签 > 条形图**。

   这将添加一个条形图悬停图。悬停在某个州上方时，您可以在条形图中看到该州商品的增长比例。

   注意：有关悬停标签的详细信息，请参见 《使用 JMP》手册。

9. 点击**完成**。
10. 悬停在某个州上方 （在此情况下为密西西比州）。

图 4.13 已完成的按种植商品英亩数的中位数着色的地图



地图上的颜色突出显示小麦、玉米和大豆种植英亩数最多的州。深蓝色的州被称为美国的粮仓。悬停图支持您探索在某个州种植的三种作物的比例。本例表明：在密西西比州，大豆是最主要的作物。

提示：要使用“图形生成器”查看其他示例，请运行 Corn Wheat Soybean Production.jmp 数据表中的其他脚本。

蝴蝶图示例

数据	本例使用针对 20 个问题的汇总调查回答，所有问题都使用同一个 5 级李克特量表。
方法	本例使用带有变量排序和颜色定制的条形图。
目标	本例的目标是创建回答 20 个调查问题的蝴蝶图。这样可以在一张图中提供大量调查问题的结果的汇总直观演示。

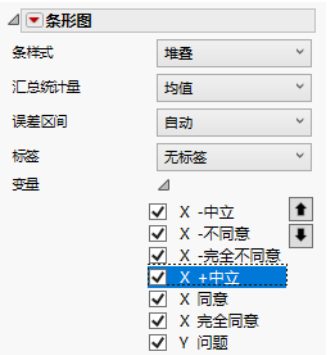
1. 选择**帮助** > **样本数据库**，然后打开 Likert Survey.jmp。

请注意关于这些数据的以下事项：

- 数据中的每个问题都有一行，而且列中包含针对每个不同回答的响应者的计数。
- **问题**列中调查问题没有按顺序排。当您在“图形生成器”中使用该列时，JMP 会按顺序排列各行。
- 公式列包含在表中，用于为否定回答创建负计数。中立计数被拆分，以便一半用于负列，一半用于正列。

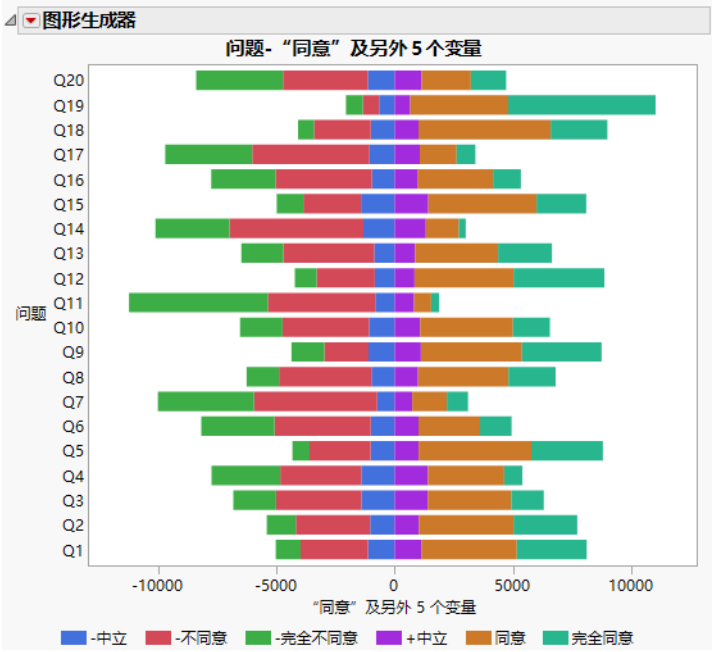
2. 选择**图形** > **图形生成器**。
3. 选择**问题**并将其拖至 **Y** 区域。
4. 选择**同意**、**完全同意**、**- 完全不同意**、**- 不同意**、**- 中立**和 **+ 中立**，并将其拖至 **X** 区域。
5. 点击“条形图”元素 。
6. 在“条形图”属性面板中，执行以下操作：
   - 将“条样式”改为**堆叠**。
   - 打开**变量**，然后突出显示某个变量，并使用箭头图标将其从上到下移动为以下顺序：**- 中立**、**-不同意**、**-完全不同意**、**+中立**、**同意**、**完全同意**。

图 4.14 完成的“条形图”属性面板



7. 点击“图形生成器”红色小三角并选择**图例位置** > **底部**。
8. 点击**完成**。

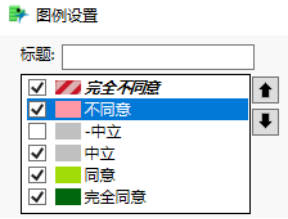
图 4.15 图例定制之前的图形



定制图例颜色和排序

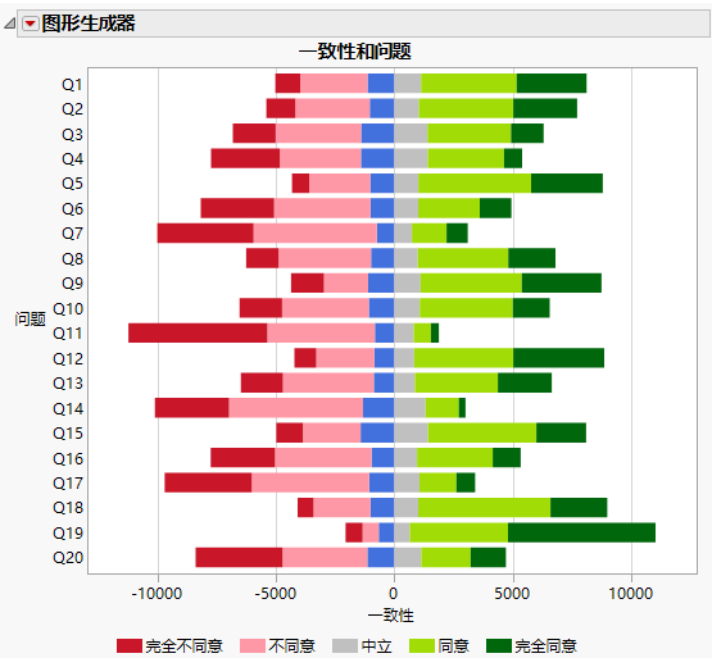
1. 右击图例中的每种颜色并更改“填充颜色”：
   - 浅灰色对应“+中立”和“-中立”
   - 浅粉色对应“-不同意”
   - 红色对应“-完全不同意”
   - 浅绿色对应“同意”
   - 深绿色对应“完全同意”
2. 悬停在图例中的任意项上方。光标应显示为手形。双击以打开 “图例设置”窗口。
3. 突出显示某个变量并使用箭头图标将顺序从上到下改为：**- 完全不同意**、**- 不同意**、**- 中立**、**+ 中立**、**同意**、**完全同意**。
4. 双击 **+ 中立**并将其重命名为**中立**。
5. 取消选中 **- 中立**旁边的框，使其不再显示在图例中，因为它是重复的。
6. 双击 **- 完全不同意**和 **- 不同意**，然后删除 “-” 符号。

图 4.16 完成的图例设置



7. 点击**确定**。
8. 更改 X 轴和图形的标题：
   a. 点击图形标题并输入 “一致性和问题”。
   b. 点击 X 轴标题并输入 “一致性”。
9. 双击 Y 轴并选择**倒序**框，然后点击**确定**。
10. 双击 X 轴并选择**主网格线**框，然后点击**确定**。
    网格线帮助您直观判断问题结果之间的差异。

图 4.17 最终李克特量表图



提示：若图例中的某项仍突出显示，请点击该项取消选定。

该图提供了 20 个调查问题的直观汇总，忽略了未回答的情况。请注意，对于任何问题，未回答的人数不到总回答人数的 1%。因此，将它们排除在直观演示之外不会影响结论。问题 19 具有最为正面的回答，问题 11 具有最为不利的回答。

散点图示例

数据	本例使用来自 442 位糖尿病患者的数据。这些数据包括基准临床和实验室数据，以及每位患者在初次访视一年后得到的糖尿病疾病进展的二元测度。该测度将疾病进展归为“低”或“高”两类。
方法	本例使用两个具有共享的 X 轴、轴定制设置和注解的散点图。
目标	本例的目标是比较和理解两种分类模型对较“高”程度疾病进展的预测概率。

运行分类模型

您需要构建一个分类模型，用来根据临床和实验室变量预测疾病进展。您将构建两个不同的分类模型、保存预测公式，并比较它们的预测分类。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 点击 **Y 二值型的决策树**脚本旁边的绿色小三角以生成决策树预测模型。
3. 点击“‘Y 二值型’分割”旁边的红色小三角，然后选择**保存列** > **保存预测公式**。您可以关闭该窗口。

   这会将概率公式保存至数据表。
4. 在数据表中，右击 Prob(Y 二值型 = = High) 列并选择**列信息**。
5. 将列名改为**分割：概率 “High”**并点击**确定**。
6. 点击 **Y 二值型的神经**脚本旁边的绿色小三角以生成神经网络预测模型。
7. 点击“模型 NTanH(3)”红色小三角并选择**保存刻画公式**。您可以关闭该窗口。

   这会将概率公式保存至数据表。
8. 在数据表中，右击**概率 (Y 二值型 =High)** 列并选择**列信息**。
9. 将列名改为**神经：概率 “High”**并点击**确定**。

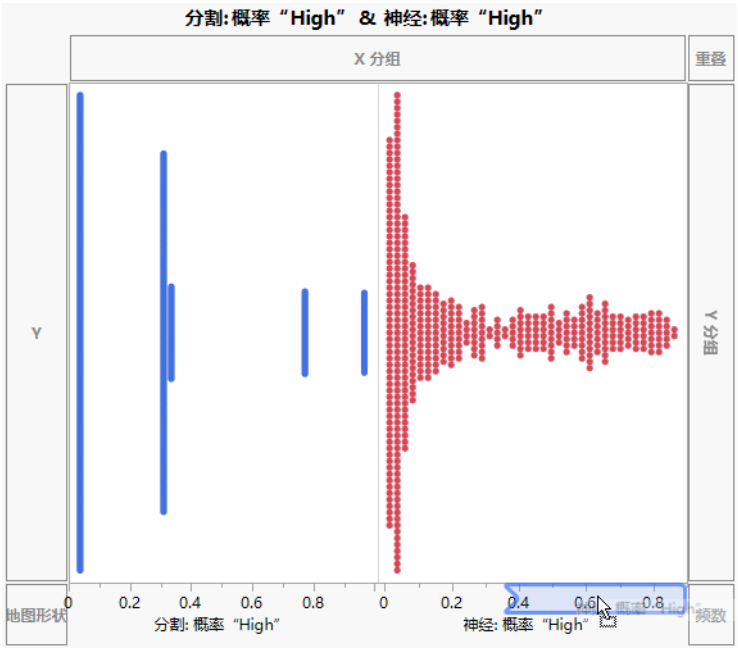
创建初始图形

您将比较模型预测的患者疾病进展程度高的概率。

1. 选择**图形** > **图形生成器**。
2. 选择**分割：概率 “High”**并将其拖至 X 区域。

3. 选择**神经：概率 “High”**，并将其拖至 X 区域中**分割：概率 “High”**的右侧。这将创建另一个 X 轴。

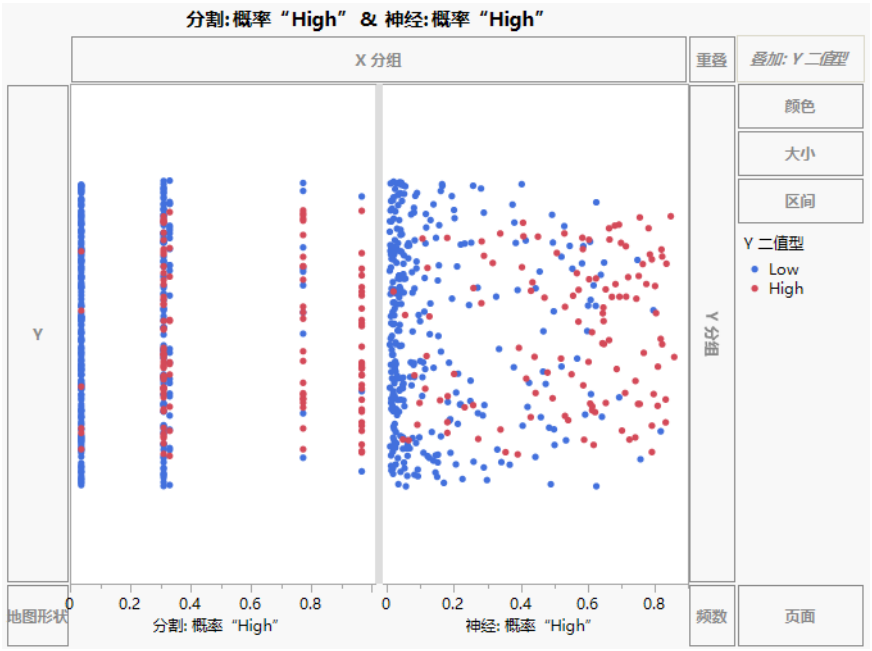
图 4.18 将 “神经：概率 ‘High’ ” 拖至 “分割：概率 ‘High’ ” 的右侧



4. 选择 Y **二值型**并将它拖至 “叠加” 区域。
5. 点击 “图形生成器” 红色小三角并选择**图形间距**。
6. 在 “图形间距” 旁边键入 6 并点击**确定**。

   这会增加两个 X 轴之间的间距。

图 4.19 初始模型概率图形

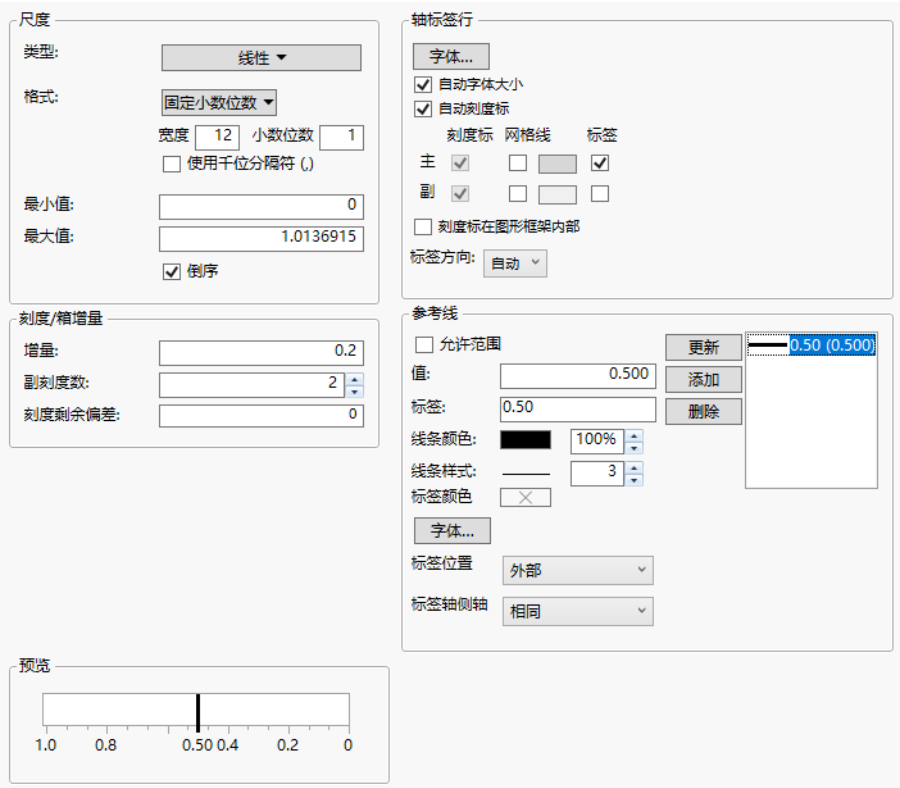


定制图形

在两个建模平台中，默认阈值为 0.50。这意味着，若患者的预测概率比“High”高 0.50，则模型将其分类预测为“High”。使用参考线在图中显示阈值。

1. 右击**分割：概率“High”** X 轴并选择**轴设置**。
2. 在“尺度”部分中，点击**倒序**旁边的框。
3. 在“刻度 / 箱增量”部分中，将**副刻度数**设置为 2。
4. 在“参考线”部分中，在**值**和**标签**旁边的框中键入 0.50。
5. 在“线条样式”旁边的框中，输入 3。3 指示参考线的粗细。
6. 点击**添加**。

图 4.20 X 轴设置

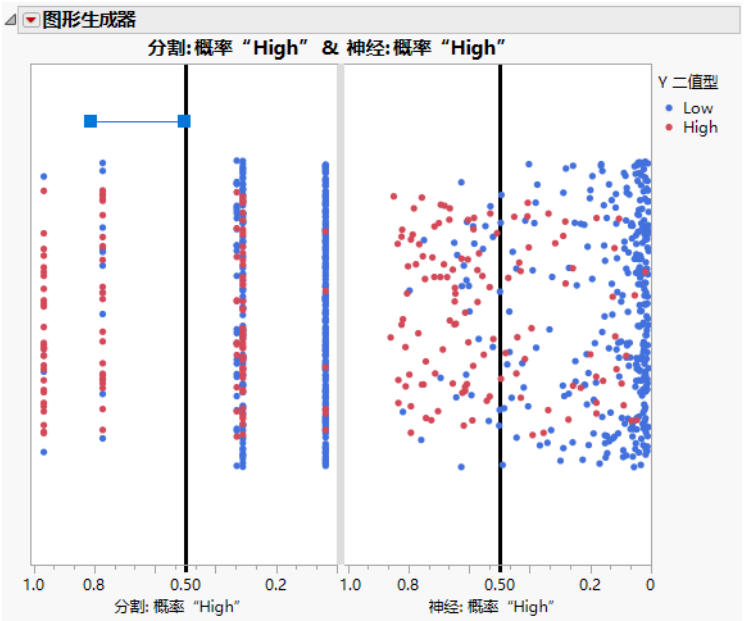


7. 点击**确定**。
8. 右击“分割：概率 ‘High’”X 轴并选择**编辑 > 复制轴设置**。右击“神经：概率 ‘High’”X 轴并选择**编辑 > 粘贴轴设置**。
9. 点击**完成**。

为图形添加注解

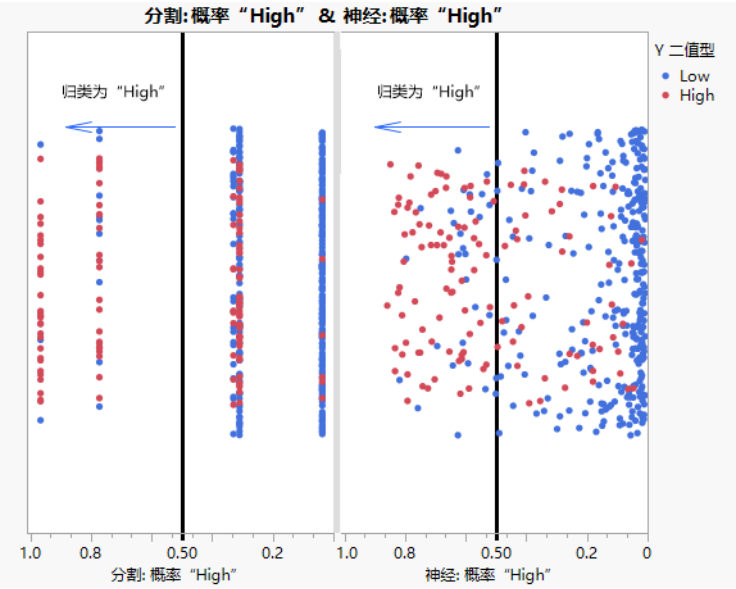
1. 在主菜单中，点击**工具 > 线条**。在“分割：概率 ‘High’”图形中的参考线左侧绘制一条水平线。

图 4.21 绘制在 “分割：概率 ‘High’ ” 参考线左侧的水平线



2. 右击这条线并选择**箭头指向**。
   箭头应指向远离参考线的方向。
3. 在主菜单中，点击**工具 > 注解**，然后在图形中的箭头上方点击。
4. 在文本框中键入**归类为 “High”**。请参见图 4.22。
5. 在 “神经：概率 ‘High’ ” 图形中重复第 1 步到第 4 步。

图 4.22 预测概率的分布



图中显示两个模型的预测概率分布不同。分割模型有五个预测得分水平，而神经模型得分分散在整个得分范围内。对于这两个模型，在每个图的最右侧都聚集了一组蓝色数据点。这是两个模型都能正确分类的较低程度测试对象，因为它们落在阈值的右侧。

堆叠填充的条形图示例

数据	本例使用美国国家海洋和大气管理局提供的 1980 年至 2018 年间美国天气和气候事件对经济产生的影响的相关数据。
方法	本例使用一个包含 10 个主要类别的堆叠填充的条形图。
目标	本例的目标是创建一个堆叠填充的条形图，用来显示对经济影响最大的事件。

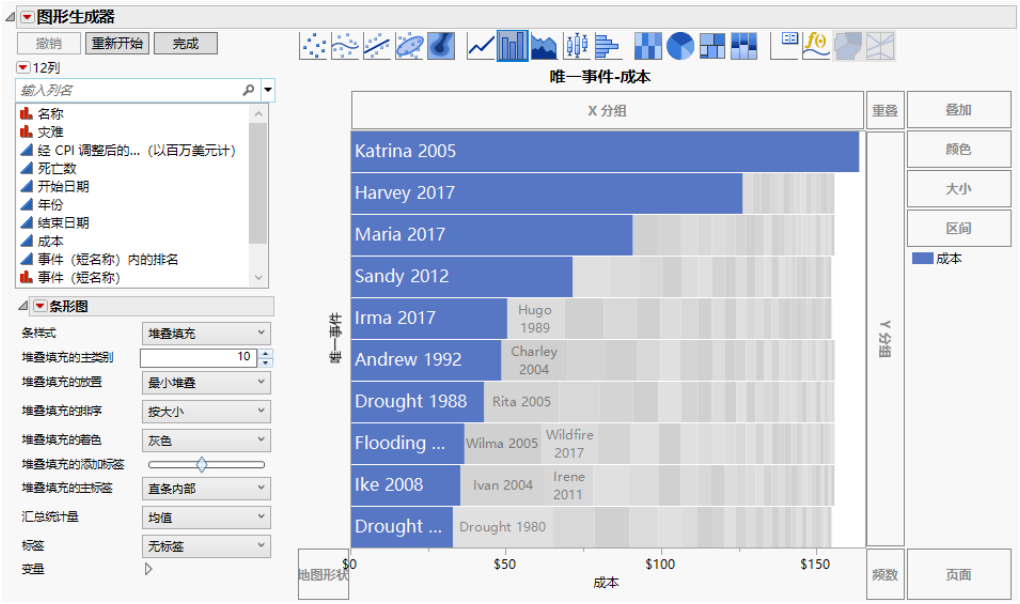
1. 选择**帮助** > **样本数据库**，然后打开 Billion Dollar Events.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**唯一事件**并将其拖至 Y 区域。
4. 选择**成本**并将其拖至 X 区域。
5. 点击 “条形图” 元素 。
6. 在 “条形图” 选项面板中：

a. 对于 “条样式”，选择**堆叠填充**。

b. 将 “堆叠填充的主类别”改为 10。

c. 将 “堆叠填充的添加标签”滑块向左移动，直到大约移动一半。

图 4.23 在堆叠填充的条形图中具有巨大经济影响的天气事件



前 10 个类别显示为带有蓝色直条的条形图。这些是您在选项中设置为 10 的 “堆叠填充的主类别”。次级类别用灰色标记。您可以悬停在任何直条上方以获取更多详细信息。

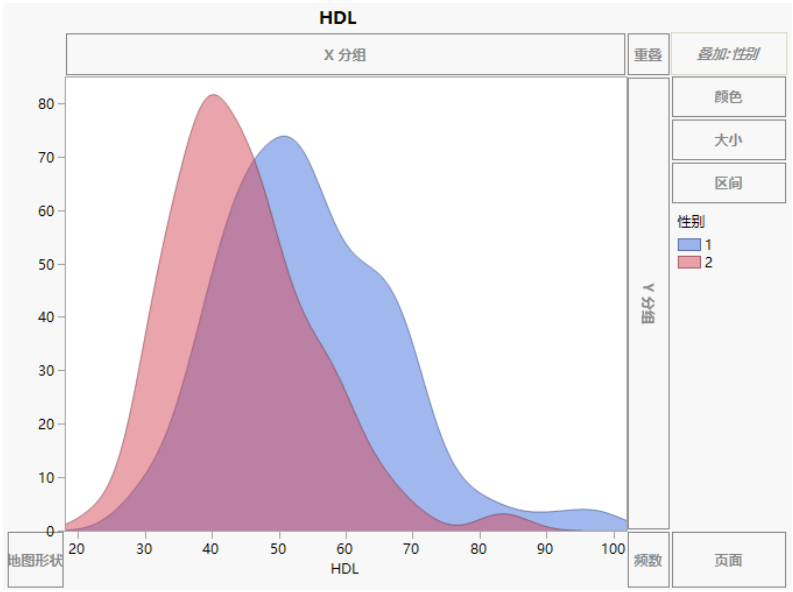
在该图中，您可以清楚地看到：2005 年的卡特里娜 (Katrina) 飓风在 1980 年至 2018 年间袭击美国的所有风暴中，对金融的影响最大。2017 年的飓风 “哈维”(Harvey) 和 “玛丽亚”(Maria) 也造成了巨大的金融影响。

叠加直方图和脊线图的示例

数据	本例使用来自 442 位糖尿病患者的数据。这些数据包括基准临床和实验室数据，以及每位患者在初次访视一年后得到的糖尿病疾病进展的二元测度。该测度将疾病进展归为 “低”或 “高”两类。
方法	本例使用叠加直方图和脊线图。
目标	本例的目标是探索按类别分组的连续变量的分布。具体来说，您的关注点是探索 HDL 的分布如何先随性别后随疾病进展而变化。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择 HDL 并将其拖至 X 区域。
4. 选择**性别**并将其拖至**叠加**区域。
5. 点击**直方图**元素图标 。
6. 在 “直方图” 属性面板中，从 “直方图样式” 列表中选择**核密度**。
7. 悬停在靠近尺度顶部的 Y 轴上方，直到光标变成水平手形。点击并拖动，直到最大值约为 85。

图 4.24 按性别划分的叠加 HDL 直方图

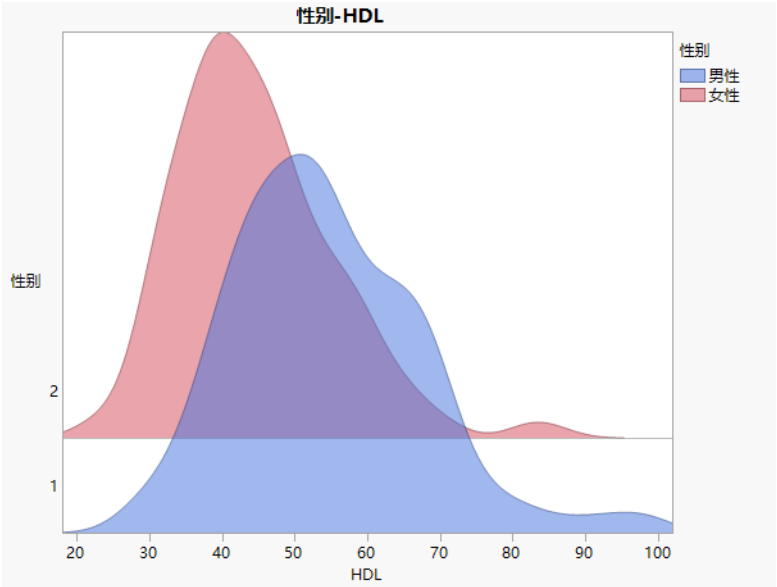


直方图显示男性 （蓝色）的 HDL 水平高于女性 （粉色）。蓝色分布落在右侧，或在 HDL 尺度上高于粉色分布。现在更改图形以查看脊线图中的性别差异。

8. 选择**性别**并将其拖至 Y 区域。
9. 在“直方图”属性面板中，移动“叠加”滑块以叠加直方图（直到滑块滑动到大约 3/4 以上）。
10. 点击**完成**。
11. 在图例中双击。在 “图例设置” 窗口中，双击 1 并键入 “男性”。双击 2 并键入 “女性”。
12. 点击**确定**。

提示：若图例中的某项仍突出显示，请点击该项取消选定。

图 4.25 按性别和糖尿病水平划分的 HDL 脊线图



脊线图显示了性别分布剩余偏差。

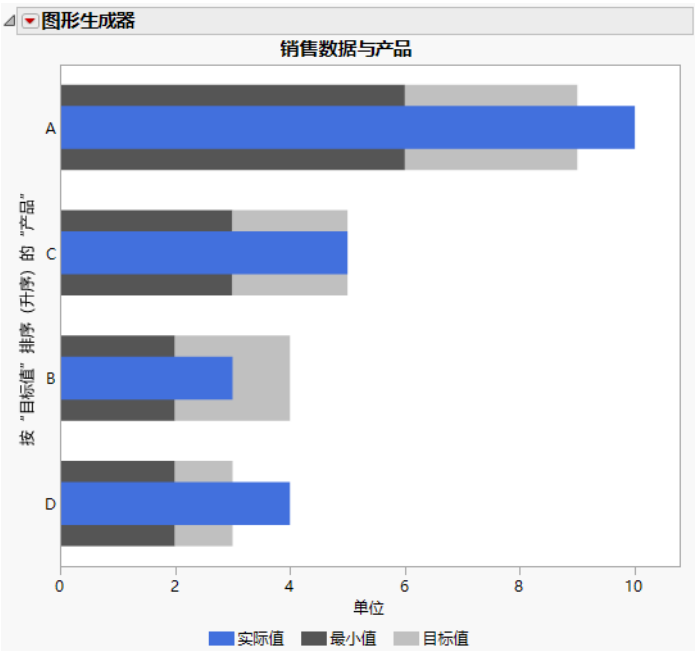
弹道条形图示例

数据	本例使用四种产品的虚构销售数据。
方法	本例使用弹道图。
目标	本例的目标是显示实际售出的单位数与最小和目标单位数的比较。

1. 选择**帮助** > **样本数据库**，然后打开 Bullet Measures.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**实际值**、**目标值**和**最小值**并将其拖至 X 区域。
4. 选择**产品**并将其拖至 Y 区域。
5. 点击**条形图**元素图标 。
6. 在 “条形图”属性面板中，执行以下操作：
   - 从“条样式”列表中，选择**弹道图**。
   - 打开**变量**，然后突出显示**最小值**并点击一次向上箭头。
7. 点击 “图形生成器”红色小三角并选择**图例位置** > **底部**。

8. 右击图例中的某种颜色并更改 “填充颜色”：
   - 不更改**实际值**
   - 将**最小值**改为深灰色
   - 将**目标值**改为浅灰色
9. 右击 Y 轴 > **排序方式 > 目标值，升序。**
10. 更改 X 轴和图形的标题：
    a. 点击图形标题并输入 “销售数据与产品”。
    b. 点击 X 轴标题并输入 “单位”。
11. 点击**完成**。

图 4.26 售出的实际、目标和最小单位数的弹道图



产品按目标单位数排序，如浅灰色直条所示。产品 A 和 D 的实际销售量高于其目标；蓝色直条超出了浅灰色目标直条。产品 C 的销售量与目标吻合。产品 B 的销售量低于目标但高于最小单位数。

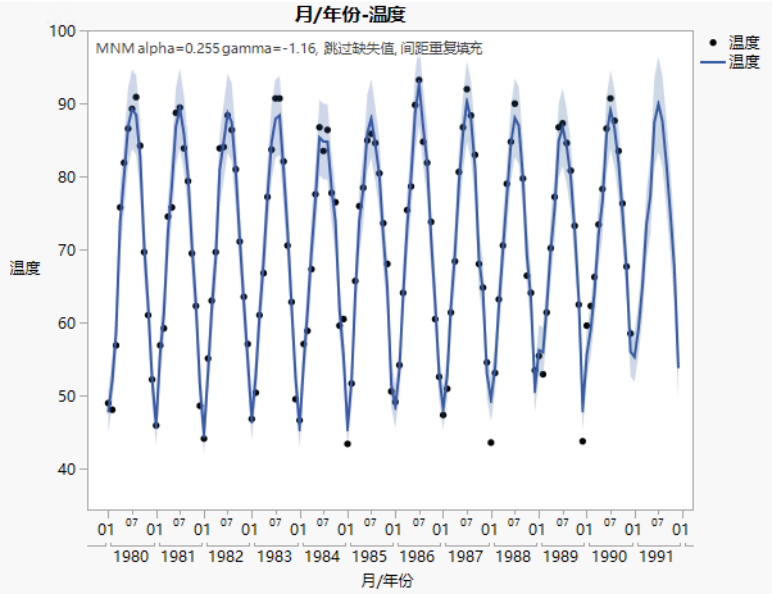
预测图示例

数据	本例使用美国北卡罗来纳州罗利市 1980 年至 1990 年间的最高每月气温数据。
方法	本例使用时间序列拟合线。
目标	本例的目标是预测下一年的月最高气温。

1. 选择**帮助** > **样本数据库**，然后打开 Time Series/RaleighTemps.jmp。
2. 选择**图形** > **图形生成器**。
3. 选择**月 / 年份**并将其拖至 X 区域。
4. 选择**温度**并将其拖至 Y 区域。
5. 点击**拟合线**元素图标 。
6. 在 “拟合线”属性面板中，执行以下操作：
   - 从“拟合”列表中，选择**时间序列**。
   - 在“季节性期间”旁边的框中，键入 12。
   - 在“预测周期数”旁边的框中键入 12。

     这会对未来 12 个月进行预测。由于数据在 1990 年结束，所以将预测到 1991 年。
7. 右击 X 轴并选择**轴设置**。
   - 在“刻度/箱增量”下，将“增量”改为 1。
   - 在“刻度/箱增量”下，将“标签行嵌套”改为 2。
   - 在“标签行 1”选项卡的“多个轴标签行”下，选择副刻度标的**标签**。
   - 点击**确定**。
8. （可选）要将预测值和预测间隔的上下限值点保存为数据表中的新列，请点击 “拟合线”红色小三角并选择**保存公式**。
9. 点击**完成**。

图 4.27 最高每月气温预测图



该图显示用于估计贯穿整个预测期间的时间序列函数的数据点。需要注意关于该图形的以下事项：

- 蓝色实线表示预测模型。着蓝色的区域是围绕预测值的 95% 置信区间。
- 预测期间覆盖 1991 年，即经过最后一个数据点之后的 12 个月。预测期间是图中没有数据点的最右侧部分。
- 预测 1991 年气温峰值出现在 7 月，估计值为 90°F。该估计值的置信区间约为 83°F 到 96°F，该区间由估计值上方和下方的蓝色区域的上下限表示。
- 顶部的文本指示所拟合模型的类型、该模型的参数值以及有关缺失值的信息。要从图形中删除该文本，请取消选择“拟合线”属性面板中的“预测模型”选项。

第 5 章

气泡图

使用气泡图查看多维数据中的模式

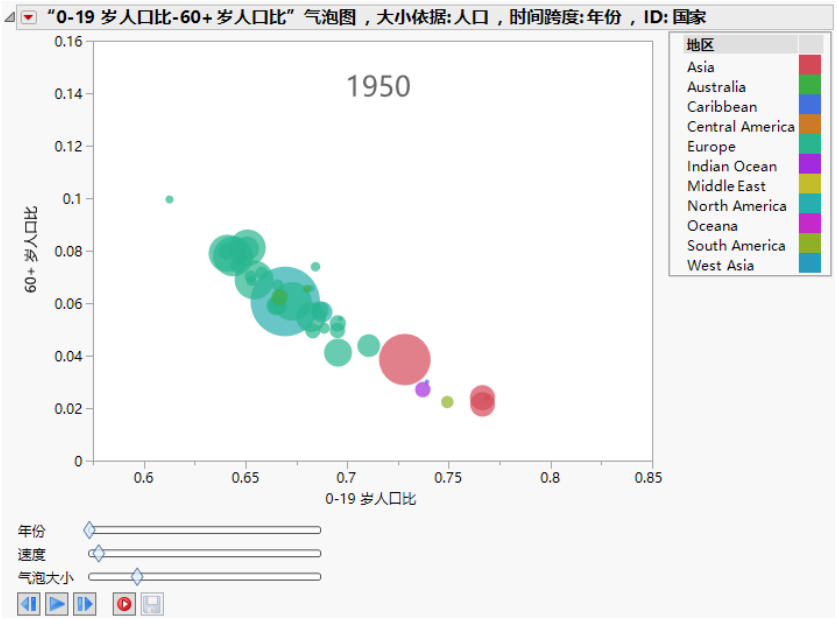
气泡图是一个将点表示为圆圈（或气泡）的散点图。气泡图可以是动态的（随时间变化播放动画），也可以是静态的（不能移动的固定气泡）。使用气泡图可以：

- 使用时间变量动态播放气泡动画，以查看气泡随时间的移动及其模式
- 使用大小和颜色明确区分不同的变量
- 将数据（行）聚合为单个气泡，简化气泡图

由于您可以最多同时查看五个维（x 位置、y 位置、大小、颜色和时间），所以气泡图可以产生动画视觉效果，使模式和趋势显而易见。

注意：最早开始动态气泡图研究的是卡罗琳学院国际卫生学教授 Hans Rosling 以及 Gapminder.org 项目的参与人员。

图 5.1 气泡图示例



目录

动态气泡图的示例 133
启动“气泡图”平台 135
指定两个 ID 变量 136
指定“时间”变量 136
与气泡图交互 136
控制动态气泡图的动画播放 136
选择气泡 138
使用笔刷工具 138
“气泡图”平台选项 138
显示角色 140
“气泡图”平台的更多示例 142
仅指定一个“时间”变量的示例 142
仅指定 ID 变量并且拆分气泡的示例 144
静态气泡图的示例 146
包含分类 Y 变量的气泡图示例 150

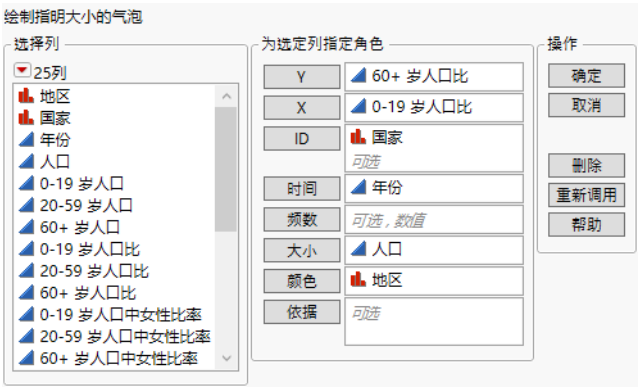
动态气泡图的示例

该示例使用 PopAgeGroup.jmp 样本数据表，其中包含世界各国和地区的人口数据。检查年轻人与老人在样本人口中所占比例之间的关系。

1. 选择**帮助** > **样本数据库**，然后打开 PopAgeGroup.jmp。
2. 选择**图形** > **气泡图**。

   随即显示启动窗口。

图 5.2 “气泡图”启动窗口



3. 选择 **60+ 岁人口比**并点击 **Y**。

   60 岁以上人口比例变为 y 坐标。
4. 选择 **0-19 岁人口比**并点击 **X**。

   0-19 岁的人口比例变为 x 坐标。
5. 选择**国家**并点击 **ID**。

   与各国对应的所有行都聚合到一个气泡中。
6. 选择**年份**并点击**时间**。

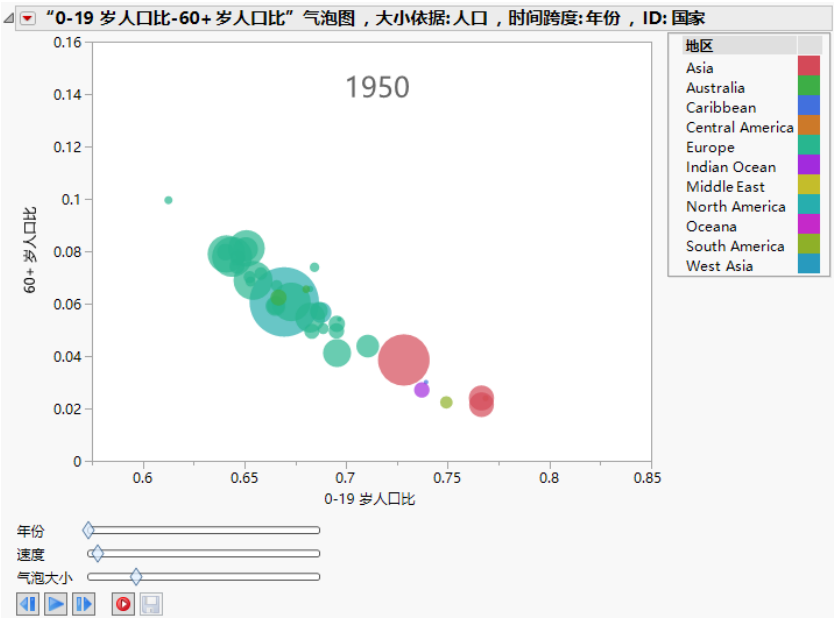
   气泡图将为每年的数据单独显示一个图。
7. 选择**人口**并点击**大小**。

   气泡大小反映总人口值。
8. 选择**地区**并点击**颜色**。

   给不同地区的气泡指定不同颜色。请参见图 5.3 中的**地区**图例。图中显示的颜色为 JMP 默认颜色。
9. 点击**确定**。

   随即显示报表窗口。

图 5.3 “气泡图”报表窗口



10. 点击播放按钮可查看制成动画的动态报表。或者，也可以点击上一个按钮一次向前移动一年。
11. （可选）要查看为图中每个地区显示一种颜色的图例，请点击“气泡图”红色小三角并选择**图例**。

随着时间的推移，您可以看到 0-19 岁的人口比例在降低，而 60 岁以上人口的比例在增高。

启动 “气泡图” 平台

通过选择**图形** > **气泡图**可启动 “气泡图” 平台。

图 5.4 “气泡图” 启动窗口

Y，X Y 列和 X 列分别变为图中气泡的 *y* 坐标和 *x* 坐标。这些值既可以是连续值，也可以是分类值 （名义型或有序型）。

ID ID 变量标识应聚合并显示为单个气泡的行。每个气泡的默认坐标都是 *x* 和 *y* 的平均值，每个气泡的默认大小是所有聚合成员的大小之和。请参见第 136 页的 “指定两个 ID 变量”。

时间 为每个唯一的时间段保持单独的坐标、大小和颜色。气泡图为单个时间段显示这些值。例如，若**时间**列包含年，气泡图将更新以显示各年数据。请参见第 136 页的 “指定 “时间” 变量”。

频数 使用 ID 变量聚合气泡时计算权重。

注意：忽略负频数值。

大小 控制气泡的大小。气泡面积与**大小**值成比例。设定了最小气泡大小，这样即便大小值为零，气泡仍然可见。若将**大小**留空，默认气泡大小将与**时间**和 ID 组合中包含的行数成比例。

颜色 根据选定变量对气泡着色。若选定变量为分类变量 （名义型或有序型），则为每个类别指定不同颜色。若选定变量为连续变量，则使用颜色梯度。您可以通过选择 “文件” > “首选项” > “图形” 并在 “颜色主题” 面板中进行选择来设置颜色主题的首选项。

依据 将某列置于此处可为变量的每个水平生成单独的气泡图。

点击**确定**后，将显示“气泡图”报表窗口。

指定两个 ID 变量

指定另一个 ID 变量会提供类别层次，但气泡不会被第二个类别拆分，除非选定这些气泡并以交互方式拆分它们。在启动窗口中，若指定另一个 ID 变量，**拆分**和**合并**按钮将显示在报表窗口中。

例如，您可能将国家指定为第一个 ID 变量，这将为每个国家生成单独的聚合气泡。另一个 ID 变量（可能指定各国内的不同地区）将在按下图形下方的交互式**拆分**按钮后进一步拆分各国。

指定“时间”变量

为每个唯一的时间段保持单独的坐标、大小和颜色。气泡图为单个时间段显示这些值。例如，若**时间**列包含年，气泡图将更新以显示各年数据。

要移动图上的时间标签，可以点击并拖动该标签。

若某时间段内的数据缺失，将线性插入该值。若缺少第一个或最后一个时间段的数据，则不会估计该值，而是保持缺失。

相关信息

- 第 136 页的“控制动态气泡图的动画播放”
- 第 142 页的“仅指定一个“时间”变量的示例”

与气泡图交互

注意：若隐藏且不排除构造气泡图形状所用的**所有**行，则在气泡图中不显示相应的气泡图形状。若在数据表中排除行，则在排除这些行的情况下构造气泡图。

采用以下两种模式之一使用“气泡图”平台：

- **静态模式**，该模式气泡是固定的，不会随时间变化播放动画（未指定**时间**变量）。请参见第 146 页的“静态气泡图的示例”。
- **动态模式**，该模式随时间变化播放气泡动画（指定了**时间**变量）。请参见第 133 页的“动态气泡图的示例”。

您可以通过不同的方式与静态气泡图和动态气泡图交互。

控制动态气泡图的动画播放

使用滑块和按钮可控制动态气泡图的动画播放。

图 5.5 动画控制

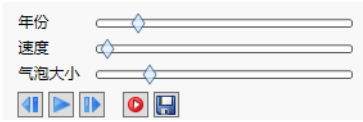


表 5.1 动画控制的说明

滑块或按钮	说明
<“时间”变量>	控制显示在气泡图中的时间值。您需要手动拖动滑块来查看时间的推移。点击并拖动气泡图中的时间变量可移动其位置。 仅当您已指定**时间**变量时才显示。
速度	调整动画播放的速度。 仅当您已指定**时间**变量时才显示。
气泡大小	调整气泡的大小。气泡保持其相对大小，但其绝对大小可以调整。 显示在所有气泡图上。
	调整一个单位的时间值并显示上一个时间值。 仅当您已指定**时间**变量时才显示。
	按下播放可播放气泡图动画。在所有时间值中按顺序移动，并在达到最后一个时间段时返回到开头。按下暂停可停止动画播放。 仅当您已指定**时间**变量时才显示。
	调整一个单位的时间值并显示下一个时间值。 仅当您已指定**时间**变量时才显示。
	记录动画（仅限 Windows）。
	将动画另存为动画 GIF 文件（仅限 Windows）。
拆分	将范围较大的第一个 ID 变量所表示的气泡拆分为范围较小的第二个 ID 列所定义的较小组成部分。选择气泡并点击**拆分**。 仅当您已指定两个 ID 变量时才显示。

表 5.1 动画控制的说明 (续)

滑块或按钮	说明
合并	通过将较小气泡重新合并为其原始气泡，反转**拆分**按钮的操作。选择分组中的任意较小气泡并点击**合并**。 仅当您已指定两个 ID 变量时才显示。

选择气泡

点击某个气泡可将其选定。请注意以下几点：

- 直观来看，选定的气泡变得更暗或更亮，未选定的气泡则更为透明。
- 若最初未填充气泡，选择气泡会将其填充。
- 若未选定任何气泡，所有气泡都是半透明的。

选择某个气泡时，数据表中与选定气泡对应的所有行都突出显示。请注意以下几点：

- 若气泡是基于 ID 列的聚合，则该 ID 对应的所有行都突出显示。否则，则只有该气泡所表示的一行突出显示。
- 若指定了 ID 变量和**时间**变量，选择某个气泡将突出显示该 ID 跨越所有**时间**水平所对应的所有行。

若从数据表中选择某一行，在相关的气泡图中也会将其选定。请注意以下几点：

- 若尚未指定**时间**变量，选择数据表中的某一行将突出显示图中相应的气泡。
- 若已指定**时间**变量，选择数据表中的某一行将突出显示动态气泡图中仅与该时间段对应的气泡。

使用笔刷工具

使用笔刷工具 可以暂时选定气泡并获取有关选定气泡的详细信息。使用笔刷工具选择气泡时，对应的行在相关数据表中会突出显示。

注意：要对突出显示的行进行更仔细地检查，可以使用**表 > 子集**命令或“行编辑器”。请参见《使用 JMP》。

“气泡图”平台选项

“气泡图”红色小三角菜单提供以下选项：

绘制 应用填充或轮廓。

填充 填充所有气泡。

轮廓 勾画所有气泡的轮廓。

填充且带轮廓 填充所有气泡并勾画其轮廓。

设置形状 更改气泡的形状。

您可以使用 JSL 创建定制形状。使用**定制**选项可打开定制形状。若未创建任何定制形状，**定制**选项将使用默认的圆圈形状。有关创建定制形状的详细信息，请参见《Scripting Guide》。

形状定向 当形状随时间朝特定方向移动以跟随数据形状时，为这些形状定向。

仅当您已指定**时间**变量时才显示该选项。

尾迹气泡 将气泡的过往历史显示为半透明的尾迹。请参见第 142 页的 “仅指定一个 “时间” 变量的示例”。

请注意以下事项：

– 仅当您已指定**时间**变量时才显示该选项。

– 若不想看到气泡标签，可以选择**标签** > **无**选项。

尾迹线 将气泡的过往历史显示为相连的线段。请参见第 142 页的 “仅指定一个 “时间” 变量的示例”。

请注意以下几点：

– 仅当您已指定**时间**变量时才显示该选项。

– 若不想看到气泡标签，可以选择**标签** > **无**选项。

标签 更改图上的标签。

无 不为图中任何气泡添加标签。

全部 为图中所有气泡添加标签。

选定 仅在选定气泡时为其添加标签。

注意：点击并拖动某个标签可移动它。

颜色主题 更改表示颜色变量的高值、中值和低值的颜色。

仅当您已指定**颜色**变量时才显示该选项。

恢复颜色主题 恢复为初始颜色主题。

仅当您应用了颜色主题之后才显示该选项。

图例 显示描述气泡图中的颜色的图例。

仅当您已指定**颜色**变量时才显示该选项。

跨数据缺失期间选择 若选定某个气泡，该选项可使该气泡在缺失数据的时间段内保持选定状态。若未选定，则不会在缺失数据的时间段内选定该气泡。

显示角色 显示在气泡图中使用的变量。您可以更改和删除变量。请参见第 140 页的 “显示角色”。

全部拆分 将所有气泡拆分为其组成部分。不用像**拆分**按钮那样先选定这些气泡。

仅当您已指定两个 ID 变量时才显示该选项。

全部合并 将分组中的所有组成气泡合并为较大的气泡。不用像**合并**按钮那样先选定这些气泡。

仅当您已指定两个 ID 变量时才显示该选项。

显示时间注解 在包含时间变量的气泡图中，“显示时间注解”选项显示或隐藏显示在气泡图中的时间。

锁定尺度 防止轴尺度和梯度图例尺度因数据变化或过滤更改而自动调整。

适应窗口大小 确定是否在您调整 JMP 窗口大小时调整图的大小。默认设置为 “自动”，基于图内容调整尺度。例如，带“依据”变量的图不会拉伸以适应调整大小后的窗口；图超出查看区域的大小。将设置更改为 “开”可始终使图适应窗口大小。将设置更改为 “关”可阻止图调整大小。

聚合选项 改变 X、Y 和**大小**角色的计算方式。默认情况下，使用 X 和 Y 的均值以及**大小**的总和来计算这些值。

表示求和的 X 或表示求和的 Y 使用总和计算 X 和 Y 值。

表示求和的大小 撤销选择该选项将使用均值计算**大小**值。

表示求和的颜色 计算数据值的求和并映射至某一颜色。该选项仅对连续变量显示。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

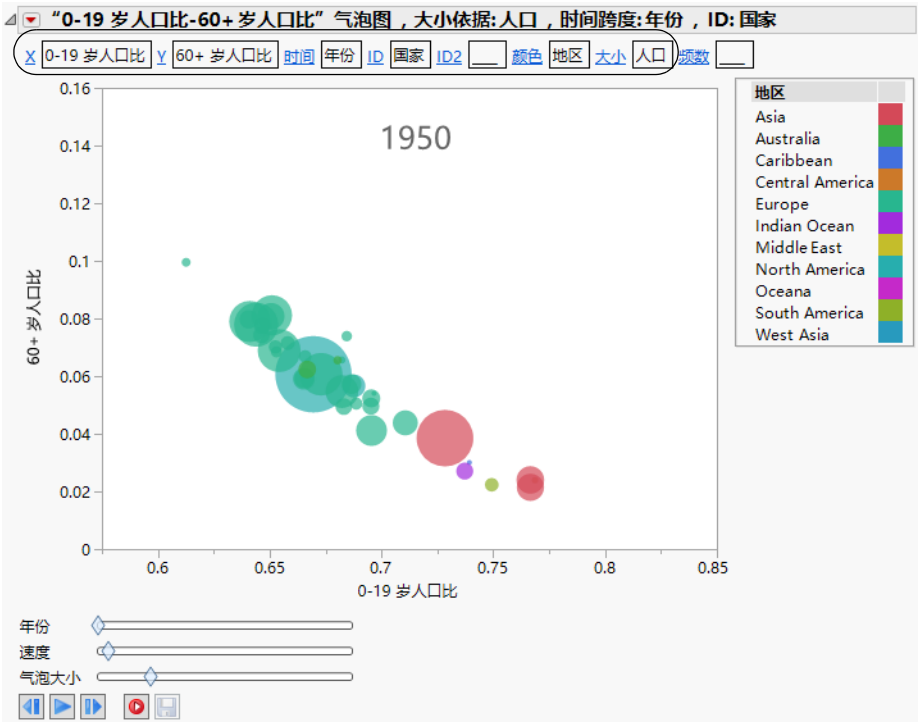
保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

显示角色

使用红色小三角菜单中的**显示角色**选项，您可以对现有变量进行更改，而不必重新启动该平台，再重新开始分析。

请按第 133 页的“动态气泡图的示例”中的说明来生成图 5.6 中所示的报表窗口。（这些颜色为 JMP 默认颜色。）

图 5.6 选定了 “显示角色”的气泡图的示例



更改指定给角色的变量

要更改指定给某个角色的变量，请点击带蓝色下划线的角色名称。例如，执行以下步骤以更改图 5.6 中的**颜色**变量：

1. 点击**颜色**链接。

   出现 “为 ‘颜色’选择列”窗口。
2. 点击**国家**。
3. 点击**确定**。

   国家现已取代**地区**成为气泡图中的**颜色**变量。

删除变量

要从气泡图中删除现有变量，请确保在 “为 < **角色** > 选择列”窗口中未选定任何列，然后点击**确定**。例如，执行以下步骤以删除图 5.6 中的**大小**变量 （人口）：

1. 点击“大小”链接。

   出现 “为 ‘大小’选择列”窗口。
2. 确保未选定任何列。若选定某个变量，通过按住 Ctrl 键并点击该变量取消对它的选定。

3. 点击**确定**。

“大小”角色现在显示为一个空框。

注意：X 和 Y 变量只能更改不能删除。

添加变量

一旦从气泡图中删除了某个现有变量，可以采用两种方法来添加新变量：

- 点击带蓝色下划线的角色名称。请参见第 141 页的“更改指定给角色的变量”。
- 在数据表中，点击列面板中的变量，然后将其拖入空角色框。

“气泡图”平台的更多示例

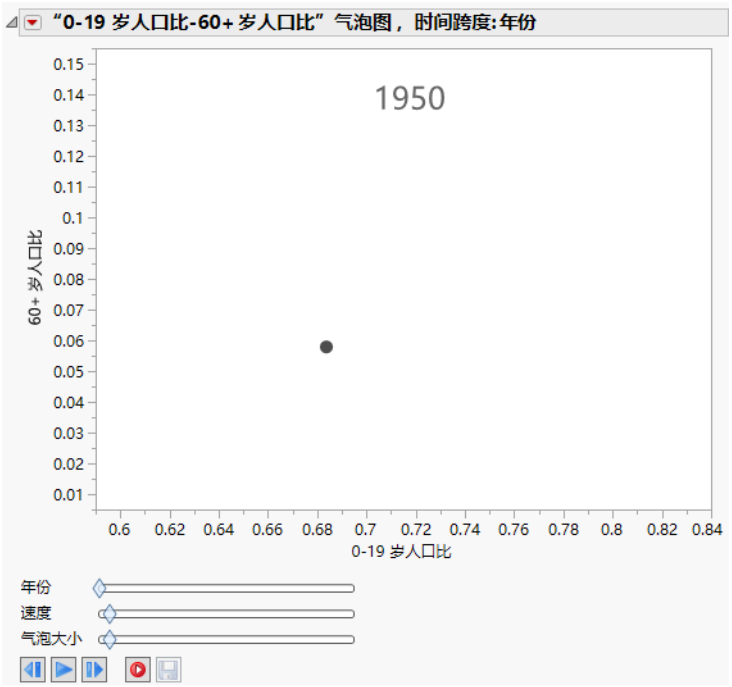
- “仅指定一个“时间”变量的示例”
- “仅指定 ID 变量并且拆分气泡的示例”
- “静态气泡图的示例”
- “包含分类 Y 变量的气泡图示例”

仅指定一个“时间”变量的示例

对于动态气泡图，您可以仅指定一个**时间**变量而不指定任何 ID 变量。生成的气泡图包含一个移动气泡，该气泡可随着**时间**值的更改跟踪数据序列。

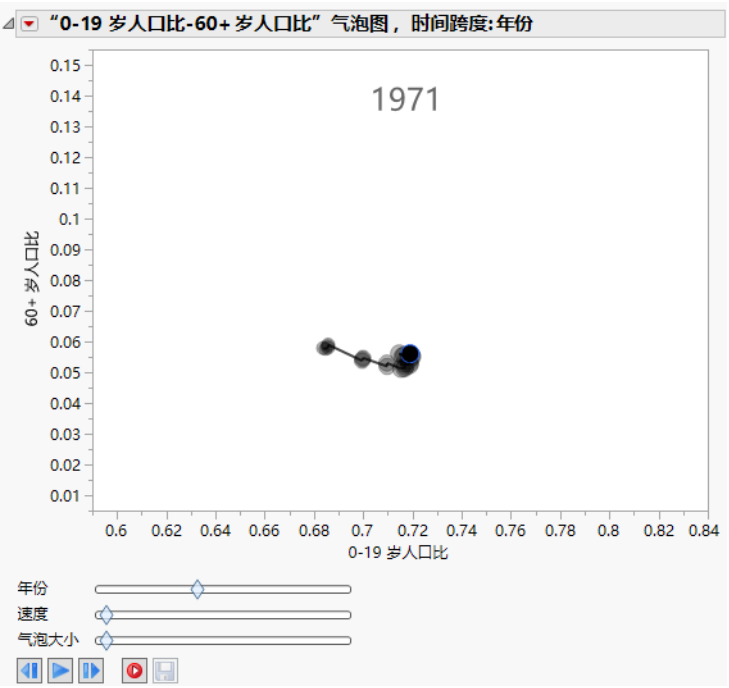
1. 选择**帮助** > **样本数据库**，然后打开 PopAgeGroup.jmp。
2. 选择**图形** > **气泡图**。
3. 选择 **60+ 岁人口比**并点击 **Y**。
4. 选择 **0-19 岁人口比**并点击 **X**。
5. 选择**年份**并点击**时间**。
6. 点击**确定**。

图 5.7 包含一个 “时间”变量的初始报表窗口



7. 点击该气泡将其选定。
   数据表中的所有行也随之突出显示。
8. 点击 “气泡图”红色小三角并选择**尾迹气泡 > 全部**和**尾迹线 > 全部**。
9. 点击播放按钮。
   气泡图开始播放动画，显示单个气泡的尾迹。

图 5.8 随时间播放的气泡图动画



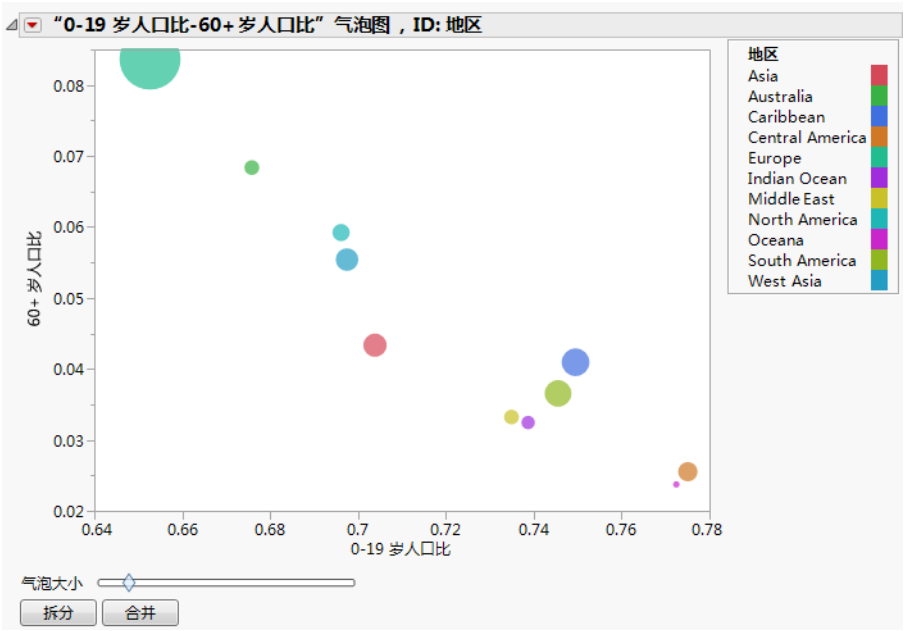
仅指定 ID 变量并且拆分气泡的示例

对于静态气泡图，您可以指定一个或两个 ID 变量而不指定**时间**变量。生成的气泡图在每个 ID 值处都包含一个气泡。请注意，尽管该气泡图是静态的，但您可以拆分这些气泡。

1. 选择**帮助** > **样本数据库**，然后打开 PopAgeGroup.jmp。
2. 选择**图形** > **气泡图**。
3. 选择 **60+ 岁人口比**并点击 **Y**。
4. 选择 **0-19 岁人口比**并点击 **X**。
5. 选择**地区**和**国家**并点击 **ID**。
6. 选择**地区**并点击**颜色**。
7. 点击**确定**。

   随即出现初始报表窗口。（图 5.9 使用默认的 JMP 颜色。）

图 5.9 包含 ID 变量的初始报表窗口

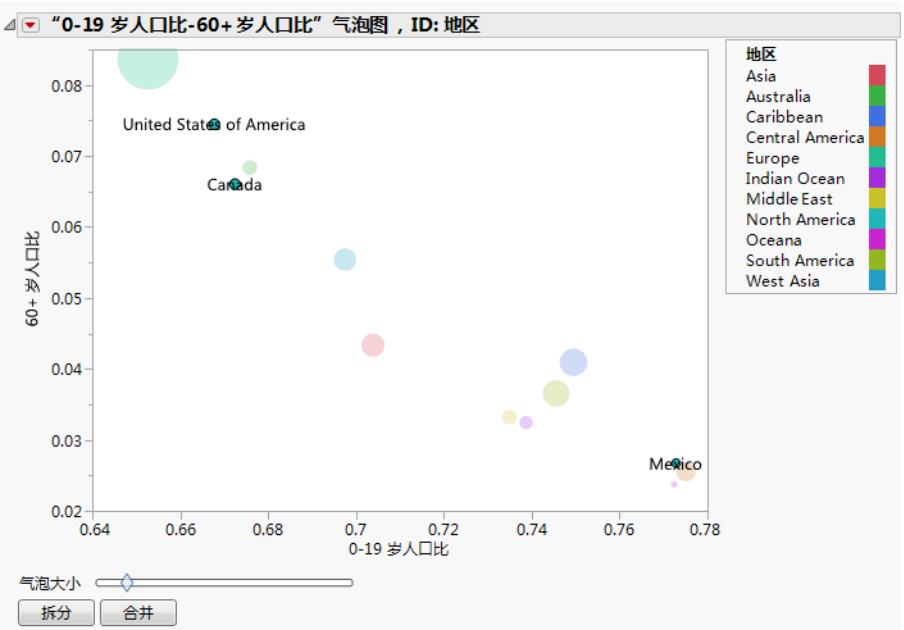


将表示 North America （北美）地区的气泡拆分为国家。

8. 点击表示 North America 的气泡 （在气泡上方悬停以查看其标签，或使用图例查找 North America 的颜色）。
9. 点击**拆分**。

   您可以看到 North America 气泡拆分为三个气泡，分别表示 North America 地区内的三个国家 — United States of America （美国）、 Canada （加拿大）和 Mexico （墨西哥）。

图 5.10 拆分 North America 气泡



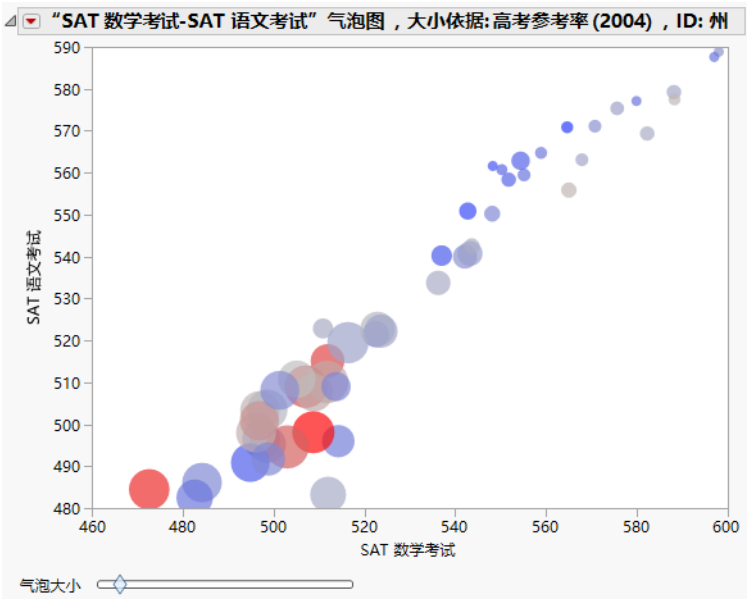
静态气泡图的示例

该示例使用 SATByYear.jmp 样本数据表，其中包含挑选出来的 2004 年美国部分人口的 SAT 语文考试和数学考试的分数。

1. 选择**帮助** > **样本数据库**，然后打开 SATByYear.jmp。
2. 选择**图形** > **气泡图**。
3. 选择 **SAT 语文考试**并点击 **Y**。
4. 选择 **SAT 数学考试**并点击 **X**。
5. 选择**州**并点击 **ID**。
6. 选择**高考参考率 (2004)** 并点击**大小**。
7. 点击**确定**。

   随即显示报表窗口。（图 5.11 使用默认的 JMP 颜色。）

图 5.11 “静态气泡图”报表窗口



可以得到以下结论：

- 较高的语文分数看起来与较高的数学分数相关，因为这两个分数的轨迹在气泡图中非常接近。这表示语文分数与数学分数之间存在相关性。
- 较大的气泡表示 2004 年 SAT 考试考生参考率较高的美国各州。这些较大的气泡全部集中在图形左下方。这表示具有较高考生参考率的州的数学和语文分数都较低。

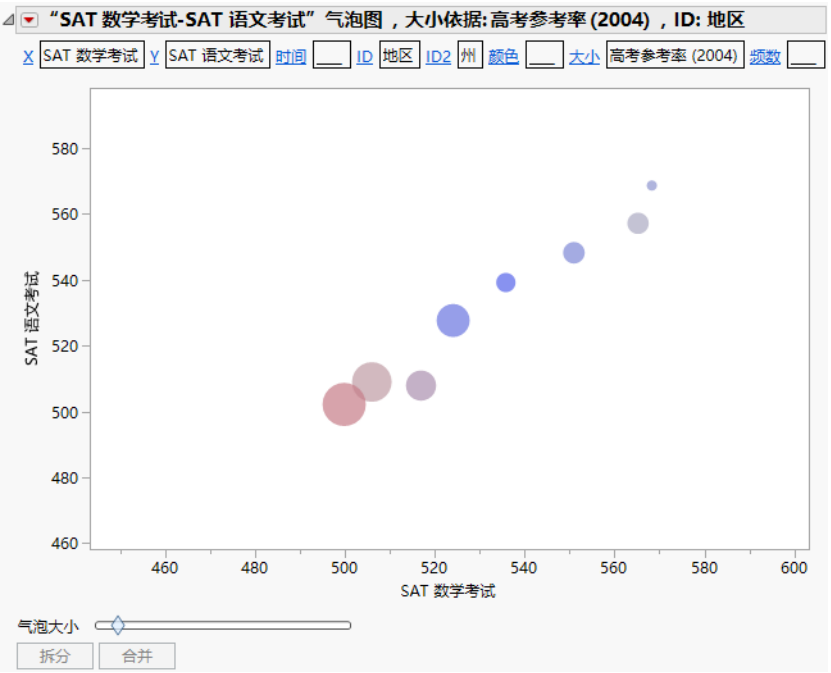
将主要按州对气泡分组改为主要按地区对气泡分组：

1. 点击“气泡图”红色小三角并选择**显示角色**。
2. 点击 ID 链接。
3. 选择**地区**并点击**确定**。

   地区现在成为主要的 ID 变量。
4. 点击 ID2 链接。
5. 选择**州**并点击**确定**。

   州现在成为次要的 ID 变量。

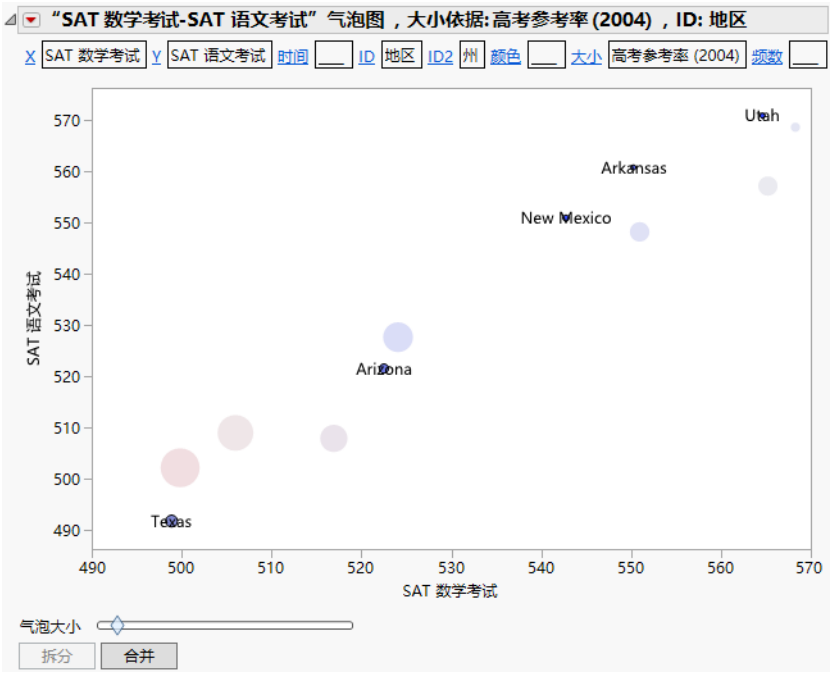
图 5.12 按 “地区” 和 “州” 分组的气泡图示例



6. 点击表示 Southwest （西南）地区的气泡 （在气泡上方悬停或点击气泡以查看其标签）。
7. 点击**拆分**。

   现在气泡将按次要 ID 变量 （即州）来拆分。您现在可以看到 Southwest 地区内的各州。

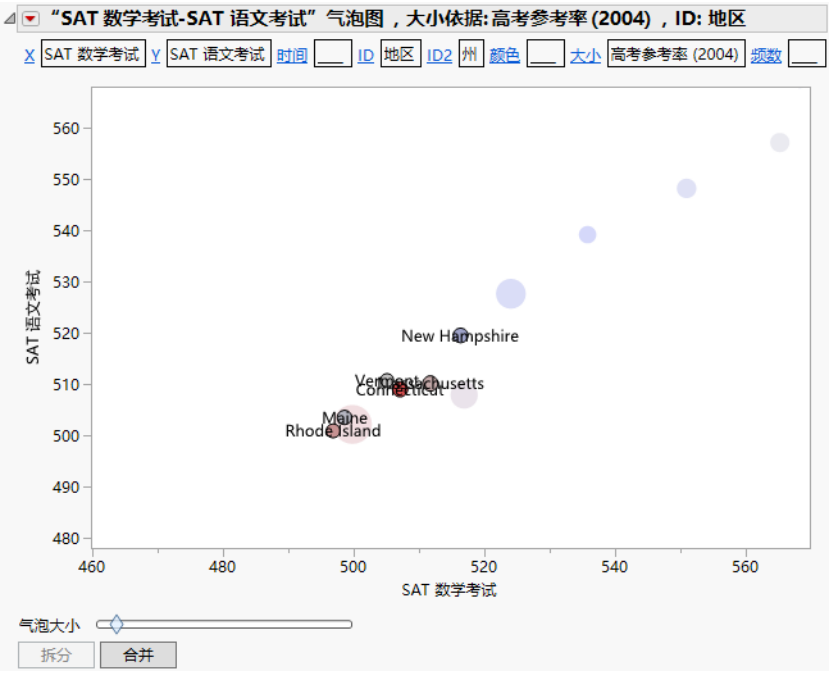
图 5.13 按 “州”拆分 Southwest 地区的示例



您可以看到 Southwest 各州之间的分数存在显著变异。

8. 点击**合并**再次合并 Southwest 各州。
9. 要进行比较，请点击 New England （新英格兰）气泡 （在气泡上方悬停或点击气泡以查看其标签）。
10. 点击**拆分**。

图 5.14 按 “州”拆分 New England 地区的示例



您可以看到 New England 各州的变异不像 Southwest 各州那么大。

包含分类 Y 变量的气泡图示例

到现在为止显示的所有示例都使用连续 Y 变量。若使用分类（名义型或有序型）Y 变量，气泡图将具有不同的外观。

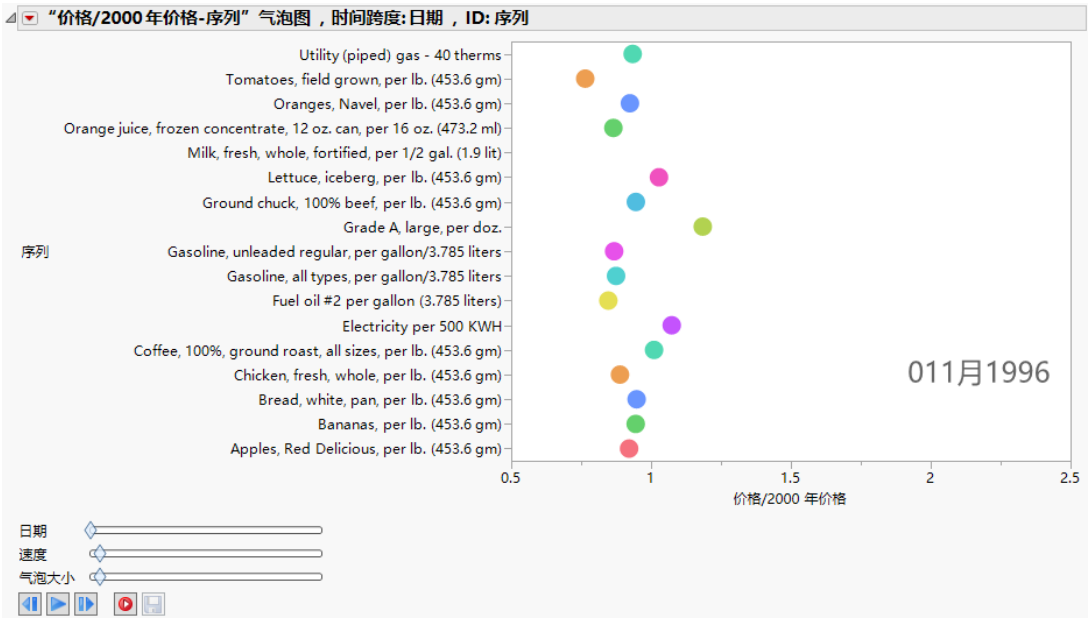
该示例使用 blsPriceData.jmp 样本数据表，其中显示若干年间的商品价格。由于美元币值随时间变动，名为 “价格 /2000 年价格”的列显示任意给定时间的商品价格与 2000 年的价格之比。

1. 选择**帮助** > **样本数据库**，然后打开 blsPriceData.jmp。
2. 选择**图形** > **气泡图**。
3. 选择**序列**并点击 Y。
4. 选择**价格 /2000 年价格**并点击 X。
5. 选择**日期**并点击**时间**。
6. 点击**确定**。

   随即显示报表窗口。（图 5.15 使用 JMP 默认颜色。）

   这将生成一个气泡图，在点击播放按钮播放动画时，该图显示价格气泡依据其价格比左右移动。

图 5.15 播放气泡动画的静态示例



为了便于解读，请添加网格线：

7. 双击分类轴。
8. 在“Y 轴设置”窗口中，选择**显示网格**。
9. 点击**确定**。

要播放气泡图动画，请点击播放按钮。价格气泡根据其价格比左右移动。

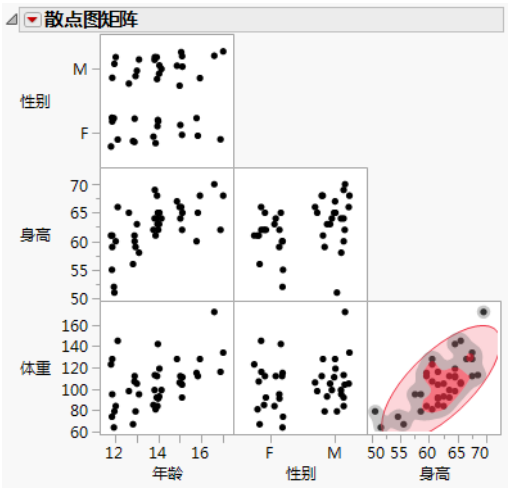
第 6 章

散点图矩阵

同时查看多个二元关系

使用“散点图矩阵”平台，您可以同时评估多个变量之间的关系。散点图矩阵是有序的二元图形的集合。要进行进一步分析，您可以定制对数据总体标绘密度椭圆的散点图，或对您分组后的数据分别标绘密度椭圆。

图 6.1 散点图矩阵的示例



目录

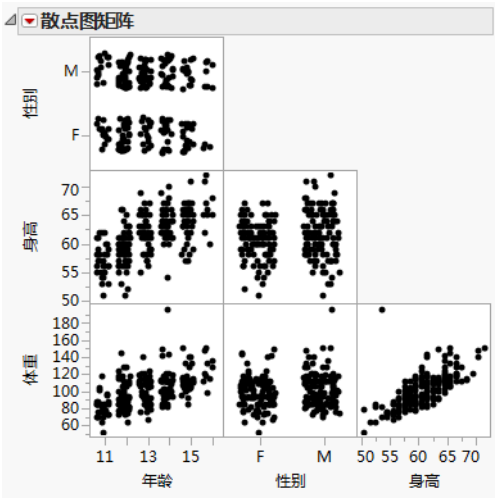
散点图矩阵的示例 155
启动“散点图矩阵”平台 156
更改矩阵格式 158
“散点图矩阵”窗口 158
“散点图矩阵”平台选项 159
使用分组变量的示例 160
创建分组变量 161

散点图矩阵的示例

该示例显示如何创建散点图矩阵。

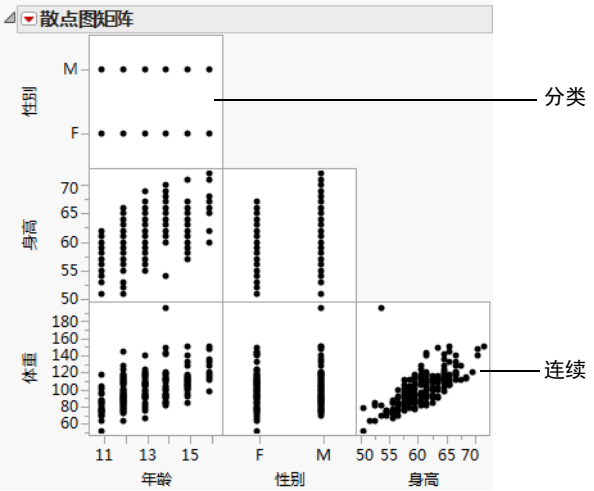
1. 选择**帮助** > **样本数据库**，然后打开 Students.jmp。
2. 选择**图形** > **散点图矩阵**。
3. 选择**年龄**、**性别**、**身高**和**体重**并点击 **Y，列**。
4. 点击**确定**。

图 6.2 散点图矩阵的示例



在该示例中，您可以看到身高-体重图形不同于年龄-性别图形。若通过点击红色小三角菜单并选择**随机散布点**来禁用随机散布，则差异变得更显著。

图 6.3 无随机散布的散点图矩阵示例

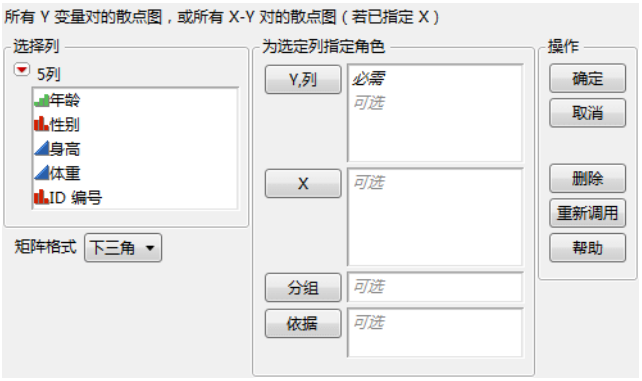


身高-体重图形显示连续数据，年龄-性别图形则显示分类数据。

启动“散点图矩阵”平台

通过选择**图形** > **散点图矩阵**可启动 “散点图矩阵”平台。

图 6.4 “散点图矩阵”启动窗口



Y，列、X 指定用于 Y 和 X 的列。

- 若仅将变量分配给 **Y，列**角色，则它们同时显示在水平轴和垂直轴上。
- 若将变量同时分配给 **Y，列**和 **X** 角色，则 **Y，列**变量显示在垂直轴上。**X** 变量显示在水平轴上。使用该方法，您可以生成矩形矩阵或具有不同但重叠以形成矩阵轴的变量集的矩阵。

分组 若将变量指定给**分组**角色，则可以为**分组**变量的每个水平添加着色密度椭圆。请参见第 160 页的 “使用分组变量的示例”。

依据 该选项为**依据**变量的每个水平生成单独的散点图矩阵。若指定了两个**依据**变量，将为这两个**依据**变量的水平的每种可能组合生成单独的图形。

矩阵格式 **矩阵格式**可以为以下三种格式之一：**上三角**、**下三角**或**正方形**。请参见第 158 页的 “更改矩阵格式”。

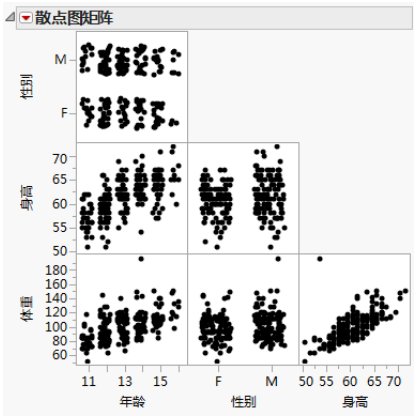
点击**确定**后，将显示“散点图矩阵”窗口。请参见第 158 页的 “ “散点图矩阵”窗口”。

更改矩阵格式

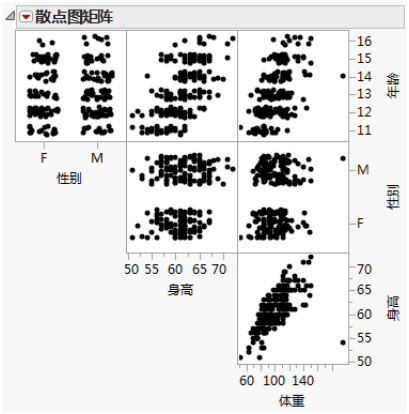
矩阵格式可以为以下三种格式之一：上三角、下三角或正方形。

图 6.5 矩阵格式的示例

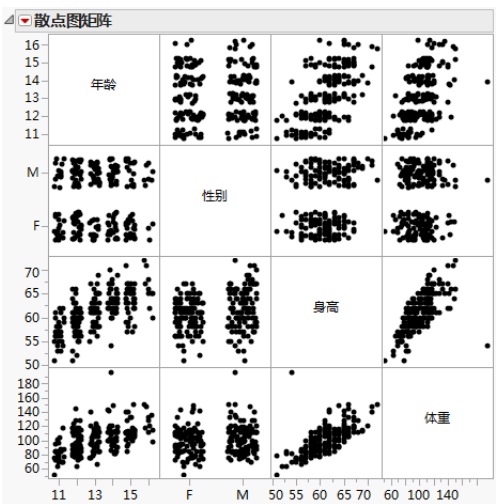
下三角



上三角



正方形



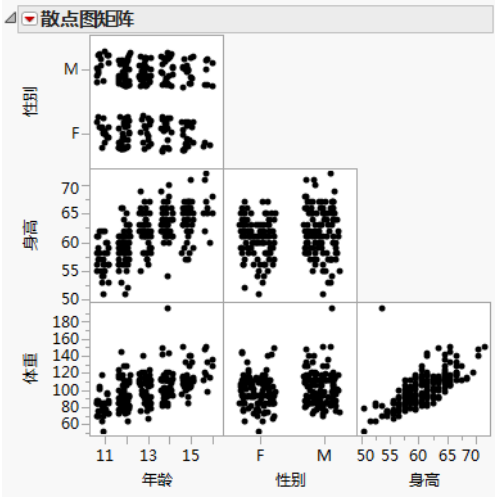
“散点图矩阵”窗口

“散点图矩阵”窗口显示二元图形的有序分组。在每个图形中，您可以检查每对变量之间的关系。

按照第 155 页的“散点图矩阵的示例”中的说明生成图 6.6 所示的图。

注意：有关其他选项的信息，请参见第 159 页的““散点图矩阵”平台选项”。

图 6.6 “散点图矩阵”窗口示例



通过以下两种方式之一拖放变量，以此替换图中变量：将一个轴上的变量拖放到另一个轴上来交换现有变量；或者，点击相关数据表的列面板中的某个变量并将其拖到轴上。该功能对于“正方形”格式的矩阵不可用。

“散点图矩阵”平台选项

“散点图矩阵”红色小三角菜单提供以下选项：

显示点 显示或隐藏散点图中的点。

随机散布点 启用或禁用散点图中点的随机散布。当至少有一个变量为有序型或名义型时，该选项可用。

拟合线 对散点图拟合简单的回归线及其均值置信区间。

密度椭圆 显示或隐藏密度椭圆的轮廓和区域。请参见第 160 页的 “使用分组变量的示例”。

着色椭圆 对每个椭圆内的区域着色。请参见第 160 页的 “使用分组变量的示例”。

椭圆覆盖率 允许您为要覆盖的椭圆选择 α 水平。

椭圆透明度 允许您选择着色椭圆的透明度，其中 0 为完全透明， 1 为完全不透明。

椭圆颜色 允许您为椭圆的轮廓和区域选择颜色。

非参数密度 显示或隐藏非参数密度，它表示数据点最密集的区域。当您具有很多点并且点的密度很难观察时，非参数密度估计很有帮助。

有两个分位数密度等高线。一个等高线包括 50% 的平滑密度，另一个等高线包括 90% 的平滑密度。因为百分比基于平滑密度，百分比可能不适合等高线内点的实际比例。

依据 ... 分组 在“依据 ... 分组”窗口中，您可以执行以下操作：

- 若未在启动窗口中选择**分组**变量，可以现在添加一个。
- 若在启动窗口中选择了**分组**变量，可以删除现有**分组**变量，或替换该**分组**变量。

请参见第 160 页的“使用分组变量的示例”。

锁定尺度 防止轴尺度和梯度图例尺度因数据变化或过滤更改而自动调整。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

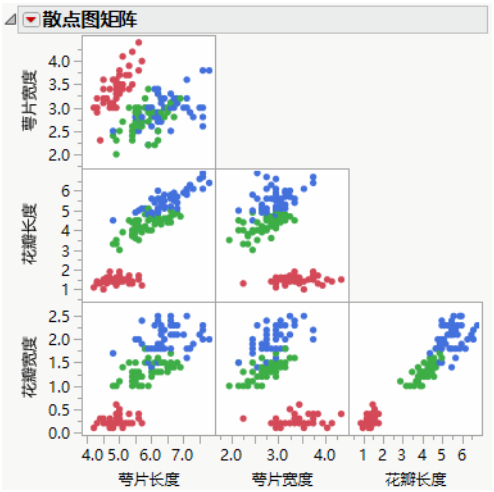
保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

使用分组变量的示例

该示例显示如何使用分组变量来创建散点图矩阵。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**图形** > **散点图矩阵**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，列**。
4. 选择**种类**并点击**分组**。
5. 点击**确定**。

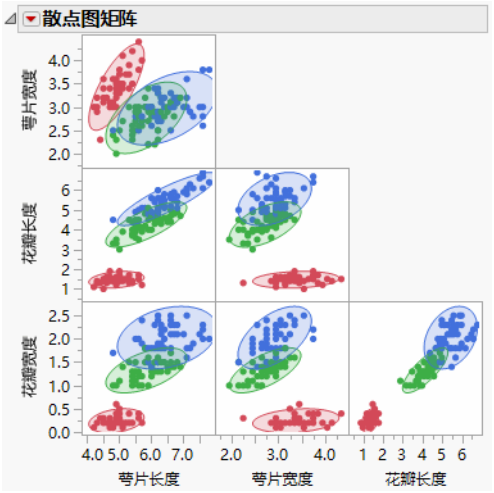
图 6.7 使用分组变量的初始示例



通过添加椭圆和着色来突出各分组：

6. 点击“散点图矩阵”红色小三角并选择**密度椭圆 > 密度椭圆**。
7. 点击“散点图矩阵”红色小三角并选择**密度椭圆 > 着色椭圆**。

图 6.8 使用椭圆的散点图矩阵示例



创建分组变量

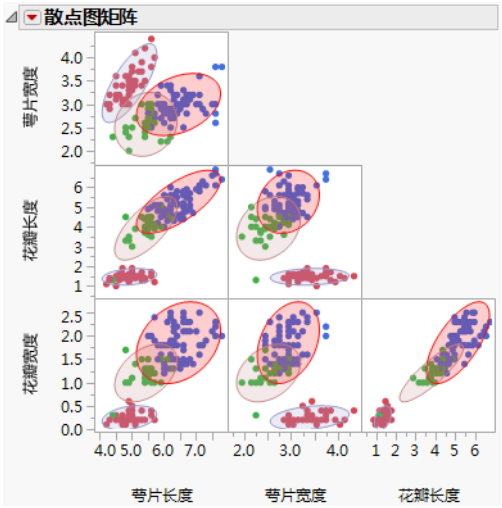
若您的数据没有分组变量，可以使用“聚类”平台创建一个。使用 Iris.jmp 数据，假定**种类**列不存在。您知道该数据来自鸢尾花的三个品种，因此要在组内创建三个聚类。

1. 使用 Iris.jmp 样本数据表，选择**分析 > 聚类 > 层次聚类**。

2. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，列**。
3. 点击**确定**。
4. 点击“层次聚类”红色小三角并选择**聚类数**。
5. 键入 3 表示三种不同的鸢尾花品种。
6. 点击**确定**。
7. 点击“层次聚类”红色小三角菜单并选择**保存聚类**。
8. 关闭“层次聚类”报表窗口，返回到 Iris.jmp 数据表。

   您可以看到已将**聚类**列添加到 Iris.jmp 数据表。
9. 执行“散点图矩阵”分析。按照第 160 页的“使用分组变量的示例”一节中的说明操作，但是使用**聚类**作为分组变量。

图 6.9 使用聚类变量的散点图矩阵示例



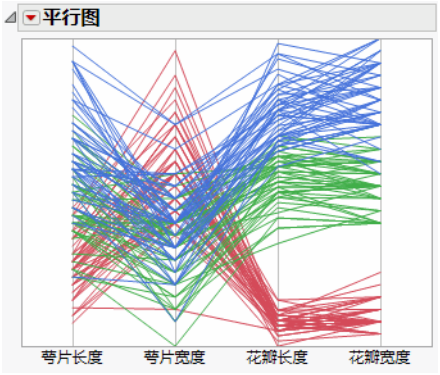
第 7 章

平行图

通过绘制平行坐标来查看多维数据中的模式

使用平行图，您可以直观表示数据表中的各个单元格。在平行图中绘制表示数据表中各行的相连线段。平行图最初由 Inselberg (1985) 开发，后来由 Wegman (1990) 推广使用。

图 7.1 平行图示例



目录

平行图示例 165
启动“平行图”平台 166
平行图 167
    解释平行图 167
“平行图”平台选项 169
“平行图”平台的更多示例 169
    检查鸢尾花测量值 170
    检查学生测量值 171

平行图示例

该示例使用 Dogs.jmp 样本数据表，其中包含对服用两种不同药物的 16 条狗的组织胺水平的测量值。组织胺水平分别取自未服药、服药 1 分钟、3 分钟和 5 分钟时的测量值。检查每种药物的组织胺水平的变动。

1. 选择**帮助** > **样本数据库**，然后打开 Dogs.jmp。

   要查看不同**药物**的区别，请按**药物**种类对平行图线条着色：

2. 选择**行** > **按列设定颜色或标记**。
3. 选择**药物**。

   若显示在 “颜色” 菜单中的选择是 **JMP 默认值**，则给 morphine 分配红色，给 trimeth 分配蓝色。

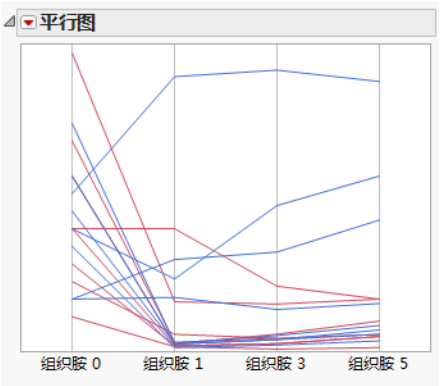
4. 点击**确定**。

创建平行图：

5. 选择**图形** > **平行图**。
6. 选择**组织胺 0**、**组织胺 1**、**组织胺 3** 和**组织胺 5**，然后点击 **Y，响应**。
7. 点击**确定**。

   随即显示报表窗口。

图 7.2 组织胺变量的平行图



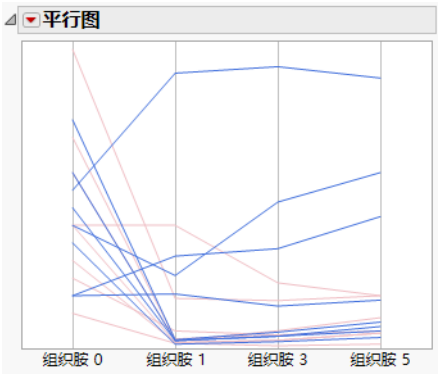
每一相连的线段表示一个观测。点击某一线段可查看其在数据表中对应的观测 （或行）。

要进一步探究数据，请将 trimeth 值隔离：

8. 选择**行** > **数据过滤器**。
9. 选择**药物**，然后点击添加 ✚。
10. 选择 trimeth。

平行图中将仅突出显示 trimeth 值。

图 7.3 突出显示的 trimeth 值



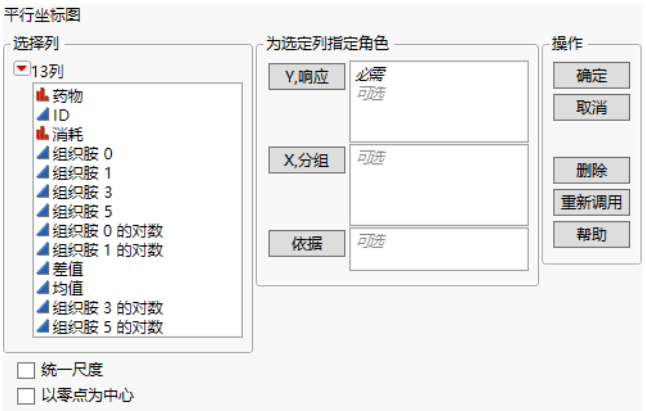
您可以观察到关于喂服 trimeth 的狗的组织胺水平的以下信息：

- 对于多数狗，组织胺水平在一分钟后急剧下降。
- 其中有四只狗的组织胺水平保持较高或进一步升高。您可以在此发现的基础上进一步研究，确定这些狗的组织胺水平有差别的原因。

启动 “平行图” 平台

通过选择**图形** > **平行图**可启动 “平行图” 平台。

图 7.4 “平行图” 启动窗口



Y，响应 变量显示在平行图的水平轴上。在平行图中绘制并连接这些值。

X，分组 为变量的每个水平生成单独的平行图。

依据 标识一列，该列创建的报表包含指定变量每个水平的单独分析。

统一尺度 以同一尺度表示所有变量，并向图中添加 *y* 轴。若不使用该选项，每个变量将采用不同的尺度。

为了确保正确比较，请在以相同尺度测量变量时选择该选项。

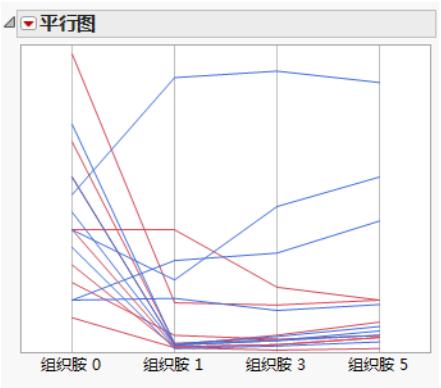
以零点为中心 使平行图 （而不是变量）以零点为中心。

点击**确定**后将出现平行图。请参见第 167 页的“平行图”。

平行图

要生成图 7.5 中所示的图，请按第 165 页的“平行图示例”中的说明操作。

图 7.5 “平行图”报表



平行图是可在一个图中显示任意数量的变量的几类坐标图之一。不过，变量之间的关系可能只有在以下情况下才显而易见：

- 变量并排显示
- 为变量的水平指定颜色来跟踪各组
- 选择线条来跟踪各组

提示：您可以为规格限添加参考线。相关信息，请参见《使用 JMP》。

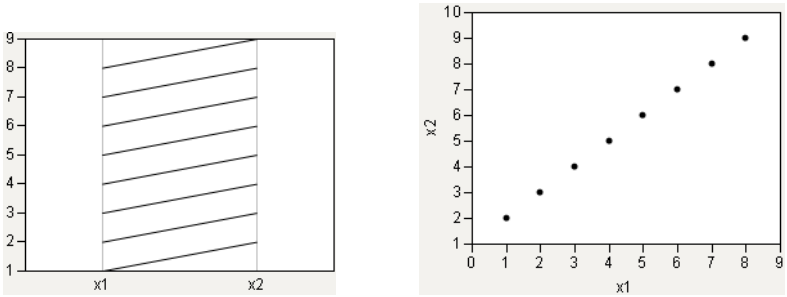
解释平行图

为帮助您解释平行图，可以将平行图与散点图作比较。在下面的每个图中，平行图都显示在左侧，散点图都显示在右侧。

强正相关

以下关系显示了强正相关。请注意平行图中各条线的一致性。

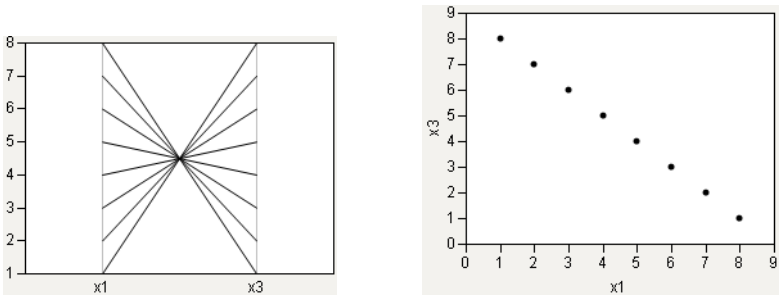
图 7.6 强正相关



强负相关

相比之下，强负相关在平行图中显示出窄颈部形式。

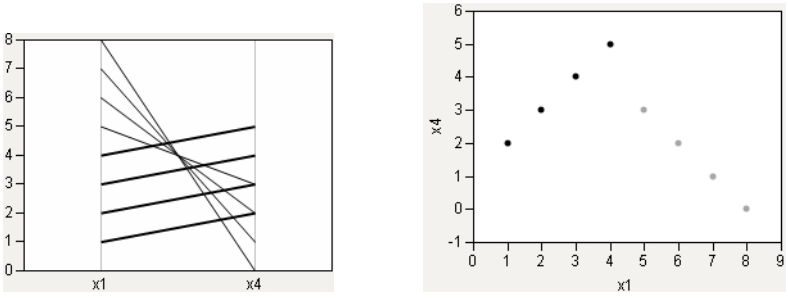
图 7.7 强负相关



共线组

现在，请考虑同时包含两种情形的情况：有两个组，均具有很强的共线性。一组具有正斜率，另一组具有负斜率。在图 7.8 中，正斜率组突出显示。

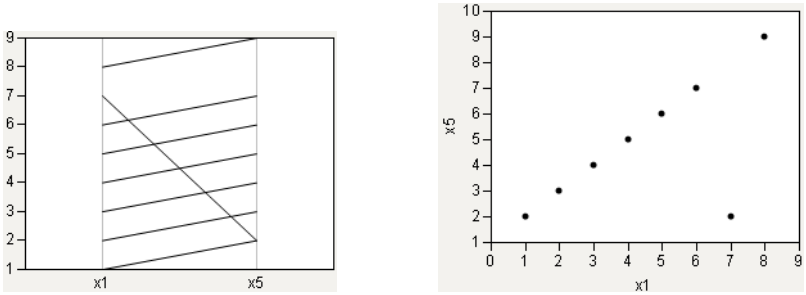
图 7.8 共线组：平行图和散点图



单个离群值

最后，请考虑单个离群值的情况。平行图显示各条线基本一致，只有一个明显的例外。

图 7.9 单个离群值：平行图和散点图



相关信息

- 第 169 页的““平行图”平台的更多示例”

“平行图”平台选项

“平行图”红色小三角菜单提供以下选项：

显示反转复选框 反转一个或多个变量的尺度。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

有关右击平行图时显示的上下文菜单选项的详细信息，请参见《使用 JMP》。

“平行图”平台的更多示例

- “检查鸢尾花测量值”
- “检查学生测量值”

检查鸢尾花测量值

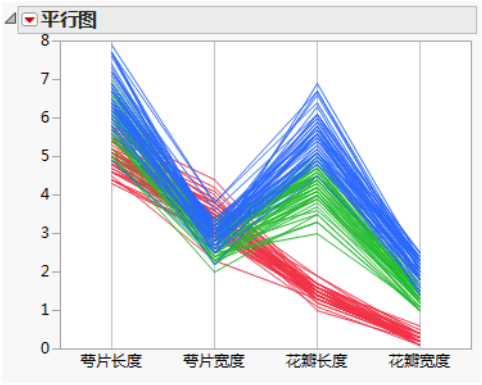
下面的示例使用 Fisher 的 Iris 数据集 (Mardia, Kent, and Bibby 1979)。Iris.jmp 样本数据表包含 setosa、versicolor 和 virginica 这三种鸢尾花的萼片长度、萼片宽度、花瓣长度和花瓣宽度的测量值（以厘米表示）。要查找可以区分这三种花的特征，请检查以下测量值。

在一个平行图中检查三个种类

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**图形** > **平行图**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，响应**。
4. 选中**统一尺度**复选框。
5. 点击**确定**。

   随即显示报表窗口。

图 7.10 一个平行图中的三个种类



在该平行图中，三个种类全都表示在同一个图中。这三个种类分别对应以下颜色：

- 蓝色对应 virginica。
- 绿色对应 versicolor。
- 红色对应 setosa。

从平行图中，您观察到：

- 对于萼片宽度，setosa 值看起来高于 virginica 和 versicolor 的值。
- 对于花瓣宽度，setosa 值看起来低于 virginica 和 versicolor 的值。

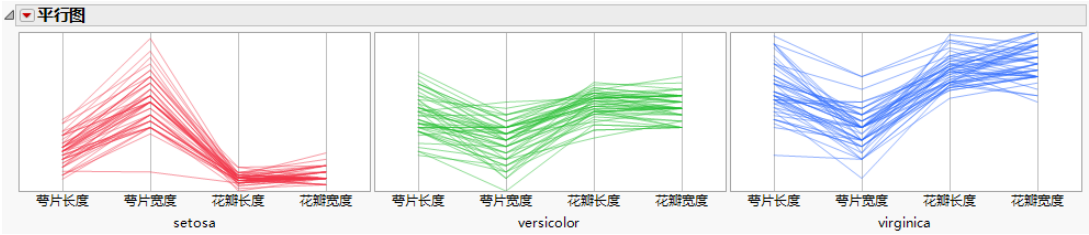
在不同的平行图中检查三个种类

1. 从 Iris.jmp 样本数据表中，选择**图形** > **平行图**。

2. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，响应**。
3. 选择**种类**并点击 **X，分组**。
4. 点击**确定**。

   随即显示报表窗口。

图 7.11 不同平行图中的三个种类



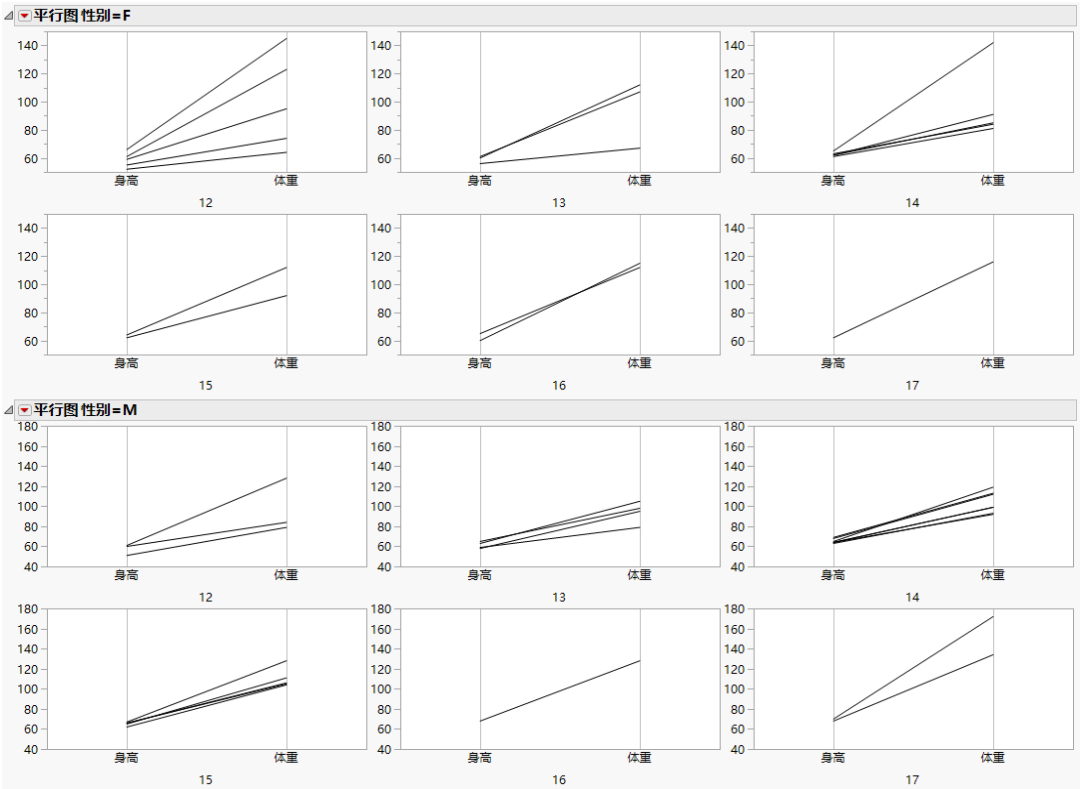
每个种类都表示在单独的平行图中。

检查学生测量值

下面的示例使用 Big Class.jmp 样本数据表，其中包含关于 40 个学生的年龄、性别、身高和体重数据。检查不同变量之间的关系。

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**图形** > **平行图**。
3. 选择**身高**和**体重**，并点击 **Y，响应**。
4. 选择**年龄**并点击 **X，分组**。
5. 选择**性别**并点击**依据**。
6. 选中**统一尺度**复选框。
7. 点击**确定**。

图 7.12 按性别排列和依据年龄分组的身高和体重



您观察到：

- 在 13 岁女生组中，有一个女生的体重低于她所在年龄组中其他女生的体重。若点击表示该更低体重的线条，相应的个体 (Susan) 将在数据表中突出显示。
- 在 14 岁女生组中，有一个女生的体重高于她所在年龄组中其他女生的体重。若点击表示该更高体重的线条，相应的个体 (Leslie) 将在数据表中突出显示。

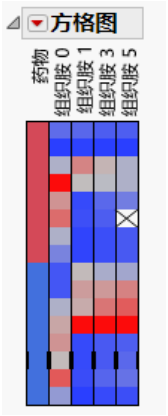
第 8 章

方格图

查看变量的颜色强度图

使用方格图，您可以直观表示数据表中的每个单元格。方格图是数据表的直接表示，因为它们绘制方格的矩形阵列，其中每个方格对应一个数据表条目。方格图因为基因组领域用于浏览大量基因表达水平的值而变得普及起来。

图 8.1 方格图的示例



目录

方格图的示例 175
启动“方格图”平台 176
方格图 176
“方格图”平台选项 177
    方格图的右击菜单 178
“方格图”平台的更多示例 179

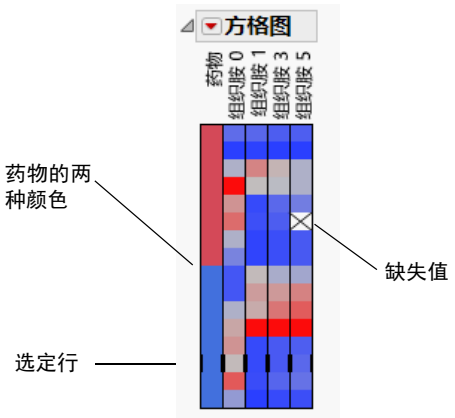
方格图的示例

了解如何在 JMP 中创建和分析方格图。该示例使用 Dogs.jmp 样本数据表，其中包含对服用两种不同药物的 16 条狗的组织胺水平的测量值。组织胺水平分别取自未服药、服药 1 分钟、 3 分钟和 5 分钟时的测量值。检查每种药物的组织胺水平的变动。

1. 选择**帮助** > **样本数据库**，然后打开 Dogs.jmp。
2. 选择从底部起的第三行 （第 14 行）。
3. 选择**图形** > **方格图**。
4. 选择**药物**、**组织胺 0**、**组织胺 1**、**组织胺 3** 和**组织胺 5**，然后点击 **Y，响应**。
5. 点击**确定**。

   随即显示报表窗口。

图 8.2 Dogs.jmp 方格图



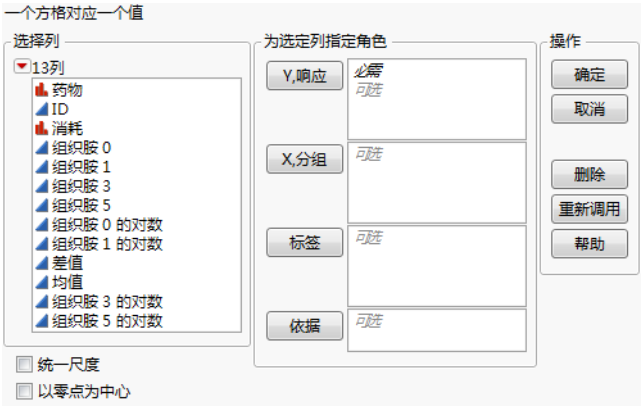
请注意以下事项：

- 有两种类型的药物，它们用两种不同的颜色表示。
- 为组织胺水平分配从蓝色到红色的颜色梯度。
- 任何缺失值用 X 表示。
- 选择从底部起的第三行，黑色的线显示在该方格旁边。

启动“方格图”平台

通过选择**图形** > **方格图**可启动“方格图”平台。

图 8.3 “方格图”启动窗口



Y，响应 变量显示在方格图的水平轴上。每个方格表示一个值。

X，分组 为变量的每个水平生成单独的方格图。

标签 按指定的变量为每行添加标签。请参见第 179 页的 ““方格图”平台的更多示例”。

依据 标识用于创建报表的一个列，该报表包括变量的每个水平的单独分析。

统一尺度 在同一尺度上表示所有变量。若不使用该选项，每个变量将采用不同的尺度。

以零点为中心 使方格图以零点为中心。

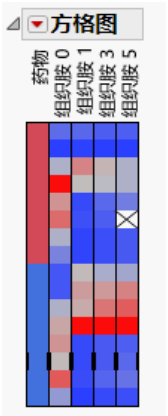
点击**确定**后，将显示“方格图”窗口。请参见第 176 页的“方格图”。

方格图

方格图呈现出着色方格与数据表条目一一对应关系。基于列中找到的值的范围和类型为每个方格指定颜色。

注意：数据表中已排除的任何行也会在方格图中隐藏。

图 8.4 方格图



注意：要生成图 8.4 中所示的图，请按第 175 页的“方格图的示例”中的说明操作。

请注意以下关于方格图的信息：

- 名义型变量对每个水平使用不同的颜色。您可以使用数据列的**值颜色**属性（位于**列信息**命令中）定制名义型和有序型颜色控制所显示的等高线，并支持您定。
- 为连续变量指定颜色梯度以显示平滑的变量值范围。
- 像连续变量那样按顺序对有序型变量确定尺度。
- 存在一些离群值时，尺度使用除极端类别之外的居分布中间 90% 的类别，使得离群值不过度影响尺度。

相关信息

- 第 179 页的“‘方格图’平台的更多示例”

“方格图”平台选项

“方格图”红色小三角菜单提供以下选项：

图例 显示或隐藏图例。

排列图形 指定在开始下一行的图形前在同一行上要放置多少个图形。

仅当指定 **X，分组**变量时该选项才可用。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

方格图的右击菜单

右击方格图可找到以下选项：

图形类型 确定图形的外观。请参见第 178 页的 “更改图形类型”。

颜色主题 显示影响色图中的连续变量的颜色主题列表。默认颜色主题为**由蓝经灰到红**（对应于小值到中值再到大值）。使用**由白到黑**可创建灰度图。

> **注意：**要查看定制颜色，必须首先创建它们。选择**文件 > 首选项 > 图形**。在“颜色主题”区域中，点击要创建的颜色主题的类型，点击**新建**，然后更改颜色。有关创建定制颜色主题的详细信息，请参见《使用 JMP》。

升序排序 将图形的行按列值从小到大排序。要排序，请在某列下的图形中右击，然后选择**升序排序**。整个图重新排列以接受排序。请参见第 179 页的 “‘方格图’平台的更多示例”。

降序排序 将图形的行按列值从大到小排序。要排序，请在某列下的图形中右击，然后选择**降序排序**。整个图重新排列以接受排序。

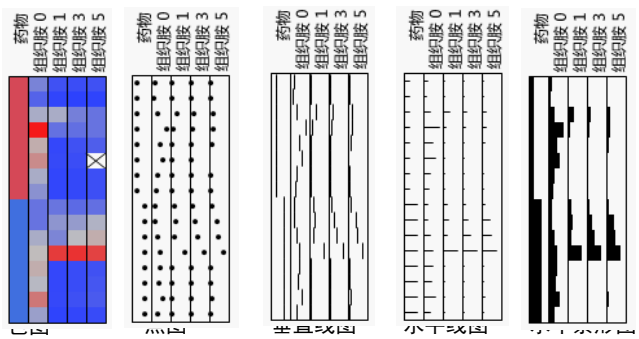
无分隔线 绘制或删除分隔各列的线。

注意：有关右击标签时显示的弹出选项的详细信息，请参见《使用 JMP》。

更改图形类型

使用**图形类型**选项可更改方格图的外观。

图 8.5 图形类型



“方格图”平台的更多示例

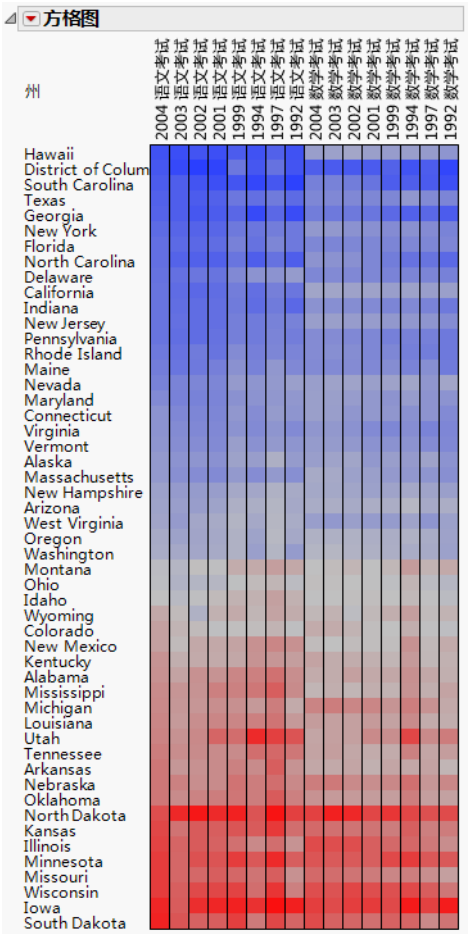
该示例使用 SAT.jmp 样本数据表，其中包含全美 50 个州的 SAT 考试分数 （被分为语文和数学两部分）。

1. 选择**帮助** > **样本数据库**，然后打开 SAT.jmp。
2. 选择**图形** > **方格图**。
3. 选择所有年份的语文考试分数，然后点击 **Y，响应**。
4. 选择所有年份的数学考试分数，然后点击 **Y，响应**。
5. 选择**州**并点击**标签**。
6. 点击**确定**。

   随即显示报表窗口。
7. 右击 **2004 语文考试** （左上角的单元格）下的图形，然后选择**升序排序**。

   这将按 2004 年的语文考试分数对方格图排序。

图 8.6 SAT 分数的方格图



您可以看到：

- 夏威夷州的 2004 年语文考试分数最低，南达科他州的 2004 年语文考试分数最高。
- 夏威夷州的数学分数和语文分数有明显差异。夏威夷州的数学分数达到平均值（用灰色颜色值表示），语文分数则低于平均值（用蓝色颜色值表示）。夏威夷州似乎是离群值，因为它具有显著不同的数学分数模式和语文分数模式。
- 北达科他州的数学分数和语文分数没有明显差异。该州的数学和语文分数普遍较高（用红色颜色值表示）。

有关颜色主题的说明，请参见第 178 页的“方格图的右击菜单”。

第 9 章

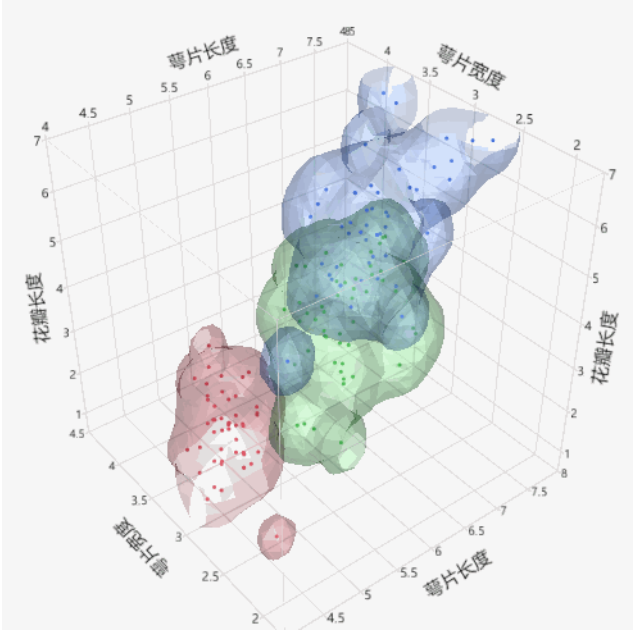
三维散点图

创建数据的旋转三维视图

“三维散点图”平台将相关数据表中数值列的值显示在可旋转的三维视图中。一次最多可以显示从相关数据表中选择的三个列。

为帮助在较高维度中直观演示变异，三维散点图可以在您请求主成分时显示点和变量的双标图表示。最显著的数据方向显示在三维散点图报表中。

图 9.1 三维散点图示例



目录

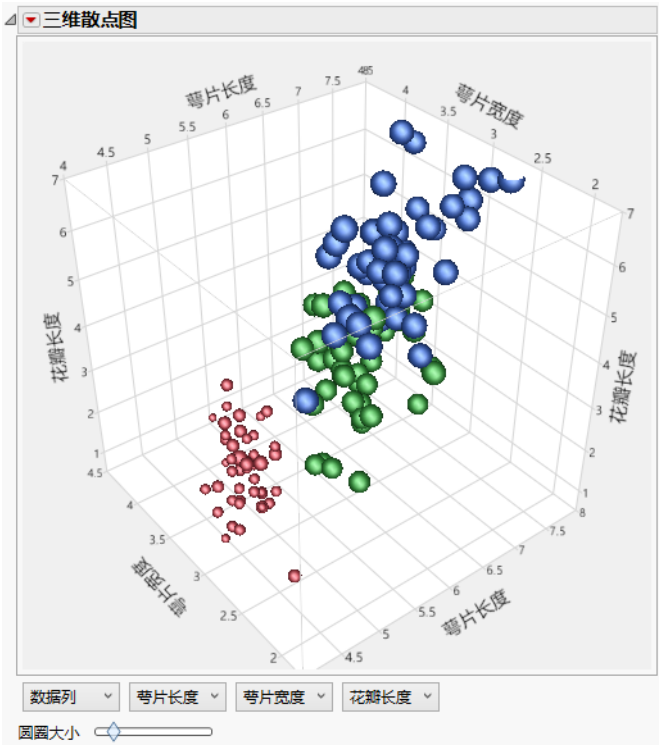
三维散点图示例 183
启动“三维散点图”平台 184
“三维散点图”报表 185
旋转三维散点图 186
更改轴上的变量 187
调整轴 187
为数据点指定颜色和标记 187
在数据表中指定颜色和标记 188
“三维散点图”平台选项 188
正态等高线椭圆 190
非参数密度等高线 190
弹出菜单 192
“三维散点图”平台的更多示例 194
未分组正态等高线椭圆的示例 194
已分组正态等高线椭圆的示例 195
已分组非参数密度等高线的示例 196

三维散点图示例

该示例使用 Iris.jmp 样本数据表，其中包含三种鸢尾花的萼片长度、萼片宽度、花瓣长度和花瓣宽度的测量值。

1. 打开 Iris.jmp 样本数据表。
2. 选择**图形** > **三维散点图**。
3. 选择**萼片长度**，**萼片宽度**和**花瓣长度**，然后点击 **Y，列**。
4. 选择**花瓣宽度**并点击**权重**。
5. 点击**确定**。

图 9.2 初始三维散点图的示例

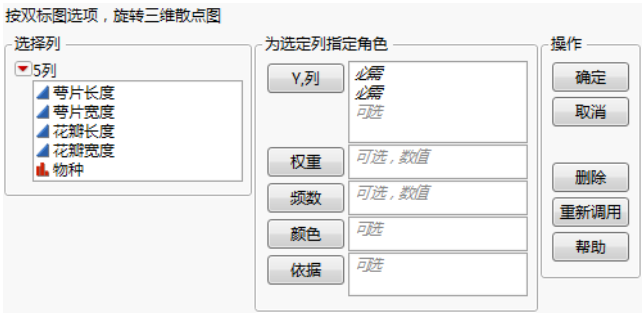


现在您可以旋转该三维散点图以查看各变量之间的关系。在本例中，数据点分别设置为蓝色、红色和绿色。您可能想要旋转该散点图，以便更清晰地查看红点和绿点之间的关系。

启动“三维散点图”平台

通过选择**图形** > **三维散点图**可启动 “三维散点图”平台。

图 9.3 “三维散点图”启动窗口



Y，列 选择要在三维散点图上标绘的变量。您选择变量的顺序决定数据点在轴上显示的位置：

- 第一个变量显示在 *x* 轴上。
- 第二个变量显示在 *y* 轴上。
- 第三个变量显示在 *z* 轴上。

您可以通过散点图下方的下拉菜单以交互方式指定其余变量。

权重 使用**权重**变量可以：

- 指定数据的权重（重要性或影响）
- 将用于设置点大小的第四个变量可视化

注意：红色小三角选项解释了**权重**变量。若不想在分析中解释该变量，请将其从启动窗口中删除。

指定**权重**变量时，JMP 会将数据点绘制为球状。这些球的尺度经过调整，用其体积来表示权重值。通过点击并拖动散点图下方的**圆圈大小**滑块可调整球的大小。

频数 标识数据表列，该列的变量为每行指定频数。当为汇总数据中的每行指定频数时，该选项很有用。

颜色 根据选定变量对点着色。若选定变量为分类变量 （名义型或有序型），则为每个类别指定不同颜色。若选定变量为连续变量，则使用颜色梯度。

依据 为每个 “依据”变量值生成单独的三维散点图。若指定了两个 “依据”变量，将为这两个 “依据”变量的各种组合生成单独的图形。

点击**确定**后将显示“三维散点图”报表窗口。请参见第 185 页的“ “三维散点图”报表”。

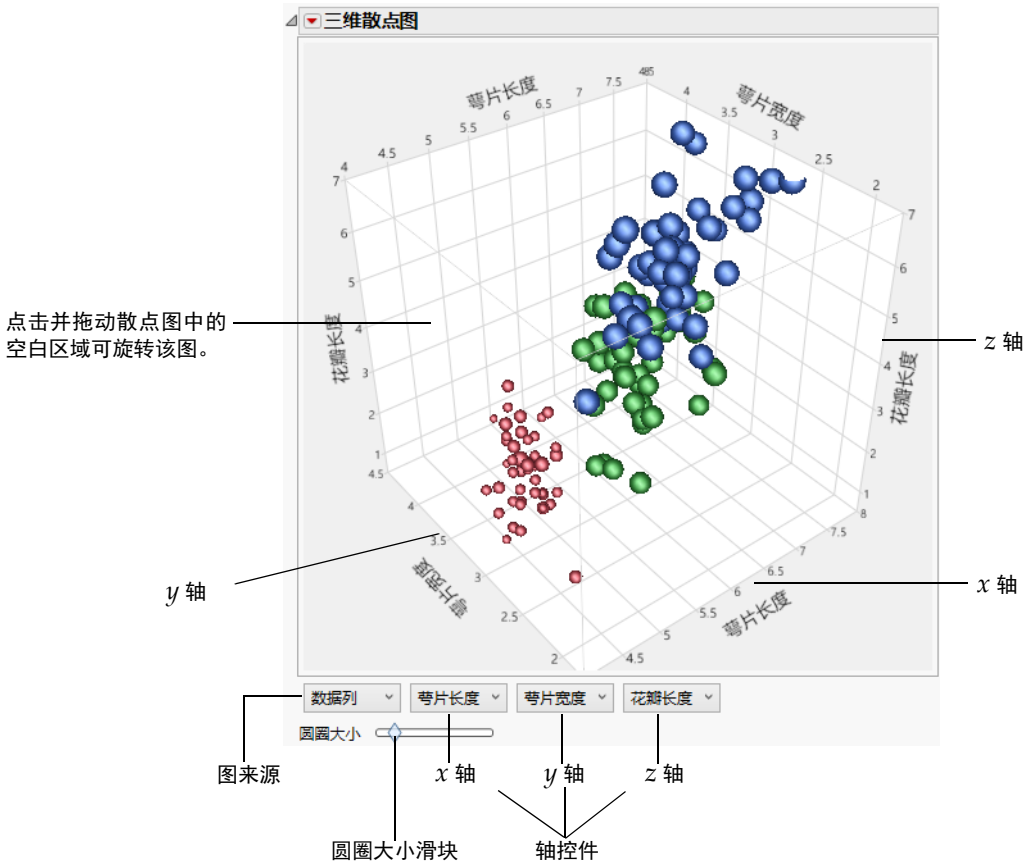
“三维散点图”报表

要生成图 9.4 所示的三维散点图，请按第 183 页的“三维散点图示例”中的说明操作。

“三维散点图”报表显示数据的三维可旋转视图。请参见图 9.4。在启动窗口中，选择变量，然后创建报表。选择的变量显示在三维散点图的 *x*、*y* 和 *z* 轴上。一次最多显示三个变量。

注意：在三维散点图中不支持十字准线工具。

图 9.4 显示在 “三维散点图”报表上的信息的示例



注意：数据表中已排除的任何行也会在三维散点图中隐藏。

图来源 图来源框指示图中数据的来源。

圆圈大小滑块 点击并拖动**圆圈大小**滑块可调整球的大小，同时保持其相对大小。

> **注意**：仅当您已指定**权重**变量时才会显示**圆圈大小**滑块。

轴控件 选择要显示在各个轴上的变量。选择**其他**选项可添加新变量。

下一个轴系 循环显示各个轴控件，查看任何隐藏变量。请参见第 187 页的“更改轴上的变量”。

> **注意**：仅当分析中包含三个以上变量时才显示**下一个轴系**按钮。

创建三维散点图之后，您可以添加如下功能：在特定数据点周围显示椭圆、显示单独的主成分、旋转成分、连接点等。请参见第 188 页的“‘三维散点图’平台选项”。

您还可以在散点图中或在其相关数据表中为数据点指定颜色和符号（或标记）。请参见第 187 页的“为数据点指定颜色和标记”和第 188 页的“在数据表中指定颜色和标记”。

旋转三维散点图

可以通过四种方式旋转三维散点图报表：

- 点击并拖动三维散点图上的空白区域。三维散点图按您拖动鼠标的方向旋转。

> **注意**：点击并拖动三维散点图上轴和数据点之外的空白区域。拖动轴将重新调整轴尺度。拖动某个数据点仅选择该点。

- 滑动鼠标滚轮。三维散点图只是上下翻转。
- 按住箭头键。（使用数字键盘上的箭头之前，请确认已经禁用 Num Lock 键。）
- 按住 Esc 键。三维散点图只是左右翻转。

在每种情况下，只要您按住鼠标按钮、箭头键或 Esc 键，三维散点图就会旋转。滑动鼠标滚轮时也会继续旋转。

您还可以采用以下任意方式连续旋转三维散点图：

- 点击并拖动：按住 Shift 键，点击并拖动图上的空白区域，然后松开 Shift 键。拖动鼠标的速度越快，三维散点图旋转的速度越快。
- 鼠标滚轮：按住 Shift 键，滑动滚轮，然后松开滚轮。三维散点图只是上下翻转。
- 箭头键：按住 Shift 键并按箭头箭，然后松开 Shift 键。
- Esc 键：按 Shift+Esc。三维散点图只是左右翻转。

除了可以自动旋转图之外，您还可以摇摆图。按 Shift+Ctrl，然后点击并拖动该图。该图将根据您拖动图的方向上下或左右摇摆。

要停止旋转或摇摆，请点击该图或按 Esc 键。

更改轴上的变量

每个轴上的变量由您在启动窗口中选择变量的顺序决定。例如，您选择的第一个变量将显示在 x 轴上。第二个变量显示在 y 轴上，第三个变量显示在 z 轴上。

创建三维散点图后，您可以更改指定给轴的变量，标绘一组不同的变量，或依次显示所有变量组合。

1. 要更改特定轴上的变量，请选择轴控件下拉菜单，从中选择不同的变量。
2. 要添加不同的变量，请点击轴控件下拉菜单，选择**其他**，再选择所需变量，然后点击**确定**。
3. 要依次显示所有变量组合，请点击**下一个轴系**按钮，直到显示您想要标绘的变量。

调整轴

您可以通过点击并拖动轴手动移动轴坐标或重新调整轴坐标的尺度。该选项在三维散点图上显示一组不同的坐标，还支持您更改坐标间距（或更改坐标尺度）。

您还可以通过双击轴并修改规格窗口中的设置来指定轴属性。

移动轴坐标

1. 悬停在轴中央上方。
2. 点击并拖动该轴。

修改坐标尺度

1. 悬停在轴末端上方。
2. 点击并拖动该轴。

重新精确调整轴尺度

1. 悬停在轴（而不是标签）中央的上方。
2. 双击该轴。
3. 更改最小和最大坐标值。

有关更新轴的详细信息，请参见《使用 JMP》。

为数据点指定颜色和标记

三维散点图中的每个点都对应相关数据表中的一行。要突出显示三维散点图上的点，需要为各点指定颜色和标记。颜色和标记随后会显示在三维散点图以及数据表中。

点击某个点时，将选定以下项：

- 三维散点图中的点
- 相关数据表中的对应行

- 任何其他已打开的图形中的点（若适用）

要选择一个点，请点击该点。

要选择若干点，请双击三维散点图并将光标拖过各点。将显示一个框，指示选定了哪些点。

要取消选择点，请双击三维散点图。

为选定的数据点指定颜色或标记：

1. 要为选定点指定颜色，请选择**行 > 着色**，然后选择所需颜色。
2. 要为选定点指定标记，请选择**行 > 标记**，然后选择所需标记。

在数据表中指定颜色和标记

您可以为数据表中的行指定颜色和标记。颜色和标记显示在数据表中的行号旁边以及三维散点图中。该选项区分每个变量的点，您可以将这些设置保存在数据表中。为特定数据点指定颜色和标记（如第 187 页的“为数据点指定颜色和标记”中所述）只会对当前已打开的图形突出显示这些数据点。

有关在数据表中指定颜色和标记的详细信息，请参见《使用 JMP》。有关更改标记的大小、质量或透明度的详细信息，请参见第 192 页的“三维散点图设置”。

“三维散点图”平台选项

“三维散点图”旁边的红色小三角菜单包含的选项用于定制显示，以及计算、旋转和保存主成分或旋转成分。

显示点 显示或隐藏图形上的数据点。

显示控制项 显示或隐藏三维散点图下方显示的来源和轴控件 （图 9.4）。

垂线 绘制从各点到您在启动窗口上选定的 *x* 和 *z* 变量所创建的平面的线条。

连接点 用线条连接各点。可以针对整体或分组数据连接各点。您还可以按特定变量对数据分组。

随机散布 在数据点之间显示小的间隔以便您可以更清楚地查看每个点。

正态等高线椭圆 绘制一个或多个正态等高线椭圆，即包含指定的一部分数据点的三维椭圆。您指定是需要针对所有数据还是各个分组的椭圆。还可以控制椭圆的大小和透明度。请参见第 190 页的“正态等高线椭圆”。

椭圆覆盖率 更改正态等高线椭圆的大小。键入介于 0 和 1 之间的值，值越大创建的椭圆就越大。实际值“0”和“1”不会生成椭圆，所以在试图使用这些值时将显示一条警告。

仅当您向三维散点图添加正态等高线椭圆后，才会显示该选项。

椭圆透明度 更改正态等高线椭圆的表面效果。值越大，椭圆越不透明。仅当您向三维散点图添加正态等高线椭圆后，才会显示该选项。

非参数密度等高线 绘制非参数密度等高线，其中大致包含指定比例的点。指定是需要针对所有数据还是各个分组的密度等高线。请参见第 190 页的 “非参数密度等高线”。

垂线粗细 更改垂线粗细。仅当您向三维散点图添加垂线后，才会显示该选项。

主成分 计算所有变量的主成分。这将更改图上的轴，以包含主成分得分。

默认显示双标图射线。通过从红色小三角菜单中选择**双标图射线**可以将其删除。有关主成分的详细信息，请参见 《多元方法》。

标准主成分 计算主成分 （与**主成分**选项一样），但已将主成分得分的尺度统一为具有单位方差。若未选定该选项，得分则具有与相应的特征值相等的方差。

通过使用标准化的主成分，变量与主成分得分之间的相关性等于特征向量中的值。这有助您快速评估变量的相对重要性。请参见 《多元方法》。

若需要 GH 双标图而不是 JK 双标图，可以选择该选项。 GH 双标图会尽量保持变量之间的关系； JK 双标图会尽量保持观测之间的关系。 GH 双标图所显示的点间距离意义不大，但 GH 双标图射线的角度可更好地测量相关性。

旋转成分 指定要旋转的因子数以及旋转方法。旋转成分可使因子方向更符合原始变量，这样可以更好地解释因子。请参见 《多元方法》。

双标图射线 显示或隐藏与主成分对应的双标图射线。您必须已经选定主成分、标准主成分或旋转成分，该选项才会显示。

显示射线标签 显示或隐藏双标图射线的标签。您必须已经选定 “双标图射线”，该选项才会显示。

删除主成分 从三维散点图报表中删除主成分、标准化主成分和旋转成分。三维散点图恢复为选定主成分之前的原始显示状态。但是，该选项不会从数据表中删除任何已保存的主成分。

仅当您向三维散点图添加主成分、标准主成分或旋转成分后，才会显示该选项。

保存主成分 将指定数量的当前主成分得分另存为当前数据表中的新列。这些列还包括用于主成分的公式。对于成分列表中的 *n* 个变量，将创建 *n* 个主成分列，并分别命名为**主成分** 1、**主成分** 2、 ... **主成分** n。

仅当您向三维散点图添加主成分、标准主成分或旋转成分后，才会显示该选项。

保存旋转成分 将所有旋转成分得分另存为当前数据表中的列。这些列还包括所用的公式。若请求了 *n* 个旋转成分，将创建 *n* 个旋转成分列，并分别命名为**旋转成分** 1、**旋转成分** 2、 ... **旋转成分** n。

仅当您向三维散点图添加旋转成分后，才会显示该选项。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

正态等高线椭圆

正态等高线椭圆是一个包含指定的一部分数据点的三维椭圆。该椭圆基于数据点的多元正态分布拟合的等高线计算得出，是图中变量的均值、标准差和相关性的函数。有关多元正态分布的详细信息，请参见《多元方法》。

添加椭圆时，提供两个格式选项：

- **覆盖率**用于更改椭圆所覆盖的数据点部分。值越大，椭圆越大。
- **透明度**用于将椭圆表面效果从透明改为不透明。值越大，椭圆越不透明。

添加椭圆之后，覆盖率和透明度这两个选项还会显示在红色小三角菜单中。

向三维散点图添加正态等高线椭圆时，可以指定是需要针对所有数据还是特定数据分组的椭圆。用于每批分组数据的椭圆会进行颜色编码，以便区分不同的分组。

通过从红色小三角菜单中选择和取消选择**正态等高线椭圆**，可以相应显示和删除正态等高线椭圆。

相关信息

- 第 194 页的“未分组正态等高线椭圆的示例”
- 第 195 页的“已分组正态等高线椭圆的示例”

非参数密度等高线

非参数密度等高线显示大致包含指定比例的点的等高线。通过添加非参数密度等高线，可以在散点图中密布数千个数据点时查看点密度中的模式。

该功能尤其适用于三维散点图中有许多点的情况；此时的等高线可能过于密集，让您无法看清结构。在这种情况下，您需要删除点，以便仅显示等高线。请参见第 191 页的“优化密集的非参数密度等高线”。

向三维散点图添加非参数密度等高线时，可以指定是需要针对所有数据还是特定数据分组的等高线。用于每批分组数据的等高线会进行颜色编码，以便区分不同的分组。

通过从红色小三角菜单中选择和取消选择**非参数密度等高线**，可以相应显示和删除非参数密度等高线。

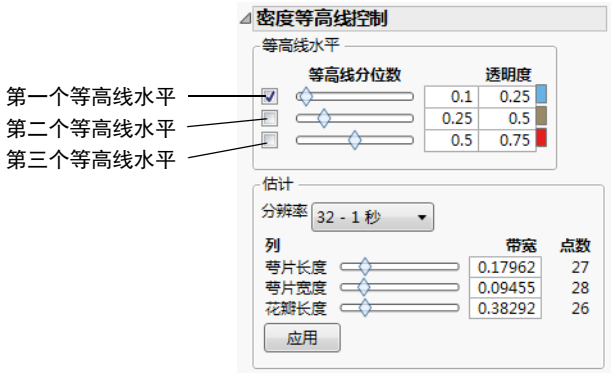
相关信息

- 第 196 页的“已分组非参数密度等高线的示例”

密度等高线控制

“密度等高线控制”选项显示在三维散点图之下。这些选项支持您选择其他等高线并更改每条等高线的格式。

图 9.5 “密度等高线控制”窗口



等高线分位数 控制所显示的等高线，并支持您定制等高线格式。

- **密度水平**表示数据点的数量和密度。随着等高线的值由小变大，等高线覆盖的数据量变小，但覆盖密度增大。等高线值为 0.9 表示总体中最密集的 10% 部分，其中的各点最为接近。点击并拖动“等高线分位数”下方的滑块，或在滑块旁边输入一个值。
- **透明度**用于更改密度等高线的表面效果。值越大，等高线越不透明。在该框中输入一个值。
- **颜色**用于更改等高线的颜色。点击着色的框并选择不同的颜色。（该选项仅对未分组的密度等高线显示。）

针对这些设置的更改会立即生效。

分辨率 更改等高线的分辨率。较高的分辨率可绘制更为精细的等高线，但显示所需的时间也更长。

列带宽 更改拟合密度的平滑度。较高的带宽会导致更平滑的拟合密度。

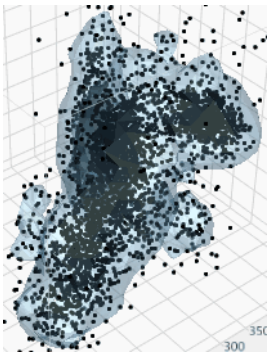
为每个变量键入新的带宽，或点击并拖动各个滑块。点击**应用**可显示所做更改。

优化密集的非参数密度等高线

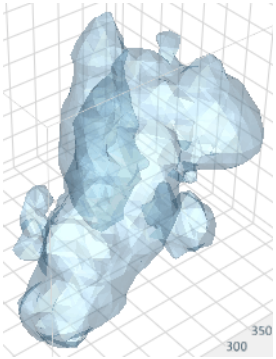
若三维散点图中有许多点，等高线可能过于密集，让您无法看清结构。在这种情况下，您需要删除点，以便仅显示等高线。

要从三维散点图中删除点，请从红色小三角菜单中选择**显示点**。您可以通过更改等高线的大小、颜色和透明度进一步优化等高线。请参见第 188 页的““三维散点图”平台选项”。

图 9.6 优化密集的非参数密度等高线的示例



包含密度等高线和数据点的三维散点图



包含密度等高线但不含数据点的三维散点图

弹出菜单

右击三维散点图可找到以下选项。

显示图例 显示和隐藏三维散点图的颜色图例。

> **注意：**该选项需要**颜色**角色。

重置 使散点图的方向恢复其初始状态。

设置 提供用来更改三维散点图外观的选项。请参见第 192 页的 “三维散点图设置”。

隐藏光源边框 显示和隐藏用于显示光源的边框。光源可突出显示三维散点图的不同部分。

右击某一光源可启用或禁用该光源并且更改颜色。

墙壁颜色 更改三维散点图的颜色。

背景颜色 更改三维散点图周围的颜色。

行 您可以针对与相关数据表中的行对应的数据点执行着色、标记、排除、隐藏和添加标签操作。选择该选项之前必须先选择数据点。请参见《使用 JMP》。

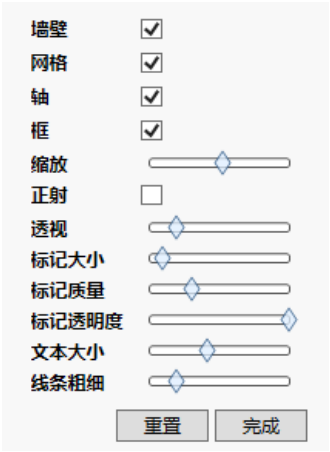
使用硬件加速 为支持加速的机器启用或禁用硬件加速。该选项可能会加快散点图的显示速度。若不能加速，请尝试更新图形驱动程序。

显示轨迹球 显示和隐藏环绕三维散点图的球体。该选项可帮助您直观演示散点图的旋转。选择是想要轨迹球始终显示、仅在拖动散点图时显示，还是从不显示。

三维散点图设置

要定制标记大小、文本大小和网格线之类的属性，可以右击三维散点图并选择**设置**。随即显示“设置”窗口。修改设置的过程中，三维散点图上会显示预览。

图 9.7 三维散点图的 “设置”窗口



请注意以下几点：

- 向左移动滑块可以减小选定的属性，向右移动滑块可以增大选定的属性。
- 要围绕散点图移动“设置”窗口，可以点击并拖动该窗口的顶部。

提供了以下选项：

重置 重置默认设置。

完成 关闭该窗口。

墙壁 添加或删除三维散点图的墙壁。若无墙壁，将显示三维散点图的背景颜色。

网格 显示或隐藏坐标线。

轴 显示或隐藏出现在每个轴上方的变量名。

框 显示或隐藏框。若无框，三维散点图显示为一个开放图形。

缩放 放大或缩小三维散点图。

正射 将散点图视图从三维改为正射投影。在正射视图中，散点图的墙壁不会聚合到一个消失点。这意味着您可以比较近距离和远距离，看清数据点之间的结构。

注意：若禁用正射视图并将透视值减至最低，则散点图的墙壁不会聚合。这与您启用正射视图时的效果相同。

透视 增大或减小透视值。较大的值会产生不自然且看起来失真的大视图。在这种情况下，您需要调整散点图窗口的大小以显示全图。

标记大小 增大或缩小数据点标记的大小。

标记质量 提高和降低数据标记质量。例如，提高标记质量时，某些标记会具有不透明的中心。其他符号标记的格式设置为粗体。放大可查看质量的这些变化。

标记透明度 提高或降低数据标记的透明度。

文本大小 增大或缩小文本大小。

线条粗细 更改坐标和轴线的粗细。

“三维散点图”平台的更多示例

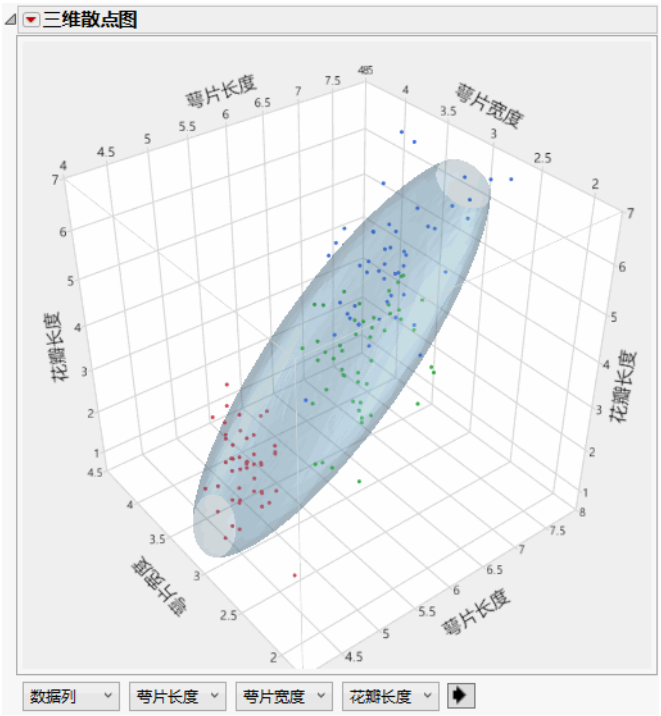
- “未分组正态等高线椭圆的示例”
- “已分组正态等高线椭圆的示例”
- “已分组非参数密度等高线的示例”

未分组正态等高线椭圆的示例

该示例显示如何为 75% 以上的数据点添加正态等高线椭圆。该椭圆的透明度为 25%。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**图形** > **三维散点图**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，列**。
4. 点击**确定**。
5. 点击“三维散点图”红色小三角并选择**正态等高线椭圆**。请注意**未分组**已被选定。
6. 在“覆盖率”旁边键入 0.75。
7. 在“透明度”旁边键入 0.25。
8. 点击**确定**。

图 9.8 未分组正态等高线椭圆的示例

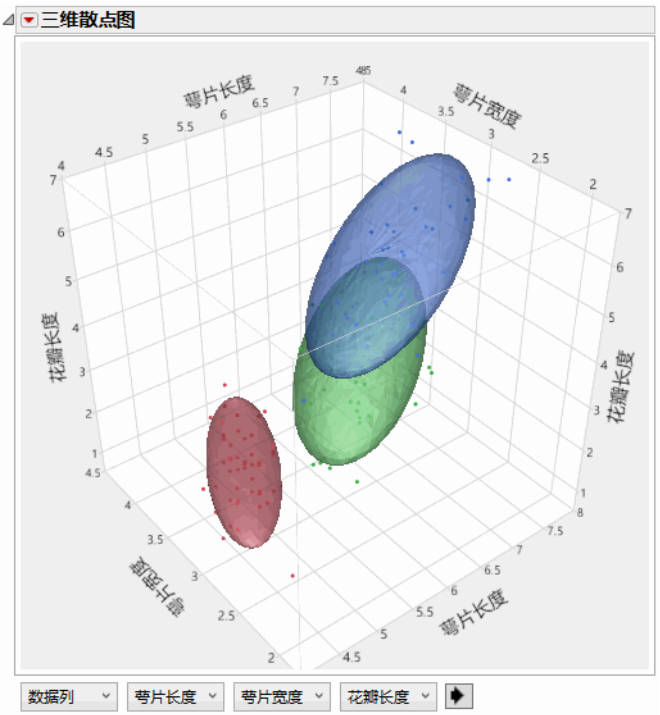


已分组正态等高线椭圆的示例

该示例显示如何按种类对测量值分组，并使用正态等高线椭圆设置每个分组的格式。这些椭圆覆盖 75% 的数据点且透明度为 50%。等高线基于种类进行颜色编码 （使用 JMP 默认颜色）。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**图形** > **三维散点图**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，列**。
4. 点击**确定**。
5. 点击 “三维散点图”红色小三角并选择**正态等高线椭圆**。
6. 选择**按列分组**。
7. 选择**种类**。
8. 在 “覆盖率”旁边键入 0.75。
9. 在 “透明度”旁边键入 0.5。
10. 点击**确定**。

图 9.9 已分组正态等高线椭圆的示例

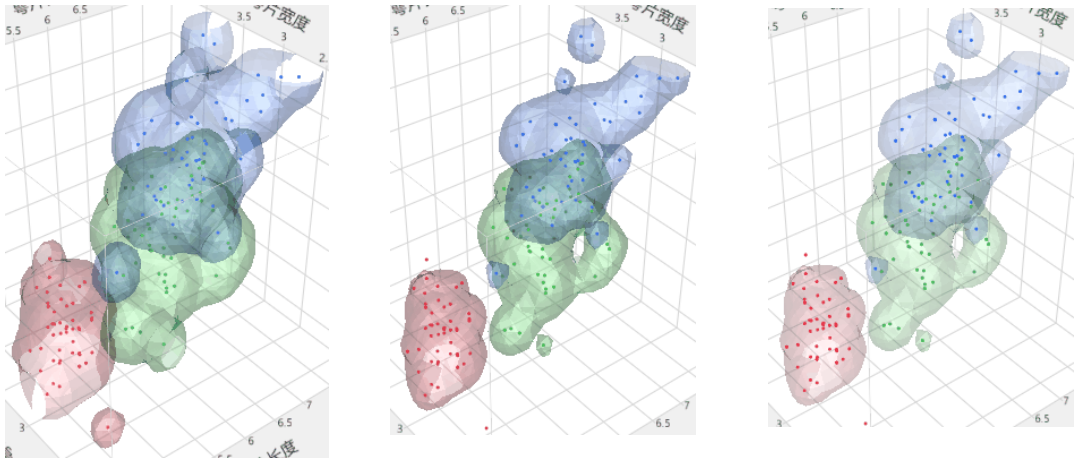


已分组非参数密度等高线的示例

该示例显示如何对数据点分组并设置各个非参数密度等高线的格式。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**图形** > **三维散点图**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，列**。
4. 点击**确定**。
5. 点击 “三维散点图”红色小三角并选择**非参数密度等高线**。
6. 选择**按列分组**。
7. 选择**种类**并点击**确定**。将为这三个种类显示颜色各不相同的等高线。
8. 在第一个 “等高线分位数”框中键入 0.25。 25% 的数据点显示在等高线曲面之外，从而生成较小的等高线。
9. 在第一个 “透明度”框中键入 0.15。等高线 15% 不透明。

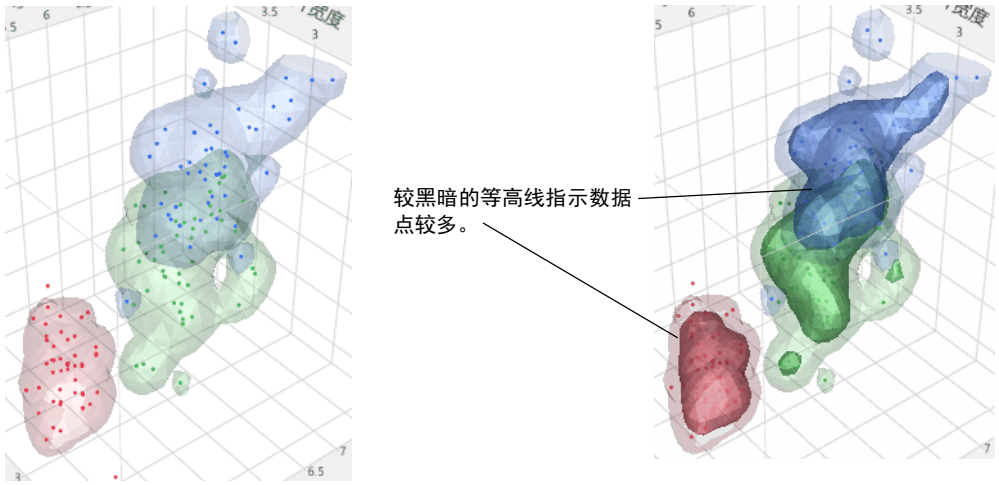
图 9.10 更改非参数密度等高线的透明度和密度



原始等高线　　密度减小的等高线　　透明度降低的等高线

10. 选中第二个复选框。等高线分位数相同 (.25)，因此新的等高线与第一组等高线重叠。
11. 在第二个“等高线分位数”框中键入 0.5。50% 的数据点显示在等高线曲面之外。第二组等高线显示在第一组之内，从而进一步演示了数据点的密度。

图 9.11 添加另一个非参数密度等高线



一组等高线水平　　两组等高线水平

您现在可以设置第二组等高线水平的格式并启用第三组等高线水平。

用于设置已分组和未分组的非参数密度等高线格式的选项基本相似。唯一的不同是您不能更改每个**已分组**非参数密度等高线的颜色。请参见第 188 页的““三维散点图”平台选项” 了解相关选项。

第 10 章

等高线图

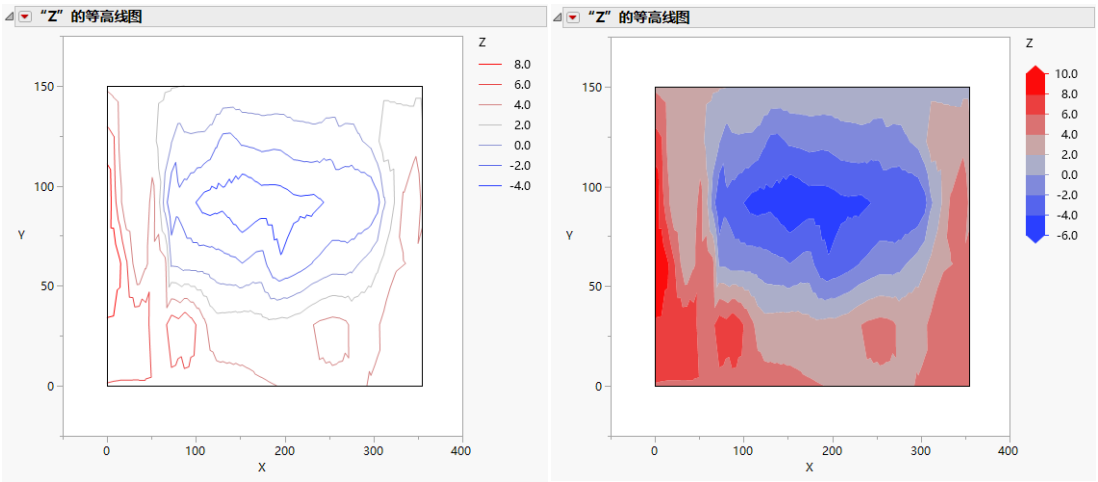
在二维中查看多维关系

图形菜单中的**等高线图**命令用于在直角坐标系中构造某个响应的等高线。等高线图在二维中显示三维曲面。等高线描绘第三个维中的变化。

以下是“等高线”平台中的一些可用选项：

- 指定等高线水平的个数
- 选择标绘等高线还是填充等高线
- 显示或隐藏数据点
- 为等高线添加响应值标签
- 定义和使用定制颜色方案

图 10.1 等高线图的示例



目录

等高线图的示例 201
使用“等高线”平台的示例 201
图形生成器中等高线图的示例 202
启动“等高线图”平台 202
等高线图 204
“等高线图”平台选项 204
填充区域 205
等高线规格 206
等高线图保存选项 210
使用公式指定等高线 211
等高线图的更多示例 211

等高线图的示例

要创建等高线图，您可以使用 “等高线”平台或图形生成器。本节提供这两者的示例。

- 要查看使用“等高线”平台的示例，请参见第 201 页的“使用“等高线”平台的示例”。
- 要查看使用图形生成器的示例，请参见第 202 页的“图形生成器中等高线图的示例”。

使用“等高线”平台的示例

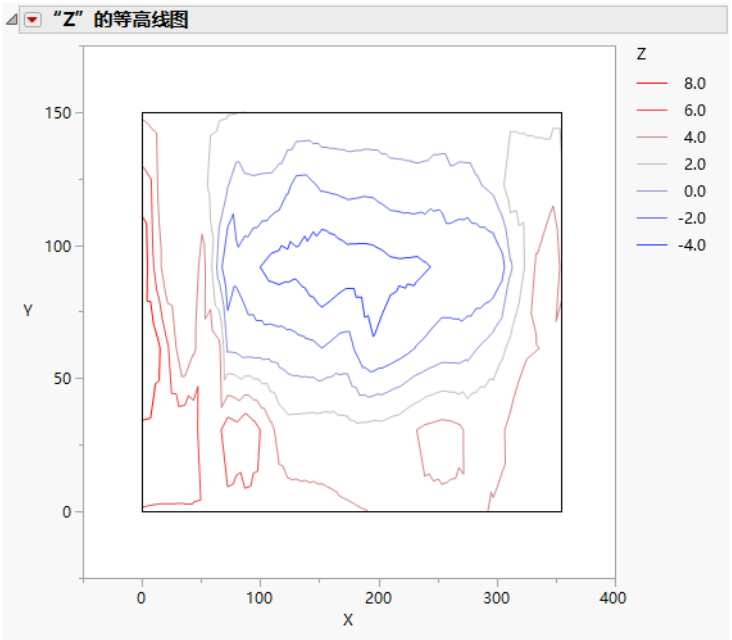
要创建等高线图，您需要分别用于 x 和 y 轴的两个变量，另外至少还要有一个用于等高线的变量。您也可以使用多个 y 变量。下面的示例使用 Little Pond.jmp 样本数据表。 X 和 Y 是池塘的坐标。 Z 是深度。

1. 选择**帮助** > **样本数据库**，然后打开 Little Pond.jmp。
2. 选择**图形** > **等高线图**。
3. 选择 X 和 Y 坐标并点击 X。
4. 选择深度 Z 并点击 Y。

注意：在等高线图中， X1 和 X2 角色分别用于 X 和 Y 轴。

5. 点击**确定**。

图 10.2 带图例的等高线图示例



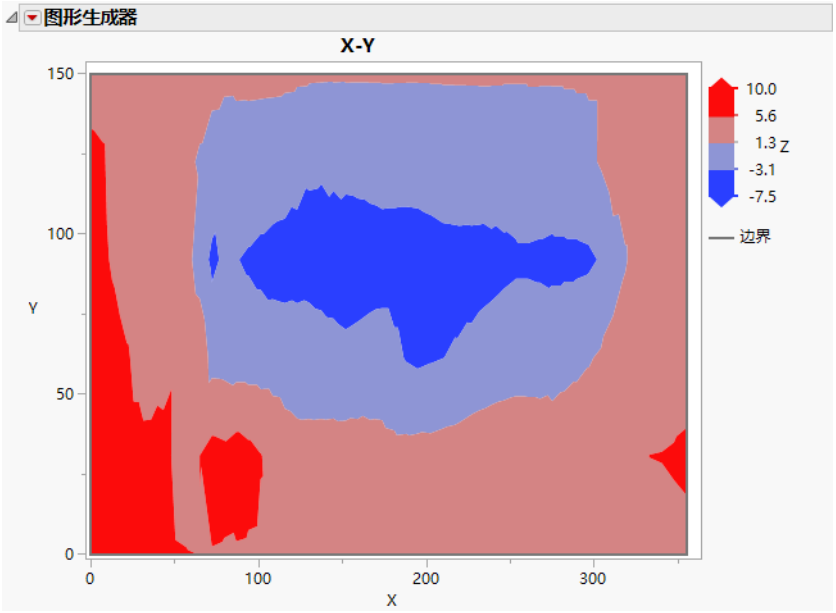
x 和 y 轴是坐标轴，等高线由深度变量定义。该等高线图实际上是一个显示深度的池塘地图。要更清楚地查看等高线，请点击 “ ‘Z’ 的等高线图” 旁边的红色小三角并选择**填充区域**。

图形生成器中等高线图的示例

您可以使用图形生成器创建与图 10.2 中所示相同的等高线图。

1. 选择**帮助** > **样本数据库**，然后打开 Little Pond.jmp。
2. 选择**图形** > **图形生成器**。
3. 点击 X 坐标并将其拖至 X 区域。
4. 点击 Y 坐标并将其拖至 Y 区域。
5. 点击深度 Z 并将其拖至**颜色**区域。
6. 点击 “等高线” 图标。

图 10.3 图形生成器中的等高线图



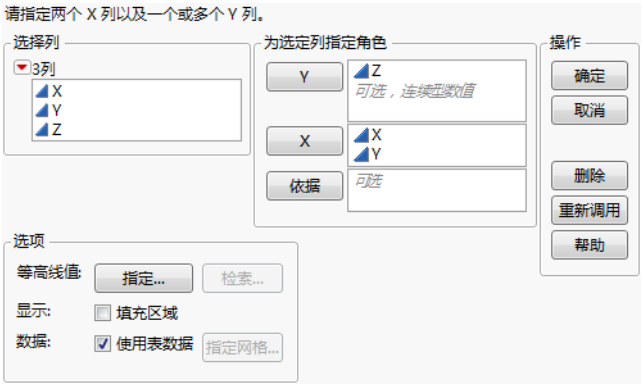
请注意，从 “等高线” 平台中生成的图与使用图形生成器生成的图看上去略有不同。

启动 “等高线图” 平台

通过选择**图形** > **等高线图**可启动 “等高线图” 平台。

默认情况下，图中使用的等高线水平是从数据中计算的值。您可以先在启动窗口中自行指定水平数和水平增量，然后再创建图形；也可以在创建图形后在“等高线图”的红色小三角菜单中进行这些指定。您可以使用列公式来计算等高线变量值。

图 10.4 “等高线图”启动窗口



Y 指定给 Y 角色的列用作确定图中等高线的变量。您必须至少指定一个变量，可以指定多个变量。

您还可以将包含公式的列指定给该角色。若这样指定，该公式应恰好是两个变量的函数。这些变量应该是在启动窗口中输入的 x 变量。

X 指定给 X 角色的列用作 x 和 y 轴的变量。您必须为 X 恰好指定两列。

依据 该选项为“依据”变量的每个水平生成单独的图形。若指定了两个“依据”变量，将为这两个“依据”变量的水平的每种可能组合生成单独的图形。

选项

等高线值 自行指定水平数和水平增量。请参见第 206 页的“等高线规格”。

填充区域 使用等高线颜色填充等高线之间的区域。

“使用表数据”和“指定网格” 很多时候，您都是为记录的响应值表构造等高线图。这种情况下，将选定**使用表数据**，且**指定网格**按钮变为不可用。

但是，若列中包含公式，而您将该列指定为响应 (Y)，则**指定网格**按钮将变为可用。点击**指定网格**时，您可以采用任意方式定义等高线网格，而与现有数据表中的行无关。该功能还适用于包含由公式定义的一个或多个列但不含任何行的表模板。请参见第 211 页的“使用公式指定等高线”。

点击**确定**后将出现等高线图。请参见第 204 页的“等高线图”。

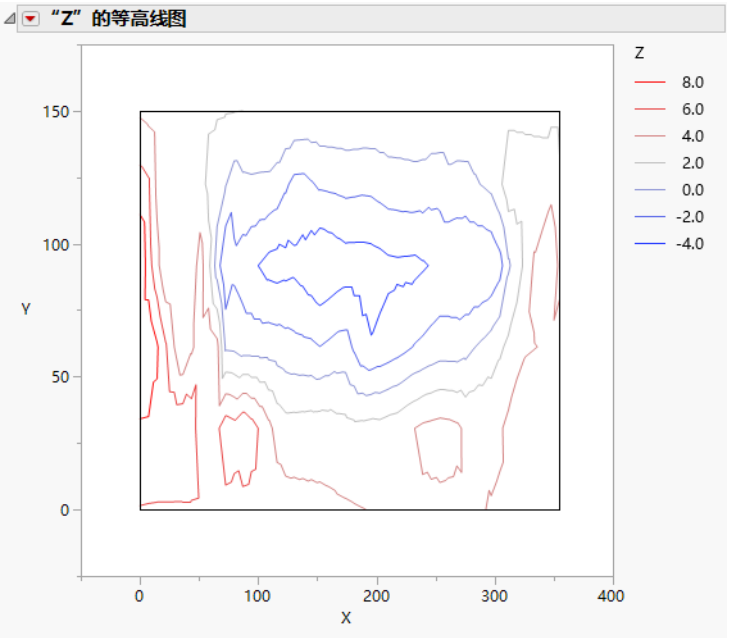
等高线图

按照第 201 页的“等高线图的示例”中的说明生成图 10.5 所示的图。

等高线图的图例显示 Y 变量的各个标记和颜色。通过以下两种方式之一拖放变量，以此替换图中变量：将一个轴上的变量拖放到另一个轴上来交换现有变量；或者，点击相关数据表的列面板中的某个变量并将其拖到轴上。

有关报表附加选项的信息，请参见第 204 页的“‘等高线图’平台选项”。

图 10.5 “等高线图”报表



“等高线图”平台选项

使用“等高线图”旁边的红色小三角菜单中的选项，您可以定制等高线图的外观，并保存有关其构造的信息。

显示数据点 显示或隐藏 (x, y) 点。默认隐藏这些点。

显示缺失数据点 显示或隐藏具有缺失 y 值的点。仅当选定**显示数据点**时才可用。

显示等高线 显示或隐藏等高线或填充色。默认显示等高线。

显示边界线 显示或隐藏整个等高线区域的边界线。默认显示边界线。

显示控制面板 显示或隐藏 Alpha 滑块，该滑块允许您更改 Alpha 形状过滤器。

变换 若等高线图包括**颜色**角色，则将启用**变换**选项。请参见第 211 页的 “等高线图的更多示例”。

无 将使用 Delaunay 三角测量计算三角测量，而不对坐标尺度进行任何调整。计算 Delaunay 三角测量以最大化三角测量中三角形的最小角度。默认选定该值。

标准化范围 计算三角测量之前，将 X1/X2 值的尺度统一调整为 [0,1]。若 X1/X2 限值是不同的，则这是一个不统一的尺度。该选项可能更适合 X1/X2 的单位极为不同的情况。

填充区域 用纯色填充等高线之间的区域。启动窗口中也提供与此相同的选项。若在启动窗口中未选定该选项，您可以在填充区域前查看线条轮廓。请参见第 205 页的 “填充区域”。

添加等高线标签 显示或隐藏等高线的标签 （z 值）。

适应窗口大小 指定报表的拉伸行为。

更改等高线 自行设置水平数和水平增量。请参见第 206 页的 “等高线规格”。

保存 该菜单提供的选项用于将有关等高线、三角测量和网格坐标的信息保存至数据表。请参见第 210 页的 “等高线图保存选项”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

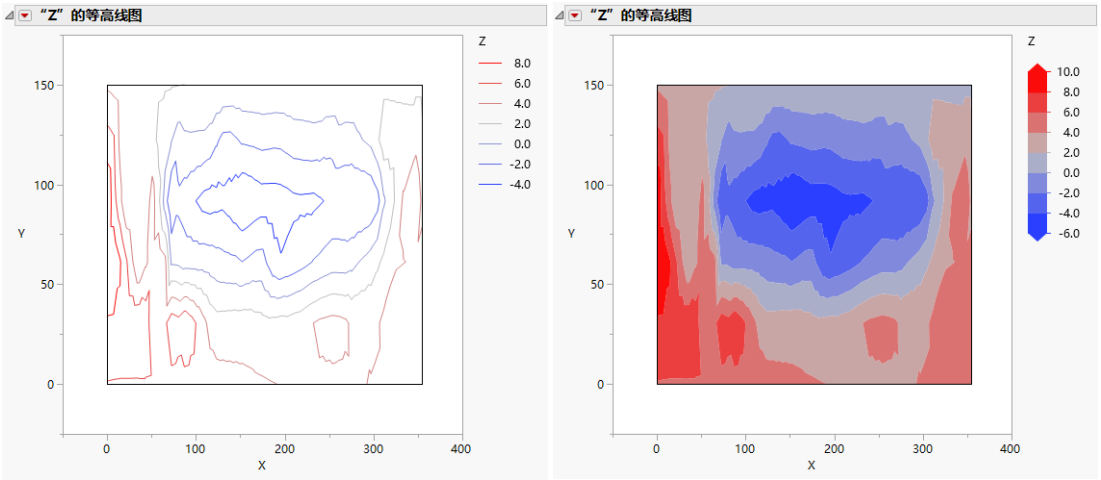
保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

填充区域

若选择**填充区域**，将使用等高线颜色来填充等高线之间的区域。该选项在启动窗口和 “等高线图”的红色小三角菜单中提供。图 10.6 的左侧显示包含等高线的图，右侧显示填充了等高线区域的图。

图 10.6 比较等高线和区域填充

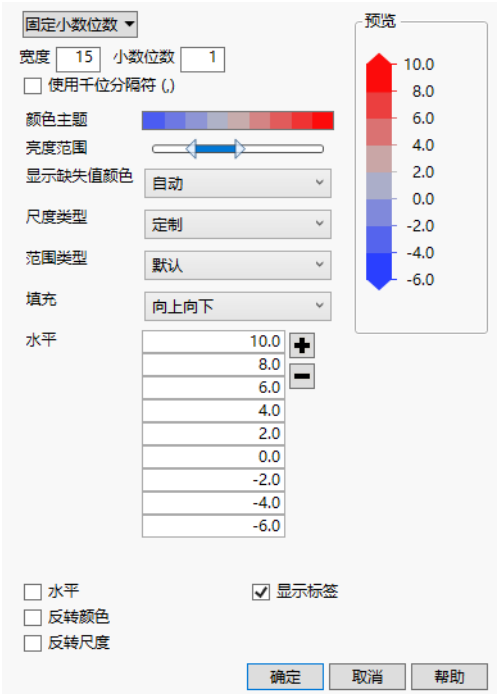


依据颜色梯度从低值到高值填充各区域。在填充的等高线图中还为最后一条最高的等高线之上的水平添加了附加颜色。

等高线规格

若未在启动窗口中选择相关选项，默认图将在 Y 变量的范围之内均等划分等高线水平的间距。您可以通过在**文件 > 首选项 > 图形**中选择“连续颜色主题”来指定颜色主题。可以通过在“等高线图”图例中右击等高线颜色来定制各个等高线的颜色。通过右击“等高线图”图例并选择“梯度”打开“梯度设置”，可以定制整个等高线颜色梯度。

图 10.7 “梯度设置”窗口



要设置图例中标签的格式，请使用格式菜单、“宽度”框和“小数位数”框，以及该窗口左上角的“使用千位分隔符 (,)”复选框。

颜色主题 更改颜色主题或定义定制颜色主题。有关颜色选项的详细信息，请参见《使用 JMP》。

亮度范围 设置梯度颜色的强度范围。

标签数 （若“尺度类型”为“定制”，则不可用。）指定图例的标签数。值为 0 则提供默认的标签数。

显示缺失值颜色 指定是否在图例中显示缺失值颜色。选择“自动”仅当有缺失值时才显示颜色，选择“关闭”从不显示颜色，选择“打开”则始终显示颜色。

尺度类型 设置梯度颜色的尺度。尺度类型的说明位于“最大值”设置下方。

线性 尺度在“最小值”与“中心值”之间以及“中心值”与“最大值”之间是分段线性的。这是默认尺度。

分位数 尺度对于梯度变量的分位数是分段线性的。

标准差 梯度变量范围细分为与均值的标准差剩余偏差。尺度在这些偏移设置之间是分段线性的。偏移数由“标签数”确定。

对数 尺度对于梯度变量的对数是线性的。

对数剩余偏差 对于梯度变量以 10 为底的对数的剩余偏差，尺度是线性的。

定制 尺度标签由用户在“水平”设置中指定。

范围类型 设置图例值的范围。

默认 除非您指定了“最小值”和“最大值”，否则最小值和最大值包含数据范围并且选定适当的取修约值。

精确数据范围 最小值和最大值是数据的精确最小值和最大值，或是您指定为“最小值”和“最大值”的值。

中间 90% 最小值和最大值分别是第 5 分位数和第 95 分位数，或是您指定为“最小值”和“最大值”的值。该选项可防止离群值。

填充 确定如何在等高线中指定范围的端点。

组间 等高线在端点处停止，端点以外的任何数据都不会显示。

之上 高于该范围的值包含在顶部等高线水平中。

之下 低于该范围的值包含在底部等高线水平中。

向上向下 两端的值包含在其各自的等高线水平中。

水平 （仅在指定 “定制尺度类型”时才可用。）支持您指定图例的标签数和标签值。

水平 将图例的方向设置为水平。

反转颜色 在颜色方案中反转颜色。

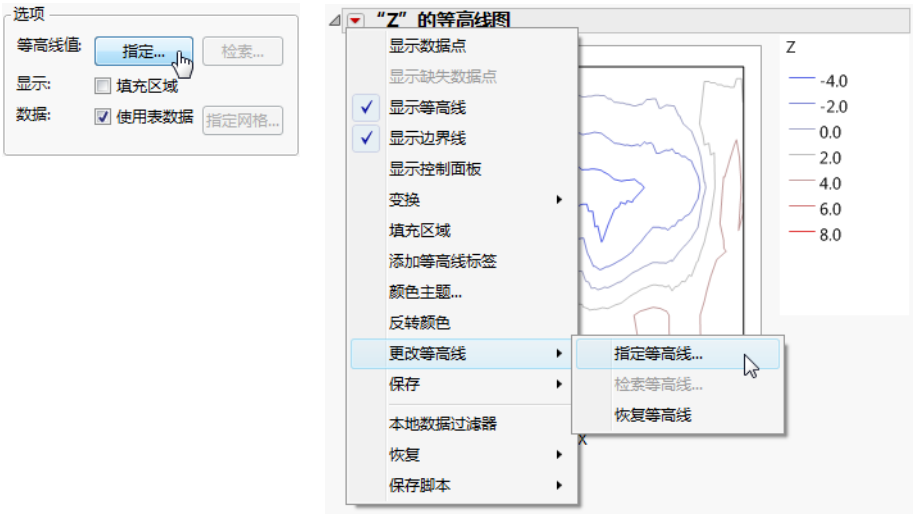
反转尺度 反转颜色方案的尺度。

显示标签 显示或隐藏图例标签。

指定

您可以在启动窗口中（**指定**按钮）或通过报表窗口中的“等高线图”的红色小三角菜单（**指定等高线**选项）指定等高线水平。选择该选项可显示“等高线规格”窗口。

图 10.8 等高线规格示例：启动窗口 （左侧）和菜单 （右侧）

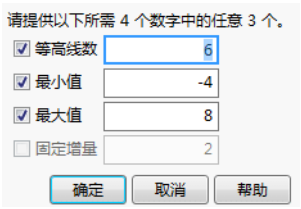


您可以使用 “等高线规格”窗口执行以下操作：

- 更改等高线的个数
- 指定最小值和最大值以定义要在图中使用的响应的范围
- 更改等高线值之间的增量

您需要提供以上四个值中的任意三个，系统将为您计算余下的值。点击复选框可取消选择这些数字之一，并自动选中余下的复选框。

图 10.9 “等高线规格”窗口



依据图中的水平数自动指定和确定颜色。出现图后，您可以右击（在 macOS 上按 Control 键并点击）该图图例中的任意等高线，然后从 JMP 调色板中选择以更改该等高线颜色。

检索

该选项在启动窗口（**检索**按钮）和“等高线图”的红色小三角菜单中（**检索等高线**选项）提供。

注意：除非在包含等高线标绘值的表**之外**还有一个打开的数据表，否则该按钮和菜单选项均不可用。当您点击**检索**或选择**检索等高线**时，将显示包含已打开数据表的列表的窗口。

使用该选项，您可以从打开的 JMP 数据表中检索以下项：

- 等高线的个数
- 每个水平的精确值
- 每个水平的颜色

从打开的数据表的列表中，选择包含等高线水平的数据表。

对于水平值规格，“等高线图”平台会查找与您在启动窗口中指定的响应列同名的数值列。数据表中的行数定义水平数。

若存在包含颜色信息的行状态列，这些颜色将用于等高线水平。若不存在，则使用默认平台颜色。

恢复等高线

该选项仅显示在“等高线图”的红色小三角菜单上。

若您自行指定了等高线，选择该选项可将您的等高线图恢复为默认等高线。

等高线图保存选项

该菜单提供的选项用于将有关等高线、三角测量和网格坐标的信息保存至数据表。

保存等高线 创建新的 JMP 数据表，其中的列包含以下内容：

- “等高线”平台为每个等高线生成的 x 和 y 坐标值
- 为每个坐标集计算的响应
- 每个坐标集的曲线编号

该表中的观测数取决于您指定的等高线数量。您可以使用坐标和响应值通过其他 JMP 平台来查看数据。例如，您可以使用 “三维散点图”平台获取池塘的三维视图。

生成网格 显示一个窗口，提示您输入所需的网格大小。当您点击**确定**后，“等高线”平台将使用以下项创建新的 JMP 数据表：

- 您请求的网格坐标的个数
- 通过线性插值计算得出的网格点的等高线值

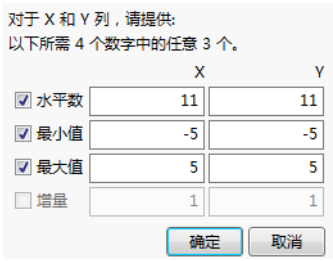
保存三角测量 创建新的 JMP 数据表，其中列出用于构造等高线的每个三角的坐标。默认情况下， JMP 使用 Delaunay 三角测量来连接最近的数据点以构成三角形。计算三角形的结果集，以便没有其他数据点位于三角形外接圆（即，穿过三角形三个顶点的圆形）的内部。要将三角测量改为标准化尺度，请选择**变换 > 标准化范围**。

使用公式指定等高线

很多时候，您都是为记录的响应值表（如 Little Pond 数据表）构造等高线图。在这种情况下，启动窗口中的**使用表数据**处于选定状态，而**指定网格**按钮不可用。但是，若列中包含公式，而您将该列指定为响应 (Y)，则**指定网格**按钮将变为可用。

点击**指定网格**后，将出现图 10.10 所示的窗口。

图 10.10 用于公式列的等高线规格的示例



您可以完成“指定网格”窗口并采用任意方式定义等高线网格，而与现有数据表中的行无关。该功能还适用于包含由公式定义的一个或多个列但不含任何行的表模板。

等高线图的更多示例

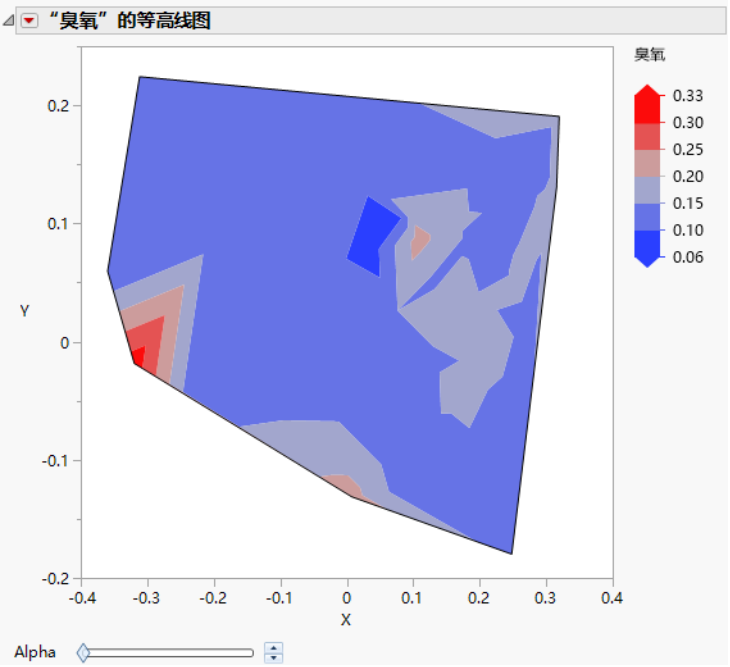
该示例演示如何创建三角测量数据表、变换三角测量以使用 Delaunay 三角形，以及过滤三角形的 Alpha 形状。

1. 选择**帮助 > 样本数据库**，然后打开 Cities.jmp。
2. 选择**图形 > 等高线图**。
3. 选择**臭氧**并点击 Y。
4. 选择 X 和 Y 并点击 X。
5. 选择**填充区域**。
6. 点击**确定**。
7. 点击 “‘臭氧’的等高线图”旁边的红色小三角并选择以下选项：
   a. **变换 > 标准化范围**

      不是使用 Delaunay 三角测量，而是将计算三角测量的方法更改为在 X 和 Y 中采用标准化尺度 ([0,1])。
   b. **显示控制面板**

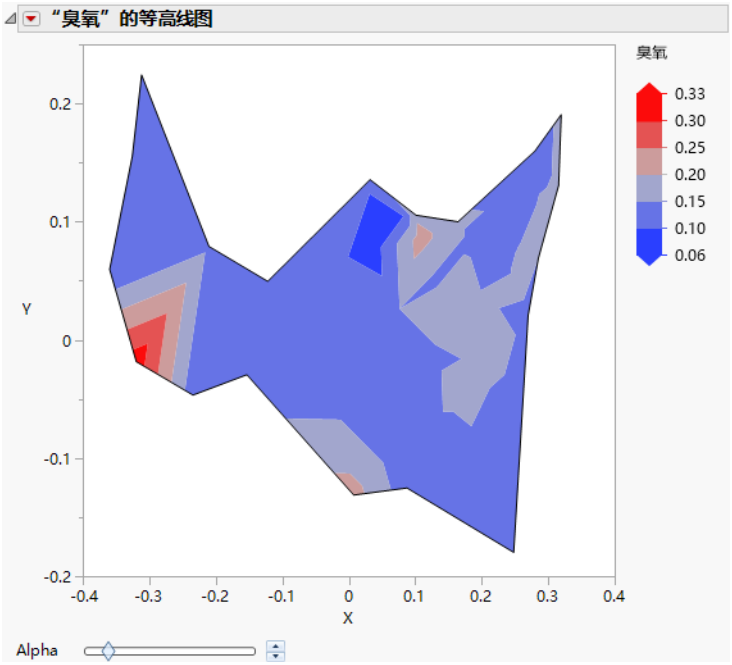
      Alpha 滑块随即显示。

图 10.11 “臭氧”的等高线图



8. 点击并将 Alpha 滑块移至右侧。

图 10.12 Alpha 形状过滤器



使用 Alpha 滑块过滤出较大的 Delaunay 三角测量区域。

第 11 章

三元图

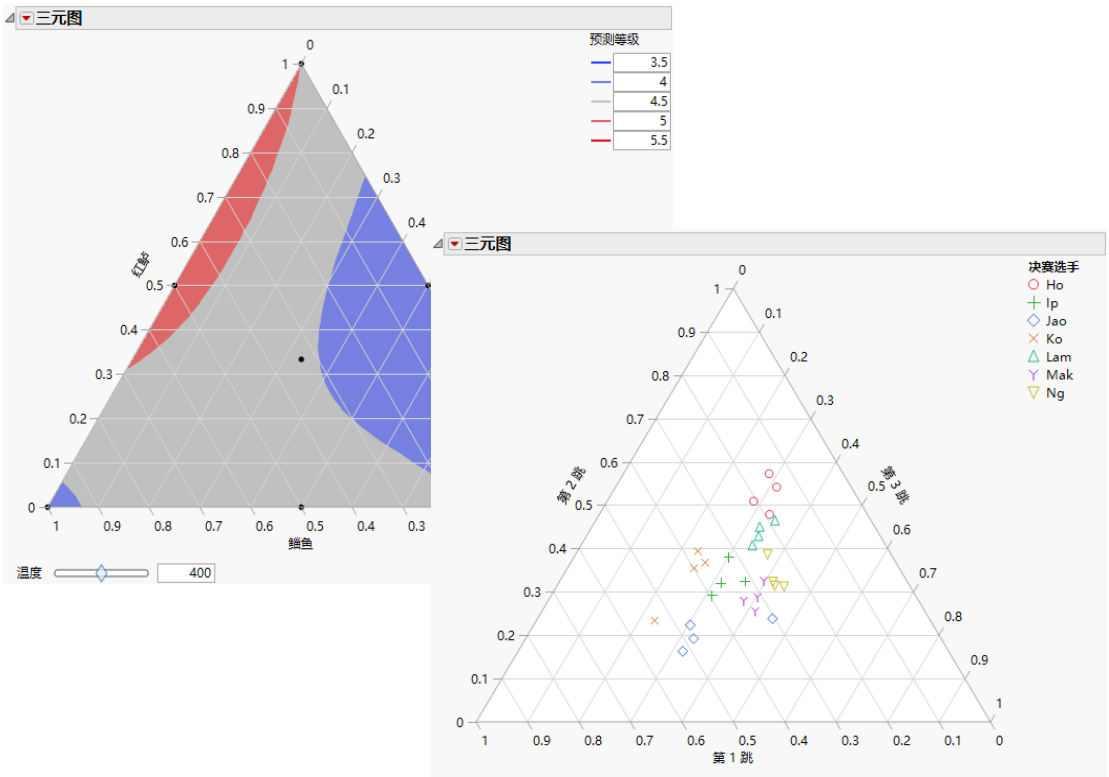
查看表示构成或混合数据的图

图形菜单中的三元图命令生成三轴图。

三元图是一种显示由三部分组成的数据的分布和变异性的方法。（例如，土壤中的沙子、淤泥和粘土的比例或实验性药物中三种化学试剂的比例。）您可以使用以比例表示的数据或使用绝对测量值。

三元显示是一个三角形，其边从 0 到 1 划分尺度。每边表示三种成分之一。绘制一个点，使得从该点出发与三角形的每条边垂直的线在该点所对应的成分值处交会。

图 11.1 三元图的示例



目录

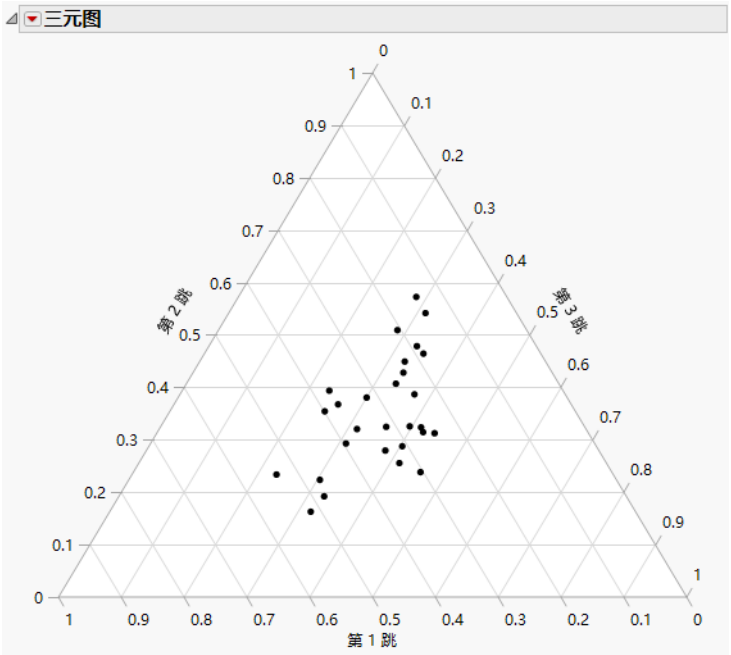
三元图的示例 217
启动“三元图”平台 219
三元图 220
    混料和约束 221
“三元图”平台选项 221
“三元图”平台的更多示例 221
    使用混料约束的示例 222
    使用等高线函数的示例 222

三元图的示例

该示例使用 Pogo Jumps.jmp 样本数据表。该数据改编自 Aitchison (1986)，显示 1985 年香港弹簧跳远锦标赛中 7 位决赛选手的弹簧跳远成绩。单次弹簧跳远成绩为三次连续弹跳中总的跳远距离，被称为**第 1 跳**、**第 2 跳**和**第 3 跳**。

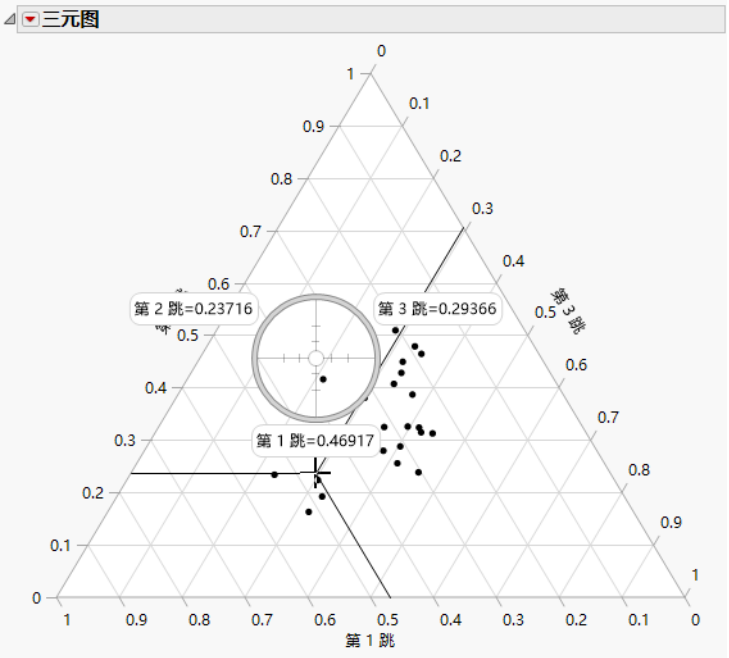
1. 选择**帮助** > **样本数据库**，然后打开 Pogo Jumps.jmp。
2. 从**图形**菜单中，选择**三元图**。
3. 选择**第 1 跳**、**第 2 跳**和**第 3 跳**并点击 **X，绘图**。
4. 点击**确定**。

图 11.2 三元图的示例



使用十字准线工具确定图中各点的精确坐标。

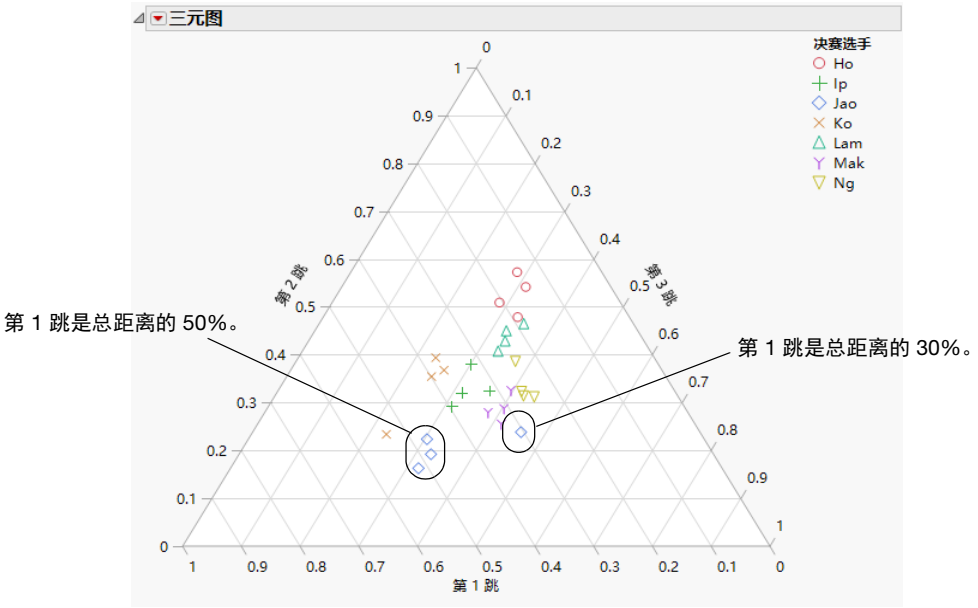
图 11.3 使用十字准线工具



为了更好地了解三次弹跳对总距离的影响，为每位选手的点指定不同的颜色和标记。

1. 右击该图并选择**行图例**。
2. 在列列表框中选择**决赛选手**。
   颜色将会自动设置为 JMP 默认。
3. 从**标记**菜单中选择**标准**。
4. 点击**确定**。

图 11.4 根据决赛选手着色的弹簧跳远数据



请注意，大多数决赛选手在总距离的成分方面是一致的。但是，有两位选手（Jao 和 Ko）的一跳与其他跳不一致。例如，在 Jao 的三次跳远成绩中，第 1 跳约是总距离的 50%，但是在他的另一次跳远成绩中，第 1 跳仅是总距离的 30%。该跳与其他跳不一致。Ko 的跳远成绩也有类似的现象。

启动“三元图”平台

通过选择**图形 > 三元图**可启动 “三元图”。

图 11.5 “三元图”启动窗口

X，绘图 指定三列来生成一个三元图。

若将 3 个以上变量指定给 **X，绘图**角色，将创建三元图矩阵。将单独的变量指定给图的前两个轴，同时将第三个轴变为其余变量的总和。必要时可以缩放变量使得它们的和为 1。

等高线公式 要绘制响应曲面的等高线，可以将包含公式的列指定给**等高线公式**角色。若在等高线公式中有未列为 **X，绘图**变量的变量，JMP 将在图下方追加滑块，以便可以通过交互方式调整值。请参见第 222 页的 “使用等高线函数的示例”。

依据 该选项为“依据”变量的每个水平生成单独的图形。

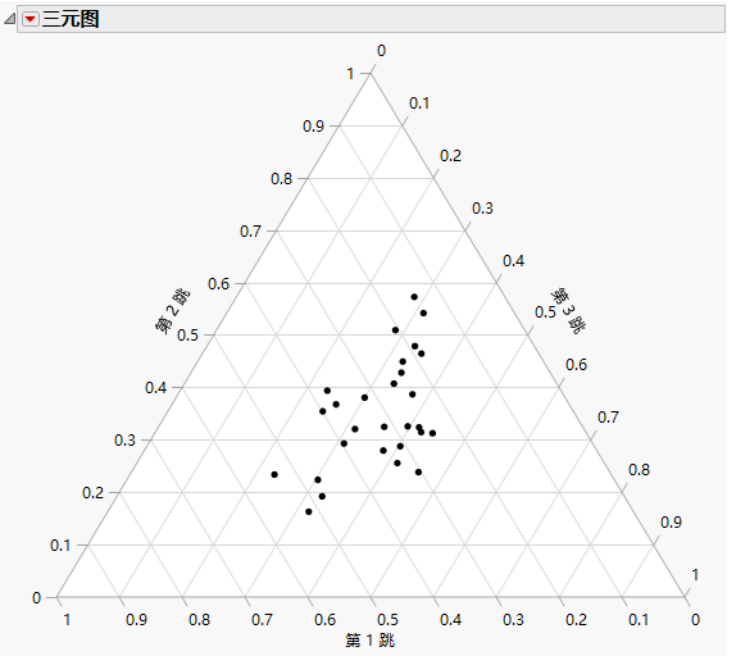
点击**确定**后，将显示“三元图”窗口。请参见第 220 页的“三元图”。

三元图

按照第 217 页的“三元图的示例”中的说明生成图 11.6 所示的图。

三元图的每条边表示比例为 0%，三角形中与底边相对的点表示比例为 100%。当任意一个样本中的比例增加时，表示该样本的点将从底边移到三角形的相对点。

图 11.6 三元图



混料和约束

三元图使用“混料”列属性为界限外的图形部分着色。三元图识别混料总和、混料界限和一般线性约束。有关在“列信息”窗口中设置“混料”列属性的信息，请参见《使用 JMP》。

相关信息

- 第 222 页的“使用混料约束的示例”

“三元图”平台选项

“三元图”红色小三角菜单包含用于修改图的选项。

注意：要查看更多详细选项，右击该图。

显示约束 显示或隐藏图的约束。默认图显示这些约束。

填充等高线 若在图中指定了等高线公式，则允许填充等高线。可以选择仅显示线条、向上填充或向下填充。默认平台仅显示线条。

颜色主题 允许您选择颜色主题。默认图显示 “由蓝经灰到红”颜色主题。

显示点 显示或隐藏绘制的点。默认图显示这些点。

适应窗口大小 指定是否在您调整 JMP 窗口大小时调整图形的大小。默认设置为 “自动”，该设置通常设置为“开”，除非该窗口与其他内容共享或是存在“页面”变量。要始终使图形适应窗口大小，请保持该设置为“开”。您还可以保持纵横比。要防止图形调整大小，请将设置改为“关”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“三元图”平台的更多示例

- “使用混料约束的示例”
- “使用等高线函数的示例”

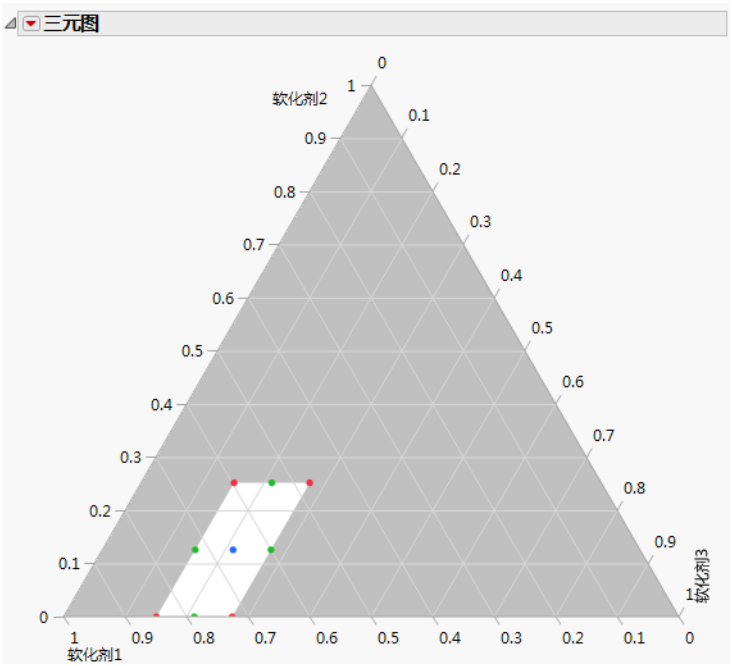
使用混料约束的示例

1. 选择**帮助** > **样本数据库**，然后打开 Plasticizer.jmp。

   软化剂 1、**软化剂 2** 和**软化剂 3** 列都定义了 “混料”列属性。
2. 从**图形**菜单中，选择**三元图**。

   请注意 p1、 p2 和 p3 显示在 **X，绘图**旁边。
3. 点击**确定**。

图 11.7 三元图中的混料约束



有关混料的详细信息，请参见《刻画器指南》。

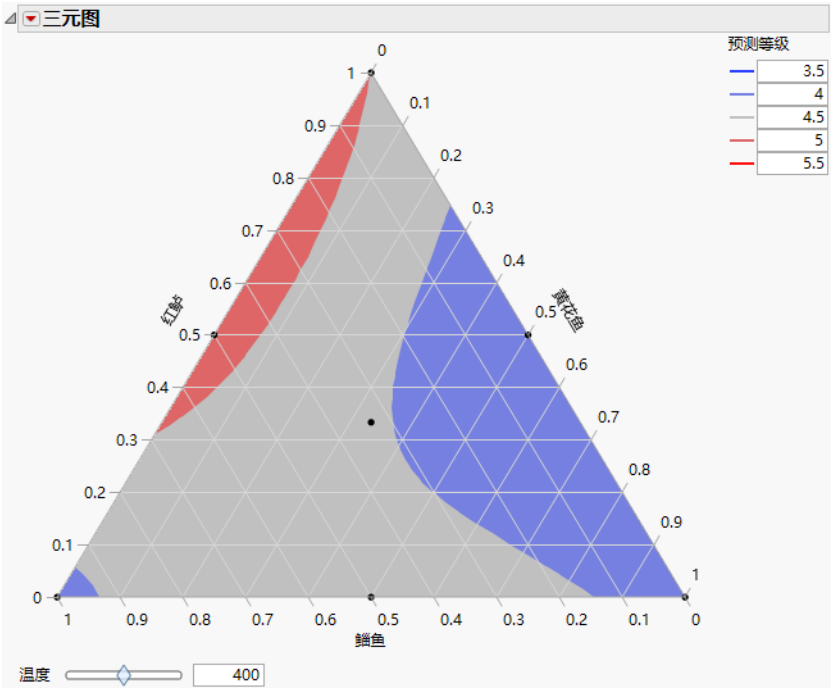
使用等高线函数的示例

Fish Patty.jmp 中的数据改编自 Cornell (1990)，来自优化鱼馅饼的配料的实验。**鳙鱼**、**红鲈**和**黄花鱼**列表示馅饼中这些鱼种类的比例。**温度**列表示用于烤馅饼的烤箱温度。**等级**列是响应，用于衡量配料被认可的程度，值越大表示越好。用响应曲面模型来拟合数据，将预测公式存储在**预测等级**列中。请参见 《刻画器指南》。

1. 选择**帮助** > **样本数据库**，然后打开 Fish Patty.jmp。
2. 从**图形**菜单中，选择**三元图**。
3. 选择**鳙鱼**、**红鲈**和**黄花鱼**并点击 **X，绘图**。

4. 选择**预测等级**并点击**等高线公式**。
5. 点击**确定**。
6. 点击“三元图”红色小三角并选择**填充等高线 > 向上填充**。

图 11.8 具有等高线公式的三元图



制造商希望等级能够达到至少 5 级。您可以拖动“温度”的滑块并查看“预测等级”变化的等高线。每个点表示这三种鱼的混料。根据烤馅饼的温度，这些鱼种类的任意给定混料会得到不同的等级。

在该示例中，红色着色区域显示等级为 5 到 5.5 的鱼混料。任意紫色区域显示等级为 4 或更低的鱼混料。在 400 度时，大部分为红鲈和鳝鱼、只含少量黄花鱼的混料会得到 5 或更高的等级。

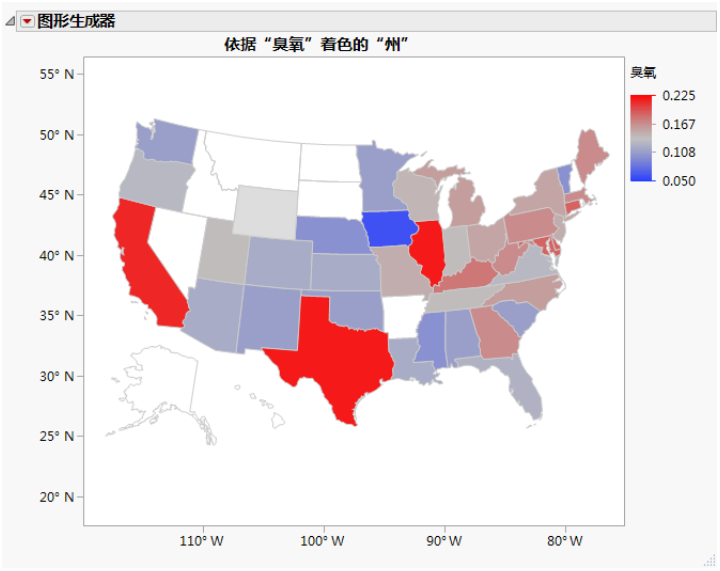
第 12 章

地图

添加地图或定制形状以增强数据可视化效果

JMP 将数字和地理数据转变为有趣的图像，将简单的数字表变为有吸引力的图片，使数据更贴近生活。JMP 可以帮助您在地图上显示数据。从内置的高品质图像中进行选择。选择“街道图服务”或“WMS(Web Map Service)”，以便从 Internet 获取定制地图图像。JMP 包括边界或许多地理区域的形状文件，并支持您针对制造厂或校园等情况添加定制形状。

图 12.1 地图示例



目录

地图绘制概述 227
在“图形生成器”中创建地图的示例 227
图形生成器 229
    地图形状 229
    颜色 231
    大小 232
    定制地图文件 235
背景地图 239
    地图中的图像 240
    边界 245
创建地图的示例 246
    路易安娜行政区域示例 246
    飓风跟踪示例 252
    办公室温度研究 257
    包含剪切元素的边界地图 262

地图绘制概述

JMP 中提供两种地图支持：一种是显示数据的地图（图形生成器），另外一种是提供数据环境的地图（背景地图）。您也可以自己创建地图。

图形生成器

您可以通过与“图形生成器”交互为数据创建有趣的图形显示。JMP 包括图形支持以使用背景地图和形状文件显示分析。您可以通过以下区域将颜色和地理边界添加到地图：

- **地图形状**区域基于数据表中的变量将地理边界指定给地图。地图形状值将确定 x 和 y 轴。

  随 JMP 安装了若干边界，比如美国各州名称、加拿大各省和日本各县。您还可以自己创建边界 （地理或其他形式）并在数据表中将它们指定为**地图角色**列属性。
- **颜色**区域基于变量将颜色应用于地理形状。
- **大小**元素根据此处的变量调整地图形状的大小，将失真最小化。

背景地图

通过“设置背景地图”窗口可以将背景地图添加到任何 JMP 图形。可以使用内置背景地图或连接到 Web Map Service (WMS) 来显示专业地图，比如卫星图像、雷达图像或道路。右击某一图形并选择**图形 > 背景地图**可以从以下图像和边界进行选择：

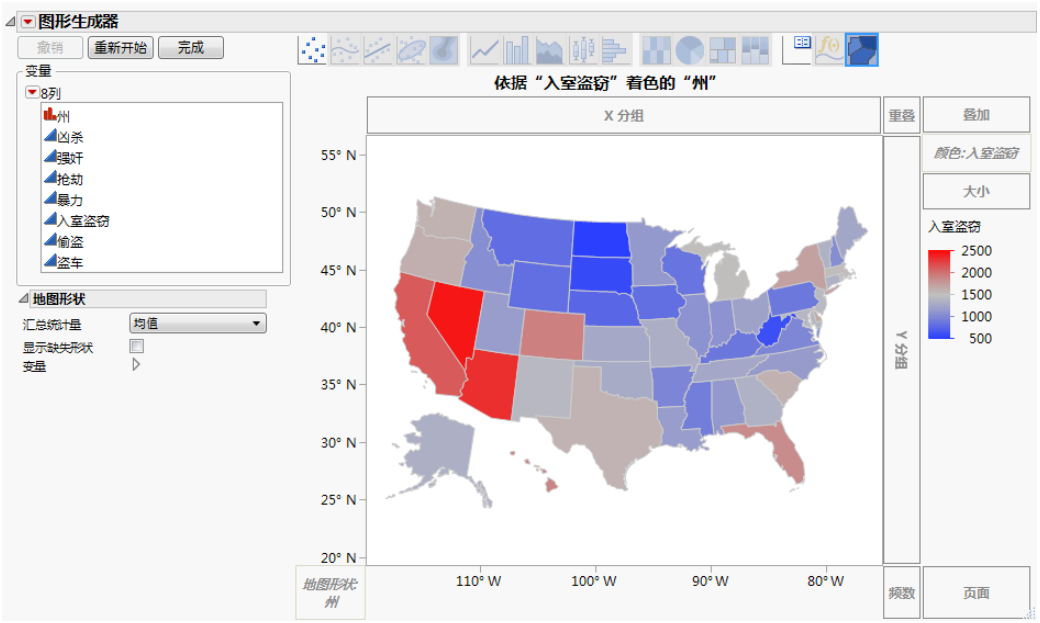
- 随 JMP 安装了**简单地球**和**精细地球**地图。
- **NASA 服务器**使用 WMS 显示最新的地图，以此来提供各种地图。
- **街道图服务**提供了街道图。OpenStreetMap 和 Open Database License 链接提供了有关街道图服务的更多信息。
- **WMS(Web Map Service)** 支持您输入使用 WMS 协议提供地图的网站的 URL。您还可以指定地图层。
- 各个地区的边界。

在“图形生成器”中创建地图的示例

该示例使用 Crime.jmp 样本数据表，表中包含美国各州的犯罪率数据。

1. 选择**帮助 > 样本数据库**，然后打开 Crime.jmp。
2. 选择**图形 > 图形生成器**。
3. 将**州**拖放到**地图形状**区域中。
4. 将**入室盗窃**拖放到**颜色**区域中。

图 12.2 依据 “州”列出的 “入室盗窃”示例



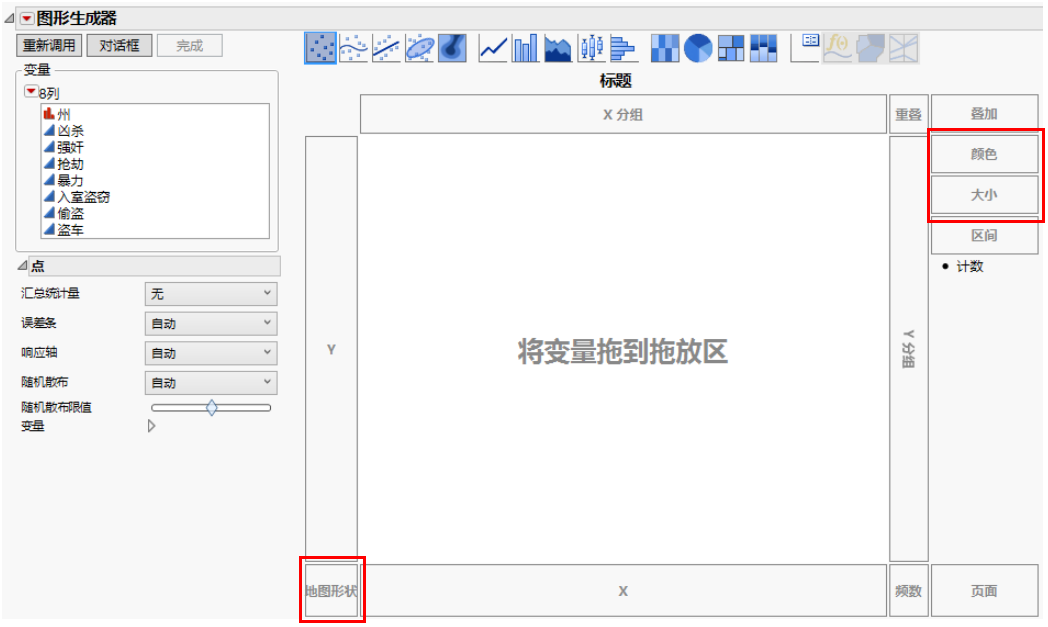
请注意以下几点：

- Y 轴和 X 轴上分别显示纬度和经度。
- 图例显示与入室盗窃率对应的颜色。因为**入室盗窃**是连续变量，颜色基于连续颜色主题。图 12.2 使用 JMP 默认连续主题。您可以在 “文件” > “首选项” > “图形” 下更改该主题。
- 地图经过投影，因此相对区域不会失真（横跨美国最上部的 49 度纬线不是直线）。

图形生成器

打开包含地理数据的数据表。通过选择**图形 > 图形生成器**可以启动“图形生成器”。“图形生成器”窗口中的主要元素是图形区域。图形区域包含多个拖放区（“地图形状”、“颜色”和“大小”），您可以将变量拖放到拖放区中。您可以从此处为包括地点名称的数据表绘制地图形状。

图 12.3 “图形生成器”窗口



地图形状

若某列包含地理区域（如国家、地区、州、省、县）的名称，您可以将该列分配至**地图形状**区域。当某个变量拖放到**地图形状**时，“图形生成器”会查找该变量各值对应的地图形状并绘制相应的地图。变量可以包含列属性来指示 JMP 查找地图数据的位置。若不包含列属性，JMP 会浏览所有已知的地图文件。若“地图形状”区域中有变量，X 和 Y 区域将消失。“地图形状”区域与位置相关，并且影响可用的图形元素类型。

图 12.4 将 “州” 拖放到 “地图形状” 后的 Cities.jmp 示例

每个地图存在两个 .jmp 文件；一个对应于名称数据 （每个实体对应一行），一个对应于坐标数据 （每个实体对应多行）。它们通过命名规则进行配对； xxx-Name.jmp 和 xxx-XY.jmp，其中 “xxx” 是某个共用的前缀。

JMP 附带的部分样本文件示例如下所示：

- World-Name.jmp
- World-XY.jmp
- US-State-Name.jmp
- US-State-XY.jmp

地图名称文件

每个 xxx-Name.jmp 可以包含任意数量的形状名称列，它们通过列属性标识。多个名称列支持本地化和备用名称样式（如缩写），但给定图形使用情况仅使用一列名称。“Name” 文件的第一列包含按升序排列的唯一 Shape ID（形状 ID）编号。JMP 为您创建该列。列值与 -XY.jmp 地图数据表 Shape ID（形状 ID）列中的值相匹配。

注意：Shape ID 列也可命名为 Shape（形状）。

图 12.5 US-State-Name.jmp 的示例

	Shape ID	Name	State FIPS	USPS Code	Abbr	Max Grow ...	Max Grow X Offset	Max Grow Y Offset
1	1	Alabama	01	AL	Ala.	1	0	0
2	2	Alaska	02	AK	Alaska	1	0	0
3	3	Arizona	04	AZ	Ariz.	1	0	0
4	4	Arkansas	05	AR	Ark.	1	0	0
5	5	California	06	CA	Calif.	3	-1.7	-3
6	6	Colorado	08	CO	Colo.	1	0	0
7	7	Connecticut	09	CT	Conn.	4	0.8	-1.1
8	8	Delaware	10	DE	Del.	3	0.5	-0.3
9	9	District of Columbia|Washi...	11	DC	D.C.	70	2	-2.5
10	10	Florida	12	FL	Fla.	4	2	-2
11	11	Georgia	13	GA	Ga.	1	0	0

地图 XY 文件

每个 xxx-XY.jmp 文件都包含四列。每一行是某个形状中的坐标。每个形状都由一个或多个部分组成。每个部分都是一个闭合多边形。第一列与 xxx-Name 文件中的 Shape ID（形状 ID）相同。第二列是 Part ID（部分 ID）。接下来的两列分别是 X 和 Y。

图 12.6 US-State-XY.jmp 的示例

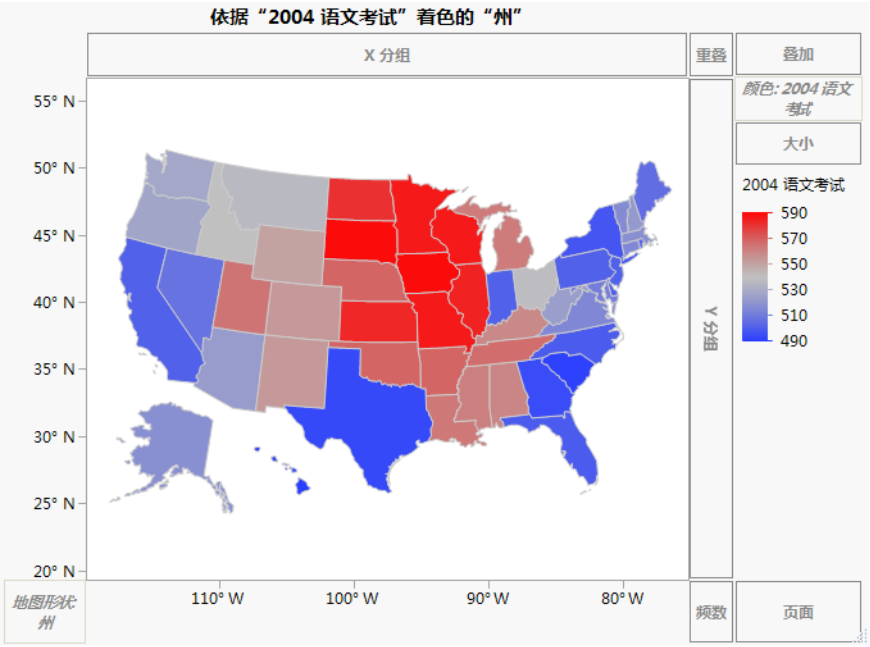
	Shape ID	Part ID	X	Y
1	1	1	85.6052° W	34.9847° N
2	1	1	85.1844° W	32.8613° N
3	1	1	84.9630° W	32.4242° N
4	1	1	85.0071° W	32.3284° N
5	1	1	84.8909° W	32.2615° N
6	1	1	85.0611° W	32.1341° N
7	1	1	85.1407° W	31.8575° N
8	1	1	85.0413° W	31.5410° N
9	1	1	85.1075° W	31.1865° N
10	1	1	85.0024° W	31.0007° N
11	1	1	87.5989° W	30.9975° N

颜色

“图形生成器”平台允许您添加颜色以创建等值线图。等值线图在保持统计变量比例的同时显示地理区域中的统计差异。

将包含地理位置名称（比如国家、地区、州或省）的列拖到**地图形状**区域并创建地图。然后将一列拖放到**颜色**区域以按该列对地图着色。在“首选项”中选定的分类或连续颜色主题会应用于每个形状。

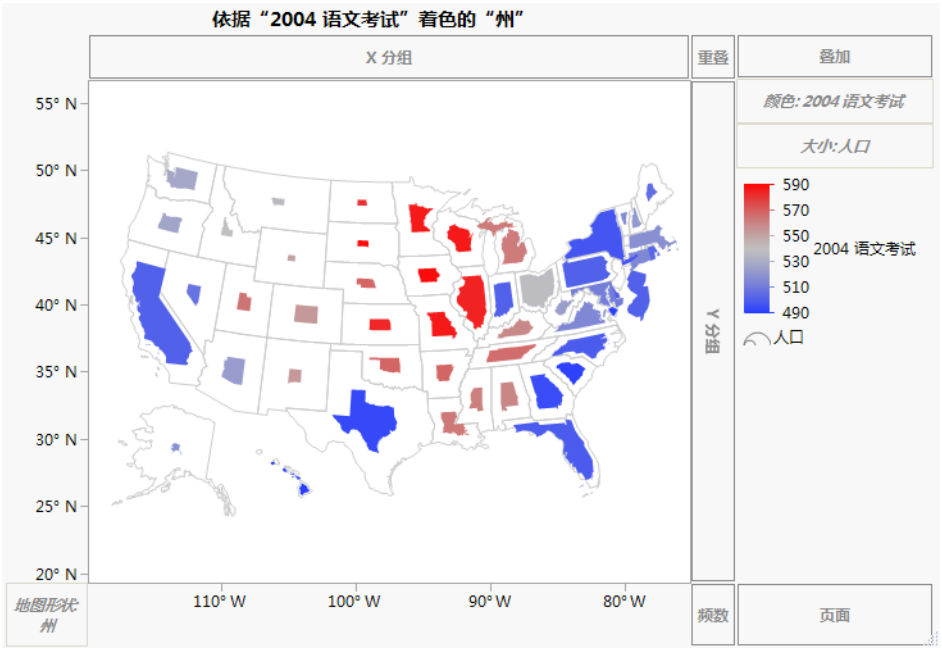
图 12.7 将 “2004 语文考试” 拖放到 “颜色” 后的 SAT.jmp 示例



大小

使用**大小**元素可以根据大小变量调整地图形状的大小，将失真最小化。

图 12.8 将 “人口” 拖放到 “大小” 后的 SAT.jmp 示例



定制图形

要更改地图的颜色和透明度，请右击图例中的颜色条。根据颜色变量是连续变量还是分类变量（名义型或有序型），右击选项有所不同。不过，对于这两种变量，您都可以更改透明度。

更改图形的透明度：

1. 右击要更改的颜色条中变量水平的颜色并选择**透明度**。
2. 指定介于 0 （透明）和 1 （不透明）之间的透明度。
3. 点击**确定**。

您还可以更改图像（例如 “简单地球” 和 “精细地球” ）的透明度。要设置透明度，在图形上右击并选择**定制...**。这将显示 “定制图形” 窗口，您可以在其中选择 “背景地图” 并为透明度分配一个值。透明度的有效值范围从 0.0（完全透明）到 1.0（完全不透明）。

分类（名义型或有序型）变量使用单一着色系统，即对变量的每个水平设置不同的颜色。

更改其中一个变量水平的颜色：

1. 右击要更改的变量水平的颜色并选择**填充颜色**。
2. 选择新颜色。

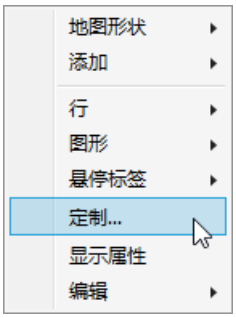
连续变量使用颜色梯度。

更改颜色主题：

1. 右击颜色条并选择**梯度**。
2. 在“梯度设置”窗口中，选择不同的**颜色主题**。

图形包含标记、线条、文本和其他可以定制的图形元素。若您右击某个图像，将出现若干用于设置图形的选项。选项根据您点击的位置而有所不同。请参见“图形生成器”一章中第 100 页的“梯度设置”和《使用 JMP》。下面是一些选项。

图 12.9 图形的右击菜单



- 地图形状：
  - 更改为 — 提供用于将地图改为其他图形类型的选项。
  - 汇总统计量 — 提供用于更改所标绘的统计量的选项。
  - 显示缺失形状 — 显示或隐藏地图中的缺失数据（默认处于关闭状态）。缺失形状意味着有些形状名称在地图文件中存在，但在用于分析的数据表中不存在。
  - 删除 — 删除当前地图形状。
- 定制 — 可以更改图形的属性（例如，内容、网格线或参考线）。您可以定制的图形元素因每个图形而异。选择**背景地图**可以更改背景地图的透明度，选择**地图形状**可以更改线条颜色、线条样式和粗细、填充颜色、缺失形状填充或缺失值填充。点击“定制图形”窗口中的**帮助**可以查看定制选项更加详细的说明。

更改形状文件：

1. 右击“地图形状”区域并选择**设置形状文件**。
2. 导航到包含名称数据的形状文件。
3. 点击**打开**。
4. （可选）要恢复更改，请右击 “地图形状”区域并选择**重置形状文件**。

定制地图文件

遵循与内置文件相同的模式，您可以自行创建地图文件。要添加您自己的地图文件，您需要两样东西：描述形状的多边形顶点所对应的一系列 XY 坐标，以及每个多边形对应的一组名称。绘制定制地图形状需要数据和形状属性，这样您才能将自己的形状添加到 JMP。有两种常见的数据来源：ESRI 形状文件和 SAS/GRAPH 地图数据集。

要让 JMP 能够自动查找您的文件，请将这些文件放在以下目录中：

- 在 Windows 上：C:/Users/<用户名>/AppData/Roaming/SAS/JMP/Maps
- 在 macOS 上：/Users/<用户名>/Library/Application Support/JMP/Maps

注意：在 Windows 的 JMP Pro 中，“JMP”文件夹命名为“JMPPro”。或者，您可以将地图文件链接至您的已显式使用**地图角色**列属性的数据文件。

请在创建地图文件时注意以下事项：

- 您创建的每组地图文件都必须包含 -Name 文件和 -XY 文件。
- 这两个文件中的第一列都必须是以升序排列的数值型 Shape ID（形状 ID）变量。-Name 文件可包含任何其他列。形状按行生成。XY 坐标必须环绕该形状一周而不能仅仅定义该形状的凸包。
- 对于**地图角色**列属性，将在标记有**形状名称定义**的列中搜索形状标识，并且这些列必须包含唯一值。
- 若您导入 Esri SHP 文件，它将以正确的格式打开。-Name 文件通常具有 .dbf 扩展名。请参见第 237 页的“Esri 形状文件”。
- SAS/GRAPH 软件包括许多可用于 JMP 的地图数据集。请参见第 239 页的“SAS/GRAPH 地图数据集”。

您可能想创建其他非地理区域（例如，办公楼的楼层）的等值线图。只需为非地理空间添加两个形状文件。若您没有 XY 坐标，但具有空间的图形图像，则可以使用 JMP 的 Custom Map Creator 插件。借助该插件，您可以跟踪空间的轮廓，并且 JMP 会为您创建 -XY 和 -Name 文件。您可以从“JMP 文件交换库”页下载该插件。

地图角色

您可以在“列信息”窗口的“列属性”中指定数据表中某列的特性和属性。可以像“列信息”窗口中的其他列属性那样设置列的**地图角色**属性。

若您自己创建的数据表中包含边界数据（例如，国家、地区、州、省或县），并且想在“图形生成器”中查看相应的地图，则可以使用“列属性”中的**地图角色**属性。您创建的每对地图文件都必须包含 -Name 文件和 -XY 文件。

请注意以下几点：

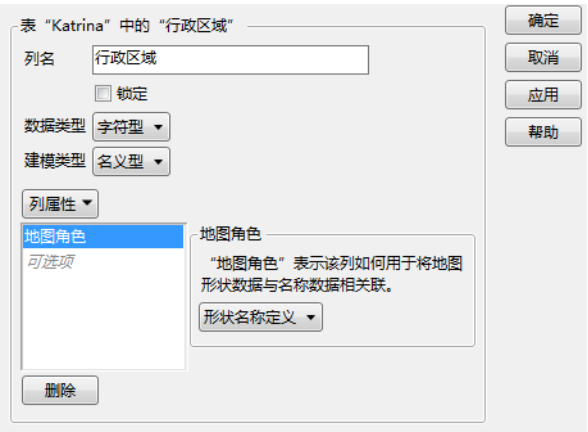
- 若定制边界文件位于默认定制地图目录中，则您只需要在 -Name 文件中指定**地图角色**属性。

- 若定制边界文件位于其他位置，则需要在 -Name 文件中和您要分析的数据表中指定**地图角色**属性。
- 包含**地图角色**属性的列必须包含相同的边界名称，但列名可以不同。

将 “地图角色” 属性添加到 -Name 数据表：

1. 右击包含边界的列，然后选择**列属性 > 地图角色**。
2. 选择**地图角色**下方的**形状名称定义**。
3. 点击**确定**。
4. 保存数据表。

图 12.10 “形状名称定义” 示例

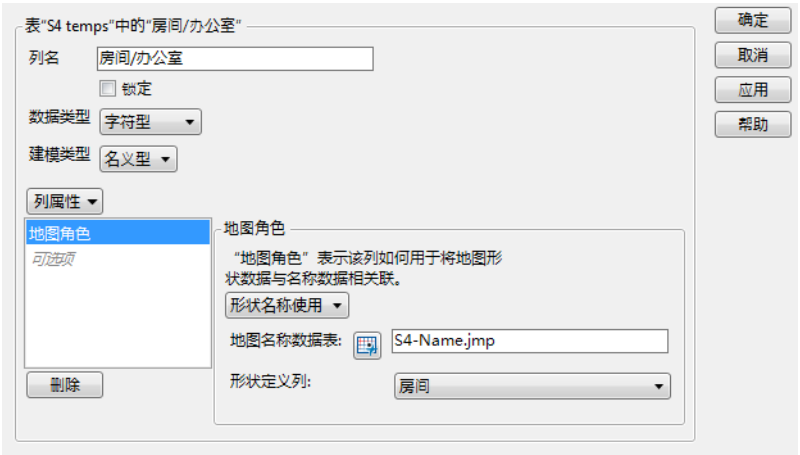


将 “地图角色” 属性添加到您要分析的数据表：

注意：仅当您的定制边界文件未包含在默认定制地图目录中时才执行以下步骤。

1. 右击包含边界的列，然后选择**列属性 > 地图角色**。
2. 选择**地图角色**下方的**形状名称使用**。
3. 在**地图名称数据表**旁边，点击 浏览至 -Name 地图数据表。您可以输入相对或绝对路径。若地图数据表位于同一文件夹中，只需输入文件名。路径包含空格时无需使用引号。
4. 从**形状定义列**列表中，选择地图数据表中其值与选定列中的值匹配的列。
   图 12.11 显示 S4 Temps.jmp 样本数据表中**房间 / 办公室**列的示例。

图 12.11 形状定义列示例



5. 点击**确定**。
6. 保存数据表。

当您在“图形生成器”中生成图形并将修改的列分配至**地图形状**区域时，边界将出现在图形中。

对于数值列，“格式”菜单将出现在 “列信息”窗口中。指定格式以指示 JMP 如何显示列中的数字。当定制轴和轴标签时，地图的纬度和经度位于**格式 > 地理**下方。

地理 显示地图的纬度和经度数字格式。纬度和经度选项如下：

- DDD（度）
- DMM（度、分）
- DMS（度、分、秒）

在每种格式中，最后一个经纬度单位可具有小数部分。可以使用带符号的经纬度单位或方向后缀来指定方向。要显示带符号的经纬度单位，如 *-59°00′00″*，请取消选择**方向指示符**。要显示方向后缀，如 *59°00′00″ S*，请选择**方向指示符**。

要将空格用作经纬度单位分隔符，请取消选择**经纬度单位符号**。要使用度、分、秒符号，请选择**经纬度单位符号**。

Esri 形状文件

Esri 形状文件是一种向量数据格式，其中包含有关如地形和海洋等地理特征的数据。它作为地图绘制软件的规范由 Esri 开发和管理。

每个形状文件都是一组具有相同名称和不同扩展名的文件。

主文件 (.shp)

.shp 文件包含组成多边形的一系列点。使用 JMP 打开时， .shp 文件会作为 JMP 表导入。

- Shape（形状）列在导入时添加，以便唯一标识每个地理区域。每个坐标点位于单独的行中。
- 指出不相连区域的 Part（部分）列，以及 XY 坐标（采用纬度和经度表示）。

JMP 支持二维 .shp 文件（无仰角信息）。

dBase 表 (.dbf)

您将 Shape ID（形状 ID）列添加至 .dbf 表，该列映射至 .shp 文件中的 Shape（形状）列。添加任意数量的列，这些列可提供公用名称或值以表示特定区域。

将 Esri 形状文件转换为 JMP 地图文件：

1. 在 JMP 中打开 .shp 文件。
2. 确保 Shape （形状）列是 .shp 文件中的第一列。为 X 和 Y 列添加格式和轴设置 （可选）。“图形生成器”对 X 和 Y 轴使用这些设置。
3. 将 .shp 文件作为 JMP 数据表保存到 Maps 文件夹，名称以 -XY.jmp 结束。
4. 打开 .dbf 文件。
5. 将 Shape ID （形状 ID）列添加为表中的第一列。该列的值应为行号 1 至 n — 数据表中的行数。

   注意：您可以使用**列 > 新建列 > 初始化数据 > 序列数据**用序列号填充列。

6. 将**地图角色**列属性分配至您用于 “图形生成器”的 “形状”角色中位置名称的任意列。为此，右击列顶部并选择**列属性 > 地图角色**。
7. 从属性定义的下拉框中选择**形状名称定义**。
8. 将表保存为 JMP 数据表，其名称与前面的表匹配并以 -Name.jmp 结束。

JMP 在两个位置查找这些文件。一个位置由机器上的所有用户共享。该位置为：

- Windows：C:/Program Files/SAS/JMP/16/Maps/
- macOS：/Library/Application Support/JMP/16/Maps

另一个位置特定于每位用户：

- 在 Windows 上：C:/Users/<用户名>/AppData/Roaming/SAS/JMP/Maps
- 在 macOS 上：/Users/<用户名>/Library/Application Support/JMP/Maps

注意：在 Windows 的 JMP Pro 中，JMP 文件夹命名为 JMPPro。

SAS/GRAPH 地图数据集

SAS/GRAPH® 软件包括的若干地图数据集经过转换可用于 JMP。数据集位于 Maps 逻辑库中。传统地图数据集包含 XY 坐标数据和具有常用位置名称的功能表。您需要将这两个文件均转换为 JMP 数据表才能在 JMP 中使用。

多数传统地图数据集具有未投影的纬度和经度变量 （以弧度表示）。一旦数据集转换为度并且经度变量进行了投影调整，这些数据集即可在 JMP 中使用。以下是一个 DATA 步，它显示了 Belize 数据集的转换过程。

```
data WORK.BELIZE;
keep id segment x y;
rename segment=Part;
set maps.belize;
if x NE .;
if y NE .;
y=lat*(180/constant('pi'));
x=-long*(180/constant('pi'));
run;
```

您现在可以导入转换的文件并将其保存为 Belize-XY.jmp。

下一步是导入匹配功能数据集（在此情况下为 MAPS.BELIZE2）。导入功能数据集后，将 ID 列移动到数据表中的第一列。然后将**地图角色**列属性分配给您用于“图形生成器”的“形状”角色中地名的列。为此，右击列顶部并选择**列属性 >地图角色**。然后从属性定义的下拉框中选择**形状名称定义**。对于 MAPS.BELIZE2，使用 IDNAME 列。将功能数据集保存为 Belize-Name.jmp。

要转换 SAS 地图，请从“JMP 文件交换库”页下载 SAS to JMP Map Converter 插件。对于每个地图，插件从两个 SAS 地图表中读取数据，重新排列并设置数据格式，然后将其放置到两个 JMP 地图表中。

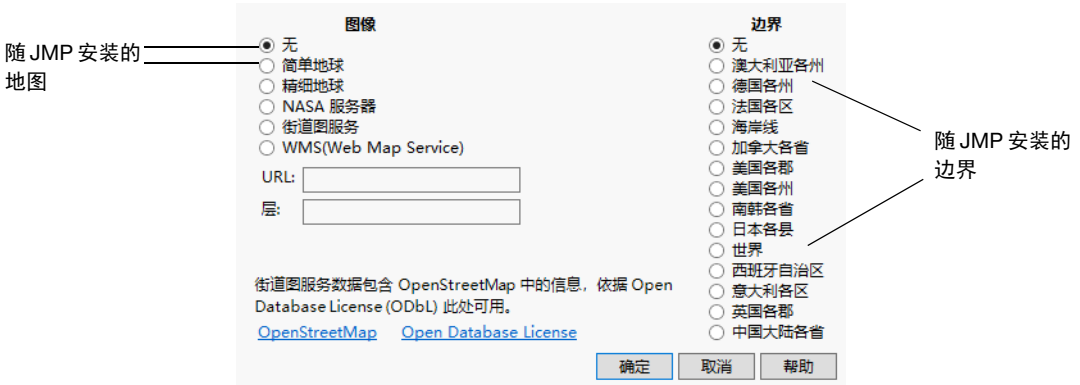
背景地图

向图形添加地图图像和边界可为地理空间数据提供可视环境。附加背景地图可以生成有吸引力的地图，从而为数据提供地理环境并提供给您一种全新的方式来查看数据。例如，您可以向图形添加显示美国图像的地图。另一个选择是显示每个州的边界（在数据包含美国的纬度和经度时）。背景地图有不同的类型。有些地图内置在 JMP 中，随 JMP 安装一起提供。有些地图从 Internet 检索得到，还有些地图是用户定义的。

数据应包含纬度和经度坐标。否则，地图在数据环境中没有意义。根据地图类型，X 和 Y 轴还具有相应的范围要求。以下各节对这些要求进行了说明。只需在 X 和 Y 轴上绘制经度和纬度，然后在图形内右击并选择**图形 > 背景地图**。

“背景地图”窗口显示两列选项：“图像”和“边界”。在窗口的左侧，您可以从两个内置地图图像中进行选择，也可以连接到 Web Map Service 以检索背景图像。在窗口的右侧，您可以选择一些地区的行政边界。

图 12.12 “背景地图”选项的示例



提供以下“背景地图”选项：

图像

无 删除在 “图像”列中选定的背景地图。

简单地球 显示提供基本地形的地图。例如，运行以下样本数据表中的关联脚本：SAS Offices.jmp 和 Aircraft Incidents.jmp。

精细地球 显示提供精细地形的高分辨率地图。

NASA 服务器 显示 NASA 服务器上的地图。要求具有 Internet 连接。

街道图服务 根据显示的缩放级别显示具有相应详细程度的地图。支持您一直缩放至街道级别。例如，运行以下样本数据表中的关联脚本：SAS Offices.jmp。

WMS(Web Map Service) 显示来自统一资源定位符 (URL) 的地图以及您指定的图层。要求具有 Internet 连接。

边界

无 删除在 “边界”列中选定的边界。

各个地区的边界 显示地图区域的边界，如加拿大各省、美国各郡、美国各州和世界各国。列表因您所在的位置而异。基于 Esri 形状文件创建的地图也列在这里。

查看地图时有两种工具尤为有用：

- 抓手工具 ( ) 支持您在地图中水平和垂直滚动。
- 放大镜工具 ( ) 支持您缩放地图。

地图中的图像

每张平面地图都会在某种程度上歪曲地球的表面。地图并不与地球完全匹配，因为它不能真正代表整个地球的表面。地图投影用于在平面上描绘圆形地球的全部或一部分。这必然会导致部

分失真。每种投影方式都有其自身的优势和劣势。地图可以显示以下一项或多项，但不能全部显示：真实方向、距离、面积或形状。JMP 对其地图使用两种投影（Albers Equal Area Conic 和 Kavrayskiy VII）。在“图像”内，您可以从两个内置地图图像中进行选择，也可以连接到 Web Map Service 检索背景图像。

随 JMP 安装的地球图像

JMP 提供两个级别的地球图像：简单和精细。这两个地图都显示地貌特征，比如水域和地域。但是，精细地图显示更清晰的地形。使用精细地图，您可以放大更多，并且地图特征仍然保持清晰。图像地图是光栅图像。地图水平换幅，因此当您从左到右滚动地图时可以继续看到地图细节。地图不会垂直换幅。超出 -90 和 90 y 轴范围，将显示白色背景而不显示地图。

图 12.13 简单和精细地图的示例

简单地球图像

精细地球图像

正如名称那样，“简单地球”是地球地貌的相对简单的图像。它不显示云团或北极冰，并且它使用绿色和棕色来表示陆地以及没有变化的深蓝色表示水域。“精细地球”比“简单地球”的颜色更加柔和，浅绿色和棕色表示陆地，各种蓝色表示水域。“精细地球”比“简单地球”还具有略高的分辨率。“精细地球”与“简单地球”相比，较高的分辨率使您在背景图像变得模糊之前能够更大程度地放大图形。

“简单地球”和“精细地球”的另一项功能是可以换幅。地球是圆的，当您横跨 180° 经度时，没有到达地球的尽头。经度值从 -180° 继续增加。地图在水平方向上不断地换幅，几乎像地球那样。背景地图在垂直方向上不会换幅。

“简单地球”和“精细地球”都支持测地线尺度。在第 241 页上的图 12.13 中，地球显示为矩形，其宽度是高度的两倍。若我们将这个矩形拿来卷起，将得到一个圆柱体。事实上，我们知道地球不会形成圆柱体，而应该是一个球形。您可以使用测地线尺度，它将地图变为以更现实的方式表示地球。要使用测地线尺度，可以更改坐标轴上尺度的类型。

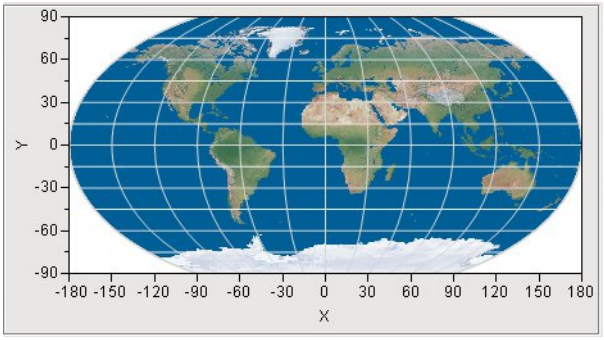
更改坐标轴尺度：

1. 在 X 或 Y 轴上右击，然后选择**轴设置**。
2. 将 “尺度类型”更改为**测地线**或**测地线 US**。

图 12.14 “Y 轴设置”窗口

这两个选项都会将地图转变为测地线尺度。若您查看的是美国大陆地图并且想在地图中包括阿拉斯加州和夏威夷州，则使用“测地线 US”。重要的是，您必须将两个轴的尺度都改为测地线才能进行变换。仅设置其中一个轴不会看到地图中有任何变化。在下图中，“简单地球”用作背景地图，而轴设置为使用测地线尺度。还启用了轴线。请注意，经度线现在是曲线而非直线。

图 12.15 简单地球背景地图 — 轴设置为测地线尺度

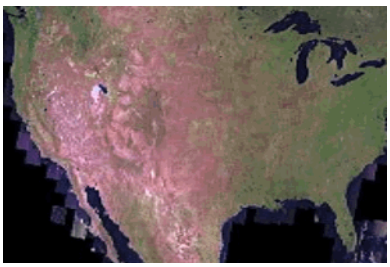


因为“精细地球”和“简单地球”均内置于 JMP 中，这些选项可以在没有网络连接的情况下随时使用。但是，这些图像可能不是您所需要的，或者它们的精细程度达不到您所需要的分辨率。若发生这样的情况，并且您有 Internet 连接，则可以连接到 Web Map Service 以检索满足您需要的地图图像。

来自 Internet 的地图

美国国家航空航天局 (NASA) 和其他组织使用名为 Web Map Service (WMS) 的协议提供地图图像数据。这些地图具有显示最新地理信息的优势。但是，由于服务器响应时间的关系，地图显示会很慢，并且站点会随时更改或消失。访问信息需要 Internet 连接。

图 12.16 NASA 和 WMS 地图的示例

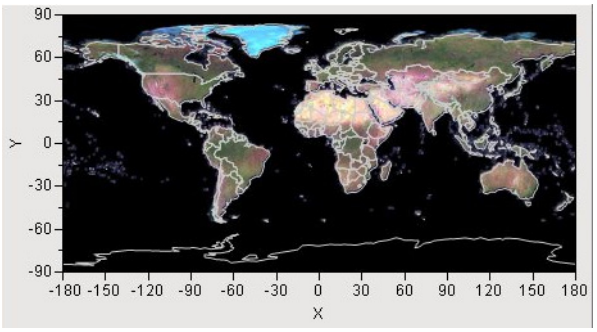


NASA 图像

来自 MetaCarta™ 服务器的 WMS 图像

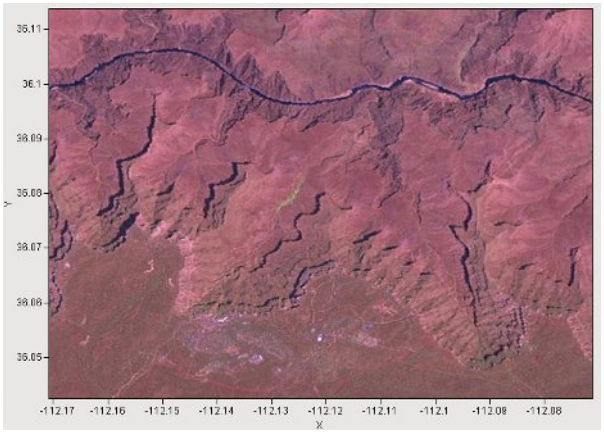
NASA 服务器提供整个地球的地图。下图使用 NASA 服务器作为其背景地图来源显示地球。边界地图显示各国/地区的轮廓。

图 12.17 NASA 服务器地图示例



该服务器不仅涵盖了整个地球，您还可以在地球上某处较小的区域放大而仍能得到适当的地图。下图显示流经亚利桑那州大峡谷的科罗拉多河。地图底部可以看到大峡谷村。

图 12.18 NASA 服务器地图示例 — 放大科罗拉多河



若您查看轴值，会发现区域小于 1/10° X 1/10° 的范围。“简单地球”和“精细地球”背景地图不会显示该分辨率的类型。NASA 服务器提供地球上任何陆地块相对精细的视图。不过，水域只是填补为黑色。NASA 服务器可以免费访问，但它也有可用性限制。若服务器临时不可用或因请求超载，它将显示错误消息而非请求的地图。

背景地图的另一个基于 Internet 的选项为 Web Map Service (WMS)。WMS 选项允许您指定任何支持 WMS 接口的服务器。例如 NASA 服务器就是一个 WMS 服务器，但我们给您提供了 URL 和层名称。使用 WMS 选项，您必须知道 WMS 服务器的 URL 和该服务器支持的层名称。大多数 WMS 服务器支持多个层。例如，一层可以显示地域，另一层可以显示公路，再一层可以显示水域（例如河流和湖泊）。通过指定服务器的 URL 和层，JMP 可以请求服务器并显示返回的地图。

与简单和精细地图不同，WMS 地图不会换幅。您可以水平和垂直滚动。但是，超过 -180 至 180（x 轴）和 -90 至 90（y 轴）的范围，将显示白色背景而非地图。轴限值用于定义所显示地图的限制范围。

为了对背景地图使用 WMS 选项，您需要确定所使用的 WMS 服务器。Internet 中免费提供了很多 WMS 服务器。大部分仅提供世界上特定区域的地图，而每个都只支持各自的层。因此您必须根据特定的情形搜索适当的 WMS 服务器。

可以使用喜爱的搜索引擎在 Internet 上搜索 WMS 服务器。一旦找到一个服务器，您需要确定它支持的层。为此，您可以使用 WMS Explorer 插件。WMS Explorer 插件将生成服务器提供的所有层列表。您可以从列表中选择一个层以查看它。您可以从“JMP 文件交换库”页下载 WMS Explorer 插件。

注意：要使用 WMS Explorer 插件和 JMP 的 WMS 背景地图功能，您的计算机必须连接到 Internet。

要定位服务器，通过以下菜单项启动插件：**插件 > 地图图像 > WMS Explorer**。插件将显示一个文本框，用于输入已知的 WMS 服务器的 URL。还可以从预先发现的 WMS 服务器的下拉列

表中进行选择（列表可能已过时）。指定 WMS 服务器后，选择**获取层**。若从下拉列表中进行选择或在输入 URL 后点击**确定**，则不必使用“获取层”。这将向 WMS 服务器发送请求服务器支持的层列表。返回的列表显示在左侧标为**层**的列表框中。世界地图将在图形靠右的位置显示轮廓。选择一层将向 WMS 服务器请求使用指定的层（表示整个地球）返回地图。选择其他层将生成不同的地图。

默认地图不涵盖整个地球（例如，某些 WMS 服务器可能提供某个州内特定县的地图绘制数据）。在这种情况下，很有可能选择一层并不会生成任何可见的地图。您必须先放大适当的区域，才能使图像地图可见。插件窗口中提供了标准 JMP 工具栏，缩放工具与任何 JMP 窗口中的操作类似。

图形是 JMP 中的典型图形，这意味着所有常规 JMP 控件均可使用。您可以像在 JMP 中那样调整轴或使用缩放工具（位于菜单栏中）。还可以右击鼠标以选择**大小/尺度 > 调整为相同大小**使图形恢复为正常的纵横比。还可以选择“背景地图”从而调整边界地图。

一旦确定了所需的地图，需要记录顶部文本框中的 URL 和“层”列表中选定的层。在 WMS 选定为图像背景地图的类型时，您需要在背景地图窗口中输入该信息。

因为通过 Internet 向服务器发送请求，有许多情况会发生错误。WMS 服务器经常限制使用，有时根本无法使用。WMS 服务器偶尔会返回它不再支持的层名称。在这些情况和其他情况下，服务器通常会返回错误消息而非地图。若发生这种情况，会在标为**错误**区域中的“层”列表下显示错误消息。

边界

JMP 可以显示边界（例如美国各州或法国各区边界）。这些边界在定义区域的四周绘制一个轮廓，它们可以在图形上单独显示，也可以结合图像数据一同显示。随 JMP 安装了若干边界。您也可以基于 Esri 形状文件自行创建边界，或从零开始创建边界。为此，您在“设置背景地图”窗口中看到的边界列表会有所不同。

当您将形状文件添加到 JMP 中的内置位置时，它们不仅可用于“图形生成器”平台，还可以用于“背景地图”窗口中的“边界”选项。通过这种方式，您可以添加更多政治边界以用于背景地图。边界样式的地图是向量式的形状。

图 12.19 美国各州边界示例

添加背景地图和边界

添加背景地图和边界：

1. 右击图形中的空白区域，然后选择**背景地图**（或在“图形生成器”中选择**图形 > 背景地图**）。
   “设置背景地图”窗口随即显示 （图 12.12）。
2. 要显示背景，请执行以下操作之一：
   – 在“图像”列中选择**简单地球**、**精细地球**、**NASA 服务器**或**街道图服务**。
   – 选择 WMS(Web Map Service) 并在 URL 旁边粘贴一个 WMS URL。在**层**旁边输入层标识符。
3. 要显示地图中的地理边界，请在边界列中选择某个选项（若已安装您自己的边界形状文件，这些文件也会列在该列中）。
4. 点击**确定**。
   若 NASA 地图、街道图或 WMS 地图在添加后并不显示，则可能是地图服务器不可用。查看错误日志以查明问题。

创建地图的示例

- “路易安娜行政区域示例”
- “飓风跟踪示例”
- “办公室温度研究”
- “包含剪切元素的边界地图”

路易安娜行政区域示例

在该示例中，您处理定制地图文件，然后通过以下两种方式创建定制地图：

- 先设置定制地图文件，并将其保存在预先确定的位置。JMP 将来会查找这些文件，并与任何适当数据一起使用。
- 从您的数据中直接指向特定的预定义地图文件。每次要指定定制地图时可能都需要执行该步骤。

设置自动定制地图

假设您从 Internet 下载了 Esri 形状文件并想使用它们作为 JMP 中的地图文件。形状文件名为 Parishes.shp 和 Parishes.dbf。这些文件包含路易斯安那州各行政区域（或县）的坐标和相关信息。

注意：本节中的路径名是指 JMP 文件夹。在 Windows 的 JMP Pro 中，JMP 文件夹命名为 JMPPro。

保存 .shp 文件

用适当的名称保存 .shp 文件，并保存在正确的目录中。

1. 在 JMP 中，从以下默认位置打开 Parishes.shp 文件：
   - 在 Windows 上： C:/Program Files/SAS/JMP/16/Samples/Import Data
   - 在 macOS 上：/Library/Application Support/JMP/16/Samples/Import Data

   注意：若看不到该文件，您可能需要将文件类型改为**所有文件**。

   JMP 将该文件打开为 Parishes。 .shp 文件包含 x 和 y 坐标。
2. 使用下面的名称和扩展名保存 Parishes 文件：Parishes-XY.jmp 将该文件保存到以下位置：
   - 在 Windows 上：C:/Users/<用户名>/AppData/Roaming/SAS/JMP/Maps
   - 在 macOS 上：/Users/<用户名>/Library/Application Support/JMP/Maps
3. 关闭 Parishes-XY.jmp 文件。

保存 .dbf 文件

执行初始设置并保存 .dbf 文件。

1. 从以下默认位置打开 Parishes.dbf 文件：
   - 在 Windows 上： C:/Program Files/SAS/JMP/16/Samples/Import Data
   - 在 macOS 上：/Library/Application Support/JMP/16/Samples/Import Data

   注意：若看不到该文件，您可能需要将文件类型改为**所有文件**。

   JMP 将该文件打开为 Parishes。 .dbf 文件包含标识信息。
2. 右击**行政区域**列并选择**列信息**。
3. 选择**列属性 > 地图角色**。
4. 选择**形状名称定义**。
5. 点击**确定**。
6. 使用下面的名称和扩展名保存 Parishes 文件：Parishes-Name.jmp。将该文件保存到以下位置：
   - 在 Windows 上：C:/Users/<用户名>/AppData/Roaming/SAS/JMP/Maps
   - 在 macOS 上：/Users/<用户名>/Library/Application Support/JMP/Maps
7. 关闭 Parishes-Name.jmp 文件。

在“图形生成器”中创建地图

地图文件一经设置即可使用。Katrina.jmp 数据表包含按行政区域划分的卡特里娜飓风影响的相关数据。您想要直观地查看卡特里娜飓风过后各行政区域人口有何变化。

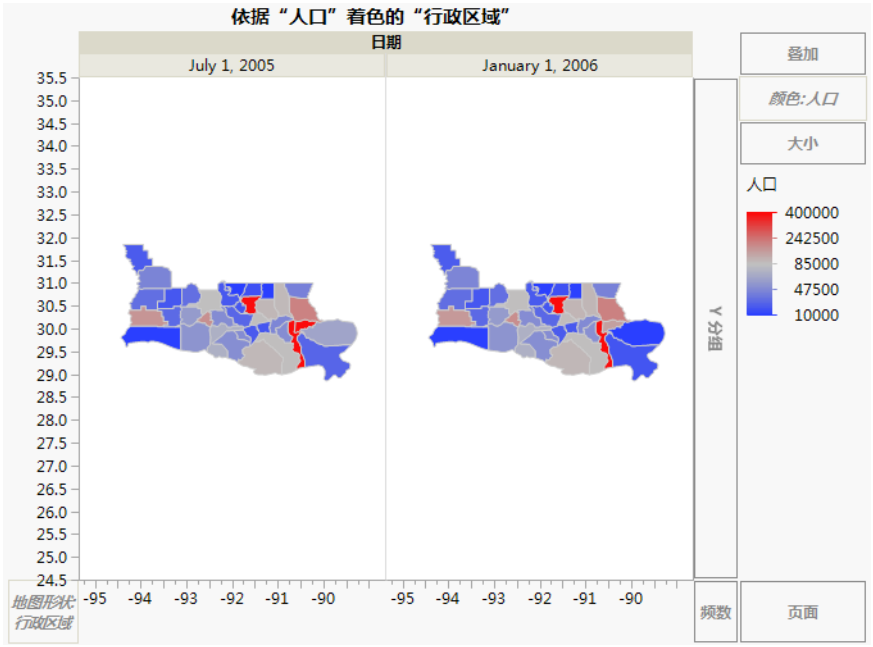
1. 选择**帮助** > **样本数据库**，然后打开 Katrina.jmp。
2. 右击**行政区域**列并选择**列属性** > **地图角色**。
3. 选择**形状名称使用**。
4. 点击**地图名称数据表**按钮 ，浏览选择您之前创建的 Parishes-Name.jmp。

   这将为 JMP 指示包含地图信息的数据表所在的位置。
5. 从**形状定义列**列表中选择**行政区域**。

   在 Parishes-Name.jmp 中，已为**行政区域**列指定**形状名称定义地图角色**属性。该列由每个行政区域的地图形状数据组成。
6. 点击**确定**。
7. 选择**图形** > **图形生成器**。
8. 将**行政区域**拖放到**地图形状**区域中。

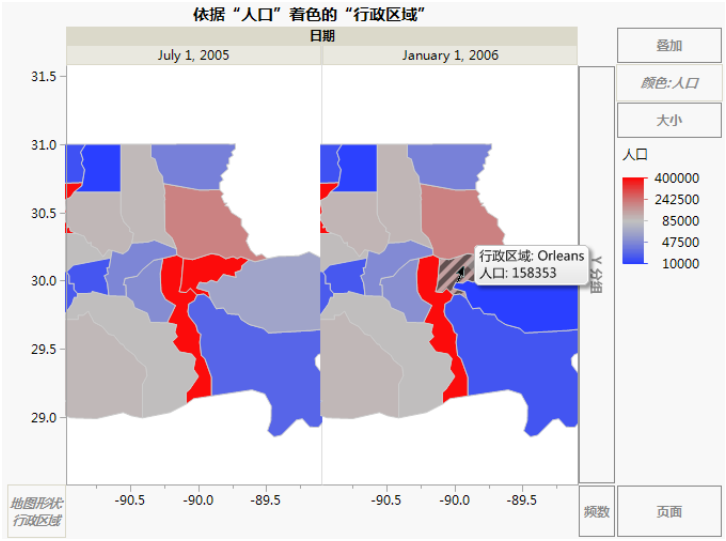
   地图自动显示，因为您已经使用定制地图文件定义了**行政区域**列。
9. 将**人口**拖放到**颜色**区域中。
10. 将**日期**拖放到 **X 分组**区域中。

图 12.20 卡特里娜飓风经过前后各行政区域的人口



11. 选择**放大镜**工具放大两个地图中的 Orleans 行政区域 (图 12.21)

图 12.21 Orleans 行政区域



您可以清晰地看到卡特里娜飓风过后人口有所下降。Orleans 行政区域的人口从 2005 年 7 月的 437,186 减至 2006 年 1 月的 158,353。

从您的数据中直接指向现有的地图文件

假定您已有定制地图文件且它们已正确命名。您的地图文件为 US-MSA-Name.jmp 和 US-MSA-XY.jmp。它们保存在 Sample Data 文件夹中。

PopulationByMSA.jmp 数据表包含美国大城市统计区域 (MSA) 2000 年和 2010 年的人口数据。本例显示如何设置数据表以创建地图。

添加 “地图角色” 列属性

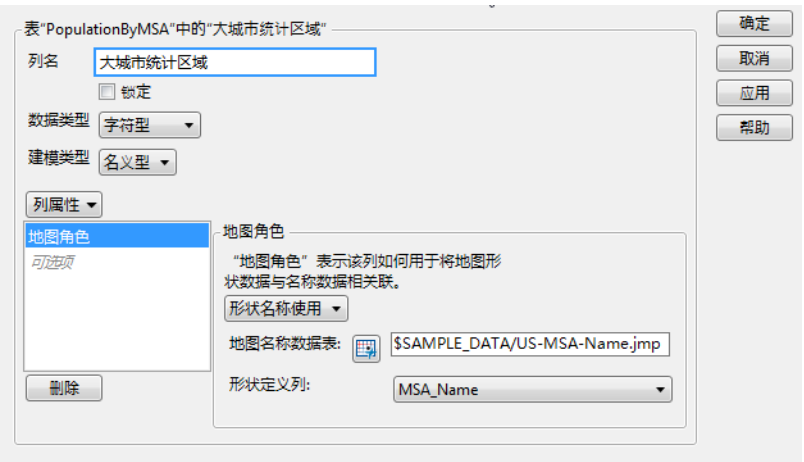
1. 选择**帮助** > **样本数据库**，然后打开 PopulationByMSA.jmp。
2. 右击**大城市统计区域**列并选择**列信息**。
3. 选择**列属性** > **地图角色**。
4. 选择**形状名称使用**。
5. 在**地图名称数据表**旁边，键入 $SAMPLE_DATA/US-MSA-Name.jmp.。

   这将为 JMP 指示包含地图信息的数据表所在的位置。
6. 从**形状定义列**列表中选择 MSA_Name。

MSA_Name 是 US-MSA-Name.jmp 数据表中包含每个大城市统计区域的唯一名称的特定列。请注意，已为 MSA_Name 列指定**形状名称定义地图角色**属性，作为正确定义地图文件的一部分。

注意：请记住， -Name 数据表中的 Shape ID （形状 ID）列映射到 -XY 数据表中的 Shape ID（形状 ID）列。这意味着指示 -Name 数据表所在的位置即可将其链接至 -XY 数据表，这样 JMP 即具备创建地图所需的所有内容。

图 12.22 “地图角色”列属性



7. 点击**确定**。

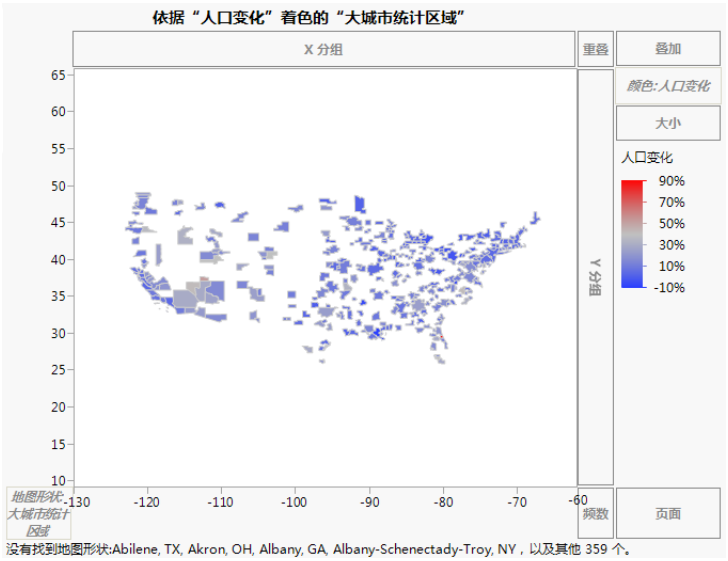
在“图形生成器”中创建地图

一旦设置了**地图角色**列属性，即可执行分析。您想要直观地查看 2000 年与 2010 年之间美国大城市统计区域中的人口变化。

1. 选择**图形 > 图形生成器**。
2. 将**大城市统计区域**拖放到**地图形状**区域中。

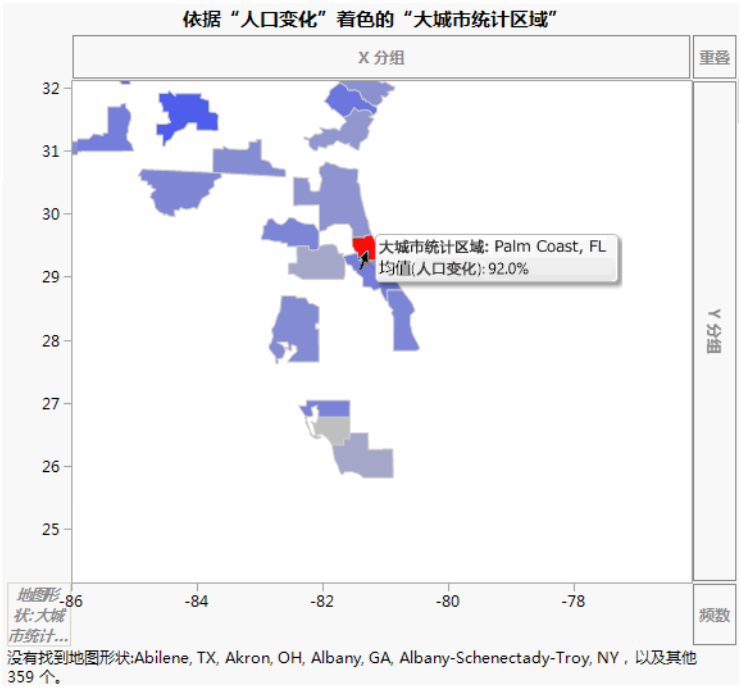
   由于您已针对该列定义了**地图角色**列属性，此时会显示地图。
3. 将**人口变化**拖放至**颜色**区域。

图 12.23 大城市统计区域的人口变化



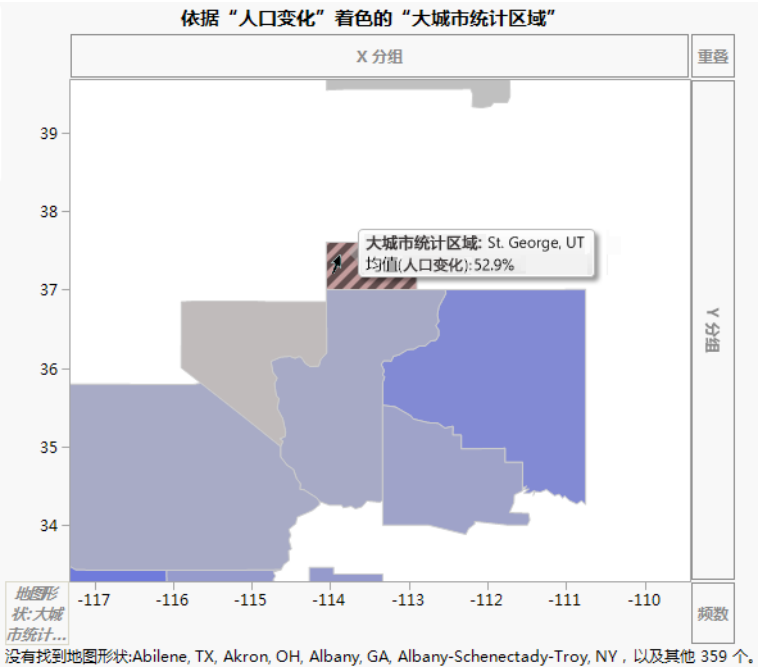
4. 选择**放大镜**工具放大 Florida 州。
5. 选择**箭头**工具并点击红色区域。

图 12.24 Florida 州 Palm Coast 的人口变化



6. 选择**放大镜**工具，并在按住 Alt 键的同时点击地图将其缩小。
7. 选择**放大镜**工具放大 Utah 州。
8. 选择**箭头**工具并点击浅红色区域。

图 12.25 Utah 州 St. George 的人口变化



您可以看到 Florida 州 Palm Coast 地区和 Utah 州 St. George 地区在 2000 年和 2010 年之间的人口变化最大。Palm Coast 地区的人口增长了 92%，St. George 地区的人口增长约 53%。

飓风跟踪示例

该示例使用 Hurricanes.jmp 样本数据表，其中包含影响美国东海岸的飓风的数据。添加背景地图可以帮助您查看飓风影响的区域。针对该示例开发了脚本，它是数据表的一部分。

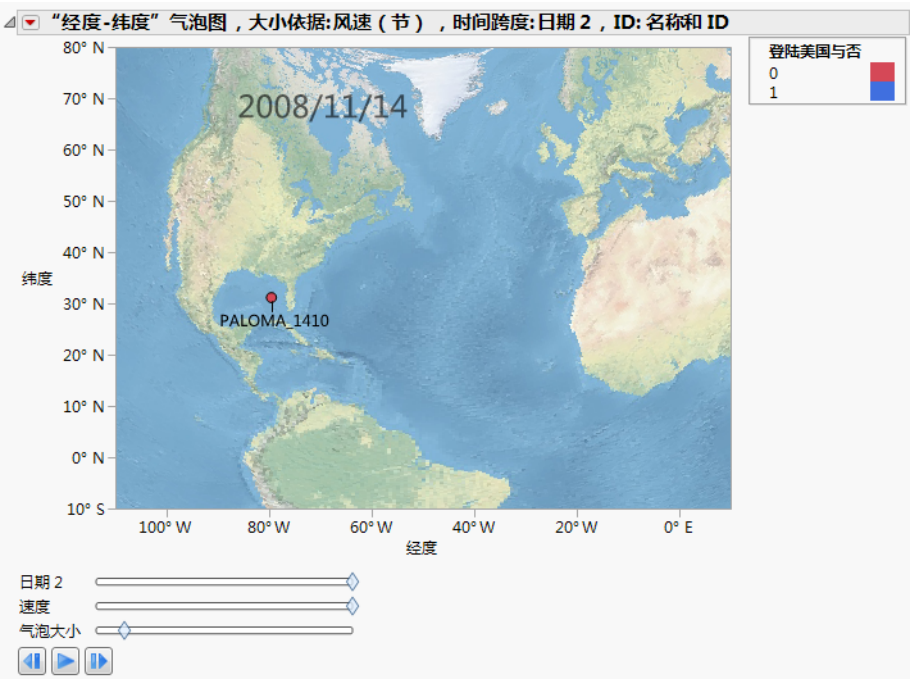
1. 选择**帮助 > 样本数据库**，然后打开 Hurricanes.jmp。
2. 在 “表” 面板中，点击**气泡图**脚本旁边的绿色小三角。
3. 将 “日期” 滑块拖至右侧，如图 12.26 中所示。
4. 点击红点显示飓风名。日期显示在窗口左上角的位置。红点显示飓风 Paloma 在 2008 年 11 月 14 日的位置。

图 12.26 Hurricanes.jmp 的气泡图

请注意，即使标绘了飓风的位置，但实际上并没有告知它的确切地点。轴信息显示北纬 27° 以及西经 86°，但我们需要更多一点的上下文信息。很有可能在大西洋中部上方，但它是在一个小岛的上方吗？这很重要，尤其是对小岛上的居民来说。显然在图形的背景中添加一个地图会得到很多有用的信息。

5. 右击图形并选择**背景地图**。“设置背景地图”窗口随即显示（图 12.12）。
6. 选择**精细地球**并点击**确定**。

图 12.27 Hurricanes.jmp 带背景地图的气泡图

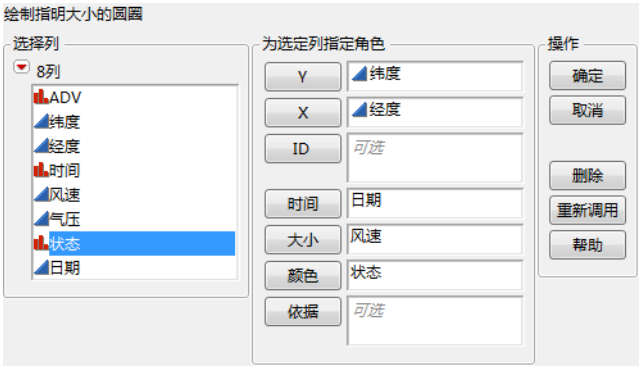


现在坐标在地理上显得有意义。点击**运行**可查看飓风数据随背景地图移动的动画。尝试使用不同的选项并查看显示。调整轴或使用缩放工具可以更改您所查看的世界部分。地图会随着视图有所调整。您还可以右击图形并选择“大小/尺度->调整为相同大小”以使图形的纵横比再一次成比例。

下一个示例使用 Katrina Data.jmp 样本数据表，其中包含有关卡特里娜飓风的数据，比如纬度、经度、日期、风速、气压和状态。添加背景地图可以帮助您查看飓风发生的路线以及基于它的规模和强度对陆地产生的影响。针对该示例开发了脚本，它是数据表的一部分。

1. 选择**帮助** > **样本数据库**，然后打开 Katrina Data.jmp。
2. 选择**图形** > **气泡图**。
3. 选择**纬度**并点击 Y。
4. 选择**经度**并点击 X。
5. 选择**日期**并点击**时间**。
6. 选择**风速**并点击**大小**。
7. 选择**状态**并点击**颜色**。

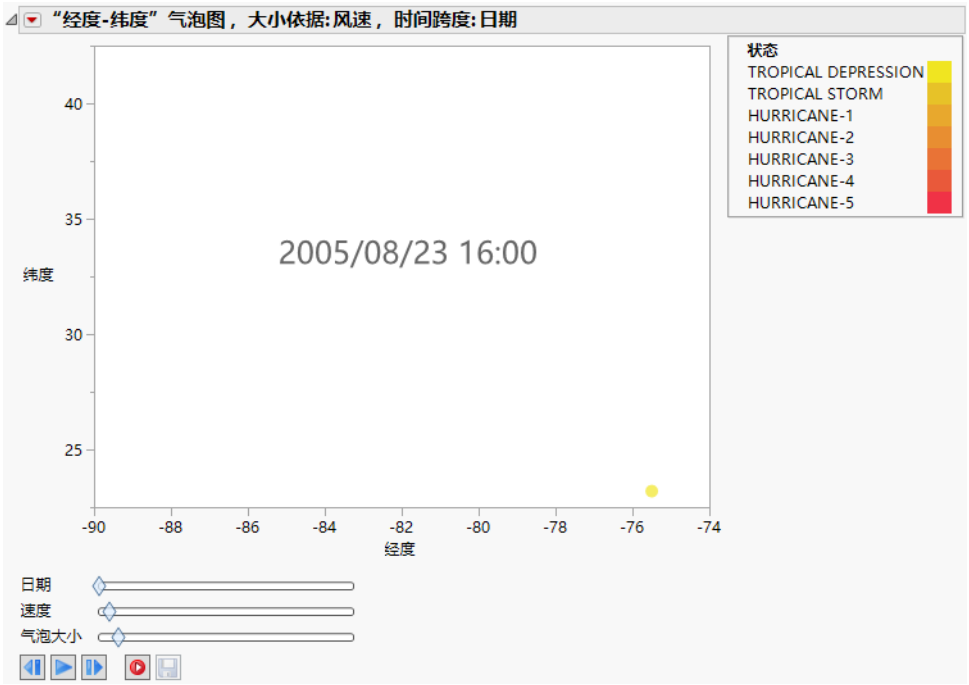
图 12.28 Katrina Data.jmp 的气泡图设置



8. 点击**确定**。

将出现以下图像。黄点显示了 2005 年 8 月 23 日热带低气压卡特里娜飓风的位置。

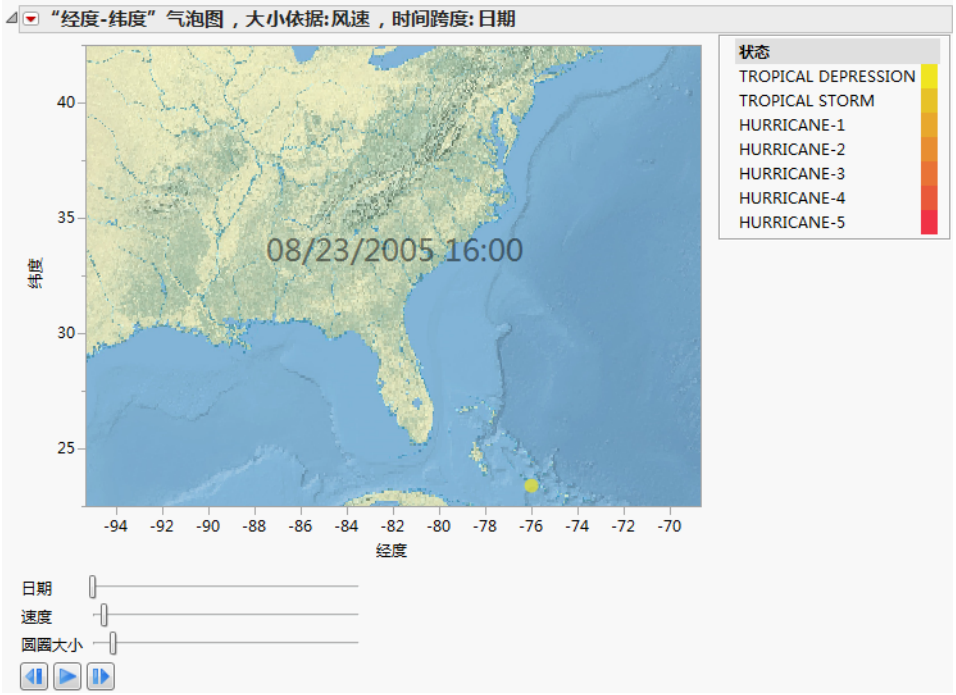
图 12.29 Katrina Data.jmp 的气泡图



请注意，即使标绘了暴风雨的位置，但实际上并没有告知它在什么地方。要添加更多上下文，请在背景中添加地图。

9. 右击图形并选择**背景地图**。“设置背景地图”窗口随即显示。
10. 选择**精细地球**并点击**确定**。

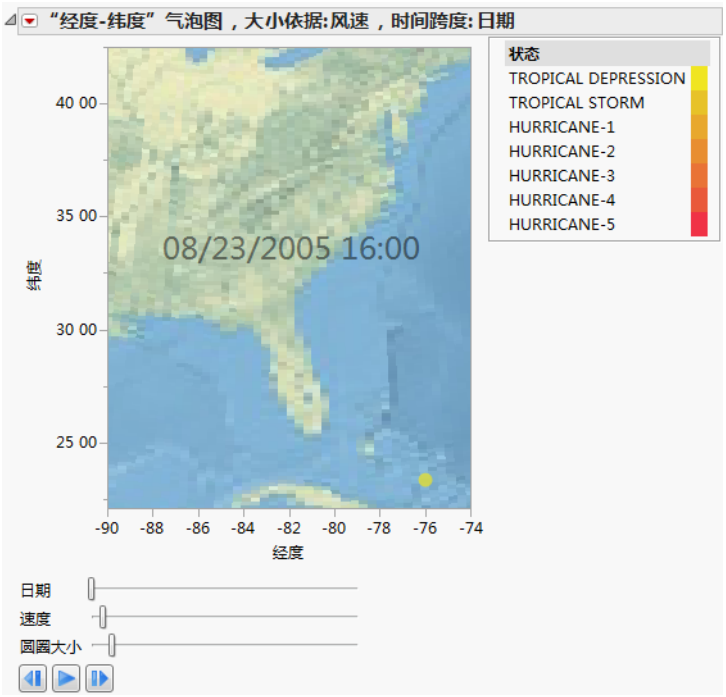
图 12.30 Katrina Data.jmp 带背景地图的气泡图



现在坐标在地理上显得有意义。您可以编辑轴和大小 / 尺度以更改图形显示的方式。

11. 右击 X 轴 （经度）并选择**轴设置**。“X 轴规格”窗口随即显示。
12. 选择**尺度** > **测地线 US**。
13. 选择**格式** > **地理** > **经度 DMM**。
14. 点击**确定**。
15. 为 Y 轴 （纬度）重复相同的操作，唯一不同的一项操作是选择**格式** > **地理** > **纬度 DMM**。
16. 右击地图并选择**大小 / 尺度** > **调整为相同大小**。

图 12.31 Katrina Data.jmp 带背景地图的气泡图



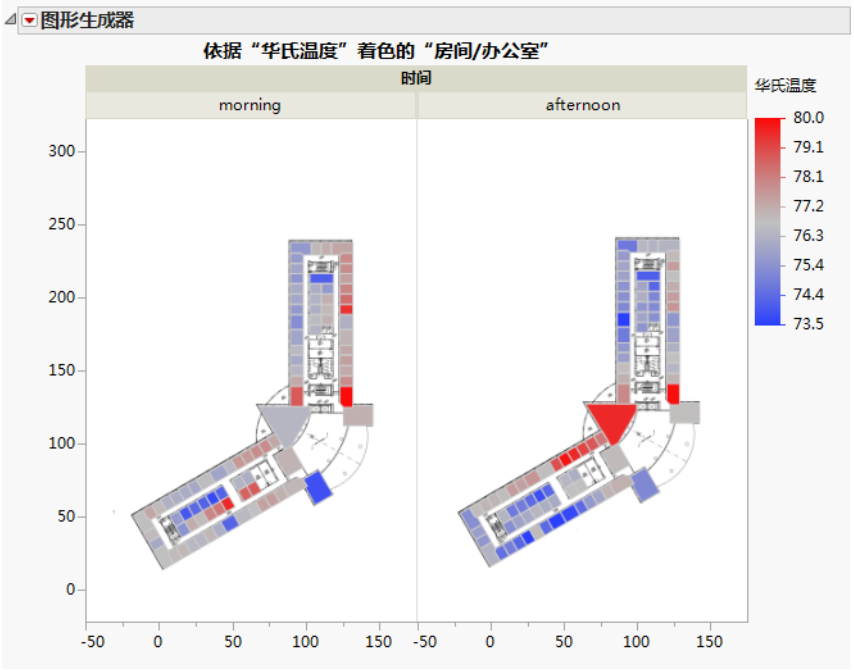
点击**运行**可查看飓风数据随背景地图移动的动画。您可以操控速度和气泡大小。尝试使用不同的选项并查看显示。调整轴或使用缩放工具可以更改您所查看的区域部分。为州添加边界。地图会随着视图有所调整。

办公室温度研究

该示例展示了创建定制背景地图进行办公室温度研究，以及如何使用 JMP 展示结果。示例中收集了某一建筑物楼层内各办公室温度的数据。使用“JMP 文件交换库”(https://community.jmp.com/docs/DOC-6218) 中的 Custom Map Creator 插件创建了楼层的地图。使用“图形生成器”，办公室温度结果随后会得到直观地分析。

下面显示的地图是按**时间**分组的楼层图。颜色反映华氏温度值。按这种方式直观地探索数据可以提示出哪些因素会影响办公室的温度。查看该地图，可以看出建筑物东侧的办公室早上比下午的温度高。建筑物西侧的办公室则出现与此刚好相反的结论。根据这个可视化结果，可以预期这两个变量会影响办公室温度，或许这些项之间的交互作用是显著的。这些直观图形有助于在分析过程中制定决策。

图 12.32 依据 “时间”分组，按 “华氏温度”着色的 “房间 / 办公室”



首先，收集数据并输入到数据表 (S4 Temps.jmp)。请注意**房间/办公室**列。它包含每间办公室的唯一名称，并指定了“地图角色”以正确定义地图文件。

图 12.33 S4 Temps.jmp 数据表的部分视图

14/0 列	日	时间	温度计	部门	房间/办公室
1	1	morning	T4	sector 1	S4068
2	1	morning	T4	sector 1	S4069
3	1	morning	T4	sector 1	S4070
4	1	morning	T4	sector 1	S4072
5	1	morning	T4	sector 1	S4074
6	1	morning	T4	sector 1	S4076
7	1	morning	T4	sector 1	S4078
8	1	morning	T4	sector 1	S4082
9	1	morning	T4	sector 1	S4084
10	1	morning	T4	sector 1	S4086
11	1	morning	T4	sector 1	S4088
12	1	afternoon	T2	sector 1	S4068
13	1	afternoon	T2	sector 1	S4069

然后，使用可从“JMP 文件交换库”（网址为 https://community.jmp.com/docs/DOC-6218）下载的 Custom Map Creator 插件创建了楼层的地图。该插件会创建两个表以定义形状；XY 表和“Name”表。下面的说明介绍了如何构建。

创建楼层地图

1. 通过以下菜单项启动插件：**插件** > **地图形状** > **定制地图创建程序**。背景中打开两个表，然后出现“定制地图创建程序”窗口。
2. 将背景图像拖放到图形框架中。此时楼层平面图的图像可用。
3. 对背景图像执行任意大小调整并对框架绘图。
4. 对表命名 （例如 S4）。
5. 点击**下一步**。
6. 命名您将定义的形状。在该示例中，针对地图分别命名了每间办公室 （例如 S4001）。
7. 在图形框架内，使用鼠标点击您想定义的形状的所有边界。将出现一条线，连接所有边界点。
8. 一旦您完成定义形状的边界，即可点击**下一个形状**。继续添加形状，直到您完成楼层平面图为止。请注意，您不需要连接最后一个边界点；当您点击**下一个形状**时插件将自动进行连接。
9. 可以更改线条粗细和颜色。此外，选中**填充形状**将使用随机颜色填充每个形状。
10. 点击**完成**。

创建并正确命名了定制地图文件。地图文件为 S4-Name.jmp 和 S4-XY.jmp，它们保存在 JMP Samples/Data 文件夹中。

添加“地图角色”列属性

注意：本节中的路径名是指 JMP 文件夹。在 Windows 的 JMP Pro 中，JMP 文件夹命名为 JMPPro。

S4 Temps.jmp 数据表包含三天的办公室数据。在数据表中设置**地图角色**列属性：

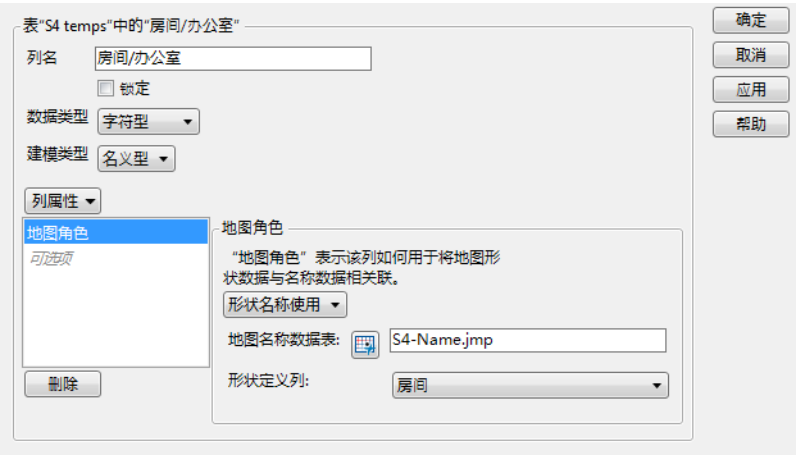
1. 选择**帮助** > **样本数据库**，然后打开 S4 Temps.jmp。
2. 右击**房间 / 办公室**列并选择**列信息**。
3. 选择**列属性** > **地图角色**。
4. 选择**形状名称使用**。
5. 点击**地图名称数据表**旁边的图标，浏览至 S4-Name.jmp 文件 （位于 JMP Samples/Data 文件夹中）。

   这将为 JMP 指示包含地图信息的数据表所在的位置。
6. 从**形状定义列**列表中选择**房间**。

   房间是 S4-Name.jmp 数据表中的特定列，其中包含每间办公室的唯一名称。请注意，已为**房间**列指定**形状名称定义地图角色**属性，作为正确定义地图文件的一部分。

注意：请记住， -Name 数据表中的 Shape ID （形状 ID）列映射到 -XY 数据表中的 Shape ID（形状 ID）列。这意味着指示 -Name 数据表所在的位置即可将其链接至 -XY 数据表，这样 JMP 即具备创建地图所需的所有内容。

图 12.34 “地图角色”列属性



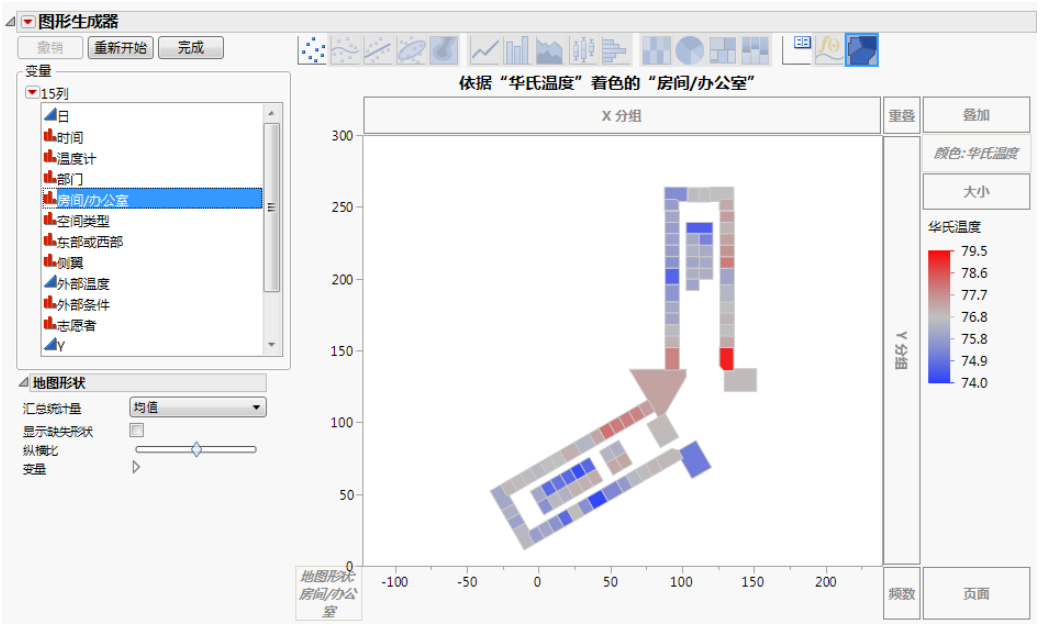
7. 点击**确定**。

一旦设置了**地图角色**列属性，即可执行分析。您想要直观地查看整个楼层各办公室温度的差异。

1. 选择**图形** > **图形生成器**。
2. 将**房间 / 办公室**拖放到**地图形状**区域中。

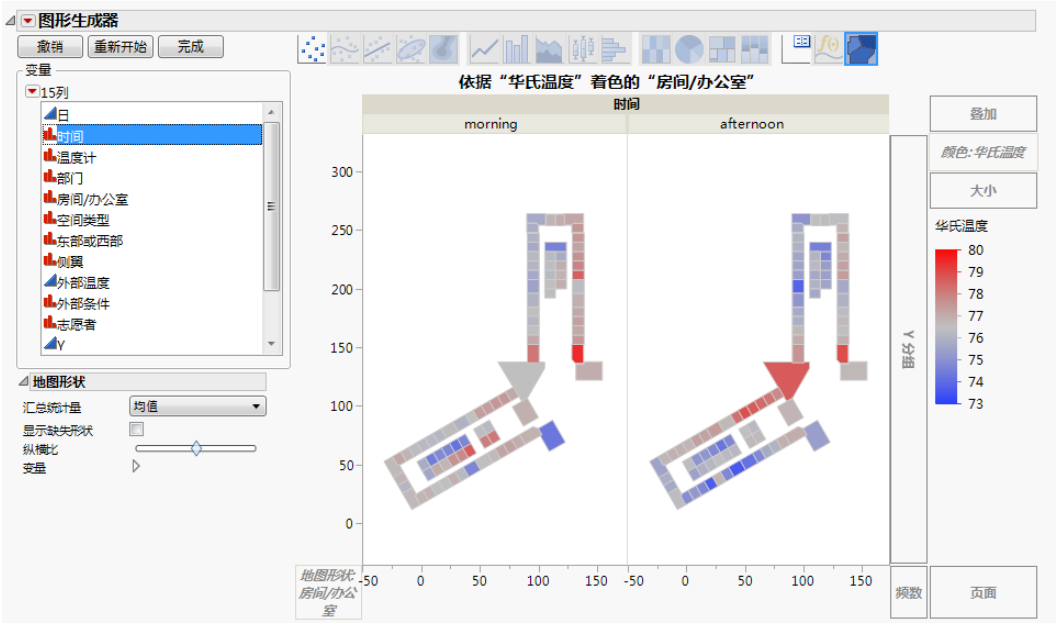
   由于您已针对该列定义了**地图角色**列属性，此时会显示地图。
3. 将**华氏温度**拖放到**颜色**区域中。

图 12.35 按 “华氏温度”着色的 “房间 / 办公室”



4. 将**时间**拖放到 “X 分组”区域。

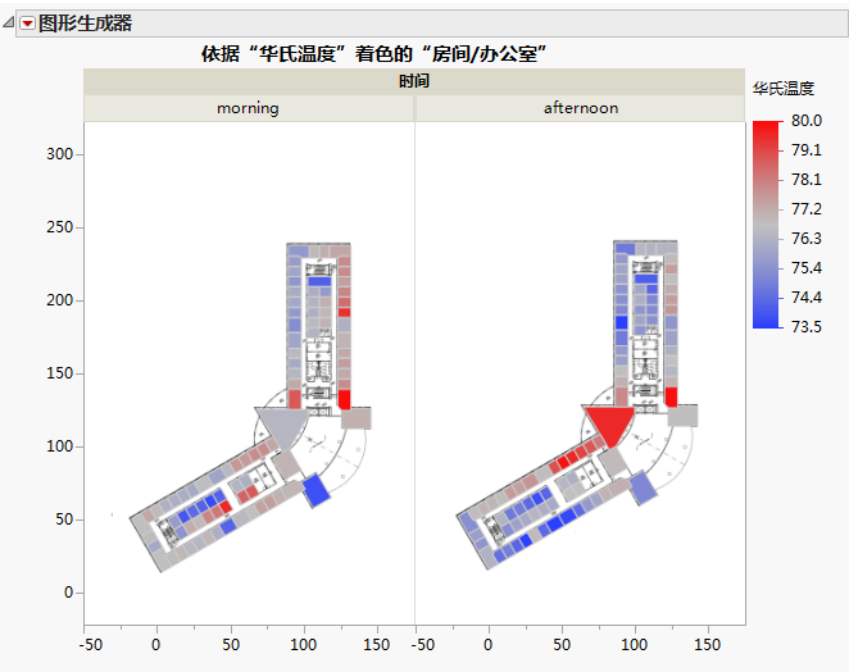
图 12.36 依据 “时间”分组，按 “华氏温度”着色的 “房间 / 办公室”



请注意，只有那些属于研究一部分并且使用 Custom Map Creator 插件创建的办公室才会显示出来。要添加整个楼层平面图的图像，原始的楼层平面图被拖放到“图形生成器”窗口上方，从而创建了图 12.37。

要查看图 12.37，选择**帮助 > 样本数据库**并打开 S4 Temps.jmp，然后运行**按一天中的时间**脚本。

图 12.37 带原始楼层平面图的房间 / 办公室图

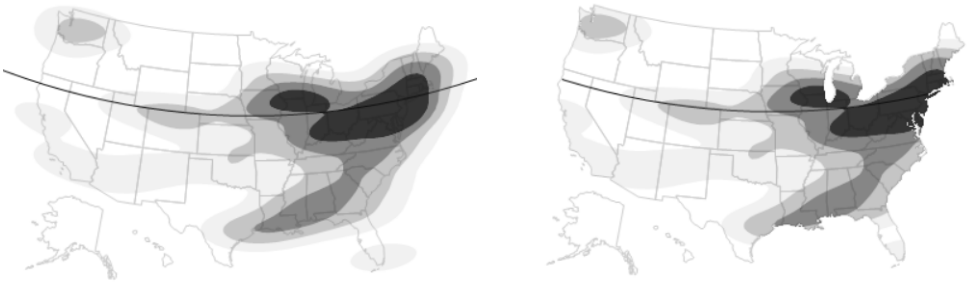


随数据表一起提供的还有一些脚本，可以运行这些脚本以查看各种分析和建模，而且可以执行并直观地显示分析和建模。

包含剪切元素的边界地图

可以剪切图形元素（显示参考线、网格线和等高线之类的线段），以便符合地图的边界。“定制图形”窗口提供“剪切形状”选项，您可以在其中选择围绕其剪切形状的边界。

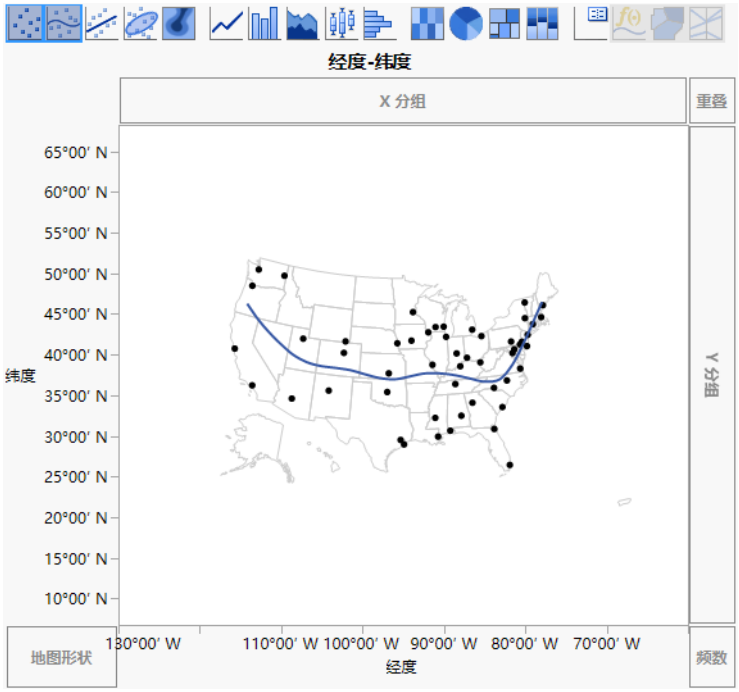
图 12.38 未剪切 （左）和已剪切 （右）的参考线和等高线



创建背景地图

1. 选择**帮助** > **样本数据库**，然后打开 Cities.jmp。
2. 选择**图形** > **图形生成器**。
3. 将**纬度**列拖至 Y 轴。
4. 将**经度**列拖至 X 轴。
5. 右击图形并选择**图形** > **背景地图**。
6. 在 “边界”列中，选择**美国各州**并点击**确定**。
   美国地图随即显示在图形上。

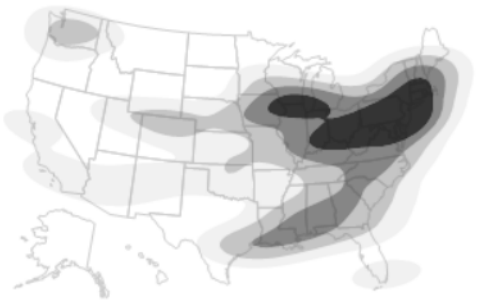
图 12.39 背景地图



7. 点击 “点” 图标 以删除各点。
8. 点击 “平滑线” 图标 以删除平滑线。
9. 点击 “等高线” 图标 。

   等高线图随即显示在地图上。

图 12.40 背景地图上的未剪切等高线



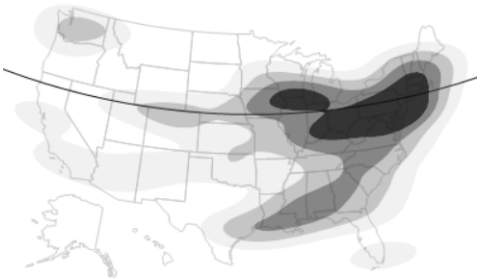
添加参考线

1. 在纬度轴上，双击 45°00′ N 以打开 “Y 轴设置” 窗口。
2. 在 “参考线” 窗格中，将 “值” 框中的第一个数字改为 “40”。

例如，在 44°40.11′ N 中，将 “44” 改为 “40”。

3. 依次点击**添加**和**确定**。

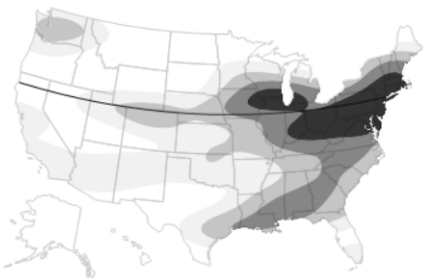
图 12.41 未剪切的参考线



分配剪切形状

1. 右击图形并选择**定制**。
2. 选择**参考线**并从 “剪切形状” 列表中选择**美国各州**。
3. 选择**等高线**，再从 “剪切形状” 列表中选择**美国各州**，然后点击**确定**。

图 12.42 已剪切的参考线和等高线



提示：

- “背景地图” 窗口中显示的边界安装在 JMP Maps 安装文件夹中。请参见第 235 页的 “定制地图文件”，了解有关自行添加边界的详细信息。
- 脚本可提供其他选项，如：在矩阵或字符串中指定剪切路径。请参见《Scripting Guide》。

附录 A

传统图形平台

以下传统图形平台已经废弃。废弃的平台指的是早期平台，其中包含的功能可在更新平台中访问。本附录中的图表和图形在“图形生成器”平台中均提供：这些传统图形平台在 JMP 的未来版本中可能不可用。

表 A.1 传统图形平台

平台	图形生成器参考
矩形树图	• “图形生成器”一章中第 80 页的“矩形树图”
图表	• “图形生成器示例”中的条形图示例 • “条形图”、“线”或“图形生成器”一章中第 77 页的“使用热图应用背景颜色”
叠加图	• “图形生成器示例”一章中第 125 页的“叠加直方图和脊线图的示例” • 第 25 页的“图形生成器”一章

矩形树图

矩形树图对于观测有很多水平的组中的模式很有用。当您的数据包含很多类别时，可以使用矩形树图来直观表示很多组。

提示：要创建矩形树图，请考虑使用图形生成器。请参见“图形生成器”一章中第 80 页的“矩形树图”。

通过选择**图形** > **传统图形** > **矩形树图**可启动 “矩形树图”平台。

当您在启动窗口中点击**确定**后，默认情况下会在“图形生成器”中创建一个矩形树图。要在传统“矩形树图”平台中生成报表，您必须首先取消选择“文件”>“首选项”>“平台”>“矩形树图”下的“使用图形生成器”平台首选项。有关传统“矩形树图”平台启动窗口、报表和选项的文档可在以下位置找到：
https://www.jmp.com/support/help/en/15.2/#page/jmp/legacy-platforms.shtml。

图表

图形菜单上的“图表”平台用于绘制分类变量与连续变量的关系图。针对每个分类水平汇总连续变量。

若要绘制各个数据点（而不是数据点汇总）的图形，我们建议改用叠加图。

提示：要创建图表，请考虑使用图形生成器。请参见“图形生成器示例”一章中第 125 页的“叠加直方图和脊线图的示例”。

通过选择**图形** > **传统图形** > **图表**可启动 “图表”平台。

当您在启动窗口中点击**确定**后，默认情况下会在“图形生成器”中创建一个图表。要在传统“图表”平台中生成报表，您必须首先取消选择“文件”>“首选项”>“平台”>“图表”下的“使用图形生成器”平台首选项。有关传统“图表”平台启动窗口、报表和选项的文档可在以下位置找到：https://www.jmp.com/support/help/en/15.2/#page/jmp/legacy-platforms.shtml。

叠加图

“叠加图”选项可以对单个 X 列和一个或多个数值 Y 绘图，但不接受非数值的 y 轴值。

提示：要创建叠加图，考虑使用“图形生成器”中的“叠加”区域。请参见“图形生成器示例”一章中第 125 页的“叠加直方图和脊线图的示例”。

通过选择**图形** > **传统图形** > **叠加图**可启动 “叠加图”。

当您在启动窗口中点击**确定**后，默认情况下会在“图形生成器”中创建一个叠加图。要在传统“叠加图”平台中生成报表，您必须首先取消选择“文件”>“首选项”>“平台”>“叠加图”下的“使用图形生成器”平台首选项。有关传统“叠加图”平台启动窗口、报表和选项的文档可在以下位置找到：
https://www.jmp.com/support/help/en/15.2/#page/jmp/legacy-platforms.shtml。

附录 B

参考书目

Aitchison, J. (1986).*The Statistical Analysis of Compositional Data*.London: Chapman & Hall.

Box, G. E. P., Hunter, W. G., and Hunter, J. S. (1978).*Statistics for Experimenters*.New York: John Wiley & Sons.

Cornell, J. A. (1990).*Experiments with Mixtures*.2nd ed. New York: John Wiley & Sons.

Few, Stephen (2008).“Dual-Scaled Axes in Graphs.”*Visual Business Intelligence Newsletter*. http://www.perceptualedge.com/articles/visual_business_intelligence/dual-scaled_axes.pdf.

Hyndman, R. J. (1996).“Computing and Graphing High Density Regions.”*The American Statistician* 50: 120 - 126.

Hyndman, R. J., Koehler, A. B., Ord, J. K., and Snyder, R. D. (2008).*Forecasting with Exponential Smoothing: The State Space Approach*.Berlin: Springer-Verlag.

Inselberg, A. (1985).“The Plane with Parallel Coordinates.”*Visual Computing* 1: 69–91.

Mardia, K. V., Kent, J. T., and Bibby J. M. (1979).*Multivariate Analysis*.New York: Academic Press.

Rousseeuw, P. J., Ruts, I., and Tukey, J. W. (1999).“The Bagplot: A Bivariate Boxplot.”*The American Statistician* 53: 382 - 387.

Wegman, E. J. (1990).“Hyperdimensional Data Analysis using Parallel Coordinates.”*Journal of the American Statistical Association* 85: 664–675.

附录 C

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

刻画器指南

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2021-2020. JMP® 16 刻画器指南. Cary, NC: SAS Institute Inc.

JMP® 16 刻画器指南

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

刻画器指南

1 了解 JMP 11

文档和其他资源

JMP 文档中的格式规范 13

JMP 帮助 14

JMP 文档库 14

学习 JMP 的其他资源 20

JMP 教程 20

样本数据表 20

学习统计和 JSL 术语 21

学习 JMP 的提示与技巧 21

JMP 工具提示 21

JMP 用户社区 21

免费在线统计思维课程 21

JMP 新用户欢迎套件 22

统计学知识门户 22

JMP 培训 22

用户编写的 JMP 手册 22

“JMP 起始页”窗口 22

JMP 技术支持 22

2 刻画器介绍 23

直观演示响应曲面和优化过程

“刻画器”平台概述 25

刻画介绍 25

JMP 中的刻画器功能 26

“刻画器”启动窗口 28

拟合组 29

解释刻画 29

“刻画器”平台选项 33
常见刻画器主题 34
线性约束 34
噪声因子 35

3 刻画器 37
探索每个因子的响应横截面

预测刻画器的示例 39
启动“预测刻画器”平台 40
“预测刻画器”选项 40
“最大化选项”窗口 47
意愿刻画和优化 47
意愿函数的构造 48
如何使用意愿函数 49
意愿刻画 50
定制意愿函数 51
评估变量重要性 52
Bagging 55
预测刻画器的其他示例 56
多重响应的意愿刻画示例 56
预测刻画器中噪声因子的示例 58
一个响应的变量重要性的示例 62
多重响应的变量重要性的示例 64
Bagging 改进预测的示例 67
指示预测准确度的 Bagging 示例 70
预测刻画器的统计详细信息 73
评估变量重要性 73
误差传播直条 74
外推控制量度 75

4 等高线刻画器 79
探索两个因子的响应等高线

“等高线刻画器”概述 81
等高线刻画器的示例 81
启动“等高线刻画器”平台 83

“等高线刻画器”报表 83
因子设置和控件 83
响应设置和控件 84
“等高线刻画器”平台选项 84
约束着色设置 86
“等高线刻画器”的其他示例 86
探索最优设置 86

5 曲面图 91
探索三个因子的响应等高线

“曲面图”和“曲面刻画器”的概述 93
“曲面图”平台的示例 93
启动“曲面图”平台 96
“曲面图”报表 98
曲面图 98
外观控件 99
自变量 100
因变量 101
“曲面图”平台选项 102
弹出式菜单选项 103
因变量选项 104
面或曲面属性 104
键盘快捷方式 105
“曲面图”平台的更多示例 106
为单个数学函数构造曲面图 106
标绘等值面 107

6 混料刻画器 109
使用三元图探索因子效应

“混料刻画器”概述 111
三元图概述 111
包含三个以上成分的三元图 111
混料刻画器的示例 113
启动“混料刻画器”平台 114

“混料刻画器”报表 115
因子设置和控件 115
响应设置和控件 116
“混料刻画器”平台选项 116
混料刻画器的定制 117
线性约束 117
“混料刻画器”平台的其他示例 118
具有过程变量的混料变量 118
具有五个混料变量的多个响应 122

7 定制刻画器 129
使用数字计算器探索响应曲面

“定制刻画器”概述 131
定制刻画器的示例 131
启动“定制刻画器”平台 133
“定制刻画器”报表 134
因子设置和控件 134
响应设置和控件 134
优化设置和控件 135
“定制刻画器”平台选项 135

8 模拟器 137
了解变异对响应的影响

模拟器概述 139
模拟器的示例 139
启动“模拟器” 141
因子模拟设置 141
响应模拟设置 143
“模拟器”报表 144
“模拟器”报表选项 144
缺陷刻画器 145
容差设计简介 145
图形尺度 146
缺陷率 146

模拟方法和详细信息 147
注意 147
缺陷参数刻画器 147
模拟实验 148
规格限 148
模拟器的其他示例 149
缺陷刻画器的示例 149
随机优化的示例 152
模拟常规公式的示例 160
模拟器的统计详细信息 163
正态加权分布 163

9 Excel 刻画器 165
直观演示 Microsoft Excel 中保存的模型
“Excel 刻画器”概述 167
Excel 模型的示例 167
运行 JMP 刻画器 169
使用线性约束 169
刻画线的分辨率 170
从 JMP 使用 Excel 刻画器 170

A 参考书目 171
B 技术许可声明 173

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 13
JMP 帮助 14
JMP 文档库 14
学习 JMP 的其他资源 20
JMP 教程 20
样本数据表 20
学习统计和 JSL 术语 21
学习 JMP 的提示与技巧 21
JMP 工具提示 21
JMP 用户社区 21
免费在线统计思维课程 21
JMP 新用户欢迎套件 22
统计学知识门户 22
JMP 培训 22
用户编写的 JMP 手册 22
“JMP 起始页”窗口 22
JMP 技术支持 22

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 14 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

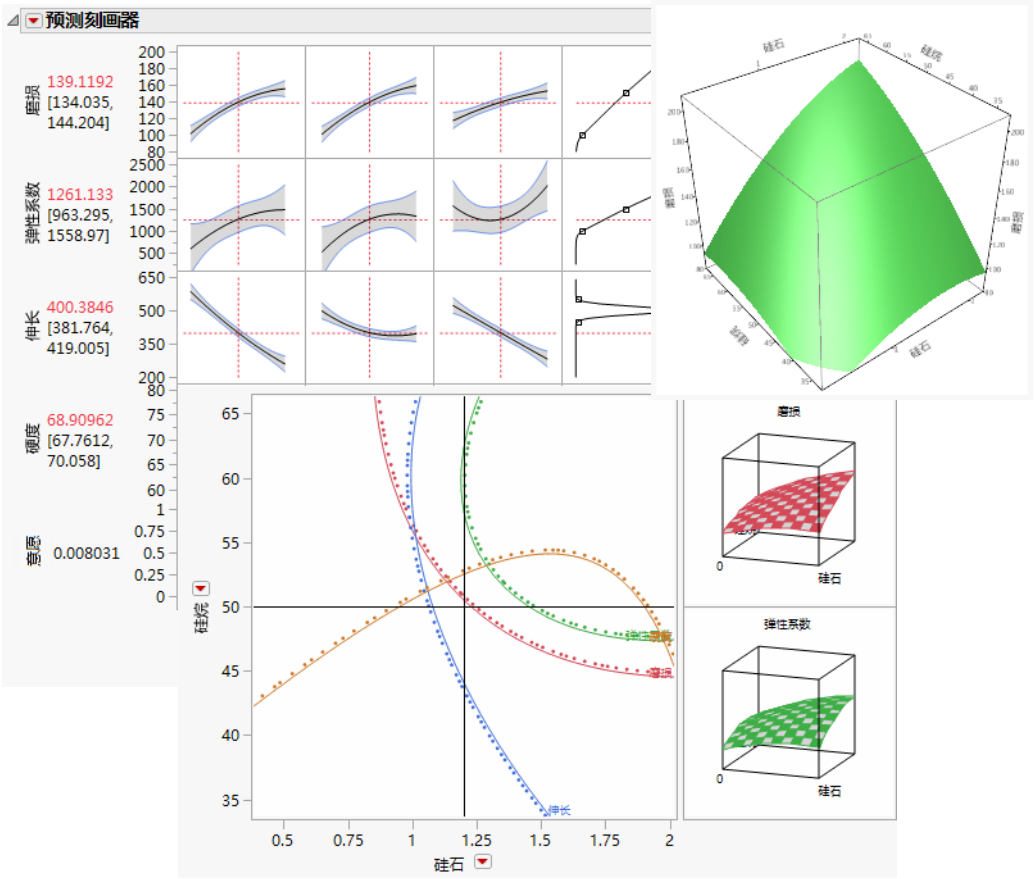
第 2 章

刻画器介绍

直观演示响应曲面和优化过程

刻画是一种方法，它可以通过观察一次仅更改一两个因子时会发生的变化来直观演示响应曲面。实际上，刻画图就是一个横截面视图。JMP 中的交互式刻画器拓宽了探索机会的空间。在用方程拟合数据的过程中，拟合只是第一步。人们还希望解释拟合、理解拟合响应曲面以及发现优化响应的因子值。

图 2.1 刻画器示例



目录

“刻画器”平台概述 25
刻画介绍 25
JMP 中的刻画器功能 26
“刻画器”启动窗口 28
拟合组 29
解释刻画 29
“刻画器”平台选项 33
常见刻画器主题 34
线性约束 34
噪声因子 35

“刻画器”平台概述

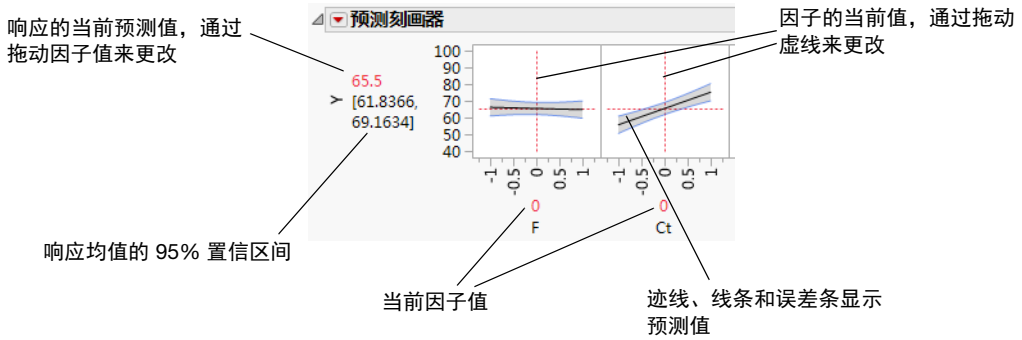
“预测刻画器”显示各个 X 变量的刻画迹线 （图 2.2）。**刻画迹线**是在一个变量发生变化而其他变量一直保持当前值的情况下的预测响应。“预测刻画器”可在您更改 X 变量值后重新计算刻画值和预测响应 （实时）。

- 每个 X 变量的垂直虚线显示其**当前值**或**当前设置**。若变量为名义型变量，则 X 轴标识类别。

  对于每个 X 变量，因子名称上方的值即其当前值。通过在图形中点击或通过将虚线拖至您所需的新当前值的位置，可以更改当前值。
- 水平虚线显示对应于 X 变量的当前值的每个 Y 变量的**当前预测值**。
- 图中的黑线显示当您更改各个 X 变量的当前值时，预测值如何变化。在拟合平台中，预测值的 95% 置信区间是使用环绕预测迹线的蓝色实曲线（对于连续变量）或是误差条的高度（对于分类变量）来表示的。对于连续变量，将置信区间区域着色。

“预测刻画器”是一次更改一个变量来观察其对预测响应的影响的一种方法。

图 2.2 迹线的演示



“预测刻画器”在某些情况下会计算每个刻画列的置信区间。若为同一列既保存了标准误差公式又保存了预测公式，“预测刻画器”会使用标准误差生成置信区间而不是将其刻画为单独的列。

刻画介绍

使用一个输入因子 X 和一个输出因子 Y 很容易直观演示响应曲面。因子和响应数量越多就越难演示。 JMP 中的刻画器可提供任意响应曲面的若干高度交互式横截面视图。在本指南中，我们使用以下可互换的术语：

- 因子、输入变量、X 列、自变量、设置、项

- 响应、输出变量、*Y* 列、因变量、结果

意愿刻画和优化功能有助于发现适合的因子设置并生成满意的响应。多数刻画器还支持多线程处理以加快计算。在因子存在变异而您需要稳健和高质量的响应时，可利用模拟和缺陷刻画功能。

- “JMP 中的刻画器功能”
- “‘刻画器’启动窗口”
- “拟合组”

JMP 中的刻画器功能

JMP 中提供了若干刻画器工具，可通过若干拟合平台和“图形”下的主菜单来访问这些工具。这些工具用于刻画数据列公式。

表 2.1 刻画器功能汇总

	说明	功能
预测刻画器	显示各因子的垂直切片，令其他因子保持当前值	意愿，优化，模拟器，误差传播
等高线刻画器	水平切片，可同时为两个因子显示等高线	模拟器
曲面刻画器	同时为两个因子显示三维响应图，或同时为三个因子显示等高线曲面图	曲面可视化
混料刻画器	混料因子的等高线刻画器	三元图和等高线
定制刻画器	非图形化刻画器和数值优化器	常规优化，模拟器
Excel 刻画器	直观演示存储在 Excel 工作表中的模型（或公式）。	使用 Excel 模型刻画

表 2.2 显示了刻画器的可用性。“定制刻画器”仅可通过“图形”菜单来使用。（在“模型比较”中**不能**使用“定制刻画器”。）

表 2.2 JMP 刻画器的提供位置

位置	刻画器	等高线刻画器	曲面刻画器	混料刻画器
“图形”菜单（作为一个平台）	是	是	是	是
拟合模型: 最小二乘法	是	是	是	是

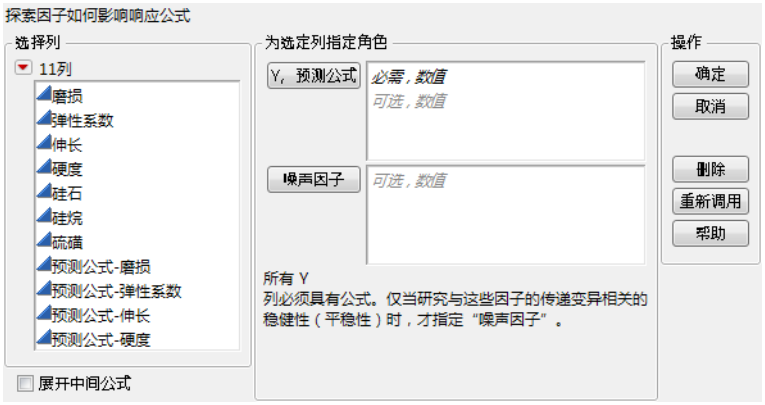
表 2.2 JMP 刻画器的提供位置 (续)

位置	刻画器	等高线刻画器	曲面刻画器	混料刻画器
拟合模型: 广义回归	是			
拟合模型: 混合模型	是	是	是	是
拟合模型: Logistic	是			
拟合模型: 对数线性方差	是	是	是	
拟合模型: 广义线性模型	是	是	是	
拟合模型: 偏最小二乘	是			
神经	是	是	是	
模型比较	是	是	是	是
非线性: 因子和响应	是	是	是	
非线性: 参数和误差平方和	是	是	是	
非线性: 拟合曲线	是			
高斯过程	是	是	是	
偏最小二乘	是			
寿命分布	是			
以 X 拟合寿命	是		是	
复发分析	是			
选择	是			
定制设计预测方差	是		是	

“刻画器”启动窗口

通过“图形”菜单将刻画器作为一个平台而不是通过拟合平台调用时，含公式的列将作为“Y，预测公式”列提供。这些公式可能已经在拟合平台中保存。

图 2.3 “刻画器”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

公式中引用的列成为 X 列 （除非该列同时也是 Y）。

Y，预测公式 包含公式的响应列。

噪声因子 仅用在对导数建模的特殊情况下。有关噪声因子的详细信息，请参见第 35 页的“噪声因子”。

展开中间公式 告知 JMP，若公式的某个成分列本身是引用其他列的公式列，则将原始列替换到内部公式中。为防止成分列展开，请添加“其他”列属性，将其命名为“展开公式”，然后指定值为 0。请参见 “展开中间公式”。

“曲面图”平台将在第 91 页的“曲面图”一章中讨论。 “曲面刻画器”与“曲面图”平台极为相似，只不过“曲面图”有多种操作模式。不论是“曲面图”平台还是“曲面刻画器”都没有与其他刻画器通用的功能。

展开中间公式

“刻画器”启动窗口包含一个“展开中间公式”复选框。选中此框时，将以不同方式处理正在刻画的公式。若刻画的公式包含另一个引用其他列的公式列，则原始列将被替换到内部公式中。因此，要刻画的公式是相对于原始列而不是引用中间列进行刻画。展开中间公式也会展开具有“向量”建模类型的公式列。

例如，当“拟合模型”拟合两个水平（A 和 B）的 Logistic 回归时，末尾公式（Prob[A] 和 Prob[B]）是 Lin[x] 列的函数，而该列本身也是另一列 x 的函数。若选定“展开中间公式”，则在刻画 Prob[A] 时，其引用的是 x 而不是 Lin[x]。

此外，使用“展开中间公式”复选框还将启用平台红色小三角菜单中的“保存展开公式”命令。这将创建包含公式的新列，该公式将刻画为末尾列（而不是中间列）的函数。

拟合组

对于“拟合模型”平台的“REML”和“逐步”特质，若多个模型拟合到多个 *Y*，则结果将合并为“拟合组”报表。这样即可在同一个刻画器中刻画不同的 *Y*。“拟合组”红色小三角菜单包含用于启动联合刻画器的选项。用于单个 *Y* 的刻画器仍可用在各自的“拟合模型”报表中。

在“逐步”分析中指定“依据”变量后，也会创建“拟合组”报表。这允许在同一个刻画器中刻画不同的模型。

“拟合组”脚本命令可用于在不同平台中拟合模型，并在“刻画器”中刻画各个模型。请参见《Scripting Guide》。

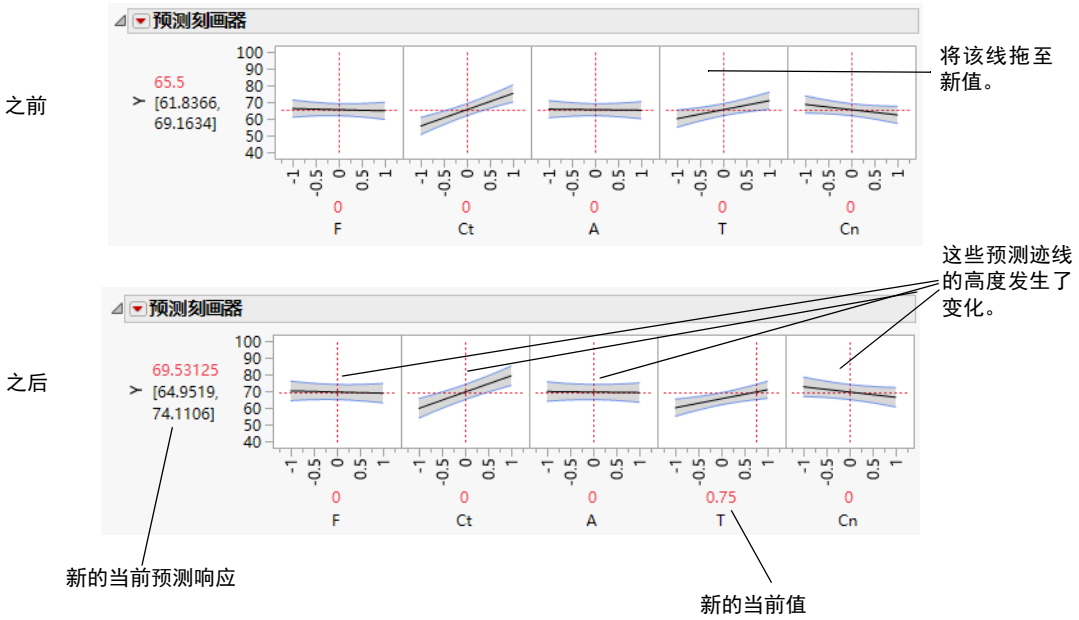
解释刻画

图 2.4 中的演示说明如何使用 “预测刻画器”组件。在解释预测刻画时需要注意几个要点：

- 可以通过预测迹线的陡峭程度在一定程度上评估因子的重要性。若模型具有曲率项（如平方项），则迹线可能是曲线。
- 若更改某个因子的值，该因子的预测迹线不受影响，但所有其他因子的预测迹线可能更改。*Y* 响应线必须穿过预测迹线与其当前值线的交点。

注意：若模型中有交互作用效应或叉积效应，预测迹线将随着您更改其他项的当前值而发生斜率和曲率偏移。这就是交互作用的含义。若没有任何交互作用效应，迹线只会发生高度变化，斜率或形状不会发生变化。

图 2.4 将一个因子的值从 0 改为 0.75



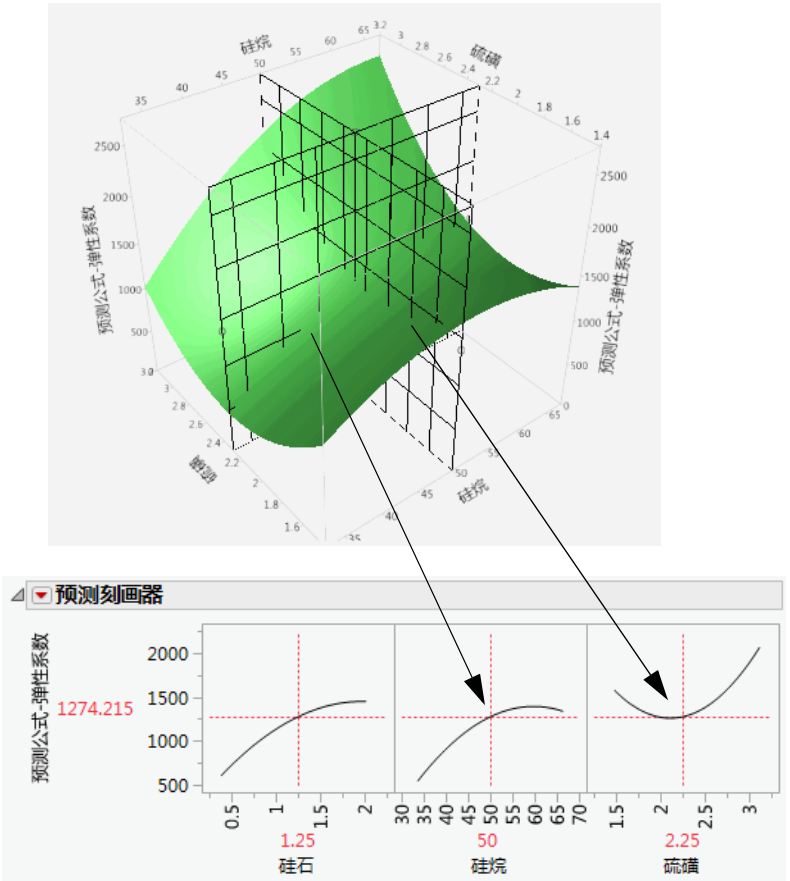
预测刻画特别适用于多重响应模型，可帮助判断哪个因子值能够优化一组复杂的条件。

点击某个图形或将当前值线向左或向右拖动，可更改因子的当前值。响应值的更改通过图形主体中的水平参考线显示出来。双击某个轴，可在调出的窗口中更改其设置。

横截面刻画器

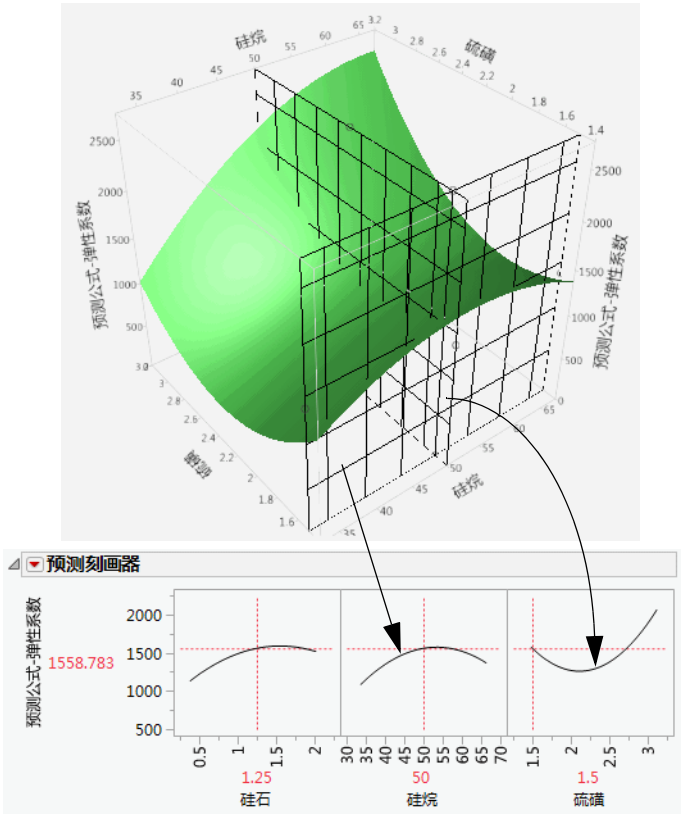
在下面使用 Tiretread.jmp 的示例中，观察**弹性系数**（作为**硫磺**和**硅烷**的函数，**硅石**保持不变）表达式的响应曲面。现在观察网格如何在**硫磺**值为 2.25 处横切**硅烷**。注意切片如何与曲面相交。若转换到下图，这将成为**硅烷**的刻画线。同样，请注意网格在**硅烷**值为 50 处横切**硫磺**。相交部分向下转换到**硫磺**图形上即为**硫磺**的刻画线。

图 2.5 横截面刻画器



现在考虑将**硫磺**的当前值从 2.25 改为 1.5。

图 2.6 横截面刻画器

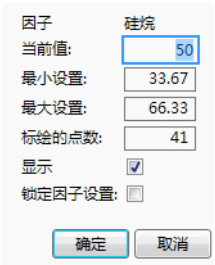


在“预测刻画器”中，请注意新值只是沿着同一条**硫磺**曲线移动，**硫磺**曲线本身并未改变。但是**硅烷**的刻画线现在是在**硫磺**的一个不同切点上取值。**硅烷**的刻画线同时也要高一些，并在更接近**硅烷**当前值 50 的另一位置达到高点。

设置或锁定因子值

在图形中，若是在按 Alt 的同时点击（在 macOS 上是在按 Option 的同时点击），则会有一个窗口提示您为因子输入特定设置。

图 2.7 连续因子设置窗口



对于连续变量，您可以指定：

当前值 用于计算刻画器中的显示值的值，相当于图形中的红色垂直线。

最小设置 因子轴的最小值。

最大值 因子轴的最大值。

标绘的点数 指定用于标绘因子的预测迹线的点数。

显示 在刻画器中显示或隐藏因子。

锁定因子设置 将因子值锁定在其当前设置。

“刻画器”平台选项

“刻画器”红色小三角菜单包含以下选项：

刻画器 显示或隐藏“预测刻画器”。

等高线刻画器 显示或隐藏“等高线刻画器”。

定制刻画器 显示或隐藏“定制刻画器”。

曲面刻画器 显示或隐藏“曲面刻画器”。

混料刻画器 显示或隐藏“混料刻画器”。

显示公式 打开 JSL 窗口，其中显示要刻画的所有公式。

OPTMODEL 的公式 创建 OPTMODEL SAS 过程的代码。按住 Ctrl+Shift，然后从红色小三角菜单中选择 OPTMODEL 的公式。

在很多平台中提供以下选项。请参见《使用 JMP》。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

常见刻画器主题

- “线性约束”
- “噪声因子”

线性约束

“预测刻画器”、“定制刻画器”和“混料刻画器”可在其操作中并入线性约束。可通过以下各节所述的两种方式输入线性约束。

红色小三角菜单项

要通过红色小三角菜单输入线性约束，请从“预测刻画器”或“定制刻画器”红色小三角菜单中选择**更改线性约束**。

从出现的窗口中选择**添加约束**，然后在相应的框中输入系数。例如，要输入约束 $p1+2*p2 \geq 0.9$，请输入如图 2.8 中所示的系数。如图所示，若刻画的是混料设计中的因子，则默认显示混料约束，且不能修改。

图 2.8 输入系数

点击**确定**后，“刻画器”将更新刻画迹线，约束将并入随后的分析和优化中。

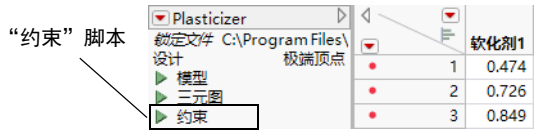
若尝试添加没有可行解的约束，则将向日志写入一条消息，且不会添加该约束。要删除约束，请为所有系数输入零。

在一个刻画器中添加的约束未经保存的情况下不能供其他刻画器访问。例如，若在“预测刻画器”下添加了约束，则这些约束不可用于“定制刻画器”。要使用约束，您可以在“定制刻画器”红色小三角菜单下添加它，或使用下一节所述的**保存线性约束**命令。

约束表属性 / 脚本

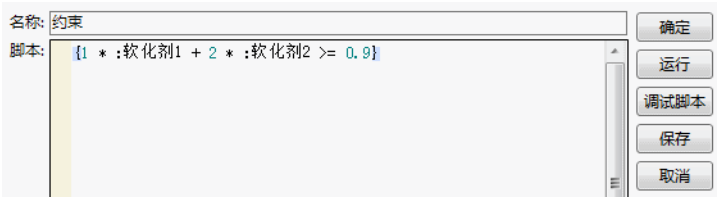
若在一个刻画器中添加了约束，并且要使其他刻画器可以访问这些约束，请使用可从平台红色小三角菜单访问的**保存线性约束**命令。例如，若在“预测刻画器”中创建了约束，请选择“预测刻画器”红色小三角菜单中的**保存线性约束**。“保存线性约束”命令可创建或更改称为“约束”的表脚本。“表属性”示例如图 2.9 所示。

图 2.9 “约束”表脚本



“约束表属性”是一个可编辑的约束列表。该列表可供其他刻画器访问，从而不必再于其他刻画器中输入这些约束。要查看或编辑“约束”，请右击红色小三角菜单并选择**编辑**。图 2.8 中的约束内容显示在下面的图 2.10 中。

图 2.10 示例 “约束”



通过从表名旁边的红色小三角菜单中选择**新建脚本**，可手动创建“约束”表脚本。

注意：手动创建“约束”表脚本时，必须严格使用“约束”两个字。此外，约束变量区分大小写且必须与列名匹配。例如，在图 2.10 中，若约束变量软化剂1 和软化剂2 用 p1 和 p2 表示，就不能写成 P1 和 P2。

在设计实验过程中指定线性约束时也会创建“约束”表脚本。

“更改线性约束”命令和“保存线性约束”命令在“混料刻画器”中不可用。要将线性约束并入“混料刻画器”的操作中，必须使用本节所述的方法之一创建“约束”表脚本。

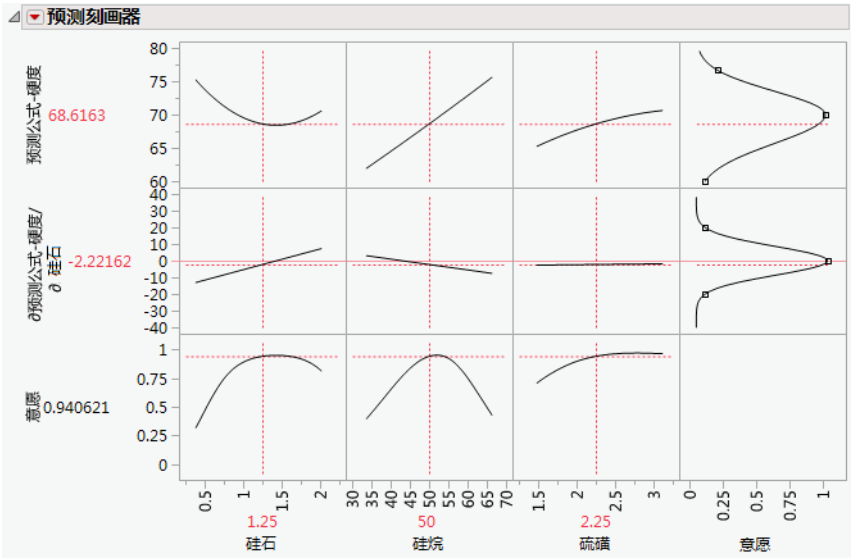
噪声因子

注意：噪声因子优化也在“预测刻画器”、“等高线刻画器”、“定制刻画器”和“混料刻画器”中提供。

通过噪声因子（稳健过程设计），您可以可靠地生产符合要求的产品，即使过程变量有变异。即便您的实验有可控因子，因子中会有一定量的不可控变异影响响应。这被称为传递变异。具有该变异的因子称为噪声因子。一些因子您根本无法控制，如环境噪声因子。一些因子的均值可以控制，但是它们的标准差无法控制。不同过程或生产步骤所生成的中间因子就经常会出现这种情况。

令过程稳健的较好方法是匹配噪声响应曲面最平稳处的目标。这样，噪声对过程就几乎没有任何影响了。从数学上讲，这个值是相对于每个噪声因子的每个响应的一阶导数为零时的值。JMP 会为您计算导数。

图 2.11 噪声因子示例



要分析具有噪声因子的模型，请执行以下步骤：

1. 拟合适当的模型（例如，使用“拟合模型”平台）。
2. 使用**保存 > 预测公式**命令将模型保存到数据表。
3. 启动**刻画器** （从**图形**菜单）。
4. 将预测公式分配给 **Y，预测公式**角色，将噪声因子分配给**噪声因子**角色。
5. 点击**确定**。

   最终的刻画器显示响应函数和它们相对于噪声因子的适当导数。导数设置为在零处具有最大意愿。
6. 从**刻画器**菜单选择**优化和意愿 > 最大化意愿**。

   在权衡使噪声因子传递的变异最小的情况下找到因子的最优设置。

第 3 章

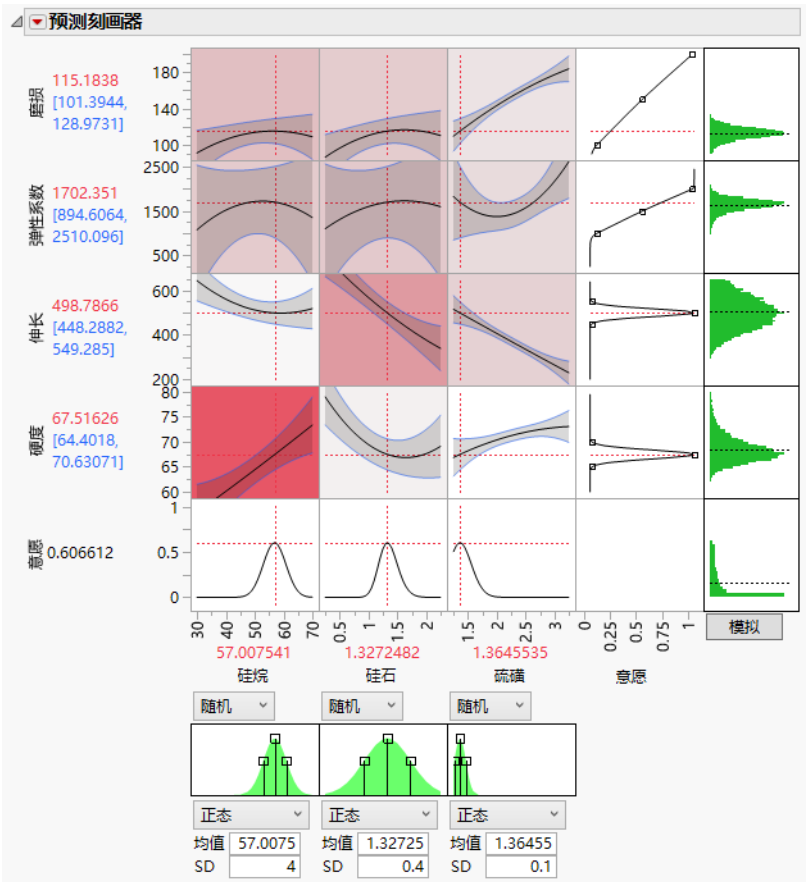
刻画器

探索每个因子的响应横截面

“预测刻画器”可为您提供关于模型的大量丰富信息。使用 “预测刻画器”可以：

- 查看在您更改各个因子的设置时，您的预测模型如何变化。
- 为一个或多个响应设置意愿目标，找出因子的最佳设置。
- 测量模型对于因子更改的灵敏度，灵敏度基于您的预测模型。
- 采用与模型无关的方式评估因子相对于模型预测的重要性。
- 基于针对因子和响应的指定分布模拟响应分布，并控制刻画器外观的各个方面。

图 3.1 具有模拟器和重要性颜色的四响应 “预测刻画器”



目录

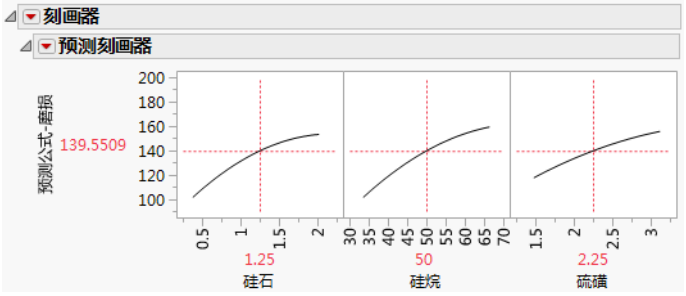
预测刻画器的示例 39
启动“预测刻画器”平台 40
“预测刻画器”选项 40
“最大化选项”窗口 47
意愿刻画和优化 47
意愿函数的构造 48
如何使用意愿函数 49
意愿刻画 50
定制意愿函数 51
评估变量重要性 52
Bagging 55
预测刻画器的其他示例 56
多重响应的意愿刻画示例 56
预测刻画器中噪声因子的示例 58
一个响应的变量重要性的示例 62
多重响应的变量重要性的示例 64
Bagging 改进预测的示例 67
指示预测准确度的 Bagging 示例 70
预测刻画器的统计详细信息 73
评估变量重要性 73
误差传播直条 74
外推控制量度 75

预测刻画器的示例

本例使用 Tiretread.jmp 样本数据表。共有三个因子（**硅石**、**硅烷**和**硫磺**）和四个响应（**磨损**、**弹性系数**、**伸长**和**硬度**）。目标是在开发轮胎胎面化合物时找到三个因子的最佳组合。请参见 Derringer and Suich (1980)。

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**图形** > **刻画器**。
3. 选择**预测公式 - 磨损**并点击 **Y，预测公式**。
4. 点击**确定**。

图 3.2 “预测刻画器”报表窗口

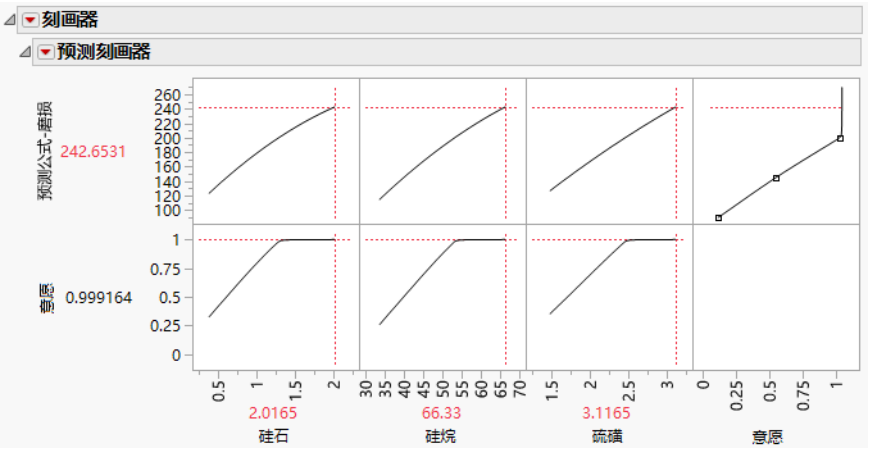


刻画器是交互式的。垂直的红线对应于因子的当前值。每个因子的当前值也用红色显示在水平轴之下。垂直轴上的红色值是基于因子当前值的预测响应。点击并拖动垂直的红线来更改因子的当前值。点击因子的红色值可以将因子设置为特定值。或者，可以播放刻画器的动画，以便循环播放多个因子值。

5. 点击“预测刻画器”红色小三角并选择**因子设置** > **动画**。
6. 点击播放按钮 以便按顺序循环播放因子值，一次一个因子。
7. 点击“预测刻画器”红色小三角，然后选择**优化和意愿** > **意愿函数**。
8. 点击“预测刻画器”红色小三角，然后选择**优化和意愿** > **最大化意愿**。

提示：您可能需要调整预测刻画器的垂直轴，以便查看图中**磨损**的最大值。

图 3.3 “最大化意愿”报表



通过最大化意愿，您可以看到要使**磨损**最大化，所有因子都设置为其最高水平。

启动“预测刻画器”平台

可以通过以下方式访问 “预测刻画器”：

- 可以直接从“图形”菜单访问“预测刻画器”。使用该方式访问“预测刻画器”时，显示“预测刻画器”启动窗口。请参见“刻画器介绍”一章中第 28 页的““刻画器”启动窗口”。
- 在很多建模平台中，“预测刻画器”可以作为红色小三角菜单选项来访问。有关在不同平台中是否提供“预测刻画器”的详细信息，请参见“刻画器介绍”一章中第 26 页的“JMP 刻画器的提供位置”。
- 可以从“模型比较”平台访问“预测刻画器”。从“模型比较”红色小三角菜单中选择“刻画器”。
- JMP PRO 可以从“公式存储库”平台访问“预测刻画器”。从“公式存储库”红色小三角菜单中选择“刻画器”。

“预测刻画器”选项

“预测刻画器”标题栏上的红色小三角菜单包含以下选项：

优化和意愿 包含以下选项的子菜单：

意愿函数 显示或隐藏意愿函数。第 47 页的“意愿刻画和优化”中对意愿进行了讨论。

最大化意愿 设置当前因子值以便将意愿函数最大化。将响应重要性权重考虑在内。

注意：在很多情况下，优化意愿函数的设置不是唯一的。“最大化意愿”选项给出一个这样的设置。等高线刻画器对于找到优化意愿的备选因子组合是很好的工具。有关示例，请参见“等高线刻画器”一章中第 86 页的 “探索最优设置”。

注意：若因子具有“离散数值”的“设计角色”列属性，在优化意愿函数时将它视为连续因子。考虑到因子只能接受离散水平的事实，在刻画器中将它作为分类项显示且选择一个允许的最佳水平。

最大化并记住 将意愿函数最大化并记住关联设置。

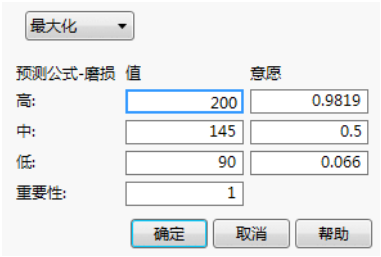
最大化选项 打开“最大化选项”窗口，您可以在其中改进优化设置。请参见第 47 页的 “‘最大化选项’窗口”。

针对每个网格点进行最大化 仅当锁定一个或多个因子时使用。锁定因子的范围划分成网格，意愿在每个网格点处最大化。若您刻画的模型具有分类因子，则该选项很有用。这样可以找到每种分类因子组合的最佳条件。

保存意愿 将每个响应的三个意愿函数设置以及关联的意愿值另存为数据表中的“响应限”列属性。这些值对应于意愿图中的控点坐标。

设置意愿 打开“响应目标”窗口，您可以在其中设置特定意愿值。

图 3.4 “响应目标”窗口



保存意愿公式 在数据表中创建包含意愿公式的列。该公式使用拟合公式（若可行），或是响应变量（若无法访问拟合公式）。

评估变量重要性 提供不同方法来计算衡量因子对模型的重要性的指标。这些指标与模型类型和拟合方法无关。请参见第 52 页的 “评估变量重要性”。

Bagging （仅当在某些建模平台中嵌入“预测刻画器”时可用。）启动 Bagging 窗口。Bootstrap 聚合 (Bagging) 支持您通过对原始数据进行放回抽样来创建多个训练数据集。对于每个训练集，使用分析平台拟合模型并生成预测。最终预测是来自所有模型的结果的组合。这通过降低方差误差提高了预测性能。请参见第 55 页的 “Bagging”。

模拟器 启动模拟器。“模拟器”支持您使用添加至因子的随机噪声和预测模型创建 Monte Carlo 模拟。典型的用法是将固定因子设置为其最佳设置，将不受控因子和模型噪声设置为随机值。随后即可计算响应超出规格限之外的比率。请参见第 137 页的 “模拟器”一章。

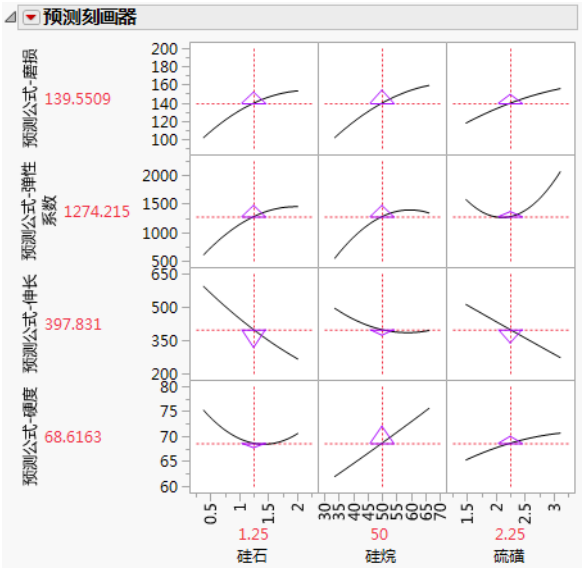
交互作用刻画器 显示或隐藏交互作用图，当您在 “预测刻画器”中更新因子值时这些图随之更新。使用该选项通过观察当因子的当前值更改时图形如何变化，来帮助可视化三次交互作用。随给定因子更改的单元格是未直接涉及该因子的单元格。

置信区间 在 “预测刻画器”图中显示或隐藏置信区间。分类因子通过直条绘制区间，连续因子通过曲线绘制区间。区间值也以蓝色显示在垂直轴上。当在某些拟合平台中使用刻画器或在 “预测刻画器”启动对话框中指定了标准误差列时，置信区间可用。

误差传播直条 （仅当在任意因子和响应变量中存在 Sigma 列属性时显示。）显示或隐藏因为因子变异而在响应上隐含的 3σ 区间。区间值也以绿色显示在垂直轴上。当因子值不太可控，而要根据因子值变异刻画响应变异的特性时，误差传播 (POE) 十分重要。请参见第 74 页的 “误差传播直条”。

灵敏度指示符 显示或隐藏一个紫色三角，其高度和方向对应于刻画函数取当前值时的偏导数值。该指示符能够快速发现敏感单元，所以在众多刻画中非常有用。

图 3.5 灵敏度指示符



边界刻画 分析混料设计时， JMP 会限制因子范围，这样混料约束范围之外的设置将被排除在外。这就是在某些混料设计中，刻画迹线的转向较为突然的原因。

若混料成分具有约束，则在通常的 0 到 1 约束之外，还有称为“边界刻画”的新子菜单显示在 “预测刻画器”红色小三角菜单上。其中包含以下两个选项：

边界处转向 让设置继续沿着约束条件的边界延伸。

边界处停止 将预测迹线截断在保持严格比例的区域中。

外推控制 显示外推控制的选项子菜单。该功能有助于确定可能的外推预测。当使用不在原始数据的因子空间内的因子点组合进行预测时，此时的预测被视为外推。在外推控制功能中，

用于确定点是否为外推的量度取决于模型拟合的类型。对于在 “拟合模型”平台的 “标准最小二乘法”特质中拟合的模型，因子设置的杠杆率用作外推量度。对于其他所有模型，正则 Hotelling T^2 值用作外推量度。请参见第 75 页的 “外推控制量度”。

“外推控制”在嵌入平台的刻画器和从 “图形”菜单启动的刻画器中提供。用于外推控制量度的数据取决于刻画器的类型。

- 构建使用验证的模型时，嵌入的刻画器和外推控制量度基于训练数据。
- 若从 “图形”菜单启动刻画器，则外推控制量度将基于所有数据，除非明确排除某些行。
- 当在模型拟合期间忽略缺失值的平台中构建模型时，这些行将从嵌入的刻画器和外推控制量度中排除。
- 若构建的模型有 “信息性缺失”，则嵌入的刻画器和外推控制量度会反映出信息性缺失。
- 要在从 “图形”菜单启动刻画器时在外推控制量度中包含信息性缺失，请使用 “信息性缺失”列属性。
- 若从 “图形”菜单启动的刻画器中调用 “外推控制”，则无论是哪种类型的模型拟合，正则 Hotelling T^2 值始终用作外推量度。因此，嵌入 “标准最小二乘法”平台的刻画器的外推控制结果将与 “图形”菜单刻画器中的结果不匹配。

“外推控制”红色小三角菜单包括警告可能的外推或限制因子设置的选项，以便不显示外推预测。

关闭 关闭所有外推控制和警告。

开 打开外推控制。若选定该选项，则会在刻画器的顶部指示该选项，并且刻画迹线仅限于不会导致外推的因子组合。

警告打开 打开外推警告。若选定该选项，则会在刻画器的顶部指示该选项。若选定生成外推的因子组合，则会出现一个警报，显示--可能的外推--。

外推详细信息 显示或隐藏预测刻画器上方的外推控制详细信息。外推控制详细信息包括当前点的外推量度值、外推阈值的值、外推量度的类型以及外推阈值的定义。

设置阈值准则 打开一个窗口，该窗口支持您调整外推阈值。当外推量度是因子设置中的杠杆率时，可以指定如何计算杠杆率以及相应乘数的值。当外推量度是正则 Hotelling T^2 值时，您可以指定乘数。请参见第 75 页的 “外推控制量度”。

重置因子网格 为每个因子显示一个窗口，允许您为因子的当前设置输入特定值，锁定该设置，以及控制网格的各方面。请参见第 32 页的 “设置或锁定因子值”一节。

图 3.6 “因子设置”窗口

因子设置 包含以下选项的子菜单：

记住设置 将一个分级显示项节点添加到报表，每当调用“记住设置”命令时，该报表累计当前设置的值。每个记住的设置前面有一个单选按钮，用于重置这些设置。“记住设置”红色小三角菜单中提供用于删除选定设置或所有设置的选项。

设置为行中的数据 将数据表行的值分配给“预测刻画器”中的 *X* 变量。

复制设置脚本 将当前“预测刻画器”的设置复制到剪贴板。

粘贴设置脚本 将剪贴板中的“预测刻画器”设置粘贴到另一报表中的“预测刻画器”。

将设置追加至表 将当前刻画器的设置追加到数据表末尾。若您想将“预测刻画器”中的设置组合添加至某一实验中以便再次运行，该选项很有用。

广播因子设置 向其他所有刻画器发送当前刻画器的因子设置，但不链接刻画器。除非再次选定“广播因子设置”，否则一个刻画器中的因子更改不会导致其他任何刻画器中的更改。

关联刻画器 将所有刻画器关联在一起。若在一个刻画器中更改了某个因子，该因子将在其他所有刻画器（包括曲面图）中都更改为该值。这是一个全局选项，针对所有刻画器设置或取消设置。

设置脚本 设置一个脚本，每当某个因子变化时调用该脚本。设置脚本接收表单的变量列表：

```
{factor1 = n1, factor2 = n2, ...}
```

例如，要将该列表写入日志，请首先定义一个函数：

```
ProfileCallbackLog = Function({arg},show(arg));
```

然后在“设置脚本”对话框中输入 `ProfileCallbackLog`。

类似函数将因子值转换为全局值：

```
ProfileCallbackAssign = Function({arg},evalList(arg));
```

或一次访问一个值：

```
ProfileCallbackAccess =
Function({arg},f1=arg["factor1"];f2=arg["factor2"]);
```

非线程 支持多线程功能无效时更改为非线程分析。

动画 显示或隐藏动画控制，这些控制支持您轻松地循环播放多个因子设置。请参见第 46 页的“动画控制”。

默认水平数 支持您设置每个连续因子的默认水平数。当“预测刻画器”特别大时，该选项很有用。首次计算迹线时，JMP 会测量其用时。若用时超过三秒，则会提醒您减少“默认水平数”会加快计算速度。

输出网格表 生成一个新数据表，其中包含含有网格值的因子所在的列、带有网格点处每个响应的计算值所对应的列，以及每个网格点处的意愿计算。

若有大量因子，使用“输出网格表”命令不太切合实际，因为这将生成一个大型表。可能会为大型网格表显示一条内存分配消息。在这些情况下，您应该锁定某些因子，使其保持锁定的值不变。要访问可指定锁定列的窗口，请在刻画器图形内按 Alt + 点击或按 Option + 点击调出包含“锁定因子设置”复选框的窗口。

输出随机表 提示指定运行试验的次数并创建一个包含该数量行数的输出表，其中包含随机因子设置以及这些设置所对应的预测值。该命令相当于打开“模拟器”，将所有因子重置为随机均匀分布，然后模拟输出的操作，但要比该操作简单得多。该命令类似于“输出网格表”，只不过其结果是一个随机表而不是一个顺序表。

生成均匀随机因子表的主要原因是：使用图形化查询以多元方式探索因子空间。该方法称为过滤的 Monte Carlo。

假定您想要查看生成所需给定响应范围的各因子所有设置的所在空间。通过选择和隐藏不合格的点（使用图形化笔刷或“数据过滤器”），您可以查看过滤后的可能性：生成您所需结果的机会空间。

某些行可能显示为已选定并用红点标记。这些表示多元意愿 Pareto 边界上的点 — 即相对于所有因子的意愿，不受其他点控制的点。

更改线性约束 支持您添加、更改或删除线性约束。这些约束将并入“预测刻画器”操作。请参见第 34 页的“线性约束”。

保存线性约束 支持您将现有的线性约束保存至称为“约束”的表脚本。请参见第 34 页的“线性约束”。

条件预测 在模型中包含随机效应时出现。随机效应预测用在使用公式表示预测值和刻画线中。

外观 包含以下选项的子菜单：

行内排列 输入在一行中显示的图的个数。该选项帮助您竖向显示各图，而不是全部显示在一个宽行中。

注意：要设置某行中显示的默认图数，请转到“文件”>“首选项”>“平台”>“刻画器”，然后编辑“行内排列”首选项。

图形间距 打开一个窗口，该窗口支持您设置图形面板之间的水平空间量。

X 变量重新排序 打开一个窗口，从中可通过按照所需顺序拖动模型主效应来对其重新排序。

Y 变量重新排序 打开一个窗口，从中可通过按照所需顺序拖动响应来对其重新排序。

适应 Y 轴 若响应超出轴范围，重新调整垂直轴尺度以便包含响应的范围。

显示创建方式 显示或隐藏创建响应列中公式的平台的名称。平台名称出现在垂直轴上。（仅当响应列的“预测”列属性中包含名为“创建平台”的参数时才可用。）

动画控制

图 3.7 动画控制

播放 / 暂停 按下播放可播放刻画器。循环播放因子设置，并在循环结束时返回到开始处。按下暂停可停止动画播放。

循环类型 列出因子设置的循环类型。

序贯 循环播放每个因子的值，一次一个因子。动画当前循环播放的因子的名称显示在速度滑块旁边。

单个因子 循环播放选定因子的值，而所有其他因子保持不变。选定因子的名称显示在速度滑块的旁边。

随机 随机循环播放因子设置/值的不同组合。

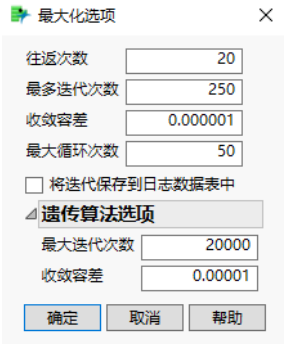
顺序数据 将因子值设置为数据表中的某行，一次一行，从第 1 行开始。当前因子设置的行号显示在速度滑块旁边。

随机数据 将因子值设置为数据表中的某个随机行，一次一行。当前因子设置的行号显示在速度滑块旁边。

速度滑块 使用滑块可调节动画播放的速度。

“最大化选项”窗口

图 3.8 “最大化选项”窗口



往返次数 优化算法中随机开始的次数。每次往返都从模型中每个因子的随机选择值开始。

最多迭代次数 每次往返中采取的最大步数。若算法在达到最大迭代次数之前没有收敛，则开始新的往返。

收敛容差 优化算法的收敛准则的上限。若收敛准则连续两次迭代都小于该值，则算法停止。

最大循环次数 （若存在线性约束，则不适用。）算法中的最大循环次数。

遗传算法选项 （仅当存在不允许的组合约束或其中一个外推控制功能已启用时才可用。）为遗传算法提供其他选项。仅当常规优化算法的解不可行时才使用遗传算法。

最大迭代次数 遗传算法的最大迭代次数。

收敛容差 遗传算法的收敛准则的上限。

意愿刻画和优化

您可以为单个响应变量或几个响应变量定义一个意愿函数。当您针对几个响应优化时，通常可以有竞争准则。例如，您可能需要将某一响应最大化，将另一响应最小化，令第三个响应接近某个目标值。

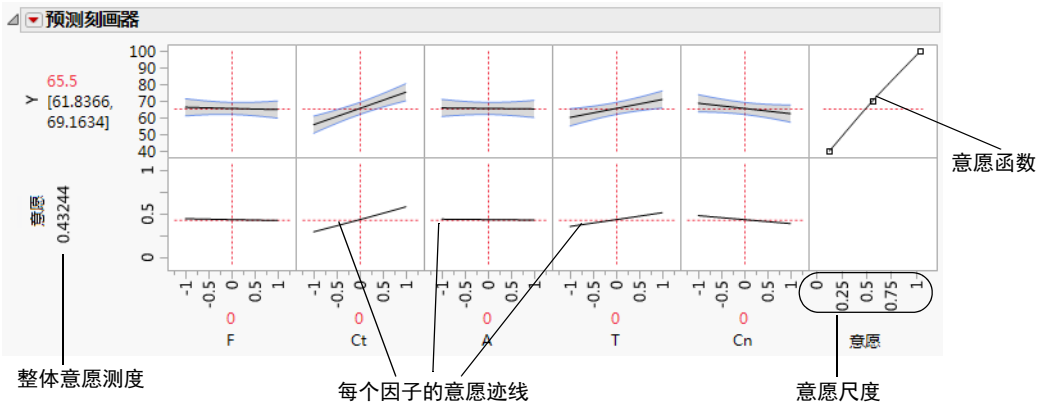
在意愿刻画中，需要为每个响应指定一个意愿函数。所有响应的整体意愿定义为各个响应的意愿函数的几何均值。有关组合响应的信息，请参见 Derringer and Suich (1980)。

要使用意愿刻画，请从“预测刻画器”红色小三角菜单选择**优化和意愿 > 意愿函数**。

注意：若响应列具有“响应限”属性，则默认启用意愿函数。

该命令向图矩阵的底部追加一个新行，专用于意愿绘图。该行为每个因子都包含一个用于显示**其意愿迹线**的图，如图 3.9 所示。它还会为每个 *Y* 变量添加一列，其中包含可调整的意愿函数。整体意愿测度显示在意愿迹线行左侧，尺度介于 0 到 1 之间。

图 3.9 意愿预测刻画器



意愿函数的构造

各个意愿函数都是经过三个定义点的平滑分段函数。这些点被称为控点（低，中间，高），可用于交互控制意愿函数的形状。

- “最小化”和“最大化”函数是三部分的分段平滑函数，它们由控点间的立方内插值和尾部的指数组成。
- “目标”函数是一个分段函数，该函数是中间值的每边上（每边具有不同的曲线）正态密度的尺度倍数，它还是分段平滑的并拟合控点。指数函数拟合尾部。
- “无”函数支持指定任意意愿函数。特别是，您可以指定“中间”值的意愿比“低”值和“高”值的意愿低。您还可以使用公式构造定制意愿函数。请参见第 51 页的“定制意愿函数”。

不允许“低”和“高”控点一直延伸到 0 或 1。当意愿值在最大化值、目标值和最小化值之间切换时，采用这种方法构造意愿函数可以得到良好的表现。

注意：JMP 不使用 Derringer and Suich (1980) 函数形式。由于它们不是平滑函数，所以不一定总能与 JMP 的优化算法配合良好。

多个优化的意愿函数

要优化多个响应时，构造并优化一个整体意愿函数。所有响应的整体意愿定义为各个响应的意愿函数的几何均值。

将 k 个响应的各个意愿函数用 $d_1, d_2,...,d_k$ 表示。整体意愿函数则是这些意愿函数的几何均值。

$$D = d_1^{1/k} d_2^{1/k} \ldots d_k^{1/k}$$

若“重要性”值已作为“响应限”列属性的一部分定义或在“响应目标”窗口中定义，则将它们集成到整体意愿函数。调整“重要性”值的尺度以使它们的和为 1。用 $w_1, w_{2,\ldots,} w_k$ 表示已统一尺度后的“重要性”值。然后将整体意愿定义为单个意愿函数的加权几何均值：

$$D = d_1^{w_1} d_2^{w_2} \ldots d_k^{w_k}$$

优化算法

用于优化整体意愿函数或单个意愿函数（若只有一个响应）的方法取决于因子类型。

- 对于分类因子，使用坐标交换算法。
- 对于连续因子，使用梯度下降算法。
- 存在约束或混料因子时，使用 Wolfe 简化梯度方法。
- 为了降低找到局部最优值的风险，JMP 使用多次随机开始。

如何使用意愿函数

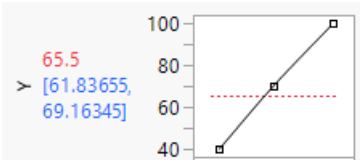
要使用变量的意愿函数，请拖动函数控点以表示响应值。

拖动这些控点时，更改的响应值显示在图左侧标为**意愿**的区域中。虚线是当前因子设置的响应。整体意愿显示在意愿迹线行左侧。或者，您可以选择**优化和意愿** > **设置意愿**来输入点的特定值。

图 3.10 显示了创建意愿设置的步骤。

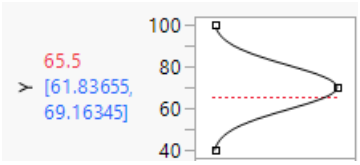
最大化 默认意愿函数设置为“最大化”（“越高越好”）。“高”控点位于最大 Y 值处并在最高意愿（接近 1）处对齐。“低”控点位于最小 Y 值处并在最低意愿（接近 0）处对齐。

图 3.10 最大化意愿



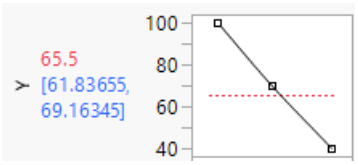
目标 您可以将目标值指定为“最佳”。在该示例中，中间控点位于 Y = 70 处并与最大意愿 1 对齐。Y 值接近 40 或 100 时就变得不可取。Y = 40 和 Y = 100 处的“高”和“低”控点位于接近 0 的最小意愿处。

图 3.11 定义目标意愿



最小化 最小化（“越小越好”）意愿函数将高响应值与低意愿关联，将低响应值与高意愿关联。该曲线是一条沿图中央水平线倒过来的最大化曲线。

图 3.12 最小化意愿



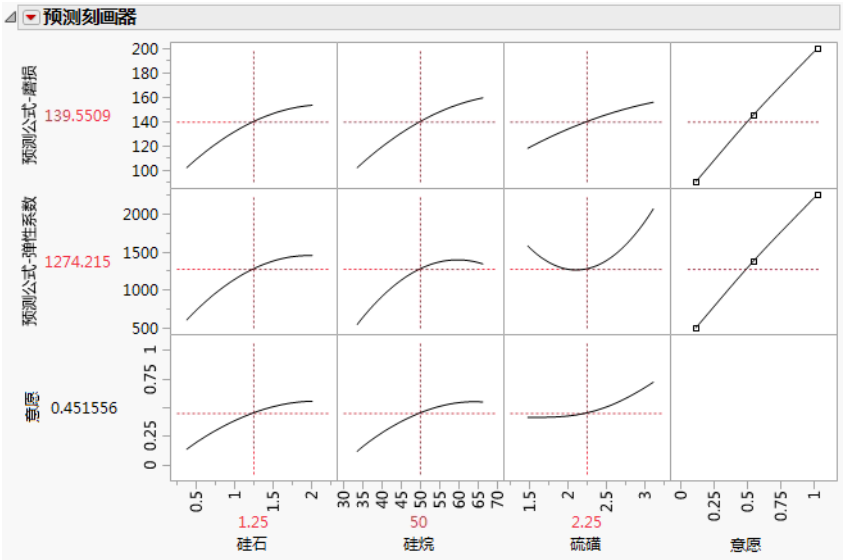
注意：将最大化或最小化意愿函数的“高”或“低”控点拖过中间点的 y 值时会出现相反的点。最小化变为最大化，而最大化变为最小化。

意愿刻画

图中最后一行显示每个因子的意愿迹线。垂直轴上的**意愿**一词旁边的数值是意愿测度的几何均值。图中的这一行显示由于一次更改一个因子而生成的当前意愿和意愿迹线。

例如，图 3.13 显示两个响应的意愿函数。您想要最大化**磨损**和**弹性系数**。意愿图指示您可以通过增大任意因子的值来增大意愿。

图 3.13 具有调整的意愿和因子值的预测刻画图



定制意愿函数

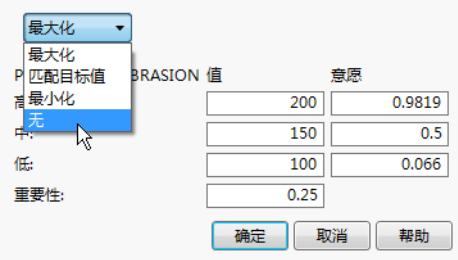
可以使用定制意愿函数。例如，假定您想要使用以下函数进行最大化。

图 3.14 最大化基于函数的意愿

$$\frac{Pred\ Formula\ ABRASION}{96} + \frac{Pred\ Formula\ MODULUS}{700} + \frac{33}{\left(\left| Pred\ Formula\ ELONG - 450 \right| + 1\right)} + \frac{2}{\left(\left| Pred\ Formula\ HARDNESS - 67 \right| + 1\right)}$$

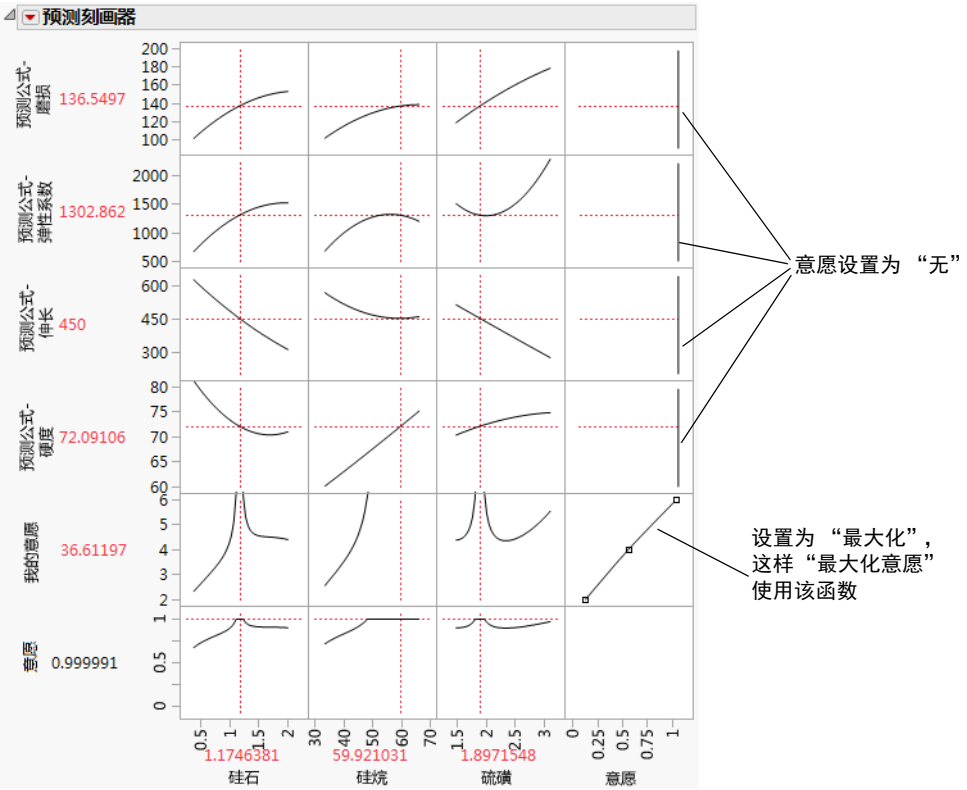
首先，创建包含以上公式的名为**定制意愿**的列。然后，选择**图形 > 刻画器**以启动平台。选择所有**预测公式**列和**定制意愿**列，然后选择 **Y，预测公式**。选择**展开中间公式**选项。通过从红色小三角菜单选择**优化和意愿 > 意愿函数**来启用意愿函数。各个效应的意愿函数必须禁用。为此，首先要在意愿图窗口中双击，然后在出现的窗口中选择**无**。将**定制意愿**的意愿设置为最大化。

图 3.15 选择无意愿目标



此时，选择**优化和意愿** > **最大化意愿**只使用**定制意愿**函数。

图 3.16 最大化定制意愿



评估变量重要性

“变量重要性”报表计算用于测量模型因子重要性的指标，这些指标与模型类型和拟合方法无关。拟合模型仅用于计算预测值。该方法根据每个因子的变异范围估计预测响应的变异性。若因子中的变异引起响应变异较大，则该效应对模型十分重要。

注意：在某些平台中，对于包含两个以上水平的分类响应，“评估变量重要性”不可用。

还可以通过“图形”菜单中包含的“预测刻画器”来访问“评估变量重要性”。

有关统计详细信息，请参见第 73 页的“评估变量重要性”。另见 Saltelli (2002)。

注意：“评估变量重要性”要求所有列位于同一数据表。

“评估变量重要性”报表

“评估变量重要性”菜单包含以下选项，说明构造重要性指标时所用的方法：

独立均匀输入 对于每个因子，从最小和最大观测值所定义的均匀分布中抽取 Monte Carlo 样本。若您确信因子之间不相关且因子的可能值均匀分布在研究中所表示的范围内，则使用该选项。这是不涉及约束或混料因子的设计实验的合适选项。

独立再抽样输入 对于每个因子，通过对因子的一组观测值再抽样来获取 Monte Carlo 样本。若您确信因子之间不相关且因子的可能值不能通过均匀分布表示，则使用该选项。

非独立再抽样输入 使用 *k* 最近邻法从观测到的组合构造因子值，以便将相关性考虑在内。该选项将观测到的方差和协方差视为因子协方差结构的表示。若您确信因子之间相关，则使用该选项。请注意，该选项对于数据表中的行数很敏感。若用于少数行，结果可能不可靠。

注意：“独立再抽样输入”和“非独立再抽样输入”选项适用于观测研究。“独立”选项快于“非独立”选项，但是“非独立”选项可以更好地处理多重共线性，且不外推到远离数据的区域。

线性约束输入 对于每个因子，从线性约束所定义的区域上的均匀分布中抽取 Monte Carlo 样本。可以在“预测刻画器”中定义线性约束，或在设计实验中构造它们。此外，样本被限制为介于最小和最大观测值之间。当您认为这些约束影响输入的分布时，在存在线性约束的地方使用该选项。

这些算法的速度取决于模型求值速度。通常，速度最快的选项是“独立均匀输入”，速度最慢的选项是“非独立再抽样输入”。若估计过程无法即刻完成，可选择“接受当前指标”。

注意：使用 Monte Carlo 抽样构造变量重要性指标。因此，对于不同的试验运行过程，重要性指标值会有些许不同。

“变量重要性”报表

每个“评估变量重要性”选项都会提供汇总报表和边缘模型图。当“评估变量重要性”报表打开时，“预测刻画器”中的因子将根据其“总效应”重要性指标重新排序。若有多重响应，因子将根据“总体”报表中的“总效应”重要性指标重新排序。当您运行若干“变量重要性”报表时，“预测刻画器”中的因子将根据其在最近使用报表中的“总效应”指标排序。

汇总报表

对于每个响应，会有一个表显示以下元素：

列 关注的因子。

主效应 重要性指标，反映该因子本身（而不涉及与其他因子的组合）的相对贡献。

总效应 重要性指标，反映该因子本身以及与其他因子的组合的相对贡献。“总效应”列显示为一个条形图。请参见第 54 页的“权重”。

主效应标准误差 “主效应”的重要性指标的 Monte Carlo 标准误差。这是一个隐藏列，您可以通过在报表中右击并选择**列 > 主效应标准误差**来访问该列。默认情况下，抽样会继续，直到该误差小于 0.01。计算详细信息在第 74 页的“变量重要性标准误差”中提供。（不适用于“非独立再抽样输入”选项。）

总效应标准误差 “总效应”的重要性指标的 Monte Carlo 标准误差。这是一个隐藏列，您可以通过在报表中右击并选择**列 > 总效应标准误差**来访问该列。默认情况下，抽样会继续，直到该误差小于 0.01。计算详细信息在第 74 页的“变量重要性标准误差”中提供。（不适用于“非独立再抽样输入”选项。）

权重 一个显示“总效应”指标的图，位于最后一列的右侧。您可以取消选择或重新选择该图，只需在报表中右击并选择**列 > 权重**即可。

带有缺失值的函数求值的比例 某些输入组合生成无法估计的预测时的 Monte Carlo 样本的比例。若该比例不为零，该消息会作为注释显示在表的底部。

注意：若有多重响应，“汇总报表”会提供一个“总体”表，后跟每个响应的表。“总体”报表中的重要性指标是所有响应的重要性指标的平均值。

边缘模型图

“边缘模型图”报表（请参见图 3.37）显示了图矩阵，每个响应都有一行，每个因子都有一列。因子根据其总体“总效应”重要性指标的大小来排序。

对于给定的响应和因子，图中显示每个因子值的响应均值，该均值是对用于计算重要性指标的所有输入所取的平均值。这些图不同于刻画器图，刻画器图显示的是响应的横截面。“边缘模型图”对于评估因子的主效应很有用。

请注意，所选的输入方法将影响在边缘模型图上标绘的值。此外，由于这些图基于生成的输入设置，标绘的响应均值可能不显示为平滑曲线。

红色小三角菜单选项支持您显示或隐藏图的以下方面：

估计值 作为因子值函数计算的模拟值均值的平滑估计。

注意：因为是模拟均值的估计值，因此当重新运行分析时这些值可能更改。

置信区间 模拟均值的 95% 置信带。这个带通常很窄，若不扩展尺度，可能无法看到。不适用于“非独立再抽样输入”。

注意：因为置信界限是模拟的，因此当重新运行分析时这些带可能更改。

数据 针对因子值标绘的响应的实际 （非模拟）值。

“变量重要性”选项

“变量重要性”红色小三角菜单包含以下选项：

按主效应重要性对因子重新排序 按照主效应的重要性指标 （主效应）对 “预测刻画器”中的单元格重新排序。

按总重要性对因子重新排序 按照因子的总重要性指标 （总效应）对 “预测刻画器”中的单元格重新排序。

着色刻画器 使用由红到白的强度尺度按照 “总效应”重要性指标对刻画器中的单元格着色。

注意：您可以点击“汇总报表”中的行以选择数据表中的列。这有利于进一步分析。

Bagging

Bootstrap 聚合 (Bagging) 是一种提高预测性能、同时也能很好揭示预测可靠性的技术。Bagging 对于不稳定的方法（包括神经网络、分类树和回归树）特别有用。

Bagging 通过从原始数据放回抽样来创建 *M* 个训练数据集。所有训练数据集的大小与原始数据集的大小相同。对于每个训练数据集，使用分析平台拟合模型并生成预测。因此，对于原始数据集中的每个观测总共有 *M* 个预测。最终预测是 *M* 个预测的平均值。

在很多分析平台中提供 Bagging 功能。要使用 Bagging，请从 “预测刻画器”红色小三角菜单中选择**保存 Bagged 预测**。将显示一个窗口，其中包含 Bagging 的以下选项：

Bootstrap 样本数 设置要再抽样数据并构建模型的次数。次数较大的能得到更精确的预测结果。默认情况下， Bootstrap 样本数为 100。

随机种子 设置一个随机种子，您可以在以后运行 Bagging 分析时重新输入它以复制当前结果。默认情况下，不设置种子。

非整数权重 执行 Bayesian Bagging 分析。在每次 Bootstrap 迭代中，给每个观测分配一个非零权重。执行预测的模型使用这个加权的观测。默认情况下，不选择 “非整数权重”选项，只执行简单 Bagging 分析。

提示：若分析中使用的观测数很小或您关注 Logistic 回归设置中的分离问题，请使用“非整数权重”选项。

假定选择 “非整数权重”。对于每次 Bootstrap 迭代，为报表中使用的每个观测分配一个非零权重。这些权重之和为模型中使用的观测数 *n*。有关如何计算并使用权重的详细信息，请参见 《基本分析》。

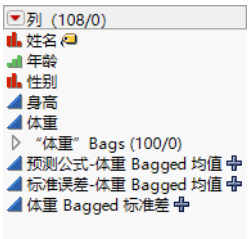
保存预测公式 对于每个 Bagged 预测，该选项保存用于在列属性中生成该预测的公式。仅在提供 Bagging 的一部分分析平台中可以使用该选项。

注意：若 “保存预测公式”不可用，将显示一个注释，说明将仅保存预测值。

Bagging 自动在原始数据表中创建新列。将所有 M 组 Bagged 预测保存为隐藏的列。最终预测保存在名为“预测公式-<列名> Bagged 均值”的列中。最终预测的标准差保存在名为“<列名> Bagged 标准差”的列中。Bagged 均值的标准误差保存在名为“标准误差-<列名> Bagged 均值”的列中。该标准误差由标准差除以 $\sqrt{M-1}$ 得到。在此处，<列名> 标识报表中的 Bagged 列。

从标准误差可以判断预测的精度。标准误差很小指示该观测的预测精度高。有关 Bagging 的详细信息，请参见 Hastie 等人 (2009)。

图 3.17 Bagging 列



预测刻画器的其他示例

本节包含说明 “预测刻画器”平台和嵌入到很多分析平台中的 “预测刻画器”的各个方面的其他示例。

- “多重响应的意愿刻画示例”
- “预测刻画器中噪声因子的示例”
- “一个响应的变量重要性的示例”
- “多重响应的变量重要性的示例”
- “Bagging 改进预测的示例”
- “指示预测准确度的 Bagging 示例”

多重响应的意愿刻画示例

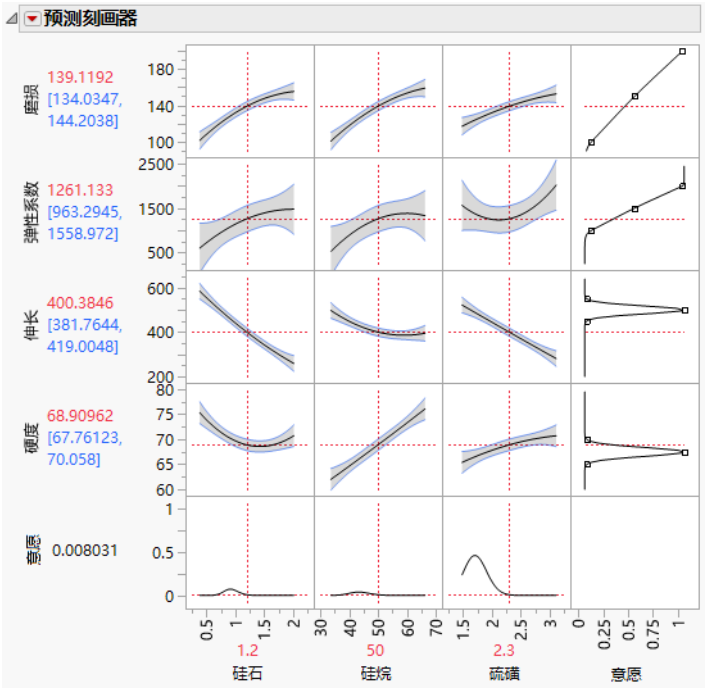
存在多重响应时，意愿指标变得尤为有用。该想法最初是由 Derringer 和 Suich (1980) 提出的，他们给出了以下示例。假定有四个响应：**磨损**、**弹性系数**、**伸长**和**硬度**。中心复合设计中使用了三个因子：**硅石**、**硅烷**和**硫磺**。

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。

2. 点击“4 种响应的响应曲面设计”旁边的绿色小三角以运行该脚本。

   该脚本为这 4 个响应定义具有完全平方响应曲面的模型。将为所有响应都显示汇总表和效应信息，后面紧随图 3.18 中所示的预测刻画器。

图 3.18 优化之前的多重响应预测刻画器

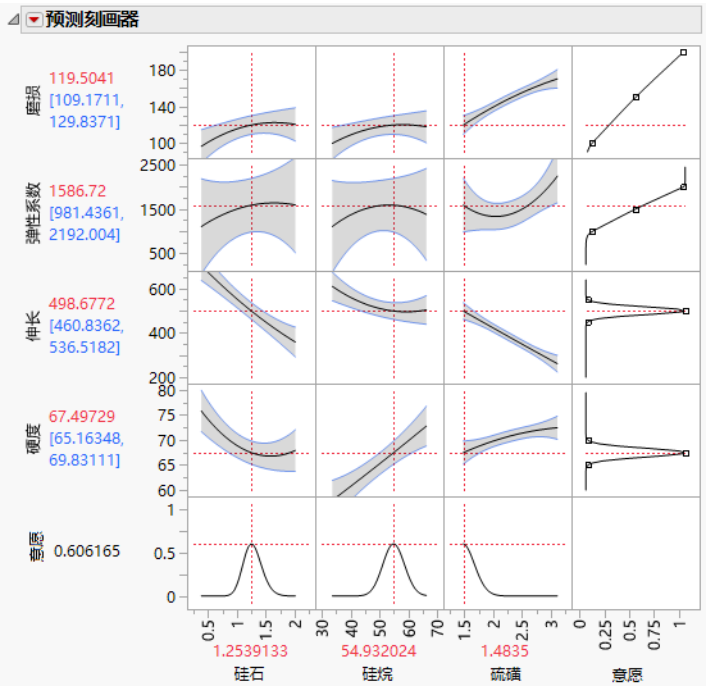


请注意以下有关意愿函数的事项：

- 最大值**磨损**和最大值**弹性系数**是最大的意愿。
- **伸长**目标值为 500 是最大的意愿。
- **硬度**目标值为 67.5 是最大的意愿。

3. 从“预测刻画器”红色小三角菜单选择**优化和意愿** > **最大化意愿**以最大化意愿。

图 3.19 优化之后的预测刻画器



底部的意愿迹线在效应当前值之外的所有位置都降低，这指出任何进一步的调整都会降低整体意愿。

预测刻画器中噪声因子的示例

本例使用 Tiretread.jmp 样本数据集。该数据集显示轮胎制造商的实验结果。实验目标是基于**硅石**、**硅烷**和**硫磺**三个因子的含量来匹配**硬度**目标值 = 70。假定**硅烷**和**硫磺**含量可以方便（精确）地进行控制，但是**硅石**不容易控制。

为了进行比较，首先我们优化硬度的因子而不考虑噪声因子**硅石**导致的变异。

1. 选择**图形 > 刻画器**以启动“预测刻画器”。
2. 将**预测公式 - 硬度**分配给 **Y，预测公式**角色。
3. 点击**确定**。
4. 点击“预测刻画器”红色小三角，然后选择**优化和意愿 > 意愿函数**。
5. 点击“预测刻画器”红色小三角并选择**最优化和意愿 > 设置意愿**以打开“响应目标”窗口。

   提示：或者，您可以在“意愿”图中双击以打开“响应目标”窗口。

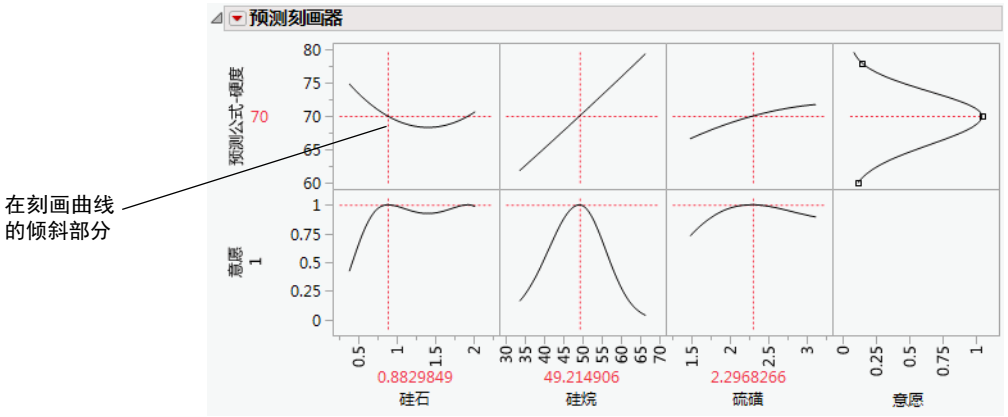
6. 从列表中选择**匹配目标值**并点击**确定**。

7. 点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**以找到**硬度**目标值的最佳因子设置。

   我们得到以下“预测刻画器”显示。请注意**硅石**因子的最佳值在刻画曲线的倾斜部分。这意味着**硅石**设置中的变异影响响应**硬度**。

   注意：您可能从中得到不同的结果，因为不同的因子值组合均可以匹配目标。

图 3.20 硬度的最大化意愿

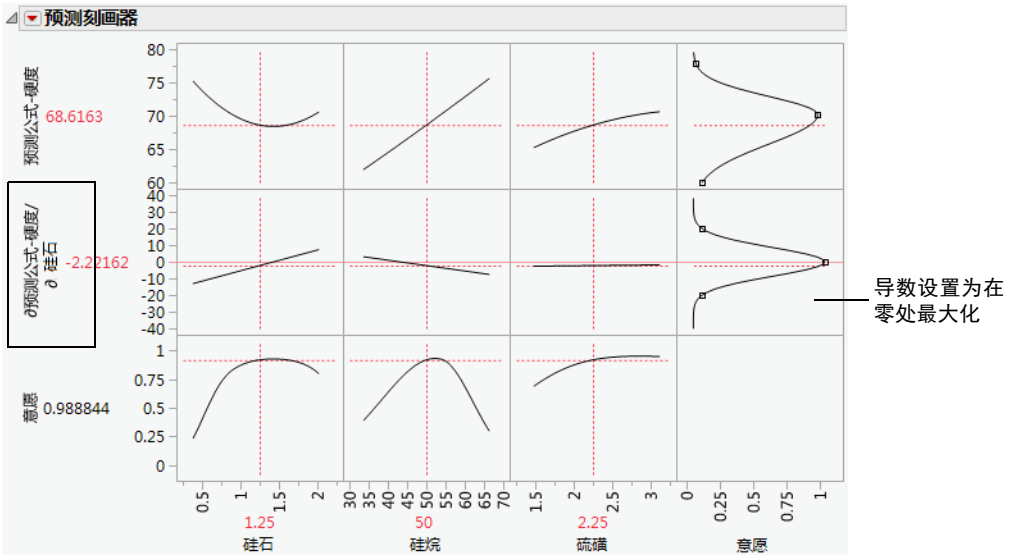


现在，我们不仅要优化**硬度**的特定目标值，还要找到一个对 “硅石”设置中的变异性不敏感的设置。以**硅石**作为噪声因子，重复优化过程。

1. 选择**图形 > 刻画器**。
2. 选择**预测公式 - 硬度**，然后点击 **Y，预测公式**。
3. 选择**硅石**并点击**噪声因子**。
4. 点击**确定**。
5. 在 “预测刻画器”菜单中选择**优化和意愿 > 意愿函数**。
6. 双击响应 “预测公式 - 硬度”的 “意愿”图，以打开 “响应目标”窗口。
7. 从列表中选择**匹配目标值**并点击**确定**。

   最终的刻画器具有相对于噪声因子的拟合模型的适当导数，该噪声因子设置在导数为零处（它最平坦的点）为最大意愿。

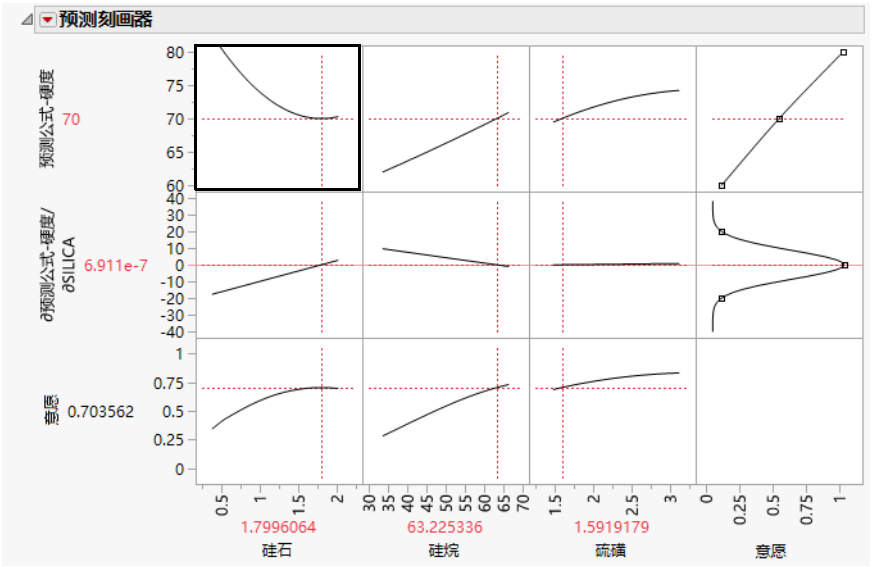
图 3.21 对硅石的预测公式的导数



8. 点击 “预测刻画器”红色小三角并选择**优化和意愿** > **最大化意愿**，以在权衡使噪声因子传递的变异最小的情况下找到过程因子的最佳值。

   这次我们也匹配了**硬度**的目标值，但是**硅石**的值位于更平坦的区域。这意味着**硅石**中的变异传递给**硬度**的变异并不像设置在 “硅石”曲线陡峭部分的值时的变异那样大。

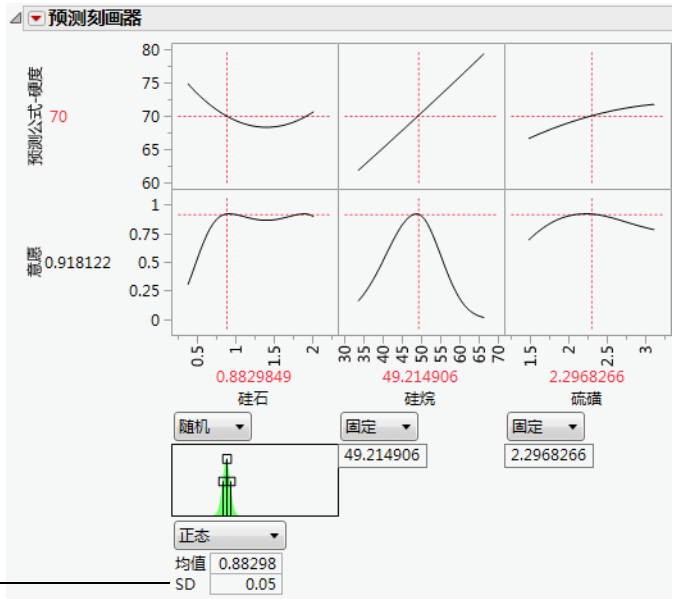
图 3.22 最大化意愿



通过对每个刻画器（一个没有噪声因子，一个有噪声因子）执行这些步骤，您可以看到考虑“硅石”变异对于预测方差的影响：

1. 在没有噪声因子的“刻画器”中，点击“预测刻画器”红色小三角并选择**模拟器**。
2. 将**硅石**分配给具有标准差 0.05 的随机正态分布。使用均值的默认设置。

图 3.23 设置随机正态分布



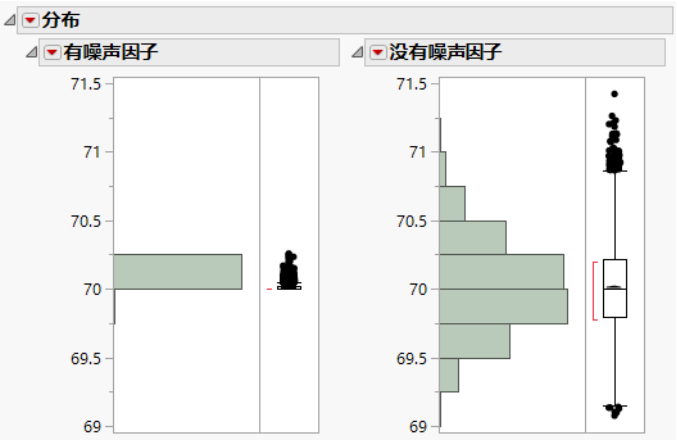
3. 点击**模拟**。
4. 点击 “模拟结果输出表”灰色小三角，然后点击**制表**按钮。
5. 将**预测公式 - 硬度**列重命名为**没有噪声因子**。
6. 在有噪声因子的 “刻画器”中，点击 “预测刻画器”红色小三角并选择**模拟器**。
7. 将**硅石**分配给具有标准差 0.05 的随机正态分布。使用均值的默认设置。
8. 点击**模拟**。
9. 点击 “模拟结果输出表”灰色小三角，然后点击**制表**按钮。

完成这些步骤对原始变量最佳模拟和对噪声因子最佳模拟将生成两个相似的数据表，每个表对应一个模拟。我们想要比较两个模拟中预测的硬度分布。为了便于比较，我们将两个预测列放在一个数据表中。

1. 选择当前模拟表中的**预测公式 - 硬度**列，然后选择**编辑 > 以全精度复制**。
2. 在另一个当前模拟表中，选择**列 > 新建列**。
3. 键入列名旁边的**有噪声因子**，然后点击**确定**。
4. 选择**有噪声因子**列，然后选择**编辑 > 粘贴**。

5. 选择**分析 > 分布**。
6. 选择**有噪声因子**和**没有噪声因子**，然后点击 **Y，列**。
7. 点击**确定**。
8. 点击 “分布”红色小三角并选择**统一尺度**。

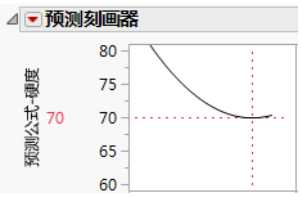
图 3.24 有噪声因子和没有噪声因子的分布比较



这些直方图显示：当噪声因子未包含在分析中时，预测的**硬度**的变异更大。

还有一件有趣的事情要注意：在包含噪声因子时直方图的形状。在比较上述直方图时，需注意**有噪声因子**分布的数据仅在一个方向上逐渐消失。若您从“分布”红色小三角菜单中取消选择“统一尺度”，则**有噪声因子**中的“偏度”就很明显。预测有偏斜，因为**硬度**相对于**硅石**为最小值，如图 3.25 中所示。因此**硅石**的变异只能使**硬度**增加。使用非稳健方案时，该变异可能向任何一个方向传递。

图 3.25 显示 “硬度”对 “硅石”的最小值的预测刻画器



一个响应的变量重要性的示例

Boston Housing.jmp 样本数据表包含可能与住房价格中位数相关的 13 个因子的数据。您使用神经网络拟合模型。由于神经网络不适用于正式的假设检验，所以不能借助这些检验来评估哪些变量在预测响应时很重要。不过，您可以使用“评估变量重要性”刻画器选项来评估变量重要性。

请注意，您的结果虽然可能不同于下面所示的结果，但应该与其相似。本例中有两个随机变异来源。拟合神经网络时，使用了 *k* 重交叉验证。这会将数据随机划分为训练集和验证集。此外，还使用 Monte Carlo 抽样来计算因子重要性指标。

1. 选择**帮助 > 样本数据库**，然后打开 Boston Housing.jmp。
2. 选择**分析 > 预测建模 > 神经**。
3. 从 “选择列”列表中选择**中位数**，然后点击 **Y，响应**。
4. 从 “选择列”列表中选择其他所有列，然后点击 **X，因子**。
5. 点击**确定**。
6. 在 “神经模型启动”面板中，从 “验证方法”下的列表中选择 **K 重**。

   选择 “K 重”后，“折数”默认为 5。
7. （可选）在 “随机种子”旁边输入 123。

   注意：由于在神经网络模型中是随机选择验证集，结果可能有所不同。输入上面的种子，这样您可以重现本例中所示的结果。
8. 点击**执行**。
9. 点击 “模型 NTanH(3)”旁边的红色小三角并选择**刻画器**。

   “预测刻画器”显示在报表最底部。请留意因子顺序以便将来比较。

   由于因子之间相关，您可以通过将 “非独立再抽样输入”选作用于评估变量重要性的抽样方法将这种相关性考虑在内。
10. 点击 “预测刻画器”红色小三角并选择**评估变量重要性 > 非独立再抽样输入**。

    随即显示 “变量重要性 : 非独立再抽样输入”报表。检查 “预测刻画器”单元格是否已经按照报表中的 “总效应”指标的量值重新排序。在图 3.26 中，检查 “总效应”重要性指标是否将**房间数**和**低收入者**标识为对预测响应影响最大的因子。

图 3.26 “非独立再抽样输入”报表

您可能会有兴趣将假定因子相关时获取的重要性指标与假定因子独立时获取的重要性指标进行比较。

11. 点击“预测刻画器”红色小三角并选择**评估变量重要性 > 独立再抽样输入**。

    再抽样输入选项在本例中有意义，因为涉及的分布不是均匀分布。“变量重要性：独立再抽样输入”报表显示在图 3.27 中。检查标识为对预测值影响最大的两个因子是否为**低收入者**和**高速公路**。

图 3.27 “独立再抽样输入”报表

多重响应的变量重要性的示例

Tiretread.jmp 样本数据表中的数据是正交因子设计实验的结果。出于以上原因，您可以使用基于独立输入的重要性估计值。假定您确信因子值实际上在整个设计空间中都有变化，而不仅仅是认定实验中定义的设置，那么您应将“独立均匀输入”选作重要性指标的抽样方案。

1. 选择**帮助 > 样本数据库**，然后打开 Tiretread.jmp。
2. 运行脚本 **4 种响应的响应曲面设计**。

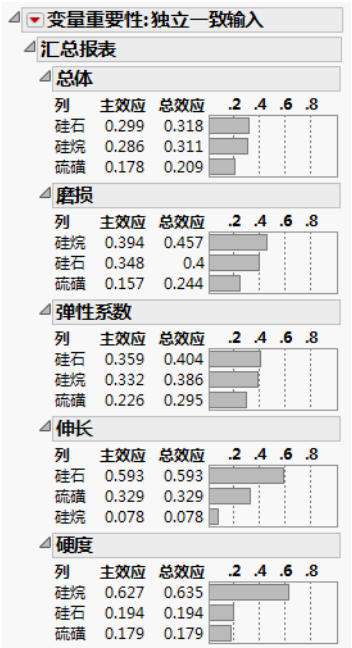
   “预测刻画器”显示在报表最底部。

3. 点击“预测刻画器”红色小三角并选择**评估变量重要性 > 独立均匀输入**。

   “汇总报表”显示在图 3.28 中。由于重要性指标基于随机抽样，您的估计值可能与图中显示的值稍有出入。

   报表中为这四个响应均显示了对应的表。“总体”表显示了不同响应的因子重要性指标的平均值。“预测刻画器”中的因子（图 3.29）已经过重新排序，符合其在“总体”表的“总效应”重要性中的顺序。

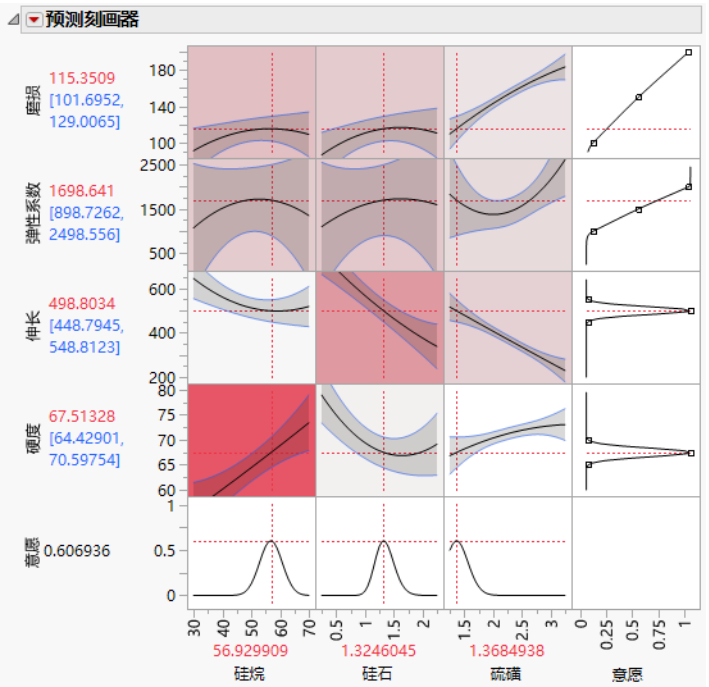
图 3.28 四个响应的汇总报表



4. 点击 “变量重要性 : 独立均匀输入”旁边的红色小三角并选择**着色刻画器**。

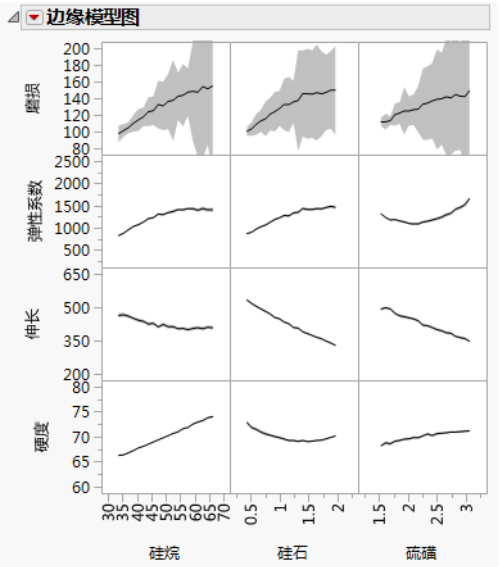
   采用由红到白强度尺度的颜色叠放在刻画器面板上，以反映 “总效应”重要性。例如，您很容易看到最为重要的效应是**硅烷**对**硬度**的效应。

图 3.29 四个响应的预测刻画器



“边缘模型图”报表显示每个因子在另外两个因子设置的均匀分布中的响应均值。

图 3.30 四个响应的边缘模型图



Bagging 改进预测的示例

在很多情况下使用 Bagging。其中之一是为了提高预测功效。Bagging 对不稳定的模型特别有用。本例使用 Tiretread.jmp 样本数据表。共有三个因子（**硅石**、**硅烷**和**硫磺**）和四个响应（**磨损**、**弹性系数**、**伸长**和**硬度**）。首先，您拟合一个神经网络模型来同时预测作为三个因子的函数的四个响应变量。然后，您对神经网络模型执行 Bagging。最后，您比较预测结果，查看通过 Bagging 获得的改进。

拟合神经网络模型

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**分析** > **预测建模** > **神经**。
3. 选择**磨损**、**弹性系数**、**伸长**和**硬度**，然后点击 **Y，响应**。
4. 选择**硅石**、**硅烷**和**硫磺**，然后点击 **X，因子**。
5. 点击**确定**。
6. （可选）在 “随机种子”旁边输入 2121。

   注意：由于在神经网络模型中是随机选择验证集，结果可能有所不同。输入上面的种子，这样您可以重现本例中所示的结果。

7. 点击**执行**。
8. 点击 “模型 NTanH(3)”旁边的红色小三角并选择**保存公式**。

   注意：该选项将神经网络模型的所有响应变量的预测值保存到数据表。稍后将这些值与从 Bagging 获得的预测进行比较。

执行 Bagging

构造了初始模型后，您可以使用该模型执行 Bagging。通过“预测刻画器”访问 Bagging 功能。

1. 点击“模型 NTanH(3)”旁边的红色小三角并选择**刻画器**。

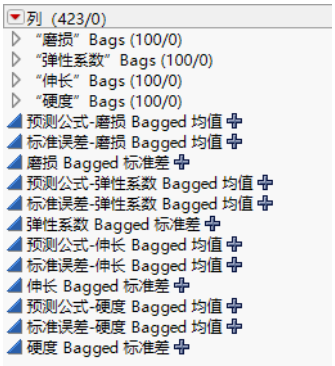
   “预测刻画器”显示在报表底部。
2. 点击 “预测刻画器”红色小三角并选择**保存 Bagged 预测**。
3. 在 “Bootstrap 样本数”旁边输入 100。
4. （可选）在 “随机种子”旁边输入 2121。

   注意：由于放回抽样是随机的，所以结果会有所不同。要重现本例的精确结果，请设置“随机种子”。

5. 点击**确定**。

返回数据表。对于每个响应变量，有三个分别表示为**预测公式 -< 列名 > Bagged 均值**、**标准误差 -< 列名 > Bagged 均值**、**< 列名 > Bagged 标准差**的新列。**预测公式 -< 列名 > Bagged 均值**列是最终预测。

图 3.31 Bagging 后添加到数据表的列



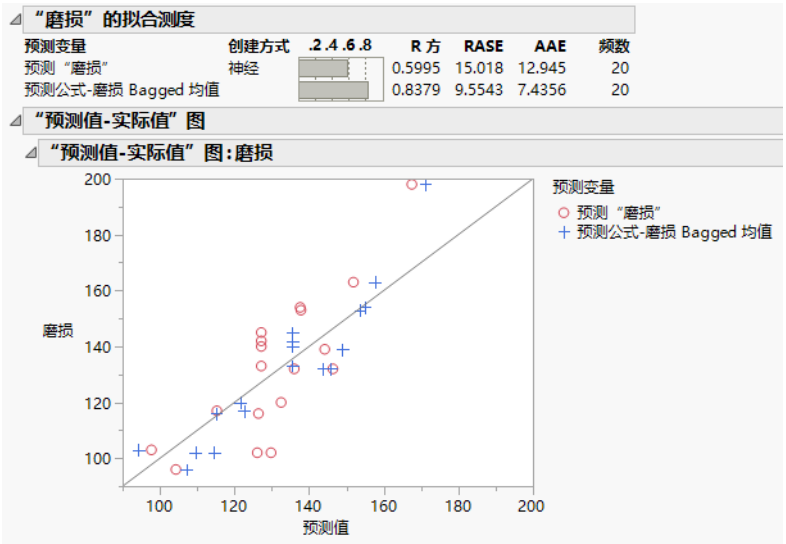
比较预测

为了查看 Bagging 如何提高预测功效，将 Bagged 模型的预测与原始模型预测进行比较。使用“模型比较”平台一次查看一个响应变量。

1. 选择**分析 > 预测建模 > 模型比较**。
2. 选择**预测 “磨损”**，然后点击 **Y，预测变量**。
3. 选择**预测公式 - 磨损 Bagged 均值**，然后点击 **Y，预测变量**。
4. 点击**确定**。

   将显示包含列列表的窗口。
5. 选择**磨损**，然后点击**确定**。
6. 点击 “模型比较”红色小三角并选择**标绘 “预测值 - 实际值”图**。

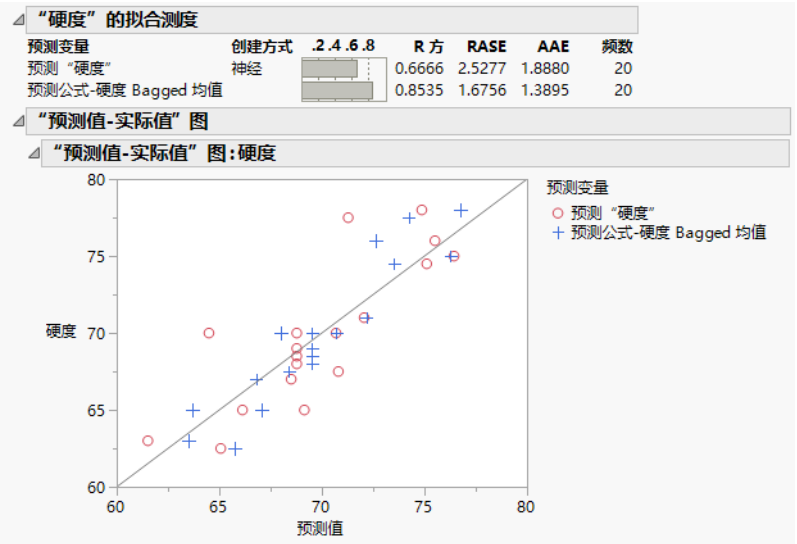
图 3.32 磨损预测的比较



“拟合测度”报表和“预测值-实际值”图显示在图 3.32 中。从 Bagging 得到的预测用蓝色显示。从原始神经网络模型得到的预测用红色显示。通常，Bagging 预测比原始模型预测更接近实际线。因为 Bagging 预测更接近实际线，因此 Bagged 预测的 R 方值 0.8379 高于原始模型预测的 R 方值。您得出结论：Bagging 改进了**磨损**的预测。

该示例比较了**磨损**的预测。要比较其他响应变量的预测，请按照第 2 步到第 6 步执行，将**磨损**替换为所需的响应变量。作为另一示例，图 3.33 显示**硬度**的“拟合测度”报表。该报表显示的结果与**磨损**的“拟合测度”报表结果相似。Bagged 预测的 R 方值略高于原始模型预测的 R 方值，这说明拟合效果更好，预测得到改进。

图 3.33 硬度预测的比较



指示预测准确度的 Bagging 示例

Bagging 还用于通过标准误差和其他分布测度指示预测的准确度。在可以在 Bagging 中使用“保存预测公式”选项的平台中，您可以对新观测进行预测并确定它们的准确度。“保存预测公式”选项在“标准最小二乘法”、“广义回归”和“广义线性模型”平台中可用。

在 Tiretread.jmp 数据表中，假定您只想预测作为三个因子变量的函数的**磨损**。在该示例中，您拟合一个广义回归模型来预测**磨损**。接着，您对该模型执行 Bagging。最后，您为新观测生成预测并调查该预测的准确度。这通过获取该预测的置信区间来完成。

拟合广义回归模型

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**磨损**，然后点击 **Y**。
4. 从“特质”列表选择**广义回归**。
5. 选择**硅石**、**硅烷**和**硫磺**，然后点击**宏** > **完全析因**。
   这会将所有项（包括交互作用）添加到该模型。
6. 点击**运行**。
7. 点击**执行**。

图 3.34 “广义回归”报表中的参数估计值

原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	-46.05865	7.526913	37.44455	<.0001*	-60.81113	-31.30617
硅石	32.987289	2.2710897	210.97194	<.0001*	28.536036	37.438543
硅烷	1.7880765	0.1194432	224.10381	<.0001*	1.5539721	2.0221809
(硅石-1.2)*(硅烷-50)	1.025	0.2316574	19.577413	<.0001*	0.5709598	1.4790402
硫磺	21.813077	1.4913761	213.92398	<.0001*	18.890033	24.73612
(硅石-1.2)*(硫磺-2.3)	28.5	4.6331484	37.838796	<.0001*	19.419196	37.580804
(硅烷-50)*(硫磺-2.3)	1.575	0.2316574	46.224125	<.0001*	1.1209598	2.0290402
(硅石-1.2)*(硅烷-50)*(硫磺-2.3)	1.55	0.4633148	11.192085	0.0008*	0.6419196	2.4580804

“正态”分布参数	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
尺度	5.2681724	0.7759427	46.095727	<.0001*	3.7473527	6.7889921

执行 Bagging

1. 点击“使用 AICc 验证的正态 Lasso”旁边的红色小三角并选择**刻画器 > 刻画器**。
   “预测刻画器”显示在报表底部。
2. 点击“预测刻画器”红色小三角并选择**保存 Bagged 预测**。
3. 在“Bootstrap 样本数”旁边输入 500。
4. （可选）在“随机种子”旁边输入 4321。

   注意：由于放回抽样是随机的，所以结果会有所不同。要重现本例的精确结果，请设置“随机种子”。

5. 确认选择了**保存预测公式**。
6. 点击**确定**。

   注意：运行它所需的时间可能比第 67 页的 “Bagging 改进预测的示例”长。更大的样本数可以给出预测分布的更好估计。

   返回数据表。对于每个响应变量，有三个分别表示为**预测公式 -< 列名 > Bagged 均值**、**标准误差 -< 列名 > Bagged 均值**、**< 列名 > Bagged 标准差**的新列。**预测公式 - 磨损 Bagged 均值**列是最终预测。

新观测的预测

您现在有数据表中每个观测的磨损预测以及这些预测的标准误差。假定您有一个观测，对应硅石、硅烷和硫磺的新值分别为 0.9、43 和 2。您可以预测磨损响应并获得该预测的置信区间，因为“保存预测公式”选项保存每个 Bagged 模型的回归方程。因此，使用新因子值生成 *M* 个预测，以创建可能的预测分布。该均值是最终预测，但是分析分布可以告诉您预测的准确度。

1. 在数据表中，选择**行 > 添加行**。
2. 在**要添加的行数**框中输入 1，然后点击**确定**。
3. 在**硅石**列下，在新行对应的框中键入 0.9。
4. 在**硅烷**列下，在新行对应的框中键入 43。

5. 在**硫磺**列下，在新行对应的框中键入 2。
   自动计算新行的所有预测列。

图 3.35 新行的值

	磨损	弹性系数	伸长	硬度	硅石	硅烷	硫磺	预测公式-磨损	预测公式-弹性系数
1	102	900	470	67.5	0.7	40	2.8	96.7475395	747.638719
2	120	860	410	65	1.7	40	1.8	113.421752	1074.97186
3	117	800	570	77.5	0.7	60	1.8	114.695993	1011.176
4	198	2294	240	74.5	1.7	60	2.8	195.496359	2173.94727
5	103	490	640	62.5	0.7	40	1.8	104.934463	865.669653
6	132	1289	270	67	1.7	40	2.8	133.734829	1333.44092
7	132	1270	410	78	0.7	60	2.8	138.00907	1310.64507
8	139	1090	380	70	1.7	60	1.8	143.683282	1497.9782
9	102	770	590	76	0.3835	50	2.3	101.492758	600.398164
10	154	1690	260	70	2.0165	50	2.3	155.361002	1476.17966
11	96	700	520	63	1.2	33.67	2.3	100.727591	525.749229
12	163	1540	380	75	1.2	66.33	2.3	159.12617	1330.8286
13	116	2184	520	65	1.2	50	1.4835	117.116503	1564.51067
14	153	1784	290	71	1.2	50	3.1165	152.737257	2020.06715
15	133	1300	380	70	1.2	50	2.3	139.119239	1261.13314
16	133	1300	380	68.5	1.2	50	2.3	139.119239	1261.13314
17	140	1145	430	68	1.2	50	2.3	139.119239	1261.13314
18	142	1090	430	68	1.2	50	2.3	139.119239	1261.13314
19	145	1260	390	69	1.2	50	2.3	139.119239	1261.13314
20	142	1344	390	70	1.2	50	2.3	139.119239	1261.13314
21	•	•	•	•	0.9	43	2	114.489091	931.321632

6. 选择**表** > **转置**。
7. 选择 “**磨损**” Bags (500/0) 并点击**转置列**。
8. 点击**确定**。
9. 选择**分析** > **分布**。
10. 选择**第 21 行**并点击 **Y，列**。

注意：第 21 行对应新观测的预测。

11. 点击**确定**。
12. 点击 “第 21 行”旁边的红色小三角并选择**显示选项** > **水平布局**。

图 3.36 “分布”报表

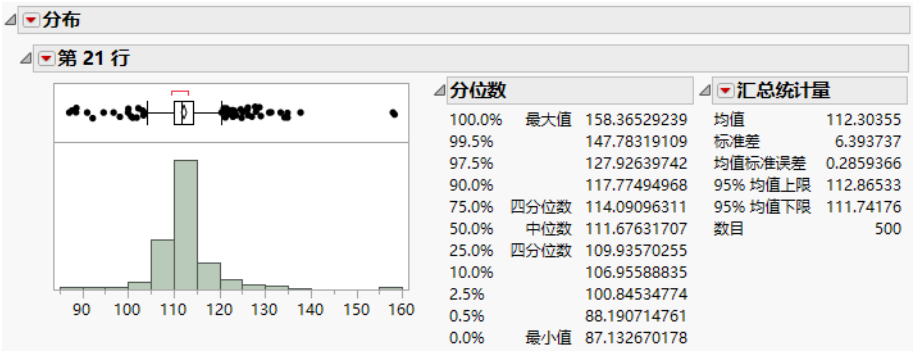


图 3.36 中的“分布”报表包含有关每个 Bagged 模型的磨损预测值的分布信息。新观测的磨损最终预测值是 112.3，它是所有 *M* 个 Bagged 预测的均值。该预测的标准误差为 6.39。您还使用分位数创建新预测的置信区间。例如，新预测的 95% 置信区间为 100.85 到 127.93。

预测刻画器的统计详细信息

- “评估变量重要性”
- “误差传播直条”
- “外推控制量度”

评估变量重要性

随后的详细信息与如何计算变量重要性指标有关。

背景

将代表预测模型的函数表示为 *f*，并假定 $x_1, x_2, ..., x_n$ 为模型中的因子或主效应。令 $y = f(x_1, x_2 ..., x_n)$。

- y 的期望值 $E(y)$ 通过使用 $x_1, x_2, ..., x_n$ 的联合分布求 y 的积分来定义。
- y 的方差 $Var(y)$ 通过使用 $x_1, x_2, ..., x_n$ 的联合分布求 $(y - E(y))^2$ 的积分来定义。

主效应

主效应 x_j 对 y 的影响可通过 $Var(E(y \mid x_j))$ 来描述。此处期望值是 $x_1, x_2, ..., x_n$ 的条件分布下给定 x_j 时的取值，而方差是在 x_j 分布下的取值。换言之，$Var(E(y \mid x_j))$ 在 x_j 分布下测量当 x_j 固定时 y 均值中的变异。

同样，$Var(E(y \mid x_j))/Var(y)$ 之比测量的是 y 对于因子 x_j 的灵敏度。“汇总报表”的“主效应”列中的重要性指标是该比率的估计值（请参见第 74 页的“抽样变异的调整”）。

总效应

“总效应”列表示涉及 x_j 的**所有**项对 $y = f(x_1, x_2 ..., x_n)$ 方差的总贡献。“主效应”的计算取决于函数分解的概念。函数 f 分解为一个常数与各函数（表示单个变量、成对变量等的效应）之和。这些构成函数类似于主效应、交互作用效应和高阶效应。请参见 Saltelli (2002)；Sobol (1993)。

这些具有包含 x_j 的项的构成函数将被标识出来。对于以上每个函数，都会计算条件预期值的方差。这些方差将被求和。总和表示因包含 x_j 的项而对 $Var(y)$ 的总贡献。对于每个 x_j，将使用用于生成输入的选定方法估计该总和。“总效应”列中报告的重要性指标即这些估计值（请参见第 74 页的“抽样变异的调整”）。

考虑一个包含 x_1 和 x_2 这两个因子的简单示例。x_1 的“总效应”重要性指标为以下公式计算得出的估计值：

$$\frac{Var(E(y\ |x_1)) + Var(E(y\ |x_1, x_2))}{Var(y)}$$

抽样变异的调整

由于“汇总表”中显示的“主效应”和“总效应”估计值是使用抽样方法获得的，因此这些估计值可能已经过调整。具体而言，若“总效应”估计值小于“主效应”估计值，则“总效应”重要性指标将设置为等于“主效应”估计值。若“主效应”估计值超过 1，则这些估计值的总和将被标准化为 1。

变量重要性标准误差

为独立输入提供的标准误差可测量 Monte Carlo 复制值的准确度。重要性指标计算如下：

- 拉丁超立方抽样用于生成一组数据值。
- 对于每组数据值，计算主效应和总效应重要性估计值。
- 该过程将重复执行，直到所有因子的“主效应”和“总效应”重要性指标的估计标准误差都降低至阈值 0.01 以下。

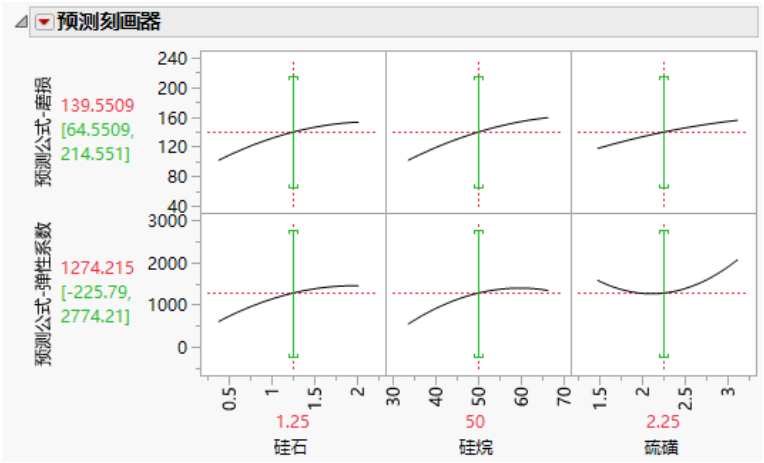
报告的标准误差是复制终止时的有效标准误差值。

误差传播直条

当因子值不太可控，而要根据因子值变异刻画响应变异的特性时，误差传播 (POE) 十分重要。

在 JMP 实现中，“预测刻画器”首先查看因子和响应变量以检查是否存在 Sigma 列属性（用于指定列的标准差，可通过“列”>“列信息”对话框访问）。若该属性存在，则可通过“预测刻画器”下拉菜单访问“误差传播直条”命令。这将会显示因为因子变异而在响应上隐含的 3σ 区间。

图 3.37 “预测刻画器”中的绿色误差传播直条



误差传播在图形中通过绿色方括号来表示。方括号指示预测值加/减三倍的误差传播方差平方根。误差传播方差可表示如下：

$$\sum_{i=1}^{N}\left(\sigma_{x_i}^2\times\left(\frac{\partial f}{\partial x_i}\right)^2\right)+\sigma_y^2$$

其中 σ_y 是响应列的用户指定的 sigma，σ_x 是因子列的用户指定的 sigma。

当前，通过数值导数来计算这些偏导数：

centered, with δ=xrange/10000

当您处于响应曲面较陡峭的部分时，响应曲面模型中的误差传播限值会显著增加。稳健过程的目标之一是在响应曲面的平缓区域操作，这样因子中的变异不会放大其对响应的效应。

外推控制量度

杠杆率

对于在“拟合模型”平台的“标准最小二乘法”特质中拟合的模型，因子设置的杠杆率用作外推量度。

第 i 个观测的杠杆率 h_{ii} 是矩阵 $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$（有时称为帽子矩阵）的第 i 个对角线元素。新预测点的杠杆率计算如下：$h_{\text{pred}}=\mathbf{x}'_{\text{pred}}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_{\text{pred}}$。以下两个准则可用于确定具有杠杆率 h_{pred} 的预测是否为外推：

- $h_{\text{pred}}>K\times\max(h_{ii})$，其中 K 是可定制的乘数
- $h_{\text{pred}}>L\times p/n$，其中 L 是可定制的乘数，p 是变量数，n 是观测数，p/n 是平均杠杆率

可以使用“设置阈值准则”选项指定使用哪种准则以及乘数的值。默认乘数值为 $K = 1$ 和 $L = 3$。

注意：从“图形”菜单运行的刻画器上的外推控制若使用保存的最小二乘法模型，则不执行杠杆方法，改为使用“正则 Hotelling T^2”方法。

正则 Hotelling T^2

在最小二乘法模型之外的模型中，正则 Hotelling T^2 值用作外推量度。训练数据的 T^2 值和预测点的 T^2 值计算如下：

$$T^2 = (x - \bar{x})' \hat{\Sigma}^{-1} (x - \bar{x})$$

$$T^2_{\text{预测}} = (x_{\text{预测}} - \bar{x})' \hat{\Sigma}^{-1} (x_{\text{预测}} - \bar{x})$$

其中 $\hat{\Sigma}$ 是对训练数据估计的 Schafer 和 Strimmer 正则协方差矩阵估计量。用于 Schafer Strimmer 估计量的目标矩阵是对角协方差矩阵。请参见 Schafer and Strimmer (2005)。在使用带有缺失值的观测训练模型的平台中，用配对删除来估计协方差矩阵。

注意：在上述计算中，分类变量要转换为指标变量。

阈值的计算取决于根据训练数据计算的非缺失 T^2 值的数量。

- 若有十个或更多非缺失 T^2 值，则阈值设置如下：

  $$\overline{T^2} + K\hat{\sigma}_{T^2}$$

  其中

  K 是可定制乘数且默认设置为 3

  $\hat{\sigma}_{T^2}$ 是 T^2 值的标准差。

- 若非缺失值 T^2 不足十个，则使用等价于 $K\sigma$ 限值的 F 分布分位数来设置阈值。

  $$Q_{T^2}(q)$$

  其中

  $q = \Phi(K)$

  $\Phi(\cdot)$ 是标准正态分布

  K 是可定制乘数且默认设置为 3

  $$T^2 \sim \frac{(n+1)(n-1)p}{n(n-p)} F(p, n-p)$$

p 是参数个数

r 是非缺失 T^2 值的个数

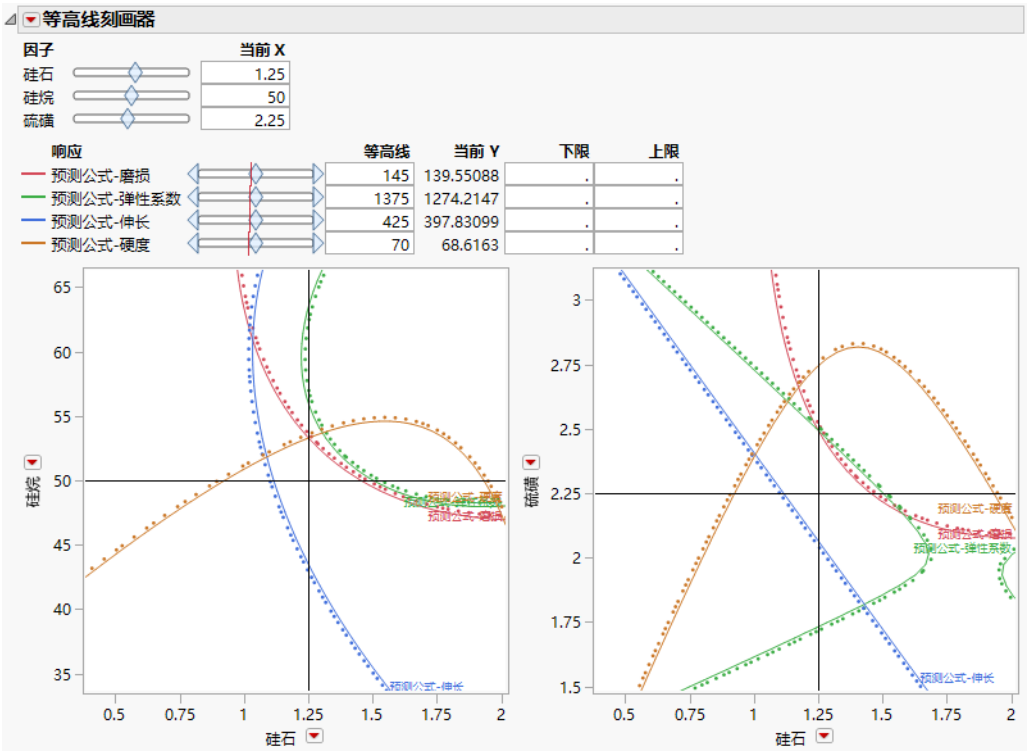
第 4 章

等高线刻画器

探索两个因子的响应等高线

“等高线刻画器”可同时显示两个因子的响应等高线图。可以同时查看不同因子组合的多个等高线图。交互式等高线刻画工具适用于以图形方式优化响应曲面。

图 4.1 “等高线刻画器”示例



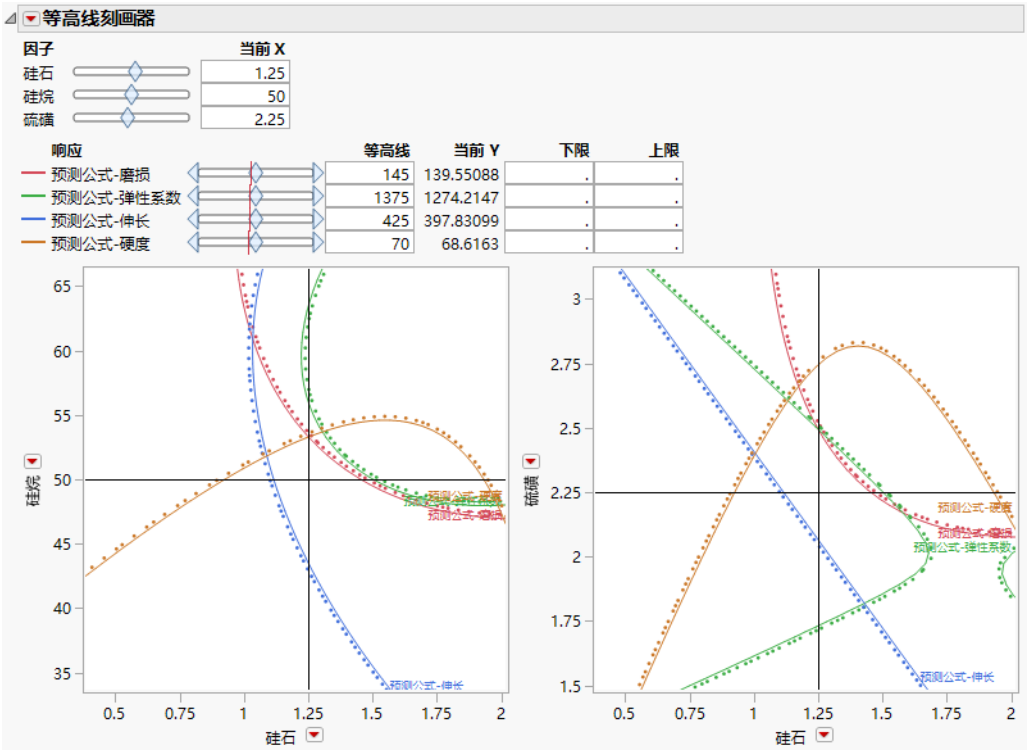
目录

“等高线刻画器”概述 81
等高线刻画器的示例 81
启动“等高线刻画器”平台 83
“等高线刻画器”报表 83
    因子设置和控件 83
    响应设置和控件 84
“等高线刻画器”平台选项 84
约束着色设置 86
“等高线刻画器”的其他示例 86
    探索最优设置 86

“等高线刻画器”概述

“等高线刻画器”可同时显示两个因子的响应等高线图。可以同时查看不同因子组合的多个等高线图。可将因子设置为特定值来显示它如何影响等高线。交互式等高线刻画工具适用于以图形方式优化响应曲面。还为每个响应提供单独的曲面图。图 4.2 显示 Tiretread.jmp 样本数据的等高线刻画器的示例。

图 4.2 等高线刻画器



等高线刻画器的示例

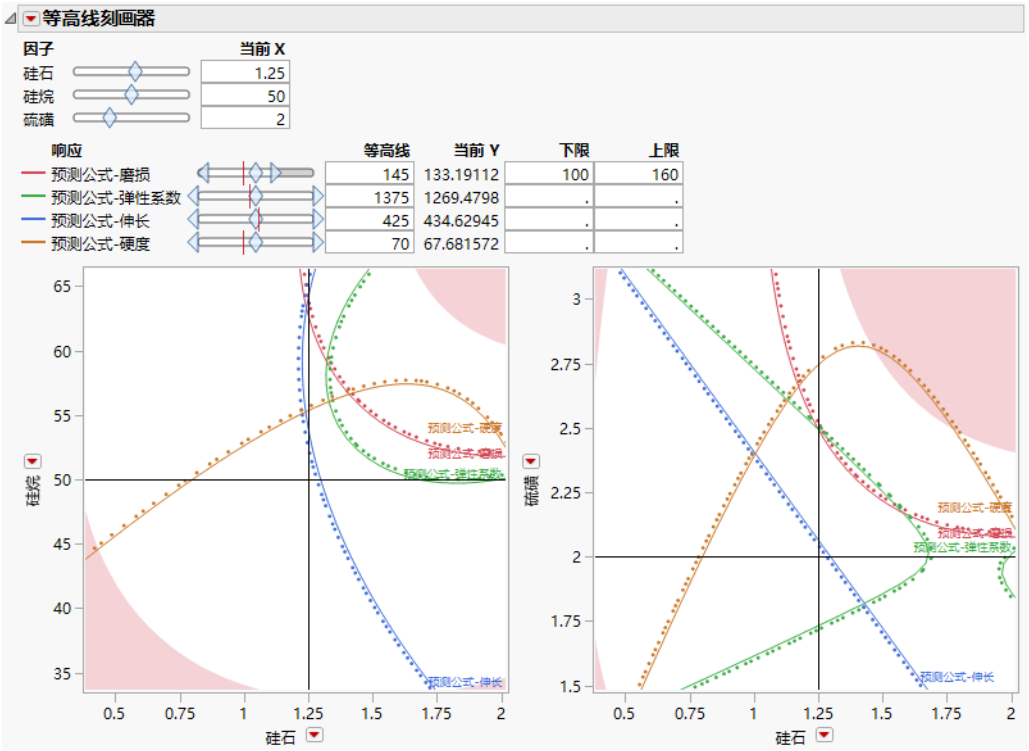
本例使用 Tiretread.jmp 样本数据表。共有三个因子（**硅石**、**硅烷**和**硫磺**）和四个响应（**磨损**、**弹性系数**、**伸长**和**硬度**）。该数据表已经包含四个响应变量的已保存预测公式列。

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**图形** > **等高线刻画器**。
3. 选择**预测公式 - 磨损**、**预测公式 - 弹性系数**、**预测公式 - 伸长**和**预测公式 - 硬度**，然后点击 **Y，预测公式**。
4. 点击**确定**。

5. 点击 “等高线刻画器”红色小三角并选择**多个等高线框架**。
6. 在 “水平因子”下，选择**硅石**，在 “垂直因子”下选择**硫磺**。
7. 点击**确定**。

   在这两个等高线图上，**硅石**的值都位于水平轴上。**硅烷**和**硫磺**的值位于垂直轴上。在第一个等高线图上，**硫磺**的值固定为 2.25。在第二个等高线图上，**硅烷**的值固定为 50 （图 4.2）。
8. 点击**硫磺**的**当前 X** 框并键入 2。点击该框外部的任意地方可以设置该值。
9. 在**预测公式 - 磨损**的**下限**框中，键入 100。
10. 在**预测公式 - 磨损**的**上限**框中，键入 160。
11. 点击该框外部的任意地方可以更新图。

图 4.3 Tiretread.jmp 的等高线刻画器



第一个等高线图中的等高线与图 4.2 中所示的内容不同，因为为**硫磺**设置的值不同。这可以从第二个等高线图中位于 2 处的水平线的位置看出来。移动**硫磺**的滑块以查看不同的值如何进一步改变第一个等高线图中的等高线。移动“硅烷”的滑块以查看不同的值如何改变第二个等高线图中的等高线。等高线图上的着色区域表示根据**磨损**的“下限”和“上限”值定义的不可行的图形区域。

启动“等高线刻画器”平台

可以通过以下方式访问 “等高线刻画器”：

- 可以直接从“图形”菜单访问“等高线刻画器”。使用该方式访问“等高线刻画器”时，显示“等高线刻画器”启动窗口。请参见“刻画器介绍”一章中第 28 页的““刻画器”启动窗口”。
- 在很多建模平台中，“等高线刻画器”可以作为红色小三角菜单选项来访问。有关在不同平台中是否提供“等高线刻画器”的详细信息，请参见“刻画器介绍”一章中第 26 页的“JMP 刻画器的提供位置”。
- 可以从“模型比较”平台访问“等高线刻画器”。从“模型比较”红色小三角菜单中选择“刻画器”。然后，从“刻画器”红色小三角菜单中选择“等高线刻画器”。
- JMP PRO 可以从“公式存储库”平台访问“等高线刻画器”。从“公式存储库”红色小三角菜单中选择“刻画器”，然后选择要刻画的模型。然后，从“刻画器”红色小三角菜单中选择“等高线刻画器”。

“等高线刻画器”报表

初始 “等高线刻画器”报表显示一个等高线刻画器图、因子设置和控件以及响应设置和控件。图形上的彩色线是用 Y 滑块控件设置的或通过在**等高线**列中输入值设置的响应的等高线。每个响应都有单独的彩色等高线。对于每条等高线，在较高的响应值方向上会有虚线，让您有方向感。若要将因子分配更改为等高线的水平轴和垂直轴，请点击轴上的红色小三角并选择一个因子。若选择相反轴上的因子，则因子分配互换。

- “因子设置和控件”
- “响应设置和控件”

因子设置和控件

因子 因子的列表。

当前 X 当前因子设置。在框中点击可以更改因子的值或使用滑块控件。右击滑块控件并选择**重新调整滑块**可更改滑块的尺度。当更改了一个值时，其他因子的值仍保持不变，但是响应设置和控件报表中的值会相应更改。**当前 X** 位置通过图形上的十字准线指示。

锁定 为包含三个以上预测变量的混料设计显示的列。该列支持您锁定混料值的设置，这样，由于其他混料效应更改而导致需要调整混料时，这些设置不会更改。若存在锁定列，混料的着色区域可识别新的限制区域。

图 4.4 锁定列

注意：若一个或多个因子有规格限，则在等高线图上绘制一个点线构成的矩形，以显示联合规格限范围。

响应设置和控件

响应 具有颜色图例的一个或多个响应的列表。要更改响应的颜色，请右击该响应的颜色图例。

等高线 等高线刻画器上等高线的当前值。在框中点击可更改该值。也可以使用滑块控件更改等高线的值。

当前 Y 基于当前 X 设置的预测响应。该值位于滑块控件的红线上。该值随着因子设置的更改更新。

下限 支持您设置响应的下限。在框中点击设置该限值或点击该响应的滑块控件左侧的三角。若响应列的“规格限”列属性具有“下规格限”值，则该值作为**下限**的初始值。

上限 支持您设置响应的上限。在框中点击设置该限值或点击该响应的滑块控件右侧的三角。若响应列的“规格限”列属性具有“上规格限”值，则该值作为**上限**的初始值。

使用响应限时，刻画器显示不可行的区域（在等高线刻画器图中着色）。

注意：若响应具有“响应概率”列属性，则“上限”自动设置为 0.5。在这种情况下，着色区域是预测响应水平的区域。

“等高线刻画器”平台选项

网格密度 设置曲面图中网格图的密度。

图形更新 包含用于更改图形更新频率的选项。（在速度较快的机器上，这种差异不易察觉。）提供以下两个选项：

每次移动鼠标 拖动鼠标时连续更新图形。这是默认设置。

每次释放鼠标 每次释放鼠标时更新图形。

曲面图 显示或隐藏网格图。

等高线标签 显示或隐藏等高线的标签。标签颜色与等高线颜色相匹配。

等高线网格 在“等高线刻画器”图上以您指定的间隔绘制等高线。

删除等高线网格 支持您在绘制等高线后删除等高线网格。

因子设置 提供命令子菜单，该子菜单支持您将“等高线刻画器”的设置保存并传输至 JMP 的其他部分。请参见“刻画器”一章中第 44 页的“因子设置”。

模拟器 启动模拟器。请参见第 137 页的“模拟器”一章。

上方带点 显示或隐藏对应每个等高线的虚线。这些虚线显示增大响应值的方向。

将等高线设置为当前位置 将等高线重置为当前 Y 值所在的位置。这意味着所有等高线将穿过等高线图上的十字准线位置，各控件的值将与 Y 滑块值一致。

左侧排列 X 控件 重新排列 X 和 Y 控件，水平排列时 X 控件在左侧，垂直排列时 X 控件在顶部。

隐藏 X 控件 显示或隐藏 X 控件 （“因子设置和控件”）。

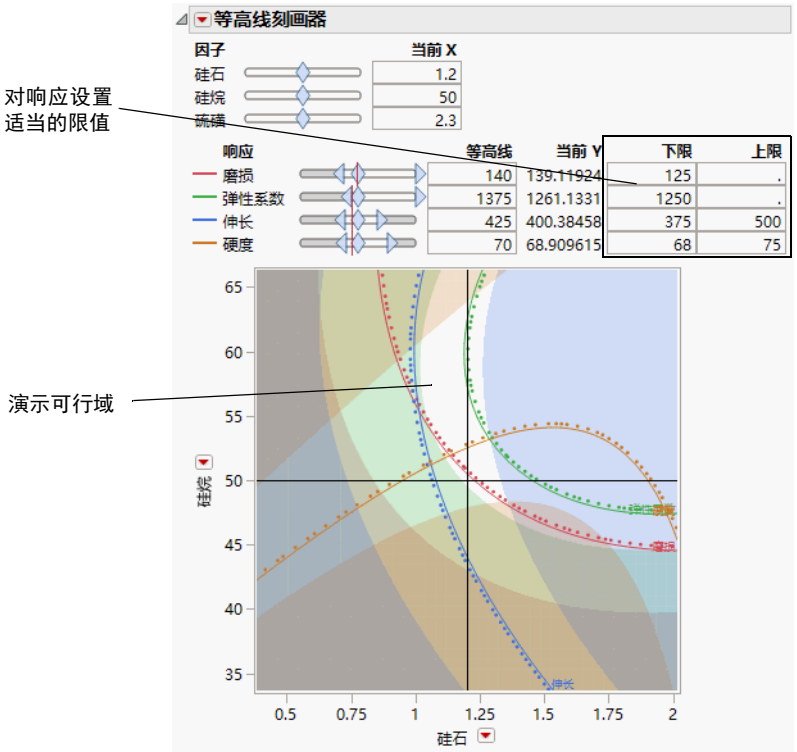
隐藏 Y 控件 显示或隐藏 Y 控件 （“响应设置和控件”）。

多个等高线框架 支持您将一个或多个等高线图添加到表示不同因子设置组合的报表中。选择该选项时，将显示“多个等高线框架”对话框。使用该对话框可以指定其他单个等高线图或等高线图矩阵。您可以指定等高线框架的下三角、上三角或全正方形矩阵。默认情况下，新等高线图将添加到报告中的当前等高线图中。点击**删除之前的框架**复选框可删除当前等高线图并用新选择替换它们。

约束着色设置

若您指定响应列的限值，则为可行域之外的图区域着色。未着色的白色区域表示可行域。

图 4.5 等高线着色设置



“等高线刻画器”的其他示例

探索最优设置

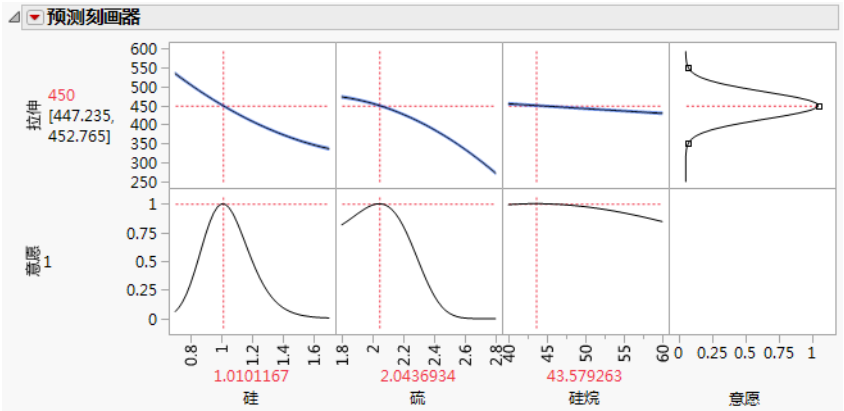
意愿函数通常用于确定响应曲面的最优设置。“预测刻画器”中的“最大化意愿”功能提供了一个因子水平设置组合，它可以得到优化意愿的预测响应。但是，通常有很多可优化意愿函数的因子水平组合。等高线刻画器对于找到优化意愿的备选设置是有用的工具。

该示例使用 Bounce Data.jmp 数据表，它包含轮胎胎面实验的数据。响应**拉伸**是**硅石**、**硅烷**和**硫磺**这三个变量的函数。将构建一个模型，并使用 “预测刻画器”中的 “最大化意愿”选项确定最优设置 （通过**拉伸**值使得意愿函数最大化）。您使用 “等高线刻画器”探索针对指定的**拉伸**值的其他最优设置。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Bounce Data.jmp。

2. 点击**模型**脚本旁边的绿色小三角，然后在启动窗口中点击**运行**。
3. 点击 “预测刻画器”红色小三角并选择**优化和意愿** > **最大化意愿**。

图 4.6 最大化意愿设置

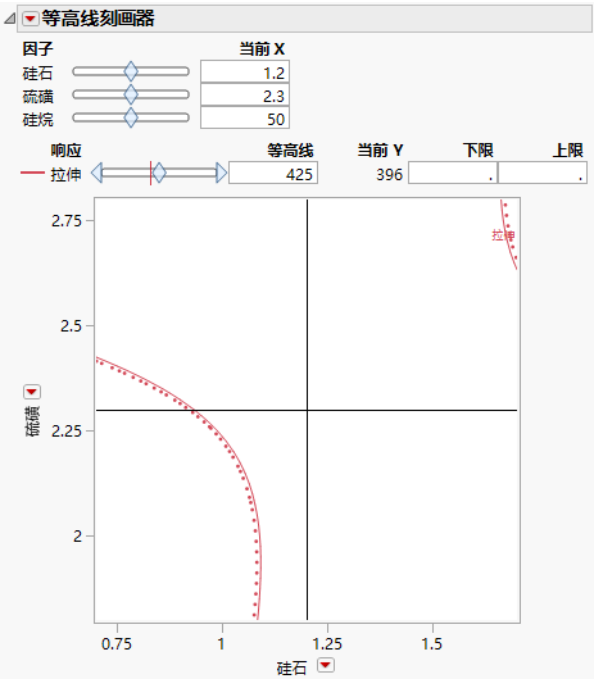


使意愿函数最大化的**拉伸**值为 450。可得到预测的**拉伸**值 450 的一个因子水平设置组合为：**硅石** = 1.0101，**硫磺** = 2.0437 且**硅烷** = 43.5792。

4. 点击 “响应拉伸”红色小三角并选择**因子刻画** > **等高线刻画器**。

   为了确保您的新设置组合使意愿最大化，您想确保设置在 450 的 2 个单位内预测**拉伸**。此外，您需要一个高水平的**硅烷**值 （固定设置为 60）。

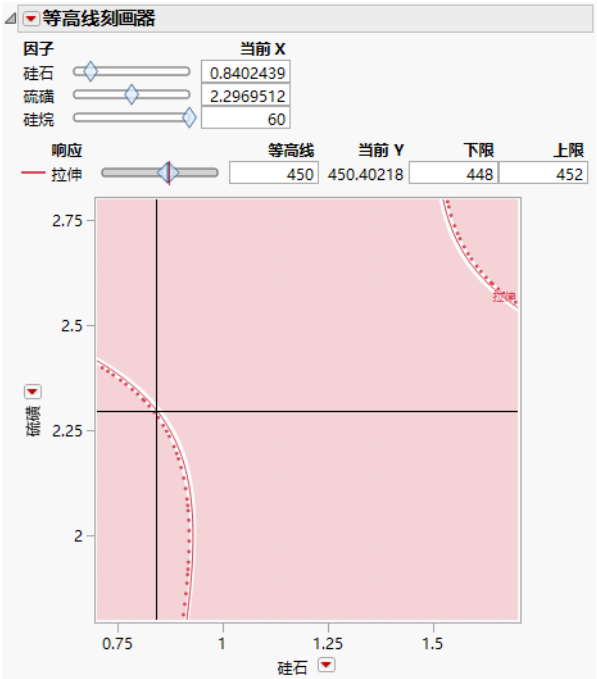
图 4.7 Bounce Data.jmp 的等高线刻画器



初始图显示**拉伸**值为 425 且**硅烷**值为 50 时**硅石**和**硫磺**的等高线值。

5. 在响应控件中，将**拉伸**的**等高线**值设置为 450。将**拉伸**的**下限**和**上限**分别设置为 448 和 452。按 Enter。
6. 将**硅烷**的**当前 X** 设置为 60。

图 4.8 最优设置的等高线刻画器



红色的实曲线上的值是预测**拉伸**值将为 450 时的**硅石**和**硫磺**值。未着色带内的值预测**拉伸**值介于 448 和 452 之间。您可以将图上显示的十字准线拖到未着色部分来查找**硅石**和**硫磺**的特定设置。图 4.8 中显示了一个这样的设置。

第 5 章

曲面图

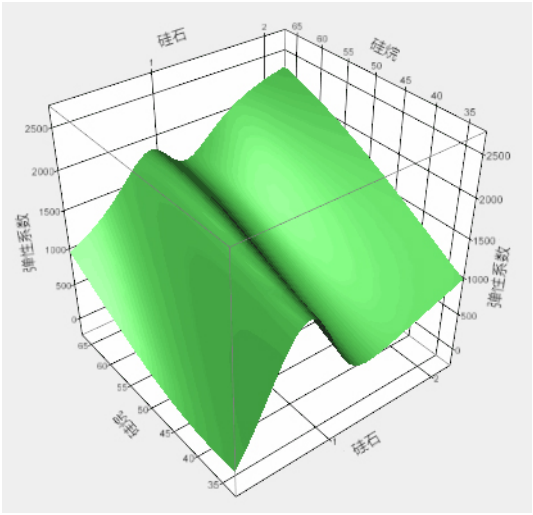
探索三个因子的响应等高线

曲面图是用平滑曲面表示一个或多个因变量的三维图。您可以在同一个曲面图上至多显示四个因变量。

JMP 在以下两种情况下生成曲面图：

- “曲面图”平台创建了包含公式的曲面图的独立报表。公式可以是数据表中的公式列，也可以是不涉及任何数据点的数学公式。
- 许多模型拟合平台中的“曲面刻画器”选项都会为现有平台报表中的拟合模型生成曲面图。

图 5.1 “曲面图”示例



目录

“曲面图”和“曲面刻画器”的概述 93
“曲面图”平台的示例 93
启动“曲面图”平台 96
“曲面图”报表 98
曲面图 98
外观控件 99
自变量 100
因变量 101
“曲面图”平台选项 102
弹出式菜单选项 103
因变量选项 104
面或曲面属性 104
键盘快捷方式 105
“曲面图”平台的更多示例 106
为单个数学函数构造曲面图 106
标绘等值面 107

“曲面图”和“曲面刻画器”的概述

“曲面图”平台和“曲面刻画器”平台用于标绘三维的点和曲面。

曲面图既可作为单独平台提供（图形 > 曲面图），也可作为许多报表中的选项提供（称为曲面刻画器）。曲面图功能在这两种情况下是类似的。

图形可由点和/或曲面构成。若点在曲面图中可见，则这些点链接到数据表。您可以点击这些点或使用笔刷工具选择数据表中的行。在数据表中分配的颜色和标记也会体现在曲面图中。

可通过数学方程或定义多边形曲面的一组点来定义曲面。每个曲面都可显示为带有或不带等高线的平滑曲面或网线。曲面图的标签、轴和光源全部支持定制。

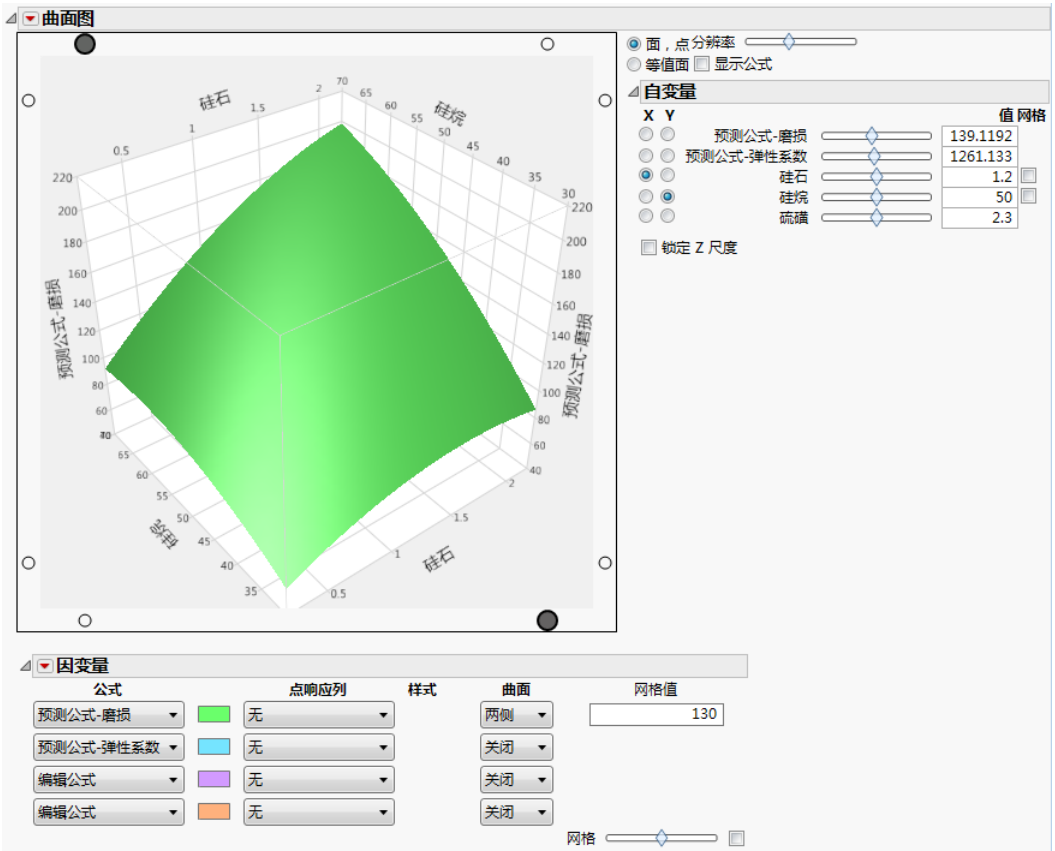
需要使用 JMP 脚本语言 (JSL) 中的三维场景命令生成曲面图。在曲面图中右击时，弹出式菜单包含 Open GL 场景命令。有关 Open GL 场景命令的详细信息，请参见《Scripting Guide》。

“曲面图”平台的示例

本例使用 Tiretread.jmp 样本数据表。共有三个因子（**硅石**、**硅烷**和**硫磺**）和四个响应（**磨损**、**弹性系数**、**伸长**和**硬度**）。该数据表已经包含四个响应变量的已保存预测公式列。使用这些预测公式列可检查其中两个响应变量的曲面。

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**图形** > **曲面图**。
3. 选择**预测公式 - 磨损**和**预测公式 - 弹性系数**，点击**列**，然后点击**确定**。

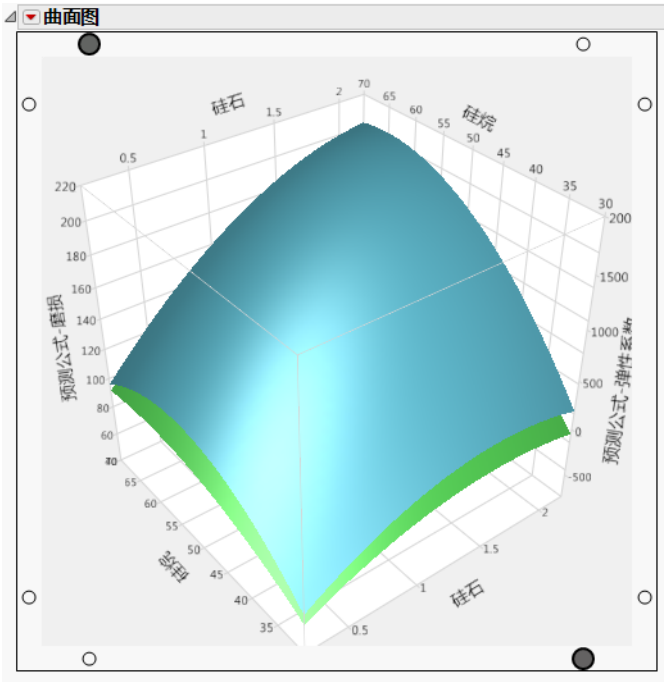
图 5.2 初始 “曲面图”报表



最初，图中仅显示 “磨损”的预测曲面。请注意，在 “自变量”报表中，“磨损”和 “弹性系数”的值之间有较大差异。这两个响应位于不同的尺度上。

4. 在 “因变量”报表中，从**预测公式 - 弹性系数**的 “曲面”列表中选择**两侧**。
5. 点击 “曲面图”红色小三角并选择**独立调整响应轴尺度**。

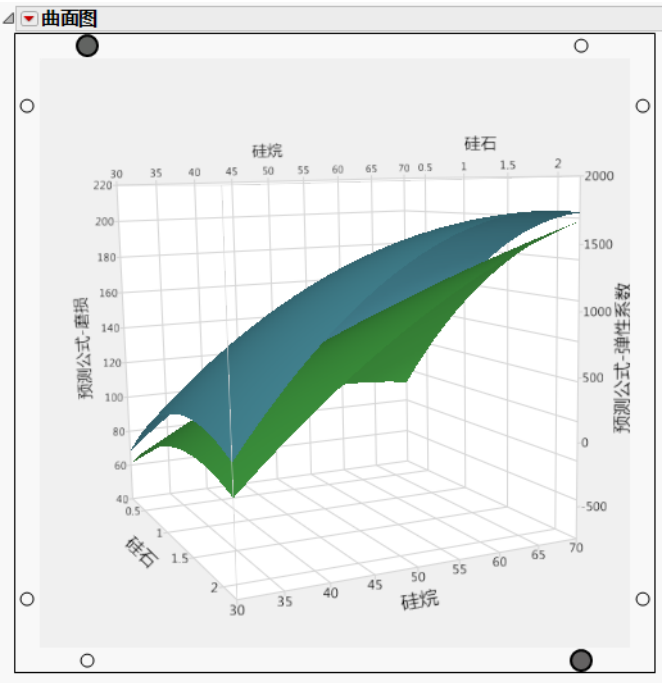
图 5.3 “磨损”和“弹性系数”的预测曲面



预测公式 - 磨损位于一个 Z 轴上，**预测公式 - 弹性系数**位于另一个 Z 轴上。每个响应都有自己的尺度。在该图中，考虑的两个因子是**硅石**和**硅烷**。您可以通过在图形中点击并拖动来旋转曲面图。这样您就可以从不同角度查看预测曲面。例如，在图 5.4 中，您可以看到基于两个列出的预测变量的“磨损”和“弹性系数”的预测具有相似的曲面形状，只是位于不同的尺度上。使用“自变量”控件可查看预测变量的不同组合如何影响曲面图。

提示：要在旋转后查看初始视图，请右击曲面图中的任意位置并选择**重置**。

图 5.4 旋转的预测曲面

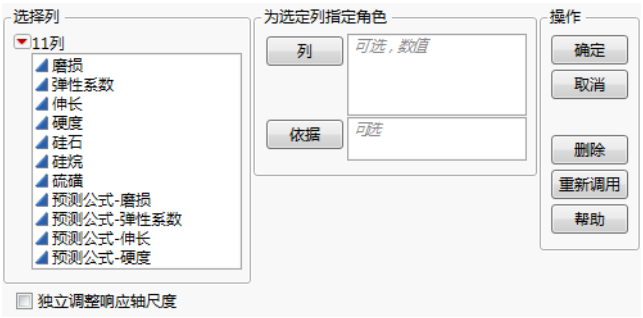


启动 “曲面图” 平台

可以通过以下方式访问 “曲面图刻画器”：

- 要直接启动 “曲面图刻画器” 平台，请从 “图形” 菜单中选择 “曲面图”。若已打开某个数据表，则将显示图 5.5 中所示的窗口。 若不想使用数据表来绘制曲面图，请点击 “确定” 而不指定任何列。若未打开任何数据表，则为您显示图 5.13 中所示的默认曲面图。
- 在许多建模平台中都可以访问作为红色小三角菜单选项的 “曲面图刻画器”。它在红色小三角菜单中称为 “曲面刻画器”。有关在不同平台中是否提供 “曲面刻画器” 的详细信息，请参见 “刻画器介绍” 一章中第 26 页的 “JMP 刻画器的提供位置”。
- 可以从 “模型比较” 平台访问 “曲面图刻画器”。从 “模型比较” 红色小三角菜单中选择 “刻画器”。然后，从 “刻画器” 红色小三角菜单中选择 “曲面刻画器”。
- JMP PRO 可以从 “公式存储库” 平台访问 “曲面图刻画器”。从 “公式存储库” 红色小三角菜单中选择 “刻画器”，然后选择要刻画的模型。然后，从 “刻画器” 红色小三角菜单中选择 “曲面刻画器”。

图 5.5 “曲面图”启动窗口



列 指定要标绘的列。只能为“列”角色分配数值变量。

依据 指定变量，以便为该变量的每个水平创建单独的曲面图。

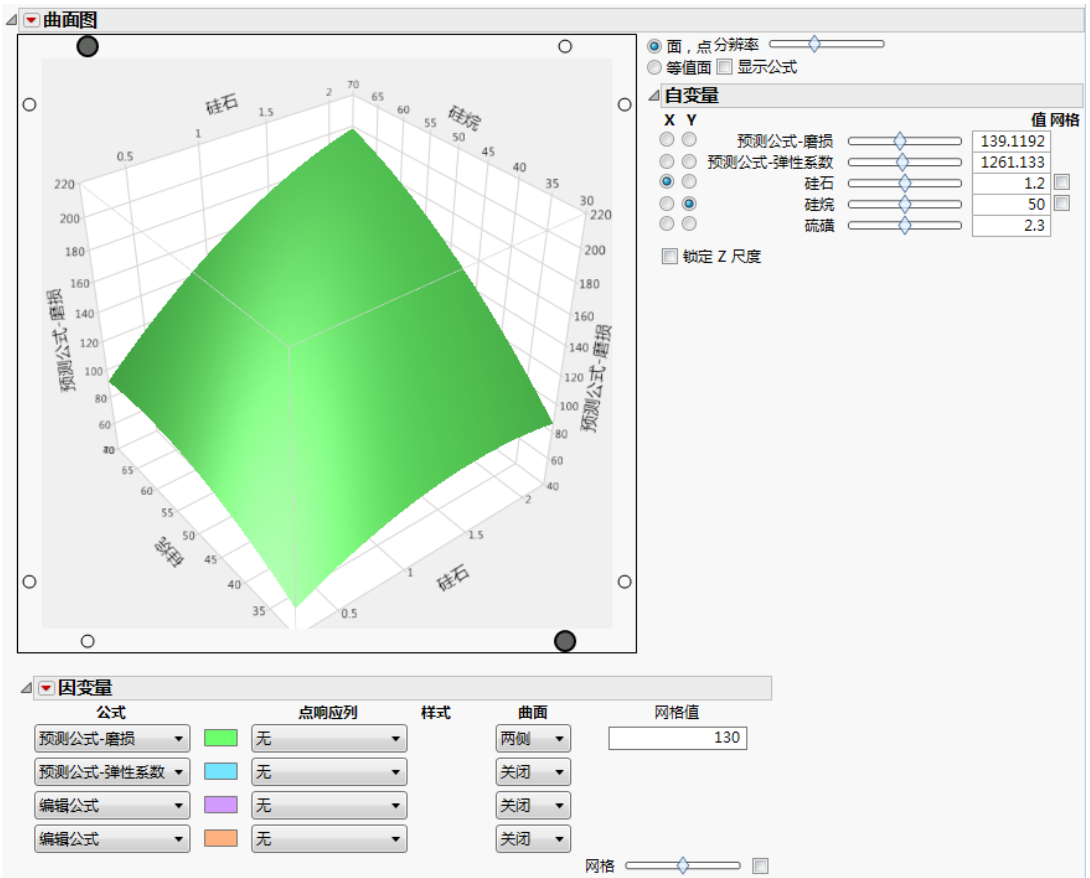
独立调整响应轴尺度 为曲面图上的每个响应分配单独的尺度。

注意: 若未选定该选项，所有响应的轴尺度都与在“列”角色中输入的第一个项的尺度相同。

“曲面图”报表

初始 “曲面图”报表显示曲面图、外观控件、自变量控件和因变量控件。若指定了不止一个预测公式列，则在初始报表中仅显示第一个曲面。不过，若指定了不止一个观测的响应列，则所有点都显示在初始报表中。

图 5.6 “曲面图”报表的示例



曲面图

曲面图具有以下控件和设置：

旋转 在曲面图上出现光标和环形箭头图标 的任意位置处点击，并拖动以便在任意方向上旋转该图。

注意：还可以使用上、下、左、右箭头键来旋转图。

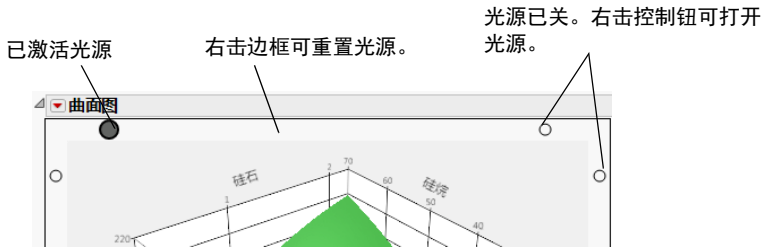
轴设置 双击某个轴可显示轴设置窗口。该轴设置窗口支持您更改尺度、刻度标增量和轴标签设置。和其他 JMP 图形一样，可以使用抓手工具调整、拉伸或收缩轴。悬停在轴上方以便使用抓手工具。

光源 默认情况下，图中有光源可照亮该图。图中有八个控制钮，可用于更改光源的位置和颜色。这有助于突出显示图中不同部分，形成鲜明对比。图 5.7 显示了八个控制钮中的四个。

您可以执行以下操作以调整照亮该曲面的光源：

- 右击某个控制钮可打开或关闭光源。打开的光源较多可使图形变亮，打开的光源较少可使图形变暗。
- 拖动某个控制钮可更改光源的位置。
- 通过右击该控制钮可更改光源的颜色。默认颜色为白色。

图 5.7 光源控制钮



外观控件

若从建模平台的红色小三角菜单中选择“曲面刻画器”选项，则“外观控制”选项不同于独立的“曲面图”平台中的“外观控件”。请参见图 5.8，了解关于外观控件在建模平台中如何显示的示例。

曲面图外观控件

报表右上的一组控件支持您指定曲面图的整体外观。

面，点 用于显示面、点和线的设置。

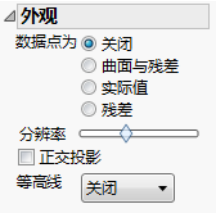
等值面 更改显示以演示等值面。请参见第 107 页的 “标绘等值面”。

显示公式 显示公式编辑框，该框支持您编辑曲面的公式。仅当在启动窗口中指定了包含公式的列时，公式编辑框才显示。

分辨率 一个影响为公式求值的点数的滑块。若分辨率太低，则可能无法非常准确地表示有急剧变化的函数。若分辨率设置得太高，则计算和显示曲面的速度会比分辨率较低时要慢。

曲面刻画器外观控件

图 5.8 外观控件



建模平台中的“外观控件”包含以下控件：

数据点为 包含控制数据点外观的以下选项：

关闭 关闭数据点的显示。

曲面与残差 在曲面上显示预测值与实际值之间的差值。

实际值 显示实际数据点。

残差 显示残差值（若在图中未关闭其显示）。

分辨率 一个影响为公式求值的点数的滑块。若分辨率太低，则可能无法非常准确地表示有急剧变化的函数。若分辨率设置得太高，则计算和显示曲面的速度会比分辨率较低时要慢。

正交投影 更改图形的视图，以便各轴构成直角。

等高线 确定等高线在图中相对于曲面的位置。选项包括关闭、下层、上层和在曲面上。

自变量

“自变量”报表包含以下控件：

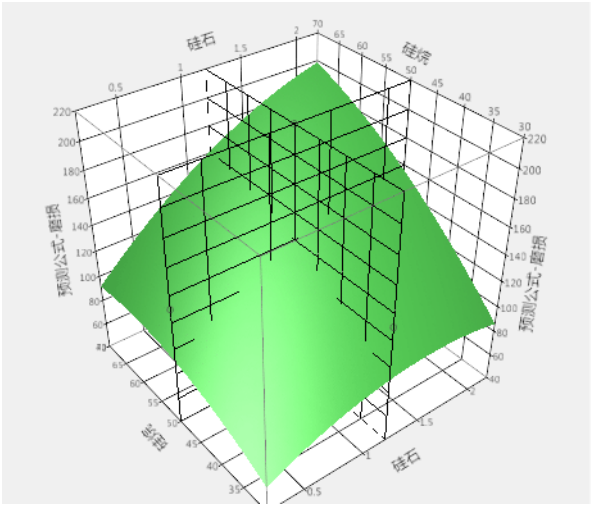
X、Y 这些单选按钮支持您选择在 x 轴和 y 轴上显示哪些自变量。该功能在有两个以上的自变量时很有用。

值 滑块和文本框设置每个变量的当前值。这些设置对于轴上未显示的变量至关重要。图中显示的是文本框中显示的值的三维曲面切片。移动滑块可查看不同的切片。

锁定 Z 尺度 将 Z 轴锁定为其当前值。这在移动轴上未显示的滑块时非常有用。

网格 选中某个复选框可激活与选定的变量轴垂直的网格。这些滑块支持您调整每个网格的位置。可通过调整轴设置控制每个网格的分辨率。图 5.9 显示已激活 X 和 Y 网格的曲面。

图 5.9 已激活 X 和 Y 网格



因变量

可用的因变量控件取决于您在“外观控件”中选定了“面，点”还是“等值面”。

用于面和点的控件

图 5.6 中显示了选定 “面，点”时默认显示的因变量控件。

公式 支持您选择在图中显示为曲面的一个或多个公式。

点响应列 支持您选择要标绘为点的值所在的列。

样式 （仅当您选择 “点响应列”后才显示的菜单。）“样式”菜单支持您选择如何显示点。该菜单包含以下选项：

点 显示各个点，这些点根据数据表中行的颜色和标记设置而更改。

针 可从 *x*-*y* 平面到点绘制线条；或者，若还标绘了曲面，则将曲面连接至点。

网线 将点连接成三角形网线。

曲面 在点之上叠加一个平滑的反射面。

关闭 根本不显示点。

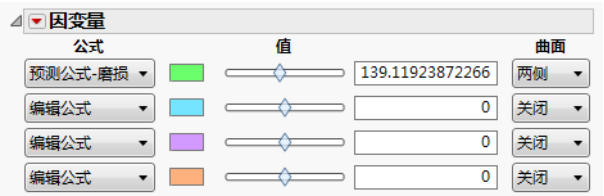
曲面 支持您显示或隐藏曲面的顶部或底部。若选定**仅上层**或**仅下层**，曲面相反的一面将变暗。

网格，网格值 提供可激活因变量的网格的滑块和复选框。您可以使用滑块调整在哪个值处绘制网格，还可以在滑块之上的 “网格值”文本框中输入值。

等值面控件

“等值面”的多数控件都与“面，点”的控件相同。图 5.10 显示默认控件，其演示的控件稍有不同。

图 5.10 用于等值面的因变量控件



值 可激活因变量的等值面的滑块和文本框。您可以使用滑块调整在哪个值处绘制等值面，还可以在滑块旁边的文本框中输入值。

“曲面图”平台选项

“曲面图”红色小三角菜单包含以下选项：

控制面板 显示或隐藏“控制面板”，其中包括针对“外观”、“自变量”和“因变量”的控件。

独立调整响应轴尺度 响应轴使用单独的尺度。请参见第 96 页的 “启动 “曲面图”平台”。

适应窗口大小 确定是否在您调整报表窗口大小时调整图的大小。

自动（默认） 基于图的内容统一尺度。例如，带 “依据”变量或在 “拟合模型”平台中用作 “曲面刻画器”的图不会拉伸以适应调整大小后的窗口；图超出查看区域的大小。

开 始终在窗口内拟合图。

关闭 防止图调整大小。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

弹出式菜单选项

您可以右击图区域以显示以下弹出式菜单选项：

面属性 （仅当您右击面时才显示的选项。）显示“面属性”窗口。仅当在“外观控件”中选定“面，点”时才可用。请参见第 104 页的“面或曲面属性”。

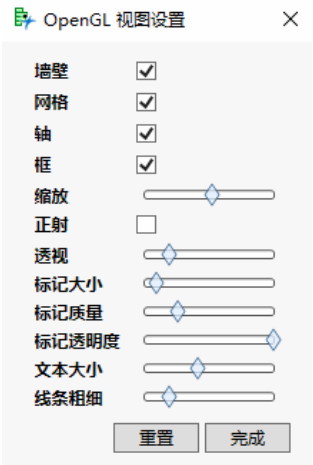
曲面属性 （仅当您点击曲面时才显示的选项。）显示“曲面属性”窗口。仅当在“外观控件”中选定“等值面”时才可用。请参见第 104 页的“面或曲面属性”。

显示图例 显示或隐藏图例。

重置 将图重置为原始视点。墙壁和背景颜色的更改不受影响。

设置 打开可用于更改许多图设置的窗口。有关 OpenGL 场景命令的详细信息，请参见《Scripting Guide》。

图 5.11 OpenGL 视图设置



隐藏光源边框 显示或隐藏光源控件。

墙壁颜色 支持您更改图的墙壁颜色。

背景颜色 支持您更改图的背景颜色。

行 支持您更改行的颜色或标记，还可以排除、隐藏和添加点标签。

使用硬件加速 支持更快地呈现显示。例如，若在旋转图时重新绘制的速度较慢，该选项有助于加快重新绘制的速度。

显示轨迹球 提供使用轨迹球的选项。轨迹球是环绕图四周绘制的球形，可帮助直观演示旋转方向。

因变量选项

“因变量”红色小三角菜单提供以下选项：

公式 在“因变量”控件中显示或隐藏“公式”列。

曲面 在“因变量”控件中显示或隐藏“曲面”列。

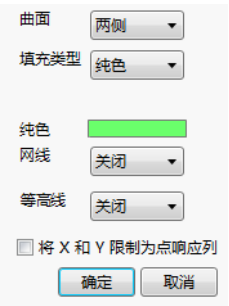
点 在“因变量”控件中显示或隐藏“点响应列”列。

响应网格 显示或隐藏“网格”控件。

面或曲面属性

若选定了“外观控件”中的“面，点”选项，则可以在图中右击面，然后选择**面属性**来显示一个窗口，该窗口支持您更改面属性。若标绘的是“等值面”，请右击曲面并选择**曲面属性**以显示相似的窗口。“面属性”和“曲面属性”窗口支持您修改曲面图的外观，包括曲面颜色、不透明性和等高线。您还可以显示或隐藏曲面图上出现的网线。

图 5.12 “面属性”窗口



曲面 支持您显示或隐藏曲面的顶部或底部。若选定仅上层或仅下层，曲面相反的一面将变暗。

填充类型 支持您使用纯色或连续 / 离散渐变为曲面着色。

纯色 （仅当将“纯色”选作“填充类型”时才可用。）支持您为曲面选择一种颜色。

曲面填充 （仅当将渐变选作“填充类型”时才可用。）指定要将填充类型应用到的因变量的曲面。将“连续渐变”选作“填充类型”时，您还可以为“曲面填充”选择“定制”选项。

定制 （仅当将“连续渐变”选作“填充类型”并且将“定制”选作“曲面填充”时才可用。）支持您指定定义响应值的方程。使用 JSL 指定该方程，并且可以使用列作用域引用数据表中的列。

曲面颜色主题 （仅当将渐变选作“填充类型”时才可用。）支持您更改曲面的颜色主题或定义定制颜色主题。有关颜色主题的详细信息，请参见《使用 JMP》。

渐变水平 （仅当将 “离散渐变”选作 “填充类型”时才可用。）支持您指定在曲面图上使用的不同颜色的数目。

曲面颜色范围 （仅当将渐变选作“填充类型”时才可用。）支持您选择颜色渐变的端点。若选择“数据”，则由为“曲面”选定的列中数据的范围来确定颜色渐变的端点。若选择“轴”，则由因变量轴的范围确定颜色渐变的端点。

警告：若为 “曲面颜色范围”选择 “轴”并更改曲面图中轴的端点，颜色渐变的截止值也随之更改。

光源 支持您更改曲面图的光源。在 “无”、“低反射”和 “正常”之间进行选择。

网线 支持您启用或禁用 *X* 和 / 或 *Y* 方向的曲面网线。

网线颜色 （仅当为“网线”选定 “关闭”之外的值时才可用。）支持您选择曲面网线的颜色。

等高线 支持您启用上层、下层或在曲面上的等高线网格选项，或者关闭等高线网格。若启用，则显示 “等高线颜色”选项，支持您更改颜色。

等高线颜色 （仅当为 “等高线”选定 “关闭”之外的值时才可用。）支持您选择等高线网格的颜色。

将 X 和 Y 限制为点响应列 支持您将图范围限定为 “点响应列”（若已激活一个）中的数据范围。若选中该选项，则基本上可防止图外推到 “点响应列”中的数据范围之外。

注意：与该选项等效的 JSL 命令为 `Clip Sheet( Boolean )`。您可以通过追加响应列的编号将该消息发送到特定的响应列。例如，`Clip Sheet2( 1 )` 可将图范围限定在第二个响应列的数据范围之内。相关示例，请参见 JMP “帮助”菜单中的 “脚本索引”。

键盘快捷方式

可以使用以下键盘快捷方式来操控曲面图。

表 5.1 曲面图键盘快捷方式

按键	功能
左、右、上、下箭头	旋转
Home、End	对角旋转
Enter（回车键）	切换轨迹球外观
Delete	逆时针旋转
Ctrl	将旋转速度提高十倍
Shift	允许连续旋转

“曲面图”平台的更多示例

- “为单个数学函数构造曲面图”
- “标绘等值面”

为单个数学函数构造曲面图

您可以使用 “曲面图”平台生成不带任何数据点的数学函数的图形。

1. 选择**图形** > **曲面图**。

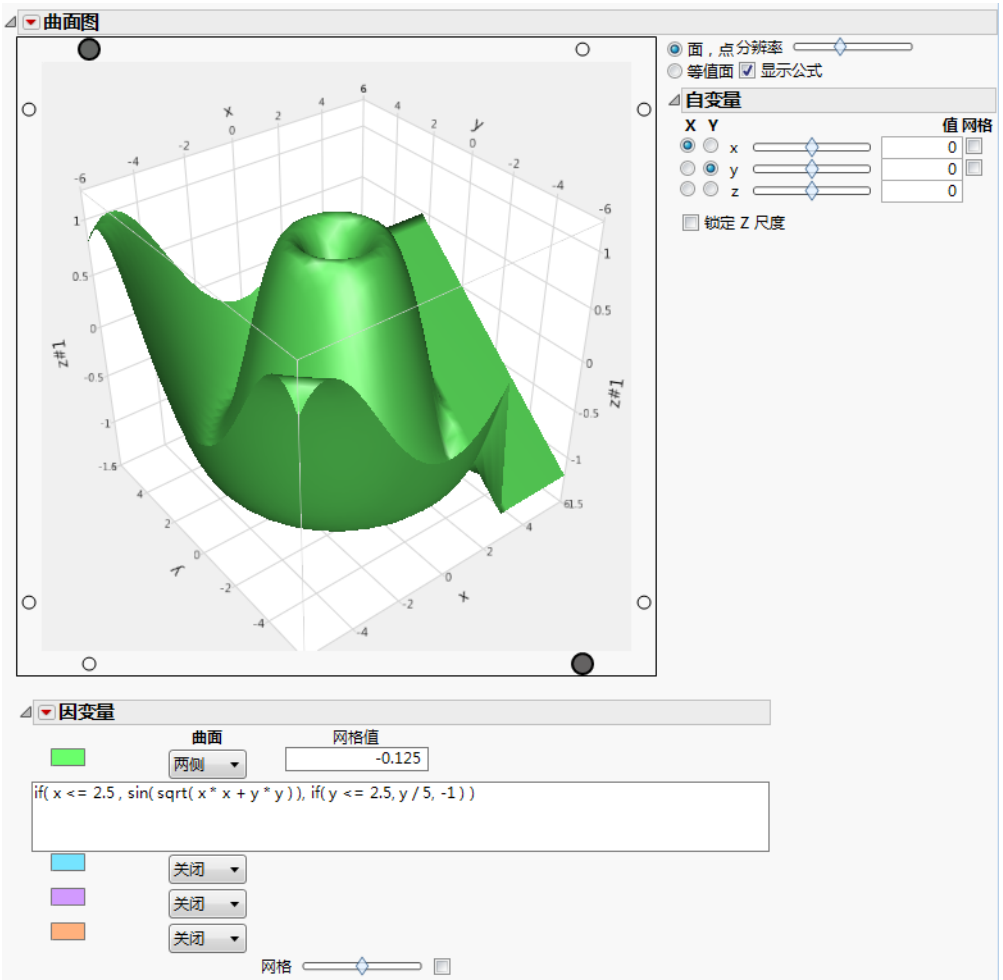
   注意：无论数据表是否打开都可以执行该操作。

2. 点击**确定**。

   若在 “曲面图”启动窗口中点击 “确定”而未选择任何列，则会显示默认的曲面图报表。若未打开任何数据表，则该启动窗口不显示，该步可以跳过。

3. 在 “外观控件”中，选中**显示公式**复选框以显示公式区域。

图 5.13 带公式的默认曲面图



默认函数显示在框中。要对您自己的函数绘图，请在该框中输入函数。

标绘等值面

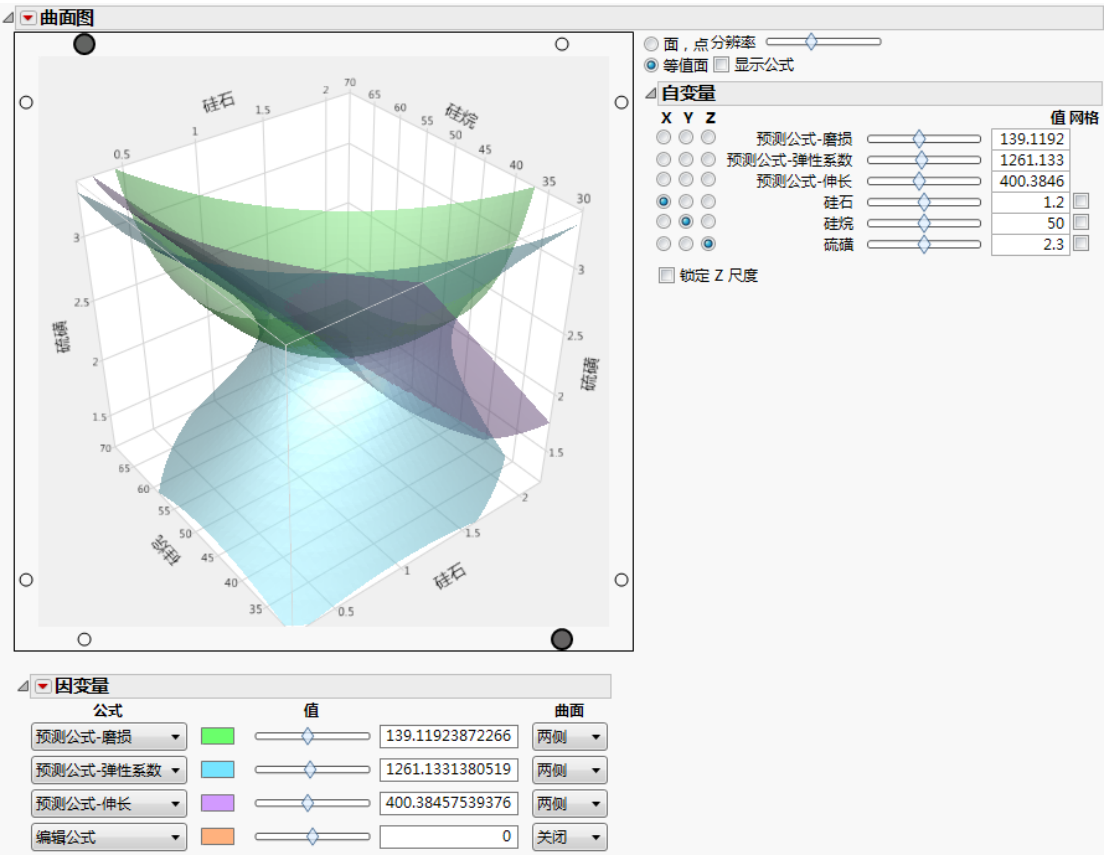
等值面是等价于二维等高线图的三维图。等值面要求公式具有三个自变量。“分辨率”滑块确定计算公式所用各点的 $n \times n \times n$ 立方体。“因变量”部分中的“值”滑块用于选择等值面（即等高线水平）值。

本例使用 Tiretread.jmp 数据表。“4 种响应的响应曲面设计”脚本生成因变量**磨损**、**弹性系数**、**伸长**和**硬度**的响应曲面模型。这四个响应的预测公式位于数据表中。

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**图形** > **曲面图**。

3. 选择**预测公式 - 磨损**、**预测公式 - 弹性系数**和**预测公式 - 伸长**，然后点击**列**。
4. 点击**确定**。
5. 选择外观控件中的 “等值面”单选按钮。
6. 在 “因变量”报表中，从全部三个变量的 “曲面”菜单中选择**两侧**。

图 5.14 三个变量的等值面



对于轮胎胎面数据，可以将磨损设置为固定的最小设置，将延伸率设置为固定的最大设置。使用**弹性系数**滑块查看哪些**弹性系数**值位于其他两个曲面所设置的限值之内。

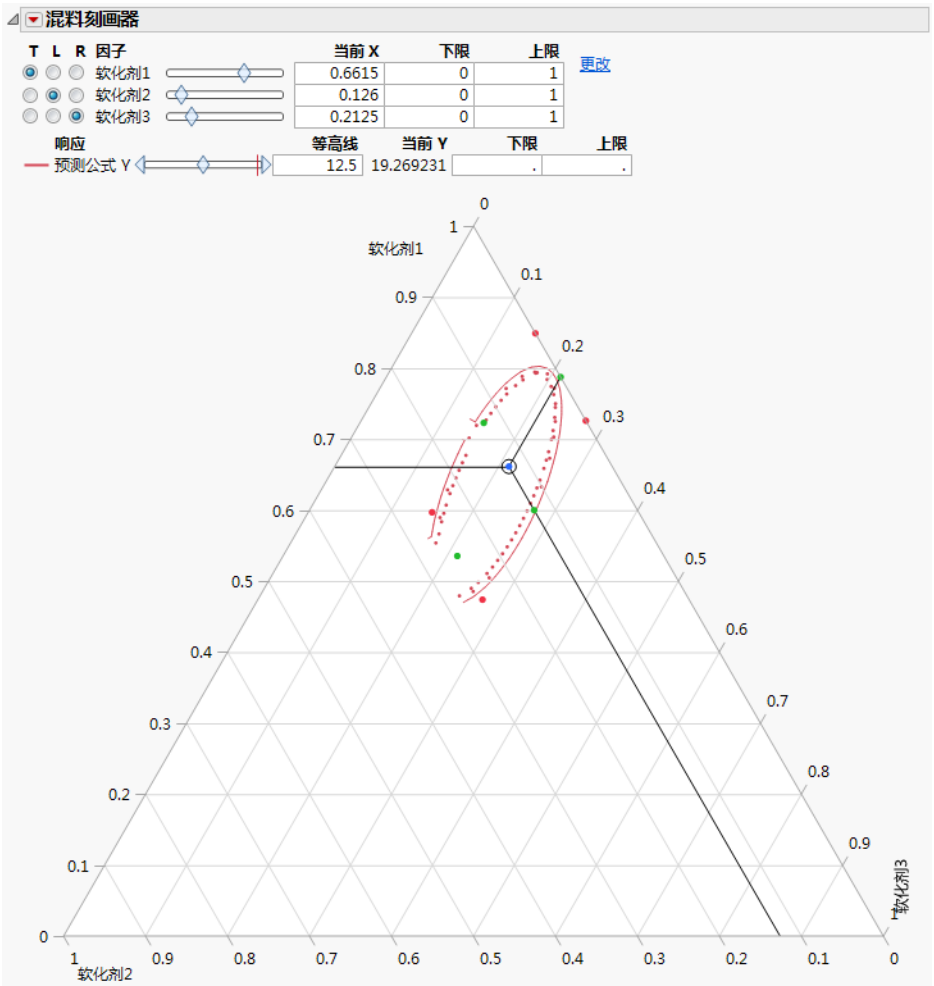
第 6 章

混料刻画器

使用三元图探索因子效应

“混料刻画器”显示混料模型的三元图上的响应等高线，其中，模型中的三个或更多因子构成混料组成部分（成分）。“混料刻画器”适用于直观演示和优化混料实验所生成的响应曲面。

图 6.1 “混料刻画器”示例



目录

“混料刻画器”概述 111
三元图概述 111
包含三个以上成分的三元图 111
混料刻画器的示例 113
启动“混料刻画器”平台 114
“混料刻画器”报表 115
因子设置和控件 115
响应设置和控件 116
“混料刻画器”平台选项 116
混料刻画器的定制 117
线性约束 117
“混料刻画器”平台的其他示例 118
具有过程变量的混料变量 118
具有五个混料变量的多个响应 122

“混料刻画器”概述

“混料刻画器”显示混料模型的三元图上的响应等高线，其中，模型中的三个或更多因子构成混料组成部分（成分）。“混料刻画器”适用于直观演示和优化混料实验所生成的响应曲面。

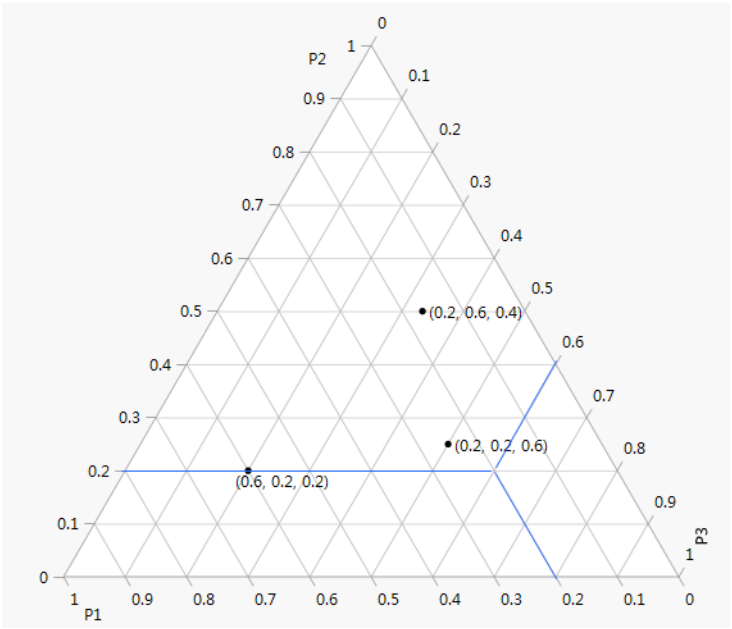
“混料刻画器”的很多功能与第 84 页的““等高线刻画器”平台选项”中所述“等高线刻画器”的功能相似。“混料刻画器”的独特之处在于使用三元图。

三元图概述

三元图是总和为常数的三个混料成分的二维表示。该图为一个等边三角形，每个成分对应一个边。无约束时，三角形的每个顶点对应于纯混合：一个成分为 1 (100%)，所有其他成分为 0。成分被约束时，用三元图的比例表示可行的混料。使用着色排除图的不可行部分。

图 6.2 显示三个成分 P1、P2 和 P3。这三个成分无约束，每个成分所占比例可为从 0 到 1 (100%) 的范围。使用其坐标 (P1, P2, P3) 标记三个点。一个点 (0.2, 0.2, 0.6) 包含沿着每个混料成分的轴网格线的蓝线，引导您读取这些轴。

图 6.2 三元图示例



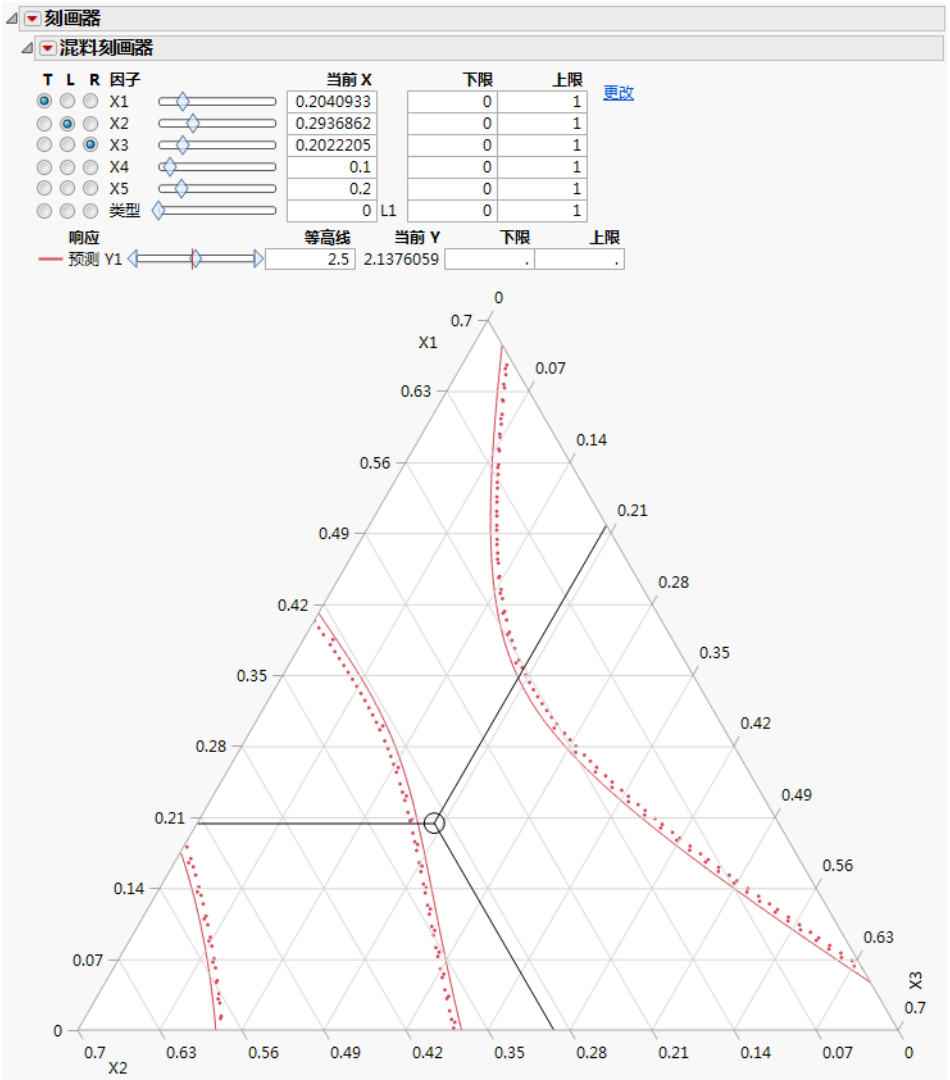
包含三个以上成分的三元图

三元图一次仅显示三个成分。对于具有三个以上成分的模型，轴上（显示的）三个成分之和为 1 减去轴下（不显示的）成分之和。按以下方式确定图轴的尺度：某个成分可以达到的最大值为 1 减去轴下成分之和。

图 6.3 显示具有 5 个因子的实验的混料刻画器三元图。它使用的是 Five Factor Mixture.jmp 数据表，**预测 Y1** 列用作公式。轴上因子为 x1、x2 和 x3，因子 x4 和 x5 为轴下因子。对任何混料成分都没有约束。请注意，所有“下限”值都设置为 0，所有“上限”值都设置为 1。x4 的值为 0.1，x5 的值为 0.2，总计为 0.3。x1、x2 和 x3 的和等于 1 – 0.3 = 0.7。请注意，图轴的最大值现在为 0.7，而非 1。

若您更改 x4 或 x5 的值，则 x1、x2 和 x3 的值将会更改，但是其相对比例保持不变，以满足因子值之和为 1 的约束条件。

图 6.3 随着轴下因子调整轴尺度的三元图



混料刻画器的示例

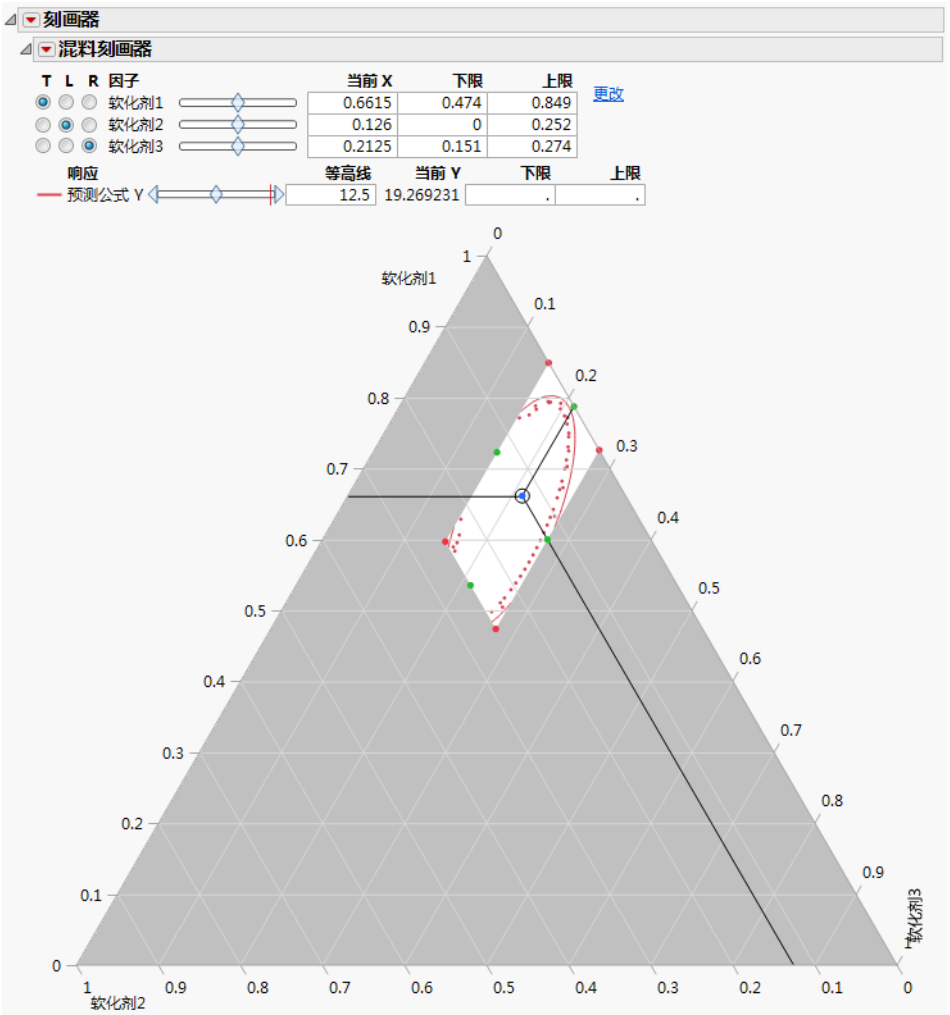
该示例使用 Plasticizer.jmp 样本数据表。有三个混料成分（p1、p2 和 p3）和一个响应 (Y)。有针对成分水平的约束。该数据表已包含响应的一个已保存预测公式列。

1. 选择**帮助** > **样本数据库**，然后打开 Plasticizer.jmp。
2. 选择**图形** > **混料刻画器**。
3. 选择**预测公式** Y 并点击 **Y，预测公式**。
4. 从 “Y，预测公式”角色选择 **Y**，然后点击**删除**。

注意：因为 “列”列表中的 Y 被设置了 Y 列角色。该角色会自动在启动窗口中分配列。在这种情况下，启动混料刻画器时将 Y 列分配给了 Y 角色。但是，Y 列不包含公式，而混料刻画器 Y 列需要公式列。

5. 点击**确定**。

图 6.4 混料刻画器的示例



使用因子滑块探索所做的更改如何影响响应。未着色的区域是该数据集的可行域。

启动“混料刻画器”平台

可以通过以下方式访问 “混料刻画器”：

- 可以直接从“图形”菜单访问“混料刻画器”。使用该方式访问“混料刻画器”时，显示“混料刻画器”启动窗口。请参见“刻画器介绍”一章中第 28 页的““刻画器”启动窗口”。

- 在很多建模平台中，“混料刻画器”可以作为红色小三角菜单选项来访问。有关在不同平台中是否提供“混料刻画器”的详细信息，请参见“刻画器介绍”一章中第 26 页的“JMP 刻画器的提供位置”。
- 可以从“模型比较”平台访问“混料刻画器”。从“模型比较”红色小三角菜单中选择“刻画器”。然后，从“刻画器”红色小三角菜单中选择“混料刻画器”。
- JMP PRO 可以从“公式存储库”平台访问“混料刻画器”。从“公式存储库”红色小三角菜单中选择“刻画器”，然后选择要刻画的模型。然后，从“刻画器”红色小三角菜单中选择“混料刻画器”。

“混料刻画器”报表

初始“混料刻画器”报表显示一个混料刻画器图、一些因子设置和控件、一些响应设置和控件。

- “因子设置和控件”
- “响应设置和控件”

因子设置和控件

图 6.5 因子设置和控件

T，L，R 一些单选按钮，用于在混料刻画器图中控制对轴的因子分配。T = 上，L = 左，R = 右。

因子 因子的列表。

当前 X 当前因子设置。在框中点击可更改因子的值。按比例调整其他因子的值以保持混料之和不变。也可以使用滑块控件更改因子设置的值。

下限 每个因子的下限。在框中点击可更改该值。

上限 每个因子的上限。在框中点击可更改该值。

注意：使用因子限时，刻画器将可行的区域显示为未着色。

更改 打开“因子设置”窗口。请参见第 116 页的“‘混料刻画器’平台选项”。

响应设置和控件

图 6.6 响应设置和控件

响应 一个或多个响应的列表。

等高线 混料刻画器上等高线的当前值。在框中点击可更改该值。也可以使用滑块控件更改等高线的值。

当前 Y 基于当前 X 设置的预测响应。该值位于刻画器图上十字准线的中心。该值随着因子设置的更改更新。

下限 支持您设置响应的下限。

上限 支持您设置响应的上限。

注意：使用响应限时，刻画器将可行的区域显示为未着色。

“混料刻画器”平台选项

指定因子值 打开用于指定因子值的对话框。输入的值之和应为 1。若需要，JMP 将指定的值调整为可行设置。还可以从报表窗口中的**更改**按钮使用该选项。

显示点 显示或隐藏图上的各个点。

显示当前值 显示或隐藏图上的三向十字准线。十字准线的交点代表当前因子值。图上方的**当前 X** 值给出了十字准线的准确坐标。

显示约束 显示或隐藏对因子施加任何约束所导致的着色区域。可以在图上方的**下限**列和**上限**列或因子的 “混料列属性”中输入这些约束。

上方带点 显示或隐藏与每条等高线对应的虚线。虚线显示增加响应值的方向，让您有方向感。

等高线网格 在图上以您指定的间隔绘制等高线。

删除等高线网格 删除图上的等高线网格 （若存在该网格）。

因子设置 命令子菜单，该子菜单支持您将 “混料刻画器”设置保存并传输至 JMP 的其他部分。有关该子菜单的详细信息，请参阅第 44 页的 “因子设置”中关于刻画器的讨论。

混料刻画器的定制

通过右击图并选择**定制**可定制混料刻画器。

等高线 更改线条颜色、填充颜色和线条透明程度。若图上有多条等高线，每条线将分别显示在选项列表中。

成分约束 更改成分约束显示属性。

线性约束 更改线性约束显示属性。

网格线 使用“轴设置”窗口修改网格线。

参考线 使用“轴设置”窗口修改参考线。

双击图上某个轴可查看“轴设置”窗口。在此可详细指定轴属性，如刻度标、网格线和参考线。

十字准线 更改图上十字准线的显示属性。

标记 更改图标记的显示属性。

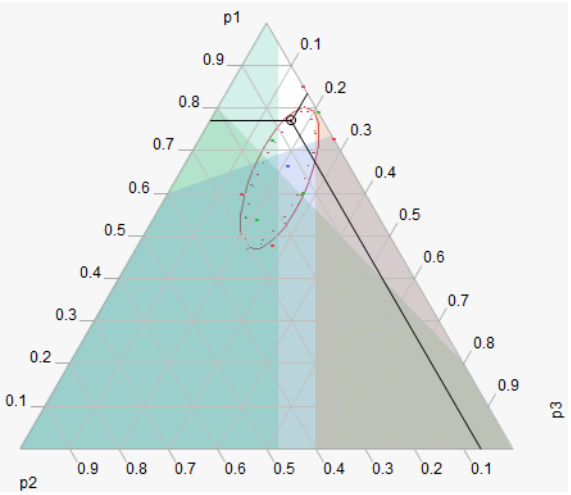
线性约束

混料刻画器可以显示线性约束。为此，数据表必须包含“约束”表脚本。有关创建表脚本的详细信息，请参见“刻画器介绍”一章中第 34 页的“线性约束”。

使用约束时，将对刻画器中的非可行域着色。图 6.7 显示了因四个约束而存在着色区域的“混料刻画器”示例。未着色部分是生成的可行域。约束如下所示：

- $4*p2 + p3 \leq 0.8$
- $p2 + 1.5*p3 \leq 0.4$
- $p1 + 2*p2 \geq 0.8$
- $p1 + 2*p2 \leq 0.95$

图 6.7 因线性约束形成的着色区域



“混料刻画器”平台的其他示例

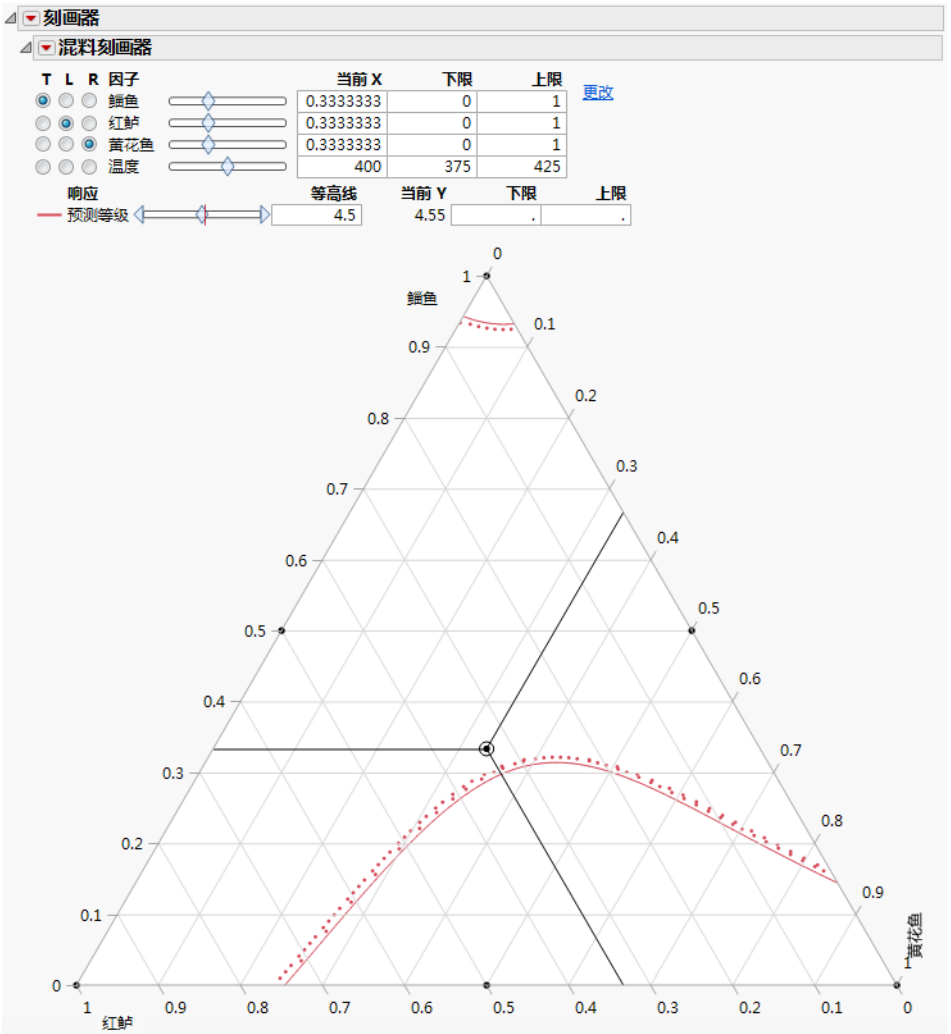
- “具有过程变量的混料变量”
- “具有五个混料变量的多个响应”

具有过程变量的混料变量

该示例使用 Fish Patty.jmp 样本数据表。该数据改编自 Cornell (1990)，来自改善鱼肉馅饼的配料的实验。**鳊鱼**、**红鲈**和**黄花鱼**列是混料成分。每个列表示鱼肉馅饼中某鱼类型的比例。**温度**列是过程变量。它是用于烤馅饼的炉温。**等级**列是响应，用于衡量配料被认可的程度，值越大表示越好。用响应曲面模型来拟合数据，将预测公式存储在**预测等级**列中。

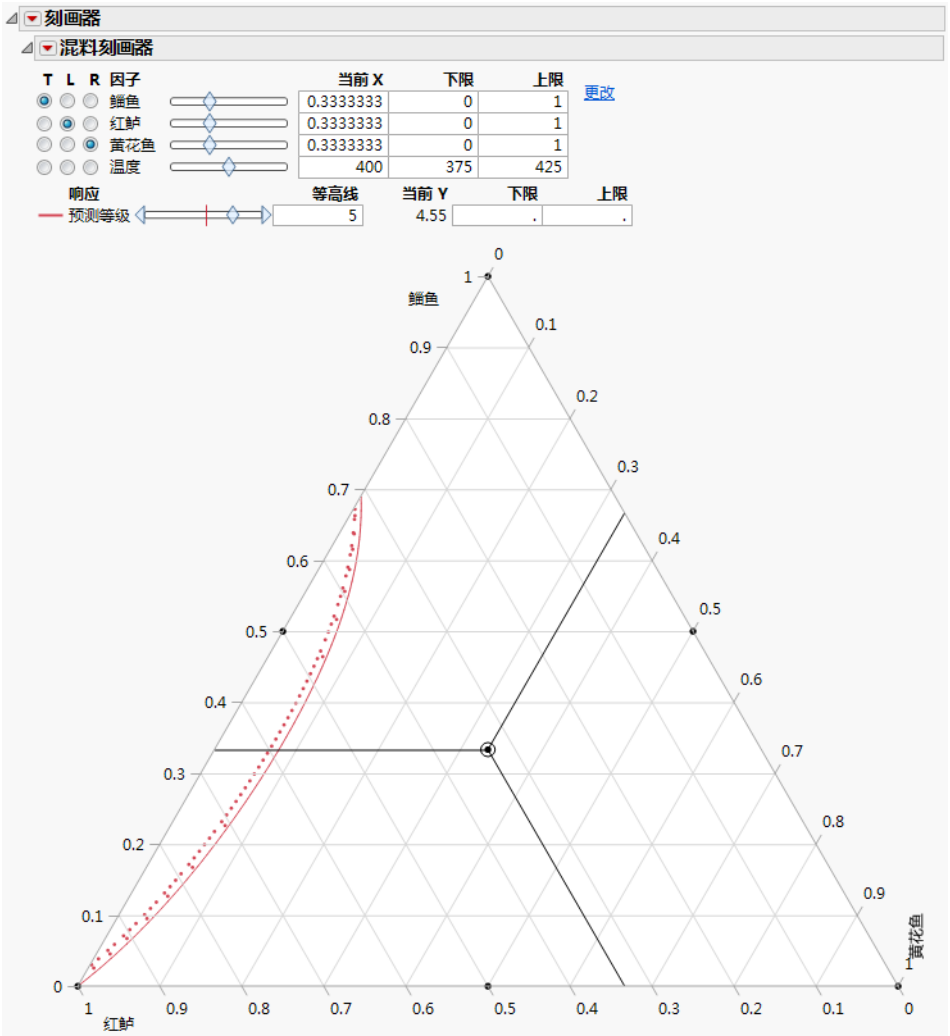
1. 选择**帮助** > **样本数据库**，然后打开 Fish Patty.jmp。
2. 选择**图形** > **混料刻画器**。
3. 选择**预测等级**并点击 **Y，预测公式**。
4. 点击**确定**。

图 6.8 “混料刻画器”的初始输出



5. 制造商希望等级能够达到至少 5 级。使用**预测等级**的滑块控件将等高线移至接近 5 的位置。或者，您还可以在**等高线**编辑框中输入 5，以将等高线设置为 5。

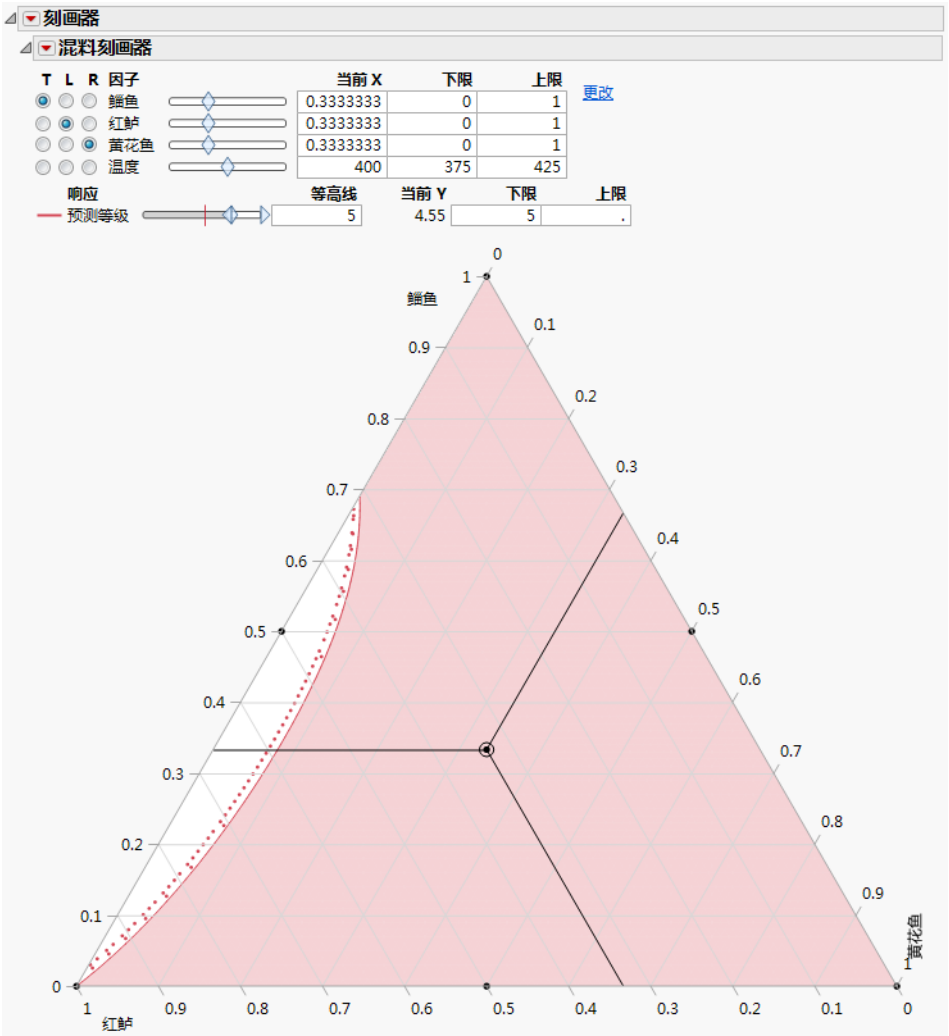
图 6.9 显示预测等级为 5 的等高线



沿着等高线显示的“上方带点”指示增加**预测等级**的方向。

6. 在“预测等级”的**下限**编辑框中输入 5。生成的着色区域代表生成的等级不到 5 级的因子组合。要生产出至少 5 级的馅饼，制造商可以将因子值设置在可行域（未着色区域）中的任意位置。请注意，该区域有较小比例的**黄花鱼** (<10%)、中低比例的**鲳鱼** (<70%) 和中高比例的**红鲈** (>30%)。该区域对应 400 度的烘焙温度。

图 6.10 显示预测等级至少为 5 的等高线着色



7. 移动**温度**的滑动控件以观测可行域对于不同温度设置是如何变化的。

进一步的分析可能包括：

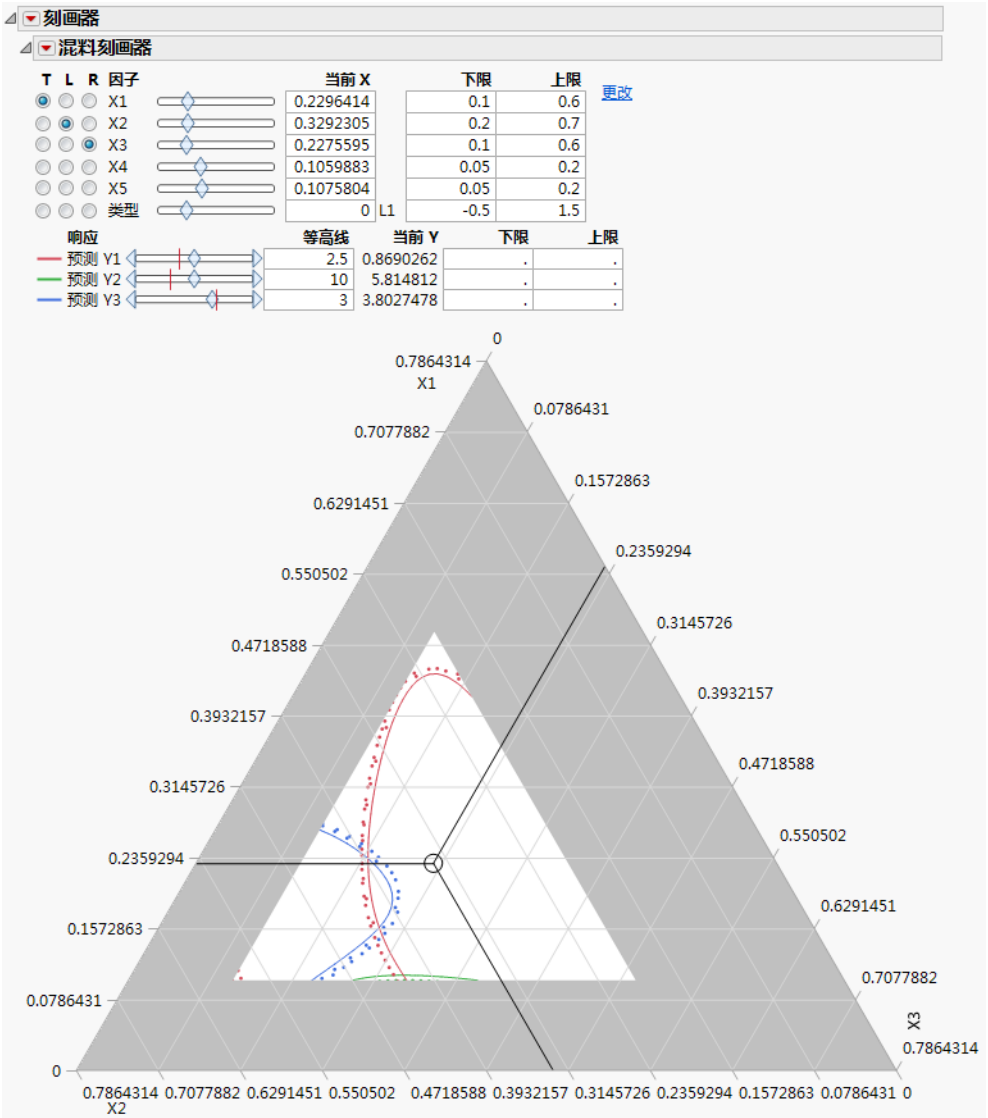
- 同时优化全部四个因子的响应。请参见第 129 页的“定制刻画器”一章或“刻画器”一章中第 47 页的“意愿刻画和优化”。
- 模拟响应，将其作为因子和模型噪声的随机变异的函数。请参见第 137 页的“模拟器”一章。

具有五个混料变量的多个响应

本例使用 Five Factor Mixture.jmp 样本数据表。有五个成分 （x1 到 x5） 、一个分类过程因子（**类型**）和三个响应（Y1、Y2 和 Y3）。为每个响应拟合了响应曲面模型，并且预测方程保存在**预测 Y1**、**预测 Y2** 和**预测 Y3** 中。

1. 选择**帮助** > **样本数据库**，然后打开 Five Factor Mixture.jmp。
2. 选择**图形** > **混料刻画器**。
3. 选择**预测 Y1**、**预测 Y2** 和**预测 Y3**，然后点击 **Y，预测**。
4. 点击**确定**。
5. 在**预测 Y3** 的 “等高线”编辑框中输入 3，以便该等高线在图上可见。

图 6.11 将 “预测 Y3”等高线设置为 3 的混料刻画器



关于输出，需要注意以下几点。

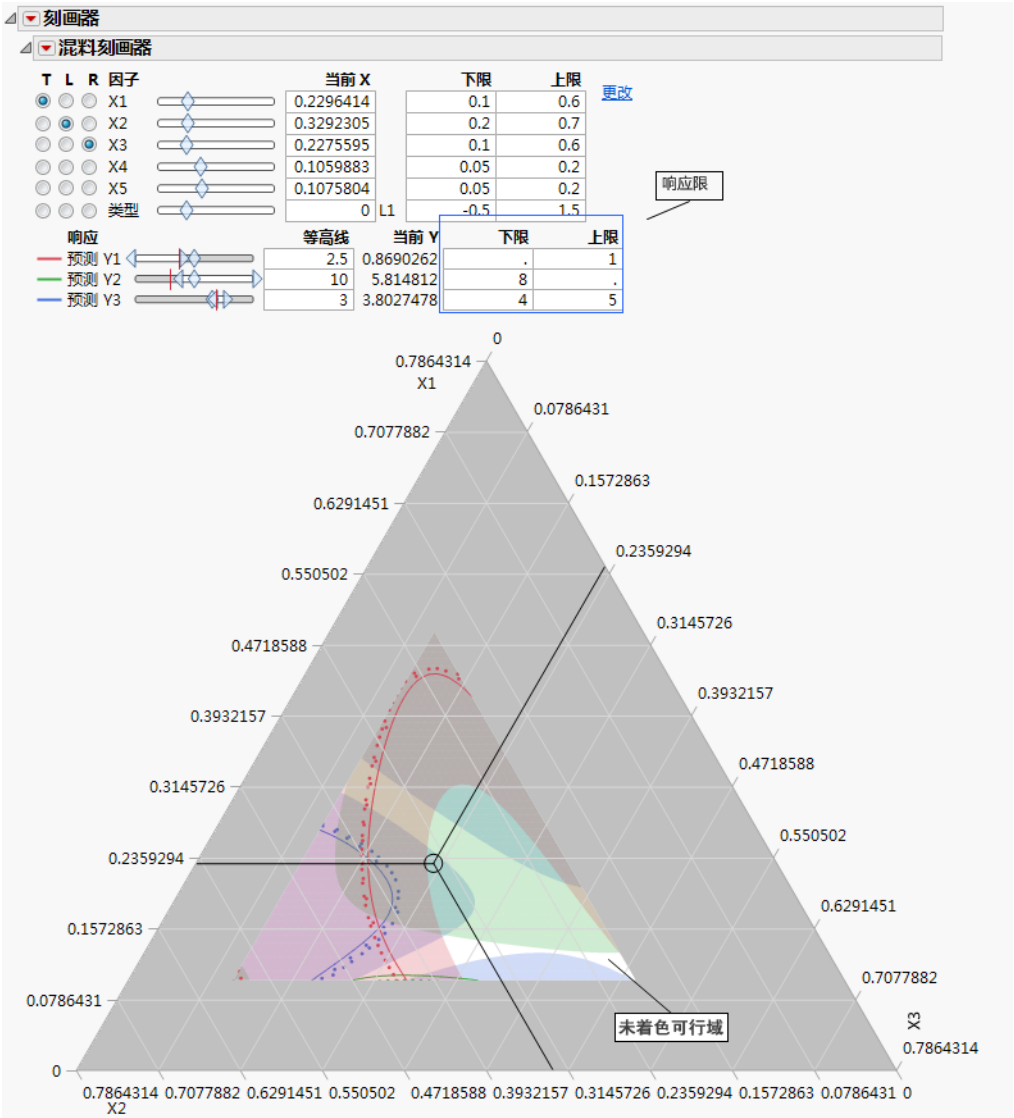
- 轴上因子为 X1、X2 和 X3，如轴标签和因子设置中的单选按钮所指示。所有因子有下限和上限，这两个限值之前是作为“列属性”输入的。有关输入列属性的详细信息，请参见《使用 JMP》。或者，也可以直接在“下限”框和“上限”框中分别输入下限和上限。
- 未着色的白色区域是可行域。这由因子限决定。
- 选定了轴上因子（x1、x2 和 x3）的单选按钮。

- 分类因子**类型**有对应的单选按钮，但无法将其分配给该图。**类型**的当前值为 L1，该值紧挨着“当前 X”框右侧列出。**类型**的“当前 X”框使用 0 来表示 L1。
- 全部三个预测方程在图上都有对应的等高线并按颜色来区分。

假定制造商希望将 Y1 控制在 1 以内，让 Y2 大于 8，将 Y3 控制在 4 和 5 之间，目标为 4.5。“混料刻画器”可以帮助您调查响应曲面并找出最佳因子设置。

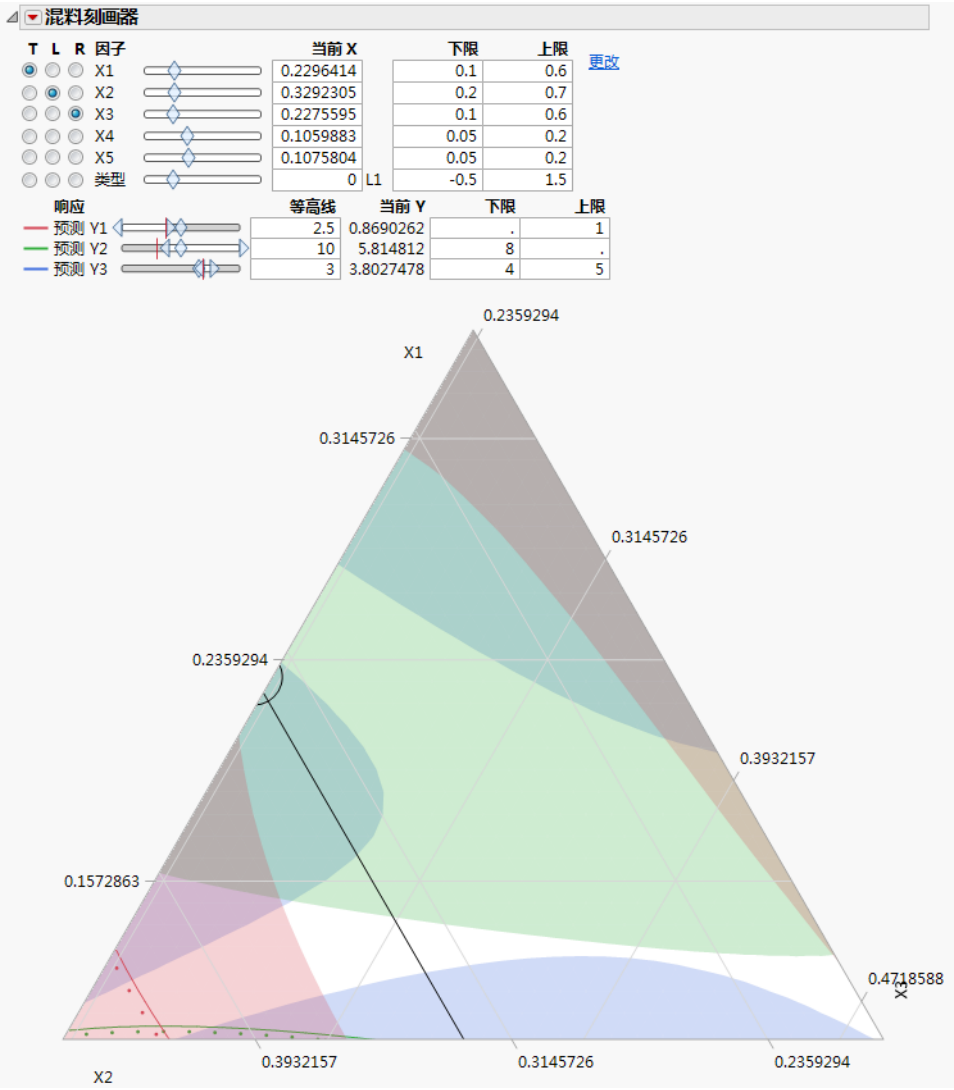
6. 在“预测 Y1 上限”编辑框中输入 1。在“预测 Y2 下限”编辑框中输入 8。在“预测 Y3 下限”编辑框中输入 4，在 “预测 Y3 上限”编辑框中输入 5。
7. 可行域仍保持白色（未着色）。使用“响应”滑块控件在可行域中放置等高线。或者，将十字准线移到可行域。

图 6.12 设置响应限后的可行域



8. 选择放大镜工具 放大可行域。

图 6.13 可行域已放大的三元图

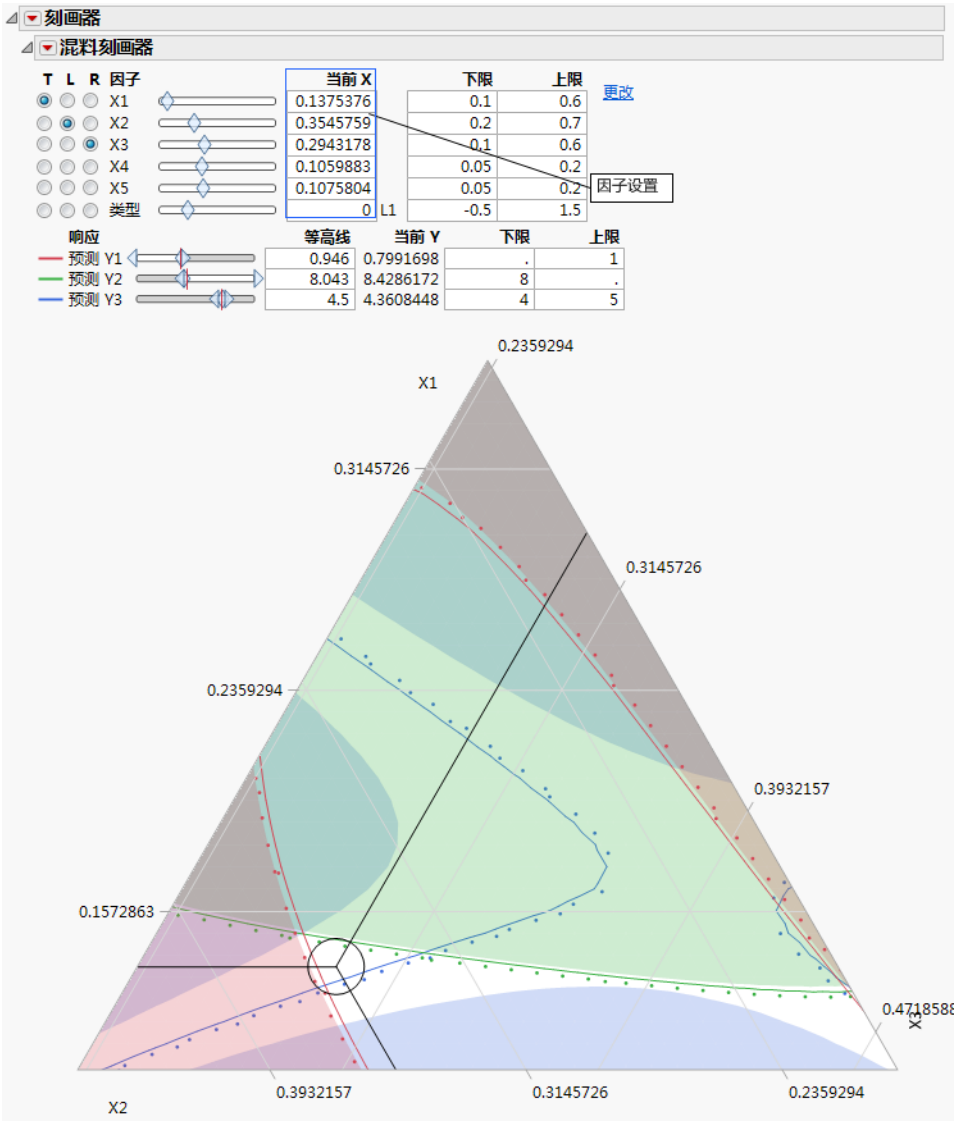


制造商希望最大化 Y1、最小化 Y2 并使 Y3 保持在 4.5。

9. 使用**预测 Y1** 的滑块控件或“等高线”编辑框将可行域中的红色等高线最大化。“上方带点”指明预测响应增加的方向。
10. 使用**预测 Y2** 的滑块控件或 “等高线”编辑框将未着色区域中的绿色等高线最小化。
11. 在**预测 Y3** 的 “等高线”编辑框中输入 4.5，令蓝色等高线保持目标值。

生成的三个等高线没有都相交于一点。Y3 为目标值时无法优化 Y1 和 Y2，因此您必须折衷一下。将三向十字准线置于等高线中央位置来探索生成这些响应值的因子水平。请注意，这些响应等高线针对 x4、x5 和**类型**的当前设置。

图 6.14 最优结果的因子设置



12. 点击“混料刻画器”红色小三角菜单并选择**因子设置** > **记住设置**来保存当前设置。这些设置追加到报表窗口的底部。

图 6.15 记住的设置

记住的设置

设置	X1	X2	X3	X4	X5	类型	预测 Y1	预测 Y2	预测 Y3
设置 1	0.1352764	0.359975	0.29118	0.1059883	0.1075804	L1	0.9244618	8.5514568	4.3705523

保存当前设置后，您现在可以更改 x4、x5 和**类型**的值以探索它们对可行域的影响。通过参考“记住的设置”报表，您可以比较**类型**每一水平的因子设置和响应值。

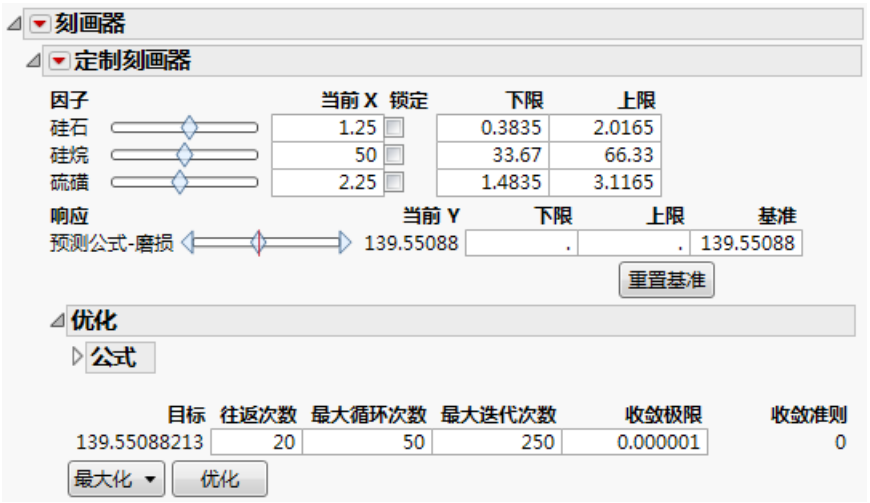
第 7 章

定制刻画器

使用数字计算器探索响应曲面

“定制刻画器”支持您优化因子设置，而不生成图形输出。这对因图形太多而无法很好地可视化的大问题很有用。

图 7.1 “定制刻画器”示例



目录

“定制刻画器”概述 131
定制刻画器的示例 131
启动“定制刻画器”平台 133
“定制刻画器”报表 134
    因子设置和控件 134
    响应设置和控件 134
    优化设置和控件 135
“定制刻画器”平台选项 135

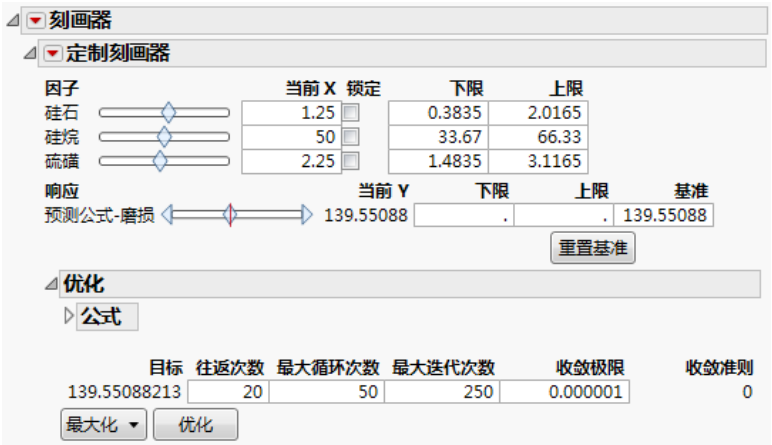
“定制刻画器”概述

“定制刻画器”支持您优化因子设置，而不生成图形输出。“定制刻画器”可用于任何规模的问题。它对于因图形太多而导致标准图形刻画器无法很好地将其可视化的大问题特别有用。

“定制刻画器”报表中的很多字段与其他刻画器中的字段相同。“基准”字段存储预测的响应的值或基准。您可以将新结果与基准值进行比较，根据当前因子设置更新该值。

通过“优化”报表，您可以指定要优化的公式和优化规格。

图 7.2 定制刻画器

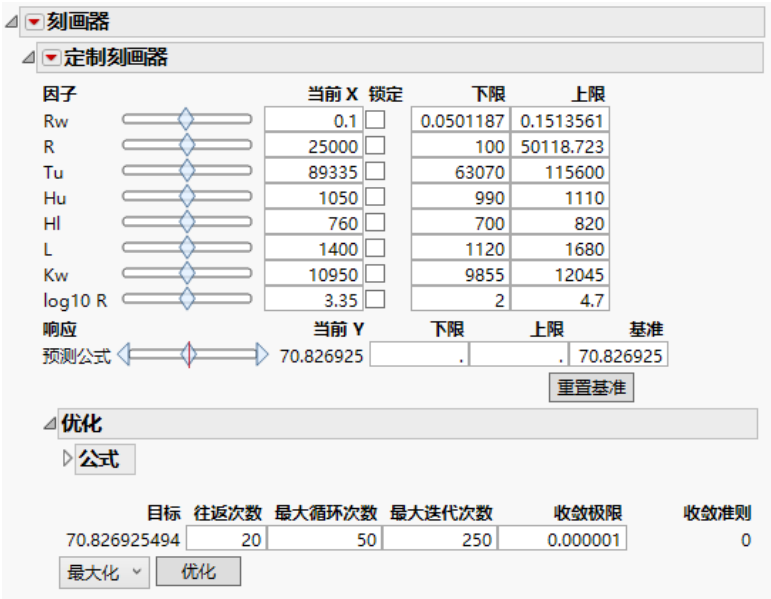


定制刻画器的示例

该示例使用用来演示通过某钻孔的水流量数据，该钻孔从地表下钻，穿过两个蓄水层。您想优化预测值。

1. 选择**帮助** > **样本数据库** > Design Experiment，然后打开 Borehole Latin Hypercube.jmp。
2. 选择**图形** > **定制刻画器**。
3. 选择**预测公式**并点击 **Y，预测公式**。
4. 点击**确定**。

图 7.3 “定制刻画器”报表



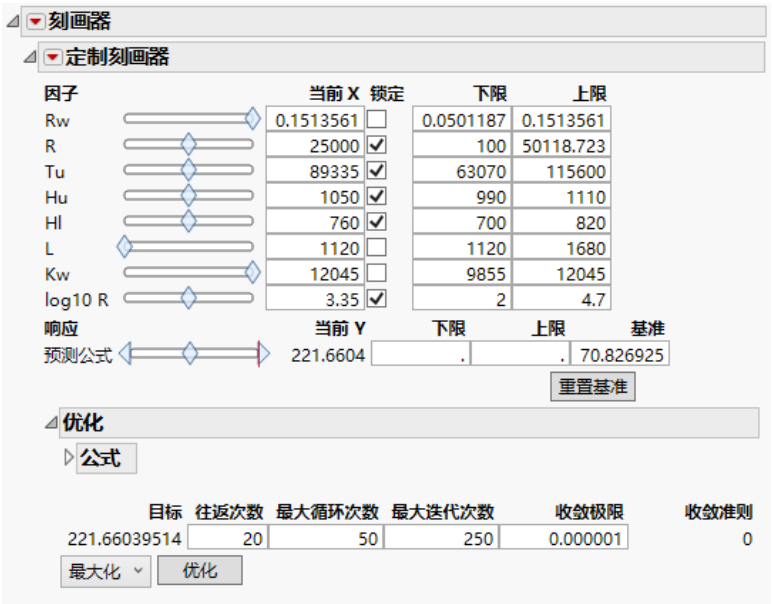
请注意 “基准”值是 70.83。“当前 Y”值也是 70.83。该值是将所有因子设置为其均值时的预测响应。

您想要查找优化预测值的因子的 X 值。不过，在该模型的因子中，只有与钻孔相关的那些因子是实际上可操控的。其他因子不受人为控制。您可以将不可控因子锁定在其均值上，并且优化可控因子。

注意：要查看因子的说明，请右击数据表中的相应列并选择**列信息**。

5. 在 “定制刻画器”报表中，选中 R、 Tu、 Hu、 Hl 和 log10 R 旁边的**锁定**复选框。
6. 在 “定制刻画器”报表中，点击**优化**。

图 7.4 优化的 “定制刻画器”报表



优化过程找到 221.66 处的最佳预测响应。为了获得 221.66 处的最优值，Rw 和 Kw 均设置为其最大值， L 设置为其最小值。最优值大于初始 “基准”值 70.83。

启动“定制刻画器”平台

可以通过以下方式访问 “定制刻画器”：

- 可以直接从“图形”菜单访问“定制刻画器”。使用该方式访问“定制刻画器”时，显示“定制刻画器”启动窗口。请参见“刻画器介绍”一章中第 28 页的““刻画器”启动窗口”。
- 在很多建模平台中，“定制刻画器”可以作为红色小三角菜单选项来访问。有关在不同平台中是否提供“定制刻画器”的详细信息，请参见“刻画器介绍”一章中第 26 页的“JMP 刻画器的提供位置”。
- 可以从“模型比较”平台访问“定制刻画器”。从“模型比较”红色小三角菜单中选择“刻画器”。然后，从“刻画器”红色小三角菜单中选择“定制刻画器”。
- JMP PRO 可以从“公式存储库”平台访问“定制刻画器”。从“公式存储库”红色小三角菜单中选择“刻画器”，然后选择要刻画的模型。然后，从“刻画器”红色小三角菜单中选择“定制刻画器”。

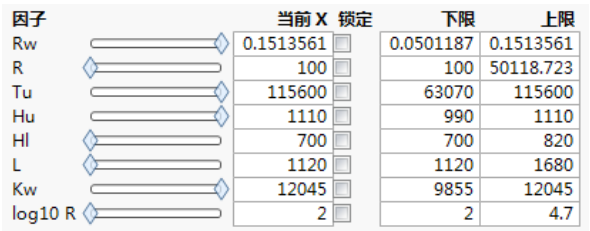
“定制刻画器”报表

初始 “定制刻画器”报表显示因子、响应以及优化的设置和控件。

- “因子设置和控件”
- “响应设置和控件”
- “优化设置和控件”

因子设置和控件

图 7.5 因子设置和控件



因子	当前 X	锁定	下限	上限
Rw	0.1513561		0.0501187	0.1513561
R	100		100	50118.723
Tu	115600		63070	115600
Hu	1110		990	1110
Hl	700		700	820
L	1120		1120	1680
Kw	12045		9855	12045
log10 R	2		2	4.7

因子 模型因子的列表。

当前 X 每个因子的当前值。在框中点击可更改因子的值。也可以使用滑块控件更改因子设置的值。

锁定 支持您锁定某个因子，以使在执行优化时固定该因子。您可以使用滑块或通过在 “当前 X”列的框中点击来更改锁定的因子。该锁定仅适用于优化。

名义型列 “锁定”列右侧的未标记列，它列出名义型因子的当前值。

> **注意：**名义型因子的 “当前 X”列显示当前名义型因子的编码值 （数值）。

下限 每个因子的下限。在框中点击可更改该值。

上限 每个因子的上限。在框中点击可更改该值。

响应设置和控件

图 7.6 响应设置和控件

响应	当前 Y	下限	上限	基准
预测公式	311.17206	.	.	70.826925

重置基准

响应 一个或多个响应的列表。

当前 Y 基于当前 X 设置的预测响应。该值随着因子设置的更改更新。

下限 支持您设置响应的下限。

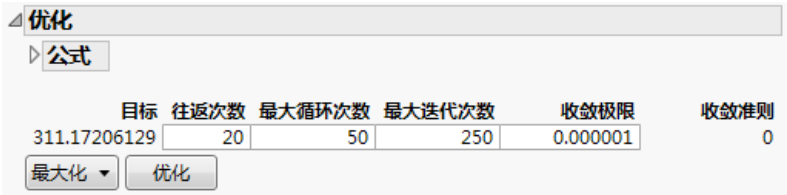
上限 支持您设置响应的上限。

基准 响应的已保存预测值。最初，将该值设置为当所有因子在其均值处的预测值。

重置基准 将基准值更新为当前预测值。

优化设置和控件

图 7.7 优化设置和控件



公式 要优化的公式。使用单个响应时，表达式为响应列名称。使用多个响应时，表达式为意愿函数之和。您可以编辑目标表达式。

目标 目标函数的当前值。使用单个响应时，目标表达式为预测响应。使用多个响应时，目标表达式为意愿函数。有关意愿函数的详细信息，请参见“刻画器”一章中第 47 页的“意愿刻画和优化”。

往返次数 优化算法中随机开始的次数。每个往返从不同的起始点重新开始执行算法。这有助于防止求得局部解。

最大循环次数 在优化算法中使用的最大循环次数。每次循环遍历一次诸多输入参数并逐一优化每个参数。

最大迭代次数 对于每个输入参数，每次循环经历的最大优化迭代次数。

收敛极限 优化算法的收敛准则的上限。若收敛准则变得小于该值，则算法停止。

收敛准则 优化算法的收敛准则的值。

最大化 支持您选择最大化或最小化目标函数。

优化 开始执行优化算法。

“定制刻画器”平台选项

因子设置 包含的选项与“预测刻画器”中的“因子设置”子菜单的选项相同。请参见“刻画器”一章中第 44 页的“因子设置”。

记录迭代次数 创建包含优化算法的迭代次数的数据表。在点击**优化**按钮后显示该数据表。

更改线性约束 支持您添加、更改或删除线性约束。这些约束在定制刻画器中使用。请参见“刻画器介绍”一章中第 34 页的 “线性约束”。

保存线性约束 将现有线性约束保存为名为“约束”的数据表脚本。请参见“刻画器介绍”一章中第 34 页的 “线性约束”。

模拟器 启动模拟器。请参见“模拟器”一章中第 137 页的 “模拟器”。

第 8 章

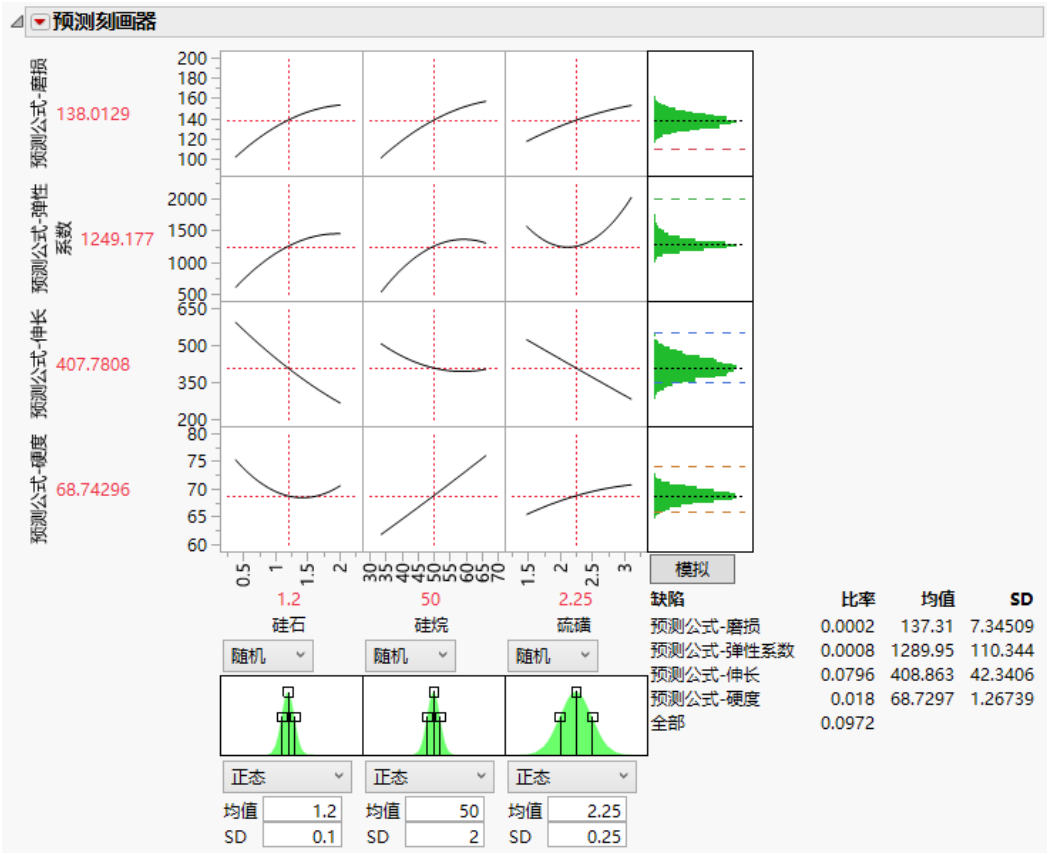
模拟器

了解变异对响应的影响

通过模拟，您可以发现作为模型输入和噪声中随机变异的函数的模型输出的分布。刻画器中的“模拟器”支持您定义随机输入、运行模拟并生成模拟值输出表。

在“预测刻画器”中，“模拟器”集成到图形布局中。因子规格显示在每个因子的刻画图下方。为每个响应都显示一个模拟直方图。

图 8.1 带 “模拟器”的 “预测刻画器”示例



目录

模拟器概述 139
模拟器的示例 139
启动“模拟器” 141
    因子模拟设置 141
    响应模拟设置 143
“模拟器”报表 144
“模拟器”报表选项 144
缺陷刻画器 145
    容差设计简介 145
    图形尺度 146
    缺陷率 146
    模拟方法和详细信息 147
    注意 147
缺陷参数刻画器 147
模拟实验 148
规格限 148
模拟器的其他示例 149
    缺陷刻画器的示例 149
    随机优化的示例 152
    模拟常规公式的示例 160
模拟器的统计详细信息 163
    正态加权分布 163

模拟器概述

通过模拟，您可以发现作为模型输入和噪声中随机变异的函数的模型输出的分布。刻画器中的“模拟器”支持您定义随机输入、运行模拟并生成模拟值输出表。

例如，您可以使用模拟器对过程的缺陷率建模，探讨缺陷率与模型因子中变异关系的稳健程度。若为响应设置了规格限，它们会显示在模拟输出中。指定规格限支持您对模拟的模型进行前瞻能力分析。有关添加规格限的信息，请参见《使用 JMP》。

您可以通过“预测刻画器”、“等高线刻画器”和“定制刻画器”访问“模拟器”。“模拟器”的外观取决于它所在的刻画器平台。

预测刻画器

在“预测刻画器”中，“模拟器”集成到图形布局中。因子规格显示在每个因子的刻画图下方。模拟直方图显示在每个响应的右侧。

“等高线刻画器”和“定制刻画器”

在“等高线刻画器”和“定制刻画器”中，“模拟器”未集成到图形布局中。没有集成的直方图，而且界面是文本的。不过，过程是相同的，而且生成的输出表与在其他刻画器中相同。

注意：“模拟器”支持混料项。该模拟将随机值分配给每个混料因子，然后按比例调整这些值以保持因子约束不变。

模拟器的示例

使用排序查看当输入的均值 （序列位置）和变异性 （定序离差）变化时响应的分布如何变化。

1. 选择**帮助 > 样本数据库**，然后打开 Tiretread.jmp。
2. 选择**图形 > 刻画器**。
3. 选择**预测公式 - 磨损**和**预测公式 - 弹性系数**，然后点击 **Y，预测公式**。
4. 点击**确定**。
5. 点击 “预测刻画器”红色小三角并选择**模拟器**。
6. 在每个因子下方选择**随机**。

   注意：默认随机分布是正态分布。

7. 将**试验次数**值更改为 1000。
8. 打开 “模拟结果输出表”分级显示项，然后打开 “排序”分级显示项。

图 8.2 “模拟器”设置

您要查看当输入因子更改位置时响应如何变化。要探索因子均值的变化，请使用序列位置。要改为探索因子变异性的变化，请使用序列散布。

9. 对于**硅石**，选择**序列位置**。保持步数设置为 5。将**下限**设置为 1，**上限**设置为 2。
10. 对于**硅烷**，选择**序列位置**。保持步数设置为 5。将**下限**设置为 40，**上限**设置为 60。
11. 对于**硫磺**，选择**序列位置**。保持步数设置为 5。将**下限**设置为 2，**上限**设置为 3。

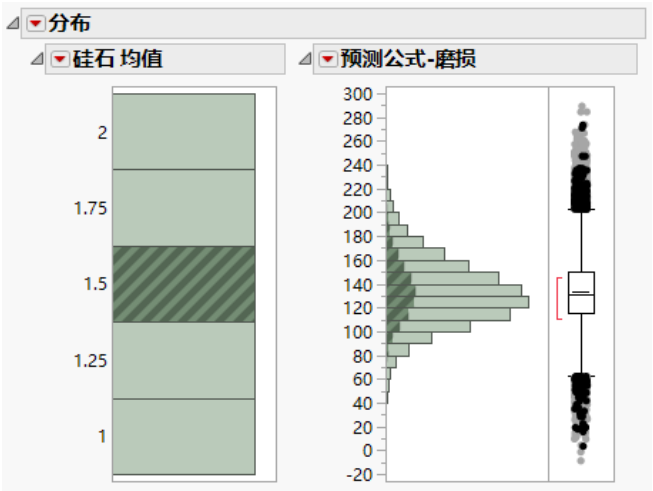
图 8.3 “排序”设置

12. 点击**制表**。

    硅石均值、**硅烷均值**和**硫磺均值**等列包含跨其各自的值范围的五个步骤。例如，**硅石均值**的各值为 1、1.25、1.5、1.75 和 2。“硅石”、“硅烷”和“硫磺”等列是来自以下正态分布的模拟值，该正态分布的均值由相应的均值列和固定标准差定义。为一组模拟值计算**预测公式 - 磨损**和**预测公式 - 弹性系数**值，以便您可以探索响应如何随因子值的变化而变化。
13. 选择**分析 > 分布**。
14. 选择**硅石均值**和**预测公式 - 磨损**，然后点击 **Y，列**。

15. 点击**确定**。

图 8.4 硅石均值的分布对预测公式 - 磨损的影响



点击与**硅石均值**对应的直方图直条，查看磨损的预测值对于选定的硅石值如何在模拟的硅烷和硫磺范围之间变动。

启动 “模拟器”

通过 “预测刻画器”、“等高线刻画器” 和 “定制刻画器” 中的红色小三角菜单选项可访问 “模拟器”。从 “预测刻画器” 启动 “模拟器” 后，因子规格显示在每个因子的刻画图下方。从 “等高线刻画器” 或 “定制刻画器” 启动 “模拟器” 时，因子规格显示在位于相应的刻画器报表下方的 “因子” 报表中。请参见第 141 页的 “因子模拟设置”。

一旦访问 “模拟器”，您就可以在 “响应” 报表中定义响应的模拟参数。请参见第 143 页的 “响应模拟设置”。“响应” 报表还支持您设置模拟数。

从 “预测刻画器” 访问 “模拟器” 时，还会向每个响应的右侧添加直方图。直方图下方是**模拟**按钮。一旦指定因子和响应设置，点击**模拟**可模拟响应值。每次点击 “模拟” 按钮都会更新模拟响应直方图。要模拟 “等高线刻画器” 和 “定制刻画器” 中的响应，打开 “模拟结果输出表” 报表并点击**制表**。请参见第 145 页的 “缺陷刻画器”。

因子模拟设置

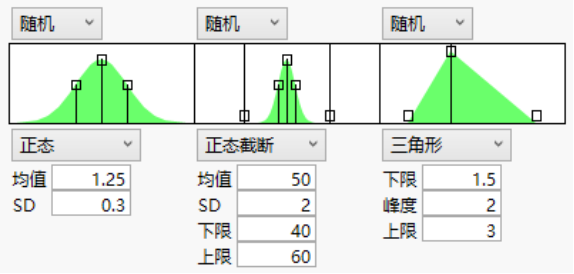
默认情况下，因子 （输入）和响应 （输出）由您在启动 “刻画器” 时选择的函数定义。模拟选项支持您修改因子和响应的模拟方式。提供针对位置、误差和噪声的控件。

固定 针对所有模拟试验将因子固定在刻画器中的当前值处。

随机 从指定的分布和分布参数为每次模拟试验抽取一个随机因子值。选择 “随机” 时会显示分布选项。有关这些选项的详细信息，请参见第 142 页的 “连续因子” 和第 143 页的 “分类因子”。

注意：默认的随机分布是均值位于当前因子设置且标准差由因子范围除以 5 估计得出的正态分布。若更改刻画器中的因子设置，正态分布的均值将自动更新。还有其他一些随机分布也会在因子设置更改时自动更新参数。“三角形” 分布更新全部三个参数， Cauchy 分布更新众数， Johnson Su 分布更新 θ。

图 8.5 三个随机分布的示例



表达式 基于 JSL 表达式生成因子值。这样可支持您灵活地使用所选的随机分布。例如，您可以创建删失的正态分布，该分布使用如下表达式确保值不为负数：

```
Max(0,RandomNormal(5,2))
```

在输入表达式后，点击**重置**按钮提交表达式。

多元 基于多元正态分布生成因子值。指定每个因子的均值和标准差。相关性矩阵在 “X 相关性规格” 报表中单独定义。您可以使用该选项适应相关因子。

图 8.6 为 “多元” 选项定义相关性矩阵

连续因子

许多可用的分布都使用 《使用 JMP》所述的标准随机函数。可用的专业分布的说明如下所示：

正态加权 从具有给定均值和标准差的加权正态分布生成值。权重是一个特定抽样方案，该方案用于模拟分布尾部的稀有事件。当您要模拟非常低的缺陷率时，这是一个不错的选择。请参见第 163 页的 “模拟器的统计详细信息”。

正态截断 从下限和上限所限定的正态分布生成值。任何超过某限值的随机生成值都将被丢弃，改为生成另一个值。当您想要模拟检查系统（其中不满足规格限的输入将被丢弃）时，这是一个不错的选择。

正态删失 从下限和上限所限定的正态分布生成值。任何超过某限值的随机生成值将被设置为该限值，这会在限值处提供密度量。当您想要模拟系统 （其中不满足规格限的输入会被返修，直到它们达到规格限）时，这是一个不错的选择。

抽样 通过从数据表中的某个因子列随机选择值，生成该因子的值。

外部 通过从另一个表中的列中随机选择值来生成因子值。系统将提示您选择表和列。

注意：当您选择“抽样”或“外部”时会显示“对齐”复选框。当两个或更多因子设置为“抽样”或“外部”时会使用该复选框。选中该框时，随机抽签来自表的同一行。这可以保持两列之间的相关性结构。若“对齐”选项用于关联不同表中的两个列，则这些列必须具有相同的行数。

分类因子

若因子为分类的，则随机分布通过为每个类别指定的概率表征。默认情况下，概率设置为数据表中的观测概率。您可以使用图中的控点或通过更改“概率”列中的数值来更改概率。概率总和必须为 1。

注意：若您选择的概率总和不为 1，则点击“模拟”按钮时，它们会自动进行规范化，使得总和为 1。

若某个因子的模拟设置为“随机”，则会显示随机分布密度的图形化表示。该图形显示连续分布的密度形式，并提供可被拖动来更改分布的控点。“正态”分布的拖点是均值以及均值加或减一个标准差。“正态截断”和“正态删失”则添加上限和下限对应的点。“均匀”和“三角形”具有限值控点，并且“三角形”添加众数。

响应模拟设置

响应的模拟设置支持您模拟响应中的随机噪声。您的因子往往解释响应中的一部分变异，而响应的其余变异由随机噪声造成。这是在模拟中指定随机噪声的方式。

无噪声 基于因子设置从模型生成响应。不向响应添加随机噪声。

添加随机噪声 向生成的响应添加均值为零且具有指定的标准差的正态随机值。

添加随机加权噪声 基于用于模拟分布尾部的稀有事件的加权抽样方案向生成的响应添加误差项。

添加多元噪声 基于多元正态分布向生成的响应添加误差项。指定响应的标准差。指定 “Y 相关性规格”报表中的响应之间的相关性结构。

按模型随机 （仅在 “拟合模型”平台的 “标准最小二乘”和 “广义回归”特质中可用。）基于 “拟合模型”启动窗口中指定的模型分布向生成的响应添加误差项。

“模拟器”报表

“模拟器”报表包含以下项：

模拟结果输出表 包含“制表”按钮，使用该按钮可将模拟结果保存到数据表中。若响应有规格限，该表包含**规格中的 Y** 列，它指示模拟的响应是否在规格限内。该表还包含一个目标列，标记为 Obj。若您在模拟前向刻画器添加了意愿函数，则该列是意愿函数的求值。若在模拟前不存在意愿函数，则该列包含所有缺失值。

排序 （仅当分布为“正态”、“均匀”或“三角形”随机分布时才可用。）“排序”支持您跨每个因子的位置或离差范围进行模拟。模拟结果保存到数据表中。选择一个选项将每个因子的位置或离差排序。指定步数和排序范围。对于步数的每个组合，模拟将运行“试验次数”。若您有两个项，每个项有 5 步且试验次数 = 100，则生成的模拟表将有 2,500 行。

“模拟器”报表选项

“模拟器”红色小三角菜单包含以下选项：

自动直方图更新 切换直方图更新。若选定该选项，当您拖动因子设置控点时将使用新的模拟值更新直方图。默认情况下未选定该选项。

缺陷刻画器 （仅当指定规格限时才可用。）将缺陷率显示为每个因子的孤立函数。请参见第 145 页的“缺陷刻画器”。

缺陷参数刻画 （当启动“缺陷刻画器”时可用。）将缺陷率显示为每个因子的分布参数的孤立函数。请参见第 147 页的“缺陷参数刻画器”。

层数 （仅在点击“模拟器”红色小三角菜单前按 Shift 键时才显示。）支持您指定“正态加权”分布的层数。请参见第 163 页的“模拟器的统计详细信息”。

设置随机种子 （仅在点击“模拟器”红色小三角菜单前按 Shift 键时才显示。）支持您指定模拟起始点的种子。这使得模拟结果可再现，除非将种子设置为零。默认情况下将种子设置为零。若种子非零，在点击“制表”按钮后输出最新的模拟结果。

模拟实验 （适用于“正态”、“均匀”和“三角形”随机分布。）在因子分布的位置上运行设计的模拟实验。将显示一个窗口，支持您指定设计点数、要在实验中使用的因子空间部分以及哪些因子在实验中发生变化。未选定用于实验的因子设置为其当前刻画器值。请参见第 148 页的“模拟实验”。

规格限 打开一个表，以便设置或编辑响应的规格限。点击**保存**按钮可将规格限保存到数据表中的响应列属性。请参见第 148 页的“规格限”。

缺陷刻画器

要使用“缺陷刻画器”，必须为至少一个响应定义规格限。“缺陷刻画器”将规格外的输出缺陷概率显示为每个因子的函数，而其他因子则随机变化。这用于帮助可视化该过程对其最为敏感的因子的分布变化，常常作为容差设计的一部分。

规格限用于定义缺陷，为每个因子分配的随机分布用于模拟响应。

在模拟设置中将至少一个因子定义为“随机”，以获取有用结果。否则，模拟输出为常数，在结果落在规格内时为 0，结果落在规格外时为 1。

提示：若您需要估计非常小的缺陷率，请考虑“正态加权”随机选项。该随机分布只用几千次模拟试验来很好地估计百万分之几的缺陷率。

容差设计简介

容差设计是调查如何通过控制输入因子的变异性来控制缺陷率。

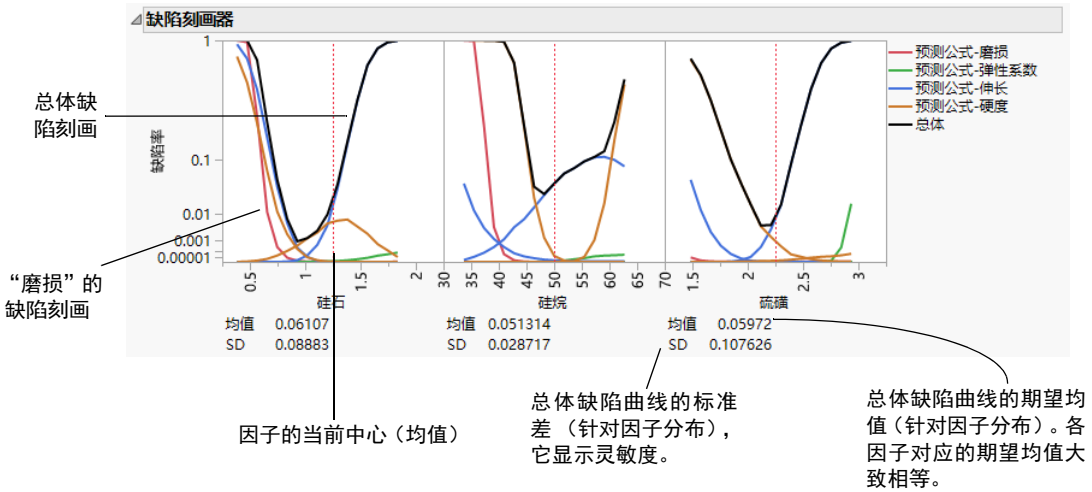
输入因子具有变异。规格限用于定义每个输入的可接受范围。输入因子中的变异性影响输出。规格限还用于定义输出变量的可接受范围。

有时，容差设计研究显示输入的规格限没有必要过于严格。放松此类限值可降低成本而不会牺牲产品质量。在这种情况下，容差设计可以省钱。

在其他情况下，容差设计研究可以发现更严格的限值或不同的目标导致更高的质量。在所有情况下，了解输入及其变异性如何影响缺陷率都十分重要。

“缺陷刻画器”显示缺陷率作为固定在指定均值处的每个因子的函数，而所有其他因子则根据它们的随机规格进行变化。若有多个带有规格限的输出，则每个输出都有不同的带颜色编码的缺陷率曲线。黑色曲线显示总缺陷率。

图 8.7 缺陷刻画器



图形尺度

缺陷率显示在立方根尺度上，这样不论是大的缺陷率还是小的缺陷率都可以较为详细地显示。

缺陷率

模拟的总缺陷率的均值和标准差 (SD) 在每个缺陷特征图下方报告。该均值是缺陷刻画曲线通过对指定因子的分布求积分计算得出。由于估计中的数值误差，在每个因子下报告的总均值缺陷率可能稍有不同。

在“缺陷刻画器”中报告的缺陷率是对模拟的总缺陷率的估计值。在“预测刻画器”分级显示项内的缺陷表的“比率”列中也报告了该缺陷率。

注意：在您运行模拟并且一个或多个响应定义了规格限后，缺陷表将添加到“预测刻画器”分级显示项。

由于通过求积分和模拟来获取比率估计值的方式不同，所以这些估计值可能稍有不同。若这些估计值极为不同，则您可能需要考虑增加模拟试验次数。此外，检查因子尺度的范围是否足够大，以便积分很好地涵盖分布。

标准差可以测量缺陷率对因子的灵敏度。若因子刻画线很平坦或因子分布具有很小的方差，则标准差很小。标准差越大，该因子的变化对于缺陷率变异性的影响就越大。通过比较各个因子的标准差，您可以选择因子以便降低缺陷率。

当您更改因子分布设置时，均值和标准差值随之更新。通过此方法，您可以用一次改变一个因子的方式来探索如何降低缺陷。您可以点击并拖动因子分布上的一个控点，在拖动时查看均值和标准差的变化。但是，在您点击“重新运行”按钮以生成更新的模拟试验前，不会更新各个因子的变化。

模拟方法和详细信息

假定 X2 和 X3 存在随机变异，您想进行因子 X1 的缺陷刻画。在 X1 的等间距值网格的 k 个点的每一个点执行一系列 n =“试验次数”的模拟试验。（默认情况下，k 设置为 17。）在每个网格点，由于规格限，假定有 m 个缺陷。在该网格点，缺陷率为 m/n。这些缺陷率被连接和标绘为 X1 的连续函数。

注意

- 分布更改时不自动重新计算刻画曲线。点击**模拟**更新该曲线。
- 缺陷刻画并不解决一般优化问题，即，在指定表示问题所有方面的函数的情况下针对成本来优化质量的问题。这个更一般的问题将受益于代理模型和空间填充设计。
- 缺陷刻画线低时，它们容易变得不平坦。一部分原因在于使用了立方尺度，因为立方尺度低值的差异会被放大。锯齿状的缺陷刻画线可能是因为模拟试验次数有限。若总体缺陷曲线（黑线）平滑且各个缺陷率比较一致，则您可能执行了足够的模拟试验以提供稳定的解。若总体缺陷曲线呈锯齿状，则考虑增加试验次数。20,000 次通常可以使曲线稳定下来。

缺陷参数刻画器

“缺陷参数刻画器”报表显示过程更改对缺陷率的影响。这些影响基于每个因子的模拟参数设置。考虑四种场景。

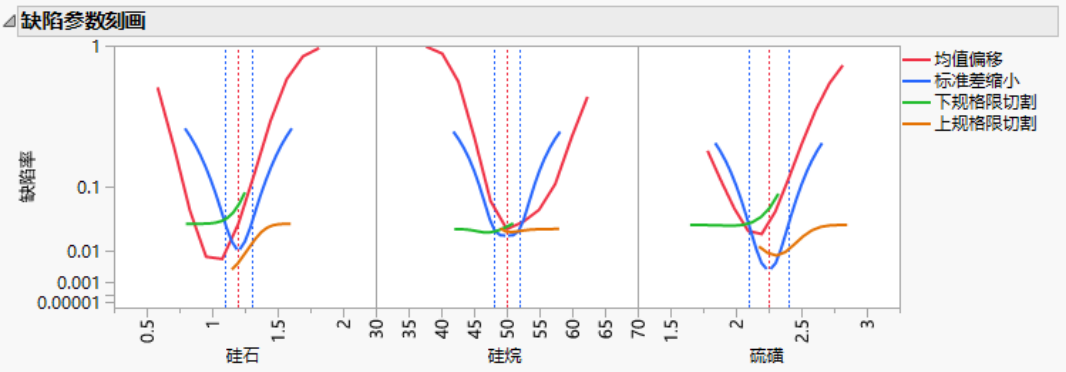
均值偏移 均值偏移的影响用红色曲线显示。当前均值用垂直的红色虚线显示。

标准差缩小 减小变异性的影响用蓝色曲线显示。在均值加上和减去一个标准差的地方设置垂直的蓝色虚线。曲线上的最小值对应于没有变异性时的缺陷率。

下规格限切割 删除下规格限以下所有部分对检查的影响通过绿色曲线显示。

上规格限切割 删除上规格限以上所有部分对检查的影响通过橙色曲线显示。

图 8.8 缺陷参数刻画



模拟实验

使用“模拟实验”选项可在因子分布的位置上运行设计的模拟实验。选择“模拟实验”时，您需要指定设计点数、要在实验中使用的因子空间部分以及哪些因子包括在实验中。对于未包括在实验中的因子，“刻画器”中设置的当前值为实验中使用的值。

实验设计为拉丁超立方设计。在每个设计点处，使用该设计点作为随机抽签的中心生成次数等于“试验次数”的随机抽签。形状和变异性来自指定的分布。对于每个设计点输出一行。响应包括具有规格限的每个响应的缺陷率以及总体缺陷率。您可以使用“高斯过程”模型对模拟实验中的总缺陷率建模。有关“高斯过程”模型的详细信息，请参见《预测和专业建模》。

规格限

要向变量添加规格限，请右击所需列并选择**列属性 > 规格限**。“列属性”窗口提供了用于输入“下规格限”、“目标”和“上规格限”的值的框。或者，您可以使用“模拟器”红色小三角菜单中的**规格限**选项定义“规格限”。

刻画器支持响应的规格限并提供若干功能：

- 在“刻画器”中，若没有在输入数据表中设置“响应限”属性来提供意愿坐标，JMP 将查找“规格限”列属性并构造适合这些规格限的意愿函数。有关“响应限”的详细信息，请参见《使用 JMP》。
- 若使用“模拟器”输出模拟表，JMP 会在输出数据表中包括“规格限”列属性。这样使得解释缺陷率和能力指标变得很容易。
- 添加规格限将启用“缺陷刻画器”。

模拟器的其他示例

- “缺陷刻画器的示例”
- “随机优化的示例”
- “模拟常规公式的示例”

缺陷刻画器的示例

为了演示使用“缺陷刻画器”的可能工作流程，我们使用 Tiretread.jmp。Tiretread.jmp 样本数据表中的实验数据来自于以下实验：研究硅石、硅烷和硫磺对轮胎胎面性能四个测度的影响。

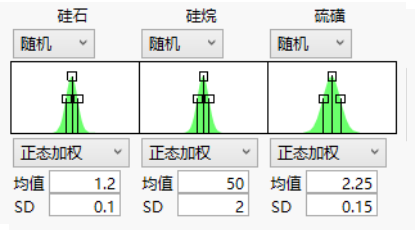
在“模拟器”中添加规格限

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**图形** > **刻画器**。
3. 选择**预测公式 - 磨损**、**预测公式 - 弹性系数**、**预测公式 - 伸长**和**预测公式 - 硬度**，然后点击 **Y，预测公式**。
4. 点击**确定**。
5. 点击“预测刻画器”红色小三角并选择**模拟器**。
6. 点击“模拟器”红色小三角并选择**规格限**。

   注意：若列的规格限另存为列属性，这些规格限显示在模拟器的“规格限”表中。

7. 设置规格限：
   - 对于“磨损”，将“下规格限”设置为 110。
   - 对于“弹性系数”，将“上规格限”设置为 2000。
   - 对于“伸长”，将“下规格限”设置为 350，“上规格限”设置为 550。
   - 对于“硬度”，将“下规格限”设置为 66，“上规格限”设置为 74。
8. 点击**保存**将“规格限”保存至数据表。
9. 在每个因子下方选择**随机**。
10. 对于每个因子，输入如图 8.9 中所示的随机规格。

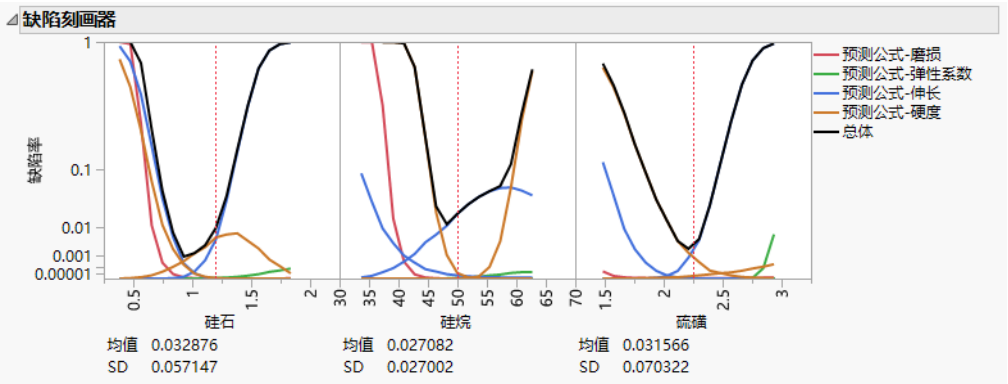
图 8.9 刻画器随机规格



打开“缺陷刻画器”和“缺陷参数刻画”

1. 点击“模拟器”红色小三角并选择**缺陷刻画器**。

图 8.10 缺陷刻画器



注意：缺陷刻画器基于带有随机输入的模拟。您获得的值可能不同于图 8.10 中显示的值。

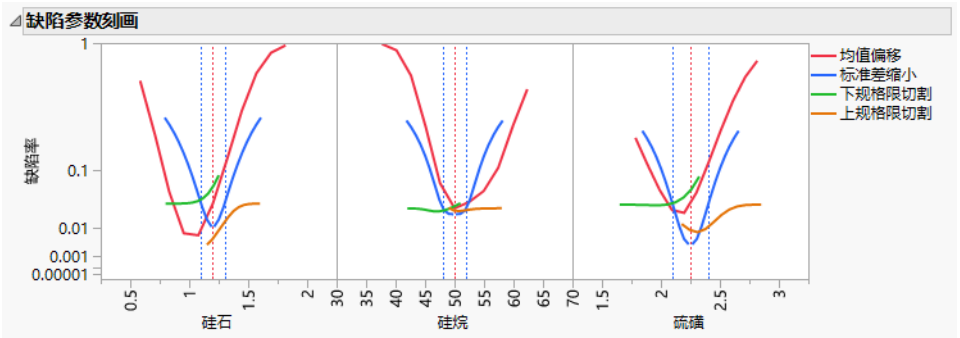
黑色曲线显示每个因子的总缺陷率固定在 X 轴上的值处，而其他所有因子都是变化的。

考虑**硅石**的总体曲线。随着硅石的变化，缺陷率在硅石大约为 1 时达到最低的 0.001，随着硅石从 1 上升或下降，缺陷率快速上升到接近 1。不过，**硅石**本身是随机的。若您对**硅石**的密度曲线求积分，可以估计出平均缺陷率大约为 0.03，该值显示为**硅石**的均值。这个由模拟所估计的总缺陷率估计值显示在模拟直方图下方的缺陷表中。所有因子的总缺陷率的均值都相似。

“缺陷刻画器”还包含针对每个因子的变异得到缺陷率标准差的估计值。对于**硅石**，该值（标记为“标准差”）为 0.057。该标准差与针对该因子分布的缺陷率的灵敏度相关。比较三个因子的标准差值，**硫磺**的标准差高于**硅石**和**硅烷**的标准差值。这指示要改善缺陷率，偏移**硫磺**的分布应该最有效。可以通过以下方式偏移分布：更改它的均值、更改它的标准差或通过拒绝不符合某些规格限的输入截去一部分分布。

2. 点击 “模拟器”红色小三角并选择**缺陷参数刻画**。

图 8.11 缺陷参数刻画



考虑**硫磺**并注意当前缺陷率 (0.03) 用四种方式表示，分别对应 “参数刻画器” 中的四条曲线。

对于红色曲线 “均值偏移”，当前缺陷率为红色曲线与垂直的红色虚线相交的地方。“均值偏移” 曲线表示**硫磺**的均值变化时总体缺陷率的变化。降低缺陷率的一个方法是将均值略微向左偏移。若您使用该图上的十字准线工具，可以看到均值向下偏移将缺陷率降低到约为 0.02。

对于蓝色曲线 “标准差缩小”，当前缺陷率表示蓝色曲线与两条蓝色虚线相交的地方。“标准差缩小” 曲线表示标准差变化时缺陷率的变化。蓝色虚线表示当前均值上下各一个标准差。围绕中心对称绘制蓝色曲线。在中心处，蓝色曲线达到最小值，这表示标准差为零的缺陷率。这表示若我们彻底消除**硫磺**的变异，缺陷率约为 0.003。这比当前的缺陷率 0.03 低得多。若您查看其他缺陷参数刻画曲线，可以看到这比减小其他因子的变异效果好得多，这就是我们之前从**硫磺**的标准差值大小就有的这种猜测。

对于绿色曲线 “下规格限切割”，缺陷率没有改进机会。绿色曲线整个都在当前缺陷率之上。这指示通过拒绝**硫磺**的过小值部分来减小变异对于降低缺陷率没有用。

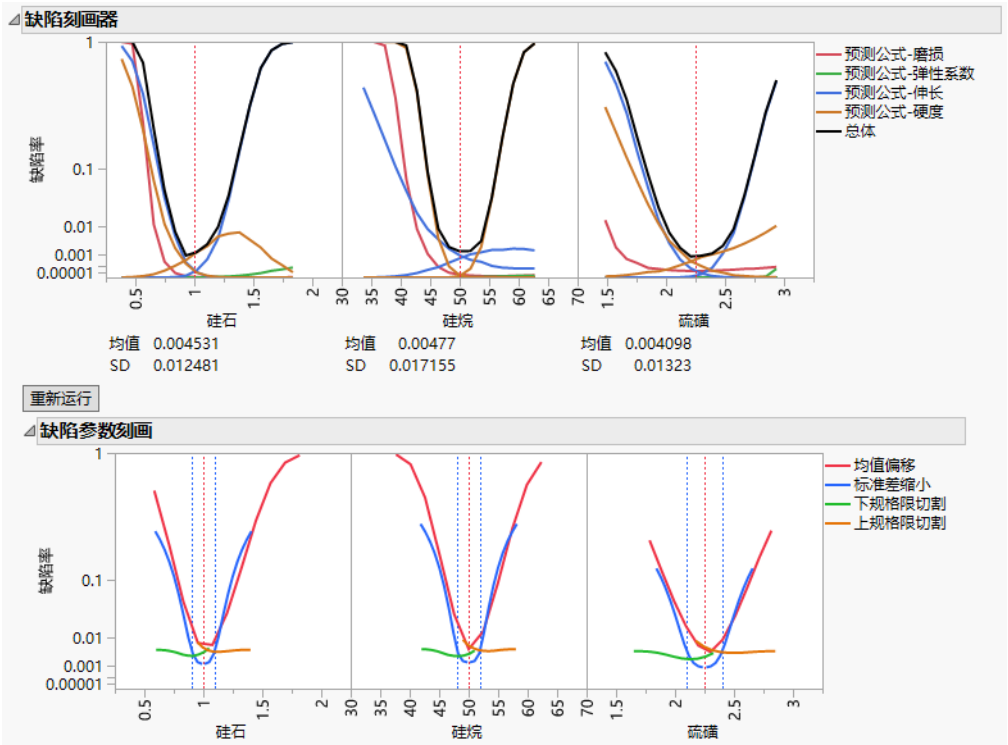
橙色曲线 “上规格限切割” 建议了一种降低缺陷率的方式。从右边读取曲线，曲线开始在当前缺陷率 (0.03) 处。然后当您通过降低**硫磺**的上规格限拒绝更多部件时，缺陷率随之改善。但是，将规格限移到分布的中心等效于丢掉了一半部件，这可能不是可行的解决方案。

查看所有因子上的所有机会，现在看起来对于进一步调查有两个不错的选项。将**硅石**的均值偏移到约为 1，或是减小**硫磺**中的变异。因为实际上更改过程均值通常比更改过程变异容易，因此首次调节的最好方法可以是将**硅石**的均值偏移到 1。

3. 在 “预测刻画器” 中，将**硅石**的均值从 1.2 调整到 1.0。
4. 在 “缺陷刻画器” 下，点击**重新运行**。

   “缺陷刻画器” 随着调整的**硅石**值更新。

图 8.12 已调整的缺陷率

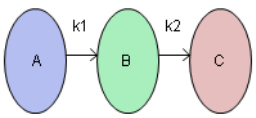


通过将“硅石”的均值从 1.2 偏移到 1.0，缺陷率从 0.03 降到了大约 0.004，这是一项不错的改进。通过更改分布和重新运行模拟，调查其他缺陷降低方案将变得简单易行。

随机优化的示例

本例改编自 Box and Draper (2007)，并使用 Stochastic Optimization.jmp。一个化学反应将化学物质 A 转换为化学物质 B，再从 B 转换为 C。化学物质“B”的生成量为反应时间和反应温度的函数。使用刻画器和模拟器探索因子设置以优化 B 的产量。

图 8.13 化学反应



对于该反应，可以使用阿伦尼乌斯定律计算 B 的产量。**产量**列包含产量的公式。该公式是**反应时间**（小时）和反应速度 k1 和 k2 的函数。反应速度是**反应温度**（开氏度）和已知的物理常数 θ_1、θ_2、θ_3、θ_4 的函数。因此，**产量**是**反应时间**和**反应温度**的函数。

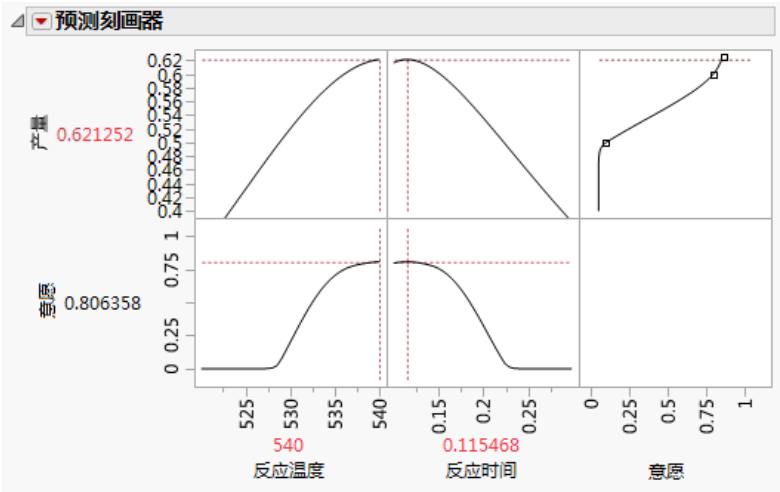
使用意愿函数使产量最大化

1. 选择**帮助** > **样本数据库**，然后打开 Stochastic Optimization.jmp。
2. 选择**图形** > **刻画器**。
3. 选择**产量**并点击 **Y，预测公式**。
4. 点击**展开中间公式**，然后点击**确定**。

   启用了意愿函数的“预测刻画器”随即出现。有关意愿函数的详细信息，请参见“刻画器”一章中第 47 页的 “意愿刻画和优化”。
5. 点击“预测刻画器”红色小三角并选择**优化和意愿** > **最大化意愿**。

   “预测刻画器”将最大化**产量**，并将图形设置为**反应时间**和**反应温度**的最优值。

图 8.14 产量最大值



最大**产量**约为 0.62，此时**反应时间**为 0.115 小时，**反应温度**为 540 开氏度，即在高温下快速反应。（由于优化过程的起始值是随机的，您的结果可能略有不同。）

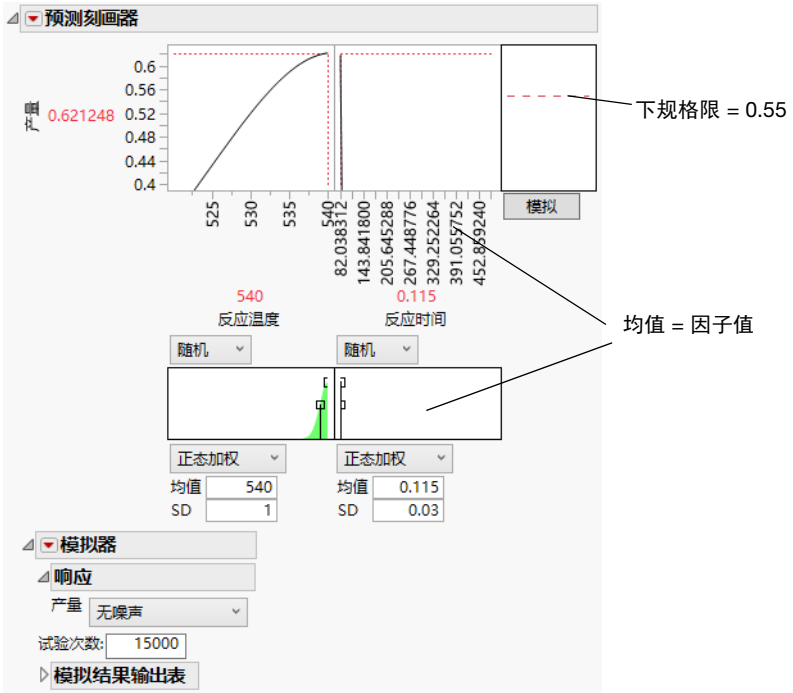
模拟产量最大化的缺陷率

在生产环境中，有时无法精确控制过程输入。若输入（**反应时间**和**反应温度**）具有随机变异，**产量**会有什么变化呢？此外，若**产量**有规格限，规格外的批次所占百分比是多少？“模拟器”还可以帮助您调查在指定**反应时间**和**反应温度**变异的条件下**产量**的变异和缺陷率情况。

1. 点击“预测刻画器”红色小三角，然后取消选择**最优化和意愿** > **意愿函数**。
2. 点击“预测刻画器”红色小三角并选择**模拟器**。

3. 将**反应温度**的模拟参数设置为**随机 > 正态加权**，且均值 = 540，标准差 = 1。
4. 将**反应时间**的模拟参数设置为**随机 > 正态加权**，且均值 = 0.115，标准差 = 0.03。
5. 将**试验次数**值设置为 15,000。

图 8.15 “模拟器”初始设置

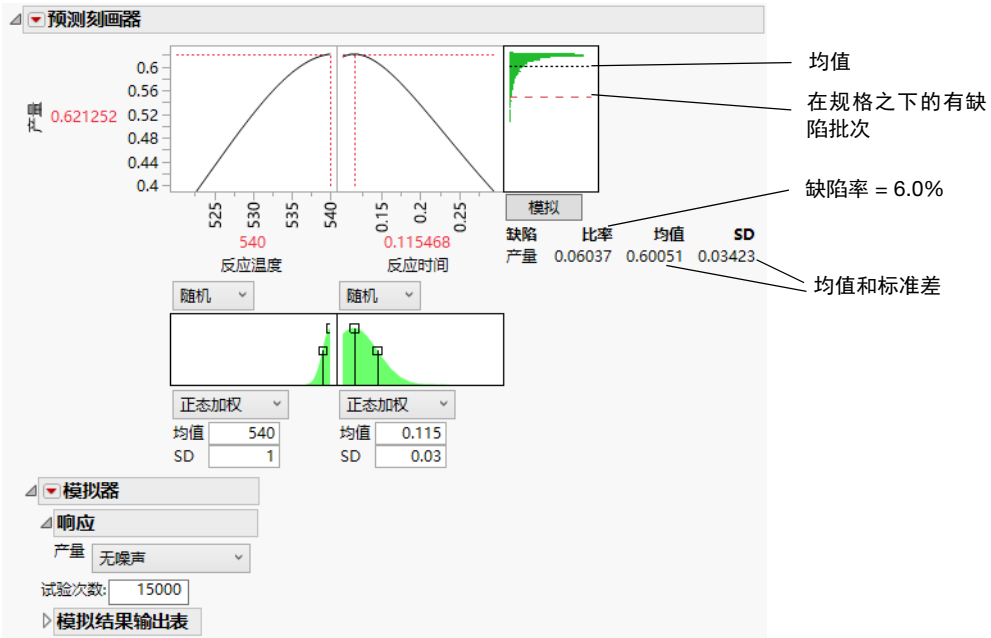


产量具有下规格限 0.55，将它设置为列属性并在图 8.15 上显示为红色虚线。

6. 点击**模拟**按钮。

注意：由于模拟中随机抽签，您的数字可能不同于图 8.16 中显示的数字。

图 8.16 模拟结果



对于 “反应温度” 540 和 “反应时间” 0.115，预测的**产量**为 0.62。在假定温度和时间存在变异的条件下，模拟估计的平均缺陷率约为 6%，标准差为 0.03。缺陷率 6.0% 指示约有 6.0% 的批次将超出规格。

使用模拟器探索**反应温度**和**反应时间**的其他设置，这些设置可能保持较高的**产量**，但缺陷率较低。更改设置前，应保存这些提供最大**产量**的因子设置以供将来使用。

7. 点击 “预测刻画器” 红色小三角并选择**因子设置 > 记住设置**。
8. 键入 “最大产量” 并点击**确定**。

   将这些设置追加到报表窗口。

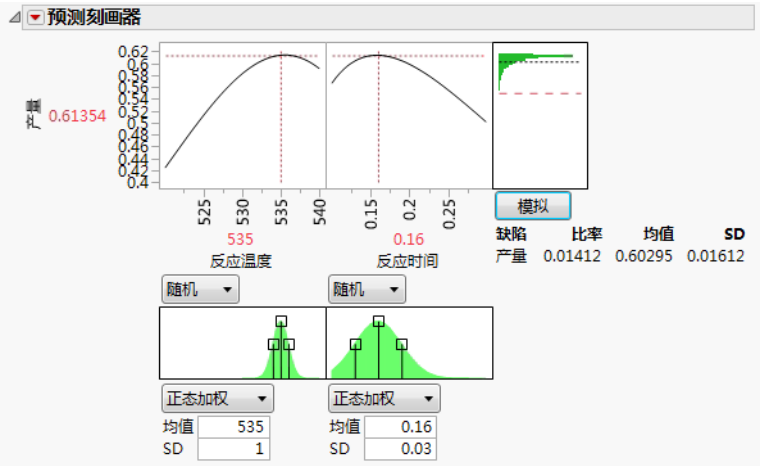
图 8.17 最大产量记住的设置

9. 将**反应温度**的均值设置为 535。

   使用**反应时间**图中的红色虚线，您可以探索和确定使**产量**最大化的近似值。该值应在 0.16 左右。
10. 将**反应时间**的均值设置为 0.16。
11. 点击**模拟**。

注意：由于模拟中随机抽签，您的数字可能不同于图 8.18 中显示的数字。

图 8.18 温度为 535 时的缺陷率



通过对输入因子设置稍作更改，缺陷率降至大约 1.8%，预测产量降低不足 0.01。使**产量**最大化的固定 （无变异性）设置与存在因子变异时缺陷率最小化的设置不同。

模拟实验

您可以运行模拟实验以便找到使缺陷率最小化的**反应温度**和**反应时间**设置。为此，模拟**反应温度**和**反应时间**的实验设计中的每个点的缺陷率。然后拟合缺陷率的预测模型并查找因子设置以使缺陷率最小化。

1. 点击“模拟器”红色小三角并选择**模拟实验**。
2. 将实验次数设置为 80 并将设计空间部分设置为 1，以便使用实验中的整个因子空间。
3. 点击**确定**。

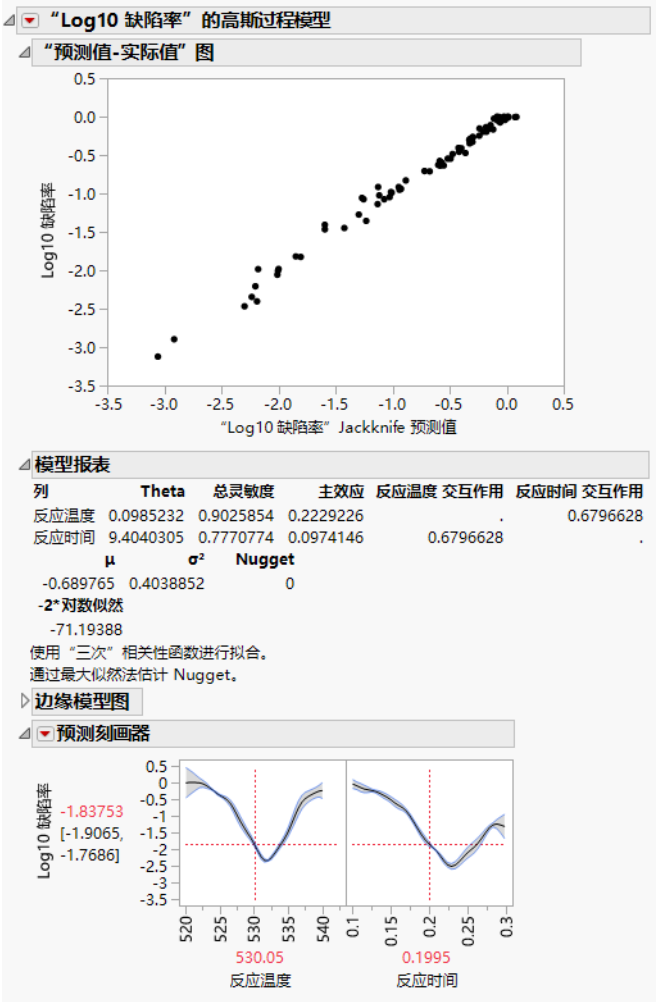
   在指定的因子空间中生成包含 80 个设计点的拉丁超立方设计，在每个设计点抽取次数等于“试验次数”的随机抽签。设计点是随机抽签的中心，随机抽签的形状和方差取自因子分布设置。

   实验结果被制成一个数据表。在每个设计点给出 “总体缺陷率” 。您现在可以拟合一个模型，该模型预测作为**反应温度**和**反应时间**的函数的缺陷率。

注意：不关闭 Stochastic Optimization “刻画器”窗口。您之后还会回到该窗口。

4. 从新的数据表，点击 “高斯过程”脚本旁边的绿色小三角。

图 8.19 “高斯过程”模型拟合的结果

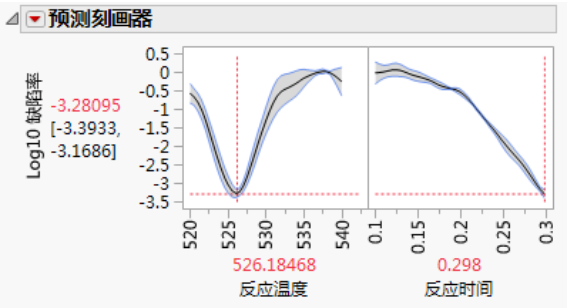


注意：由于模拟中随机抽签，您的结果可能略有不同。

5. 要查找使缺陷率最小化的**反应温度**和**反应时间**设置，请点击“预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

   已设置意愿函数使缺陷率最小化。

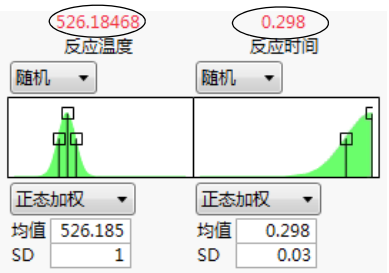
图 8.20 最小缺陷率的设置



使缺陷率最小化的设置约是**反应温度** = 526，**反应时间** = 0.3。

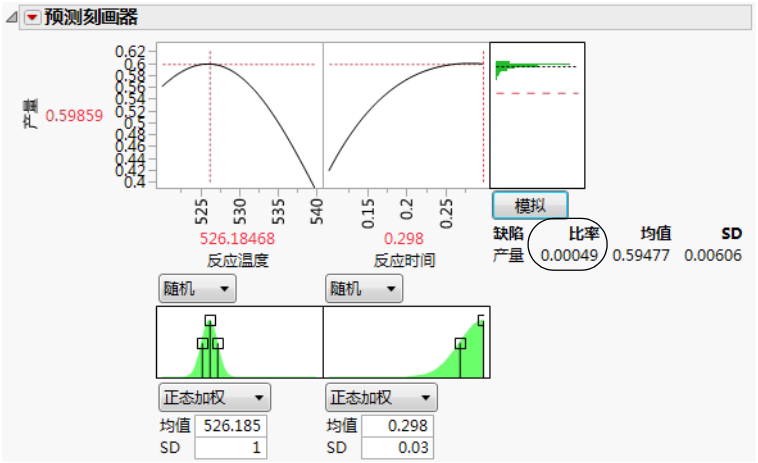
6. 点击**传输回因子设置**按钮。
   这将更新原始的刻画器报表，以便使用使得缺陷率最小化的**反应温度**和**反应时间**。
7. 返回到原始 “刻画器” 报表窗口。
8. 点击 “预测刻画器” 红色小三角并选择**因子设置 > 记住设置**。
9. 键入 “最小缺陷” 并点击**确定**。

图 8.21 最小缺陷设置



10. 指定新设置后，点击**模拟**按钮以估计新设置下的缺陷率。

图 8.22 更低的缺陷率

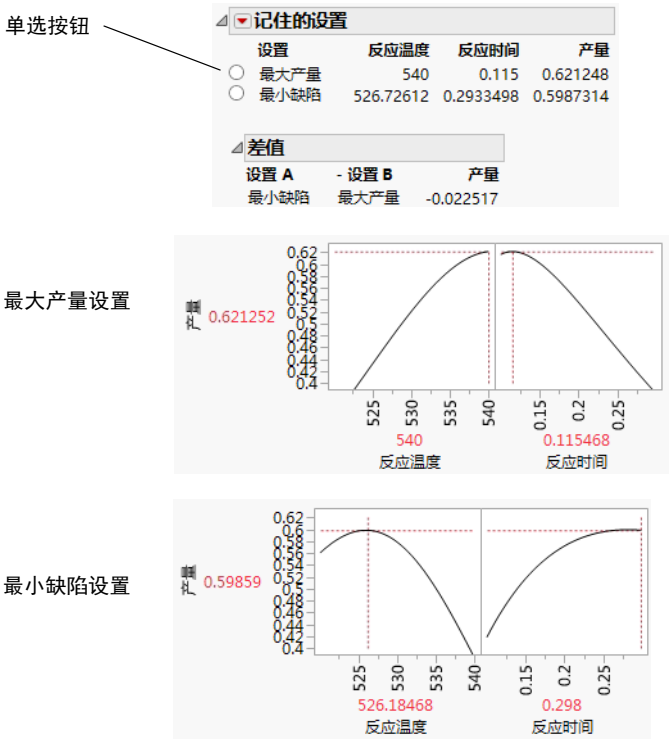


新设置下的缺陷率为 0.05%。这比使**产量**最大化的设置下的 6.0% 低很多。这几乎降低了大约 120 倍。我们回想一下，第一种设置下的平均**产量**为 0.62，而新平均产量为 0.59。缺陷率降低 120 倍的代价是平均**产量**减少了 0.03。

因为您使用“记住设置”保存了设置，所以可以轻松比较旧设置和新设置。“差值”报表总结了这些差值。

11. 点击**记住的设置**单选按钮可以查看每种设置的刻画器。

图 8.23 设置比较



化学家现在知道了将哪个设置用于质量过程。若因子没有变异，则使**产量**最大化的设置为高温下快速反应。但是，若该过程输入有类似于我们模拟的变异，使**产量**最大化的设置会导致高的缺陷率。因此，要在有因子变异的情况下使缺陷率最小化，设置应为低温下慢速反应。

模拟常规公式的示例

尽管刻画器和模拟器被设计为使用从模型拟合存储的公式，但是它们适用于在列中存储的任意公式。模拟的典型应用是在某些概率场景下运用财务模型以获取目标对象的分布情况。可以在 JMP 中完成该操作。关键是要将公式存储到列中、设置范围，然后执行模拟。

1. 在“JMP 主窗口”中，选择**文件 > 新建 > 脚本**。这将打开新的脚本窗口。
2. 将以下 JSL 脚本复制并粘贴到新的脚本窗口中。

```
dt = New Table( "销售量模型" );
dt << New Column( "单位销售量", Values( {1000, 2000} ) );
dt << New Column( "单价", Values( {2, 4} ) );
dt << New Column( "单位成本", Values( {2, 2.5} ) );
dt << New Column( "收入",
   Formula( :Unit Sales * :Unit Price )
);
dt << New Column( "总成本",
```

```
   Formula( :Unit Sales * :Unit Cost + 1200 )
);
dt << New Column( "利润",
   Formula( :Revenue - :Total Cost ),
   Set Property( "规格限", {LSL( 0 )} )
);
Profiler(
   Y( :Revenue, :Total Cost, :Profit ),
   Objective Formula( Profit )
);
```

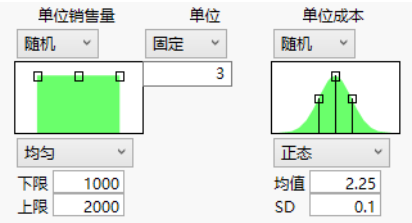
3. 点击“运行脚本”图标 运行该脚本。也可以选择 Ctrl-R。

   该脚本使用一些初始尺度数据创建图 8.24 中的数据表并将公式存储到输出变量。它还启动“预测刻画器”。

图 8.24 从脚本创建的数据表

4. 点击“预测刻画器”红色小三角并选择**模拟器**。
5. 在**单位销售量**和**单位成本**中，选择**随机**。
6. 填充因子参数：
   - **单位销售量**是**均匀**的，下限为 1000 且上限为 2000。
   - **单价固定**在 3。
   - **单位成本**呈均值为 2.25 且标准差为 0.1 的**正态**分布。

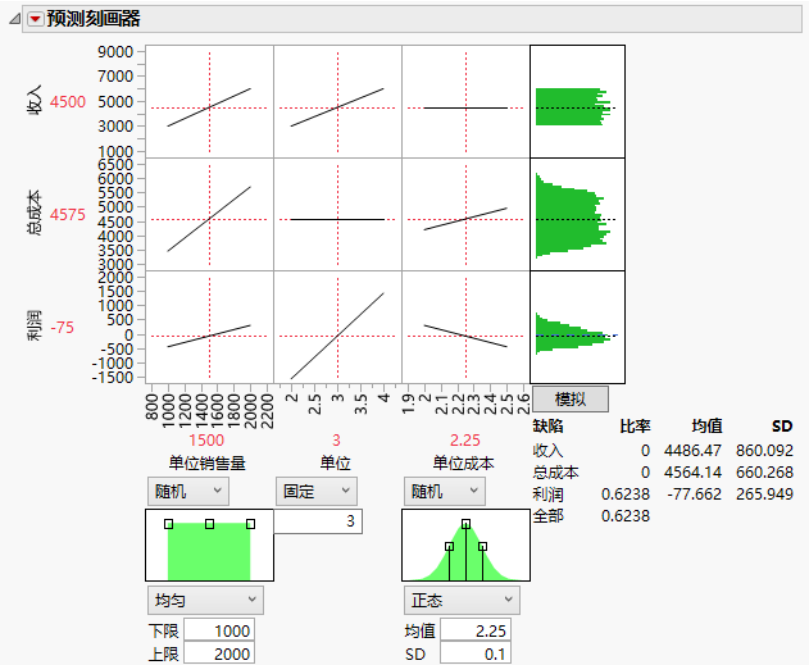
图 8.25 “预测刻画器”规格



7. 点击**模拟**按钮。

注意：由于模拟中随机抽签，您的数字可能不同于图 8.26 中显示的数字。

图 8.26 模拟器



看起来这些模型不太可能盈利。通过对**利润**设置为零的下规格限，缺陷报表显示不盈利的概率为 62%。

8. 将**单价**的固定值改为 3.25。
9. 点击**模拟**按钮。

图 8.27 结果

缺陷	比率	均值	SD
收入	0	4860.35	931.767
总成本	0	4564.14	660.268
利润	0.214	296.211	327.227
全部	0.214		

现在不盈利的概率下降到约为 21%。

若单价不能再提高了，而您想进一步降低不盈利的概率，现在应设法降低成本或提高销售量。

模拟器的统计详细信息

正态加权分布

JMP 将**多元径向分层**方法用于使用“正态加权”分布的每个因子。这似乎比使用很多重要性抽样方法效果好，并且在极端尾部中估计时是准确的。

首先，定义层数。层是一个以 0 为中心的超球面网。对于 d 个随机因子，层按其径向间隔定义。请参见表 8.1。

表 8.1 层间隔

层数	内部距离	外部距离
0	0	$\sqrt{d}$
1	$\sqrt{d}$	$\sqrt{d+\sqrt{2d}}$
2	$\sqrt{d+\sqrt{2d}}$	$\sqrt{d+2\sqrt{2d}}$
i	$\sqrt{d+(i-1)\sqrt{2d}}$	$\sqrt{d+i\sqrt{2d}}$
$N_{层}-1$	$\sqrt{d+\{(N_{层}-1)-1\}\sqrt{2d}}$	$\sqrt{d+(N_{层}-1)\sqrt{2d}}$

默认层数为 16。要更改层数，请使用隐藏的命令“层数”，在点击“模拟器”旁边的红色小三角菜单时按住 Shift 键可显示该命令。根据需要增加样本大小以保持层数为偶数。

对于每个模拟试验都执行以下操作：

1. 选择一个层作为 $\mathrm{mod}(i-1, N_{层})$（针对第 i 次运行）。
2. 通过将多元正态分布 (0,1) 偏差缩放为单位范数，确定随机 n 维方向。
3. 使用对随机均匀参数的层合适的卡方分位数确定随机距离。

4. 缩放变量以便范数为随机距离。
5. 分别按每个因子指定的方式来缩放并变量中心化。

使用正确的权重估计时，所得的因子分布为具有合适的均值和标准差的多元正态分布。请注意，您不能将分布标准差与权重一起使用，因为它不估计期望的值。但是，将权重与一个较大的值（如 10^{12}）相乘，然后使用所得的值作为频数值可以得到正确的标准差。

第 9 章

Excel 刻画器

直观演示 Microsoft Excel 中保存的模型

Excel 的 JMP 插件使用 JMP 刻画器直观演示 Excel 工作表中储存的模型（或公式）。安装 JMP 时您可以安装 Excel 插件。

图 9.1 使用 Excel 模型的 “预测刻画器”示例

目录

“Excel 刻画器”概述 167
Excel 模型的示例 167
运行 JMP 刻画器 169
使用线性约束 169
刻画线的分辨率 170
从 JMP 使用 Excel 刻画器 170

“Excel 刻画器”概述

Excel 的 JMP 插件使用 JMP 刻画器直观演示 Excel 工作表中储存的模型 （或公式）。安装 JMP 时您可以选择安装 Excel 插件。在 Excel 插件中刻画是包含两个步骤的过程：

1. 点击**创建 / 编辑模型**按钮（Excel 2010 - 2016）以输入有关 JMP 所需的模型的信息。这对每个模型仅需要执行一次。有关详细信息，请点击“创建 / 编辑模型”窗口中的**帮助**。
2. 点击**运行模型**按钮（Excel 2010 - 2016）以启动 JMP 刻画器并运行 Excel 模型。请参见第 169 页的 “运行 JMP 刻画器”。

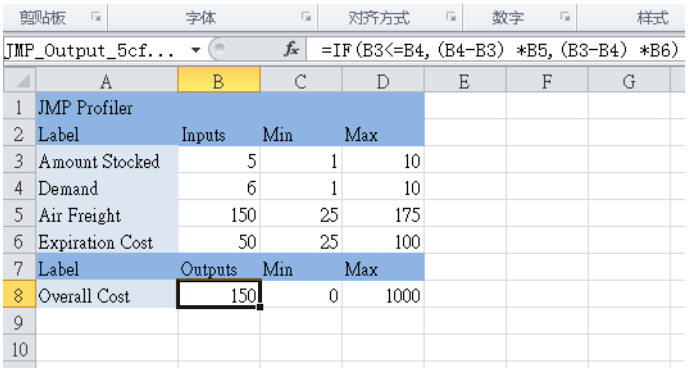
注意：

- 无需使用“首选项”、“数据表”、“图形生成器”、“分布”、“以 X 拟合 Y”、“拟合模型”、“时间序列”和“控制图”等按钮来刻画 Excel 模型。有关这些功能的详细信息，请参见《使用 JMP》。
- 安装该插件时会将一个 JMP 功能区添加到 Microsoft Excel。若没有 JMP 功能区，请通过以下方式安装该插件：双击 JMP setup.exe 文件，依次选择**修改**和 **Excel 插件**，然后点击**下一步**。

Excel 模型的示例

Excel 模型包含一个或多个 Excel 公式。每个公式必须是一个或多个其他单元格的函数。该示例使用位于 Samples/Import Data 文件夹中的 Demand.xls 样本导入数据。

图 9.2 Excel 中的需求模型



	A	B	C	D	E	F	G
1	JMP Profiler						
2	Label	Inputs	Min	Max			
3	Amount Stocked	5	1	10			
4	Demand	6	1	10			
5	Air Freight	150	25	175			
6	Expiration Cost	50	25	100			
7	Label	Outputs	Min	Max			
8	Overall Cost	150	0	1000			
9							
10							

关于 Demand.xls 样本导入数据

单元格 B8 中的公式是与不同的库存产品量有关的 Overall Cost 的计算。该公式显示在公式栏中，是四个单元格的函数：

- Amount Stocked 是库存产品量。

- Demand 是客户对该产品的需求。
- Air Freight 是指当需求超过库存量时空运更多产品所导致的单位成本。
- Expiration Cost 是指当需求小于库存量时处置未用产品所导致的单位成本。

公式计算取决于 Amount Stocked 与 Demand 之间的关系。

- 若 Amount Stocked 小于 Demand，则公司必须运送更多产品，其成本为：(Demand-Amount Stocked) × Air Freight。例如，若需求为 8，但是公司库存量为 6，则必须空运 8 - 6 = 2 个单位，成本为：2 × 150 = 300。
- 若 Amount Stocked 大于 Demand，则公司必须处置未用的产品，成本为：(Amount Stocked-Demand) × Expiration Cost。例如，若需求为 5，但是公司库存量为 8，则必须处置 8 - 5 = 3 个单位，成本为：3 × 50 = 150。
- 若 Amount Stocked 等于 Demand，则没有运送成本或处置成本。
- 不可能同时具有运送成本和处置成本。

在 Excel 中创建模型

1. 选择**帮助 > 样本数据库**，然后向上导航一级到 Samples/Import Data 文件夹。
2. 双击 Demand.xls 以在 Microsoft Excel 中打开该文件。
3. 在 Microsoft Excel 中，点击 JMP 功能区。
4. 点击**创建 / 编辑模型**按钮 。

   工作簿的名称显示在 “模型”和 “模型名称”字段中。

   使用工作表中的数据填充 “输入”和 “输出”字段。
5. 在 “模型名称”字段中输入 “客户需求”，然后点击**应用**。

   将更新 “模型”字段。
6. 在 “输入”框中选择 Air Freight，然后点击向下箭头按钮 。

   Air Freight 移到列表底部，因为您想让它在刻画器中最后一个显示。
7. 点击**确定**。

   将 Excel 模型保存到工作表。

注意：

- 若您设置模型时未填充“创建 / 编辑模型”窗口中的字段，请点击**选择**，然后选择包含输入或输出名称的单元格。
  - 对于输入，指定值并点击**应用**。输入必须为值，不能是公式。
  - 对于输出，指定的单元格必须是仅包含来自输入单元格的信息的公式。
- 您可以为一个工作表创建一个以上的模型。在“创建 / 编辑模型”窗口中，点击“模型”名称旁边的加号按钮 。在“模型名称”字段中，必要时更改该名称，然后点击**应用**按钮。您可以根据需要接着更改输入和输出，然后点击**确定**。

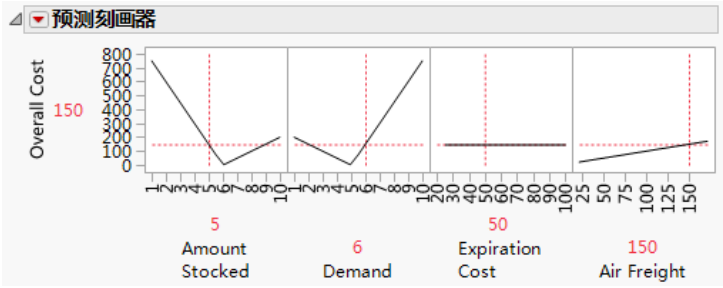
- 您必须在一个工作表上定义整个模型。一个模型不能引用另一工作表上的单元格。

运行 JMP 刻画器

使用 Excel 插件创建模型后，可以在 JMP 刻画器中运行它。从 Excel 插件执行以下操作：

1. 在 Microsoft Excel 中，点击 JMP 功能区。
2. 点击**运行模型**按钮。
3. 选择要运行的模型。
4. 点击**在 JMP 中刻画**。
5. 使用 JMP 刻画器可同时查看所有输入值对输出的影响。您还可以模拟一组输入组合来查看输出值的结果范围。

图 9.3 使用 Excel 模型的 “预测刻画器”的示例



注意：为确保原始 Excel 工作表不会更改，JMP 在控制所有刻画器计算的后台中运行 Excel 的隐藏副本。由于 Excel 工作表和刻画器未动态链接，在您更新工作表中的公式后，您必须重新启动 JMP 刻画器以反映更改。

使用线性约束

在 JMP 刻画器内，您可以更改线性约束以限制模型输入值。系统会提示您将约束保存至 Excel 工作簿。将约束保存到 Excel 工作簿后，无论何时从 Excel 插件刻画模型，都将带着约束条件。

1. 点击“预测刻画器”红色小三角并选择**更改线性约束**。
2. 点击**添加约束**。
3. 键入约束值。
4. 点击**确定**。
5. 点击 “预测刻画器”红色小三角并选择**保存线性约束**。

   系统会提示您将约束保存至 Excel 工作簿。

6. 点击**是**。

注意：当您保存 .xls 文件时，可能会看到兼容性错误。若出现这样的错误，请点击**继续**保存该文件。

工作簿将在 Excel 中打开。运行模型时，这些约束反映在 JMP 刻画器中。有关线性约束的详细信息，请参见“刻画器介绍”一章中第 34 页的“线性约束”。

提示：要删除某个线性约束，请将所有约束值设置为零。

刻画线的分辨率

“预测刻画器”红色小三角菜单上的“默认水平数”选项影响刻画线的分辨率。请注意下列信息：

- 刻画器运行在 Excel 中储存的模型时，该选项默认值为 17。
- 模型直接储存在 JMP 中时，该选项默认值为 41。

若同一模型同时储存在 Excel 和 JMP 中，则刻画模型时，显示的刻画线会不同。增加该值将导致“Excel 刻画器”运行更慢。

从 JMP 使用 Excel 刻画器

您在 Excel 文件中定义了模型的输入和输出后，可以从 JMP 内刻画模型。

1. 选择**图形 > Excel 刻画器**。
2. 找到包含模型的 Excel 文件，然后点击**打开**。
3. 若 Excel 文件包含多个模型，系统将提示您选择要刻画的模型。

使用以下代码为“Excel 刻画器”编写脚本：

```
Excel Profiler( "path to workbook", <"model name"> ) ;
```

若有多个模型且未指定模型，则显示包含可用模型列表的窗口。有关为“Excel 刻画器”编写脚本的详细信息，请参见《Scripting Guide》。

附录 A

参考书目

Box, G. E. P., and Draper, N. R. (1987). *Empirical Model–Building and Response Surfaces*. New York: John Wiley & Sons.

Box, G. E. P., and Draper, N. R. (2007). *Response Surfaces, Mixtures, and Ridge Analyses*. 2nd ed. New York: John Wiley & Sons.

Cornell, J. A. (1990). *Experiments with Mixtures*. 2nd ed. New York: John Wiley & Sons.

Derringer, G., and Suich, R. (1980). "Simultaneous Optimization of Several Response Variables." *Journal of Quality Technology* 12.4:214–219.

Hastie, T., Tibshirani, R., and Friedman, J. H. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2nd ed. New York: Springer-Verlag.

Saltelli, A. (2002). "Making best use of model evaluations to compute sensitivity indices." *Computer Physics Communications* 145:280–297.

Schafer, J., and Strimmer, K. (2005). "A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics." *Statistical Applications in Genetics and Molecular Biology* 4:1–30.

Sobol, I. M. (1993). "Sensitivity Estimates for Nonlinear Mathematical Models." *MMCE* 1.4:407–414.

附录 B

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

实验设计指南

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020–2021。JMP® 16 实验设计指南. Cary, NC: SAS Institute Inc.

JMP® 16 实验设计指南

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

实验设计指南

1 了解 JMP 23

文档和其他资源

JMP 文档中的格式规范 25

JMP 帮助 26

JMP 文档库 26

学习 JMP 的其他资源 32

JMP 教程 32

样本数据表 32

学习统计和 JSL 术语 33

学习 JMP 的提示与技巧 33

JMP 工具提示 33

JMP 用户社区 33

免费在线统计思维课程 33

JMP 新用户欢迎套件 34

统计学知识门户 34

JMP 培训 34

用户编写的 JMP 手册 34

“JMP 起始页”窗口 34

JMP 技术支持 34

2 实验设计介绍 35

“实验设计”平台概述

3 从实验设计入手 39

示例和关键概念

实验设计和实验设计工作流程概述 41

咖啡浓度实验 42

定义研究和目标 42

创建设计 44

运行实验 49
分析数据 50
实验设计工作流程：描述、指定、设计 56
定义响应和因子 57
指定模型 57
生成设计 58
评估设计 59
制表 60
实验设计的原则和方针 60
效应等级 60
效应遗传 60
效应稀疏 61
中心点、重复试验和检验 61

4 定制设计 63
构造符合您需要的设计

定制设计概述 65
定制设计示例 66
构建定制设计 81
响应 82
因子 83
定义因子约束 88
别名项 93
生成设计 94
设计 95
设计评估 95
输出选项 96
定制设计选项 98
模拟响应 101
保存 X 矩阵 103
开始数 105
设计搜索时间 105
定制设计的技术细节 106
包含随机化限制的设计 106
包含难以更改的水平的协变量 113

整区和子区数 114
最优性准则 114
D 效率 118
坐标交换算法 118

5 定制设计示例 121
执行满足您需要的实验

筛选实验 123
仅估计主效应的设计 123
估计所有双因子交互作用的设计 126
避免主效应与双因子交互作用存在别名关系的设计 128
超饱合筛选设计 131
固定区组的设计 138
响应曲面实验 142
响应曲面设计 142
包含灵活分区组的响应曲面设计 147
D 最优和 I 最优响应曲面设计的比较 150
包含约束和分类因子的响应曲面设计 154
混料实验 160
包含非混料因子的混料设计 160
混合混料设计 164
包含协变量的实验 168
包含固定协变量的设计 168
包含难以更改的协变量的设计 171
带有线性时间趋势的设计 174
包含随机化限制的实验 177
裂区实验 177
双因子裂区实验 181
针对稳健过程和产品设计的实验 187

6 扩充设计 191
向现有设计添加额外试验

使用“扩充设计”平台的示例 193
启动“扩充设计”平台 199
“扩充设计”窗口 200

因子 ... 200
定义因子约束 ... 201
扩充选择 ... 203
“扩充设计”平台选项 ... 204
扩充选择的更多示例 ... 207
重复设计 ... 207
向设计添加中心点 ... 209
创建折叠设计 ... 209
向设计添加轴点 ... 210
空间填充扩充 ... 211
统计详细信息 ... 212

7 确定性筛选设计 ... 215

确定性筛选设计概述 ... 217
确定性筛选设计示例 ... 219
确定性筛选设计 ... 219
与部分析因设计进行比较 ... 222
包含分区组的确定性筛选设计 ... 224
确定性筛选设计与 Plackett-Burman 设计的比较 ... 227
“确定性筛选设计”窗口 ... 231
响应 ... 231
因子 ... 233
设计选项 ... 235
设计 ... 237
设计评估 ... 237
输出选项 ... 237
确定性筛选设计选项 ... 239
模拟响应 ... 241
技术细节 ... 243
确定性筛选设计的结构 ... 243
实验数据分析 ... 245

8 “拟合确定性筛选”平台 249
分析确定性筛选实验中的数据
“拟合确定性筛选”平台概述 251
标识 DSD 中的活跃效应 251
DSD 的有效模型选择 251
“拟合确定性筛选”平台示例 252
拟合模型 252
检查结果 252
简化模型 254
启动“拟合确定性筛选”平台 255
“拟合确定性筛选”报表 256
阶段 1 - 主效应估计值 256
阶段 2 - 偶数阶效应估计值 257
组合模型参数估计值 258
主效应残差图 258
预测刻画器 258
“拟合确定性筛选”平台选项 258
“拟合确定性筛选”平台的技术细节 259
“DSD 的有效模型选择”算法 259
9 筛选设计 263
筛选设计概述 265
基础原则 265
筛选设计结果的分析 265
筛选设计示例 266
比较部分析因设计和主效应筛选设计 266
不存在标准设计的主效应筛选设计 273
“筛选设计”窗口 274
响应 275
因子 277
选择筛选类型 279
从部分析因设计列表中选择 279
设计类型 280

分辨率衡量混杂程度 282
显示和修改设计 283
主效应筛选设计 285
生成设计 286
设计 287
设计评估 287
输出选项 287
制表 288
筛选设计选项 289
筛选设计的其他示例 291
在部分析因设计中修改生成规则 292
Plackett-Burman 设计 297

10 “拟合两水平筛选”平台 303
分析筛选实验中的数据
“拟合两水平筛选”平台概述 305
“拟合两水平筛选”平台的示例 305
启动“拟合两水平筛选”平台 307
筛选报表 308
对比 309
半正态图 309
构建或运行模型 309
其他拟合两水平筛选分析示例 310
分析 Plackett-Burman 设计 310
效应遗传示例 311
分析超饱和设计 313
“拟合两水平筛选”平台的技术细节 314
效应的进入顺序 315
拟合两水平筛选为正交旋转 315
Lenth 伪标准误差 315
Lenth t 比 316

11 响应曲面设计 319

响应曲面设计概述 321

响应曲面设计的示例 323
构造 Box-Behnken 设计 324
分析实验数据 325
探索最优设置 327
“响应曲面设计”窗口 330
响应 330
因子 332
选择设计 334
指定输出选项 335
制表 336
响应曲面设计选项 337

12 完全析因设计 339

完全析因设计概述 341
完全析因设计的示例 341
构造设计 341
分析实验数据 343
构建完全析因设计 347
响应 347
因子 349
选择输出选项 351
制表 352
完全析因设计选项 353

13 混料设计 355

配方设计

混料设计概述 357
混料设计窗口 357
响应 357
因子 359
三元图概述 360
混料设计类型的示例 361
最优混料设计 361
单纯形重心设计 363

单纯形格点设计 365
极端顶点设计 366
ABCD 设计 370
空间填充设计 370
拟合混料设计 373
整体模型检验和方差分析报表 374
理解响应曲面报表 374
化学混料示例 374
创建设计 374
分析混料模型 377
预测刻画器 377
混料刻画器 378

14 田口设计 381
含有噪声因子的设计
田口设计概述 383
田口设计的示例 383
构建田口设计 387
响应 387
因子 388
选择田口内表和外表设计 388
显示编码设计 388
制作设计表 389
田口设计选项 389

15 评估设计 393
探索设计的属性
“评估设计”平台概述 395
评估设计示例 395
评估丢失试验的影响 396
针对指定模型计算功效 403
“评估设计”启动窗口 405
“评估设计”窗口 405
因子 406
模型 406

别名项 407
设计 408
设计评估 408
功效分析 408
预测方差刻画 413
设计空间比例图 415
预测方差曲面 416
估计效率 417
别名矩阵 418
相关性色图 419
设计诊断 420
“评估设计”选项 422

16 比较设计 425
同时比较和评估设计
“比较设计”平台概述 427
比较设计的示例 427
试验规模相同的设计 427
试验规模不同的设计 432
具有不同整区数的裂区设计 435
启动“比较设计”平台 438
“比较设计”窗口：指定模型和别名项 440
参考设计 441
因子 441
模型 441
别名项 441
“比较设计”窗口：设计评估 441
功效分析 442
预测方差刻画 445
设计空间比例图 446
相对估计效率 447
别名矩阵汇总 449
绝对相关性 450
设计诊断 452
比较设计选项 454

17 前瞻样本大小与功效 455

启动“样本大小与功效”平台 458
单样本均值计算器 459
“单样本均值”计算器的示例 459
“单样本均值”计算器字段 463
“单样本均值”计算器的统计详细信息 464
“双样本均值”计算器 465
“双样本均值”计算器的示例 465
“双样本均值”计算器字段 466
“双样本均值”计算器的统计详细信息 467
“k 个样本均值”计算器 467
“k 个样本均值”计算器的示例 468
“k 个样本均值”计算器字段 469
“k 个样本均值”计算器的统计详细信息 470
“单样本标准差”计算器 471
“单样本标准差”计算器的示例 471
“单样本标准差”计算器字段 472
“单样本标准差”计算器的统计详细信息 473
“单样本比例”计算器 474
“单样本比例”计算器的示例 474
“单样本比例”计算器字段 475
单比例计算器的统计详细信息 476
“双样本比例”计算器 477
“双样本比例”计算器的示例 477
“双样本比例”计算器字段 478
双比例计算器的统计详细信息 479
“每单位计数”计算器 479
“每单位计数”计算器的示例 480
“每单位计数”计算器字段 480
“每单位计数”计算器的统计详细信息 481
“Sigma 质量水平”计算器 482
“Sigma 质量水平”计算器的示例 482
计算缺陷数的示例 482

“Sigma 质量水平”计算器字段 483
Sigma 质量计算器的统计详细信息 483
“可靠性检验计划”计算器 484
“可靠性检验计划”计算器的示例 484
“可靠性检验计划”计算器字段 485
“可靠性检验计划”计算器的统计详细信息 487
“可靠性验证”计算器 487
“可靠性验证”计算器的示例 487
“可靠性验证”计算器字段 489
“可靠性验证”计算器的统计详细信息 489

18 选择设计 491

创建用于选择偏好产品特征的设计

选择设计概述 493
选择设计术语 493
Bayesian D 最优性 493
选择设计示例 494
带有分析的选择设计示例 496
创建用于试行研究的选择设计 496
分析试行研究数据 498
使用先验信息设计最终选择实验 500
运行设计和分析结果 501
“选择设计”窗口 503
特性 504
模型 505
生成设计 506
设计 507
制表 508
选择设计选项 508
技术细节 509
Bayes D 最优性和设计构造 509
效用平均和局部 D 最优设计 510

19 MaxDiff 设计 511
创建选择最优项和最差项的设计
“MaxDiff 设计”平台概述 513
MaxDiff 设计的示例 513
MaxDiff 设计启动窗口 516
MaxDiff 窗口 517
“设计选项”分级显示项 517
“设计”分级显示项 517
配对关联矩阵 517
制表 517
MaxDiff 选项 518

20 覆盖阵列 519
检测组件交互作用失败
覆盖阵列概述 521
不含因子水平限制的覆盖阵列的示例 522
创建设计 523
分析实验数据 525
包含因子水平限制的覆盖阵列的示例 526
创建设计 526
分析实验数据 531
“覆盖阵列”窗口 532
因子 532
限制因子水平组合 533
设计 535
量度 538
输出选项 538
“覆盖阵列”数据表 538
覆盖阵列选项 540
覆盖阵列的技术细节 540
用于优化的算法 541
用于量度的公式 541

21 空间填充设计 543

空间填充设计概述 545
构建空间填充设计 545
响应 546
因子 547
定义因子约束 549
空间填充设计方法 552
设计 552
设计诊断 552
设计表 553
空间填充设计选项 553
球堆积设计 554
创建球堆积设计 555
直观演示球堆积设计 556
拉丁超立方设计 557
创建拉丁超立方设计 558
直观演示拉丁超立方设计 559
均匀设计 561
比较球堆积、拉丁超立方和均匀方法 563
最低位势设计 564
最大熵设计 566
高斯过程 IMSE 最优设计 568
快速灵活填充设计 569
FFF 最优性准则 569
为 FFF 设计设置平均聚类大小 570
约束 570
创建和查看受约束的快速灵活填充设计 571
为地图形状创建空间填充设计 572
球堆积设计的示例 576
为钻孔数据创建球堆积设计 577
确定性数据分析准则 578

22 加速寿命试验设计 581

为加速寿命试验设计实验

加速寿命试验设计概述 583
加速寿命试验设计的示例 583
扩充加速寿命试验设计的示例 590
构建加速寿命试验设计 594
- 指定设计结构 595
- 指定加速因子 595
- 指定设计详细信息 596
- 检查和更新 ALT 计划规格 598
- 创建和评估最优 599
- 创建设计表 600

加速寿命试验选项 601
“ALT 设计”平台的统计详细信息 602
- ALT 设计的失效分布 602
- R 精度因子（95% 置信区间） 603
- 加速寿命试验的最优性准则 604

23 非线性设计 607

非线性设计概述 609
非线性设计的示例 610
- 创建无先验数据的非线性设计 610
- 使用先验数据扩充设计 614
- 创建二项响应的设计 618

非线性设计启动窗口 621
“非线性设计”窗口 621
- 因子 622
- 参数 622
- 生成设计 622
- 设计 623
- 制表或扩充表 623

非线性设计选项 623
非线性设计的统计详细信息 624

非线性模型 624
最优性准则的径向球积分 624
查找最优设计 625

24 平衡不完整区组设计 627

不完整区组设计的示例 629
构建平衡不完整区组设计 631
指定处理、区组和制作设计 631
验证设计并制表 632
平衡不完整区组设计选项 632
平衡不完整区组设计的统计详细信息 633

25 MSA 设计 635
创建测量系统分析实验设计

MSA 设计概述 637
MSA 设计示例 637
构建 MSA 设计 639
响应 639
因子 641
MSA 设计 642
MSA 设计诊断 642
MSA 设计表 642
MSA 设计选项 643
“MSA 设计”平台的统计详细信息 644

26 组正交超饱和设计 645
创建组正交超饱和设计

组正交超饱和设计概述 647
组正交超饱和设计的示例 647
构建组正交超饱和设计 648
指定因子、结构和制作设计 649
验证设计并制表 649
组正交超饱和设计选项 650
组正交超饱和设计的统计详细信息 651

27 “拟合组正交超饱和”平台 653

分析组正交超饱和设计中的数据

“拟合组正交超饱和”平台概述 655

拟合组正交超饱和的示例 655

启动“拟合组正交超饱和”平台 657

“拟合组正交超饱和”报表 658

“拟合组正交超饱和”平台选项 659

28 样本大小分析器 661

样本大小与功效计算器

“样本大小分析器”平台概述 663

样本大小分析器示例 663

启动“样本大小分析器” 664

假设检验功效分析器 664

单样本均值的功效 665

单样本比例的功效 667

单样本方差的功效 669

单样本等价性的功效 670

两个独立样本均值的功效 672

两个独立样本比例的功效 675

两个独立样本方差的功效 677

两个独立样本等价性的功效 679

方差分析的功效 681

置信区间计算器 682

单样本均值的误差幅度 683

单样本比例的误差幅度 685

单样本方差的误差幅度 686

两个独立样本均值的误差幅度 687

两个独立样本比例的误差幅度 689

两个独立样本方差的误差幅度 691

“可靠性验证”计算器 693

参数可靠性验证 693

非参数可靠性验证 694

样本大小分析器中的已保存设置 695

A 列属性 697
了解实验设计分配的列属性
添加和查看列属性 699
响应限 700
响应限示例 701
编辑响应限 704
检测限值 706
函数响应 706
设计角色 706
设计角色示例 707
编码 709
低值和高值 709
“编码”列属性和中心多项式 710
编码示例 710
分配编码 713
混料 715
混料示例 716
因子更改 720
因子更改示例 721
值顺序 723
值顺序示例 724
指定值顺序 726
值标签 727
值标签示例 727
每区组试验次数 729
每区组试验次数示例 729
约束状态 730
约束状态示例 730
B 技术细节 733
别名矩阵 735
具有“困难”或“极难”因子更改的设计 735
具有“若可能”效应的设计 735

功效计算 736
单个参数的功效 736
分类效应的功效 737
相对预测方差 738
C 参考书目 739
D 技术许可声明 745

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 25
JMP 帮助 26
JMP 文档库 26
学习 JMP 的其他资源 32
JMP 教程 32
样本数据表 32
学习统计和 JSL 术语 33
学习 JMP 的提示与技巧 33
JMP 工具提示 33
JMP 用户社区 33
免费在线统计思维课程 33
JMP 新用户欢迎套件 34
统计学知识门户 34
JMP 培训 34
用户编写的 JMP 手册 34
“JMP 起始页”窗口 34
JMP 技术支持 34

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 26 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

实验设计介绍
“实验设计”平台概述

JMP“实验设计”平台可协助您设计、评估和分析实验。大多数平台都侧重于构造设计，其余平台则用来支持设计工作。本节为“实验设计”菜单下的每个平台提供了简要的概述。

设计构造平台

定制设计 构造适合众多设置的设计。与完全基于经典设计的方法相比，定制设计往往更经济和灵活。

定制设计可适应各种类型的因子、约束和不被允许的组合。在试验次数给定后，您可以指定哪些效应是必须要估计的，哪些是期望估计的。还可以指定符合您实验情形预算的试验次数。定制设计还支持难以更改的因子和极难更改的因子，从而允许您构造裂区及相关设计。

“定制设计”平台可构造许多特殊的设计类型：

- 筛选
- 响应曲面
- 混料
- 随机区组
- 裂区
- 裂-裂区
- 双因子裂区

您可以使用其他平台构造经典筛选设计、响应曲面设计和混料设计。不过，“定制设计”平台可赋予您其他平台所不具备的灵活性。只能使用“定制设计”平台来构造适用于裂区情形的设计。

确定性筛选设计 针对连续因子和两水平的分类因子构造筛选设计。若您怀疑存在活跃的交互作用或曲率，则确定性筛选设计会十分有用。确定性筛选设计支持您标识强非线性效应的来源，同时避免最高达到二阶的任何效应之间的完全混杂。

“确定性筛选设计”最适用于包含四个或更多因子的实验。确定性筛选设计支持将试验分组到区组中。区组数是用户指定的。

筛选设计 为具有任意水平数的连续、离散数值和分类因子构造筛选设计。存在标准设计时，您有两个选项：

- 从经典筛选设计的列表中进行选择。这些设计允许使用两水平连续因子，或是两水平或三水平分类或离散连续因子。

– 为主效应生成正交或接近正交的设计。**接近正交设计**允许分类和离散数值因子包含任意数量的水平，也允许两水平连续因子。这些设计侧重于在交互作用可忽略不计的情况下估计主效应。

对于许多筛选情形，标准设计都不可用。在这些情形下，您可以构造接近正交筛选设计。

响应曲面设计 构造对连续因子的二次函数建模的设计。要拟合二次效应，响应曲面设计要求每个因子有三个设置。 JMP 可为至多八个因子提供响应曲面设计。

您可以从中心复合设计或 Box-Behnken 设计的列表中进行选择。在适用的情况下，以正交方式分区组的中心复合设计包含在该列表内。支持对中心复合设计进行多种修改。

完全析因设计 为任意数量的连续或分类因子（均可包含任意数量的水平）构造完全析因设计。完全析因设计对因子设置的每种组合都进行一次试验。完全析因设计的规模往往较大。试验次数等于因子水平个数的乘积。

混料设计 构造在因子为混料成分时使用的设计。在混料实验中，一种成分比例的变化要求其余的一种或多种成分随之变化以保持总和不变。可从若干设计类型（包括一些经典混料设计方法）中进行选择：最优、单纯形重心、单纯形格点、极端顶点、 ABCD 和空间填充。对于最优、极端顶点和空间填充混料设计，您可以指定线性不等式约束以限制设计空间。

田口数组 构造可用于信噪比分析的设计。这种设计基于田口内表和外表方法。控制因子设置构成内表，噪声因子设置构成外表。均值和信噪比是关注的响应。

若不使用田口数组，也可以通过构造定制设计来包含控制因子、噪声因子和噪声控制交互作用。此类设计称为**组合数组**，该数组通常比田口数组更经济且包含信息更多。

选择设计 构造您可用来比较前瞻产品的设计。选择设计中的因子是产品特性。该设计按照两个、三个或四个一组来安排产品特征（即各种特性的组合）。该实验会让响应者指明他们倾向于一对特征中的哪一个特征。您可以生成反映有关产品特性的先验信息的选择设计。

MaxDiff 构造一个由选择集构成的设计，在 MaxDiff 研究中，可将这些选择集提供给响应者。响应者仅从一组较小的选择集内报告最偏好和最不偏好的选项。这会强制响应者按照偏好对选项排名，这样生成的排名往往比使用标准偏好尺度获得的排名更具有确定性。

JMP PRO **覆盖阵列** 构造您可以用来测试软件、网络和其他系统的组合设计。强度为 *t* 的覆盖阵列具有以下属性：每 *t* 个因子的每种水平组合都会出现在至少一个试验中。覆盖阵列允许使用任意数量的分类因子，每个分类因子可含有任意数量的水平。可以指定不被允许的组合。

空间填充设计 构造的设计适用于关注的系统为确定性或接近确定性系统的情形。标准应用涉及为高度复杂的确定性计算机模拟模型创建较为简单的代理模型。

确定性系统中不存在变异。其目标是要最小化拟合模型与真正模型之间的差异 （偏倚）。空间填充设计尝试通过让设计点的分布尽量远离彼此或是在设计区域中均匀分布各点来满足该目标。

JMP 提供七种空间填充设计方法。快速灵活填充设计是其中一种设计方法，它可适应包含任意数量水平的分类因子并支持线性约束。

加速寿命试验设计 构造和扩充的设计可用于检验旨在加速失效时间的极端条件下的产品。使用实验结果来预测正常操作条件下的可靠性。

寿命分布可以为对数正态分布或 Weibull 分布。设计中可包含一个或两个加速因子。若有两个加速因子，您可以选择包含其交互作用。您可以为加速模型参数指定先验分布。提供 D 最优设计和两类 I 最优设计。

非线性设计 构造和扩充的设计可用于拟合参数为非线性的模型。您可以使用现有数据的模型拟合的估计值构造设计。若没有基于模型的估计值，则还可以通过应用先验知识来构造设计。

平衡不完整区组设计 构造用于试验 b 个区组中的 a 个处理的设计，此时只能在任一个区组中运行 k 个处理 （且 $k < a$）

组正交超饱和设计 构造超饱合筛选设计。这些设计适用于待调查因子数大于可行试验次数情况下的早期工作。组正交超饱和设计是一类特殊的二水平超饱和设计，该设计具有模型选择所需的属性。

支持平台

扩充设计 向现有设计添加试验，目的是使生成的设计最优。扩充设计支持您以迭代方式进行实验。您可以重复设计、添加中心点、创建折叠设计、添加轴点、添加点以创建空间填充设计，或是以指定的试验次数扩充设计。您可以将试验分组到区组中，以便将原始试验与扩充试验区别开来。您可以添加原始模型中不存在的模型效应，并为这些效应指定要求。

拟合确定性筛选设计 使用称为 DSD 的有效模型选择的方法分析确定性筛选设计。该方法利用确定性筛选设计的特殊结构。

拟合组正交超饱和 分析组正交超饱和设计。该分析方法利用组正交超饱和设计的组正交结构。

评估设计 为现有实验设计提供诊断。“评估设计”平台为您提供多种方法来评估设计的优点和局限性。该平台可与任何数据表一同使用，不仅仅是使用 JMP 创建的设计。

提供了以下若干诊断：

- 功效分析
- 预测方差图
- 参数的估计效率
- 别名矩阵，显示模型效应的偏倚结构
- 色图，显示效应间的绝对相关性
- 设计效率值

比较设计 将多达四个设计与参考设计进行比较。用于探索、评估和比较设计性能。诊断显示各个设计相对于彼此的性能如何，以及在绝对意义上的性能如何。

样本大小与功效 为以下多种检验情形提供样本大小与功效计算：一个或多个样本均值、标准差、一个或两个比例、每单位计数（Poisson 均值），以及 sigma 质量水平。对于这些选项，

您可以通过指定三个量中的两个量来计算第三个量。这三个量分别是您要检测的差值、样本大小和功效。若您仅提供这三个值中的一个，系统会提供另外两个值之间的关系图。

您可以计算可靠性检验计划（目标是估计失效概率）所需的样本大小。还可以计算可靠性验证 （目标是证明某个产品符合或超过指定的标准）所需的样本大小。

第 3 章

从实验设计入手
示例和关键概念

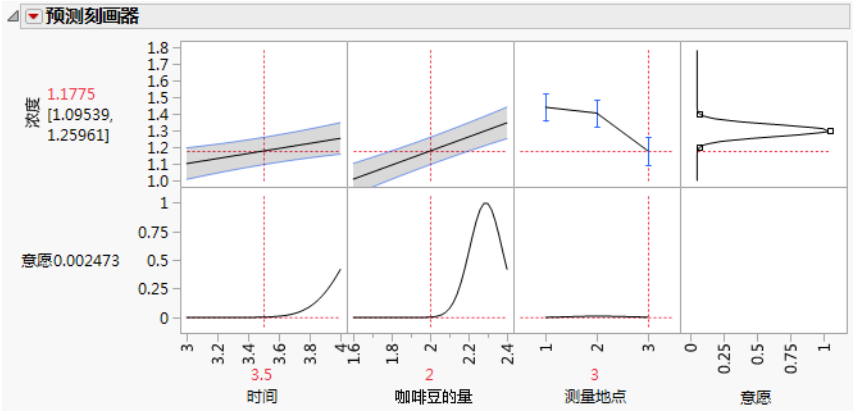
设计实验是一组受控检验，设计这些检验的目的是：对因子与一个或多个响应之间的关系进行建模和探索。JMP 包含多种工具，可帮助您创建适用于您的特定情形的高效实验设计。具体而言，提供了以下几类设计：

- “定制设计”平台专为您特有的情形定制设计。该平台构建的设计可适应任意类型、任意数量的因子以及难以更改的因子（裂区情形）。您可以控制试验次数。
- “确定性筛选设计”平台构造一类创新型筛选设计，其中的主效应彼此之间没有别名关系，且主效应与双因子交互作用之间也没有别名关系。这些设计还允许估计二次项。
- “筛选设计”、“响应曲面设计”、“完全析因设计”、“混料设计”和“田口数组”平台构造的是传统实验设计。
- “覆盖阵列”、“选择设计”、“空间填充设计”、“加速寿命试验设计”和“非线性设计”平台构造的是专业设计类型。

“评估设计”和“扩充设计”平台提供用于评估和扩充现有设计的工具。“样本大小与功效”平台处理特殊化情形的样本大小与功效计算。

本章通过一个示例演示 JMP 的实验设计方法。本章还讨论实验设计框架、用于支持设计创建的工作流程，以及实验设计的基本原则。

图 3.1 刻画器图示例



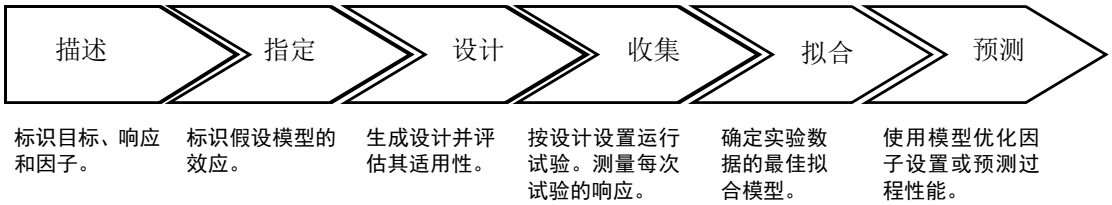
目录

实验设计和实验设计工作流程概述 41
咖啡浓度实验 42
定义研究和目标 42
创建设计 44
运行实验 49
分析数据 50
实验设计工作流程：描述、指定、设计 56
定义响应和因子 57
指定模型 57
生成设计 58
评估设计 59
制表 60
实验设计的原则和方针 60
效应等级 60
效应遗传 60
效应稀疏 61
中心点、重复试验和检验 61

实验设计和实验设计工作流程概述

下面的六步框架提供了用来设计实验、运行试验和分析结果的结构。适当的工程设计与流程知识对于理解所有这些步骤至关重要。

图 3.2 实验设计框架



您可以在实验设计平台中执行前三步。最终结果是一种能够在您的工作环境中运行的设计。有关这三步工作流程的详细说明，请参见第 56 页的 “实验设计工作流程：描述、指定、设计”。

描述 确定实验目标。标识响应和因子。

您的目标可以是标识活跃因子、查找最优因子设置或构建预测模型。

指定 确定或指定您认为能够充分描述实际情形的假设模型。

您的**假设模型**是一个初始模型，理想情况下包含要评估的所有效应。在某些平台中，您可以明确构建关注的模型。在另一些平台中，模型隐含在您所做的选择中。例如，在 “筛选设计” 平台中，您可以选择具有给定分辨率的模型。设计的分辨率确定哪些效应混杂。效应混杂很可能导致不能明确哪个效应是真正活跃的。

设计 生成与假设模型一致的设计。评估该设计以了解其优点和局限性，并确保它可以按照您的模型和目标提供您所需的信息。

设计生成平台中的 “设计评估” 或 “设计诊断” 分级显示项可让您深入了解关于设计的属性。

下一步进入数据收集阶段，该阶段将在受控条件下运行实验。

收集 执行每次试验并记录响应值。

运行实验之后，借助生成的数据表中的脚本，您可以使用 “拟合模型” 和 “筛选” 之类的平台来拟合模型。根据您的目标，该模型可帮助您标识活跃效应或找到最优设置。

拟合 用实验数据来拟合假设模型。

使用 JMP 建模平台拟合并微调模型。在某些情形下，您可能需要扩充设计并执行额外试验来解决模型的不确定性。

预测 使用经过微调的模型来实现实验目标。

确定哪些效应是活跃的、找到可以优化响应的因子水平，或构建预测模型。

设计的实验通常接着用来构造流程知识。设计策略往往从筛选设计入手，用来缩小可能活跃因子的范围。随后要在设计中研究已经确定的因子，这些设计侧重于更好地理解交互作用和二次效应。有时，需要扩充设计来消除与引起效应的因子相关的不确定性。本节概述的步骤涉及如何执行和分析单个实验。不过，您可能需要通过一系列实验来实现目标。

第 42 页的“咖啡浓度实验”中的示例明确演示了实验设计工作流程中的步骤，还说明了如何使用数据表脚本来分析您的实验数据。《实验设计指南》中的许多示例既演示了支持好设计的工作流程，又提供了对所研究的实验数据的分析。

咖啡浓度实验

- “定义研究和目标”
- “创建设计”
- “运行实验”
- “分析数据”

定义研究和目标

您在您当地的一家中等规模的咖啡焙烘厂工作。您需要处理分别冲泡的 12 盎司杯的咖啡的浓度。您的目标是确定哪些因子对咖啡浓度有影响，并找出这些因子的最优设置。

响应

响应是咖啡的**浓度**。该浓度是使用折射计测量的总溶解固体量。咖啡是使用单杯咖啡滤杯冲泡的，并且是在从咖啡粉中析出液体之后五分钟测量的。

以往的研究表明浓度读数为 1.3 最为理想，而浓度在 1.2 和 1.4 之间也是可接受的。

因子

该研究中共标识了四个因子：**研磨**、**温度**、**时间**和**咖啡豆的量**。在工作区中的三个测量地点冲泡咖啡。为解释因冲泡位置引起的变异，**测量地点**作为分区组因子包含在本研究中。下面说明了这些因子：

- **研磨**是指研磨的粗细程度。**研磨**设定为两个水平：Medium（中等）和 Coarse（粗糙）。
- **温度**是指水浇注到咖啡粉上的前一刻测量的水温（华氏度）。**温度**设定为 195 华氏度和 205 华氏度。
- **时间**是以分钟计的冲泡时间。**时间**设定为 3 或 4 分钟。
- **咖啡豆的量**是放入锥形过滤器的咖啡量，用每盎司水冲泡的咖啡豆克数来测量。**咖啡豆的量**设定为 1.6 和 2.4。
- **测量地点**是冲泡咖啡的位置。三个测量地点分别标记为 1、2 和 3。

表 3.1 汇总了有关这些因子及其设置的信息。还可以在 Design Experiment 文件夹中的 Coffee Factors.jmp 样本数据表中找到这些因子和水平。

表 3.1 咖啡实验的因子和设置范围

因子	角色	设置范围
研磨	分类	Medium（中 等），Coarse（粗糙）
温度	连续	195 - 205
时间	连续	3 - 4
咖啡豆的量	连续	1.6 - 2.4
测量地点	分区组	1，2，3

请注意以下几点：

- **研磨**是包含两个水平的分类因子。
- **温度**、**时间**和**咖啡豆的量**是连续因子。
- **测量地点**是包含三个水平的分区组因子。

所有因子都可变化，可在每次试验中重新设置。该实验没有难以更改的因子。

运行咖啡实验所用的器具显示在图 3.3 中。这是三个冲泡测量地点之一的设置。另外两个测量地点也使用同类型的器具。

图 3.3 咖啡实验器具

试验次数

根据可用的资源和允许的时间，您决定总共可进行 12 次试验。由于共有三个测量地点，每个测量地点可进行 4 次试验。

创建设计

按照 “实验设计工作流程：描述、指定、设计”中概括的设计工作流程中的步骤创建设计：

- “定义响应和因子”
- “指定模型”
- “生成设计”
- “评估设计”
- “制表”

定义响应和因子

在出现的第一个分级显示项中，输入有关响应和因子的信息。

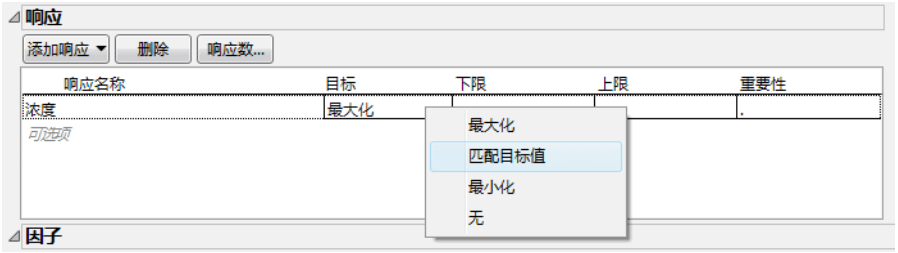
响应

1. 选择**实验设计** > **定制设计**。
2. 双击 “响应名称”下的 Y 并键入**浓度**。

请注意，默认 “目标”为**最大化**。您的目标是找到实现咖啡冲泡目标浓度为 1.3 （介于 1.2 到 1.4 的限值之内）的因子设置。

3. 点击默认 “目标”**最大化**并将其更改为**匹配目标值**。

图 3.4 将 “匹配目标值”选作 “目标”



4. 在**下限**下点击并键入 1.2。
5. 在**上限**下点击并键入 1.4。
6. 将**重要性**下面的区域留空。

   由于只有一个响应，默认为该响应的 “重要性”指定 1。

完成的“响应”分级显示项如图 3.5 所示。

因子

手动输入因子，或通过包含因子和设置的预存表输入因子。若要设计的是一个新实验，您必须首先手动输入因子。一旦使用“保存因子”选项将因子保存至数据表，即可使用保存的表来加载这些因子。

对于本例，您可以在两个选项中任选其一。请参见第 45 页的“手动输入因子”或第 46 页的“使用加载因子输入因子”。

手动输入因子

1. 点击**添加因子 > 分类 > 2 水平**。
2. 键入**研磨**，代替默认名称 **X1**。

   请注意，“角色”按要求设置为 “分类”。“更改”属性默认设置为 “容易”，指示可以为每次试验重置**研磨**设置。
3. 点击默认值 L1 和 L2，将其分别更改为 “Coarse”和 “Medium”。
4. 在**添加因子**数旁边键入 3。然后点击**添加因子 > 连续**。
5. 键入因子名称和值，代替默认条目：
   - **温度**（195 和 205）
   - **时间**（3 和 4）
   - **咖啡豆的量**（1.6 和 2.4）

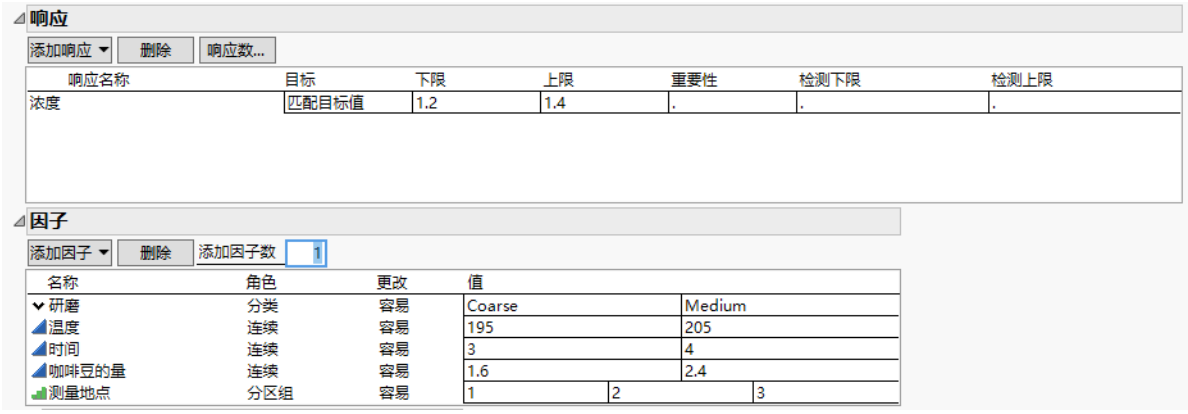
6. 点击**添加因子 > 分区组 > 每个区组 4 次试验**。

   回想一下您的试验预算允许进行 12 次试验。您想要在三个测量地点之间平均分配试验次数。

7. 键入**测量地点**，代替默认名称 X5。

   请注意，“角色”设置为“分区组”，并且仅显示“值”的一个设置。这是因为在尚未指定所需的试验次数之前 JMP 无法确定区组数。一旦在“生成设计”分级显示项中指定了“试验次数”， JMP 会将**测量地点**水平数更新为所需数目。

图 3.5 已完成的“响应”和“因子”分级显示项



8. 点击**继续**。

   以下分级显示项将添加到“定制设计”窗口：

   - 定义因子约束（本例未使用）
   - 模型
   - 别名项
   - 生成设计

使用加载因子输入因子

使用包含因子信息的表输入因子：

1. 点击“定制设计”红色小三角并选择**加载因子**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Coffee Factors.jmp。

   加载因子之后，“定制设计”窗口将更新以显示更多分级显示项：

   - 定义因子约束（本例未使用）
   - 模型
   - 别名项
   - 生成设计

定义因子约束

一旦手动输入因子并点击了“继续”，或是一旦从因子表加载了因子，即显示“定义因子约束”分级显示项。添加因子约束（若有）是“定义响应和因子”步骤中的一部分。由于本设计对因子设置没有任何约束，请不要更改该分级显示项。

指定模型

“模型”分级显示项

图 3.7 显示了“模型”分级显示项。在“模型”分级显示项中，您可以指定假设模型，其中包含您要估计的效应。请参见第 41 页的“指定”。显示的列表默认将所有主效应显示为**必需**，指示该设计能够估计所有主效应。由于您此时主要关注的是因子的主效应，所以您不需要向“模型”分级显示项添加任何效应。

图 3.6 仅包含主效应的 “模型”分级显示项

用于复制结果的步骤（可选）

由于 “定制设计”算法是从随机起始设计开始的，所以您的设计可能与图 3.8 中显示的设计有所不同。要获得包含完全相同试验数的设计，请在生成设计之前设置随机种子和开始数：

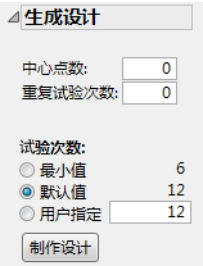
1. 点击“定制设计”红色小三角并选择**设置随机种子**。
2. 键入 569534903。
3. 点击**确定**。
4. 点击 “定制设计”红色小三角并选择**开始数**。
5. 键入 100。
6. 点击**确定**。

注意：设置“随机种子”和“开始数”将精确重现本例中显示的设计。但是，设计表中的行顺序可能不同。自行构造设计时，这些步骤不是必需的。

生成设计

在“生成设计”分级显示项中，您可以输入有关设计的结构和大小的其他详细信息。“默认值”设计显示为包含 12 次试验。回想一下您的设计预算允许进行 12 次试验（第 44 页的“试验次数”）。

图 3.7 “生成设计”分级显示项



1. 点击**制作设计**。

   “设计”和“设计评估”分级显示项将添加到“定制设计”窗口中。“输出选项”面板也会显示出来。

   “设计”分级显示项显示设计（图 3.8）。若您未按第 47 页的“用于复制结果的步骤（可选）”中所述设置随机种子和开始数，您的设计可能不同于图 3.8 中的设计。这是因为用于生成设计的算法是从随机起始设计开始的。

图 3.8 咖啡实验设计

设计

试验	研磨	温度	时间	咖啡豆的量	测量地点
1	Coarse	195	3	1.6	1
2	Medium	195	4	2.4	2
3	Medium	195	4	1.6	3
4	Medium	205	4	2.4	1
5	Coarse	205	3	2.4	2
6	Medium	205	3	2.4	3
7	Medium	205	3	1.6	1
8	Coarse	205	4	1.6	2
9	Coarse	205	4	1.6	3
10	Coarse	195	4	2.4	1
11	Medium	195	3	1.6	2
12	Coarse	195	3	2.4	3

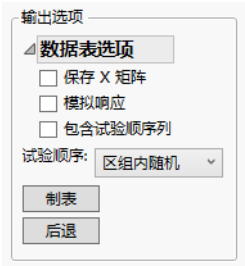
评估设计

“设计评估”分级显示项提供多种评估设计的方法。这是一个重要的主题，但为简便起见，本例的上下文中并未涉及。请参见第 393 页的“评估设计”一章。

制表

使用“输出选项”面板指定数据表中的试验顺序。默认选择**区组内随机**即为合适的选项。这种选择将以随机顺序为各个**测量地点**安排各次试验。

图 3.9 输出选项



1. 点击**制表**。

   图 3.10 中显示的数据表随即打开。请记住，若您未按第 47 页的“用于复制结果的步骤（可选）”中所述设置随机种子和开始数，您的设计可能有所不同。您的设计表代表备择最优设计。

图 3.10 定制设计表

定制设计
设计 定制设计
准则 D 最优
模型
评估设计
广义回归
实验设计对话框

列 (6/0)
研磨 *
温度 *
时间 *
咖啡豆的量 *
测量地点 *
浓度 *

	研磨	温度	时间	咖啡豆的量	测量地点	浓度
1	Coarse	195	4	2.4	1	•
2	Medium	205	3	1.6	1	•
3	Medium	205	4	2.4	1	•
4	Coarse	195	3	1.6	1	•
5	Medium	195	4	2.4	2	•
6	Medium	195	3	1.6	2	•
7	Coarse	205	4	1.6	2	•
8	Coarse	205	3	2.4	2	•
9	Medium	195	4	1.6	3	•
10	Medium	205	3	2.4	3	•
11	Coarse	205	4	1.6	3	•
12	Coarse	195	3	2.4	3	•

请注意“列”面板中因子和响应右侧的星号。这些星号指示列属性已保存至数据表中的列。这些列属性用在数据分析中。请参见第 83 页的“因子”和第 88 页的“因子列属性”。

运行实验

现在可以开始执行实验了。在每个**测量地点**，按照设计表中显示的顺序执行四次试验。在不同试验中重置设备和材料。例如，若先后两次试验要求水温为 195 度，则在分别两次将 12 盎司杯中的水加热至 195 度之后，需要让加热容器冷却下来后再加热。**浓度**测量值将被记录下来。

您针对**浓度**的设计和实验结果包含在 Design Experiment 文件夹的 Coffee Data.jmp 样本数据表中（图 3.11）。

图 3.11 带有浓度结果的咖啡设计

	研磨	温度	时间	咖啡豆的量	测量地点	浓度
1	Medi...	205	4	2.4	1	1.78
2	Coarse	195	3	1.6	1	1.25
3	Medi...	205	3	1.6	1	1.10
4	Coarse	195	4	2.4	1	1.63
5	Coarse	205	4	1.6	2	1.26
6	Medi...	195	4	2.4	2	1.63
7	Medi...	195	3	1.6	2	1.22
8	Coarse	205	3	2.4	2	1.51
9	Coarse	205	4	1.6	3	1.07
10	Coarse	195	3	2.4	3	1.26
11	Medi...	195	4	1.6	3	1.13
12	Medi...	205	3	2.4	3	1.25

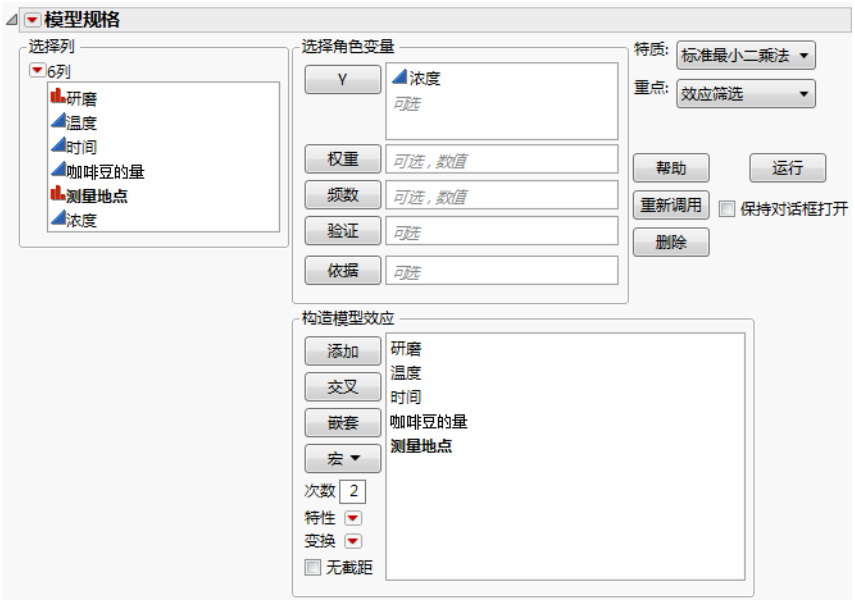
分析数据

“定制设计”平台通过将“模型”脚本保存至该平台所创建的设计表，为数据分析提供便利（图 3.10）。在执行实验并输入数据后运行该脚本。使用该脚本可打开“拟合模型”窗口，其中包含您在“定制设计”窗口的“模型”分级显示项中指定的效应。

拟合模型

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Coffee Data.jmp。
   在“表”面板中，请注意“定制设计”所创建的“模型”脚本。
2. 点击“模型”脚本旁边的绿色小三角。
   “模型规格”窗口显示您在“模型”分级显示项中指定的效应。

图 3.12 “模型规格”窗口

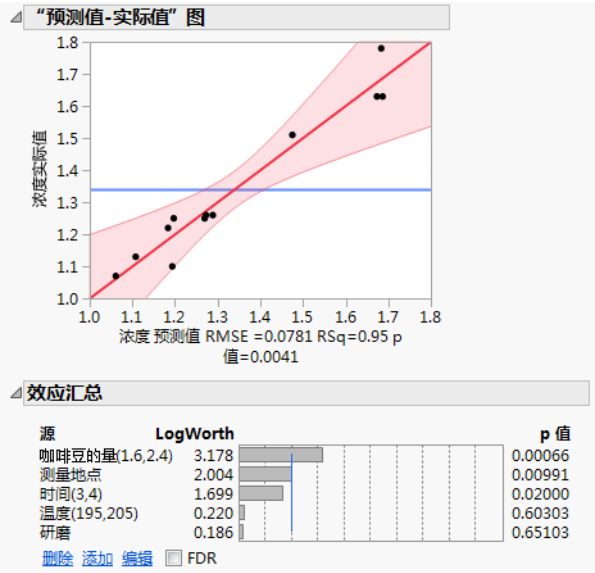


3. 选中**保持对话框打开**选项。
4. 点击**运行**。

分析模型

“效应汇总”和“‘预测值-实际值’图”报表提供有关模型的高级别信息。

图 3.13 完整模型的 “效应汇总” 和 “‘预测值 - 实际值’ 图”



请注意以下几点：

- “‘预测值-实际值’ 图” 显示没有证据表明失拟。
- 正如 “‘预测值-实际值’ 图” 指明的那样，该模型是显著的。图下方显示的符号 p = 0.0041 给出了总体模型检验的显著性水平。
- “效应汇总” 报表显示**咖啡豆的量**、**测量地点**和**时间**在 0.05 水平下显著。
- “效应汇总” 报表还显示**温度**和**研磨**不显著。

简化模型

由于**温度**和**研磨**看起来不活跃，这两个效应对模型贡献随机噪声。去除这些效应后重新拟合模型，以获取与活跃效应有关的模型参数的更精确估计值。

1. 在 “模型规格” 窗口的 “构造模型效应” 列表中选择**温度**和**研磨**。
2. 点击**删除**。
3. 确认模型**重点**设置为**效应筛选**。

   “效应筛选” 重点菜单提供了有助于分析实验设计的报表 （如 “预测刻画器”）。
4. 点击**运行**。

图 3.14 简化模型的部分输出

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	1.3408333	0.020047	66.88	<.0001*
时间(3,4)	0.0758333	0.020047	3.78	0.0069*
咖啡豆的量(1.6,2.4)	0.1691667	0.020047	8.44	<.0001*
测量地点[1]	0.0991667	0.028351	3.50	0.0100*
测量地点[2]	0.0641667	0.028351	2.26	0.0580

效应检验

源	参数数目	自由度	平方和	F 比	概率>F
时间(3,4)	1	1	0.06900833	14.3093	0.0069*
咖啡豆的量(1.6,2.4)	1	1	0.34340833	71.2078	<.0001*
测量地点	2	2	0.16251667	16.8494	0.0021*

效应详细信息

统一尺度估计值

名义型因子 - 扩展至所有水平

项	统一尺度估计值	标准误差	t 比	概率>|t|
截距	1.3408333	0.020047	66.88	<.0001*
时间(3,4)	0.0758333	0.020047	3.78	0.0069*
咖啡豆的量(1.6,2.4)	0.1691667	0.020047	8.44	<.0001*
测量地点[1]	0.0991667	0.028351	3.50	0.0100*
测量地点[2]	0.0641667	0.028351	2.26	0.0580
测量地点[3]	-0.163333	0.028351	-5.76	0.0007*

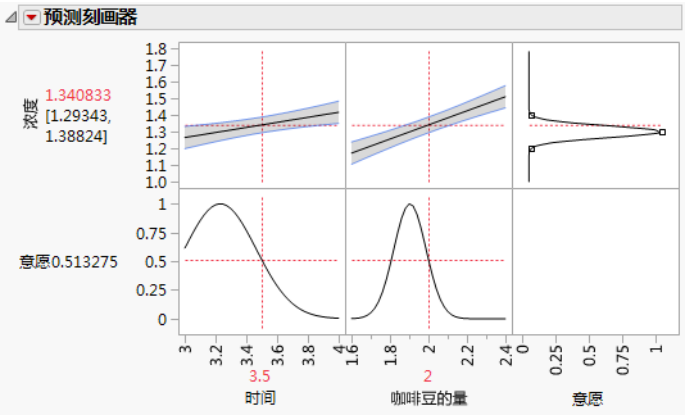
请注意以下几点：

- “效应检验”报表显示这三个效应都仍保持显著。
- “统一尺度估计值”报表进一步指明测量地点 [1] 和测量地点 [3] 的均值与**浓度**的平均响应显著不同。
- 请注意，“参数估计值”报表中出现的“估计值”与“统一尺度估计值”报表中的对等项完全相同。这是因为效应经过了编码。请参见“列属性”附录中第 709 页的“编码”。
- 测量地点 [3] 效应的估计值仅显示在“统一尺度估计值”报表中，其中名义型因子经过扩展，可显示其所有水平的估计值。
- “参数估计值”报表给出了模型系数的估计值，其中的模型是以编码效应的形式指定的。

探索模型

“预测刻画器”显示在报表底部。

图 3.15 预测刻画器



回想一下在设计实验时，您设置的响应目标为“匹配目标值”，限值为 1.2 和 1.4。JMP 使用该信息构造意愿函数以反映您的指定。请参见第 83 页的“因子”。

请注意图 3.15 中的以下事项：

- 图形顶部行中的前两个图显示，在给定其中一个因子设置的情况下，**浓度**如何随另一个因子发生变化。例如，当**咖啡豆的量**为 2 时，**时间**图中对应的线条显示**浓度**预测值如何随**时间**变化。
- 图顶部行左侧的值给出了选定因子设置下的**浓度**（红色）预测值以及**浓度**均值的置信区间。
- 顶部行最右侧的图显示**浓度**的意愿函数。意愿函数指示目标为 1.3 最理想。意愿随着您远离该目标而降低。意愿在限值 1.2 和 1.4 处接近 0。
- 底部行中的图显示每个因子在另一个因子设置处的意愿迹线。
- 图底部行左侧的值给出了选定因子设置所对应的响应值的意愿。

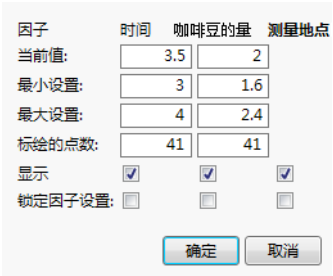
通过在**时间**和**咖啡豆的量**对应的列中拖动红色的垂直虚线，可以探索各种因子设置。由于模型中没有交互作用，刻画器指示增加**咖啡豆的量**即增加**浓度**。此外，**浓度**看起来对于**咖啡豆的量**的更改较之**时间**的更改更为敏感。

由于**测量地点**是一个分区组因子，它不会显示在 “预测刻画器”中。不过，您可能想要了解**浓度**预测值如何随**测量地点**变化。在 “预测刻画器”中加入 “测量地点”：

1. 点击“预测刻画器”红色小三角并选择**重置因子网格**。

   随即显示包含**时间**、**咖啡豆的量**和**测量地点**列的 “因子设置”窗口。在**测量地点**下，请注意与**显示**对应的框并未选中。这指示**测量地点**不显示在 “预测刻画器”中。
2. 选中**测量地点**下面与**显示**相对应行中的框。
3. 取消选中**测量地点**下面与**锁定因子设置**相对应行中的框。

图 3.16 “因子设置”窗口



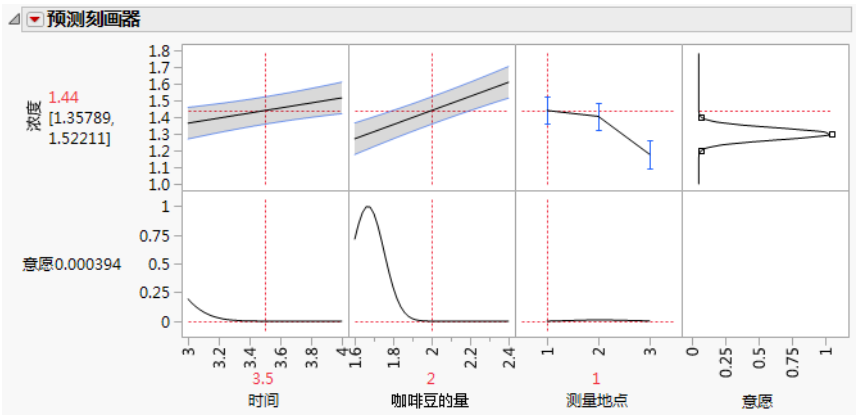
4. 点击**确定**。

   测量地点对应的图显示在 “预测刻画器”中。

5. 点击**测量地点**上方两个图中的任一个，插入一条红色垂直虚线。

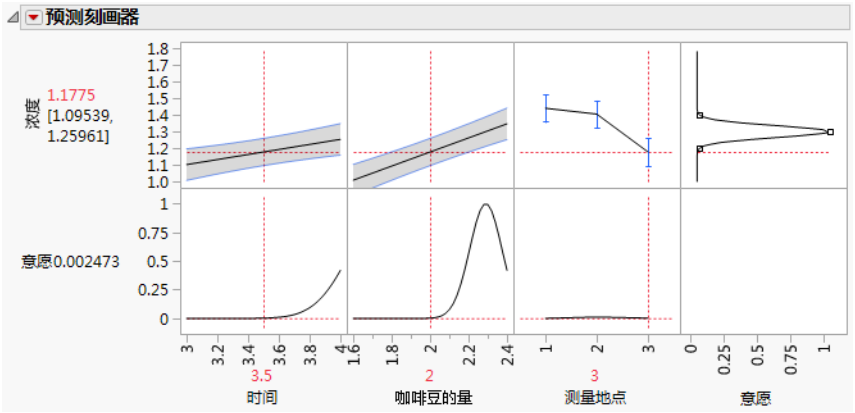
6. 将红色垂直虚线移至**测量地点** 1。

图 3.17 显示测量地点 1 的结果的预测刻画器



7. 将红色垂直虚线移至**测量地点** 3。

图 3.18 显示测量地点 3 的结果的预测刻画器



对于**测量地点** 1，位于设计区域中间的**浓度**预测值为 1.44。对于**测量地点** 3，**浓度**预测值约为 1.18。差异的量值指示您需要处理**测量地点**变异性。更好地控制**测量地点**变异应能使**浓度**更加一致。一旦使各个**测量地点**保持一致，您就可以确定**时间**和**咖啡豆的量**的共同最优设置。

构造咖啡实验设计所使用的过程遵从实验设计工作流程中的步骤。下一节详细描述实验设计工作流程。

实验设计工作流程：描述、指定、设计

实验设计平台由一系列表示工作流程的步骤构成，该工作流程是设计实验所必经的流程。完成每个步骤后，需要点击**继续**移至下一步。本节所述的要素对于九个实验设计平台是通用的。以下是本节要讨论的平台：

- 定制设计
- 确定性筛选设计
- 筛选设计
- 响应曲面设计
- 完全析因设计
- 混料设计
- JMP PRO 覆盖阵列
- 空间填充设计
- 田口数组

三个专用平台差别较大，它们分别是：选择设计、加速寿命试验设计和非线性设计。本节不对这三个平台加以讨论。

本节说明实验设计工作流程中的步骤，其中还会讨论如何在不同设计平台中实施这些步骤。

定义响应和因子

在实验设计框架的 “描述” 步骤中：

- 您需要标识关注的响应和因子。
- 需要确定实验目标。您是要让响应最大化，还是要实现特定目标？目标是什么？还是您只想确定哪些因子对响应有影响？
- 您需要确定描述您的实验范围或设计空间的因子设置。

多数 JMP 实验设计平台会在打开时显示分级显示项，您可在其中列出您的响应和因子。“响应”分级显示项在不同平台中是通用的。您需要在此插入您的响应和其他信息，如响应目标、下限、上限和重要性。

“因子”分级显示项在不同平台中各有不同。这是为了适应每个平台所处理的不同类型因子和特定设计情形的需要。在某些平台中，一旦输入响应和因子，点击“继续”后即显示“定义因子约束”分级显示项。在该分级显示项中，您可以限制可用于设计的因子值。

图 3.19 显示了使用“定制设计”平台构造在 Box Corrosion Split-Plot.jmp 样本数据表中构造设计所用到的“响应”和“因子”分级显示项。该数据表位于 Design Experiment 文件夹中。同时还显示了“定义因子约束”分级显示项，一点击“继续”即显示该分级显示项。“定义因子约束”分级显示项支持您指定因子设置必须满足的限制。

图 3.19 箱体腐蚀裂区实验的响应和因子

指定模型

一旦完成填写“响应”和“因子”分级显示项，即可点击**继续**按钮。您将进入下一个设计构造阶段，在该阶段中您需要显式或隐式选择一个假设模型。

“定制设计”平台支持您显式指定您要拟合的模型，这样生成的设计对该模型而言是最优的。其他设计平台则不允许显式指定模型。例如，在筛选平台中，有一个选项支持您从完全析因、部分析因和 Plackett-Burman 设计列表中进行选择。这些设计中的别名关系隐式定义了您可以拟合的模型。

在“定制设计”中，填写完“响应”和“因子”后点击“继续”，即可看到“模型”分级显示项。图 3.20 显示了 Box Corrosion Split-Plot.jmp 样本数据表中使用的设计的示例。假设模型要求**炉温**和**涂层**主效应及其交互作用都是可估计的。生成的设计可保证这些效应的可估计性。

在其他多数平台中，点击“继续”将为您提供一组可供选择的设计；但在“完全析因”中，点击“继续”会直接将您引至“输出选项”，因为完成填写“因子”分级显示项后设计就已经确定了。

图 3.20 箱体腐蚀裂区实验的 “模型”分级显示项

生成设计

多数实验设计平台都会让您在一定程度上控制最终设计的规模。在“定制设计”中，您可以指定试验次数，并在适当时指定中心点数和重复试验次数。在其他平台中，您可以有众多灵活选择。通常您可以指定中心点数、重复试验次数或设计的重复次数。

一旦指定了与试验次数相关的选项，即可点击“制作设计”。“实验设计”窗口随之更新，以在“设计”分级显示项中显示设计。

图 3.21 显示了针对 Box Corrosion Split-Plot.jmp 实验的定制设计（共计 24 次试验）的“设计”分级显示项。由于**炉温**的**更改**指定为**困难**，所以构建了**整区**因子，以表示**炉温**的随机区组设置。

图 3.21 箱体腐蚀裂区实验的 “设计” 分级显示项

设计

试验	整区	炉温	涂层
1	1	370	C2
2	1	370	C3
3	1	370	C1
4	1	370	C4
5	2	370	C3
6	2	370	C4
7	2	370	C2
8	2	370	C1
9	3	360	C3
10	3	360	C2
11	3	360	C4
12	3	360	C1
13	4	380	C4
14	4	380	C3
15	4	380	C2
16	4	380	C1
17	5	380	C4
18	5	380	C2
19	5	380	C1
20	5	380	C3
21	6	360	C3
22	6	360	C1
23	6	360	C4
24	6	360	C2

注意：对于定制设计，一旦您创建了设计，“模型” 和 “别名” 分级显示项中的控件即被禁用。使用 “后退” 按钮可更改模型或别名项。

评估设计

在多数平台中，若点击 “制作设计”，“设计评估” 分级显示项随即显示。通过该分级显示项，您可以在以下方面探索您创建的设计：检测效应的能力、预测方差、估计效率、别名关系、效应之间的相关性，以及其他设计效率测度。“定制设计” 的 “设计评估” 分级显示项如图 3.22 所示：“设计评估” 在第 393 页的 “评估设计” 一章中介绍。

某些平台适用其他类型的设计诊断。例如，“空间填充设计” 提供的 “设计诊断” 分级显示项包含专用于空间填充设计的量度。“覆盖阵列” 提供的 “量度” 分级显示项包含专用于覆盖的测度。

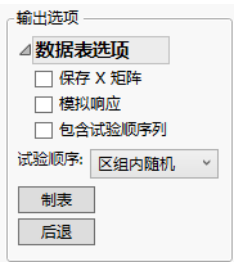
图 3.22 “定制设计” 中的 “设计评估” 分级显示项

制表

多数平台都提供“输出选项”节点或面板。根据具体的平台和设计，您可以使用“输出选项”面板指定其他设计结构。例如，您可以指定试验次数、中心点、重复次数，或者您希望设计试验以哪种顺序显示在生成的数据表中。

图 3.23 中显示的“输出选项”面板针对 Wine Data.jmp 样本数据表中的实验，该数据表位于 Design Experiment 文件夹中。在本例中，您可以选择各种“试验顺序”选项来构造设计数据表。或者，您可以选择“后退”重建设计。

图 3.23 葡萄酒实验的 “输出选项”面板



实验设计的原则和方针

实验设计与实验数据分析需要遵从特定的原则。效应等级、效应遗传和效应稀疏原则主要与模型选择相关。这些原则可帮助您在搜索最佳模型的过程中缩小可能的模型范围。请参见 Hamada and Wu (1992)、 Wu and Hamada (2009) 以及 Goos and Jones (2011)。

- “效应等级”
- “效应遗传”
- “效应稀疏”
- “中心点、重复试验和检验”

效应等级

在回归建模中，**效应等级**原则主张主（一阶）效应趋向于解释响应中最大的变异量。二阶效应（即：交互作用效应和二次项）解释的变异量紧接其后。再后是更高阶的项（按等级顺序）。

以下是建模的含义：主效应更有可能比二阶效应重要；二阶效应更有可能比三阶效应重要；依此类推到更高阶的项。

效应遗传

效应遗传原则指的是在模型中应该加入高阶效应对应的低阶成分。该原则来源于观测上的证据，即：具有较小主效应的因子往往不会有显著的交互作用效应。

强效应遗传要求将模型效应的所有低阶成分都包含在模型中。假定模型中有一个三因子交互作用 (ABC)，那么其所有成分主效应和双因子交互作用（A、B、C、AB、AC、BC）也都必须包含在模型中。

弱效应遗传要求仅在模型中包含模型效应的一个序列的低阶成分。若模型中有一个三因子交互作用，那么该模型必须包含涉及到的某一个因子以及涉及该因子的某一个双因子交互作用。假定模型中有一个三因子交互作用 (ABC)，那么，若 B 和 BC 也包含在模型中，则该模型满足弱效应遗传。

对于连续因子，效应遗传可确保模型对于因子位置和尺度的变化保持不变。

效应稀疏

效应稀疏原则断定响应中的多数变异由相对少量的效应来解释。筛选设计要研究许多效应，所以该设计非常依赖效应稀疏原则。经验表明，筛选设计中使用的试验次数应该至少两倍于可能显著的效应的数量。

中心点、重复试验和检验

有些实验设计平台支持您向设计添加中心点（对于连续因子）、重复试验或完整重复的设计。以下是与添加设计点相关的一些背景介绍。

添加中心点

连续因子的中心点支持您检验因非线性效应引起的失拟，检验失拟可帮助您确定误差方差估计值是否因缺失模型项而膨胀，所以这可能是在试验上的一项明智投资。

您可以只在中心点重复试验，也可以重复其他设计试验。JMP 使用重复试验来构造独立于模型的误差估计值（纯误差估计值）。这个纯误差估计值支持您检验失拟。

请注意，中心点无法帮助您获取关于模型效应的更精确的估计值。这些点支持您检验是否存在曲率，但并不确定是哪个非线性效应导致出现曲率。

要标识曲率来源，您必须设置连续因子至少有三个水平。确定性筛选设计是三水平设计，该设计能够检测和标识对响应造成强非线性效应的所有因子。请参见第 7 章，“确定性筛选设计”。

添加重复试验

若您的试验预算允许，您可以重复试验或是在设计空间内以最佳方式分配新试验。添加重复试验可提高某些估计值的精确性，并改善失拟检验的能力。不过，对于给定试验次数的预算，使用重复试验通常会降低设计估计模型效应的能力。您无法通过在设计空间内以最佳方式分布试验来估计尽可能多的项。

失拟检验

在构造设计的实验时，通常要求试验次数尽量少并且与实验目标保持一致。但若是试验次数过少，则只能检测极大的效应。例如，对于给定的效应，*t* 检验统计量是响应均值的变化与响应

均值标准误差之比。若只有一个误差自由度 (df)，则检验临界值将超过 12。所以，要让这种近乎饱和的设计检测某个效应，该临界值必须极大。

您观察到失拟检验也有类似情况发生。该检验检测失拟的能力取决于分子与分母中的自由度。若每种仅有 1 个自由度，则您需要 *F* 值超过 150 才能声明在 0.05 水平下显著。若每种有 2 个自由度，则 *F* 值必须超过 19。为了使检验在第二种情况下显著，失拟均方必须是纯误差均方的 19 倍。此外，失拟检验对离群值也很敏感。

有关“失拟”检验的详细信息，请参见《拟合线性模型》。

确定试验次数

在工业应用中，每次试验的成本往往都极高，所以有必要将试验次数降至最低。要估计所关注的固定效应，您只需要与模型中项数一样多的试验次数。要确定效应是否活跃，您需要对误差方差进行合理估计。除非您已对该方差进行了合理估计，否则请考虑在估计模型项所需的试验次数基础上至少添加 4 次试验。

第 4 章

定制设计

构造符合您需要的设计

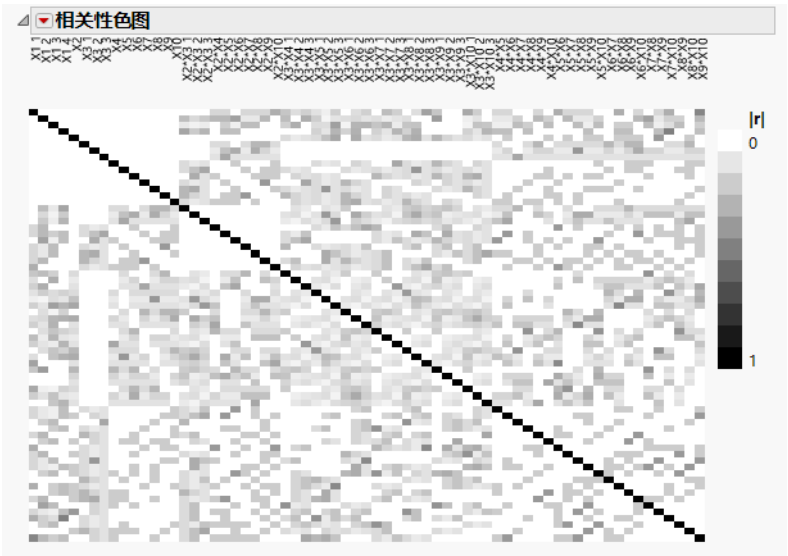
使用“定制设计”平台可构造为您特定的实验设置量身定制的最优设计。通常，定制设计比使用其他方法获得的设计更为经济高效。您可以执行下列任务：

- 输入多种不同类型的因子。
- 指定设计空间约束。
- 在试验次数给定后，如果可能的话，指出哪些效应是必须要估计的，哪些是期望估计的。
- 指定符合您预算的试验次数。

“定制设计”平台可以构造包括以下特殊设计类型在内的众多设计：筛选、响应曲面、混料、随机区组、裂区、裂-裂区和双因子裂区。

本章包含关于如何使用“定制设计”平台的详细示例，随后介绍关于该平台的信息。另见第 121 页的“定制设计示例”一章。

图 4.1 绝对相关性色图



目录

定制设计概述 65
定制设计示例 66
构建定制设计 81
响应 82
因子 83
定义因子约束 88
别名项 93
生成设计 94
设计 95
设计评估 95
输出选项 96
定制设计选项 98
模拟响应 101
保存 X 矩阵 103
开始数 105
设计搜索时间 105
定制设计的技术细节 106
包含随机化限制的设计 106
包含难以更改的水平的协变量 113
整区和子区数 114
最优性准则 114
D 效率 118
坐标交换算法 118

定制设计概述

使用“定制设计”平台可构造为您特定的实验需要量身定制的最优设计。

您可以包括众多因子类型，具体如下：

- 连续
- 离散数值（包含任意水平数）
- 分类（包含任意水平数）
- 分区组（每个区组包含指定数量的试验）
- 协变量
- 混料
- 常数
- 不受控

指定可操作区域

您可以使用线性因子约束或不被允许的组合来限定您的实验区域，以反映您的操作条件。特别是可以为分类、连续和离散数值因子指定限制。请参见第 88 页的“定义因子约束”。

指定包含难以更改的水平的因子

对于连续、离散数值、分类和混料因子，您可以指出两个级别的难以更改的因子。这些难度级别可通过整区或整区及裂区来表示。您还可以指定难以更改的协变量。

指定主要关注的效应

您可以明确指定**假设**模型。您的假设模型是一个初始模型，理想情况下包含要评估的所有效应。模型中可包含主效应、交互作用、响应曲面效应和多项式效应（最高五阶）的任意组合。您可以指定哪些效应是必须估计的，哪些效应是期望估计的。根据试验次数，“定制设计”使用 Bayesian 最优性方法来估计哪些效应是期望估计的。请参见第 91 页的“模型”。

指定试验次数

“定制设计”平台支持您指定符合您实验情形预算的试验次数。该平台指示用于估计所需效应的最小试验次数，并提供默认试验次数。这些值可用作指导信息，帮助确定可行的试验次数。请参见第 94 页的“生成设计”。

构造适合的设计类型

“定制设计”可构造众多设计类型。这些类型包括经典设计和随机区组设计。有关不同设计类型的示例，请参见第 121 页的“定制设计示例”一章。

构造最优设计

根据您的特定要求，“定制设计”平台可构造最优设计。算法支持若干最优性准则：

- D 最优性
- I 最优性
- Bayesian D 和 I 最优性（使用“若可能”效应）
- A 最优性
- 别名最优性

请参见第 114 页的“最优性准则”。

设计是使用坐标交换算法构造的 (Meyer and Nachtsheim, 1995)。请参见第 118 页的“坐标交换算法”。

定制设计示例

利用定制设计器设计一个实验，调查葡萄酒加工因子对葡萄酒口感的影响。您的雇主种植了两种黑皮诺葡萄，可采用不同方式加工这两种葡萄。您的目标就是确定哪些因子影响黑皮诺葡萄酒的口感。在加工葡萄之前，您需要创建实验设计。加工之后，葡萄酒样本将陈酿 12 个月，然后过滤并装瓶。届时，专业品酒师会对葡萄酒样本的质量进行评级。

响应

酒庄多数产品都会出售给五大葡萄酒分销商。您需要安排每个分销商派一名专业品酒师对葡萄酒样本的质量进行评级。为了让您可以研究的因子数最大化，您决定每名专家必须品尝八个不同的样本。这意味着您的设计需要包含 40 个葡萄酒样本或试验。

评级以 0 – 20 为尺度，其中 0 表示最差，20 表示最佳。**分级**，该变量包含专家的评级，是关注的响应。您想要确定将该响应最大化的葡萄酒相关因子。

分区组因子

分区组因子用于解释不一定直接关注的变异。若在某一因子水平上所取的观测要比在不同水平上的观测更加相似，此时分区组因子尤其有效。在您的实验中，某一位专家给出的评级很可能具有相似的特征，而与其他专家给出的评级会有差别。但是，您关注的是葡萄酒的哪些属性导致所有专家都给出高评级。

由于每个评级员都品尝了八种葡萄酒，**评级员**成为分区组因子，每个区组八次试验。对于该实验，只需要考虑这五个评级员。您并不想让更多的评级员参与进来。

过程因子

您已经确定该项研究涉及九个过程因子。这些因子包括葡萄品种、葡萄产地，以及其他七个与加工相关的因子。您可以使用这些因子的任意组合进行实验。同样，可以在实验过程中按需更

改这些因子。相对于本次实验，这些因子全部为“容易”更改。有关指定因子更改的信息，请参见第 86 页的“更改和随机区组”。

因子及其水平显示在表 4.1 中。请注意，所有这些因子都为分类因子。Sample Data 的 Design Experiment 文件夹的因子表 Wine Factors.jmp 中同样给出了这些因子及其水平。

要使用这些因子的所有可能组合进行实验，需要执行多达 $4 \times 2^8 = 1024$ 次试验。不过在本例中，只需 40 次试验就可以构建令人满意的设计。

表 4.1 葡萄酒品尝实验的过程因子和水平

因子	水平
品种	Bernard，Dijon
产地	1，2，3，4
除梗	No，Yes
酵母	Cultured，Wild
温度	High，Low
压榨	Hard，Soft
桶龄	New，2 Years
桶干燥处理	Air，Kiln
过滤	No，Yes

既然已经定义了实验目标、因子和响应，接下来就可以构建定制设计了。

添加响应

对于这一定制设计示例，添加响应、响应 “目标”以及 “下限”、“上限”和 “重要性”（若适用）。响应在此为**分级**。

1. 选择**实验设计 > 定制设计**。
2. 双击 “响应名称”下的 Y，然后键入**分级**。

   请注意，默认 “目标”为最大化。由于您想要最大化口味等级，所以不要更改该目标。
3. 在 “下限”下点击并键入 0。

   最小的满意等级为 0。
4. 在 “上限”下点击并键入 20。

   最大的满意等级为 20。
5. 将 “重要性”下面的区域留空。

由于只有一个响应，默认为该响应的 “重要性” 指定 1。

第 69 页上的图 4.2 显示完成的 “响应” 分级显示项。

手动或自动添加因子

对于这一定制设计示例，手动输入因子，或使用包含因子和设置的预存表自动输入因子。

- 若要设计的是一个新实验，您必须首先手动输入因子。请参见第 68 页的 “手动添加因子”。
- 一旦使用 “保存因子” 选项保存了因子，即可使用保存的表自动加载这些因子。请参见第 69 页的 “使用 “加载因子” 自动添加因子”。

两种方法都可以将这四个分级显示项添加至 “定制设计” 窗口：定义因子约束、模型、别名项和生成设计。

手动添加因子

1. 首先，添加分区组因子**评级员**。点击**添加因子 > 分区组 > 每个区组 8 次试验**。
2. 键入**评级员**，代替默认名称 X1。

   请注意，“角色” 设置为 “分区组”。还请注意，只为 “值” 显示了一个设置。这是因为在尚未指定所需的试验次数之前无法确定区组数。一旦在 “生成设计” 分级显示项中指定了 “试验次数”，**评级员**水平数即更新为所需数目。
3. 点击**添加因子 > 分类 > 2 水平**。
4. 键入**品种**，代替默认名称 X2。

   请注意，“角色” 已按要求设置为 “分类”，“更改” 默认设置为 “容易”。
5. 点击 L1 和 L2，将其分别改为 Bernard 和 Dijon。
6. 点击**添加因子 > 分类 > 4 水平**。
7. 键入**产地**，代替默认名称 X3。
8. 点击 L1、 L2、 L3、 L4，将其分别改为 1、 2、 3、 4。
9. 点击**添加因子 > 分类 > 2 水平**。
10. 键入**除梗**，代替默认名称 X4。
11. 点击 L1 和 L2，将其分别改为 No 和 Yes。
12. 在**添加因子数**旁边键入 6，然后点击**添加因子 > 分类 > 2 水平**。这将向设计中添加六个分类两水平因子。
13. 更改默认值因子名称和值：
    - **酵母**（Cultured 和 Wild）
    - **温度**（High 和 Low）
    - **压榨**（Hard 和 Soft）

- 桶龄（New 和 Two Years）
- 桶干燥处理（Air 和 Kiln）
- 过滤（No 和 Yes）

图 4.2 已完成的 “响应” 和 “因子” 分级显示项

14. 点击**继续**。

以下分级显示项将添加到 “定制设计” 窗口：

- 定义因子约束（本例未使用）
- 模型
- 别名项
- 生成设计

使用 “加载因子” 自动添加因子

使用包含因子信息的表输入因子：

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Wine Factors.jmp。
2. 点击 “定制设计” 红色小三角并选择**加载因子**。

加载因子之后，“定制设计” 窗口自动更新。以下分级显示项将添加到 “定制设计” 窗口：

- 定义因子约束（本例未使用）
- 模型
- 别名项

- 生成设计

定义模型

对于这一定制设计示例，“模型”分级显示项将所有主效应显示为“必需”，指示该设计需能够估计所有主效应。您的假设模型反映您仅关注主效应。但是，若您想要估计其他效应，则可以将其添加至“模型”分级显示项。有关模型的详细信息，请参见第 91 页的“模型”。

图 4.3 “模型”分级显示项

定义别名项

“别名项”分级显示项指定要显示在之后出现的“别名矩阵”中的效应。“别名矩阵”显示模型项与“别名项”分级显示项所列的效应之间的别名关系。打开“别名项”分级显示节点以验证是否已列出所有双因子交互作用。有关别名矩阵的详细信息，请参见第 73 页的“别名矩阵”。

图 4.4 “别名项”分级显示项的部分视图

设置随机种子以复制设计

“定制设计”算法以随机开始设计开头。要复制本例或教学设置中的设计，请设置用于定义开始设计的随机种子。若想获得包含与图 4.5 中所示完全相同的试验和试验顺序的设计，请设置随机种子和开始数：

1. 点击“定制设计”红色小三角并选择**设置随机种子**。
2. 键入 100526291 （随机种子）。
3. 点击**确定**。
4. 点击 “定制设计”红色小三角并选择**开始数**。
5. 键入 2。
6. 点击**确定**。

注意：设置“随机种子”和“开始数”将精确重现本例中显示的结果。自行构造设计时，这些步骤不是必需的。

生成设计

在“生成设计”分级显示项中，您可以输入有关设计的结构和大小的其他详细信息。在本例中，默认设计显示 16 次试验。但您有五位评级员，每一位可以抽取八种葡萄酒的样本。这意味着您的设计包含 40 次试验。更改设计试验次数：

1. 在**试验次数**下的**用户指定**框中键入 40。

   由于您不想重复试验，请保持**重复试验次数**设置为 0。
2. 点击**制作设计**。

“设计”和“设计评估”分级显示项将添加到“定制设计”窗口中。“输出选项”面板也会显示出来。

验证设计

“设计”分级显示项显示您已经构造的定制设计中的试验。您随后可以在“输出选项”中随机安排试验顺序。当前需要确认该设计是否适合您的实验。例如，确认每一位评级员（共五位）各评估八种葡萄酒，所有必需因子都显示出来，没有任何设置代表不可行的组合。

图 4.5 葡萄酒实验设计

设计

试验	评级员	品种	产地	除梗	酵母	温度	压榨	桶龄	桶干燥处理	过滤
1	1	Dijon	3	No	Wild	High	Soft	New	Kiln	Yes
2	2	Bernard	4	Yes	Cultured	Low	Hard	New	Kiln	No
3	3	Bernard	2	Yes	Wild	High	Hard	New	Air	No
4	4	Bernard	1	Yes	Wild	Low	Hard	2 Years	Air	Yes
5	5	Dijon	1	Yes	Wild	Low	Soft	New	Kiln	No
6	1	Dijon	2	Yes	Cultured	Low	Soft	2 Years	Air	Yes
7	2	Dijon	1	No	Cultured	High	Hard	New	Air	No
8	3	Dijon	4	No	Cultured	High	Soft	New	Air	Yes
9	4	Dijon	4	Yes	Wild	High	Hard	New	Kiln	Yes
10	5	Bernard	2	No	Wild	Low	Hard	New	Air	Yes
11	1	Bernard	3	Yes	Wild	High	Hard	New	Air	No
12	2	Dijon	1	Yes	Wild	High	Soft	2 Years	Air	Yes
13	3	Bernard	4	Yes	Wild	Low	Soft	2 Years	Kiln	No
14	4	Bernard	3	No	Cultured	Low	Soft	New	Air	No
15	5	Dijon	2	No	Cultured	High	Soft	2 Years	Kiln	No
16	1	Dijon	1	Yes	Cultured	Low	Hard	2 Years	Kiln	Yes
17	2	Bernard	2	No	Cultured	Low	Hard	2 Years	Air	No
18	3	Dijon	2	Yes	Cultured	High	Hard	New	Kiln	No
19	4	Bernard	1	No	Cultured	Low	Hard	New	Kiln	Yes
20	5	Bernard	3	Yes	Cultured	High	Hard	2 Years	Air	Yes
21	1	Bernard	4	No	Wild	High	Hard	2 Years	Kiln	No
22	2	Bernard	4	Yes	Cultured	High	Soft	2 Years	Kiln	Yes
23	3	Dijon	1	No	Wild	Low	Hard	2 Years	Kiln	No
24	4	Dijon	2	Yes	Wild	High	Soft	2 Years	Air	No
25	5	Bernard	1	No	Cultured	High	Soft	New	Kiln	No
26	1	Bernard	2	No	Wild	Low	Soft	New	Kiln	Yes
27	2	Dijon	3	No	Wild	High	Hard	New	Kiln	Yes
28	3	Bernard	1	No	Wild	High	Soft	2 Years	Air	Yes
29	4	Bernard	3	Yes	Cultured	High	Soft	2 Years	Kiln	No
30	5	Dijon	3	Yes	Wild	Low	Hard	2 Years	Kiln	No
31	1	Dijon	4	No	Cultured	Low	Hard	2 Years	Air	No
32	2	Bernard	2	Yes	Wild	Low	Soft	New	Kiln	Yes
33	3	Bernard	3	No	Cultured	Low	Soft	2 Years	Kiln	Yes
34	4	Dijon	2	No	Cultured	High	Hard	2 Years	Kiln	Yes
35	5	Dijon	4	Yes	Cultured	Low	Soft	New	Air	Yes
36	1	Bernard	1	Yes	Cultured	High	Soft	New	Air	No
37	2	Dijon	3	No	Wild	Low	Soft	2 Years	Air	No
38	3	Dijon	3	Yes	Cultured	Low	Hard	New	Air	Yes
39	4	Dijon	4	No	Wild	Low	Soft	New	Air	No
40	5	Bernard	4	No	Wild	High	Hard	2 Years	Air	Yes

评估设计

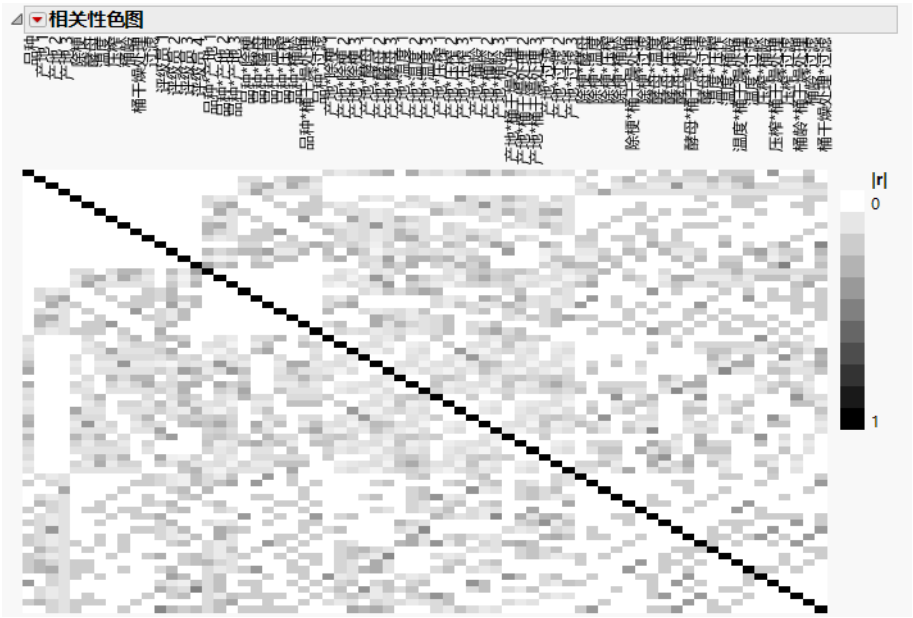
“设计评估”分级显示项提供多种评估定制设计的方法。对于本例，打开“设计评估”分级显示项，检查“相关性色图”、“别名矩阵”和“设计诊断”。

注意：有关“设计评估”分级显示项的详细信息，请参见第 393 页的“评估设计”一章。

相关性色图

“相关性色图”显示“模型”或“别名项”分级显示项中显示的任意两个效应之间的相关性绝对值。

图 4.6 相关性色图



主效应由色图左上角的 15 个项表示。白色表示的主效应与其他主效应之间的相关性为 0。这意味着所有主效应都是正交的，可以独立进行估计。

图 4.6 中唯一的黑色位于主对角线上。黑色指示绝对相关性为 1，它反映出每个项都与其自身完全相关。由此断定没有任何主效应与任何双因子交互作用完全混杂。事实上，主效应与双因子交互作用相关性的绝对值相当低。这意味着主效应估计值因为存在活跃的双因子交互作用而只产生轻微的偏倚。

提示：悬停在色图中的单元格上方可查看效应之间的绝对相关性。在图例下右击，将相关性保存至数据表。

别名矩阵

在“别名矩阵”中，模型效应列在左侧列中。对于给定的模型效应，列条目将指明列效应（若活跃）使模型效应估计值产生偏倚的程度。

图 4.7 别名矩阵的部分视图

别名矩阵

效应	品种*产地 1	品种*产地 2	品种*产地 3	品种*除梗	品种*酵母	品种*温度	品种*压榨	品种*桶龄
截距	0	0	0	0	0	0	0	0
品种	0	0	0	0	0	0	0	0
产地 1	0	0	0	0.141	0.141	0	-0.14	0.283
产地 2	0	0	0	0.082	-0.24	-0.33	0.082	0.327
产地 3	0	0	0	-0.12	0.346	0.115	-0.12	-0.12
除梗	0.141	0.082	-0.12	0	0	-0.2	0	0
酵母	0.141	-0.24	0.346	0	0	0	-0.2	0
温度	0	-0.33	0.115	-0.2	0	0	0	-0.2
压榨	-0.14	0.082	-0.12	0	-0.2	0	0	0
桶龄	0.283	0.327	-0.12	0	0	-0.2	0	0
桶干燥处理	0.283	0.163	0.115	0	-0.2	0.2	0.4	0
过滤	-0.14	-0.08	0.115	-0.2	0.2	0.2	0	0.2
评级员 1	0	0	0	-0.16	-0.16	0	-0.16	0.158
评级员 2	-0.26	0.149	-0.21	-0.27	0.274	-0.37	-0.09	-0.09
评级员 3	0.091	-0.05	0.075	0	-0.19	-0.06	-0.26	-0.26
评级员 4	0.354	-0.2	0.289	0	0.25	-0.25	0	0

例如，考虑模型效应**桶干燥处理**。若**品种*压榨**活跃，则**桶干燥处理**效应的期望估计值不同于该效应的无偏估计值。其差值是**品种*压榨**效应的 0.4 倍。因此，看起来显著的**桶干燥处理**估计效应实际上可能是显著的**品种*压榨**效应。

设计诊断

“设计诊断”分级显示项提供有关设计效率的信息。效率测度将您的设计与可能并不存在的理论最优设计加以对比。效率值是您设计的效率与该最优设计的效率之比，表示为百分比。有关效率测度的详细信息，请参见“评估设计”一章中第 417 页的“估计效率”。

图 4.8 “设计诊断”分级显示项

设计诊断

D 最优设计	
D 效率	100
G 效率	100
A 效率	100
预测平均方差	0.4
设计创建时间（秒）	0.05

请注意，D、G 和 A 效率值均为 100%。由于您的设计对主效应是正交的，所以相对于全部三种效率准则，该设计对主效应模型是最优的。

“设计诊断”分级显示项的第一行指示构造设计是为优化 D 效率准则。请参见第 98 页的“定制设计选项”中的“最优性准则”说明。在这种情况下，您的设计具有 100% 的 D 效率。

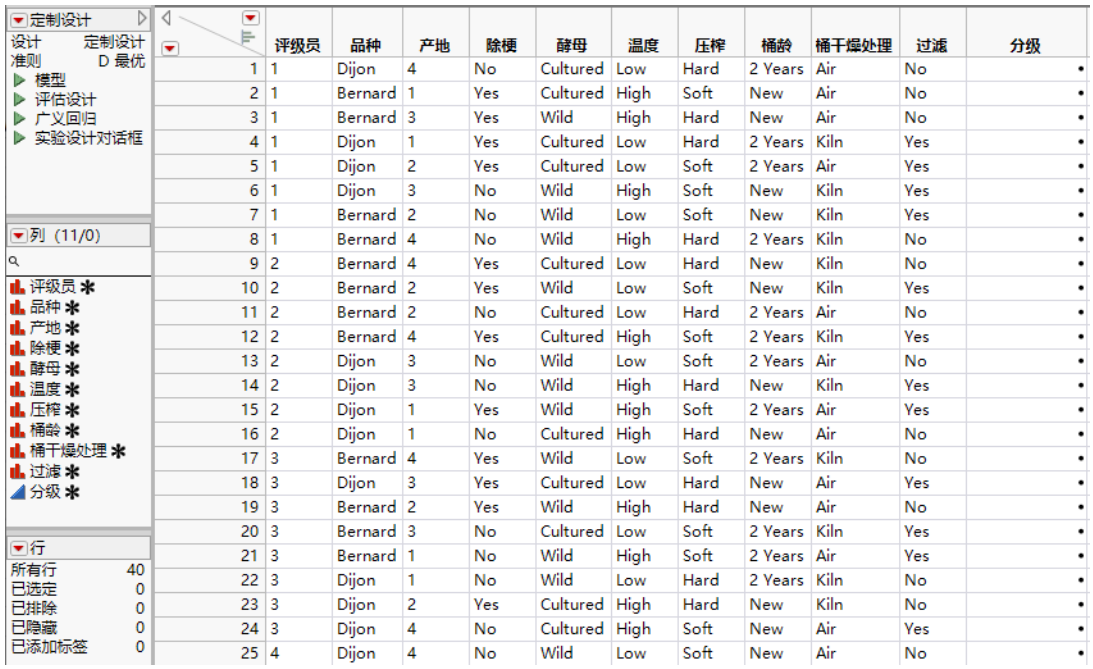
指定设计表选项

定制设计生成过程的最后步骤是为设计表指定选项并生成表。使用“输出选项”面板指定数据表中的试验顺序。默认选择**区组内随机**即为适合本例的选项。只需点击**制表**。

一个定制设计表随即创建并打开，该表类似于图 4.9 中的表。

注意：您的表可能看起来有所不同，因为创建它的算法使用随机开始设计。要获得图 4.9 中所示的精确表，请按第 71 页的“设置随机种子以复制设计”中所述设置随机种子和开始数。

图 4.9 定制设计表



	评级员	品种	产地	除梗	酵母	温度	压榨	桶龄	桶干燥处理	过滤	分级
1	1	Dijon	4	No	Cultured	Low	Hard	2 Years	Air	No	•
2	1	Bernard	1	Yes	Cultured	High	Soft	New	Air	No	•
3	1	Bernard	3	Yes	Wild	High	Hard	New	Air	No	•
4	1	Dijon	1	Yes	Cultured	Low	Hard	2 Years	Kiln	Yes	•
5	1	Dijon	2	Yes	Cultured	Low	Soft	2 Years	Air	Yes	•
6	1	Dijon	3	No	Wild	High	Soft	New	Kiln	Yes	•
7	1	Bernard	2	No	Wild	Low	Soft	New	Kiln	Yes	•
8	1	Bernard	4	No	Wild	High	Hard	2 Years	Kiln	No	•
9	2	Bernard	4	Yes	Cultured	Low	Hard	New	Kiln	No	•
10	2	Bernard	2	Yes	Wild	Low	Soft	New	Kiln	Yes	•
11	2	Bernard	2	No	Cultured	Low	Hard	2 Years	Air	No	•
12	2	Bernard	4	Yes	Cultured	High	Soft	2 Years	Kiln	Yes	•
13	2	Dijon	3	No	Wild	Low	Soft	2 Years	Air	No	•
14	2	Dijon	3	No	Wild	High	Hard	New	Kiln	Yes	•
15	2	Dijon	1	Yes	Wild	High	Soft	2 Years	Air	Yes	•
16	2	Dijon	1	No	Cultured	High	Hard	New	Air	No	•
17	3	Bernard	4	Yes	Wild	Low	Soft	2 Years	Kiln	No	•
18	3	Dijon	3	Yes	Cultured	Low	Hard	New	Air	Yes	•
19	3	Bernard	2	Yes	Wild	High	Hard	New	Air	No	•
20	3	Bernard	3	No	Cultured	Low	Soft	2 Years	Kiln	Yes	•
21	3	Bernard	1	No	Wild	High	Soft	2 Years	Air	Yes	•
22	3	Dijon	1	No	Wild	Low	Hard	2 Years	Kiln	No	•
23	3	Dijon	2	Yes	Cultured	High	Hard	New	Kiln	No	•
24	3	Dijon	4	No	Cultured	High	Soft	New	Air	Yes	•
25	4	Dijon	4	No	Wild	Low	Soft	New	Air	No	•

请注意以下几点：

- 在“表”面板中，设计创建过程中会添加“模型”脚本、“评估设计”脚本和“实验设计对话框”脚本。使用“模型”脚本可打开“拟合模型”窗口，其中包含您在“定制设计”对话框窗口中指定为“必需”的效应。“实验设计对话框”脚本可重新创建用于生成设计表的窗口。
- 在“列”面板中，因子和响应右侧的星号指示列属性已保存至数据表中的列。这些列属性用在数据分析中。有关列属性的详细信息，请参见第 83 页的“因子”和第 88 页的“因子列属性”。

分析定制设计

在运行实验和收集数据之后，您就可以分析结果了。使用“定制设计”表的**分级**列记录**分级**数据。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Wine Data.jmp。

Wine Data.jmp 表与图 4.9 中所示的定制设计表完全相同，只不过其中包含您记录的实验结果。

2. 在 “表”面板中，点击**模型**脚本旁边的绿色小三角。

图 4.10 葡萄酒实验的 “模型规格”窗口

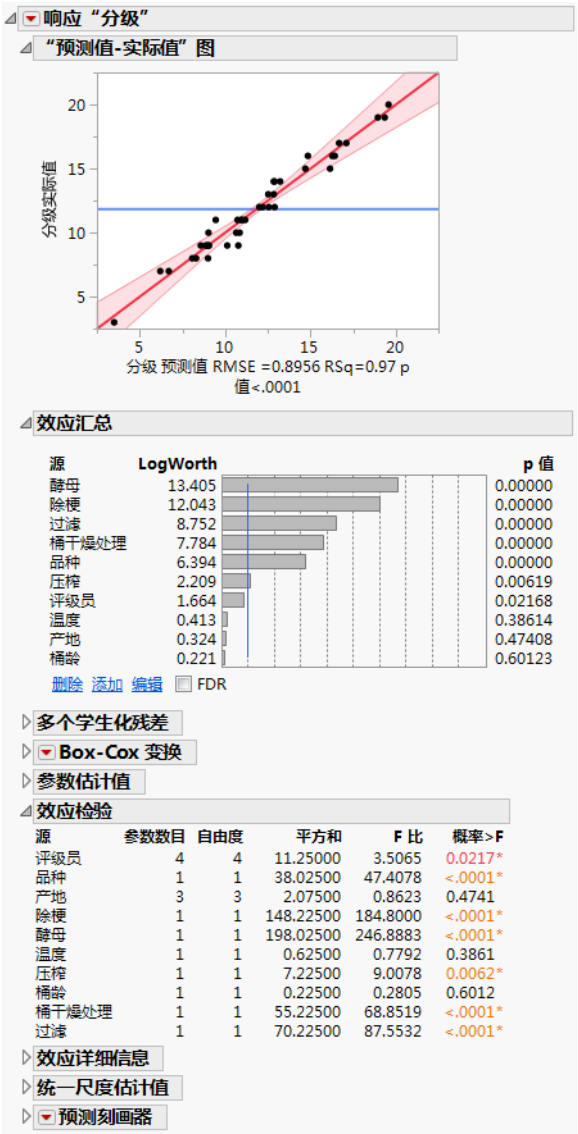
请注意，**评级员**分区组因子添加为固定效应，而不是随机区组效应。这样添加很合理，因为这五位评级员是专门挑选的，而不是从较大总体中随机抽取的样本。

3. 点击**运行**。

解释完整模型结果

来自定制设计示例的模型输出显示如下。

图 4.11 部分模型拟合结果



请注意以下几点：

- “‘预测值-实际值’图”显示没有明显证据表明失拟。
- 正如“‘预测值-实际值’图”和其下的 *p* 值指明的那样，该模型是显著的。
- “效应检验”报表指示模型项中有七项在 0.05 水平下显著。**产地**、**温度**和**桶龄**不显著。
- “效应汇总”报表按照显著性从高到低列出这些效应。较大的 LogWorth 值对应于较小的 p 值和较大的显著性。

简化模型

通过删除您确认为不活跃的效应来简化定制设计结果模型：

1. 在“效应汇总”报表中，按 Ctrl 键并选择**温度**、**产地**和**桶龄**。
2. 点击**删除**。

   报表随之更新，以显示删除这三个效应之后的模型拟合。

使用刻画器解释模型结果

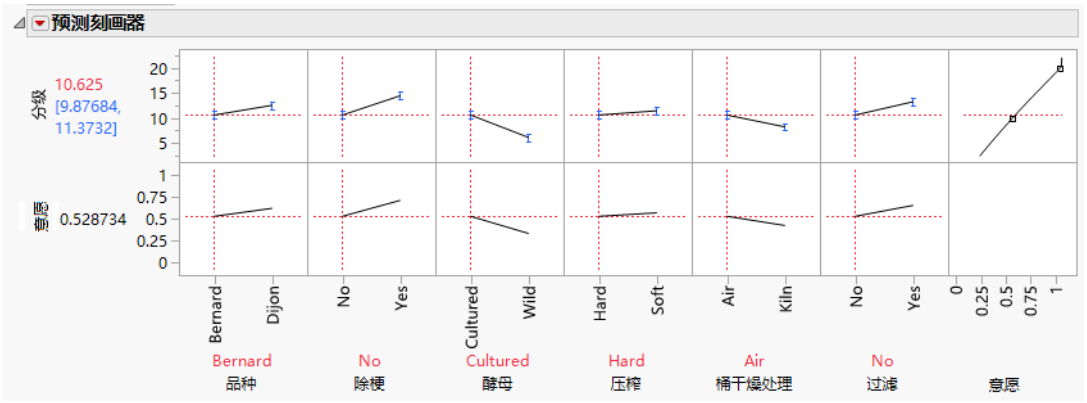
简化模型的“‘预测值-实际值’图”显示不存在失拟问题。“效应汇总”和“效应检验”报表显示其余七项在 0.05 水平下显著。使用预测刻画器可以进一步探索简化模型。

图 4.12 显示了“预测刻画器”。回想一下，您已经指定响应目标为“最大化”，下限和上限分别为 0 和 20。设置这些限值导致“响应限”列属性保存到定制设计表的**分级**列。“预测刻画器”使用“响应限”信息构造意愿函数，该函数显示在图 4.12 的顶行中的最右侧图内。底部行显示意愿迹线。

顶部行中的前六个图显示预测模型的迹线。对于每个因子，图中的线条显示当其他所有因子都设置为红色垂直虚线定义的值时，**分级**如何变化。默认情况下，刻画器中的分类因子设置为其低值设置。通过更改这些因子的设置，您可以查看葡萄酒的预测**分级**如何变化。请注意，已为预测**分级**均值指定了置信区间。

请注意，**评级员**并未包括在刻画器所示的因子中。这是因为**评级员**是一个区组变量。包括**评级员**是为了解释变异，但**评级员**与优化过程因子设置不直接相关。具有给定设置的葡萄酒的预测**分级**是所有评级员对该葡萄酒的预测等级的平均值。

图 4.12 简化模型的刻画器



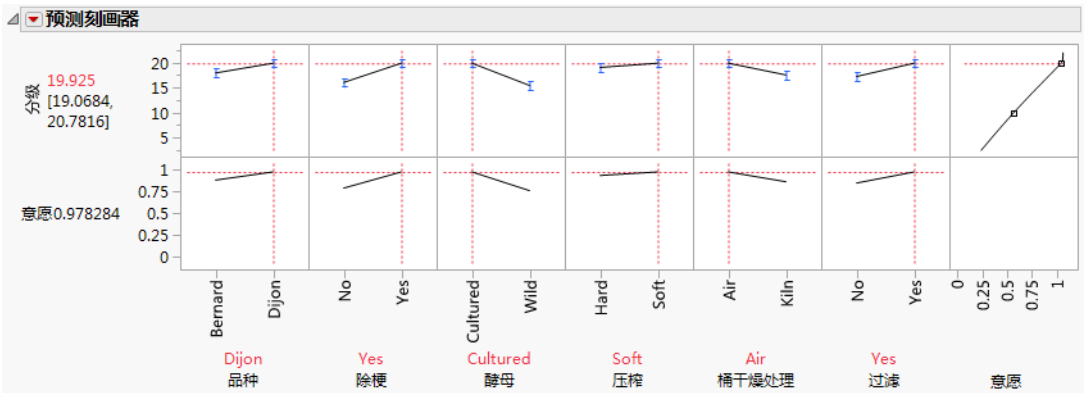
优化因子设置

使用预测刻画器可以基于定制设计结果标识最优设置。您希望确定可将各个评级员的**分级**最大化的设置。

1. 点击“预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

   “预测刻画器”中的红色垂直虚线随之更新，以显示每个因子的最优设置。最优设置导致预测分级为 19.925。通常，可能有多组不同的因子设置都会生成相同的最优值。

图 4.13 已优化因子设置的 “预测刻画器”



2. 要查看所有试验的预测分级，请保存“预测公式”。点击“响应‘分级’”红色小三角并选择**保存列 > 预测公式**。

   名为**预测公式：分级**的列将添加到数据表中。请注意，其中一个试验，第 33 行，得到了**评级员** 5 给出的最大分级 20。**评级员** 5 给出的该试验的预测分级为 19.550。但是第 33 行试验是以最优设置运行的。“预测刻画器”中为这些设置给出的预测值 19.925 是通过对该试验全部五位评级员的预测分级取平均值得到的。

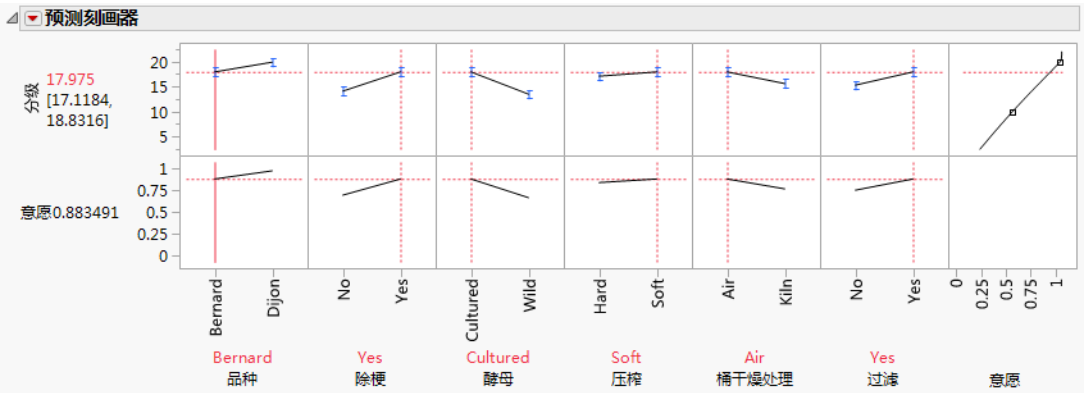
锁定因子水平

您可以在刻画器中锁定因子来探索特定条件下的优化。将设计响应最大化时，您了解到 Dijon 品种的葡萄实现了最优分级 （图 4.13）。您的经理指出在种植 Bernard 葡萄的土地上重新种植 Dijon 葡萄树幼苗在成本上不可行。因此，您需要查找 Bernard 葡萄的最优过程设置和预测分级。

1. 在“预测刻画器”的**品种**图中，将红色垂直虚线拖至 Bernard。
2. 按 Ctrl 键并点击**品种**图之一。

   “因子设置”窗口随即显示。
3. 选择**锁定因子设置**并点击**确定**。
4. 点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 4.14 包含 Bernard 品种的最优设置的 “预测刻画器”



最优设置没有变化，因为模型不包含交互作用项。在这些设置处的预测分级为 17.975。

向刻画器添加评级员

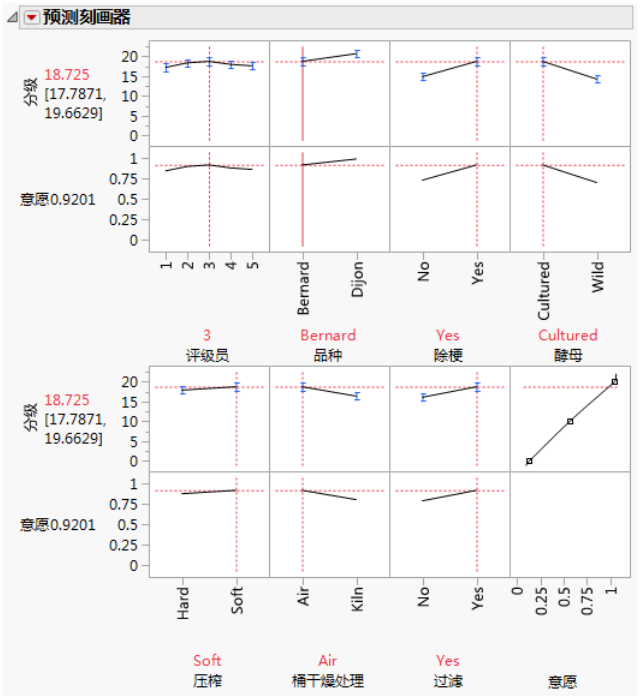
若您想要查看**评级员**各水平的刻画器迹线，请执行以下步骤：

1. 点击“预测刻画器”红色小三角并选择**重置因子网格**。

   随即显示“因子设置”窗口，其中包含所有因子（包括**评级员**）所对应的列。**评级员**下面位于**显示**旁边的框并未选中。这指示**评级员**不显示在“预测刻画器”中。
2. 选中**评级员**下面与**显示**相对应行中的框。
3. 取消选中**评级员**下面与**锁定因子设置**相对应行中的框。
4. 点击**确定**。

   刻画器更新以显示针对**评级员**的图。
5. 点击**评级员**上方两个图中的任一个。

图 4.15 针对显示评级员的简化模型的刻画器



随即显示红色垂直虚线。拖动该线可查看每一个评级员的迹线。请记住，**品种**仍在 Bernard 处锁定。要解锁**品种**，请按 Ctrl 键并点击**品种**图之一。在出现的 “因子设置” 窗口中，取消选择**锁定因子设置**。

定制设计汇总示例

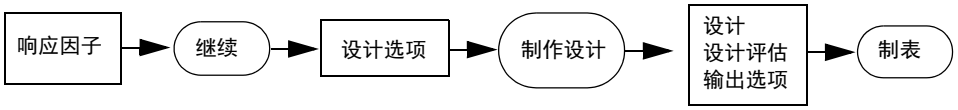
在葡萄酒品尝实验中，仅使用 40 次试验，就从总共九个因子中确定了六个因子对 Pinot Noir 葡萄的评级具有影响。您发现：在这些因子采用最优设置时，可以实现的预测分级为 19.925（最大可能为 20）。您还确定了两个葡萄品种的最优设置。

在本节中，您使用 “定制设计” 窗口中的分级显示项构造了一个设计。下一节会更详细地解释各个分级显示项和设计步骤。

构建定制设计

通过选择**实验设计 > 定制设计**构建定制设计。首先定义响应和因子。接下来，继续执行设计选项、生成和评估过程。设计过程遵循图 4.16 中的流程。

图 4.16 定制设计流程图

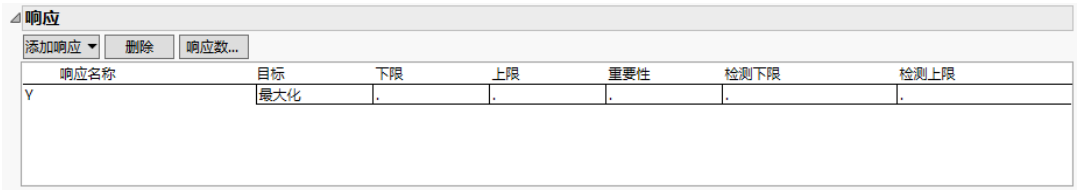


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 4.17 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的 “响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。

– “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
– “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
– “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定“广义回归”平台中的删失响应。请参见《拟合线性模型》中的“广义回归”一章。

编辑“响应”分级显示项

在“响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

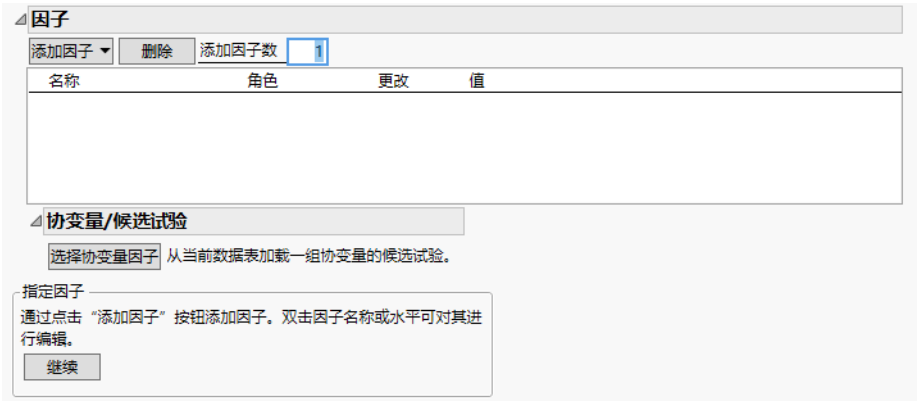
若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

在“因子”分级显示项中添加要在定制设计中研究的因子。

提示：填写完“因子”分级显示项后，考虑从红色小三角菜单中选择**保存因子**。这会将因子名称、角色、更改和值保存到一个数据表中，以便将来重新加载。

图 4.18 “因子”分级显示项



添加因子 选择因子类型。请参见第 85 页的 “因子类型”。

删除 删除选定的因子。

> **注意：**若您尝试在点击“继续”或“后退”按钮后删除所有因子，仍有一个连续因子保留下来。您可以在添加新因子后删除它。

添加因子数 添加多个因子。输入要添加的因子数，点击**添加因子**，然后选择因子类型。重复执行**添加因子数**，添加不同类型的多个因子。

“因子”分级显示项

“因子”分级显示项包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、X2 等。要更改该名称，双击它，然后输入所需名称。

角色 指定因子的设计角色。因子的“设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。

更改 指示因子水平为“容易”、“困难”还是“极难”更改。点击默认值“容易”对其进行更改。若您将因子指定为“困难”或“极难”更改，您的设计将反映出这些随机化限制。除非“因子”列表包含指定为“困难”的因子，否则不能将因子指定为“极难”。“因子更改”列属性将保存至数据表。请参见第 86 页的 “更改和随机区组”。

值 因子的实验设置。要插入值，点击默认值并输入所需值。

编辑“因子”分级显示项

在“因子”分级显示项中，请注意以下事项：

- 要编辑某个因子名称，双击该因子名称。
- 分类因子的因子名称左侧有一个向下箭头。点击该箭头可添加水平。

- 要删除因子水平，点击该值，再点击**删除**，然后在文本框外点击。
- 要修改“更改”之下的条目，请点击**更改**列中的值并选择适当的条目。
- 要编辑某个值，在**值**列中点击该值。

因子类型

要选择某种因子类型，请点击“定制设计”中的**添加因子**。

注意：包含各个因子角色的“设计角色”列属性添加到生成的设计表中该因子对应的列中。“设计角色”列属性确保对因子正确建模。

连续 仅限数值数据类型。在符合过程和测量系统的限制条件下，若一个因子在概念上可以设置为您提供的下限与上限之间的任何值，则这样的因子为连续因子。

离散数值 仅限数值数据类型。离散数值因子只能采用一组离散值。这些值具有隐含的顺序。

有 k 个水平 （其中， $k > 2$）的离散数值因子的默认值是整数 1, 2, ..., k。有 $k = 2$ 个水平的离散数值因子的默认值为 -1 和 1。用您计划在实验中使用的设置替换这些默认值。

注意：并不是离散数值因子的所有水平都显示在设计中。而是根据您在“模型”分级显示项中的指定来确定显示的水平。若想在设计中显示所有水平，请考虑使用“筛选设计”平台。

在假设模型中，含 k 个水平的离散数值因子的效应包含该效应中一直到 k-1 阶的多项式项。对于 k 大于 6 的情况，则包含一直到第 5 个水平的幂次。多项式效应（二次幂或更高次幂）的“可估计性”设置为“若可能”。这允许算法使用试验规模所允许的多个水平。若不包含多项式项，则创建仅包含主效应的设计。有关如何在假设模型中处理离散数值因子的详细信息，请参见第 91 页的 “模型”。

“拟合模型”将离散数值因子视为连续预测变量。保存至设计表的“模型”脚本不包含大于二阶的任何多项式项。

分类 数值或字符数据类型。生成的数据表中的数据类型为“分类”。各水平的值排序就是从左到右输入值的顺序。在创建设计数据表后，该排序将保存在“值顺序”列属性中。

分区组 数值或字符数据类型。分区组因子是一种特殊类型的分类因子，它只能作为主效应进入模型。定义分区组因子时，您需要指定每区组试验次数。“每区组试验次数”列属性将保存至设计表。“默认”试验规模始终假定至少有两个区组。若指定的试验规模不是每区组试验次数的整数倍，JMP 将尝试尽可能令设计平衡。通过平衡设计，JMP 可确保每区组至少有两次试验。

协变量 数值或字符数据类型。协变量因子的值是在实验之前已知的实验单元的测量值。选定协变量值是为确保相对于最优性准则所生成的设计是最优的。请参见第 86 页的 “更改和随机区组”和第 113 页的 “包含难以更改的水平的协变量”。

JMP 从包含可用实验单元的已测量过的协变量的打开数据表中获取协变量因子及其值。令该数据表成为您当前的数据表。**使用添加因子**或点击“协变量候选试验”分级显示项中的**选**

择协变量因子来添加协变量。添加协变量时，将打开当前数据表中列的列表，您可以从该列表中选择包含协变量的列。添加协变量因子会将包含候选行的表加载到“协变量候选试验”分级显示项中。一旦加载，就可以选择要强加到设计中的行。要将选定的协变量行强加到设计中，请使用“设计生成”选项**在设计中包括所有选定的协变量行**。

若设计中的试验次数少于协变量行数，则设计表会包含**协变量行索引**列。该列指示协变量表中与每次试验相对应的行。

在某些情况下，您可能需要从较大的一组候选设置中选择较小的一组设计点。例如，对于一大批单元，您可能有多个测量值列（因子）。您想要将每个单元的测量值都视为一个候选试验。从这些候选试验中，您想要优选出一小部分来测量响应。在这种情况下，将所有候选试验的数据表设置为活跃表，选择“添加因子”>“协变量”，然后将所有测量值列输入为协变量。指定需要的试验规模。“定制设计”平台标识设计设置的最优集合。

注意：在包含协变量时，“设计生成”分级显示项包含两个选项。第一个选项允许您指定将协变量表中的选定行纳入设计。第二个选项允许在设计中重复协变量行。

混料 代表混料成分的连续因子。混料因子的各个值的总和必须为一个常数。默认情况下，所有混料因子的值的总和为 1。要将混料成分的总和设置为其他某个正数值，请从红色小三角菜单中选择**高级选项 > 混料总和**。“混料”列属性将保存至数据表。

常数 数值或字符数据类型。常数因子是其值在实验过程中保持固定不变的因子。常数因子不包含在“模型”分级显示项中，也不包含在保存到数据表的“模型”脚本中。

不受控 数值或字符数据类型。不受控因子是其值在生产过程中不受控制的因子，但却是您想要加入模型的因子。假定您可以记录每次试验的因子值。

设计表中会创建一个具有“连续建模类型”的空列。您可以根据需要在“列信息”窗口中更改该列的“数据类型”和“建模类型”。在该列中输入数据。不受控因子包含在“模型”分级显示项或保存到数据表的“模型”脚本中。

更改和随机区组

指定各次试验之间对因子进行更改的相对难度在工业实验中很有用。当难以更改的因子固定保持某种设置时往往很方便可以执行一些实验。“更改”值为“困难”将生成裂区设计。“更改”值为“极难”将生成裂-裂区设计或双因子裂区设计。

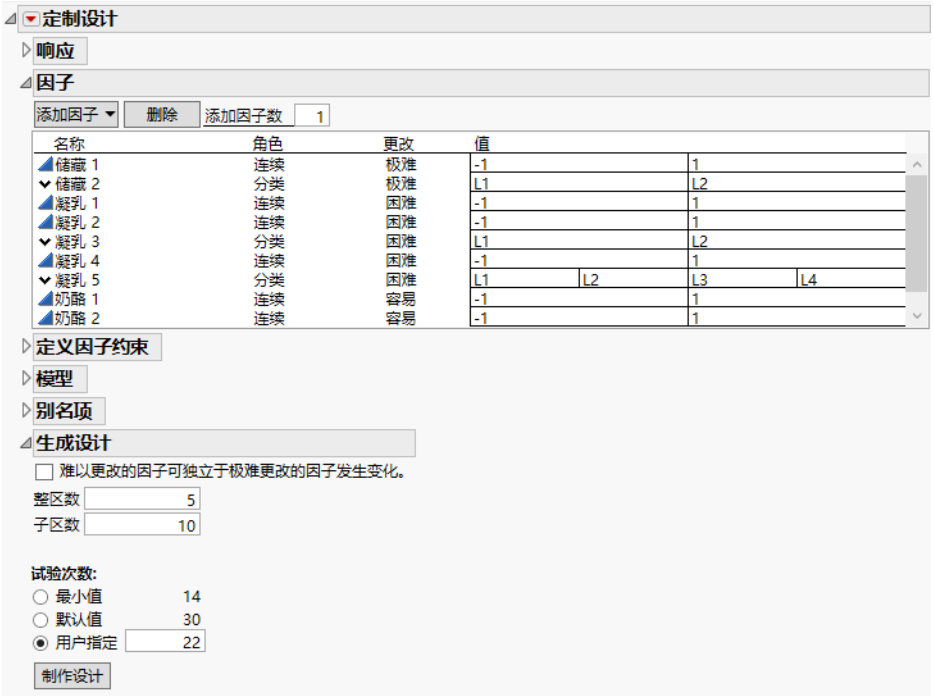
您可以将连续、离散数值、分类和混料因子的“更改”值设置为“困难”和“极难”。要将因子设置为“极难”，列表中必须含有设置为“困难”的其他因子。

您可以将协变量因子的“更改”值设置为“困难”。在这种情况下，所有其他协变量也将设置为“困难”，而其余因子则设置为“容易”。该算法需要结合使用行交换和坐标交换。出于此原因，即便规模适中的设计也需要一些生成时间。

对于包含“困难”或“极难”更改因子的设计，“定制设计”会根据您指定的最优性准则尽力查找最优设计。请参见第 114 页的“最优性准则”。有关用于生成裂区设计的方法的详细信息，请参见 Jones and Goos (2007)。有关与包含难以更改协变量的设计相关的详细信息，请参见 Jones and Goos (2015)。

图 4.19 显示了一种裂-裂区方案，其中使用了 Cheese Factors.jmp 样本数据表中的因子（该数据表位于 Design Experiment 文件夹中）。

图 4.19 裂 - 裂区设计的 “因子” 和 “生成设计” 分级显示项



若为一个或多个因子的 “更改” 值分配了 “困难” ，但没有为任何因子分配 “更改” 值 “极难”，则会向设计添加一个名为**整区**的分类因子。此情形生成一个裂区设计：

- **整区**每个水平都对应一个区组，这个区组的难以更改的因子的设置不变。
- 设计表中的 “模型” 脚本将 “随机效应” 特性应用到因子**整区**。
- 因子**整区**分配有 “设计角色” 列属性，属性值为 “随机区组” 。

若同时为 “更改” 值指定 “困难” 和 “极难”，则会向设计添加名为**子区**和**整区**的分类因子。此情形生成一个裂 - 裂区设计：

- **子区**的每个水平对应一个区组，这个区组的难以更改的因子的设置不变。
- **整区**的每个水平都对应一个区组，这个区组的极难更改的因子的设置不变。
- 设计表中的 “模型” 脚本将 “随机效应” 特性应用到**整区**和**子区**效应。
- 默认情况下，假定难以更改的因子的水平嵌套在极难更改因子的水平之内。
- 在设计表中，**整区**和**子区**这两个因子都分配有 “设计角色” 列属性，属性值为 “随机区组” 。

要构造双因子裂区设计，请选择“生成设计”下的**难以更改的因子可独立于极难更改的因子发生变化**选项。该选项可使难以更改的因子的水平与极难更改的因子的水平交叉。请参见第 111 页的“双因子裂区设计”。

使用“整区数”和“子区数”文本框指定整区数或子区数的值。这些框初始化为整区和子区的建议数目。有关如何获取这些值的信息，请参见第 114 页的“整区和子区数”。

有关演示随机区组裂区、裂-裂区和双因子裂区设计的详细信息和方案，请参见第 106 页的“包含随机化限制的设计”。有关与包含难以更改协变量的设计相关的详细信息，请参见第 113 页的“包含难以更改的水平的协变量”。

因子列属性

对于每个因子，各个列属性保存在数据表中。您可以在附录 A，“列属性”中找到关于这些列属性和相关示例的详细信息。

设计角色 每个因子都分配有“设计角色”列属性。在定义因子时指定的角色确定其“设计角色”列属性的值。在“生成设计”下添加随机区组时，该因子将分配有“随机区组”值。“设计角色”属性反映在对实验数据建模时要如何使用因子。“设计角色”值用在“扩充设计”平台中。请参见“列属性”附录中第 706 页的“设计角色”。

因子更改 每个因子都分配有“因子更改”列属性。您在“更改”下指定的值确定其“因子更改”列属性的值。“因子更改”属性反映在对实验数据建模时要如何使用因子。“因子更改”值用在“扩充设计”和“评估设计”平台中。请参见“列属性”附录中第 720 页的“因子更改”。

编码 若“角色”为“连续”、“离散数值”，连续的“协变量”或“不受控”，则会保存因子的“编码”列属性。该属性变换因子值，以便低值与高值分别对应于 –1 和 +1。请参见“列属性”附录中第 709 页的“编码”。

值顺序 若“角色”为“分类”或“分区组”，将保存因子的“值顺序”列属性。该属性确定因子水平的显示顺序。请参见“列属性”附录中第 723 页的“值顺序”。

混料 若“角色”为“混料”，将保存因子的“混料”列属性。该属性指示因子的限值和混料总和。它还支持您选择混料因子的编码。请参见“列属性”附录中第 715 页的“混料”。

每区组试验次数 对于分区组因子，指示每个区组中所允许的最多试验次数。在“因子”分级显示项中指定“分区组”因子后，将保存该因子的“每区组试验次数”列属性。请参见“列属性”附录中第 729 页的“每区组试验次数”。

定义因子约束

使用“定义因子约束”来限制设计空间。除非您已经加载某个约束或在脚本中包含了某个约束，否则会选定**无**选项。要指定约束，请选择以下其他选项之一：

指定线性约束 指定针对因子的线性组合的不等式约束。仅适用于具有“连续”或“混料”角色的因子。请参见“指定线性约束”。

注意: 保存涉及线性约束的设计的脚本时，该脚本将线性约束表示为**小于等于**不等式 (≤)。

使用不被允许的组合过滤器 基于各个因子的限定值定义各组约束。您可以定义“与”和“或”约束。请参见 “使用不被允许的组合过滤器”。

使用不被允许的组合脚本 定义不被允许的组合和其他约束，作为脚本编辑器框中的布尔 JSL 表达式。请参见 “使用不被允许的组合脚本”。

注意： 当分析具有因子约束的设计时，模型刻画器将接受这些约束。

指定线性约束

当连续因子不可能在设计空间上独立变化时，可以指定线性不等式约束。线性不等式描述了允许的因子水平设置。

点击**添加**以输入一个或多个线性不等式约束。

添加 添加涉及设计中所有连续因子的线性表达式的模板。输入因子的系数值并选择不等式的方向来反映您的线性约束。在不等式右侧的框中指定约束值。要添加更多约束，请再次点击**添加**。

注意： 若您已通过指定球形半径限制了设计区域，将禁用 “添加”选项。

删除最后一个约束 删除最后一个约束。

检查约束 检查约束的一致性。该选项删除多余的约束并执行可行性检查。若有问题将显示 JMP 警示。若约束等效于因子的界限， JMP 警示将指示 “因子”分级显示项中的界限已更新。

使用不被允许的组合过滤器

该选项使用经过改编的数据过滤器帮助指定不被允许的组合。有关详细信息，请参见《使用 JMP》中的“JMP 报表”一章。

从“添加过滤器因子”列表中选择因子并点击**添加**。然后通过使用滑块（对于连续因子）或选择水平（对于分类因子）来指定不被允许的组合。

“添加过滤器因子”菜单的红色小三角选项在许多平台启动窗口的“选择列”面板中都可以找到。请参见《使用 JMP》中的“入门”一章。

点击“添加”时，“不被允许的组合”控制面板显示选定的因子并提供用于进一步控制的选项。基于建模类型，按以下方式表示因子：

连续因子 对于连续因子，将显示跨越因子设置范围的双端滑块。您可以指定不允许的设置，方法是拖动滑块两端或通过点击滑块下的文本值来设置端点。在滑块中，用纯蓝色突出表示不被允许的值。

分类因子 对于分类因子，可能的水平显示为带标签的区组，或者是在水平数较大时显示为列表条目。选择某个水平使其不被允许。要选择多个水平，请按住 Ctrl 键。区组或列表条目将突出显示，以指明不被允许的水平。向“不被允许的组合”面板添加分类因子时，在因子名称后面的括号中给出分类因子的水平数。

不被允许的组合选项

控制面板具有以下控件：

清除 清除您已指定的所有不被允许的因子水平设置。这不会清除选定的因子。

重新开始 删除所有选定因子，并返回最初的因子列表。

与 打开“添加过滤器因子”列表。选定的因子将变为一个“与”组。在“与”组内指定的任意因子水平组合都不被允许。

之后要向“与”组添加因子，请点击该组的分级显示项以查看突出显示的矩形。选择“与”并添加因子。

要删除单个因子，请从该因子的红色小三角菜单中选择**删除**。

或 打开“添加过滤器因子”列表。选定的因子将变为单独的“与”组。对于由“或”分隔的“与”组，若至少在一个“与”组中指定了某个组合，则该组合不被允许。

因子的红色小三角选项

某个因子可以显示在多个“或”组中。特定“或”组中因子的一次出现称为该因子的一个实例。

删除 从“不被允许的组合”面板中删除选定的因子实例。

清除选择 清除该因子实例的任何选择。

反转选择 取消选择选定值，并选择之前没有为该因子实例选定的那些值。

显示选项 仅可用于分类因子。更改显示的外观。选项包括：将每个水平显示为区组、列表或单个类别，或者在每个值旁边添加复选框。

查找 仅可用于分类因子。在因子名称下提供一个文本框，在此可以为因子水平输入搜索字符串。按 Enter 键或在该文本框外点击可执行搜索。一旦选定**查找**，“查找”选项即显示在红色小三角菜单中，如：“不包含”、“区分大小写”、“开始字符为”或“结束字符为”，或者“清除查找”。

使用不被允许的组合脚本

使用该选项来通过 JSL 脚本指定不被允许的因子水平的特定组合。该选项可以用于连续因子或既有连续因子又有分类因子的情况。

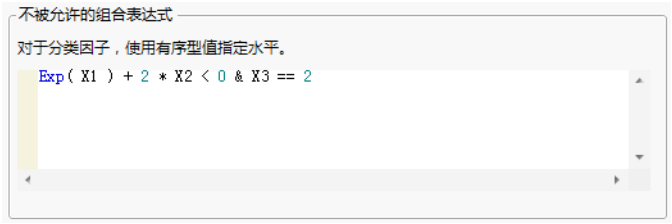
该选项将打开一个脚本窗口，您可以在其中插入用来标识您不想允许的组合的脚本。该脚本必须作为一个布尔表达式来求值。若表达式的计算结果为真，则不允许指定的组合。

创建分类因子的表达式时，使用水平的序数值而非水平的名称。若因子的水平是高、中和低（在“因子”分级显示项中以该顺序指定），则相关的序数值为 1、2 和 3。例如，假定您有两个连续因子 X1 和 X2，还有一个具有三个水平（水平的顺序为 L1、L2 和 L3）的分类因子 X3。您希望不允许符合以下条件的水平：

$$e^{X_1} + 2X_2 < 0 \text{ 并且 } X_3 = L2$$

在脚本窗口中输入表达式 `(Exp(X1) + 2*X2 < 0) & (X3 == 2)`。

图 4.20 脚本编辑器中的表达式



（在图中，通过解析删除了不必要的括号。）请注意函数可以作为布尔表达式的一部分来输入。表达式 `(Exp(X1) + 2*X2 < 0) & (X3 == “L2”)` 也有效。

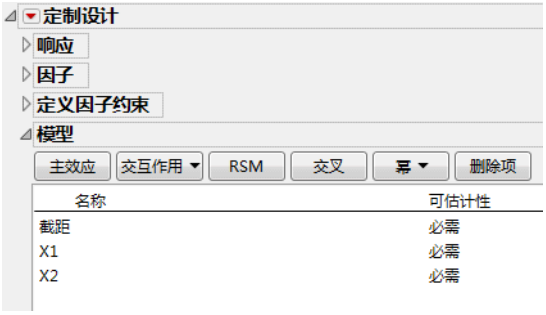
模型

在“模型”分级显示项中指定您的假设模型（其中包含您要估计的所有效应）。对于您指定的每个效应，您都可以指定该效应的“可估计性”。“可估计性”值指示对该效应进行估计是否是“必需”的，或者您是否愿意在“若可能”的情况下估计该效应。

初始“模型”分级显示项包含所有因子的主效应。若输入具有三个或三个以上水平的离散数值因子，则多项式项也将包含在初始模型中。二阶和高阶项的“可估计性”设置为**若可能**。若想要确保这些项可估计，请将其“可估计性”改为**必需**。

注意：您可以确保离散数值多项式项的可估计性始终设置为“必需”。选择**文件 > 首选项 > 平台 > 实验设计**。选中“离散数值幂设置为必需”。

图 4.21 “模型”分级显示项



构造设计表时，JMP 会将“模型”脚本保存至数据表。除了离散数值因子的情况，“模型”脚本都会包含“模型”分级显示项中显示的效应。对于离散数值因子，“模型”脚本仅包含其主效应和二次项。

“模型”分级显示项包含以下按钮和字段：

主效应 为模型中的所有因子添加主效应，为离散数值因子添加多项式项。

交互作用 添加交互作用效应。若未在“因子”分级显示项中选择任何因子，选择“二次”、“三次”、“四次”或“五次”，可添加最高达到该阶数的所有适当交互作用。通过在“因子”分级显示项中选择因子名称，再选择“交互作用”，然后指定适当的阶数，可以添加特定因子最高达到给定阶数的交互作用。不会添加非混料因子与混料因子之间的交互作用，以及带有分区组因子和带有常数因子的交互作用。

RSM 添加所有连续因子最高达到二阶的交互作用和二次项（响应曲面模型项）。分类因子不包括在 RSM 项中。将删除与所有混料因子交互的非混料因子的主效应。

交叉 添加特定的交互作用项。在“因子”分级显示项中选择因子名称，在“模型”分级显示项中选择效应名称。点击“交叉”将交叉项添加到“模型”分级显示项。

幂 添加多项式项。若未在“因子”分级显示项中选择因子名称，则为所有连续因子添加多项式项。若在“因子”分级显示项中选择了因子名称，则仅为这些因子添加多项式项。选择“二次”、“三次”、“四次”或“五次”添加该阶数的多项式项。

Scheffé 三次项 （可用于混料因子。）为所有混料因子添加 Scheffé 三次项。这些项用于指定包含三次多项式项的混料模型。

删除项 删除选定的效应。

名称 效应的名称。

可估计性 指定您是否需要估计效应。

- 若“可估计性”设置为“必需”，则算法确保效应是可估计的。
- 若“可估计性”设置为“若可能”，则算法会尝试在您指定的试验次数允许的前提下令该效应可估计。

除了离散数值因子的多项式项，所有效应都默认指定为“必需”。点击效应的“可估计性”值可更改该值。

Bayesian D 最优性和“若可能”效应的估计

Bayesian D 最优设计方法获取所有**必需**项的精确估计，同时对**若可能**项提供综合检测能力（和部分估计能力）。有关更多详细信息，请参见“定制设计示例”一章中第 142 页的“响应曲面实验”和第 115 页的“Bayesian D 最优性”。若有任何主效应模型项设置为**若可能**，则将 Bayes 信息矩阵用于设计诊断计算。

别名项

在“别名项”分级显示项中添加可能的活跃效应，它们不在您的假设模型中，但是可能使模型项的估计值发生偏倚。**未**包含在假设模型中的效应有可能是活跃的。一旦生成设计，“别名矩阵”分级显示项将显示在“设计评估”之下。“别名矩阵”条目表示您在“别名项”分级显示项中指定的效应传导给模型参数的偏倚度。请参见“技术细节”附录中第 735 页的“别名矩阵”。

默认情况下，“别名项”分级显示项包含“模型”分级显示项中未包含的所有双因子交互作用效应（涉及分区组因子的项除外）。使用这些按钮添加项。有关如何使用这些按钮向“别名项”表中添加效应的说明，请参见第 91 页的“模型”。

注意：对于交互作用项，仅添加所选次数的交互作用。这不同于“模型”分级显示项，其中要向模型添加所选次数以及更低次数的交互作用。

例如，假定您指定了一个包含三个连续因子的设计。您在“模型”分级显示项中指定的假设模型仅包含这三个主效应。您的预算只能执行六次试验。您想要查看活跃的双因子交互作用和三因子交互作用如何使这些主效应的估计值产生偏倚。

默认情况下，“别名项”表包括所有双因子交互作用。您可以通过选择**交互作用 > 三次**添加三因子交互作用。

图 4.22 “别名项”分级显示项

一旦您在“生成设计”分级显示项中指定了六次试验并点击**制作设计**之后，“设计评估”分级显示项随即显示。打开“设计评估”分级显示项和“别名矩阵”分级显示项。

图 4.23 别名矩阵

效应	X1*X2	X1*X3	X2*X3	X1*X2*X3
截距	0	0	0.5	-0.5
X1	0	0	-0.5	0.5
X2	0.5	-0.5	0	0
X3	-0.5	0.5	0	0

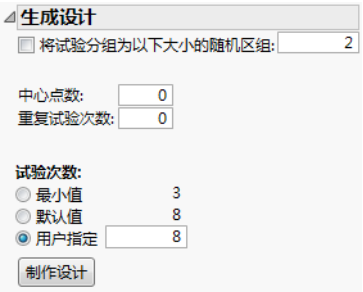
“别名矩阵”指示每个主效应都与两个交互作用存在部分别名关系。请参见“评估设计”一章中第 418 页的“别名矩阵”和“技术细节”附录中第 735 页的“别名矩阵”。

生成设计

“生成设计”分级显示项为您提供与设计中的分区组、中心点、重复和试验次数相关的选项。通常，输入区域包含两个部分：

- 设计结构选项
- 试验次数选项

图 4.24 “生成设计”分级显示项



设计结构选项

将试验分组为以下大小的随机区组 （若指定分区组因子则不可用。）要构造随机区组设计，请输入您希望每个区组包含的试验次数。指定样本大小时，将创建名为**随机区组**的因子。根据指定的试验次数，其水平定义大小与您输入的区组大小一致的区组。若试验次数是区组大小的整数倍，则区组大小等于您指定的值。

整区数 在您指定困难或极难更改的因子时显示。因子**整区**对应于 （若存在）极难更改的因子（裂 - 裂区设计），否则对应于难以更改的因子 （裂区设计）。 JMP 建议整区数使用可将模型系数相关信息最大化的值。或者，您可以为整区数输入一个值。请参见第 114 页的“整区和子区数”。

子区数 在您指定极难更改的因子时显示。因子**子区**对应于裂 - 裂区设计中的难以更改的因子。JMP 建议整区数和子区数使用可将模型系数相关信息最大化的值。或者，您可以为子区数输入一个值。请参见第 114 页的 “整区和子区数”。

难以更改的因子可独立于极难更改的因子发生变化 选择该选项可创建带区 （亦称双因子裂区或分裂区组）设计。在该选项创建的设计中，难以更改的因子在极难更改因子的水平之内随机分布。它们**不**嵌套在极难更改的因子之内。

中心点数 仅当设计包含 “连续”或 “混料”因子类型的因子时才显示。指定您要作为中心点向设计额外添加的试验次数。中心点是对于每个连续因子而言，其设置位于高值设置与低值设置中央的试验。请参见“从实验设计入手”一章中第 61 页的“中心点、重复试验和检验”。

若设计包含连续因子和其他类型的因子，则中心点相对于其他因子的水平可能并不均衡。“定制设计”选择中心点以使设计的 D、 I 或别名效率最大化。

重复试验次数 指定您要添加到设计中的重复试验次数。这不会重复整个设计，而只是选择最优设计点来重复。请参见“从实验设计入手”一章中第 61 页的“中心点、重复试验和检验”。

试验次数选项

最小值 为避免生成设计失败而必需的试验次数的下限。若选择**最小值**，生成的设计是饱和的。不存在误差自由度。

注意：若对试验次数选择 “最小值” ，则没有用于检验的误差项。您将无法检验参数估计值。仅当因成本限制无法执行额外试验时，此选项才适合。

默认值 给出建议的试验次数。该值通过启发方式来创建比最小试验次数多至少四次试验的平衡设计。

用户指定 指定您所需的试验次数。在**试验次数**文本框中输入该值。该选项支持您在信息上可能带来的收益与额外试验成本的增加之间进行权衡。

试验次数 在指定协变量因子 （“更改”值设置为 “容易”）时，这是唯一显示的选项。显示的试验次数是数据表中与一个或多个协变量关联的行数。您可以指定较少的试验次数。在这种情况下，选定的协变量试验可优化设计准则。

制作设计

一旦完成“生成设计”分级显示项，请点击**制作设计**。使用“定制设计”可生成设计，在“设计”分级显示项中提供它，并在“设计评估”分级显示项中提供评估信息。请注意，“模型”和“别名”分级显示项中的控件被禁用。使用“后退”按钮可更改模型或别名项。还会显示“输出选项”面板，支持您创建设计表。

注意：有时，若干设计可优化最优性准则。若是这种情况，设计算法可能在反复点击**后退**和**制作设计**按钮后生成不同设计。

设计

“设计”分级显示项显示用于定制设计的试验。在您指定的条件下，设计是最优的。这些试验看起来可能并未随机化。您可以在生成设计表之前选择“输出选项”面板中的“试验顺序”选项。

设计评估

“设计评估”红色小三角菜单和分级显示项提供多种方式来评估生成的定制设计的属性。

使用 Bayes 信息 选择对于含有不可估计的**若可能**模型效应的设计，在设计诊断计算中使用 Bayes 信息矩阵。有关 Bayes 信息调整的详细信息，请参见第 115 页的 “Bayesian D 最优性”。

注意：若所有的模型效应都是可估计的，则对所有效应都提供“设计诊断”，而不进行 Bayes 信息调整。若有些项不可估计，则除非选定**使用 Bayes 信息**选项，否则仅显示**必需**模型项。

功效分析 支持您探索检测给定大小的效应的能力。

预测方差刻画 显示针对因子设置范围的预测方差。

设计空间比例图 显示模型预测方差位于给定值之下或之上的比例。

预测方差曲面 显示任意两个连续因子的预测方差曲面图。

估计效率 对于每个参数，给出与理想化（正交）设计（该设计可能不存在）的置信区间长度相比，置信区间长度的部分增加。还给出参数的相对标准误差。

别名矩阵 给出系数，这些系数指示因为可能是活跃的却不在模型中的效应导致模型参数的偏倚程度。在“别名项”表中指定表示可能的活跃效应的项。请参见“技术细节”附录中第 735 页的“别名矩阵”。

相关性色图 使用强度尺度在图上显示两个效应之间的绝对相关性。

注意：默认强度尺度为灰色尺度。要更改颜色尺度，请点击“相关性色图”红色小三角并选择**由蓝经灰到红**。对于定制颜色尺度，在图中右击并选择**颜色主题**。

设计诊断 指出用于构造设计的最优性准则。还给出设计的效率测度。请参见第 98 页的“定制设计选项”和第 114 页的“最优性准则”中的“最优性准则”。

注意：在模型包含“更改”设置为“困难”或“极难”或“可估计性”设置为“若可能”的因子时，“设计诊断”分级显示项不提供以下统计量：D 效率、G 效率、A 效率。

有关“设计评估”分级显示项的详细信息，请参见“评估设计”一章中第 408 页的“设计评估”。

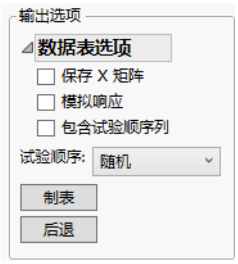
输出选项

使用“输出选项”面板可保存您的定制设计 X 矩阵、模拟响应或定制您的设计数据表。您可以执行下列任务：

- 指定您希望定制设计数据表如何显示
- 构造设计表
- 返回“定制设计”窗口中的上一个点

多数情况下，“输出选项”面板如图 4.25 所示：

图 4.25 “输出选项”面板



“输出选项”面板包含以下选项：

- 第 97 页的“数据表选项”
- 第 97 页的“试验顺序”
- 第 97 页的“制表”
- 第 98 页的“后退”

数据表选项

数据表选项包括**保存 X 矩阵**选项（第 103 页的“保存 X 矩阵”）、**模拟响应**选项（第 101 页的“模拟响应”）和**包含试验顺序列**选项。

包含试验顺序列 向设计数据表添加与 “设计”分级显示项中的设计顺序相对应的一列。

提示：若保存的数据具有排序的试验顺序，则使用“包含试验顺序列”。这将提供“设计”分级显示项中的随机化试验顺序列。

试验顺序

试验顺序选项确定试验在设计表中的顺序。包括以下选项：

保持相同 设计表中的行与 “设计”分级显示项中的行具有相同的顺序。

从左至右排序 设计表中的列按从左至右排序。

随机 设计表中的行采用随机顺序。

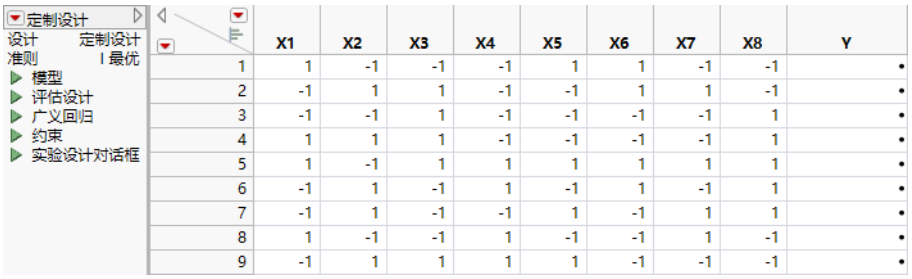
从右至左排序 设计表中的列按从右至左排序。

区组内随机 设计表中的行在区组内采用随机顺序。

制表

点击**制表**可构造定制设计数据表。在定制设计表中，“表”面板（左上角）可根据设计需要包含脚本。始终提供“模型”、“评估设计”和“实验设计对话框”脚本。要运行脚本，请点击脚本名称旁边的绿色小三角。若设计中包括协变量，则设计表包括协变量行索引列，除非所有协变量行都用在设计中。

图 4.26 显示脚本的定制设计表



	X1	X2	X3	X4	X5	X6	X7	X8	Y
1	1	-1	-1	-1	1	1	-1	-1	•
2	-1	1	1	-1	-1	1	1	-1	•
3	-1	-1	1	-1	-1	-1	-1	1	•
4	1	1	1	-1	-1	-1	-1	1	•
5	1	-1	1	1	1	1	1	1	•
6	-1	1	-1	1	-1	1	-1	1	•
7	-1	1	-1	-1	1	-1	1	1	•
8	1	-1	-1	1	-1	-1	1	-1	•
9	-1	1	1	1	1	-1	-1	-1	•

可能的脚本包括：

模型 运行**分析 > 拟合模型**平台。脚本所描述的模型由您在 “模型”分级显示项的选择和设计类型来决定。

评估设计 运行**实验设计 > 设计诊断 > 评估设计**平台。脚本所描述的模型由您在 “模型”分级显示项的选择和设计类型来决定。

约束 显示您在 “定义因子约束”分级显示项中使用**指定线性约束**选项输入的模型约束。

不被允许的组合 显示您在 “定义因子约束”分级显示项中使用**使用不被允许的组合过滤器**或**使用不被允许的组合脚本**选项输入的模型约束。

实验设计对话框 重新创建您用于生成设计表的 “定制设计”窗口。该脚本还包含用于生成设计的随机种子。

后退

“后退”按钮令您回到点击“制作设计”之前您所处的位置。您可以对之前的分级显示项进行更改并重新生成设计。

注意：若您尝试在点击“后退”按钮后删除所有因子，仍有一个连续因子保留下来。您可以在添加新因子后删除该连续因子。

定制设计选项

“定制设计”红色小三角菜单包含用于设计设置和生成的选项。

保存响应 将 “响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用 “保存响应”选项保存的响应。

保存因子 将 “因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在 “设计角色”区域中，选择合适的角色。

加载因子 加载使用 “保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在 “定义因子约束”或 “线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为 “小于”或 “大于”约束。请参见 “列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用 “保存约束”选项保存的因子约束。

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的 “试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击 “制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示 “模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一： “正态”、 “二项式”或 “Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开 “模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在 “模型”窗口中做出选择以控制模拟的响应值的分布。当您点击 “应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中， Y 是响应的名称。再次点击 “应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见 “定制设计”一章中第 101 页的 “模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见 《基本分析》中的 “模拟”一章。

保存 X 矩阵 将 “矩量矩阵”和 “模型矩阵”保存至设计数据表中的表脚本。这些脚本包含矩量和设计矩阵。请参见第 103 页的 “保存 X 矩阵”。

警告：对于包含名义型因子的设计，使用 “保存 X 矩阵”选项保存的 “模型矩阵”**不是**拟合线性模型时使用的编码矩阵。通过在运行 “模型”脚本时获取的 “拟合模型”报表中选择选项 “保存列” > “保存编码表”，您可以获取用于拟合模型的编码矩阵。

最优性准则 更改设计最优性准则。默认准则**推荐**为所有设计类型指定 D 最优性，除非您使用 “模型”分级显示项中的 “RSM”按钮添加了二次效应。有关 D、I 和别名最优设计的详细信息，请参见第 114 页的 “最优性准则”。

注意：您可以设置首选项，以便始终使用给定的最优性准则。选择**文件** > **首选项** > **平台** > **实验设计**。选中 “最优性准则”并选择您偏好的准则。

开始数 支持您指定用于构造设计的随机开始数。请参见第 105 页的 “开始数”。

设计搜索时间 搜索设计所用的最大秒数。默认搜索时间基于设计的复杂性。请参见第 105 页的 “设计搜索时间”和第 105 页的 “开始数”。

若算法迭代需要不止几秒，“正在计算设计”进度窗口随即显示。若点击进度窗口中的**取消**，计算将停止，同时给出目前为止的最佳设计。当设计中不包含 “更改”设置为 “困难”或 “极难”或 “可估计性”设置为 “若可能”的因子时，该进度窗口还显示 D 最优设计的 D 效率。

注意：您可以为 “设计搜索时间”设置首选项。选择**文件** > **首选项** > **平台** > **实验设计**。选中 “设计搜索时间”并输入最大秒数。在需要更长时间的特定情形下， JMP 会延长搜索时间。

球形半径 将设计中的连续因子限定到一个超球面中。指定半径并点击**确定**。挑选的设计点与 0 之间的距离等于球形半径。先选择该选项，然后再点击 “制作设计”。

注意：不能将球形半径约束与使用 “指定线性约束”选项添加的约束结合使用。同样，当包含难以更改的因子时 （裂区设计），该选项也不可用。

高级选项 > 混料总和 将混料因子的总和设置为任意正值。使用该选项可令混料的成分在实验过程中始终保持不变。

高级选项 > 裂区方差比 指定随机整区和子区 （若存在）方差与误差方差之间的比值。设置该值之前，您必须为裂区设计定义难以更改的因子，或是为裂 - 裂区设计定义难以更改和极难更改的因子。然后，您可以为方差比输入一个或两个正数，具体取决于您指定的是裂区设计还是裂 - 裂区设计。

高级选项 > 先验参数方差 （仅当 “模型” 分级显示项可用时才可用。）指定对 “可估计性” 设置为 “若可能” 的因子使用的权重。当您点击 “制作设计” 时，该选项随之更新以显示默认权重。为您要指定权重的每个项输入一个正数。您输入的值是先验方差倒数的平方根。较大的值代表较小的方差，因此表明效应不活跃的先验信息越多。

使用 Bayesian D 或 I 最优性来构造包含 “若可能” 因子的设计。在算法中使用的默认值为：“必需” 项为 0，涉及 “若可能” 项的交互作用为 4，“若可能” 项为 1。请参见 “技术细节” 附录中第 735 页的 “别名矩阵” 和第 114 页的 “最优性准则”。

高级选项 > A 最优参数权重 （用于 A 最优设计。）指定模型参数的权重。这样您就可以对主效应（相对于二阶效应（举例来说））的方差赋予更大的权重。有关参数权重的详细信息，请参见 Morgan and Stallings (2017)。

高级选项 > D 效率权重 指定构造设计时 D 效率相对于别名最优性的相对重要性。选择该选项可在减小系数方差与获取所需别名结构之间进行权衡。值应介于 0 到 1 之间。较大值赋予 D 效率更大权重。默认值为 0.5。仅当您将 “制作别名最优设计” 选作 “最优性准则” 时，该选项才有效果。

有关 D 效率的定义，请参见第 114 页的 “最优性准则”。有关别名最优性的详细信息，请参见第 118 页的 “别名最优性”。

将脚本保存至脚本窗口 为您在 “定制设计” 窗口中指定的设计创建脚本，并将其放入打开的脚本窗口。

模拟响应

在数据收集之前或在课堂环境中，使用模拟响应进行设计探索。选定模拟响应并点击 “制表” 以创建设计表时，“模拟响应” 选项将为每个响应执行以下操作：

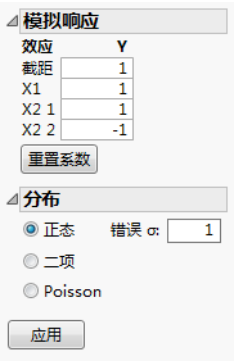
- 将随机响应值添加至设计表中的响应列。
- 将包含模拟模型公式的新列添加至设计表。公式和值基于在设计窗口中指定的模型。

显示 “模型” 窗口，您可在其中指定参数值并选择用于模拟的响应分布。在 “模型” 窗口中点击 “应用” 后，包含模拟模型公式的每一列都随之更新。

控制窗口

图 4.27 为具有一个连续因子 (X1) 和一个三水平分类因子 (X2) 的设计显示 “模型” 窗口。请注意，X2 用两个模型项表示。

图 4.27 模拟响应控制窗口



最初的窗口显示系数值为 1 或 -1，以及误差标准差为 1 的正态分布。若已在“实验设计”窗口中“设计评估”下方将“预期系数”设置为“功效分析”的一部分，则“模拟响应”分级显示项中的默认值就是您在“功效分析”分级显示项中指定为“预期系数”和“预期 RMSE（标准误差）”的值。若不可能拟合在数据表的“模型”脚本中指定的模型，则截距和系数具有默认值 0。

模拟响应

要为模拟值指定模型，请执行以下操作：

1. 对于效应列表中的每个项，输入用于模拟响应值的线性模型的系数。这将定义一个线性函数 $L(\mathbf{x}, \beta) = \mathbf{x}'\beta$。请参见图 4.27 中的“模拟响应”分级显示项：
   - 向量 **x** 包含定义“效应”下列出的效应的项。
   - 向量 β 是您在 Y 下指定的模型系数的向量。
2. 在“分布”下，选择一种响应分布。
3. 点击**应用**。包含模拟值的 **<Y> 模拟**列及其公式将添加至设计表，其中，Y 是响应列的名称。

分布

从“模拟响应”窗口中的可用分布中选择其一：

正态 模拟正态分布中的值。为误差 σ（正态误差分布的标准差）输入一个值。若您已在“因子”分级显示项中指定因子“更改”值为“困难”，则可以为整区 σ（整区误差）输入一个值。若您已指定因子的“更改”值为“困难”和“极难”，则可以为同时为子区和整区误差输入值。点击“应用”后，随机值和包含基于模型的随机响应向量的公式将输入列 **<Y> 模拟**中。

二项 模拟二项分布中的值。输入 N（即试验次数）的值。根据基于成功概率为 $1/(1 + exp(-L(\mathbf{x}, \beta))$ 的 N 次试验的二项分布生成随机整数值。当您点击“应用”后，将在列 **<Y> 模拟**中输入随机值及其公式。名为 **N 次试验**的包含值 N 的列也会添加至数据表。

Poisson 根据带有参数 $exp((L(\mathbf{x}, \beta))$ 的 Poisson 分布模拟随机整数值。当您点击 “应用” 后，将在列 <Y> **模拟**中输入随机值及其公式。

注意：您可以设置首选项以便在每次点击 “制表” 时模拟响应。为此，请选择**文件 > 首选项 > 平台 > 实验设计**。选择**模拟响应**。

保存 X 矩阵

该选项将名为 “矩量矩阵” 和 “模型矩阵” 的脚本（它们分别包含矩量矩阵和模型矩阵）保存至设计数据表。矩量矩阵和模型矩阵用于计算 “预测平均方差”，该方差显示在 “设计评估” 分级显示项的 “设计诊断” 部分中。请参见 Goos and Jones (2011)。若设计属于裂区设计，则还会保存 “V 求逆” 脚本。“V 求逆” 脚本包含响应的协方差矩阵的逆协方。

警告：对于包含名义型因子的设计，使用 “保存 X 矩阵” 选项保存的 “模型矩阵” **不是**拟合线性模型时使用的编码矩阵。通过在运行 “模型” 脚本时获取的 “拟合模型” 报表中选择选项 “保存列” > “保存编码表”，您可以获取用于拟合模型的编码矩阵。

注意：您可以通过设置首选项始终保存矩阵脚本。选择**文件 > 首选项 > 平台 > 实验设计**。选中 “保存 X 矩阵”。

模型矩阵

模型矩阵描述实验设计。在模型矩阵中，每次试验都对应一行，“模型” 分级显示项中指定的每一个模型项都对应一列。对于每次试验，模型矩阵的相应行中都会包含模型项的编码值：

- 连续项按照从 -1 到 1 的范围来编码。
- 对在线性模型拟合中使用到的编码向量，通过应用 Gram-Schmidt 正交过程对名义项编码。

有关 Gram-Schmidt 正交过程的详细信息，请参见 Horn and Johnson (2012)。

注意：功效分析的编码与拟合线性模型的编码相匹配。有关用于拟合线性模型的名义型项的编码的信息，请参见《拟合线性模型》。

矩量矩阵

矩量矩阵依赖模型效应但不依赖设计。其定义如下：

$$\mathbf{M} = \int_R f(\mathbf{x}) f(\mathbf{x})' d\mathbf{x}$$

其中，$f(x)$ 表示与因子向量 x 的因子组合对应的模型效应，R 表示设计空间。有关矩量和设计矩阵的更多详细信息，请参见 Goos and Jones (2011) 和 Myers et al. (2009)。请注意，矩量矩阵在 Myers et al. (2009)。

脚本

从“定制设计”红色小三角菜单中，选择**保存 X 矩阵**。创建设计和表之后，在定制设计表中，“矩量矩阵”和“模型矩阵”脚本以及“V 求逆”脚本（若设计为裂区）都将另存为表脚本。

- 右击并从“矩量矩阵”、“模型矩阵”或“V 求逆”脚本选择**编辑**。脚本将显示对应的矩阵。您可以将该矩阵复制到您编写的脚本中。
- 运行“矩量矩阵”脚本时，日志中显示矩量矩阵中的行数（称为“矩”）。
- 运行“模型矩阵”脚本时，日志中显示模型矩阵中的行数（称为 X）。
- 运行“V 求逆”脚本时，日志中显示逆协方差矩阵中的行数（称为“V 求逆”）。

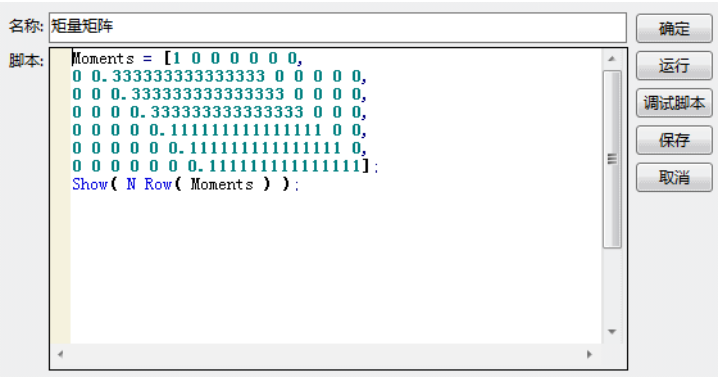
示例

本例演示模型矩阵脚本的用法：

提示：要查看日志，请选择**视图 > 日志**（在 macOS 上为**窗口 > 日志**）。

1. 选择**实验设计 > 定制设计**。
2. 添加 3 个连续因子并点击**继续**。
3. 点击**交互作用 > 二次**。
4. 点击“定制设计”红色小三角并选择**保存 X 矩阵**。
5. 使用默认试验次数 (12)，依次点击**制作设计**和**制表**。
6. 在“表”面板中，右击“矩量矩阵”脚本并选择**编辑**。
   脚本显示在脚本窗口中。该脚本显示称为**矩**的矩量矩阵。

图 4.28 “矩量矩阵”脚本



7. 若该脚本尚未打开，请选择**视图 > 日志**（在 macOS 上为**窗口 > 日志**）。
8. 在“表”面板中，点击“矩量矩阵”脚本旁边的绿色小三角。
   日志中显示的行数为 `N Row(Moments)=7`。

9. 在“表”面板中，右击“模型矩阵”脚本并选择**编辑**。

   脚本显示在脚本窗口中。该脚本显示称为 X 的模型矩阵。

10. 点击**运行**。

    日志中显示的行数为 N Row(X)=12。

11. 要以数据表形式查看模型矩阵，请向脚本中添加以下行：

```
dt = New Table( "Model Matrix" );
dt << Set Matrix( X );
```

12. 点击**运行**。

开始数

开始数指的是坐标交换算法开始构造新设计时所采用的次数。请参见第 118 页的“坐标交换算法”。您可以使用**开始数**选项指定自己的值。增大随机开始数往往会改进生成的设计的最优性。

除非您为“开始数”指定了某个值并点击“确定”，否则开始数将由“设计搜索时间”控制。要查看构造设计所用的开始数，请点击“制作设计”。然后选择“开始数”。文本框中的值是用于构造特定设计的开始数。

在某些特殊情况下，全局最优设计可自理论得知。若坐标交换算法检测到已发现最优设计，它就会停止搜索并返回该设计。

提示：要重现特定设计，您需要指定最初用于生成该设计的“开始数”和“随机种子”。点击“制作设计”之后从红色小三角选项中获取这些值。

在本文的定制设计示例中，通常会提供随机种子和开始数，以便您精确重建所述的设计。

设计搜索时间

“设计搜索时间”是为查找最优设计而分配的时间量。在定制设计的坐标交换算法中，需要根据随机开始设计查找接近最优的设计。请参见第 118 页的“坐标交换算法”。“设计搜索时间”基于随机开始设计确定构造多少个设计。

您可以使用**设计搜索时间**选项指定自己的值。增加搜索时间往往会改进生成的设计的最优性。

谨记：通过重新运行该算法所生成的设计可能有所不同。即便使用相同的随机种子，用于构造最终设计的开始设计数也可能不同，因为计算速度和其他因子会有变化。

注意：开始设计数通过“开始数”文本框中的值来指定。不过，只有在您通过点击“制作设计”构造了设计之后，该值才会更新。

在某些特殊情况下，全局 D 最优设计可自理论得知。这些情况包括：

- 两水平部分析因设计或非常规正交表。这些对于所有主效应和双因子交互作用模型都是全局 D 最优设计。

- 拉丁方设计。这些对于主效应模型（假设采用正确的样本大小和因子水平数）是 D 最优设计。
- Plackett-Burman 设计。这些对于主效应模型是 D 最优设计。

若坐标交换算法检测到已发现最优设计，它就会停止搜索并返回该设计。

定制设计的技术细节

- “包含随机化限制的设计”
- “包含难以更改的水平的协变量”
- “整区和子区数”
- “最优性准则”
- “D 效率”
- “坐标交换算法”

包含随机化限制的设计

完全随机设计并不总是可行的。在设计实验时，必须考虑随机化的逻辑或物理限制。本节介绍“定制设计”平台如何处理无法向因子水平设置随机分配实验单元的各种设计。这些设计包括随机区组设计和各种类型的裂区设计。

随机区组设计

随机区组设计将实验中的各次试验归组到区组中，这些区组被视为是从较大总体中随机选取的。试验区组中的试验通常比不同区组中的试验更具齐性。在这些情况下，若您能估计由分区组变量解释的变异，则往往能够更好地识别出其他效应。

随机区组设计方案

Goos (2002) 提供的示例涉及一个糕点面团混合实验。该实验的目的是要了解面团的特定属性如何依赖于三个因子：进料流速、初始含水量和螺杆转速。由于每天只能执行四次试验，该实验需要运行几天时间。环境变量中各天的随机差异对给定某天执行的所有试验有一定影响。要解释各天之间的变异，试验归入四个一组的区组中，这样这种变异不会影响关于这三个因子的信息。

分区组因子**日期**包含每天的试验。执行试验的日期在具有不同环境条件的较大日期总体中具有代表性。**日期**是一个随机分区组因子。

随机区组设计的设置

要创建随机区组设计，请照常使用“定制设计”平台输入响应和因子并定义您的模型。在“生成设计”分级显示项中，选择**将试验分组为以下大小的随机区组**选项，并输入您希望每个区组包含的试验次数。请参见第 94 页的“设计结构选项”。

注意：要定义固定分区组因子，请在“因子”分级显示项中输入分区组因子。要定义随机分区组因子，请不要在“因子”分级显示项中输入分区组因子；而应选择“生成设计”下的**将试验分组为以下大小的随机区组**选项。

裂区设计

裂区设计用于以下情形：特定因子的设置在各组试验之间保持不变。在工业中，这些因子通常是在各次试验之间难以更改或更改成本高昂的因子。设置在各组试验之间需保持不变的因子在 JMP 中归入**难以更改**一类。

由于特定因子难以更改，将其随机分配给实验单元并不可行。相反，应将其分配给各组单元。这就对随机化施加了限制，在生成设计和分析结果时必须要考虑这种限制。

裂区设计方案

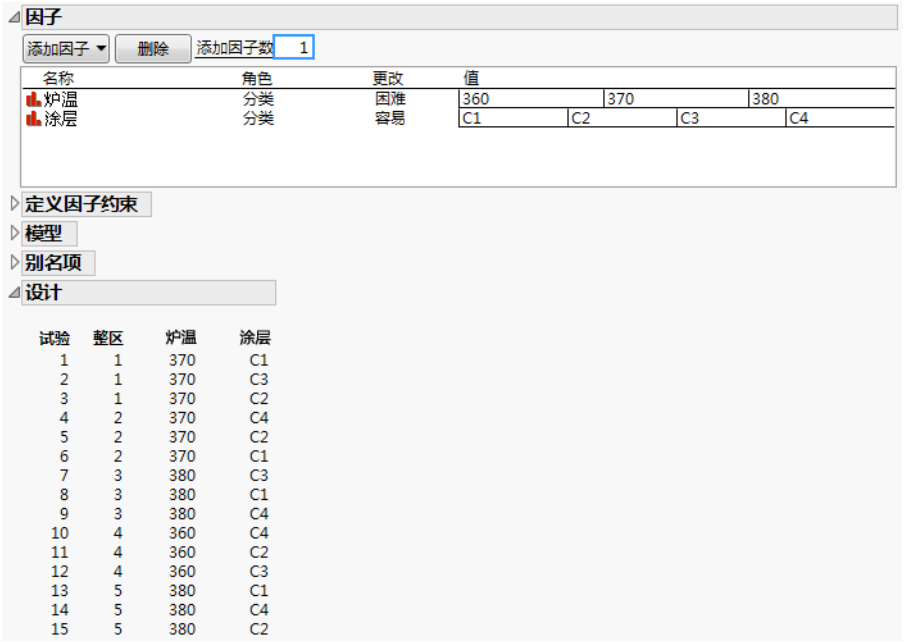
Box et al. (2005) 提供了一个实验，用于研究钢筋的耐腐性。钢筋被放入熔炉进行固化。之后覆上涂层以增加耐腐性。关注的两个因子为：

- **炉温**，采用摄氏度，包含 360、370 和 380 三个水平
- **涂层**，包含 C1、C2、C3 和 C4 四个水平，分别描述四种不同的涂层类型

炉温为难以更改的因子，因为重设炉温要花费较长时间。出于上述原因，针对每个炉温设置加工四根钢筋。在之后的阶段中，对这四根钢筋随机分配四种涂层。

实验单元为钢筋。**炉温**是难以更改的因子，其水平定义整区。在每个整区内，**涂层**因子随机分配给应用了整区因子的实验单元。

图 4.29 裂区设计的“因子”和“设计”分级显示项



试验	整区	炉温	涂层
1	1	370	C1
2	1	370	C3
3	1	370	C2
4	2	370	C4
5	2	370	C2
6	2	370	C1
7	3	380	C3
8	3	380	C1
9	3	380	C4
10	4	360	C4
11	4	360	C2
12	4	360	C3
13	5	380	C1
14	5	380	C4
15	5	380	C2

在腐蚀实验的“因子”分级显示项中，**炉温**的“更改”值设置为“困难”，**涂层**的“更改”值设置为“容易”。这个 15 试验设计包含五个整区，在这些整区内**温度**设置保持不变。

裂区设计的设置

通常，可以对设置为难以更改的加工步骤应用若干因子。在熔炉示例中，您可以考虑熔炉位置因子以及温度。在“因子”分级显示项中的“更改”列下，您需要为此类因子指定“更改”值“困难”。

若定制设计仅涉及易于更改和难以更改的因子，则使用称为**整区**的新因子对难以更改因子的试验分组。**整区**的值使用与难以更改的因子完全相同的设置指定试验区组。保存至设计表的“模型”脚本将**整区**视为随机效应。请参见第 84 页的“更改”和第 94 页的“设计结构选项”。

有关创建裂区设计和分析实验数据的示例，请参见“定制设计示例”一章中第 177 页的“裂区实验”。

裂 - 裂区设计

若难以更改的因子有两个水平，则使用裂-裂区设计。在工业中，当不同批次的材料或实验单元从一个处理阶段过渡到另一个处理阶段时，往往就适用上述设计。因子在第一阶段应用到各批材料。之后这些批次在第二阶段处理过程中被拆分，在此过程中研究其他因子。第一阶段因子被视为**极难**更改，第二阶段因子被视为**难以**更改。第二个处理阶段后，可以将其他因子应用到实验单元。这些因子被视为**容易**更改。

在裂-裂区设计中，各批次被视为随机区组。由于在第二阶段处理过程中拆分了各批次，第二阶段因子嵌套在第一阶段因子内。

裂-裂区设计方案

Schoen (1999) 提供了与奶酪质量相关的裂 - 裂区设计的示例。这些因子在 Design Experiment 文件夹中的 Cheese Factors.jmp 数据表中提供。实验中包含三个处理阶段：

- 从农民处接收牛奶，并存储到大型储奶罐中。
- 大型储奶罐中的牛奶分装到用于加工凝乳的小型储奶罐中。
- 每个储奶罐中的凝乳都运送到压块机以加工奶酪块。

实验中包含以下检验：

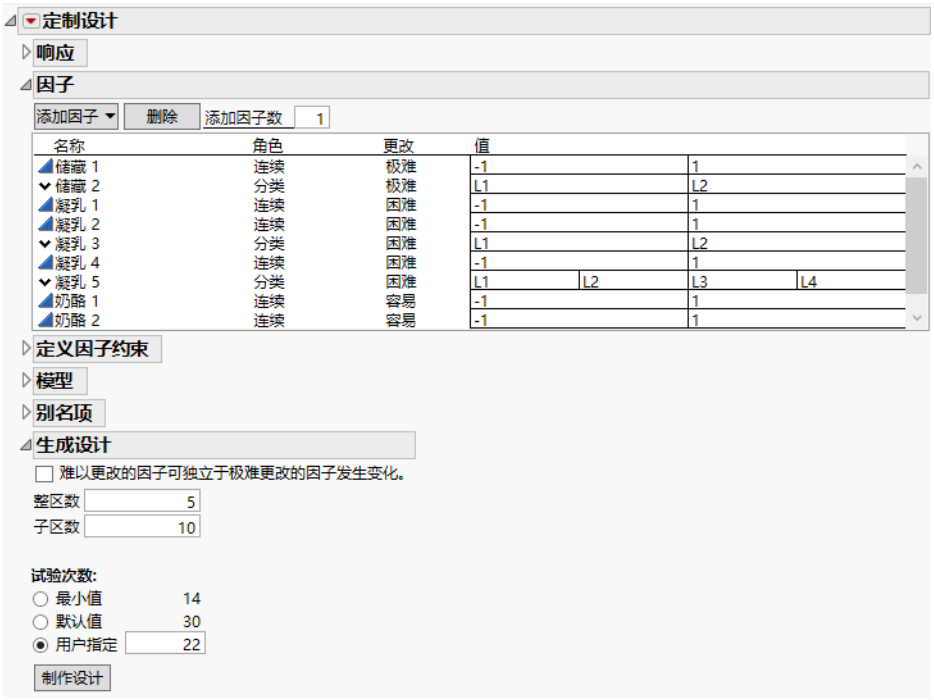
- 当牛奶存储在大型储奶罐中时应用的两个因子。
- 应用到小型凝乳加工罐的五个因子。
- 对凝乳加工罐中的奶酪块应用的三个因子。

请注意，对凝乳加工罐（子区）应用的因子水平嵌套在对储奶罐（整区）应用的因子水平之内。

“因子”分级显示项中的 “更改”设置为 “极难”、“困难”或 “容易”：

- 两个储奶罐因子设置为“极难”
- 五个凝乳加工罐因子设置为“困难”
- 可随机分配给奶酪的三个因子设置为“容易”

图 4.30 裂 - 裂区设计的 “因子”和 “生成设计”分级显示项



默认整区数为 5，默认子区数为 10。试验次数设置为 22。若仅使用 10 个子区，则设计没有足够的整区来估计子区方差。将子区数改为 11 并点击**制作设计**以查看 22 次试验设计。

图 4.31 奶酪的裂 - 裂区设计方案

设计

试验	整区	子区	储藏 1	储藏 2	凝乳 1	凝乳 2	凝乳 3	凝乳 4	凝乳 5	奶酪 1	奶酪 2	奶酪 3
1	1	1	-1	L2	1	1	L1	-1	L2	1	-1	L3
2	1	1	-1	L2	1	1	L1	-1	L2	-1	1	L1
3	1	2	-1	L2	-1	-1	L2	-1	L3	1	1	L1
4	1	2	-1	L2	-1	-1	L2	-1	L3	-1	-1	L3
5	1	3	-1	L2	1	-1	L2	1	L1	1	1	L3
6	2	4	1	L1	-1	1	L1	-1	L1	1	-1	L2
7	2	4	1	L1	-1	1	L1	-1	L1	1	1	L3
8	2	4	1	L1	-1	1	L1	-1	L1	-1	1	L1
9	2	5	1	L1	1	1	L1	1	L3	-1	-1	L1
10	2	5	1	L1	1	1	L1	1	L3	1	1	L2
11	3	6	1	L2	1	-1	L2	-1	L2	1	-1	L1
12	3	6	1	L2	1	-1	L2	-1	L2	-1	1	L2
13	3	7	1	L2	-1	1	L1	1	L4	1	1	L2
14	3	7	1	L2	-1	1	L1	1	L4	-1	-1	L3
15	4	8	1	L1	-1	1	L2	1	L2	1	1	L3
16	4	8	1	L1	-1	1	L2	1	L2	-1	-1	L1
17	4	9	1	L1	1	-1	L1	-1	L4	1	-1	L1
18	4	9	1	L1	1	-1	L1	-1	L4	-1	1	L2
19	5	10	-1	L1	1	1	L2	-1	L4	-1	-1	L2
20	5	10	-1	L1	1	1	L2	-1	L4	1	1	L1
21	5	11	-1	L1	-1	-1	L1	1	L2	-1	1	L3
22	5	11	-1	L1	-1	-1	L1	1	L2	1	-1	L2

五个整区对应于储藏因子；**储藏 1** 和**储藏 2**。储藏因子的设置在整区内保持不变。若若干连续的整区对于整区因子的设置都相同，则应在图之间重置该因子。例如，您应在第 10 次和第 11 次试验之间以及第 14 次和第 15 次试验之间重置**储藏 1** 的水平；在第 18 次和第 19 次试验之间重置**储藏 2** 的水平。即便指定的设置相同，也要求重置整区之间的因子，这样才能捕获整区变异。

11 个子区对应于凝乳因子。在某一子区内，凝乳因子的设置保持不变。**子区**的每个水平仅显示在**整区**的一个水平之内，这表明**子区**水平嵌套在**整区**水平之内。

奶酪因子的水平在不同试验之间随机变化。

裂-裂区设计的设置

在裂 - 裂区设计中，“因子”分级显示项包含 “更改”值设置为 “极难”和 “困难”的因子。该设计还可以包含 “更改”值设置为 “容易”的因子。创建了两个因子：

- 名为**整区**的因子代表“更改”值设置为“极难”的不变因子水平区组。
- 名为**子区**的因子代表“更改”值设置为“困难”的不变因子水平区组。
- 因子**子区**反映出“更改”值设置为“困难”的因子水平嵌套在“更改”值设置为“极难”的因子水平之内。
- “更改”值设置为“容易”的因子水平随机分配给子区内的单元。
- 因子**整区**和**子区**被视为保存到设计表的“模型”脚本中的随机效应。

有关详细信息，请参见第 84 页的“ “因子”分级显示项”和第 94 页的“设计结构选项”下的“更改”说明。

双因子裂区设计

双因子裂区（亦称带区或分裂区组）设计由两个裂区部分构成。在工业中，当诸多批次的材料或实验单元从一个处理阶段过渡到另一个处理阶段时，就适用上述设计。但是，在第一个处理阶段之后，可以将批次细分为子批次。将向这些子批次随机应用第二阶段处理因子。对于特定的第二阶段实验设置，分配至该设置的所有子批次可以同时处理。第二个处理阶段后，可以将其他因子应用到实验单元。

与裂-裂区设计相反，第二阶段因子**不嵌套**在第一阶段因子内。第一阶段之后，这些批次被细分并形成新的批次。因此，第一阶段因子和第二阶段因子将同时应用到整个批次。

尽管因子在这两个阶段可能同样难以更改，但为了区分这些因子，JMP 将第一阶段因子表示为**极难**更改，将第二阶段因子表示为**难以**更改。在第二个处理阶段之后应用到实验单元的其他因子被视为**易于**更改因子。

双因子裂区设计方案

Vivacqua and Bisgaard (2004) 描述的实验改善了电池中的开路电压。两个处理阶段受到关注：

- 第一阶段：连续组装过程

- 第二阶段：以 5 天为一个周期的固化过程

工程师想要研究六个两水平因子：

- 四个因子 X1、X2、X3 和 X4，应用于组装过程
- 两个因子 X5 和 X6，应用于固化过程。

所有因子都具有两个水平的完全析因设计将需要 $2^6 = 64$ 次试验，而且需要不可行的 64*5 = 320 天。此外，变动单个电池的组装条件实际上也不可行。不过，可以更改大批次（比如，包含 2000 粒电池的批次）的组装条件。

第一阶段和第二阶段因子均难以更改。在一定意义上，有两个裂区设计。不过，第一阶段实验的 2,000 粒电池的批次可细分为 500 粒电池为一批的四个子批次。这些子批次可随机分配到两个第二阶段因子的四个设置。分配到给定的一组固化条件的所有批次都可以同时处理。换言之，第一阶段和第二阶段因子**交叉**。

要区分第一阶段因子和第二阶段因子，您可以将第一阶段因子的“更改”指定为“极难”，将第二阶段因子的“更改”指定为“困难”（图 4.32）。此外，在“生成设计”下，请注意以下选项：**难以更改的因子可独立于极难更改的因子发生变化**。若未选中该选项，该设计将被视为裂-裂区设计，因子在两个水平上嵌套。选中该选项将创建双因子裂区设计。

图 4.32 双因子裂区设计的 “因子”和 “生成设计”分级显示项

整区默认数目为 7，子区默认数目为 14。点击**制作设计**可查看 28 次试验的设计。

图 4.33 电池的双因子裂区设计

设计

试验	整区	子区	X1	X2	X3	X4	X5	X6
1	1	1	1	1	1	-1	1	-1
2	1	2	1	1	1	-1	-1	1
3	1	3	1	1	1	-1	-1	-1
4	1	4	1	1	1	-1	1	1
5	2	5	-1	-1	-1	-1	-1	1
6	2	6	-1	-1	-1	-1	1	-1
7	2	7	-1	-1	-1	-1	-1	-1
8	2	8	-1	-1	-1	-1	1	1
9	3	9	1	1	-1	-1	-1	1
10	3	10	1	1	-1	-1	1	-1
11	3	11	1	1	-1	-1	-1	-1
12	3	12	1	1	-1	-1	1	1
13	4	13	1	-1	-1	1	-1	-1
14	4	14	1	-1	-1	1	1	1
15	4	1	1	-1	-1	1	1	-1
16	4	2	1	-1	-1	1	-1	1
17	5	3	-1	1	-1	1	-1	-1
18	5	4	-1	1	-1	1	1	1
19	5	5	-1	1	-1	1	-1	1
20	5	6	-1	1	-1	1	1	-1
21	6	7	1	-1	1	1	-1	-1
22	6	8	1	-1	1	1	1	1
23	6	9	1	-1	1	1	-1	1
24	6	10	1	-1	1	1	1	-1
25	7	11	-1	-1	1	-1	-1	-1
26	7	12	-1	-1	1	-1	1	1
27	7	13	-1	-1	1	-1	-1	-1
28	7	14	-1	-1	1	-1	1	1

七个整区对应于第一阶段因子 X1、X2、X3 和 X4。这些因子的设置在整区内保持不变。14 个子区对应于第二阶段因子 X5 和 X6。例如，试验 1 和 15（来自不同的整区）对应的子批次要接受相同的子区处理，其中，X5 设置为 1，X6 设置为 -1。

双因子裂区设计的设置

双因子裂区设计要求因子的“更改”值设置为“极难”和“困难”。如第 111 页的“裂-裂区设计的设置”中所述，将创建名为**整区**和**子区**的因子。不过，在双因子裂区设计中，**子区**中“更改”值设置为“困难”的因子水平并不嵌套在“更改”值设置为“极难”的因子水平之内。**整区**和**子区**同时被视为保存到设计表的“模型”脚本中的随机效应。

您需要确保因子**子区未嵌套**在因子**整区**内。在“生成设计”分级显示项中，选择选项**难以更改的因子可独立于极难更改的因子发生变化**（图 4.32）。请参见第 84 页的“更改”和第 94 页的“设计结构选项”。

有关创建裂区设计和分析实验数据的示例，请参见“定制设计示例”一章中第 181 页的“双因子裂区实验”。

包含难以更改的水平的协变量

“定制设计”平台支持您将协变量指定为难以更改。假定您具有可用于检验实验因子的各批材料的测量值。或是假定您具有可能入选参与检验实验因子的个人的测量值。这些测量值在实验之前已知，所以被视为协变量。

批次或个人对应于整区。您可能只希望在实验中使用其中一些整区。由于提供了关于协变量形式的整区的信息，设计中应以最优方式选取整区。

模型（如您在“模型”分级显示项中包含的项所指定）可以包含使用协变量和实验因子构造的交互作用和幂。

注意：若您将某个协变量因子的“更改”值设置为“困难”，所有其他协变量也将设置为“困难”。其余因子必须设置为“容易”。由于该算法需要结合使用行交换和坐标交换，即便是规模适中的设计也需要花一些时间才能生成。

包含难以更改的协变量的实验方案

Goos and Jones (2011, Chapter 9) 以及 Jones and Goos (2015) 对涉及各批次聚丙烯板的实验进行了讨论。根据若干变量确定的各种配方生产出大批的聚丙烯板。有些聚丙烯板立即使用，而其余的则存储起来供将来实验使用。这些存储起来的批次的成分是已知的。

有客户对聚丙烯板的配方有特定要求。将来实验涉及将气体等离子体的处理定制为客户要求的配方类型。构成变量被视为难以更改的协变量。气体等离子体处理因子可应用到具有给定配方的聚丙烯板的子批次。

最优设计确定要使用的批次（由协变量定义）、确定每批次要使用的聚丙烯板的数量，并提供气体等离子体水平的设置。请注意，最优批次数量和给定批次中的聚丙烯板的数量取决于协变量。

第 121 页的“定制设计示例”一章中提供了一个示例。

整区和子区数

JMP 会为“整区数”和“子区数”建议默认值。这些值通过启发方式来创建平衡设计，从而允许对“模型”分级显示项中指定的效应进行估计。

若您为“整区数”或“子区数”输入缺失值，JMP 会选择可将设计的 D 效率最大化的值。该算法使用“裂区方差比”选项中指定的值。请参见第 100 页的“高级选项 > 裂区方差比”。D 效率通过 $\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}$ 的行列式得出，其中，$\mathbf{V}^{-1}$ 是响应的方差矩阵的逆矩阵。请参见 Goos (2002)。

若您为“整区数”或“子区数”输入值，“定制设计”会尝试使所生成设计的最优性最大化。有关裂区设计的详细信息，请参见 Jones and Goos (2007)。有关与包含难以更改协变量的设计相关的详细信息，请参见 Jones and Goos (2015)。

最优性准则

使用依赖于最优性准则的搜索例程创建定制设计。本节提供有关用于创建定制设计的最优性选项的信息：

- 第 115 页的“D 最优性”
- 第 115 页的“Bayesian D 最优性”
- 第 116 页的“I 最优性”

- 第 117 页的“Bayesian I 最优性”
- 第 117 页的“A 最优性”
- 第 118 页的“别名最优性”

D 最优性

默认情况下，“定制设计”平台优化 D 最优性准则，除非使用“RSM”按钮创建了完全二次模型。这种情况下会构造 I 最优设计。

D 最优性准则将模型系数估计值的协方差矩阵的行列式最小化。由此断定 D 最优性着重于对效应的精确估计。该准则在以下情况下较为理想：

- 筛选设计
- 侧重于估计效应或检验显著性的实验
- 标识活跃因子是实验目标的设计

D 最优性准则依赖于假设模型。这造成了一定的局限性，因为真正模型的形式往往事先不可知。D 最优设计的试验优化假设模型的系数精度。在极端情况下，D 最优设计可能饱和，试验次数即参数个数，没有失拟自由度。

具体而言， D 最优设计最大化 D，其中， D 定义如下：

$$D = \det[\mathbf{X}'\mathbf{X}]$$

其中，$\mathbf{X}$ 是在第 101 页的“模拟响应”中定义的模型矩阵。

D 最优裂区设计最大化 D，其中， D 定义如下：

$$D = \det[\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}]$$

$\mathbf{V}^{-1}$ 是响应的区组对角协方差矩阵 (Goos 2002)。

由于 D 最优设计侧重于最小化系数的标准误差，所以可能不允许检查模型是否正确。例如，D 最优设计在一阶模型中不包含中心点。若存在可能活跃的项未包含在假设模型中，更好的方法是指定“若可能”项并使用 Bayesian D 最优设计。

Bayesian D 最优性

Bayesian D 最优性是经过修改的 D 最优性准则。Bayesian D 最优性准则在存在可能活跃的交互作用或非线性效应的情况下很有用。请参见 DuMouchel and Jones (1994) 以及 Jones et al (2008)。

Bayesian D 最优性可准确估计指定的一组模型参数。这些是您在“模型”分级显示项中将“可估计性”指定为“必需”的效应。但同时，Bayesian D 最优性有能力估计其他效应，通常为高阶效应（在试验规模允许的情况下）。这些是您在“模型”分级显示项中将“可估计性”指定为“若可能”的效应。Bayesian D 最优设计会在试验规模限制所允许的最大范围内，允许检测仅包含“必需”效应的模型中的不充分性。

在试验次数大于“必需”项个数但小于“必需”项与“若可能”项个数之和的情况下，Bayesian D 最优性准则最为有效。若出现这种情况，试验次数小于您要估计的参数个数。使用 Bayesian 设置中的先验信息允许精确估计所有“必需”项，同时提供检测和估计某些“若可能”项的能力。

要允许将有意义的先验分布应用到模型参数，需要对响应和因子的尺度进行统一以具备特定属性 (DuMouchel and Jones 1994, Section 2.2)。

考虑以下符号：

- **X** 是在第 101 页的“模拟响应”中定义的模型矩阵
- **K** 是具有以下值的对角矩阵：
  - $k = 0$ 代表“必需”项
  - $k = 1$ 代表涉及含有两个以上水平的分类因子的“若可能”主效应、幂和交互作用
  - $k = 4$ 代表其他所有“若可能”项

对“若可能”参数的向量施加的先验分布属于多元正态分布，均值向量为 **0** 并且对角协方差矩阵的对角线元素为 $1/k^2$。因此，值 k^2 是相应参数先验方差的倒数。

k 的值凭经验决定。对于“若可能”主效应、幂和具有多个自由度的交互作用，其先验方差为 1。其他“若可能”项的先验方差为 1/16。在 DuMouchel and Jones (1994) 采用的符号表示法中，$k = 1/\tau$。

要控制“若可能”项的权重，请从红色小三角菜单中选择**高级选项 > 先验参数方差**。请参见第 101 页的“高级选项 > 先验参数方差”。

参数的后验分布具有协方差矩阵 $(\mathbf{X}'\mathbf{X} + \mathbf{K}^2)^{-1}$。通过将后验协方差矩阵的逆矩阵的行列式最大化，可获得 Bayesian D 最优设计：

$$\det(\mathbf{X}'\mathbf{X} + \mathbf{K}^2)$$

I 最优性

I 最优设计可最小化设计空间上的预测平均方差。若您的主要实验目标不是估计系数，而是执行以下操作，则 I 最优性准则比 D 最优性更合适：

- 预测响应
- 确定最佳操作条件
- 确定设计空间中响应落在可接受范围内的区域

在以上情况下，响应的精确预测优先于参数的精确估计。

在设计空间中的 x_0 点处，相对于未知误差方差的预测方差可计算如下：

$$\text{方差}(\hat{\mathbf{Y}} | \mathbf{x}_0) = f(\mathbf{x}_0)'(\mathbf{X}'\mathbf{X})^{-1} f(\mathbf{x}_0)$$

其中，**X** 是在第 101 页的“模拟响应”中定义的模型矩阵。

I 最优设计在整个设计空间上将预测方差的积分 I 最小化，其中， I 计算如下：

$$I = \int_R f(\mathbf{x})'(\mathbf{X}'\mathbf{X})^{-1} f(\mathbf{x}) d\mathbf{x} = \text{迹}[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{M}]$$

此处 **M** 是矩量矩阵：

$$\mathbf{M} = \int_R f(\mathbf{x}) f(\mathbf{x})' d\mathbf{x}$$

请参见第 101 页的“模拟响应”。有关模拟响应的进一步信息，请参见 Goos and Jones (2011b)。

矩量矩阵不依赖设计，可提前计算。行向量 $f(\mathbf{x})'$ 包含 1，后跟与假设模型对应的效应。例如，对于含有两个连续因子的完全二次模型， $f(\mathbf{x})'$ 定义如下：

$$f(\mathbf{x})' = (1, x_1, x_2, x_1 x_2, x_1^2, x_2^2)$$

A 最优性

A 最优性准则将模型系数估计值的协方差矩阵的迹最小化。迹就是矩阵的主对角线元素的总和。A 最优设计可使回归系数的方差之和最小。

Bayesian I 最优性

Bayesian I 最优设计可将设计区域中“必需”和“若可能”项的平均预测方差最小化。

考虑以下符号：

- **X** 是在第 101 页的“模拟响应”中定义的模型矩阵
- **K** 是具有以下值的对角矩阵：
  - $k = 0$ 代表“必需”项
  - $k = 1$ 代表涉及含有两个以上水平的分类因子的“若可能”主效应、幂和交互作用
  - $k = 4$ 代表其他所有“若可能”项

对“若可能”参数的向量施加的先验分布属于多元正态分布，均值向量为 **0** 并且对角协方差矩阵的对角线元素为 $1/k^2$。（有关值 k 的详细信息，请参见第 115 页的“Bayesian D 最优性”。）

$\mathbf{x}_0$ 点处的预测值后验方差如下所示：

$$\text{方差}(\hat{\mathbf{Y}} | \mathbf{x}_0) = f(\mathbf{x}_0)'(\mathbf{X}'\mathbf{X} + \mathbf{K}^2)^{-1} f(\mathbf{x}_0)$$

Bayesian I 最优设计可将设计区域中的平均预测方差最小化，如下所示：

$$I_B = \text{迹}[(\mathbf{X}'\mathbf{X} + \mathbf{K}^2)^{-1}\mathbf{M}]$$

其中，**M** 是矩量矩阵。请参见第 101 页的“模拟响应”。

别名最优性

别名最优性的目的在于：将假设模型中的效应与该模型中不存在但可能活跃的效应之间的别名关系最小化。模型中不存在但可能需要关注的效应称为**别名效应**。有关别名最优设计的详细信息，请参见 Jones and Nachtsheim (2011b)。

具体而言，令 $\mathbf{X}_1$ 成为与假设模型中的项对应的模型矩阵，如第 101 页的“模拟响应”中定义。该设计定义与别名效应对应的模型。将别名效应的模型项矩阵表示为 $\mathbf{X}_2$。

别名矩阵是矩阵 **A**，定义如下：

$$\mathbf{A} = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2$$

别名矩阵中的条目代表与模型项估计值关联的偏倚度。请参见“技术细节”附录中第 735 页的“别名矩阵”，了解别名矩阵的由来。

A 中条目的平方和提供了偏倚的汇总测度。该平方和可以使用迹形式表示，如下所示：

$$\text{迹}(\mathbf{A}'\mathbf{A})$$

通常，减小迹准则的设计的 D 效率比 D 最优设计要低。因此，别名最优性寻求在受制于 D 效率下限的前提下将 $\mathbf{A}'\mathbf{A}$ 的迹最小化。有关 D 效率的定义，请参见第 114 页的“最优性准则”。D 效率的下限通过 D 效率权重给出，可在“高级选项”下指定该权重。请参见第 101 页的“高级选项 > D 效率权重”。

D 效率

用 **X** 表示包含 p 个参数的给定假设模型的设计（或模型）矩阵。有关模型矩阵的定义，请参见第 101 页的“模拟响应”。用 $\mathbf{X}_D$ 表示假设模型的 D 最优设计的模型矩阵。因此，**X** 给出的设计的 D 效率如下所示：

$$\text{D 效率} = \left[\frac{\det(\mathbf{X}'\mathbf{X})}{\det(\mathbf{X}_D'\mathbf{X}_D)}\right]^{1/p}$$

坐标交换算法

“定制设计”构造的设计用来优化若干最优性准则之一。（请参见第 114 页的“最优性准则”。）要优化准则，“定制设计”使用**坐标交换算法** (Meyer and Nachtsheim, 1995)。该算法最初在指定设计区域内为每个因子和每次试验随机选择值，以此构造初始设计。

假定您的研究需要连续因子（无因子约束）和一个主效应模型。迭代涉及检验模型矩阵的各个值，如下所示：

- 每个因子的当前值由其最极端的两个值来替代。

- 为这两个替代值计算最优性准则。
- 若其中某个值增大了最优性准则，则该值将替代旧值。

该过程持续进行，直到整个迭代不会出现任何替代。

将对该算法进行适当调整来解释多项式项、名义型因子和因子约束。

使用该过程获取的设计在一大类邻近设计范围内是最优的。但它只是**局部**最优。要改进查找全局最优设计的似然性，坐标交换算法将被重复很多次。Goos and Jones (2011, p. 36) 建议对除最不重要的设计情形之外的所有设计都至少采用 1,000 为随机开始数。开始设计数受“开始数”选项控制。请参见第 105 页的“开始数”。“定制设计”提供的设计可将所有构造的设计中的最优性准则最大化。

第 5 章

定制设计示例
执行满足您需要的实验

将“定制设计”平台用作您构造广泛实验设计的主要工具。您可以构造多种设计，并通过微调这些设计来满足您特定的实验需要和资源预算。

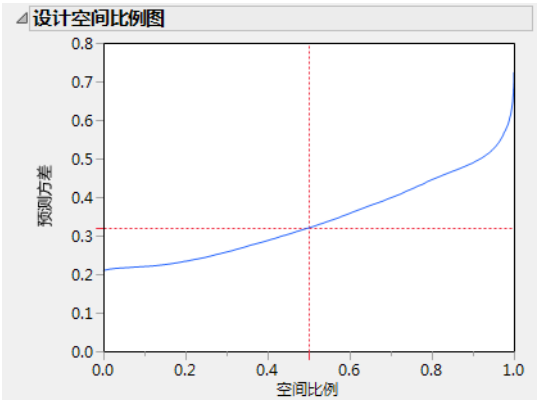
与“筛选”、“响应曲面”、“完全析因”和“混料设计”平台相比，“定制设计”能提供更多的选项和更强的控制。您构造的设计专为满足您的目标而创建。这样，您就不必再费力寻找某种经典设计，而其只是接近满足您的目标。

使用“定制设计”可构造的灵活专用设计包括：

- 筛选设计，包括超饱和筛选设计
- 响应曲面设计，包括含有分类因子的那些设计
- 混料设计，包括含有过程因子的混料设计以及混合混料设计
- 包含协变量或对线性时间趋势稳健的设计
- 固定和随机区组设计
- 裂区、裂-裂区和双因子裂区（带区）设计

在本章中，您将在“定制设计”平台中构造上述多数设计类型。多数情况下，您还可以分析实验结果。有关使用“定制设计”平台的帮助，请参见第 63 页的“定制设计”一章。

图 5.1 设计空间比例图



目录

筛选实验 123
仅估计主效应的设计 123
估计所有双因子交互作用的设计 126
避免主效应与双因子交互作用存在别名关系的设计 128
超饱合筛选设计 131
固定区组的设计 138
响应曲面实验 142
响应曲面设计 142
包含灵活分区组的响应曲面设计 147
D 最优和 I 最优响应曲面设计的比较 150
包含约束和分类因子的响应曲面设计 154
混料实验 160
包含非混料因子的混料设计 160
混合混料设计 164
包含协变量的实验 168
包含固定协变量的设计 168
包含难以更改的协变量的设计 171
带有线性时间趋势的设计 174
包含随机化限制的实验 177
裂区实验 177
双因子裂区实验 181
针对稳健过程和产品设计的实验 187

筛选实验

在研究某个过程的最初阶段，您需要标识可能影响您的一个或多个响应的因子列表。您关注的是标识**活跃**因子，也就是真正影响您的一个或多个响应的因子。**筛选设计**可帮助您确定哪些因子可能是活跃的。一旦标识了活跃因子，您就可以构造更复杂的设计（例如，响应曲面设计），以便对交互作用和曲率建模。

使用“定制设计”平台构造的筛选设计往往等同于“筛选设计”平台提供的经典设计。不过，“定制设计”可以为没有可用的经典筛选设计的情况构造设计。

“定制设计”平台使用 D 最优性准则或 Bayesian D 最优性准则构造筛选设计。D 最优性准则将模型系数估计值的协方差矩阵的行列式最小化。由此断定 D 最优性着重于对效应的精确估计。请参见 “定制设计”一章中第 114 页的 “最优性准则”。

- “仅估计主效应的设计”
- “估计所有双因子交互作用的设计”
- “避免主效应与双因子交互作用存在别名关系的设计”
- “超饱合筛选设计”
- “固定区组的设计”

仅估计主效应的设计

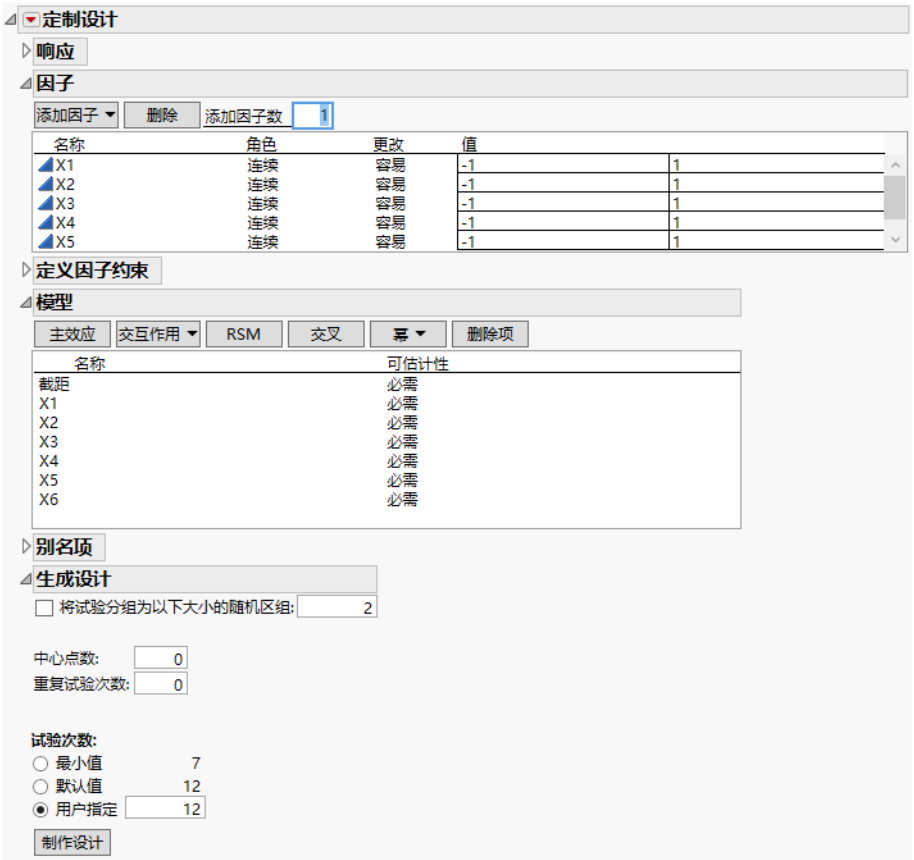
注意：有关仅限于主效应的筛选设计的详细信息，请参见第 263 页的“筛选设计”一章。

在本例中，您关注的是研究六个因子的主效应。您构造一个筛选设计，其中所有主效应都是正交的。不过，主效应与双因子交互作用存在别名关系。

1. 选择**实验设计 > 定制设计**。
2. 在 “因子”分级显示项中的**添加因子数**旁边键入 6。
3. 点击**添加因子 > 连续**。
4. 点击**继续**。

   “模型”分级显示项随即显示。其中仅包含“可估计性”指定为“必需”的主效应。这意味着所有主效应在生成的设计中都是可估计的。

图 5.2 显示 “模型” 分级显示项的 “定制设计” 窗口



保留默认值 12 次试验。

注意：在第 5 步中设置 “随机种子”，在第 6 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

5. （可选）点击 “定制设计” 红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
6. （可选）点击 “定制设计” 红色小三角，选择**开始数**，键入 1，然后点击**确定**。
7. 点击**制作设计**。

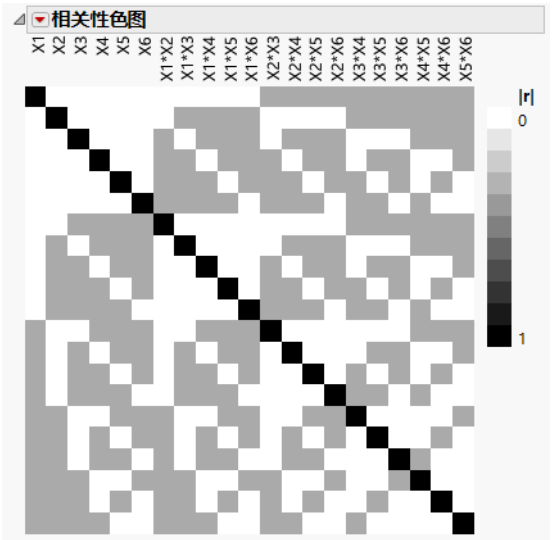
图 5.3 仅限于主效应的设计

设计

试验	X1	X2	X3	X4	X5	X6
1	-1	-1	-1	-1	-1	-1
2	1	-1	1	-1	-1	-1
3	1	1	-1	1	-1	-1
4	-1	1	1	-1	1	-1
5	1	-1	1	1	-1	1
6	1	1	-1	1	1	-1
7	1	1	1	-1	1	1
8	-1	1	1	1	-1	1
9	-1	-1	1	1	1	-1
10	-1	-1	-1	1	1	1
11	1	-1	-1	-1	1	1
12	-1	1	-1	-1	-1	1

8. 打开**设计评估 > 相关性色图**分级显示项。

图 5.4 相关性色图



请注意以下事项：

- 主效应由色图左上角的六个项表示。
- 白色对应六个主效应与其他主效应之间的相关性，白色指示二者之间为 0 相关。这意味着所有主效应都是正交的，可以独立于彼此进行估计。
- 方块中的浅灰色对应于一些双因子交互作用，浅灰色指示对应的效应之间相关。这意味着这些效应不能独立于其他效应进行估计。悬停在这些方块上方可查看精确的相关性。
- 请注意，没有任何效应彼此之间完全混杂。黑色方块（指示绝对相关性为 1）仅位于主对角线上。

9. 打开**设计评估 > 别名矩阵**分级显示项。

图 5.5 别名矩阵

别名矩阵

效应	X1*X2	X1*X3	X1*X4	X1*X5	X1*X6	X2*X3	X2*X4	X2*X5	X2*X6	X3*X4	X3*X5	X3*X6	X4*X5	X4*X6	X5*X6
截距	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X1	0	0	0	0	0	-0.33	0.333	0.333	-0.33	-0.33	-0.33	0.333	-0.33	-0.33	0.333
X2	0	-0.33	0.333	0.333	-0.33	0	0	0	0	-0.33	0.333	0.333	-0.33	-0.33	-0.33
X3	-0.33	0	-0.33	-0.33	0.333	0	-0.33	0.333	0.333	0	0	0	-0.33	0.333	-0.33
X4	0.333	-0.33	0	-0.33	-0.33	-0.33	0	-0.33	-0.33	0	-0.33	0.333	0	0	-0.33
X5	0.333	-0.33	-0.33	0	0.333	0.333	-0.33	0	-0.33	-0.33	0	-0.33	0	-0.33	0
X6	-0.33	0.333	-0.33	0.333	0	0.333	-0.33	-0.33	0	0.333	-0.33	0	-0.33	0	0

“别名矩阵”显示模型中主效应项的系数如何因可能活跃的双因子交互作用效应而产生偏倚。列标签标识交互作用。例如，在 X1 行中，列 X2*X3 的值为 -0.33；列 X2*X4 的值为 0.333。这意味着 X1 主效应的期望值是以下值之和：X1 主效应加上 X2*X3 效应的 -0.33 倍，加上 X2*X4 效应的 0.333 倍，一直加到 X1 行的最后一项。为使 X1 主效应的估计值有意义，您必须假定与 X1 的效应相比，这些交互作用的大小可忽略不计。

提示：别名矩阵是部分析因设计中的混杂模式的广义形式。

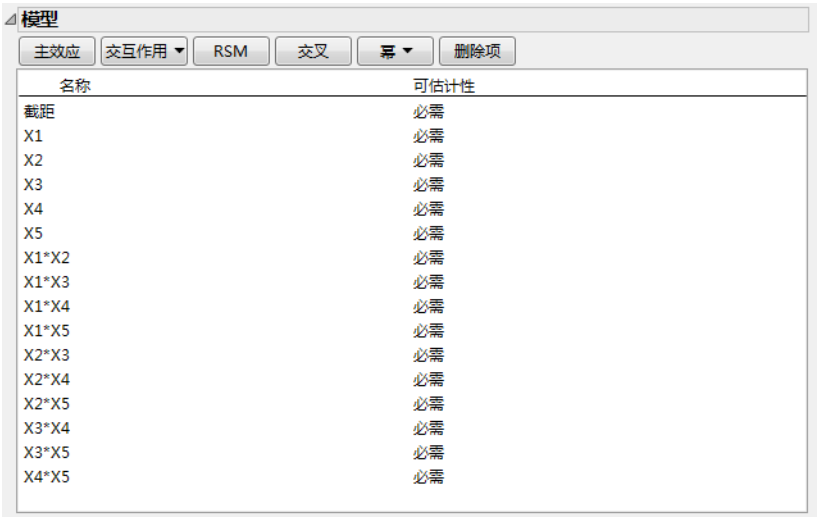
估计所有双因子交互作用的设计

图 5.5 中的“别名矩阵”显示了效应的部分别名关系。在其他情况下，主效应与双因子交互作用可能有完全别名关系或**混杂**。在以上两种情况下，强双因子交互作用都可能混淆仅限于主效应的实验的结果。为避免这种风险，需创建可分辨所有双因子交互作用的设计。

在本例中，您创建一个分辨率 V 筛选设计。双因子交互作用彼此正交，但它们与三因子交互作用混杂。

1. 选择**实验设计** > **定制设计**。
2. 在**添加因子数**旁边键入 5。
3. 点击**添加因子** > **连续**。
4. 点击**继续**。
5. 在 “模型”分级显示项中，选择**交互作用** > **二次**。

图 5.6 显示交互作用的 “模型” 分级显示项


模型

主效应 | 交互作用 | RSM | 交叉 | 幂 | 删除项

名称	可估计性
截距	必需
X1	必需
X2	必需
X3	必需
X4	必需
X5	必需
X1*X2	必需
X1*X3	必需
X1*X4	必需
X1*X5	必需
X2*X3	必需
X2*X4	必需
X2*X5	必需
X3*X4	必需
X3*X5	必需
X4*X5	必需

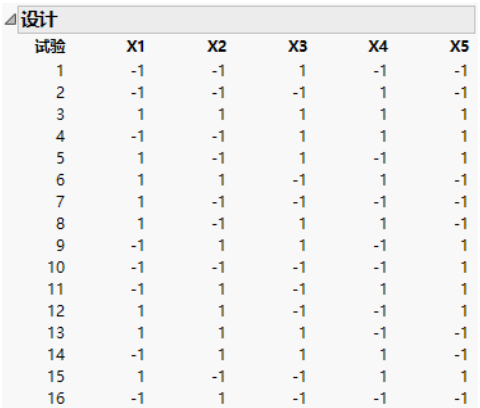
6. 点击**最小值**接受试验次数为 16。

注意：在第 7 步中设置 “随机种子”，在第 8 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

7. （可选）点击 “定制设计” 红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
8. （可选）点击 “定制设计” 红色小三角，选择**开始数**，键入 1，然后点击**确定**。
9. 点击**制作设计**。

图 5.7 显示了设计的试验。所有主效应和双因子交互作用都是可估计的，因为其 “可估计性” 已在 “模型” 分级显示项中指定为 “必需”（默认值）。

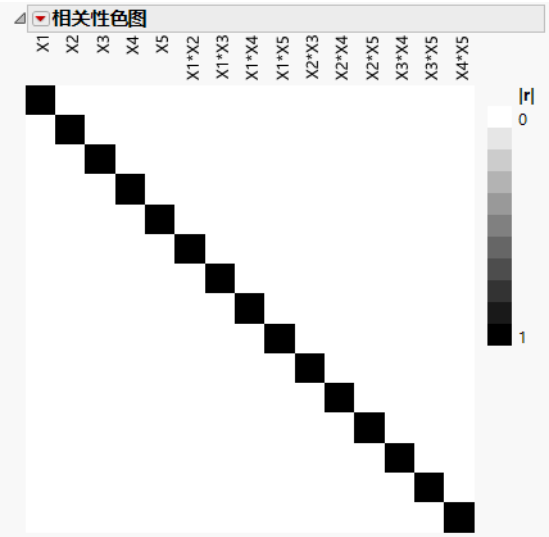
图 5.7 用于估计所有双因子交互作用的设计


设计

试验	X1	X2	X3	X4	X5
1	-1	-1	1	-1	-1
2	-1	-1	-1	1	-1
3	1	1	1	1	1
4	-1	-1	1	1	1
5	1	-1	1	-1	1
6	1	1	-1	1	-1
7	1	-1	-1	-1	-1
8	1	-1	1	1	-1
9	-1	1	1	-1	1
10	-1	-1	-1	-1	1
11	-1	1	-1	1	1
12	1	1	-1	-1	1
13	1	1	1	-1	-1
14	-1	1	1	1	-1
15	1	-1	-1	1	1
16	-1	1	-1	-1	-1

10. 打开**设计评估** > **相关性色图**分级显示项。

图 5.8 相关性色图



色图指示五个主效应与十个双因子交互作用全部两两正交。

避免主效应与双因子交互作用存在别名关系的设计

假定您主要关注的是估计六个连续因子的主效应。不过，您希望在估计时尽量减少主效应与可能活跃的双因子交互作用之间的别名关系。

您的预算仅允许进行 16 次试验。六个因子有 15 个可能的双因子交互作用。拟合常数、六个主效应和 15 个双因子交互作用所需的最小试验次数为 22。

在本例中，您需要在仅限主效应的 8 次试验设计（请参见第 123 页的“仅估计主效应的设计”）与能够拟合所有双因子交互作用的 22 次试验设计之间作出取舍。您将“别名最优性”用作实现目标的最优性准则。

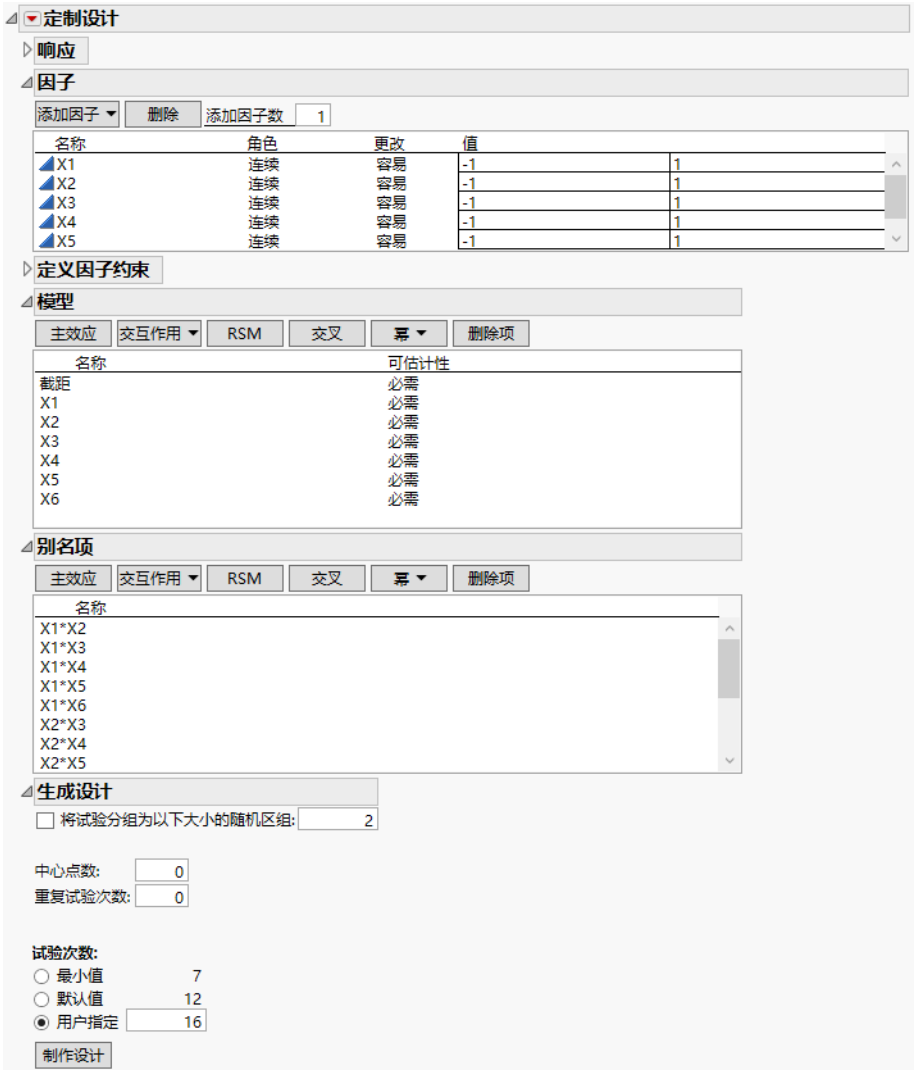
1. 选择**实验设计** > **定制设计**。
2. 在**添加因子数**旁边键入 6。
3. 点击**添加因子** > **连续**。
4. 点击**继续**。

   模型默认包含主效应项。这些项的默认可估计性为“必需”。在“别名项”分级显示项中，请注意添加了二阶交互作用。默认情况下，未包含在假设模型中的所有双因子交互作用都添加到“别名项”列表中。
5. 点击“定制设计”红色小三角并选择**最优性准则** > **制作别名最优设计**。

选择“制作别名最优设计”指示 JMP 生成在减少别名关系与提高 D 效率之间达到平衡的设计。请参见“定制设计”一章中第 118 页的“别名最优性”。

6. 点击**用户指定**，将试验次数改为 16。

图 5.9 因子、模型、别名项和试验次数



注意：在第 7 步中设置“随机种子”，在第 8 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

7. （可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
8. （可选）点击“定制设计”红色小三角，选择**开始数**，键入 10，然后点击**确定**。

9. 点击**制作设计**。
10. 打开**设计评估** > **别名矩阵**分级显示项。

图 5.10 别名矩阵

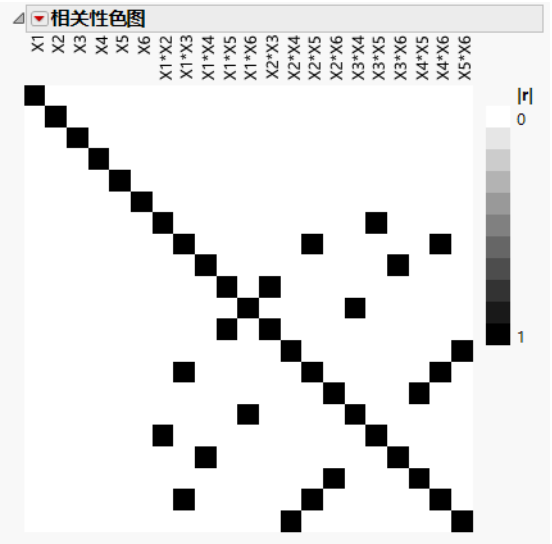
别名矩阵

效应	X1*X2	X1*X3	X1*X4	X1*X5	X1*X6	X2*X3	X2*X4	X2*X5	X2*X6	X3*X4	X3*X5	X3*X6	X4*X5	X4*X6	X5*X6
截距	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

所有行包含的都是零，这意味着“截距”和主效应项没有因任何双因子交互作用而产生偏倚。

11. 打开**设计评估** > **相关性色图**分级显示项。

图 5.11 相关性色图



“相关性色图”显示可以独立于双因子交互作用对主效应进行估计。不过，某些双因子交互作用与其他双因子交互作用之间具有完全别名关系。悬停在对角线之外的红色方块上，可检查哪些双因子交互作用混杂。

结果表明：这一特定设计是一个分辨率 IV 正交筛选设计。主效应相互之间以及与双因子交互作用之间均没有别名关系。不过，双因子交互作用与其他双因子交互作用之间具有完全别名关系。

超饱合筛选设计

一般来说，头脑风暴讨论后通常会发现几十个可能的活跃因子。与其精简列表而享受不到数据优势，不如使用超饱和设计。

在**饱和**设计中，试验次数等于模型项的个数。在**超饱和**设计中，模型项数超过了试验次数 (Lin, 1993)。超饱和设计可使用比因子数一半还少的试验次数来检查几十个因子。于是在因子数众多而试验成本高昂的情况下，超饱和设计便成了颇具吸引力的因子筛选选择。

或者，用组正交超饱和设计代替传统超饱和设计来更好地识别活跃效应。请参见第 645 页的“组正交超饱和设计”一章。

超饱和设计的局限性

超饱和设计有一些缺点：

- 若实验中的**活跃**因子个数超过了试验次数的一半，则这些因子可能无法识别。一般规则是：试验次数应至少是活跃因子个数的四倍之多。换言之：若您预计活跃因子可能多达五个，那么应计划执行至少 20 次试验。
- 超饱和设计分析目前尚无法简化为自动过程。不过，使用前向逐步回归较为合理。此外，“筛选”平台（**实验设计 > 经典 > 两水平筛选 > 拟合两水平筛选**）提供集成分析。

生成超饱和设计

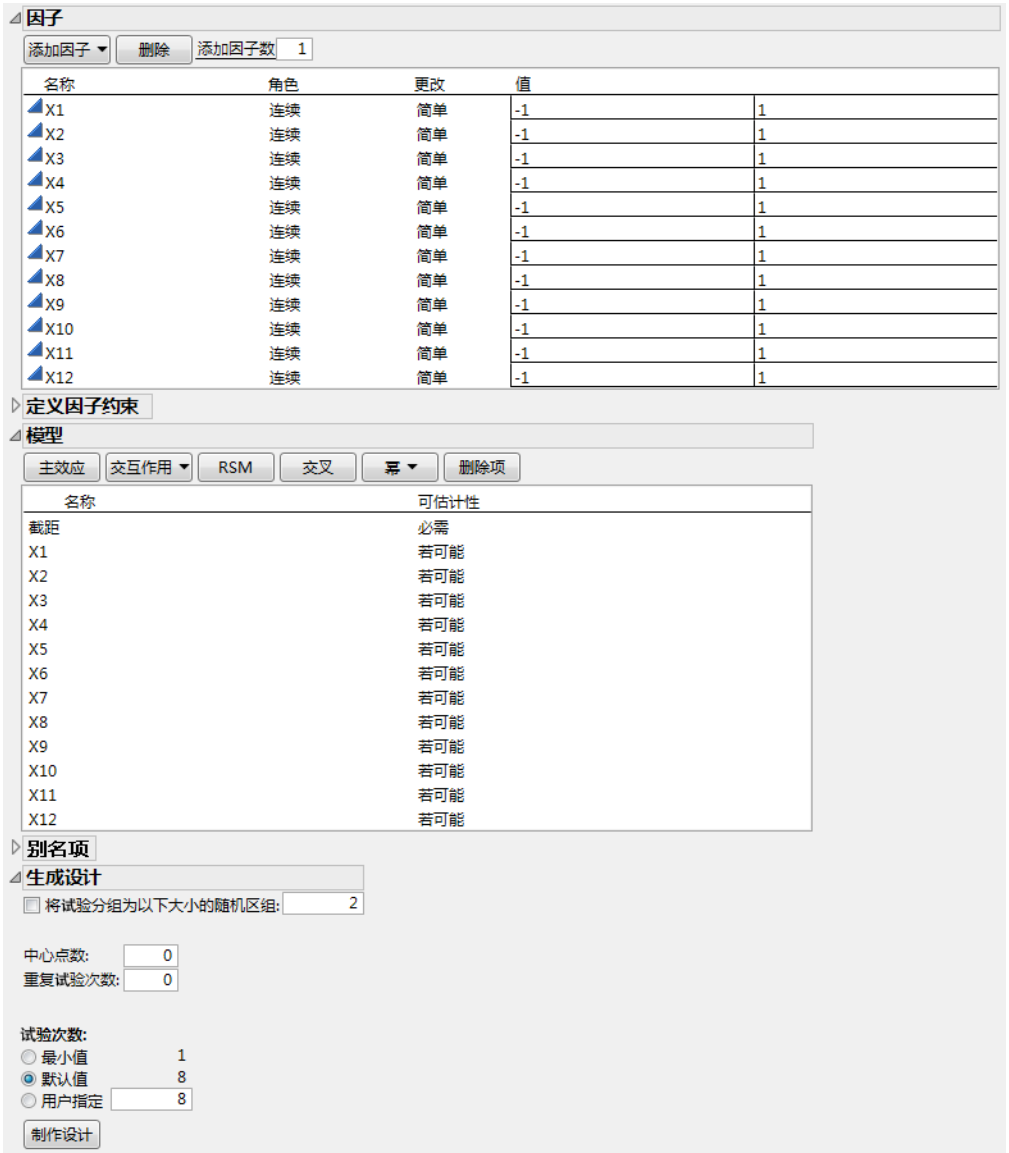
在本例中，您想要通过构造超饱和设计在 8 次试验中研究 12 个因子。要创建超饱和设计，需要将所有模型项（截距除外）的“可估计性”设置为“若可能”。

注意：本例仅用于演示目的。在任意超饱和设计中都应至少进行 14 次试验。若活跃因子多达四个，则很难解释 8 次试验设计的结果。请参见第 131 页的“超饱和设计的局限性”。

1. 选择**实验设计 > 定制设计**。
2. 在**添加因子数**旁边键入 12。
3. 点击**添加因子 > 连续**。
4. 点击**继续**。
5. 在 “模型”分级显示项中，选择除 “截距”之外的所有项。
6. 点击任意效应旁边的**必需**，将其更改为**若可能**。

   将效应设置为**若可能**可确保 JMP 使用 Bayes D 最优性准则获取设计。

图 5.12 因子、模型和试验次数



7. 在 “别名项” 分级显示项中，选择所有效应，然后点击**删除项**。
   这将确保只有主效应才会显示在 “相关性色图” 中。一旦创建设计即构造该图。
8. 点击 “定制设计” 红色小三角，然后选择**模拟响应**。
   该选项将生成显示在设计表中的随机响应。您需要使用这些响应来查看如何分析实验数据。
   保持 “试验次数” 设置为默认值 8。

注意：在第 9 步中设置 “随机种子”，在第 10 步中设置 “开始数”，这将会重现本例中显示的设计。自行构造设计时，这些步骤不是必需的。**Y 模拟**值可能与图 5.13 中所示有所不同。

9. （可选）点击 “定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
10. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 5，然后点击**确定**。
11. 点击**制作设计**。
12. 点击**制表**。

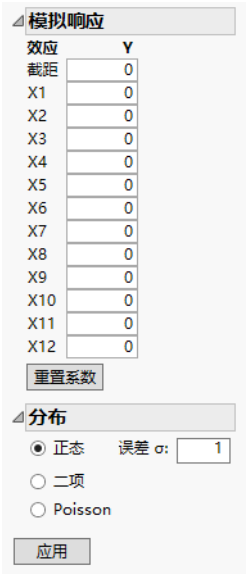
请不要关闭 “定制设计”窗口。在本例中，您稍后会返回到该窗口。

图 5.13 包含模拟响应的设计表

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	Y	Y 模拟
1	1	-1	1	-1	-1	1	-1	-1	-1	1	1	-1	-0.747583018	-0.747583
2	1	-1	-1	1	-1	-1	-1	1	1	1	-1	-1	-0.264373478	-0.2643735
3	-1	1	-1	-1	1	-1	-1	-1	1	1	1	1	0.7304848363	0.73048484
4	1	1	1	-1	1	-1	-1	1	-1	-1	-1	-1	-0.102964121	-0.1029641
5	-1	-1	-1	1	1	1	-1	1	-1	-1	1	1	-1.020102579	-1.0201026
6	-1	-1	1	1	1	-1	1	-1	-1	1	-1	-1	0.3251963898	0.32519639
7	1	-1	-1	-1	1	1	1	-1	1	-1	-1	-1	0.2557031962	0.2557032
8	1	1	-1	1	-1	-1	1	-1	-1	-1	1	1	-1.042077705	-1.0420777

响应列 Y 和 Y **模拟**的初始值具有相同的模拟值。这些值是来自 *N(0, s)* 分布的随机值。其中，*s* 是来自功效分析对话框的 RMSE，默认值为 1。 Y **模拟**值随着使用 “模拟响应”窗口中的参数值定义的模型随机生成的值而更新。 Y 列专用于运行实验后的真实响应。

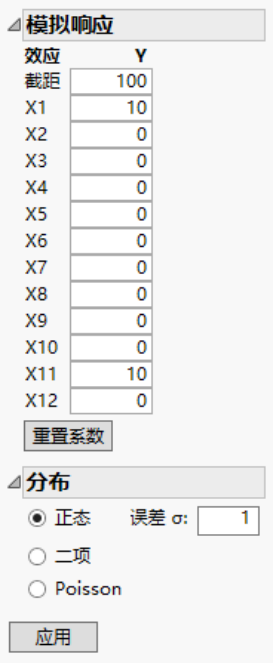
图 5.14 “模拟响应”窗口



“模拟响应”窗口显示所有项的默认系数均为 0，分布选择为 “正态”，“误差 σ”为 1。 Y 和 Y 模拟列中的值当前仅反映随机变异。

13. 更改 “模拟响应”窗口中的系数的值，如图 5.15 所示。

图 5.15 模拟响应的参数值



14. 点击应用。

Y 模拟列中的响应值随即更改。

注意：响应值随机生成。您的值不会与图 5.16 中的值精确匹配。

图 5.16 X1 和 X11 活跃时的 Y 模拟响应列

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	Y	Y 模拟
1	1	-1	1	-1	-1	1	-1	-1	-1	1	1	-1	1.8987814056	120.360314
2	1	-1	-1	1	-1	-1	-1	1	1	1	-1	-1	0.1543361111	99.8981745
3	-1	1	-1	-1	1	-1	-1	-1	1	1	1	1	1.4977493852	101.260989
4	1	1	1	-1	1	-1	-1	1	-1	-1	-1	-1	-0.638707667	100.686157
5	-1	-1	-1	1	1	1	-1	1	-1	-1	1	1	-1.239897281	98.5968775
6	-1	-1	1	1	1	-1	1	-1	-1	1	-1	-1	-1.032536321	80.722591
7	1	-1	-1	-1	1	1	1	-1	1	-1	-1	-1	0.9357329106	100.856697
8	1	1	-1	1	-1	-1	1	-1	-1	-1	1	1	1.0008548409	120.906062

在您的模拟中，您将 X1 和X11 指定为活跃因子；相对于误差变异，这两个因子的效应较大。出于此原因，您在分析数据时应将这两个因子标识为活跃因子。

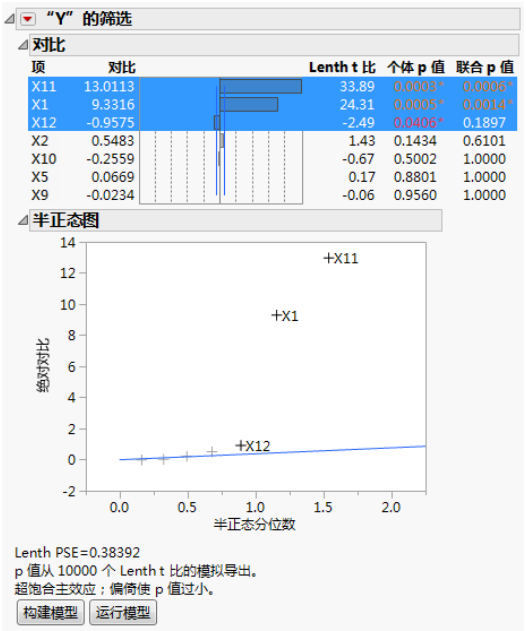
使用“筛选”平台分析超饱和设计

“筛选”平台提供了用于标识活跃因子的方法。使用“筛选”平台来分析设计表中的 Y 模拟值（图 5.16）。“筛选”平台位于**实验设计 > 经典**菜单下。

注意：您的数据表中的 Y 模拟值与本例中的值稍有不同。报表中的精确值将不同于此处显示的结果。

1. 选择**实验设计 > 经典 > 两水平筛选 > 拟合两水平筛选**。
2. 选择 Y 模拟并点击 Y。
3. 从 X1 一直选择到 X12，然后依次点击 X 和**确定**。

图 5.17 超饱和设计的筛选报表



因子 X1 和 X11 具有较大的对比以及 Lenth t 比值。此外，这两个因子的“联合 p 值”较小。在半正态图中， X1 和 X11 均落在距离线条较远的位置。“对比”报表和 “半正态图”报表指示 X1 和 X11 是活跃的。尽管 X12 的 “个体 p 值”小于 0.05，但其效应比 X1 和 X11 的效应小得多。

由于设计超饱和，*p* 值可能比所有效应都可估计的模型中的 p 值要小。这是因为效应估计值因其他可能活跃的主效应产生偏倚。在图 5.17 中，“构建模型”按钮正上方的注释提醒您这一可能性。

您可能还需要检查显示为活跃的效应是否与其他效应高度相关。若高度相关，则一个效应可能会掩盖另一个效应的真正显著性。图 5.19 中的色图显示效应之间的绝对相关性。

4. 点击**构建模型**。

   构造的模型仅包含效应 X1、 X11 和 X12。

5. 在 “模型规格”窗口中，点击**运行**。

图 5.18 模型的参数估计值

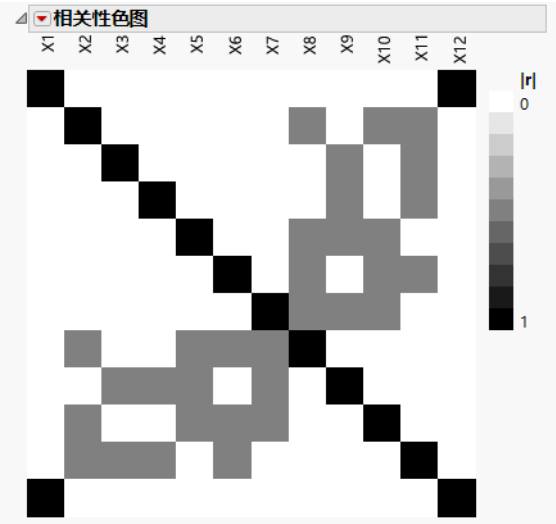
参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	100.12645	0.319502	313.38	<.0001*
X11	11.347787	0.360447	31.48	<.0001*
X1	9.8209584	0.319502	30.74	<.0001*
X12	-1.1329	0.360447	-3.14	0.0347*

请注意， X11 和 X1 的参数估计值接近您用来模拟模型的理论值。请参见图 5.15，您在该窗口中指定了模型的 X1 = 10， X11 = 10。因子 X12 的显著性是假阳性的一个示例。

6. 在 “定制设计”窗口中，打开**设计评估 > 相关性色图**分级显示项。

图 5.19 “相关性色图”分级显示项



悬停在单元格上方可查看绝对相关性。请注意，X1 与其他主效应（X4、X5、X7）的相关性高达 0.5。（图 5.19 使用 JMP 默认颜色。）

使用逐步回归分析超饱和设计

逐步回归是另一种标识活跃因子的方式。图 5.16 中的设计表包含三个脚本。“模型”脚本使用“拟合模型”平台中的逐步回归分析数据。

注意：您的数据表中的 Y **模拟**值与本例中的值稍有不同。报表中的精确值将不同于此处显示的结果。

1. 在设计表的“表”面板中，点击**模型**脚本旁边的绿色小三角。
2. 将**特质**从**标准最小二乘**改为**逐步**。
3. 点击**运行**。
4. 在 “‘Y’ 的逐步拟合”报表中，将**停止规则**改为**最小 AICc**。

   对于设计的实验，BIC 通常是比 AICc 更为宽松的停止规则，因为 BIC 往往允许在模型中包含不活跃效应。
5. 点击**执行**。

图 5.20 超饱和设计的逐步回归

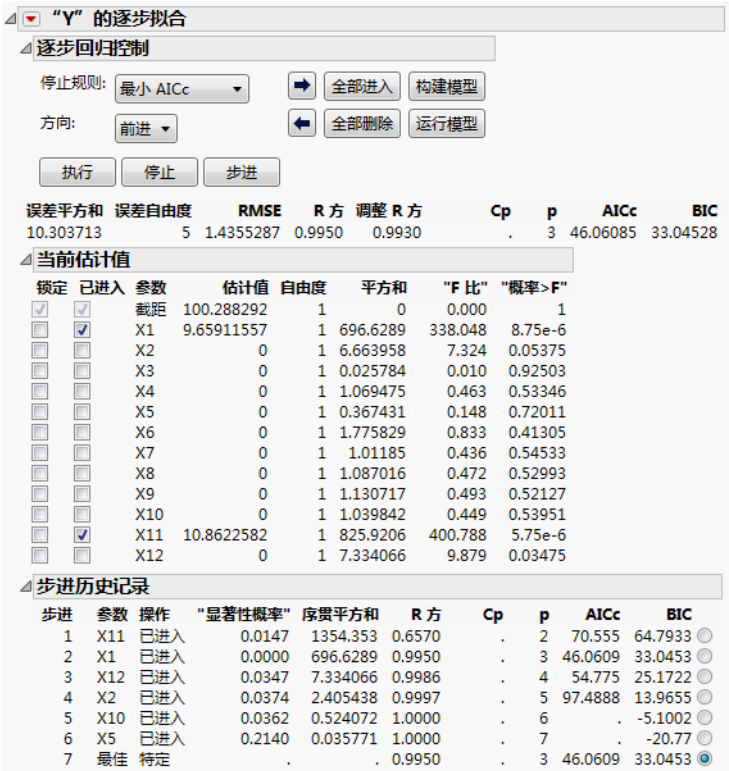


图 5.20 显示选定模型包含两个活跃因子：X1 和 X11。步进历史记录显示在报表的底部。请记住， X1 和 X11 与其他因子的相关性可能掩盖其他活跃因子的效应 （图 5.19）。

注意：本例定义了两个较大的主效应并将其他效应设置为零。现实中不太可能有这种区分明显的效应。

固定区组的设计

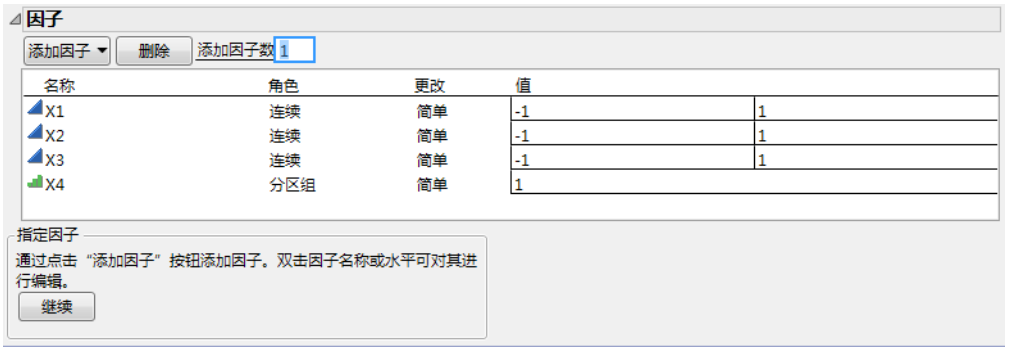
传统的筛选设计要求区组大小为 2 的幂次方。不过，“定制设计”平台可创建包含任意大小固定区组的设计。

假定您想要研究三个因子。您每天只能执行三次试验，并且期望每天都会有相当大的变异。因此，您需要将设计划分区组到若干天内。此外，在本研究中，您关注的是估计所有双因子交互作用。在本例中，您构造的设计中每个区组有三次试验。

1. 选择**实验设计** > **定制设计**。
2. 在“因子”分级显示项中的**添加因子数**旁边键入 3。
3. 点击**添加因子** > **连续**。
4. 点击**添加因子** > **分区组** > **每个区组 3 次试验**。

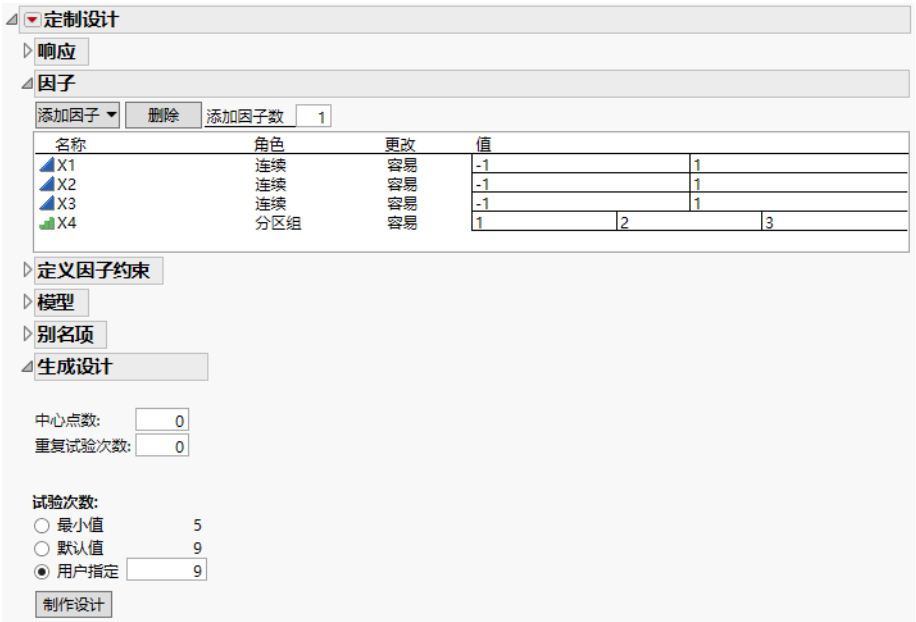
   分区组因子 X4 在**值**的下方仅显示一个水平。这是因为试验规模在此时还是未知的。

图 5.21 为 X4 显示一个区组的“因子”分级显示项



5. 点击**继续**。

图 5.22 为 X4 显示三个区组的 “因子” 分级显示项

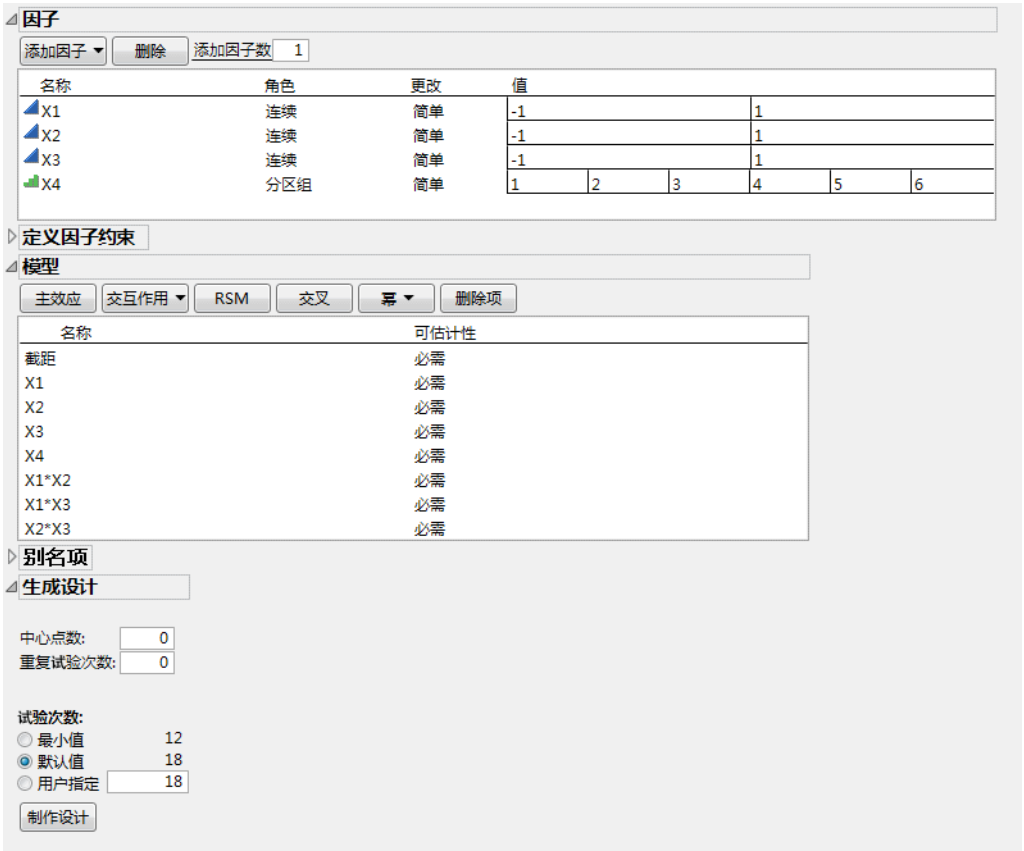


“因子” 分级显示项现在显示适当的区组数，区组数是通过用默认试验规模除以每个区组的试验次数计算得出。对于本例，默认样本大小 9 需要三个区组。“因子” 分级显示项现在显示 X4 有三个值，指示这三个区组。

注意：若指定不同的试验次数，“因子” 分级显示项会更新以显示该分区组因子的正确值数。

6. 在 “因子” 分级显示项中选择三个连续因子：X1、 X2 和 X3。
7. 在 “模型” 分级显示项中，点击**交互作用 > 二次**。

图 5.23 为 X4 显示六个区组的 “因子” 分级显示项



“试验次数”面板现在显示 18 是默认试验规模。“因子”分级显示项现在显示 X4 的 6 个值，指示 6 个区组。

注意：在第 8 步中设置 “随机种子”，在第 9 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

8. （可选）点击 “定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
9. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 5，然后点击**确定**。
10. 点击**制作设计**。

图 5.24 固定区组设计

设计

试验	X1	X2	X3	X4
1	1	-1	-1	1
2	-1	-1	-1	2
3	1	1	-1	3
4	-1	-1	-1	4
5	1	1	1	5
6	-1	-1	1	6
7	-1	1	-1	1
8	1	1	-1	2
9	-1	1	1	3
10	1	-1	1	4
11	-1	-1	1	5
12	-1	1	-1	6
13	-1	-1	1	1
14	-1	1	1	2
15	1	-1	1	3
16	-1	1	1	4
17	1	-1	-1	5
18	1	1	1	6

在该设计中，查看分区组因子 X4。六个区组已表示出来。在执行实验时，每天都要执行三次试验，您将在第一天执行 X4 = 1 对应的三次试验，将在第二天执行 X4 = 2 对应的三次试验，依此类推。所以您希望设计表在区组内随机执行试验。在“输出选项”面板中，已经为“试验顺序”选定 “区组内随机”选项。

11. 点击**制表**。

图 5.25 固定区组设计的设计表

	X1	X2	X3	X4	Y
1	1	-1	-1	1	•
2	-1	1	-1	1	•
3	-1	-1	1	1	•
4	-1	1	1	2	•
5	1	1	-1	2	•
6	-1	-1	-1	2	•
7	1	-1	1	3	•
8	-1	1	1	3	•
9	1	1	-1	3	•
10	-1	1	1	4	•
11	1	-1	1	4	•
12	-1	-1	-1	4	•
13	1	-1	-1	5	•
14	-1	-1	1	5	•
15	1	1	1	5	•
16	-1	-1	1	6	•
17	-1	1	-1	6	•
18	1	1	1	6	•

设计表中的行按照每天的试验分组。该设计支持您估计区组效应、所有主效应和双因子交互作用。

响应曲面实验

响应曲面实验通常涉及已标识为活跃的少量连续因子（通常为 2 到 8 个）。响应曲面实验的主要目标是产生因子与响应之间关系的预测模型。通常，您使用预测模型来找到对过程更好的操作设置。出于此原因，用于响应曲面实验的假设模型通常为二次模型。

由于筛选设计侧重于标识活跃效应，衡量其质量的一个测度就是系数相对方差的大小。您希望这些相对方差尽量小。D 最优性准则可处理这些相对方差。

在响应曲面实验中，因子范围内的预测方差比系数方差更重要。设计空间上的预测方差可通过 I 最优性准则来处理。与 D 最优设计相比，I 最优设计往往会在设计空间的极点放置较少的试验。有关 D 最优性和 I 最优性的详细信息，请参见“定制设计”一章中第 114 页的“最优性准则”。

默认情况下，“定制设计”中的“最优性准则”使用“推荐”选项。只要使用“模型”分级显示项中的“RSM”按钮添加二次效应，“定制设计”就会将 I 最优性准则用作“推荐”准则。如若不然，“定制设计”会将 D 最优性准则用作“推荐”准则。请参见“定制设计”一章中第 114 页的“最优性准则”。

- “响应曲面设计”
- “包含灵活分区组的响应曲面设计”
- “D 最优和 I 最优响应曲面设计的比较”
- “包含约束和分类因子的响应曲面设计”

响应曲面设计

本例包含以下部分：

- 第 142 页的“构造响应曲面设计”
- 第 145 页的“分析实验结果”

构造响应曲面设计

为您标识为活跃的三个连续因子构造响应曲面设计。您想要找到可令响应 (Y) 保持在规格内的过程设置。Y 的下规格限和上规格限分别为 54 和 56，目标为 55。

1. 选择**实验设计** > **定制设计**。
2. 在“响应”分级显示项中，点击**最大化**并选择**匹配目标值**。
3. 在**下限**中键入 54，在**上限**中键入 56。
4. 保留**重要性**为空。

   由于只有一个响应，“重要性”值默认设置为 1。
5. 在**添加因子数**旁边键入 3。
6. 点击**添加因子** > **连续**。

这将添加三个连续因子：X1、 X2 和 X3。

7. 点击**继续**。
8. 在 “模型” 分级显示项中，点击 RSM 按钮。

   这将向模型添加二次项和交互作用项。同时还会将 “推荐” 最优性准则的值设置为 I 最优性。一旦点击 “制作设计”，您就可以在 “设计诊断” 分级显示项中验证该设置。

   保持默认试验次数设置为 16。

   注意：在第 9 步中设置 “随机种子”，在第 10 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

9. （可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 929281409，然后点击**确定**。
10. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 40，然后点击**确定**。
11. 点击**制作设计**。

    “设计” 分级显示项显示设计。

图 5.26 RSM 设计

设计

试验	X1	X2	X3
1	0	-1	0
2	1	-1	1
3	0	0	0
4	-1	1	1
5	-1	0	0
6	1	1	1
7	-1	1	-1
8	-1	-1	-1
9	0	0	1
10	1	0	0
11	0	1	0
12	0	0	0
13	1	1	-1
14	0	0	-1
15	-1	-1	1
16	1	-1	-1

为估计二次效应，响应曲面设计为每个因子使用三个水平。请注意，图 5.26 中的设计是具有两个中心点的面心 “中心复合设计”。

12. 打开**设计评估 > 设计诊断**分级显示项。

图 5.27 “设计诊断” 分级显示项

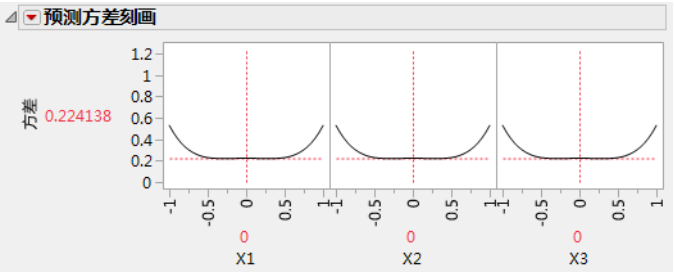
设计诊断

I 最优设计	
D 效率	42.99905
G 效率	88.62743
A 效率	30.68134
预测平均方差	0.340517
设计创建时间（秒）	0.05

“设计诊断” 分级显示项中的第一行标识所使用的最优性准则。该设计为 I 最优。

13. 打开**设计评估 > 预测方差刻画**分级显示项。

图 5.28 预测方差刻画

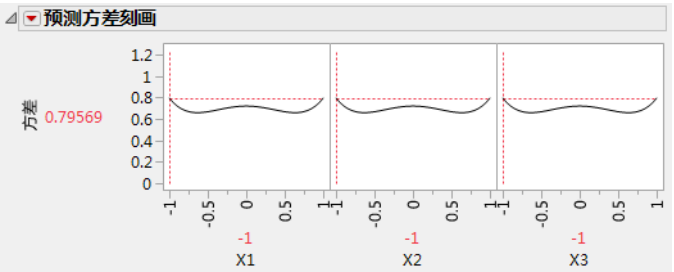


垂直轴显示响应期望值的相对预测方差。**相对预测方差**是用预测方差除以误差方差所得的值。当相对预测方差为 1 时，其绝对方差等于回归模型的误差方差。

刻画器显示设计空间上的相对预测方差的值。您可以移动滑块以探索预测方差的行为。预测方差在设计空间的中心最小。对于 -0.5 到 0.5 之间的因子设置，预测方差基本保持不变，其值只是稍微大于 0.2。设置越趋向设计空间的边界，预测方差的值越大。

14. 点击 “预测方差刻画”红色小三角并选择**最大化方差**。

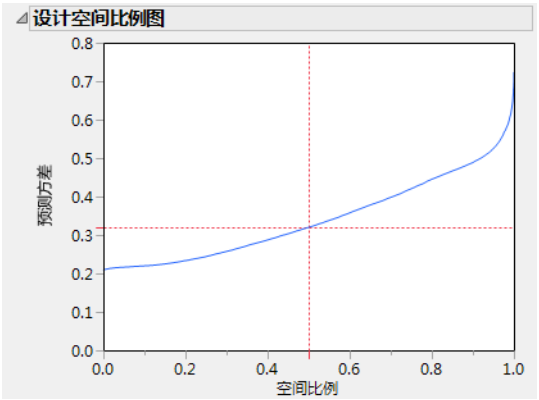
图 5.29 相对方差最大化之后的预测方差刻画



刻画器显示相对预测方差的最大值为 0.79569。

15. 打开**设计评估 > 设计空间比例图**分级显示项。

图 5.30 设计空间比例图



图中蓝色曲线显示相对预测方差是设计空间比例的函数。红色虚十字准线指示：对于 50% 的设计空间，预测方差大约为 0.32 或更小。使用十字准线工具绘制其他推断。例如，若“空间比例”为 0.95，“预测方差”约为 0.52。这意味着：对于 95% 的设计空间，相对预测方差低于 0.52。

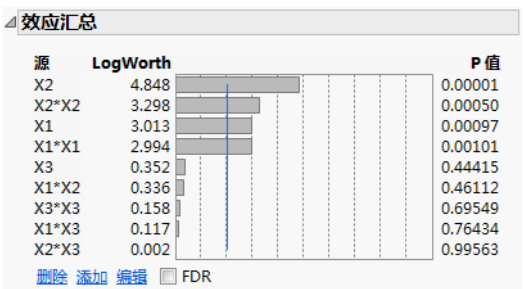
分析实验结果

Custom RSM.jmp 样本数据表包含实验结果。使用“模型”脚本可打开“拟合模型”窗口，其中包含您在“实验设计”窗口的“模型”分级显示项中指定的所有效应。该脚本通过“定制设计”平台另存到数据表中。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Custom RSM.jmp。
2. 在“表”面板中，点击**模型**脚本旁边的绿色小三角。
3. 点击**运行**。

   “效应汇总”报表显示模型中每个效应的 LogWorth 和 P 值。图中的蓝色竖线设置在值 2 处。 LogWorth 值超过 2 的效应在 0.01 水平下显著。

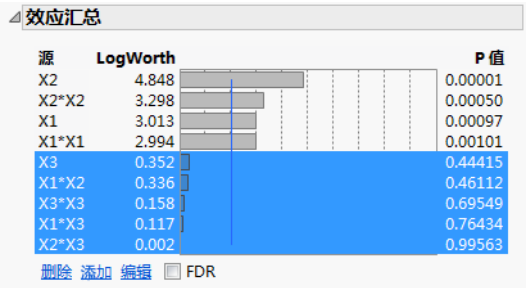
图 5.31 “效应汇总”报表



该报表显示 X1、X2、X1*X1 和 X2*X2 在 0.01 水平下显著。其他任何效应即便是在 0.10 水平下也不显著。可通过删除这些不显著的效应来简化模型。

4. 在 “效应汇总”报表中，选择 X3、 X1*X2、 X3*X3、 X1*X3 和 X2*X3。

图 5.32 已选定不显著效应的 “效应汇总”报表



5. 点击**删除**。

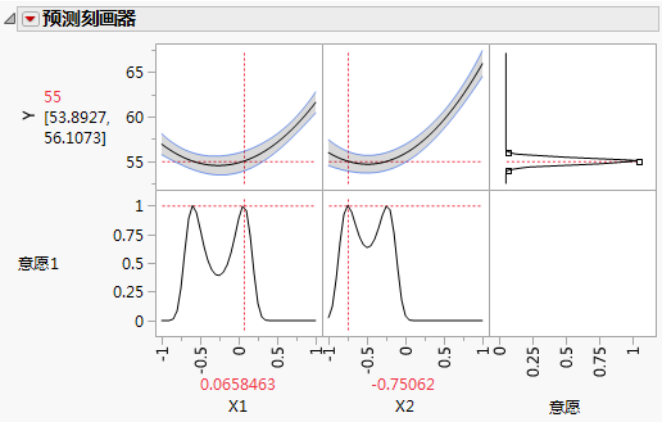
“拟合最小二乘法”报表随之更新，以显示仅包含显著效应的模型：X1、 X2、 X1*X1 和 X2*X2。

使用 “预测刻画器”（位于 “拟合最小二乘法”窗口底部）探索预测响应 (Y) 如何随着您更改因子 X1 和 X2 而变化。请注意在 X1 和 X2 各取值下 Y 的二次特性。

回想一下您在 “定制设计”窗口的 “响应”分级显示项中为 Y 输入了响应限。因此，“响应限”列属性附加到设计表中的 Y 列。 Y 的意愿函数 （位于右上图中）基于 “响应限”列属性中包含的信息。 JMP 使用该函数计算作为 X1 和 X2 设置的函数的意愿值。意愿函数的迹线显示在位于底部行的图中。

6. 点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

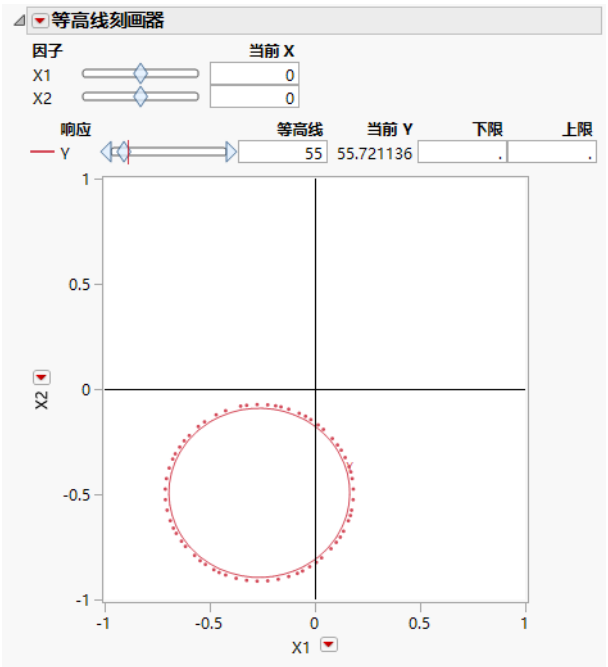
图 5.33 意愿最大化后的预测刻画器



预测响应在 X1 和 X2 上方显示为红色的过程设置处实现了目标值 55。图 5.33 显示：有个接近 –0.65 的 X1 值在 X2 = -0.75062 时也实现了预测值 55。事实上，预测刻画器可能显示与最大化意愿的设置不同的设置。这是因为 X1 和 X2 的许多设置都可以实现预测响应 55。

7. 点击 “响应 Y” 红色小三角并选择**因子刻画 > 等高线刻画器**。
8. 在 “等高线刻画器” 报表中，键入 55 作为**等高线**的值。

图 5.34 等高线刻画器



与红色等高线对应的 X1 和 X2 的设置具有预测响应值 55。您最好从这些过程设置中基于成本效益进行选择。

包含灵活分区组的响应曲面设计

优化某个过程时，您可能需要在实验中包含定性因子和连续因子。您可能需要按照批次或天之类的定性因子来分区组，或是包含机器或传送机制之类的定性因子。但 “响应曲面设计” 平台仅支持连续因子。要获得包含定性因子的响应曲面设计，您可以对该因子每个水平重复该设计。不过，这样做效率较低。“定制设计” 平台可使用较少的试验构造最优设计。

在本例中，您需要构造一个响应曲面设计，以适应两个连续因子和一个分区组因子 （每个区组四次试验）的需要。您可以采用类似方式包含分类或离散数值因子。

1. 选择**实验设计 > 定制设计**。
2. 在**添加因子数**旁边键入 2。

3. 点击**添加因子** > **连续**。
4. 点击**添加因子** > **分区组** > **每个区组 4 次试验**。

   请注意，“值”下仅显示一个水平。这是因为在尚未确定试验次数之前无法确定区组数。

图 5.35 包含两个连续因子和一个分区组因子的 “因子”分级显示项

5. 点击**继续**。

   默认试验次数为 12。“因子”分级显示项更新以显示分区组因子 X3 的三个水平。由于您要求 X3 每个区组执行四次试验， 12 次试验需要三个区组。
6. 点击 RSM。

   将有一个说明性的 “JMP 警示”窗口提示您分区组因子不能显示在交互作用或二次项中。JMP 只会将适当的 RSM 项添加到列表中。
7. 点击**确定**取消显示该消息。

   X1 和 X2 的二次项和交互作用项随即添加至模型。由于已经添加了 RSM 项，“推荐”最优性准则将从 D 最优改为 I 最优。稍后您将在 “设计诊断”分级显示项中看到这一变化。

图 5.36 包含响应曲面效应的 “模型”分级显示项

注意：在第 8 步中设置 “随机种子”，在第 9 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

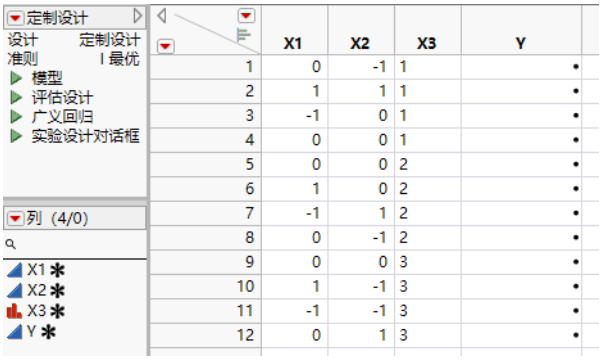
8. （可选）点击 “定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
9. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 5，然后点击**确定**。
10. 点击**制作设计**。
11. 打开**设计评估** > **设计诊断**分级显示项。

图 5.37 “设计诊断”分级显示项

“设计诊断”分级显示项中的第一行标识所使用的最优性准则。该设计为 I 最优。

12. 点击**制表**。

图 5.38 包含分区组因子的设计表

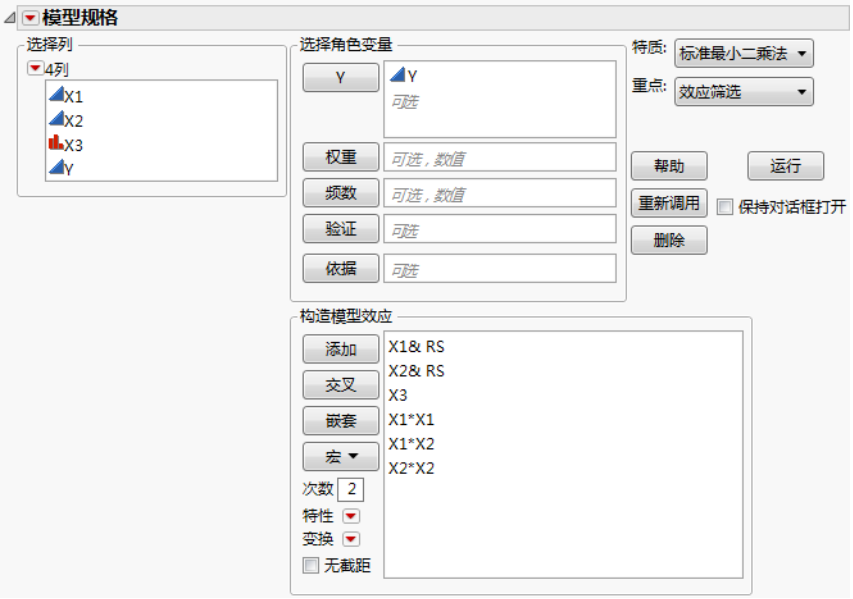


	X1	X2	X3	Y
1	0	-1	1	•
2	1	1	1	•
3	-1	0	1	•
4	0	0	1	•
5	0	0	2	•
6	1	0	2	•
7	-1	1	2	•
8	0	-1	2	•
9	0	0	3	•
10	1	-1	3	•
11	-1	-1	3	•
12	0	1	3	•

由于默认的“试验顺序”为“区组内随机”，分区组因子 (X3) 的水平经过排序。

13. 在设计表的“表”面板中，点击**模型**脚本旁边的绿色小三角。

图 5.39 拟合模型窗口



请注意以下事项：

- 分区组因子 (X3) 作为效应输入。
- 未包含涉及 X3 的任何交互作用。
- 其他五个效应定义 X1 和 X2 的响应曲面模型。

D 最优和 I 最优响应曲面设计的比较

在本例中，您将要探索在双因子响应曲面设计的情形下 I 最优性与 D 最优性之间的区别。

I 最优设计

1. 选择**实验设计** > **定制设计**。
2. 在**添加因子数**旁边键入 2。
3. 点击**添加因子** > **连续**。
4. 点击**继续**。
5. 点击 RSM。

   X1 和 X2 的二次项和交互作用项随即添加至模型。由于已经添加了 RSM 项，“推荐”最优性准则将从 D 最优改为 I 最优。稍后您将在 “设计诊断”分级显示项中看到这一变化。

注意：在第 6 步中设置“随机种子”，在第 7 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

6. （可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 383570403，然后点击**确定**。
7. （可选）点击“定制设计”红色小三角，选择**开始数**，键入 8，然后点击**确定**。
8. 点击**制作设计**。

图 5.40 I 最优设计

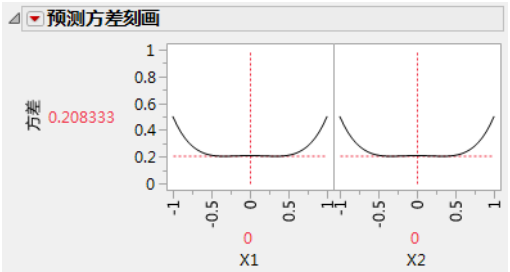
设计

试验	X1	X2
1	0	0
2	1	0
3	-1	-1
4	0	0
5	0	1
6	-1	0
7	0	0
8	-1	1
9	1	-1
10	0	0
11	0	-1
12	1	1

在该 I 最优设计中，试验 1、4、7 和 10 位于中心点（X1 = 0 且 X2 = 0）。与 D 最优设计相比，I 最优设计往往会在设计空间的中心放置较多的试验（因而在极点放置较少的试验）。您可以将该设计与图 5.42 所示的 D 最优设计加以比较。

9. 打开**设计评估 > 预测方差刻画**分级显示项。

图 5.41 I 最优模型的预测方差刻画



期望响应的相对预测方差在设计空间的中心最小。

10. 打开**设计空间比例图**分级显示项。

“设计空间比例图”显示在图 5.44 的左侧。当“空间比例”为 0.95 时，蓝色曲线的垂直坐标约为 0.5。这意味着对于大约 95% 的设计空间，相对预测方差低于误差方差的 50%。

该“定制设计”窗口包含您的 I 最优设计。请让该窗口保持打开状态。在下一节中，您将生成一个 D 最优设计，并对这两个设计加以比较。

D 最优设计

要对 I 最优设计和 D 最优设计的 “预测方差刻画” 和 “设计空间比例图” 加以比较，请执行以下步骤：

1. 在包含您的 I 最优设计的 “定制设计” 窗口中，点击 “定制设计” 红色小三角并选择**将脚本保存至脚本窗口**。

   随即显示一个窗口，显示可重现您的工作的脚本。
2. 在这个新脚本窗口中，选择**编辑 > 运行脚本**。

   随即显示一个重复的 “定制设计” 窗口，但其中的 “设计评估” 分级显示项已关闭。
3. 在这个新的定制设计窗口中，点击**后退**。
4. 点击 “定制设计” 红色小三角并选择**最优性准则 > 制作 D 最优设计**。
5. （可选）点击 “定制设计” 红色小三角，选择**设置随机种子**，键入 383570403，然后点击**确定**。
6. 点击**制作设计**。

   您当前的 “定制设计” 窗口包含 D 最优设计。

图 5.42 D 最优设计

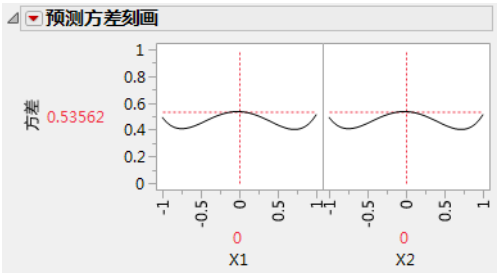
设计

试验	X1	X2
1	0	0
2	1	0
3	-1	-1
4	1	-1
5	0	1
6	-1	0
7	-1	-1
8	-1	1
9	1	-1
10	-1	1
11	0	-1
12	1	1

在该 D 最优设计中，试验 1 是唯一位于中心点的试验。与 I 最优设计相比，D 最优设计往往会在设计空间的极点放置较多试验。回想一下， I 最优设计有四次中心试验 （图 5.40）。

7. 打开**设计评估 > 预测方差刻画**分级显示项。

图 5.43 D 最优模型的预测方差刻画



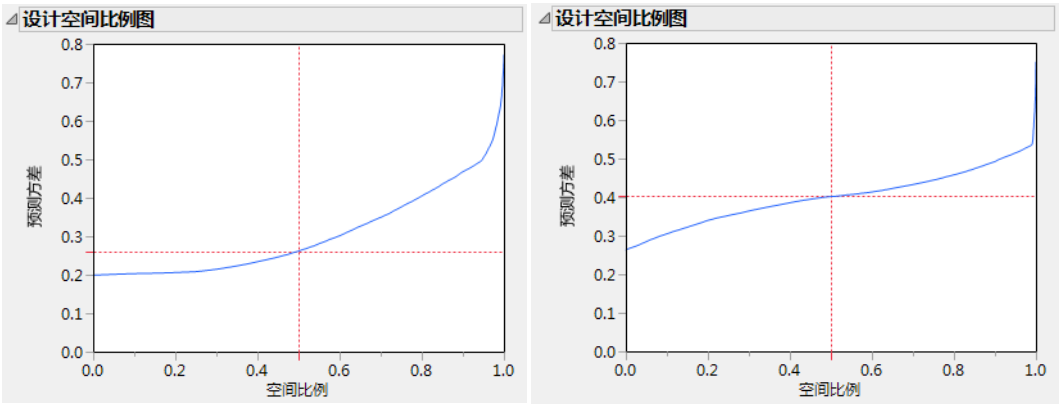
在设计区域的中心，相对预测方差为 0.53562，而不是 I 最优设计中的 0.208333（图 5.41）。这意味着相对标准误差对于 D 最优设计为 0.732，对于 I 最优设计为 0.456。若其他所有值都相同，在设计区域的中心，基于 D 最优设计的期望响应的置信区间比基于 I 最优设计的期望响应的置信区间宽大约 60%。

“设计”分级显示项显示 D 最优设计有九个设计点，X1 和 X2 设置为 -1、0、1 时的每种组合都对应一个设计点。D 最优性准则尝试在以上每一个设计点保持相对预测方差较低。通过将 X1 和 X2 的滑块移至 -1 和 1 来探索设计区域极点处的方差。请注意，这些极点处的方差通常小于 I 最优设计中这些点处的方差。

8. 打开**设计评估** > **设计空间比例图**分级显示项。

“设计空间比例图”显示在图 5.44 的右侧。

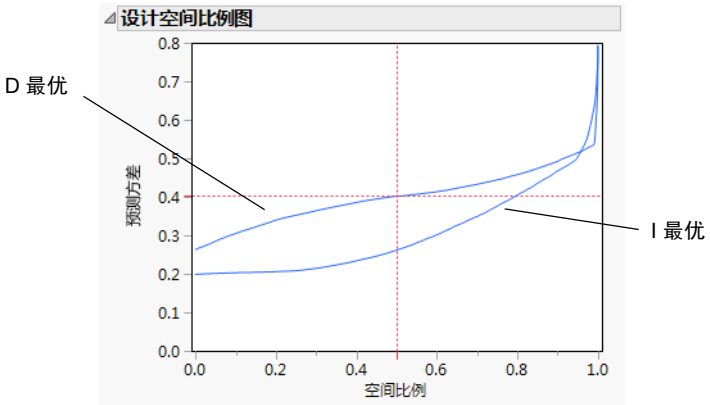
图 5.44 设计空间比例图 （左侧为 I 最优，右侧为 D 最优）



每个图中的红色十字准线指示 50% 的设计空间的最大预测方差。对于 50% 的设计空间，I 最优设计的预测方差落在大约 0.27 以下。对于 D 最优设计，预测方差约为 0.4。

9. 在 I 最优设计的 “设计空间比例图”中右击。选择**编辑** > **复制框架内容**。
10. 在 D 最优设计的 “设计空间比例图”中右击。选择**编辑** > **粘贴框架内容**。

图 5.45 叠加的两个设计空间比例图



在至少 90% 的设计空间内，I 最优设计的方差曲线都位于 D 最优设计的曲线下方。这反映出这样的事实：I 最优设计尝试在整个设计空间内最小化预测方差。相比之下， D 最优设计则侧重于在设计点处减小预测方差。

包含约束和分类因子的响应曲面设计

在本例中，您需要创建一个设计来优化化学反应的产量。实验因子包括一个有三个水平的分类因子，其中的约束涉及其中两个水平。在本例中，您需要使用“不被允许的组合过滤器”输入约束。

响应为**产量**。您有三个因子：

- **时间**：关注的范围介于 500 到 560 秒之间。
- **温度**：关注的范围介于 350 到 750 开氏温度之间。
- **催化剂**：必须测试 A、B、C 这三种催化剂。

设计必须考虑**催化剂**的两个水平的约束条件：

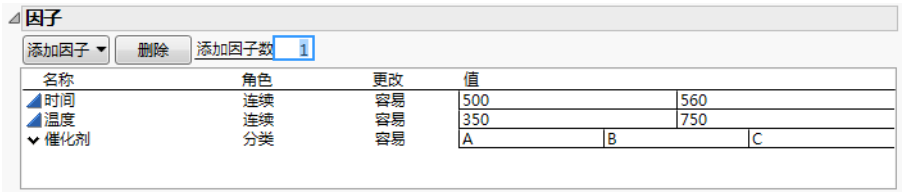
- 使用催化剂 B 时，温度必须高于 400。
- 使用催化剂 C 时，温度必须低于 650。

定义响应和因子

1. 选择**实验设计** > **定制设计**。
2. 在 “响应”分级显示项中，双击 Y 并将其改为**产量**。
   由于您的目标是将**产量**最大化，所以保持**目标**设置为 “最大化”。
3. 在 “因子”分级显示项中的**添加因子数**旁边键入 2。
4. 点击**添加因子** > **连续**。
5. 点击**添加因子** > **分类** > **3 水平**。

6. 重命名因子**时间**、**温度**和**催化剂**。
7. 将**时间**的**值**改为 500 和 560。
8. 将**温度**的**值**改为 350 和 750。
9. 将**催化剂**的**值**改为 A、 B、 C。

图 5.46 因子设置



10. 点击**继续**。

定义约束

1. 在“定义因子约束”分级显示项中选择**使用不被允许的组合过滤器**。
2. 从**添加过滤器因子**列表中选择**温度**和**催化剂**。
3. 点击**添加**。
4. 在“温度”滑块上方显示的等式中点击 750 并将其改为 400。按 Enter 键（或点击其他位置）。

   这会禁止使用**温度**值低于 400 的因子设置。
5. 点击**催化剂**下的 B 区组。

   连同**温度**约束，这会禁止使用**催化剂**为 B 且**温度**值低于 400 的因子设置。
6. 点击**或**。

   这将允许您定义第二个约束。
7. 从**添加过滤器因子**列表中选择**温度**和**催化剂**。（您之前的选择可能已经保留。）
8. 点击**添加**。
9. 在 “或”一词下方显示的面板中，在 “温度”滑块上方显示的等式中点击 350 并将其改为 650。按 Enter 键 （或点击其他位置）。

   这会禁止使用**温度**值高于 650 的因子设置。
10. 点击**催化剂**下的 C 区组。

    连同**温度**约束，这会禁止使用**催化剂**为 C 且**温度**值高于 650 的因子设置。

图 5.47 定义的约束

注意：催化剂之后的 (3) 指示催化剂水平数。

添加响应曲面项并制作设计

1. 在“模型”分级显示项中，选择 **RSM**。

   会有 JMP 警示提醒您只有连续因子的二次项才会添加到模型中。

2. 点击**确定**取消显示该提醒。

   JMP 只会将适当的响应曲面项添加到模型中。

   注意：在第 3 步中设置“随机种子”，在第 4 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

3. （可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 654321，然后点击**确定**。
4. （可选）点击“定制设计”红色小三角，选择**开始数**并将其设置为 1000。点击**确定**。
5. 点击**制作设计**。

图 5.48 满足约束的设计

设计

试验	时间	温度	催化剂
1	500	750	B
2	500	410	B
3	560	750	A
4	521	750	A
5	542	350	A
6	500	648.7299	C
7	500	350	A
8	533	630	C
9	506	350	C
10	560	530	A
11	560	410	B
12	530	570	B
13	530	550	A
14	530	570	B
15	557	645.1294	C
16	500	570	A
17	560	750	B
18	560	350	C

由于您向“模型”分级显示项添加了 RSM 项，所以这是一个 I 最优设计。它满足您的约束：

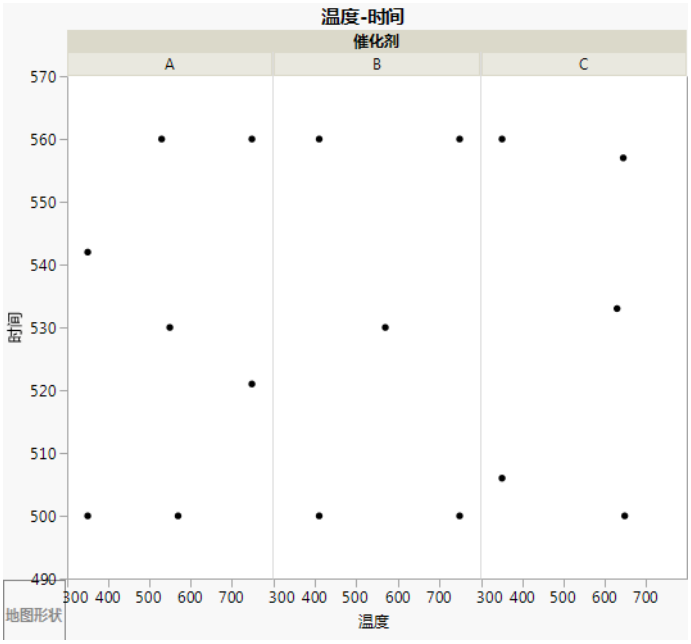
- 使用催化剂 B 时，温度必须高于 400。
- 使用催化剂 C 时，温度必须低于 650。

查看设计

回想一下设计区域包含介于 500 和 560 之间的**时间**设置和介于 350 到 750 之间的**温度**设置。生成一个图形以获取该设计区域内设计点的几何视图：

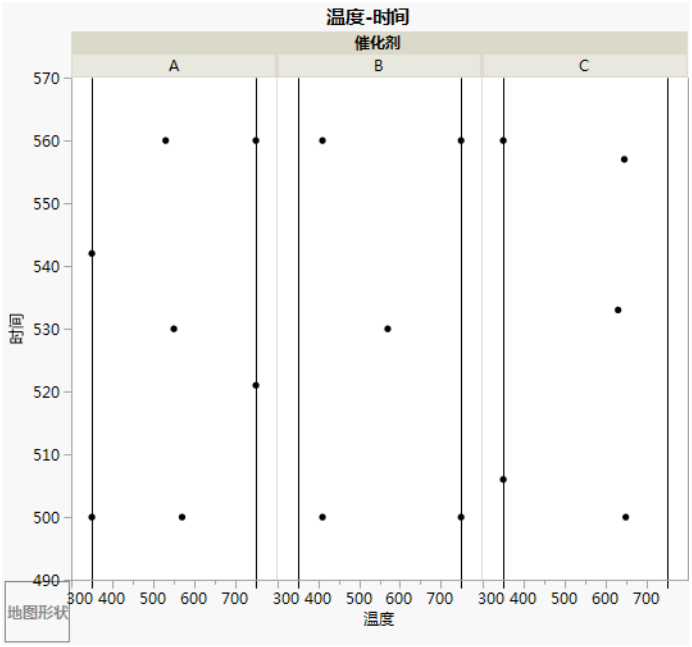
1. 点击**制表**。
2. 选择**图形 > 图形生成器**。
3. 选择**时间**和**温度**，然后将其拖至模板中央。
4. 取消选择**平滑线**图标。这是模板上方左侧的第二个图标。
5. 选择**催化剂**并将其拖至模板顶部的 **X 分组**区域。

图 5.49 针对催化器三个水平的设计点



6. 双击**温度**轴。
7. 在“参考线”下方，添加两条参考线：
   - 在**值**旁边，输入 350。
   - 点击**添加**。
   - 在**值**旁边，输入 750。
   - 点击**添加**。
8. 点击**确定**。

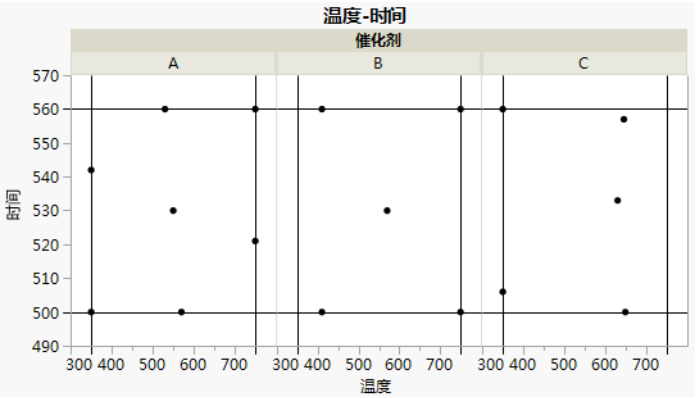
图 5.50 三水平催化剂设计中的温度边界



9. 双击**时间**轴。
10. 在 “参考线”下方，添加两条参考线：
    - 在**值**旁边，输入 500。
    - 点击**添加**。
    - 在**值**旁边，输入 560。
    - 点击**添加**。
11. 点击**确定**。
12. 点击**完成**。

注意：图 5.51 中的图已在垂直方向调整大小。

图 5.51 针对催化器三个水平的设计区域



催化剂 A 的六个设置落在设计区域的边缘。**催化剂** B 的**温度**设置位于 400 度约束的上方，**催化剂** C 的所有**温度**设置均位于 650 度约束下方。

混料实验

“定制设计”和“混料设计”平台都能构造所有因子均为混料成分情形的设计。不过，混料实验可能会涉及非混料过程变量，即**过程因子**。除了仅混料设计之外，“定制设计”平台还可以构造同时适应混料和过程因子的设计。“定制设计”平台还允许混料成分的总和为任意正数。请参见“定制设计”一章中第 85 页的“因子类型”中的“混合”说明。

- “包含非混料因子的混料设计”
- “混合混料设计”

包含非混料因子的混料设计

在 Atkinson 和 Donev (1992) 所提供的这一示例中，您针对同时涉及混料因子和过程因子的实验创建设计。该设计涉及 18 次试验，它在分类因子的各水平得到平衡。该设计支持您拟合完全响应曲面。您使用设计评估图和结果来检查该设计的相对预测方差。

设计由一个响应、三个混料因子和一个非混料因子构成：

- 响应为**阻尼**，用来测量丙烯腈粉的电磁阻尼。
- 三种混料成分分别为：
  - **硫酸铜** (CuSO4)，介于 0.2 到 0.8 之间
  - **亚硫酸钠** (Na2S2O3)，介于 0.2 到 0.8 之间
  - **乙二醛**，介于 0 到 0.6 之间
- 关注的非混料环境因子为**波长**（电磁波的波长），该因子有三个水平，分别标为 L1、L2 和 L3。

波长是一个连续变量。不过，研究人员仅关注三个特定波长的预测值。出于此原因，您可以将**波长**视为包含三个水平的分类因子。

本例包含两部分：

- 第 161 页的“创建设计”
- 第 163 页的“评估设计”

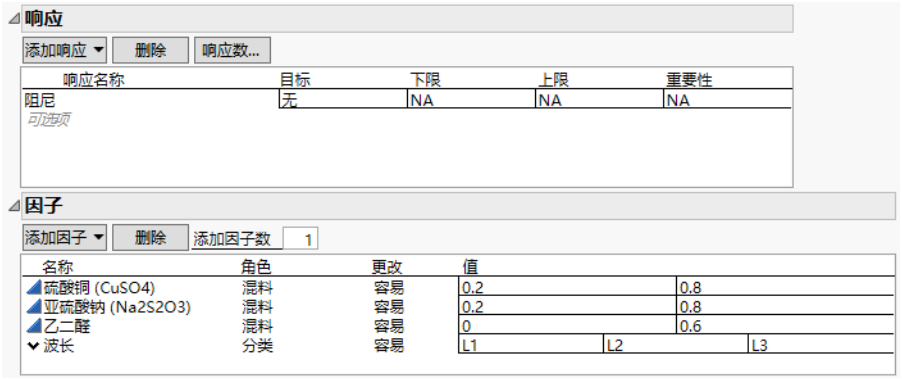
创建设计

1. 选择**实验设计 > 定制设计**。
2. 双击“响应名称”下的 Y 并键入**阻尼**。
3. 点击“目标”下的**最大化**，然后将其改为**无**。

   目标设置为“无”，因为本研究的作者并未提及期望的阻尼值。选择**帮助 > 样本数据库**并打开 Design Experiment/Donev Mixture Factors.jmp。
4. 点击“定制设计”红色小三角并选择**加载因子**。

   这将加载三个混料成分和分类环境因子。请注意，三个混料因子的值界限也随之加载。

图 5.52 “响应”分级显示项和“因子”分级显示项

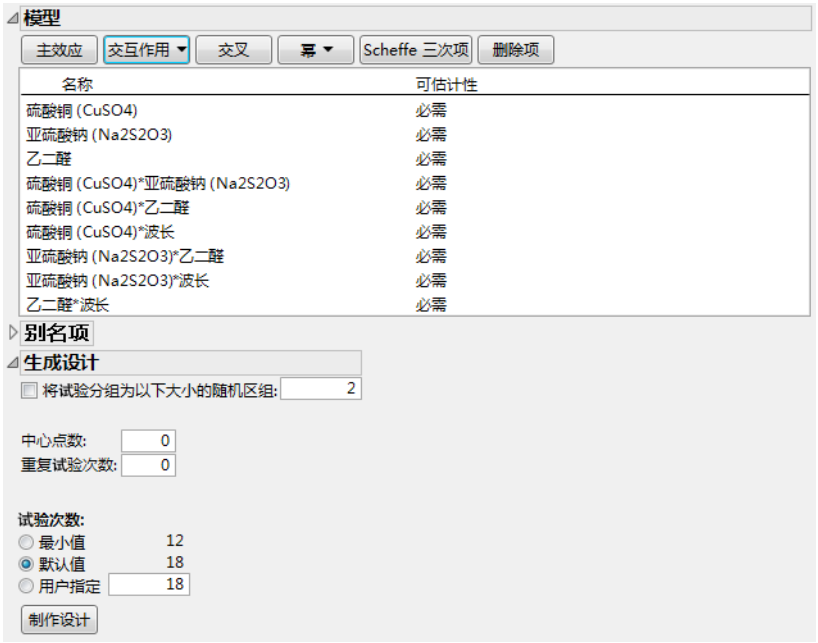


5. 在“模型”分级显示项中，点击**交互作用 > 二次**。

   将有一个说明性的“JMP 警示”窗口提示您 JMP 会删除与所有混料因子产生交互作用的非混料因子的主效应项。这意味着将删除**波长**主效应，但是会添加混料因子与**波长**的所有双因子交互作用。
6. 点击**确定**取消显示该消息。

   “模型”分级显示项中的效应以混料成分连同波长这个附加效应一起的形式定义响应曲面模型。请参见 Scheffé (1958)。

图 5.53 “模型”和“生成设计”分级显示项



7. 保持默认试验次数设置为 18。

   选择 18 次试验将允许对波长因子三个水平的每个水平都执行六次试验。

   注意：在第 8 步中设置“随机种子”，在第 9 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

8. （可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
9. （可选）点击“定制设计”红色小三角，选择**开始数**，键入 5，然后点击**确定**。
10. 点击**制作设计**。

图 5.54 显示 18 次试验设计的 “设计” 分级显示项

设计

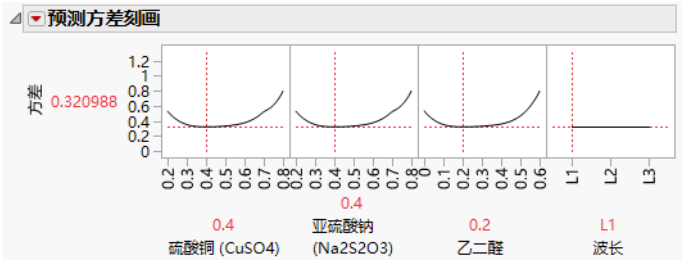
试验	硫酸铜 (CuSO4)	亚硫酸钠 (Na2S2O3)	乙二醛	波长
1	0.8	0.2	0	L3
2	0.2	0.2	0.6	L3
3	0.8	0.2	0	L2
4	0.2	0.8	0	L2
5	0.5	0.5	0	L3
6	0.2	0.2	0.6	L1
7	0.5	0.2	0.3	L1
8	0.8	0.2	0	L1
9	0.2	0.2	0.6	L2
10	0.5	0.5	0	L1
11	0.2	0.8	0	L3
12	0.2	0.5	0.3	L2
13	0.5	0.5	0	L2
14	0.2	0.5	0.3	L1
15	0.5	0.2	0.3	L3
16	0.2	0.5	0.3	L3
17	0.5	0.2	0.3	L2
18	0.2	0.8	0	L1

您可以检查**波长**的每个水平是否有六次试验。

评估设计

1. 打开**设计评估** > **预测方差刻画**分级显示项。

图 5.55 18 次试验设计的预测方差刻画



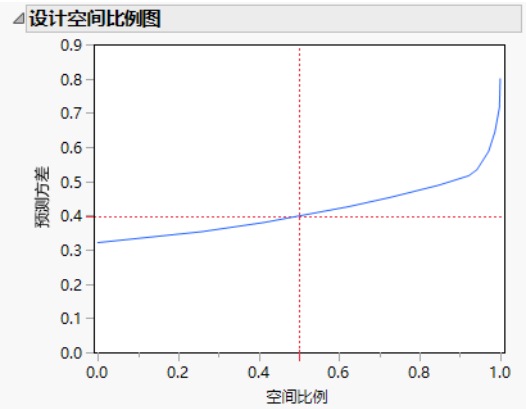
移动**波长**滑块，确认混料因子的相对预测方差刻画不随**波长**的水平变化。移动任一混料因子的滑块。另外两个混料因子的滑块将随之调整，以使各混料成分的总和为 1。请注意，最小相对预测方差出现在混料因子的中心设置附近。

2. 点击 “预测方差刻画” 红色小三角并选择**最大化方差**。

   请注意，设计空间上的最大相对预测方差为误差方差的 0.8 倍。

3. 打开**设计空间比例图**分级显示项。

图 5.56 18 次试验设计的设计空间比例图



在整个设计空间上，相对预测方差都低于 0.8。最小相对预测方差约为 0.32。如图 5.55 所示，最小相对预测方差出现在混料因子的中心设置附近。

4. 打开**设计诊断**分级显示项。

图 5.57 18 次试验设计的 “设计诊断” 分级显示项

设计诊断

D 最优设计	
D 效率	3.627407
G 效率	83.33333
A 效率	0.328715
预测平均方差	0.412892
设计创建时间（秒）	0.066667

相对于 D 最优性准则，该设计是最优的，尽管其 D 效率非常低 (3.6%)。由于存在混料约束，混料设计远非正交，所以它们通常具有非常低的 D 效率。预测的平均（相对）方差为 0.412864。这与图 5.56 中的 “设计空间比例图” 一致。

混合混料设计

在本例中，针对混料成分情形构造设计。

考虑制作蛋糕的原料成分。干性成分包括面粉、糖和可可粉。湿性成分包括牛奶、溶化黄油和鸡蛋。蛋糕的干性成分和湿性成分是先分别混合而后又混合在一起的两种混料。表 5.1 列出了相关因子以及在实验中您对这些因子取值的范围。

表 5.1 干性和湿性成分以及实验范围

混料	成分	低水平和高水平
干性	可可粉	0.1 - 0.2
	糖	0 - 0.15
	面粉	0.2 - 0.3
湿性	黄油	0.1 - 0.2
	牛奶	0.25 - 0.35
	鸡蛋	0.05 - 0.20

干性成分（可可粉、糖和面粉的混料）构成组合混料的 45%。湿性成分（黄油、牛奶和鸡蛋）构成 55%。

注意：在干性成分和湿性成分的约束下，若要在混料模型中包含全部 6 个因子，则会存在模型奇异性。

您的实验目标是要优化“口味”等级。“口味”分为 1 到 10 十个级别，10 表示最佳口味。

您需要构造一个 10 试验设计以拟合主效应模型。由于干性成分和湿性成分比例的约束，您需要在“模型”分级显示项中仅包含五个因子以避免奇异性。可任选一个因子不放进模型。

本例包含两部分：

- 第 165 页的“创建设计”
- 第 167 页的“分析实验结果”

创建设计

1. 选择**实验设计 > 定制设计**。
2. 双击 “响应名称”下的 Y 并键入**口味**。

   请注意，默认 “目标”为最大化。由于您想要最大化**口味**等级，所以不要更改该目标。
3. 在 “下限”下点击并键入 0。

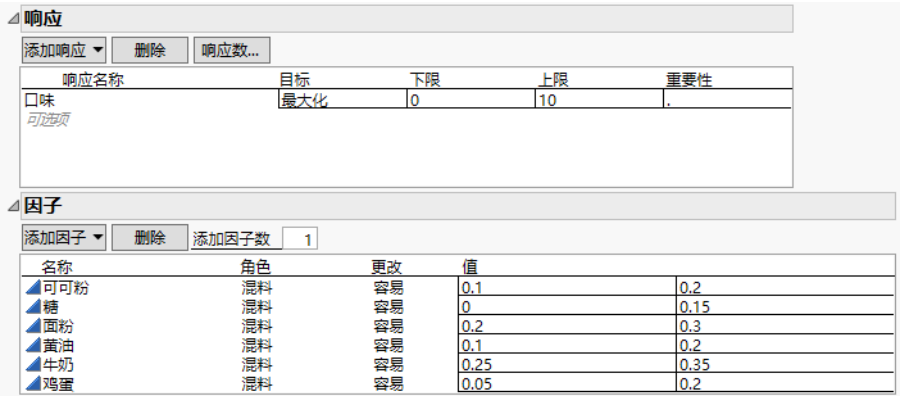
   最小的满意等级为 0。
4. 在 “上限”下点击并键入 10。

   最大的满意等级为 10。
5. 将 “重要性”下面的区域留空。

   由于只有一个响应，默认为该响应的 “重要性”指定 1。
6. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Cake Factors.jmp。

7. 点击“定制设计”红色小三角并选择**加载因子**。

图 5.58 已完成的“响应”和“因子”分级显示项



请注意，因子全部为混料因子。为实验定义设置范围的值随因子而异。

8. 在“定义因子约束”分级显示项中，选择**指定线性约束**。
9. 在“线性约束”面板中，点击**添加**两次。
10. 输入图 5.59 中所示的约束。

图 5.59 定义因子约束

请注意，第二个约束等同于指定**可可粉**、**糖**和**面粉**的总和大于或等于 0.45。这两个约束共同意味着**可可粉**、**糖**和**面粉**构成恰好 45% 的混料，确保湿性因子构成剩下的 55%。

11. 在“模型”分级显示项中，选择“鸡蛋”并点击**删除项**。

    由于等式约束的作用，包含全部六个效应的模型为奇异模型。

12. 在**用户指定**旁边键入 10。

    您的实验要求烘培 10 个蛋糕。

注意：在第 13 步中设置 “随机种子”，在第 14 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

13. （可选）点击 “定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
14. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 40，然后点击**确定**。
15. 点击**制作设计**。

    “JMP 警示”告知您因子约束中包含一个等式约束。这正如您所愿，因为干性成分比例的总和不得超过 45%。
16. 点击**确定**关闭 JMP 警示。
17. 点击**制表**。

图 5.60 混合混料设计

	可可粉	糖	面粉	黄油	牛奶	鸡蛋	口味
1	0.2	0	0.25	0.2	0.3	0.05	•
2	0.1	0.15	0.2	0.15	0.35	0.05	•
3	0.1	0.05	0.3	0.2	0.25	0.1	•
4	0.2	0.05	0.2	0.1	0.25	0.2	•
5	0.2	0.05	0.2	0.1	0.35	0.1	•
6	0.1	0.15	0.2	0.1	0.25	0.2	•
7	0.1	0.05	0.3	0.1	0.25	0.2	•
8	0.1	0.15	0.2	0.2	0.3	0.05	•
9	0.15	0	0.3	0.1	0.35	0.1	•
10	0.2	0.05	0.2	0.2	0.25	0.1	•

干性成分**可可粉**、**糖**和**面粉**的设置的总和为混料的 45%，湿性成分**黄油**、**牛奶**和**鸡蛋**的设置的总和为混料的 55%。这些设置还符合 “因子”分级显示项中给定的上限和下限。

分析实验结果

Cake Data.jmp 样本数据表显示了实验结果。设计表包含一个“模型”脚本，该脚本可打开“拟合模型”窗口，其中显示在 “实验设计”窗口的 “模型”分级显示项中指定的五个主效应。请注意，该设计的 “模型”分级显示项中不包含主效应**鸡蛋**。“鸡蛋”的水平可通过其他 5 个因子的设置和干性成分的约束来确定。在“定制设计”平台中制表时，该脚本被保存到数据表中。

1. 打开 Design Experiment 文件夹中的 Cake Data.jmp 样本数据表。
2. 在设计表的 “表”面板中，点击**模型**脚本旁边的绿色小三角。

   鸡蛋对应的主效应未包含在内。要避免模型奇异性，在 “定制设计”窗口的 “模型”分级显示项中已排除了**鸡蛋**效应。五个效应均指定为响应曲面和混料效应。
3. 点击**运行**。

   随即显示 JMP 警示，通知您由于因子子集存在混料约束的混杂而无法显示刻画器。
4. 点击**确定**关闭 JMP 警示。

“参数估计值”报表指示**糖**和**面粉**在 0.05 水平下显著。

图 5.61 “参数估计值”报表

参数估计值

项	估计值	标准误差	t 比	概率>|t|
(可可粉-0.1)/0.35	5.1056115	2.360215	2.16	0.0829
糖/0.35	16.873989	1.675197	10.07	0.0002*
(面粉-0.2)/0.35	15.750757	2.207353	7.14	0.0008*
(黄油-0.1)/0.35	5.3740398	2.654689	2.02	0.0988
(牛奶-0.25)/0.35	1.5973046	2.435648	0.66	0.5409

包含协变量的实验

有时，供实验使用的实验单元具有某种测量值。这些测量值可能会影响实验响应。将这些称为**协变量**的变量作为设计因子包括在内会非常有用。尽管您无法直接控制这些值，但您可以确保选取的其他设计因子的水平能够生成所有效应的最精确估计值。

“定制设计”平台构造的设计可采用最优方式选择协变量值。从现有数据表中选择协变量值，该数据表提供关于可能的实验单元的协变量信息。您可以指定比数据表中所列实验单元数少的试验次数。您还可以指定难以更改的协变量。制作设计时，若设计中的试验次数少于协变量表中的行数，则设计表会包含一个**协变量行索引**列。该列指示协变量表中与每次试验相对应的行。

注意：在协变量数据表中，协变量因子包含非缺失值的行数必须大于等于指定的试验次数。

- “包含固定协变量的设计”
- “包含难以更改的协变量的设计”
- “带有线性时间趋势的设计”

包含固定协变量的设计

在本例中，您关注的是对注塑成型过程生产的部件的**收缩**建模。Design Experiment 文件夹中的 Thermoplastic.jmp 样本数据表列出了研究中可能会用到的 25 个批次的原材料（热塑材料）。对于每个批次，都获取些材料以获得**比重**和**抗拉强度**的测量值。还提供了第三个协变量**供应商**。

您想要研究三个可控因子**温度**（模具温度）、**速度**（转速）和**时间**（保留时间）对于**收缩**的效应。但您还想要研究协变量**比重**、**抗拉强度**和**供应商**的效应。您的资源允许执行 12 次试验。

创建设计

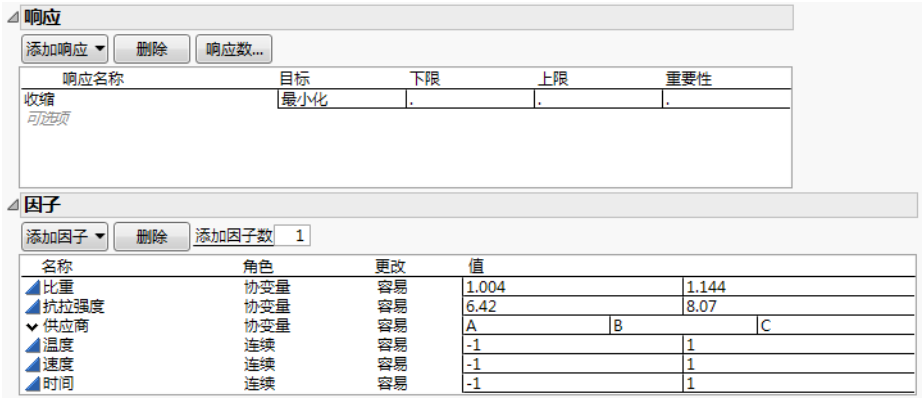
1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Thermoplastic.jmp。
2. 选择**实验设计** > **定制设计**。
3. 双击“响应名称”下的 Y 并键入**收缩**。
4. 点击“目标”下的**最大化**，然后将其改为**最小化**。

5. 点击**添加因子**并选择**协变量**。
6. 从列表中选择**比重**、**抗拉强度**和**供应商**，然后点击**确定**。

   这些是协变量，不受控制。
7. 在**添加因子数**旁边键入 3。
8. 点击**添加因子 > 连续**。
9. 将三个连续因子重命名为**温度**、**速度**和**时间**。

   这些是可控因子。

图 5.62 “响应”和“因子”分级显示项



10. 点击**继续**。

    “试验次数”显示包含协变量值的行数。您有 25 个批次带有测量过的协变量。
11. 在**试验次数**旁边键入 12。

> **注意：**在第 12 步中设置“随机种子”，在第 13 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

12.（可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 84951，然后点击**确定**。
13.（可选）点击“定制设计”红色小三角，选择**开始数**，键入 40，然后点击**确定**。
14. 点击**制作设计**。

图 5.63 12 次试验最优设计

设计

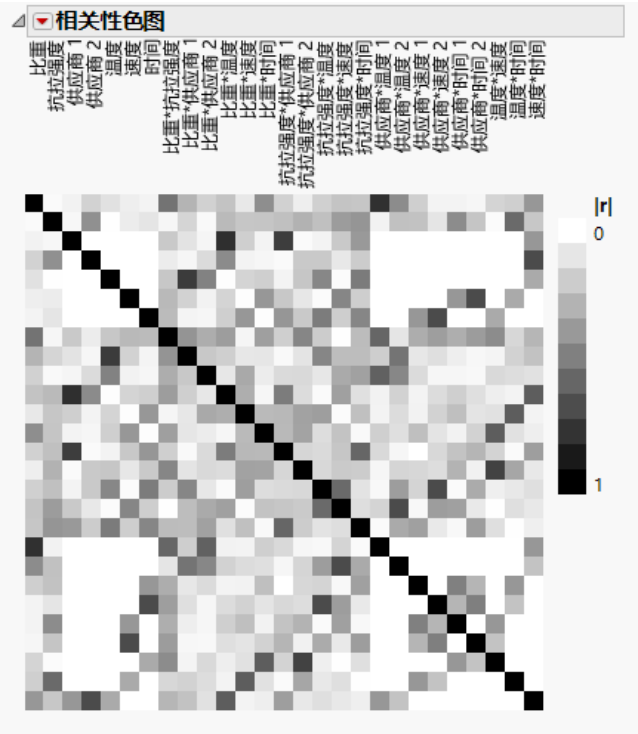
试验	比重	抗拉强度	供应商	温度	速度	时间
1	1.107	8.07	B	1	1	1
2	1.144	6.82	B	1	-1	-1
3	1.004	6.52	C	1	1	-1
4	1.015	7.46	A	-1	1	-1
5	1.129	6.97	A	1	-1	-1
6	1.139	6.94	C	-1	1	-1
7	1.004	7.12	C	1	-1	1
8	1.067	6.7	B	-1	1	1
9	1.047	6.42	A	-1	-1	1
10	1.03	7.54	B	-1	-1	-1
11	1.094	6.48	A	1	1	1
12	1.113	6.89	C	-1	-1	1

考虑到可能的协变量值，该设计为 D 最优设计。该设计选择最优的协变量值组以及适用于这三个可控因子的最优设置。

评估设计

1. 打开**设计评估** > **相关性色图**分级显示项。

图 5.64 相关性色图



与主效应对应的七项显示在色图左上角。请注意，这七项接近正交。最大绝对相关性出现在**抗拉强度**与**供应商 2** 之间。这个大约 0.43 的绝对相关性是可用协变量值所带来的结果。（图 5.64 使用 JMP 默认颜色。）

包含难以更改的协变量的设计

在本例中，您需要构造一种设计，为选手级跑步者开发一款具有耐磨 （**损耗**）特点的跑鞋。您的实验包含四个因子：

- 鞋底厚度（**厚度**）
- 减震胶体量（**胶体**）
- 鞋底材料（**外底**）
- 鞋底夹层材料（**鞋底夹层**）

要获得可概况的结论，您需要针对大量选手级跑步者检验这款鞋。为了适应实验预算，每名跑步者都必须检验若干实验组合。

您的公司已收集 100 名愿意参与您的研究的合适跑步者的数据。对这些跑步者测量的相伴变量（**协变量**）包括平均每天跑步英里数（**英里数**）、体重（**体重**）和脚的着地点（**着地点**）。

创建设计：

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Runners Covariates.jmp。
2. 选择**实验设计 > 定制设计**。
3. 双击 “响应名称”下的 Y 并键入**损耗**。
4. 点击 “目标”下的**最大化**，然后将其改为**最小化**。
5. 点击**添加因子**并选择**协变量**。
6. 从列表中选择**英里数**、**体重**和**着地点**，然后点击**确定**。

   这些是与跑步者关联的难以更改的协变量。
7. 针对**英里数**、**体重**和**着地点**这三个因子之一，在 “更改”下点击**容易**，并将其改为**困难**。

   请注意，这三个协变量的 “更改”值全部变为 “困难”。

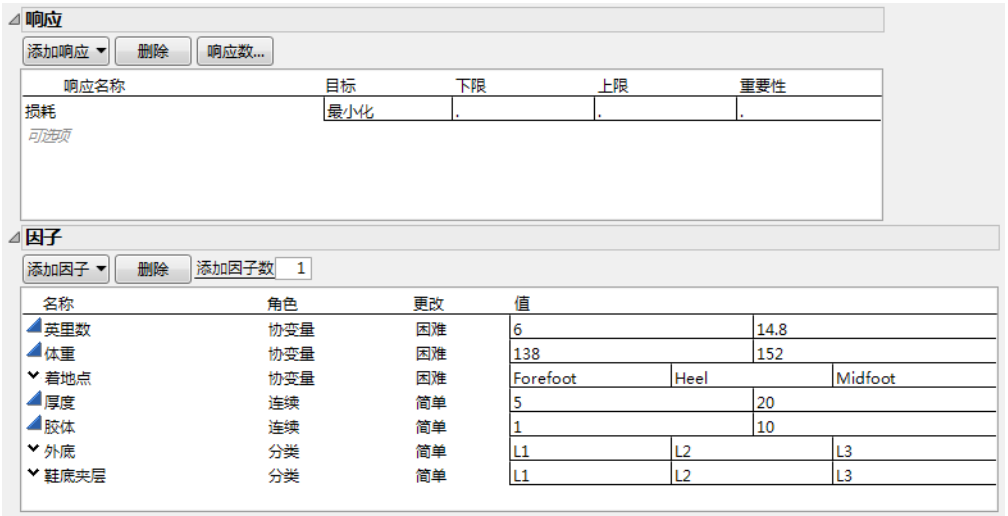
要手动添加其余因子，请执行第 8 步到第 16 步。或者，要从保存的表加载因子，请从“定制设计”红色小三角中选择**加载因子**。打开 Design Experiment 文件夹中的 Runners Factors.jmp 样本数据表。若选择**加载因子**，则可跳过第 8 步到第 16 步。

8. 在**添加因子数**旁边键入 2。
9. 点击**添加因子 > 连续**。
10. 重命名**厚度**和**胶体**这两个因子。
11. 将**厚度**的**值**改为 5 和 20。
12. 将**胶体**的**值**改为 1 和 10。

13. 在**添加因子数**旁边键入 2。
14. 点击**添加因子 > 分类 > 3 水平**。
15. 重命名**外底**和**鞋底夹层**这两个因子。

    保留这些因子的默认**值**。

图 5.65 “响应”和“因子”分级显示项



16. 点击**继续**。
17. 选择**交互作用 > 二次**。

    指定的模型拟合所有双因子交互作用，包括协变量与实验因子的交互作用。
18. 将“整区数”（即跑步者数量）设置为 32（若尚未设置为该数值）。
19. 在“试验次数”下的“用户指定”旁边键入 64（若尚未设置为该数值）。

> **注意：**在第 20 步中设置“随机种子”，在第 21 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

20.（可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
21.（可选）点击“定制设计”红色小三角，选择**开始数**，并将其设置为 1（若尚未设置为该数值）。点击**确定**。
22. 点击**制作设计**。

图 5.66 针对难以更改的协变量的设计中的前 20 次试验

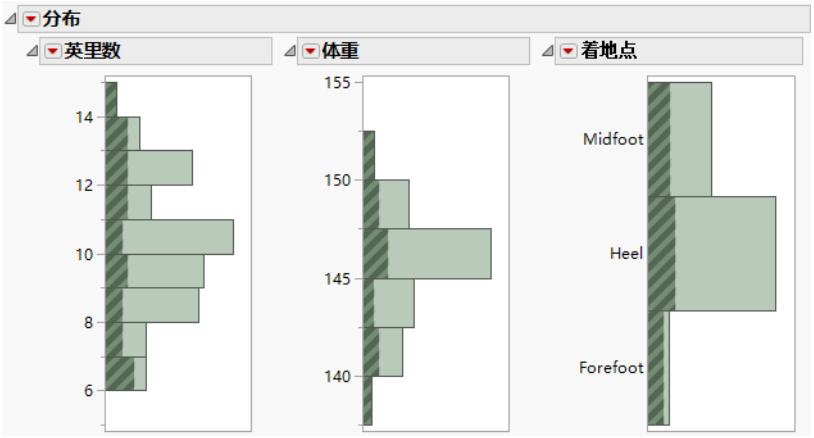
设计

试验	整区	英里数	体重	撞击点	厚度	胶体	外底	鞋底夹层
1	1	7.6	144	Midfoot	5	10	L3	L2
2	1	7.6	144	Midfoot	20	1	L2	L1
3	2	7.6	145	Midfoot	20	10	L3	L3
4	2	7.6	145	Midfoot	5	1	L1	L1
5	3	14.8	141	Midfoot	20	1	L1	L3
6	3	14.8	141	Midfoot	5	10	L2	L1
7	4	6.4	146	Heel	20	10	L1	L2
8	4	6.4	146	Heel	5	1	L2	L3
9	5	11.2	138	Heel	5	1	L2	L3
10	5	11.2	138	Heel	20	10	L3	L2
11	6	11.2	144	Forefoot	20	1	L3	L3
12	6	11.2	144	Forefoot	5	10	L2	L1
13	7	12.8	139	Heel	5	10	L3	L1
14	7	12.8	139	Heel	20	1	L1	L2
15	8	14	139	Midfoot	5	1	L1	L2
16	8	14	139	Midfoot	20	10	L2	L3
17	9	6	150	Heel	20	10	L2	L2
18	9	6	150	Heel	5	1	L3	L3
19	10	10.8	143	Forefoot	20	1	L3	L1
20	10	10.8	143	Forefoot	5	10	L2	L2

在 100 名跑步者中，根据其协变量值选定了其中的 32 名跑步者。与选定跑步者对应的行也在 RunnersCovariates.jmp 样本数据表中选定。确定实验因子**厚度**、**胶体**、**内底**和**外底**的设置，以使设计对于 “模型” 分级显示项中描述的模型是最优的。

23. 在 RunnersCovariates.jmp 样本数据表成为当前表时，选择**分析 > 分布**。
24. 将所有三列都选作 **Y，列**。
25. 选中**仅显示直方图**。
26. 点击**确定**。

图 5.67 100 名跑步者的直方图 （已选定的跑步者数据用阴影表示）



这些直方图演示设计选定的跑步者如何在协变量之间平衡各次跑步。选定跑步者的分布类似于每个协变量的所有跑步者的总体分布。

带有线性时间趋势的设计

通常，依照一定时间顺序执行的实验会遇到响应的线性漂移。若您随机执行试验，则漂移的效应通常不会使估计的因子效应发生偏倚。不过，若将漂移考虑在内，您可以减小这些效应的方差。

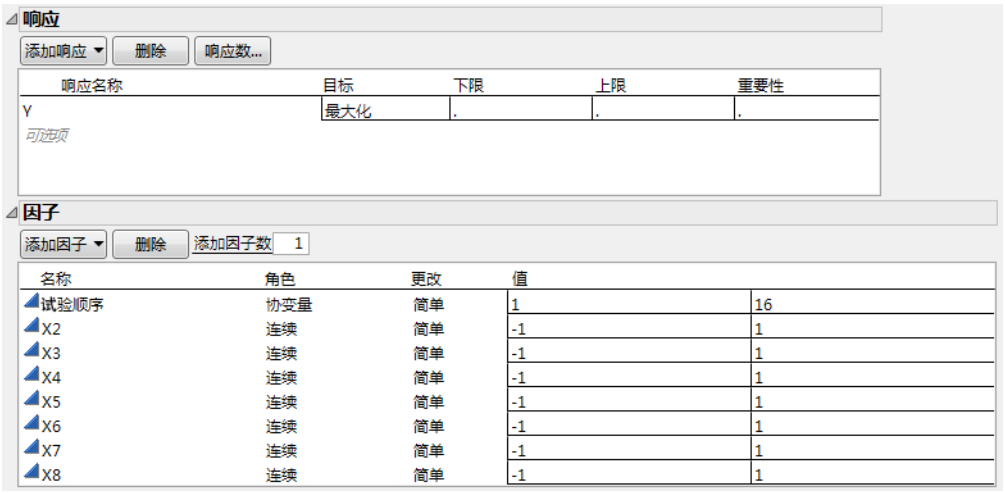
假定有理由怀疑随着时间的推移，响应中存在与因子变化无关的强线性趋势。那么您可以构造一种设计，在其中包含解释该趋势的线性协变量。在考虑到这一趋势协变量后，生成的设计将是最优的。

在本例中，您针对 7 个因子设计实验。您构造一个对于线性趋势稳健的包含 16 次试验的设计。

1. 选择**文件 > 新建 > 数据表**。
2. 右击**第 1 列**，然后选择**列信息**。
3. 将列名改为**试验顺序**。
4. 从**初始化数据**选项列表中，选择**序列数据**。
5. 在**至**旁边键入 16。
6. 点击**确定**。

   数据表中已输入从 1 到 16 的连续整数。
7. 选择**实验设计 > 定制设计**。
8. 点击**添加因子 > 协变量**。
9. 选择**试验顺序**并点击**确定**。
10. 在**添加因子数**旁边键入 7。
11. 点击**添加因子 > 连续**。

图 5.68 “响应”和“因子”分级显示项



12. 点击**继续**。

13. 打开**别名项**分级显示项。
14. 选择列表中的所有效应并点击**删除项**。

    这会从相关性色图中去除交互作用效应，仅保留主效应。

    注意：在第 15 步中设置“随机种子”，在第 16 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

15.（可选）点击“定制设计”红色小三角，选择**设置随机种子**，键入 1234，然后点击**确定**。
16.（可选）点击“定制设计”红色小三角，选择**开始数**，键入 100，然后点击**确定**。
17. 点击**制作设计**。

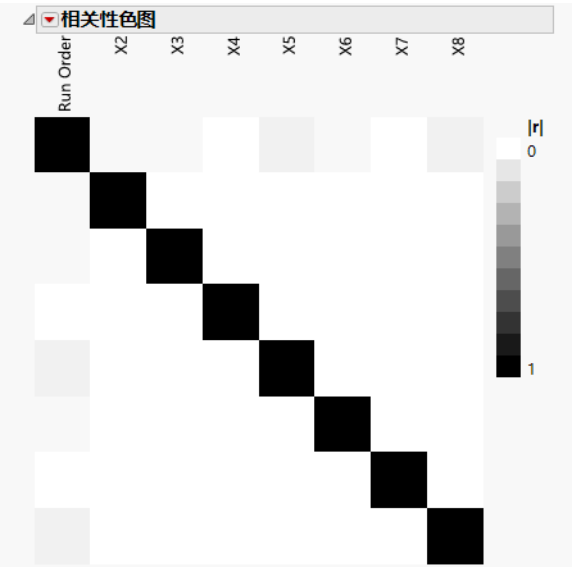
图 5.69 “设计”分级显示项

设计

试验	试验顺序	X2	X3	X4	X5	X6	X7	X8
1	7	-1	-1	1	1	-1	1	1
2	1	-1	1	-1	-1	1	1	-1
3	4	1	1	1	-1	-1	-1	1
4	12	1	1	-1	1	-1	1	-1
5	15	-1	1	-1	-1	-1	-1	1
6	8	1	-1	1	-1	1	1	1
7	11	-1	1	1	-1	-1	1	-1
8	3	1	-1	1	-1	-1	-1	-1
9	5	-1	-1	-1	1	-1	1	1
10	13	-1	-1	1	1	1	-1	-1
11	16	1	1	1	1	1	1	1
12	14	-1	-1	-1	-1	1	-1	1
13	10	1	-1	-1	1	-1	-1	-1
14	9	1	-1	-1	-1	1	1	-1
15	2	1	1	-1	1	1	-1	1
16	6	-1	1	1	1	1	-1	-1

18. 打开**设计评估 > 相关性色图**分级显示项。

图 5.70 显示具有试验顺序的绝对相关性的色图



色图显示以下信息：

- 七个连续因子 X2 到 X8 彼此正交。
- **试验顺序**（线性时间趋势变量）与 X2 到 X8 之间存在极低的绝对相关性。

19. 打开**设计评估** > **估计效率**分级显示项。

图 5.71 “估计效率”分级显示项

估计效率

项	CI 长度部分增加	估计的相对标准误差
截距	0	0.25
试验顺序	0.65	0.412
X2	0.006	0.252
X3	4e-4	0.25
X4	0	0.25
X5	0.003	0.251
X6	4e-4	0.25
X7	4e-4	0.25
X8	0.003	0.251

试验顺序与 X2 到 X8 之间较小的绝对相关性导致置信区间长度极小幅的增加 （相对于理想的正交设计而言）。 X2 到 X8 的置信区间长度的增加全部小于约 0.1%。

在本例中，试验顺序因子与因子效应接近正交。在某些情况下，您设计中的试验顺序因子与其他因子之间可能存在程度更大的相关性。即便在这样的情形下，将试验顺序作为因子包含在内可解释任何线性趋势效应。包含试验顺序还使得其他因子效应的估计更加精确。

包含随机化限制的实验

“定制设计”平台可构造 D 最优或 I 最优的裂区、裂 - 裂区和双因子裂区 （带区）设计。有关构造这些设计的详细信息，请参见 Goos (2002)。

- “裂区实验”
- “双因子裂区实验”

裂区实验

裂区设计发源于农业，但在制造业和工程设计研究中也很普遍。在裂区实验中，只有在一个整区与下一个整区之间才重置难以更改的因子。整区细分为子区，易于更改的因子的水平随机分配给各个子区。

本节中的示例改编自 Kowalski、 Cornell 和 Vining (2002)。您关注的是五个因子的效应，这些因子影响用于制作汽车座罩的乙烯基塑料的厚度。实验中的响应和因子如下所述：

- 响应是生产的乙烯基塑料的**厚度**。您想要使**厚度**最大化。厚度值的下限为 10。
- 整区因子包含挤压的速度（**挤压速度**）和烘干温度（**温度**）。这些是过程变量，难以更改。
- 子区因子包含三种增塑剂，这三种增塑剂的比例（m1、m2 和 m3）之和为 1。这些因子为混料成分。

您的实验预算允许运行这些整区因子的 7 个设置。对于每个整区，您都可以针对子区因子执行 4 次试验。因而试验总数为 28 次。

创建设计

1. 选择**实验设计** > **定制设计**。
2. 双击 “响应名称”下的 Y 并键入**厚度**。

   保持默认目标设置为 “最大化”。
3. 输入下限 10。

要手动添加因子，请执行第 4 步到第 11 步。或者，要从保存的表加载因子，请从“定制设计”红色小三角中选择**加载因子**。打开 Design Experiment 文件夹中的 Vinyl Factors.jmp 样本数据表。若选择**加载因子**，则可跳过第 4 步到第 11 步。

4. 在**添加因子数**旁边键入 2。
5. 点击**添加因子** > **连续**。
6. 将这些因子重命名为**挤压速度**和**温度**。

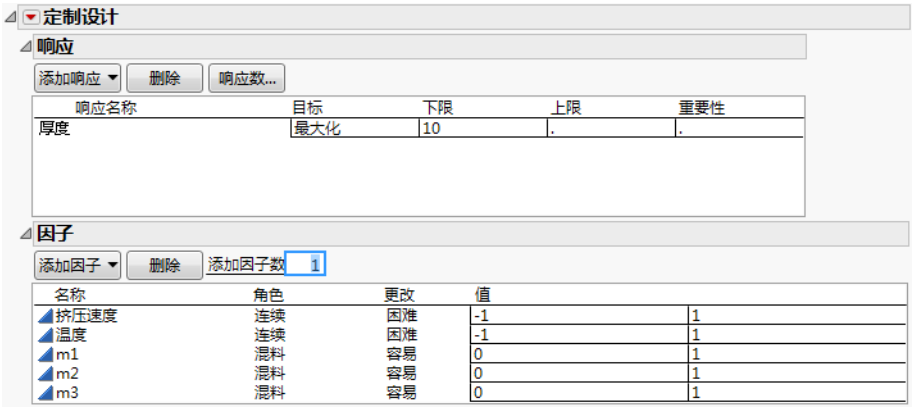
   保留这两个因子的默认值为 –1 和 1。
7. 点击**容易**，然后为**挤压速度**和**温度**均选择**困难**。

   这样即可将**挤压速度**和**温度**都设置为整区因子。

8. 在**添加因子数**旁边键入 3。
9. 点击**添加因子** > **混料**。
10. 将三个混料因子重命名为 m1、 m2 和 m3。

    保留这三个因子的默认值为 0 和 1。

图 5.72 “响应”和“因子”分级显示项



11. 点击**继续**。
12. 点击**交互作用** > **二次**。
13. 点击**确定**取消显示说明性消息。

    请注意， 3 是 “整区数”的默认值。
14. 在**整区数**旁边键入 7。
15. 在**用户指定**旁边键入 28。

注意：在第 16 步中设置 “随机种子”，在第 17 步中设置 “开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

16. （可选）点击 “定制设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
17. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 5，然后点击**确定**。
18. 点击**制作设计**。

图 5.73 “设计”分级显示项

设计

试验	整区	挤压速度	温度	m1	m2	m3
1	1	1	1	1	0	0
2	1	1	1	0.5	0.5	0
3	1	1	1	0	0.5	0.5
4	1	1	1	0	0	1
5	2	-1	1	0	0	1
6	2	-1	1	0	1	0
7	2	-1	1	0.5	0	0.5
8	2	-1	1	1	0	0
9	3	-1	-1	0	1	0
10	3	-1	-1	0.4	0	0.6
11	3	-1	-1	0	0.5	0.5
12	3	-1	-1	1	0	0
13	4	1	-1	0	0.5	0.5
14	4	1	-1	1	0	0
15	4	1	-1	0	1	0
16	4	1	-1	0	0	1
17	5	1	1	1	0	0
18	5	1	1	0	1	0
19	5	1	1	0.5	0	0.5
20	5	1	1	0	0	1
21	6	-1	-1	1	0	0
22	6	-1	-1	0	1	0
23	6	-1	-1	0.5	0.5	0
24	6	-1	-1	0	0	1
25	7	1	-1	0	0	1
26	7	1	-1	0.6	0	0.4
27	7	1	-1	0	1	0
28	7	1	-1	0.560358	0.439642	0

请注意：整区因子**挤压速度**和**温度**根据**整区**因子的水平重置了七次。在**整区**的每个水平内，混料成分 m1、m2 和 m3 的设置都是随机分配的。

分析结果

Vinyl Data.jmp 样本数据表包含使用以前版本的 JMP 所创建的设计的实验结果。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Vinyl Data.jmp。

   该样本数据表包含 28 次试验和响应值。在该表中，您使用 “定制设计”平台创建的设计设置不同于 Vinyl Data.jmp 设计中使用的设置。

2. 在 “表”面板中，点击**模型**脚本旁边的绿色小三角。

图 5.74 拟合模型窗口

请注意 “拟合模型”窗口中的以下事项：

- 因子**整区**具有称为“随机效应”（**& 随机**）的属性。这样即可指定**整区**的水平为随机实现值。这些水平值有关联的误差项。
- 使用的分析方法为 **REML（推荐）**。因为模型包含随机效应，所以准确指定了该方法。有关 REML 模型的详细信息，请参见《拟合线性模型》。

提示：在 “拟合模型”窗口中， JMP Pro 用户可将 “特质”更改为 “混合模型”。

3. 点击**运行**。

图 5.75 裂区分析结果

参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|
m1	9.4464782	0.989604	7.99	9.55	<.0001*
m2	5.378571	1.120267	10.35	4.80	0.0007*
m3	5.5546291	1.030955	8.786	5.39	0.0005*
挤压速度*温度	-1.866921	0.740748	2.858	-2.52	0.0903
挤压速度*m1	-0.647611	0.958111	7.261	-0.68	0.5200
挤压速度*m2	0.2944425	0.997579	8.162	0.30	0.7752
挤压速度*m3	2.6611621	0.979087	7.733	2.72	0.0272*
温度*m1	-1.8151	0.961061	7.328	-1.89	0.0990
温度*m2	1.2044413	0.993319	8.065	1.21	0.2596
温度*m3	-1.498647	0.97904	7.731	-1.53	0.1657
m1*m2	-6.458845	5.83806	13.36	-1.11	0.2881
m1*m3	8.5532871	5.607183	13.15	1.53	0.1508
m2*m3	10.575333	5.226599	13.04	2.02	0.0640

随机效应预测

REML 方差分量的估计值

随机效应	方差比	方差分量	标准误差	95% 下限	95% 上限	占合计的百分比
整区	0.6350069	2.476748	2.9711536	-3.346606	8.3001021	38.838
残差		3.9003483	1.5898101	2.0073487	10.607283	61.162
合计		6.3770963	3.1515558	2.9308168	22.945461	100.000

-2 对数似然 = 82.032495402
注意: 合计是正方差分量的总和。
包括负估计值的合计 = 6.3770963

“参数估计值”报表显示：三个混料成分以及**挤压速度***m3 交互作用在 0.05 水平下显著。

“REML 方差分量的估计值”报表指示与**整区**关联的方差分量为 2.476748。这占了总变异的 38.838%。由此断定与整区复制值关联的误差项小于残差（或区内）误差项。

双因子裂区实验

双因子裂区（亦称带区或分裂区组）设计由两个裂区部分构成。在工业中，当诸多批次的材料或实验单元从一个处理阶段过渡到另一个处理阶段时，就适用上述设计。要使用双因子裂区设计，您必须能够在两个阶段之间对单元重新排序。

在第一个处理阶段之后，您必须能够将批次细分为子批次。将向这些子批次随机应用第二阶段处理因子。对于特定的第二阶段实验设置，分配至该设置的所有子批次可以同时处理。第二个处理阶段后，可以将其他因子应用到实验单元。

与裂-裂区设计相反，第二阶段因子**不嵌套**在第一阶段因子内。第一阶段之后，这些批次被细分并形成新的批次。因此，第一阶段因子和第二阶段因子将同时应用到整个批次。

尽管因子在这两个阶段可能同样难以更改，但为了区分这些因子，JMP 将第一阶段因子表示为**极难更改**，将第二阶段因子表示为**难以更改**。在第二个处理阶段之后应用到实验单元的其他因子被视为**易于**更改因子。

双因子裂区设计方案

该示例基于一项旨在改善电池中的开路电压 (OCV) 的实验 (Vivacqua and Bisgaard, 2004)。您需要最小化 OCV 以便防止电池自行放电。

电池需经过两个处理阶段：

- 第一阶段：一个连续组装过程，该过程加工电池时是以 2000 粒电池为一批。
- 第二阶段：在可以容纳 4000 粒电池的室内，以 5 天为一个周期的固化过程。

您想要研究六个两水平连续因子：

- 四个因子（A1、A2、A3 和 A4）应用于组装过程。您可以为第一阶段因子运行 16 次试验。
- 两个因子（C5 和 C6）应用于固化过程。由于固化过程需要以 5 天为一个周期的时间，您只能为第二阶段因子运行 6 个周期（30 天）。使用六个固化周期让您能够部分复制固化设置，这样您便可以检验固化效应。

第一阶段因子和第二阶段因子均难以更改，建议两个裂区。不过，第一阶段实验的 2,000 粒电池的批次可细分为 500 粒电池为一批的子批次。可随机选择八个这样的子批次并在固化室中同时处理。

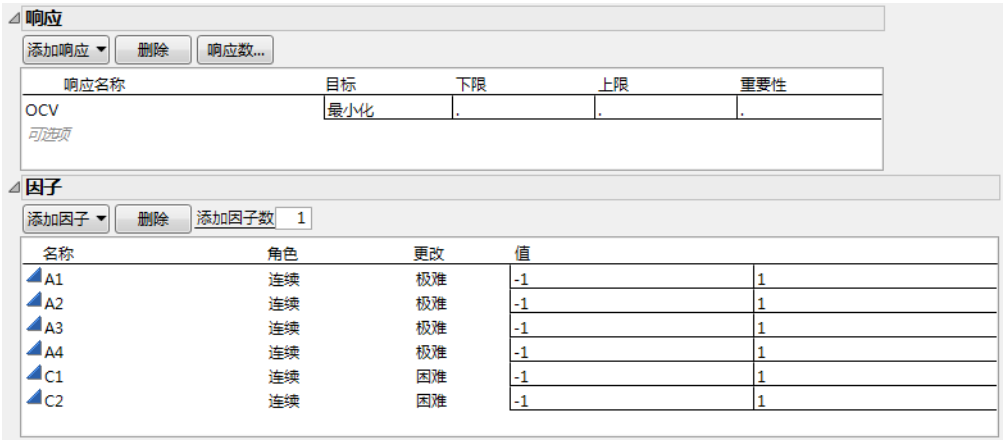
该实验有 48 个实验单元。请注意，第一阶段因子和第二阶段因子之间**交叉**。

创建设计

要设计双因子裂区实验，请执行以下步骤：

1. 选择**实验设计 > 定制设计**。
2. 双击 “响应名称”下的 Y 并键入 OCV。
3. 在 “目标”下，点击**最大化**并选择**最小化**。
4. 要手动添加因子，请执行第 5 步到第 10 步。或者，要从保存的表加载因子：
   a. 点击 “定制设计”红色小三角并选择**加载因子**。
   b. 打开 Design Experiment 文件夹中的 Battery Factors.jmp 样本数据表。
   c. 前进到第 11 步。
5. 在**添加因子数**旁边键入 6。
6. 点击**添加因子 > 连续**。
7. 将因子重命名为 A1、 A2、 A3、 A4、 C1 和 C2。
   保留这些因子的默认值为 -1 和 1。
8. 对于每个因子 （A1、 A2、 A3 和 A4），在 “更改”下，点击**容易**，然后将其更改为**极难**。
   要区分第一阶段因子和第二阶段因子，您可以将第一阶段因子的 “更改”指定为 “极难”，将第二阶段因子的 “更改”指定为 “困难”。
9. 对于每个因子 （C1 和 C2），在 “更改”下，点击**容易**，然后将其更改为**困难**。

图 5.76 “响应”和“因子”分级显示项



10. 点击**继续**。
11. 在“模型”分级显示项中，选择**交互作用 > 二次**。
12. 在“生成设计”分级显示项中，选择选项**难以更改的因子可独立于极难更改的因子发生变化**。

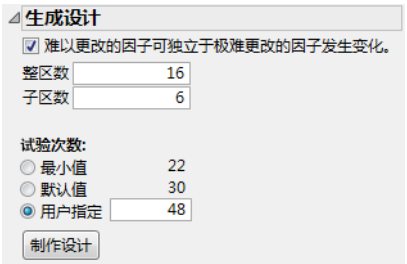
    请参见图 5.77。选中该选项将创建双因子裂区设计。若未选中该选项，该设计将被视为裂 - 裂区设计，因子在两个水平上嵌套。
13. 在**整区数**旁边键入 16。

    这是您可以对第一阶段因子运行的试验数。
14. 在**子区数**旁边键入 6。

    这是您可以对第二阶段因子运行的试验数。
15. 在“试验次数”下的**用户指定**旁边键入 48。

    这是实验单元的总数。

图 5.77 “生成设计”分级显示项



注意：在第 16 步中设置“随机种子”，在第 17 步中设置“开始数”，这将会重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

16. （可选）点击 “定制设计”红色小三角，选择**设置随机种子**，键入 1866762673，然后点击**确定**。
17. （可选）点击 “定制设计”红色小三角，选择**开始数**，键入 21，然后点击**确定**。
18. 点击**制作设计**。
19. 点击**制表**。

图 5.78 设计表的部分视图

	整区	子区	A1	A2	A3	A4	C1	C2	OCV
1	1	1	-1	1	-1	1	-1	1	•
2	1	2	-1	1	-1	1	1	1	•
3	1	3	-1	1	-1	1	1	-1	•
4	2	4	1	1	-1	1	1	1	•
5	2	5	1	1	-1	1	-1	1	•
6	2	6	1	1	-1	1	-1	-1	•
7	3	1	-1	-1	-1	1	-1	1	•
8	3	2	-1	-1	-1	1	1	1	•
9	3	3	-1	-1	-1	1	1	-1	•
10	4	4	-1	-1	-1	-1	1	1	•
11	4	5	-1	-1	-1	-1	-1	1	•
12	4	6	-1	-1	-1	-1	-1	-1	•
13	5	1	1	1	-1	-1	-1	1	•
14	5	2	1	1	-1	-1	1	1	•
15	5	3	1	1	-1	-1	1	-1	•
16	6	4	1	-1	-1	-1	1	1	•

设计表为**整区**显示了 16 个水平。对于**整区**的每个水平，四个组装因子的设置是不变的。对于**整区**的每个水平，三个批次、每批 500 粒电池（**子区**）随机分配给固化因子的设置。重复了两组固化条件（C1 = -1，C2 = 1 和 C1 = 1，C2 = 1）。要查看这些条件，请选择列 C1 和 C2，在列标题区域中右击，然后选择**排序 > 升序**。

分析结果

Battery Data.jmp 样本数据表包含您生成的设计的实验结果。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Battery Data.jmp。
2. 在 “表”面板中，点击**模型**脚本旁边的绿色小三角。

   请注意 “拟合模型”窗口中的以下事项：

   – 因子**整区**具有称为“随机效应”（**& 随机**）的属性。这样即可指定**整区**的水平为随机实现值。这些水平值有关联的误差项。
   – 因子**子区**也具有“随机效应”属性（**&随机**）。
   – 使用的分析方法为 **REML（推荐）**。因为模型包含随机效应，所以准确指定了该方法。有关 REML 模型的详细信息，请参见《拟合线性模型》。

提示：在“拟合模型”窗口中，JMP Pro 用户可将“特质”更改为“混合模型”。

3. 选中**保持对话框打开**选项。

4. 点击运行。

图 5.79 完整模型的报表

参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|
截距	38.183196	1.673404	2.075	22.82	0.0016*
A1	-4.657571	1.093074	2.916	-4.26	0.0251*
A2	0.147742	1.093074	2.916	0.14	0.9013
A3	0.9486795	1.093074	2.916	0.87	0.4510
A4	4.0716925	1.190175	3.214	3.42	0.0376*
C1	-1.827487	1.606207	0.868	-1.14	0.4804
C2	-15.73288	1.35446	0.719	-11.62	0.1065
A1*A2	0.1289012	1.080526	2.784	0.12	0.9131
A1*A3	-1.114015	1.080526	2.784	-1.03	0.3837
A1*A4	-1.92398	1.176342	3.04	-1.64	0.1993
A1*C1	-3.271232	0.480531	20.65	-6.81	<.0001*
A1*C2	3.737258	0.478635	20.09	7.81	<.0001*
A2*A3	0.3768179	1.080526	2.784	0.35	0.7520
A2*A4	1.4693535	1.176342	3.04	1.25	0.2992
A2*C1	1.9072313	0.480531	20.65	3.97	0.0007*
A2*C2	-0.161179	0.478635	20.09	-0.34	0.7398
A3*A4	0.0764369	1.176342	3.04	0.06	0.9522
A3*C1	0.1761619	0.480531	20.65	0.37	0.7176
A3*C2	-0.445242	0.478635	20.09	-0.93	0.3633
A4*C1	0.5306856	0.511896	20.84	1.04	0.3118
A4*C2	1.3506119	0.502894	20.12	2.69	0.0142*
C1*C2	-0.425326	1.606207	0.868	-0.26	0.8401

随机效应预测

REML 方差分量的估计值

随机效应	方差比	方差分量	标准误差	95% 下限	95% 上限	占合计的百分比
整区	1.6095523	15.451509	15.876576	-15.66601	46.569027	45.968
子区	0.8919389	8.562507	16.314188	-23.41271	40.537728	25.473
残差		9.5998804	3.0299487	5.6239301	19.985723	28.559
合计		33.613897	16.824615	15.325965	123.59174	100.000

-2 对数似然 = 227.810084
注意: 合计是正方差分量的总和。
包括负估计值的合计 = 33.613897

“参数估计值”报表指示四个双因子交互作用（A1*C1、A1*C2、A2*C1 和 A4*C2）以及两个主效应（A1 和 A4）在 0.05 水平下显著。

5. 在 Battery Data.jmp 的“表”面板中，点击简化模型 1 脚本旁边的绿色小三角。

该脚本可打开“拟合模型”窗口，其中已删除不显著的交互作用。剩余效应均为主效应，另有四个双因子交互作用（A1*C1、A1*C2、A2*C1 和 A4*C2）。您采用了保守方式简化模型。

6. 点击运行。

图 5.80 初步简化模型的报表

参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|
截距	38.249531	1.373127	7.723	27.86	<.0001*
A1	-4.723906	1.175023	8.5	-4.02	0.0034*
A2	0.0497917	1.165418	8.228	0.04	0.9669
A3	0.7560417	1.165418	8.228	0.65	0.5342
A4	4.2328205	1.207654	9.16	3.50	0.0065*
C1	-2.195626	0.866042	1.083	-2.54	0.2234
C2	-15.79922	0.840903	1.008	-18.79	0.0331*
A1*C1	-3.283225	0.452505	25.17	-7.26	<.0001*
A1*C2	3.8035937	0.4498	24.42	8.46	<.0001*
A2*C1	1.8797279	0.452505	25.17	4.15	0.0003*
A4*C2	1.3877006	0.482556	24.95	2.88	0.0081*

请注意，主效应 C2 现在在 0.05 水平下显著 (概率 >|t| = 0.0331)

7. 在 “拟合模型” 窗口中，删除 A3。

   主效应 A3 是唯一不显著且不涉及双因子交互作用的主效应。

8. 点击**运行**。

图 5.81 简化模型的报表

参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|
截距	38.249531	1.362184	6.639	28.08	<.0001*
A1	-4.723906	1.137313	9.131	-4.15	0.0024*
A2	0.0497917	1.127389	8.818	0.04	0.9658
A4	4.2131272	1.173862	9.921	3.59	0.0050*
C1	-2.166553	0.899138	1.037	-2.41	0.2432
C2	-15.79922	0.874257	0.963	-18.07	0.0389*
A1*C1	-3.282201	0.452234	25.24	-7.26	<.0001*
A1*C2	3.8035938	0.449735	24.43	8.46	<.0001*
A2*C1	1.8843473	0.452234	25.24	4.17	0.0003*
A4*C2	1.3851686	0.4823	24.91	2.87	0.0082*

随机效应预测

REML 方差分量的估计值

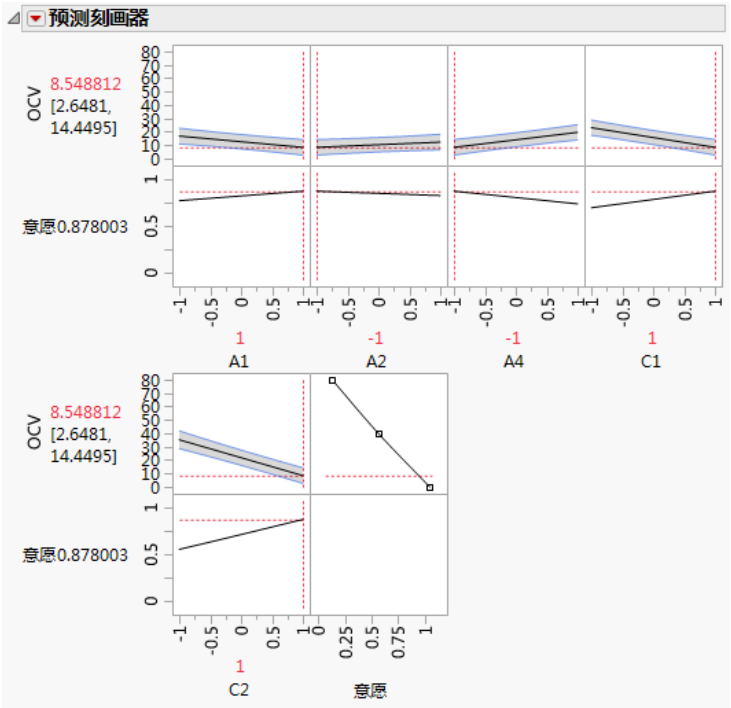
随机效应	方差比	方差分量	标准误差	95% 下限	95% 上限	占合计的百分比
整区	2.0231563	17.459497	9.7154555	-1.582446	36.50144	60.025
子区	0.3473621	2.9976761	5.8732929	-8.513766	14.509119	10.306
残差		8.6298312	2.4692519	5.2816799	16.588508	29.669
合计		29.087005	9.1941655	17.028461	60.633509	100.000

-2 对数似然 = 254.17861349
注意: 合计是正方差分量的总和。
包括负估计值的合计 = 29.087005

“REML 方差分量的估计值” 报表显示与**整区**关联的方差分量大约为**子区**方差分量的六倍。这表明组装过程比固化过程更容易发生变化。此外，内部（残差）误差也比**子区**的内部误差大。要减小变异，应侧重于尽量减少组装过程中的差异以及电池的个体差异。

9. 点击 “响应 ‘OCV’” 红色小三角并选择**因子刻画 > 刻画器**。
10. 点击 “预测刻画器” 红色小三角并选择**优化和意愿 > 最大化意愿**。

图 5.82 具有可最小化 OCV 的设置的预测刻画器



该刻画器显示了标识为活跃的五个因子以及可最小化 OCV 的设置。

针对稳健过程和产品设计的实验

过程或产品往往会受到噪声因子的影响。有一些难以控制或控制成本较高的噪声因子。最好找到受噪声因子影响最小的可控因子的过程的或产品的设置。以往，田口设计提供了一种在存在噪声因子时进行实验的方法。请注意，必须在实验期间控制噪声因子。

田口设计的备选设计包括组合数组和混合分辨率设计 (Borror and Montgomery, 2000)。混合分辨率设计用于获取同时包含控制因子和噪声因子的设计。这种设计的目标是查找控制变量的稳健设置。关键是估计特定效应的能力。特别是，人们关注的是估计控制变量的主效应、噪声变量的主效应、控制*控制交互作用、噪声*控制交互作用和控制变量的二次项。

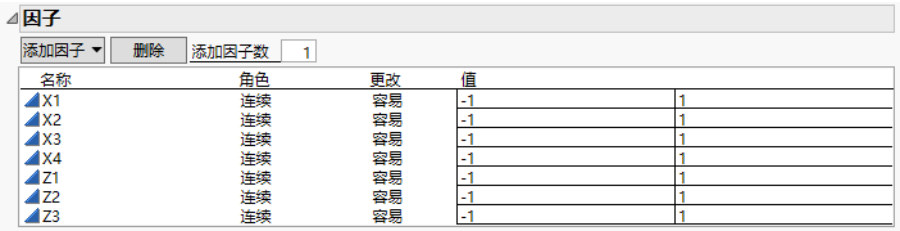
使用定制设计器可为研究稳健过程设置生成设计。

在本例中，您关注的是如何在存在三个噪声因子的情况下查找四个控制设置的最优设置。

1. 选择**实验设计** > **定制设计**。
2. 在 “因子”分级显示项中的**添加因子数**旁边键入 4。
3. 点击**添加因子** > **连续**。

4. 在 “因子”分级显示项中的**添加因子数**旁边键入 3。
5. 点击**添加因子** > **连续**。
6. 双击 X5 并将其改为 Z1 以指定噪声因子。重复该过程以将 X6 和 X7 改为 Z2 和 Z3。

图 5.83 使用 4 个控制因子和 3 个噪声因子的设计的 “因子”分级显示项



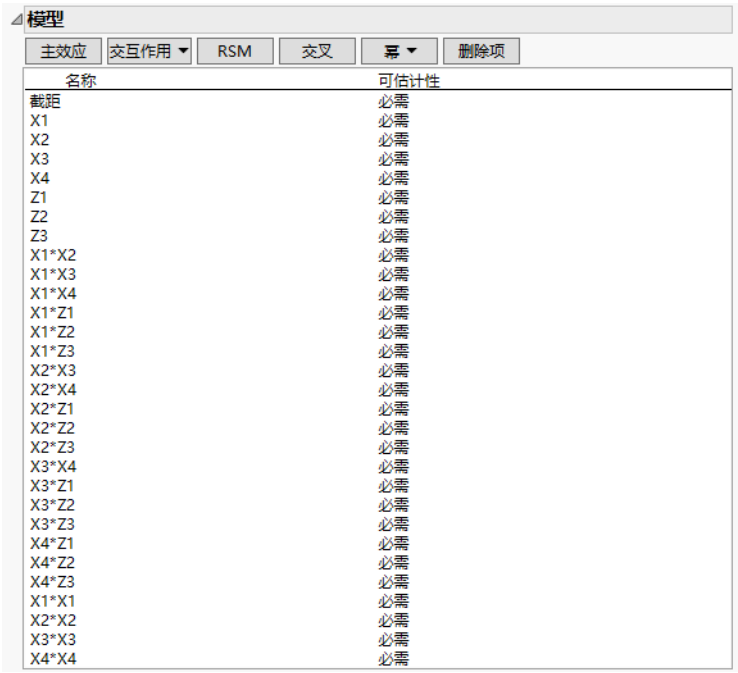
7. 点击 “继续”。

接下来，您将建立模型，以探索受噪声因子影响最小的控制因子的最佳设置。

8. 该模型默认包含所有主效应。
9. 点击**交互作用** > **二次**以添加所有双因子交互作用。

   这将输入控制 * 控制交互作用、控制 * 噪声交互作用和噪声 * 噪声交互作用。您不想估计噪声因子之间的相互作用。
10. 滚动至列表底部，突出显示 Z1*Z2、Z1*Z3 和 Z2*Z3，然后点击 “删除项”。这将删除噪声 * 噪声交互作用。
11. 在 “因子”分级显示项中，选择 X1、 X2、 X3 和 X4。
12. 在 “模型”分级显示项中，点击 “幂” > “二次”以添加控制变量二次项。

图 5.84 具有噪声因子的设计的 “模型” 分级显示项


模型

主效应 | 交互作用 | RSM | 交叉 | 幂 | 删除项

名称	可估计性
截距	必需
X1	必需
X2	必需
X3	必需
X4	必需
Z1	必需
Z2	必需
Z3	必需
X1*X2	必需
X1*X3	必需
X1*X4	必需
X1*Z1	必需
X1*Z2	必需
X1*Z3	必需
X2*X3	必需
X2*X4	必需
X2*Z1	必需
X2*Z2	必需
X2*Z3	必需
X3*X4	必需
X3*Z1	必需
X3*Z2	必需
X3*Z3	必需
X4*Z1	必需
X4*Z2	必需
X4*Z3	必需
X1*X1	必需
X2*X2	必需
X3*X3	必需
X4*X4	必需

13. 在**中心点数**：文本框中输入 2 来添加两个中心点。
14. 点击**制作设计**。

结果是一个有 36 次试验的设计，该设计支持您在存在三个噪声因子的情况下评估和优化控制因子。分析的目的是确定哪些效应是重要的，然后为受噪声因子影响最小的控制因子选择设置。

第 6 章

扩充设计

向现有设计添加额外试验

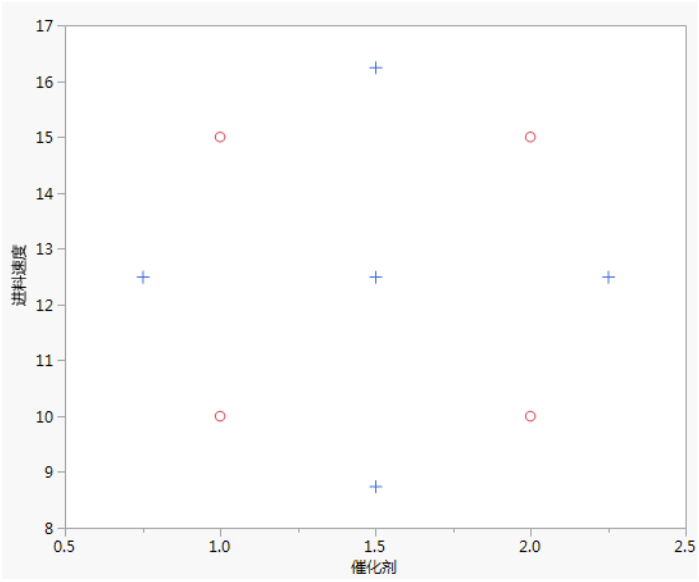
使用“扩充设计”平台可向现有实验设计添加试验。实验往往是一个序贯过程，扩充已设计的实验有助于解决单个设计带来的不明确性。

对于现有设计表，“扩充设计”平台构造额外试验来优化总体设计。您可以添加试验来实现以下目标：

- 将设计重复指定的次数。
- 添加中心点。
- 创建折叠设计。
- 将轴点随中心点一起添加，以将筛选设计转换为响应曲面设计。
- 向设计添加空间填充点。
- 向设计添加试验以便拟合特定模型。

有关扩充设计的详细信息，请参见 Goos and Jones (2011)。

图 6.1 原始设计试验 (+) 和扩充试验 (o)



目录

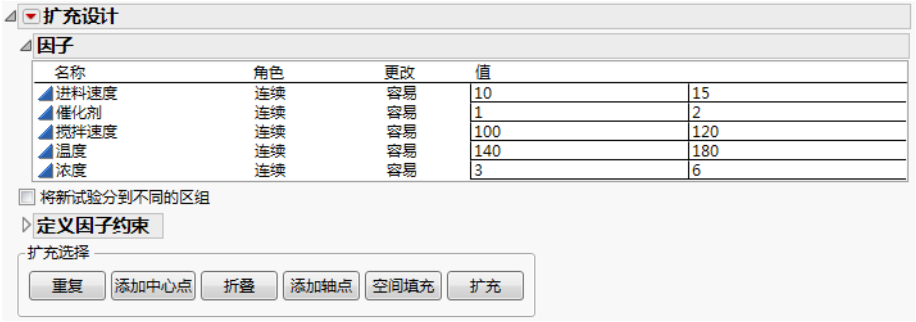
使用“扩充设计”平台的示例 193
启动“扩充设计”平台 199
“扩充设计”窗口 200
因子 200
定义因子约束 201
扩充选择 203
“扩充设计”平台选项 204
扩充选择的更多示例 207
重复设计 207
向设计添加中心点 209
创建折叠设计 209
向设计添加轴点 210
空间填充扩充 211
统计详细信息 212

使用 “扩充设计”平台的示例

使用 “扩充设计”平台向设计添加额外试验。本例演示如何使用 “扩充设计”平台解决筛选设计结果中的不确定性。在本研究中，一位化学工程师调查了五个因子对于某化学反应过程的反应百分比的效应。原始设计是一个适于拟合主效应的 8 次试验的设计。本例扩充这一 8 次试验设计，以便可以使用总共 16 次试验估计所有双因子交互作用。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor 8 Runs.jmp。
2. 选择**实验设计** > **扩充设计**。
3. 选择**反应百分比**并点击 **Y，响应**。
4. 选择**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**，然后点击 **X，因子**。
5. 点击**确定**。

图 6.2 反应器示例的因子



注意：您可以选择**将新试验分到不同的区组**以将分区组因子添加到任何设计。但是，请记住本示例是在 16 次试验中估计所有双因子交互作用，当在模型中有其他分区组因子时这不可能做到。

6. 点击**扩充**。

图 6.3 中所示的模型使用数据表中的 “模型”脚本定义。通过 8 次试验，您可以估计至多 7 个效应。该脚本中的模型包含全部 5 个主效应和两个交互作用项。所有项都定义为模型的必需项。有关效应可估计性的详细信息，请参见 “定制设计”一章中第 91 页的 “模型”。

图 6.3 初始 “扩充设计”平台

7. 在 “模型”分级显示项中，选择**交互作用 > 二次**。

   这将所有双因子交互作用添加到模型。为指定模型给出的最小试验次数为 16，如 “生成设计”文本编辑框中所示。

   注意：在第 8 步中设置 “随机种子”，在第 9 步中设置 “开始数”，这将会重现图 6.4 中显示的结果。自行构造设计时，这些步骤是可选的。

8. （可选）从“扩充设计”红色小三角中，选择**设置随机种子**，键入 282322901，然后点击**确定**。
9. （可选）点击 “扩充设计”红色小三角，选择**开始数**，键入 800，然后点击**确定**。
10. 点击**制作设计**。
11. 点击**设计**展开图标以查看设计。

图 6.4 16 次试验的扩充设计

设计

试验	进料速度	催化剂	搅拌速度	温度	浓度
1	15	2	100	140	3
2	10	2	120	140	6
3	15	1	120	140	3
4	10	2	100	180	3
5	10	1	120	180	3
6	10	1	100	140	6
7	15	2	120	180	6
8	15	1	100	180	6
9	15	1	120	180	6
10	10	2	100	140	6
11	10	1	100	180	6
12	15	2	120	140	6
13	15	2	120	180	3
14	10	2	120	180	3
15	10	1	100	140	3
16	10	1	120	140	3

12. 点击**制表**生成包含带结果的原始设计和扩充试验的设计表。

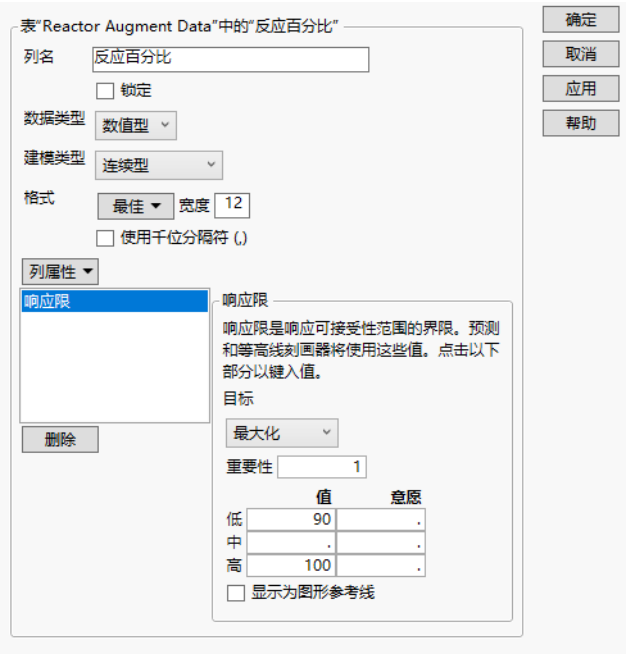
分析扩充设计

继续演示扩充设计示例，假定您已经执行了附加的试验并且已经记录了要分析的结果。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor Augment Data.jmp。

   您想要最大化**反应百分比**。请注意，该样本数据表中该列的 “响应限”列属性设置为**最小化**。下限和上限分别设置为 90 和 100。

图 6.5 “反应百分比”列的 “响应限”列属性



2. 在数据表的 “表”面板中，点击**模型**脚本旁边的绿色小三角。

   模型脚本打开 “拟合模型”窗口，其中所有的主效应和双因子交互作用都显示为效应。
3. 将拟合特质从**标准最小二乘法**改为**逐步**，然后点击**运行**。
4. 在 “逐步回归控制”面板中，从**停止规则**菜单中选择 **p 值阈值**，从**方向**菜单中选择**混合**。为**进入的概率**输入 0.05，为**剔除的概率**输入 0.05。
5. 在 “当前估计值”面板中，为所有主效应项 （进料速度 - 浓度）选中**已进入**复选框。

图 6.6 已完成的逐步模型

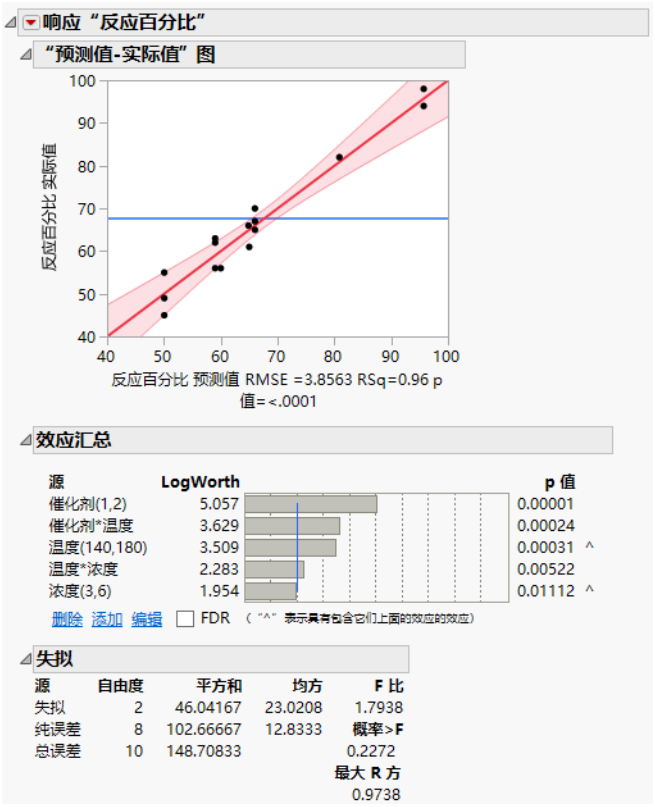
6. 点击**执行**。

   逐步回归，从主效应开始并使用基于 p 值的搜索，将生成包含三个主效应和两个交互作用项的模型。点击 “逐步回归”控制面板上的**构建模型**。

   这将为使用逐步过程构建的模型启动 “拟合模型”平台。

7. 点击**运行**。

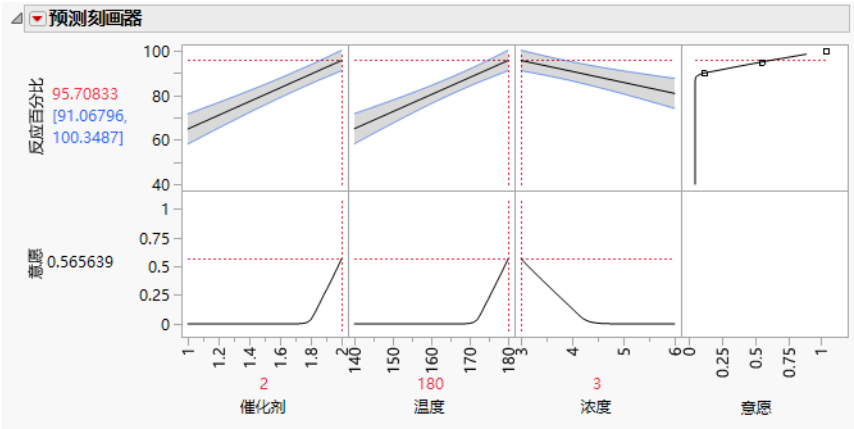
图 6.7 预测模型 “方差分析”和 “失拟检验”



“‘预测值 - 实际值’图”指示总体模型显著 （p 值 < .0001）。“‘预测值 - 实际值’图”和“失拟”检验均显示没有证据表明模型指定有误。“效应汇总”报表显示**催化剂**是最显著的效应。

8. 点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 6.8 最大反应百分比

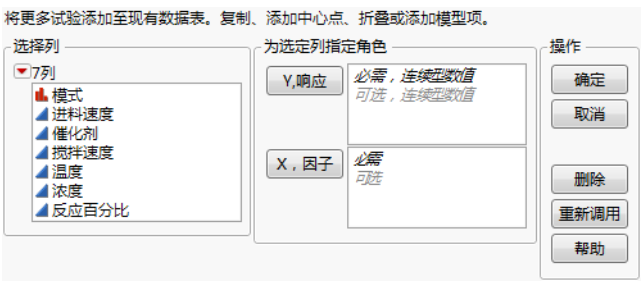


预测刻画图显示最大反应百分比出现在**催化剂**和**温度**的高水平以及**浓度**的低水平上。在这些极端设置上，**反应百分比**的估计值为 95.7。在预测响应落在响应限值之外的区域，意愿刻画变平缓。

启动 “扩充设计” 平台

通过从包含您要扩充的设计的数据表中选择**实验设计** > **扩充设计**，启动 “扩充设计” 平台。

图 6.9 “扩充设计” 启动窗口



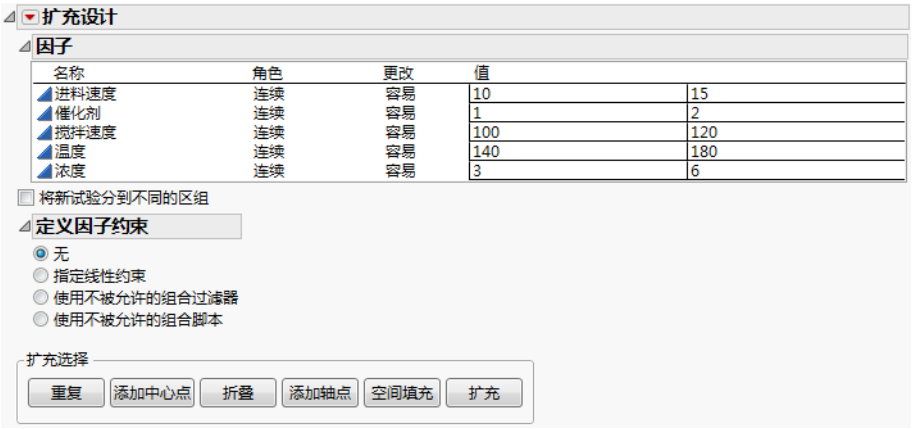
Y，响应 输入一个或多个数值响应列。必须输入响应。

X，因子 输入因子列。因子可以为任意数据类型或建模类型。

“扩充设计”窗口

初始“扩充设计”窗口包含“因子”和“定义因子约束”分级显示项以及“扩充选择”面板。

图 6.10 使用 Reactor 8 Runs.jmp 的初始“扩充设计”窗口



因子

这些因子是您在“扩充设计”启动窗口中指定的因子。

名称 列出在“扩充设计”启动窗口中列为 **X，因子**的所有因子，具有“随机区组”设计角色列属性的因子除外。

角色 指定在列属性中指定的因子的“设计角色”。若该因子不具有“设计角色”列属性而且是一个常数，则**常数**会显示在“角色”列中。如若不然，该因子的建模类型会显示在“角色”列中。

更改 指示因子水平是否如数据表中的“因子更改”列属性所指定的那样设置为“容易”、“困难”或“极难”更改。若因子不具有“因子更改”列属性，则“更改”指定为“容易”。

注意：若某个因子具有“因子更改”列属性且属性设置为“困难”或“极难”，则必须在“扩充设计”启动窗口的 **X，因子**列表中包括相应的整区因子。

值 对于连续因子，显示最小值和最大值。对于分类因子，显示水平。

提示：具有“分类”或“常数”角色的因子在“名称”列中显示有向下箭头图标。点击向下箭头可添加水平。若该因子为“常数”且具有分类建模类型，则可以添加多个水平。若该因子为“常数”且具有连续建模类型，则只能添加一个水平。

将新试验分到不同的区组 向设计添加分区组因子，使一个区组包含原始设计，一个区组包含扩充试验。

定义因子约束

若您使用“空间填充”或“扩充”选项扩充设计，可以在设计空间上针对添加的试验定义限制条件。

使用“定义因子约束”来限制设计空间。除非您已经加载某个约束或在脚本中包含了某个约束，否则会选定**无**选项。要指定约束，请选择以下其他选项之一：

指定线性约束 指定针对因子的线性组合的不等式约束。仅适用于具有“连续”或“混料”角色的因子。请参见“指定线性约束”。

注意: 保存涉及线性约束的设计的脚本时，该脚本将线性约束表示为**小于等于**不等式 (≤)。

使用不被允许的组合过滤器 基于各个因子的限定值定义各组约束。您可以定义“与”和“或”约束。请参见“使用不被允许的组合过滤器”。

使用不被允许的组合脚本 定义不被允许的组合和其他约束，作为脚本编辑器框中的布尔 JSL 表达式。请参见“使用不被允许的组合脚本”。

注意：当分析具有因子约束的设计时，模型刻画器将接受这些约束。

指定线性约束

当连续因子不可能在设计空间上独立变化时，可以指定线性不等式约束。线性不等式描述了允许的因子水平设置。

点击**添加**以输入一个或多个线性不等式约束。

添加 添加涉及设计中所有连续因子的线性表达式的模板。输入因子的系数值并选择不等式的方向来反映您的线性约束。在不等式右侧的框中指定约束值。要添加更多约束，请再次点击**添加**。

注意：若您已通过指定球形半径限制了设计区域，将禁用“添加”选项。

删除最后一个约束 删除最后一个约束。

检查约束 检查约束的一致性。该选项删除多余的约束并执行可行性检查。若有问题将显示 JMP 警示。若约束等效于因子的界限， JMP 警示将指示“因子”分级显示项中的界限已更新。

使用不被允许的组合过滤器

该选项使用经过改编的数据过滤器帮助指定不被允许的组合。有关详细信息，请参见《使用 JMP》中的“JMP 报表”一章。

从“添加过滤器因子”列表中选择因子并点击**添加**。然后通过使用滑块（对于连续因子）或选择水平（对于分类因子）来指定不被允许的组合。

“添加过滤器因子”菜单的红色小三角选项在许多平台启动窗口的“选择列”面板中都可以找到。请参见《使用 JMP》中的“入门”一章。

点击“添加”时，“不被允许的组合”控制面板显示选定的因子并提供用于进一步控制的选项。基于建模类型，按以下方式表示因子：

连续因子 对于连续因子，将显示跨越因子设置范围的双端滑块。您可以指定不允许的设置，方法是拖动滑块两端或通过点击滑块下的文本值来设置端点。在滑块中，用纯蓝色突出表示不被允许的值。

分类因子 对于分类因子，可能的水平显示为带标签的区组，或者是在水平数较大时显示为列表条目。选择某个水平使其不被允许。要选择多个水平，请按住 Ctrl 键。区组或列表条目将突出显示，以指明不被允许的水平。向“不被允许的组合”面板添加分类因子时，在因子名称后面的括号中给出分类因子的水平数。

不被允许的组合选项

控制面板具有以下控件：

清除 清除您已指定的所有不被允许的因子水平设置。这不会清除选定的因子。

重新开始 删除所有选定因子，并返回最初的因子列表。

与 打开“添加过滤器因子”列表。选定的因子将变为一个“与”组。在“与”组内指定的任意因子水平组合都不被允许。

之后要向“与”组添加因子，请点击该组的分级显示项以查看突出显示的矩形。选择“与”并添加因子。

要删除单个因子，请从该因子的红色小三角菜单中选择**删除**。

或 打开“添加过滤器因子”列表。选定的因子将变为单独的“与”组。对于由“或”分隔的“与”组，若至少在一个“与”组中指定了某个组合，则该组合不被允许。

因子的红色小三角选项

某个因子可以显示在多个“或”组中。特定“或”组中因子的一次出现称为该因子的一个**实例**。

删除 从“不被允许的组合”面板中删除选定的因子实例。

清除选择 清除该因子实例的任何选择。

反转选择 取消选择选定值，并选择之前没有为该因子实例选定的那些值。

显示选项 仅可用于分类因子。更改显示的外观。选项包括：将每个水平显示为区组、列表或单个类别，或者在每个值旁边添加复选框。

查找 仅可用于分类因子。在因子名称下提供一个文本框，在此可以为因子水平输入搜索字符串。按 Enter 键或在该文本框外点击可执行搜索。一旦选定**查找**，“查找”选项即显示在红

色小三角菜单中，如：“不包含”、“区分大小写”、“开始字符为”或“结束字符为”，或者“清除查找”。

使用不被允许的组合脚本

使用该选项来通过 JSL 脚本指定不被允许的因子水平的特定组合。该选项可以用于连续因子或既有连续因子又有分类因子的情况。

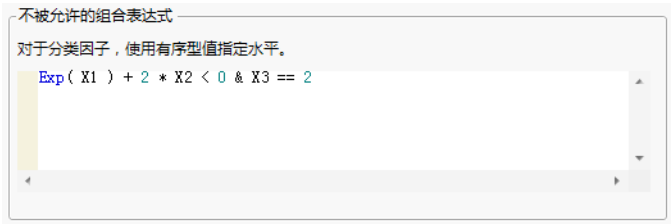
该选项将打开一个脚本窗口，您可以在其中插入用来标识您不想允许的组合的脚本。该脚本必须作为一个布尔表达式来求值。若表达式的计算结果为真，则不允许指定的组合。

创建分类因子的表达式时，使用水平的序数值而非水平的名称。若因子的水平是高、中和低（在“因子”分级显示项中以该顺序指定），则相关的序数值为 1、2 和 3。例如，假定您有两个连续因子 X1 和 X2，还有一个具有三个水平（水平的顺序为 L1、L2 和 L3）的分类因子 X3。您希望不允许符合以下条件的水平：

$$e^{X_1} + 2X_2 < 0 \text{ 并且 } X_3 = L2$$

在脚本窗口中输入表达式 `(Exp(X1) + 2*X2 < 0) & (X3 == 2)`。

图 6.11 脚本编辑器中的表达式



（在图中，通过解析删除了不必要的括号。）请注意函数可以作为布尔表达式的一部分来输入。表达式 `(Exp(X1) + 2*X2 < 0) & (X3 == “L2”)` 也有效。

扩充选择

“扩充设计”平台具有以下扩充选择：

重复 将设计重复指定的次数。请参见第 207 页的“重复设计”。

添加中心点 添加中心点。指定您要作为中心点向设计添加的试验次数。中心点是对于每个连续因子而言，其设置位于高值设置与低值设置中央的试验。请参见“从实验设计入手”一章中第 61 页的“中心点、重复试验和检验”。

若设计包含连续因子和其他类型的因子，则中心点相对于其他因子的水平可能并不均衡。“扩充设计”选择中心点以使设计的 D、I 或别名效率最大化。

请参见第 209 页的“向设计添加中心点”。

折叠 创建折叠设计。请参见第 209 页的“创建折叠设计”。

添加轴点 向设计中添加轴点和中心点。“添加轴点”可用于基于筛选设计生成中心复合设计。请参见第 210 页的“向设计添加轴点”。

空间填充 向包含连续因子的任意设计添加额外试验。使用快速灵活填充方法构造额外试验。请参见第 211 页的“空间填充扩充”。“空间填充”不支持将新试验分组到单独的区组中。

扩充 使用模型向设计添加试验。该模型的项通常比原始模型多。请参见第 193 页的“使用“扩充设计”平台的示例”。

“扩充设计”平台选项

“加速寿命试验计划”红色小三角菜单包含以下选项：

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用“保存响应”选项保存的响应。

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在“设计角色”区域中，选择合适的角色。

加载因子 加载使用“保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在“定义因子约束”或“线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为“小于”或“大于”约束。请参见“列属性”附录中第 730 页的“约束状态”。

加载约束 （对于一些平台不可用）加载使用“保存约束”选项保存的因子约束。

设置随机种子 设置随机种子，JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的“试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示“模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一：“正态”、“二项式”或“Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在 “模型”窗口中做出选择以控制模拟的响应值的分布。当您点击 “应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中， Y 是响应的名称。再次点击 “应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见 “定制设计”一章中第 101 页的 “模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见 《基本分析》中的 “模拟”一章。

保存 X 矩阵 将名为“矩量矩阵”和“模型矩阵”的脚本保存至设计数据表。这些脚本包含矩量和设计矩阵。请参见 “定制设计”一章中第 103 页的 “保存 X 矩阵”。

警告：对于包含名义型因子的设计，使用 “保存 X 矩阵”选项保存的 “模型矩阵”脚本**不是**拟合线性模型时使用的编码矩阵。通过在运行 “模型”脚本时获取的 “拟合模型”报表中选择选项 “保存列” > “保存编码表”，您可以获取用于拟合模型的编码矩阵。

最优性准则 更改设计最优性准则。默认准则**推荐**为所有设计类型指定 D 最优性，除非您使用“模型”分级显示项中的“RSM”按钮添加了二次效应。有关 D、I 和别名最优设计的详细信息，请参见 “定制设计”一章中第 114 页的 “最优性准则”。

注意：您可以设置首选项，以便始终使用给定的最优性准则。选择**文件** > **首选项** > **平台** > **实验设计**。选择 “最优性准则”并选择您偏好的准则。

开始数 支持您指定用于构造设计的随机开始数。请参见 “定制设计”一章中第 105 页的 “开始数”。

设计搜索时间 搜索设计所用的最大秒数。默认搜索时间基于设计的复杂性。请参见 “定制设计”一章中第 105 页的 “设计搜索时间”和 “定制设计”一章中第 105 页的 “开始数”。

若算法迭代需要不止几秒，“正在计算设计”进度窗口随即显示。若点击进度窗口中的**取消**，计算将停止，同时给出目前为止的最佳设计。当设计中不包含 “更改”设置为 “困难”或 “极难”或 “可估计性”设置为 “若可能”的因子时，该进度窗口还显示 D 最优设计的 D 效率。

注意：您可以为 “设计搜索时间”设置首选项。选择**文件 > 首选项 > 平台 > 实验设计**。选择设计搜索时间并输入最大秒数。在需要更长时间的特定情形下， JMP 会延长搜索时间。

球形半径 将设计中的连续因子限定到一个超球面中。指定半径并点击**确定**。挑选的设计点与 0 之间的距离等于球形半径。先选择该选项，然后再点击 “制作设计”。

注意：不能将球形半径约束与使用 “指定线性约束”选项添加的约束结合使用。同样，当包含难以更改的因子时 （裂区设计），该选项也不可用。

高级选项

混料总和 将混料因子的总和设置为任意正值。使用该选项可令混料的成分在实验过程中始终保持不变。

裂区方差比 指定随机整区和子区（若存在）方差与误差方差之间的比值。设置该值之前，您必须为裂区设计定义难以更改的因子，或是为裂-裂区设计定义难以更改和极难更改的因子。然后，您可以为方差比输入一个或两个正数，具体取决于您指定的是裂区设计还是裂-裂区设计。

先验参数方差 （仅当 “模型”分级显示项可用时才可用。）指定对 “可估计性”设置为 “若可能”的因子使用的权重。当您点击 “制作设计”时，该选项随之更新以显示默认权重。为您要指定权重的每个项输入一个正数。您输入的值是先验方差倒数的平方根。较大的值代表较小的方差，因此表明效应不活跃的先验信息越多。

使用 Bayesian D 或 I 最优性来构造包含 “若可能”因子的设计。在算法中使用的默认值为：“必需”项为 0，涉及 “若可能”项的交互作用为 4， “若可能”项为 1。请参见 “技术细节”附录中第 735 页的 “别名矩阵”和 “定制设计”一章中第 114 页的 “最优性准则”。

A 最优参数权重 （用于 A 最优设计。）指定模型参数的权重。这样您就可以对主效应而不是二阶效应（举例来说）的方差赋予更大的权重。有关参数权重的详细信息，请参见 Morgan and Stallings (2017)。

D 效率权重 指定构造设计时 D 效率相对于别名最优性的相对重要性。选择该选项可在减小系数方差与获取所需别名结构之间进行权衡。值应介于 0 到 1 之间。较大值赋予 D 效率更大权重。默认值为 0.5。仅当您将 “制作别名最优设计”选作 “最优性准则”时，该选项才有效果。

有关 D 效率的定义，请参见 “定制设计”一章中第 114 页的 “最优性准则”。有关别名最优性的详细信息，请参见 “定制设计”一章中第 118 页的 “别名最优性”。

将脚本保存至脚本窗口 为您在“定制设计”窗口中指定的设计创建脚本，并将其放入打开的脚本窗口。

扩充选择的更多示例

- “重复设计”
- “向设计添加中心点”
- “创建折叠设计”
- “向设计添加轴点”
- “空间填充扩充”

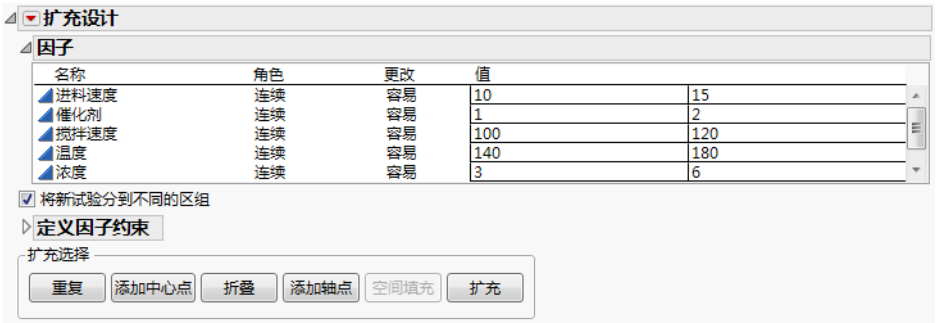
重复设计

使用重复来扩充设计提供了对误差方差是常数这个假设的直接检查。在过程或测量变异性很大时，重复还可减小回归系数的变异性。

本例重复 8 次试验设计。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor 8 Runs.jmp。
2. 选择**实验设计** > **扩充设计**。
3. 选择**反应百分比**并点击 **Y，响应**。
4. 选择**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**，然后点击 **X，因子**。
5. 点击**确定**。

图 6.12 选择扩充类型



6. 选择**将新试验分到不同的区组**以将分区组变量添加到设计中。
7. 点击**重复**，输入 2，然后点击**确定**。

注意：重复对话框要求提供要执行每次试验的次数。输入 2 将指定您希望每个试验在生成的设计中出现两次。这将对原始设计进行一次重复。

图 6.13 使用重复扩充的反应器数据设计

因子设计

试验	进料速度	催化剂	搅拌速度	温度	浓度	区组
1	15	2	100	140	3	1
2	10	2	120	140	6	1
3	15	1	120	140	3	1
4	10	2	100	180	3	1
5	10	1	120	180	3	1
6	10	1	100	140	6	1
7	15	2	120	180	6	1
8	15	1	100	180	6	1
9	15	2	100	140	3	2
10	10	2	120	140	6	2
11	15	1	120	140	3	2
12	10	2	100	180	3	2
13	10	1	120	180	3	2
14	10	1	100	140	6	2
15	15	2	120	180	6	2
16	15	1	100	180	6	2

最初的 8 次试验位于 “区组 1”中，重复的试验位于 “区组 2”中。

注意：可以从“扩充设计”平台评估完整设计。有关设计评估的详细信息，请参见第 15 章，“评估设计”。

8. 点击**制表**。

图 6.14 重复的设计

	进料速度	催化剂	搅拌速度	温度	浓度	区组	反应百分比
1	15	2	100	140	3	1	61
2	10	2	120	140	6	1	67
3	15	1	120	140	3	1	56
4	10	2	100	180	3	1	94
5	10	1	120	180	3	1	66
6	10	1	100	140	6	1	56
7	15	2	120	180	6	1	82
8	15	1	100	180	6	1	45
9	15	2	100	140	3	2	•
10	10	2	120	140	6	2	•
11	15	1	120	140	3	2	•
12	10	2	100	180	3	2	•
13	10	1	120	180	3	2	•
14	10	1	100	140	6	2	•
15	15	2	120	180	6	2	•
16	15	1	100	180	6	2	•

该表包括 “区组 1”中的原始设计和结果。扩充试验位于 “区组 2”中。反应百分比值可在测量扩充试验后添加到该表中。

向设计添加中心点

使用中心点扩充设计可以检查曲率和减小因子区域中心处的预测误差。中心点通常是允许独立估计纯误差的重复点，可以在失拟检验中使用它们。

本例向 8 次试验设计添加两个中心点。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor 8 Runs.jmp。
2. 选择**实验设计** > **扩充设计**。
3. 选择**反应百分比**并点击 **Y，响应**。
4. 选择**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**，然后点击 **X，因子**。
5. 点击**确定**。
6. 选择**将新试验分到不同的区组**以将分区组变量添加到设计中。
7. 点击**添加中心点**，输入 2，然后点击**确定**。
8. 点击**制表**。

图 6.15 添加了两个中心点的设计

	进料速度	催化剂	搅拌速度	温度	浓度	区组	反应百分比
1	15	2	100	140	3	1	61
2	10	2	120	140	6	1	67
3	15	1	120	140	3	1	56
4	10	2	100	180	3	1	94
5	10	1	120	180	3	1	66
6	10	1	100	140	6	1	56
7	15	2	120	180	6	1	82
8	15	1	100	180	6	1	45
9	12.5	1.5	110	160	4.5	2	•
10	12.5	1.5	110	160	4.5	2	•

为两个中心试验收集数据后，您可以随后检验模型中的曲率。

创建折叠设计

使用折叠扩充设计可以消除双因子交互作用与主效应的混淆。作为饱和或近饱和部分析因或 Plackett-Burman 设计的后续设计，折叠设计特别有用。

本例创建 8 次试验设计的折叠设计。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor 8 Runs.jmp。
2. 选择**实验设计** > **扩充设计**。
3. 选择**反应百分比**并点击 **Y，响应**。
4. 选择**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**，然后点击 **X，因子**。
5. 点击**确定**。

6. 选择**将新试验分到不同的区组**以将分区组变量添加到设计中。
7. 点击**折叠**，然后点击**确定**。

   在“折叠”对话框中，若您未选择任何因子，则所有设计因子都将包括在折叠中。若您选择因子的一个子集进行折叠，则其余因子是原始试验的重复。

图 6.16 所有因子的折叠设计

因子设计

试验	进料速度	催化剂	搅拌速度	温度	浓度	区组
1	15	2	100	140	3	1
2	10	2	120	140	6	1
3	15	1	120	140	3	1
4	10	2	100	180	3	1
5	10	1	120	180	3	1
6	10	1	100	140	6	1
7	15	2	120	180	6	1
8	15	1	100	180	6	1
9	10	1	120	180	6	2
10	15	1	100	180	3	2
11	10	2	100	180	6	2
12	15	1	120	140	6	2
13	15	2	100	140	6	2
14	15	2	120	180	3	2
15	10	1	100	140	3	2
16	10	2	120	140	3	2

8. 点击**制表**。

可通过更改“区组 1”中试验的因子水平（从低到高或从高到低）来获取“区组 2”中的每个试验。

向设计添加轴点

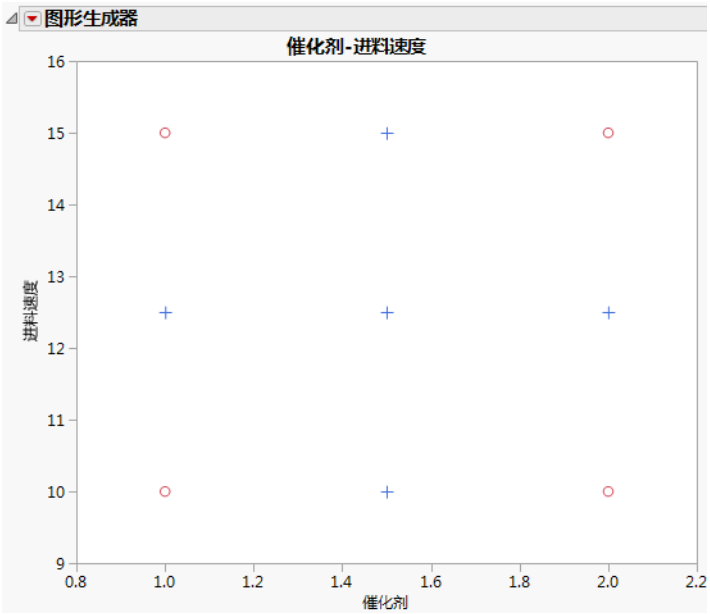
使用轴点和中心点来扩充设计，通过扩充筛选设计来获取响应曲面设计。这样您就可以拟合数据的响应曲面模型，以便优化相对于响应的因子设置。

本例使用轴点和中心点扩充 8 试验筛选设计。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Reactor 8 Runs.jmp。
2. 选择**实验设计 > 扩充设计**。
3. 选择**反应百分比**并点击 **Y，响应**。
4. 选择**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**，然后点击 **X，因子**。
5. 点击**确定**。
6. 选择**将新试验分到不同的区组**以将分区组变量添加到设计中。
7. 点击**添加轴点**，为轴值输入 1，为中心点数输入 2，然后点击**确定**。
8. 点击**制表**。
9. 从 “扩充设计”表中，选择**行 > 按列设定颜色或标记**。
10. 选择**区组**，从 “标记”中选择**标准**，然后点击**确定**。

11. 选择**图形** > **图形生成器**。
12. 将**进料速度**拖至 Y，将**催化剂**拖至 X。然后点击平滑线图标删除平滑线。

图 6.17 两个因子的扩充设计的图形化视图



“催化剂 - 进料速度”图中带 “+” 的点是添加的轴点和中心点。轴点位于设计空间的表面上。扩充设计时增大或减小轴值会将轴点移出或移入设计空间。

空间填充扩充

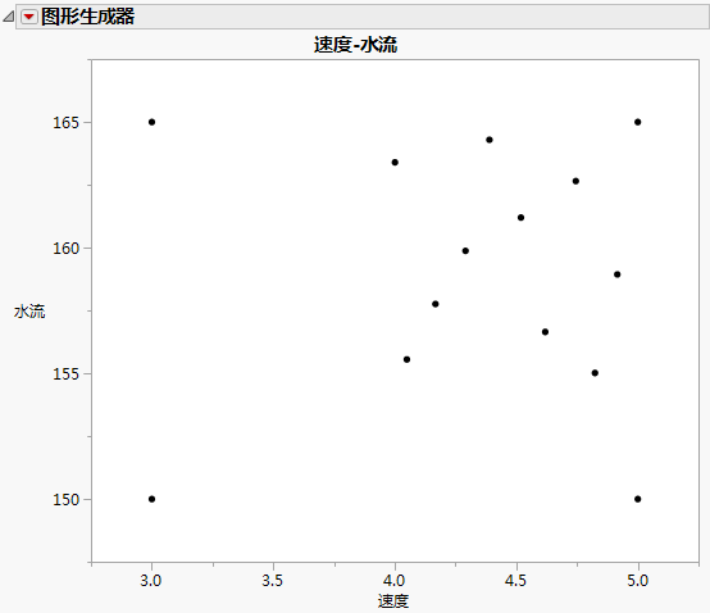
使用空间填充来扩充设计可以向由连续因子构成的设计添加点。 “空间填充”选项支持对设计空间设置约束。您可以指定线性约束或不被允许的组合。有关所用算法的详细信息，请参见第 212 页的“统计详细信息”。

本例扩充了设计以探索设计空间的某一约束区域，空间中的响应**深度**已最大化，同时对两个因子都添加了约束。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/DOE Example 1.jmp。
2. 选择**实验设计** > **扩充设计**。
3. 选择**深度**并点击 **Y，响应**。
4. 选择**速度**和**水流**，点击 **X，因子**，然后点击**确定**。
5. 打开 “定义因子约束”分级显示项。
6. 选择**指定线性约束**，然后点击**添加**。
7. 为**速度**输入 1，选择大于或等于，然后为**速度**下限输入 4。

8. 点击**添加**。
9. 为**水流**输入 1，选择大于或等于，然后为**水流**下限输入 155。
10. 点击**空间填充**。
11. 输入 10 并点击**确定**。
12. 点击**制表**。
13. 从 “扩充设计” 表中，选择 “图形生成器”。
14. 将**速度**拖至 X 拖放区，将**水流**拖至 Y 拖放区。

图 6.18 空间填充扩充试验



扩充试验填充设计空间右上角的空间。这是约束设计空间。

统计详细信息

本节包含关于用于为空间填充扩充选择选择各点的算法的统计详细信息。

用于扩充设计的算法首先在指定的设计区域内生成大量随机点。随后使用快速 Ward 算法将这些点聚类到个数与您指定的 “额外试验数” 相同的聚类中。

通过对现有和额外试验优化 MaxPro （**最大投影**）准则来得到最终设计点。对于 p 个因子以及 n 等于现有和额外试验数的情况， MaxPro 准则尝试查找聚类中可最小化以下准则的点：

$$C_{MaxPro} = \sum_{i}^{n-1} \sum_{j=i+1}^{n} \left[1 / \prod_{k=1}^{p} (x_{ik} - x_{jk})^2 \right]$$

MaxPro 准则可在考虑所有因子的前提下，令设计点之间的距离乘积最大化。这样即可支持为因子投影提供良好的空间填充属性的目标。请参见 Joseph et al. (2015)。

第 7 章

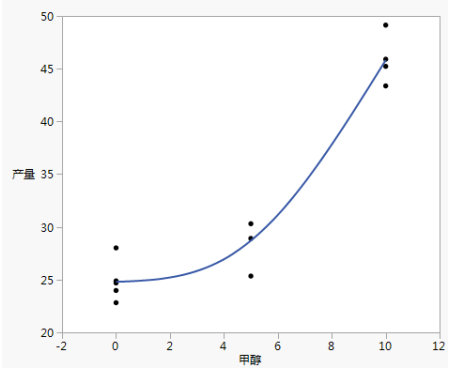
确定性筛选设计

确定性筛选设计 (DSD) 属于筛选设计。这些设计适用于早期实验工作，特别适用于具有四个或更多因子的实验工作。DSD 可用于连续或两水平分类因子组合。该设计在多数因子为连续型时性能最佳。每个连续因子都有三个水平，允许调查连续因子的二次模型项。

以下列出了确定性筛选设计优越于标准筛选设计的方面：

- 通过将各个连续因子设定在三个水平，确定性筛选设计可帮助标识非线性效应的原因。在标准筛选设计中，连续因子仅有两个水平。您可以向筛选设计添加中心点，但仅当存在曲率时才建立这些点。此类设计不允许您标识引起二次效应的因子。
- 它们可以避免最高达到二阶的任何效应之间的混杂。对于连续因子，确定性筛选设计的主效应彼此之间正交，并与双因子交互作用和二次效应正交。双因子交互作用彼此之间不完全混杂。但在具有类似试验次数的许多标准筛选设计中都会出现混杂。
- 它们可以避免为解决标准筛选设计初始结果的不确定性而进行的成本高昂的额外试验。

图 7.1 显示曲率的 “响应 - 因子值” 图



目录

确定性筛选设计概述 217
确定性筛选设计示例 219
确定性筛选设计 219
与部分析因设计进行比较 222
包含分区组的确定性筛选设计 224
确定性筛选设计与 Plackett-Burman 设计的比较 227
“确定性筛选设计”窗口 231
Responses 256
因子 233
设计选项 235
设计 237
设计评估 237
输出选项 237
确定性筛选设计选项 239
模拟响应 241
技术细节 243
确定性筛选设计的结构 243
实验数据分析 245

确定性筛选设计概述

调查人员在想要识别对响应影响最大的因子时会使用筛选设计。筛选设计支持您通过较小规模的实验研究大量因子。

许多标准筛选设计都侧重于估计主效应。确定性筛选设计提供标准筛选设计所不具备的优势。这些设计可避免效应混杂，并且可以标识对响应造成非线性效应的因子。有关确定性筛选设计的优势和构造的详细信息，请参见 Jones and Nachtsheim (2011a)。

针对仅包含连续因子的设计，对确定性筛选设计和标准筛选设计的以下属性进行了比较：

注意：在提及二次效应时，假定标准筛选设计具有中心点。

- 主效应与双因子交互作用正交。
  - 确定性筛选设计：始终
  - 标准筛选设计：仅适用于分辨率 IV 或更高分辨率
- 没有任何双因子交互作用与任何其他双因子交互作用完全混杂。
  - 确定性筛选设计：始终
  - 标准筛选设计：仅适用于分辨率 V 或更高分辨率
- 在仅包含主效应和二次效应的模型中，所有二次效应都可估计。
  - 确定性筛选设计：始终
  - 标准筛选设计：从不

本节下文中对这些属性进行了更为详尽的介绍。

标准筛选设计

标准筛选设计，如部分析因或 Plackett-Burman 设计，尝试使用相对较少的分配资源研究许多因子。不过，标准筛选设计有若干不可取的功能：

- 这些设计中的某些主效应可能与双因子交互作用存在别名关系。例如，在 Plackett-Burman 设计中，主效应与若干双因子交互作用相关。若一个或多个双因子交互作用效应很强，那么必须在实验中执行额外试验来解决不确定性。
- 这些设计中的某些双因子交互作用之间还可能相互混杂。因此，若某个双因子交互作用效应很强，那么必须在实验中执行额外试验来解决遗留的不确定性。
- 连续因子通常设置为仅有两个水平（低和高）。不过，工程师和科学家往往希望设计中的连续因子设置为三个水平（低、中、高）。这是因为两个水平不足以检测物理系统中很常见的非线性。您可以使用添加了中心点的传统筛选设计来检测非线性，但这种设计并不识别引起效应的因子。

确定性筛选设计

使用确定性筛选设计，您可以执行以下操作：

- 避免模型不确定性，支持您更快更高效地识别重要因子。
- 识别非线性效应的原因，同时避免混杂最高达到二阶的任何项。因此，您不仅能够**检测**非线性，就像您在传统筛选设计中使用中心点那样，而且可以**识别**引起效应的因子。

确定性筛选设计提供以下优势：

- 确定性筛选设计仅需要少量试验。对于六个或六个以上的因子，所需的最小试验次数通常只比因子数的两倍稍多。有关试验次数的更多详细信息，请参见第 244 页的“会议矩阵和试验次数”。
- 主效应与双因子交互作用正交。这意味着主效应估计值不会因为存在活跃的双因子交互作用而产生偏倚，不论这些交互作用是否包含在模型中。请注意，在分辨率 III 筛选设计中，某些主效应与交互作用效应混杂。此外，若存在活跃的双因子交互作用，Plackett-Burman 设计会生成有偏的主效应估计值。
- 没有任何双因子交互作用与任何其他双因子交互作用完全混杂。不过，某个双因子交互作用可能与其他双因子交互作用相关。请注意，分辨率 IV 筛选设计完全混杂某些双因子交互作用效应。
- 在仅包含主效应和二次项的模型中，所有二次效应都可估计。这样您就可以识别解释非线性的因子。请注意，添加了中心点的传统筛选设计**不**允许估计包含主效应和二次效应的模型中的所有二次效应。
- 二次效应与主效应正交，且不与双因子交互作用完全混杂。二次效应可能与交互作用效应相关。
- 对于 6 到至少 30 个因子，可以采用高精度估计涉及三个或三个以下因子的所有完全二次模型的参数。
- 对于 18 个或 18 个以上的因子，则可以用任意 4 个因子拟合完全二次模型。对于 24 个或 24 个以上的因子，则可以用任意 5 个因子拟合完全二次模型。

“确定性筛选设计”平台

“确定性筛选设计”平台支持您为连续因子和两水平分类因子构造确定性筛选设计。它还支持您构造分区组设计。您可以添加额外的非中心试验，用来增强设计在有许多效应活跃时能可靠地检测效应的能力。

要查看效应之间相关性的绝对值，请使用“确定性筛选设计”窗口的“设计评估”分级显示项中提供的“相关性色图”。您可以通过比较相关性色图，将确定性筛选设计的别名关系结构与其他设计的别名关系结构加以比较。请参见“评估设计”一章中第 419 页的“相关性色图”。

有关确定性筛选设计的结构的详细信息，请参见第 243 页的“确定性筛选设计的结构”。有关包含区组的确定性筛选设计的信息，请参见第 235 页的“确定性筛选设计中的分区组”。有关如何对使用确定性筛选设计获取的数据进行分析的建议，请参见第 245 页的“实验数据分析”。

“拟合确定性筛选”平台

运行“确定性筛选设计”(DSD) 后，使用“拟合确定性筛选”平台分析您的结果。应用于 DSD 的标准模型选择方法可能无法标识活跃效应。要标识活跃的主效应和二阶效应，“拟合确定性筛选”平台使用名为 DSD 的有效模型选择的算法。该算法利用 DSD 的特殊结构。请参见第 8 章，“拟合确定性筛选”平台”。

若在 JMP 中创建 DSD，设计表则包含名为“拟合确定性筛选”的脚本，该脚本可使用“DSD 的有效模型选择”方法自动运行分析。

确定性筛选设计示例

- “确定性筛选设计”
- “与部分析因设计进行比较”
- “包含分区组的确定性筛选设计”
- “确定性筛选设计与 Plackett-Burman 设计的比较”

确定性筛选设计

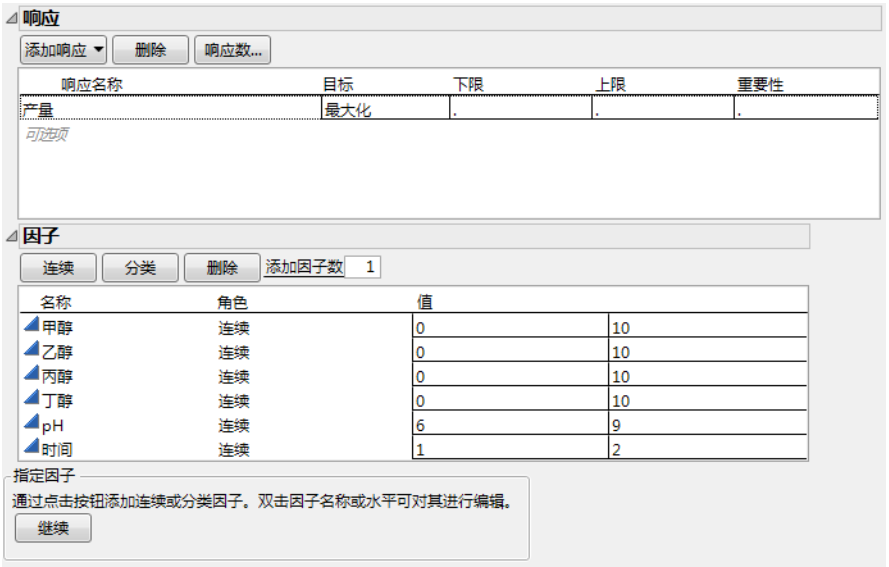
假定您需要确定六个因子中的哪些因子对萃取过程的产量有影响。

创建设计

因子及其设置在数据表 Extraction Factors.jmp 中提供。您通过创建确定性筛选设计来展开调查。

1. 选择**实验设计 > 确定性筛选 > 确定性筛选设计**。
2. 双击“响应名称”下的 Y，然后键入**产量**。
3. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Extraction Factors.jmp。
4. 点击“确定性筛选设计”旁边的红色小三角并选择**加载因子**。
   因子名称和范围将添加到“因子”分级显示项。

图 7.2 萃取设计的响应和因子



5. 点击**继续**。

   “设计选项”分级显示项随即打开。您可以在此指定分区组结构。本例中无需分区组，所以您可以接受默认选项**不需要区组**。

   您还可以选择添加额外试验，这会极大地提高您检测二阶效应的能力。极力推荐您使用最小的额外试验数，这也是默认设置。

6. 点击**制作设计**。

   “确定性筛选设计”窗口更新，以显示 “设计”分级显示项和 “设计评估”分级显示项。

图 7.3 “设计”分级显示项

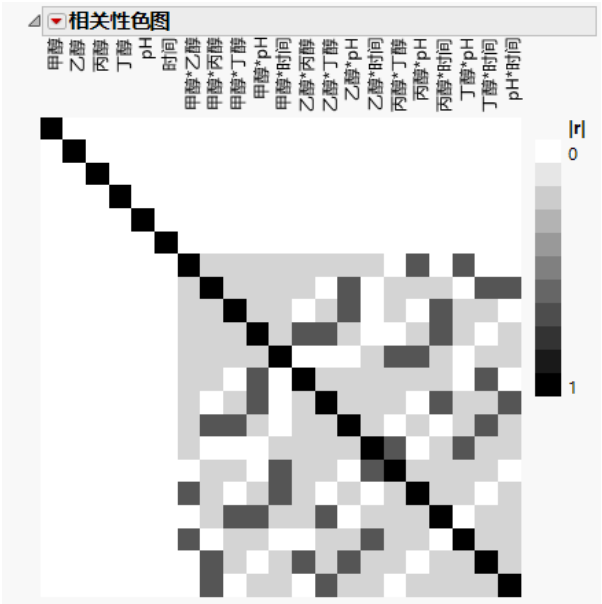
设计

试验	甲醇	乙醇	丙醇	丁醇	pH	时间
1	5	10	10	10	9	2
2	5	0	0	0	6	1
3	10	5	10	10	6	2
4	0	5	0	0	9	1
5	10	0	5	10	9	1
6	0	10	5	0	6	2
7	10	0	0	5	9	2
8	0	10	10	5	6	1
9	10	10	0	0	7.5	2
10	0	0	10	10	7.5	1
11	10	0	10	0	6	1.5
12	0	10	0	10	9	1.5
13	10	10	0	10	6	1
14	0	0	10	0	9	2
15	10	10	10	0	9	1
16	0	0	0	10	6	2
17	5	5	5	5	7.5	1.5

7. 打开**设计评估** > **相关性色图**分级显示项。

“相关性色图”为所有主效应与双因子交互作用之间的相关性绝对值分配了颜色强度尺度。

图 7.4 萃取设计的相关性色图



请注意以下几点：

- 纯白区域显示主效应之间或主效应与双因子交互作用之间无相关性。
- 灰色区域指示双因子交互作用之间的绝对相关性较小。
- 黑色方块指示绝对相关性为 1。全红方块全部显示在对角线上，反映出效应与其本身之间的预料中的相关性。

在 “输出选项”面板中，请注意 “试验顺序”设置为**随机化**。

8. 点击**制表**获取图 7.5 中显示的数据表。

注意：您设计中的试验次数的显示顺序可能不同于图 7.5 中显示的顺序。

图 7.5 萃取过程的确定性筛选设计

	甲醇	乙醇	丙醇	丁醇	pH	时间	产量
1	5	0	0	0	6	1	•
2	0	0	10	0	9	2	•
3	5	10	10	10	9	2	•
4	0	5	0	0	9	1	•
5	10	0	10	0	6	1.5	•
6	10	10	10	0	9	1	•
7	10	5	10	10	6	2	•
8	0	10	0	10	9	1.5	•
9	0	10	5	0	6	2	•
10	5	5	5	5	7.5	1.5	•
11	10	10	0	10	6	1	•
12	10	10	0	0	7.5	2	•
13	10	0	5	10	9	1	•
14	0	10	10	5	6	1	•
15	10	0	0	5	9	2	•
16	0	0	0	10	6	2	•
17	0	0	10	10	7.5	1	•

与部分析因设计进行比较

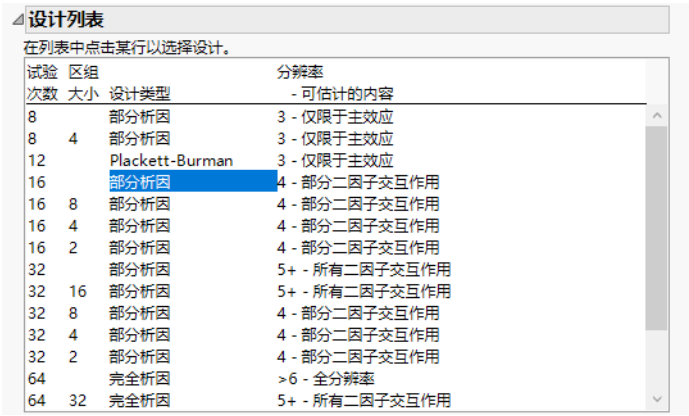
假定您挑选的是传统筛选设计而不是第 219 页的 “确定性筛选设计” 中的确定性筛选设计。本例在混杂方面对两个设计进行比较。

1. 选择**实验设计 > 经典 > 两水平筛选 > 筛选设计**。
2. 双击 “响应名称” 下的 Y，然后键入**产量**。
3. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Extraction Factors.jmp。
4. 点击 “筛选设计” 旁边的红色小三角并选择**加载因子**。

   因子名称和范围将添加到 “因子” 分级显示项。
5. 点击**继续**。
6. 选择**从部分析因设计列表中选择**。
7. 点击**继续**。

   可能的设计显示在 “设计列表” 中。

图 7.6 针对六个连续因子的筛选设计列表



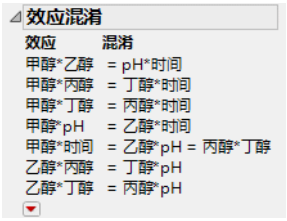
设计列表

在列表中点击某行以选择设计。

试验次数	区组大小	设计类型	分辨率 - 可估计的内容
8		部分析因	3 - 仅限于主效应
8	4	部分析因	3 - 仅限于主效应
12		Plackett-Burman	3 - 仅限于主效应
16		部分析因	4 - 部分二因子交互作用
16	8	部分析因	4 - 部分二因子交互作用
16	4	部分析因	4 - 部分二因子交互作用
16	2	部分析因	4 - 部分二因子交互作用
32		部分析因	5+ - 所有二因子交互作用
32	16	部分析因	5+ - 所有二因子交互作用
32	8	部分析因	4 - 部分二因子交互作用
32	4	部分析因	4 - 部分二因子交互作用
32	2	部分析因	4 - 部分二因子交互作用
64		完全析因	>6 - 全分辨率
64	32	完全析因	5+ - 所有二因子交互作用

8. 选择不含区组的 16 次试验的部分析因设计，该设计在图 7.6 中突出显示。
9. 点击**继续**。
10. 打开**显示和修改设计** > **效应别名**分级显示项。

图 7.7 针对部分析因设计的效应别名



效应混淆

效应	混淆
甲醇*乙醇	= pH*时间
甲醇*丙醇	= 丁醇*时间
甲醇*丁醇	= 丙醇*时间
甲醇*pH	= 乙醇*时间
甲醇*时间	= 乙醇*pH = 丙醇*丁醇
乙醇*丙醇	= 丁醇*pH
乙醇*丁醇	= 丙醇*pH

16 次试验的部分析因设计的“效应别名”分级显示项显示：每个双因子交互作用都与其他至少一个双因子交互作用混杂。在这一部分析因设计中，**乙醇*时间**交互作用与**甲醇*pH** 混杂。要确定哪个交互作用活跃，您需要运行额外试验。若以不同顺序输入因子，**乙醇*时间**交互作用则可能与另外两个双因子交互作用存在别名关系。

在第 219 页的“确定性筛选设计”一节中，您构造了包含 17 次试验的确定性筛选设计。该 DSD 的相关性色图（图 7.4）显示：没有任何双因子交互作用与任何其他双因子交互作用混杂。对于部分析因设计，有七次双因子交互作用混杂。若您怀疑有活跃的双因子效应，则 DSD 是较好的选择。

您可以使用“比较设计”平台（实验设计 > 设计诊断 > 比较设计）对两个设计执行更全面的比较。请参见第 16 章，“比较设计”。

包含分区组的确定性筛选设计

假定因原材料约束，萃取实验要求您使用两个不同批的材料来执行该实验。您可以生成包含分区组变量的确定性筛选设计来解释可能存在的批变异。

创建设计

萃取因子及其设置在数据表 Extraction Factors.jmp 中提供。生成包含区组的确定性筛选设计：

1. 选择**实验设计** > **确定性筛选** > **确定性筛选设计**。
2. 双击 “响应名称” 下的 Y，然后键入**产量**。
3. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Extraction Factors.jmp。
4. 点击 “确定性筛选设计” 旁边的红色小三角并选择**加载因子**。

   因子名称和范围将添加到 “因子” 分级显示项。
5. 点击**继续**。

   “设计选项” 分级显示项随即打开。您可以在此指定分区组结构。
6. 选择**添加包含中心试验的区组以估计二次效应**。

   保持**区组数**设置为 2。

   您要为 Extraction2 Data.jmp 样本数据表重新创建设计，该数据表在创建时未使用额外试验。尽管极力推荐使用额外试验，在本例中仍不添加额外试验。
7. 在**额外试验次数**旁边，选择 0。
8. 点击**制作设计**。

   “确定性筛选设计” 窗口更新，以显示 “设计” 分级显示项和 “设计评估” 分级显示项。

   检查该**区组**是否已添加至 “因子” 分级显示项和设计中。
9. 在 “因子” 分级显示项中，双击**区组**并键入**批**。

   在 “输出选项” 面板中，请注意 “试验顺序” 设置为 “区组内随机”。
10. 点击**制表**。

注意：您设计中的试验次数的显示顺序可能不同于图 7.8 中显示的顺序。

图 7.8 针对萃取过程的包含区组的确定性筛选设计

	区组	甲醇	乙醇	丙醇	丁醇	pH	时间	产量
1	1	5	5	5	5	7.5	1.5	•
2	1	5	10	10	10	9	2	•
3	1	5	0	0	0	6	1	•
4	1	10	0	5	0	9	2	•
5	1	0	0	0	10	7.5	2	•
6	1	0	10	5	10	6	1	•
7	1	10	10	10	0	7.5	1	•
8	2	10	5	0	10	9	1	•
9	2	0	10	0	0	9	1.5	•
10	2	0	0	10	5	9	1	•
11	2	10	0	10	10	6	1.5	•
12	2	0	5	10	0	6	2	•
13	2	10	10	0	5	6	2	•
14	2	5	5	5	5	7.5	1.5	•

请注意，试验 1 是 “第 1 批”中的中心点试验，试验 14 是 “第 2 批”中的中心点试验。

分析实验数据

此时，执行实验并将数据记录在设计表的**产量**列中（图 7.8）。Extraction2 Data.jmp 样本数据表包含您的实验结果。Extraction2 Data.jmp 样本数据表中试验的顺序不同于图 7.8 中的顺序。

要探索所有二阶效应，其中一个选项就是使用 “所有可能的模型”回归。另一个选项是使用前向逐步回归。不过，这些标准方法往往无法标识活跃效应。出于此原因，您需要使用 “拟合确定性筛选”平台。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Extraction2 Data.jmp。
2. 在设计表的 “表”面板中，点击**拟合确定性筛选**脚本旁边的绿色小三角。

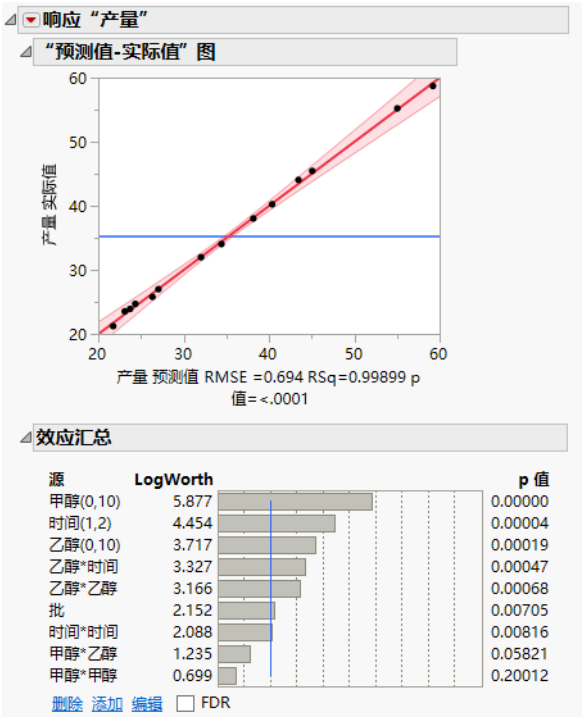
图 7.9 “拟合确定性筛选”报表

“拟合确定性筛选”标识为可能活跃的效应列在 “组合模型参数估计值”报表中。

3. 点击 “组合模型参数估计值”报表底部的**运行模型**按钮。

   这将使用标准最小二乘法模型来拟合标识为可能活跃的效应。

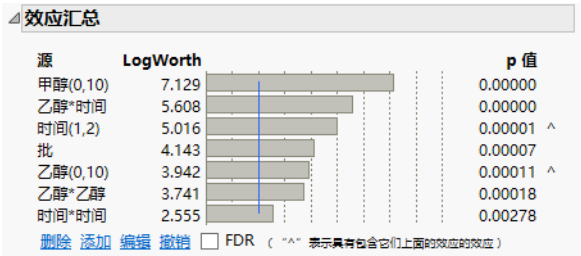
图 7.10 “标准最小二乘法”报表



“‘预测值 - 实际值’图”显示没有证据表明失拟。“效应汇总”报表显示**甲醇 * 乙醇**和**甲醇 * 甲醇**不显著。您决定从模型中删除这些效应。

4. 在“效应汇总”报表中选择“**甲醇 * 乙醇**”和“**甲醇 * 甲醇**”，然后点击**删除**。

图 7.11 最终的活跃效应集



其余效应全部显著。您推断这些是活跃效应。

确定性筛选设计与 Plackett-Burman 设计的比较

Plackett-Burman 设计是部分析因筛选设计的备选设计。不过，Plackett-Burman 设计中的主效应与双因子交互作用之间存在复杂的别名关系。

本例显示如何使用“评估设计”平台比较确定性筛选设计和 Plackett-Burman 设计。有关使用“比较设计”平台的更深入的示例，请参见“比较设计”一章中第 427 页的“试验规模相同的设计”。

确定性筛选设计

1. 选择**实验设计** > **确定性筛选** > **确定性筛选设计**。
2. 在**添加因子数**框中键入 4，然后点击**连续**。
3. 在**添加因子数**框中键入 2，然后点击**分类**。

   窗口应如图 7.12 所示。

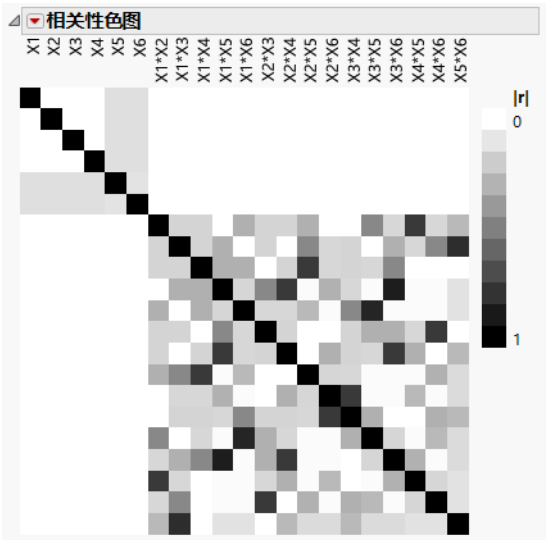
图 7.12 包含 4 个连续因子和 2 个分类因子的确定性筛选对话框

4. 点击**继续**。

   本例不需要区组。在“设计选项”分级显示项下，检查是否已选中“不需要区组”选项。为了比较大小基本相同的设计，请不要添加额外试验。
5. 在**额外试验次数**旁边，选择 0。
6. 点击**制作设计**。

   生成的设计包含 14 次试验。
7. 打开**设计评估** > **相关性色图**分级显示项。

图 7.13 确定性筛选设计的色图



请注意，分类主效应相互之间以及与连续因子主效应之间的相关性都很小。这种相关性导致估计值的精度略微降低。

8. 请不要关闭“确定性设计筛选”窗口，直到您将该色图与下面的 Plackett-Burman 设计的色图进行了比较。

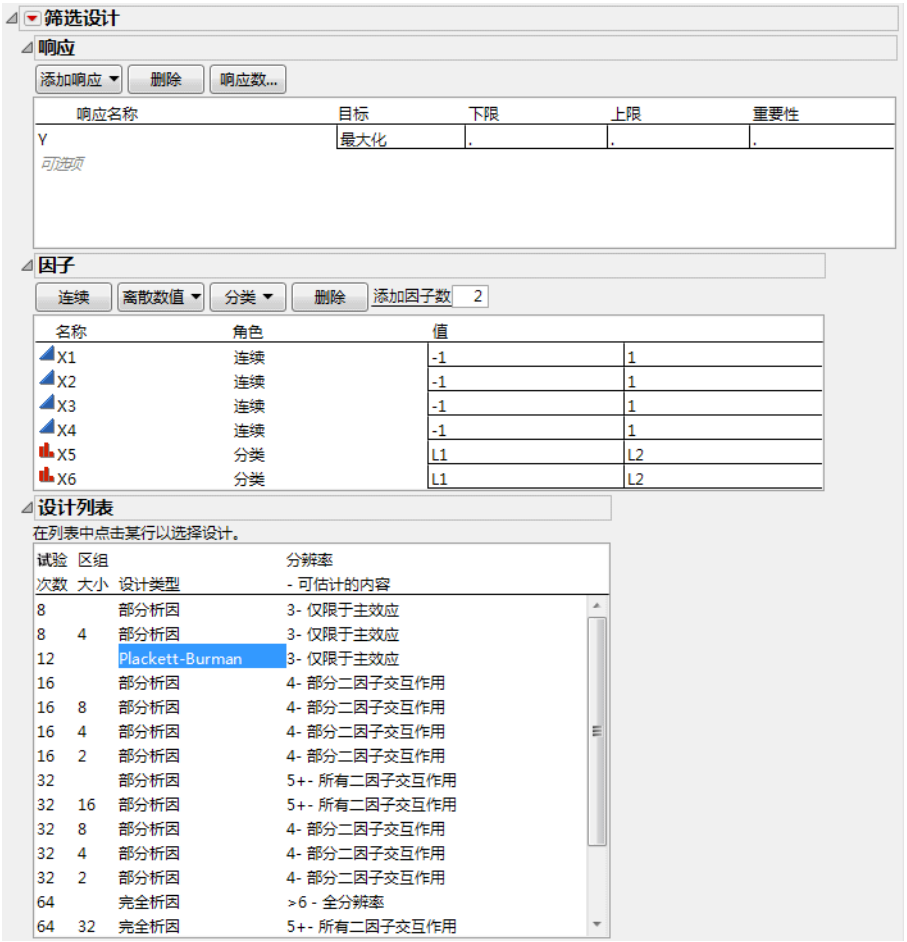
Plackett-Burman 设计

现在使用相同的因子结构创建 Plackett-Burman 设计。

1. 选择**实验设计 > 经典 > 两水平筛选 > 筛选设计**。
2. 在**添加因子数**框中键入 4，然后点击**连续**。
3. 在**添加因子数**框中键入 2，然后点击**分类 > 2 水平**。
4. 点击**继续**。
5. 选择**从部分析因设计列表中选择**，然后点击**继续**。

   可能的设计显示在 “设计列表”中。

图 7.14 Plackett-Burman 设计



6. 选择 12 试验 Plackett-Burman 设计 （图 7.14）。
7. 点击**继续**。
8. 打开**设计评估** > **相关性色图**分级显示项。

   将 12 试验 Plackett-Burman 设计的色图与 14 试验确定性筛选设计的色图加以比较。

图 7.15 Plackett-Burman 相关性 （左）和确定性筛选相关性 （右）

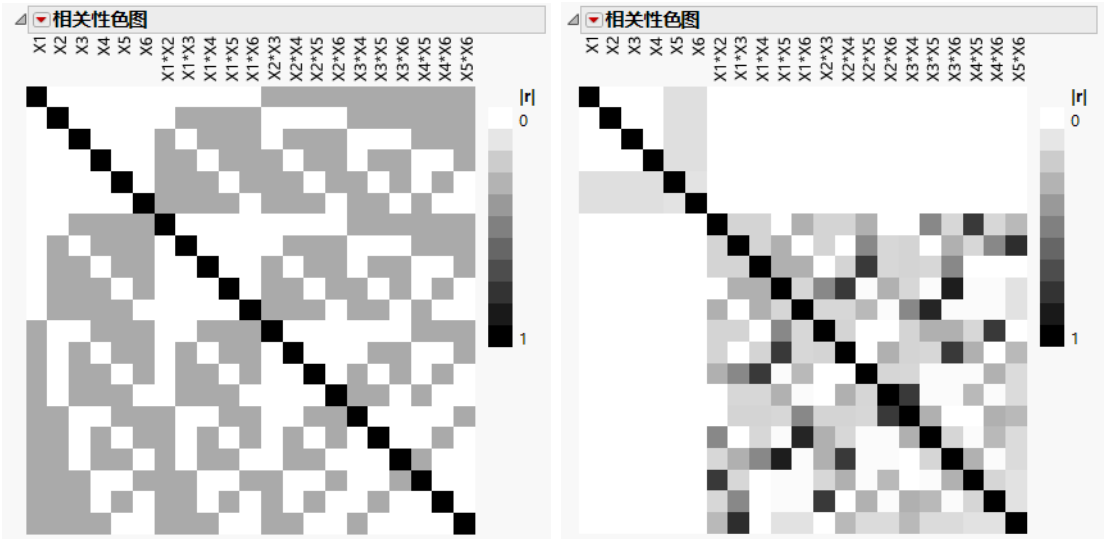


图 7.15 同时显示了两个色图，但仅显示了 Plackett-Burman 色图中涉及主效应和双因子交互作用的部分。（要构造不包含三因子交互作用的 Plackett-Burman 设计的色图，请构造上述设计。然后使用“评估设计”获取色图。）

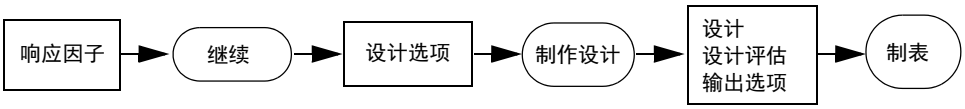
在左侧的 Plackett-Burman 设计的色图中，您会看到多数双因子交互作用都与主效应相关。这意味着任何不可忽略的双因子交互作用都将使若干主效应产生偏倚。这可能导致无法识别活跃的主效应，或得出主效应活跃（实际上不活跃）的错误结论。

将此色图与右侧的确定性筛选设计的色图进行对比。只增加了两次试验，确定性筛选设计以主效应变异的略微增加换取主效应和双因子交互作用的完全独立。

“确定性筛选设计”窗口

当您执行各个设计步骤时，“确定性筛选设计”窗口会相应更新。请参见“从实验设计入手”一章中第 56 页的“实验设计工作流程：描述、指定、设计”。分级显示项遵循图 7.16 中的流程图，它们被一些按钮分开，点击这些按钮后便有不同的分级显示出现。

图 7.16 确定性筛选设计的流程图

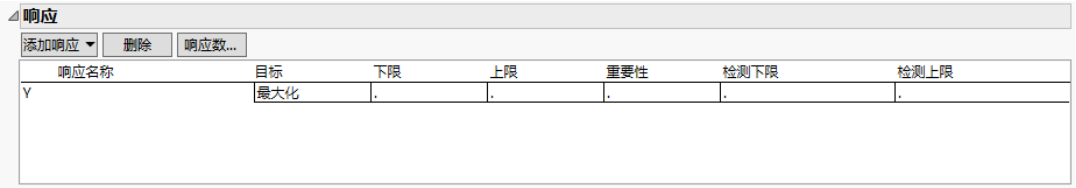


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 7.17 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的“响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
- “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定“广义回归”平台中的删失响应。请参见《拟合线性模型》中的“广义回归”一章。

编辑“响应”分级显示项

在“响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

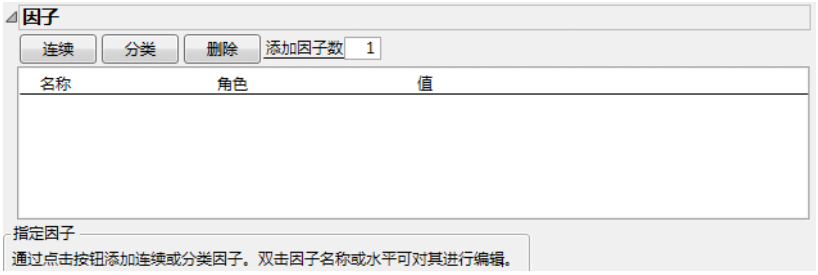
在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

在“因子”分级显示项中添加因子。

图 7.18 “因子”分级显示项



“因子”分级显示项包含以下按钮：

连续型 输入在**添加因子数**中指定的连续因子数。

分类 输入在**添加因子数**中指定的名义型因子数。

删除 删除选定的因子。

添加因子数 添加给定类型的多个因子。输入要添加的因子数并点击 “连续”或 “分类”。重复执行**添加因子数**，添加不同类型的多个因子。

提示：填写完“因子”面板后，从红色小三角菜单中选择保存因子。这将在数据表中保存因子名称和值，以便将来重新加载。请参见第 239 页的“确定性筛选设计选项”。

“因子”分级显示项包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、 X2 等。要更改该名称，双击它，然后输入所需名称。

角色 指定因子的设计角色。因子的 “设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。

值 因子的实验设置。要插入值，点击默认值并键入所需值。

编辑“因子”分级显示项

在 “因子”分级显示项中，请注意以下事项：

- 要编辑某个因子名称，双击该因子名称。
- 要编辑某个值，在**值**列中点击该值。

因子类型

连续 仅限数值数据类型。在符合测量系统的限制条件下，若一个因子在概念上可以设置为您提供的下限与上限之间的任何值，则这样的因子为连续因子。

分类 包含两个水平的数值或字符数据类型。对于分类因子，值排序就是从左到右输入值时值的顺序。在创建设计数据表后，该排序将保存在 “值顺序”列属性中。

因子列属性

对于每个因子，各个列属性保存在数据表中。

设计角色 每个因子都分配有 “设计角色”列属性。在定义因子时指定的角色确定其 “设计角色”列属性的值。在 “设计选项”下添加区组时，该因子将分配有 “分区组”值。“设计角色”属性反映在对实验数据建模时要如何使用因子。“设计角色”值用在“扩充设计”平台中。

因子更改 每个因子都分配有 “因子更改”列属性，该属性设置为 “容易”。在确定性筛选设计中，假定对于每次试验都可以更改因子水平。“因子更改”值用在 “评估设计”和 “扩充设计”平台中。

编码 若 “设计角色”为 “连续”，将保存因子的 “编码”列属性。该属性变换因子值，以便低值与高值分别对应于 –1 和 +1。“拟合最小二乘法”报表中的估计值和检验基于变换值。

值顺序 若“设计角色”为“分类”或“分区组”，将保存因子的“值顺序”列属性。该属性确定因子水平的显示顺序。

每区组试验次数 指示每个区组中的试验次数。在“设计选项”分级显示项中选定某个区组并点击“制作设计”后，将向“因子”列表添加默认名称为“区组”的因子。将为该因子保存“每区组试验次数”列属性。

设计选项

“设计选项”分级显示项支持您指定分区组结构、区组数和额外试验次数。区组效应与主效应正交。区组大小无需相等。

该分级显示项包含以下选项：

不需要区组 指示设计将不包含分区组因子。这是默认选项。

添加包含中心试验的区组以估计二次效应 添加在“区组数”文本框中指定的区组数。构造一种设计，其中的区组效应与主效应正交，并且包含所有主效应和二次效应的模型可估计。请参见第 236 页的“添加包含中心试验的区组以估计二次效应”

添加不需要额外中心试验的区组 添加在“区组数”文本框中指定的区组数。仅添加设计结构所需的中心试验数。构造一种设计，其中的区组效应与主效应正交，但包含所有主效应和二次效应的模型可能**无法**估计。请参见第 236 页的“添加不需要额外中心试验的区组”。

注意：仅当您认定并不是所有二次效应都重要时，才使用**添加不需要额外中心试验的区组**选项。

区组数 指示要添加的区组数。您可以添加的区组数介于 2 到因子数之间。

额外试验次数 添加支持您执行**有效模型选择**的非中心试验。请参见第 245 页的“额外试验”和“‘拟合确定性筛选”平台”一章中第 251 页的“DSD 的有效模型选择”。

提示：使用“额外试验”选项向设计添加试验可以增强在有许多效应活跃的情况下检测效应的能力。推荐的额外试验次数为 4，这样可以显著改善设计在标识活跃二阶效应方面的功效。

制作设计 生成设计、在“设计”分级显示项中提供它，并在“设计评估”分级显示项中提供评估信息。还会显示“输出选项”面板，支持您创建设计表。

确定性筛选设计中的分区组

本节介绍两个分区组选项：

- 第 236 页的“添加包含中心试验的区组以估计二次效应”
- 第 236 页的“添加不需要额外中心试验的区组”

添加包含中心试验的区组以估计二次效应

注意：有关估计二次效应的分区组设计的构造和属性的详细信息，请参见 Jones and Nachtsheim (2016)。该论文还包含有关将区组视为随机效应的信息。

添加包含中心试验的区组以估计二次效应选项用于构造具有以下属性的设计：

- 区组效应与主效应正交。
- 包含所有主效应和二次效应的模型可估计。

若设计仅包含连续因子，则可以构造具备这些属性的 *k* 个因子分区组设计：

- 从针对 *k* 个因子的 DSD 设计中删除中心试验。
- 向同一个区组分配会议矩阵折叠对。
- 向每个区组添加一个中心试验。

若某些因子是分类因子，则**添加包含中心试验的区组以估计二次效应**选项将在特定区组内添加成对中心试验。该结构可确保正交性以及所有主效应和二次效应的可估计性。

由于区组大小唯一要求的就是区组包含一个折叠对，所以区组数的范围可介于 2 到 *k* 之间（若 *k* 为偶数），或是介于 2 到 *k+1* 之间（若 *k* 为奇数）。请参见第 244 页的“会议矩阵和试验次数”。JMP 尝试构造大小相同的区组。

添加不需要额外中心试验的区组

添加不需要额外中心试验的区组选项用于构造这样的设计：在所有因子都是连续因子时，设计中有一个中心试验；在某些因子为分类因子时，设计中有两个中心试验。生成的设计具有以下属性：

- 区组效应与主效应正交。
- 区组效应可能与二次效应的线性组合混杂。这意味着包含所有主效应和二次效应的模型可能**不**可估计。

出于此原因，仅当您可以假定某些二次效应可忽略时才使用该选项。

构造没有额外中心试验的 *k* 个因子的分区组设计：

- 向同一个区组分配会议矩阵折叠对。
- 若所有因子都是连续因子，则将单个中心试验分配到单个区组。
- 若存在分类因子，则不分区组的确定性筛选设计要求向会议矩阵定义的折叠对添加两个中心试验。请参见第 244 页的“会议矩阵和试验次数”。要构造不需要额外中心试验的分区组设计，这两个中心试验会添加到单个区组。

由于区组大小唯一要求的就是区组包含一个折叠对，所以区组数的范围可介于 2 到 *k* 之间（若 *k* 为偶数），或是介于 2 到 *k+1* 之间（若 *k* 为奇数）。请参见第 244 页的“会议矩阵和试验次数”。JMP 尝试构造大小相同的区组。

设计

“设计”分级显示项显示确定性筛选设计的试验。这些试验以标准顺序提供。要更改设计表的试验顺序，您可以在生成设计表之前选择“输出选项”面板中的“试验顺序”选项。

注意：包含四个或四个以下因子的确定性筛选设计基于五因子设计。请参见第 245 页的“四个或四个以下因子的确定性筛选设计”。

设计评估

注意：对于 Cotter 设计，不显示“设计评估”分级显示项。

“设计评估”分级显示项提供若干方式来评估所生成设计的属性。打开 “设计评估”分级显示项可查看以下选项：

功效分析 支持您探索检测给定大小的效应的能力。

预测方差刻画 显示针对因子设置范围的预测方差。

设计空间比例图 显示模型预测方差位于给定值之下或之上的比例。

预测方差曲面 显示任意两个连续因子的预测方差曲面图。

估计效率 对于每个参数，给出与理想 （正交）设计 （该设计可能不存在）的置信区间长度相比，置信区间长度的部分增加。还给出参数的相对标准误差。

别名矩阵 给出系数，这些系数指示因为可能是活跃的却不在模型中的效应导致模型参数的偏倚程度。

相关性色图 使用强度尺度在图上显示两个效应之间的绝对相关性。

设计诊断 指出用于构造设计的最优性准则。还给出设计的效率测度。

注意：当所有二因子交互作用可估计时，用于设计诊断的模型将包含所有主效应和二因子交互作用。否则，该模型包含所有主效应。

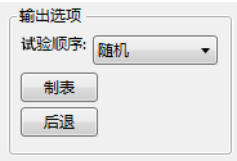
有关“设计评估”面板的更多详细信息，请参见“评估设计”一章中第 408 页的“设计评估”。

输出选项

使用 “输出选项”面板可执行以下任务：

- 指定设计数据表中的试验顺序
- 构造设计表
- 返回“确定性筛选设计”窗口中的上一个点

图 7.19 “输出选项”面板



“输出选项”面板包含以下选项：

试验顺序

试验顺序选项确定试验在设计表中的顺序。包括以下选项：

保持相同 设计表中的行与 “设计”分级显示项中的行具有相同的顺序。

从左至右排序 设计表中的列按从左至右排序。

随机 设计表中的行采用随机顺序。

从右至左排序 设计表中的列按从右至左排序。

区组内随机 设计表中的行在区组内采用随机顺序。

制表

点击**制表**可构造“确定性筛选设计”数据表。

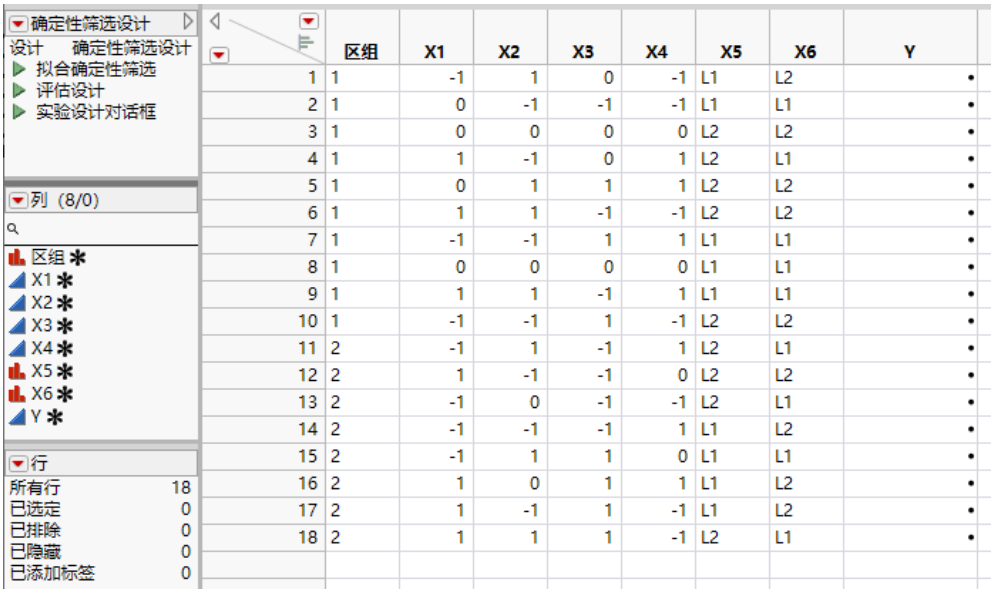
在 “确定性筛选设计”表中，“表”面板 （位于左上角）包含以下脚本。要运行脚本，请点击脚本名称旁边的绿色小三角。

拟合确定性筛选 运行**实验设计 > 确定性筛选 > 拟合确定性筛选**平台。请参见第 8 章，“拟合确定性筛选”平台”。

评估设计 运行**实验设计 > 设计诊断 > 评估设计**平台。请参见第 15 章，“评估设计”。

实验设计对话框 重新创建您用于生成设计表的 “确定性筛选设计”窗口。

图 7.20 显示脚本的确定性筛选设计表



	区组	X1	X2	X3	X4	X5	X6	Y
1	1	-1	1	0	-1	L1	L2	•
2	1	0	-1	-1	-1	L1	L1	•
3	1	0	0	0	0	L2	L2	•
4	1	1	-1	0	1	L2	L1	•
5	1	0	1	1	1	L2	L2	•
6	1	1	1	-1	-1	L2	L2	•
7	1	-1	-1	1	1	L1	L1	•
8	1	0	0	0	0	L1	L1	•
9	1	1	1	-1	1	L1	L1	•
10	1	-1	-1	1	-1	L2	L2	•
11	2	-1	1	-1	1	L2	L1	•
12	2	1	-1	-1	0	L2	L2	•
13	2	-1	0	-1	-1	L2	L1	•
14	2	-1	-1	-1	1	L1	L2	•
15	2	-1	1	1	0	L1	L1	•
16	2	1	0	1	1	L1	L2	•
17	2	1	-1	1	-1	L1	L2	•
18	2	1	1	1	-1	L2	L1	•

后退

“后退”按钮令您回到点击“制作设计”之前您所处的位置。您可以对之前的分级显示项进行更改并重新生成设计。

确定性筛选设计选项

“确定性筛选设计”平台中的红色小三角菜单包含以下选项：

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用“保存响应”选项保存的响应。

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

> **注意：**可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在“设计角色”区域中，选择合适的角色。

加载因子 加载使用“保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在 “定义因子约束”或 “线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为 “小于”或 “大于”约束。请参见 “列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用 “保存约束”选项保存的因子约束。

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的 “试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击 “制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示 “模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一： “正态”、 “二项式”或 “Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开 “模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在 “模型”窗口中做出选择以控制模拟的响应值的分布。当您点击 “应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中， Y 是响应的名称。再次点击 “应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见 “定制设计”一章中第 101 页的 “模拟响应”。

注意: JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见 《基本分析》中的 “模拟”一章。

保存 X 矩阵 将 “矩量矩阵”和 “模型矩阵”保存至设计数据表中的表脚本。这些脚本包含矩量和设计矩阵。请参见 “定制设计”一章中第 103 页的 “保存 X 矩阵”。

警告：对于包含名义型因子的设计，使用“保存 X 矩阵”选项保存的“模型矩阵”**不是**拟合线性模型时使用的编码矩阵。通过在“拟合模型”报表中选择选项“保存列”>“保存编码表”（在您从确定性模型拟合中选择**运行模型**时可获得“拟合模型”报表），您可以获得用于拟合模型的编码矩阵。

将脚本保存至脚本窗口 为您在 “确定性筛选设计”窗口中指定的设计创建脚本，并将其保存到打开的脚本窗口中。

模拟响应

点击 “制表”以创建设计表时，“模拟响应”选项将为每个响应执行以下操作：

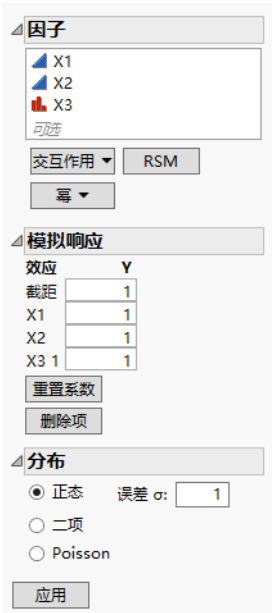
- 将随机响应值添加至设计表中的响应列。
- 将包含模拟模型公式的新列添加至设计表。公式和值基于主效应模型。

“模型”窗口随即打开，您可在其中添加和删除效应以定义模型、指定参数值以及选择用于模拟的响应分布。在“模型”窗口中点击“应用”后，包含模拟模型公式的每一列都随之更新。

控制窗口

图 7.21 为具有两个连续因子（X1 和 X2）和一个两水平分类因子 (X3) 的设计显示“模型”窗口。

图 7.21 模拟响应控制窗口



该窗口具有三个分级显示项：

- 因子
- 模拟响应
- 分布

最初的“模拟响应”分级显示项为所有系数的值均为 1 的主效应模型显示项。“分布”分级显示项显示误差标准差为 1 的正态分布。若已在“实验设计”窗口中“设计评估”下方将“预期系数”设置为“功效分析”的一部分，则“模拟响应”分级显示项中的初始值就是您在“功效分析”分级显示项中指定为“预期系数”和“预期 RMSE（标准误差）”的值。

因子

使用 “因子”分级显示项向模拟模型添加项。

交互作用 选择列表中的因子。然后从 “交互作用”菜单中选择交互作用阶数。这些交互作用将添加至 “模拟响应”分级显示项中的效应列表。

RSM 将所有可能的响应曲面项都添加至 “模拟响应”分级显示项中的效应列表。

幂 选择列表中的因子。然后从 “幂”菜单中选择阶数。这些幂将添加至 “模拟响应”分级显示项中的效应列表。

模拟响应

要为模拟值指定模型，请执行以下操作：

1. 对于效应列表中的每个项，输入用于模拟响应值的线性模型的系数。这将定义一个线性函数 $L(\mathbf{x}, \beta) = \mathbf{x}'\beta$。请参见图 7.21 中的“模拟响应”分级显示项：
   - 向量 **x** 包含定义“效应”下列出的效应的项。
   - 向量 β 是您在 Y 下指定的模型系数的向量。
2. 在 “分布”下，选择一种响应分布。
3. 点击**应用**。包含模拟值的 <Y> **模拟**列及其公式将添加至设计表，其中，Y 是响应列的名称。

重置系数 将所有系数设置为 0。

删除项 从效应列表中删除项。选择要删除的效应并点击 “删除项” 。请注意，您无法删除主效应。

分布

从“模拟响应”窗口中的可用分布中选择其一：

正态 模拟正态分布中的值。为误差 σ（正态误差分布的标准差）输入一个值。若您已在 “因子”分级显示项中指定因子 “更改”值为 “困难”，则可以为整区 σ（整区误差）输入一个值。若您已指定因子的 “更改”值为 “困难”和 “极难”，则可以为同时为子区和整区误差输入值。点击“应用”后，随机值和包含基于模型的随机响应向量的公式将输入列 <Y> **模拟中**。

二项 模拟二项分布中的值。输入 N（即试验次数）的值。根据基于成功概率为 *1/(1 + exp(-L(**x,** β))*) 的 N 次试验的二项分布生成随机整数值。当您点击 “应用” 后，将在列 <Y> **模拟**中输入随机值及其公式。名为 **N 次试验**的包含值 N 的列也会添加至数据表。

Poisson 根据带有参数 *exp(L(**x,** β))* 的 Poisson 分布模拟随机整数值。当您点击 “应用” 后，将在列 <Y> **模拟**中输入随机值及其公式。

注意：您可以设置首选项以便在每次点击 “制表” 时模拟响应。为此，请选择**文件 > 首选项 > 平台 > 实验设计**。选择**模拟响应**。

技术细节

- “确定性筛选设计的结构”
- “实验数据分析”

确定性筛选设计的结构

图 7.22 显示了确定性设计的示例，其中有八个连续因子和四个对应于假因子的额外试验。请注意以下事项：

- 每一对行都是一个折叠对；每个偶数行的值都是上一行乘以 -1。设计的折叠特性去除了双因子交互作用与主效应的混杂。
- 每个因子都有在三次试验中设置为其中心值；这种设置与设计的构造一起令所有二次效应都可估计。
- 行 17 到 20 是对应于假因子的额外试验。
- 在最后一行中添加中心试验，这样您就可以拟合包含截距以及所有主效应和二次效应的模型。

该结构对于包含连续因子的确定性筛选设计很典型。

图 7.22 针对八个连续因子的确定性筛选设计

设计

试验	X1	X2	X3	X4	X5	X6	X7	X8
1	0	1	1	1	1	1	1	1
2	0	-1	-1	-1	-1	-1	-1	-1
3	1	0	-1	-1	-1	-1	1	1
4	-1	0	1	1	1	1	-1	-1
5	1	-1	0	-1	1	1	-1	-1
6	-1	1	0	1	-1	-1	1	1
7	1	-1	-1	0	1	1	1	1
8	-1	1	1	0	-1	-1	-1	-1
9	1	-1	1	1	0	-1	-1	1
10	-1	1	-1	-1	0	1	1	-1
11	1	-1	1	1	-1	0	1	-1
12	-1	1	-1	-1	1	0	-1	1
13	1	1	-1	1	-1	1	0	-1
14	-1	-1	1	-1	1	-1	0	1
15	1	1	-1	1	1	-1	-1	0
16	-1	-1	1	-1	-1	1	1	0
17	1	1	1	-1	-1	1	-1	1
18	-1	-1	-1	1	1	-1	1	-1
19	1	1	1	-1	1	-1	1	-1
20	-1	-1	-1	1	-1	1	-1	1
21	0	0	0	0	0	0	0	0

会议矩阵和试验次数

JMP 中的确定性筛选设计是使用**会议矩阵** (Xiao et al., 2012) 构造的。会议矩阵是一个 m x m 矩阵 **C**，其中，m 是偶数。矩阵 **C** 在对角线上具有 *0* 值，非对角线元素等于 *1* 或 *–1*，并且满足

$$\mathbf{C}'\mathbf{C} = (m-1)\mathbf{I}_{mxm}。$$

注意：对于 m 的特定偶数值，是否存在会议矩阵尚未知晓。

假定因子数 k 为 5 或更大的数字。对于 $k \leq 4$ 个因子的情况，请参见第 245 页的“四个或四个以下因子的确定性筛选设计”。

考虑 k 个连续因子的情况，并假定存在会议矩阵。

- 若 k 为偶数，k x k 会议矩阵用来定义设计的 k 次试验。其负值 **–C** 则定义折叠试验。向设计添加了一个中心点以确保包含截距、主效应和二次效应的模型可估计。所以，对于 k 为偶数的情况，确定性筛选设计中的最小试验次数为 $2k + 1$。
- 若 k 为奇数，将使用 $(k+1)$ x $(k+1)$ 会议矩阵，并删除其末列。添加了一个中心点。所以，对于 k 为奇数的情况，确定性筛选设计中的最小试验次数为 $2k + 3$。

在某些因子为分类因子并且存在会议矩阵的情况下，会使用相似的过程。请参见 Jones and Nachtsheim (2013)。

- 需要添加两个额外试验，而不是一个中心点。这两次试验是中心试验，其中所有连续因子都设置为其中间值。
- 若有 k 个因子且 k 为偶数，设计中的试验次数为 $2k + 2$。

- 若 k 为奇数，试验次数为 $2k + 4$。

对于会议矩阵不可用的那些 m 值，可使用下一个最大的会议矩阵构造确定性筛选设计。因此，所需的试验次数可能超过 $2k + 3$（对于连续因子的情况），或超过 $2k + 4$（对于分类因子的情况）。

额外试验

使用虚构或**假**因子构造额外试验。向设计中添加 f 个假因子可生成 2^f 个额外试验。

用 k 表示实验研究中的因子数。可以在设计中添加四到八个额外试验。通过为 $k + f$ 个因子创建设计来构造额外试验（如第 244 页的“会议矩阵和试验次数”中所述），然后删除后 f 列。仅仅四个额外试验就会对模型选择大有益处。

有关如何使用额外试验的信息，请参见“‘拟合确定性筛选”平台”一章中第 251 页的“DSD 的有效模型选择”。

四个或四个以下因子的确定性筛选设计

四个或四个以下因子的确定性筛选设计是使用五因子确定性筛选设计为基础构造的。这是因为严格根据会议矩阵法构造 $k \leq 4$ 个因子的设计具有不理想的属性。特别是很难分隔二阶效应。

若指定 $k \leq 4$ 个因子，则会构造针对五个因子的确定性筛选设计并删除不必要的列。因此，包含 $k \leq 4$ 个因子的不分区组设计的试验次数为 13（若所有因子都是连续因子）或 14（若部分因子为分类因子）。

实验数据分析

通常，您想要拟合一个模型，该模型允许双因子交互作用可能是活跃的。您也可能想要在模型中包含纯二次项。您可能需要假设一个完全二阶模型，或是您想要指定一个仅包含一些二阶项的先验模型。

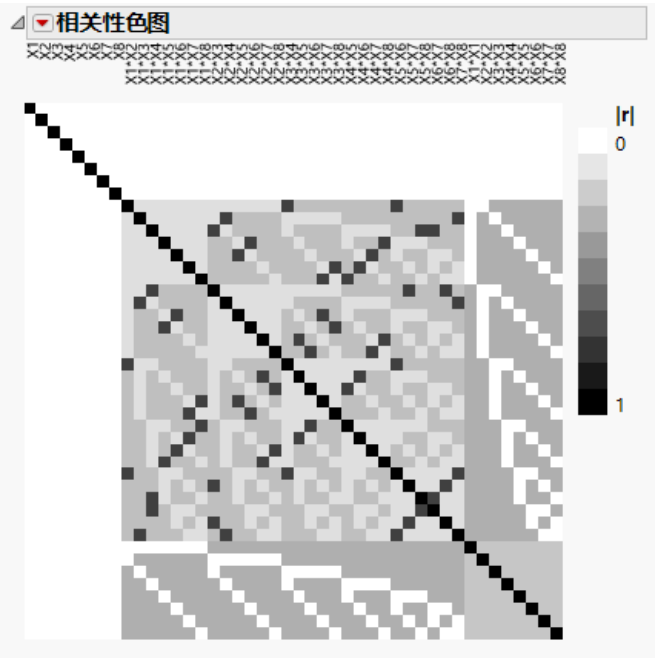
双因子交互作用

在拟合此类模型时，您需要注意两个事实：

- 双因子交互作用效应与二次效应往往相关
- 无法同时估计**所有**双因子交互作用和二次效应

图 7.23 为包含八个连续因子的设计显示了相关性色图，如图 7.22 所示。该色图针对完全二次设计。八个纯二次效应列在最右侧。通过使用“实验设计”>“设计诊断”>“评估设计”并在“别名项”列表中输入适当的项，您可以构造该图。请参见“评估设计”一章中第 407 页的“别名项”。

图 7.23 完全二次模型的相关性色图



悬停在色图单元格上方，以便查看效应之间的绝对相关性。您会看到主效应与所有双因子交互作用和纯二次效应都不相关。还会看到没有任何效应与其他效应完全混杂，因为唯一的黑色单元格位于主对角线上。但请注意，双因子交互作用之间的某些绝对相关性很大，有些甚至高达 0.75。还要注意，双因子交互作用与纯二次效应之间的绝对相关性为 0 或 0.3118。

若只有主效应和纯二次效应活跃，您可以拟合包含主效应和二次效应的饱和模型。该模型将生成无偏的效应估计值，假定没有任何活跃的三因子或更高阶的效应。

由于相关性涉及二阶效应，您必须小心地拟合包含双因子交互作用的模型。包括以下分析方法，首选第一种：

- “有效模型选择”方法的效果较好，特别是在可能有许多效应活跃时。请参见““拟合确定性筛选”平台”一章中第 251 页的“DSD 的有效模型选择”。
- 若以下条件成立，前向逐步或所有可能的子集回归都会充分执行。
  - 活跃效应数不超过试验次数的一半。
  - 至多有两个活跃的双因子交互作用或至多有一个活跃的二次效应。

  请参见第 247 页的“前向逐步回归或所有可能的子集回归”。

前向逐步回归或所有可能的子集回归

该方法包括首先指定一个完全响应曲面模型。然后执行以下操作之一：

- 使用前向逐步回归，将“停止规则”设置为“最小 AICc”并将“规则”设置为“合并”，以确保模型的遗传性。
- 使用“所有可能的模型”回归，您需要选择强制“遗传限制”的选项，并使用 AICc 准则进行模型选择。

您无法拟合完整响应曲面模型，因为试验次数小于参数个数。所以您的分析取决于效应稀疏假设，此时您假设活跃效应数小于试验次数。该方法有以下一些限制：

- 若活跃效应数超过试验次数的一半，则逐步回归和所有可能的模型回归都很难找到正确的模型。
- 检测适合的二次效应的检验功效很低。二次效应必须超过三倍的误差标准差，功效才能超过 0.9。
- 由于效应混杂，若干模型可能等效。将需要额外试验来解决混杂。

第 8 章

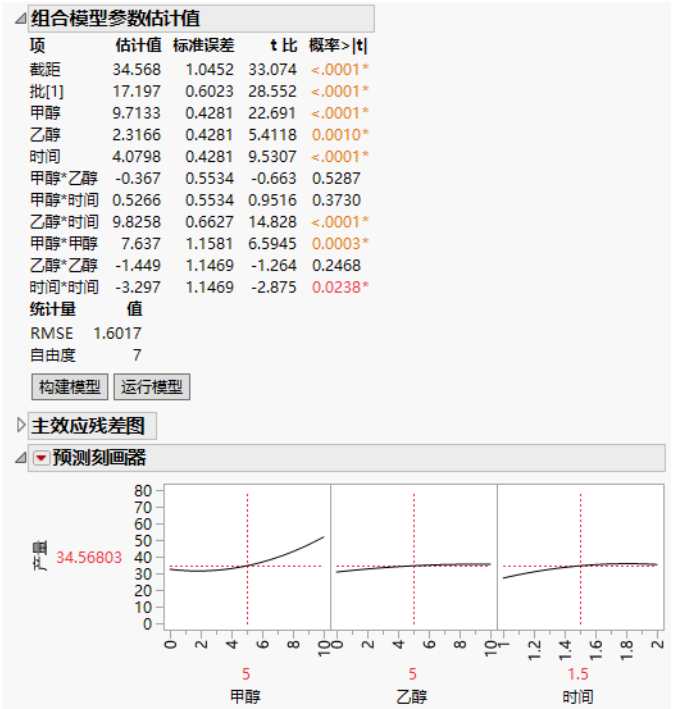
“拟合确定性筛选”平台

分析确定性筛选实验中的数据

使用“拟合确定性筛选”平台可通过称为 **DSD 的有效模型选择**的方法分析确定性筛选设计 (DSD)。该方法利用确定性筛选设计的独特结构。

应用于 DSD 的标准模型选择方法可能无法标识活跃效应。为标识活跃的主效应和二阶效应，“DSD 的有效模型选择”算法会利用 DSD 的结构。

图 8.1 拟合确定性筛选结果



目录

“拟合确定性筛选”平台概述 251
标识 DSD 中的活跃效应 251
DSD 的有效模型选择 251
“拟合确定性筛选”平台示例 252
拟合模型 252
检查结果 252
简化模型 254
启动“拟合确定性筛选”平台 255
“拟合确定性筛选”报表 256
阶段 1 - 主效应估计值 256
阶段 2 - 偶数阶效应估计值 257
组合模型参数估计值 258
主效应残差图 258
预测刻画器 258
“拟合确定性筛选”平台选项 258
“拟合确定性筛选”平台的技术细节 259
“DSD 的有效模型选择”算法 259

“拟合确定性筛选”平台概述

“拟合确定性筛选”平台使用利用确定性筛选设计 (DSD) 的特殊结构的方法分析确定性筛选设计。该方法称为 **DSD 的有效模型选择**。若在 JMP 中创建了 DSD，设计表则包含名为“拟合确定性筛选”的脚本，该脚本可使用“DSD 的有效模型选择”方法自动运行分析。

标识 DSD 中的活跃效应

DSD 是有利于在单个实验中标识主效应和二阶效应的三水平设计。若活跃效应数小于试验次数约一半，并且效应大小超过标准差的两倍，则最小试验规模的 DSD 能够高概率地正确标识活跃项。

不过，若用四个或更多选定试验扩充最小试验规模的 DSD，则可以标识更多的具有较高概率的效应。这些试验称为**额外试验**并且对应于虚构的称为**假因子**的非活跃因子。有关额外试验的信息，请参见“确定性筛选设计”一章中第 243 页的“确定性筛选设计的结构”。

额外试验可大幅度增加设计检测二阶效应的能力。出于此原因，Jones and Nachtsheim (2016) 强烈推荐包含至少四个额外试验。

DSD 的有效模型选择

将标准模型选择方法应用于 DSD 时，这些方法可能无法标识活跃效应。请参见 Errore et al. (2017)。此外，标准选择方法不利用 DSD 的结构。“拟合确定性筛选”平台使用“DSD 的有效模型选择”方法，该方法充分利用 DSD 的结构。

Jones and Nachtsheim (2016) 为使用“DSD 的有效模型选择”方法和标准方法的模拟研究提供了报告。用 *c* 来表示 DSD 中的因子数与假因子数的总和。在许多情况下，若活跃主效应数超过三个，则可以可靠标识 *c*/2 个活跃二阶效应。假定存在强效应遗传，若有三个或三个以下的活跃主效应，则可以可靠地标识所有活跃的二阶效应。**可靠标识**意味着系数绝对值与误差标准差的比值超过了 3 并且检测效应的功效超过了 0.80。

“拟合确定性筛选”平台的默认设置假定存在**强效应遗传**。强效应遗传意味着仅当同时包含 A 和 B 时才考虑在模型中包含 A*B 交互作用。强效应遗传要求将模型效应的所有低阶成分都包含在模型中。在标识活跃的二阶效应时，该算法使用强效应遗传和之前引述的关于可以可靠标识多少活跃二阶效应的结果。

在 DSD 中，主效应和二阶效应彼此正交。“DSD 的有效模型选择”方法利用了这一事实。响应的线性空间划分到主效应生成的子空间和该子空间的正交补空间中。Miller and Sitter (2005) 将主效应横跨的线性子空间称为**奇空间**，因为它包含关于奇效应（主效应、3 因子效应、5 因子效应等）的所有信息。他们将其正交补称为**偶空间**，因为它包含关于偶效应的所有信息：截距、2 因子效应、4 因子效应等。

拟合确定性筛选采用的就是上述思路。主效应横跨的子空间就是奇空间。其正交补（偶空间）包含二阶效应和区组变量（若存在）。有关算法的详细信息，请参见第 259 页的“ “DSD 的有效模型选择”算法”和 Jones and Nachtsheim (2016)。

“拟合确定性筛选”平台示例

样本数据表 Extraction 3 Data.jmp 中的设计是一个位于两个区组中的六因子的确定性筛选设计。**添加包含中心试验的区组以估计二次效应**选项处于选定状态，并且向设计生成过程添加了 4 个额外试验。生成的设计包含 18 次试验。该设计用于探索确定性筛选设计的分析。

- “拟合模型”
- “检查结果”
- “简化模型”

注意：有关使用连续型和分类因子的设计的示例，请打开样本数据文件 Peanut Data.jmp 并运行“拟合确定性筛选”表脚本。

拟合模型

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Extraction 3 Data.jmp。
2. 选择**实验设计 > 确定性筛选 > 拟合确定性筛选**。
3. 选择**产量**并点击 Y。
4. 从**批**一直选择到**时间**，然后点击 X。
5. 点击**确定**。

该拟合执行两阶段分析。有关算法的详细信息，请参见第 259 页的““拟合确定性筛选”平台的技术细节”。

检查结果

阶段 1：主效应估计值

阶段 1 确定哪些主效应可能活跃。

图 8.2 阶段 1 的主效应报表

阶段 1 - 主效应估计值

项	估计值	标准误差	t 比	概率>|t|
甲醇	9.7133	0.3674	26.438	<.0001*
乙醇	2.3166	0.3674	6.3055	0.0015*
时间	4.0798	0.3674	11.104	0.0001*

统计量	值
RMSE	1.3747
自由度	5

☑ 二次项服从强遗传
☑ 交互作用服从强遗传

注意：假因子不显示在设计中也不作为分析中的因子。

从对应两个假因子的四次试验计算了两个自由度的平方和误差。由于根据假因子的构造，假因子非活跃因子，所以该误差方差的估计值无偏。对于每个主效应，针对该估计值来检验主效应响应 Y_{ME}。在本例中，**甲醇**、**乙醇**和**时间**这三个因子的 p 值小于阈值，作为活跃因子留下。有关阈值的详细信息，请参见第 259 页的“阶段 1 方法”。

丙醇、**丁醇**和 pH 这三个非活动因子的变异性与假因子平方和合并，共同生成五个自由度的 RMSE 统计量（如图 8.2 中所示）。

阶段 2：偶数阶效应估计值

阶段 2 使用经指导的子集选择得到可能活跃的二阶效应列表。交互作用和二次项是二阶或偶数阶效应。

图 8.3 阶段 2 的偶数阶效应报表

阶段 2 - 偶数阶效应估计值

项	估计值	标准误差	t 比	概率>|t|
截距	34.568	1.3459	25.683	0.0015*
批[1]	17.197	0.7757	22.171	0.0020*
甲醇*乙醇	-0.367	0.7127	-0.515	0.6581
甲醇*时间	0.5266	0.7127	0.7389	0.5369
乙醇*时间	9.8258	0.8534	11.514	0.0075*
甲醇*甲醇	7.637	1.4914	5.1208	0.0361*
乙醇*乙醇	-1.449	1.477	-0.981	0.4299
时间*时间	-3.297	1.477	-2.232	0.1552

统计量	值
RMSE	2.0626
自由度	2

由于在阶段 1 中标识了三个主效应为活跃效应，针对活跃二阶效应的经指导的子集选择过程将继续进行，直到包含所有二阶效应。由于阶段 2 报告了全部六个二阶效应，由此判定：阶段 2 RMSE 仍大于阶段 1 RMSE。请参见第 260 页的“阶段 2 方法”。

阶段 2 报表中提供的两个自由度的 RMSE 是从全部六个二阶效应的最终子集获得的误差估计值。

合并后的结果

为模型选定的效应列在“组合模型参数估计值”报表中。

图 8.4 “组合模型参数估计值”报表

组合模型参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	34.568	1.0452	33.074	<.0001*
批[1]	17.197	0.6023	28.552	<.0001*
甲醇	9.7133	0.4281	22.691	<.0001*
乙醇	2.3166	0.4281	5.4118	0.0010*
时间	4.0798	0.4281	9.5307	<.0001*
甲醇*乙醇	-0.367	0.5534	-0.663	0.5287
甲醇*时间	0.5266	0.5534	0.9516	0.3730
乙醇*时间	9.8258	0.6627	14.828	<.0001*
甲醇*甲醇	7.637	1.1581	6.5945	0.0003*
乙醇*乙醇	-1.449	1.1469	-1.264	0.2468
时间*时间	-3.297	1.1469	-2.875	0.0238*

统计量	值
RMSE	1.6017
自由度	7

报表底部提供的 RMSE 和自由度是通常的标准最小二乘法数量。将这些效应用作您最终模型的潜在因子。

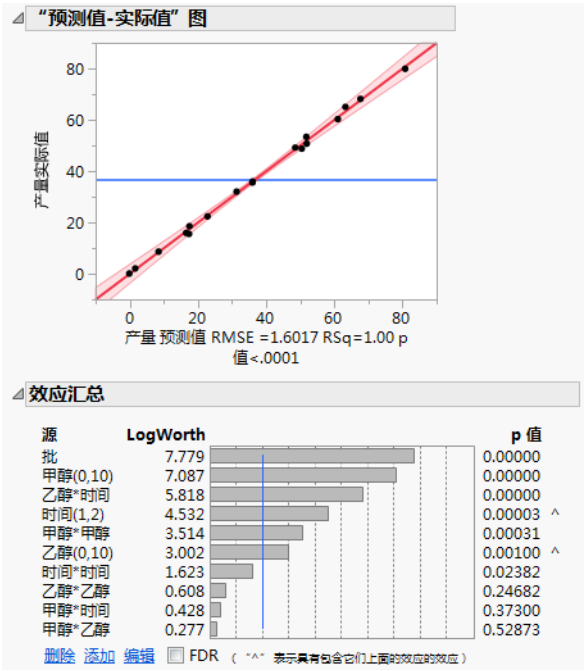
简化模型

“构建模型”按钮为列出的项在 “拟合模型”规格窗口中输入模型。要直接使用标准最小二乘法运行模型，请点击 “运行模型”按钮。

1. 点击**运行模型**。

   “‘预测值 - 实际值’图”显示无失拟。“效应汇总”报表建议您可以进一步缩减模型。

图 8.5 “预测值 - 实际值”图和 “效应汇总”报表



2. 在 “效应汇总”报表中选择**甲醇 * 乙醇**，然后点击**删除**。

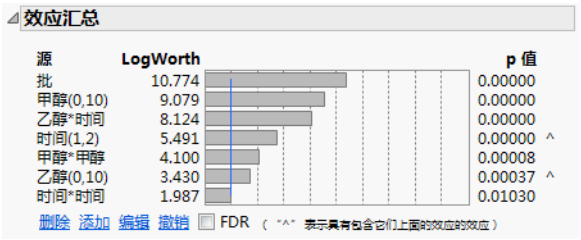
   甲醇 * 时间的 *p* 值为 0.33750。接下来删除它。

3. 在 “效应汇总”报表中选择**甲醇 * 时间**，然后点击**删除**。

   乙醇 * 乙醇的 *p* 值为 0.15885。接下来删除它。

4. 在 “效应汇总”报表中选择**乙醇 * 乙醇**，然后点击**删除**。

图 8.6 显示最终模型中的效应的“效应汇总”报表



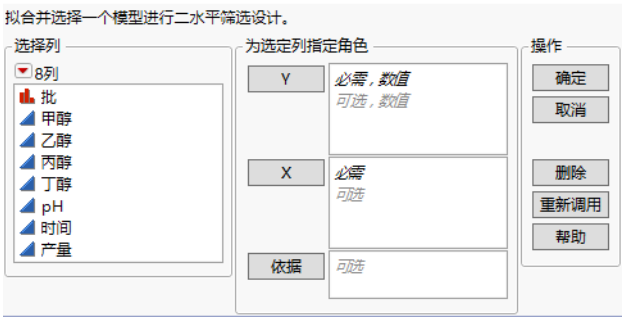
其余效应全部显著。您推断这些是活跃效应。

启动“拟合确定性筛选”平台

要启动“拟合确定性筛选”平台，请选择**实验设计 > 确定性筛选 > 拟合确定性筛选**。图 8.7 中的启动窗口使用 Extraction3 Data.jmp。

注意：若您在 JMP 中创建了设计，则设计表包含称为“拟合确定性筛选”的脚本。运行该脚本可直接运行分析。

图 8.7 “拟合确定性筛选”启动窗口



Y 一个或多个数值响应变量。

X 连续或两水平分类因子。由于该平台使用 DSD 的独特功能执行分析，这些因子必须定义 DSD 或折叠设计。

依据 一列，其水平定义不同的分析。对于指定列的每个水平，都分析相应行。结果显示在单独的报表中。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独分析。

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“拟合确定性筛选”报表

- “阶段 1 - 主效应估计值”
- “阶段 2 - 偶数阶效应估计值”
- “组合模型参数估计值”
- “主效应残差图”
- “预测刻画器”

阶段 1 - 主效应估计值

“主效应估计值”报表列出标识为活跃的主效应。*p* 值小于阈值 *p* 值的主效应被认为是活跃效应。

- 若具有可用的假因子或中心点重复，则构造独立于模型的误差方差估计量。针对该估计值检验主效应。
- 若假因子或中心点重复不可用，则针对从非活跃主效应构造的误差方差的估计值按顺序检验主效应子集。为使该过程可行，必须至少有一个非活跃的主效应。

在任一种情况下，非活跃主效应的变异性都合并到用于检验主效应的误差方差中。

图 8.8 阶段 1 报表

阶段 1 - 主效应估计值

项	估计值	标准误差	t 比	概率>|t|
甲醇	9.7133	0.3674	26.438	<.0001*
乙醇	2.3166	0.3674	6.3055	0.0015*
时间	4.0798	0.3674	11.104	0.0001*

统计量	值
RMSE	1.3747
自由度	5

☑ 二次项服从强遗传
☑ 交互作用服从强遗传

项 标识为活跃的主效应。这些效应的 *p* 值在检验时小于阈值 （如第 259 页的 “阶段 1 方法”中所述）。

估计值 针对主效应的 Y 回归拟合的参数估计值。

标准误差 使用阶段 1 RMSE 计算得出的估计值的标准误差。

t 比 估计值除以其标准误差。

概率 >|t| 使用 t 比和误差自由度 (DF) 计算所得的 *p* 值。

RMSE 从阶段 1 分析中得出的均方误差的平方根。

– 若假因子或中心点重复可用，均方误差即：假因子和中心点的方差估计值与由未标识为活跃的主效应构造的方差估计值合并之后的结果。

– 若没有任何假因子或中心点重复可用，则均方误差就是从未标识为活跃的主效应构造的方差的估计值。

自由度 与用于构造 RMSE 的误差估计值关联的自由度。

– 若假因子或中心点重复可用，则自由度是假因子、中心点重复和未标识为活跃的主效应的数目的总和。

– 若没有任何假因子或中心点重复可用，则自由度是未标识为活跃的主效应的数目。

二次项服从强遗传 选中或取消选中以控制应用到阶段 2 中的二次项的遗传规则。

交互作用服从强遗传 选中或取消选中以控制应用到阶段 2 中的交互作用的遗传规则。

注意：强效应遗传意味着仅当同时包含 A 和 B 时才考虑在模型中包含 A*B 交互作用。强效应遗传要求将模型效应的所有低阶成分都包含在模型中。

阶段 2 - 偶数阶效应估计值

“偶数阶效应估计值”报表列出标识为活跃的二阶效应。活跃二阶效应使用第 260 页的“阶段 2 方法”中所述的经指导的变量选择过程或向前选择来标识。区组效应（若包含一个区组）也同时列出，不论其是否显著。

图 8.9 阶段 2 报表

阶段 2 - 偶数阶效应估计值

项	估计值	标准误差	t 比	概率>|t|
截距	34.568	1.3459	25.683	0.0015*
批[1]	17.197	0.7757	22.171	0.0020*
甲醇*乙醇	-0.367	0.7127	-0.515	0.6581
甲醇*时间	0.5266	0.7127	0.7389	0.5369
乙醇*时间	9.8258	0.8534	11.514	0.0075*
甲醇*甲醇	7.637	1.4914	5.1208	0.0361*
乙醇*乙醇	-1.449	1.477	-0.981	0.4299
时间*时间	-3.297	1.477	-2.232	0.1552

统计量	值
RMSE	2.0626
自由度	2

项 标识为活跃的区组因子和二阶效应。

估计值 由 Y_{2nd} 定义的阶段 2 二阶效应的 Y 回归拟合的参数估计值。请参见第 259 页的“响应分解”。

标准误差 使用阶段 2 RMSE 计算得出的估计值的标准误差。

t 比 估计值除以其标准误差。

概率 >|t| 使用 t 比和误差自由度 (DF) 计算所得的 *p* 值。

RMSE 从阶段 2 分析中得出的均方误差的平方根。RMSE 估计为拟合二阶效应（标识为活跃）之后的 Y_{2nd} 的残差方差。请参见第 259 页的 “响应分解”。

自由度 与用于构造 RMSE 的误差估计值关联的自由度。

组合模型参数估计值

“组合模型参数估计值”报表列出最终模型中的项及其通常的标准最小二乘法估计值、标准误差、t 比、*p* 值、RMSE 和模型自由度。

该报表下是用于构造或运行组合模型的按钮。

图 8.10 “组合模型参数估计值”报表

组合模型参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	34.568	1.0452	33.074	<.0001*
批[1]	17.197	0.6023	28.552	<.0001*
甲醇	9.7133	0.4281	22.691	<.0001*
乙醇	2.3166	0.4281	5.4118	0.0010*
时间	4.0798	0.4281	9.5307	<.0001*
甲醇*乙醇	-0.367	0.5534	-0.663	0.5287
甲醇*时间	0.5266	0.5534	0.9516	0.3730
乙醇*时间	9.8258	0.6627	14.828	<.0001*
甲醇*甲醇	7.637	1.1581	6.5945	0.0003*
乙醇*乙醇	-1.449	1.1469	-1.264	0.2468
时间*时间	-3.297	1.1469	-2.875	0.0238*

统计量	值
RMSE	1.6017
自由度	7

构建模型 运行模型

构建模型 在“拟合模型”窗口中创建一个模型，该模型包含了“组合模型参数估计值”报表中的模型项以及在“拟合确定性筛选”分析指定的响应。指定了“标准最小二乘法”特质。

运行模型 针对“组合模型参数估计值”报表中的模型项以及为“拟合确定性筛选”分析指定的响应运行标准最小二乘法拟合。

主效应残差图

显示相对于绘制因子各水平的残差图，该残差来自于拟合绘制因子之外所有其他所有因子的主效应模型。残差在因子的各水平之间有所不同时指示该因子为一个重要的主效应。

注意：没有区组因子图。

预测刻画器

显示组合模型的预测刻画器。有关预测刻画器的详细信息，请参见《刻画器指南》。

“拟合确定性筛选”平台选项

设置阶段 1 p 值 设置用于阶段 1 的 *p* 值阈值。请参见第 259 页的“阶段 1 方法”。

设置阶段 2 比值 指定阶段 2 均方误差与阶段 1 均方误差的比值。请参见第 260 页的“阶段 2 方法”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

“拟合确定性筛选”平台的技术细节

“DSD 的有效模型选择”算法

本节汇总了“拟合确定性筛选”平台中使用的算法。请参见 Jones and Nachtsheim (2016)。

响应分解

“有效模型选择”算法以两个响应 Y_{ME} 和 Y_{2nd} 的形式表示响应 Y，以便 $Y = Y_{ME} + Y_{2nd}$。

- Y_{ME} 是从针对主效应和假因子的 Y 回归获取的预测值。
  由于设计的折叠结构，无需在 Y_{ME} 中包含区组因子。区组因子包括在 Y_{2nd} 中。
- Y_{2nd} 的计算公式为 $Y_{2nd} = Y - Y_{ME}$。

注意：在 DSD 中，列 Y_{ME} 和 Y_{2nd} 正交。

该分析分为两个阶段：

- 阶段 1：响应 Y_{ME} 用于标识主效应。阶段 1 标识视为活跃的主效应。
- 阶段 2：响应 Y_{2nd} 用于标识二阶效应。阶段 2 考虑阶段 1 中活跃主效应的二阶项，并确定这些包含效应中视为活跃的效应子集。

注意：若有分区组因子，则该因子包含在阶段 2 效应列表中，即便它不显著。

阶段 1 方法

阶段 1 方法依赖于该设计是否包含假因子或中心点重复。

案例 1：假因子或中心点重复可用

1. 使用假因子或中心点重复，将构造独立于模型的误差方差估计量。假定没有活跃的三阶或更高的奇阶效应，该估计值则无偏。
2. 使用 Y_{ME}，针对该估计值检验主效应。*p* 值小于阈值 *p* 值的主效应被认为是活跃的。阈值如下：
   - 对于有一个误差自由度的情况，阈值为 0.20。

- 对于有两个误差自由度的情况，阈值为 0.10。
- 对于两个以上误差自由度的情况，阈值为 0.05。
- 用户指定的 *p* 值是阈值。

注意: 要指定不同的 *p* 值阈值，请从“拟合确定性筛选”小三角菜单中选择**设置阶段 1 p 值**。

3. 若没有任何主效应的 *p* 值小于阈值，则可认定没有活跃主效应和活跃的双因子效应。该过程到此为止。
4. 若找到活跃的主效应，则非活跃主效应的变异性都合并到在 (1) 中构造的误差方差中。

注意：若设计中有分类因子，则每次将主效应选作活跃效应时，都会重新计算估计的系数。

案例 2：无假因子或无中心点重复

在这种情况下，不存在独立于模型的误差方差估计量。针对从非活跃主效应构造的误差方差的估计值按顺序检验主效应子集。假定有 *m* 个主效应。

1. 估计效应的绝对值（将 Y_{ME} 用作响应）从最大到最小排列。
2. 对于每个 $1 \le i < m$，具有第 *i* 个最大绝对值的效应针对包含该效应和所有具有更大绝对值的效应的模型的调整残差平方和来进行检验。
3. 模型中具有最小 *p* 值的效应被视为活跃效应。
4. 若找到活跃主效应，则将 Y_{ME} 用作响应，使用非活跃主效应的变异性构造误差方差的估计值。

注意：要使“拟合确定性筛选”过程在情况 2 中正常工作，至少有一个主效应必须活跃，并且至少有一个必须非活跃。若没有主效应活跃，或是所有主效应都活跃，则该过程将标识一组主效应，但实现该子集的过程会被打折扣。

阶段 2 方法

在阶段 2 中，考虑的因子取决于“强遗传”选项。选定强遗传时，只考虑涉及主效应在阶段 1 中标识为活跃的那些因子的二阶效应。阶段 2 方法取决于在阶段 1 中标识的活跃主效应数。

案例 1：七个或少于七个活跃主效应

阶段 2 使用经指导的子集选择过程。目标是：只要阶段 2 的 RMSE 与阶段 1 的 RMSE 之比大于指定阈值，就继续向模型添加二阶效应。当该比值小于等于阈值时，这指示没有任何其他二阶效应要添加到模型中。默认阈值为 1。与较大阈值相比，较小的阈值会增加可能被标识为活跃项的项数。

注意：要将 RMSE 比率阈值指定为 1 之外的其他值，请从 “拟合确定性筛选”红色小三角菜单中选择**设置阶段 2 比值**。

对于阶段 2：

- 对于有一个误差自由度的情况，阈值为 0.20。
- 对于有两个误差自由度的情况，阈值为 0.10。
- 对于两个以上误差自由度的情况，阈值为 0.05。
- 用户指定的 *p* 值是阈值。

1. 针对阶段 1 的误差估计值检验 Y_{2nd} 的变异性，以确定是否存在因二阶效应导致更多的变异性。
   – 若该检验的 *p* 值超过阈值，该过程将终止，不标识任何活跃的二阶效应。
2. 若该检验的 *p* 值小于等于阈值，则会从 *k = 1* 开始，接连检验大小为 *k*、*k = 1,2,3,...* 的子集。
3. 对于每个 *k*，都针对阶段 1 的误差估计值检验该大小的每个子集的残差平方和。标识具有最小 RMSE 的子集。
4. 该过程会一直持续到找到与阶段 1 RMSE 的 RMSE 比值小于阶段 2 RMSE 比值的 *k* 为止。
5. 与 *k* 的末尾值对应的子集之前的子集中的效应被视为活跃的双因子效应。

案例 2：八个或更多活跃主效应

当在阶段 1 中标识 8 个或更多的活跃主效应时，阶段 2 对二阶项使用向前选择。

第 9 章

筛选设计

筛选设计是面向工业实验的最受欢迎的设计之一。通常在实验的初始阶段使用，它们检查很多因子以便确定对响应影响最大的那些因子。然后使用更敏感的设计研究确定的因子。因为筛选设计所需的试验次数通常比其他设计少，因此它们是改进过程的相对经济高效的方式。

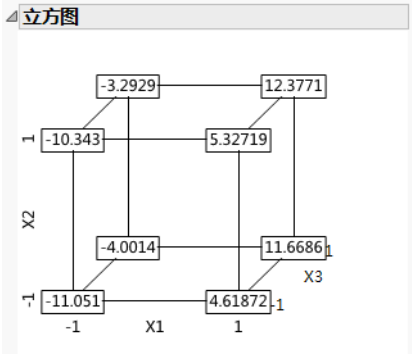
若存在适用于您的实验情形的标准筛选设计，您可以从几种标准筛选设计中选择一个。该列表包括适用的区组设计。您的因子可以是两个水平的连续因子、三个水平的分类因子或只能取离散值的连续因子（**离散数值**因子）。

不过，有些情形并不存在标准筛选设计。在这些情形中，“筛选设计”平台构造一个**主效应筛选设计**。主效应筛选设计是正交或接近正交的。它侧重于在交互作用可忽略不计的情况下估计主效应。

请注意，JMP 还为筛选设计提供两种好的备选设计：

- 若您怀疑有活跃的双因子交互作用或怀疑某连续因子对响应的效应图可能显示强曲率，则确定性筛选设计特别有用。请参见第 7 章，“确定性筛选设计”。
- 定制设计特别灵活，通常比使用其他方法获取的设计更经济高效。请参见第 4 章，“定制设计”。

图 9.1 部分析因设计的结果



目录

筛选设计概述 265
基础原则 265
筛选设计结果的分析 265
筛选设计示例 266
比较部分析因设计和主效应筛选设计 266
不存在标准设计的主效应筛选设计 273
“筛选设计”窗口 274
响应 275
因子 277
选择筛选类型 279
从部分析因设计列表中选择 279
设计类型 280
分辨率衡量混杂程度 282
显示和修改设计 283
主效应筛选设计 285
生成设计 286
设计 287
设计评估 287
输出选项 287
制表 288
筛选设计选项 289
筛选设计的其他示例 291
在部分析因设计中修改生成规则 292
Plackett-Burman 设计 297

筛选设计概述

筛选实验一般很小，旨在确定影响响应的因子。因为目标是确定因子（而非复杂的建模），通常只在两个水平上设置筛选设计中的连续因子。但是，筛选情形也可能涉及离散数值或分类因子，在这种情况下经典筛选设计可能不合适。“筛选设计”平台可以处理全部三种类型的因子：两水平的连续因子、分类因子和离散数值因子。

有两种类型的设计：

- 经典设计：对适用于标准筛选设计的情形，您可以从包括部分析因设计、Plackett-Burman、Cotter 和混合水平设计的列表中选择一种设计。
- 主效应筛选设计：无论是否能提供标准设计，您都可以请求 JMP 构造一个主效应筛选设计。这些设计是正交或接近正交的，侧重于在交互作用可忽略不计的情况下估计主效应。请参见第 285 页的“主效应筛选设计”。

基础原则

强调提前在实验过程中研究主效应得到效应等级的经验原则支持。该原则认为低阶效应通常更有可能比高阶效应重要。因此，筛选设计侧重于确定活跃的主效应。在需要**关注**高阶交互作用的情形中，筛选设计假定双因子交互作用比三因子交互作用重要，以此类推。请参见“从实验设计入手”一章中第 60 页的“效应等级”以及 Wu and Hamada (2009)。

筛选设计的效率还依赖于效应稀疏原则。效应稀疏声称响应中的多数变异由相对少量的效应来解释。请参见“从实验设计入手”一章中第 61 页的“效应稀疏”。

为了说明效应稀疏的重要性，请看一个含有七个两水平因子的示例。将完全析因设计与筛选设计进行比较：

- 完全析因设计包含因子水平的所有组合。试验次数是每个因子的水平数之积。在本例中，完全析因设计具有 $2^7 = 128$ 次试验。
- 相比之下，筛选设计只需要完全析因设计中的一部分试验。可以在包含八次试验的筛选设计中研究七个因子的主效应。

筛选设计结果的分析

筛选设计通常用于检验大量因子或交互作用。当有误差的自由度时，允许构造误差估计值，可以使用常规回归技术（“分析”>“拟合模型”）来分析实验结果。

但是，有时没有误差的自由度。在这种情况下，假定效应稀疏，“筛选”平台（“实验设计”>“经典”>“两水平筛选”>“拟合两水平筛选”）提供一种分析两水平设计结果的方式。“筛选”平台支持多重响应和多个因子。它自动使用图和统计量显示显著效应。请参见第 303 页的“‘拟合两水平筛选’平台”一章。有关当前章中的示例，请参见第 292 页的“在部分析因设计中修改生成规则”和第 297 页的“Plackett-Burman 设计”。

筛选设计示例

- “比较部分析因设计和主效应筛选设计”
- “不存在标准设计的主效应筛选设计”

比较部分析因设计和主效应筛选设计

在本例中，假定某个工程师要调查使用电子束焊接机焊接两个部件的过程。该工程师将两个部件安放在焊接固定装置中，使其紧贴彼此。对电子束发生器加以一定的电压所产生的电子流会对两个部件加热，致使二者熔接在一起。熔接区的理想深度为 0.17 英寸。该工程师想要研究焊接过程，以确定电子束发生器采用什么最佳设置才能生成所需的熔接区深度。

对于该研究，工程师想探索以下七个因子：

- **操作员**是操作焊接机器的技术人员。通常由两个技术人员来操作机器。
- **速度**（每分钟转数）是部件在电子束下旋转的速度。
- **电流**（安培）是影响电子束强度的电流。
- **模式**是使用的焊接方法。
- **部件壁尺寸**（毫米）是部件壁的厚度。
- **几何形状**指示是单面斜接还是双面斜接。
- **材料**是要焊接的材料类型。

请注意，其中三个因子是连续的：**速度**、**电流**和**部件壁尺寸**。四个是分类因子：**操作员**、**模式**、**几何形状**和**材料**。这些分类因子均具有两个水平。

每次加工处理后，工程师会将部件一切为二。这会显露出两个部件熔接的区域。该熔接区域的长度（用英寸表示）是焊接的穿透深度。穿透深度是本研究的响应。

本研究的目标如下：

- 探寻哪些因子影响焊接深度。
- 量化这些效应。
- 查找预测焊接深度为 0.17 英寸且容差为 ±0.05 英寸的特定因子设置。

您的实验预算最多允许进行 12 次试验。针对您的实验情形构造并比较两个设计。第一个设计是使用 8 次试验的经典部分析因设计。第二个设计是使用 12 次试验的主效应筛选设计。

构造标准筛选设计

在本节中，为该实验情形构造一个标准筛选设计。

指定响应

1. 选择**实验设计** > **经典** > **两水平筛选** > **筛选设计**。

2. 在 “响应”面板中，双击 “响应名称”下的 Y 并键入**深度**。

   请注意，默认 “目标”为最大化。您的目标是找到实现目标深度为 0.17 英寸 （介于 0.12 到 0.22 的限值之内）的因子设置。
3. 点击默认 “目标”最大化并将其更改为**匹配目标值**。
4. 在**下限**下点击并键入 0.12。
5. 在**上限**下点击并键入 0.22。
6. 将**重要性**下面的区域留空。

   由于只有一个响应，默认为该响应的 “重要性”指定 1。

完成的“响应”分级显示项如图 9.2 所示。现在，指定因子。

指定因子

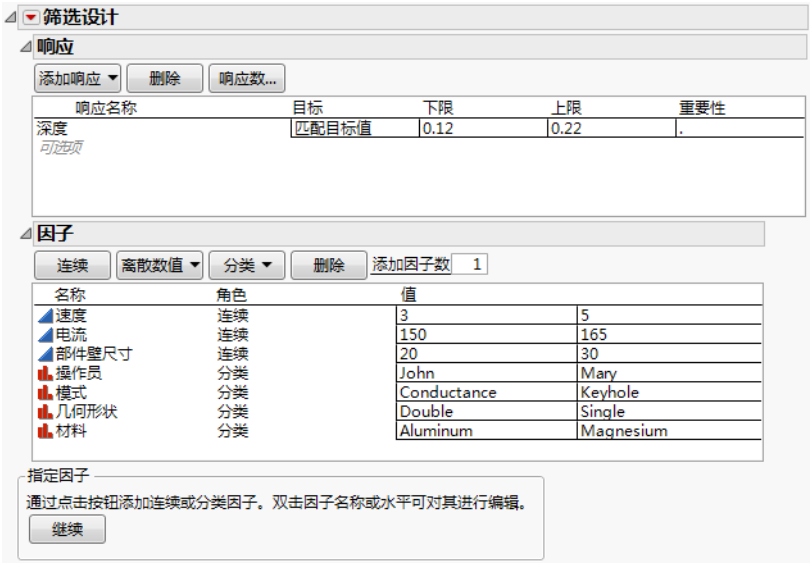
您可以手动输入因子，也可以自动输入。

- 要手动输入因子，请参见第 267 页的“手动指定因子”。
- 要自动输入因子，请使用 Weld Factors.jmp 数据表。
  1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Weld Factors.jmp。
  2. 点击 “筛选设计”红色小三角并选择**加载因子**。前进到第 268 页的 “选择设计”。

手动指定因子

1. 在**添加因子数**框中键入 3 并点击**连续**。
2. 双击 X1 并键入**速度**。
3. 使用 Tab 键在其余值和因子中移动。做出以下更改：
   a. 将**速度**值更改为 3 和 5。
   b. 将 X2 更改为**电流**，设置其值为 150 和 165。
   c. 将 X3 更改为**部件壁尺寸**，设置其值为 20 和 30。
4. 在**添加因子数**框中键入 4 并选择**分类 > 2 水平**。
5. 双击 X4 并键入**操作员**。
6. 使用 Tab 键在其余值和因子中移动。做出以下更改：
   a. 将**操作员**值更改为 John 和 Mary。
   b. 将 X5 更改为**模式**，设置其值为 Conductance 和 Keyhole。
   c. 将 X6 更改为**几何形状**，设置其值为 Double 和 Single。
   d. 将 X7 更改为**材料**，设置其值为 Aluminum 和 Magnesium。

图 9.2 焊接实验的 “响应” 和 “因子” 分级显示项



选择设计

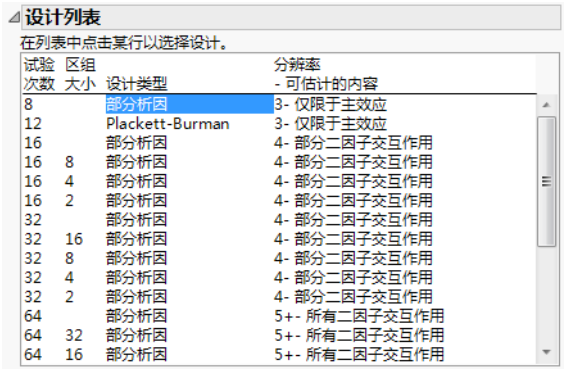
1. 点击**继续**。

   因为您指定的因子和水平组合满足标准部分析因设计的要求，因此显示 “选择筛选类型” 面板。您可以从列表中选择一个标准设计，也可以构造一个主效应设计。

   注意：在下一步设置随机种子将重现本例中所示的结果。自行构造设计时，该步骤不是必需的。

2. （可选）点击 “筛选设计” 红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
3. 接受默认选择**从部分析因设计列表中选择**，然后点击**继续**。
4. 选择第一个部分析因设计。

图 9.3 三个连续因子和四个分类因子的设计列表



设计列表

在列表中点击某行以选择设计。

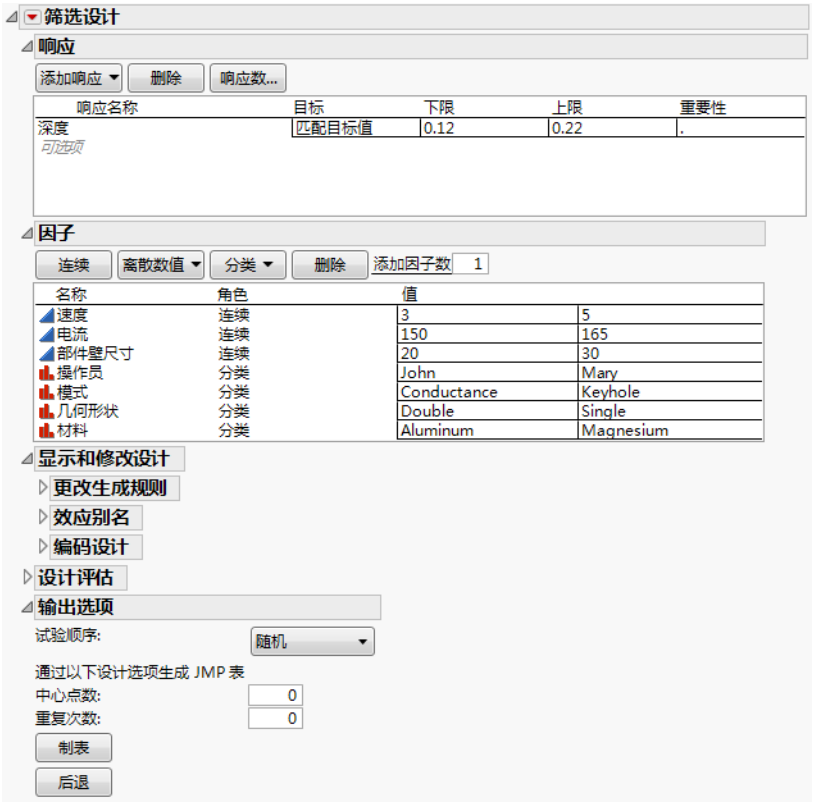
试验次数	区组大小	设计类型	分辨率 - 可估计的内容
8		部分析因	3- 仅限于主效应
12		Plackett-Burman	3- 仅限于主效应
16		部分析因	4- 部分二因子交互作用
16	8	部分析因	4- 部分二因子交互作用
16	4	部分析因	4- 部分二因子交互作用
16	2	部分析因	4- 部分二因子交互作用
32		部分析因	4- 部分二因子交互作用
32	16	部分析因	4- 部分二因子交互作用
32	8	部分析因	4- 部分二因子交互作用
32	4	部分析因	4- 部分二因子交互作用
32	2	部分析因	4- 部分二因子交互作用
64		部分析因	5+- 所有二因子交互作用
64	32	部分析因	5+- 所有二因子交互作用
64	16	部分析因	5+- 所有二因子交互作用

它指定一个包含 8 次试验、分辨率为 3 的部分析因设计。有关分辨率的信息，请参见第 282 页的 “分辨率衡量混杂程度”。

5. 点击**继续**。

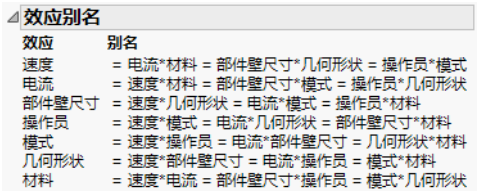
   在“输出选项”分级显示项中，我们看到**试验顺序**设置为**随机**。这意味着将以随机顺序显示设计试验。这是您进行试验时应使用的顺序。

图 9.4 已完成的 “筛选设计” 窗口



6. 打开 “显示和修改设计” 下的 “效应别名” 分级显示项。

图 9.5 包含 8 次试验的部分析因设计的别名



回想一下您选择了一个分辨率为 3 的设计 （图 9.3）。在分辨率为 3 的设计中，一些主效应与双因子交互作用混杂。“效应别名” 分级显示项指示：对于这个分辨率为 3 的设计，每个主效应都与三个双因子交互作用完全混杂。若您怀疑双因子交互作用是活跃的，则这是一个很差的设计。有关混杂的说明，请参见第 280 页的 “两水平常规部分析因”。

7. 点击**制表**。

图 9.6 设计数据表

	模式	速度	电流	部件壁尺寸	操作员	模式 2	几何形状	材料	深度
1	+++++++	5	165	30	Mary	Keyhole	Single	Magnesium	•
2	-+-+-+-	3	165	20	Mary	Conductance	Single	Aluminum	•
3	+-+--+-	5	150	30	John	Conductance	Single	Aluminum	•
4	----+++	3	150	20	John	Keyhole	Single	Magnesium	•
5	-++-+--	3	165	30	John	Keyhole	Double	Aluminum	•
6	++----+	5	165	20	John	Conductance	Double	Magnesium	•
7	+--++--	5	150	20	Mary	Keyhole	Double	Aluminum	•
8	--++--+	3	150	30	Mary	Conductance	Double	Magnesium	•

请注意以下事项：

- 该表使用您为响应、因子和水平指定的名称。
- **模式**列显示为设计试验分配的高设置和低设置。
- 这个部分析因设计是分辨率为 3 的设计。它允许您在八次试验中研究七个因子的主效应。

构造主效应筛选设计

主效应筛选设计是正交设计或接近正交的设计。在本节中，为七个因子构造主效应筛选设计。

1. 打开“筛选设计”窗口。若您关闭了它，则运行“设计数据”表中的**实验设计对话框**脚本。
2. 点击**后退**。
3. 点击**继续**。

注意：在接下来的两个步骤中设置随机种子和开始数，这将完全重现本例中显示的结果。自行构造设计时，这些步骤不是必需的。

4. （可选）点击“筛选设计”红色小三角，选择**设置随机种子**，键入 12345，然后点击**确定**。
5. （可选）点击“筛选设计”红色小三角，选择**开始数**，键入 50，然后点击**确定**。
6. 在“选择筛选类型”面板中，选择**构造主效应筛选设计**选项。
7. 点击**继续**。

   在“试验次数”下选择的选项为**默认**，即将试验次数设置为 12。保留该设置。
8. 点击**制作设计**。

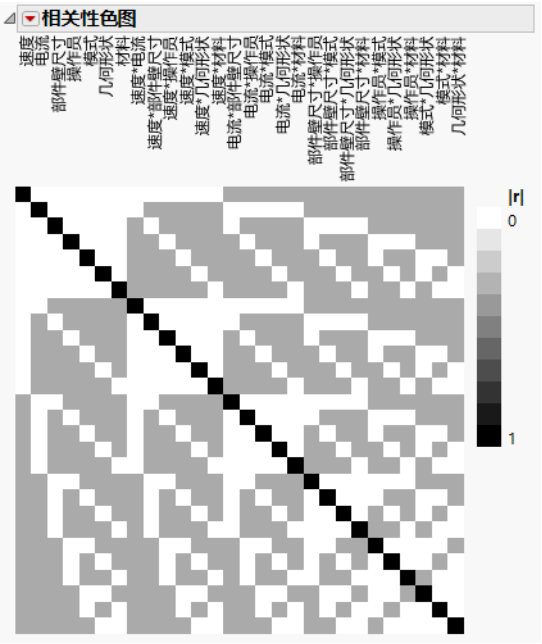
图 9.7 主效应筛选设计

设计

试验	速度	电流	部件壁尺寸	操作员	模式	几何形状	材料
1	3	165	20	John	Conductance	Single	Aluminum
2	5	150	30	Mary	Conductance	Double	Aluminum
3	3	150	30	John	Keyhole	Double	Aluminum
4	5	165	20	Mary	Keyhole	Double	Magnesium
5	3	165	20	Mary	Conductance	Double	Aluminum
6	5	150	20	John	Conductance	Double	Magnesium
7	3	150	30	Mary	Conductance	Single	Magnesium
8	5	150	20	John	Keyhole	Single	Aluminum
9	3	165	30	John	Keyhole	Double	Magnesium
10	5	165	30	John	Conductance	Single	Magnesium
11	3	150	20	Mary	Keyhole	Single	Magnesium
12	5	165	30	Mary	Keyhole	Single	Aluminum

9. 打开 “设计评估”分级显示项，然后打开 “相关性色图”分级显示项。

图 9.8 包含 12 次试验的主效应筛选设计的相关性色图



“相关性色图”显示主效应相互不相关。这用色图左上角的白色非对角方格表示。每个主效应都与某些双因子交互作用（如灰色方格所示）存在部分别名关系。悬停在其中一个灰色单元格上方可看到绝对相关性为 0.333。

在本例中，包含 12 次试验的主效应筛选设计为 Plackett-Burman 设计，您可以在 “设计列表”中获得它。但是，在很多设计情形中，主效应设计中存在的部分别名关系比在您的实验情形中采用的部分析因设计中的完全混杂更可取。

下一节将显示不存在标准设计的情形的示例。在这种情况下，JMP 构造一个主效应筛选设计。

不存在标准设计的主效应筛选设计

主效应筛选设计是主效应的正交设计或接近正交的设计。您可以在不存在标准设计的情形中使用它们来替代标准设计。主效应筛选设计是在交互作用可忽略不计的情况下估计主效应的绝佳设计。

在这种实验情况下，不存在标准设计。您需要通过一种设计来研究 13 个因子：2 个为分类因子（一个因子有 4 个水平，另一个因子有 6 个水平），另外 11 个为连续因子。

1. 选择**实验设计 > 经典 > 两水平筛选 > 筛选设计**。

   在 “响应”面板中，有一个称为 Y 的默认响应。将它继续作为默认响应。
2. 在 “因子”面板中，点击**分类**并选择 **4 水平**。

   这将添加变量 X1，其水平为 L1 到 L4。
3. 点击**分类**并选择 **6 水平**。

   这将添加变量 X2，其水平为 L1 到 L6。
4. 在**添加因子数**旁边输入 11。
5. 点击**连续**。

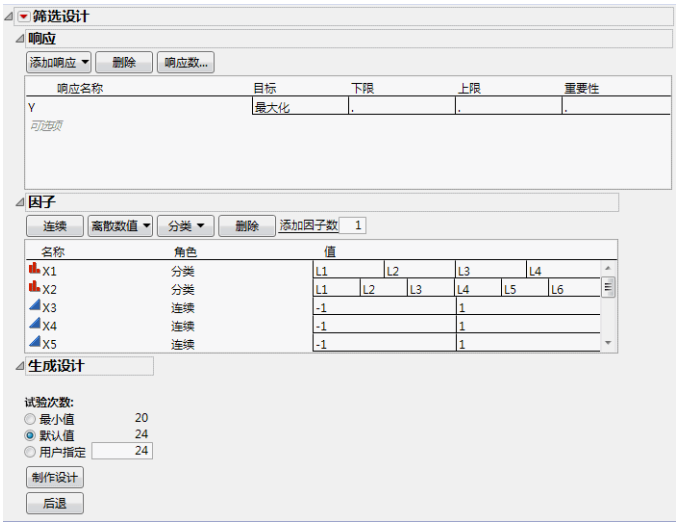
   这将添加 X3 到 X13 共 11 个因子，每个因子有两个水平：-1 和 1。
6. 点击**继续**。

   将显示 “生成设计”面板。

   不提供从 “设计列表”中选择设计的选项，因为在这种情形下没有可用的标准设计。

   保留默认试验次数 (24)。

图 9.9 带 “生成设计”面板的 13 个因子设计的 “筛选设计”窗口



7. 点击**制作设计**。

   将显示一个 “设计”和 “设计评估”分级显示项。

8. 打开 “设计”分级显示项以查看随机设计。

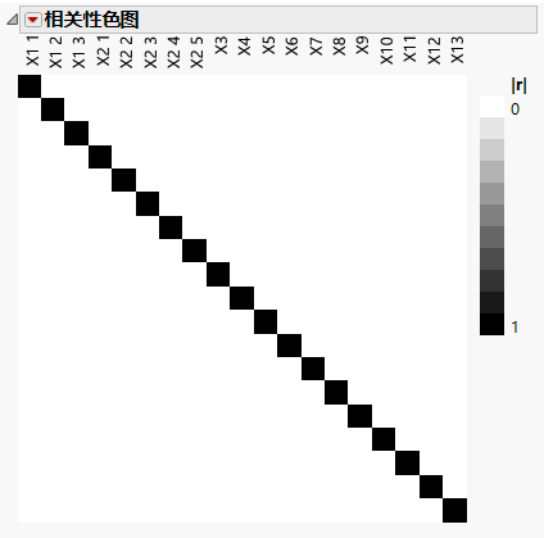
   注意：生成设计的算法使用随机起始设计。要重现该设计，请通过从报表标题旁边的红色小三角菜单选择**将脚本保存至脚本窗口**保存包含随机种子的脚本。

   接着查看 “相关性色图”，看看这个特定设计是否是正交的。

9. 打开**设计评估 > 相关性色图**分级显示项。

   色图（图 9.10）在主对角线上显示黑色条目（使用 JMP 默认颜色），这指示相关性为 1。这是因为每个对角单元格对应于项与自身的相关性，而这是 1。非对角相关性都是白色，这指示非重复项之间的相关性为 0。悬停在任意单元格上方可查看相关项和它们的绝对相关性。

图 9.10 相关性色图



10. 点击**制表**以构造设计表。

    该表包含以随机顺序执行的一定次数的试验。以这个随机顺序构造实验并在列 Y 中插入结果。运行数据表中的**模型**脚本以分析结果。

“筛选设计”窗口

逐步执行设计步骤时，“筛选设计”窗口将相应更新。请参见第 56 页的“实验设计工作流程：描述、指定、设计”。显示的分级显示项用更新窗口的按钮分隔。它们遵循下图中的流程。

图 9.11 存在标准设计时的筛选设计流程

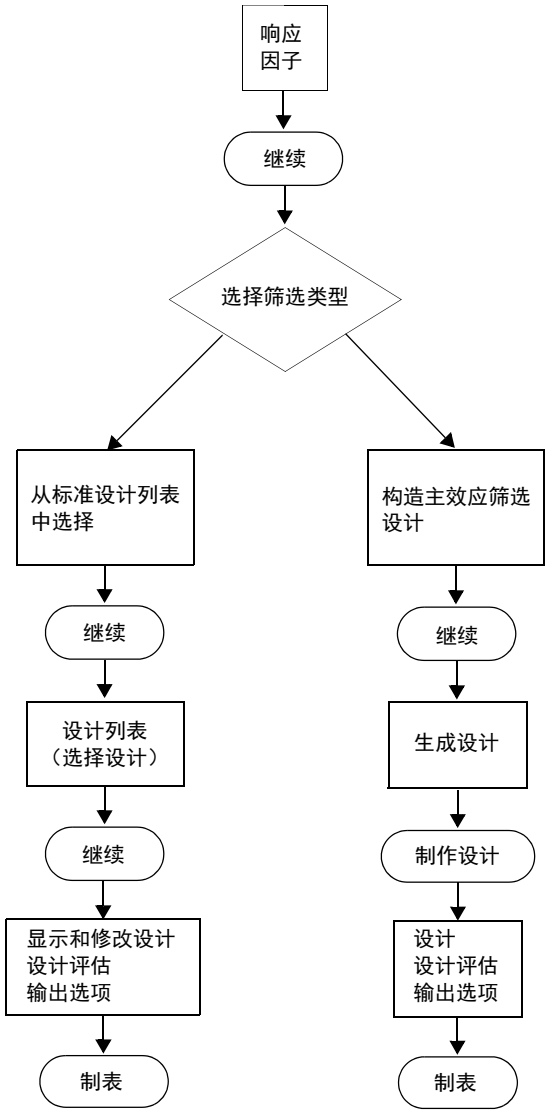
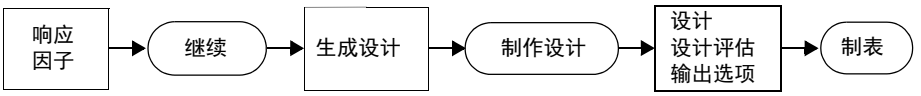


图 9.12 不存在标准设计时的筛选设计流程

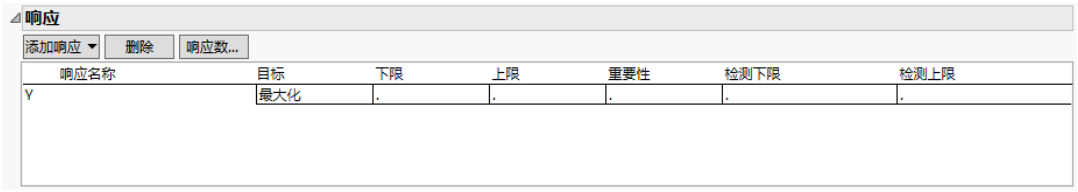


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 9.13 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的“响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
- “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定“广义回归”平台中的删失响应。请参见《拟合线性模型》中的“广义回归”一章。

编辑“响应”分级显示项

在“响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

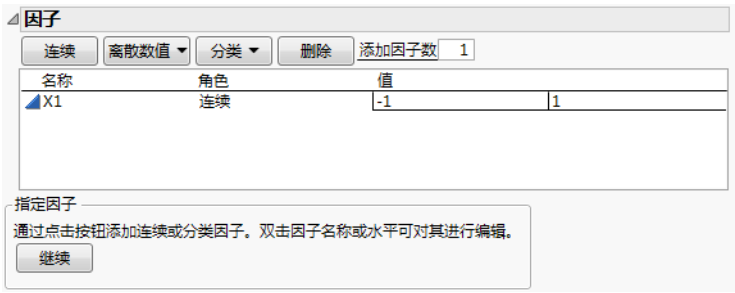
若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

在“因子”分级显示项中添加因子。

提示：填写完“因子”分级显示项后，考虑从红色小三角菜单中选择**保存因子**。该选项会将因子名称、角色、更改和值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 9.14 “因子”分级显示项



连续型 添加一个“连续”因子。生成的数据表中的数据类型为“数值”。在符合过程和测量系统的限制条件下，若一个因子在概念上可以设置为您提供的下限与上限之间的任何值，则这样的因子为连续因子。

离散数值 添加“离散数值”因子。离散数值因子只能采用一组离散数值。这些值具有隐含的顺序。生成的数据表中的数据类型为“数值”。

筛选设计包含离散数值因子的所有水平并尝试平衡这些水平。“拟合模型”将离散数值因子视为连续预测变量。

有 k 个水平（其中，$k > 2$）的离散数值因子的默认值是整数 1, 2, ..., k。有 $k = 2$ 个水平的离散数值因子的默认值为 -1 和 1。用您计划在实验中使用的设置替换这些默认值。

分类 添加一个“分类”因子。点击以选择或指定水平数。生成的数据表中的数据类型为“字符”。各水平的值排序就是从左到右输入值的顺序。在创建设计数据表后，该排序将保存在“值顺序”列属性中。

分类因子的默认值为 L1, L2, ..., Lk，其中 k 是您指定的水平数。用与实验相关的水平名称替换默认值。

删除 删除选定的因子。

添加因子数 添加多个因子。输入要添加的因子数，点击**添加因子**，然后选择因子类型。重复执行**添加因子数**，添加不同类型的多个因子。

“因子”分级显示项

“因子”分级显示项包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、X2 等。要更改该名称，双击它，然后输入所需名称。

角色 因子的“设计角色”。因子的“设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。因子的“角色”决定保存至数据表的其他因子属性。请参见第 279 页的“因子列属性”。

值 因子的实验设置。

编辑“因子”分级显示项

在“因子”分级显示项中，请注意以下事项：

- 要编辑某个因子名称，双击该因子名称。
- 要编辑某个值，在值列中点击该值。

因子列属性

对于每个因子，当您通过在 “筛选设计”窗口中选择 “制表”创建设计后，各种列属性都会保存至设计表。这些属性还会自动保存至选择 “保存因子”选项后创建的数据表中。您可以在附录 A, “列属性”中找到关于这些列属性和相关示例的详细信息。

编码 若 “角色”为 “连续”或 “离散数值”，将保存因子的 “编码”列属性。该属性变换因子值，以便低值与高值分别对应于 –1 和 +1。请参见 “列属性”附录中第 709 页的 “编码”。

值顺序 若 “角色”为 “分类”或构造了 “区组”变量，则将保存因子的 “值顺序”列属性。该属性确定因子水平的显示顺序。请参见 “列属性”附录中第 723 页的 “值顺序”。

设计角色 每个因子都分配有 “设计角色”列属性。在定义因子时指定的角色确定其 “设计角色”列属性的值。选择带区组的设计时，给 “区组”因子分配 “分区组”值。“设计角色”属性反映在对实验数据建模时要如何使用因子。“设计角色”值用在 “扩充设计”平台中。请参见 “列属性”附录中第 706 页的 “设计角色”。

因子更改 每个因子都分配有 “因子更改”列属性，该属性的值为 “容易” 。“因子更改”属性反映在对实验数据建模时要如何使用因子。“因子更改”值用在 “扩充设计”和 “评估设计”平台中。请参见 “列属性”附录中第 720 页的 “因子更改”。

每区组试验次数 对于分区组因子，指示每个区组中所允许的最多试验次数。在 “因子”分级显示项中指定 “分区组”因子后，将保存该因子的 “每区组试验次数”列属性。请参见 “列属性”附录中第 729 页的 “每区组试验次数”。

选择筛选类型

输入响应和因子并点击**继续**后，将出现以下结果之一：

- 若标准设计适合您的因子和水平，将在 “选择筛选类型”面板中显示两个选项。请参见第 279 页的 “ “选择筛选类型”选项”。
- 若针对您的因子和水平并不存在所列的标准设计，则 “选择筛选类型”面板不会显示。将打开 “生成设计”分级显示项，以构造主效应筛选设计。请参见第 286 页的 “生成设计”。

“选择筛选类型”选项

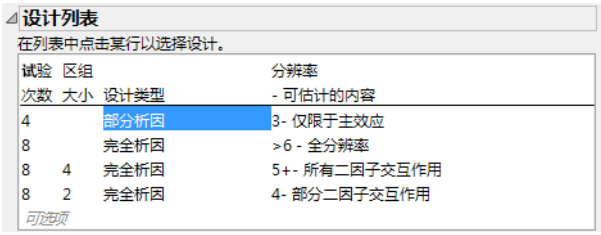
从部分析因设计列表中选择 支持您从设计列表中选择。该选项是默认选项。请参见第 279 页的 “从部分析因设计列表中选择”。

构造主效应筛选设计 打开 “生成设计”分级显示项，从中可以指定主效应筛选设计中的试验次数。有关主效应筛选设计的详细信息，请参见第 285 页的 “主效应筛选设计”。

从部分析因设计列表中选择

您可以从中选择的筛选设计列表包括将试验次数分组为大小相等的区组的设计，区组大小是 2 的幂次方。选择要使用的筛选设计类型，然后点击**继续**。

图 9.15 选择部分析因设计类型



“设计列表”包含以下列：

试验次数 设计中的总试验次数。

区组大小 区组中的试验次数。区组数通过试验次数除以区组大小得到。

设计类型 设计类型的说明。请参见第 280 页的 “设计类型”。

分辨率 指定设计的分辨率和别名类型的简短说明。请参见第 282 页的“分辨率衡量混杂程度”。

设计类型

“设计列表”提供以下设计类型：

- 第 280 页的“两水平完全析因”
- 第 280 页的“两水平常规部分析因”
- 第 281 页的“Plackett-Burman 设计”
- 第 281 页的“混合水平设计”
- 第 282 页的“Cotter 设计”

两水平完全析因

完全析因设计会针对因子水平的所有组合执行试验。样本大小是因子水平数的乘积。对于两水平设计，样本大小是 2^k，其中 k 是因子数。

完全析因设计对于所有效应都是正交的。它要求效应的估计值不相关。此外，若您从分析中删除一个效应，则其他估计值不会改变。它们的 p 值稍有改变，因为误差方差的估计值和自由度有所不同。

完全析因设计允许估计所有阶数的交互作用，其最高阶等于因子数目。但是，多数经验建模都涉及只用一阶或二阶近似表示因子与响应之间的真实函数关系。从这个角度看，完全析因设计没有高效使用试验次数。

两水平常规部分析因

常规部分析因设计的样本大小也是 2 的幂次方。对于两水平的设计，若 k 是因子数，则常规部分析因设计中的试验次数为 2^{k-p}，其中 $p < k$。2^{k-p} 部分析因设计是 k 个因子的完全析因设计的 2^{-p} 部分。像完全析因设计一样，常规部分析因设计是正交的。

k 个因子的完全析因设计提供所有交互作用（最多 *k* 次）的估计值。但是因为试验费用通常很高，因此人们更喜欢选择较小的设计。在较小的设计中，一些高阶效应与其他效应**混杂**在一起，这意味着它们无法相互区分。尽管可以估计混杂效应的线性组合，但是不可能将该变异都归属于某个特定效应或某些效应。

事实上，在设计部分析因时，需要事先决定哪些交互作用效应与其他交互作用效应混杂。实验通常不关注涉及两个以上因子的交互作用。三因子和更高阶交互作用效应通常可以忽略不计。

Plackett-Burman 设计

Plackett-Burman 设计是常规部分析因筛选设计的备选设计。Plackett-Burman 设计中的试验次数是 4 的倍数而非 2 的幂。没有试验次数在 16 和 32 之间的两水平部分析因设计，但是有试验次数为 20、24 和 28 的 Plackett-Burman 设计。

在 Plackett-Burman 设计中，主效应为正交的，且双因子交互作用只与主效应部分混杂。相比之下，在常规分辨率为 3 的部分析因设计中，一些双因子交互作用与主效应无法区分。当您想在很多因子间检测大的主效应且交互作用可以忽略不计时，Plackett-Burman 设计很有用。

混合水平设计

对于多数涉及包含三个或三个以上水平的分类或离散数值因子的设计，不存在标准设计。在这些情况下，筛选平台生成**主效应筛选设计**。这些设计对于主效应来说是正交设计或接近正交的设计。

对于存在标准混合水平设计的情形来说，在“设计列表”中给出可能的设计。“设计列表”为最多 13 个因子的纯三水平析因提供了部分析因设计。对于混合两水平和三水平设计，“设计列表”包含完整的析因和正交表设计，如表 9.1 中所列。

若您的因子数未超过该表所列的设计对应的数值，您可以通过使用适当的列子集来采用该设计。

表 9.1 混合水平设计表

	因子数	
设计	两水平	三水平
L18 John 和 L18 田口	1	7
L18 Chakravarty	3	6
L18 Hunter	8	4
L36 田口	11	12

Cotter 设计

注意：默认情况下，Cotter 设计不包含在“设计列表”中。要包含 Cotter 设计，请在“筛选设计”红色小三角菜单中取消选择**禁止 Cotter 设计**。要始终显示 Cotter 设计，请选择**文件 > 首选项 > 平台 > 实验设计**并取消选择**禁止 Cotter 设计**。

当您必须以很少的试验次数检验很多因子且其中一部分因子之间可能有交互作用时，Cotter 设计很有用。Cotter 设计依赖于效应稀疏原则。它们假定若效应和中的某个成分有活跃效应，则效应和显示一种效应。缺点是几种效果相反的活跃效应之和可能接近为零，从而无法显示为效应。因为存在这种假阴性风险，很多统计学家不建议使用这个设计。

对于 *k* 个因子，Cotter 设计有 *2k + 2* 次试验。该设计结构类似于“一次改变一个因子”方法。

按以下方式构造 Cotter 设计：

- 定义一次试验，使所有因子设置为高水平。
- 对于后面 *k* 次试验中的每一次试验，依次将一个因子设置为低水平而将其他因子设置为高水平。
- 在下一试验中将所有因子设置为低水平。
- 对于后面 *k* 次试验中的每一次试验，依次将一个因子设置为高水平而将其他因子设置为低水平。
- 试验是随机的。

当您构造 Cotter 设计时，设计数据表包含一组作为回归变量的列。这些列名采用**<因子名称>奇数**和**<因子名称>偶数**形式。通过分别累加包含给定因子的奇数阶和偶数阶交互作用项来构造它们。

例如，假定有三个因子：A、B 和 C。表 9.2 显示如何计算回归变量列中的值。

表 9.2 Cotter 设计表

奇数和偶数回归变量列累加的效应	
A 奇数 = A + ABC	A 偶数 = AB + AC
B 奇数 = B + ABC	B 偶数 = AB + BC
C 奇数 = C + ABC	C 偶数 = BC + AC

奇数和偶数列定义一个正交变换。因为这个原因，奇数和偶数列的参数检验等效于检验原始效应的组合。

分辨率衡量混杂程度

设计的**分辨率**衡量设计中的混杂程度。筛选设计需要在试验次数与设计的分辨率之间权衡利弊。

实验按**分辨率数**分为三组：

- 分辨率 3 表示一些主效应与一个或多个双因子交互作用混杂。要使主效应有意义，必须假定这些交互作用忽略不计。
- 分辨率 4 表示主效应不与其他主效应或双因子交互作用混杂。但是，一些双因子交互作用与其他双因子交互作用混杂。
- 分辨率 5 表示主效应之间、主效应与双因子交互作用之间以及双因子交互作用对之间均无混杂。一些双因子交互作用与三因子交互作用混杂。
- 分辨率 5+ 表示设计的分辨率大于 5 但是不是完全析因设计。
- 分辨率 6 表示任意阶效应之间不混杂。该设计是完全析因设计。

最小混杂设计是对于给定的分辨率使混杂数最小的设计。给定分辨率的最小混杂设计使最小长度的定义关系中的**单词**数最少。有关**单词**说明，请参见第 283 页的“更改生成规则”。有关最小混杂设计的讨论，请参见 Fries 和 Hunter (1984)。

显示和修改设计

在“设计列表”中，若您选择了包含所有连续因子或两水平分类因子或可能的分区组因子的部分析因设计，在点击“继续”后将显示“显示和修改设计”分级显示项。使用该分级显示项中的报表修改您的设计。有关更改生成规则以构造设计的示例，请参见第 292 页的“在部分析因设计中修改生成规则”。

注意：对于 Plackett-Burman 设计或 Cotter 设计，不显示“更改生成规则”和“效应别名”分级显示项，因为交互作用不完全等于主效应。

更改生成规则 指定设计的定义关系。定义关系可确定 JMP 提供完全析因设计的哪个部分。请参见第 283 页的“更改生成规则”。

效应别名 显示部分析因设计的混杂模式。点击面板底部的红色箭头以查看对指定阶的交互作用。在数据表中提供交互作用及其别名。

编码设计 显示每次试验中因子高值和低值的模式。

注意：对于 Cotter 设计，“更改生成规则”和“效应别名”分级显示项不适用，不显示它们。

更改生成规则

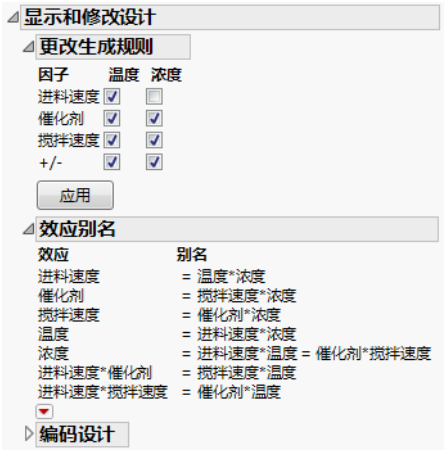
生成规则定义用于构造特定部分析因设计的关系。筛选平台中的默认生成器会生成最小混杂设计。最小混杂设计将低阶效应的别名最小化。您的实验情形可能要求您定义设计的一部分，它提供不同于标准部分的编码或别名结构。您可以通过在“显示和修改设计”分级显示项中更改生成规则来做到这一点。有关定义关系和生成规则的详细信息，请参见 Montgomery (2009)。

提示：若您想要创建一种设计以估计特定效应，请考虑使用“定制设计器”。有关定制设计的详细信息，请参见第 63 页的“定制设计”一章。

设计的定义关系由生成规则中的**单词**确定。**单词**用因子乘积表示，但是被解释为这些列的设计矩阵中条目的元素级乘积。定义关系包含其积为 1 的列的单词，称为**单位**。

图 9.16 显示 2^{5-2} 部分析因设计（5 个因子和 8 次试验）的默认生成规则。

图 9.16 标准 2^{5-2} 设计的生成规则



在“更改生成规则”面板的每个列中，在顶部列出的因子和列中其框被选定的因子构成了定义关系中的一个单词。例如，第一列指示**温度 = 进料速度 * 催化剂 * 搅拌速度**是定义关系中的一个单词。

- 若选定 +/- 框，则与生成规则相关的符号为正数，相应的单词等于单位。
- 若未选定 +/- 框，则与生成规则相关的符号为负数，相应的单词等于减单位。

请参见第 285 页的“获取 2^{5-3} 设计中的定义关系”。

完全析因设计的**主部分**是通过将所有定义关系设置为等于单位得到的部分析因设计。默认情况下，JMP 提供的析因设计是主部分。请注意，默认为所有生成规则选定 +/- 框，因此定义关系中的每个单词等于单位。

生成规则确定设计的效应的编码和别名。在某些情况下，您可能要使用这样的部分：它导致得到不同于标准部分的编码或别名结构。

- 要更改生成规则，请选中相应的框。
- 要查看您的选择对“效应别名”结果和“编码设计”的影响，请点击**应用**。

有关示例，请参见第 294 页的“更改生成规则以获取不同部分”。

获取 2^{5-3} 设计中的定义关系

图 9.16 显示两列复选框。

- 第一列表示单词**温度 = 进料速度*催化剂*搅拌速度**。
- 第一列表示单词**浓度 = 催化剂*搅拌速度**。

定义 I 以表示包含 “值 +1” 的列。因为所有因子水平为 -1 或 +1，第一列中的单词等价于**温度 * 进料速度 * 催化剂 * 搅拌速度 = I**。第二列中的单词等价于**浓度 * 催化剂 * 搅拌速度 = I**。它们一起给出了 2^{5-2} 设计的定义关系：

I = 温度 * 进料速度 * 催化剂 * 搅拌速度 = 浓度 * 催化剂 * 搅拌速度

获取 2^{5-3} 设计中的别名效应关系

“效应别名” 分级显示项中的别名结构由定义关系和因子水平为 +1 和 -1 这个事实决定。回想一下第一个生成规则是**温度 = 进料速度*催化剂*搅拌速度**，第二个规则是**浓度 = 催化剂*搅拌速度**。

为了获取 “效应别名” 分级显示项中的第一个关系，我们看到应用这两个生成规则得到了表达式：

温度 = 进料速度 * 催化剂 * 搅拌速度 = 进料速度 * 浓度

– 第二个等式使用第二个生成规则将**催化剂*搅拌速度**替换为**浓度**。

现在，将第一个和第三个表达式与**浓度**自右乘得到以下表达式：

温度 * 浓度 = 进料速度 * 浓度 * 浓度

因为设计矩阵中的**浓度**列包含值 -1 和 +1，项**浓度 * 浓度**表示值为 +1 的列。该表达式成为 “效应别名” 分级显示项中显示的第一个别名关系。

温度 * 浓度 = 进料速度 *I = 进料速度

可以使用类似计算得到其他别名关系。

主效应筛选设计

若实验涉及分类或离散数值因子，或是试验次数受限，则有可能无法构造正交设计来筛选主效应。但是，可以构造主效应筛选设计。请参见 Lekivetz et al. (2015)。

主效应筛选设计是一种包含良好的平衡属性的设计（如卡方准则所述）。请参见第 286 页的 “卡方效率”。这种设计包含主效应模型所需的统计属性。

用于生成设计的算法尝试构建强度为 2 的正交表。强度为 2 的正交表允许在交互作用可忽略不计的情况下对主效应进行正交估计。这些表是筛选设计的理想选择。分辨率为 3 的常规部分析因设计和 Plackett-Burman 设计就是强度为 2 的正交表示例。

考虑设计中因子的所有可能水平对。该算法尝试尽量平衡水平对的数目。假设已经生成了固定数目的列，将随机构建新的平衡列。定义了一个测度来反映涉及新列的水平对所达到的平衡度。该算法尝试通过在新列内交换水平使该测度最小化。

卡方效率

假定某个设计具有 n 次试验和 p 个与设计矩阵列对应的因子。

- 将因子 k 和 l 的水平分别表示为 $a = 0, 1, \ldots, s_k - 1$ 和 $b = 0, 1, \ldots, s_l - 1$。
- 将水平组合 (a,b) 出现在列 k 和 l 中的次数表示为 $n_{kl}(a,b)$。

列 k 和 l 所证明的非正交测度可通过下面的表达式得出：

$$\chi^2_{kl} = \sum_{a=0}^{s_k-1} \sum_{b=0}^{s_l-1} \frac{[n_{kl}(a,b) - n/(s_k s_l)]^2}{n/(s_k s_l)}$$

设计的平均非正交测度通过下面的表达式得出：

$$\chi^2 = \sum_{1 \le k < l \le p} \chi^2_{kl} / [p(p-1)/2]$$

可以得到 χ^2 的最大可能值，表示为 χ^2_{max}。设计的卡方效率定义如下：

$$\text{卡方效率} = 100(1 - \chi^2/\chi^2_{max})$$

卡方效率指示相对于成对水平显示极端失衡的设计，χ^2 与零接近的程度。

生成设计

当您构造主效应筛选设计时，可使用 “生成设计”分级显示项指定试验次数。要生成设计，请点击**制作设计**。

最小值 为避免生成设计失败而必需的试验次数的下限。若选择**最小值**，生成的设计是饱和的。不存在误差自由度。

注意：若对试验次数选择 “最小值”，将没有用于检验的误差项。您不能检验参数估计值。仅当因成本限制无法执行额外试验时，此选项才适合。

默认值 给出建议的试验次数。该值通过启发方式来创建比最小试验次数多至少四次试验的平衡设计。

用户指定 指定您所需的试验次数。在**试验次数**文本框中输入该值。该选项支持您在信息上可能带来的收益与额外试验成本的增加之间进行权衡。

设计

“设计”分级显示项显示主效应筛选设计的试验次数。要更改设计表的试验顺序，您可以在生成设计表之前选择“输出选项”面板中的“试验顺序”选项。

设计评估

注意：对于 Cotter 设计，不显示“设计评估”分级显示项。

“设计评估”分级显示项提供若干方式来评估所生成设计的属性。打开 “设计评估”分级显示项可查看以下选项：

功效分析 支持您探索检测给定大小的效应的能力。

预测方差刻画 显示针对因子设置范围的预测方差。

设计空间比例图 显示模型预测方差位于给定值之下或之上的比例。

预测方差曲面 显示任意两个连续因子的预测方差曲面图。

估计效率 对于每个参数，给出与理想 （正交）设计 （该设计可能不存在）的置信区间长度相比，置信区间长度的部分增加。还给出参数的相对标准误差。

别名矩阵 给出系数，这些系数指示因为可能是活跃的却不在模型中的效应导致模型参数的偏倚程度。

相关性色图 使用强度尺度在图上显示两个效应之间的绝对相关性。

设计诊断 指出用于构造设计的最优性准则。还给出设计的效率测度。

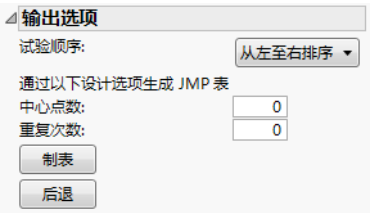
注意：当所有二因子交互作用可估计时，用于设计诊断的模型将包含所有主效应和二因子交互作用。否则，该模型包含所有主效应。

有关“设计评估”面板的更多详细信息，请参见“评估设计”一章中第 408 页的“设计评估”。

输出选项

在“输出选项”面板中指定输出数据表的详细信息。完成后，点击**制表**以构造设计的数据表。图 9.17 显示从“设计列表”中选择的某个标准设计的“输出选项”面板。对于主效应筛选设计，只有“试验顺序”可用。

图 9.17 选择输出选项



试验顺序

试验顺序选项确定试验在设计表中的顺序。

保持相同 设计表中的行与“编码设计”或“设计”分级显示项中的行具有相同的顺序。

从左至右排序 设计表中的列按从左至右排序。

随机 设计表中的行采用随机顺序。

从右至左排序 设计表中的列按从右至左排序。

区组内随机 设计表中的行在区组内采用随机顺序。（若您选择**构造主效应筛选设计**，则该选项不可用。）

中心点和重复

中心点数 指定要作为设计的中心点额外添加的试验次数。中心点是每个连续因子都设置在因子范围中心位置的试验。若您选择**构造主效应筛选设计**，则该选项不可用。

假定您的设计既包含连续因子又包含分类因子。若您在“输出选项”面板中请求中心点，这些中心点将按如下方式分配：

1. 使用在“因子”分级显示项中指定的值排序对分类因子的设置进行排序。
2. 每一个中心点都按顺序分配至分类因子设置的每种组合，重复分配过程，直到所有中心点都分配完毕。

重复次数 对于“设计列表”中的设计，指定要重复整个设计（包括中心点）的次数。一次重复会使试验次数增加一倍。若您选择**构造主效应筛选设计**，则该选项不可用。

注意：若您请求“中心点”或“重复”，并重复点击“制表”，这些操作将应用到最近构建的设计表。

制表

点击**制表**可创建包含各次试验的数据表。在该表中，将为每次试验显示您所指定的高值和低值。

图 9.18 设计数据表

	模式	进料速度	催化剂	搅拌速度	温度	浓度	反应百分比
1	−++−+	10	2	120	140	6	•
2	+−+−−	15	1	120	140	3	•
3	−+−+−	10	2	100	180	3	•
4	+++++	15	2	120	180	6	•
5	−−++−	10	1	120	180	3	•
6	+−−++	15	1	100	180	6	•
7	−−−−+	10	1	100	140	6	•
8	++−−−	15	2	100	140	3	•

表名就是生成该表的设计类型。

设计表包括以下脚本：

筛选 运行“实验设计”>“经典”>“两水平筛选”>“拟合两水平筛选”平台。仅当所有因子为两个水平时可用。

模型 运行“分析”>“拟合模型”平台。

评估设计 运行“实验设计”>“设计诊断”>“评估设计”平台。

实验设计对话框 重新创建您用于生成设计表的“筛选设计”窗口。该脚本还包含用于生成设计的随机种子。

运行**筛选**或**模型**脚本以分析数据。

若已从“设计列表”选择了设计，则设计表包含**模式**列。**模式**列包含在给定行中汇总试验的条目。低设置用“–”表示，高设置用“+”表示，中心点用“0”表示。**模式**在图中是很有用的标签变量。

筛选设计选项

“筛选设计”红色小三角菜单包含以下选项：

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用“保存响应”选项保存的响应。

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在 “设计角色”区域中，选择合适的角色。

加载因子 加载使用 “保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在 “定义因子约束”或 “线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为 “小于”或 “大于”约束。请参见 “列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用 “保存约束”选项保存的因子约束。

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的 “试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击 “制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示 “模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一： “正态”、 “二项式”或 “Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开 “模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在 “模型”窗口中做出选择以控制模拟的响应值的分布。当您点击 “应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中， Y 是响应的名称。再次点击 “应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见 “定制设计”一章中第 101 页的 “模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见 《基本分析》中的 “模拟”一章。

保存 X 矩阵 将 “矩量矩阵”和 “模型矩阵”保存至设计数据表中的表脚本。这些脚本包含矩量和设计矩阵。请参见 “定制设计”一章中第 103 页的 “保存 X 矩阵”。

警告：对于包含名义型因子的设计，使用 “保存 X 矩阵”选项保存的 “模型矩阵”**不是**拟合线性模型时使用的编码矩阵。通过在运行 “模型”脚本时获取的 “拟合模型”报表中选择选项 “保存列”>“保存编码表”，您可以获取用于拟合模型的编码矩阵。

禁止 Cotter 设计 从 “设计列表”中排除 Cotter 设计。默认选择该选项。取消选择它可以在 “设计列表”中显示 Cotter 设计。

注意：您可以设置首选项来始终显示 Cotter 设计。选择**文件 > 首选项 > 平台 > 实验设计**，然后取消选择 “禁止 Cotter 设计”。

开始数 （仅适用于主效应筛选设计。）指定算法从头开始重新生成整个设计以尝试优化最终设计的最大次数。

设计搜索时间 （仅适用于主效应筛选设计。）指定搜索设计所用的最大秒数。默认搜索时间为 15 秒。

若算法迭代需要不止几秒，“正在计算设计”进度窗口随即显示。进度条显示**卡方效率**。请参见第 286 页的 “卡方效率”。若点击进度窗口中的**取消**，计算将停止，同时给出目前为止的最佳设计。

注意：您可以为 “设计搜索时间”设置首选项。选择**文件 > 首选项 > 平台 > 实验设计**。选择**设计搜索时间**并输入最大秒数。若找到正交表，则搜索终止。在某些需要更多时间的情形中， JMP 自动延长搜索时间。

列开始数 （仅适用于主效应筛选设计。）指定该算法在转而构建下一列之前尝试优化给定列的最大次数。默认列开始数为 50。请参见第 285 页的 “主效应筛选设计”。

将脚本保存至脚本窗口 为您在 “筛选设计”窗口中指定的设计创建脚本，并将其放入打开的脚本窗口。

筛选设计的其他示例

- “在部分析因设计中修改生成规则”
- “Plackett-Burman 设计”

在部分析因设计中修改生成规则

本例改编自 Meyer, et al.(1996)，其中显示了如何在您有许多因子的情况下使用“筛选设计”平台。在该示例中，一位化学工程师要调查五个因子对于某化学反应过程的反应百分比的效应。这些因子包括：

- 进料速度 - 向反应室添加原材料的数量，以每分钟升数为单位
- 催化剂（作为百分比）
- 搅拌速度 - 反应室中推进器的每分钟转数
- 温度（摄氏度）
- 反应物的浓度

生产上的约束将实验大小限制为不超过 12 次试验。您决定考虑包含 8 次试验的部分析因设计和包含 12 次试验的 Plackett-Burman 设计。此外，您怀疑以下表述是正确的：

- **温度*浓度**交互作用是活跃的，因此您需要一个不用主效应作为该交互作用的别名的设计。
- **催化剂*温度*** 交互作用不太可能活跃。
- **搅拌速度*浓度**交互作用不太可能活跃。

在构造设计时用到该信息。

创建标准部分析因设计

要创建标准部分析因设计，请执行以下步骤：

- 第 292 页的“指定响应”
- 第 293 页的“指定因子”或第 293 页的“手动指定因子”
- 第 293 页的“选择设计”

指定响应

1. 选择**实验设计 > 经典 > 两水平筛选 > 筛选设计**。
2. 双击“响应名称”下的 Y 并键入**反应百分比**。

   请注意，默认“目标”为最大化。**目标**是最大化响应，但是最小可接受的反应百分比为 90（**下限**），上限为 100（**上限**）。
3. 在“下限”下点击并键入 90。
4. 在“上限”下点击并键入 100。
5. 将“重要性”下面的区域留空。

   由于只有一个响应，默认为该响应的“重要性”指定 1。

   完成的“响应”分级显示项请参见图 9.19。现在指定因子。

指定因子

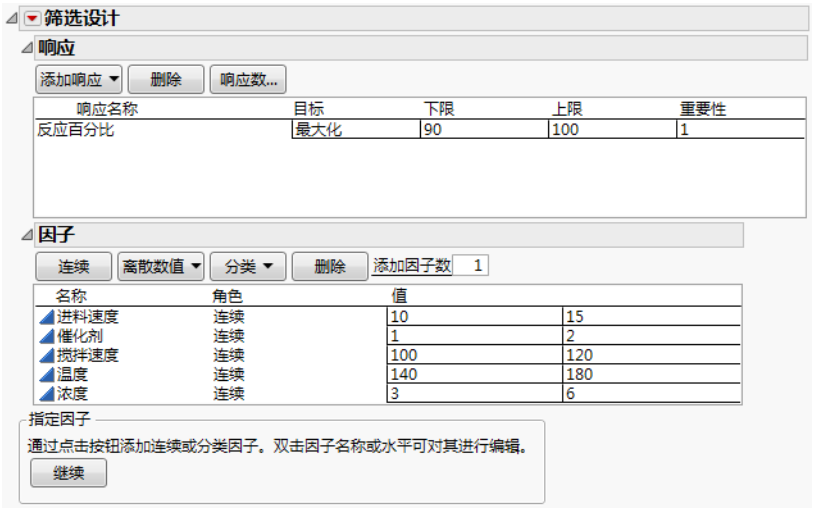
您可以手动输入因子，也可以自动输入：

- 要自动输入因子，请使用 Reactor Factors.jmp 数据表：
  1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Reactor Factors.jmp。
  2. 点击“筛选设计”红色小三角并选择**加载因子**。前进到第 293 页的“选择设计”。
- 要手动输入因子，请执行以下步骤。

手动指定因子

1. 通过在**添加因子数**框中输入 5 并点击**连续**，添加五个连续因子。
2. 将默认因子名称 (X1-X5) 分别改为**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**。
3. 输入每个因子的低值和高值：
   - 进料速度：10，15
   - 催化剂：1，2
   - 搅拌速度：100，120
   - 温度：140，180
   - 浓度：3，6

图 9.19 “响应”和“因子”分级显示项

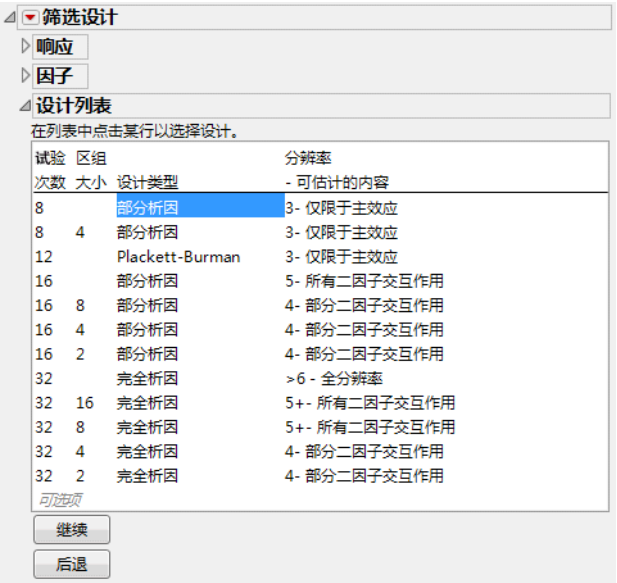


选择设计

1. 点击**继续**。
2. 从“选择筛选类型”面板中，接受默认选择**从部分析因设计列表中选择**，然后点击**继续**。

针对您指定的因子和水平的设计列在 “设计列表”中 （图 9.20）。

图 9.20 五个连续因子的部分析因设计



3. 您所要的设计是列表中第一个且碰巧默认已选定 （图 9.20）。接受该选择并点击**继续**。

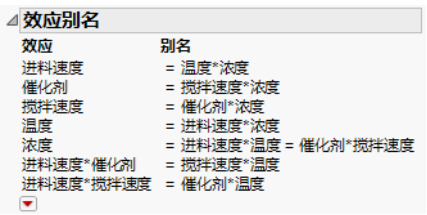
   因为您被限制为八次试验且没有分区组因子，最好的设计选项是包含 8 次试验的部分析因设计且没有区组。该设计为 2^{5-2} 部分析因设计。它是五个因子的完全析因设计的四分之一。

更改生成规则以获取不同部分

在本例中，您想知道**温度 * 浓度**交互作用是否与主效应混杂。使用 “显示和修改设计”分级显示项查看您选择的设计的别名结构，根据需要可以更改它。

1. 打开“效应别名”分级显示项。

图 9.21 “效应别名”分级显示项



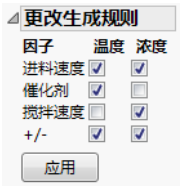
温度 * 浓度交互作用（您怀疑是活跃的）与**进料速度**主效应混杂。您想更改生成规则来构造这样的设计：**进料速度**与您怀疑不活跃的效应具有别名关系，并且**温度 * 浓度**交互作用与主效应不具有别名关系。

2. 打开 “更改生成规则”分级显示项。

   默认生成规则为您提供完全析因设计的标准（或**主**）四分之一部分。回想一下，您怀疑**催化剂 * 温度**和**搅拌速度 * 浓度**交互作用不太可能活跃。重新定义生成规则以便这两个交互作用与**进料速度**混杂。重新定义的生成规则为您提供完全析因设计的另一四分之一部分。

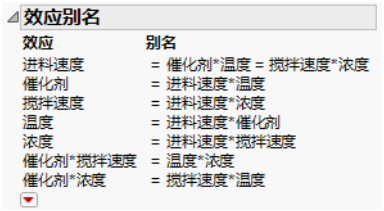
3. 请执行以下操作：
   - 在**温度**列中取消选择**搅拌速度**。
   - 在**浓度**列中取消选择**催化剂**。
   - 在**浓度**列中取消选择**进料速度**。

图 9.22 新生成规则



4. 点击**应用**。

图 9.23 已修改的生成规则的 “效应别名”分级显示项



   在您定义的设计中，**进料速度**与**催化剂 * 温度**和**搅拌速度 * 浓度**混杂。而且，**温度 * 浓度**交互作用现在与双因子交互作用**催化剂 * 搅拌速度**混杂。

5. 在 “输出选项”分级显示项中，接受**随机**的默认 “试验顺序”设置并点击**制表**。

图 9.24 包含八次试验的部分析因设计表

	模式	进料速度	催化剂	搅拌速度	温度	浓度	反应百分比
1	−−−++	10	1	100	180	6	•
2	−++−−	10	2	120	140	3	•
3	+−+−+	15	1	120	140	6	•
4	−−++−	10	1	120	180	3	•
5	++−+−	15	2	100	180	3	•
6	+++++	15	2	120	180	6	•
7	−+−−+	10	2	100	140	6	•
8	+−−−−	15	1	100	140	3	•

该设计表显示您构造的设计。请注意，该表包含您在“筛选”窗口中定义的响应**反应百分比**的列，您可以在此记录实验结果。

还包含**筛选**、**模型**和**实验设计对话框**脚本。有关这些脚本的详细信息，请参见第 288 页的“制表”。

分析结果

接着您构造实验、记录数据并分析结果。

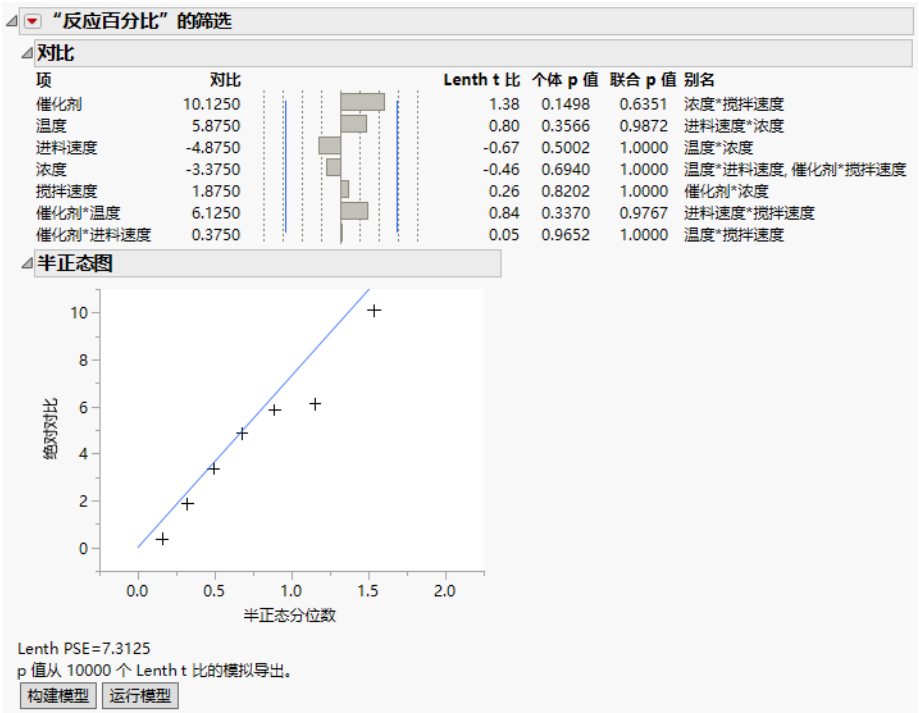
1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Reactor 8 Runs.jmp。

   您可以使用八次试验估计七个效应。在这些效应中，您期望只有几个效应是活跃的。因为您想估计七个效应，因此没有误差的自由度。因为这些原因，您使用“筛选”平台来分析结果。

2. 运行数据表中的**筛选**脚本。

   筛选脚本为您的响应和因子启动“筛选”平台（**实验设计 > 经典 > 两水平筛选 > 拟合两水平筛选**）。

图 9.25 筛选报表示例



注意：由于 *p* 值是使用基于模拟的技术获取的，您的 *p* 值可能不会与此处所示的 p 值精确匹配。

报表显示基于 Lenth t 比的个体和联合 *p* 值。即使从个体 *p* 值来看，也没有效应是显著的。半正态图显示这些效应只反映了随机噪声。

有关“筛选”报表的详细信息，请参见““拟合两水平筛选”平台”一章中第 308 页的“筛选报表”。

Plackett-Burman 设计

部分析因示例显示了针对五个连续因子的 8 次试验的部分析因设计。但假定您能够额外执行 4 次试验。在本例中，构造 12 次试验的 Plackett-Burman 设计。为便于完成“筛选”窗口，使用了“加载响应”和“加载因子”命令。

创建 Plackett-Burman 设计

1. 选择**实验设计 > 经典 > 两水平筛选 > 筛选设计**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Reactor Response.jmp。
3. 点击“筛选设计”红色小三角并选择**加载响应**。

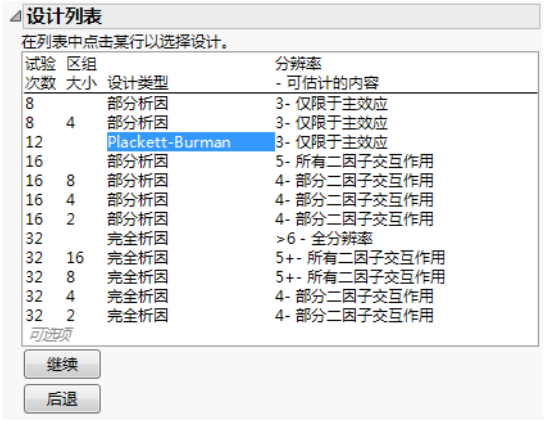
4. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Reactor Factors.jmp。
5. 点击 “筛选设计”红色小三角并选择**加载因子**。

   加载响应和**加载因子**命令使用响应和因子名称、响应的目标和限值、因子值填充 “响应”和 “因子”分级显示项。完成的 “响应”和 “因子”分级显示项请参见图 9.19。
6. 点击**继续**。

   注意：在下一步设置随机种子将重现本例中所示的试验顺序。自行构造设计时，该步骤不是必需的。
7. （可选）点击 “筛选设计”红色小三角，选择**设置随机种子**，键入 34567，然后点击**确定**。
8. 从 “选择筛选类型”面板中，接受默认选择**从部分析因设计列表中选择**，然后点击**继续**。
9. 选择 Plackett-Burman 设计。

   试验规模不是 2 的幂的 Plackett-Burman 设计倾向于具有复杂的别名结构。尤其是主效应可能与几个双因子交互作用有部分别名关系。请参见第 299 页的 “评估设计” 。我们看到 12 次试验的 Plackett-Burman 设计的分辨率指定为 3。

图 9.26 显示 Plackett-Burman 筛选设计的设计列表



10. 点击**继续**。
11. 点击**制表**。

图 9.27 Placket-Burman 设计的设计表

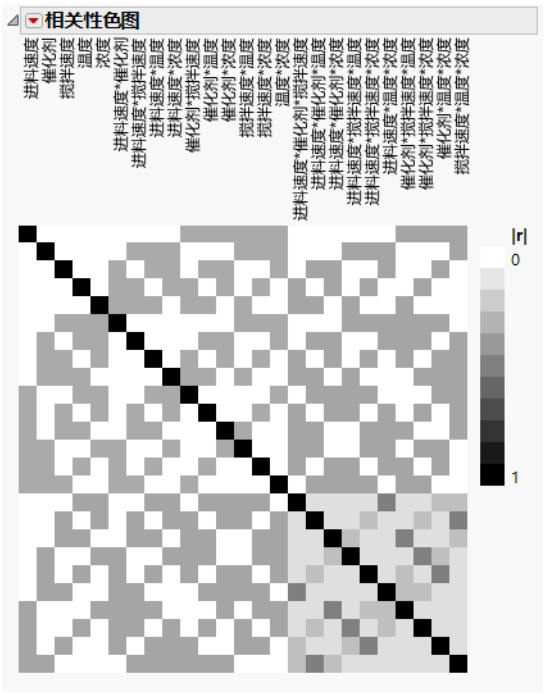
	模式	进料速度	催化剂	搅拌速度	温度	浓度	反应百分比
1	+++++	15	2	120	180	6	•
2	+−−−+	15	1	100	140	6	•
3	+−+++	15	1	120	180	6	•
4	−+++−	10	2	120	180	3	•
5	++−−−	15	2	100	140	3	•
6	−−−+−	10	1	100	180	3	•
7	−+−++	10	2	100	180	6	•
8	+++−−	15	2	120	140	3	•
9	−−+−+	10	1	120	140	6	•
10	−−+−−	10	1	120	140	3	•
11	−+−−+	10	2	100	140	6	•
12	+−−+−	15	1	100	180	3	•

设计表中包含名为**反应百分比**的列。您应按表中所示的顺序执行试验，并在**反应百分比**列中记录结果。

评估设计

1. 返回到“筛选设计”窗口。若您关闭了该窗口，请运行设计表中的“实验设计对话框”脚本。
2. 打开**设计评估** > **相关性色图**分级显示项。

图 9.28 绝对相关性色图



对角单元格的相关性为 1，这符合预期。白色单元格对应于效应间相关性等于 0 的情形。灰色和黑色单元格对应于效应间相关性大于 0 的情形。悬停在单元格上方可查看涉及的效应及其绝对相关性。例如，我们看到**进料速度**与几个双因子和三因子交互作用相关。

注意：筛选设计中的色图包含针对 8 个或不足 8 个因子的设计的三因子交互作用。设计评估脚本中的色图仅包含至多两个因子的交互作用。

3. 打开 “别名矩阵”分级显示项。

图 9.29 别名矩阵 - 最多显示双因子交互作用的部分视图

别名矩阵

效应	进料速度*催化剂	进料速度*搅拌速度	进料速度*温度	进料速度*浓度	催化剂*搅拌速度	催化剂*温度	催化剂*浓度	搅拌速度*温度	搅拌速度*浓度	温度*浓度
截距	0	0	0	0	0	0	0	0	0	0
进料速度	0	0	0	0	0.333	-0.33	-0.33	0.333	0.333	0.333
催化剂	0	0.333	-0.33	-0.33	0	0	0	0.333	-0.33	0.333
搅拌速度	0.333	0	0.333	0.333	0	0.333	-0.33	0	0	0.333
温度	-0.33	0.333	0	0.333	0.333	0	0.333	0	0.333	0
浓度	-0.33	0.333	0.333	0	-0.33	0.333	0	0.333	0	0

因为设计对于主效应是正交的，“别名矩阵”给出两个效应之间的相关性的数值。请参见“评估设计”一章中第 418 页的 “别名矩阵”。例如，我们看到**进料速度**与 6 个双因子交互作用和 4 个三因子交互作用具有部分别名关系。这些是对应于 “进料速度”行中的 0.333 和 -0.33 条目。

分析结果

数据表 Plackett-Burman.jmp 包含设计的实验的结果。回想一下，您曾怀疑**温度 * 浓度**交互作用是活跃的。以下您将假设这是唯一可能活跃的交互作用。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Plackett-Burman.jmp。
2. 通过点击左边的图标运行**模型**脚本。
3. 在 “选择列”列表中选择**温度**，在 “构造模型效应”列表中选择**浓度**。
4. 点击**交叉**。
5. 点击**运行**。

图 9.30 完整模型的参数估计值

参数估计值

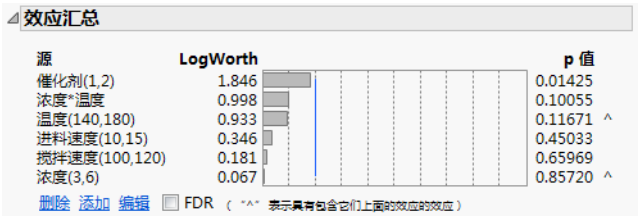
项	估计值	标准误差	t 比	概率>|t|
截距	66.166667	2.639444	25.07	<.0001*
进料速度(10,15)	-2.333333	2.850926	-0.82	0.4503
催化剂(1,2)	10.5	2.850926	3.68	0.0142*
搅拌速度(100,120)	1.3333333	2.850926	0.47	0.6597
温度(140,180)	5	2.639444	1.89	0.1167
浓度(3,6)	-0.5	2.639444	-0.19	0.8572
浓度*温度	-6.5	3.232646	-2.01	0.1006

“‘预测值 - 实际值’图”指示不存在模型失拟。“参数估计值”报表显示**催化剂**在 0.05 水平显著，**浓度 * 温度**交互作用在 0.10 水平几乎是显著的。

简化模型

您想确定对响应影响最大的那些效应。为了更清楚地看到这些活跃效应，使用“效应汇总”分级显示项去除了不显著的效应。

图 9.31 完整模型的 “效应汇总”分级显示项



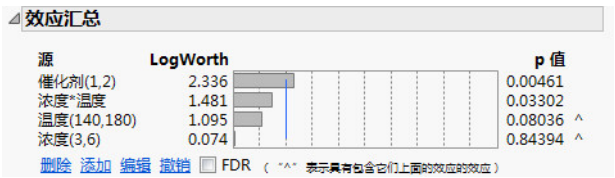
尽管**浓度**是最不显著的效应，但是在高阶交互作用 （**浓度 * 温度**）中会涉及它，这用 p 值右侧的补注符号来指示。根据效应遗传的原则，当在模型中保留**浓度 * 温度**交互作用时，不应将**浓度**从模型中删除。请参见“从实验设计入手”一章中第 60 页的 “效应遗传”。下一个最不显著的效应为**搅拌速度**。

1. 在“效应汇总”分级显示项中，选择**搅拌速度**并点击**删除**。

   进料速度是下一个最不显著的效应，可以删除它。

2. 在 “效应汇总”分级显示项中，选择**进料速度**并点击**删除**。

图 9.32 简化模型的 “效应汇总”分级显示项



p 值列指示**催化剂**主效应和**浓度*温度**交互作用在 0.05 水平上都是显著的。不能再进一步简化模型了。若所有其他交互作用不活跃或忽略不计，则您可以得出结论：**催化剂**和**浓度*温度**交互作用是活跃的效应。

第 10 章

“拟合两水平筛选”平台

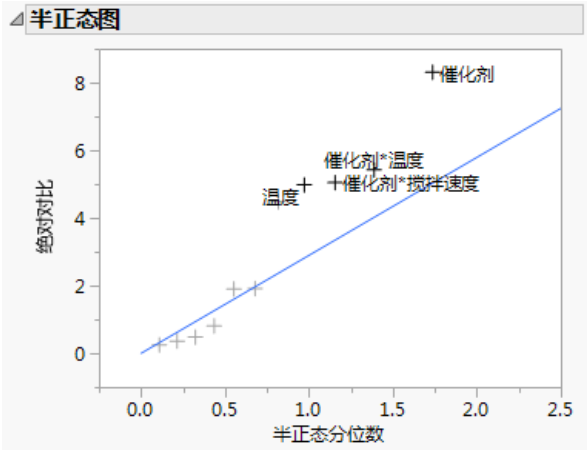
分析筛选实验中的数据

“拟合两水平筛选”平台是一个建模平台，您可以使用该平台分析从筛选设计得到的实验数据。“拟合两水平筛选”平台可帮助您识别对响应影响大的效应。

“拟合两水平筛选”平台基于效应稀疏原则 (Box and Meyer, 1986)。该原则主张在筛选设计所研究的效应中只有**相对较少**的效应是活跃的。多数效应是非活跃的，这表示其真正效应是可忽视的，并且其估计值可被视为随机误差。

当关注的模型包含交互作用项时，筛选设计往往不提供误差自由度。因此，经典效应检验不可用。在这些情形下，“拟合两水平筛选”平台尤其有用。

图 10.1 “拟合两水平筛选”报表中的半正态图



目录

“拟合两水平筛选”平台概述 305
“拟合两水平筛选”平台的示例 305
启动“拟合两水平筛选”平台 307
筛选报表 308
对比 309
半正态图 309
构建或运行模型 309
其他拟合两水平筛选分析示例 310
分析 Plackett-Burman 设计 310
效应遗传示例 311
分析超饱和设计 313
“拟合两水平筛选”平台的技术细节 314
效应的进入顺序 315
拟合两水平筛选为正交旋转 315
Lenth 伪标准误差 315
Lenth t 比 316

“拟合两水平筛选”平台概述

筛选设计的分析依赖于**效应稀疏**原则，该原则规定响应中的多数变异由少量的效应来解释。按照该原则，具有小估计值的效应用来估计模型中的误差。这样便可以检测较大效应是否活跃。

可以使用“拟合模型”（“分析”>“拟合模型”）或“拟合两水平筛选”（“实验设计”>“经典”>“两水平筛选”>“拟合两水平筛选”）分析筛选实验的数据。使用以下指导原则来选择合适的建模平台：

- 若您的因子全部为两水平且正交，则“拟合两水平筛选”平台中的所有统计量都适用。
- 若您的数据来自高度超饱和的主效应设计，则“拟合两水平筛选”平台对于选择活跃因子很有效，但对于估计误差或显著性无效。生成 *p* 值的 Monte Carlo 模拟使用的假设不适用于此类情形。
- 若您的分类或离散数值因子包含两个以上的水平，则“拟合两水平筛选”平台并不适合。JMP 将关联的模型项视为连续项。对于此类因子，变异散布在主效应和多项式效应中。在这种情况下，建议您使用“拟合模型”平台。
- 若您的数据非正交，则“拟合两水平筛选”平台中构造的估计值不同于标准回归估计值。JMP 可标识大效应，但它不会有效地检验每个效应。这是因为效应在输入模型时被人为正交化，这使得较早进入模型的效应显得没有在标准回归中显著。请参见第 315 页的“效应的进入顺序”。
- 对于混料设计，“拟合两水平筛选”平台不合适。改为参见“混料设计”一章中第 373 页的“拟合混料设计”。

“拟合两水平筛选”平台的示例

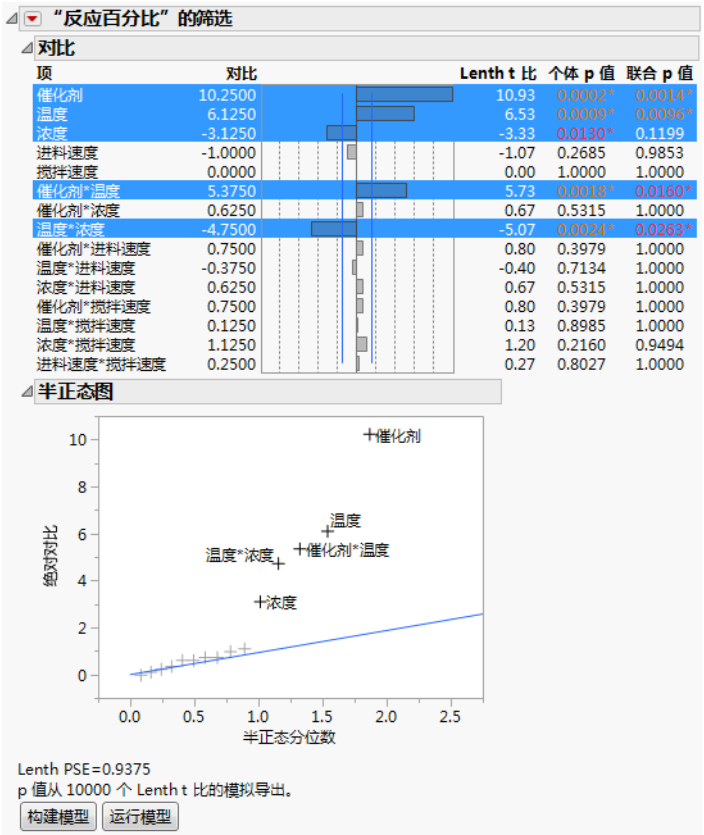
Reactor Half Fraction.jmp 样本数据表包含源自 Box et al. (1978) 中所述的设计的实验结果。您关注的是标识包含主效应和双因子交互作用的模型中的显著效应。该示例针对包含 16 次试验的设计使用一个包含 15 个参数的模型。示例中演示了“拟合两水平筛选”和“拟合模型”这两个平台的分析。

拟合两水平筛选

1. 选择**帮助** > **样本数据库**，然后打开 Reactor Half Fraction.jmp。
2. 选择**实验设计** > **经典** > **两水平筛选** > **拟合两水平筛选**。
3. 选择**反应百分比**并点击 Y。
4. 从**进料速度**一直选择到**浓度**，点击 X，然后点击**确定**。

注意：“拟合两水平筛选”平台自动构造包含交互作用项的模型。这与“拟合模型”平台形成对比，您需要在“拟合模型”中手动指定要包括在模型中的交互作用。

图 10.2 Reactor Half Fraction.jmp “拟合两水平筛选设计”报表



请注意筛选报表的以下特性：

- 个体 *p* 值小于 0.10 的效应将被选定。
- *t* 比使用 Lenth PSE（伪标准误差）计算。Lenth PSE 值显示在“半正态图”下方。
- 同时显示个体和联合 *p* 值。小于 0.05 的 p 值标有星号。
- “半正态图”支持您快速检查效应。最初在效应列表中选定的效应也在该图中进行了标记。

在本例中，已选定**催化剂**、**温度**和**浓度**以及它们的两个双因子交互作用。或者，您可以在“拟合模型”平台中拟合相同的模型。

拟合模型

1. 从 Reactor Half Fraction.jmp 表中选择**分析 > 拟合模型**。
2. 选择**反应百分比**并点击 Y。
3. 从**进料速度**一直选择到**浓度**，然后选择**宏 > 析因次数**。
4. **点击运行**。

5. 打开 “参数估计值”分级显示项。

图 10.3 饱和的 Reactor Half Fraction.jmp 设计参数估计值

响应 “反应百分比”

效应汇总

参数估计值

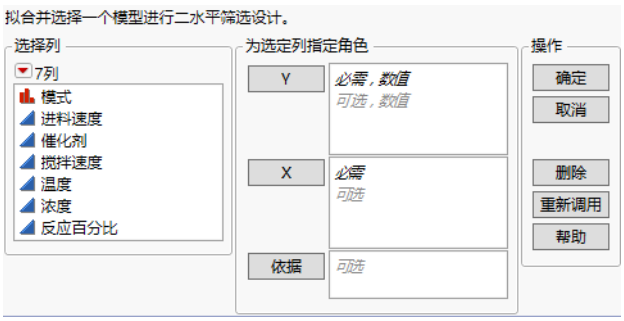
项	估计值	标准误差	t 比	概率>|t|
截距	65.25	.	.	.
进料速度(10,15)	-1	.	.	.
催化剂(1,2)	10.25	.	.	.
搅拌速度(100,120)	0	.	.	.
温度(140,180)	6.125	.	.	.
浓度(3,6)	-3.125	.	.	.
进料速度*催化剂	0.75	.	.	.
进料速度*搅拌速度	0.25	.	.	.
进料速度*温度	-0.375	.	.	.
进料速度*浓度	0.625	.	.	.
催化剂*搅拌速度	0.75	.	.	.
催化剂*温度	5.375	.	.	.
催化剂*浓度	0.625	.	.	.
搅拌速度*温度	0.125	.	.	.
搅拌速度*浓度	1.125	.	.	.
温度*浓度	-4.75	.	.	.

由于有 16 个观测和 16 个模型项，所以没有足够的观测来估计误差项。没有估计误差项则无法执行标准检验。尽管提供了参数估计值，但因为误差没有自由度，标准误差、*t* 比和 *p* 值全部缺失。这显示出“拟合两水平筛选”平台能够从筛选设计中获取最多信息的优势。

启动“拟合两水平筛选”平台

通过选择**实验设计** > **经典** > **两水平筛选** > **拟合两水平筛选**启动 “拟合两水平筛选”平台。

图 10.4 “拟合两水平筛选”平台的启动窗口



Y 要分析的列。这些必须具有 “数值”数据类型。

X 连续或离散两水平正交因子。

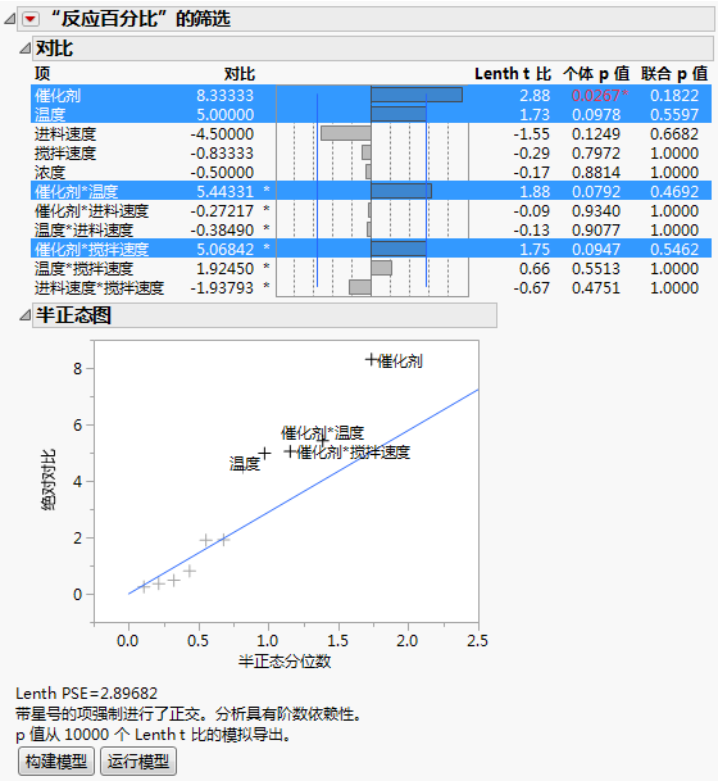
依据 一列，其中的水平定义单独分析。对于指定列的每个水平，都分析相应行。结果显示在单独的报表中。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独分析。

注意：若您在“筛选设计”平台中从部分析因设计列表创建了设计，设计表则包含名为**筛选**的脚本。运行该脚本可直接运行分析。

筛选报表

“拟合两水平筛选”报表显示设计对比 （候选模型项）汇总表和半正态图。您可以从该报表启动或运行选定模型。

图 10.5 筛选报表



- 第 309 页的“对比”
- 第 309 页的“半正态图”
- 第 309 页的“构建或运行模型”

对比

“对比”分级显示项列出模型效应、对比值、Lenth *t* 比、单个和联合 *p* 值以及别名（若存在）。效应依照等级顺序进入分析。请参见第 315 页的 “效应的进入顺序”。个体 *p* 值小于 0.10 的效应将被选定。

项 因子的名称。

对比 因子的估计值。对于正交设计，该数值与回归参数估计值相同。非正交设计则不然。对比旁边可能显示一个星号，指示非正交。

条形图 用蓝色垂线显示 Lenth *t* 比，指示在 0.10 水平下显著的值。

Lenth t 比 Lenth *t* 比，计算为**对比** /PSE，其中 PSE 是 Lenth 伪标准误差。请参见第 315 页的 “Lenth 伪标准误差”。

个体 p 值 类似于线性模型的标准 *p* 值。较小的个体 *p* 值指示显著效应。请参见第 315 页的 “Lenth 伪标准误差”。

注意：不要期望重新运行分析后 *p* 值会完全相同。Monte Carlo 方法应能在重复执行同一分析后给出类似但不是完全相同的值。

联合 p 值 经过多重比较调整的 *p* 值。

别名 仅当后面的效应确实与前面的效应存在别名关系时才显示。

半正态图

“半正态图”显示针对半正态分布的分位数绝对值绘制的对比绝对值。蓝线通过圆点，斜率为 Lenth 估计值 σ。较小的效应被视为误差项，并且假定服从均值为 0 且标准差为 σ 的正态分布。这些项落在蓝线上。显著效应是那些具有非零均值且落在蓝线以外的效应。

“半正态图”是交互的。您可以通过在所关注效应周围拖曳出一个矩形来选择模型效应。您还可以按 Ctrl 键并点击报表中的效应名称。

构建或运行模型

使用**构建模型**按钮可启动已填充选定效应的 “拟合模型”窗口。

使用**运行模型**按钮可运行选定效应的效应筛选模型。

注意：若选定的模型不服从效应遗传，则会显示 JMP 警示。您可以**取消**，使用**构建模型**按钮启动“拟合模型”窗口，然后添加缺失项。或者，您可以**继续**并构建不服从效应遗传规则的模型。请参见“从实验设计入手”一章中第 60 页的 “效应遗传”。

其他拟合两水平筛选分析示例

- “分析 Plackett-Burman 设计”
- “效应遗传示例”
- “分析超饱和设计”

分析 Plackett-Burman 设计

Plackett-Burman 设计是部分析因筛选设计的备选设计。两水平部分析因设计的试验次数必须是 2 的幂次方，这是由其本质决定的。不过，Plackett-Burman 设计存在 12、24 和 28 次试验的设计。

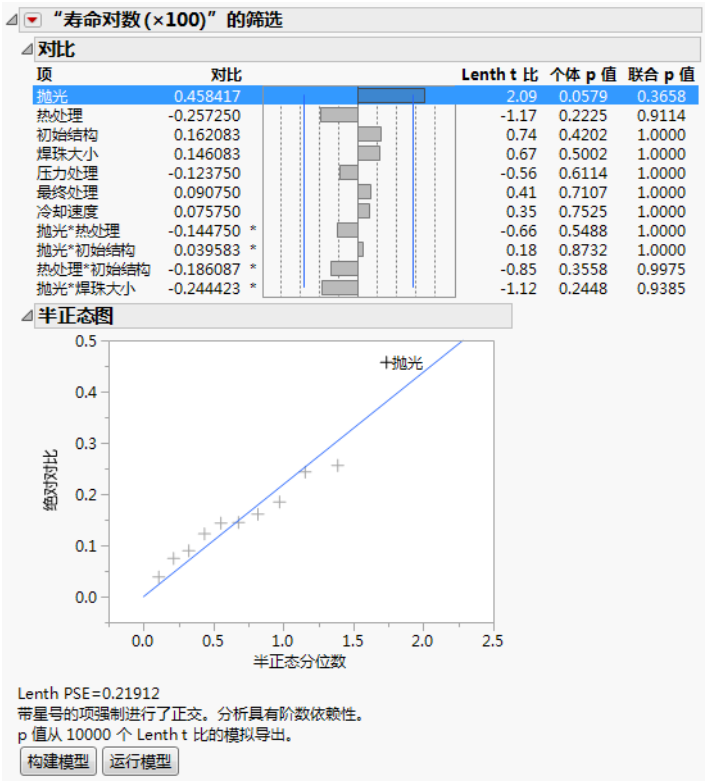
Weld-Repaired Castings.jmp 样本数据表使用 Plackett-Burman 设计，Box et al. (1978) 中所述的设计的实验结果。有七个因子被认为对焊接质量有影响。这七个因子分别是：**初始结构**、**焊珠大小**、**压力处理**、**热处理**、**冷却速度**、**抛光**和**最终处理**。使用包含 12 次试验的 Plackett-Burman 设计来调查这七个因子的重要性。响应为 100× log(lifetime)。（该样本数据表还包含对误差建模时使用的四项，但本分析中不使用这些项。）

1. 选择**帮助** > **样本数据库**，然后打开 Weld-Repaired Castings.jmp。
2. 选择**实验设计** > **经典** > **两水平筛选** > **拟合两水平筛选**。

   基于列属性填充启动窗口。寿命对数 (x100) 是响应 Y。

   为 X 指定了七个因子：**初始结构**、**焊珠大小**、**压力处理**、**热处理**、**冷却速度**、**抛光**和**最终处理**。
3. 滚至 X 窗口底部。选择 ε1、 ε2、 ε3 和 ε4，然后点击**删除**。
4. **点击确定**。

图 10.6 Weld-Repaired Castings.jmp 的筛选报表



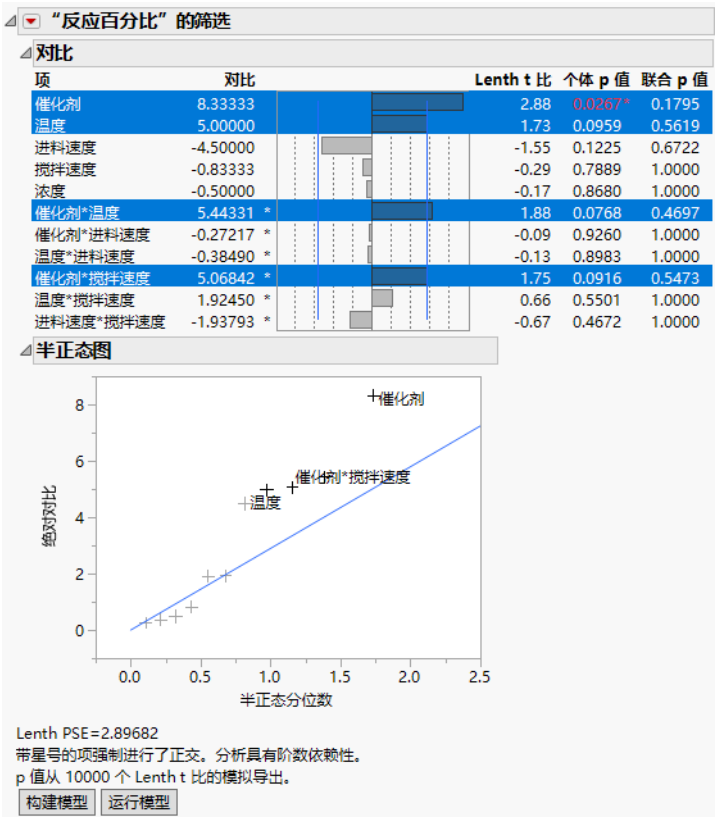
唯一标识的显著效应是**抛光**。请注意：有四项带有星号标记，指示它们不与其之前的效应正交。为这些效应获取的对比值是正交化之后的结果。因此，这些估计值不与从相应回归分析获取的估计值匹配。您可以使用**运行模型**按钮拟合单因子模型。

效应遗传示例

本例演示当“拟合两水平筛选”平台发现模型不服从效应遗传时如何构建模型。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Plackett-Burman.jmp。
2. 选择**实验设计** > **经典** > **两水平筛选** > **拟合两水平筛选**。
3. 选择**反应百分比**并点击 Y。
4. 从**进料速度**一直选择到**浓度**，然后点击 X。
5. 点击**确定**。

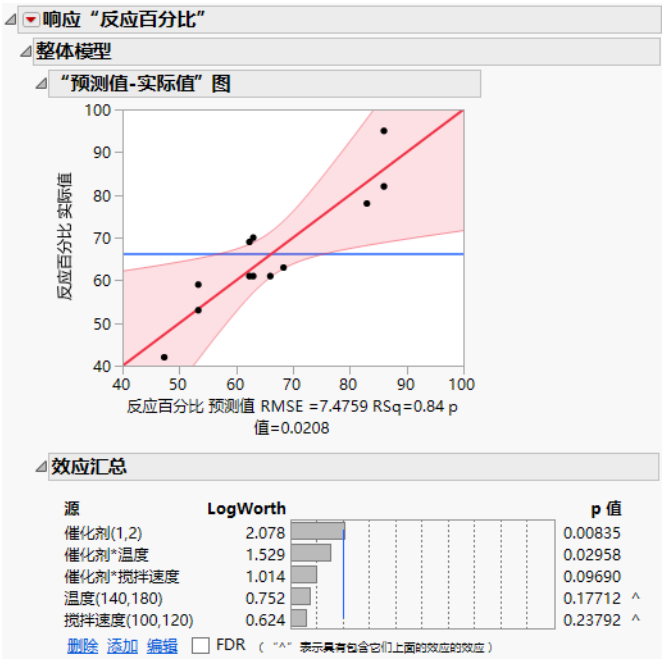
图 10.7 模型对比



催化剂 * **搅拌速度**交互作用已突出显示，但**搅拌速度**主效应未突出显示。按照效应遗传原则，您应该向模型添加**搅拌速度**主效应。请参见“从实验设计入手”一章中第 60 页的“效应遗传”。

6. 点击**构建模型**。
7. 在“构造模型效应”部分中选择**搅拌速度**并点击**添加**。
8. 点击**运行**。

图 10.8 部分 “拟合模型”报表



搅拌速度和**温度**的 p 值右侧的 ^ 符号说明：尽管不显著，这些效应还是包含在显著高阶效应中。

分析超饱和设计

超饱和设计的因子数超过了试验次数。它们在很大程度上依赖效应稀疏原则来执行分析，所以“拟合两水平筛选”平台非常适用于此类分析。其目的是要确定哪些效应是活跃的。

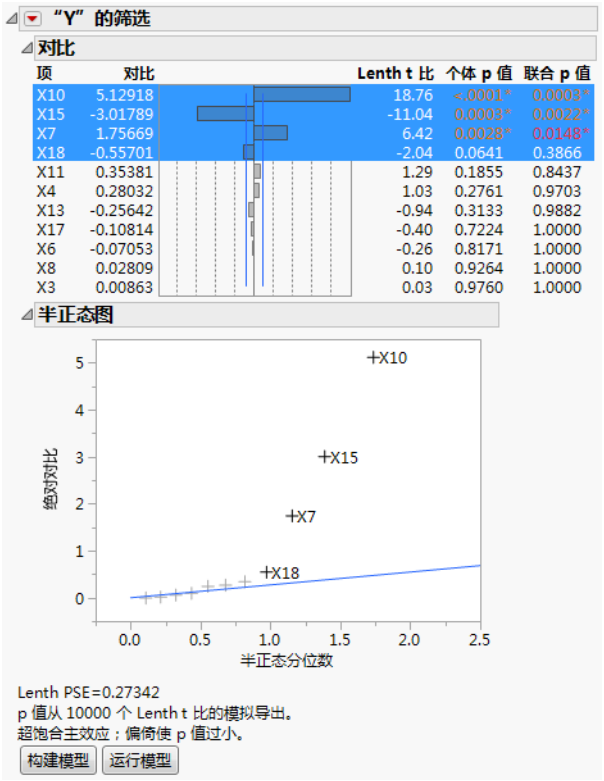
本例考虑的是包含 18 个因子而仅有 12 次试验的模拟设计。 Y 通过以下公式生成：

$$Y = 2(X7) + 5(X10) - 3(X15) + \varepsilon$$

其中， $\varepsilon \sim N(0,1)$。所以， Y 由三个活跃因子构建。

1. 选择**帮助** > **样本数据库**，然后打开 Supersaturated.jmp。
2. 选择**实验设计** > **经典** > **两水平筛选** > **拟合两水平筛选**。
3. 选择 Y 并点击 Y。
4. 从 X1 一直选择到 X18，然后点击 X。
5. 点击**确定**。

图 10.9 Supersaturated.jmp 的筛选报表



请注意，突出显示了四个因子。根据 0.05 临界值准则，因子 X10、X15 和 X7 是活跃的。在“半正态图”中，X18 接近蓝线，指示其接近 0.1 这个截止显著值。0.1 这个临界值在因子选择上很宽松，所以您不会错过可能活跃的因子。

X10、X15 和 X7 的对比估计值 5.1、–3 和 1.8 接近其模拟值（5、–3、2）。

p 值在统计上不完全有效，因为它基于的模拟假定设计正交，这不适用于超饱和设计的情况。不过，如图所示，p 值有助于标识效应以便于进一步调查。

“拟合两水平筛选”平台的技术细节

- “效应的进入顺序”
- “拟合两水平筛选为正交旋转”
- “Lenth 伪标准误差”
- “Lenth t 比”

效应的进入顺序

“拟合两水平筛选”平台具有严格定义的操作顺序：

1. 首先，主效应项根据其对比值的绝对大小进入模型。所有效应都与模型中排在它们之前的效应正交化。该方法确保这些效应的顺序与前向逐步回归中的顺序完全一致。按主效应排序还可以帮助在之后的过程中选择首选的别名项。
2. 在主效应之后，所有二阶交互作用进入模型，然后是三阶交互作用，依此类推。二阶交互作用先与前面的项交叉，而后再引入新项。例如，对于按大小排序的主效应 A、 B、 C 和 D，B*C 在 A*D 之前进入，因为 B 和 C 都大于 D。进入的完整先后顺序为：AB、 AC、 BC、 AD、 BD、 CD。
3. 若效应是模型中已有效应的完全别名关系，则它会显示在“别名”列中。若它是前面若干效应的线性组合，则不会显示。若存在部分别名关系（因为非正交），涉及的效应将用星号标记出来。
4. 该过程会一直持续到获得 n 个效应（其中， n 是数据表中的行数），这会让模型完全饱和。若因子不可能完全饱和，JMP 会生成随机正交化的效应以吸收剩余的变异。这些效应标记为**空 n**，其中 n 是一个数字。例如，若设计中有完全重复的行就会出现这种情形。

拟合两水平筛选为正交旋转

从数学上讲，“拟合两水平筛选”平台取响应向量中的 n 个值并将其旋转为 n 个新值。旋转值随后通过因子及其交互作用空间进行映射。

$$\text{对比} = \mathbf{T}' \times \text{响应}$$

在上面的等式中，**T** 是从截距、因子主效应、双因子交互作用、三因子交互作用，依此类推，一直到获取 n 个值后的一组 $n * n$ 正交化值。由于 **T** 的第一列是截距，其他所有列都与其正交，所以其他列全部为对比，即：它们总和为零。由于 **T** 正交，所以它可以用作线性模型中的 **X**。它不需要求逆，因为 $\mathbf{T}'$ 等价于 $\mathbf{T}^{-1}$ 并且 $(\mathbf{T}'\mathbf{T})\mathbf{T}'$。对比是线性模型中估计的参数。

若在除截距外模型中没有任何效应是活跃的，对比则只是随机自变量正交旋转到不同的随机自变量。这些正交旋转随机变量具有与原始随机自变量相同的方差。即便在一定程度上某些效应是活跃的，非活跃效应仍表示与模型中误差相同的变异。希望在于效应与设计足够强大，可以将活跃效应与随机误差效应分开。

Lenth 伪标准误差

Lenth 的方法 (Lenth, 1989) 称为 **Lenth 伪标准误差** (*PSE*)，它使用看起来不活跃的效应构造残差标准误差的估计值。Lenth 的 PSE 可用于估计以下实验的标准误差：对比是独立的且具有公共方差。

若有 n 行，则平台构造 $n - 1$ 个对比。用 $\hat{C}_i$ 表示这些对比，其中 $i = 1, \ldots, n - 1$。

要获得 Lenth PSE，请首先计算以下项：

$$v = 1.5[median_{i = 1, \ldots, n-1}|\hat{C}_i|]$$

Lenth PSE 基于可能不活跃的效应，定义如下：

$$PSE = 1.5\left[median_{|\hat{C}_i| < 2.5v}|\hat{C}_i|\right]$$

Lenth PSE 的值显示在筛选报表底部。

Lenth t 比

对于每个对比，*t* 比是用对比除以 PSE 计算得到的。原假设下这些 *t* 比的参考分布在计算上很难处理，因此是通过模拟得到的。使用的方法（如下所述）基于 Ye and Hamada (2000) 中的论述。

用 t_i 表示第 *i* 个对比的 *t* 比：

$$t_i = \hat{C}_i/(PSE)$$

筛选实验中最重要的是**个体误差率**，即声明给定效应为活跃的但是其实它不活跃的概率。对于第 *i* 个效应，当 $|t_i|$ 很大、落入其参考分布的上尾时会出现这种情况。

因为平台构造相对较大数量的效应，**实验级误差率**也很重要。实验级误差率是当没有效应是活跃的时候声明**任何**效应是活跃的概率。当没有效应是活跃的且绝对 *t* 比的最大值 $\max|t_i|$ 很大并落入其参考分布的上尾时，出现实验级误差。

“拟合两水平筛选”平台使用 Monte Carlo 模拟获取两种误差率类型的参考分布。考虑包含 *n - 1* 个值的一个集合，它们采用均值为 0 且标准差等于 PSE 的正态分布来模拟。这些值表示在无活跃效应的原假设下实验的可能对比值。一共生成了 10,000 组 *n - 1* 随机对比值。

个体 p 值

考虑第 *i* 个对比。使用这 10,000*(*n - 1*) 个模拟值中的每一个模拟值构造 Lenth *t* 比。用这些 *t* 比的绝对值近似表示个体误差率的参考分布。个体 *p* 值是在以降序排列的 10,000*(*n - 1*) 个模拟的绝对 *t* 比中观测到的绝对 Lenth t 比的小数位置内插值。对于参考分布，这近似于观测的绝对 Lenth t 比的绝对值右侧区域。

联合 p 值

若当所有效应都不活跃时任意 *t* 比导致拒绝原假设，会出现实验级误差。即若绝对*t* 比的最大值 $\max|t_i|$ 导致拒绝原假设，会出现实验级误差。

要获取这种情况下的参考分布，在 10,000 个模拟中考虑每个模拟的绝对 *t* 比的最大值。这 10,000 个最大值构成了参考分布。联合 *p* 值是在以降序排列的 10,000 个模拟的最大绝对 *t* 比中观测到的绝对 Lenth t 比的小数位置内插值。对于基于模拟的最大绝对 *t* 比的参考分布，这近似于绝对 Lenth t 比的绝对值右侧区域。

Monte Carlo 模拟选项

要更改集合中包含的默认模拟个数 (10,000)，您必须为名为 LenthSimN 的全局 JSL 变量赋值。用于生成 *p* 值的模拟数在报表窗口中报告。

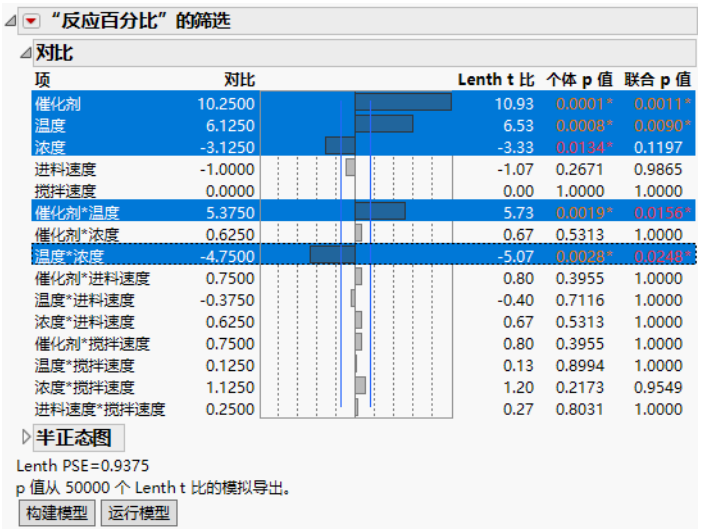
下例将模拟数设置为 50,000。

1. 选择**帮助** > **样本数据库**，然后打开 Reactor Half Fraction.jmp。
2. 选择**实验设计** > **经典** > **两水平筛选** > **拟合两水平筛选**。
3. 选择**反应百分比**并点击 **Y**。
4. 从**进料速度**一直选择到**浓度**，点击 **X**，然后点击**确定**。
5. 点击“‘反应百分比’的筛选”旁边的红色小三角菜单，然后选择**保存脚本** > **至脚本窗口**。
6. 插入 LenthSimN=50000; 作为脚本的第一行。

```
LenthSimN=50000;
Fit Two Level Screening(
  Y( :Percent Reacted),
  X( :Feed Rate, :Catalyst, :Stir Rate, :Temperature, :Concentration )
);
```

7. 在脚本窗口中右击并选择**运行脚本**。

图 10.10 使用用于生成 *p* 值的 N = 50,000 次模拟来拟合两水平



模拟数在报表窗口中说明。

注意：若 LenthSimN=0，则使用标准 *t* 分布且不提供联合 *p* 值。不推荐使用该方法。

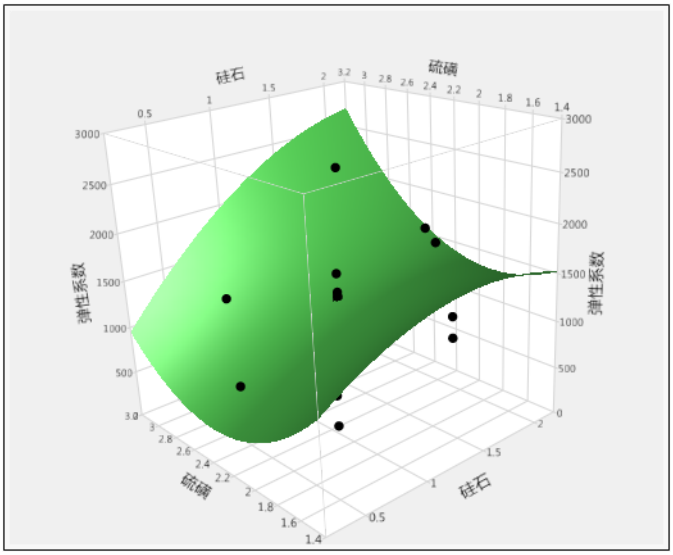
第 11 章

响应曲面设计

响应曲面设计对于二次曲面建模很有用。响应曲面模型可以标识响应的最小值或最大值所在的点（若设计区域内有这样的点）。拟合二次函数要求为每个因子提供三个非重复值，因此标准两水平设计对于拟合曲面不合适。

响应曲面设计能够拟合响应的二阶预测方程。这些方程中的二次项可对真正响应函数中的曲率建模。若设计区域内存在最大值或最小值，则可以估计出现该值的点。

图 11.1 响应曲面设计结果的模型



目录

响应曲面设计概述 321
响应曲面设计的示例 323
构造 Box-Behnken 设计 324
分析实验数据 325
探索最优设置 327
“响应曲面设计”窗口 330
响应 330
因子 332
选择设计 334
指定输出选项 335
制表 336
响应曲面设计选项 337

响应曲面设计概述

“响应曲面设计”平台提供经典中心复合设计和 Box-Behnken 设计，包括这些设计的区组版本。对于中心复合设计，您可以控制轴点的放置和设计的其他方面。响应曲面设计只能用于连续因子，最多允许有八个因子。

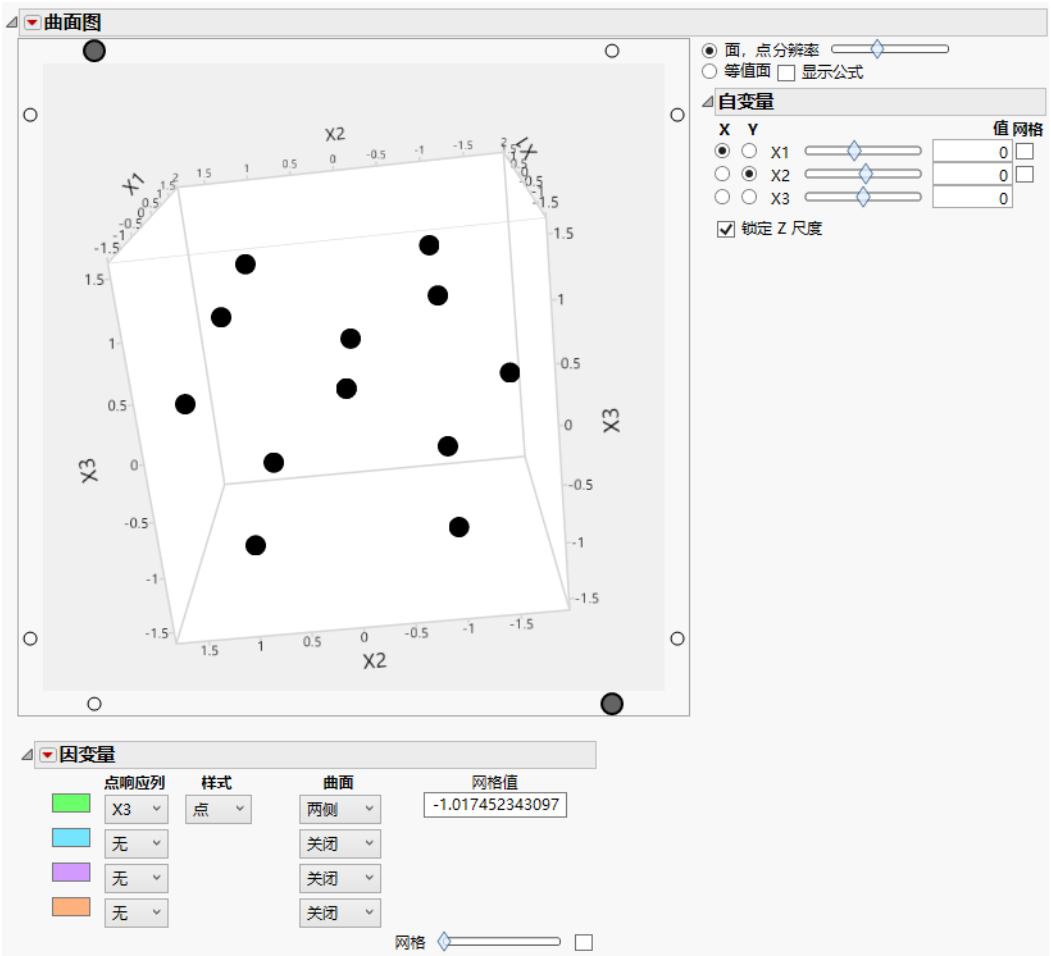
提示：您可以使用“实验设计” >“定制设计”来构造适合特定实验情形的最优响应曲面设计。与经典响应曲面设计相比，“定制设计”构造的响应曲面设计更加灵活。尤其是您可以使用“定制设计”平台创建涉及分类因子或八个以上连续因子的响应曲面设计。您还可以指定试验次数和对设计空间的限制。有关示例，请参见“定制设计示例”一章中第 142 页的“响应曲面实验”。

中心复合设计（图 11.2）组合了两水平的部分析因设计和其他两种类型的点：

- **中心点**，其中将所有因子值设置为中间值。
- **轴点**，其中将一个因子设置为高值或低值（**轴**值），将所有其他因子设置为中间值。

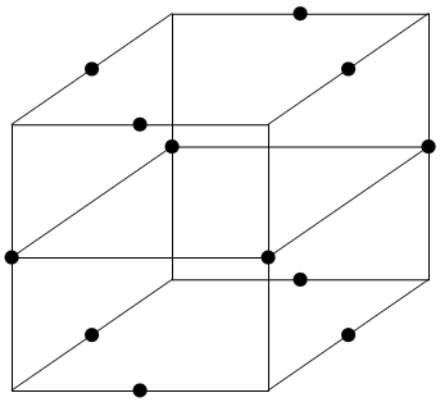
根据您相对轴点的选择，中心复合设计可以最多为每个因子设置五个非重复设置，轴点可以扩展到因子的指定范围之外。

图 11.2 三个因子的中心复合设计



Box-Behnken 设计（图 11.3）中的每个因子只有三个水平，并且在因子范围定义的立方体的顶点处没有设计点。当您由于工程考虑必须避免这些点时，这类设计很有用。但是，在立方体的顶点处没有设计点意味着与中心复合设计相比，Box-Behnken 设计具有更高的预测方差，因此靠近顶点精度更低。

图 11.3 三个因子的 Box-Behnken 设计



在 JMP 中，您可以通过两种方式构造响应曲面设计：

- 使用“响应曲面设计”平台（最多允许八个连续因子）
- 使用“定制设计”平台（并点击“模型”分级显示项中的“RSM”按钮）

在这两种情况下，设计表都包含一个“模型”脚本，您可以运行它来拟合模型。“模型”脚本将“响应曲面效应”特性应用到每个主效应，以便在“拟合模型”窗口中显示主效应时包含 &RS 后缀。该特性确保“拟合最小二乘法”报表包含“效应曲面”报表。有关该报表的详细信息，请参见《拟合线性模型》。

注意：在“标准最小二乘法”报表中没有为包含 20 个以上连续因子的响应曲面设计显示“响应曲面”分级显示项。

响应曲面设计的示例

在本例中，您为轮胎胎面实验构造 Box-Behnken 设计。您的目标是为伸长率的测度（**拉伸**）匹配目标值 450。拉伸值的变化可以表示为用于制造轮胎胎面化合物的**硅石**、**硅烷**和**硫磺**的量的函数。您想在很大范围的因子设置上进行实验，以找到可实现目标的设置。

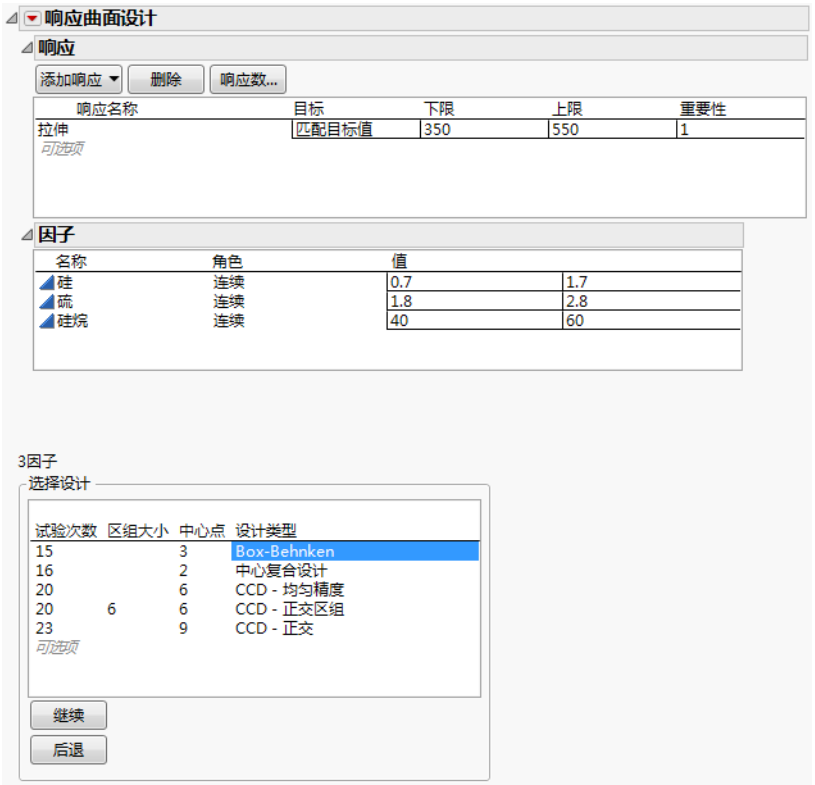
- “构造 Box-Behnken 设计”
- “分析实验数据”
- “探索最优设置”

构造 Box-Behnken 设计

在本例中，为了方便起见，您从现有表加载响应和因子。自行设计新实验时，手动输入响应和因子。请参见第 330 页的 “响应”和第 332 页的 “因子”。

1. 选择**实验设计 > 经典 > 响应曲面设计**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Bounce Response.jmp。
3. 点击 “响应曲面设计”红色小三角并选择**加载响应**。
4. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Bounce Factors.jmp。
5. 点击 “响应曲面设计”红色小三角并选择**加载因子**。

图 11.4 轮胎胎面设计的 “响应”和 “因子”分级显示项



在 “响应”分级显示项中，显示将 “拉伸”的 “目标”设置为 “匹配目标值”。

在 “选择设计”面板中，显示可能的设计。

注意：在第 6 步中设置随机种子可确切重现本例所示的结果。自行构造设计时，该步骤不是必需的。

6. （可选）点击 “响应曲面设计”红色小三角并选择**设置随机种子**。键入 12345 并点击**确定**。
7. 点击**继续**以保留 Box-Behnken 设计选择。
8. 点击**制表**。

图 11.5 Box-Behnken 设计表

	模式	硅	硫	硅烷	拉伸
1	−0+	0.7	2.3	60	•
2	0+−	1.2	2.8	40	•
3	000	1.2	2.3	50	•
4	000	1.2	2.3	50	•
5	+0−	1.7	2.3	40	•
6	+0+	1.7	2.3	60	•
7	0−+	1.2	1.8	60	•
8	0++	1.2	2.8	60	•
9	−0−	0.7	2.3	40	•
10	−+0	0.7	2.8	50	•
11	000	1.2	2.3	50	•
12	−−0	0.7	1.8	50	•
13	0−−	1.2	1.8	40	•
14	+−0	1.7	1.8	50	•
15	++0	1.7	2.8	50	•

此时，执行实验并将响应值输入数据表。

分析实验数据

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Bounce Data.jmp。

   文件 Bounce Data.jmp 包含您的实验结果。
2. 运行**模型**脚本。

   请注意，“构造模型效应”列表中的主效应后跟 & RS 后缀。该后缀指示这些是响应曲面效应，它们将在 “标准最小二乘法”报表中生成 “响应曲面”报表。
3. 点击**运行**。

图 11.6 “失拟”和“效应检验”报表

失拟

源	自由度	平方和	均方	F 比
失拟	3	11.750000	3.91667	0.9792
纯误差	2	8.000000	4.00000	概率>F
总误差	5	19.750000		0.5411
				最大 R 方
				0.9999

Box-Cox 变换

参数估计值

效应检验

源	参数数目	自由度	平方和	F 比	概率>F
硅(0.7,1.7)	1	1	12880.125	3260.791	<.0001*
硫(1.8,2.8)	1	1	5778.125	1462.816	<.0001*
硅烷(40,60)	1	1	968.000	245.0633	<.0001*
硅*硫	1	1	52441.000	13276.20	<.0001*
硅*硅烷	1	1	0.250	0.0633	0.8114
硫*硅烷	1	1	8556.250	2166.139	<.0001*
硅*硅	1	1	2592.923	656.4362	<.0001*
硫*硫	1	1	4653.231	1178.033	<.0001*
硅烷*硅烷	1	1	0.231	0.0584	0.8186

没有失拟。“效应检验”报表则指示除两个更高阶的项 （**硅石 * 硅烷**和**硅烷 * 硅烷**）之外的所有项具有低于 0.0001 的 *p* 值。有关图 11.6 中各表的解释的详细信息，请参见 《拟合线性模型》。

4. 点击 “响应曲面”旁边的展开图标以打开报表。
5. 点击 “典型曲率”旁边的展开图标。

图 11.7 “响应曲面”报表

响应曲面

系数

	硅(0.7,1.7)	硫(1.8,2.8)	硅烷(40,60)	拉伸
硅(0.7,1.7)	26.5	114.5	-0.25	-40.125
硫(1.8,2.8)	.	-35.5	46.25	-26.875
硅烷(40,60)	.	.	0.25	11

解

变量	临界值
硅(0.7,1.7)	1.7912411
硫(1.8,2.8)	2.1986422
硅烷(40,60)	23.424426

解是鞍点
临界值超出数据范围
对应于解的预测值 360.38388

典型曲率

特征值和特征向量

特征值	62.9095	3.2989	-74.9584
硅(0.7,1.7)	0.82779	-0.29879	-0.47486
硫(1.8,2.8)	0.52687	0.12315	0.84097
硅烷(40,60)	0.19280	0.94634	-0.25937

显示为报表第一部分的“系数”表给出估计的模型参数的简要汇总。前三列给出二阶项的系数。最后一列提供线性项的系数。要查看完整的预测表达式，请从“响应拉伸”红色小三角中选择**估计值 > 显示预测表达式**。

“解”报表给出出现单个临界值的点的坐标。在本例中，该点是鞍点 （出现的既非最大值也非最小值的点），位于设计空间范围之外。

“典型曲率”报表显示效应的特征值和特征向量。它们给出有关曲面的曲率的性质和方向的信息。大的正特征值 62.9095 指示正曲率，特征向量值指示曲率主要位于**硅石**方向。大的负特征值 -74.9584 指示负曲率，特征向量值指示曲率主要位于**硫磺**方向。

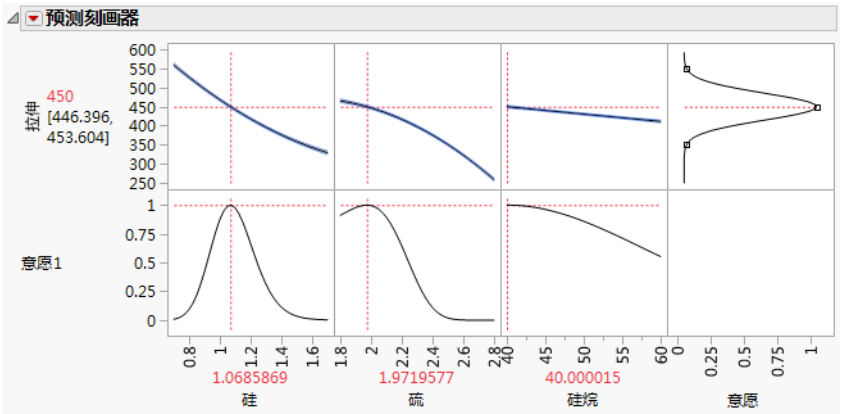
有关图 11.7 中的响应曲面分析表的详细信息，请参见 《拟合线性模型》。

接着，使用预测刻画器和等高线刻画器查找最优设置。

探索最优设置

1. 点击“预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 11.8 意愿已最大化的 Bounce Data.jmp 的预测刻画器



注意：您的最优设置可能有所不同。这是因为对于预测的**拉伸**为 450 有很多点。

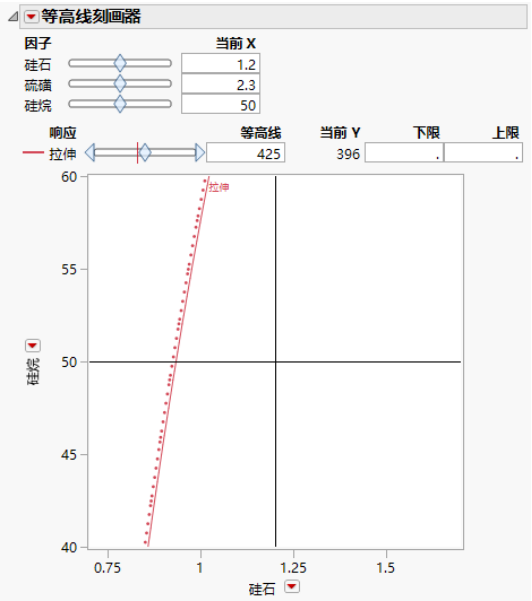
您指定了响应后，将目标设置为匹配目标值 450 （上下限分别为 350 和 550）。将该目标传递到设计表，在**拉伸**的 “响应限”列属性中设置这些限制。从这些响应限 （图 11.8 中的右上角单元格）构造意愿函数。请参见 “列属性”附录中第 700 页的 “响应限”。

您最大化意愿函数时，JMP 确定一个因子水平设置组合（通常来自很多可能的组合），该组合可实现预测的**拉伸**值 450。图 11.8 显示这些设置为：硅石 = 1.069，硫磺 = 1.972 且硅烷 = 40.000。接着，您使用等高线刻画器确定最大化意愿函数的其他点。

有关 “预测刻画器”的详细信息，请参见 《刻画器指南》。

2. 点击 “响应 ‘拉伸’”红色小三角并选择**因子刻画 > 等高线刻画器**。假定您要在将**硫磺**设置为值 2.0 时实现目标。此外，您要确保您为**硅烷**和**硅石**选择的设置可让预测的**拉伸值**保持在 450 附近的 5 个单位之内。
3. 在垂直轴上，点击**硫磺**旁边的红色小三角并选择**硅烷**。

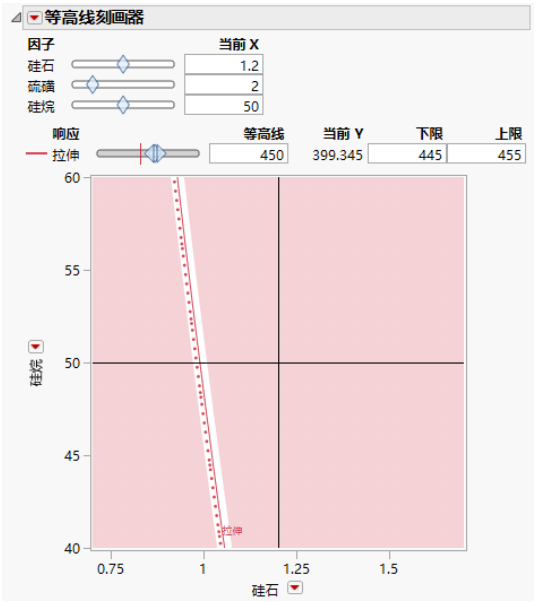
图 11.9 Bounce Data.jmp 的等高线刻画器



该图显示**拉伸**为 425 且**硫磺**为 2.3 时的**硅烷**和**硅石**的值的等高线。

4. 设置**硫磺**的**当前 X** 为 2。
5. 设置**拉伸**的**等高线**为 450。
6. 设置**拉伸**的**下限**和**上限**分别为 445 和 455。按 Enter。

图 11.10 显示硅石和硅烷的最优设置的等高线刻画器

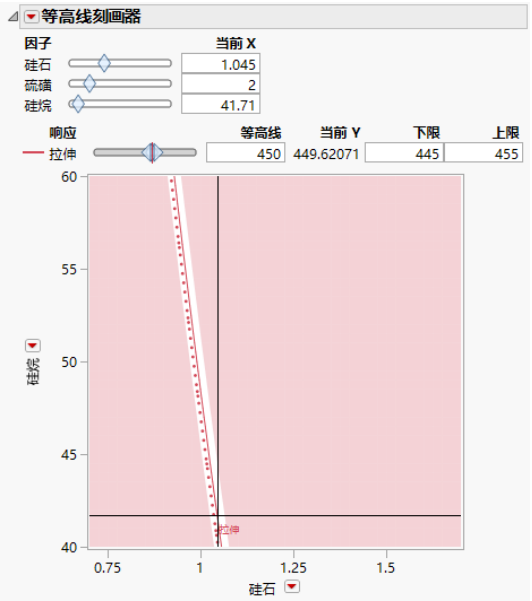


硅石和**硅烷**值的未着色带给出当**硫磺**设置为 2.0 时预测的介于 445 和 455 的**拉伸**值。红色曲线 （实线）上的值给出预测的**拉伸**值 450。

7. 将图中显示的十字准线拖到未着色带以查找从实践角度对您的过程最佳的**硅石**和**硅烷**的设置。

   假定已知您的过程在低水平的**硅烷**处比高水平处更稳健。这样您可能考虑图 11.11 中的设置。

图 11.11 显示硅石和硅烷的特定设置的等高线刻画器



对于**硫磺** = 2.0，十字准线标识的因子设置为**硅石** = 1.045 且**硅烷** = 41.71。这些设置显示在“当前 X”下。在这些设置上，预测的**拉伸**为 449.62071，显示在“当前 Y”旁边。

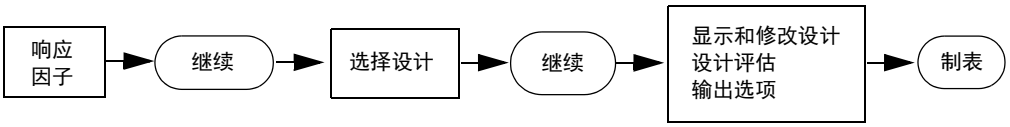
有关“等高线刻画器”的进一步信息，请参见《刻画器指南》。

“响应曲面设计”窗口

“响应曲面设计”窗口引导您逐步构造为二次曲面建模的设计。您可以选择一个中心复合设计、Box-Behnken 设计或这些设计类型的区组版本。若您选择中心复合设计，可以调整轴点。

当您执行各个设计步骤时，“响应曲面设计”窗口会相应更新。分级显示项遵循图 11.12 中的流程图，它们被一些按钮分开，点击这些按钮后便有不同的分级显示出现。

图 11.12 响应曲面设计流程

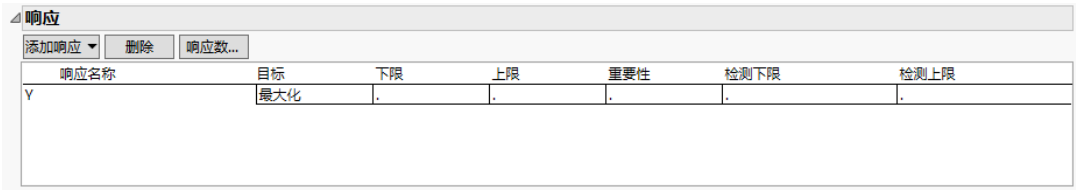


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 11.13 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的 “响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
- “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的 “重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定 “重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定 “广义回归”平台中的删失响应。请参见 《拟合线性模型》中的 “广义回归”一章。

编辑“响应”分级显示项

在 “响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

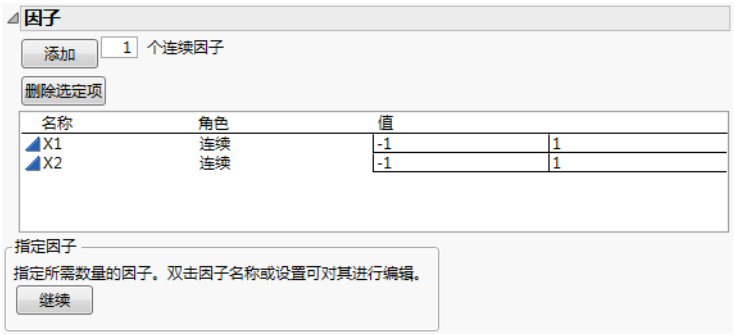
因子

响应曲面设计中的因子只能是连续的。

提示：使用**实验设计 > 定制设计**创建涉及分类因子的响应曲面设计。

响应曲面设计的初始 “因子”面板显示有两个连续因子。

图 11.14 “因子”分级显示项



“因子”分级显示项包含以下按钮：

添加 输入指定的连续因子数。

删除选定项 删除选定的因子。

提示：填写完“因子”面板后，从红色小三角菜单中选择“保存因子”。这将在数据表中保存因子名称和值，以便将来重新加载。请参见第 337 页的“响应曲面设计选项”。

“因子”分级显示项包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、X2 等。要更改该名称，双击它，然后输入所需名称。

角色 指定因子的“设计角色”为“连续”。因子的“设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。

值 因子的实验设置。要插入值，点击默认值并输入所需值。

因子列属性

对于每个因子，将各个列属性保存到已完成设计的数据表。

设计角色 每个因子都分配有“设计角色”列属性。在定义因子时指定的角色确定其“设计角色”列属性的值。选择带区组的设计时，给该因子分配“分区组”值。“设计角色”属性反映在对实验数据建模时要如何使用因子。“设计角色”值用在“扩充设计”平台中。

因子更改 每个因子都分配有“因子更改”列属性，该属性设置为“容易”。在“响应曲面设计”平台中，假定对于每次试验都可以更改因子水平。“因子更改”值用在“评估设计”和“扩充设计”平台中。

编码 若“设计角色”为“连续”，将保存因子的“编码”列属性。该属性变换因子值，以便低值与高值分别对应于 –1 和 +1。“拟合最小二乘法”报表中的估计值和检验基于变换值。

每区组试验次数 指示每个区组中的试验次数。选择带区组的设计并点击“制表”时，会将具有默认名称“区组”的因子添加到“因子”列表。将为该因子保存“每区组试验次数”列属性。

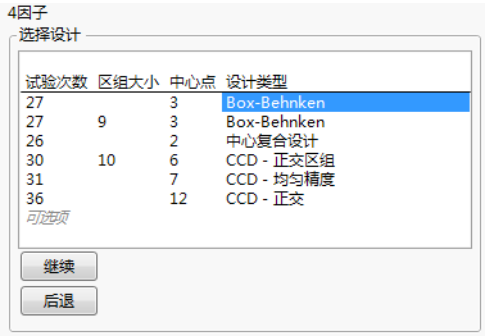
选择设计

在输入响应和因子并点击**继续**后，从设计列表中选择一种设计。设计包括两种类型：

- 第 334 页的“Box-Behnken 设计”
- 第 334 页的“中心复合设计”

选择要使用的设计并点击**继续**。

图 11.15 选择包含四个因子的设计面板



Box-Behnken 设计

Box-Behnken 设计中的每个因子只有三个水平，并且在因子范围定义的立方体的顶点处没有设计点。当出于工程考虑希望避免极端设置时，这些设计很有用。但是，与中心复合设计相比，这些设计导致顶点附近有更高的预测方差。

中心复合设计

中心复合设计具有中心点和轴点。**轴点**是将一个因子设置为高值或低值（**轴值**）并将所有其他因子设置为中间值（或中心值）的点。

中心复合设计可以有位于指定因子范围所定义的超立方面之外的轴点。这意味着每个因子可能需要五个非重复设置，其中包括位于“因子”分级显示项中指定的值范围外的两个设置。但是，JMP 允许您在该面上放置设计点。

提供以下类型的中心复合设计：

中心复合设计 针对指定因子数目的普通中心复合设计。

CCD - 均匀精度 选择一定的中心点数以使设计空间中心附近的预测方差极为平缓。

CCD - 正交 选择一定的中心点数和轴值以使二阶参数估计值与其他参数估计值最小相关。

CCD - 正交区组 二阶参数估计值和区组效应与其他参数估计值最小相关。

轴值

选择一个中心复合设计并点击**继续**时，您可以使用选项来提供轴尺度信息。在放置轴值时，显示的值用于乘以指定因子范围的一半。若您在“轴值”旁边指定值 1.0，则将所得设计中的轴点放置在因子所定义的立方体面上。您可以根据以下选项设置轴值：

图 11.16 “轴值”面板

可旋转 预测方差仅依赖于距设计中心的统一尺度距离。轴点比因子范围还更远。若该因子范围无法实际达到，请选择**位于表面**或自行指定值。

正交 效应是正交的。轴点比因子范围还更远。若该因子范围无法实际达到，请选择**位于表面**或自行指定值。

位于表面 将轴点放置在指定因子范围的极端。

用户指定 将轴点放置在您在 “轴值”文本框中输入的值所指定的距离处。

内切 重新设置设计尺度以便轴点位于因子范围的低端和高端。根据该尺度收缩析因设计点。

指定输出选项

您可以在“输出选项”面板中指定输出数据表的详细信息。完成时，点击**制表**以构造设计的数据表。

试验顺序

试验顺序选项确定试验在设计表中的顺序。包括以下选项：

保持相同 设计表中的行与 “设计”和 “预期系数”分级显示项中的行具有相同的顺序。

从左至右排序 设计表中的列按从左至右排序。

随机 设计表中的行采用随机顺序。

从右至左排序 设计表中的列按从右至左排序。

区组内随机 设计表中的行在区组内采用随机顺序。

中心点和重复

中心点数 设计中显示的中心点数。中心点是每个连续因子都设置在因子范围中心位置的试验。显示的初始值为您选择的设计中的中心点数。

重复次数 要重复整个设计 （包括中心点）的次数。一次重复会使试验次数增加一倍。

制表

点击**制表**以创建包含您的实验的试验次数的设计表。

图 11.17 Bounce Factors 和 Response 的正交中心复合设计

	模式	硅石	硫磺	硅烷	拉伸
1	000	1.2	2.3	50	•
2	+++	1.7	2.8	60	•
3	00A	1.2	2.3	66.680317692	•
4	--+	0.7	1.8	60	•
5	A00	2.0340158846	2.3	50	•
6	-+-	0.7	2.8	40	•
7	000	1.2	2.3	50	•
8	000	1.2	2.3	50	•
9	00a	1.2	2.3	33.319682308	•
10	a00	0.3659841154	2.3	50	•
11	000	1.2	2.3	50	•
12	000	1.2	2.3	50	•

“表”面板中左上角的**设计**注释给出生成表的设计类型（“中心复合设计”）。若您正在比较多个设计，此信息很有用。

“模式”列

模式列根据因子值给出每行中试验的符号说明。

提示：模式在图中是很有用的标签变量。

表 11.1 “模式”列说明

-	低值
+	高值
0	中间（中心）值
a	低轴值
A	高轴值

设计表脚本

设计表包括以下脚本：

模型 运行**分析 > 拟合模型**平台。

评估设计 运行 “实验设计” > “设计诊断” > “评估设计” 平台。

实验设计对话框 重新创建您用于生成设计表的 “响应曲面设计” 窗口。该脚本还包含用于生成设计的随机种子。

响应曲面设计选项

“响应曲面设计” 平台中的红色小三角菜单包含以下选项：

保存响应 将 “响应” 面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用 “保存响应” 选项保存的响应。

保存因子 将 “因子” 面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在 “设计角色” 区域中，选择合适的角色。

加载因子 加载使用 “保存因子” 选项保存的因子。

保存约束 （对于一些平台不可用）将您在 “定义因子约束” 或 “线性约束” 分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为 “小于” 或 “大于” 约束。请参见 “列属性” 附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用 “保存约束” 选项保存的因子约束。

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的 “试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示“模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一：“正态”、“二项式”或“Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在 “模型”窗口中做出选择以控制模拟的响应值的分布。当您点击 “应用”后，模拟的响应值的公式将保存在名为 <Y> **模拟**的新列中，其中， Y 是响应的名称。再次点击 “应用”将更新 <Y> **模拟**中的公式和值。

有关其他详细信息，请参见 “定制设计”一章中第 101 页的 “模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见 《基本分析》中的 “模拟”一章。

保存 X 矩阵 将 “矩量矩阵”和 “模型矩阵”保存至设计数据表中的表脚本。这些脚本包含矩量和设计矩阵。请参见 “定制设计”一章中第 103 页的 “保存 X 矩阵”。

警告：对于包含名义型因子的设计，使用 “保存 X 矩阵”选项保存的 “模型矩阵”**不是**拟合线性模型时使用的编码矩阵。通过在运行 “模型”脚本时获取的 “拟合模型”报表中选择选项 “保存列” > “保存编码表”，您可以获取用于拟合模型的编码矩阵。

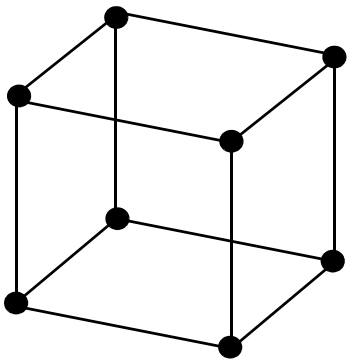
将脚本保存至脚本窗口 为您在 “响应曲面设计”窗口中指定的设计创建脚本，并在打开的脚本窗口中保存它。

第 12 章

完全析因设计

在完全析因设计定义的实验中，针对所有可能的因子设置组合都会运行试验。完全析因设计允许估计所有可能的交互作用。与筛选设计相比，完全析因设计规模较大，并且因高水平交互作用往往不活跃而可能效率低下。通常会在有少量因子和水平并且想要得到关于可能的交互作用信息的情况下使用完全析因设计。例如，完全析因设计往往会构成测量系统分析 (MSA) 的基础。

图 12.1 三个两水平因子的完全析因设计



目录

完全析因设计概述 341
完全析因设计的示例 341
构造设计 341
分析实验数据 343
构建完全析因设计 347
响应 347
因子 349
选择输出选项 351
制表 352
完全析因设计选项 353

完全析因设计概述

在完全析因设计中，您对因子水平的每种组合都执行一次试验。样本大小就是因子水平数的乘积。例如，包含一个两水平因子、一个三水平因子和一个四水平因子的析因设计需要执行 2 x 3 x 4 = 24 次试验。

“完全析因设计”平台同时支持具有任意水平数的连续因子和分类因子。假定您可采用完全随机的方式运行试验。

完全析因设计是所有设计类型中最为保守的设计。遗憾的是，由于样本大小以因子数的指数级增长，所以运行完全析因设计的成本往往太高。定制设计、确定性筛选设计和筛选设计不像完全析因设计那样保守，但更经济高效。

完全析因设计的示例

在本例中，您构造一个完全析因设计，以便研究五个两水平因子（**进料速度**、**催化剂**、**搅拌速度**、**温度**和**浓度**）对于反应器的产量的影响。由于有五个因子，每个因子有两个水平，所以完全析因设计包括至少 $2^5 = 32$ 次试验。有关涉及该实验情形的较小的筛选设计，请参见“筛选设计”一章中第 291 页的“筛选设计的其他示例”。

在本例中，您从现有表加载响应和因子。自行设计新实验时，手动输入响应和因子。请参见第 347 页的“响应”和第 349 页的“因子”。

构造设计

1. 选择**实验设计** > **经典** > **完全析因设计**。
2. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor Response.jmp。
3. 点击“完全析因设计”红色小三角并选择**加载响应**。
4. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor Factors.jmp。
5. 点击“完全析因设计”红色小三角并选择**加载因子**。
6. 点击**继续**。

图 12.2 完全析因示例 “响应”和 “因子”面板

注意：在第 7 步中设置随机种子可确保设计表中的试验采用本例所用的顺序显示。自行构造设计时，该步骤不是必需的。

7. （可选）点击 “完全析因设计”红色小三角并选择**设置随机种子**。键入 12345 并点击**确定**。

   “输出选项”面板中的 “试验顺序”设置为 “随机”。设计表中的试验顺序将采用随机种子确定的随机顺序。

   试验次数设置为 32。这是所有可能的因子水平组合的数目。

8. 点击**制表**。

   设计数据表中的第一列以 + 和 - 符号的形式（分别指示高、低因子设置）显示每次试验的因子水平组合。该表还包含一个名为**反应百分比**的空 Y 列，用于在执行实验时输入响应值。

图 12.3 反应器实验的完全析因设计

	模式	进料速度	催化剂	搅拌速度	温度	浓度	反应百分比
1	++−+−	15	2	100	180	3	•
2	+++++	15	2	120	180	6	•
3	−++−−	10	2	120	140	3	•
4	−+−−+	10	2	100	140	6	•
5	−++++	10	2	120	180	6	•
6	−+−−−	10	2	100	140	3	•
7	++−−+	15	2	100	140	6	•
8	−−+−+	10	1	120	140	6	•
9	−−−−−	10	1	100	140	3	•
10	++−++	15	2	100	180	6	•
11	+−++−	15	1	120	180	3	•
12	−+++−	10	2	120	180	3	•
13	+−−+−	15	1	100	180	3	•
14	−−++−	10	1	120	180	3	•
15	−++−+	10	2	120	140	6	•
16	+++−+	15	2	120	140	6	•
17	+−−++	15	1	100	180	6	•
18	++++−	15	2	120	180	3	•
19	−−−+−	10	1	100	180	3	•
20	+−−−+	15	1	100	140	6	•
21	−+−++	10	2	100	180	6	•
22	−−−++	10	1	100	180	6	•
23	++−−−	15	2	100	140	3	•
24	+−+−+	15	1	120	140	6	•
25	−−+++	10	1	120	180	6	•
26	+++−−	15	2	120	140	3	•
27	+−+−−	15	1	120	140	3	•
28	−−+−−	10	1	120	140	3	•
29	−−−−+	10	1	100	140	6	•
30	−+−+−	10	2	100	180	3	•
31	+−−−−	15	1	100	140	3	•
32	+−+++	15	1	120	180	6	•

分析实验数据

下一步，继续分析已完成的实验中的数据。您将使用两种方法分析结果：筛选和逐步回归。然后，您需要使用预测刻画器找到最优设置。

使用“筛选”平台进行分析

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor 32 Runs.jmp。
2. 运行**筛选**脚本。

   “筛选”报表显示一个“对比”报表和一个“半正态图”。“对比”报表显示所有 31 个可能效应（最多包括五因子交互作用）的估计值。

图 12.4 Reactor 32 Runs.jmp 的 “对比” 报表

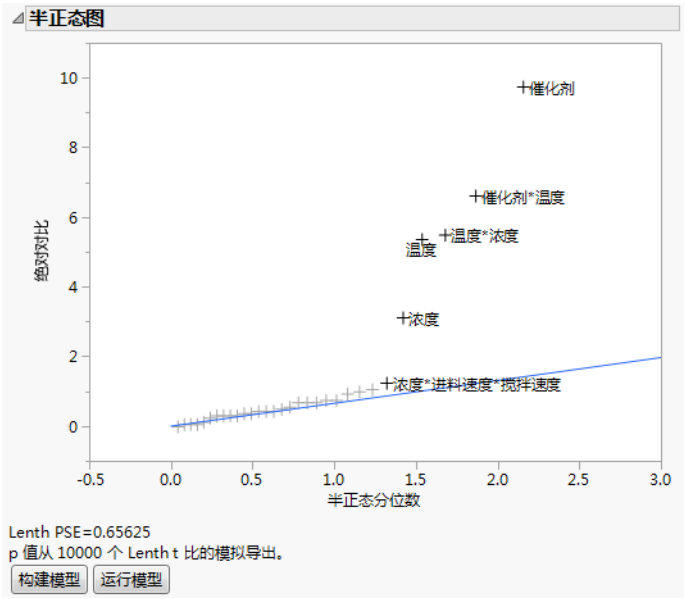
对比

项	对比	Lenth t 比	个体 p 值	联合 p 值
催化剂	9.75000	14.86	<.0001*	<.0001*
温度	5.37500	8.19	<.0001*	0.0005*
浓度	-3.12500	-4.76	0.0011*	0.0152*
进料速度	-0.68750	-1.05	0.2883	0.9998
搅拌速度	-0.31250	-0.48	0.6494	1.0000
催化剂*温度	6.62500	10.10	<.0001*	0.0002*
催化剂*浓度	1.00000	1.52	0.1335	0.9553
温度*浓度	-5.50000	-8.38	<.0001*	0.0004*
催化剂*进料速度	0.68750	1.05	0.2883	0.9998
温度*进料速度	-0.43750	-0.67	0.4941	1.0000
浓度*进料速度	0.06250	0.10	0.9266	1.0000
催化剂*搅拌速度	0.43750	0.67	0.5212	1.0000
温度*搅拌速度	1.06250	1.62	0.1139	0.9205
浓度*搅拌速度	0.43750	0.67	0.4941	1.0000
进料速度*搅拌速度	0.37500	0.57	0.5859	1.0000
催化剂*温度*浓度	-0.12500	-0.19	0.8557	1.0000
催化剂*温度*进料速度	0.68750	1.05	0.2883	0.9998
催化剂*浓度*进料速度	-0.93750	-1.43	0.1587	0.9784
温度*浓度*进料速度	0.31250	0.48	0.6494	1.0000
催化剂*温度*搅拌速度	0.56250	0.86	0.3772	1.0000
催化剂*浓度*搅拌速度	0.06250	0.10	0.9266	1.0000
温度*浓度*搅拌速度	0.06250	0.10	0.9266	1.0000
催化剂*进料速度*搅拌速度	0.75000	1.14	0.2478	0.9989
温度*进料速度*搅拌速度	-0.37500	-0.57	0.5859	1.0000
浓度*进料速度*搅拌速度	-1.25000	-1.90	0.0705	0.7592
催化剂*温度*浓度*进料速度	0.31250	0.48	0.6494	1.0000
催化剂*温度*浓度*搅拌速度	-0.31250	-0.48	0.6494	1.0000
催化剂*温度*进料速度*搅拌速度	0.00000	0.00	1.0000	1.0000
催化剂*浓度*进料速度*搅拌速度	0.75000	1.14	0.2478	0.9989
温度*浓度*进料速度*搅拌速度	0.50000	0.76	0.4328	1.0000
催化剂*温度*浓度*进料速度*搅拌速度	-0.25000	-0.38	0.7150	1.0000

注意：由于 “对比” 报表中的 *p* 值是使用 Monte Carlo 模拟获取的，所以您可能得不到图 12.4 中所示的相同值。请参见 ““拟合两水平筛选” 平台” 一章中第 315 页的 “Lenth 伪标准误差”。

“对比” 分级显示项中六个突出显示的效应已在 “半正态图” 中标记。

图 12.5 Reactor 32 Runs.jmp 的 “半正态图”



“半正态图”提供强有力的证据，表明至少五个标记的效应要比预期大（若这些效应是随机变异的结果）。这表明这些效应是活跃的。该图**并不能**明显地指出三因子**浓度 * 进料速度 * 搅拌速度**交互作用是活跃的。

在图 12.4 中的“对比”分级显示项中，三因子**浓度 * 进料速度 * 搅拌速度**交互作用的个体 p 值为 0.0705，其联合 p 值为 0.7592。由于该效应在“半正态图”中不突出并且其 *p* 值较大，您决定在模型中不包括该效应。

3. 在“半正态图”中，拖动某个矩形以选择除**浓度 * 进料速度 * 搅拌速度**之外的所有标记效应。
4. 点击**构建模型**以打开包含五个效应的 “拟合模型”窗口。
5. 点击**运行**。

“预测值 - 实际值”图显示：没有证据表明失拟，并且 “效应汇总”分级显示项显示全部五个效应都显著。

使用逐步回归进行分析

1. 返回 Reactor 32 Runs.jmp 数据表，或通过选择**帮助 > 样本数据库**并打开 Design Experiment/Reactor 32 Runs.jmp 重新打开该数据表。
2. 运行**模型**脚本。

“构造模型效应”列表仅包含到双因子交互作用；而您想要考虑所有交互作用。

3. 从 “选择列”列表中，从**进料速度**一直选择到**浓度**。
4. 点击**宏 > 完全析因**。

所有可能的效应都添加到 “构造模型效应” 列表中。

5. 将**特质**改为**逐步**。
6. 点击**运行**。
7. 将**停止规则**改为**最小 AICc**。

   对于设计的实验，AICc 优先于 BIC。这是因为 BIC 通常是比 AICc 更为宽松的停止规则，因为 BIC 往往允许在模型中包含不活跃效应。
8. 点击**执行**。

   逐步过程将六个效应选作可能活跃的效应。
9. 点击**运行模型**。

   这将使用这六个效应拟合模型。“效应汇总” 分级显示项指示**催化剂 * 浓度**在 0.05 显著性水平下不显著，因为 *p* 值为 0.0896。
10. 在 “效应汇总” 分级显示项中选择**催化剂 * 浓度**，然后点击**删除**。

    剩下的五个效应全部高度显著。这也是您使用 “筛选” 平台标识的五个效应 （第 343 页的 “使用 “筛选” 平台进行分析”）。

使用预测刻画器的最优设置

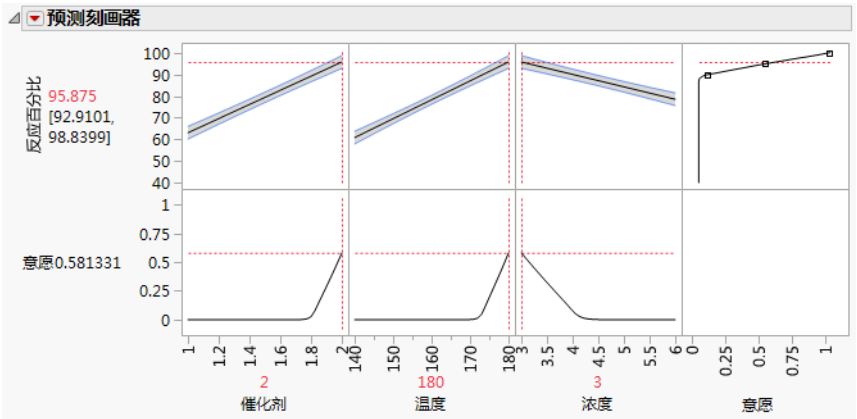
现在，找出您在模型中保留的五个显著效应所涉及的三个活跃因子的最优设置。

1. 在 Reactor 32 Runs.jmp 数据表中，运行**简化模型**脚本。

   “简化模型” 脚本打开您在第 343 页的 “使用 “筛选” 平台进行分析”和第 345 页的 “使用逐步回归进行分析”中标识的五效应模型的 “拟合模型” 窗口。
2. 点击**运行**。

   “预测刻画器” 报表显示 “意愿”，因为在 “完全析因” 窗口中，您在定义响应时指定的 “目标” 为 “最大化”。显示在刻画器顶行最右侧单元格中的意愿函数显示值为 100 最为理想，值为 90 或 90 以下最不理想。
3. 点击 “预测刻画器” 红色小三角菜单并选择**优化和意愿 > 最大化意愿**。

图 12.6 显示优化意愿的设置的预测刻画器

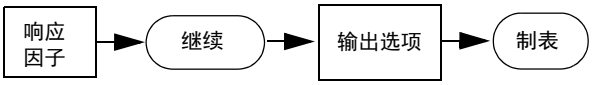


所显示设置下的预测均值**反应百分比**为 95.875，置信区间介于 92.91 到 98.84 之间。请注意，对于全部三个因子，标识的设置位于实验中使用的范围的极值处。在将来实验中，您应该探索这些设置之外的过程行为。

构建完全析因设计

通过选择**实验设计 > 经典 >完全析因设计**构建完全析因设计。首先定义响应和因子。下一步，继续执行输出选项。设计过程遵循图 12.7 中的流程。有关工作流的详细信息，请参见第 56 页的“实验设计工作流程：描述、指定、设计”。

图 12.7 完全析因设计流程

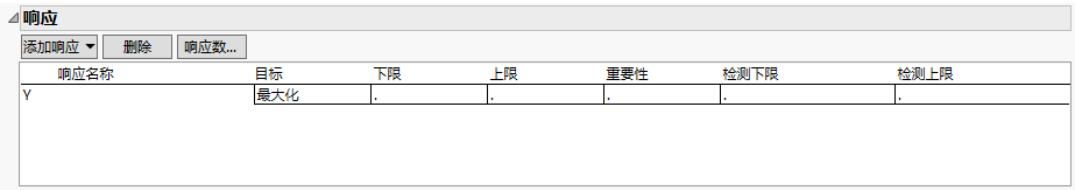


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 12.8 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的 “响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
- “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定“广义回归”平台中的删失响应。请参见《拟合线性模型》中的“广义回归”一章。

编辑“响应”分级显示项

在“响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

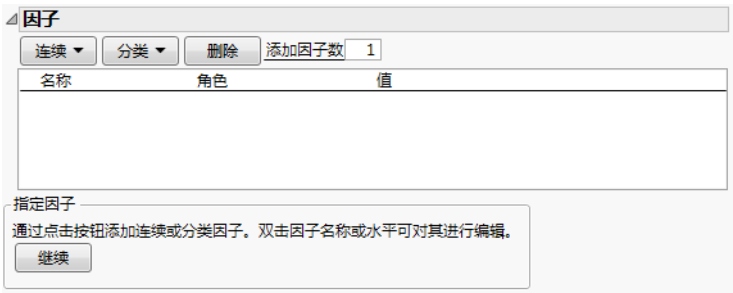
若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

完全析因设计中的因子可以是连续因子，也可以是分类因子。

提示：填写完“因子”分级显示项后，考虑从红色小三角菜单中选择**保存因子**。该选项会将因子名称、角色、更改和值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 12.9 “因子”分级显示项



连续型 添加一个“连续”因子。生成的数据表中的数据类型为“数值”。在符合过程和测量系统的限制条件下，若一个因子在概念上可以设置为您提供的下限与上限之间的任何值，则这样的因子为连续因子。

分类 添加一个“分类”因子。点击以选择或指定水平数。生成的数据表中的数据类型为“字符”。各水平的值排序就是从左到右输入值的顺序。在创建设计数据表后，该排序将保存在“值顺序”列属性中。

分类因子的默认值为 L1, L2, ..., Lk，其中，*k* 是您指定的水平数。用与实验相关的水平名称替换默认值。

删除 删除选定的因子。

添加因子数 添加多个因子。输入要添加的因子数，点击**添加因子**，然后选择因子类型。重复执行**添加因子数**，添加不同类型的多个因子。

“因子”分级显示项

“因子”分级显示项包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、X2 等。要更改该名称，双击它，然后输入所需名称。

角色 因子的“设计角色”。因子的“设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。因子的“角色”决定保存至数据表的其他因子属性。请参见第 350 页的“因子列属性”。

值 因子的实验设置。

编辑“因子”分级显示项

在“因子”分级显示项中，请注意以下事项：

- 要编辑某个因子名称，双击该因子名称。
- 要编辑某个值，在值列中点击该值。

因子列属性

对于每个因子，当您通过在“筛选设计”窗口中选择“制表”创建设计后，各种列属性都会保存至设计表。这些属性还会自动保存至选择“保存因子”选项后创建的数据表中。您可以在附录 A，“列属性”中找到关于这些列属性和相关示例的详细信息。

编码 若“角色”为“连续”，将保存因子的“编码”列属性。该属性变换因子值，以便低值与高值分别对应于 –1 和 +1。请参见“列属性”附录中第 709 页的“编码”。

值顺序 若“角色”为“分类”或构造了“区组”变量，则将保存因子的“值顺序”列属性。该属性确定因子水平的显示顺序。请参见“列属性”附录中第 723 页的“值顺序”。

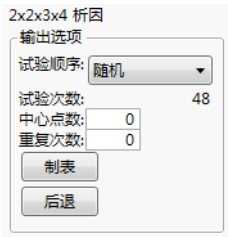
设计角色 每个因子都分配有“设计角色”列属性。在定义因子时指定的角色确定其“设计角色”列属性的值。“设计角色”属性反映在对实验数据建模时要如何使用因子。“设计角色”值用在“扩充设计”平台中。请参见“列属性”附录中第 706 页的“设计角色”。

因子更改 每个因子都分配有“因子更改”列属性，该属性的值为“容易”。“因子更改”属性反映在对实验数据建模时要如何使用因子。“因子更改”值用在“扩充设计”和“评估设计”平台中。请参见“列属性”附录中第 720 页的“因子更改”。

选择输出选项

输入响应和因子并点击**继续**后，您可以在“输出选项”分级显示项中为设计表做出选择。完全析因设计的结构显示在该分级显示项的顶部。

图 12.10 “输出选项”面板



试验顺序

试验顺序选项确定试验在设计表中的顺序。

保持相同 设计表中的行按从左至右排序。

从左至右排序 设计表中的列按从左至右排序。

随机 设计表中的行采用随机顺序。

从右至左排序 设计表中的列按从右至左排序。

中心点和重复

试验次数 在添加中心点或重复之前，显示设计中的试验次数。

中心点数 指定要作为设计的中心点额外添加的试验次数。中心点是每个连续因子都设置在因子范围中心位置的试验。

假定您的设计既包含连续因子又包含分类因子。若您在“输出选项”面板中请求中心点，这些中心点将按如下方式分配：

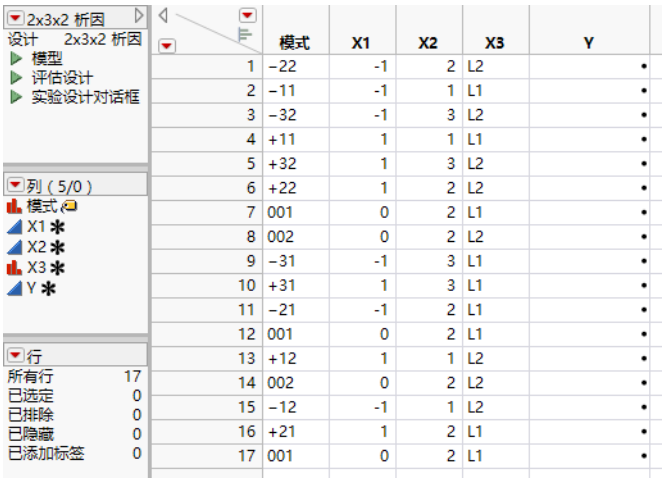
1. 使用在“因子”分级显示项中指定的值排序对分类因子的设置进行排序。
2. 每一个中心点都按顺序分配至分类因子设置的每种组合，重复分配过程，直到所有中心点都分配完毕。

重复次数 要重复整个设计 （包括中心点）的次数。一次重复会使试验次数增加一倍。

制表

点击“制表”创建包含实验中的试验的数据表。图 12.11 中的示例显示了有五个中心点的一个完全析因设计，其中的三个因子为 X1（两水平连续因子）、X2（三水平连续因子）和 X3（两水平分类因子）。该设计使用因子水平的默认值。中心点位于行 7、8、12、14 和 17 中。请参见第 352 页的“ “模式”列”。

图 12.11 设计数据表



	模式	X1	X2	X3	Y
1	−22	-1	2	L2	•
2	−11	-1	1	L1	•
3	−32	-1	3	L2	•
4	+11	1	1	L1	•
5	+32	1	3	L2	•
6	+22	1	2	L2	•
7	001	0	2	L1	•
8	002	0	2	L2	•
9	−31	-1	3	L1	•
10	+31	1	3	L1	•
11	−21	-1	2	L1	•
12	001	0	2	L1	•
13	+12	1	1	L2	•
14	002	0	2	L2	•
15	−12	-1	1	L2	•
16	+21	1	2	L1	•
17	001	0	2	L1	•

表名（显示在左上角）是生成该表的设计类型。

设计表脚本

设计表包括以下脚本：

模型 运行 “分析” > “拟合模型”平台。

评估设计 运行 “实验设计” > “设计诊断” > “评估设计”平台。

实验设计对话框 重新创建您用于生成设计表的 “完全析因设计”窗口。该脚本还包含用于生成设计的随机种子。

运行**筛选**或**模型**脚本以分析数据。

“模式”列

模式列包含在给定行中汇总试验的条目。您可以将**模式**用作图中的标签变量。

- 对于两水平连续因子，低设置用“−”表示，高设置用“+”表示，中心点用“0”表示。
- 对于具有两个以上水平的连续因子：

  - 对于非中心点，因子设置用对应于该试验的值水平的整数表示。
  - 对于中心点，因子设置用“0”表示。
- 对于分类因子，因子设置用对应于该试验的值水平的整数表示。

完全析因设计选项

“完全析因设计”平台中的红色小三角菜单包含以下选项：

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用“保存响应”选项保存的响应。

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在“设计角色”区域中，选择合适的角色。

加载因子 加载使用“保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在“定义因子约束”或“线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为“小于”或“大于”约束。请参见“列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用“保存约束”选项保存的因子约束。

设置随机种子 设置随机种子，JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

  - 初始化用于生成设计的搜索算法
  - 随机化用于构造设计的“试验顺序”
  - 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的“实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击“制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示“模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一：“正态”、“二项式”或“Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在“模型”窗口中做出选择以控制模拟的响应值的分布。当您点击“应用”后，模拟的响应值的公式将保存在名为 <Y> **模拟**的新列中，其中，Y 是响应的名称。再次点击“应用”将更新 <Y> **模拟**中的公式和值。

有关其他详细信息，请参见“定制设计”一章中第 101 页的 “模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见《基本分析》中的“模拟”一章。

高级选项 无。

将脚本保存至脚本窗口 为您在“完全析因设计”窗口中指定的设计创建脚本，并将其保存到打开的脚本窗口中。

第 13 章

混料设计

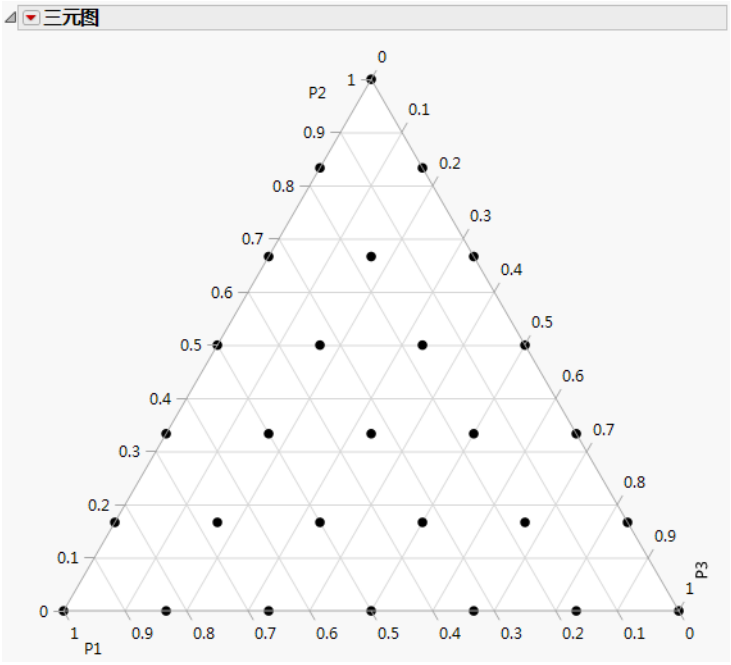
配方设计

使用“混料设计”平台来构建是混料成分的因子实验。您可以在若干经典混料设计方法中进行选择，例如单纯形、极端顶点和格点。对于极端顶点方法，您可以提供一组线性不等式约束，用来限制混料因子空间的几何图形。

在混料实验中，因子的值是其在混料中的比例，介于 0 到 1 之间。混料实验具有三个或更多因子，这些因子比例之和等于 1 (100%)。混料实验不同于其他实验类型：您不能独立更改某个因子。当您更改一个因子的比例时，一个或多个因子的比例也必须随之更改。这一简单的事实对混料实验的方方面面（因子空间、设计属性和结果解释）都产生了深远的影响。

因为混料成分的比例之和为 1，混料设计的可行域采用单纯形形式。例如，使用三角形状的二维图形可视化三因子设计，使用三维四面体可视化四因子设计。

图 13.1 三元图



目录

混料设计概述 357
混料设计窗口 357
响应 357
因子 359
三元图概述 360
混料设计类型的示例 361
最优混料设计 361
单纯形重心设计 363
单纯形格点设计 365
极端顶点设计 366
ABCD 设计 370
空间填充设计 370
拟合混料设计 373
整体模型检验和方差分析报表 374
理解响应曲面报表 374
化学混料示例 374
创建设计 374
分析混料模型 377
预测刻画器 377
混料刻画器 378

混料设计概述

您可以从以下混料设计类型中进行选择：

最优设计 使用定制设计器生成最优混料设计。请参见第 361 页的 “最优混料设计”。

单纯形重心设计 （可用于无约束因子或具有下限约束的因子。）生成纯混料的设计，混合因子最多到指定程度。在纯混料中，一个成分为 100%。所有其他成分为 0%。请参见第 363 页的 “单纯形重心设计”。

单纯形格点设计 （可用于无约束因子或具有下限约束的因子。）生成纯混料的设计，在试验的空间填充三角形网格中混合。您可以指定网格水平数。请参见第 365 页的 “单纯形格点设计”。

极端顶点设计 （当对一个或多个因子进行线性约束或指定受限制的上限或下限时，可使用该设计。）根据约束因子空间的顶点和顶点组合生成设计。请参见第 366 页的 “极端顶点设计”。

ABCD 设计 生成 Snee (1975) 提出的混料筛选设计。请参见第 370 页的 “ABCD 设计”。

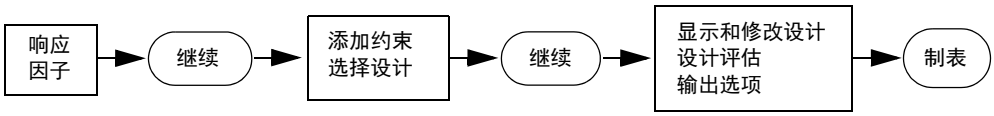
空间填充 生成适用于线性约束的空间填充设计。请参见第 370 页的 “空间填充设计”。

注意：最优设计、极端顶点设计和空间填充设计可以纳入成分的线性约束和限制。

混料设计窗口

随着您执行的设计步骤不同，混料设计窗口将相应更新。请参见第 56 页的 “实验设计工作流程：描述、指定、设计”。工作流程和分级显示项随设计类型而异。

图 13.2 混料设计工作流程

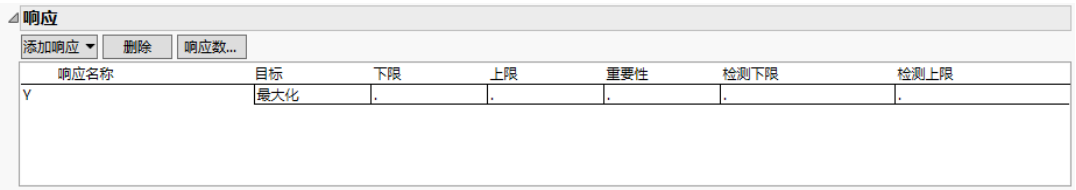


响应

使用 “响应” 分级显示项来指定一个或多个响应。

提示：填写完 “响应” 分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 13.3 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的 “响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
- “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定 “广义回归”平台中的删失响应。请参见 《拟合线性模型》中的 “广义回归”一章。

编辑“响应”分级显示项

在 “响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

在 “因子”分级显示项中添加因子。

提示：填写完“因子”分级显示项后，考虑从红色小三角菜单中选择**保存因子**。这会将因子名称、角色、更改和值保存到一个数据表中，以便将来重新加载。

图 13.4 “因子”分级显示项

添加 输入要添加的因子数并点击 “添加”。

删除选定项 删除选定的因子。

“因子”列表

“因子”列表包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、 X2 等。要更改该名称，双击它，然后输入所需名称。

角色 指定因子的设计角色。因子的 “设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。

值 因子的实验设置。要插入值，点击默认值并输入所需值。

编辑“因子”列表

在 “因子”列表中，执行以下操作：

- 要编辑某个因子名称，双击该因子名称。
- 要编辑某个值，在**值**列中点击该值。

线性约束

点击**线性约束**按钮可输入一个或多个线性不等式约束。将显示一个模板，它用于指定涉及设计中所有连续因子的线性表达式。输入因子的系数值并选择不等式的方向来反映您的线性约束。在不等式右侧的框中指定约束值。要添加更多约束，请再次点击**线性约束**。

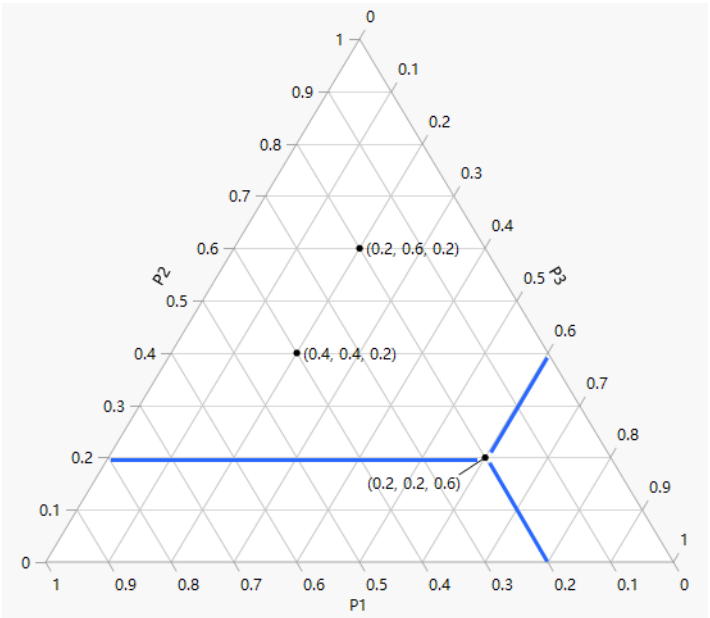
注意：保存涉及线性约束的设计的脚本时，该脚本将线性约束表示为**小于等于**不等式 (≤)。

三元图概述

三元图是总和为常数的三个混料成分的二维表示。该图为一个等边三角形，每个成分对应一个边。无约束时，三角形的每个顶点对应于纯混合：一个成分为 1 (100%)，所有其他成分为 0。成分被约束时，用三元图的比例表示可行的混料。使用着色排除图的不可行部分。

图 13.5 显示三个成分 P1、P2 和 P3。这三个成分无约束，每个成分所占比例可为从 0 到 1 (100%) 的范围。使用其坐标 (P1, P2, P3) 标记三个点。一个点 (0.2, 0.2, 0.6) 包含沿着每个混料成分的轴网格线的蓝线，引导您读取这些轴。

图 13.5 三元图示例



混料设计类型的示例

- “最优混料设计”
- “单纯形重心设计”
- “单纯形格点设计”
- “极端顶点设计”
- “ABCD 设计”
- “空间填充设计”

最优混料设计

可以使用“混料设计”平台或“定制设计”平台生成最优混料设计。

在此您使用 “混料”平台使用三个因子创建最优混料设计示例：

1. 选择**实验设计** > **经典** > **混料设计**。
2. 在该示例中使用三个因子。无需对 “因子”部分做出更改。
3. 点击**继续**。
4. 在 “选择混料设计类型”面板上点击**最优设计**。

5. （可选）要匹配该示例的输出，请点击“混料设计”红色小三角并选择“设置随机种子”，然后输入 1409。
6. （可选）要匹配该示例的输出，请点击“混料设计”红色小三角菜单并选择“开始数”，然后输入 2。
7. 在**定义因子约束**面板中，点击**指定线性约束**，然后点击**添加**。
8. 在 X1 和 X2 框中输入 1，在约束框中输入 0.8。

   这将约束 X1+X2 小于混料的 80%。
9. 在**模型**面板中，点击**交互作用 > 二次**。

   这将交互作用效应添加到模型。
10. 在**生成设计**面板中，对于**中心点数**输入 2。
11. 点击**制作设计**。

图 13.6 12 次试验的最优设计

设计

试验	X1	X2	X3
1	0.8	0	0.2
2	0	0	1
3	0	0.8	0.2
4	0	0	1
5	0	0.4	0.6
6	0.4	0	0.6
7	0	0.8	0.2
8	0.4	0	0.6
9	0.4	0.4	0.2
10	0.8	0	0.2
11	0.264183	0.269687	0.46613
12	0.264183	0.269687	0.46613

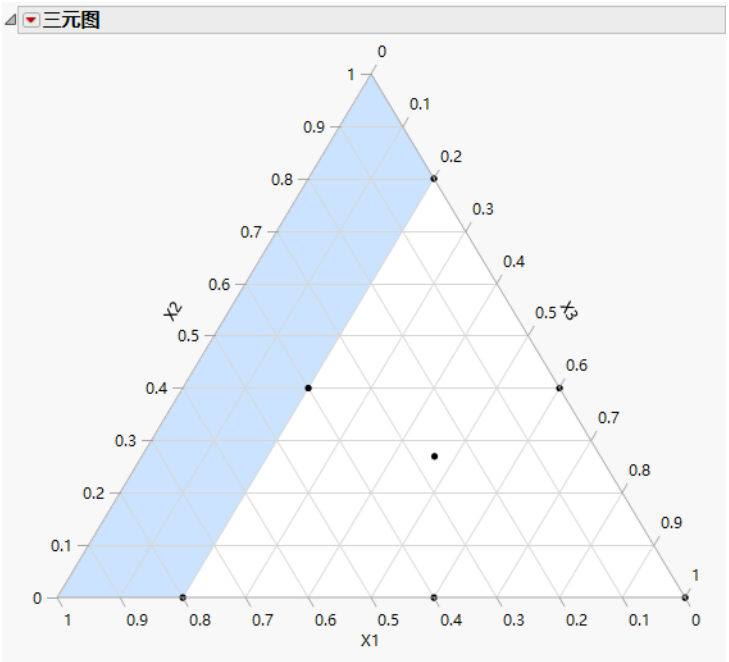
12. 点击**制表**。

注意：实际上，您会将混料值舍入为合适的精度级别。

直观演示设计

1. 从设计表中，选择**图形 > 三元图**。
2. 选择 X1、 X2 和 X3 并点击 **X，绘图**，然后点击**确定**。

图 13.7 最优设计的三元图



该三元图显示了构成设计的 7 个独特混料。着色指示受约束的区域。有关三元图的详细信息，请参见第 360 页的“三元图概述”。

单纯形重心设计

包含 n 个因子且阶数为 k 的单纯形重心设计由具有以下特征的混料组成：

- 一个因子的混料或纯混合
- 两个因子按相同水平混合
- 其他混合，最多 k 个因子在某一时间按 k 个相等水平混合

当 k 等于因子数时，将包含一个所有成分都等量的中心点试验。

注意：该设计可用于一个或多个因子的下限。

创建设计

要创建示例单纯形重心设计，请执行以下步骤：

1. 选择**实验设计 > 经典 > 混料设计**。
2. 在该示例中使用三个因子。无需对“因子”部分做出更改。

3. 点击**继续**。
4. 点击**单纯形重心设计**以生成设计。

注意：使用 $k = 2$ 的默认设置。有一个文本框，用于调整 k 的值。

图 13.8 阶数为 2 的三因子单纯形重心设计

设计

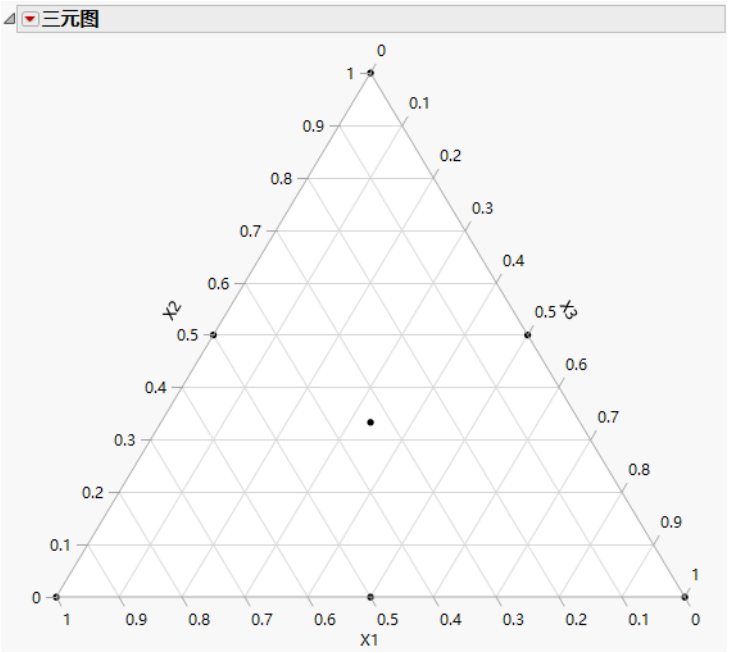
试验	X1	X2	X3
1	1.00000	0.00000	0.00000
2	0.00000	1.00000	0.00000
3	0.00000	0.00000	1.00000
4	0.50000	0.50000	0.00000
5	0.50000	0.00000	0.50000
6	0.00000	0.50000	0.50000
7	0.33333	0.33333	0.33333

5. 点击**制表**。

直观演示设计

6. 从设计表中，选择**图形 > 三元图**。
7. 选择 X1、 X2 和 X3 并点击 **X，绘图**，然后点击**确定**。

图 13.9 单纯形重心设计的三元图



有关三元图的详细信息，请参见第 360 页的“三元图概述”。

单纯形格点设计

单纯形格点设计是一种空间填充设计，它可以创建三角形网格形式的试验。该设计是混合集，其中因子的值是 i/m，其中 i 的范围为 0 到 m，因子和为 1。对于三水平设计，可以在 0、0.33、0.66 或 1 处设置因子。

注意：该设计可用于一个或多个因子的下限。

要创建示例单纯形格点设计，请执行以下步骤：

1. 选择**实验设计 > 经典 > 混料设计**。
2. 在该示例中使用三个因子。无需对 “因子”部分做出更改。
3. 点击**继续**。
4. 点击**单纯形格点设计**以生成设计。

   注意：使用了 5 个水平的默认设置。有一个文本框，用于调整水平数。

图 13.10 三因子五水平的单纯形格点设计

设计

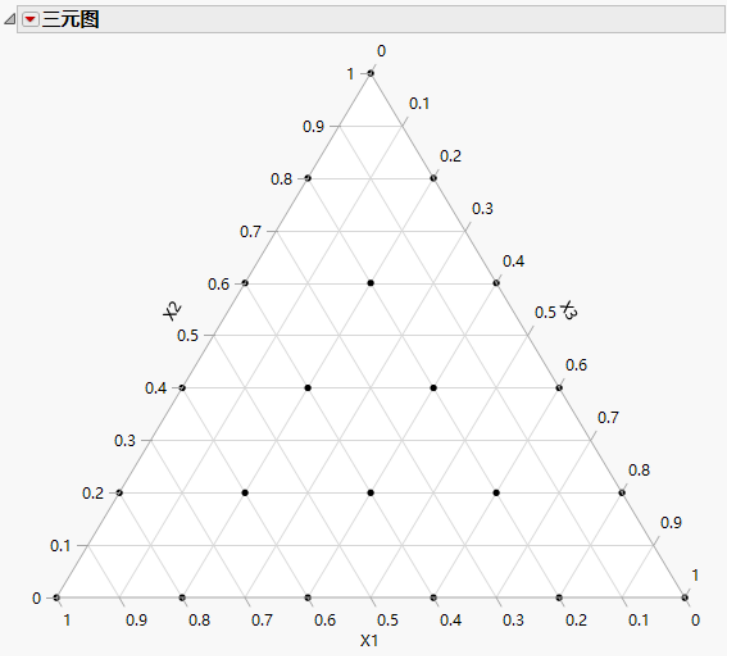
试验	X1	X2	X3
1	0.00000	0.00000	1.00000
2	0.00000	0.20000	0.80000
3	0.00000	0.40000	0.60000
4	0.00000	0.60000	0.40000
5	0.00000	0.80000	0.20000
6	0.00000	1.00000	0.00000
7	0.20000	0.00000	0.80000
8	0.20000	0.20000	0.60000
9	0.20000	0.40000	0.40000
10	0.20000	0.60000	0.20000
11	0.20000	0.80000	0.00000
12	0.40000	0.00000	0.60000
13	0.40000	0.20000	0.40000
14	0.40000	0.40000	0.20000
15	0.40000	0.60000	0.00000
16	0.60000	0.00000	0.40000
17	0.60000	0.20000	0.20000
18	0.60000	0.40000	0.00000
19	0.80000	0.00000	0.20000
20	0.80000	0.20000	0.00000
21	1.00000	0.00000	0.00000

5. 点击**制表**。

直观演示设计

6. 从设计表中，选择**图形 > 三元图**。
7. 选择 X1、 X2 和 X3 并点击 **X，绘图**，然后点击**确定**。

图 13.11 单纯形格点设计的三元图



有关三元图的详细信息，请参见第 360 页的“三元图概述”。

随着水平数的增加，设计点数会增加，减少了设计点之间的空间。通过与单纯形重心设计对比发现，单纯形格点设计不一定包含重心。

极端顶点设计

极端顶点设计需要对一个或多个因子的范围进行限制，这由“因子”面板中的限值或线性约束来指定。极端顶点设计包含因子空间顶点处的混料和指定次数的顶点平均值处的混料。由一条线连接的两个邻近顶点的重心点为二次重心。共享一个平面的顶点的重心点为三次重心。

包含范围约束的极端顶点示例

要创建一个示例极端顶点设计，请执行以下步骤：

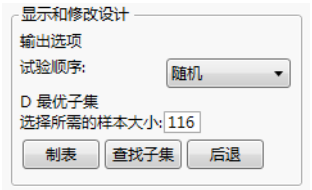
1. 选择**实验设计 > 经典 > 混料设计**。
2. 在“因子”下，输入 2 作为要添加的其他因子数，然后点击**添加**。
3. 如图 13.12 中所示设置五个因子的范围，然后点击**继续**。

图 13.12 五个因子的范围

名称	角色	值	
X1	混料	0.05	0.25
X2	混料	0.1	0.3
X3	混料	0.1	0.3
X4	混料	0.1	0.4
X5	混料	0.05	0.25

4. 在**次数**文本框中输入 4，然后点击**极端顶点设计**。

图 13.13 极端顶点设计示例的 “显示和修改” 面板



这个五个因子、次数为 4 的极端顶点设计的默认试验次数为 116。这是在给定范围约束的条件下五个因子的极端顶点数和次数为 4 的平均数。 JMP 使用混料的默认集作为候选集来生成更小的 D 最优设计。

5. （可选）要匹配该示例的输出，请点击 “混料设计” 红色小三角并选择 “设置随机种子”，然后输入 1409。
6. 在 “选择所需的样本大小” 框中输入 10，然后点击**查找子集**来生成设计。

注意：查找子集选项使用行交换方法 （不是坐标交换）来查找最佳行子集。

图 13.14 试验次数为 10 的最优极端顶点设计

设计

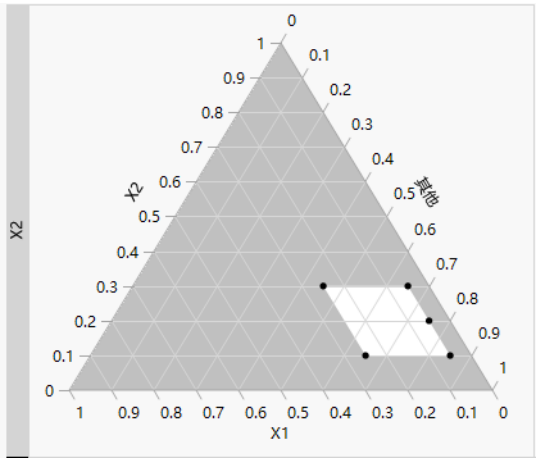
试验	X1	X2	X3	X4	X5
1	0.05000	0.30000	0.20000	0.40000	0.05000
2	0.05000	0.10000	0.30000	0.40000	0.15000
3	0.25000	0.30000	0.10000	0.10000	0.25000
4	0.05000	0.30000	0.30000	0.10000	0.25000
5	0.25000	0.10000	0.30000	0.10000	0.25000
6	0.25000	0.30000	0.30000	0.10000	0.05000
7	0.25000	0.10000	0.30000	0.30000	0.05000
8	0.05000	0.20000	0.10000	0.40000	0.25000
9	0.25000	0.30000	0.10000	0.30000	0.05000
10	0.25000	0.10000	0.10000	0.30000	0.25000

7. 点击**制表**。

直观演示设计

8. 从设计表中，选择**图形 > 三元图**。
9. 选择 X1、 X2、 X3、 X4 和 X5，点击 **X，绘图**，然后点击**确定**。

图 13.15 五因子设计的三元图的部分输出



当您有三个以上的混料成分时，三元图显示一系列图。每个图具有两个成分轴，第三个轴是所有其他成分之和。若您有受约束的区域，则将这些图着色。未着色的部分为可行域。有关三元图的详细信息，请参见第 360 页的“三元图概述”。

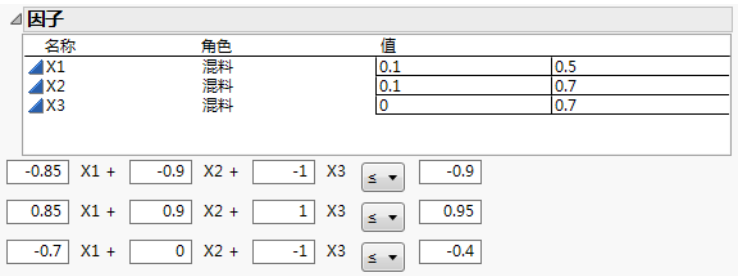
包含线性约束的极端顶点示例

考虑 Snee (1979) 和 Piepel (1988) 提供的经典示例。该示例具有三个因子（X1、X2 和 X3），且具有因子边界和三个线性约束。

要为该示例创建极端顶点设计，请执行以下步骤：

1. 选择**实验设计 > 经典 > 混料设计**。
2. 为 X1、 X2 和 X3 输入图 13.16 中的值，然后点击**继续**。

图 13.16 Snee 和 Piepel 示例的值和线性约束

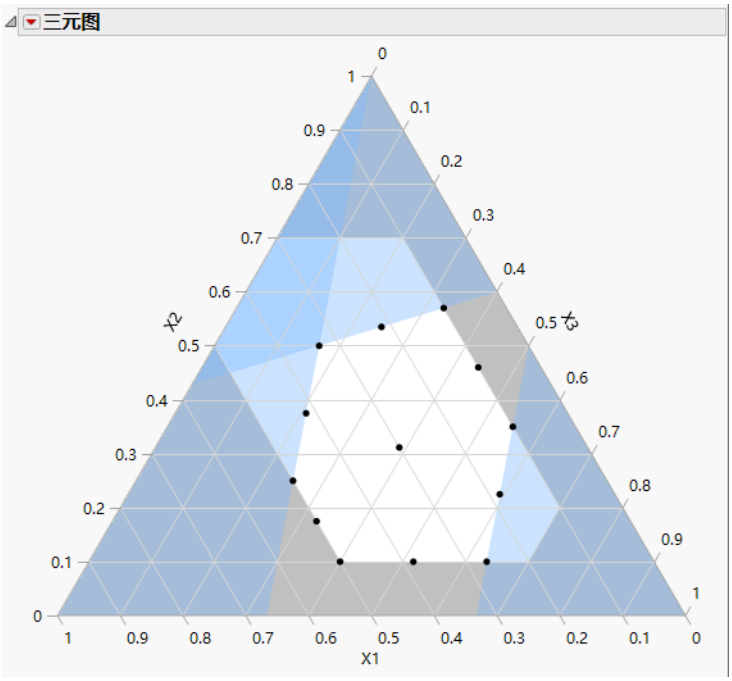


3. 点击三次**线性约束**。输入图 13.16 中所示的约束。
4. 点击**极端顶点设计**按钮。
5. 点击**制表**。

直观演示设计

6. 从设计表中，选择**图形 > 三元图**。
7. 选择 X1、 X2 和 X3 并点击 **X，绘图**，然后点击**确定**。

图 13.17 显示带有约束的 Piepel 示例的三元图



三元图显示因子限值和线性约束所定义的可行域。设计点位于可行域的六个顶点、六边的中点和整体重心。有关三元图的详细信息，请参见第 360 页的“三元图概述”。

极端顶点方法的统计详细信息

若存在线性约束，JMP 使用 R.E. Wheeler 开发的 CONSIM 算法，该算法在 Snee (1979) 中进行了说明，由 Piepel (1988) 作为 CONVRT 提供。该方法在 Cornell (1990, Appendix 10a) 中也有说明。方法中结合了约束和检查，以查看顶点是否与它们冲突。若冲突，它会删除顶点并计算新顶点。用来计算重心点的名为 CONAEV 的方法是由 Piepel (1988) 提出的。

若不存在线性约束（只有范围约束），则会使用 Snee and Marquardt (1974) 以及 Snee (1975) 开发的 XVERT 方法构造极端顶点设计。找到顶点之后，单纯形重心设计方法会生成直到指定阶数的顶点组合。

XVERT 方法首先使用具有最小范围的 $nf-1$ 个因子的给定低值和高值创建完整的 2^{nf-1} 设计。然后，它根据因子值总和必须为 1 的限制计算剔除的一个因子的值。它保留不在因子范围之内

的点。若不在范围之内，该方法会增大或减小该值使其位于范围之内，并且相应地让其他每个因子的值减小或增大相同的量。该方法使这些点仍满足最初的限制。

上述算法可在因子约束定义的单纯形中创建可行域的顶点。不过，Snee (1975) 已证明具有可行域的边和面的重心也会很有用。可行域的广义 n 维面通过 $nf - n$ 条边界来定义，而面的重心通过落在面上的顶点的平均值来定义。该算法生成边界条件的所有可能组合，然后对第一步生成的顶点取平均值。

ABCD 设计

ABCD 设计是混料的筛选设计。请参见 Snee (1975)。

要创建一个示例 ABCD 设计，请执行以下步骤：

1. 选择**实验设计 > 经典 > 混料设计**。
2. 在 “因子”下，输入 3 作为要添加的其他因子数，然后点击**添加**。
3. 点击**继续**。
4. 点击 **ABCD 设计**。

图 13.18 6 因子 ABCD 设计的 14 次试验 （一共 28 次）

设计

试验	X1	X2	X3	X4	X5	X6
1	1.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
3	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000
4	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000
5	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000
6	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000
7	0.50000	0.50000	0.00000	0.00000	0.00000	0.00000
8	0.50000	0.00000	0.50000	0.00000	0.00000	0.00000
9	0.50000	0.00000	0.00000	0.50000	0.00000	0.00000
10	0.50000	0.00000	0.00000	0.00000	0.50000	0.00000
11	0.50000	0.00000	0.00000	0.00000	0.00000	0.50000
12	0.00000	0.50000	0.50000	0.00000	0.00000	0.00000
13	0.00000	0.50000	0.00000	0.50000	0.00000	0.00000
14	0.00000	0.50000	0.00000	0.00000	0.50000	0.00000

5. 点击**制表**。

空间填充设计

“空间填充”混料设计将设计点分布在整个设计区域中。该设计支持线性约束。该设计采用类似于“实验设计”>“特殊目的”>“空间填充设计”下的“快速灵活填充”设计方法方式生成（第 569 页的“快速灵活填充设计”）。

两个 “混料设计”红色小三角选项与空间填充设计相关：

FFF 最优性准则 对于 “快速灵活填充”混料设计类型，支持您选择 MaxPro 准则 （默认值）或重心法准则。请参见第 372 页的 “FFF 最优性准则”。

高级选项 > 设置平均聚类大小 对于 “快速灵活填充”混料设计类型，支持您指定用于定义每个聚类（或等效的每个设计点）的随机生成点的平均数。请参见第 373 页的 “设置平均聚类大小”。

空间填充示例

要创建一个示例空间填充设计，请执行以下步骤：

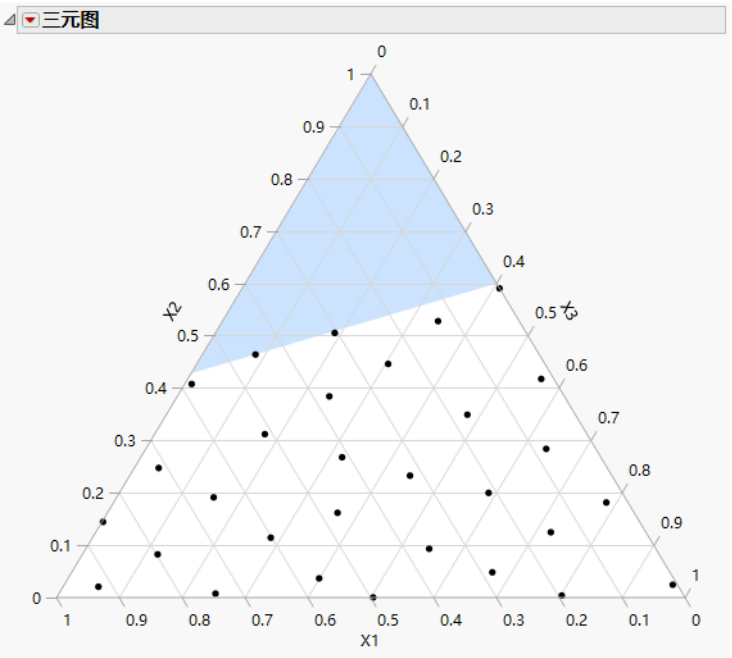
1. 选择**实验设计 > 经典 > 混料设计**。
2. （可选）要匹配该示例的输出，请点击 “混料设计”红色小三角菜单，然后选择 “设置随机种子”并输入 1409。
3. 在该示例中使用三个因子。无需对 “因子”部分做出更改。
4. 点击**继续**。
5. 点击**线性约束**。
6. 在 X1 框中输入 0.7，在 X3 框中输入 1，将不等式方向更改为 ≥ 并在约束框中输入 0.4。

   这将 0.7*X1*+*X3* 约束为混料的至少 40%。
7. 在**试验数**框中输入 30。
8. 点击**空间填充**。
9. 点击**制表**。

直观演示设计

10. 从设计表中，选择**图形 > 三元图**。
11. 选择 X1、 X2 和 X3 并点击 **X，绘图**，然后点击**确定**。

图 13.19 带有一个线性约束的空间填充设计



使用重心 FFF 最优性准则构造该设计。请注意，点落在约束设计区域中，并且在整个约束设计区域中分布较为均匀。有关三元图的详细信息，请参见第 360 页的“三元图概述”。

混料空间填充设计的统计详细信息

FFF 最优性准则

“快速灵活填充”设计的算法最初会在指定设计区域内生成大量随机点。随后使用快速 Ward 算法将这些点聚类到个数与您指定的“试验次数”相同的聚类中。

可使用默认 MaxPro（**最大投影**）最优性准则或通过选择重心法准则来获取最终设计点。您可以在该报表的红色小三角菜单中的“FFF 最优性准则”下找到这些选项。

MaxPro 对于 p 个因子以及 n 等于指定“试验次数”的情况，MaxPro 准则尝试查找聚类中可最小化以下准则的点：

$$C_{MaxPro} = \sum_{i}^{n-1} \sum_{j=i+1}^{n} \left[1 / \prod_{k=1}^{p} (x_{ik} - x_{jk})^2 \right]$$

MaxPro 准则可在考虑所有因子的前提下，令可能设计点之间的距离乘积最大化。这样即可支持为因子投影提供良好的空间填充属性的目标。请参见 Joseph et al. (2015)。 Max Pro 选项是默认选项。

重心法 该方法在每个聚类的重心放置一个设计点。它具有这样的属性：从设计空间中的任意点到其最近邻近设计点的平均距离比其他设计都要小。

注意：您可以设置首选项，以便始终使用给定的最优性准则。选择**文件 > 首选项 > 平台 > 实验设计**。选择 FFF 最优性准则并选择您首选的准则。

设置平均聚类大小

设置平均聚类大小选项位于“混料设计”红色小三角菜单的“高级选项”下方。该选项支持您指定用来定义每个聚类（或等效的每个设计点）的均匀生成点的平均数。

默认情况下，若“空间填充”设计类型的试验次数为 200 或更小，则聚类算法以总共 10,000 个随机均匀生成的点作为基础。当试验次数超过 200 时，默认值为 50。增大该值尤其适用于因子数量较多的设计。

注意：根据因子数和为试验指定的值，您可能需要通过选择**高级选项 > 设置平均聚类大小**增大每个设计点的初始点平均数。

线性约束

可以通过选择“线性约束”分级显示项中的**线性约束**选项限定设计区域。

当您指定线性约束时，构成聚类算法基础的随机点在受约束的设计区域内随机分布。聚类算法使用这些点。

拟合混料设计

拟合混料设计的模型时，必须考虑因子之和为常数。传统完全线性模型不是完全可估的。

混料响应的合适响应曲面模型是 Scheffé 多项式 (Scheffé 1958)。请参见 《拟合线性模型》中的关于 “Cox 混料”和 “Scheffé 三次项”宏的讨论。 Scheffé 多项式模型执行以下任务：

- 禁用截距
- 包括所有线性主效应项
- 排除所有平方项（如 X1*X1）
- 包括所有交叉项（如 X1*X2）

在该模型中，参数很容易解释 (Cornell 1990)。线性项的系数是极点处（对应混料由单个因子构成）的拟合响应。交叉项的系数指示跨越因子空间每边的曲率。

对于具有三阶多项式项的混料模型，可以使用 Scheffé 三次项模型。Scheffé 三次项模型包含 X1*X2*(X1-X2) 形式的项。

- 要拟合 Scheffé 多项式模型，请使用“拟合模型”平台中的“混料响应曲面”宏。

- 要拟合 Scheffé 三次项模型，请使用“拟合模型”平台中的 Scheffe 三次项宏。
- 若您选择拟合另一混料模型，“拟合模型”平台将包含“无截距”模型的选项，将“混料效应”作为效应特性。

提示：定制设计模型分级显示项具有添加“Scheffé 三次项”模型项的选项。生成的设计表将包含“Scheffé 三次项”模型的脚本。

整体模型检验和方差分析报表

在整体模型的“方差分析”表中，JMP 通常会检验除截距之外的所有参数是否为零。在不含截距的混料模型中，JMP 会查找隐藏截距（其意义为效应的线性组合是一个常数）。若 JMP 找到隐藏截距，它会针对截距模型而不是零截距模型执行整体模型检验。该检验等同于检验除线性参数之外的其他所有参数是否为零，并检验这些参数是否相等。

隐藏截距属性还会导致针对截距模型报告 R^2，而不是报告为缺失。

理解响应曲面报表

若有效应标记为响应曲面效应“&RS”，JMP 会额外创建报表来分析拟合响应曲面。这些报表最初是为完全响应曲面而不是混料模型设计的。不过，JMP 可能遇到无截距模型而找到包含线性响应曲面项（而不是平方项）的隐藏截距。那么 JMP 会**折叠**其计算，在最后一个响应曲面项上折叠以计算最佳值的临界值。可以为生成常数并涉及最后一个响应曲面项的任意组合执行该操作。

化学混料示例

有三种增塑剂（p1、p2 和 p3）构成了用于制作汽车座罩的乙烯基中 79.5% 的成分 (Cornell, 1990)。在这 79.5% 中，各增塑剂分别受到以下约束的限制：$0.409 \leqslant x1 \leqslant 0.849$、$0 \leqslant x2 \leqslant 0.252$ 且 $0.151 \leqslant x3 \leqslant 0.274$。Cornell 使用 14 次试验设计来对收集的响应进行二次混料模型拟合。该设计为 3 次极端顶点设计，其中包含四个顶点和整体重心的重复测量值。

- “创建设计”
- “分析混料模型”
- “预测刻画器”
- “混料刻画器”

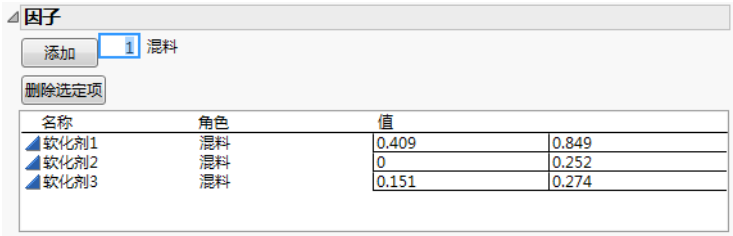
创建设计

在 JMP 中创建 Cornell 的混料设计

1. 选择**帮助 > 样本数据库**，然后打开 Plastifactors.jmp。

2. 选择**实验设计 > 经典 > 混料设计**。
3. 点击“混料设计”红色小三角菜单并选择**加载因子**。

图 13.20 增塑剂实验的因子和因子约束



4. 点击**继续**。
5. 在**次数**框中输入 3。
6. 点击**极端顶点设计**。
7. 点击**制表**。 JMP 使用 9 个因子设置来生成 JMP 表。

图 13.21 极端顶点混料设计

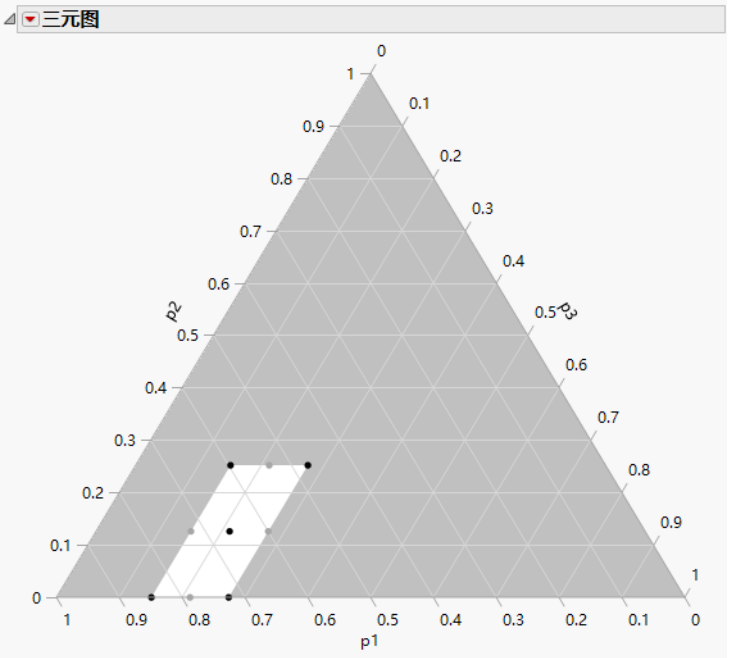
	软化剂1	软化剂2	软化剂3	Y
1	0.597	0.252	0.151	•
2	0.726	0	0.274	•
3	0.6615	0.126	0.2125	•
4	0.6	0.126	0.274	•
5	0.7875	0	0.2125	•
6	0.849	0	0.151	•
7	0.5355	0.252	0.2125	•
8	0.474	0.252	0.274	•
9	0.723	0.126	0.151	•

注意：您的表可能因用于生成设计的随机种子而有所不同。

接下来，您通过复制顶点和整体重心增加了额外的 5 次设计试验，以生成 14 次试验的设计。

8. 为了确定顶点和重心，使用了三元图。从设计表中，选择**图形 > 三元图**。
9. 选择 p1、 p2 和 p3 并点击 **X，绘图**，然后点击**确定**。

图 13.22 设计的三元图



10. 突出显示顶点和重心点。

图 13.23 确定顶点和中心点

	软化剂1	软化剂2	软化剂3	Y
1	0.597	0.252	0.151	•
2	0.726	0	0.274	•
3	0.6615	0.126	0.2125	•
4	0.6	0.126	0.274	•
5	0.7875	0	0.2125	•
6	0.849	0	0.151	•
7	0.5355	0.252	0.2125	•
8	0.474	0.252	0.274	•
9	0.723	0.126	0.151	•

11. 选择**编辑 > 复制**，将选定行复制到剪贴板。
12. 点击第 10 行的第一个单元格，然后选择**编辑 > 粘贴**以将复制的行添加到表。

注意：可使用定制设计器（二次模型、2 个中心点、4 个重复设计和 14 次试验）得到类似的设计。

增塑剂设计和 Cornell 得到的结果（*Y* 值）在样本数据文件 Plasticizer.jmp 中提供。

分析混料模型

为了分析增塑剂实验，使用了样本数据表 Plasticizer.jmp。

1. 选择**帮助** > **样本数据库**，然后打开 Plasticizer.jmp。
2. 点击数据表左上角的**模型**脚本旁边的绿色小三角。

   完整的 “拟合模型”启动窗口随即显示。

注意：当您创建设计时，将模型脚本保存到设计表。

3. 点击**运行**以查看响应曲面分析。
4. Plasticizer.jmp 包含名为**预测公式 Y** 的列。在完成分析之后，通过从 “响应 Y”红色小三角菜单中选择**保存列** > **预测公式**可添加该列。
5. 要查看预测公式，请点击列列表中的加号：

   0−50.1465**p1* − 282.198**p2* − 911.648**p3* + *p2***p1**317.363 + *p3***p1**1464.330 + *p3***p2**1846.218

注意：这些结果更正了 Cornell (1990) 中报告的系数。

响应曲面的“解”报表显示最大预测值 19.570299 出现在点 (0.63505, 0.15568, 0.20927) 处。

图 13.24 混料响应曲面分析

响应曲面

系数

	(软化剂1-0.474)/0.375	软化剂2/0.375	(软化剂3-0.151)/0.375	Y
(软化剂1-0.474)/0.375	0	44.62923	205.92138	7.4923077
软化剂2/0.375	.	0	259.62437	-1.491318
(软化剂3-0.151)/0.375	.	.	0	-138.2039

解

变量	临界值
(软化剂1-0.474)/0.375	0.635052
软化剂2/0.375	0.1556806
(软化剂3-0.151)/0.375	0.2092675

解是最大值
假定以下混料总和: 软化剂1+软化剂2+软化剂3=1
对应于解的预测值 19.570299

预测刻画器

报表包含预测刻画器。

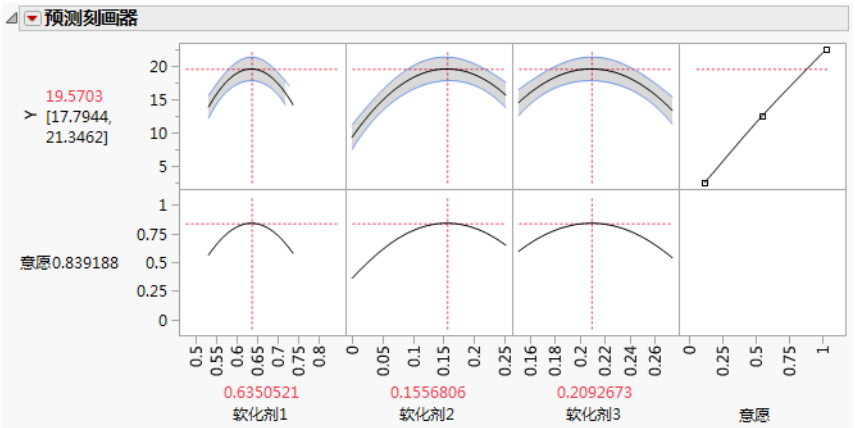
1. 从预测刻画器，您可以看到交叉效应如何显示为预测迹线中的曲率。当您更改一个成分的值（拖动红线）时，其他两个成分的值会向相反方向移动以保持其比例和整体混料约束不变（成分之和必须为 100%）。

注意：预测刻画器迹线的轴范围为因子 p1、p2 和 p3 的上限和下限。当您探索成分的效应且达到第二个成分的限值时，它将无法进一步移动，而仅更改第三个变量来保持混料约束不变。

2. 要在可见的刻画曲线上显示考虑所有三个成分的水平的边界，请点击“预测刻画器”红色小三角菜单并选择**边界刻画 > 边界处停止**。
3. 要优化混料成分，请点击“预测刻画器”红色小三角菜单并选择**优化和意愿 > 意愿函数**。
4. 点击“预测刻画器”红色小三角菜单并选择**优化和意愿 > 最大化意愿**以获取最大 Y 的最优因子设置。

该刻画器显示最优设置（已四舍五入）：p1 为 0.6350、p2 为 0.1557，p3 为 0.2093，这些设置给出的估计响应为 19.5703。

图 13.25 针对混料分析示例的刻画器中的最大意愿



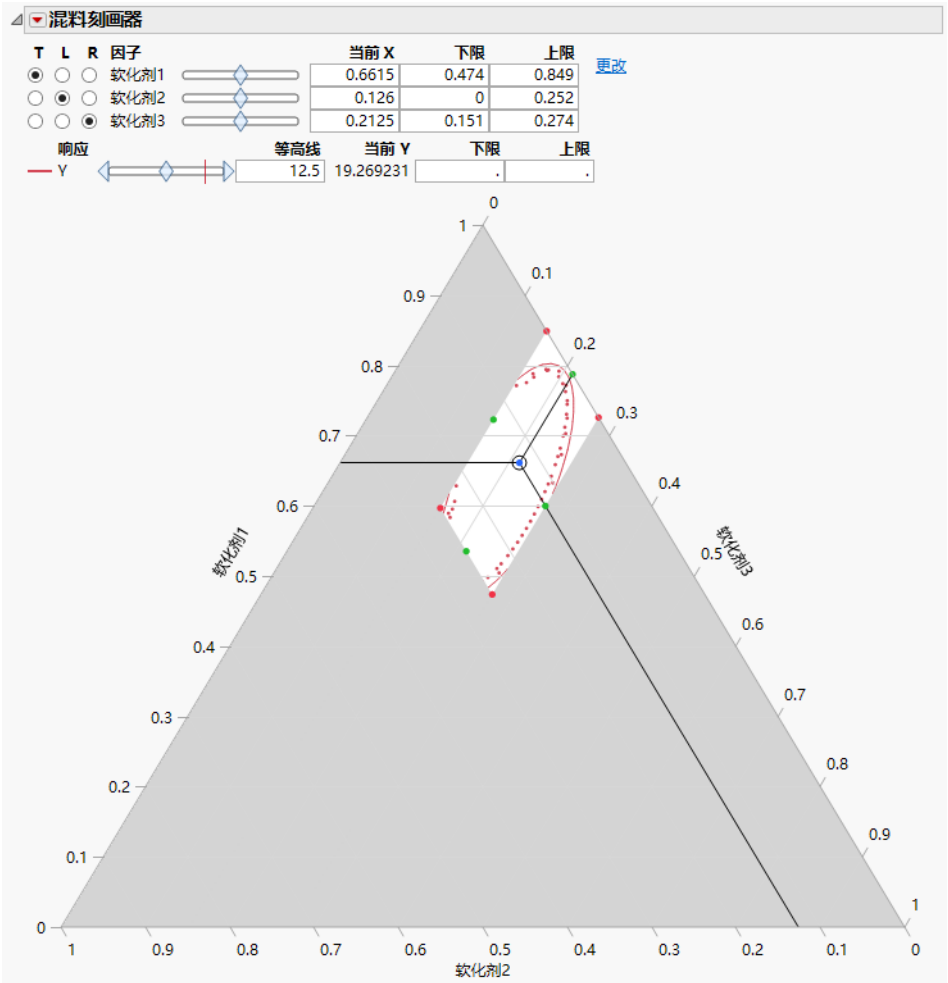
混料刻画器

“拟合模型”报表也包含一个**混料刻画器**，可用于直观演示和优化混料实验中的响应曲面。许多功能都与“等高线刻画器”中的功能相同，但有些功能是“混料刻画器”特有的：

- 三元图代替了笛卡尔图，该图可支持您一次查看三个混料因子。
- 若有超过三个因子，可通过单选按钮来选择要标绘的因子。
- 若对因子有约束，您可以在“下限”列和“上限”列中分别输入其下限和上限。该设置为刻画器中的非可行域着色。

点击“响应 Y”红色小三角并选择**因子刻画 > 混料刻画器**，以查看增塑剂数据的混料刻画器。

图 13.26 增塑剂示例的混料刻画器



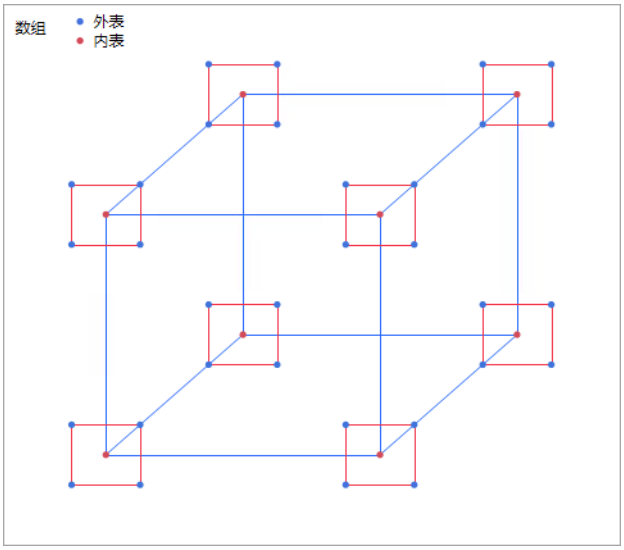
第 14 章

田口设计
含有噪声因子的设计

田口设计使用交叉数组探索存在噪声的环境中的因子。田口设计包含两部分：内表是针对控制因子的设计，外表是针对噪声因子的设计。完整设计是这两个数组的叉积。

请注意，生产中难以控制或控制成本较高的噪声因子必须在实验期间加以控制。田口设计的备选设计包括组合数组和混合分辨率设计 (Borror and Montgomery, 2000)。有关使用定制设计器生成含有噪声因子的设计的示例，请参见“定制设计示例”一章中第 187 页的“针对稳健过程和产品设计的实验”。

图 14.1 包含三个控制因子和两个噪声因子的田口设计



目录

田口设计概述 383
田口设计的示例 383
构建田口设计 387
响应 387
因子 388
选择田口内表和外表设计 388
显示编码设计 388
制作设计表 389
田口设计选项 389

田口设计概述

田口设计定义两类因子：控制因子和噪声因子。

- 为控制因子构造了**内部**设计。
- 为噪声因子构造了**外部**设计。

针对内部和外部设计试验的所有组合执行该实验。必须在实验期间控制噪声因子。内表中每个设置的均值和标准差组合到信噪比测度中进行分析。表 14.1 列出了信噪比性能统计量。

表 14.1 推荐的性能统计量

目标	信噪比公式
望目特性	$\frac{S}{N} = 10\log\left(\frac{\bar{Y}^2}{s^2}\right)$
望大特性（最大化）	$\frac{S}{N} = -10\log\left(\frac{1}{n}\sum_i \frac{1}{Y_i^2}\right)$
望小特性（最小化）	$\frac{S}{N} = -10\log\left(\frac{1}{n}\sum_i Y_i^2\right)$

田口设计的示例

使用田口设计研究针对三个噪声因子评估的四个控制因子。本例采用 Byrne and Taguchi (1986) 中所述的实验。该实验的目的是要找出可将尼龙管的粘度（黏着力）最大化的控制因子设置。

设计中包含四个信号因子：

干扰 尼龙管和连接器干扰。具有 3 个水平的符号因子。

壁厚 连接器壁厚。具有 3 个水平的符号因子。

深度 尼龙管插入连接器的深度。具有 3 个水平的符号因子。

粘度 粘度百分比。具有 3 个水平的符号因子。

设计中包含三个噪声因子：

时间 条件时间。具有 2 个水平的噪声因子。

温度 温度。具有 2 个水平的噪声因子。

湿度 相对湿度。具有 2 个水平的噪声因子。

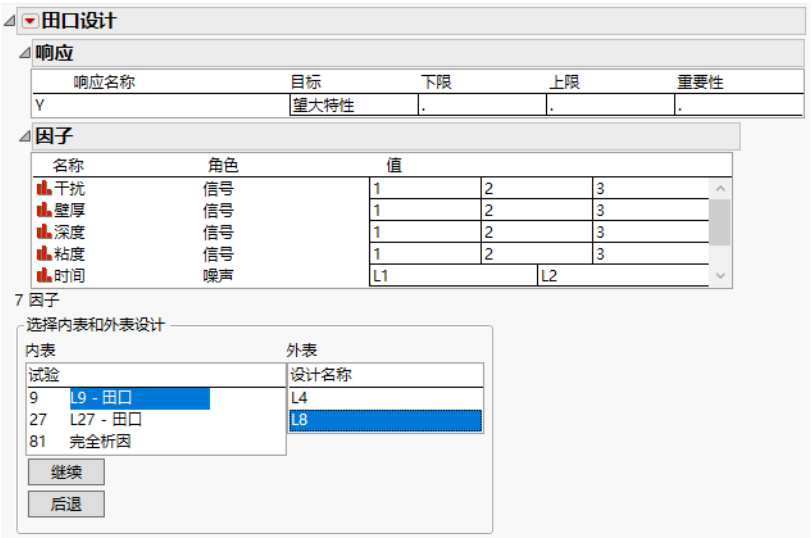
创建设计

1. 选择**实验设计 > 经典 > 田口数组**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Byrne Taguchi Factors.jmp。
3. 在 “田口数组”窗口中，点击 “田口设计”红色小三角菜单并选择**加载因子**。
   因子面板随后显示四个三水平控制 （信号）因子和三个噪声因子。

   注意：为响应选定的 “目标”确定添加到设计数据表的信噪比公式。

4. 确保为内表选定了 L9 - 田口。
5. 为外表设计点击 L8。

图 14.2 完成的 “田口设计”窗口



6. 点击**继续**。
7. 点击**制表**创建图 14.3 中显示的设计表。
   内表针对四个信号因子（每个因子都有三个水平）有九次试验。信号因子设置在表中前四列中提供。
   外部设计是针对三个两水平噪声因子的完全析因设计。外表在设计表的八个列中提供。列名是外表试验的模式。例如，名为 “---”的列用于在所有噪声水平都位于其低水平时收集的结果。针对八个噪声因子组合中的每一个执行这九次试验中的每一次试验，总共有 72 次试验。

图 14.3 输入数据之前的田口设计

	干扰	壁厚	深度	粘度	模式	---	--+	-+-	-++	+--	+-+	++-	+++	均值	信噪比
1	1	1	1	1	----	•	•	•	•	•	•	•	•	•	•
2	1	2	2	2	-000	•	•	•	•	•	•	•	•	•	•
3	1	3	3	3	-+++	•	•	•	•	•	•	•	•	•	•
4	2	1	2	3	0-0+	•	•	•	•	•	•	•	•	•	•
5	2	2	3	1	00+-	•	•	•	•	•	•	•	•	•	•
6	2	3	1	2	0+-0	•	•	•	•	•	•	•	•	•	•
7	3	1	3	2	+-+0	•	•	•	•	•	•	•	•	•	•
8	3	2	1	3	+0-+	•	•	•	•	•	•	•	•	•	•
9	3	3	2	1	++0-	•	•	•	•	•	•	•	•	•	•

设计表允许输入 72 个实验结果。

8. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Byrne Taguchi Data.jmp。

图 14.4 完成的田口设计表 (Byrne Taguchi Data.jmp)

	干扰	壁厚	深度	粘度	模式	Y---	Y--+	Y-+-	Y-++	Y+--	Y+-+	Y++-	Y+++	Y 均值	Y 信噪比
1	1	1	1	1	----	15.6	9.5	16.9	19.9	19.6	19.6	20	19.1	17.525	24.02534
2	1	2	2	2	-000	15	16.2	19.4	19.6	19.7	19.8	24.2	21.9	19.475	25.52164
3	1	3	3	3	-+++	16.3	16.7	19.1	15.6	22.6	18.2	23.3	20.4	19.025	25.33476
4	2	1	2	3	0-0+	18.3	17.4	18.9	18.6	21	18.9	23.2	24.7	20.125	25.90425
5	2	2	3	1	00+-	19.7	18.6	19.4	25.1	25.6	21.4	27.5	25.3	22.825	26.90753
6	2	3	1	2	0+-0	16.2	16.3	20	19.8	14.7	19.6	22.5	24.7	19.225	25.32574
7	3	1	3	2	+-+0	16.4	19.1	18.4	23.6	16.8	18.6	24.3	21.6	19.85	25.71081
8	3	2	1	3	+0-+	14.2	15.6	15.1	16.8	17.8	19.6	23.2	24.4	18.3375	24.83231
9	3	3	2	1	++0-	16.1	19.9	19.3	17.3	23.1	22.7	22.6	28.6	21.2	26.15198

信噪比 Y 列是将响应最大化的性能统计量（越大越好）。该统计量用 –10 乘以响应的平方倒数平均值的常用对数计算得出：

$$-10Log10\left[Mean\left[\frac{1}{(Y---)^2}, \frac{1}{(Y--+)^2}, \frac{1}{(Y-+-)^2}, \frac{1}{(Y-++)^2}, \frac{1}{(Y+--)^2}, \frac{1}{(Y+-+)^2}, \frac{1}{(Y++-)^2}, \frac{1}{(Y+++)^2}\right]\right]$$

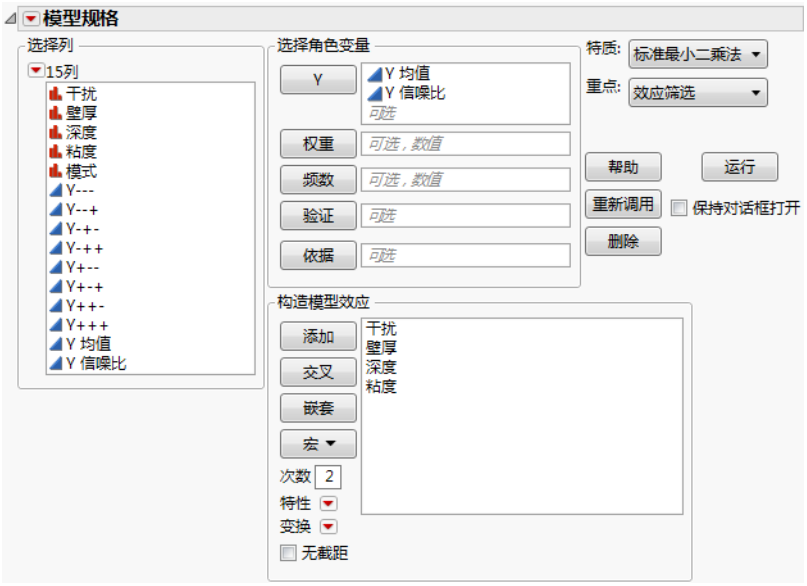
当所有单个 y 响应值都很大时，该表达式也会很大。也就是说，您要尽量查找针对噪声设置生成最大响应的信号设置。

分析数据

现在可以分析数据了。分析目标是要找出可将均值和信噪比同时最大化的因子设置。

1. 在 Byrne Taguchi Data.jmp 数据表中，点击绿色箭头以运行“模型”脚本。

图 14.5 田口数据的 “拟合模型” 启动窗口

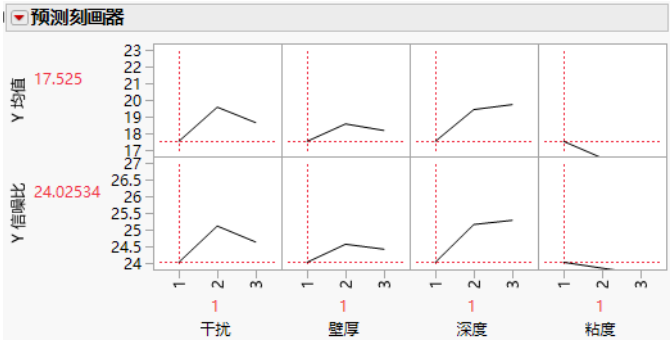


该脚本启动 “拟合模型” 窗口。模型包含四个信号因子的主效应，以便对均值 （**均值 Y**） 和信噪比 （**信噪比 Y**） 响应建模。

2. 点击**运行**。

   可使用报表底部的 “预测刻画器” 快速找出为该实验提供最高信噪比的设置以及该最大值处的均值。

图 14.6 预测刻画器

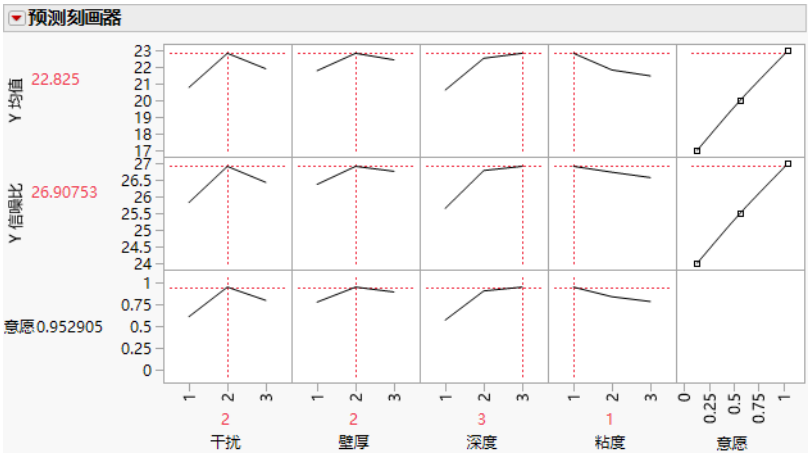


3. 点击 “预测刻画器” 红色小三角，然后选择**优化和意愿 > 意愿函数**。

   这将向刻画器添加一行迹线和一列函数设置，如图 14.7 所示。默认意愿函数设置为 “望大特性”，这正是您要在该实验中实现的目标。有关预测刻画器中的意愿函数的详细信息，请参见 《刻画器指南》。

4. 点击“预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 14.7 Byrne Taguchi 数据的最佳因子设置

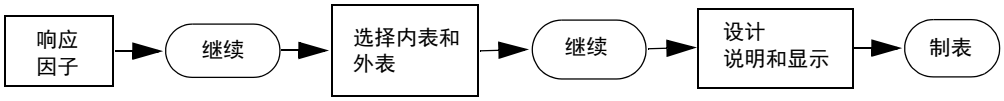


在本例中，**干扰**和**壁厚**的最优设置为 2，**深度**的最优设置为 3，**粘度**的最优设置为 1。这些设置生成的预测均值为 22.8，信噪比为 26.9。

构建田口设计

通过选择**实验设计 > 经典 > 田口数组**首先定义响应和因子。接下来，继续执行设计选项、生成和评估过程。设计过程遵循图 14.8 中的流程。请参见第 56 页的“实验设计工作流程：描述、指定、设计”。

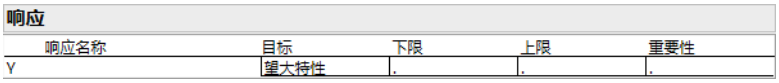
图 14.8 田口设计工作流程



响应

使用“响应”分级显示项指定田口设计的响应。

图 14.9 “响应”分级显示项



“响应”分级显示项包含以下列：

响应名称 响应的名称 （默认为 Y）。要更改该名称，双击它，然后输入不同名称。

目标、下限、上限 “目标”确定信噪比函数的形式（第 383 页上的表 14.1）。选择以下目标之一：“望大特性”、“望目特性”、“望小特性”或“无”。当您创建设计时，JMP 将信噪比公式保存到反映您选定目标的数据表中。要更改目标，请点击它，然后输入一个不同目标。若响应有限制，请使用 “下限”和 “上限”。

重要性 当您有几个响应时，指定的 “重要性”值用于计算总意愿函数。这些值被视为响应的权重。田口设计只有一个响应，“重要性”默认设置为 1。无需为 “重要性”输入值。

因子

使用 “因子”分级显示项指定田口设计的因子。

信号 指定两个或更多 2 水平或 3 水平信号因子。信号因子是系统控制输入。这些是您可以在生产中控制的因子。

噪声 指定一个或多个噪声因子。噪声因子是在生产中难以控制或控制成本较高的变量。不过，您必须能够在实验过程中控制噪声因子。点击 “噪声”添加噪声因子。

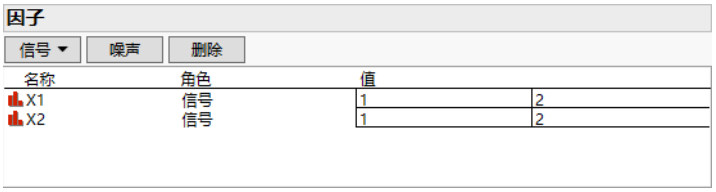
删除 删除选定的因子。

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、 X2 等。要更改该名称，双击它，然后输入不同名称。

角色 指定因子的设计角色 （信号或噪声）。这是在添加因子时设置的。

值 因子的实验设置。要插入值，点击默认值并输入所需值。

图 14.10 “因子”分级显示项



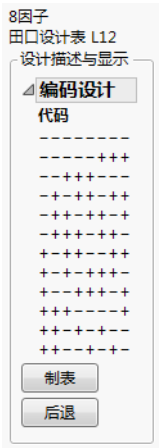
选择田口内表和外表设计

点击“指定因子”面板中的“继续”后，就会出现“选择内表和外表设计”选项。根据您指定的信号因子，您会有一个或多个针对内表的正交设计选择。针对正交“L”设计的外表选择取决于设计中的因子数量和类型。

显示编码设计

选择内表和外表的设计类型后，当内表具有两水平因子时就会显示“编码设计”。

图 14.11 八因子 L12 设计的编码



“编码设计”显示每次试验中因子高值和低值的模式。

制作设计表

当您点击**制表**后，将显示类似于图 14.12 所示的田口设计表。在数据表中，每行代表一次内表试验。在**模式**变量的值中，加号代表高水平，减号代表低水平。在该表中，每个因子对应一列，“模式”列中显示的每个模式对应一列。模式列包含实验响应。若选定“田口设计”红色小三角菜单中的“模拟响应”，模式列将填充模拟的响应值。

图 14.12 适用于八因子 L12 设计的田口设计表

	X1	X2	X3	X4	X5	X6	X7	X8	模式	run 1	run 2	均值	信噪比
1	1	1	1	1	1	1	1	1	--------	•	•	•	•
2	1	1	1	1	1	2	2	2	-----+++	•	•	•	•
3	1	1	2	2	2	1	1	1	--+++---	•	•	•	•
4	1	2	1	2	2	1	2	2	-+-++-++	•	•	•	•
5	1	2	2	1	2	2	1	2	-++-++-+	•	•	•	•
6	1	2	2	2	1	2	2	1	-+++-++-	•	•	•	•
7	2	1	2	2	1	1	2	2	+-++--++	•	•	•	•
8	2	1	2	1	2	2	2	1	+-+-+++-	•	•	•	•
9	2	1	1	2	2	2	1	2	+--+++-+	•	•	•	•
10	2	2	2	1	1	1	1	2	+++-----+	•	•	•	•
11	2	2	1	2	1	2	1	1	++-+-+--	•	•	•	•
12	2	2	1	1	2	1	2	1	++--+-+-	•	•	•	•

田口设计选项

“田口设计”红色小三角菜单包含用于设计设置和生成的选项。

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用“保存响应”选项保存的响应。

保存因子 将 “因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在 “设计角色”区域中，选择合适的角色。

加载因子 加载使用 “保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在 “定义因子约束”或 “线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为 “小于”或 “大于”约束。请参见 “列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用 “保存约束”选项保存的因子约束。

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的 “试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击 “制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示 “模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一： “正态”、 “二项式”或 “Poisson”。

注意：并非所有分布都可用于所有设计类型。

– 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在“模型”窗口中做出选择以控制模拟的响应值的分布。当您点击“应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中，Y 是响应的名称。再次点击“应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见“定制设计”一章中第 101 页的“模拟响应”。

注意: JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见《基本分析》中的“模拟”一章。

第 15 章

评估设计
探索设计的属性

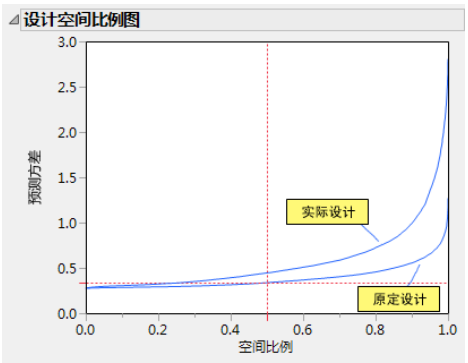
使用“评估设计”平台，您可以：

- 查看现有实验设计的优点和局限性。
- 确定设计用于检测与响应中有意义的变化相关的效应的能力。
- 处理预测方差和估计值的精度。
- 揭示别名现象。
- 获得效率测度。

“评估设计”平台生成显示在几个实验设计平台所提供的“设计评估”分级显示项中的结果。“评估设计”和“设计评估”分级显示项提供的诊断都包括以下项：

- 功效分析
- 预测方差刻画器和曲面图
- 设计空间比例图，显示有多少设计空间的预测方差位于给定值之上
- 参数估计值的估计效率测度
- 别名矩阵，显示模型效应的偏倚结构
- 色图，显示效应间的绝对相关性
- 设计效率值

图 15.1 比较两个设计空间比例图



目录

“评估设计”平台概述 395
评估设计示例 395
评估丢失试验的影响 396
针对指定模型计算功效 403
“评估设计”启动窗口 405
“评估设计”窗口 405
因子 406
模型 406
别名项 407
设计 408
设计评估 408
功效分析 408
预测方差刻画 413
设计空间比例图 415
预测方差曲面 416
估计效率 417
别名矩阵 418
相关性色图 419
设计诊断 420
“评估设计”选项 422

“评估设计”平台概述

“评估设计”平台提供可评估设计属性的强大工具，无论该设计是由 JMP 还是其他工具创建的。在进行实验前，您可以使用该平台从几个设计中进行选择。您还可以评估在执行的设计中不正确的设置或丢失试验的影响。可以修改假设模型中的项来查看对修改后的模型进行估计有什么影响。还可以修改在“别名项”分级显示项中显示的项以查看对别名矩阵的影响。

首先在启动窗口中输入有关设计的信息。然后，您可以修改假设模型，指定模型中未包括的哪些效应可能需要关注。根据您指定的信息，“设计评估”平台会提供很多方法来评估生成的设计的属性：

功效分析 支持您探索检测给定大小的效应的能力。

预测方差刻画 显示针对因子设置范围的预测方差。

设计空间比例图 显示模型预测方差位于给定值之下或之上的比例。

预测方差曲面 显示任意两个连续因子的预测方差曲面图。

估计效率 对于每个参数，给出与理想（正交）设计（该设计可能不存在）的置信区间长度相比，置信区间长度的部分增加。还给出参数的相对标准误差。

别名矩阵 给出系数，这些系数指示因为可能是活跃的却不在模型中的效应导致模型参数的偏倚程度。

相关性色图 使用强度尺度在图上显示两个效应之间的绝对相关性。

注意：默认强度尺度为灰色尺度。要更改颜色尺度，请点击“相关性色图”红色小三角并选择**由蓝经灰到红**。对于定制颜色尺度，在图中右击并选择**颜色主题**。

设计诊断 给出设计的效率测度。

注意：在几个实验设计平台中，当您构造设计时，将显示“设计评估”分级显示项。该分级显示项显示“评估设计”平台提供的结果。提供“设计评估”分级显示项的平台有：“定制设计”、“确定性筛选设计”、“筛选设计”、“响应曲面设计”和最优设计类型的“混料设计”。

评估设计示例

- “评估丢失试验的影响”
- “针对指定模型计算功效”

评估丢失试验的影响

研究人员做了一个实验来探索三个因子（**硅石**、**硫磺**和**硅烷**）对网球弹性（**拉伸**）的影响。实验的目标是建立**拉伸**的预测模型。使用“响应曲面设计”平台选择了包含 15 次试验的 Box-Behnken 设计。实验后，研究人员得知**硅石** = 0.7 和**硅烷** = 50 的两次试验未正确处理。这些试验不能包括在数据分析中。

使用“评估设计”平台评估未包含这两次试验的影响。获取原定 15 次试验的设计的诊断，并将其与实际只有 13 次试验的设计（缺少两次试验）的诊断进行比较。

构造原定设计和实际设计

原定设计

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Bounce Data.jmp。
2. 选择**实验设计** > **设计诊断** > **评估设计**。
3. 选择**硅石**、**硫磺**和**硅烷**，然后点击 **X，因子**。

   可以根据需要添加**拉伸**作为 **Y，响应**。但是指定响应对设计属性没有任何影响。
4. 点击**确定**。

保持原定设计的“评估设计”窗口打开。

提示：将原定设计的“评估设计”窗口放在屏幕的左侧区域。在执行后续步骤后，将实际设计的相应窗口放在它的右侧。

缺失一些试验的实际设计

在本节中，您将排除**硅石** = 0.7 和**硅烷** = 50 的两次试验。它们是数据表的第 3 和第 7 行。

1. 在 Bounce Data.jmp 中，选择第 3 和第 7 行，在突出显示的区域中右击，然后选择**隐藏和排除**。
2. 选择**实验设计** > **设计诊断** > **评估设计**。
3. 点击**重新调用**。
4. 点击**确定**。

保持实际设计的“评估设计”窗口打开。

提示：将实际设计的“评估设计”窗口放在原定设计的“评估设计”窗口的**右侧**，以便于比较这两个设计。

比较原定设计和实际设计

现在可以使用以下方法比较这两个设计：

- 第 397 页的“功效分析”
- 第 398 页的“预测方差刻画”
- 第 400 页的“设计空间比例图”
- 第 401 页的“估计效率”
- 第 402 页的“相关性色图”
- 第 402 页的“设计诊断”

功效分析

在每个窗口中，执行以下操作：

1. 打开“功效分析”分级显示项。

   该分级显示项为所有预期系数显示默认值 1。这些值在主效应项的各水平上检测到预期响应 2 个单位的变化，前提假定交互作用和二次项不活跃。

   功效计算假定误差项 （预期 RMSE）为 1。从以前的研究中，您认定 RMSE 大约为 2。
2. 在**预期 RMSE** 旁边键入 2。

   在文本框外点击时，功效值随即更新。

   您关注的是检测相应于跨越主效应各水平 6 个单位量级的预测响应的差值。前提假设交互作用和二次项不活跃。要统一设置这些值，请使用红色小三角选项。
3. 点击 “功效分析”红色小三角并选择**设置功效的 Delta**
4. 键入 6 作为 delta 值。
5. 点击**确定**。

图 15.2 显示两个分级显示项，其中关闭了“设计和预期响应”分级显示项。

图 15.2 “功效分析”分级显示项，原定设计 （左）和实际设计 （右）

功效分析

显著性水平 0.05
预期 RMSE 2

项	预期系数	功效
截距	3	0.553
硅石	3	0.919
硫磺	3	0.919
硅烷	3	0.919
硅石*硫磺	3	0.672
硅石*硅烷	3	0.672
硫磺*硅烷	3	0.672
硅石*硅石	3	0.639
硫磺*硫磺	3	0.639
硅烷*硅烷	3	0.639

将更改应用于预期系数

设计和预期响应

功效分析

显著性水平 0.05
预期 RMSE 2

项	预期系数	功效
截距	3	0.432
硅石	3	0.533
硫磺	3	0.533
硅烷	3	0.798
硅石*硫磺	3	0.231
硅石*硅烷	3	0.533
硫磺*硅烷	3	0.533
硅石*硅石	3	0.379
硫磺*硫磺	3	0.379
硅烷*硅烷	3	0.379

将更改应用于预期系数

设计和预期响应

实际设计的功效值均小于原定设计的功效值。对于**硅石**和**硫磺**，原定设计中检验的功效几乎是实际设计中功效的两倍。对于**硅石*硫磺**交互作用，实际设计中检验的功效为 0.231，而原定设计中检验的功效为 0.672。与原定设计相比，实际设计的功效显著下降。

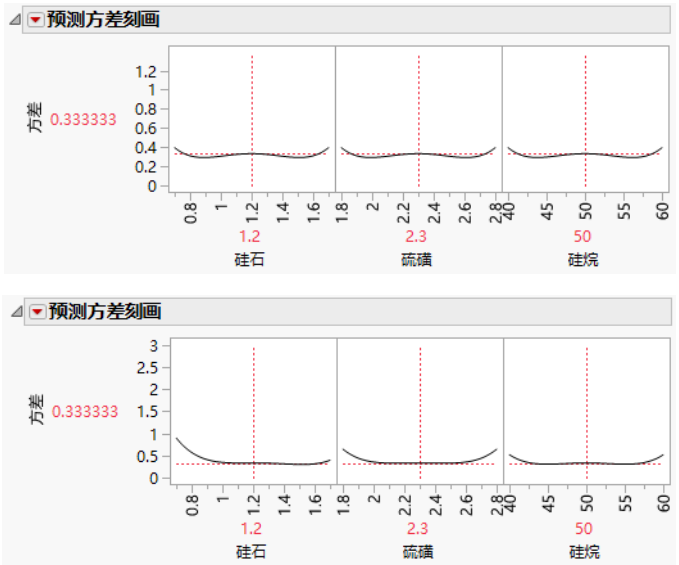
预测方差刻画

1. 在每个窗口中，打开“预测方差刻画”分级显示项。
2. 在实际设计的窗口中，右击垂直轴并从菜单中选择**编辑 > 复制轴设置。**

   该操作会创建一个包含轴设置的脚本。接下来，将这些轴设置应用到原定设计的预测方差刻画图。
3. 在原定设计的 “评估设计”窗口中，找到 “预测方差刻画”分级显示项。右击垂直轴并从菜单中选择**编辑 > 粘贴轴设置。**

   这些图显示在图 15.5 中，其中原定设计的图显示在顶部，实际设计的图显示在底部。

图 15.3 预测方差刻画，原定设计 （顶部）和实际设计 （底部）



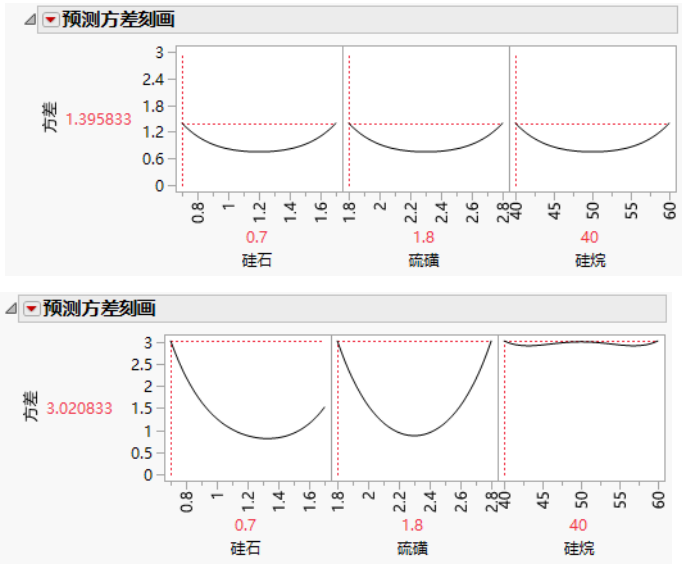
“预测方差刻画”图是相对预测方差的刻画器视图。您可以在设计空间的各个区域中探索相对预测方差。

两个图都显示设计空间中心处的相对预测方差。但是，靠近设计空间边缘的点的方差大于原定设计中的同一点的方差。通过将所有三条竖线移到靠近因子设置边缘的点来探索该现象。

4. 在两个窗口中，点击 “预测方差刻画”红色小三角并选择**最大化方差**。

图 15.4 显示原定设计和实际设计的最大相对预测方差。

图 15.4 最大化的预测方差刻画，原定设计 （顶部）和实际设计 （底部）



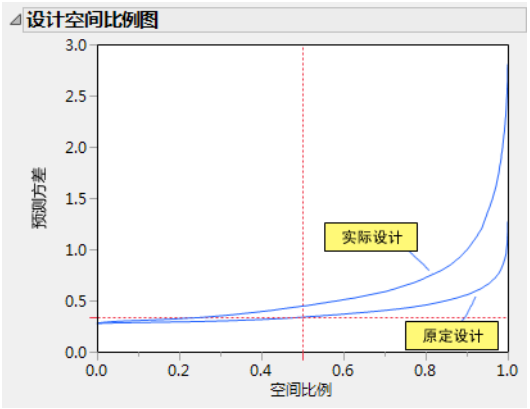
两个设计的刻画器在众多的设计点中标识了相同的设计点：**硅石** =0.7、**硫磺** =1.8、**硅烷** =40，在这点上预测方差最大。原定设计的最大预测方差为 1.396，实际设计的最大预测方差为 3.021。请注意有其他点可使预测方差最大化。实际设计的最大预测方差较大意味着设计空间的某些部分中的预测没有它们在原定设计中的预测准确。

设计空间比例图

1. 在每个窗口中，打开“设计空间比例图”分级显示项。
2. 在原定设计的窗口中，在图中右击，然后选择**编辑 > 复制框架内容**。
3. 在实际设计的窗口中，找到 “设计空间比例图”分级显示项。
4. 在图中右击，然后选择**编辑 > 粘贴框架内容**。

图 15.5 显示带注解的图。每个设计空间比例图显示相对预测方差位于特定值之下的设计空间的比例。

图 15.5 设计空间比例图



在整个设计空间上实际设计的相对预测方差大于原定设计的相对预测方差。随着设计空间覆盖面的增加偏差增大。

估计效率

在每个窗口中，打开“估计效率”分级显示项。

图 15.6 “估计效率”分级显示项，原定设计 （左）和实际设计 （右）

估计效率

项	CI 长度部分增加	估计的相对标准误差
截距	1.236	0.577
硅石	0.369	0.354
硅烷	0.369	0.354
硫磺	0.369	0.354
硅石*硅烷	0.936	0.5
硅石*硫磺	0.936	0.5
硫磺*硅烷	0.936	0.5
硅石*硅石	1.016	0.52
硅烷*硅烷	1.016	0.52
硫磺*硫磺	1.016	0.52

估计效率

项	CI 长度部分增加	估计的相对标准误差
截距	1.082	0.577
硅石	0.803	0.5
硫磺	0.803	0.5
硅烷	0.275	0.354
硅石*硫磺	2.122	0.866
硅石*硅烷	0.803	0.5
硫磺*硅烷	0.803	0.5
硅石*硅石	1.268	0.629
硫磺*硫磺	1.268	0.629
硅烷*硅烷	1.268	0.629

在实际设计（右）中，所有参数的相对标准误差均超过或等于原定设计（左）的标准误差。对于三个非截距参数外的所有参数，实际设计中的相对标准误差均超过原定设计中的标准误差。

“CI 长度部分增加”比较当前设计所给出的参数的置信区间长度与相同试验次数的理想设计所给出的置信区间长度。置信区间长度以及与之有关的“CI 长度部分增加”受试验次数的影响。请参见第 417 页的“CI 长度部分增加”。 尽管试验次数减少了，对于实际设计，项**硅烷**、**硅石*硅烷**和**硫磺*硅烷**的增加量比原定设计的增加量小。这是因为定义实际设计而被删除的两次试验将**硅烷**设置为其中心点。通过删除这两次试验，这些参数的置信区间宽度更类似于理想正交设计（该设计没有中心点）的参数置信区间宽度。

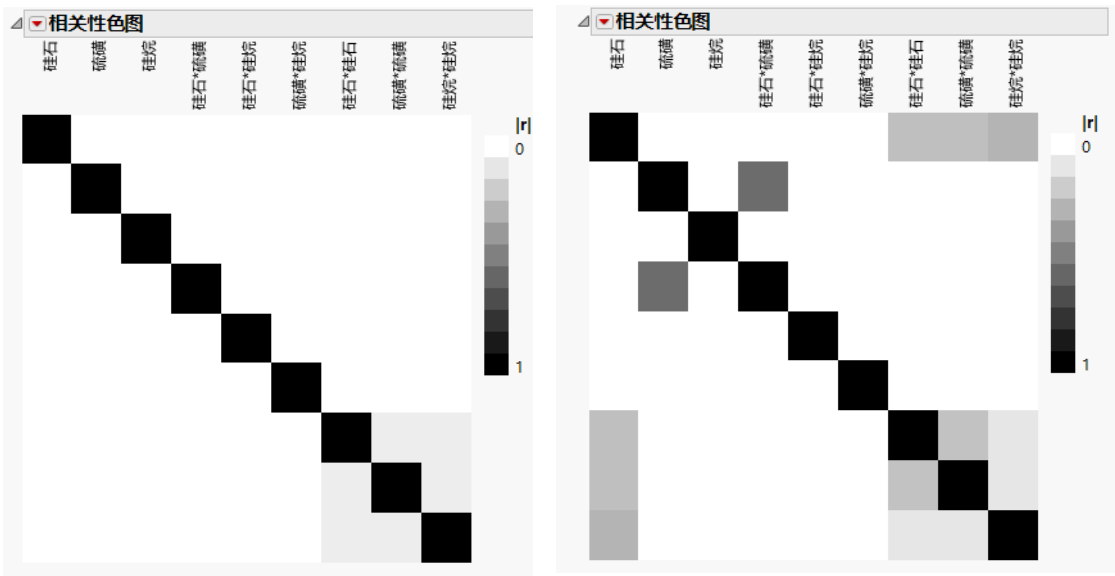
相关性色图

在每个报表中，执行以下操作：

1. 打开**相关性色图**分级显示项。

   这两个色图显示“模型”分级显示项中的效应。每个图使用 JMP 默认的白色到黑色的着色强度尺度显示两个效应之间的绝对相关性。理想的情况下，您希望两个效应之间的相关性为零或很小。

图 15.7 相关性色图，原定设计 （左）和实际设计 （右）



相关性的绝对值范围从 0（白色）到 1（黑色）。在单元格上悬停可查看绝对相关性的值。实际设计的色图与原定设计的色图相比显示更多值较大的绝对相关性。例如，对于原定设计，**硫磺**和**硅石*硫磺**之间的相关性 < .0001，而对于实际设计，该值为 0.5774。

设计诊断

在每个报表中，打开“设计诊断”分级显示项。

图 15.8 设计诊断，原定设计 （左）和实际设计 （右）

原定设计（左）具有比实际设计（右）更高的效率值和更低的平均预测方差。“设计评估”分析的结果指示丢失的两次试验对设计具有负面影响。

请注意效率测度的计算需要考虑试验次数和模型矩阵因素。尤其是，*D*、*G* 和 *A* 效率针对给定设计的试验规模相对于理想设计计算得到。较大的设计不一定比较小设计的效率高。但是，对于给定的因子数，较大的设计与较小的设计相比通常具有更小的预测平均方差。有关如何定义效率测度的详细信息，请参见第 420 页的“设计诊断”。

针对指定模型计算功效

对于本例，您构造了一个确定性筛选设计来确定六个因子中的哪些因子对萃取过程的产量有影响。数据位于 Design Experiment 文件夹下的 Extraction Data.jmp 样本数据表中。因为设计是**确定性**筛选设计，每个因子具有三个水平。请参见第 215 页的“确定性筛选设计”一章。

您关注的是检测到强二次效应的检验功效。其量值至少是误差变异的三倍才被称为强效应。

尽管实验研究了六个因子，效应稀疏原则表明只有几个因子是有效的。因此，您在基于较少因子数的模型中研究功效时把握更大。此外，过去对相关过程的研究充分证明**丙醇**、**丁醇**和 pH 这三个因子的主效应可以忽略不计，它们不与其他因子交互，也没有二次效应。因此您认为可能的模型只包含**甲醇**、**乙醇**和**时间**的主效应、交互作用效应和二次效应。您决定在三因子响应曲面模型的环境下研究功效。

使用“评估设计”平台确定您的设计对检测**甲醇**、**乙醇**或**时间**的强二次效应的功效。

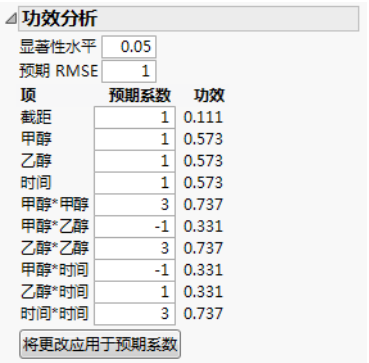
1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Extraction Data.jmp。
2. 选择**实验设计** > **设计诊断** > **评估设计**。
3. 选择**甲醇**、**乙醇**和**时间**，然后点击 **X，因子**。

   可以根据需要添加**产量**作为 **Y，响应**。但是指定响应对设计属性没有任何影响。
4. 点击**确定**。
5. 在“模型”分级显示项中，点击 RSM。

   这为三个因子添加了交互作用和二次项。
6. 打开“功效分析”分级显示项。

   请注意，默认情况下“预期 RMSE”设置为 1。尽管您具有来自过去研究的 RMSE 估计值，但是不需要输入它。这是因为关注效应的大小是误差变异的三倍。
7. 在**预期系数**下，在**甲醇 * 甲醇**、**乙醇 * 乙醇**和**时间 * 时间**旁边键入 3。
8. 点击**将更改应用于预期系数**。

图 15.9 将更改应用于系数后的 “功效分析”分级显示项

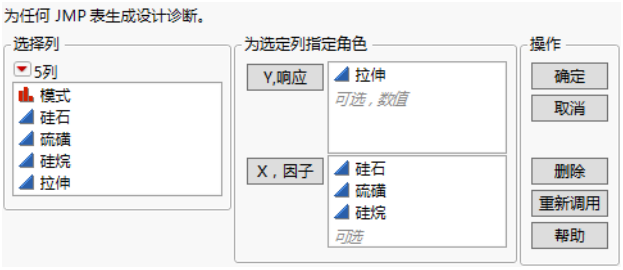


检测二次效应的功效为 0.737，其量值是误差变异的三倍。这假定最终模型是包含三个因子的响应曲面。它还假定检验的显著性水平为 0.05。

“评估设计”启动窗口

要启动“评估设计”平台，需要打开关注的数据表，然后选择**实验设计 > 设计诊断 > 评估设计**。图 15.10 中的示例使用 Design Experiment 文件夹下的 Bounce Data.jmp 样本数据表。

图 15.10 “评估设计”启动窗口



启动窗口包含以下按钮：

Y，响应 输入一个或多个响应列。响应的输入是可选的。评估设计时不使用响应值。响应必须是数值。

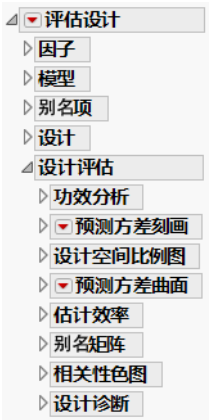
X，因子 输入因子列。因子可以为任意数据类型或建模类型。

“评估设计”窗口

“评估设计”窗口包含两个部分。请参见图 15.11，其中所有分级显示项节点均已关闭。

- “因子”、“模型”、“别名项”和“设计”分级显示项定义模型和设计。
- “设计评估”分级显示项提供描述您设计属性的结果。

图 15.11 显示所有可能的分级显示项的 “评估设计”窗口



“因子”、“模型”、“别名项”和 “设计”分级显示项包含您输入的有关因子、假设模型、关注的可能别名效应以及实际设计的信息。 JMP 使用您在启动窗口和设计表中所作的选择填充这些分级显示项。但是，您可以在 “模型”和 “别名项”分级显示项中修改效应。指定相关信息后，“设计评估”分级显示项会随即更新。您可以打开这些分级项以查看报表或控件窗口，其中提供了有关您的设计的信息。

- “因子”
- “模型”
- “别名项”
- “设计”
- “设计评估”
- “功效分析”
- “预测方差刻画”
- “设计空间比例图”
- “预测方差曲面”
- “估计效率”
- “别名矩阵”
- “相关性色图”
- “设计诊断”

因子

“因子”分级显示项列出在启动窗口中输入的因子。您可以选择因子来构造“模型”分级显示项中的效应。

图 15.12 “因子”分级显示项

因子

名称	角色
硅石	连续
硫磺	连续
硅烷	连续

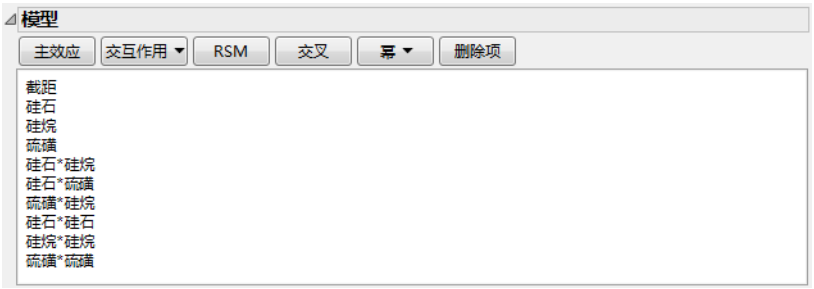
模型

“模型”分级显示项中显示的内容取决于关联数据表是包含 “模型”还是 “拟合模型”脚本：

- 若存在脚本，“模型”分级显示项将包含在该脚本中指定的可估计效应。效应将按照在该脚本中指定的顺序进入模型。若模型脚本包含的效应超出了可估计的效应，那些不可估计的效应不会包含在模型中进行评估。
- 若不存在脚本，“模型”分级显示项将仅包含主效应。

图 15.13 显示 Design Experiment 文件夹下的 Bounce Data.jmp 数据表的“模型”分级显示项。数据表中的“模型”脚本包含三个因子**硅石**、**硅烷**和**硫磺**的响应曲面效应。因此，“模型”分级显示项包含这三个因子的主效应、双因子交互作用和二次效应。

图 15.13 Bounce Data.jmp 的 “模型”分级显示项



您可以使用以下按钮向 “模型”分级显示项添加效应：

主效应 为模型中的所有因子添加主效应。

交互作用 添加交互作用效应。若未在 “因子”分级显示项中选择任何因子，选择 “二次”、“三次”、“四次”或 “五次”，可添加最高达到该阶数的所有适当交互作用。通过在 “因子”分级显示项中选择因子名称，再选择 “交互作用”，然后指定适当的阶数，可以添加特定因子最高达到给定阶数的交互作用。不会添加非混料因子与混料因子之间的交互作用，以及带有分区组因子和带有常数因子的交互作用。

RSM 添加连续因子最高达到二阶的交互作用和二次项 （响应曲面模型项）。分类因子不包括在 RSM 项中。将删除与所有混料因子交互的非混料因子的主效应。

交叉 添加特定的交互作用项。在 “因子”分级显示项中选择因子名称，在 “模型”分级显示项中选择效应名称。点击 “交叉”将交叉项添加到 “模型”分级显示项。

幂 添加多项式项。若未在 “因子”分级显示项中选择因子名称，则为所有连续因子添加多项式项。若在 “因子”分级显示项中选择了因子名称，则仅为这些因子添加多项式项。选择 “二次”、“三次”、“四次”或 “五次”添加该阶数的多项式项。

Scheffé 三次项 为所有混料因子添加 Scheffé 三次项。这些项用于指定包含三次多项式项的混料模型。

删除项 删除选定的效应。

别名项

未包含在假设模型中的效应有可能是活跃的。在“别名项”分级显示项中列出可能的活跃效应，它们不在您的假设模型中，但是可能使模型项的估计值偏倚。“别名矩阵”条目表示您在“别名项”分级显示项中指定的效应传导给模型参数的偏倚度。请参见“技术细节”附录中第 735 页的“别名矩阵”。

默认情况下，“别名项”分级显示项包含“模型”分级显示项中未包含的所有双因子交互作用效应（涉及分区组因子的项除外）。使用按钮添加或删除项。有关如何使用这些按钮向“别名项”表中添加效应的说明，请参见第 406 页的“模型”。

在“评估设计”平台中，“别名矩阵”分级显示项会立即更新以反映对“别名矩阵”效应的更改。在“定制设计”平台中，在“别名项”分级显示项中修改效应后您必须点击“制作设计”。在构造设计的其他实验设计平台中，没有“别名项”分级显示项。但是，在构造设计后，包含适当效应的“别名矩阵”分级显示项将显示在“设计评估”下。

设计

“设计”分级显示项针对您在启动窗口中指定的因子显示这些设计的试验。当您在“设计评估”分级显示项中探索设计属性时可以轻松查看设计。

设计评估

“评估设计”平台内的“设计评估”基于您的设计以及在“模型”和“别名项”分级显示项中指定的内容。几个实验设计平台提供“设计评估”分级显示项：“定制”、“确定性筛选”、“筛选”、“响应曲面”和具有最优设计类型的“混料”。这些平台内的“设计评估”基于您构造的设计。

功效分析

“功效分析”分级显示项为您模型中的参数计算检验功效。功效是检测到给定大小的活跃效应的概率。“功效分析”分级显示项帮助您评估您的设计检测具有实际意义的效应的能力。功效越高，您越可能检测到显著效应（假设前提是您的系数和 RMSE 假设均正确）。功效取决于试验次数、显著性水平和估计的误差变异。特别是您可以决定是否需要增加额外的试验。

本节涉及以下主题：

- 第 408 页的“功效分析概述”
- 第 410 页的“功效分析详细信息”
- 第 410 页的“针对各个参数的检验”
- 第 411 页的“针对具有两个以上水平的分类效应的检验”
- 第 411 页的“‘设计和预期响应’分级显示项”
- 第 412 页的“咖啡实验的功效分析”

功效分析概述

为“模型”分级显示项中列出的效应计算功效。它们包括连续因子、离散数值因子、分类因子、分区组因子、混料因子和协变量因子。这些检验针对各个模型参数和整体效应。有关如何计算功效的详细信息，请参见“技术细节”附录中第 736 页的“功效计算”。

功效是在模型参数的指定值下拒绝“无效应”原假设的概率。实际上，您关注的不是模型参数的值，而是检测响应均值中具有实际意义的差值。在“功效分析”分级显示项中，您可以针对预期系数的指定值计算预期响应。这帮助您确定与要检测的响应均值差值相关的系数值。

图 15.14 显示 Design Experiment 文件夹下 Coffee Data.jmp 样本数据表中设计的“功效分析”分级显示项。“模型”脚本中指定的模型是仅包含主效应的模型。

图 15.14 Coffee Data.jmp 的功效分析

功效分析

显著性水平 0.05

预期 RMSE 1

项	预期系数	功效
截距	1	0.789
研磨	-1	0.789
温度	1	0.789
时间	1	0.789
咖啡豆的量	1	0.789
测量地点 1	1	0.507
测量地点 2	1	0.507

将更改应用于预期系数

效应	功效
测量地点	0.888

设计和预期响应

预期响应	研磨	温度	时间	咖啡豆的量	测量地点
6	Medium	205	4	2.4	1
-2	Coarse	195	3	1.6	1
2	Medium	205	3	1.6	1
2	Coarse	195	4	2.4	1
2	Coarse	205	4	1.6	2
4	Medium	195	4	2.4	2
0	Medium	195	3	1.6	2
2	Coarse	205	3	2.4	2
-1	Coarse	205	4	1.6	3
-3	Coarse	195	3	2.4	3
-1	Medium	195	4	1.6	3
1	Medium	205	3	2.4	3

将更改应用于预期响应

在“功效分析”分级显示项中，您可以：

- 指定反映要检测的差值的系数值。您在分级显示项的上面部分将这些值作为预期系数输入。
- 指定预期响应值并应用它们来确定相应的预期系数。您在“设计和预期响应”面板中指定预期响应。

从“功效分析”红色小三角菜单中，您可以：

- 指定一个功效以确定实现该功效的预期系数。若不存在满足指定功效的系数，则将系数设置为零。
- 指定 delta 的值，其中预期系数将是指定 delta 的一半。随后基于预期系数更新功效。

功效分析详细信息

指定 “显著性水平”和 “预期 RMSE”的值。它们用于计算针对模型参数检验的功效。

显著性水平 在 “无效应”假设为真时拒绝该假设的概率。功效计算在您输入值后立即更新。

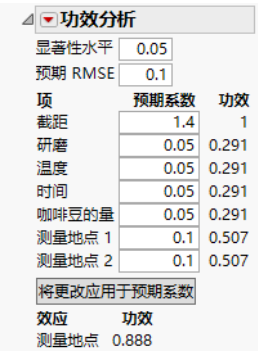
预期 RMSE 误差变异的平方根的估计值。功效计算在您输入值后立即更新。

“功效分析”报表的上面部分打开时显示预期系数的默认值（图 15.14）。默认值基于 Delta。请参见第 422 页的“高级选项 > 设置功效的 Delta”。

注意：若设计超饱和（这意味着要评估的参数个数超过试验次数），则预期系数设置为 0。

图 15.15 显示“功效分析”报表的上面部分，其中已指定了预期系数的值。这些值反映您要检测的差值。

图 15.15 为 Coffee Data.jmp 指定的可能的预期系数值



针对各个参数的检验

“项”列包含模型项的列表。对于每个项，“预期系数”列包含该项的值。“功效”列中的值是以下检验的功效：若预期系数给定了系数的真实值，该项的系数为 0。

项 与要检测的系数相关的模型项。

注意：模型项在“功效分析”报表中显示的顺序可能不同于它们在使用标准最小二乘法得到的“参数估计值”报表中的顺序。仅当模型包含的交互作用具有大于 1 的自由度时，该顺序才会不同。

预期系数 与模型项相关的系数的值。该值用于功效的计算。这些值还用于计算 “设计和预期响应”分级显示项中的 “预期响应”列。当您在 “预期系数”列中设置新值时，点击**将更改应用于预期系数**可更新 “功效”和 “预期响应”列。

注意：预期系数对于连续效应的默认值为 1。它们对于分类效应具有交替值 1 和 –1。您可以通过从红色小三角菜单中选择**高级选项 > 设置功效的 Delta** 来指定 Delta 的值。若您更改 Delta 的值，预期系数的值会更新以便它们的绝对值为 Delta 的一半。请参见第 422 页的“高级选项 > 设置功效的 Delta”。

功效 当指定的预期系数给出真实系数值时拒绝 “无效应”原假设的概率。对于与数值因子相关的系数，响应均值的变化量（基于模型）是系数值的两倍。对于与分类因子相关的系数，因子各个水平上响应均值的变化量 （基于模型）是预期系数的绝对值的两倍。

计算中使用指定的 “显著性水平”和 “预期 RMSE” 。有关功效计算的详细信息，请参见 “技术细节”附录中第 736 页的 “单个参数的功效”。

将更改应用于预期系数 当您在 “预期系数”列中设置新值时，点击**将更改应用于预期系数**可更新 “功效”和 “预期响应”列。

针对具有两个以上水平的分类效应的检验

若您的模型包含具有两个以上水平的分类效应，则以下列显示在 “将更改应用于预期系数”按钮下方：

效应 分类效应。

功效 检验无效应的功效计算。检验的原假设是：与效应对应的所有模型参数为零。要检测的差值由 “预期系数”列中的值定义，这些值对应于效应的模型项。功效计算反映由预期系数决定的响应均值的差值。

计算中使用指定的 “显著性水平”和 “预期 RMSE” 。有关功效计算的详细信息，请参见 “技术细节”附录中第 737 页的 “分类效应的功效”。

“设计和预期响应”分级显示项

“设计和预期响应”分级显示项在 “预期响应”列之后显示设计。第一列中的每个条目是对应于设计设置的 “预期响应”。使用 “预期系数”计算 “预期响应”。

图 15.16 显示对应于图 15.15 中所示的 “预期系数”值的 “设计和预期响应”分级显示项。

图 15.16 Coffee Data.jmp 的预期响应

设计和预期响应

预期响应	研磨	温度	时间	咖啡豆的量	测量地点
1.6	Medium	205	4	2.4	1
1.4	Coarse	195	3	1.6	1
1.4	Medium	205	3	1.6	1
1.6	Coarse	195	4	2.4	1
1.6	Coarse	205	4	1.6	2
1.5	Medium	195	4	2.4	2
1.3	Medium	195	3	1.6	2
1.6	Coarse	205	3	2.4	2
1.3	Coarse	205	4	1.6	3
1.2	Coarse	195	3	2.4	3
1.1	Medium	195	4	1.6	3
1.2	Medium	205	3	2.4	3

将更改应用于预期响应

在“预期响应”列中，您可以为因子的每个设置指定值。这些值反映您要检测的差值。

点击**将更改应用于预期响应**可同时更新 “预期系数”和 “功效”列。

预期响应 使用 “预期系数”值作为模型中的系数获得的响应值。该分级显示项首次显示时，“预期响应”值的计算基于 “预期系数”列中的默认值。当您在 “预期响应”列中设置新值时，点击**将更改应用于预期响应**可更新 “预期系数”和 “功效”列。

设计 “预期响应”列右侧的列显示设计中所有试验的因子设置。

将更改应用于预期响应 当您在 “预期响应”列中设置新值时，点击**将更改应用于预期响应**可更新 “预期系数”和 “功效”列。

咖啡实验的功效分析

考虑 Coffee Data.jmp 数据表中的设计。假定您关注的是设计检测各个量对**浓度**影响的功效。回想一下，**研磨**是两水平的分类因子，**温度**、**时间**和**咖啡豆的量**是连续因子，**测量地点**是三水平的分类（分区组）因子。

在该示例中，忽略**测量地点**作为分区组因子的角色。您要关注的是**测量地点**对**浓度**的影响。因为**测量地点**是三水平的分类因子，它在“参数”列表中用两项来表示：测量地点 1 和测量地点 2。

您尤其关注检测**浓度**均值的以下变化量的概率：

- 您将**研磨**从“Coarse”变为“Medium”时均值变化了 0.10 个单位。
- 您将**温度**、**时间**和**咖啡豆的量**从低水平变为高水平时均值变化了 0.10 个单位或更多。
- 测量地点 1 和 2 分别导致均值比整体预期均值增加了 0.10 个单位。这对应于测量地点 3 导致均值比整体预期均值减少了 0.20 个单位。

您将 0.05 设置为“显著性水平”。对于固定设计设置，估计的**浓度**标准差为 0.1，输入它作为“预期 RMSE”。

图 15.17 显示输入了这些值后的“功效分析”节点。您明确指定了“显著性水平”、“预期 RMSE”和每个“预期系数”的值。

点击“将更改应用于预期系数”时，“预期响应”值会更新以反映您指定的模型。

图 15.17 在 “预期系数”面板中输入了用户指定内容的 “功效分析”分级显示项

功效分析

显著性水平 0.05
预期 RMSE 0.1

项	预期系数	功效
截距	1.4	1
研磨	0.05	0.291
温度	0.05	0.291
时间	0.05	0.291
咖啡豆的量	0.05	0.291
测量地点 1	0.1	0.507
测量地点 2	0.1	0.507

将更改应用于预期系数

效应	功效
测量地点	0.888

设计和预期响应

预期响应	研磨	温度	时间	咖啡豆的量	测量地点
1.6	Medium	205	4	2.4	1
1.4	Coarse	195	3	1.6	1
1.4	Medium	205	3	1.6	1
1.6	Coarse	195	4	2.4	1
1.6	Coarse	205	4	1.6	2
1.5	Medium	195	4	2.4	2
1.3	Medium	195	3	1.6	2
1.6	Coarse	205	3	2.4	2
1.3	Coarse	205	4	1.6	3
1.2	Coarse	195	3	2.4	3
1.1	Medium	195	4	1.6	3
1.2	Medium	205	3	2.4	3

将更改应用于预期响应

回想一下，**温度**是具有编码水平 -1 和 1 的连续因子。考虑原假设为**温度**对**浓度**没有影响的检验。图 15.17 显示该检验在**温度**的各水平上检测 0.10 (=2*0.05) 个单位差值的功效仅为 0.291。

现在考虑检验整个**测量地点**效应，其中**测量地点**是三水平的分类因子。考虑原假设为**测量地点**对**浓度**没有影响的检验。这其实是当您运行**分析** > **拟合模型**时“效应检验”报表中针对分类因子提供的普通 *F* 检验。请参见《拟合线性模型》。

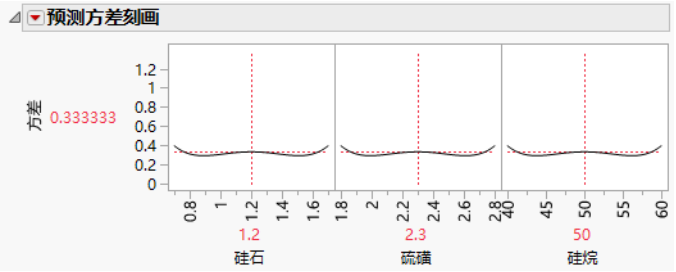
该检验的功效直接显示在“将更改应用于预期系数”按钮下方。模型项“测量地点 1”和“测量地点 2”的“预期系数”下的条目都为 0.10。这些设置暗示两个测量地点的效应都使**浓度**高于整体预期均值 0.10 个单位。对于“测量地点 1”和“测量地点 2”系数的这些设置，“测量地点 3”对**浓度**的影响是使浓度比整体预期均值减少 0.20 个单位。图 15.17 显示检验用于检测至少这个量的差值的功效为 0.888。

预测方差刻画

“预测方差刻画”可帮助您了解您的预测在设计空间中的哪些位置具有较多或较少的变异。期望得到较低的预测方差。使用**最大化方差**选项查找最大方差。请参见第 414 页的“最大化方差”。

“预测方差刻画”将预测的相对方差标绘为每个因子的函数（固定其他因子的值）。图 15.18 显示 Design Experiment 文件夹下 Bounce Data.jmp 数据表的“预测方差刻画”。

图 15.18 预测方差刻画器



相对预测方差

对于因子的给定设置，预测方差为误差方差与依赖于设计和因子设置的量的乘积。在执行实验前，误差方差是未知的，因此预测方差也是未知的。但是，预测方差与误差方差的比值不是误差方差的函数。该比值称为**相对预测方差**，它仅依赖于设计和因子设置。因此，可以在采集数据前计算预测的相对方差。请参见“技术细节”附录中第 738 页的“相对预测方差”。

在执行实验并拟合最小二乘模型后，您可以使用模型拟合的均方误差 (MSE) 估计误差方差。可以通过乘以任意设置处预测的相对方差来估计该设置处预测的实际方差。

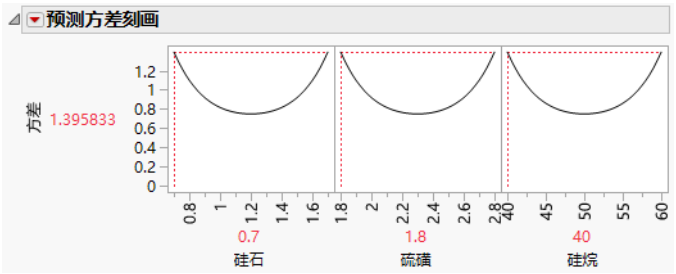
预测方差在整个设计空间上都很小是理想情况。通常，误差方差随着样本大小的增加而减小。在比较设计时，您可能要将两个设计的预测方差刻画器并排放置。首选平均预测方差较小的设计。

最大化方差

您可以根据最大相对预测方差来评估设计或比较设计。从“预测方差刻画”红色小三角菜单中选择**最大化方差**选项。JMP 使用使相对预测方差最大化的意愿函数。“预测方差刻画”中的最大方差值是相对预测方差的最差（从设计角度看最不想要的）值。最大方差可能出现在不止一个因子设置组合中。

图 15.19 显示已选定“最大化方差”的预测方差刻画。该图针对 Design Experiment 文件夹下的 Bounce Data.jmp 样本数据表。当**硅石** = 0.7，**硫磺** = 1.8，**硅烷** = 40 时，相对预测方差的最大值为 1.3958。但是，请记住有多个因子设置都可以得到这个相对方差。**硅石** = 1.7，**硫磺** = 2.8，**硅烷** = 60 这个设计点的相对预测方差也是 1.3958。要使用曲面计算预测方差，请参见第 416 页的“预测方差曲面”。

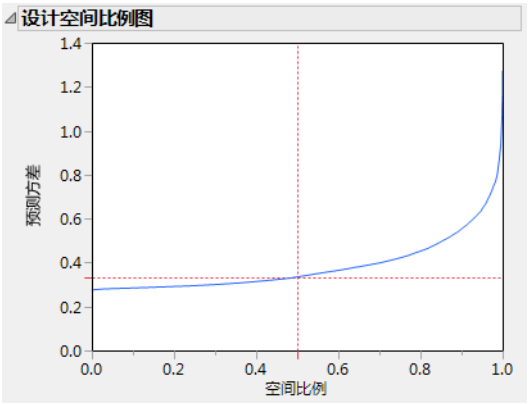
图 15.19 显示最大方差的预测方差刻画



设计空间比例图

设计空间比例图显示相对预测方差位于给定值之下的设计空间的比例。希望得到的结果是：有较大比例的设计空间具有较低的预测方差值。图 15.20 显示 Design Experiment 文件夹下 Bounce Data.jmp 样本数据表的设计空间比例图。

图 15.20 设计空间比例图



图中的 *X* 轴表示设计空间的比例，范围为从 0 到 100%。*Y* 轴表示相对预测方差值。对于落在蓝色曲线上的点 *(x, y)*，值 *x* 是其方差小于等于 *y* 的设计空间的比例。使用红点构成的十字准线标记 50% 的设计空间的相对预测方差的上限值。

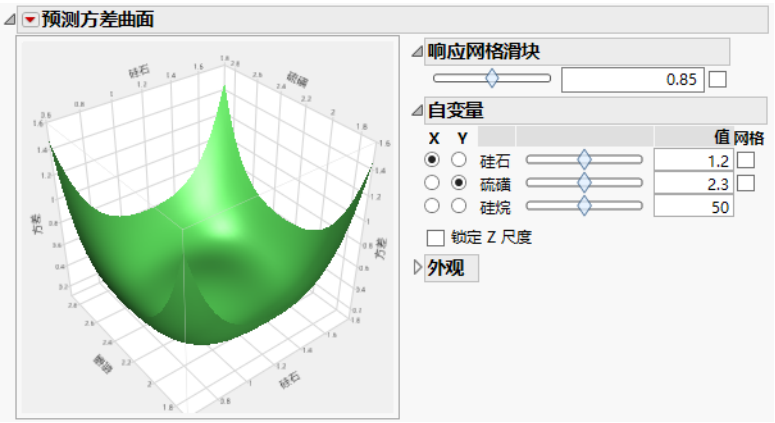
图 15.20 显示最小相对预测方差比 0.3 略小，而最大值则低于 1.4。（实际最大值为 1.395833，如图 15.19 中所示。）使用红点构成的十字准线指示：在大约 50% 的设计空间上，相对预测方差小于 0.34。您可以使用十字准线工具来找到对应于任意空间比例值的最大相对预测方差。使用图 15.20 中的十字准线工具可得到 90% 的预测方差值低于约 0.55。

注意：构造设计空间比例图时使用了设计空间的 Monte Carlo 抽样。因此，同一设计的图可能略有不同。

预测方差曲面

“预测方差曲面”报表标绘相对预测方差曲面为任意两个设计因子的函数。图 15.21 显示 Design Experiment 文件夹下 Bounce Data.jmp 样本数据表的“预测方差曲面”分级显示项。通过在红色小三角菜单上选择**控制面板**来显示或隐藏控件。请参见第 416 页的“控制面板”。

图 15.21 预测方差曲面



当有两个或更多因子时，“预测方差曲面”分级显示项显示任意两个变量的相对预测方差图。“预测方差曲面”分级显示项标绘相对预测方差公式。拖动图可以旋转和更改透视。

控制面板

“控制面板”包含用于调整该图的选项：

响应网格滑块 **网格**复选框叠加一个网格，用于显示“方差”的常数值。“方差”的值显示在文本框中。使用滑块可以调整网格的位置。或者，您可以在文本框中输入“方差”值。在框外点击可以更新图。

自变量 该面板允许您选择使用哪两个因子作为图的轴，以及指定不作为轴的因子的设置。通过点击相应的列，为 X 和 Y 轴选择因子。使用滑块和文本框为**未**选择作为轴的每个因子指定值。该图显示不在图中用作轴的因子取指定值时曲面的三维切片。移动滑块可查看不同的切片。

每个网格复选框激活相应因子的网格。使用滑块调整每个网格的位置。

“锁定 Z 尺度”可将 *z* 轴锁定到其当前值。这在移动轴上未显示的滑块时非常有用。

外观 **分辨率**滑块影响有多少个点用于计算公式。太粗略的分辨率意味着有急剧变化的函数可能不能很好表示。但是设置高分辨率可能使计算和显示曲面变慢。

正射投影复选框显示图在二维空间的投影。

等高线菜单控制等高线曲线的位置。等高线曲线是一组其“响应”值为常数的点。您可以选择关闭等高线 （默认值）或将它们放置在下层、上层或在曲面上。

估计效率

该报表给出模型中每个参数估计值的“CI（置信区间）长度部分增加”和“估计的相对标准误差”。这两个值越小越好。图 15.22 显示 Design Experiment 文件夹下 Bounce Data.jmp 样本数据表的“估计效率”分级显示项。

图 15.22 “估计效率”分级显示项

估计效率

项	CI 长度部分增加	估计的相对标准误差
截距	1.236	0.577
硅石	0.369	0.354
硅烷	0.369	0.354
硫磺	0.369	0.354
硅石*硅烷	0.936	0.5
硅石*硫磺	0.936	0.5
硫磺*硅烷	0.936	0.5
硅石*硅石	1.016	0.52
硅烷*硅烷	1.016	0.52
硫磺*硫磺	1.016	0.52

CI 长度部分增加

“CI 长度部分增加”比较当前设计给出的参数的置信区间长度与理想设计给出的置信区间长度：

- 将从参数的实际置信区间长度减去其理想置信区间长度。
- 然后用该差值除以理想置信区间长度。

对于 D 最优正交设计，部分增加量为零。选择设计时，您希望置信区间长度的部分增加量尽量的小。

理想设计

普通最小二乘估计量的协方差矩阵为 $\sigma^2(\mathbf{X}'\mathbf{X})^{-1}$。$(\mathbf{X}'\mathbf{X})^{-1}$ 的对角线元素是参数估计值的**相对方差**（除以 σ^2 后的方差）。对于两水平设计和使用效应编码规则（请参见“列属性”附录中第 709 页的“编码”），任何参数估计值的相对方差的最小值为 $1/n$，其中，n 是试验次数。当设计的所有效应都正交并且设计为 D 最优时就会发生这种情况。

用 $\hat{\beta}$ 表示参数估计值的向量。**理想设计**（可能不存在）是具有如下给出的协方差矩阵的设计：

$$\mathrm{Var}(\hat{\beta}) = (\sigma^2/n)\mathbf{I}_n$$

其中，$\mathbf{I}_n$ 是 $n*n$ 单位矩阵，σ 是响应的标准差。

若存在正交 D 最优设计，则该设计为理想设计。不过，上述定义将理想设计的范围延伸到了正交和 D 最优均不存在的情况。

该定义还适用于有多水平分类因子的设计。用于分类因子的正交编码允许此类设计具有理想的协方差矩阵。对于定制设计，您可以查看编码矩阵，方法是：在“定制设计”窗口的选项中选择“保存 X 矩阵”、制作设计表并查看保存到设计表中的脚本“模型矩阵”。

置信区间长度的部分增加

请注意，在理想设计中，参数估计值的标准误差将计算如下：

$$SE_{Ideal}(\hat{\beta}) = (\sigma/\sqrt{n})\mathbf{I}_n$$

置信区间的长度由标准误差决定。置信区间长度的部分增加是给定设计的标准误差与理想设计的标准误差之间的差值除以理想设计标准误差之后的值。

具体而言，对于第 i 个参数，按以下方式定义 “置信区间长度的部分增加”：

$$FI = \frac{\sigma\sqrt{(\mathbf{X}'\mathbf{X}_{ii})^{-1}} - (\sigma/(\sqrt{n}))}{(\sigma/\sqrt{n})} = \sqrt{n(\mathbf{X}'\mathbf{X}_{ii})^{-1}} - 1$$

其中

σ^2 是未知的响应方差，

$\mathbf{X}$ 是给定设计的模型矩阵 （在 “技术细节”附录中第 735 页的 “别名矩阵”中定义），

$(\mathbf{X}'\mathbf{X})^{-1}_{ii}$ 是 $(\mathbf{X}'\mathbf{X})^{-1}$ 的第 i 个对角线元素，

n 是试验次数。

估计的相对标准误差

“估计的相对标准误差”提供参数估计值的标准差与误差标准差的比值。这些值指示模型参数估计值的标准误差相对于误差标准差有多大。对于第 i 个参数估计值，按以下方式定义 “估计的相对标准误差”：

$$SE = \sqrt{(\mathbf{X}'\mathbf{X})^{-1}_{ii}}$$

其中

$\mathbf{X}$ 是 “技术细节”附录中第 735 页的 “别名矩阵”中定义的模型矩阵，

$(\mathbf{X}'\mathbf{X})^{-1}_{ii}$ 是 $(\mathbf{X}'\mathbf{X})^{-1}$ 的第 i 个对角线元素。

别名矩阵

“别名矩阵”解决不包括在模型中的项如何影响模型项的估计值的问题（若这些项确实是活跃的）。在“别名项”分级显示项中列出可能的活跃效应，它们不在您的假设模型中，但是可能使模型项的估计值偏倚。“别名矩阵”条目表示别名项效应给模型参数带来的偏倚程度。请参见第 407 页的“别名项”。

“别名矩阵”的行是对应于“模型”分级显示项中所列模型效应的项。列是对应于“别名项”分级显示项中所列效应的项。给定的行和列中的条目指示别名项在多大程度上影响对应于模型项的参数估计值。

在评估设计时，您理想上希望相对于“别名矩阵”中的任何条目，符合两种情形之一：一种情况是条目很小，另一种情况是条目不小，但别名项的效应很小，以至于偏倚很小。若您怀疑别名项的效应可能很大，则该项应包括在模型中或您应考虑别名最优设计。

有关计算别名矩阵的详细信息，请参见“技术细节”附录中第 735 页的“别名矩阵”。另见 Lekivetz, R. (2014)。

请注意以下几点：

- 若设计对于假设模型而言正交，则别名矩阵中的相关性对应于相关性色图中的绝对相关性。
- 根据设计的复杂性，别名矩阵条目可能大于 1 或小于 -1。

别名矩阵示例

考虑 Design Experiment 文件夹下的 Coffee Data.jmp 样本数据表。该设计假定一个主效应模型。您可以通过运行数据表中的“模型”脚本来进行查看。这样，在“评估设计”窗口的“模型”分级显示项中，只显示“截距”和五个主效应。“别名项”分级显示项包含双因子交互作用。

图 15.23 Coffee Data.jmp 的别名矩阵

别名矩阵

效应	研磨*温度	研磨*时间	研磨*咖啡豆的量	温度*时间	温度*咖啡豆的量	时间*咖啡豆的量
截距	0	0	0	0	0	0
研磨	0	0	0	0.333	-0.33	-0.33
温度	0	0.333	-0.33	0	0	-0.33
时间	0.333	0	-0.33	0	-0.33	0
咖啡豆的量	-0.33	-0.33	0	-0.33	0	0
测量地点 1	-0.41	0	0	0	0	0.816
测量地点 2	0.707	0	0	0	0	0

“别名矩阵”在第一列中显示“模型”项，用来定义行。“别名项”中的双因子交互作用在顶部列出，用来定义列。考虑以模型效应**温度**为例。若**研磨*时间**交互作用为唯一活跃的双因子交互作用，则**温度**系数估计值的偏倚量为**研磨*时间**效应的真实值的 0.333 倍。若其他交互作用是活跃的，则“别名矩阵”中的值指示**温度**系数估计值导致的其他偏倚量。

相关性色图

相关性色图显示“模型”或“别名项”分级显示项中显示的任意两个效应之间的相关性绝对值。色图的单元格在图上方标识。“模型”分级显示项中的每个效应和“别名项”分级显示项中的每个效应都分别对应一个单元格。需要较小值。

默认情况下，相关性的绝对大小用白色-灰色-黑色强度颜色主题表示。通常来说，优秀设计的色图显示对角线外有很多白色，这表示不同项之间是正交的或相关性很小。效应之间的绝对相关性很大会导致估计值的标准误差膨胀。

要查看两个效应之间相关性的绝对值，请悬停在相应的单元格上方。要更改颜色尺度，请点击“相关性色图”红色小三角并选择**由蓝经灰到红**。对于定制颜色尺度，在图中右击并选择**颜色主题**。要保存相关性表，请在图例下方的图右侧右击，然后选择**相关性表**。

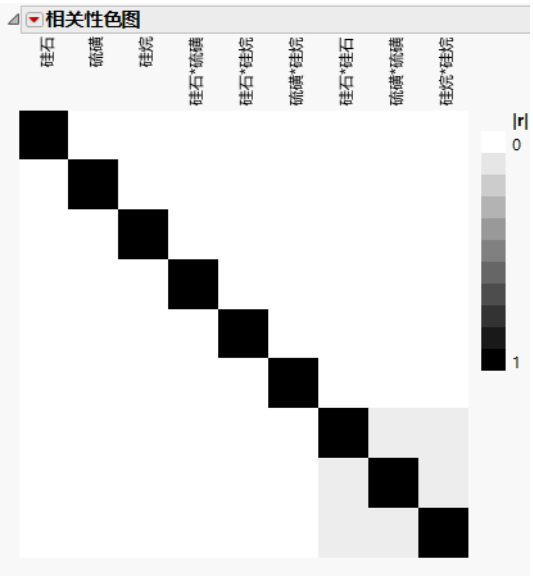
色图示例

图 15.24 显示 Design Experiment 文件夹下 Bounce Data.jmp 样本数据表的相关性色图。黑色指示绝对相关性为 1。请注意对角线上有黑色单元格，它们显示模型项与自身的相关性。

其他所有单元格为“白色”或“灰色”。灰色方块对应于二次项之间的相关性。要查看相关性，请悬停在每个灰色方块上。二次项彼此之间的绝对相关性很小，为 0.0714。

从相关性角度看，这是一个好设计。效应高度相关时，更难确定哪个效应对响应有影响。

图 15.24 相关性色图



提示：要保存相关性表，请在图例下方的图右侧右击，然后选择**相关性表**。

设计诊断

“设计诊断”分级显示项显示 *D*、*G* 和 *A* 效率以及预测的平均方差。若设计包含设置为“困难”或“极难”更改的因子或包含“可估计性”指定为“若可能”的效应，则不显示这些诊断。

若是从“实验设计”平台访问“设计诊断”，而不是从“评估设计”访问“设计诊断”，则“设计创建时间”会给出创建设计所需的时间。从“评估设计”平台访问“设计诊断”时，“设计创建时间”给出“评估设计”平台计算结果所需的时间。

图 15.25 显示 Design Experiment 文件夹下 Bounce Data.jmp 样本数据表的“设计诊断”分级显示项。

图 15.25 “设计诊断”分级显示项

警告：效率测度不应由自身来解释。但是可以使用它们来比较设计。对于给定的两个设计，效率测度值更高的设计好一些。尽管任何准则的最高效率均为 100，但是 100% 的效率对于很多设计问题来说是不可能实现的。

符号

下文给出的效率测度说明中使用以下符号：

- **X** 是模型矩阵
- n 是设计中的试验次数
- p 是模型中的项数（包括截距）
- $Var(\hat{y}|\underset{\sim}{\mathbf{x}})$ 是点 $\underset{\sim}{\mathbf{x}}$ 处的相对预测方差。请参见“技术细节”附录中第 738 页的“相对预测方差”。
- $Var(\hat{y}|\underset{\sim}{\mathbf{x}})_{max}$ 是设计区域上的最大相对预测方差

D 效率

根据 D 最优性准则，设计相对理想正交设计的效率。若设计使回归系数的向量的联合置信区域最小，则设计是 D 最优的：

$$D效率 = 100\left(\frac{1}{n}|\mathbf{X}'\mathbf{X}|^{1/p}\right)$$

G 效率

根据 G 最优性准则，设计相对理想正交设计的效率。若设计使设计区域上的最大预测方差最小，则设计是 G 最优的：

$$G效率 = 100p/(nVar(\hat{y}|\underset{\sim}{\mathbf{x}})_{max})$$

用 D 表示设计区域，

$$Var(\hat{y}|\underset{\sim}{\mathbf{x}})_{max} = \underset{\underset{\sim}{\mathbf{x}} \text{ 在 D 中}}{maximum}[\underset{\sim}{\mathbf{x}}'(\mathbf{X}'\mathbf{X})^{-1}\underset{\sim}{\mathbf{x}}]$$

注意：使用设计空间的 Monte Carlo 抽样计算 G 效率。因此，同一设计的再次计算可能略有不同。

A 效率

根据 A 最优性准则，设计相对理想正交设计的效率。若设计使回归系数的方差之和最小，则设计是 A 最优的：

$$A\text{效率} = 100p/(n\text{Trace}(\mathbf{X}'\mathbf{X})^{-1})$$

预测的平均方差

在设计空间中的点 $\underset{\sim}{\mathbf{x}}$ 处，按以下方式定义相对预测方差：

$$Var(\hat{y}|\underset{\sim}{\mathbf{x}}) = \underset{\sim}{\mathbf{x}}'(\mathbf{X}'\mathbf{X})^{-1}\underset{\sim}{\mathbf{x}}$$

这是预测方差除以误差方差得到的值。有关计算的详细信息，请参见 Goos and Jones (2011) 中的第 4.3.5 节。

注意：若设计区域受到线性约束或不被允许的组合的约束，则预测的平均方差将根据“设计空间比例”图中的预测方差值计算得出。

设计创建时间

“设计创建时间”给出“评估设计”平台计算结果所需的时间。

“评估设计”选项

“评估设计”红色小三角菜单包含以下选项：

高级选项 > 裂区方差比 指定随机整区和子区 （若存在）方差与误差方差之间的比值。设置该值之前，您必须为裂区设计定义难以更改的因子，或是为裂 - 裂区设计定义难以更改和极难更改的因子。然后，您可以为方差比输入一个或两个正数，具体取决于您指定的是裂区设计还是裂 - 裂区设计。

高级选项 > 设置功效的 Delta 指定要检测的差值，该值将应用到 “功效分析”报表中的 “预期系数”。对于连续效应，预期系数的值设置为 Delta/2。对于分类效应，它们为 Delta/2 和 –Delta/2 交替出现。有关功效分析的详细信息，请参见第 408 页的 “功效分析”。

默认情况下，Delta 设置为 2。这样，预期系数的默认值对于连续效应为 1，对于分类效应则为 1 和 –1 交替出现。当 Delta 为 2 时作为预期系数输入的默认值确保以下属性：

– 数值效应的功效计算假定当因子从设计区域的最低设置变为最高设置时，由于线性主效应导致响应均值变化了 Delta。

- 与两水平的分类因子相关的参数的功效计算假定因子水平变化而导致响应均值相应变化了 Delta。
- 具有两个以上水平的分类效应的功效计算基于多自由度 *F* 检验，其原假设为所有水平具有相同的响应均值。在预期系数所确定的响应均值处计算功效。预期系数的各种配置可以定义像 Delta 这么大的水平差值。但是，由于其他水平的预期系数的影响，这类配置的功效值将有所不同。

将脚本保存至脚本窗口 创建再现 “评估设计”窗口的脚本并将其放到打开的脚本窗口中。

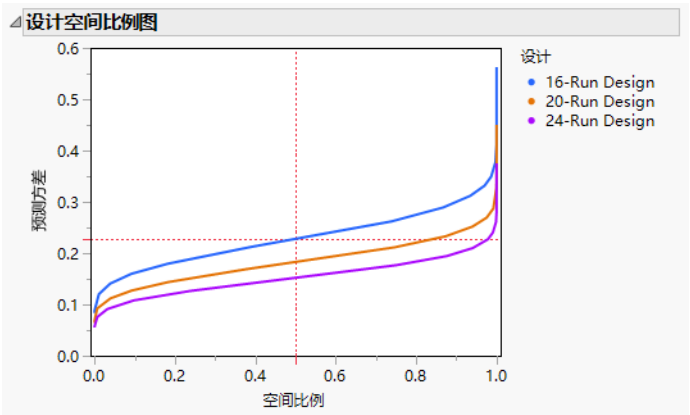
第 16 章

比较设计

同时比较和评估设计

“比较设计”平台可将多达四个设计与参考设计进行比较。用于探索、评估和比较设计性能。诊断显示各个设计相对于彼此的性能如何，以及在绝对意义上的性能如何。要根据您的特定需要比较设计，您可以更改假设模型和别名项列表中的项。

图 16.1 比较具有不同试验规模作的三个设计



目录

“比较设计”平台概述 427
比较设计的示例 427
试验规模相同的设计 427
试验规模不同的设计 432
具有不同整区数的裂区设计 435
启动“比较设计”平台 438
“比较设计”窗口：指定模型和别名项 440
参考设计 441
因子 441
模型 441
别名项 441
“比较设计”窗口：设计评估 441
功效分析 442
预测方差刻画 445
设计空间比例图 446
相对估计效率 447
别名矩阵汇总 449
绝对相关性 450
设计诊断 452
比较设计选项 454

“比较设计”平台概述

“比较设计”平台是扩展了的“评估设计”平台，它支持您轻松地将多达四个设计与参考设计进行比较。要比较一个或两个设计相对于其他设计的性能，请选择一个被视为基准设计的参考设计。您可以在“模型”分级显示项中指定效应，在“别名项”分级显示项中指定关注的效应。

“设计评估”报表显示诊断结果和覆盖这些区域的图：

- 功效分析
- 预测方差
- 设计空间比例
- 相对估计效率
- 别名矩阵诊断
- 效应之间的相关性（包括混杂）
- 整体设计的相对效率测度

比较设计的示例

- “试验规模相同的设计”
- “试验规模不同的设计”
- “具有不同整区数的裂区设计”

试验规模相同的设计

在本例中，您要比较针对六个因子（每个因子涉及 13 个试验）的两个设计。其中一个是一个 12 次试验的 Plackett-Burman (PB) 设计，该设计使用一个中心点扩充。另一个设计是确定性筛选设计 (DSD)。

比较仅限于主效应

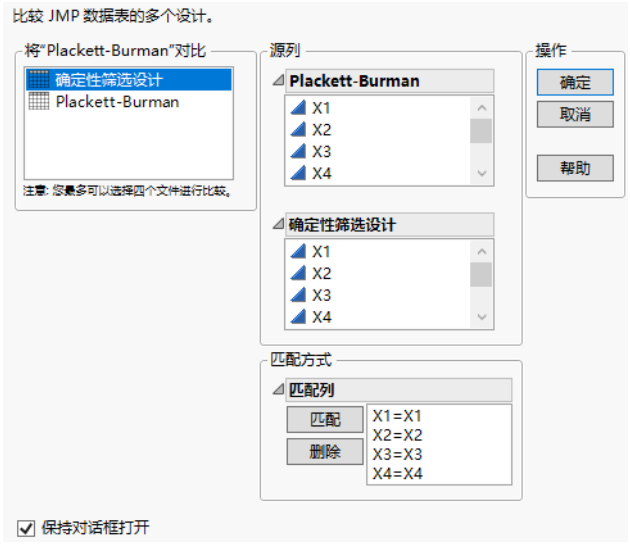
首先，比较两个设计时假定要估计的模型仅包含主效应。

1. 选择**帮助 > 样本数据**，点击**打开样本脚本目录**，然后选择 Compare Same Run Size.jsl。
2. 在脚本窗口中右击并选择**运行脚本**。

   随之构造两个 13 次试验设计表：确定性筛选设计和 Plackett-Burman 设计。您想要比较这两种设计。由于 Plackett-Burman 表是活跃的，它就是与确定性筛选设计 (DSD) 作比较的参考设计。
3. 在 Plackett-Burman 数据表中，选择**实验设计 > 设计诊断 > 比较设计**。

4. 从**将 “Plackett-Burman” 对比**列表中，选择**确定性筛选设计**。
5. 在 “Plackett-Burman” 面板和 “确定性筛选设计” 面板中，从 X1 一直选择到 X6。
6. 打开 “匹配列” 分级显示项，然后点击**匹配**。

图 16.2 包含 “匹配列” 的启动窗口

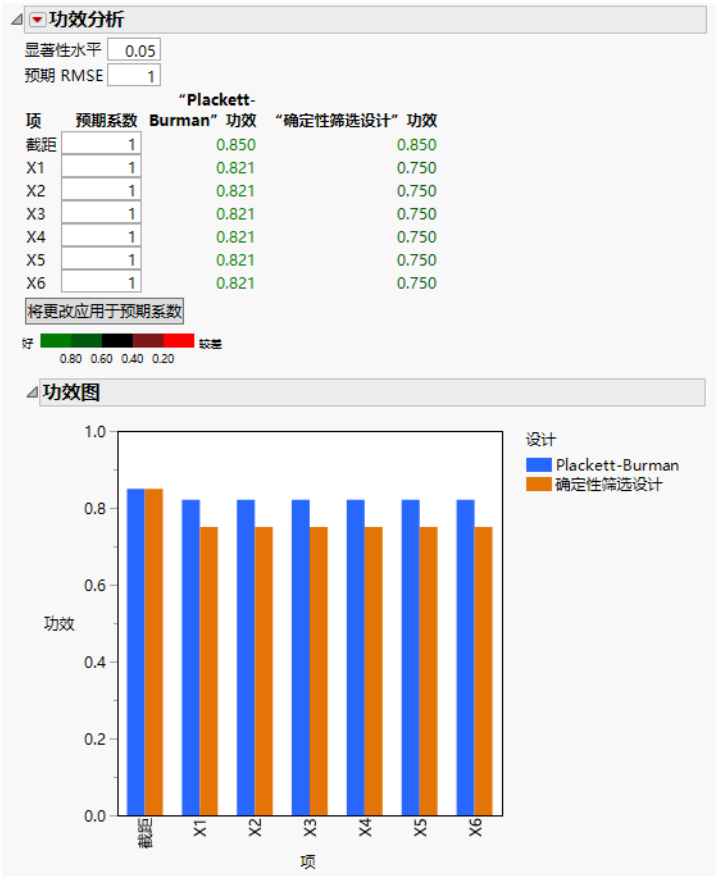


这将定义您的两个设计中因子之间的对应关系。

7. 点击**确定**。

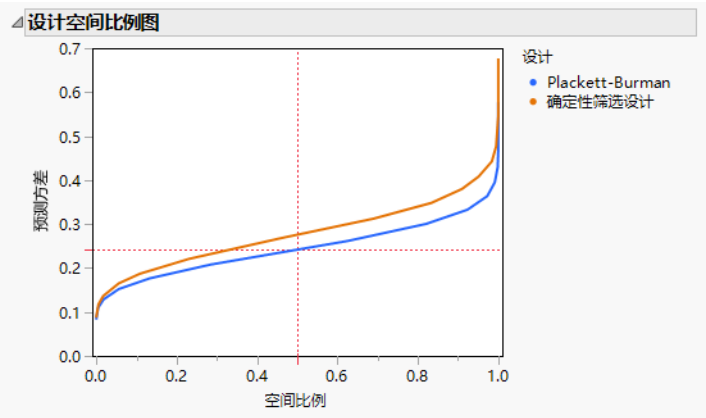
参考设计是 Plackett-Burman 设计。在 “设计评估” 分级显示项中，比较量度用来比较 PB 与 DSD。这些设计将相对于功效、预测方差、估计效率、别名关系和设计效率等测度作比较。

图 16.3 PB 与 DSD 比较的功效分析



在功效、预测方差和估计效率等方面，PB 设计优于 DSD。图 16.3 显示“功效分析”报表，其中包含显著性水平、预期 RMSE 和系数的默认设置。对于针对主效应的试验，PB 设计的功效比 DSD 高。

图 16.4 PB 和 DSD 比较的设计空间比例图



设计空间比例图指示 PB 设计在整个设计空间中的预测方差小于 DSD。

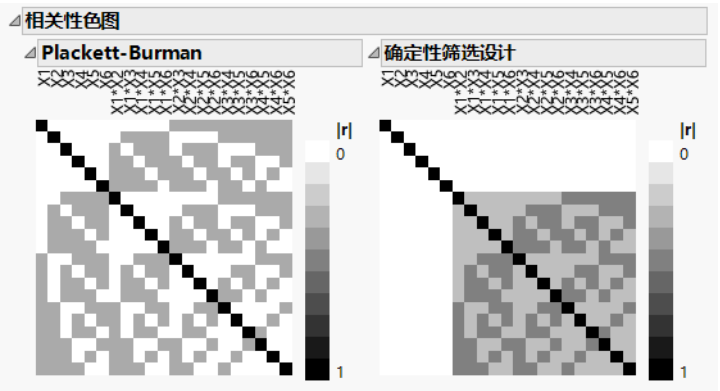
您可以推断出：若怀疑只有主效应活跃，则 PB 设计更可取。

双因子交互作用方面的比较

现在假定您怀疑某些双因子交互作用可能活跃。下面的分析显示：若这些双因子交互作用实际上是活跃的，则 PB 设计可能不如 DSD。

1. 在“绝对相关性”报表中，打开“相关性色图”报表和其下的色图报表。

图 16.5 PB 与 DSD 比较的色图



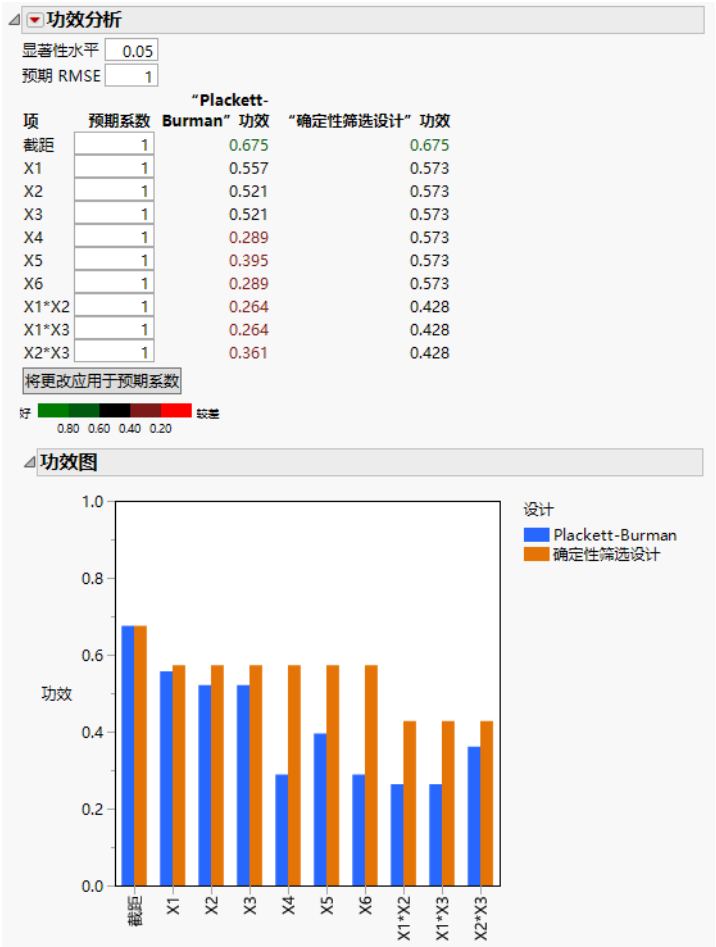
相关性色图显示 PB 设计中主效应与双因子交互作用之间存在别名关系。相比之下，DSD 不会使主效应与双因子交互作用之间存在别名关系。

若要在某些双因子交互作用活跃时获取关于设计之间如何比较的更深入信息，请在“模型”分级显示项中添加双因子交互作用。

2. 在“因子”分级显示项中，从 X1 一直选择到 X3。

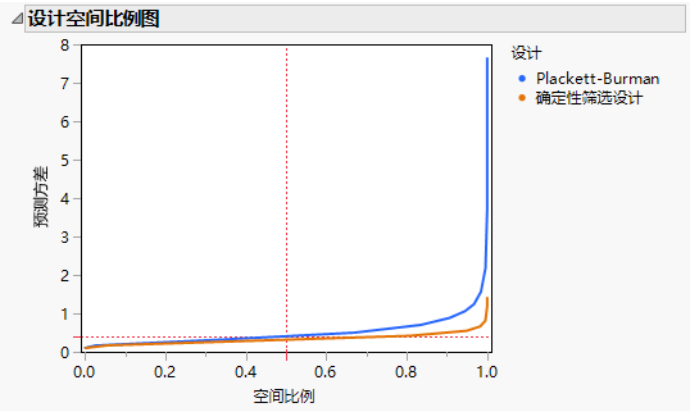
3. 在 “模型”分级显示项中，选择**交互作用 > 二次**。

图 16.6 包含交互作用的 PB 和 DSD 比较的功效分析



“项”列表显示三个双因子交互作用。若这些双因子交互作用是活跃的，则 DSD 在所有效应的功效方面的表现优于 PB。

图 16.7 包含交互作用的 PB 和 DSD 比较的预测方差



当模型中包含三个交互作用时， DSD 在预测方差方面也优于 PB 设计。您可以检查其他报表，查看 DSD 在可能存在活跃的交互作用的情况下是否是首选设计。

试验规模不同的设计

在本例中，比较试验规模分别为 16、20 和 24 的三个设计。这些设计是为主效应模型构造的。使用 “比较设计”平台确定使用较大试验规模可能带来的益处是否值得花费额外的资源成本。

1. 选择**帮助 > 样本数据**，点击**打开样本脚本目录**，然后选择 Compare Three Run Sizes.jsl。
2. 在脚本窗口中右击并选择**运行脚本**。

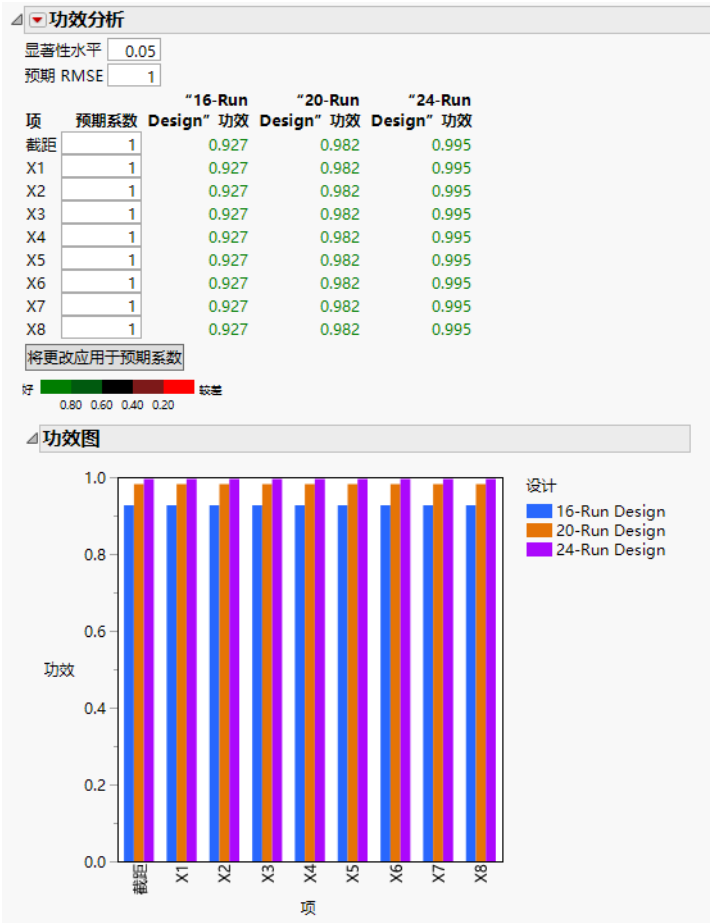
   三个设计表是使用 “定制设计”构造的，“模型”分级显示项中只输入了主效应：

   – 16-Run Design
   – 20-Run Design
   – 24-Run Design

   您想要比较这三种设计。请注意 16-Run Design 表是活跃的。
3. 在 16-Run Design 表中，选择**实验设计 > 设计诊断 > 比较设计**。
4. 从**将 “16-Run Design”对比**列表中，选择 20-Run Design 和 24-Run Design。

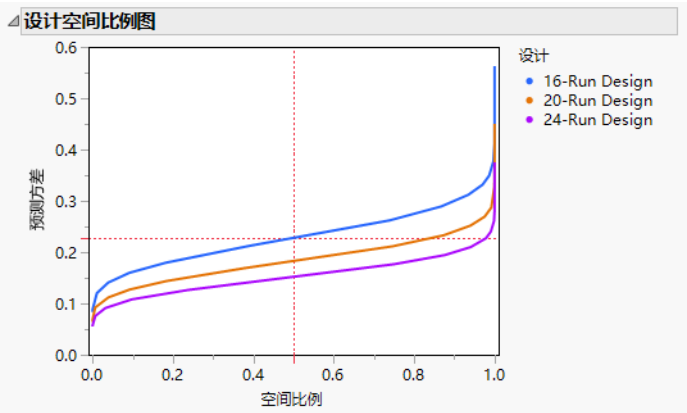
   以上每个设计的面板都添加至启动窗口。 JMP 按照各列出现在三个设计表中的顺序自动匹配各列。
5. 点击**确定**。

图 16.8 功效分析比较



若系数具有相近的预期 RMSE，则全部三种设计检测主效应的功效都很高。

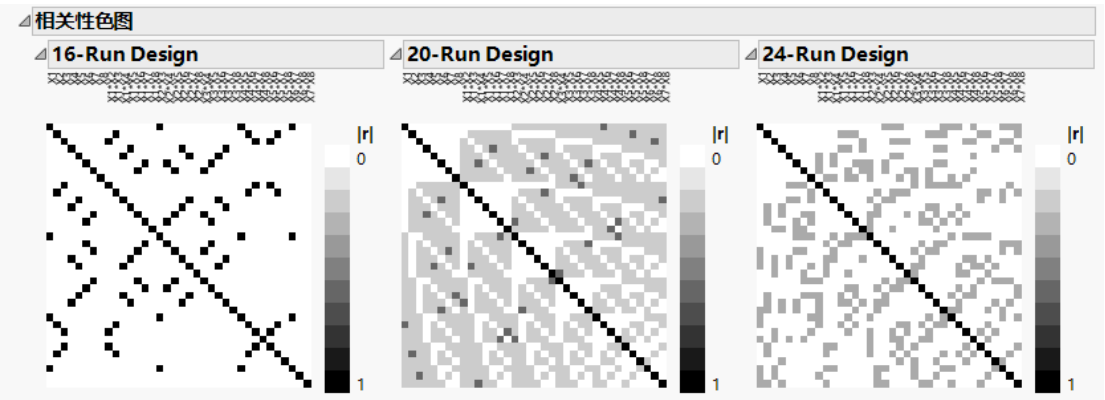
图 16.9 设计空间比例的比较



正如预期的那样， 24 次试验设计 (24-run design) 在整个设计空间的预测方差方面优于另外两种设计；而 20 次试验设计 (20-run design) 又优于 16 次试验设计 (16-run design)。

6. 在 “绝对相关性”报表中，打开 “相关性色图”报表和其下的三个色图报表。

图 16.10 相关性色图比较

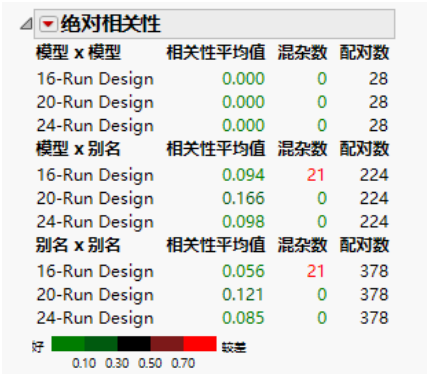


对于 16 次试验设计，“相关性色图”指示某些主效应与某些双因子交互作用混杂，而且双因子交互作用之间也有混杂。

对于 20 次试验设计，“相关性色图”指示某些主效应与某些双因子交互作用之间以及某些双因子交互作用之间都存在较大的相关性。

24 次试验设计仅显示主效应与双因子交互作用以及双因子交互作用之间有中等程度的相关性。

图 16.11 绝对相关性比较



绝对相关性

模型 x 模型	相关性平均值	混杂数	配对数
16-Run Design	0.000	0	28
20-Run Design	0.000	0	28
24-Run Design	0.000	0	28
模型 x 别名	**相关性平均值**	**混杂数**	**配对数**
16-Run Design	0.094	21	224
20-Run Design	0.166	0	224
24-Run Design	0.098	0	224
别名 x 别名	**相关性平均值**	**混杂数**	**配对数**
16-Run Design	0.056	21	378
20-Run Design	0.121	0	378
24-Run Design	0.085	0	378

“绝对相关性”表汇总了“相关性色图”中显示的信息。回想一下，全部三种设计的模型都仅包含主效应，并且别名矩阵包含双因子交互作用。

对于 16 次试验设计，该表的“模型 x 别名”部分指示主效应与双因子交互作用有 9 个混杂。“别名 x 别名”部分指示有 6 个双因子交互作用混杂。

图 16.12 设计诊断比较

设计诊断

	“16-Run Design”相对于“20-Run Design”的效率	“16-Run Design”相对于“24-Run Design”的效率
D 效率	0.800	0.667
G 效率	0.800	0.667
A 效率	0.800	0.667
I 效率	0.800	0.667
额外试验规模	-4	-8

“设计诊断”报表依据若干效率测度将 16 次试验设计与 20 次试验设计和 24 次试验设计都进行了效率比较。超过 1 的相对效率值指示参考设计对于给定测度而言更可取。小于 1 的值指示与参考设计比较的设计更可取。16 次试验设计的效率在所有量度上都比其他两种设计低，这表示较大规模的设计更可取。

具有不同整区数的裂区设计

在本例中，比较具有不同整区数的两个裂区设计。这些设计针对三个因子：

- 一个连续的难以更改的因子
- 一个连续的易于更改的因子
- 一个三水平分类易于更改的因子

这些设计包括假定模型中的所有双因子交互作用。您可以最多进行 20 次试验，而且您想要使用 4 或 8 个整区进行比较。

启动比较设计

1. 选择**帮助** > **样本数据**，点击**打开样本脚本目录**，然后选择 Compare Split Plots.jsl。
2. 在脚本窗口中右击并选择**运行脚本**。

   使用定制设计构造了两个设计表：

   – 4 Whole Plots
   – 8 Whole Plots

   您想要比较这两种设计。请注意 4 Whole Plots 表是活跃的。
3. 在 4 Whole Plots 表中，选择**实验设计** > **设计诊断** > **比较设计**。
4. 从**将 “4 Whole Plots” 对比**列表中，选择 8 Whole Plots。

   针对该设计的面板添加到启动窗口中。 JMP 按照各列出现在两个设计表中的顺序自动匹配各列。

图 16.13 已完成的启动窗口

5. 点击**确定**。
6. 打开 “参考设计：20 次试验 ‘4 Whole Plots’” 下的**匹配方式**分级显示项。

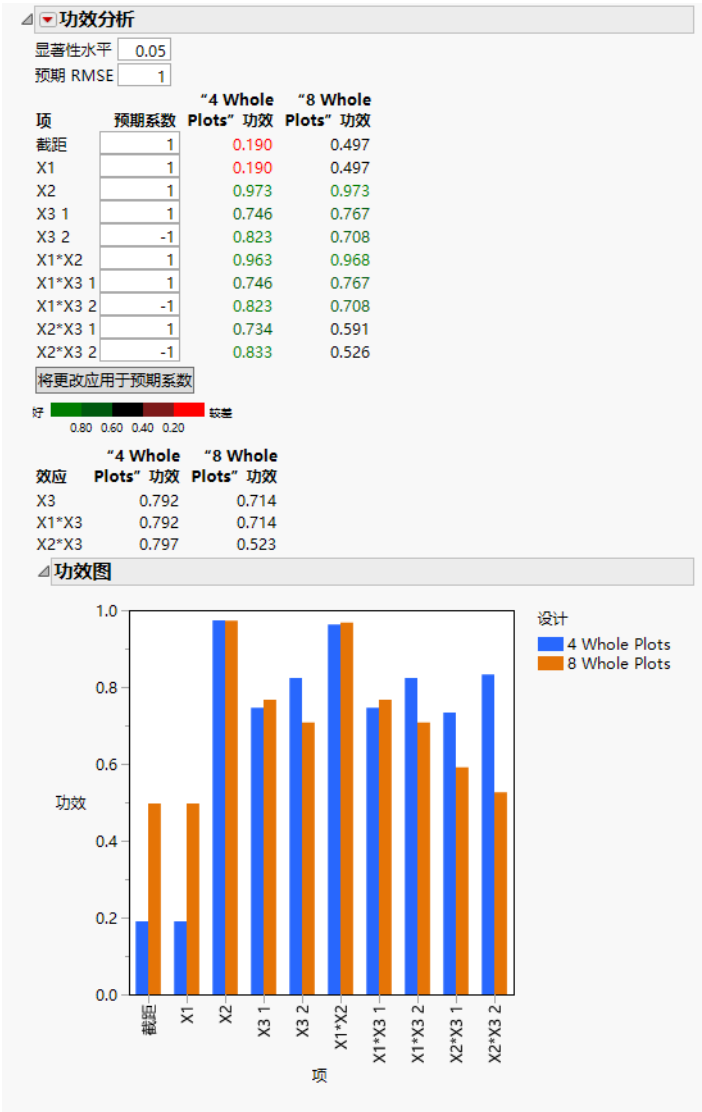
图 16.14 裂区设计的匹配方式

请注意，“整区”列输入为设计的一部分。这是必需的，因为“比较设计”需要了解整区结构。

检查报表

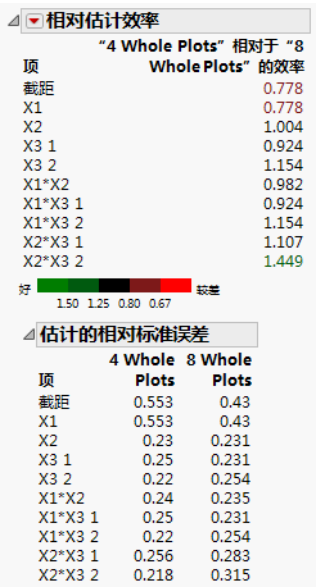
“设计评估”报表提供用于比较两个设计的多种诊断信息。

图 16.15 两个裂区设计的功效分析



“功效分析”报表显示：整区因子 X1 的功效对于 4 个整区设计 (0.19) 比起对于 8 个整区设计 (0.497) 要小得多。不过，4 个整区设计检测裂区效应的功效较高，特别是两个裂区因子的交互作用 X2*X3（0.797 对比 0.523）。请注意，色条和图例下面给出了组合效应 X2*X3 的功效。

图 16.16 比较裂区设计的相对估计效率



相对估计效率

项	“4 Whole Plots” 相对于 “8 Whole Plots” 的效率
截距	0.778
X1	0.778
X2	1.004
X3 1	0.924
X3 2	1.154
X1*X2	0.982
X1*X3 1	0.924
X1*X3 2	1.154
X2*X3 1	1.107
X2*X3 2	1.449

估计的相对标准误差

项	4 Whole Plots	8 Whole Plots
截距	0.553	0.43
X1	0.553	0.43
X2	0.23	0.231
X3 1	0.25	0.231
X3 2	0.22	0.254
X1*X2	0.24	0.235
X1*X3 1	0.25	0.231
X1*X3 2	0.22	0.254
X2*X3 1	0.256	0.283
X2*X3 2	0.218	0.315

“相对估计效率” 报表显示了 X1 的相对估计效率为 0.778。这指示 4 个整区设计的 X1 的标准误差比 8 个整区设计明显大得多。

打开 “估计的相对标准误差” 报表。您可以看到 4 个整区设计中的 X1 的相对标准误差为 0.553，而 8 个整区的误差为 0.43。

在 “相对估计效率” 报表中，X2*X3 2 的相对估计效率为 1.449，这表明 8 个整区设计的与 X2*X3 2 关联的参数的标准误差比 4 个整区设计明显大得多。

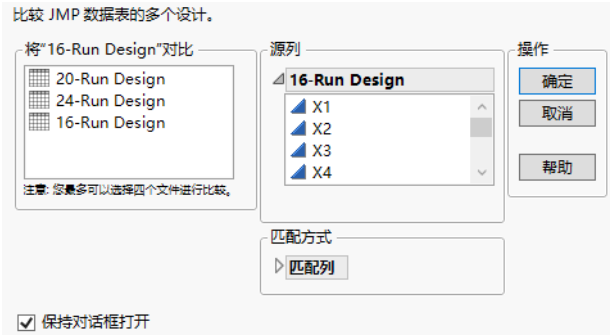
“功效分析” 报表和 “相对估计效率” 报表指示设计的选择围绕着检测整区效应 X1 的重要性。8 个整区设计为您提供了更好的检测整区效应的机会。4 个整区设计对于检测涉及分类变量的裂区效应要好一些。

启动 “比较设计” 平台

通过选择**实验设计** > **设计诊断** > **比较设计**启动 “比较设计” 平台。所有打开的数据表都显示在左侧的列表中。活跃数据表及其列显示在 “源列” 面板中。最初的 “源列” 面板中的设计为**参考设计**，即：其他设计所比照的设计。添加与参考设计比照的设计时，其列显示在参考设计面板下的面板中。

图 16.17 显示第 432 页的 “试验规模不同的设计” 中三个设计的启动窗口。

图 16.17 “比较设计” 启动窗口



设计表选择

从左侧列表中选择多达四个设计表。

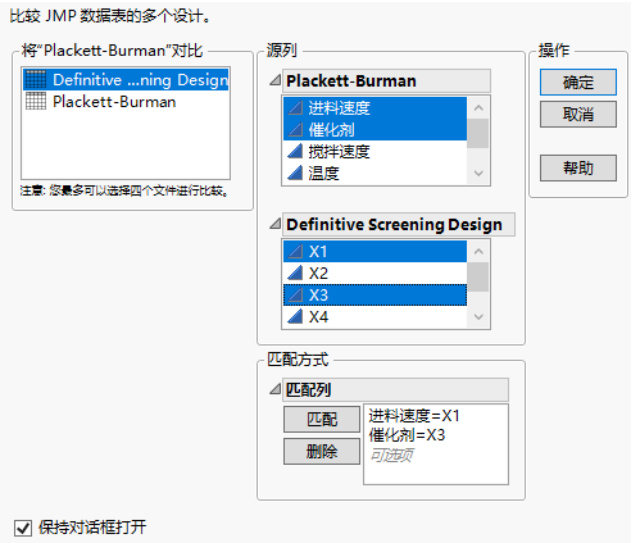
- 要将多个设计与参考设计作比较，您必须从左侧的列表中同时选择其设计表。
- 要替换 “源列” 列表中的一个或多个设计，请从左侧的列表中选择所需的一个或多个表。参考设计表下的一个或多个表将被替换。

注意：参考设计表可与自身作比较，这在探索如何将设计列分配给因子时很有用。

匹配列

指定在 “匹配列” 面板中各设计表中的哪些列相互对应。要匹配列，请在每个设计表的 “源列” 列表中选择要匹配的列，然后点击 “匹配” 。

图 16.18 选择要匹配的列



- 要匹配每个列表中的单个列，在每个列表中选择单个列，然后点击“匹配”。
- 要匹配每个列表中按正确匹配顺序显示的若干列，请在每个列表中选择它们。点击“匹配”按钮。这些列将按它们排列的顺序匹配（图 16.18）。在本例中，**进料速度**与 X1 匹配，**催化剂**与 X3 匹配。
- 若列表包含相同数量的列而您所需的匹配顺序就是列在列表中的显示顺序，您无需点击“匹配”。点击“确定”运行启动窗口后，JMP 会按照列的显示顺序自动匹配它们。您可以在报表的“匹配方式”分级显示项中检查匹配情况。

“比较设计”窗口：指定模型和别名项

“比较设计”窗口由两组分级显示项构成：

- 使用“因子”、“模型”和“别名项”分级显示项指定模型中有哪些效应，哪些效应可能活跃。
- 使用“设计评估”分级显示项中的诊断信息比较设计。您在“模型”和“别名项”分级显示项中所做的更改将在“设计评估”分级显示项中更新。

“比较设计”报表使用参考设计中的列名。

本节说明“参考设计”、“因子”、“模型”和“别名项”等分级显示项。有关“设计评估”分级显示项的说明，请参见第 441 页的“ “比较设计”窗口：设计评估”。

参考设计

参考设计窗口的名称显示在分级显示项标题中。“匹配方式”分级显示项列出您在启动窗口中输入的规格。

因子

使用“因子”分级显示项向“模型”和“别名项”列表添加效应。

“因子”分级显示项使用参考设计中的列名列出因子和编码值。由于整区和子区列并非因子，所以不显示在“因子”分级显示项中。不过，分析中要用到它们。

模型

添加或删除效应，以便针对您确信应包含在模型中的效应比较设计。“模型”分级显示项最初列出位于参考设计表的“模型”脚本中的所有要比较的设计都要评估的效应。若参考设计表中没有“模型”脚本，则“模型”分级显示项仅显示可由所有要比较的设计评估的主效应。有关如何添加和删除效应的详细信息，请参见“评估设计”一章中第 406 页的“模型”。

注意：若有任何设计超饱和（这意味着要评估的参数个数超过试验次数），则“模型”分级显示项仅列出可评估的一组效应。

别名项

添加或删除效应，以便针对可能活跃的效应比较设计。“别名项”分级显示项最初包含“模型”分级显示项中未包含的所有双因子交互作用。该分级显示项中的效应影响“别名矩阵汇总”和“绝对相关性”分级显示项中的计算。请参见第 449 页的“别名矩阵汇总”和第 450 页的“绝对相关性”。

有关如何添加和删除效应的详细信息，请参见“评估设计”一章中第 407 页的“别名项”。

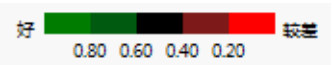
“比较设计”窗口：设计评估

- “功效分析”
- “预测方差刻画”
- “设计空间比例图”
- “相对估计效率”
- “别名矩阵汇总”
- “绝对相关性”
- “设计诊断”

颜色仪表板

有些“设计评估”分级显示项显示根据色条着色的值。这些颜色应用到诊断测度，有助您查看哪些值（和设计）反映行为的好坏。您可以编辑图例值以应用反映关于行为好坏界定的颜色。

图 16.19 颜色仪表板



通过从分级显示项的红色小三角菜单中选择以下两个选项或右击色条，您可以修改色条：

显示图例值 显示或隐藏在色条下出现的值。

编辑图例值 指定定义颜色的值。

功效分析

功效是检测到给定大小的活跃效应的概率。“功效分析”报表帮助您评估和比较您的设计检测具有实际意义的效应的能力。对于每个设计，“功效分析”报表都为“模型”分级显示项中的效应计算检验功效。

“功效分析”报表提供针对各个模型参数和整体效应的检验功效。其中还提供一个“功效图”和一个“样本大小-功效”图。

功效取决于试验次数、显著性水平和估计的误差变异。有关如何计算功效的详细信息，请参见“技术细节”附录中第 736 页的“功效计算”。

图 16.20 三个设计的 “功效分析”分级显示项

图 16.20 显示在第 432 页的“试验规模不同的设计”中构造的三个设计的“功效分析”分级显示项。两个双因子交互作用已添加到“模型”分级显示项。

“功效分析”报表

为“显著性水平”和“预期 RMSE”指定值时，这些值用于计算模型参数的检验功效。输入系数值，这些值反映您想要检测的“预期系数”的差值。要更新所有设计的结果，请点击“将更改应用于预期系数”。

显著性水平 在“无效应”假设为真时拒绝该假设的概率。功效计算在您输入值后立即更新。

预期 RMSE 误差变异的平方根的估计值。功效计算在您输入值后立即更新。

功效值将根据“将更改应用于预期系数”按钮下显示的颜色梯度来着色。您可以使用“功效分析”红色小三角菜单中的选项控制颜色图例。请参见第 442 页的“颜色仪表板”。

有关功效图的详细信息，请参见第 444 页的“功效图”。

注意：若设计超饱和（这意味着要评估的参数个数超过试验次数），则“模型分析”分级显示项仅列出可评估的一组效应。

针对各个参数的检验

“项”列包含模型项的列表。对于每个项，“预期系数”列包含该项的值。“功效”值是以下检验的功效：在给定设计和“模型”分级显示项中的项的前提下，若预期系数给定了系数的真实值，该项的系数为 0。

项 与要检测的系数相关的模型项。

预期系数 与模型项相关的系数的值。该值用于功效的计算。当您在“预期系数”列中设置新值时，点击**将更改应用于预期系数**可更新“功效”计算。

注意：预期系数对于连续效应的默认值为 1。它们对于分类效应具有交替值 1 和 –1。

功效 当指定的预期系数给出真实系数值时拒绝“无效应”原假设的概率。

- 对于与数值因子相关的系数，从高设置变为低设置时响应均值的变化量（基于模型）是系数值的两倍。
- 对于与分类因子相关的系数，因子各个水平上响应均值的变化量（基于模型）是预期系数的绝对值的两倍。有关分类因子的功效的详细信息，请参见第 444 页的“针对具有两个以上水平的分类效应的检验”

计算中使用指定的“显著性水平”和“预期 RMSE”。有关功效计算的详细信息，请参见“技术细节”附录中第 736 页的“单个参数的功效”。

将更改应用于预期系数 当您在“预期系数”列中设置新值时，点击**将更改应用于预期系数**可更新“功效”值。

针对具有两个以上水平的分类效应的检验

若您的模型包含具有两个以上水平的分类效应，则以下列显示在 “将更改应用于预期系数”按钮下方：

效应 分类效应。

功效 检验无效应的功效计算。检验的原假设是：与效应对应的所有模型参数为零。要检测的差值由“预期系数”列中的值定义，这些值对应于效应的模型项。功效计算反映由预期系数决定的响应均值的差值。

计算中使用指定的 “显著性水平”和 “预期 RMSE” 。有关功效计算的详细信息，请参见“技术细节”附录中第 737 页的 “分类效应的功效”。

功效图

功效图以图形方式显示功效分析的功效值。该图在并排显示的条形图中显示每个效应和每个设计的功效。

图 16.21 三个设计的功效图

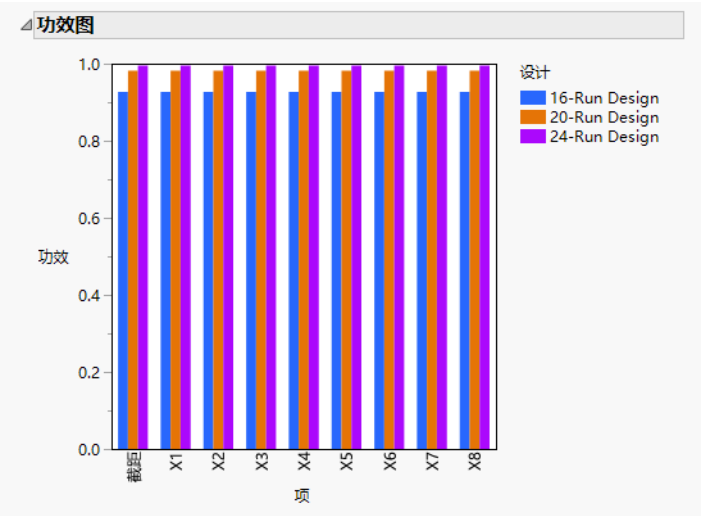
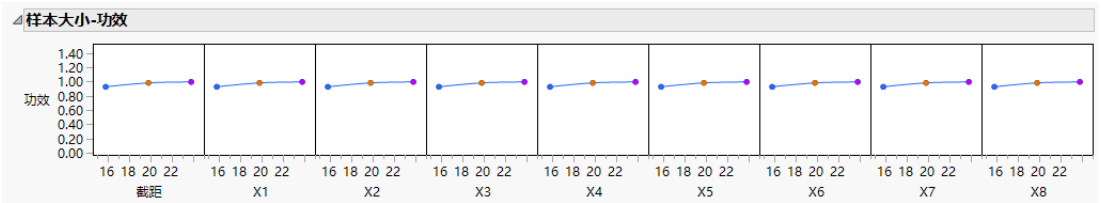


图 16.21 中的功效图针对第 432 页的 “试验规模不同的设计”中构造的三个设计的主效应。

样本大小 - 功效

仅当您比较的设计的试验规模不同时，才会显示“样本大小-功效”图。这些图支持您查看样本大小如何影响模型中每个效应的功效。它还传达与“功效图”中相同的信息，只不过形式不同。整数样本大小处的功效值使用线段连接。

图 16.22 三个设计的 “样本大小 - 功效”刻画器



预测方差刻画

“预测方差刻画”分级显示项显示要比较的每个设计的相对预测方差的刻画器。每个图都将预测的相对方差标绘为每个因子的函数（固定其他因子的值）。

要为参考设计查找设计空间上相对预测方差的最大值，请从“预测方差刻画”旁边的红色小三角中选择**最大化方差**选项。请参见第 446 页的“最大化方差”。

图 16.23 三个设计的预测方差刻画

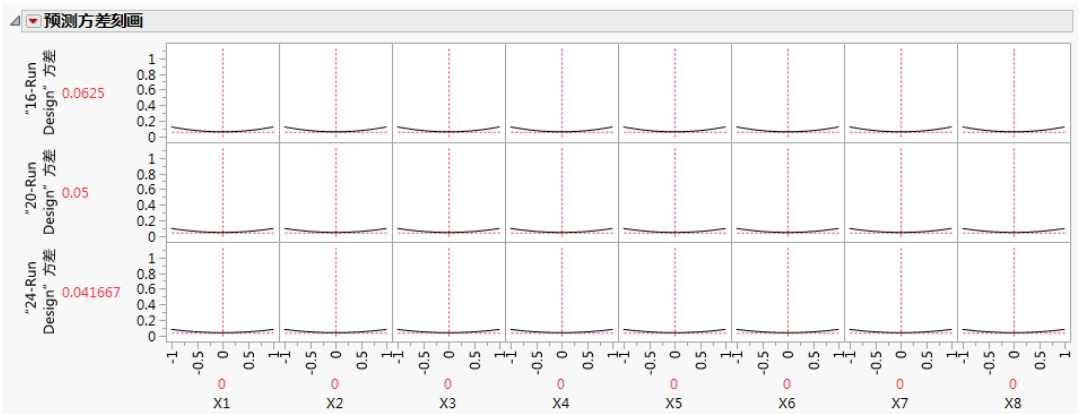


图 16.23 中的“预测方差刻画”图针对第 432 页的“试验规模不同的设计”中构造的三个设计。两个双因子交互作用 X1*X3 和 X2*X3 已添加至“模型”分级显示项。图中每个连续因子的初始值是其设计设置的中点值。左侧的方差值指示，随着试验次数的增加，方差在中心点处减小。

相对预测方差

对于因子的给定设置，预测方差为误差方差与依赖于设计和因子设置的量的乘积。在执行实验前，误差方差是未知的，因此预测方差也是未知的。但是，预测方差与误差方差的比值不是误差方差的函数。该比值称为**相对预测方差**，它仅依赖于设计和因子设置。因此，可以在采集数据前计算预测的相对方差。请参见“技术细节”附录中第 738 页的“相对预测方差”。

在执行实验并拟合最小二乘模型后，您可以使用模型拟合的均方误差 (MSE) 估计误差方差。可以通过乘以任意设置处预测的相对方差来估计该设置处预测的实际方差。

理想情况下，预测方差在整个设计空间中都较小。通常，误差方差随着样本大小的增加而减小。在比较设计中，具有较低预测方差的设计一般而言更可取。

最大化方差

您还可以根据最大相对预测方差来评估设计或比较设计。从“预测方差刻画”旁边的红色小三角中选择**最大化方差**选项。JMP 使用使相对预测方差最大化的意愿函数。预测方差刻画中的“方差”值是相对预测方差的最差（从设计角度看最不想要的）值。

图 16.24 显示三个设计的最大方差的预测方差刻画

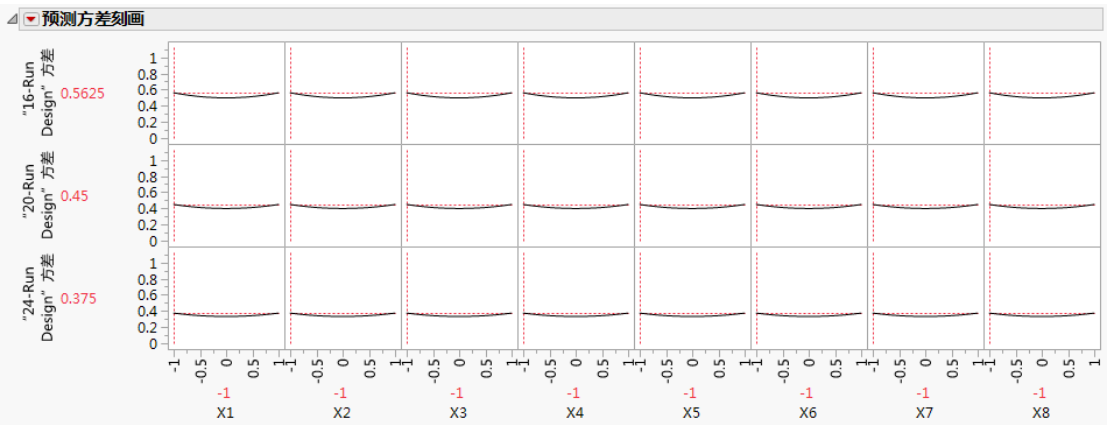


图 16.24 显示在为第 432 页的“试验规模不同的设计”中构造的三个设计选择“最大化方差”后的预测方差刻画。如预期的那样，最大相对预测方差随着试验规模的增大而减小。该图还显示给出最差情况的相对方差时的因子值。但是，请记住很多设置都可以得到这个最大相对方差。

设计空间比例图

设计空间比例图显示相对预测方差位于给定值之下的设计空间的比例。

图 16.25 三个设计的设计空间比例图

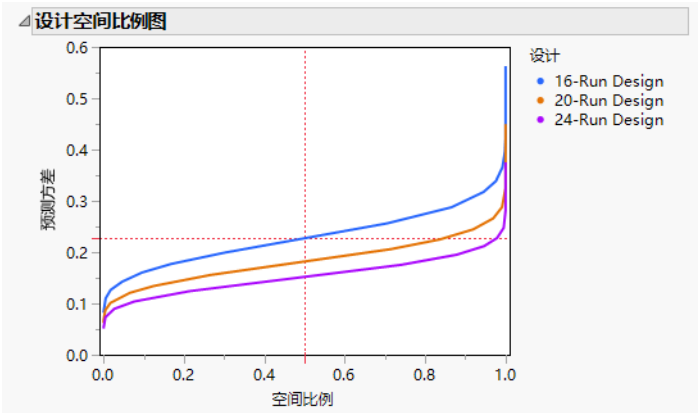


图 16.25 显示在第 432 页的 “试验规模不同的设计”中构造的三个设计的设计空间比例图。请注意以下几点：

- 图中的 *X* 轴表示设计空间的比例，范围为从 0 到 100%。
- *Y* 轴表示相对预测方差值。
- 对于落在给定曲线上的点 (x, y)，值 x 是其方差小于等于 y 的设计空间的比例。
- 使用红点构成的十字准线标记 50% 的参考设计设计空间的相对预测方差的限值。

图 16.25 显示 24 次试验设计的相对预测方差均小于另外两个设计的相对预测方差。20 次试验设计的预测方差均小于 16 次试验设计的预测方差。使用红点构成的十字准线指示：在大约 50% 的设计空间上，20 次试验设计的相对预测方差比 0.23（近似值）小。

您可以使用十字准线工具来找到对应于任意空间比例值的最大相对预测方差。例如，使用十字准线工具可看到：对于 24 次试验设计，90% 的预测方差值都在大约 0.20 以下。

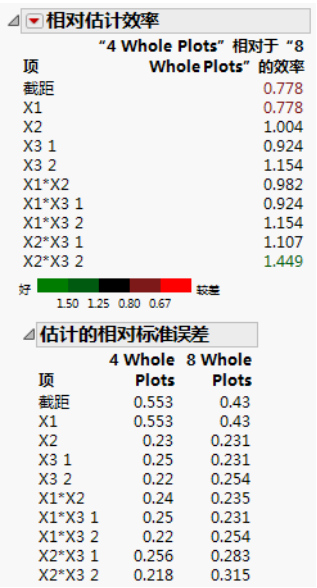
注意：同一设计的图可能略有不同，因为构造设计空间比例图时使用了设计空间的 Monte Carlo 抽样。

相对估计效率

“相对估计效率”报表针对假设模型中参数的参数估计值的标准误差对设计进行比较。标准误差控制参数估计值的置信区间长度。该报表提供效率比和相对标准误差。

根据相对估计效率值表下方显示的颜色梯度对相对估计效率值着色。您可以使用“相对估计效率”红色小三角菜单中的选项控制颜色图例。请参见第 442 页的“颜色仪表板”。

图 16.26 比较两个裂区设计的相对估计效率



相对估计效率

项	“4 Whole Plots” 相对于 “8 Whole Plots” 的效率
截距	0.778
X1	0.778
X2	1.004
X3 1	0.924
X3 2	1.154
X1*X2	0.982
X1*X3 1	0.924
X1*X3 2	1.154
X2*X3 1	1.107
X2*X3 2	1.449

估计的相对标准误差

项	4 Whole Plots	8 Whole Plots
截距	0.553	0.43
X1	0.553	0.43
X2	0.23	0.231
X3 1	0.25	0.231
X3 2	0.22	0.254
X1*X2	0.24	0.235
X1*X3 1	0.25	0.231
X1*X3 2	0.22	0.254
X2*X3 1	0.256	0.283
X2*X3 2	0.218	0.315

图 16.26 显示在第 435 页的“具有不同整区数的裂区设计”中比较的裂区设计的“相对估计效率”分级显示项。

相对估计效率

对于给定的某项，参考设计相对于比较设计的估计效率是指：该比较设计的该项的相对标准误差除以参考设计的该项的相对标准误差的结果。小于 1 的值指示该参考设计不像比较设计那样高效。大于 1 的值指示该参考设计更高效。

估计的相对标准误差

“估计的相对标准误差”报表提供参数估计值的标准差与误差标准差的比值。这些值指示模型参数估计值的标准误差相对于误差标准差有多大。对于第 i 个参数估计值，按以下方式定义“估计的相对标准误差”：

$$SE = \sqrt{(\mathbf{X}'\mathbf{X})^{-1}_{ii}}$$

其中：

$\mathbf{X}$ 是“技术细节”附录中第 735 页的“别名矩阵”中定义的模型矩阵，

$(\mathbf{X}'\mathbf{X})^{-1}_{ii}$ 是 $(\mathbf{X}'\mathbf{X})^{-1}$ 的第 i 个对角线元素。

别名矩阵汇总

“别名矩阵”解决不包括在模型中的项如何影响模型项的估计值的问题（若这些项确实是活跃的）。在“别名项”分级显示项中列出可能的活跃效应，它们不在您的假设模型中，但是可能使模型项的估计值偏倚。“别名矩阵”条目表示别名项效应给模型参数带来的偏倚程度。请参见第 441 页的“别名项”和第 449 页的“别名矩阵”。

“别名矩阵汇总”表列出假设模型中的项。这些是对应于“模型”分级显示项中所列效应的项。在给定设计的前提下，对于“项”列中的每个条目，将为对应于“别名项”分级显示项中的效应的项计算别名矩阵条目平方和的平方根。在给定设计的“均方根值”列中报告该值。相关示例，请参见第 450 页的“别名矩阵汇总值的计算示例”。

注意：仅当“别名项”列表中有效应时才会显示“别名矩阵汇总”报表。

图 16.27 两个设计的别名矩阵汇总

别名矩阵汇总

项	“Plackett-Burman” 均方根值	“确定性筛选设计” 均方根值
截距	0.0000	0.0000
X1	0.2722	0.0000
X2	0.2722	0.0000
X3	0.2722	0.0000
X4	0.2722	0.0000
X5	0.2722	0.0000
X6	0.2722	0.0000
合计	0.2520	0.0000

好 较差
0.10 0.30 0.50 0.70

图 16.27 显示在第 427 页的“试验规模相同的设计”中构造的 Plackett-Burman 和确定性筛选设计的“别名矩阵汇总”报表（“模型”分级显示项中仅有主效应）。所有双因子交互作用都位于“别名项”列表中。该表显示：对于确定性筛选设计，主效应与双因子交互作用不相关。

根据“别名矩阵汇总”表下方显示的颜色梯度对“均方根值”着色。您可以使用“别名矩阵汇总”红色小三角菜单中的选项控制颜色图例。请参见第 442 页的“颜色仪表板”。

别名矩阵

“别名矩阵”的行是对应于“模型”分级显示项中所列模型效应的项。列是对应于“别名项”分级显示项中所列效应的项。给定的行和列中的条目指示别名项在多大程度上影响对应于模型项的参数估计值。

在评估设计时，您理想上希望相对于“别名矩阵”中的任何条目，符合两种情形之一：一种情况是条目很小，另一种情况是条目不小，但别名项的效应很小，以至于偏倚很小。若您怀疑别名项的效应可能很大，则该项应包括在模型中或您应考虑别名最优设计。实际上，别名最优性取决于别名矩阵的平方值。

有关别名矩阵的其他背景信息，请参见“技术细节”附录中第 735 页的“别名矩阵”。另见 Lekivetz, R. (2014)。

别名矩阵汇总值的计算示例

本例演示如何计算 “别名矩阵汇总”分级显示项中显示的值。在本例中，您在假定只有主效应活跃的前提下比较两个设计。

1. 选择**帮助** > **样本数据**，点击**打开样本脚本目录**，然后选择 Compare Same Run Size.jsl。
2. 在脚本窗口中右击并选择**运行脚本**。

   随之构造两个 13 次试验设计表：

   – 确定性筛选设计
   – Plackett-Burman

   您只关注 Plackett-Burman 设计。这是活跃表。
3. 从 Plackett-Burman 表中，选择**实验设计** > **设计诊断** > **评估设计**。
4. 从 X1 一直选择到 X6，然后点击 **X，因子**。
5. 点击**确定**。
6. 打开 “别名项”分级显示项，确认所有双因子交互作用都位于 “别名项”列表中。
7. 打开 “别名矩阵”分级显示项。

   对于“效应”列中列出的每个模型项，给定列中该行的条目都指示别名项在多大程度上影响对应于模型项的参数估计值。

例如，要获取图 16.27 中对应于 X1 的“别名矩阵汇总”条目，将“别名矩阵”中的 X1 对应的行中的各项取平方，取其平均值，然后再取平方根。可以得到 0.2722。

绝对相关性

“绝对相关性”报表汇总关于模型项和别名项之间的相关性的信息。

图 16.28 三个设计的“绝对相关性”报表

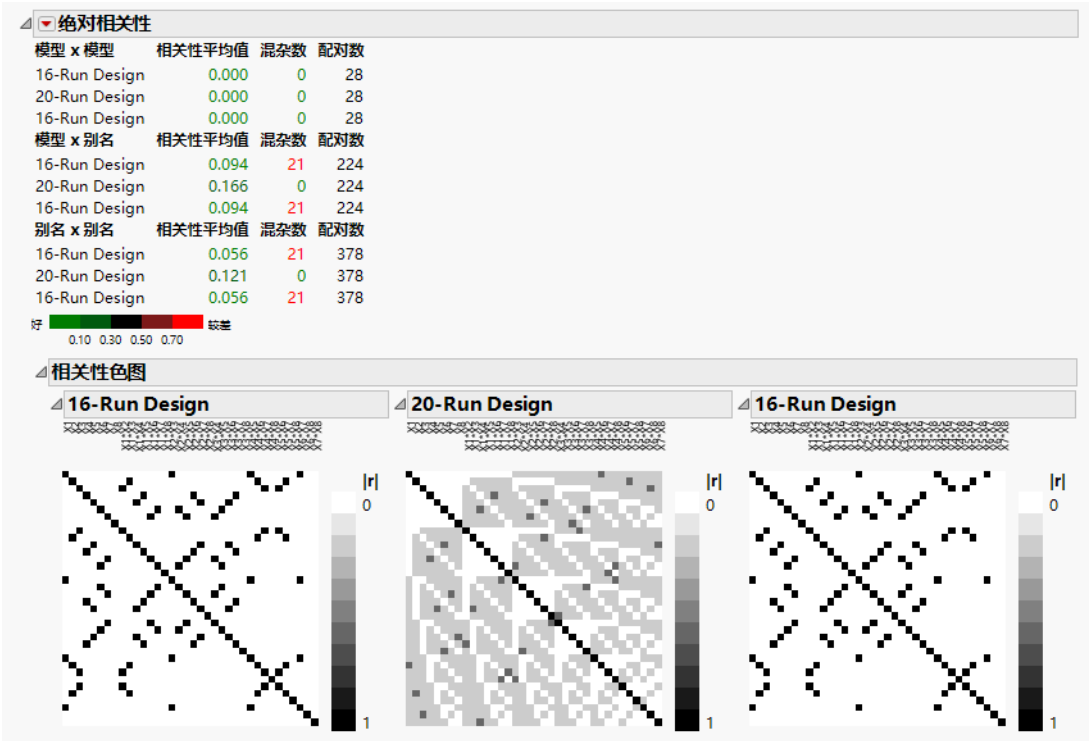


图 16.28 显示在第 432 页的“试验规模不同的设计”中构造的三个设计的“绝对相关性”报表（“模型”分级显示项中仅有主效应）。

绝对相关性表

“绝对相关性”报表中的表分为三部分：

- “模型 x 模型”考虑对应于“模型”列表中的效应的项之间的相关性。
- “模型 x 别名”考虑对应于“模型”列表中的效应的项之间的相关性以及对应于“别名”列表中的效应的项之间的相关性。
- “别名 x 别名”考虑对应于“别名”列表中的效应的项之间的相关性。

注意：若没有别名项，则仅显示“模型 x 模型”部分。

对于报表中的每个部分，都提供以下信息：

相关性平均值 报表这部分考虑的所有配对项的相关性的平均值。

混杂数 构成混杂项的配对项数目。

项数 报表这部分考虑的配对项总数。

根据该表下方显示的颜色梯度对“绝对相关性”表中的值着色。您可以使用“绝对相关性”红色小三角菜单中的选项控制颜色图例。请参见第 442 页的“颜色仪表板”。

相关性色图

“相关性色图”分级显示项为每个设计显示图。色图的单元格在图上方标识。对应于“模型”分级显示项或“别名项”分级显示项中显示的效应的所有项都有对应的单元格。根据两个项之间的相关性的绝对值对每个单元格着色。

默认情况下，相关性的绝对大小用白色-灰色-黑色强度颜色主题表示。通常来说，优秀设计的色图显示对角线外有很多白色，这表示不同项之间是正交的或相关性很小。效应之间的绝对相关性很大会导致估计值的标准误差膨胀。

要查看两个效应之间相关性的绝对值，请悬停在相应的单元格上方。要更改颜色尺度，请点击“相关性色图”红色小三角并选择**由蓝经灰到红**。对于定制颜色尺度，在图中右击并选择**颜色主题**。要保存相关性表，请在图例下方的图右侧右击，然后选择**相关性表**。

绝对相关性和相关性色图的示例

图 16.28 显示了在第 432 页的“试验规模不同的设计”中构造的 Plackett-Burman 和确定性筛选设计的“绝对相关性”报表。“模型”分级显示项仅包含主效应，所以“别名项”分级显示项包含所有双因子交互作用。所有主效应和双因子交互作用都显示在色图中。

在 16 次试验设计的相关性色图中，主对角线之外的黑色单元格指示相应项的相关性为 1，因此完全混杂。模型项（主效应）与别名项（双因子交互作用）共混杂九次，其中有六次是别名项相互之间混杂。上述情况显示在“配对混杂”下的报表中。

20 次试验设计和 24 次试验设计的色图没有落在对角线之外的黑色实心单元格。由此判定：这些设计表明所有成对的主效应或双因子交互作用之间都不混杂。不过，值得注意的是：与 16 次试验设计相比，20 次试验设计和 24 次试验设计的“模型 x 别名项”相关性平均值均较高。尽管 16 次试验设计显示出混杂，但其相关性平均值小于 20 次试验设计和 24 次试验设计。

设计诊断

“设计诊断”分级显示项显示相对于比较设计的参考设计的 *D*、*G*、*A* 和 *I* 效率。其中还显示“额外试验规模”。对于给定的两个设计，相对效率测度值更高的设计好一些。

图 16.29 三个设计的设计诊断

设计诊断

	“16-Run Design” 相对于 “20-Run Design” 的效率	“16-Run Design” 相对于 “24-Run Design” 的效率
D 效率	0.800	0.667
G 效率	0.800	0.667
A 效率	0.800	0.667
I 效率	0.800	0.667
额外试验规模	-4	-8

好 较差
1.50 1.25 0.80 0.67

图 16.29 显示在第 432 页的“试验规模不同的设计”中构造的三个设计的“设计诊断”报表（“模型”分级显示项中仅有主效应）。

根据该表下方显示的颜色梯度对“设计诊断”表中的值着色。您可以使用“设计诊断”红色小三角菜单中的选项控制颜色图例。请参见第 442 页的“颜色仪表板”。

效率和额外试验规模

D、*G*、*A* 和 *I* 效率中每个效率的相对效率都显示在“设计诊断”报表中。通过计算每个设计的效率值再取合适的比率可以得到这些值。相对效率测度的说明在第 453 页的“相对效率测度”中提供。

“额外试验规模”是参考设计中的试验次数减去比较设计中的试验次数的结果。若参考设计的试验数多于比较设计，则“额外试验规模”会告知您需要额外执行多少次试验才能实现参考设计的效率。

相对效率测度

符号

- $\mathbf{X}$ 是模型矩阵
- p 是模型中的项数（包括截距）
- $Var(\hat{y}|\underset{\sim}{\mathbf{x}})$ 是点 $\underset{\sim}{\mathbf{x}}$ 处的相对预测方差。请参见“技术细节”附录中第 738 页的“相对预测方差”。

相对效率

参考设计 (*Ref*) 与比较设计 (*Comp*) 的相对效率通过以下表达式得到：

D 效率 Eff_{Ref} / Eff_{Comp}，其中，每个设计的 *Eff* 计算如下：

$$Eff = |\mathbf{X}'\mathbf{X}|^{1/p}$$

G 效率 Eff_{Comp} / Eff_{Ref}，其中，每个设计的 *Eff* 计算如下：

$$Eff = Var(\hat{y}|\underset{\sim}{\mathbf{x}})_{max} = \underset{\underset{\sim}{\mathbf{x}} \text{ 在 D 中}}{maximum}[\underset{\sim}{\mathbf{x}}'(\mathbf{X}'\mathbf{X})^{-1}\underset{\sim}{\mathbf{x}}]$$

在此， D 表示设计区域。

注意： 使用设计空间的 Monte Carlo 抽样计算 G 效率。报告的值基于 $Var(\hat{y}|\underset{\sim}{\mathbf{x}})_{max}$ 或 Monte Carlo 抽样中的预测方差中的较大者。因此，同一设计的再次计算可能略有不同。

A 效率 Eff_{Comp} / Eff_{Ref}，其中，每个设计的 *Eff* 计算如下：

$$Eff = \text{Trace}[(\mathbf{X}'\mathbf{X})^{-1}]$$

I 效率 Eff_{Comp} / Eff_{Ref}，其中，每个设计的 *Eff* 计算如下：

$$Eff = \frac{\int \tilde{\mathbf{x}}'(\mathbf{X}'\mathbf{X})^{-1}\tilde{\mathbf{x}}d\mathbf{x}}{\int d\mathbf{x}}$$

有关计算的详细信息，请参见 Goos and Jones (2011) 中的第 4.3.5 节。

比较设计选项

高级选项 > 裂区方差比 指定随机整区和子区 （若存在）方差与误差方差之间的比值。设置该值之前，您必须为裂区设计定义难以更改的因子，或是为裂 - 裂区设计定义难以更改和极难更改的因子。然后，您可以为方差比输入一个或两个正数，具体取决于您指定的是裂区设计还是裂 - 裂区设计。

第 17 章

前瞻样本大小与功效

在规划实验或研究时，使用“样本大小与功效”平台调查样本大小对您回答特定问题的能力的影响。

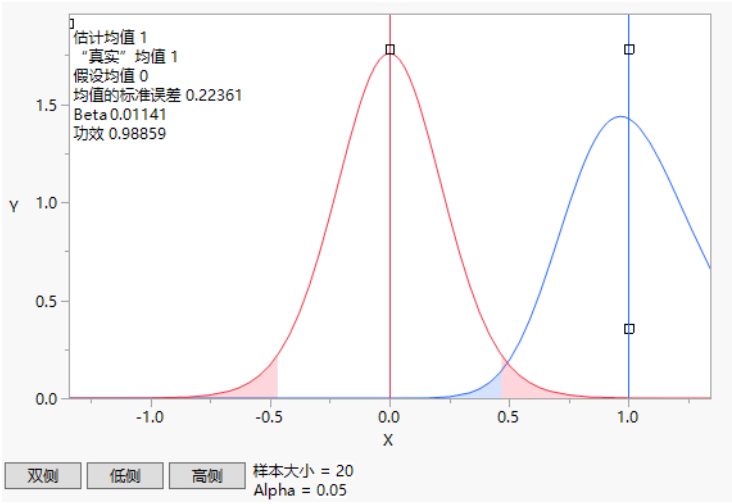
- 我应该检验多少个单元？
- 我是否能在我的处理方式中检测到差值？
- 我必须检验多少个单元来估计失效时间？

“样本大小与功效”平台是一个计算器的集合。这些计算器具有用于数据输入的字段和计算出的值。您可以选择要指定的值和要计算的值。例如，您可能指定功效和要检测的效应大小，然后计算所需的样本大小。或者，您可能指定样本大小和功效以计算可以检测的效应大小。

假设检验的样本大小计算基于 I 类和 II 类误差的权衡。I 类误差（或 α）是当原假设为真时拒绝它的概率。II 类误差（或 β）是当原假设为假时不拒绝它的概率。通常，您希望 α 和 β 都很小。功效（或 1 - β）是当原假设为假时拒绝它的概率。

有关 JMP 中的功效和样本大小的详细信息，请参见 Barker (2011)。

图 17.1 单样本均值的功效动画



目录

启动“样本大小与功效”平台 458
单样本均值计算器 459
“单样本均值”计算器的示例 459
“单样本均值”计算器字段 463
“单样本均值”计算器的统计详细信息 464
“双样本均值”计算器 465
“双样本均值”计算器的示例 465
“双样本均值”计算器字段 466
“双样本均值”计算器的统计详细信息 467
“k 个样本均值”计算器 467
“k 个样本均值”计算器的示例 468
“k 个样本均值”计算器字段 469
“k 个样本均值”计算器的统计详细信息 470
“单样本标准差”计算器 471
“单样本标准差”计算器的示例 471
“单样本标准差”计算器字段 472
“单样本标准差”计算器的统计详细信息 473
“单样本比例”计算器 474
“单样本比例”计算器的示例 474
“单样本比例”计算器字段 475
单比例计算器的统计详细信息 476
“双样本比例”计算器 477
“双样本比例”计算器的示例 477
“双样本比例”计算器字段 478
双比例计算器的统计详细信息 479
“每单位计数”计算器 479
“每单位计数”计算器的示例 480
“每单位计数”计算器字段 480
“每单位计数”计算器的统计详细信息 481
“Sigma 质量水平”计算器 482
“Sigma 质量水平”计算器的示例 482
计算缺陷数的示例 482
“Sigma 质量水平”计算器字段 483
Sigma 质量计算器的统计详细信息 483
“可靠性检验计划”计算器 484
“可靠性检验计划”计算器的示例 484
“可靠性检验计划”计算器字段 485

“可靠性检验计划”计算器的统计详细信息 487
“可靠性验证”计算器 487
“可靠性验证”计算器的示例 487
“可靠性验证”计算器字段 489
“可靠性验证”计算器的统计详细信息 489

启动“样本大小与功效”平台

“样本大小与功效”平台是用于规划实验或研究的计算器的集合。可通过选择**实验设计 > 设计诊断 > 样本大小与功效**来启动 “样本大小与功效”平台。

图 17.2 “样本大小与功效”启动窗口

从以下计算器中选择：

- 第 459 页的“单样本均值计算器”
- 第 465 页的“‘双样本均值’计算器”
- 第 467 页的“‘k 个样本均值’计算器”
- 第 471 页的“‘单样本标准差’计算器”
- 第 474 页的“‘单样本比例’计算器”
- 第 477 页的“‘双样本比例’计算器”
- 第 479 页的“‘每单位计数’计算器”
- 第 482 页的“‘Sigma 质量水平’计算器”
- 第 484 页的“‘可靠性检验计划’计算器”
- 第 487 页的“‘可靠性验证’计算器”

单样本均值计算器

使用“单样本均值”计算器计算有关一个均值的假设检验的样本大小。研究如何权衡样本大小、功效、显著性和假设的待检差值。样本大小与功效与以下假设检验相关联：

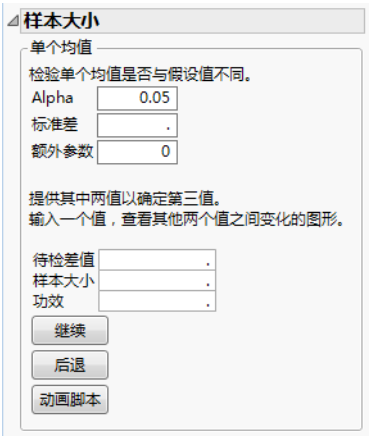
$$H_0: \mu = \mu_0$$

与双侧备择假设：

$$H_a: \mu \neq \mu_0$$

其中 μ 是真均值，μ_0 是原假设均值或参考值。均值偏离 μ_0 的差值 δ，是您认为重要，需要检测的量（基于一组样本检测）。对于相同的显著性水平和功效，检测小差值与检测大差值相比，前者需要更大的样本大小。假定关注的总体服从均值为 μ、标准差为 σ 的正态分布。

图 17.3 单均值的初始样本大小与功效计算器



“单样本均值”计算器的示例

在以下示例中，假定您想验证您公司研发的新织物与当前材料相比可燃性有改进。以前的检验表明该织物的点燃时间的标准差为 2 秒。

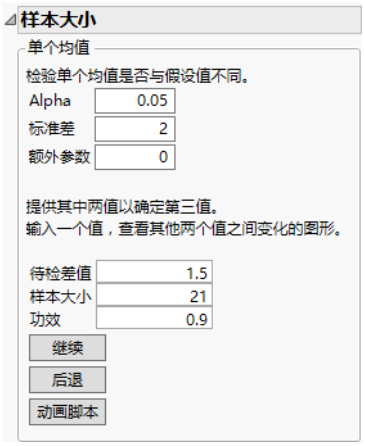
样本大小计算的示例

在该初始示例中，您想设计一个功效为 90% 的实验在显著性水平 $\alpha = 0.05$ 下检测差值 （1.5 秒）。使用 “单样本均值”计算器计算您需要检验的织物样本数。

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**单样本均值**按钮。
3. 保持 Alpha 设置为 0.05。

4. 为**标准差**输入 2。
5. 保持**额外参数**设置为 0。
6. 为**待检差值**输入 1.5。
7. 将**样本大小**留空。
8. 为**功效**输入 0.9。
9. 点击**继续**。

图 17.4 “单样本均值”计算器



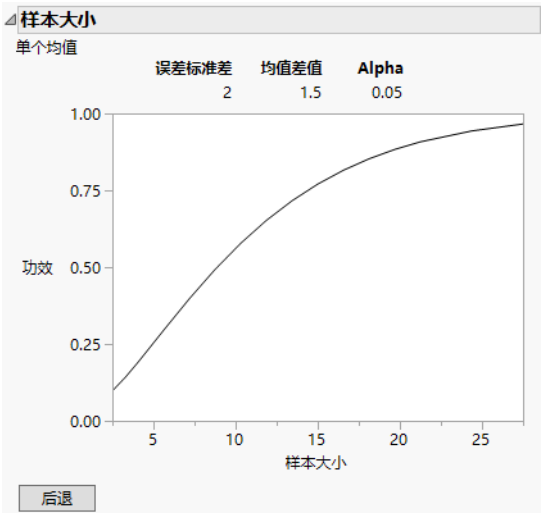
显著性水平为 0.05 时，要想有 90% 机会检测到显著的点燃时间差值 1.5 秒，需要 21 个织物样本。

“样本大小 - 功效”图的示例

要探索如何在您的织物实验中权衡样本大小和功效，请使用 “样本大小 - 功效”图。

1. 选择**实验设计** > **设计诊断** > **样本大小与功效**。
2. 点击**单样本均值**。
3. 保持 Alpha 设置为 0.05。
4. 为**标准差**输入 2。
5. 保持**额外参数**设置为 0。
6. 为**待检差值**输入 1.5。
7. 将**样本大小**留空。
8. 保留**功效**为空。
9. 点击**继续**以启动 “样本大小 - 功效”图。

图 17.5 样本大小 - 功效



该图显示功效从大约 0.1 变到大约 0.95 时样本大小的范围。您可以在实验中将检验减少到大约 15 次而维持功效在 75% 以上。但是，若您只进行 10 次检验，检测到显著差值 1.5 的功效将下降到大约 50%。

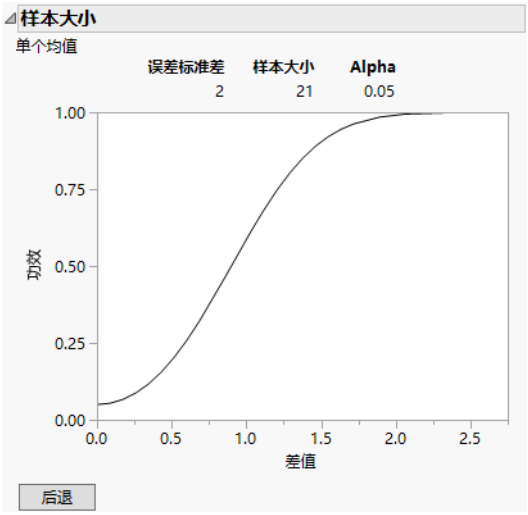
提示：使用十字准线工具从图中获取样本大小和功效的组合值。

“差值 - 功效”图的示例

要了解在织物实验中如何权衡功效和差值量值 （可以使用 21 次观测来检测），请使用 “差值 - 功效”图。

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**单样本均值**。
3. 保持 Alpha 设置为 0.05。
4. 为**标准差**输入 2。
5. 保持**额外参数**设置为 0。
6. 将**待检差值**留空。
7. 为**样本大小**输入 21。
8. 保留**功效**为空。
9. 点击**继续**。

图 17.6 样本大小为 21 时的 “待检差值 - 功效”图



在显著性水平为 0.05、观测数为 21 时，您可以检测 1.5 秒的差值（功效为 90%）。若该差值只有 1 秒那么小，在织物样本数为 21 时，您检测到该差值的功效大约为 50%。

动画脚本的示例

使用一个动画脚本来探索样本大小如何影响功效。

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**单样本均值**。
3. 保持 Alpha 设置为 0.05。

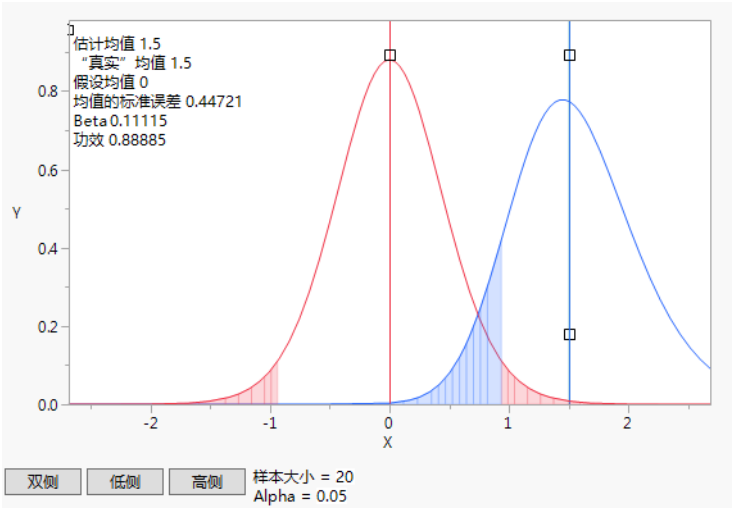
   提示：启动动画后，通过点击动画窗口中的 “Alpha”值，您可以更改 Alpha 水平。

4. 为**标准差**输入 2。
5. 保持**额外参数**设置为 0。
6. 输入 1.5 作为**待检差值**。
7. 将**样本大小**留空。

   提示：将样本大小留空时，默认样本大小设置为 20。在启动动画脚本后，您将看到默认值 20。您可以在启动动画之前或之后更改样本大小。要在启动动画后更改样本大小，请在动画窗口中点击 “样本大小”值。

8. 保留**功效**为空。
9. 点击**动画脚本**。

图 17.7 演示功效的初始动画脚本



初始动画图显示两个 *t* 密度曲线：

- 红色曲线显示真实均值为零时的 *t* 分布。
- 蓝色曲线显示真实均值为 1.5（此为待检差值）时的 *t* 分布。
- 蓝色着色指示发生 II 类误差的概率。II 类误差为当有差异时未检测到差异的概率。II 类误差的概率通常用 β 表示。
- 红色着色指示发生 I 类误差的概率。I 类误差是当没有差异时得出均值有显著差异结论的概率。I 类误差通常用 α 表示。

选择并拖动方形控点来查看统计量随曲线位置的变化。要更改样本大小和 Alpha 的值，请点击图下对应的值。

默认情况下，动画显示双侧检验，使用**双侧**、**低侧**和**高侧**按钮在不等于、小于或大于备择假设之间切换。

“单样本均值”计算器字段

指定以下数量：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

> **提示：**对于单侧检验，使用 α = α *2。对于 α = 0.05 的单侧检验，将双侧计算器与 α = 0.10 配合使用。所得值是 α = 0.05 的单侧检验所需的那些值。

标准差 假定的标准差。误差标准差的估计值可以是先前的模型拟合的均方根误差 (RMSE)。

提示：使用标准差 1 估计检测差值 （用标准差单位数表示）所需的样本大小。

额外参数 假设检验中除 μ 之外的参数数目。该选项可用于多因子设计。在简单情形中，保持该字段中的默认值为零。

在效应正交的多因子设计中，您可以在此处指定额外模型参数的数目。例如，在三因子、两水平的设计（具有三个双因子交互作用）中，额外参数数目为 5：2 个参数用于其他主效应，3 个参数用于交互作用。

指定以下值中的两个来计算第三个值，或指定一个值来获取其他两个参数之间的关系图：

待检差值 真均值和假设均值 （或参考均值）之间的最小差值，您希望能称为在统计上显著。

样本大小 您实验中的总观测数 （试验次数、实验单元数或样本数）。

功效 当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

计算器按钮

继续 在指定两个参数时计算缺失值，在仅指定一个参数时启动用于比较两个缺失参数的图。

后退 返回之前的 “样本大小与功效”启动窗口。

动画脚本 启动一个交互图来说明和探索功率和待检差值之间的关系。请参见第 462 页的 “动画脚本的示例”。

“单样本均值”计算器的统计详细信息

单样本均值计算基于以下假设检验的传统完全与简化的 *F* 检验：

$H_0: \mu = \mu_0$ vs $H_a: \mu \neq \mu_0$

JMP 按如下方式计算功效：

$$\Pr(\text{拒绝 } H_0 \mid \mu = \mu_0 + \delta) = 1 - F\left(f_{1-\alpha}, 1, n-p-1, \frac{n\delta^2}{\sigma^2}\right)$$

其中：

α 是显著性水平。

n 是样本大小。

p 额外参数的数目。

δ 是待检差值。

$f_{1-\alpha}$ 是 $F(1, n-p-1)$ 分布的第 $(1-\alpha)$ 分位数。

$F(x, df_1, df_2, nc)$ 是非中心 F 分布的累积分布函数，该函数的自由度为 df_1 和 df_2，非中心参数 nc 在 x 处求值。

由于 δ 和 n 的分析解不存在，则使用数值解对其求解。

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 2.1 节）。

“双样本均值”计算器

使用 “双样本均值”计算器计算有关两个均值的假设检验的样本大小。探索诸如以下的问题：对于给定的样本大小、显著性水平、所需的功效和假定的数据中的变异，处理效应需要多大才能检测出来？样本大小与功效与以下假设检验相关联：

$$H_0: \mu_1 - \mu_2 = 0$$

与双侧备择假设：

$$H_a: \mu_1 - \mu_2 \neq 0$$

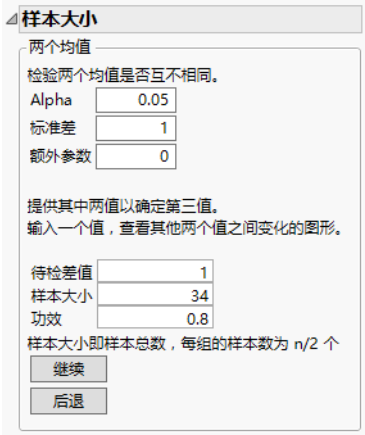
其中 μ_1 和 μ_2 是两个总体的真实均值。假定相关总体服从正态分布，且您要检测两个均值之间的差值 δ。

“双样本均值”计算器的示例

假定您在比较两个组，想在显著性水平为 0.05、功效为 80% 时检测 1 个标准差的组均值差值。

1. 选择**实验设计** > **设计诊断** > **样本大小与功效**。
2. 点击**双样本均值**。
3. 保留 **Alpha** 为 0.05。
4. 为**标准差**输入 1。
5. 保留**额外参数**为 0。
6. 输入 1 作为**待检差值**。
7. 将**样本大小**留空。
8. 为**功效**输入 0.8。
9. 点击**继续**。

图 17.8 “双样本均值”计算器结果



计算的样本大小为 34。每个组有 17 个样本时，在 $\alpha = 0.05$ 的显著性水平下，您有 80% 的机会检测出大小为 1 个标准差的双样本均值差值。

“双样本均值”计算器字段

指定以下数量：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

标准差 假定的标准差（假设两个组的标准差相等）。

额外参数 假设检验中除 μ_1 和 μ_2 之外的参数数目。该选项可用于多因子设计。在简单情形中，保持该字段中的默认值为零。

在多因子平衡设计中，可以在此处指定额外模型参数的数目。例如，在三因子、两水平的设计（具有三个双因子交互作用）中，额外参数数目为 4：1 个参数用于其他主效应，3 个参数用于交互作用。

指定以下参数中的两个来计算第三个值，或指定一个值来获取其他两个参数之间的关系图：

待检差值 两个均值之间的最小差值，您希望能称为在统计上显著。

样本大小 您实验中的总观测数（试验次数、实验单元数或样本数）。在每个组中使用此样本大小的一半。

功效 当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

计算器按钮

继续 在指定两个参数时计算缺失值，在仅指定一个参数时启动用于比较两个缺失参数的图。

后退 返回之前的 “样本大小与功效”启动窗口。

“双样本均值”计算器的统计详细信息

双样本均值计算基于以下假设检验的传统完全与简化的 *F* 检验：

H_0: $\mu_1 - \mu_2 = 0$ vs H_a: $\mu_1 - \mu_2 \neq 0$

JMP 按如下方式计算功效：

$$\text{功效} = \Pr(\text{拒绝 } H_0 \mid (\mu_1 - \mu_2 = \delta)) = 1 - F\left(f_{1-\alpha}, 1, 2n - p - 2, \frac{n\delta^2}{2\sigma^2}\right)$$

其中：

α 是显著性水平。

n 是每组样本大小。

p 额外参数的数目。

δ 是待检差值。

$f_{1-\alpha}$ 是 $F(1, 2n - p - 2)$ 分布的第 $(1 - \alpha)$ 分位数。

$F(x, df_1, df_2, nc)$ 是非中心 *F* 分布的累积分布函数，该函数的自由度为 df_1 和 df_2，非中心参数 nc 在 x 处求值。

由于 δ 和 n 的分析解不存在，则使用数值解对其求解。

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 2.2 节）。

“k 个样本均值”计算器

使用 “k 个样本均值”计算器确定有关 2 到 10 个均值的假设检验的适当样本大小。样本大小与功效与以下假设检验相关联：

H_0: $\mu_1 = \mu_2 = \ldots = \mu_k$

与双侧备择假设：

H_a：并非所有均值都相等

其中：

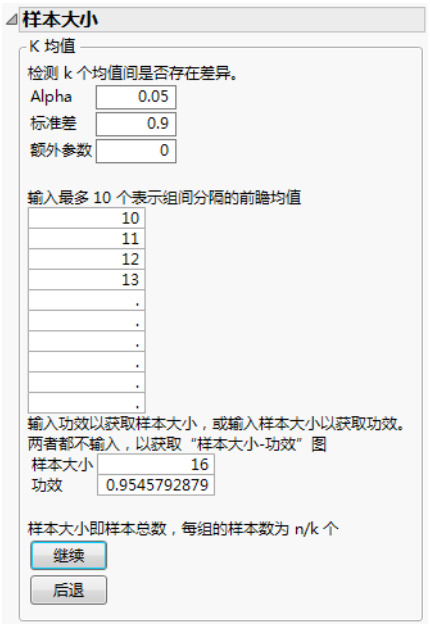
$X_{ij} \sim N(\mu_j, \sigma^2)$ for $i = 1, \ldots, n,\ j = 1, \ldots, k$

“k 个样本均值”计算器的示例

假定您的 4 个组每个组有 4 个观测值，想检测所有组均值是否相等。您期望组均值为 10、 11、 12 和 13，标准差为 0.9。计算在 0.05 显著性水平下检测一个或多个均值的差值的功效。

1. 选择**实验设计** > **设计诊断** > **样本大小与功效**。
2. 点击 **k 个样本均值**。
3. 保留 Alpha 为 0.05。
4. 为**标准差**输入 0.9。
5. 保留**额外参数**为 0。
6. 输入 10、 11、 12 和 13 作为前瞻均值的四个水平。
7. 为**样本大小**输入 16。
8. 保留**功效**为空。
9. 点击**继续**。

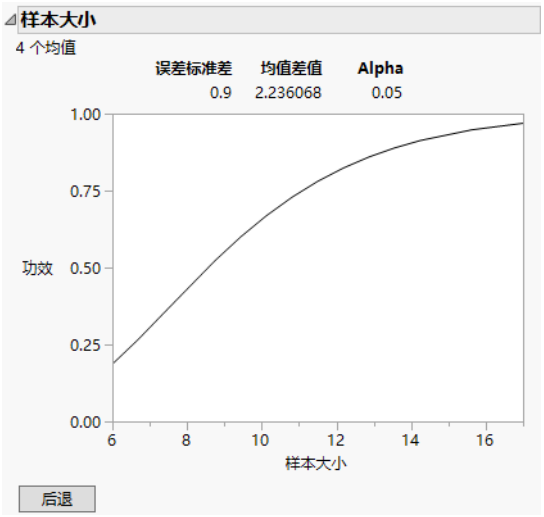
图 17.9 4 个均值的前瞻功效



计算出的功效大约为 0.95。这意味着：假定总体均值为 10、11、12 和 13，标准差为 0.9，总样本大小为 16 时，在 0.05 显著性水平下有 95% 的机会检测出至少有一个均值不同。

10. 为“样本大小”点击 16，然后删除它。
11. 为“功效”点击 0.95，然后删除它。
12. 点击**继续**以启动“样本大小 - 功效”图。

图 17.10 k 个均值的前瞻功效和“样本大小 - 功效”图



从“样本大小 - 功效”图，您可以确认样本大小为 16 是合适的。悬停在“样本大小”轴上方，并拖动以将轴延伸到 16 之外。您可以看到样本大小超过 16 时，功效的增加量很小。

该图还报告均值差值，它由与总均值的偏差平方和的平方根计算得出。在本例中，它是 $(-1.5)^2 + (-0.5)^2 + (0.5)^2 + (1.5)^2$ 的平方根，即 0.5 的平方根。因此，本例中的均值差值约为 2.236。

“k 个样本均值”计算器字段

指定以下数量：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

标准差 假定的标准差 （假设所有组的标准差相等）。

额外参数 假设检验中除 μ 之外的参数数目。该选项可用于多因子设计。在简单情形中，保持该字段中的默认值为零。

前瞻均值 均值的假定值。至少必须指定两个均值。

指定以下值中的一个来计算第二个值，或将两个值均留空以获取这两者之间的关系图：

样本大小 您实验中的总观测数 （试验次数、实验单元数或样本数）。在每个组中使用样本大小 n/k，其中 n 是总观测数。

功效 当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

计算器按钮

继续 在指定一个参数时计算缺失值，或在未指定任何参数时启动用于比较两个参数的图。

后退 返回之前的 “样本大小与功效”启动窗口。

“k 个样本均值”计算器的统计详细信息

k 个样本均值计算基于以下公式：

$$\text{功效} = 1 - F\left(f_{1-\alpha}, 1k-1, 2kn-p-2k, \frac{n\sum_{j=1}^{k}(\mu_j-\bar{\mu})^2}{\sigma^2}\right)$$

其中：

α 是显著性水平。

σ 是 k 个组的共同标准差。

n 是每组样本大小。

p 额外参数的数目。

$\bar{\mu}$ 是 *k* 个组前瞻均值的均值。

$f_{1-\alpha}$ 是 $F(k - 1, kn - p - k)$ 分布的第 $(1 - \alpha)$ 分位数。

$F(x, df_1, df_2, nc)$ 是非中心 *F* 分布的累积分布函数，该函数的自由度为 df_1 和 df_2，非中心参数 *nc* 在 *x* 处求值。

使用数值方法对 *n* 求解。

均值差值

按以下方式计算均值差值：

$$\sqrt{\sum_{j=1}^{k}(\mu_j-\bar{\mu})^2}$$

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 2.3 节）。

“单样本标准差”计算器

使用 “单样本标准差”计算器计算有关一个标准差的假设检验的样本大小。样本大小与功效与以下假设检验相关联：

H_0: $\sigma = \sigma_0$

与单侧备择假设：

H_a: $\sigma > \sigma_0$ 或 H_a: $\sigma < \sigma_0$

其中 σ 是总体标准差，σ_0 是假设值。标准差偏离 σ_0 的差值 δ，是您认为重要，需要检测的量（基于一组样本检测）。

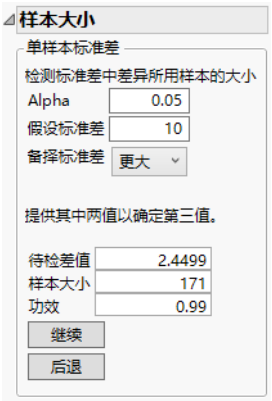
“单样本标准差”计算器的示例

注意：该示例来自美国国家标准和技术研究所 (NIST) 的在线手册。您可以在 https://www.itl.nist.gov/div898/handbook/prc/section2/prc232.htm 上查看 NIST 手册示例。

对许多硅晶片的电阻率进行测量所得的方差要求为 100 欧姆平方厘米。若某一批次的方差大于 155 欧姆平方厘米，买家将不愿接受。您必须检验多少晶片来估计批次方差（要求针对目标 100 欧姆平方厘米，可以检测 55 欧姆平方厘米的增量）？从标准差角度，假设的标准差 σ_0 为 10（100 的平方根）且 σ 为 12.4499（100 + 55 = 155 的标准差）。待检差值为 12.4499 – 10 = 2.4499。所需的功效为 0.99，显著性水平为 0.05。

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**单样本标准差**。
3. 保留 **Alpha** 为 0.05。
4. 为**假设标准差**输入 10。
5. 为**备择标准差**选择 “更大”。
6. 为**待检差值**输入 2.4499。
7. 将**样本大小**留空。
8. 为**功效**输入 0.99。
9. 点击**继续**。

图 17.11 “单样本标准差”计算器



您必须检验 171 个晶片，才能在 alpha 为 0.05 的情况下有 99% 的机会检测出标准差增加了 2.4499（针对标准差 10）。

“单样本标准差”计算器字段

指定以下信息：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

假设标准差 样本标准差要与之比较的假设标准差或基准标准差。

备择标准差 要检测的变动的方向 （“更大”或 “更小”）。

> **注意：**若从 “备择标准差”菜单选择 “更小”，请输入负的 “待检差值”。

指定以下值中的两个以计算第三个值：

待检差值 最小可检测差值 （多小的差值您希望能称为在统计上显著）。它是假设值与真实值的差值。

样本大小 您实验中的总观测数 （试验次数、实验单元数或样本数）。

功效 当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

计算器按钮

继续 计算缺失值。

后退 返回之前的 “样本大小与功效”启动窗口。

“单样本标准差”计算器的统计详细信息

JMP 用于计算功效的公式取决于“备择标准差”选项的设置。

对于备择标准差更大的情形， JMP 按以下方式计算功效：

$$\text{功效} = 1 - F\left(\frac{\sigma_0^2 \cdot \chi_{1-\alpha}}{(\sigma_0 + \delta)^2}, n-1\right)$$

$$\delta = \sigma_0 \sqrt{\frac{\chi_{1-\alpha}}{\chi_\beta}} - \sigma_0$$

其中：

σ_0 是假设标准差。

σ 是真正的标准差。

X_p 是 X_{n-1}^2 分布的第 $(1 - p)$ 分位数。

$F(x, n\text{-}1)$ 是在 x 处求值的 X_{n-1}^2 分布的累积分布函数。

对于备择标准差更小的情形， JMP 按以下方式计算功效：

$$\text{功效} = 1 - F\left(\frac{\sigma_0^2 \cdot \chi_{\alpha}}{(\sigma_0 + \delta)^2}, n-1\right)$$

$$\delta = \sigma_0 \sqrt{\frac{\chi_{\alpha}}{\chi_{1-\beta}}} - \sigma_0$$

其中：

σ_0 是假设标准差。

σ 是真正的标准差。

X_p 是 X_{n-1}^2 分布的第 $(1 - p)$ 分位数。

$F(x, n\text{-}1)$ 是在 x 处求值的 X_{n-1}^2 分布的累积分布函数。

使用数值方法对 n 求解。

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 2.4 节）。

“单样本比例”计算器

使用 “单样本比例”计算器计算有关一个比例的假设检验的样本大小。样本大小与功效与以下假设检验相关联：

$$H_0: p = p_0$$

与双侧备择假设：

$$H_a: p \neq p_0$$

或与单侧备择假设：

$$H_a: p < p_0 \text{ 或 } H_a: p > p_0$$

其中 p 是总体比例，p_0 是原假设比例。

单比例计算器还提供实际检验大小。这是指定假设下的实际 I 类误差率。因为二项分布是离散的，对于接近 0 或 1 的小样本或比例，实际检验大小可能与指定的 Alpha 水平有很大不同。为了确保 alpha 水平等于或大于指定的水平，请使用 Clopper-Pearson 精确方法。

“单样本比例”计算器的示例

假定一条装配线过去的缺陷比例等于 0.1。在样本大小为 100 且 alpha 水平为 0.05 时，您想要计算功效以检测与历史比率有 0.1 或更大差异的缺陷率。

1. 选择**实验设计** > **设计诊断** > **样本大小与功效**。
2. 点击**单样本比例**。
3. 保留 **Alpha** 为 0.05。
4. 将 0.1 作为**比例**的值。
5. 将**方法**保留为 “Agresti-Coull 精确区间估计方法”。
6. 将检验类型保留为**双侧**。
7. 将 0.2 作为**原假设比例**的值。
8. 输入 100 作为**样本大小**。
9. 点击**继续**。

图 17.12 “单样本比例”计算器

样本大小为 100 时，该功效约为 70%。“实际检验大小”大约为 0.0467，略小于所需的 0.05。样本大小为 100 时，若缺陷率为 0.2，I 类误差（拒绝原假设）的概率为 5%。或者，II 类误差（不拒绝原假设）的概率为 30%（1 - 功效）。

“单样本比例”计算器字段

指定以下数量和检验设置：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

比例 真实比例，它可以是已知或假设的。默认值为 0.1。

提示：若所有其他参数保持不变，当该比例为 0.5 时，样本大小最大。

方法 选择一种方法。选项为“Agresti-Coull 精确区间估计方法”或“Clopper-Pearson 精确区间估计方法”。Clopper-Pearson 方法与 Agresti-Coull 方法相比更为保守（使用更大的样本大小）。

单侧或双侧 指定单侧或双侧检验。

指定以下参数中的两个来计算第三个值，或指定一个值来获取其他两个参数之间的关系图。

注意：该图使用正态近似法而非精确方法。

原假设比例 要对照进行检验的比例 (p_0)。默认值为 0.2。

样本大小 实验中的总观测数 （试验次数、实验单元数或样本数）。

功效 当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

计算器按钮

继续 在指定两个参数时计算缺失值，在仅指定一个参数时启动用于比较两个缺失参数的图。

后退 返回之前的 “样本大小与功效”启动窗口。

单比例计算器的统计详细信息

单比例样本大小计算使用基于二项分布的精确方法。精确计算确保达到指定的功效水平。

Agresti-Coull 方法

Agresti-Coull 精确区间估计方法使用调整的基于 Wald 的检验统计量。JMP 在双侧原假设下按以下方式计算功效：

$$\text{功效} = \sum_{y=0}^{n} \Pr\{Y = y | Y \sim \text{二项}(n, p)\} I\{T(y) \geq \chi_{1-\alpha}\}$$

其中：

$$T(y) = \frac{(\hat{p} - p_0)^2}{\frac{\hat{p}(1-\hat{p})}{n+4}}$$

$$\hat{p} = \frac{y+2}{n+4}$$

$I\{T(y_1,y_2) \geq \chi_{1-\alpha}\} = 1$ 若 $(T(y_1,y_2) \geq \chi_{1-\alpha})$ 其他情况则为 0，并且

$\chi_{1-\alpha}$ 是 χ_1^2 分布的第 $(1 - \alpha)$ 分位数。

因为对于 n 或 p_0 没有近似的表达式，使用数值方法求解 n 或 p_0 的值。

有关调整的 Wald 检验统计量的详细信息，请参见 Agresti and Coull (1998)。有关 JMP 中计算的详细信息，请参见 Barker（2011，第 3.3 节）。

Clopper-Pearson 方法

Clopper-Pearson 精确区间估计方法基于二项分布。该方法导致 alpha 水平等于或大于指定的水平。Clopper-Pearson 方法比 Agresti-Coull 方法更保守（使用更大的样本大小）。

Clopper-Pearson 精确区间估计方法直接使用二项分布。使用数值方法求解未知参数的值。

有关 Clopper-Pearson 精确区间估计方法的详细信息，请参见 Clopper and Pearson (1934) 或 Agresti and Coull（1998，第 1 节）。

“双样本比例”计算器

使用 “双样本比例”计算器计算有关两个比例的假设检验的样本大小。样本大小与功效与以下假设检验相关联：

$$H_0: p_1 - p_2 = D_0$$

与双侧备择假设：

$$H_a: p_1 - p_2 \neq D_0$$

或与以下某个单侧备择假设：

$$H_a: (p_1 - p_2) < D_0 \text{ 或 } H_a: (p_1 - p_2) > D_0$$

其中 p_1 和 p_2 是来自两个总体的总体比例，D_0 是假设的比例差值。

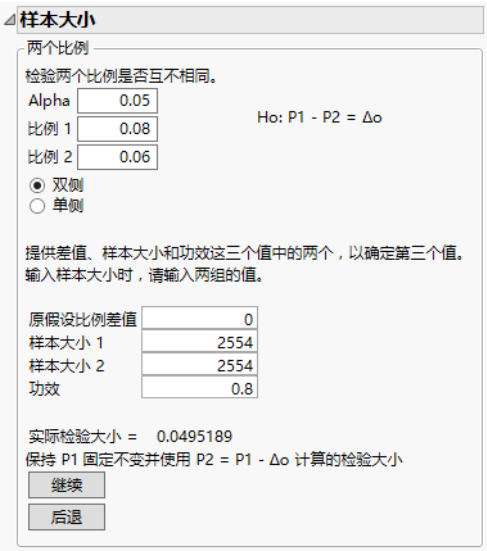
双比例计算器还提供实际检验大小。这是指定假设下的实际 I 类误差率。因为二项分布是离散的，对于小样本或接近 0 或 1 的比例，实际检验大小可能与指定的 Alpha 水平有很大不同。

“双样本比例”计算器的示例

假定您负责两条硅晶片装配线。根据很多次试验的了解，其中一条装配线的缺陷率为 8%，另一条装配线的缺陷率为 6%。您想知道在功效为 80% 时得出缺陷率有差异结论所需的样本大小。

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**双样本比例**。
3. 将 Alpha 设置为 0.05。
4. 为**比例 1** 输入 0.08。
5. 为**比例 2** 输入 0.06。
6. 选择**双侧**检验。
7. 为**原假设比例差值**输入 0。
8. 将**样本大小 1** 和**样本大小 2** 留空。
9. 为**功效**输入 0.8。
10. 点击**继续**。

图 17.13 针对双侧检验的双比例差值



计算器显示您应对每条生产线检验 2,554 个晶片的样本大小，才能检测缺陷率差异。

“双样本比例”计算器字段

指定以下数量和检验设置：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

比例 1 总体 1 的比例，它可以是已知或假设的。默认值为 0.5。

比例 2 总体 2 的比例，它可以是已知或假设的。默认值为 0.1。

单侧或双侧 指定单侧或双侧检验。

指定以下参数中的两个以计算第三个参数 （其中样本大小被视为一个参数） 。必须同时输入两个样本大小，或者将它们留空以进行计算。

原假设比例差值 原假设下的比例差异。默认值为 0.2。

样本大小 1 实验中的总观测数 （试验次数、实验单元数或样本数），用于估计 “比例 1”。

样本大小 2 实验中的总观测数 （试验次数、实验单元数或样本数），用于估计 “比例 2”。

注意：计算样本大小时，样本大小 1 = 样本大小 2。若您输入样本大小，则它们不必相等。

功效 当原假设为假时拒绝原假设的概率。希望高功效。所有其他参数保持不变时，功效随样本大小增大而增大。

计算器按钮

继续 计算缺失值。

后退 返回之前的 “样本大小与功效”启动窗口。

双比例计算器的统计详细信息

双比例计算所基于的精确方法与用于单比例计算器的精确方法类似。对于双侧假设

$$H_0: p_1 - p_2 = D_0 \text{ vs } H_a: p_1 - p_2 \neq D_0$$

JMP 在原假设下按以下方式计算功效：

$$\text{功效} = \sum_{y_1=0}^{n_1} \sum_{y_2=0}^{n_2} \Pr(Y_1 = y_1)\Pr(Y_2 = y_2) I\{T(y_1,y_2) \geq \chi_{1-\alpha}\}$$

其中，按以下方式定义调整的 Wald 统计量：

$$T(y_1,y_2) = \frac{(\hat{p}_1 - \hat{p}_2 - \delta_0)^2}{\frac{\hat{p}_1(1-\hat{p}_1)}{n_1+2} + \frac{\hat{p}_2(1-\hat{p}_2)}{n_2+2}} \text{ 其中 } \hat{p}_j = \frac{y_j+1}{n_j+2} \text{ 对于 } j = 1, 2$$

并且：

$$I\{T(y_1,y_2) \geq \chi_{1-\alpha}\} = 1 \text{ 若 } (T(y_1,y_2) \geq \chi_{1-\alpha}) \text{ 其他情况则为 } 0$$

$\chi_{1-\alpha}$ 是 χ^2 分布的第 (1 - α) 分位数。

因为没有 n_i 或 δ_0 的近似表达式，使用数值方法求解 n_i 和 δ_0 的值。

有关调整的 Wald 检验统计量的详细信息，请参见 Agresti and Coull (1998)。有关 JMP 中计算的详细信息，请参见 Barker（2011，第 3.3 节）。

“每单位计数”计算器

使用 “每单位计数”计算器来计算有关每个单位的缺陷 （计数）的单侧假设检验的样本大小。样本大小与功效与以下假设检验相关联：

$$H_0: \lambda \leq \lambda_0$$

与单侧备择假设：

$$H_a: \lambda > \lambda_0$$

其中 λ 是 Poisson 分布的比率。

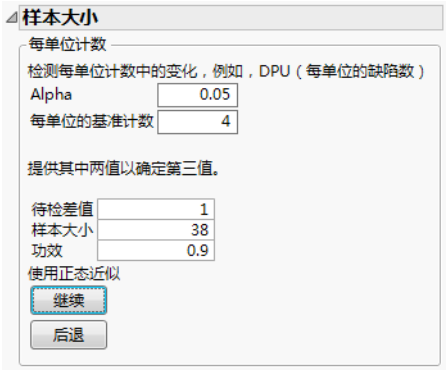
“每单位计数”计算器的示例

考虑晶片制造过程，目标值为每个晶片 4 个或更少的缺陷。您想验证新过程是否满足目标要求。在 0.05 显著性水平下，要使功率为 90%，您应检验多少晶片来检测差值 1（与每个晶片的目标缺陷计数相比）？

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**每单位计数**。
1. 保留 Alpha 为 0.05（若新过程和目标一样好但检验失败的机会）。
2. 输入 4 作为**每单位的基准计数**，指示目标是每个晶片有 4 个缺陷。
3. 输入 1 作为**待检差值**。
4. 输入功率值 0.9。

   这是检测大于 1 的变动的机会，它等价于每个晶片有 5 个或更多缺陷。在这种情形下，显著性水平 (alpha) 有时称为**生产者的风险**，功效 (beta) 称为**消费者的风险**。
5. 点击**继续**。

图 17.14 “每单位计数”计算器



您必须检验 38 个晶片。若缺陷数少于 173，则该过程满足目标。

注意：173 个缺陷是在拒绝原假设的情况下 38 个晶片中允许的最大缺陷数。

“每单位计数”计算器字段

指定以下数量：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

每单位的基准计数 定为目标的每单位缺陷数。默认值为 0.1。

在以下数量中指定两个以计算第三个数量：

待检差值 针对指定的每单位缺陷数要检验的最小可检测差值。

样本大小 单位数。

功效 当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

计算器按钮

继续 计算缺失值。

后退 返回之前的“样本大小与功效”启动窗口。

“每单位计数”计算器的统计详细信息

“每单位计数”计算器的计算基于将正态近似应用到 Poisson 分布。该检验使用以下假设：

$$H_0: \lambda \le \lambda_0 \text{ vs } H_a: (\lambda > \lambda_o)$$

上述假设使用以下检验统计量：

$$T = \frac{\bar{d} - \lambda_0}{\sqrt{\lambda_0 / n}}$$

其中 n 是样本大小，$\bar{d}$ 是样本中每单位缺陷数的均值。我们假定 T 近似于正态分布。功效计算基于原假设和备择假设下的 T 分布。

$$\text{功效} = \Pr(\text{拒绝} H_0 \mid \lambda = \lambda_0 + \delta)$$

$$= \Pr[T > Z_{1-\alpha} \mid (\lambda = \lambda_0 + \delta)] = 1 - \Phi\left(\frac{Z_{1-\alpha} - \delta\sqrt{n/\lambda_0}}{\sqrt{(\lambda_0 - \delta)/\lambda_0}}\right)$$

其中 Φ() 是标准正态累积分布函数，Z_{1-p} 是标准正态分布的第 (1 - p) 分位数。

使用 1 - β 表示拒绝原假设的所需功效，样本大小按以下方式计算：

$$n = \frac{\lambda_0}{\delta^2}\left(Z_{1-\alpha} - Z_{\beta}\sqrt{\frac{\lambda_0 + \delta}{\lambda_0}}\right)^2$$

因为 δ 的分析解不存在，使用数值方法在给定功效 δ 和 n 的条件下求解。

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 2.5 节）。

“Sigma 质量水平”计算器

使用“Sigma 质量水平”计算器探索缺陷数、缺陷的机会数和 Sigma 质量水平之间的关系。Sigma 质量水平是一个统计量，它将给定的缺陷率与 6 sigma 尺度关联。例如，每一百万个机会中，3.4 个缺陷的结果是 6 sigma 过程。

“Sigma 质量水平”计算器的示例

假定您要估计工厂 1,000,000 个机会中有 50 个缺陷的 Sigma 质量水平。一个机会是缺陷计数的单位。它可以是单个物品，单个物品可能有多个缺陷 （其中每个潜在缺陷是一个机会）。

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击 **Sigma 质量水平**。
3. 为**缺陷数**输入 50。
4. 输入 1000000 作为**机会数**。
5. 点击**继续**。

图 17.15 “Sigma 质量水平”计算器

Sigma 质量水平约为 5.4。

计算缺陷数的示例

要计算在 6 sigma 质量水平过程中一百万次机会中您可以具有的最大缺陷数，请执行以下步骤:

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击 **Sigma 质量水平**按钮。
3. 将**缺陷数**留空。
4. 输入 1000000 作为**机会数**。
5. 输入 6 作为 **Sigma 质量水平**。
6. 点击**继续**。

图 17.16 1,000,000 次机会的 6 Sigma 质量水平

该计算显示在 1,000,000 次机会中**缺陷数**不能超过 3 才能实现 6 sigma 质量水平过程。

“Sigma 质量水平”计算器字段

在以下数量中指定两个以计算第三个数量：

缺陷数 缺陷数。

机会数 缺陷的机会数。

Sigma 质量水平 用标准差单位表示的缺陷率。

计算器按钮

继续 计算缺失值。

后退 返回之前的“样本大小与功效”启动窗口。

Sigma 质量计算器的统计详细信息

对于 n 次机会中的 d 个缺陷， sigma 质量计算器的计算基于 sigma 质量水平 σ_q 的定义：

$$\sigma_q = Z_{1-\frac{d}{n}} + 1.5$$

从该定义中，可以按以下方式计算 d 和 n：

$$d = n[1 - \Phi(\sigma_q - 1.5)]$$

$$n = d[1 - \Phi(\sigma_q - 1.5)]^{-1}$$

其中 $\Phi()$ 是标准正态累积分布函数，$Z_{1-d/n}$ 是标准正态分布的第 $(1 - d/n)$ 分位数

“可靠性检验计划”计算器

使用“可靠性检验计划”计算器确定可靠性研究所需的样本大小、研究长度或精度。可靠性研究用于估计产品的失效时间和失效概率。该检验计划计算器使用期望的置信区间来定义估计值的精度。这些估计值基于拟合的失效分布。要拟合失效分布，必须观测最小失效次数。

“可靠性检验计划”计算器的示例

某公司开发了一个新产品，想计算要检验多少个单元来估计 20% 的单元失效所需的时间（置信区间为 95%，双侧绝对精度为 200 小时）。换句话说，当为估计的时间创建置信区间时，上限与下限之间的差值大约应为 200 小时。该公司可执行 2,500 小时的检验。此外，根据类似产品的研究，他们认定近似失效分布为位置参数 $\alpha = 2000$、尺度参数 $\beta = 3$ 的 Weibull 分布。

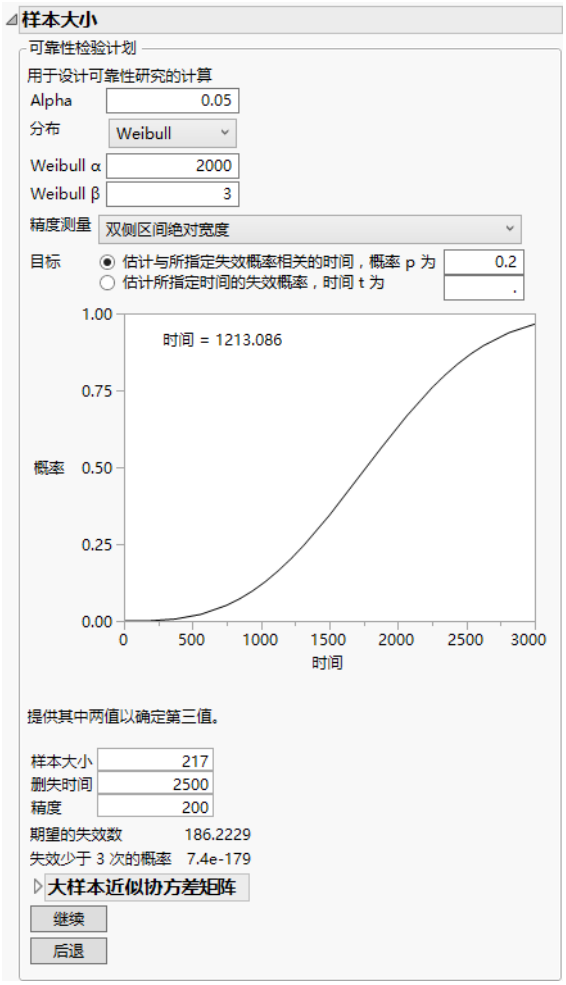
要计算所需的样本大小，请执行以下步骤：

1. 选择**实验设计 > 设计诊断 > 样本大小与功效**。
2. 点击**可靠性检验计划**。
3. 将 Alpha 设置为 0.05。
4. 从分布列表中选择 Weibull。
5. 为 Weibull α 参数输入 2000。
6. 为 Weibull β 参数输入 3。
7. 从 “精度测量”列表中选择**双侧区间绝对宽度**。
8. 选择**估计与所指定失效概率相关的时间**，并为 p 输入 0.2。

   注意：根据研究目标标记累积分布函数图。在此处，时间 = 1213 是 20% 的单元失效的时间估计值。

9. 将**样本大小**留空。
10. 为**删失时间**输入 2500。
11. 为**精度**输入 200。
12. 点击**继续**。

图 17.17 “可靠性检验计划”计算器



对于 2,500 小时的检验，要估计 20% 的单位失效的时间（精度为 200 小时）需要 217 个单元。本检验中期望的失效次数为 186，远大于估计失效分布所需的最小值 3。

“可靠性检验计划”计算器字段

指定以下数量和检验设置：

Alpha 用于定义精度测量的置信区间的显著性水平。

分布 假定的失效分布。可用分布包括：Weibull、对数正态、 Frechet、对数 Logistic、最小极值、正态、最大极值和 Logistic。有关这些分布的详细信息，请参见《可靠性和生存方法》。

位置 失效分布的位置参数。

注意：将 “分布”设置为 “Weibull”时，“位置”字段用 Weibull α 表示。

尺度 失效分布的尺度参数。

注意：将 “分布”设置为 “Weibull”时，“尺度”字段用 Weibull β 表示。

精度测量 精度测量的定义。这些定义基于要估计的数量（失效时间或概率）的期望置信区间。精度测量的选项如下所示：

区间比 将精度定义为上限与下限之比的平方根。该比值始终大于 1，因为上限大于或等于下限。随着估计精度的增大，区间比将下降。

双侧区间绝对宽度 将精度定义为置信区间宽度或上限与下限的差值。

下单侧区间绝对宽度 将精度定义为区间下侧宽度或估计值与估计值置信区间下限的差值。

双侧区间相对宽度 将精度定义为置信区间相对于估计值的宽度。它用上限与下限的差值除以估计值计算而来。

下单侧区间相对宽度 将精度定义为区间下侧相对于估计值的宽度。它用估计值与下限的差值除以估计值计算而来。

目标 研究的目标。选择以下目标之一并输入相应的值：

- 估计与指定失效概率 *p* 相关的时间。
- 估计时间 *t* 时的失效概率。

注意：该图是失效分布的累积分布函数。该图用基于研究目标的时间或概率估计值标记。

在以下数量中指定两个以计算第三个数量：

样本大小 要包括在可靠性检验中的单元数。

删失时间 运行可靠性检验的可用时间量。

精度 精度水平。该值的单位定义与所选 “精度测量”对应。

计算器按钮

继续 计算缺失值。

后退 返回之前的 “样本大小与功效”启动窗口。

其他可靠性检验计划计算

除了计算样本大小、删失时间或精度外，还计算以下数量：

期望的失效数 指定的可靠性检验的期望失效数。

失效少于 3 次的概率 指定的可靠性检验得到小于三次失效结果的概率。这很重要，因为至少需要三次失效来获取失效分布的位置和尺度参数的稳定估计值。当只有一次或两次失效时，

估计是不稳定的。若该概率很大，您即使没有观测足够的失效次数也可能可靠估计分布参数。增大样本大小或删失时间都可降低小于三次失效的概率。

大样本近似协方差矩阵 提供失效分布的位置和尺度参数的近似方差和协方差。

“可靠性检验计划”计算器的统计详细信息

可靠性检验计划设计用于估计某个分位数

$$q(p) = \exp[\sigma\Phi^{-1}(p) + \mu]$$

或失效概率：

$$p(t) = \Phi\left[\frac{\log(t) - \mu}{\sigma}\right]$$

（在指定的显著性水平下，具有假定的失效时间分布和精度测量）。Φ() 是位置参数为 μ、尺度参数为 σ 的假定失效时间分布的标准累积分布函数。Wald 置信区间和精度测量将精度定义为 t（时间）和 n（样本大小）的函数。使用数值方法来求解所需的数量。

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 6 节）。

“可靠性验证”计算器

使用 “可靠性验证”计算器在指定的时间段检验指定的单元数。若失效的单元少于 k 个，则通过验证，您可以认为产品可靠性满足或超过可靠性标准。我们将可靠性验证视为假设检验：

$$H_o: p < p^* \text{ vs } H_a: p \geq p^*$$

其中 p 是新产品在 t^* 时的生存概率，p^* 是在时间 t^* 时生存的标准概率。

“可靠性验证”计算器的示例

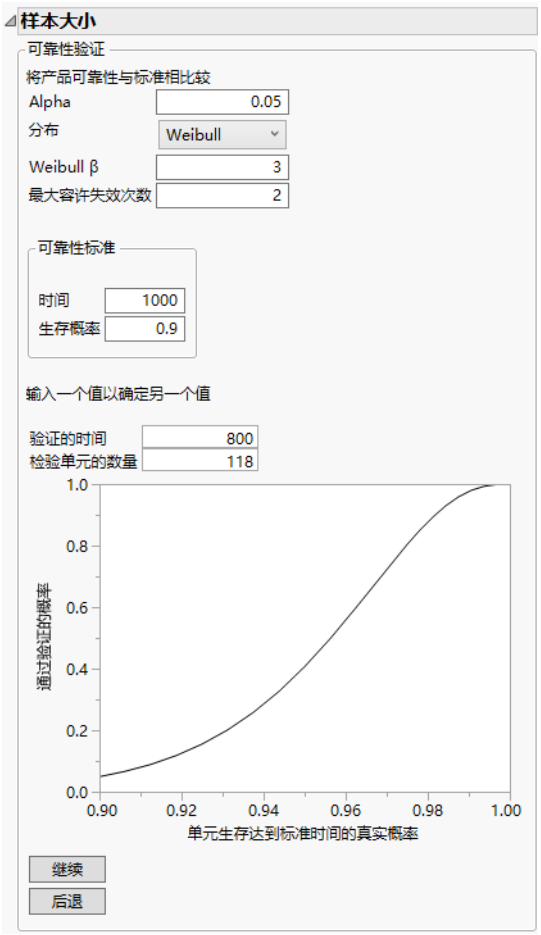
一家公司想估计用来评估新产品的可靠性所需的样本大小，该评估是针对以往可靠性标准（在 1,000 小时后有 90% 的概率生存）而言。根据以前对类似产品的研究结果，认定失效时间分布为 Weibull 分布（β 参数为 3）。公司可承受执行 800 小时的验证，希望实验中发生的失效不多于 2 次。

要计算所需的样本大小，请执行以下步骤：

1. 选择**实验设计** > **设计诊断** > **样本大小与功效**。
2. 点击**可靠性验证**。
3. 将 Alpha 设置为 0.05。
4. 从分布列表中选择 Weibull。

5. 为 Weibull β 输入 3。
6. 为**最大容许失效次数**输入 2。
7. 为**时间**输入 1000。
8. 为**生存概率**输入 0.9。
9. 为**验证的时间**输入 800。
10. 将**检验单元的数量**留空。
11. 点击**继续**。

图 17.18 “可靠性验证”计算器



公司需要在验证中试验 118 个单元。此外，若公司在 800 小时内观测到 2 次或更少的失效，可以认定新产品的可靠性至少与标准一样可靠。

该图显示随着某单元生存到可靠性标准中的时间的真实概率增大，通过验证的概率将增大。在该示例中，若一个单元只有 92% 的机会生存到标准时间，则新产品通过验证的机会只有大约 12.5%。

“可靠性验证”计算器字段

指定以下数量和检验设置：

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

分布 假定的失效时间分布。可用分布包括：Weibull、对数正态、 Frechet、对数 Logistic、最小极值、正态、最大极值和 Logistic。有关这些分布的详细信息，请参见《可靠性和生存方法》。

尺度 失效时间分布的尺度参数。

注意：指定的 “分布”为 “Weibull”时，“尺度”字段用 Weibull β 表示。

最大容许失效次数 成功的检验验证允许的最大失效次数。

可靠性标准 您正在检验的标准的定义。该标准基于一个时间长度和某个物品生存到该时间的概率。

时间 某个物品在检验下应生存的最短时间长度。

生存概率 该物品在检验下生存到可靠性标准定义的 “时间”的概率。

输入以下两个值中的一个来计算另一个值：

验证的时间 验证的时间长度。

检验单元的数量 验证所需的单元数。

计算器按钮

继续 计算缺失值。

后退 返回之前的 “样本大小与功效”启动窗口。

“可靠性验证”计算器的统计详细信息

可靠性验证取决于尺度参数为 σ 的假定失效时间分布。可靠性标准 （或时间 *t* 时的生存概率）和位置 μ 指定如下：

$$p = 1 - \Phi\left[\frac{\log t - \mu}{\sigma}\right]$$

其中， μ 使用以下等式求解：

$$\mu = \log(t) - \sigma\Phi^{-1}(1-p)。$$

为了计算样本大小和检验大小，将在时间 *t* 时生存的概率视为假设检验：

$$H_o: p < p^* \text{ vs } H_a: p \geq p^*$$

其中 p^* 是在时间 t^* 时生存的标准概率。

我们想按以下方式检验该假设 （α 水平下）：

$\alpha = \Pr$ （*k* 或更少失效 | H_0 为 true）。

因为该检验涉及 *n* 个独立单元，失效数具有二项 (*n*, *p*) 分布，其中 *p* 是某单元在时间 *t* 前失效的概率。因此，我们可以将 α 表示为 *t* 和 *n* 的函数：

$$\alpha = \Pr\left(x \leq k | x \sim \text{Bin}\left\{n, \Phi\left[\frac{\log t - \mu^*}{\sigma^*}\right]\right\}\right)$$

其中 μ* 和 σ* 来自假定的可靠性标准。

二项和 beta 分布的属性使我们可以使用以下公式求解 *t* 值：

$$t = \frac{t^* \exp\{\sigma^* \Phi^{-1}[1 - \text{B}^{-1}(\alpha; n-k, k+1)]\}}{\exp[\sigma^* \Phi^{-1}(1-p^*)]}$$

对于 *n*， Brent 方法用于查找以下等式的根：

$$\text{B}^{-1}(\alpha; n-k, k+1) - 1 + \Phi\left[\frac{\log t - (\log t^*) + \sigma^* \Phi^{-1}(1-p^*)}{\sigma^*}\right] = 0$$

其中：

$\text{B}^{-1}(\alpha; n-k, k+1)$ 是 $\text{Beta}(n-k; k+1)$ 分布的 α 分位数

Φ() 是假定的失效时间分布的累积分布函数。

有关 JMP 中计算的详细信息，请参见 Barker（2011，第 5 节）。

第 18 章

选择设计

创建用于选择偏好产品特征的设计

选择（或**离散选择**）实验提供用于对离散偏好建模的数据。研究参与者可从具有不同特性的多组可能产品（或产品特征）中进行选择。参与者从每组特征中都选择一个偏好特征。例如，在设计高端笔记本电脑时，电脑公司可能关注关键功能的相对重要性，如：处理器速度、硬盘大小、屏幕大小、电池寿命和价格。选择实验可将这些功能的相对价值传达给客户，并指出各产品功能在权衡利弊后的最佳取舍。

使用关联分析方法分析选择实验的结果。请参见《消费者研究》。

图 18.1 八个选择集的调查

	选择集	选择 ID	硬盘大小	速度	电池寿命	价格
1	1	1	40 GB	2.0 GHz	6 Hrs	$1000
2	1	2	80 GB	1.5 GHz	4 Hrs	$1500
3	2	1	80 GB	1.5 GHz	4 Hrs	$1200
4	2	2	40 GB	1.5 GHz	6 Hrs	$1500
5	3	1	40 GB	1.5 GHz	4 Hrs	$1200
6	3	2	80 GB	2.0 GHz	6 Hrs	$1500
7	4	1	40 GB	2.0 GHz	4 Hrs	$1000
8	4	2	80 GB	1.5 GHz	6 Hrs	$1500
9	5	1	80 GB	2.0 GHz	6 Hrs	$1200
10	5	2	40 GB	1.5 GHz	6 Hrs	$1000
11	6	1	40 GB	1.5 GHz	4 Hrs	$1500
12	6	2	80 GB	1.5 GHz	6 Hrs	$1000
13	7	1	80 GB	1.5 GHz	4 Hrs	$1000
14	7	2	40 GB	2.0 GHz	6 Hrs	$1200
15	8	1	80 GB	1.5 GHz	6 Hrs	$1200
16	8	2	80 GB	2.0 GHz	4 Hrs	$1500

目录

选择设计概述 493
选择设计术语 493
Bayesian D 最优性 493
选择设计示例 494
带有分析的选择设计示例 496
创建用于试行研究的选择设计 496
分析试行研究数据 498
使用先验信息设计最终选择实验 500
运行设计和分析结果 501
“选择设计”窗口 503
特性 504
模型 505
生成设计 506
设计 507
制表 508
选择设计选项 508
技术细节 509
Bayes D 最优性和设计构造 509
效用平均和局部 D 最优设计 510

选择设计概述

离散选择实验支持设计产品的流程。选择实验可帮助公司针对公司面向的市场确定产品功能的主次，从而设计出满足人们购买需求的产品。“选择设计”平台使用作为产品特性的因子创建实验。选择要研究的特性及特性值至关重要。您必须包括可能影响消费者的产品购买决策的所有特性。有关用于设计选择实验的详细信息和指导原则，请参见 Sall (2008)。

选择设计术语

以下列出了与选择设计相关的术语：

- **特性**即产品功能。
- **特征**指的是一组产品特性。
- **选择集**就是特征集合。
- **调查**就是选择集的集合。
- **部分特征**是指选择设计中的一个特征，在该设计中每个选择集中只有指定数量的特性变动，其余特性不变。

在离散选择实验中，向响应者提供包含若干选择集的调查。选择集通常只包含少量的特征来促进决策过程。在每个选择集内，每个响应者都需指定他/她偏好的特征。例如，笔记本电脑实验的特性可能包括速度、存储、屏幕大小、电池寿命和价格。这些特性的不同组合构成了产品特征。一个选择集可能包含两个特征。从每个选择集中，响应者选择他/她偏好的特征。

对于涉及许多特性的情况，您可以构造多个调查，让每个选择集包含**部分特征**。在包含部分特征的选择集内，只有指定数量的特性变动，其余特性不变。这会降低选择任务的复杂性。

Bayesian D 最优性

由于离散选择模型的参数是非线性的，选择设计的效率取决于未知参数。“选择设计”平台使用 Bayes 方法，该方法基于您指定的可能参数值的先验分布优化设计。Bayes D 最优性准则是信息矩阵对先验分布取得的行列式的对数期望值。“选择设计”平台最大化相对于先验概率分布的这一期望值。请参见第 509 页的“Bayes D 最优性和设计构造”和 Kessels et al.(2011)。

您还可以生成以下类型的设计：

- 效用平均设计 - 在效用平均设计中，选择集中的所有选择的可能性都是同等的。先验均值设置为 0。
- 局部 D 最优设计 - 局部 D 最优设计考虑先验均值，但不包括来自先验协方差矩阵的任何信息。

有关效用平均和局部 D 最优设计的详细信息，请参见第 510 页的“效用平均和局部 D 最优设计”。

选择设计示例

关于实验

在本例中，一家咖啡馆关注于如何冲泡理想的咖啡以满足多数客户的需求。经理要求您确定哪些因子影响客户偏好。具体而言，您需要确定以下因子（特性）的哪些设置会冲泡出理想的咖啡：

- 研磨颗粒大小（中等或粗糙）
- 温度（195°、200°、205°）
- 冲泡时间（3 分钟、3.5 分钟或 4 分钟）
- 咖啡豆的量（1.6 克/盎司、2 克/盎司或 2.4 克/盎司）

因子水平的每一组合都是一个**特征**。尝试通过获取每个响应者样本的每个可能的特征来获取关于偏好的信息是不可行的。不过，您可以让响应者从包含少量特征的**选择集**中选择偏好的特征。

在本例中，您需要设计一个实验，其中每个响应者都表明其在若干选择集中的偏好。您的设计将具有以下结构：

- 10 个响应者
- 12 个选择集
- 每个选择集两个特征
- 每个响应者一个调查，调查包含全部 12 个选择集

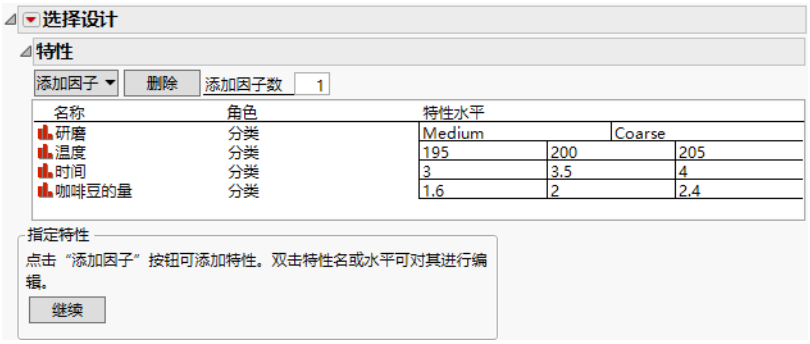
该实验中的每个响应者都有 12 个响应。借助针对这些偏好的分析，可以得出关于如何冲泡咖啡可取悦多数客户的结论。

创建设计

您可以手动输入因子，或使用包含因子和设置的预存表自动输入因子。在本例中，为方便起见，您使用一个预存表。但若设计的是一个新实验，您必须首先手动输入因子。有关手动输入因子的详细信息，请参见第 504 页的 “特性”。

1. 选择**实验设计 > 消费者研究 > 选择设计**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Coffee Choice Factors.jmp。
3. 点击 “选择设计”红色小三角并选择**加载因子**。

图 18.2 定义了特性的 “选择设计” 窗口



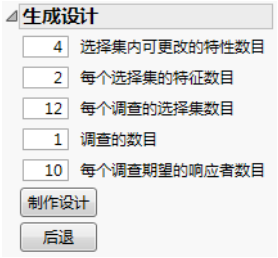
4. 点击**继续**。
5. （可选）打开 “实验设计模型控制” 分级显示项。

   请注意只有主效应。在本例中，您仅关注包含您的四个因子的主效应的模型。不过，若想要设计能够估计其他效应，您可以在该分级显示项中添加它们。
6. 在 “生成设计” 面板中：
   - 保持**选择集内可更改的特性数目**设置为 4。
   - 保持**每个选择集的特征数目**设置为 2。
   - 为**每个调查的选择集数目**键入 12。

     在本例中，响应者评估 12 个选择集。
   - 保持**调查的数目**设置为 1。
   - 为**每个调查期望的响应者数目**键入 10。

     在本例中，共有 10 个响应者。

图 18.3 完成的 “生成设计” 面板



注意：在第 7 步中设置随机种子可确切重现本例所示的结果。自行构造设计时，该步骤不是必需的。

7. （可选）点击 “选择设计” 红色小三角并选择**设置随机种子**。键入 12345 并点击**确定**。

8. 点击**制作设计**。

   共有 12 个选择集，每个选择集包含两个咖啡特征。

9. 选择**为特征和响应输出单独的表**。

   这会在名为“选择特征”的数据表中放入选择集的说明。还会构造第二个数据表（称为“选择试验”）以便于输入响应信息。

10. 点击**制表**。

    “选择特征”表显示 12 个选择集，每个选择集都包含两个特征。“选择试验”表支持您使用**特征 ID** 将首选的特征记录在**响应**列中。若 “选择 1”是首选特征，则输入 1 ；若选择 2 是首选特征，则输入 2。或者，若响应者无首选项，则将响应保持为缺失。

    “选择特征”表中的**选择**脚本有助于分析实验结果。该脚本可打开已完成的“选择模型”启动窗口。有关 “选择模型”的信息，请参见 《消费者研究》。

    使用 “选择特征”表中的**实验设计对话框**脚本可重新启动设计对话框。

带有分析的选择设计示例

在本例中，某一计算机厂商关注如何制造新型笔记本电脑，并想要在开始成本高昂的开发过程之前掌握关于客户偏好的信息。该厂商决定构造一个包含两组特征并要提供给 10 个响应者的设计。该选择设计的目标在于：了解潜在笔记本电脑购买者如何看待一组特性（共四个）的优势：

- 硬盘驱动器大小（40 GB 或 80 GB）
- 处理器速度（1.5 GHz 或 2.0 GHz）
- 电池寿命（4 小时或 6 小时）
- 电脑成本（$1000、$1200 或 $1500）

要为 10 个响应者构造该设计，首先应只使用一个响应者开展小型试行研究。然后分析结果，并在为 10 个响应者设计最终研究时使用该参数估计值作为先验信息。

创建用于试行研究的选择设计

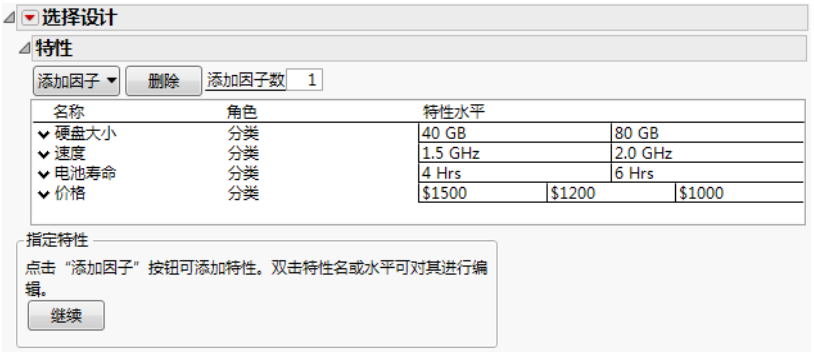
在本节中，您需要为单个响应者研究构造一个选择设计。

定义因子和水平

在本例中，您从现有表加载因子。自行设计新实验时，手动输入因子。

1. 选择**实验设计 > 消费者研究 > 选择设计**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Laptop Factors.jmp。
3. 点击 “选择设计”红色小三角并选择**加载因子**。

图 18.4 定义了特性的 “选择设计” 窗口



创建设计

1. 点击**继续**。
2. 本试行调查将发送给单个响应者。“实验设计模型控制”、“先验规格” 和 “生成设计” 面板中的默认值原样提供即合适。
3. （可选）点击 “选择设计” 红色小三角并选择**设置随机种子**。键入 12345 并点击**确定**。
4. 点击**制作设计**。

注意：在第 3 步中设置随机种子可确切重现本例所示的结果。自行构造设计时，该步骤不是必需的。

图 18.5 试行设计

这一调查包含八个选择集，每个选择集都包含两个笔记本电脑特征。

5. 确认已选定**将特征和响应合并在一个表中**选项。

   这会将选择集和调查结果放入同一个表。

6. 点击**制表**。

设计该调查时假定无先验信息。出于此原因，某些选择集可能不会提供有用信息。按照计划，需要从这一个响应者获取调查结果，分析结果，然后将试行调查中的结果用作设计最终调查时的先验信息。

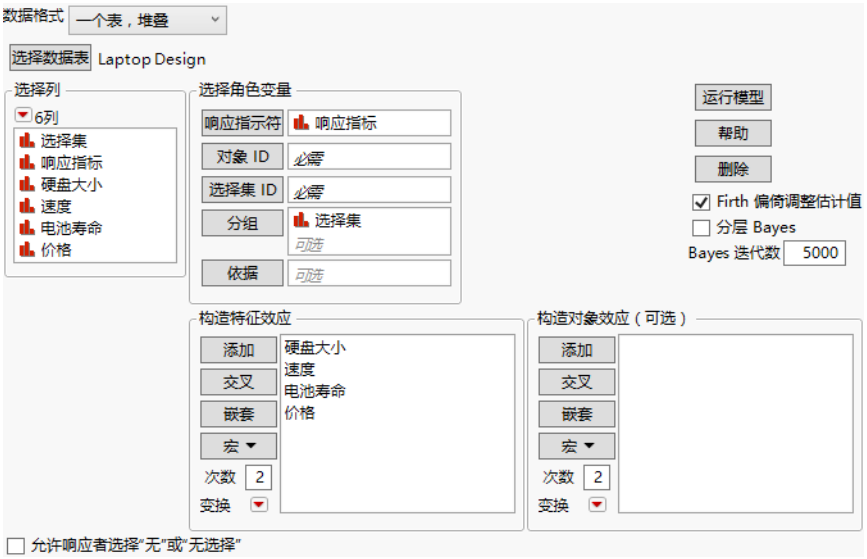
分析试行研究数据

现在试行调查设计已完成，接下来需要提供给这一个响应者。该响应者从每个集中选择一个特征，为选定的特征输入 1，为拒绝的特征输入 0。您需要使用“选择”平台分析结果。

注意：有关“选择”平台的详细信息，请参见《消费者研究》。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Laptop Design.jmp。
2. 运行**选择**脚本。

图 18.6 “选择模型”启动窗口



唯一的分组变量为**选择集**，因为只有一个调查和一个响应者。

3. 点击**运行模型**。

图 18.7 试行调查的参数估计值

参数估计值

项	估计值	标准误差
硬盘大小[40 GB]	-0.007125177	0.4991763970
速度[1.5 GHz]	0.214880936	0.5265848733
电池寿命[4 Hrs]	0.052925888	0.4895109018
价格[$1500]	-0.978517706	0.7062009731
价格[$1200]	0.076206148	0.7442825255

AICc	43.487115
BIC	13.884322
-2*对数似然	3.4871145
-2*Firth 对数似然	-2.203446

梯度收敛
Firth 偏倚调整估计值

要构造将要提供给 10 个响应者的最终选择设计，您需要参数估计值的先验均值和方差。图 18.7 中的分析提供了参数均值的估计值（估计值）及其标准误差的估计值（标准误差）。您需要将标准误差视为标准差的先验估计值。下一步，要计算特性的方差估计值，需构造一个 JMP 表并将标准误差乘方。

4. 右击“参数估计值”报表并选择**制成数据表**。
5. 在新数据表中，右击**标准误差**列标题并选择**新建公式列 > 变换 > 平方**。

名为**标准误差 ^2** 的列将添加到数据表中。其值将作为选择模型参数的先验方差估计值。

图 18.8 最后一列中包含方差估计值的 “无标题” 数据表

	项	估计值	标准误差	标准误差^2
1	硬盘大小[40 GB]	-0.007125177	0.4991763970	0.2491770753
2	速度[1.5 GHz]	0.214880936	0.5265848733	0.2772916288
3	电池寿命[4 Hrs]	0.052925888	0.4895109018	0.2396209229
4	价格[$1500]	-0.978517706	0.7062009731	0.4987198144
5	价格[$1200]	0.076206148	0.7442825255	0.5539564778

注意：此时不要关闭 “无标题” 数据表。

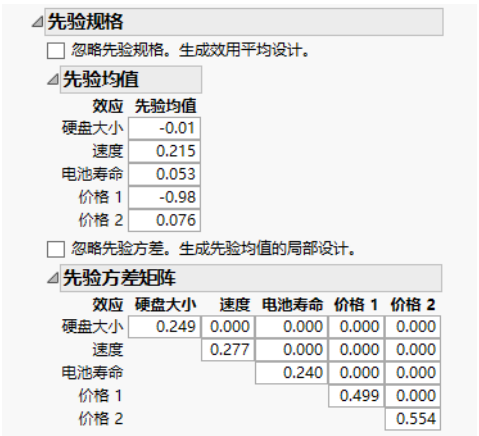
使用先验信息设计最终选择实验

在本节中，您要使用从笔记本电脑试行研究中获取的先验信息构造最终设计。最终设计将提供给一组 10 个响应者。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Laptop Factors.jmp。
2. 选择**实验设计 > 消费者研究 > 选择设计**。
3. 点击 “选择设计” 红色小三角并选择**加载因子**。
4. 点击**继续**。
5. 从 “无标题” 表中，将**估计值**列中的值输入 “选择设计” 窗口中的 “先验均值” 分级显示项 （图 18.9）。

   您可以从 “无标题” 表中复制和粘贴整个列，然后点击 “先验均值” 分级显示项中**先验均值**下的**磁盘大小**文本框，右击并选择 “粘贴”。
6. 从 “无标题” 表中，将**标准误差 ^2** 列中的值输入 “选择设计” 窗口的 “先验方差矩阵” 分级显示项中的对角线元素 （图 18.9）。逐一输入这些值，并将这些值舍入为三位小数。

图 18.9 试行研究中的先验均值和方差信息



7. 在 “生成设计”面板中，为**调查的数目**输入 2，为**每个调查期望的响应者数目**输入 5。

   这将为总共 10 个响应者提供设备，并允许使用两组不同的特征。

8. 点击**制作设计**。
9. 点击**制表**。

   设计表包含 160 行。 10 个研究响应者中每一个都对应 16 行。每个响应者具有 8 个选择集，每个选择集都有 2 个特征。共有两个调查，每个调查提供给 5 个响应者。通过以下计算得到 160 行：2 个特征 * 8 个选择集 * 2 个调查 * 5 个响应者 = 160 行。

   最终设计现已可提供给 10 个响应者。

运行设计和分析结果

在本节中，您要分析您从最终设计中获取的结果。特别是，您想要知道更改笔记本电脑的价格或其他特征会如何影响潜在购买人所感受到的意愿。该意愿称为笔记本电脑特性的**效用值**。

确定显著特性

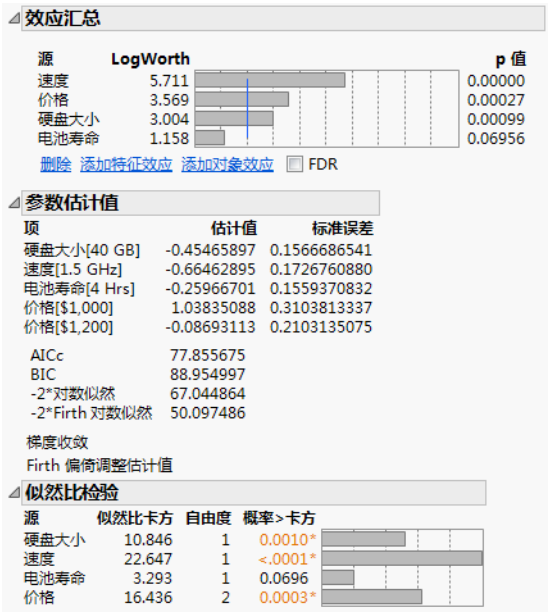
1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Laptop Results.jmp。
2. 运行**选择**脚本。

图 18.10 “选择模型”启动窗口

由于有多个调查和响应者，所以共有三个分组变量：**响应者**、**调查**和**选择集**。

3. 点击**运行模型**。

图 18.11 最终笔记本电脑设计的初始分析



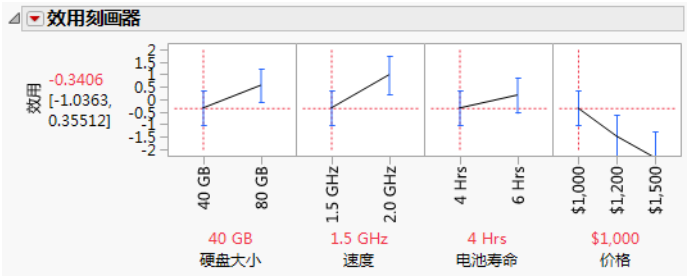
“效应汇总”分级显示项和“似然比检验”分级显示项指示**磁盘大小**、**速度**和**价格**在 0.05 水平下显著，“电池寿命”略微显著。

查找单位成本和权衡成本

下一步，使用 “刻画器”可查看效用值以及效用值如何随着笔记本电脑特性的更改而变化。

1. 点击“选择模型”红色小三角并选择**效用刻画器**。

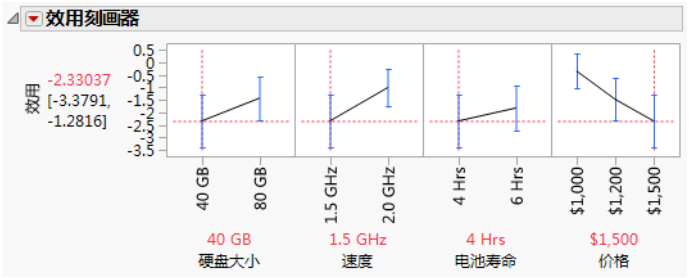
图 18.12 价格 = $1000 的效用刻画器



当每个特性值设置为其最低值时，效用值为 –0.3406。您最首要的任务是确定单位效用成本。

2. 将**价格**滑块移至 $1,500。

图 18.13 价格 = $1500 的效用刻画器

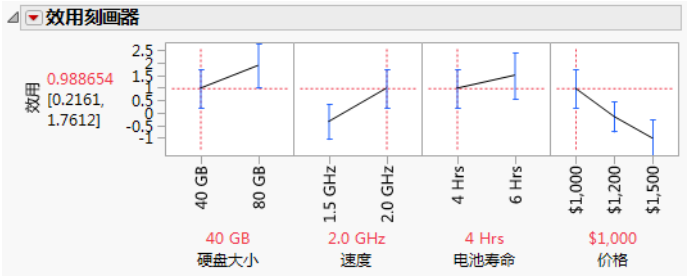


当**价格**从 $1,000 改为 $1,500 时，“效用”从 –0.3406 改为 –2.3303。也就是说，将笔记本电脑的价格提高 $500.00 会将效用（或意愿）降低大约 2 个单位。因此，您可以估计单位效用成本大约为 $250.00。

借助这一单位效用成本估计值，您现在可以改变其他特性，记下效用变化，而后求出与这一特性变化关联的近似货币值。例如，最显著的特性为**速度**（图 18.11）。

3. 在“效用刻画器”中，将“价格”重新设置为 $1,000（即其最低值）并将“速度”改为 2.0 GHz（其较高值）。

图 18.14 较高速度的效用值



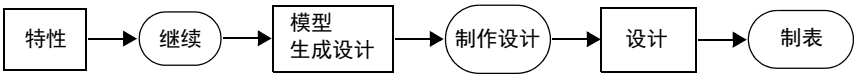
效用值从图 18.12 所示的初始值 –0.3406 变为 0.9886，总共改变了 1.3292 个单位。使用效用成本估计值或 $250.00，则 2.0 GHz 笔记本电脑相对于 1.5 GHz 笔记本电脑的价格涨幅可以计算为：1.3292*$250.00 = $332.30。这是选择研究所指示的美元价值，厂商可据此对该笔记本电脑特性定价。您可以对其他特性执行类似的计算。

“选择设计”窗口

“选择设计”窗口可逐步指导您构造用于对特性偏好建模的选择设计。您可以指定假设模型和先验信息。

“选择设计”窗口将在您逐步执行设计步骤时随之更新。分级显示项遵循图 18.15 中的流程图，它们被一些按钮分开，点击这些按钮后便有不同的分级显示出现。

图 18.15 选择设计流程

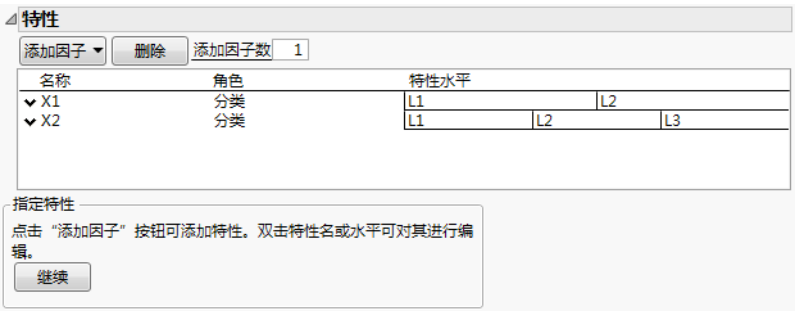


特性

选择设计中的特性只能是分类特性。

提示：填写完“特性”分级显示项后，考虑从红色小三角菜单中选择**保存因子**。这会将特性名称、角色和水平保存到一个数据表中，以便将来重新加载和重新使用。

图 18.16 “特性”分级显示项



“特性”分级显示项包含以下按钮：

添加因子 添加具有选定水平数的特性。

删除 删除选定的特性。

添加因子数 添加多个特性。输入要添加的特性数，点击**添加因子**，然后选择水平数。重复执行**添加因子数**，添加包含不同水平数的多个特性。

“特性”分级显示项包含以下列：

名称 特性的名称。特性具有 X1、X2 之类的给定默认名称。要更改某个名称，双击它，然后键入所需名称。

角色 将特性的“设计角色”指定为“分类”。

特性水平 特性名称或说明。要插入“特性水平”，点击默认水平并键入所需名称。

编辑“特性”分级显示项

- 要编辑某特性的“名称”，双击该特性名称。
- 要编辑某个“特性水平”，点击该水平。

特性列属性

对于每个特性，JMP 都将“值顺序”列属性保存到由“选择”平台构造的数据表中。“值顺序”列属性指定水平按照“特性”分级显示项中指定的顺序显示在报表中。请参见“列属性”附录中第 723 页的“值顺序”。

模型

“模型”分级显示项包含两部分：

- 您可以指定假设模型，其中包含您想要估计的所有效应。请参见第 505 页的“实验设计模型控制”。
- 您可以指定关于特性水平的先验知识，这样可以生成更好的设计。请参见第 505 页的“先验规格”。

实验设计模型控制

在“实验设计模型控制”分级显示项中指定假设模型。默认包含所有主效应。点击“交互作用”按钮可添加所有双因子交互作用。

图 18.17 “实验设计模型控制”分级显示项

构造设计表时，JMP 会将“选择”脚本保存至数据表。“选择”脚本使用在“实验设计模型控制”分级显示项中选定的模型启动“选择”平台。

“实验设计模型控制”分级显示项包含以下按钮：

主效应 为模型中的所有特性添加主效应。

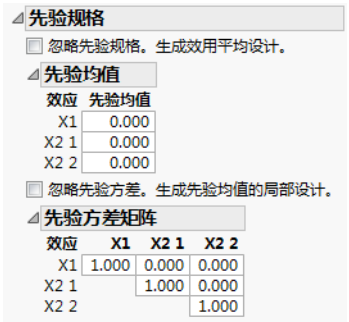
交互作用 添加所有二阶交互作用。若不想包括所有交互作用，请选择您想要删除的交互作用并点击**删除项**。

删除项 删除选定的效应。

先验规格

为模型参数的多元正态先验分布输入规格。在“先验均值”分级显示项中输入先验分布的均值，在“先验方差”分级显示项中输入其协方差矩阵。

图 18.18 “先验规格”分级显示项



您可以使用以下选项忽略先验规格：

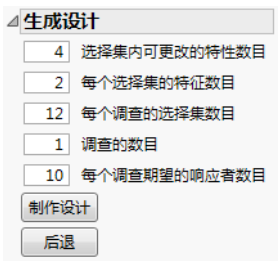
忽略先验规格。生成效用平均设计。 将先验均值设置为 0 并生成局部 D 最优设计。该设计称为**效用平均**设计。请参见 Huber and Zwerina (1996)。

忽略先验方差。生成先验均值的局部设计。 生成局部 D 最优设计。该局部设计考虑先验均值但忽略协方差矩阵。请参见 Huber and Zwerina (1996)。

生成设计

输入用于在“生成设计”面板中定义设计结构的规格。

图 18.19 咖啡示例的 “生成设计”面板



注意：图 18.19 摘自咖啡示例。请参见第 494 页的“选择设计示例”。

为以下每个项输入一个数字：

选择集内可更改的特性数目 输入小于等于特性总数的数目。该值经常设置为特性总数。不过，若是比较许多特性，而您想要简化响应者的选择流程，请将该值设置为小于特性总数的数目。

每个选择集的特征数目 输入响应者在表明偏好时必须从中选择的特征数。

每个调查的选择集数目 输入您想要从每个响应者获取的偏好数。

调查的数目 输入选择集的非重复集合数。若您想要将调查提供给多个响应者，该项很有用。

每个调查期望的响应者数目 输入响应者总数除以调查数目后的数目。

制作设计

一旦完成“生成设计”分级显示项，请点击**制作设计**生成该设计。该设计显示在“设计”分级显示项中。

设计

在给定的选择下，“设计”分级显示项显示最佳设计的试验。检查该设计以确保其满足您的需要。

图 18.20 咖啡示例的 “设计”分级显示项

设计

选择集	研磨	温度	时间	咖啡豆的量
1	Coarse	195	3	2
1	Medium	200	4	1.6
2	Medium	205	3	2
2	Coarse	200	3.5	2.4
3	Coarse	195	3	1.6
3	Medium	200	3.5	2.4
4	Coarse	200	3	2.4
4	Medium	205	3.5	2
5	Coarse	195	3.5	1.6
5	Coarse	200	3	2
6	Medium	195	3.5	2
6	Medium	205	4	2.4
7	Coarse	195	3.5	2.4
7	Medium	200	3	1.6
8	Medium	195	4	2
8	Coarse	205	3	1.6
9	Coarse	200	3.5	1.6
9	Medium	195	3	2.4
10	Medium	200	3	1.6
10	Coarse	205	4	2
11	Coarse	200	3.5	2
11	Medium	195	4	1.6
12	Coarse	200	4	2
12	Medium	205	3.5	2.4

○ 为特征和响应输出单独的表
◉ 将特征和响应合并在一个表中

注意：图 18.20 摘自咖啡示例。请参见第 494 页的“选择设计示例”。

注意：查找最优设计的算法基于随机开始设计。有鉴于此，您获取的设计不是唯一的。当您反复点击**后退**和**制作设计**按钮时，设计算法将生成不同的设计。

输出选项

选择以下输出选项之一：

为特征和响应输出单独的表 显示两个数据表：

- “选择特征”表列出每行中由**调查**和**选择集**列标识的特征。在每个选择集内，特征都由**选择 ID** 标识。该表有助于构造调查手段。

– “选择试验”表提供一个空**响应**列，您可在其中输入响应者偏好。每行都对应一个选择集。这些行按照**响应者**、**调查**和**选择集**排序。选择集 ID 在接下来的列中提供，再后面是响应列。为“响应”列中的响应者偏好输入选择集 ID。

将特征和响应合并在一个表中 提供一个“选择特征”表，其中有一个空的**响应指示符**列可供您输入响应者偏好。每行都对应一个特征。该表按照**响应者**、**调查**和**选择集**排序。为响应者偏好的特征输入值 1（或其他非零数值指示符），为该选择集内的其他特征输入指示符 0。

注意：您在**响应指示符**列中输入的值必须为数值。

制表

点击**制表**可构造您在第 507 页的“输出选项”中选定的一个或多个表。在“选择特征”表的表面板中，有一个**选择**脚本。运行该脚本，然后点击**运行模型**分析您的实验结果。

有关“选择模型”报表的信息，请参见《消费者研究》。

选择设计选项

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在“设计角色”区域中，选择合适的角色。

加载因子 加载使用“保存因子”选项保存的因子。

设置随机种子 设置随机种子，JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

– 初始化用于生成设计的搜索算法
– 随机化用于构造设计的“试验顺序”
– 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的“实验设计对话框”脚本中。

模拟响应 将响应概率和包含选择模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击“制表”时，将出现以下情况：

– 包含模拟选择响应公式的新列将添加至设计表。公式和值基于在设计窗口中指定的模型。

– 随即显示“模型”窗口，您可在其中设置模型效应的边缘效用值。
– 名为**选择模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对边缘效用做出更改。

当您点击“应用”时，模拟的选择响应值的公式将在“模拟的选择”列中更新。若再次点击“应用”，该列中的公式和值将随之更新。

请参见“定制设计”一章中第 101 页的“模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见《基本分析》。

开始数 支持您指定用于构造设计的随机开始数。请参见第 509 页的“Bayes D 最优性和设计构造”。

高级选项 不可用于“选择设计”。

将脚本保存至脚本窗口 为您在“选择设计”窗口中指定的设计创建脚本，并将其保存到打开的脚本窗口中。

技术细节

Bayes D 最优性和设计构造

Bayes D 最优性准则是多项式 Logit 模型中参数估计量最大似然的信息矩阵对先验分布取得的行列式的对数期望值。“选择设计”平台针对表示先验概率分布的参数向量的样本将该期望值最大化。请参见 Kessels et al.(2011)。

对于部分特征设计， JMP 使用两阶段设计算法：

1. 使用**特性平衡**方法可确定每个选择集内的不变特性。
2. 使用 Bayes D 最优性来确定非不变特性的水平。

特性平衡意味着该算法尝试平衡每个特性在整个设计中保持不变的次数。若两个或更多特性保持不变，该算法尝试平衡在设计中保持不变的成对特性的发生。

通过确定非不变特性的水平以优化 Bayes D 最优准则。找到随机开始设计。然后使用坐标交换算法生成非不变特性的水平并进行评估，直到 Bayes D 最优性准则得到优化。计算过程（涉及相对于多元正态先验求积分）使用 Gotwalt et al. (2009)。

注意：Bayes D 最优性准则可生成选择集，这些选择集中的某些非不变特性具有相同的水平。当改变特征中的非不变水平将生成无法提供信息的选择集时（此时所有特征都具有极高或极低的概率）就会发生这种情况。

效用平均和局部 D 最优设计

您可以使用“选择设计”平台通过将先验均值设置为 0 来生成效用平均设计。在效用平均设计中，选择集内的所有选择的概率都相同。请参见 Huber and Zwerina (1996)。

您还可以生成局部 D 最优设计。局部设计考虑先验均值，但不包括来自先验协方差矩阵的任何信息。请参见 Huber and Zwerina (1996)。

第 19 章

MaxDiff 设计

创建选择最优项和最差项的设计

MaxDiff（**差值最大化尺度**）研究可替代使用标准偏好尺度确定要评定的项的相对重要性的研究。在 MaxDiff 研究中，响应者仅报告在一个小的选择集中最想选的选项和最不想选的选项。这会强制响应者按照偏好对选项排名，这样生成的排名往往比使用标准偏好尺度获得的排名更具有确定性。

当您需要构造一个包含某个因子的选择集的设计（该因子可以作为 MaxDiff 研究的一部分向响应者提供）时，使用 MaxDiff 平台。进行您的研究，然后使用 MaxDiff 分析平台分析数据。

图 19.1 MaxDiff 设计表

	对象	选择集	糖果	选择
1	1	1	Reese's Cups	•
2	1	1	Hershey Bar	•
3	1	1	Snickers	•
4	1	1	Butterfinger	•
5	1	2	Butterfinger	•
6	1	2	Heath Bars	•
7	1	2	Plain M&Ms	•
8	1	2	Snickers	•
9	1	3	Plain M&Ms	•
10	1	3	Snickers	•
11	1	3	Peanut M&Ms	•
12	1	3	Hershey Bar	•
13	1	4	Snickers	•
14	1	4	Peanut M&Ms	•
15	1	4	Reese's Cups	•
16	1	4	Heath Bars	•

目录

“MaxDiff 设计”平台概述 513
MaxDiff 设计的示例 513
MaxDiff 设计启动窗口 516
MaxDiff 窗口 517
“设计选项”分级显示项 517
“设计”分级显示项 517
配对关联矩阵 517
制表 517
MaxDiff 选项 518

“MaxDiff 设计”平台概述

选择集是一个项的集合，响应者必须从中选择最想选的项（**最优**项）和最不想选的项（**最差**项）。使用 MaxDiff 平台指定要在选择集中显示的项数和要提供给响应者的选择集数。

MaxDiff 平台为您提供一个调查。若您有几个响应者，可以为所有响应者提供同一调查或为每个响应者构造一个调查。

构造 MaxDiff 设计以尽可能匹配平衡不完整区组设计的关联矩阵。这意味着每对项在设计中出现的频数相等。若选择集数允许针对选择集中指定的项数构造平衡不完整区组，则构造的设计为平衡不完整区组设计。

考虑的项或特征被视为处理因子的水平。选择集被视为区组。在 MaxDiff 研究中，区组大小小于处理数。

MaxDiff 设计的示例

您是公司办公用品的采购员，需要为节日聚会购买糖果。首先，您需要确定员工喜欢哪几种糖果。为了了解这一信息，您执行了 MaxDiff 研究。

您请求五个随机选择的同事来对七种糖果进行评级。根据您以前研究的经验，您知道评级员很难按偏好程度对七种糖果排名。为此，您创建了一个设计，它包含大小为 4 的选择集。为了便于管理，您按 MaxDiff 研究的要求来组织调查：您要求每个同事在七个选择集的每一个中选择自己最喜欢的糖果和最不喜欢的糖果。（这些选择构造了一个平衡不完整区组设计。请参见 Cochran and Cox (1957)。）您对每个同事进行同样的调查。

创建设计

构造一个表，它列出您的选择集的项或特征。在本例中，已构造了项表 Candy Profiles.jmp。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Candy Profiles.jmp。
   该表列出关注的七种糖果类型。
2. 选择**实验设计** > **消费者研究** > **MaxDiff 设计**。
3. 从“选择列”列表选择**糖果**并点击 **X，因子**。
4. 点击**确定**。
5. 在“设计选项”分级显示项中，执行以下操作：
   - 将**每个选择集的特征数目**设置为 4。
   - 将**选择集数目**设置为 7。

注意：在第 6 步中设置随机种子可确切重现本例所示的结果。自行构造设计时，该步骤不是必需的。

6. （可选）点击“MaxDiff 研究”红色小三角菜单，然后选择**设置随机种子**。键入 12345 并点击**确定**。
7. 点击**制作设计**。

   “设计”分级显示项显示 7 个选择集，每个包含 4 种糖果类型。“配对关联矩阵”提供选择集数矩阵，该矩阵中每对糖果类型都显示在一起。矩阵的对角线是每对糖果类型所在的选择集数。对于本例，每一种糖果都显示在 4 个选择集中，并且每一个其他糖果类型在四个选择集中的两个中显示。
8. 点击**制表**。

图 19.2 糖果偏好调查的部分设计表

	对象	选择集	糖果	选择
1	1	1	Reese's Cups	•
2	1	1	Hershey Bar	•
3	1	1	Snickers	•
4	1	1	Butterfinger	•
5	1	2	Butterfinger	•
6	1	2	Heath Bars	•
7	1	2	Plain M&Ms	•
8	1	2	Snickers	•
9	1	3	Plain M&Ms	•
10	1	3	Snickers	•
11	1	3	Peanut M&Ms	•
12	1	3	Hershey Bar	•
13	1	4	Snickers	•
14	1	4	Peanut M&Ms	•
15	1	4	Reese's Cups	•
16	1	4	Heath Bars	•

设计表包含一个**选择**列用于记录糖果偏好。对于每个选择集，记录 1 表示最喜欢的糖果，-1 表示最不喜欢的糖果， 0 表示其他两种糖果。

分析研究结果

您执行了研究并在 Candy Survey.jmp 中记录了数据。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Candy Survey.jmp。

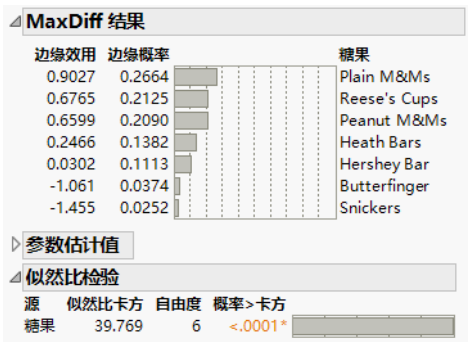
   该表显示分别向**对象**列中所列的五名响应者进行调查的结果。
2. 选择**分析** > **消费者研究** > MaxDiff。
3. 点击 “选择数据表”，选择 Candy Survey，然后点击**确定**。
4. 分配列角色：
   - 选择**选择**并点击**响应指示符**。
   - 选择**试验对象**并点击**对象 ID**。
   - 选择**选择集**并点击**选择集 ID**。
   - 在“构造特征效应”面板中选择**糖果**并点击**添加**。

图 19.3 已完成的 MaxDiff 分析启动窗口

因为您指定了最优选择为 1、最差选择为 -1，因此不用修改启动窗口左下角的 “最优”和“最差”选择指示符。

5. 点击**运行模型**。

图 19.4 MaxDiff 报表



该报表指示**糖果**是显著的。具有最高效用的三种糖果为 Plain M&Ms、 Reese’s Cups 和 Peanut M&Ms。

6. 点击 “MaxDiff 模型”红色小三角菜单并选择 **“所有水平比较”报表**。

图 19.5 糖果类型的 “所有水平比较” 报表

“所有水平比较” 报表

差值（行-列） 差值的标准误差 Wald P 值	Butterfinger	Heath Bars	Hershey Bar	Peanut M&Ms	Plain M&Ms	Reese's Cups	Snickers
Butterfinger	0	-1.3079	-1.0915	-1.7212	-1.964	-1.7378	0.39343
	0	0.52954	0.52038	0.54692	0.56495	0.54938	0.5059
		0.01964	0.04478	0.0038	0.00162	0.00364	0.44306
Heath Bars	1.30792	0	0.21642	-0.4133	-0.656	-0.4299	1.70135
	0.52954	0	0.47619	0.48741	0.49094	0.46959	0.54758
	0.01964		0.65286	0.40343	0.19185	0.36752	0.0042
Hershey Bar	1.0915	-0.2164	0	-0.6297	-0.8725	-0.6463	1.48493
	0.52038	0.47619	0	0.47406	0.49984	0.49096	0.53808
	0.04478	0.65286		0.19443	0.09149	0.19835	0.00992
Peanut M&Ms	1.72121	0.41329	0.62971	0	-0.2427	-0.0166	2.11463
	0.54692	0.48741	0.47406	0	0.49706	0.49054	0.56901
	0.0038	0.40343	0.19443		0.62896	0.97325	0.00086
Plain M&Ms	1.96396	0.65604	0.87246	0.24275	0	0.22616	2.35738
	0.56495	0.49094	0.49984	0.49706	0	0.49327	0.58562
	0.00162	0.19185	0.09149	0.62896		0.65002	0.00037
Reese's Cups	1.7378	0.42988	0.6463	0.01659	-0.2262	0	2.13123
	0.54938	0.46959	0.49096	0.49054	0.49327	0	0.5714
	0.00364	0.36752	0.19835	0.97325	0.65002		0.00083
Snickers	-0.3934	-1.7013	-1.4849	-2.1146	-2.3574	-2.1312	0
	0.5059	0.54758	0.53808	0.56901	0.58562	0.5714	0
	0.44306	0.0042	0.00992	0.00086	0.00037	0.00083	

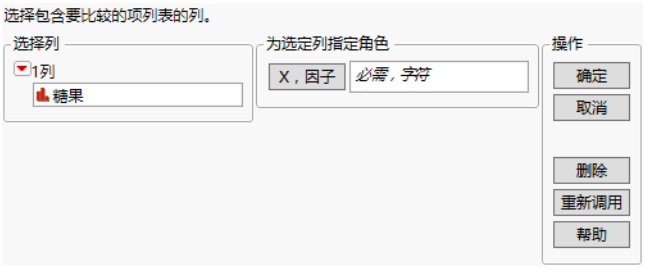
该比较报表指示哪些糖果类型对在效用上有显著差异。每个单元格中的第三个条目是由行项的效用减列项的效用定义的差值的 *p* 值。*p* 值颜色的强度指示差值的显著程度。蓝色或红色着色指示差值（行 - 列）是正数还是负数。不调整 *p* 值来控制多重比较误差率，该值只应作为参考。有关 “所有水平比较” 报表的详细信息，请参见 《消费者研究》。

MaxDiff 设计启动窗口

要使用 “MaxDiff 设计” 平台，您需要一个起始数据表。您的起始数据表必须包含特征数据列，它列出要提供给响应者进行评级的项。

在 JMP 中激活起始数据表后，选择**实验设计** > **消费者研究** > **MaxDiff 设计**。若您没有活动数据表，系统会提示您导航到起始数据表。

图 19.6 使用 Candy Profiles.jmp 的 MaxDiff 启动窗口

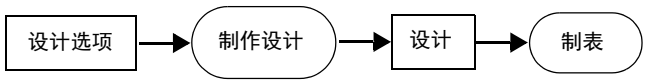


X，因子 包含项的字符列，响应者将对这些项进行评级。

MaxDiff 窗口

“MaxDiff 研究”窗口随着您执行各个设计步骤而更新。分级显示项遵循图 19.7 中的流程图，它们被一些按钮分开，点击这些按钮后便有不同的分级显示出现。

图 19.7 MaxDiff 设计流程



“MaxDiff 研究”窗口打开，显示“设计选项”分级显示项。点击“制作设计”后，将显示“设计”分级显示项。要构造设计表，请点击“制表”。

“设计选项”分级显示项

指定以下信息：

每个选择集的特征数目 要包括在每个选择集中的项数。

选择集数目 要提供给响应者进行评级的项集总数。

“设计”分级显示项

“设计”分级显示项使用相邻正整数标识选择集。列出构成每个选择集的项。

配对关联矩阵

“配对关联矩阵”分级显示项提供选择集数矩阵，该矩阵中每对因子水平都显示在一起。矩阵的对角线是每对因子水平所在的选择集数。

制表

“制表”按钮用于创建设计表。设计表包含四列：

对象 初始填充为 1。使用响应者的相应标识符替换。

> **提示：**要轻松添加响应者标识符，请参见《使用 JMP》。使用“列”>“重新编码”将标识符更改为名称。

选择集 每个选择集的指示符。

因子 您指定的因子的水平。这些是选择集中的项。

选择 您可以输入结果的列。使用数值来指示最优选择、最差选择以及介于这两者之间的选择。通常使用值 1、-1 和 0，它们是 MaxDiff 分析平台（位于“分析”>“消费者调查”>“MaxDiff”中）进行分析所必需的。

MaxDiff 选项

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的“试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 选择该选项以模拟响应值。当您点击 “制表”时，将显示一个 “模拟选择”窗口以及设计表，已将 “概率”和 “模拟的选择”列添加到该设计表。“模拟的选择”公式列包含随机响应，“概率”公式列包含其概率。

警告：要更改在模拟响应中使用的模型，请在 “模拟选择”窗口的 “边缘效用”列中为所有因子水平 （最后一个除外）输入值。因为边缘效用的和必须为零，因此不允许您编辑最后一个水平的边缘效用。您指定剩余 “边缘效用”值后，会相应调整因子的最后一个水平。

开始数 在构造设计时使用的随机开始数。默认情况下将该值设置为 10。

高级选项 不适用于 “MaxDiff 设计”。

将脚本保存至脚本窗口 为您指定的设计创建脚本并将其放入打开的脚本窗口。

第 20 章

JMP PRO 覆盖阵列

检测组件交互作用失败

覆盖阵列用于检验因组件或子系统之间的交互作用而导致出现失败的确定性系统。设计目标是要揭示是否有任何交互作用引起了系统失败。应用领域包括软件、电路和网络设计。

由于检验是确定性的，所以推动这种设计的重点就是对覆盖所有必需交互作用的需要。“覆盖阵列”平台构造高效的覆盖阵列。您还可以排除对您的检验方案不可行的因子水平组合。

图 20.1 强度为 3 的覆盖阵列

设计

试验	X1	X2	X3	X4	X5
1	L1	L1	L1	L1	L1
2	L1	L1	L1	L2	L2
3	L2	L2	L2	L2	L2
4	L2	L1	L2	L2	L1
5	L2	L2	L2	L1	L2
6	L2	L2	L1	L1	L1
7	L2	L1	L1	L1	L2
8	L1	L2	L2	L1	L1
9	L1	L2	L1	L2	L2
10	L1	L2	L2	L2	L1
11	L1	L1	L2	L1	L2
12	L2	L2	L1	L2	L1

目录

覆盖阵列概述 521
不含因子水平限制的覆盖阵列的示例 522
创建设计 523
分析实验数据 525
包含因子水平限制的覆盖阵列的示例 526
创建设计 526
分析实验数据 531
“覆盖阵列”窗口 532
因子 532
限制因子水平组合 533
设计 535
量度 538
输出选项 538
“覆盖阵列”数据表 538
覆盖阵列选项 540
覆盖阵列的技术细节 540
用于优化的算法 541
用于量度的公式 541

JMP PRO 覆盖阵列概述

您可以使用覆盖阵列检验因组件或子系统之间的交互作用而导致出现失败的系统。覆盖阵列经常用在软件、电路和网络设计领域，这些领域中往往存在以下状况：

- 检验成本通常很高。
- 检验侧重于揭示出现失败的交互作用。
- 检验试验通常是确定性的，结果为成功或失败。
- 重复试验会因为生成完全相同的结果而造成浪费。
- 设计效率取决于在不包含冗余试验的前提下覆盖了多少可能条件。

由于系统检验成本很高，所以减少检验的数量至关重要。通常无法检验所有可能的交互作用，而且往往也没有必要。经验证明多数失败都来自少数组件的交互作用。可能导致失败的组件的最多组合的大小（称为**强度**）决定了设计的规模。

在“覆盖阵列”平台中，您可以指定所需的设计强度。在适当的情况下，您可以定义不允许的因子水平组合。“覆盖阵列”平台构造满足您需求的高效设计。它提供了量度，支持您用设计覆盖率来评估设计质量。它还在设计的数据表中提供了脚本，可支持您分析结果。

覆盖阵列通常用在特定的因子水平设置组合不可行的情形中。“覆盖阵列”平台能够找到非常高效的覆盖阵列，即便因子水平组合受限时也不例外。

有关覆盖阵列的结构和用于计算覆盖阵列的算法的背景信息，请参见 Colbourn (2004)、Colbourn et al. (2011)、Hartman and Raskin (2004) 以及 Martirosyan (2003)。有关因子水平受限的覆盖阵列的详细信息，请参见 Cohen et al.(2007) 和 Morgan (2009)。

JMP PRO 覆盖阵列和强度

强度为 t 的覆盖阵列是一种检验 *t* 个因子水平设置的所有组合的设计。考虑使用 *k* 个因子的特定设置定义的某个交互作用。若涉及该交互作用的所有检验都出现失败，则该交互作用**检测**到失败。使用该术语，强度为 *t* 的设计支持您检测与多达 *t* 个因子的任意交互作用关联的失败。

覆盖阵列在文献中还称为**因子覆盖设计**。有关背景和详细信息，请参见 Yilmaz et al.(2014)、Cohen et al. (2003) 以及 Dalal and Mallows (1998)。

要演示覆盖阵列的性质，可以考虑涉及七个分类因子且每个因子包含两个水平的情形。您想要检验因子水平的所有成对组合。

该情形中有可能使用的一个设计是一个 8 试验的分辨率 III 主效应设计：

图 20.2 强度为 2 的分辨率 III 设计

设计

试验	X1	X2	X3	X4	X5	X6	X7
1	L2	L2	L2	L2	L2	L2	L2
2	L1	L2	L1	L2	L1	L2	L1
3	L2	L1	L1	L2	L2	L1	L1
4	L1	L1	L2	L2	L1	L1	L2
5	L2	L2	L2	L1	L1	L1	L1
6	L1	L2	L1	L1	L2	L1	L2
7	L2	L1	L1	L1	L1	L2	L2
8	L1	L1	L2	L1	L2	L2	L1

请注意，该因子设计是一个强度为 2 的覆盖阵列，因为任意两个因子的水平的所有成对组合均显示出来。例如，对于 X1 和 X2，以下组合中每个组合显示两次：

- L2 和 L1
- L2 和 L2
- L1 和 L1
- L1 和 L2

不过，强度为 2 的覆盖阵列仅需要 6 次试验...：

图 20.3 强度为 2 的覆盖阵列

设计

试验	X1	X2	X3	X4	X5	X6	X7
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	1
3	2	2	2	1	1	1	2
4	2	1	1	2	2	1	2
5	1	2	1	2	1	2	2
6	1	1	2	1	2	2	1

任意两个因子的水平的所有成对组合在六次试验中都至少出现一次。“覆盖阵列”设计比“分辨率 III”设计更高效，因为它可以用更少的试验实现强度为 2 的覆盖率。

覆盖阵列的效率通过为实现所需覆盖率而需要执行的试验次数来衡量。试验次数越少，设计越高效。

JMP PRO 不含因子水平限制的覆盖阵列的示例

本例中的数据与软件测试领域中的互操作性相关。关注的四个因子为：

- Web 浏览器（Safari、IE、Firefox、Chrome、其他）
- 操作系统（Windows 或 macOS）
- RAM（4 MB 、8 MB 或 16 MB）
- 连接速度有三个设置（0-1 Mbps、1-5 Mbps 或大于 5 Mbps）

您关注的是找出这些因子的哪些组合可能引起失败。

响应是针对因子设置的每种组合系统能否正常运行。

检验设置的每种组合需要执行 90 = 5x2x3x3 次试验。要保持试验规模的可操作性，您决定需要强度为 3 的覆盖率，指示检验任意三个因子的所有组合。

JMP PRO 创建设计

通过执行以下步骤创建强度为 3 的覆盖阵列。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Software Factors.jmp。
   Software Factors.jmp 数据表包含因子及其设置。
2. 选择**实验设计** > **特殊目的** > **覆盖阵列**。
3. 从**强度 : t =** 旁边的菜单中，选择 3。
4. 点击“覆盖阵列”红色小三角，选择**加载因子**。
   “因子”分级显示项中填充了四个因子及其水平。

图 20.4 Software Factors 的“因子”分级显示项

5. 点击**继续**。
   “限制因子水平组合”分级显示项随即打开，您可在其中输入对设计设置的限制。由于该设计没有限制，您无需更改默认选择**无**。
6. 点击**制作设计**。
   “设计”分级显示项打开，显示一个 45 次试验的设计。

图 20.5 Software Factors 的“设计”和“量度”分级显示项

设计

试验	Web 浏览器	操作系统	RAM	连接速度
1	Safari	Macintosh	4 MB	>5 Mbps
2	Safari	Windows	4 MB	0-1 Mbps
3	Safari	Macintosh	4 MB	1-5 Mbps
4	Safari	Windows	8 MB	>5 Mbps
5	Safari	Macintosh	8 MB	0-1 Mbps
6	Safari	Macintosh	8 MB	1-5 Mbps
7	Safari	Macintosh	16 MB	>5 Mbps
8	Safari	Macintosh	16 MB	0-1 Mbps
9	Safari	Windows	16 MB	1-5 Mbps
10	IE	Windows	4 MB	>5 Mbps
11	IE	Macintosh	4 MB	0-1 Mbps
12	IE	Macintosh	4 MB	1-5 Mbps

优化 最多迭代次数: 250

量度

试验次数: 45

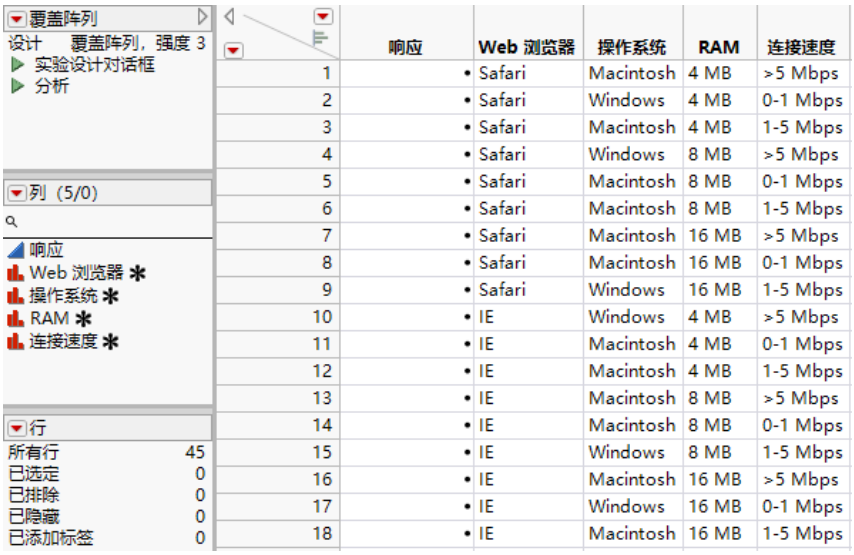
t	覆盖率	多样性
3	100.00	68.33
4	50.00	100.00

在“量度”分级显示项中，考虑 $t = 3$ 对应的行。“覆盖率”为 100%，指示该设计覆盖 100% 的三因子交互作用。这正是您所需要的，因为您请求的是强度为 3 的设计。对于 $t = 3$，“多样性”列指示所显示的 68.33% 的三因子交互作用都是非重复的。三因子组合只有少量重复。

对于 $t = 4$，“覆盖率”为 50%，指示该设计覆盖一半的四因子交互作用。四因子设置共有 90 个可能的非重复组合。设计中的 45 次试验构成了这些非重复组合的一半。“多样性”值为 100% 再次印证了这一事实：没有任何四因子交互作用是重复的。

7. 点击**制表**。

图 20.6 Software Factors 的部分设计表



覆盖阵列
设计 覆盖阵列，强度 3
实验设计对话框
分析

列 (5/0)
响应
Web 浏览器 *
操作系统 *
RAM *
连接速度 *

行
所有行 45
已选定 0
已排除 0
已隐藏 0
已添加标签 0

	响应	Web 浏览器	操作系统	RAM	连接速度
1	•	Safari	Macintosh	4 MB	>5 Mbps
2	•	Safari	Windows	4 MB	0-1 Mbps
3	•	Safari	Macintosh	4 MB	1-5 Mbps
4	•	Safari	Windows	8 MB	>5 Mbps
5	•	Safari	Macintosh	8 MB	0-1 Mbps
6	•	Safari	Macintosh	8 MB	1-5 Mbps
7	•	Safari	Macintosh	16 MB	>5 Mbps
8	•	Safari	Macintosh	16 MB	0-1 Mbps
9	•	Safari	Windows	16 MB	1-5 Mbps
10	•	IE	Windows	4 MB	>5 Mbps
11	•	IE	Macintosh	4 MB	0-1 Mbps
12	•	IE	Macintosh	4 MB	1-5 Mbps
13	•	IE	Macintosh	8 MB	>5 Mbps
14	•	IE	Macintosh	8 MB	0-1 Mbps
15	•	IE	Windows	8 MB	1-5 Mbps
16	•	IE	Macintosh	16 MB	>5 Mbps
17	•	IE	Windows	16 MB	0-1 Mbps
18	•	IE	Macintosh	16 MB	1-5 Mbps

该设计在数据表中提供。请注意位于左上角的 “表”面板中的以下内容：

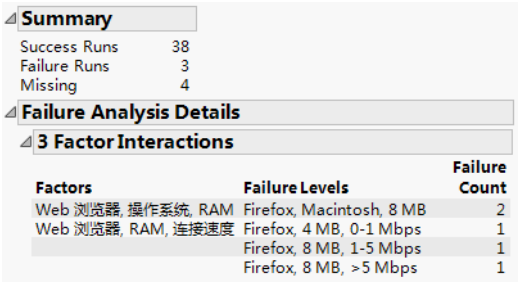
- “设计”注释指示这是一个强度为 3 的覆盖阵列。
- “实验设计对话框”脚本重现“覆盖阵列”窗口设置。
- “分析”脚本分析实验数据。

JMP PRO 分析实验数据

现在已经有了设计表，现在可以执行实验并将数据记录在设计表的**响应**列中 （图 20.6）。您的实验结果包含在 Software Data.jmp 样本数据表中。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Software Data.jmp。
2. 在 “表”面板中，点击**分析**脚本旁边的绿色小三角。

图 20.7 软件实验数据分析



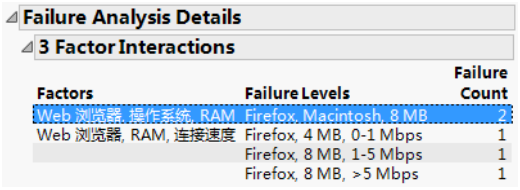
“汇总”分级显示项指示三个检验失败，四个检验未生成成功或失败结果。

“失败分析详细信息”分级显示项根据关联的三因子交互作用给出了细分的失败信息。该分级显示项仅列出所有检验都导致失败的因子水平的组合。若有任何涉及设置的给定三因子组合的任何检验导致成功，则设置的这个三因子组合不能对系统失败负责。

两次失败与 Web 浏览器设置为 Firefox、**操作系统**设置为 macOS 且 RAM 设置为 8 MB 相关。请注意，无论**连接速度**如何设置，该组合都导致失败。

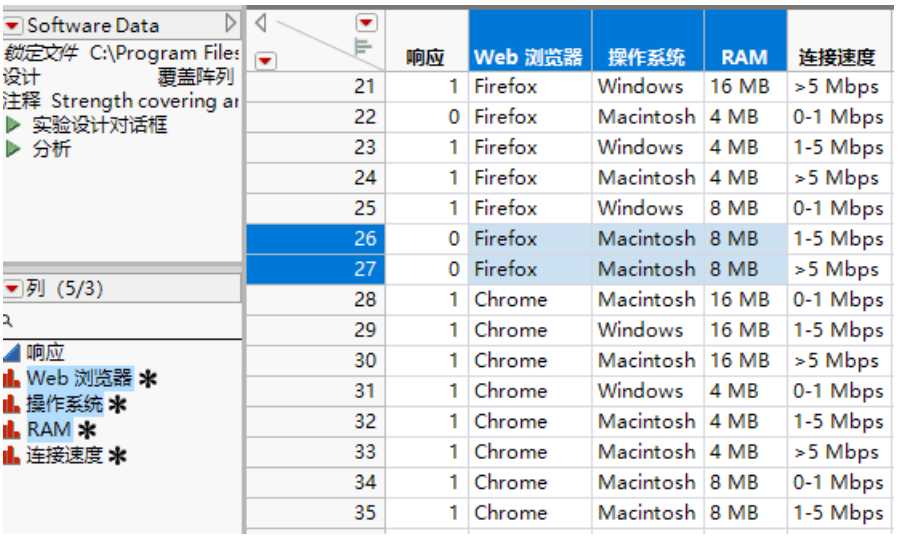
3. 选择 “3 因子交互作用”报表中的第一行。

图 20.8 在分析报表中选择交互作用



该操作将选定数据表中的相应行和列。

图 20.9 数据表中相应的行列选择



	响应	Web 浏览器	操作系统	RAM	连接速度
21	1	Firefox	Windows	16 MB	>5 Mbps
22	0	Firefox	Macintosh	4 MB	0-1 Mbps
23	1	Firefox	Windows	4 MB	1-5 Mbps
24	1	Firefox	Macintosh	4 MB	>5 Mbps
25	1	Firefox	Windows	8 MB	0-1 Mbps
26	0	Firefox	Macintosh	8 MB	1-5 Mbps
27	0	Firefox	Macintosh	8 MB	>5 Mbps
28	1	Chrome	Macintosh	16 MB	0-1 Mbps
29	1	Chrome	Windows	16 MB	1-5 Mbps
30	1	Chrome	Macintosh	16 MB	>5 Mbps
31	1	Chrome	Windows	4 MB	0-1 Mbps
32	1	Chrome	Macintosh	4 MB	1-5 Mbps
33	1	Chrome	Macintosh	4 MB	>5 Mbps
34	1	Chrome	Macintosh	8 MB	0-1 Mbps
35	1	Chrome	Macintosh	8 MB	1-5 Mbps

有三次失败与 Web 浏览器、RAM 和连接速度的组合相关联。请注意，其中两次失败（Firefox、8 MB、1-5 Mbps 和 Firefox、8 MB、>5 Mbps）位于 Web 浏览器、操作系统和 RAM 交互作用的两次失败之中。选择报表的这些行中的任意行将选定数据表中的相应行列。

JMP PRO 包含因子水平限制的覆盖阵列的示例

本例借鉴 Dalal and Mallows (1998) 中所述的示例。拨打电话（近距离电话）呼叫接听电话（远距离电话）。每一次通话都会经由类型为 A 或 B 的界面。关注的五个因子为：

- 市场：USA、UK、Canada、France、Mexico
- 近距离电话：ISDN、Bus（商用）、Coin、Res（家用）
- 近距离界面：A 或 B
- 远距离电话：ISDN、Bus（商用）、Coin、Res（家用）
- 远距离界面：A 或 B

您关注的是这些因子对的哪些组合可能引起失败。不过，有些组合是不可能的：

- 两部电话（近距离或远距离）上的 ISDN 线路都不能使用界面 A。
- 拨打电话（近距离）上的商用和家用线路都不能使用界面 B。

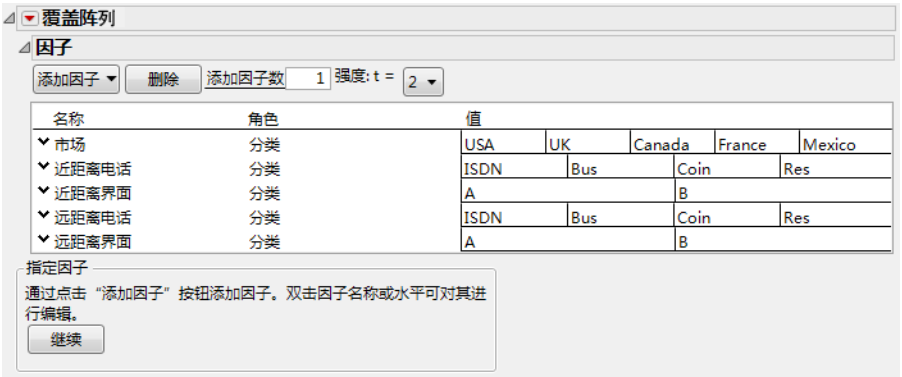
JMP PRO 创建设计

因子及其设置在数据表 Phone Factors.jmp 中提供。通过执行以下步骤创建强度为 2 的覆盖阵列。

JMP PRO 加载因子

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Phone Factors.jmp。
   Phone Factors.jmp 数据表包含因子及其设置。
2. 选择**实验设计** > **特殊目的** > **覆盖阵列**。
   请注意，**强度 : t =** 旁边的菜单默认设置为 2。
3. 点击 “覆盖阵列” 红色小三角并选择**加载因子**。
   “因子” 分级显示项中填充了五个因子及其水平。

图 20.10 电话因子的 “因子” 分级显示项



4. 点击**继续**。
   “限制因子水平组合” 分级显示项随即打开。

JMP PRO 限制因子水平组合

您可以采用两种方式指定不被允许的组合：

- 使用不被允许的组合过滤器
- 使用不被允许的组合脚本

过滤器可以采用直观方式指定不被允许的组合。脚本可以采用快速简单的方式指定不被允许的组合，但要求您已编写或保存脚本。在本例中，若不想使用过滤器指定组合，可跳至第 529 页的“使用脚本指定不被允许的组合”。

回想一下存在以下限制：

- 两部电话（近距离或远距离）上的 ISDN 线路都不能使用界面 A。
- 拨打电话（近距离）上的商用和家用电话都不能使用界面 B。

JMP PRO 使用过滤器指定不被允许的组合

使用该方法可通过过滤器界面输入不被允许的组合。或者，您可以粘贴第 529 页的 “使用脚本指定不被允许的组合”中所示的脚本。

1. 选择**使用不被允许的组合过滤器**。
2. 从 “添加过滤器因子”列表中，选择**近距离电话**和**近距离界面**，然后点击**添加**按钮。
3. 按 Ctrl 键并点击**近距离电话**之下的 ISDN 以及**近距离界面**之下的 A。

图 20.11 显示第一个约束的 “不被允许的组合”面板

这两个区组都应变暗。您已经添加了限定拨打电话（近距离）上的 ISDN 线路不能使用界面 A 的约束。

4. 点击**或**。
5. 从 “添加过滤器因子”列表中，选择**远距离电话**和**远距离界面**，然后点击**添加**按钮。
6. 按住 Ctrl 键并点击**远距离电话**之下的 ISDN 以及**远距离界面**之下的 A。

   您已经添加了限定接听电话 （远距离）上的 ISDN 线路不能使用界面 A 的约束。
7. 点击**或**。
8. 从 “添加过滤器因子”列表中，选择**近距离电话**和**近距离界面**，然后点击**添加**按钮。
9. 按 Ctrl 键并点击**近距离电话**之下的 Bus 和 Res 以及**近距离界面**之下的 B。

   您已经添加了限定拨打电话（近距离）上的“商用”和“家用”线路不能使用界面 B 的约束。

图 20.12 完成的不被允许的组合过滤器

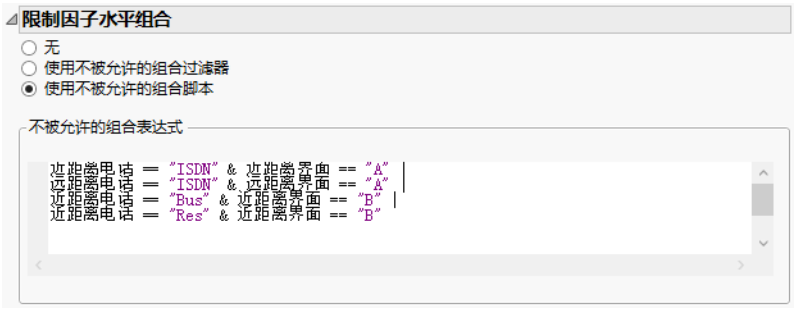
JMP PRO 使用脚本指定不被允许的组合

您也可以通过构造脚本来指定不被允许的组合。加载因子后 （第 527 页的 “加载因子”），请执行以下操作：

1. 点击**继续**。
2. 选择**使用不被允许的组合脚本**。
3. 将以下脚本复制并粘贴到 “不被允许的组合表达式” 脚本框中：

```
(近距离电话 == "ISDN" & 近距离界面 == "A") |
(远距离电话 == "ISDN" & 远距离界面 == "A")  |
(近距离电话 == "Bus" & 近距离界面 == "B") |
(近距离电话 == "Res" & 近距离界面 == "B")
```

图 20.13 完成的 “不被允许的组合脚本”窗口



JMP PRO 构造设计表

注意：设置下两步中的“随机种子”将重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

1. 点击“覆盖阵列”红色小三角并选择**设置随机种子**。
2. 输入 632 并点击**确定**。
3. 点击**制作设计**。

   “设计”分级显示项打开，显示一个 20 次试验的设计。“量度”分级显示项添加到窗口中。

图 20.14 针对电话设计的 “量度”分级显示项

“量度”分级显示项指示强度为 2 的覆盖率为 100%。这意味着所有允许的双因子组合都可以在设计中表示出来。设计中还覆盖了所有三因子组合中的 65%。

4. 点击**制表**。

   该设计随即放入设计表。表中提供了一个响应列以及各种脚本。

图 20.15 覆盖阵列设计表

	响应	市场	近距离电话	近距离界面	远距离电话	远距离界面
1	•	Mexico	Res	A	Res	A
2	•	Mexico	Bus	A	Bus	A
3	•	France	Coin	B	Bus	B
4	•	UK	Res	A	Coin	A
5	•	UK	ISDN	B	Bus	B
6	•	France	ISDN	B	Res	A
7	•	Canada	Bus	A	Res	A
8	•	Mexico	ISDN	B	ISDN	B
9	•	Canada	ISDN	B	Coin	B
10	•	UK	Bus	A	ISDN	B
11	•	Canada	Res	A	Bus	B
12	•	USA	ISDN	B	ISDN	B
13	•	UK	Coin	B	Res	B
14	•	Mexico	Coin	B	Coin	B
15	•	USA	Bus	A	Bus	A
16	•	USA	Coin	A	Coin	A
17	•	USA	Res	A	Res	A
18	•	France	Res	A	ISDN	B
19	•	France	Bus	A	Coin	B
20	•	Canada	Coin	B	ISDN	B

JMP PRO 分析实验数据

既然已经有了设计表，现在可以执行实验并将数据记录在设计表的**响应**列中。您的实验结果包含在 Phone Data.jmp 样本数据表中。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Phone Data.jmp。
2. 在 “表”面板中，点击**分析**脚本旁边的绿色小三角。

图 20.16 电话实验数据分析

Summary

Success Runs	17
Failure Runs	3
Missing	0

Failure Analysis Details

2 Factor Interactions

Factors	Failure Levels	Failure Count
近距离界面, 远距离电话	A, Coin	3
远距离电话, 远距离界面	Coin, A	2
市场, 近距离电话	USA, Coin	1
市场, 近距离电话	UK, Res	1
市场, 近距离电话	France, Bus	1
市场, 远距离电话	USA, Coin	1
市场, 远距离电话	UK, Coin	1
市场, 远距离电话	France, Coin	1
市场, 远距离界面	UK, A	1
近距离电话, 近距离界面	Coin, A	1
近距离电话, 远距离电话	Bus, Coin	1
近距离电话, 远距离电话	Res, Coin	1
近距离电话, 远距离界面	Coin, A	1

“汇总”分级显示项指示三个检验失败。

“失败分析详细信息”分级显示项包含一个“双因子交互作用”报表，因为双因子交互作用是可检测失败的最低水平交互作用。

“双因子交互作用”报表显示可能引起三次失败的组合。有可能是**近距离界面**设置为 A 且**远距离电话**设置为 Coin 的这一组合引起了全部三次失败。或者，也有可能是其他两个或三个组合引起了这三次失败。

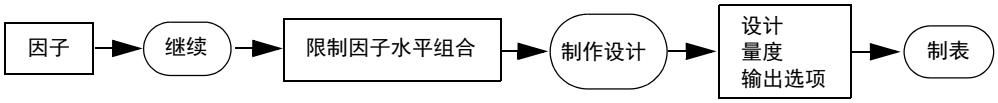
3. 选择“双因子交互作用”报表中的第一行。

在该数据表中，已选中行 4、16 和 19。不论**市场**、**近距离电话**和**远距离界面**如何设置，这些组合都出现了失败。但请注意，因子设置的其他组合也可能引起这些失败。

JMP PRO “覆盖阵列”窗口

“覆盖阵列”窗口将在您逐步执行设计步骤时随之更新。有关实验设计窗口的常规流程的详细信息，请参见第 56 页的“实验设计工作流程：描述、指定、设计”。分级显示项遵循图 20.17 中的流程图，它们被一些按钮分开，点击这些按钮后便有不同的分级显示出现。

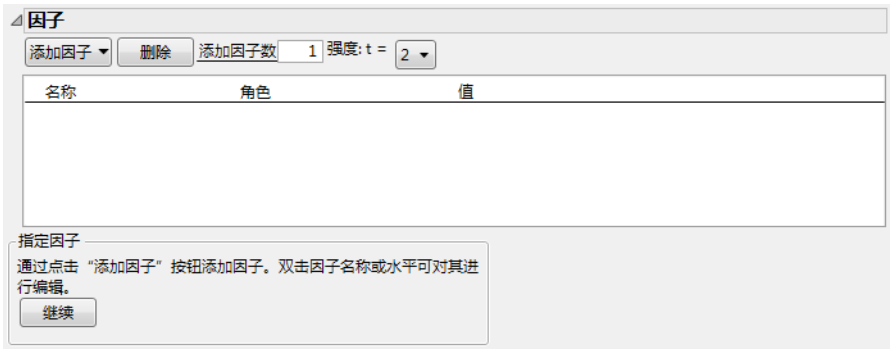
图 20.17 覆盖阵列流程



JMP PRO 因子

在“因子”分级显示项中添加因子。

图 20.18 “因子”分级显示项



添加因子 输入在**添加因子数**中指定的因子数。所有因子都是分类因子。选择或指定水平数。

删除 删除选定的因子。

添加因子数 添加包含特定水平数的多个因子。输入要添加的因子数，点击**添加因子**，然后选择或指定水平数。重复执行**添加因子数**，添加包含不同水平数的多个因子。

强度 t = 选择某个值以指定矩阵的强度。

提示：填写完“因子”面板后，从红色小三角菜单中选择**保存因子**。这将在数据表中保存因子名称和值，以便将来重新加载。请参见第 540 页的“覆盖阵列选项”。

JMP PRO 因子表

“因子”表包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、X2 等。要更改该名称，双击它，然后输入所需名称。

角色 指定因子的设计角色。所有覆盖阵列因子的设计角色均为分类。因子的“设计角色”列属性将保存至数据表。该属性可确保正确地对因子建模。

值 因子的设置。要插入值，点击默认值并输入所需值。设计表中的值排序就是从左到右输入值的顺序。

JMP PRO 编辑因子表

在“因子”分级显示项中，请注意以下事项：

- 要编辑某个因子名称，双击该因子名称。
- 分类因子的因子名称左侧有一个向下箭头。点击该箭头可添加水平。
- 要删除因子水平，点击该值，再点击**删除**，然后在文本框外点击。
- 要编辑某个值，在**值**列中点击该值。

JMP PRO 因子列属性

对于每个因子，“值标签”列属性将保存至设计表。“值标签”列属性代表列中具有指定标签的值。这些标签显示在数据表中，并用在图和报表中。请参见“列属性”附录中第 727 页的“值标签”。

JMP PRO 限制因子水平组合

完成“因子”分级显示项并点击“继续”后，“限制因子水平组合”分级显示项随即显示。该分级显示项支持您指定禁止的因子水平组合。除非您已经加载某个约束或在脚本中包含了某个约束，否则会选定**无**选项。要指定约束，请选择以下其他选项之一：

使用不被允许的组合过滤器 基于各个因子的限定值定义各组约束。您可以定义“与”和“或”约束。请参见第 534 页的“使用不被允许的组合过滤器”。

使用不被允许的组合脚本 定义不被允许的组合和其他约束，作为脚本编辑器框中的布尔 JSL 表达式。请参见第 535 页的 “使用不被允许的组合脚本”。

JMP PRO 使用不被允许的组合过滤器

该选项使用经过改编的数据过滤器帮助指定不被允许的组合。有关使用数据过滤器的详细信息，请参见《使用 JMP》。

要添加不被允许的组合，请执行以下步骤：

1. 从“添加过滤器因子”列表中选择因子并点击**添加**。
2. 通过选择水平来指定不被允许的组合。

注意：“添加过滤器因子”菜单中的红色小三角选项与许多平台启动窗口的“选择列”面板中的选项都相同。请参见《使用 JMP》。

点击“添加”后，将更新初始面板。“不被允许的组合”控制面板显示选定的因子并提供用于进一步控制的选项。

“覆盖阵列”平台仅允许使用分类因子。对于分类因子，可能的水平显示为带标签的区组，或者是在水平数较大时显示为列表条目。选择某个水平使其不被允许。要选择多个水平，请按 Ctrl 键。区组或列表条目将突出显示，以指明不被允许的水平。向“不被允许的组合”面板添加因子时，在因子名称后面的括号中给出分类因子的水平数。

不被允许的组合选项

清除 清除您已指定的所有不被允许的因子水平设置。这不会清除选定的因子。

与 打开“添加过滤器因子”列表。选定的因子将变为一个“与”组。在“与”组内指定的任意因子水平组合都不被允许。

之后要向“与”组添加因子，请点击该组的分级显示项以查看突出显示的矩形。选择“与”并添加因子。

要删除单个因子，请从该因子的红色小三角菜单中选择**删除**。

或 打开“添加过滤器因子”列表。选定的因子将变为单独的“与”组。对于由“或”分隔的“与”组，若至少在一个“与”组中指定了某个组合，则该组合不被允许。

因子的红色小三角选项

某个因子可以显示在多个“或”组中。特定“或”组中因子的一次出现称为该因子的一个**实例**。

删除 从“不被允许的组合”面板中删除选定的因子实例。

清除选择 清除该因子实例的任何选择。

反转选择 取消选择选定值，并选择之前没有为该因子实例选定的那些值。

显示选项 更改显示的外观。

区组显示 将每个水平显示为一个区组。

列表显示 将每个水平显示为列表成员。

单一类别显示 显示各个水平。

复选框显示 在每个值旁边添加一个复选框。

单选框显示 在单个选择项的各值旁边添加一个单选框。

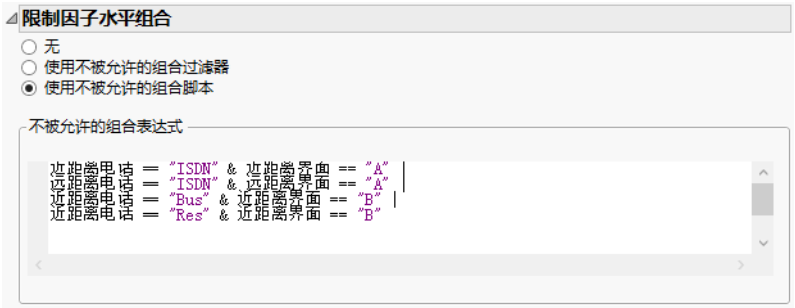
查找 （可用于分类因子。）在因子名称下提供一个文本框，在此可以为因子水平输入搜索字符串。按 Enter 键或在该文本框外点击可执行搜索。一旦选定**查找**，“查找”选项即出现在搜索面板黑色小三角菜单中。选项包含 “包含词条”、“包含短语”、“以短语开头”或 “以短语结尾”以及 “反转结果”。

JMP PRO 使用不被允许的组合脚本

该选项将打开一个脚本窗口，您可以在其中插入用来标识您不想允许的组合的脚本。该脚本必须作为一个布尔表达式来求值。若表达式的计算结果为真，则不允许指定的组合。

创建表达式时，在引号中使用水平名称。请不要使用水平的有序型值。例如，图 20.19 显示了在第 529 页的“使用脚本指定不被允许的组合”的电话界面示例中输入的脚本。

图 20.19 将水平名称显示在引号中的脚本窗口



JMP PRO 设计

点击“制作设计”后，“设计”和“量度”分级显示项随即显示。对于要求大量计算的设计，将显示一个进度栏。

“设计”分级显示项显示您已经构造的设计。第一列列出“试验顺序”。您可能需要使用滚动条查看所有试验。其余列显示每次试验的因子设置。

优化

选择“优化”可减小使用“覆盖阵列”平台构造的设计规模，或减小使用“加载设计”红色小三角选项加载的设计的规模。“优化”不适用于使用“覆盖阵列”平台构造的已知为最优的设计。特别是，通过该平台为两水平因子构造的所有强度为 2 的无约束设计都是最优的。此外，为 *t+1* 个因子构造的所有强度为 *t* 的无约束设计对于任何 *t* 而言都是最优的。

有关算法的详细信息，请参见第 541 页的“用于优化的算法”。

注意：“优化”需要耗费时间，但可以反复运行以生成逐渐改善的设计。

使用**最多迭代次数**选项可指定优化设计时要使用的最多迭代次数。

JMP PRO 无法满足的约束

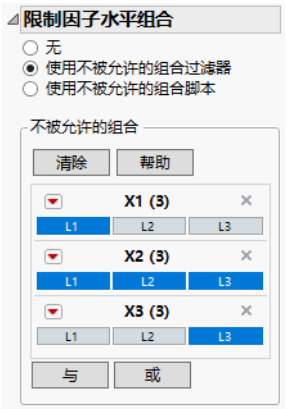
若有一组约束妨碍构造表示出所有必需因子水平的覆盖阵列，则认为这组约束**无法满足**。

无法满足的约束的示例

考虑针对三个因子（每个因子三个水平）的强度为 2 的设计。

1. 选择**实验设计 > 特殊目的 > 覆盖阵列**。
2. 在**添加因子数**旁边键入 3。
3. 从**添加因子**菜单中，选择 **3 水平**。
4. 点击**继续**。
5. 选择**使用不被允许的组合过滤器**。
6. 从**添加过滤器因子**列表中，选择全部三个因子并点击**添加**按钮。
7. 按 Ctrl 键并选择以下水平：
   - 为 X1 选择 L1。
   - 为 X2 选择 L1、L2 和 L3。
   - 为 X3 选择 L3。

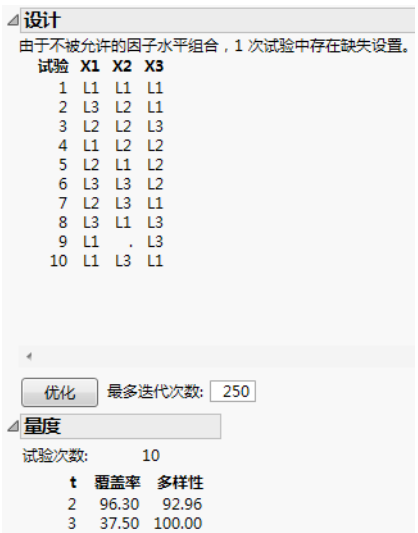
图 20.20 完成的“限制因子水平组合”面板



注意：设置下两步中的“随机种子”将重现本例中显示的确切结果。自行构造设计时，这些步骤不是必需的。

8. 点击“覆盖阵列”红色小三角并选择**设置随机种子**。
9. 输入 12345 并点击**确定**。
10. 点击**制作设计**。

图 20.21 “设计”和“量度”分级显示项



“设计”分级显示项下的注释指出由于存在约束导致一次试验包含缺失设置。该试验是第 9 次试验。要确保覆盖阵列具有强度 2，需要 X1 设置为 L1 且 X3 设置为 L3 的组合。但是对于这些设置，这些约束阻止了 X2 的所有设置。

JMP PRO 量度

“量度”分级显示项为您提供有关设计在多大程度上满足强度要求的信息。请参见 Dalal and Mallows (1998)，了解有关无约束设计的这些量度的背景信息。

t 因子数。

覆盖率 显示在设计中的非重复 *t* 因子设置数量与可能的 *t* 因子设置总数的比率，用百分比表示。*t* 覆盖率为 100% 指示设计覆盖了所有可能的 *t* 因子设置。请注意，每个 *t* 因子设置都可以显示多次。

对于受约束和无法满足的设计，一旦对所有 *t* 因子组合应用了约束，覆盖率的定义将针对可能的 *t* 因子设置数进行调整。请参见第 541 页的 “用于量度的公式”。

多样性 设计中的非重复 *t* 因子设置数量与设计中出现的 *t* 因子设置总数的比率，用百分比表示。*t* 多样性测量设计在多大程度上避免重复。*t* 多样性为 100% 指示未重复任何 *t* 因子设置。50% 的 *t* 多样性指示非重复 *t* 因子设置出现的平均次数为 2。

对于受约束和无法满足的设计，多样性的定义将针对包含缺失设置的试验次数进行调整。请参见第 541 页的 “用于量度的公式”。

JMP PRO 输出选项

制表 构造 “覆盖阵列”数据表。

后退 使您后退到“因子”分级显示项。您可以对之前的分级显示项进行更改并重新生成设计。

注意：若已经在“限制因子水平组合”分级显示项中定义“不被允许的组合”，这些组合将保留为脚本。当您点击“继续”时，该脚本显示在“使用不被允许的组合脚本”面板中。

JMP PRO “覆盖阵列”数据表

“覆盖阵列”数据表包含的第一列是用于输入响应的列。其余列给出了因子设置。

“表”面板 （位于左上角）包含一个设计注释，指示该设计为 “覆盖阵列”并给出了设计的强度。“表”面板还包含以下脚本。

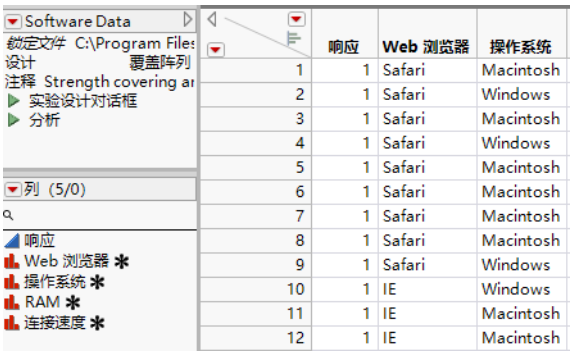
实验设计对话框 重新创建您用于生成设计表的 “覆盖阵列”窗口。

不被允许的组合 显示您在 “限制因子水平组合”分级显示项中输入的因子水平限制。

分析 提供实验数据的分析。有关脚本的详细信息，请参见第 539 页的 ““分析”脚本”。相关背景信息，请参见 Zhang and Zhang (2011)。

提示：要运行脚本，请点击脚本名称旁边的绿色小三角。

图 20.22 显示脚本的 Software Data.jmp 覆盖阵列表的部分视图



	响应	Web 浏览器	操作系统
1	1	Safari	Macintosh
2	1	Safari	Windows
3	1	Safari	Macintosh
4	1	Safari	Windows
5	1	Safari	Macintosh
6	1	Safari	Macintosh
7	1	Safari	Macintosh
8	1	Safari	Macintosh
9	1	Safari	Windows
10	1	IE	Windows
11	1	IE	Macintosh
12	1	IE	Macintosh

JMP PRO “分析”脚本

“分析”脚本假定**响应**列具有以下特点：

- 响应将失败记录为 0，将成功记录为 1。
- 允许缺失值。
- **响应**列是连续的。
- 您可以重命名或移动**响应**列。

“分析”脚本生成包含两个分级显示项的报表：

- “汇总”分级显示项提供导致“成功”、“失败”的试验次数以及响应为“缺失”的试验次数。
- “失败分析详细信息”报表包含一个“<k> 因子交互作用”报表。*k* 的值是检测到失败的最小交互作用数。（要了解关于**检测**的定义，请参见第 521 页的“覆盖阵列和强度”。）这三列包含以下信息：
  - **因子**列列出检测到失败的所有 *k* 因子组合。
  - **失败水平**列列出**因子**列中检测到失败的 *k* 个因子的值。
  - **失败计数**列提供与**失败水平**的 *k* 因子组合对应的失败数。

  注意：一个失败观测可能出现在**失败水平**列中所列的不止一个 *k* 因子组合中。

“<k> 因子交互作用”报表中的行以动态方式链接至数据表。若您选择了报表中的一行或多行，则也会在数据表中选定相应行。

JMP PRO 覆盖阵列选项

“覆盖阵列”平台中的红色小三角菜单包含以下选项：

保存因子 创建一个数据表，其中为每个因子都包含一列，该列包含因子的诸水平。每个因子的列都包含以下列属性：设计角色、值顺序和因子更改。“保存因子”支持您快速将其加载到 “实验设计”窗口中。

注意：您可以为“覆盖阵列”创建因子表，只需在空表中输入数据，但要记住为每列分配适当的“分类”设计角色。右击数据网格中的列名并选择**列属性 > 设计角色**。在“设计角色”区域中，选择 “分类”。

加载因子 将已经使用 “保存因子”选项保存的因子加载到 “因子”分级显示项中。

加载设计 从活跃设计表中加载设计。若没有任何活跃数据表，系统会提示您打开一个数据表。在您选择“加载设计”后，会显示一个菜单，该菜单支持您选择您想要指定为设计中的因子的列。所有列都作为分类列导入。列及其值列在 “因子”分级显示项中。“设计”分级显示项为数据表中的每行显示一次试验，并提供每次试验的因子值。

“加载设计”选项支持您获取现有设计的量度，修改或构造现有设计的 “分析”脚本：

- “量度”分级显示项显示指定设计的“*t* 覆盖率”和“*t* 多样性”。
- 您可以点击**后退**以施加因子水平限制，然后构造新设计。
- 点击**制表**会构造您可以在其中输入响应的设计表。该表包含设计的“分析”脚本。

设置随机种子 设置随机种子， JMP 用来控制具有随机成分的一些操作。对于覆盖阵列，种子会选择开始设计和迭代计数。对于多数设计，随机种子可保证重现设计，但不能重现设计的试验顺序。

注意：时间以及迭代计数的上限用于限制设计构造时间。对于高强度的大型设计，根据所用的计算机，时间限制可能优先于迭代限制。对于此类设计，随机种子不保证再现性。

要重现某个设计，输入用于生成该设计的随机种子，然后点击 “制作设计”。

注意：与设计关联的随机种子包含在保存至设计数据表的“实验设计对话框”脚本中。

高级选项 不适用于 “覆盖阵列”。

将脚本保存至脚本窗口 为您在 “覆盖阵列”窗口中指定的设计创建脚本并将其放到打开的脚本窗口中。

JMP PRO 覆盖阵列的技术细节

- “用于优化的算法”

- “用于量度的公式”

JMP PRO 用于优化的算法

“优化”按钮会调用一种算法，它在概念上类似于一类覆盖阵列优化器，有时称为**构造后随机化优化器** (Nayeri et al., 2013)。不过，JMP 中的算法与多数算法不同，因为它还处理包含约束的设计。尤其是它还优化受约束覆盖阵列以及无法满足的受约束覆盖阵列。

该算法假定要优化的设计是具有指定强度的覆盖阵列。对于强度为 t 的 K 因子设计，该算法以迭代方式检查所有 ${}_K\mathrm{C}_t$ 因子投影，以确定是否可以剔除或合并试验。因此，随着 K 或 t 的增加，算法的运行时间也将快速增加。要改进性能，JMP 采用多线程实现，利用您工作站上可用的尽可能多的 CPU 内核。

JMP PRO 用于量度的公式

计算覆盖率和多样性的公式取决于是否存在约束。将使用以下符号：

${}_u\mathrm{C}_v$ 是从 u 个元素中一次选取 v 个元素的所有组合数

t 是设计的强度

K 是因子数

$M = {}_K\mathrm{C}_t$

$i = 1, 2, \ldots, M$ 是对 t 个因子的所有组合 （或**投影**）进行排序的索引

v_{ik} 是第 k 个因子的水平数

n_i 是设计中针对第 i 个投影的非重复 t 元组数

p_i 是第 i 个投影中的因子的 v_{ik} 乘积

r 是设计中的试验次数

JMP PRO 无约束设计

覆盖率和多样性分别用以下公式计算：

$$\text{覆盖率} = \frac{1}{M}\sum_{i=1}^{M} n_i/p_i$$

$$\text{多样性} = \frac{1}{M}\sum_{i=1}^{M} n_i/r$$

JMP PRO 受约束设计

在受约束设计中，不允许某些特定的 t 元组。这可能导致某些 t 元组缺失值。对于 t 个因子的某些组合，可能根本不存在有效的 t 元组。覆盖率和多样性必须按照可能的有效组合来定义。因此，受约束设计的公式需要使用其他符号：

a_i 是因第 i 个投影中的因子产生的无效 t 元组数

m 是不存在有效 t 元组的投影数

q_i 是设计中包含第 i 个投影中的任何因子的缺失值的试验次数

$r_i = r - q_i$

$M' = M - m$

覆盖率和多样性分别用以下公式计算：

$$\text{覆盖率} = \frac{1}{M'} \sum_{i=1}^{M'} n_i/(p_i - a_i)$$

$$\text{多样性} = \frac{1}{M'} \sum_{i=1}^{M'} n_i/r_i$$

若不存在无效 t 元组 ($M' = M$) 且不存在缺失值（对所有 i 均为 $r_i = r$），则受约束设计的覆盖率和多样性定义可简化为无约束设计的定义。请参见 Morgan (2014)。

第 21 章

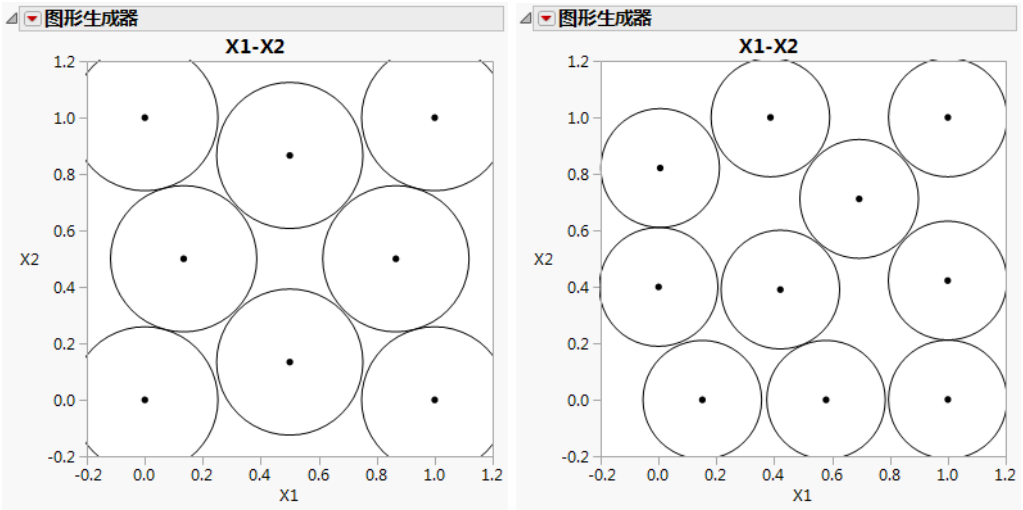
空间填充设计

空间填充设计在试验间变异远小于模型本身的变异的情况下很有用。考虑一个计算机模拟模型的敏感度研究。在这种情况下，对于任何机理或确定性建模问题，任何变异都很小，可以忽略不计。对于没有变异的系统，无需考虑重复、随机化和分区组问题。

“空间填充”平台提供针对连续和分类因子这两种情形的设计。对于连续因子，空间填充设计具有两个目标：

- 使任何两个设计点之间的距离最大化
- 均匀放置设计点

图 21.1 空间填充设计



目录

空间填充设计概述 545
构建空间填充设计 545
响应 546
因子 547
定义因子约束 549
空间填充设计方法 552
设计 552
设计诊断 552
设计表 553
空间填充设计选项 553
球堆积设计 554
创建球堆积设计 555
直观演示球堆积设计 556
拉丁超立方设计 557
创建拉丁超立方设计 558
直观演示拉丁超立方设计 559
均匀设计 561
比较球堆积、拉丁超立方和均匀方法 563
最低位势设计 564
最大熵设计 566
高斯过程 IMSE 最优设计 568
快速灵活填充设计 569
FFF 最优性准则 569
为 FFF 设计设置平均聚类大小 570
约束 570
创建和查看受约束的快速灵活填充设计 571
为地图形状创建空间填充设计 572
球堆积设计的示例 576
为钻孔数据创建球堆积设计 577
确定性数据分析准则 578

空间填充设计概述

空间填充设计对于确定性或近似确定性的建模系统很有用。比如计算机模拟就是确定性系统。这类模拟可能很复杂，涉及很多具有复杂相互关系的变量。对这些系统设计实验的目标是找到更简单的经验模型，该模型足以预测系统在因子的有限范围上的行为。

在针对有很大随机噪声的系统的实验中，目标是使预测的方差最小化。在针对确定性系统的实验中，没有方差但有**偏倚**。偏倚是近似模型和真实数学函数之间的差值。空间填充设计的目标是限制偏倚。

一种限制偏倚的方法是让设计点的分布尽量远离彼此，同时它们要保持在实验边界内。另一种方法是使点均匀分布在相关区域。

空间填充设计器支持以下设计方法：

球堆积 使每对设计点之间的最小距离最大化。请参见第 554 页的 “球堆积设计”和第 577 页的 “为钻孔数据创建球堆积设计”。

拉丁超立方 使每对设计点之间的最小距离最大化，但是要求每个因子的水平是等间距的。这个方法将生成模拟均匀分布的设计。拉丁超立方方法是球堆积方法和均匀设计方法的折衷。请参见第 557 页的 “拉丁超立方设计”。

均匀 使设计点 （具有经验均匀分布）和理论均匀分布之间的偏差最小化。请参见第 561 页的 “均匀设计”。

最低位势 将点围绕着中心分布在球内。请参见第 564 页的 “最低位势设计”。

最大熵 测量在一组数据的分布中包含的信息量。请参见第 566 页的 “最大熵设计”。

高斯过程 IMSE 最优 创建一个设计，它使在实验区域内高斯过程的积分均方误差最小。请参见第 568 页的 “高斯过程 IMSE 最优设计”。

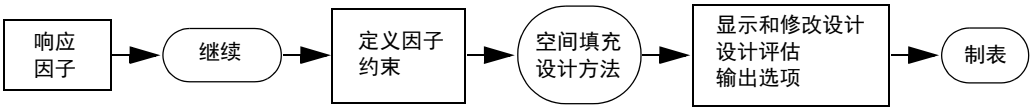
快速灵活填充 快速灵活填充方法在设计空间中形成来自随机点的聚类。这些聚类用于根据优化准则选择设计点。这是唯一可在设计空间上容纳分类因子和约束的方法。您可以指定线性约束和不被允许的组合。请参见第 569 页的 “快速灵活填充设计”和第 571 页的 “创建和查看受约束的快速灵活填充设计”。

注意：若试验次数为 500 或更少，则将高斯过程模型保存到数据表。若试验次数超过 500，则将神经模型保存到数据表。

构建空间填充设计

通过选择**实验设计** > **特殊目的** > **空间填充设计**构建空间填充设计。首先定义响应和因子。接下来，继续执行设计选项、生成和评估过程。设计过程遵循图 21.2 中的流程。

图 21.2 空间填充设计流程图

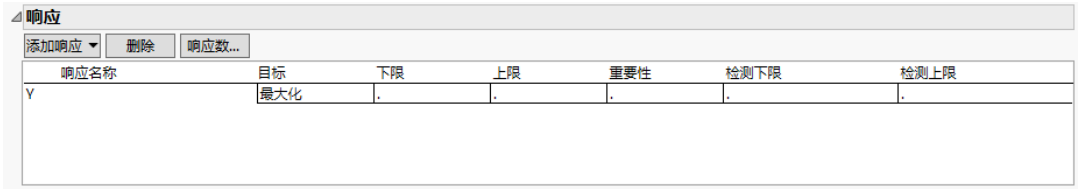


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 21.3 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO **函数** （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见 《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的 “响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。

– “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
– “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
– “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定“广义回归”平台中的删失响应。请参见《拟合线性模型》中的“广义回归”一章。

编辑“响应”分级显示项

在“响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

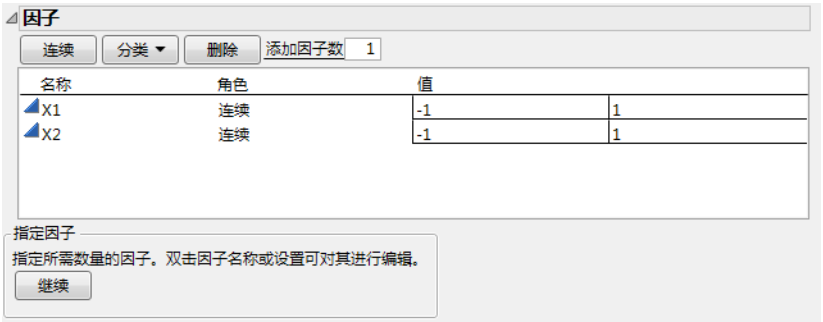
在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

在“因子”分级显示项中为空间填充设计添加因子。

图 21.4 “因子”分级显示项



“因子”分级显示项包含以下选项：

连续型 输入在**添加因子数**中指定的连续因子数。

分类 输入在**添加因子数**中指定的名义型因子数。

删除 删除选定的因子。

添加因子数 添加给定类型的多个因子。输入要添加的因子数并点击“连续”或“分类”。重复执行**添加因子数**，添加不同类型的多个因子。

提示：填写完“因子”面板后，从红色小三角菜单中选择**保存因子**。这将在数据表中保存因子名称和值，以便将来重新加载。请参见第 553 页的“空间填充设计选项”。

“因子”分级显示项

“因子”分级显示项包含以下列：

名称 因子的名称。在添加因子时，会给因子指定默认名称 X1、X2 等。要更改该名称，双击它，然后输入所需名称。

角色 指定因子的设计角色。因子的“设计角色”列属性将保存至数据表。该属性可确保正确地对因子类型建模。

值 因子的实验设置。要插入值，点击默认值并输入所需值。

编辑“因子”分级显示项

在“因子”分级显示项中，请注意以下事项：

- 要编辑某个因子名称，双击该因子名称。
- 分类因子的因子名称左侧有一个向下箭头。点击该箭头可添加水平。
- 要删除因子水平，点击该值，再点击**删除**，然后在文本框外点击。
- 要编辑某个值，在**值**列中点击该值。

因子类型

连续 仅限数值数据类型。在符合过程和测量系统的限制条件下，若一个因子在概念上可以设置为您提供的下限与上限之间的任何值，则这样的因子为连续因子。

分类 数值或字符数据类型。对于分类因子，值排序是指从左到右输入值时值的顺序。在创建设计数据表后，该排序将保存在 “值顺序” 列属性中。

因子列属性

对于每个因子，各个列属性保存在数据表中。

设计角色 每个因子都分配有 “设计角色” 列属性。在定义因子时指定的角色确定其 “设计角色” 列属性的值。“设计角色” 属性反映在对实验数据建模时要如何使用因子。“设计角色” 值用在 “扩充设计” 平台中。

因子更改 每个因子都分配有 “因子更改” 列属性，该属性设置为 “容易”。在空间填充设计中，假定对于每次试验都可以更改因子水平。“因子更改” 值用在 “评估设计” 和 “扩充设计” 平台中。

编码 若 “角色” 为 “连续”，将保存因子的 “编码” 列属性。该属性变换因子值，以便低值与高值分别对应于 –1 和 +1。“拟合最小二乘法” 报表中的估计值和检验基于变换值。

值顺序 若 “角色” 为 “分类” 或 “分区组”，将保存因子的 “值顺序” 列属性。该属性确定因子水平的显示顺序。

定义因子约束

只能为使用 “快速灵活填充” 方法构造的空间填充设计指定因子约束。

使用 “定义因子约束” 来限制设计空间。除非您已经加载某个约束或在脚本中包含了某个约束，否则会选定**无**选项。要指定约束，请选择以下其他选项之一：

指定线性约束 指定针对因子的线性组合的不等式约束。仅适用于具有 “连续” 或 “混料” 角色的因子。请参见 “指定线性约束”。

注意： 保存涉及线性约束的设计的脚本时，该脚本将线性约束表示为**小于等于**不等式 (≤)。

使用不被允许的组合过滤器 基于各个因子的限定值定义各组约束。您可以定义 “与” 和 “或” 约束。请参见 “使用不被允许的组合过滤器”。

使用不被允许的组合脚本 定义不被允许的组合和其他约束，作为脚本编辑器框中的布尔 JSL 表达式。请参见 “使用不被允许的组合脚本”。

注意： 当分析具有因子约束的设计时，模型刻画器将接受这些约束。

指定线性约束

当连续因子不可能在设计空间上独立变化时，可以指定线性不等式约束。线性不等式描述了允许的因子水平设置。

点击**添加**以输入一个或多个线性不等式约束。

添加 添加涉及设计中所有连续因子的线性表达式的模板。输入因子的系数值并选择不等式的方向来反映您的线性约束。在不等式右侧的框中指定约束值。要添加更多约束，请再次点击**添加**。

注意：若您已通过指定球形半径限制了设计区域，将禁用 “添加”选项。

删除最后一个约束 删除最后一个约束。

检查约束 检查约束的一致性。该选项删除多余的约束并执行可行性检查。若有问题将显示 JMP 警示。若约束等效于因子的界限， JMP 警示将指示 “因子”分级显示项中的界限已更新。

使用不被允许的组合过滤器

该选项使用经过改编的数据过滤器帮助指定不被允许的组合。有关详细信息，请参见《使用 JMP》中的“JMP 报表”一章。

从“添加过滤器因子”列表中选择因子并点击**添加**。然后通过使用滑块（对于连续因子）或选择水平（对于分类因子）来指定不被允许的组合。

“添加过滤器因子”菜单的红色小三角选项在许多平台启动窗口的“选择列”面板中都可以找到。请参见《使用 JMP》中的“入门”一章。

点击“添加”时，“不被允许的组合”控制面板显示选定的因子并提供用于进一步控制的选项。基于建模类型，按以下方式表示因子：

连续因子 对于连续因子，将显示跨越因子设置范围的双端滑块。您可以指定不允许的设置，方法是拖动滑块两端或通过点击滑块下的文本值来设置端点。在滑块中，用纯蓝色突出表示不被允许的值。

分类因子 对于分类因子，可能的水平显示为带标签的区组，或者是在水平数较大时显示为列表条目。选择某个水平使其不被允许。要选择多个水平，请按住 Ctrl 键。区组或列表条目将突出显示，以指明不被允许的水平。向“不被允许的组合”面板添加分类因子时，在因子名称后面的括号中给出分类因子的水平数。

不被允许的组合选项

控制面板具有以下控件：

清除 清除您已指定的所有不被允许的因子水平设置。这不会清除选定的因子。

重新开始 删除所有选定因子，并返回最初的因子列表。

与 打开“添加过滤器因子”列表。选定的因子将变为一个“与”组。在“与”组内指定的任意因子水平组合都不被允许。

之后要向“与”组添加因子，请点击该组的分级显示项以查看突出显示的矩形。选择“与”并添加因子。

要删除单个因子，请从该因子的红色小三角菜单中选择**删除**。

或 打开“添加过滤器因子”列表。选定的因子将变为单独的“与”组。对于由“或”分隔的“与”组，若至少在一个“与”组中指定了某个组合，则该组合不被允许。

因子的红色小三角选项

某个因子可以显示在多个“或”组中。特定“或”组中因子的一次出现称为该因子的一个**实例**。

删除 从“不被允许的组合”面板中删除选定的因子实例。

清除选择 清除该因子实例的任何选择。

反转选择 取消选择选定值，并选择之前没有为该因子实例选定的那些值。

显示选项 仅可用于分类因子。更改显示的外观。选项包括：将每个水平显示为区组、列表或单个类别，或者在每个值旁边添加复选框。

查找 仅可用于分类因子。在因子名称下提供一个文本框，在此可以为因子水平输入搜索字符串。按 Enter 键或在该文本框外点击可执行搜索。一旦选定**查找**，“查找”选项即显示在红色小三角菜单中，如：“不包含”、“区分大小写”、“开始字符为”或“结束字符为”，或者“清除查找”。

使用不被允许的组合脚本

使用该选项来通过 JSL 脚本指定不被允许的因子水平的特定组合。该选项可以用于连续因子或既有连续因子又有分类因子的情况。

该选项将打开一个脚本窗口，您可以在其中插入用来标识您不想允许的组合的脚本。该脚本必须作为一个布尔表达式来求值。若表达式的计算结果为真，则不允许指定的组合。

创建分类因子的表达式时，使用水平的序数值而非水平的名称。若因子的水平是高、中和低（在“因子”分级显示项中以该顺序指定），则相关的序数值为 1、2 和 3。例如，假定您有两个连续因子 X1 和 X2，还有一个具有三个水平（水平的顺序为 L1、L2 和 L3）的分类因子 X3。您希望不允许符合以下条件的水平：

$$e^{X_1} + 2X_2 < 0 \text{ 并且 } X_3 = L2$$

在脚本窗口中输入表达式 `(Exp(X1) + 2*X2 < 0) & (X3 == 2)`。

图 21.5 脚本编辑器中的表达式

（在图中，通过解析删除了不必要的括号。）请注意函数可以作为布尔表达式的一部分来输入。表达式 (Exp(X1) + 2*X2 < 0) & (X3 == “L2”) 也有效。

空间填充设计方法

可以使用以下方法来构造空间填充设计：

- 第 554 页的“球堆积设计”
- 第 557 页的“拉丁超立方设计”
- 第 561 页的“均匀设计”
- 第 564 页的“最低位势设计”
- 第 566 页的“最大熵设计”
- 第 568 页的“高斯过程 IMSE 最优设计”
- 第 569 页的“快速灵活填充设计”

设计

“设计”分级显示项显示空间填充设计的试验。

设计诊断

“设计诊断”分级显示项显示尺度为 0-1 的空间填充设计因子的值。最小距离基于这些统一尺度的值，是每个点到其最近邻的最小距离。最近邻的行号在“最近的点”列中给出。表下面显示的偏差值为基于设计点的分布和均匀分布之间的积分差值。

将为完整设计和分类变量（若使用）的每个水平都提供 MaxPro（最大投影）准则。值越小越好。有关 MaxPro 准则的详细信息，请参见第 569 页的“MaxPro”。

注意：若对于任何维，两个点具有相同值（即 $x_{ik} = x_{jk}$），则未定义 MaxPro 准则。在这种情况下，设计诊断为 MaxPro 显示缺失值。

设计表

使用“设计表”按钮可完成“空间填充设计”构造。

制表 构造 “空间填充设计”数据表。

后退 使您返回在点击**制作设计**前所在的位置。您可以对之前的分级显示项进行更改并重新生成设计。

空间填充设计选项

“空间填充设计”平台中的红色小三角菜单包含以下选项：

保存响应 将 “响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用 “保存响应”选项保存的响应。

保存因子 将 “因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在 “设计角色”区域中，选择合适的角色。

加载因子 加载使用 “保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在 “定义因子约束”或 “线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为 “小于”或 “大于”约束。请参见 “列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用 “保存约束”选项保存的因子约束。

设置随机种子 设置随机种子， JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的“试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击“制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示“模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一：“正态”、“二项式”或“Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在“模型”窗口中做出选择以控制模拟的响应值的分布。当您点击“应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中，Y 是响应的名称。再次点击“应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见“定制设计”一章中第 101 页的“模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见《基本分析》中的“模拟”一章。

FFF 最优性准则 对于快速灵活填充设计方法，使您可以选择 MaxPro 准则（默认值）或重心法准则。请参见第 569 页的“FFF 最优性准则”。

开始数 指定算法针对所选设计类型开始构造新设计的次数。基于给定设计类型的准则，返回最佳设计。默认情况下对于所有设计类型设置为 1。不用于快速灵活填充设计。

高级选项 > 设置平均聚类大小 对于快速灵活填充设计方法，使您可以指定用于定义每个聚类（或等效的每个设计点）的随机生成的平均点数。

高级选项 > MaxPro 分类权重 对于“快速灵活填充”设计方法，支持您指定用于 MaxPro 计算中的分类因子的权重。具有相同分类水平的点与该权重相乘。这将为分类因子的每个水平在子设计一级提供更好的空间填充属性。

将脚本保存至脚本窗口 为您在“空间填充设计”平台中指定的设计创建脚本，并将其保存到打开的脚本窗口中。

球堆积设计

球堆积设计方法使每对设计点之间的最小距离最大化。该最大化的效果是在设计区域内将设计点尽可能地分布开来。

- “创建球堆积设计”

- “直观演示球堆积设计”

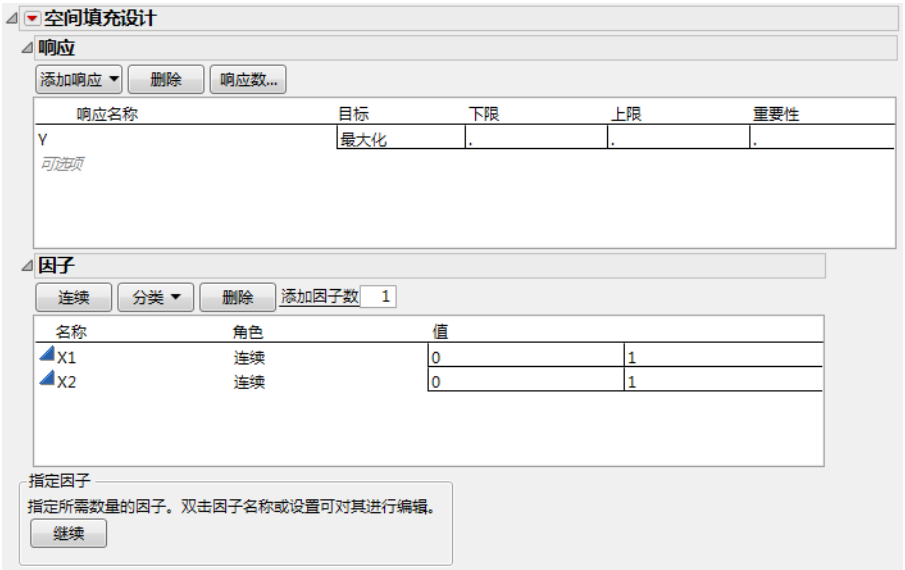
创建球堆积设计

要创建球堆积设计，请按以下步骤执行：

1. 选择**实验设计 > 特殊目的 > 空间填充设计**。
2. 输入响应和因子。

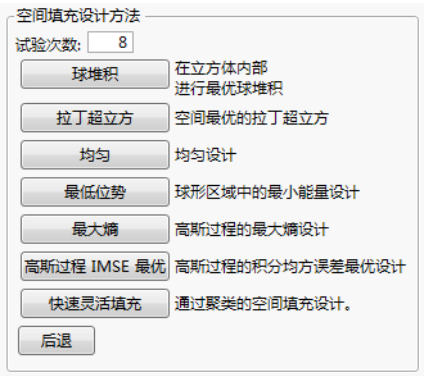
   请参见第 546 页的 “响应”。
3. 按需更改因子水平值。例如，图 21.6 显示两个现有因子 X1 和 X2 以及它们的值 （范围为 0-1，而非默认的 –1 到 1）。

图 21.6 针对两个因子的空间填充对话框



4. 点击**继续**。
5. 在设计规格对话框中，指定样本大小 （**试验次数**）。图 21.7 显示样本大小为 8。

图 21.7 空间填充设计对话框



6. 点击**球堆积**。

   JMP 创建设计并显示设计试验和设计诊断。图 21.8 显示 “设计诊断”面板，其中 0.518 为**最小距离**。您的结果可能与下面的结果略有不同，但是最小距离是相同的。没有为球堆积定义 MaxPro 统计量，因为点在一个维中可能有相同值。

图 21.8 球堆积设计诊断

设计诊断

试验	统一尺度X1	统一尺度X2	最小距离	最近的点
1	0.50000	0.86602	0.518	3
2	0.86603	0.50000	0.518	6
3	0.13397	0.50000	0.518	1
4	0.50000	0.13398	0.518	7
5	0.00000	1.00000	0.518	1
6	1.00000	1.00000	0.518	2
7	1.00000	0.00000	0.518	4
8	0.00000	0.00000	0.518	3

偏离度 = 0.046969
MaxPro
完整设计 .

设计表
制表 后退

7. 点击**制表**。使用该表完成可视化示例 （将在下文中介绍）。

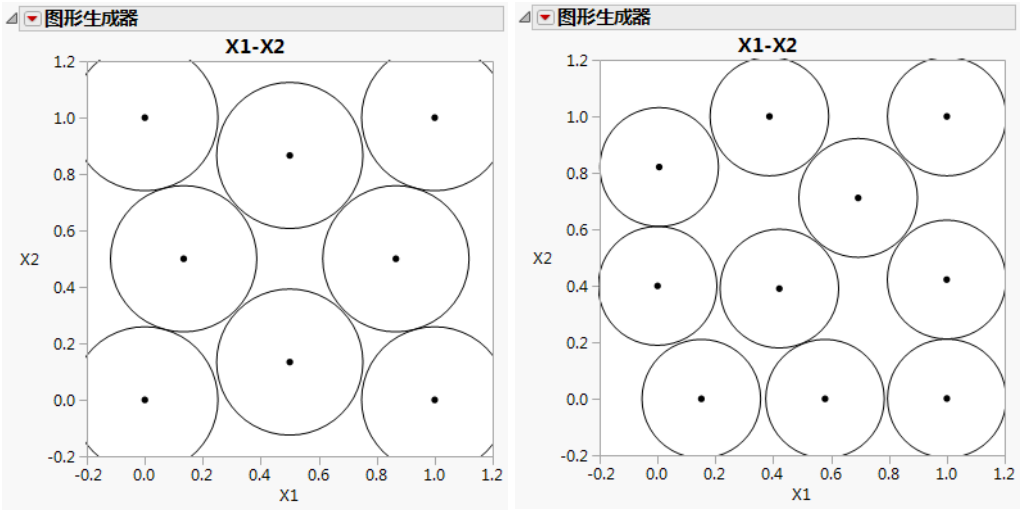
直观演示球堆积设计

要直观演示球堆积设计，请使用球堆积设计表通过 “图形生成器”来创建图：

1. 选择**图形 > 图形生成器**。
2. 指定 X1 作为 X，指定 X2 作为 Y。
3. **点击**平滑线图标可从图中删除平滑线 （蓝线）。
4. 通过右击该图并选择**图形 > 大小/尺度 > 调整为相同大小**调整框架大小，使得框架为正方形。

5. 右击该图并选择**定制**。显示“定制”面板时，点击加号以显示一个文本编辑区域，然后输入以下脚本：
   `For Each Row(Circle({:X1, :X2}, 0.518/2))`
   其中 `0.518` 是您在“设计诊断”面板中记下的最小距离值。该脚本绘制一个以每个设计点为圆心的圆，其半径为 0.259（直径 0.518 的一半），如图 21.9 中的左侧所示。该图显示 JMP 堆积设计点的有效方式。
6. 现在完全按照上一节中的所述重复过程，但是样本大小为 10 而非 8。
   切记将图形脚本中的 `0.518` 更改为 10 次试验所得到的最小距离。显示该图时，再次设置框架大小并使用诊断报告中的最小距离作为圆的直径创建图形脚本。您应该看到与图 21.9 中右侧图形相似的图形。请注意球堆积的不规则特性。实际上，您可以第三次重复该过程得到一个略有不同的图，因为排列取决于随机开始点。

图 21.9 具有 8 次试验 （左侧）和 10 次试验 （右侧）的球堆积示例



拉丁超立方设计

在拉丁超立方中，每个因子具有与设计中试验次数相同的水平数。这些水平在因子的下限和上限之间等间距分布。与球堆积方法类似，拉丁超立方方法选择点使设计点之间的最小距离最大化，但是具有约束。约束可保持因子水平之间等间距。

- “创建拉丁超立方设计”
- “直观演示拉丁超立方设计”

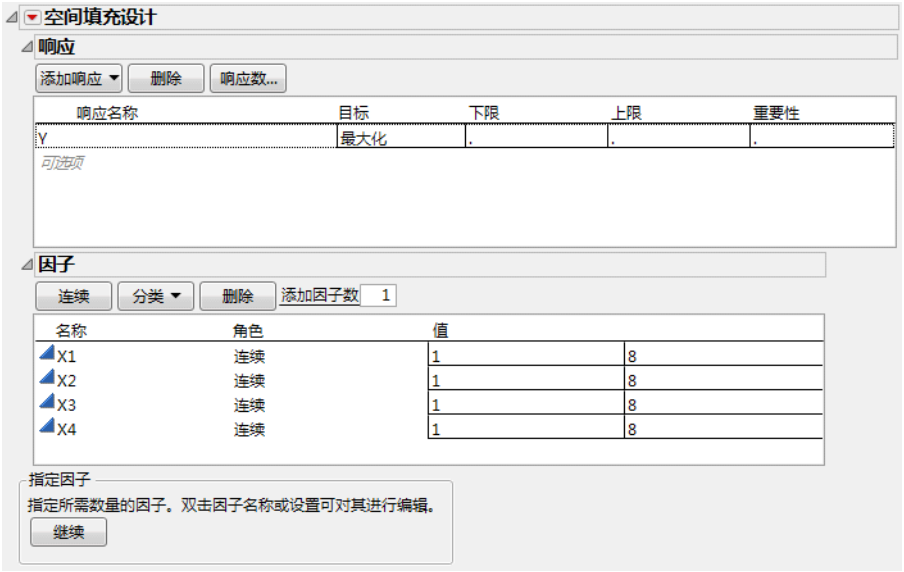
创建拉丁超立方设计

要创建拉丁超立方设计，请按以下步骤执行：

1. 选择**实验设计** > **特殊目的** > **空间填充设计**。
2. 按需输入响应和因子。

   请参见第 546 页的 “响应”。
3. 按需更改因子水平值。图 21.10 显示添加四个因子并将其值更改为 1 和 8 以替代默认值 –1 和 1。

图 21.10 4 个因子的空间填充对话框



4. 点击**继续**。
5. 在设计规格对话框中，指定样本大小 （**试验次数**）。该示例使用的样本大小为 8。
6. 点击**拉丁超立方** （图 21.7）。显示的因子设置和设计诊断结果类似于图 21.11 中的内容，该图显示具有 4 个因子和 8 次试验的拉丁超立方设计。

注意：该示例的目的是显示只给每列（因子）分配每个水平一次且每列都是水平的不同置换。

图 21.11 针对具有 8 个水平的 4 个因子和 8 次试验的拉丁超立方设计

4 因子
空间填充 - 拉丁超立方

设计

试验	X1	X2	X3	X4
1	4.00000	7.00000	1.00000	4.00000
2	2.00000	6.00000	5.00000	7.00000
3	1.00000	4.00000	6.00000	1.00000
4	7.00000	5.00000	3.00000	8.00000
5	5.00000	2.00000	8.00000	6.00000
6	6.00000	8.00000	7.00000	3.00000
7	8.00000	3.00000	4.00000	2.00000
8	3.00000	1.00000	2.00000	5.00000

设计诊断

试验	统一尺度X1	统一尺度X2	统一尺度X3	统一尺度X4	最小距离	最近的点
1	0.42857	0.85714	0.00000	0.42857	0.782	2
2	0.14286	0.71429	0.57143	0.85714	0.782	1
3	0.00000	0.42857	0.71429	0.00000	0.926	2
4	0.85714	0.57143	0.28571	1.00000	0.795	2
5	0.57143	0.14286	1.00000	0.71429	0.845	2
6	0.71429	1.00000	0.85714	0.28571	0.892	7
7	1.00000	0.28571	0.42857	0.14286	0.892	6
8	0.28571	0.00000	0.14286	0.57143	0.892	1

偏离度 = 0.039317

	MaxPro
完整设计	11.40992

直观演示拉丁超立方设计

为了直观演示拉丁超立方技术的特性，使用 “图形生成器”创建一个图。

1. 创建另一个使用两个因子的拉丁超立方设计。
2. 选择**实验设计 > 特殊目的 > 空间填充设计**。
3. 输入两个因子，并确保更改因子值以使它们为 0 和 1 而非默认值 –1 和 1。
4. 点击**继续**。
5. 指定样本大小为 8 （**试验次数**）。
6. 点击**拉丁超立方**。

图 21.12 具有 2 个因子和 8 次试验的拉丁超立方设计

2 因子
空间填充 - 拉丁超立方

设计

试验	X1	X2
1	0.28571	0.00000
2	0.00000	0.85714
3	0.57143	0.28571
4	0.42857	0.71429
5	0.14286	0.42857
6	1.00000	0.14286
7	0.85714	0.57143
8	0.71429	1.00000

设计诊断

试验	统一尺度X1	统一尺度X2	最小距离	最近的点
1	0.28571	0.00000	0.404	3
2	0.00000	0.85714	0.452	4
3	0.57143	0.28571	0.404	1
4	0.42857	0.71429	0.404	5
5	0.14286	0.42857	0.404	4
6	1.00000	0.14286	0.452	7
7	0.85714	0.57143	0.404	3
8	0.71429	1.00000	0.404	4

偏离度 = 0.009279

	MaxPro
完整设计	11.32801

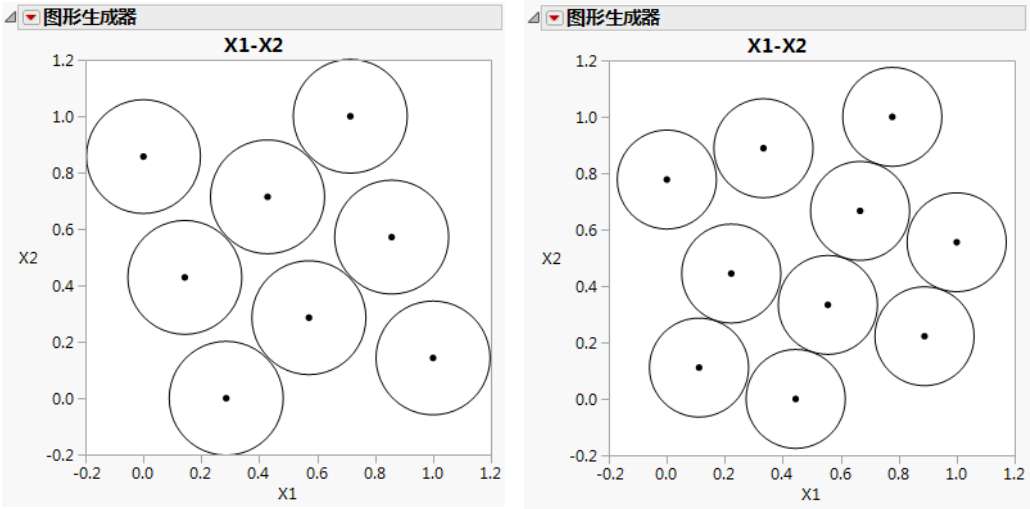
7. 点击**制表**。
8. 选择**图形 > 图形生成器**。
9. 指定 X1 作为 X，指定 X2 作为 Y。
10. 点击平滑线图标可从图中删除平滑线 （蓝线）。
11. 右击该图并选择**图形 > 大小 / 尺度 > 调整为相同大小**调整框架大小，使得框架为正方形。
12. 右击该图，从菜单中选择**定制**。在“定制”面板中，点击大加号以显示一个文本编辑区域，然后输入以下脚本：

    ```
    For Each Row(Circle({:X1, :X2}, 0.404/2))
    ```

    其中 0.404 是您在“设计诊断”面板中记录的最小距离值 （图 21.12）。该脚本绘制一个以每个设计点为圆心的圆，其半径为 0.202 （直径 0.404 的一半），如图 21.13 中的左侧所示。该图显示 JMP 堆积设计点的有效方式。
13. 完全按照上一节中的所述重复过程，但是试验次数为 10 而非 8（步骤 5）。切记将图形脚本中的 0.404 更改为 10 次试验所得到的最小距离。

您应该看到与图 21.13 中右侧图形相似的图形。请注意球堆积的不规则特性。实际上，您可以重复该过程得到一个略有不同的图，因为排列取决于随机开始点。

图 21.13 具有 8 次试验 （左侧）和 10 次试验 （右侧）的拉丁超立方设计的比较



请注意，拉丁超立方设计中每对点之间的最小距离比球堆积设计中每对点之间的最小距离小。这是因为拉丁超立方设计有约束条件，要求每个因子的水平等间距。球堆积设计在没有任何约束的情况下使最小距离最大化。

均匀设计

均匀设计使设计点（经验均匀分布）和理论均匀分布之间的偏差最小化。

注意：这些设计对于获得未知函数积分的简单准确的估计值最有用。估计值是通过实验得到的观测响应的平均值。

1. 选择**实验设计 > 特殊目的 > 空间填充设计**。
2. 按需输入响应和因子。

   请参见第 546 页的 “响应”。
3. 将因子水平值更改为 0 和 1。
4. 点击**继续**。
5. 在设计规格对话框中，指定样本大小。该示例使用的样本大小为 8 （**试验次数**）。
6. 点击**均匀**按钮。 JMP 创建该设计并显示设计试验和设计诊断 （如图 21.14 中所示）。

注意：均匀设计方法的重点不在于分散这些点。图 21.14 中的最小距离变化很大。

图 21.14 具有 8 次试验的均匀空间填充设计的因子设置和诊断

2 因子
空间填充 - 均匀设计

设计

试验	X1	X2
1	0.93092	0.69041
2	0.69276	0.18514
3	0.06860	0.32218
4	0.18696	0.81304
5	0.56496	0.93540
6	0.81410	0.43443
7	0.32218	0.06860
8	0.43643	0.56357

设计诊断

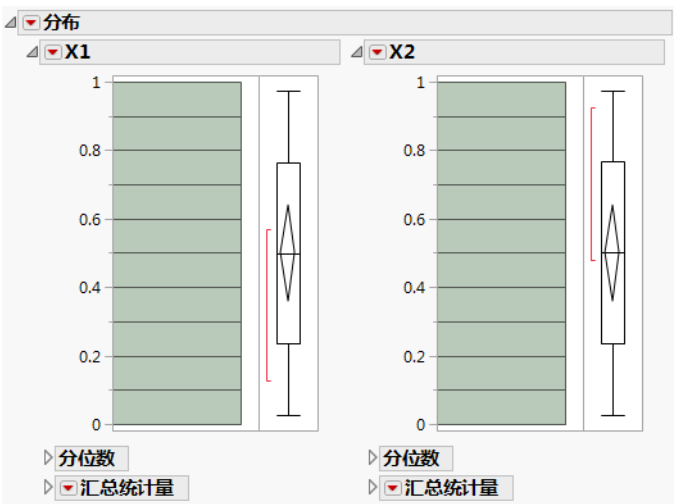
试验	统一尺度X1	统一尺度X2	最小距离	最近的点
1	0.93092	0.69041	0.281	6
2	0.69276	0.18514	0.277	6
3	0.06860	0.32218	0.359	7
4	0.18696	0.81304	0.353	8
5	0.56496	0.93540	0.393	8
6	0.81410	0.43443	0.277	2
7	0.32218	0.06860	0.359	3
8	0.43643	0.56357	0.353	4

偏离度 = 0.004663
MaxPro
完整设计 15.40271

7. 点击**制表**。

均匀设计不保证因子水平的等间距。但是，增加试验次数并对每个因子运行分布（使用**分析** > **分布**）会显示平坦的直方图。

图 21.15 当试验次数增加到 20 时每个因子对应的直方图是平坦的



比较球堆积、拉丁超立方和均匀方法

要比较空间填充设计方法，需要创建**球堆积**、**拉丁超立方**和**均匀**设计，如以上示例中所示。“设计诊断”表显示统一尺度为 0 到 1 的因子值。最小距离基于这些统一尺度的值，是每个点到最近邻的最小距离。偏差值为设计点和均匀分布之间的积分差值。

图 21.16 显示三种试验次数为 8 的空间填充设计的设计诊断比较。请注意均匀设计的偏差最小（最佳）。球堆积设计的偏差最大（最差）。拉丁超立方的偏差介于这两者之间，更接近最优值。与均匀设计相比，MaxPro 准则对于拉丁超立方设计较小。

还请注意，球堆积方法的每对点之间的最小距离最大（最佳）。均匀设计的每对点之间的最小距离大约只有前者的一半。拉丁超立方设计在点的分散程度上更接近于球堆积设计。

对于分散度和偏差，拉丁超立方设计代表了一个合理的折衷方案。

图 21.16 试验次数为 8 的三种空间填充方法的诊断比较

空间填充 - 球堆积

▷设计

◢设计诊断

试验	统一尺度X1	统一尺度X2	最小距离	最近的点
1	0.50000	0.86602	0.518	3
2	0.86603	0.50000	0.518	6
3	0.13397	0.50000	0.518	1
4	0.50000	0.13398	0.518	7
5	0.00000	1.00000	0.518	1
6	1.00000	1.00000	0.518	2
7	1.00000	0.00000	0.518	4
8	0.00000	0.00000	0.518	3

偏离度 = 0.046969

MaxPro

完整设计 .

空间填充 - 拉丁超立方

▷设计

◢设计诊断

试验	统一尺度X1	统一尺度X2	最小距离	最近的点
1	0.28571	0.00000	0.404	3
2	0.00000	0.85714	0.452	4
3	0.57143	0.28571	0.404	1
4	0.42857	0.71429	0.404	5
5	0.14286	0.42857	0.404	4
6	1.00000	0.14286	0.452	7
7	0.85714	0.57143	0.404	3
8	0.71429	1.00000	0.404	4

偏离度 = 0.009279

MaxPro

完整设计 11.32801

空间填充 - 均匀设计

▷设计

◢设计诊断

试验	统一尺度X1	统一尺度X2	最小距离	最近的点
1	0.93092	0.69041	0.281	6
2	0.69276	0.18514	0.277	6
3	0.06860	0.32218	0.359	7
4	0.18696	0.81304	0.353	8
5	0.56496	0.93540	0.393	8
6	0.81410	0.43443	0.277	2
7	0.32218	0.06860	0.359	3
8	0.43643	0.56357	0.353	4

偏离度 = 0.004663

MaxPro

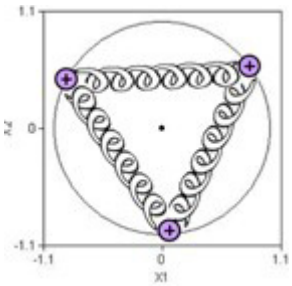
完整设计 15.40271

比较的另一个方面是计算设计所需的时间。均匀设计方法计算花费的时间最长。此外，计算设计所需的时间随着试验次数的增加而快速增加。对于类似的问题，所有空间填充设计方法所花的计算时间均比定制设计器中的 *D* 最优设计所花的计算时间长。

最低位势设计

最低位势设计在球内分散点。要了解这个设计是如何创建的，请将点设想为电子，它具有连接其他点的弹簧，如图 21.17 中所示。库伦力将这些点推离开，但是弹簧则将它们拉在一起。该设计得到使系统的势能最小化的点间距离。

图 21.17 最低位势设计



最低位势设计：

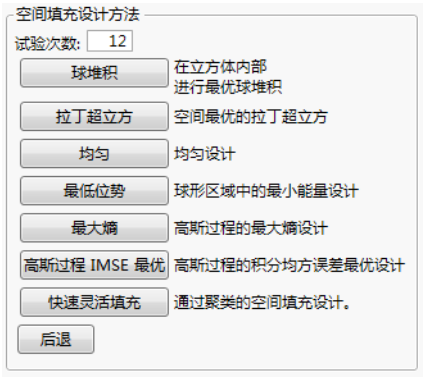
- 具有球对称性
- 接近正交设计
- 具有均匀间距

要构造 “最低位势”示例设计：

1. 选择**实验设计 > 特殊目的 > 空间填充设计**。
2. 添加 3 个连续因子。

   请参见第 547 页的 “因子”。
3. 将因子水平值更改为 0 和 1。
4. 点击**继续**。
5. 在设计规格对话框中 （显示在图 21.18 中的左侧），输入样本大小 （**试验次数**）。该示例使用的样本大小为 12。
6. 点击**最低位势**按钮。JMP 创建该设计并显示设计试验和设计诊断（显示在图 21.18 中的右侧）。

图 21.18 空间填充方法和最低位势设计的设计诊断



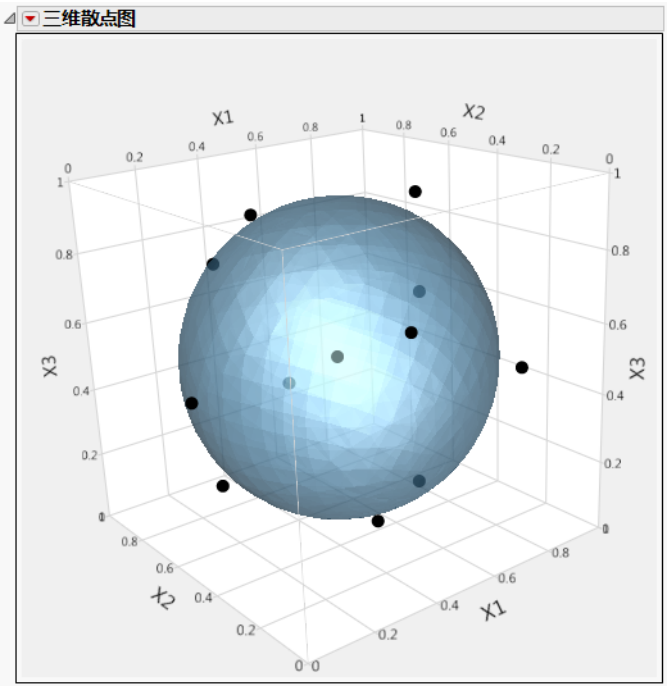
7. 点击**制表**。

使用 “三维散点图”图形平台，您可以看到最低位势设计的球对称性。

1. 在创建 JMP 设计表后，选择**图形 > 三维散点图**命令。
2. 在 “三维散点图”启动对话框中，选择 X1、X2 和 X3 作为 **Y, 列**，然后点击**确定**以查看设计点的初始三维散点图。
3. 要得到与图 21.19 中类似的结果，请执行以下步骤：
   - 点击 “三维散点图”红色小三角并选择**正态等高线椭圆**。
   - 右击该图并选择**设置**，然后使用**标记大小**滑块增加标记大小。

现在可以看到点均匀分布在椭圆表面上。

图 21.19 球上的最低位势设计点

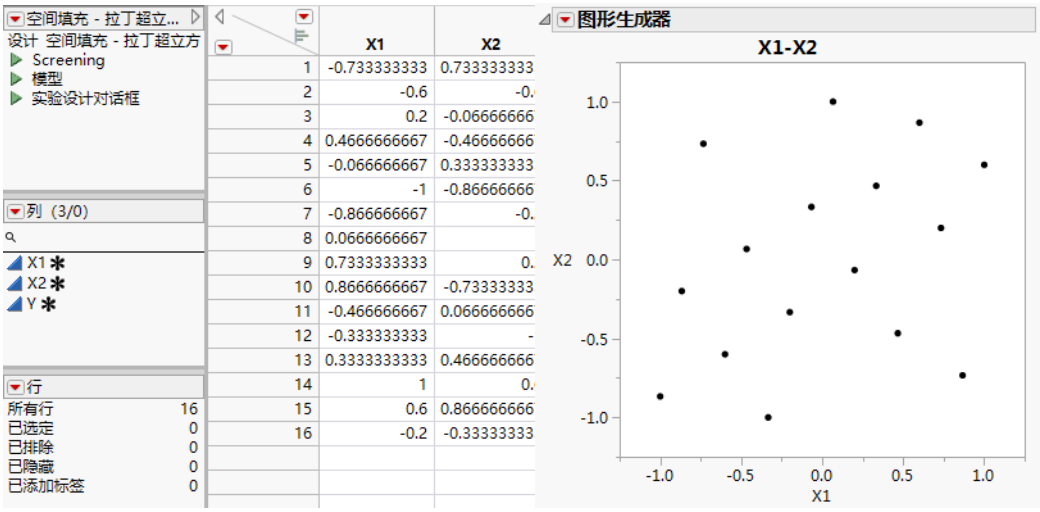


最大熵设计

最大熵设计是用于计算机实验的拉丁超立方体设计的一种替代方案。拉丁超立方体设计是一种很受欢迎的与高斯过程模型配合使用的设计。计算机模拟专家喜欢使用拉丁超立方设计，这是因为到坐标轴的所有投影是均匀的。

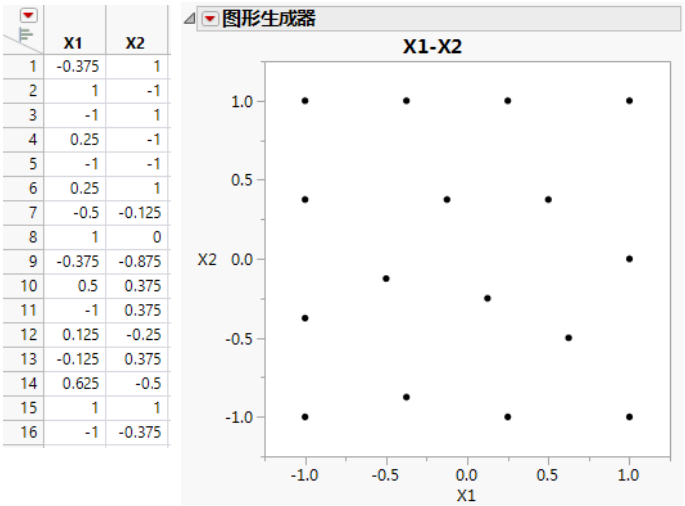
但是，正如图 21.20 中的示例所示，拉丁超立方设计可能不能为您提供最优空间填充属性。该示例是双因子 16 次试验的拉丁超立方设计，它将因子水平设置在 -1 到 1 之间。双因子图显示该设计具有缺失覆盖的区域。特别是对接近 0 的 X1 和接近 -1 的 X2 覆盖很差。

图 21.20 双因子的拉丁超立方设计



最大熵设计优化了实验中所含信息量的测量。请参见下面的技术说明。将因子水平设置在 -1 到 1 之间，图 21.21 中所示的双因子最大熵设计比图 21.20 中所示的拉丁超立方设计更好地覆盖了该区域。当试验次数增加时，设计的空间填充属性会改善。

图 21.21 双因子最大熵设计



技术说明：假定数据来自正态 (m, s^2R) 分布，最大熵设计使实验的 Shannon 信息（Shewry 和 Wynn (1987)）最大化，其中

$$R_{ij} = \exp\left(-\sum_k \theta_k (x_{ik} - x_{jk})^2\right)$$

是两个不同设计点 x_i 和 x_j 上响应值的相关性。在计算上而言，这些设计使 $|\mathbf{R}|$（样本的相关性矩阵的行列式）最大化。若 x_i 和 x_j 相距很远，则 $\mathbf{R}_{ij}$ 接近于 0。若 x_i 和 x_j 相距很近，则 $\mathbf{R}_{ij}$ 接近于 1。

高斯过程 IMSE 最优设计

高斯过程 IMSE 最优设计方法构造适合高斯过程模型的设计。高斯过程模型拟合各种曲面。高斯过程 IMSE 最优设计使实验区域上的高斯过程模型的积分均方误差最小。高斯过程 IMSE 最优设计方法使用的相关性结构类似于 kriging 模型的相关性结构。请参见 Jones and Johnson (2009)。

协方差参数向量

在高斯过程模型的高斯过程 IMSE 最优设计构想中，协方差参数向量决定相关性结构。每个因子有 theta。theta 等于 0 对应于相关性为 1，这使拟合的曲面在相应因子方向上是平坦的。随着 theta 增大，相关性减小，允许曲面在因子方向上弯曲。

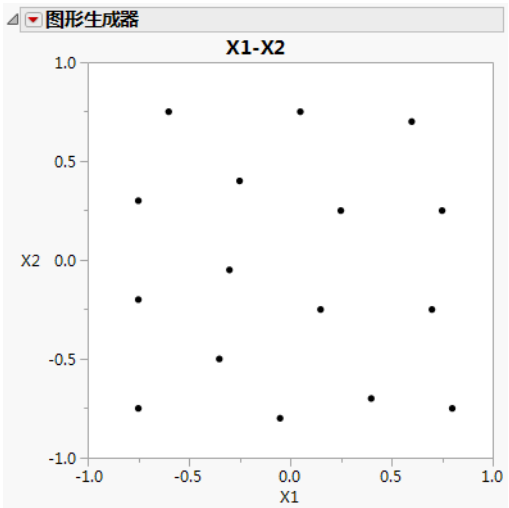
在“协方差参数向量”分级显示项中，在“Thetas”下的值列表中，您可以输入反映曲面先验知识的值。

高斯过程 IMSE 最优设计和拉丁超立方设计的比较

高斯过程 IMSE 最优设计是拉丁超立方设计的备选设计。您可以将 IMSE 最优设计与拉丁超立方设计（前文的图 21.20 中所示）进行比较。图 21.22 中的表和叠加图显示了高斯 IMSE 最优设计。您可以看到该设计实现了因子区域的均匀覆盖。

图 21.22 双因子拉丁超立方和高斯 IMSE 最优设计的比较

	X1	X2
1	-0.05	-0.8
2	-0.75	0.3
3	0.25	0.25
4	0.7	-0.25
5	-0.3	-0.05
6	-0.75	-0.2
7	-0.6	0.75
8	0.6	0.7
9	-0.75	-0.75
10	0.8	-0.75
11	0.4	-0.7
12	-0.35	-0.5
13	-0.25	0.4
14	0.15	-0.25
15	0.75	0.25
16	0.05	0.75



注意：最大熵设计和高斯过程 IMSE 最优设计都是使用 100 作为随机开始数创建的。

快速灵活填充设计

“快速灵活填充”设计的算法最初会在指定设计区域内生成大量随机点。随后使用快速 Ward 算法将这些点聚类到个数与您指定的“试验次数”相同的聚类中。

可使用默认 MaxPro （**最大投影**）最优性准则或通过选择重心法准则来获取最终设计点。

注意：若您有分类因子或因子约束，则“快速灵活填充”是唯一可用的空间填充设计方法。

FFF 最优性准则

“空间填充设计”红色小三角菜单包含 “FFF 最优性准则”下的最优性准则。

MaxPro 对于 p 个因子以及 n 等于指定“试验次数”的情况，MaxPro 准则尝试查找聚类中可最小化以下准则的点：

$$C_{MaxPro} = \sum_{i}^{n-1} \sum_{j=i+1}^{n} \left[1 / \prod_{k=1}^{p} (x_{ik} - x_{jk})^2 \right]$$

MaxPro 准则可在考虑所有因子的前提下，令可能设计点之间的距离乘积最大化。这样即可支持为因子投影提供良好的空间填充属性的目标。请参见 Joseph et al. (2015)。 Max Pro 选项是默认选项。

重心法 该方法在每个聚类的重心放置一个设计点。它具有这样的属性：从设计空间中的任意点到其最近邻近设计点的平均距离比其他设计都要小。

注意：您可以设置首选项，以便始终使用给定的最优性准则。选择**文件 > 首选项 > 平台 > 实验设计**。选择 **FFF 最优性准则**并选择您首选的准则。

分类因子

当您有分类因子时，算法如下所示：

- 设计点总数在分类因子的水平组合总数上是平衡分布的。假定有 m 个水平组合，给 k 个设计点分配这些组合。
- 生成由连续变量定义的设计空间内的大量点。将它们分为 k 个主聚类。
- 将点的 k 个主聚类中的每个主聚类进一步分为 m 个子聚类。
- 在每个主聚类中，使用指定的 FFF 最优性准则计算 m 个子聚类中每个子聚类的设计点。
- 对于 k 个主聚类中的每个主聚类，将 m 个水平组合之一随机分配给 m 个子聚类设计点的每个设计点。这一共得到 km 个设计点。

- 对于每个 *k* 主聚类，都会根据 MaxPro 准则为水平的每一 *m* 组合选择一个设计点，其中，具有相同分类因子水平的点要乘以 MaxPro 分类权重。（有关 MaxPro 分类权重的说明，请参见第 553 页的“空间填充设计选项”。）该过程在全部 *k* 个主聚类中持续执行 10 次，或者持续到更改给定设计点不再有任何改进为止。

为 FFF 设计设置平均聚类大小

“设置平均聚类大小”选项支持您指定用来定义每个聚类（或等效的每个设计点）的随机生成点的平均数。该选项位于“空间填充设计”红色小三角菜单的“高级选项”下方。

默认情况下，若“试验次数”设置为 200 或更小，则聚类算法以总共 10,000 个随机均匀生成的点作为基础。当试验次数超过 200 时，将使用默认值 50。对于因子数很大的设计或当不被允许的组合限制了聚类算法中使用的点的分布时，增加该值可能特别有用。

注意：根据因子数和为“试验次数”指定的值，您可能需要通过选择**高级选项 > 设置平均聚类大小**增大每个设计点的初始点平均数。

约束

完成“因子”分级显示项后，点击**继续**。将显示“定义因子约束”分级显示项。使用该分级显示项可限制设计区域。有关该分级显示项的详细信息，请参见第 549 页的“定义因子约束”。

您可以使用“使用不被允许的组合过滤器”和“使用不被允许的组合脚本”选项来指定不被允许的因子水平组合。或者，您可以使用“指定线性约束”选项来指定线性不等式形式的界限。但是，这两种方法所生成的设计有所不同。

“使用不被允许的组合过滤器”和“使用不被允许的组合脚本”

当您指定不被允许的组合时，构成聚类算法基础的随机点在不受约束的设计区域内随机分布。然后会删除不被允许的点，使用其余的点继续执行聚类。

注意：根据约束的性质和指定的试验次数，初始随机生成的点所形成的不受约束的设计空间的默认覆盖范围可能不足以得到所需的试验次数。在这种情况下，您可能收到 JMP 警示，指出算法“无法找到足够的点数”。要增大构成聚类算法基础的初始点数，需要通过选择**高级选项 > 设置平均聚类大小**来指定更大的每个设计点所对应的初始点平均数。（请参见第 570 页的“为 FFF 设计设置平均聚类大小”）。

指定线性约束

当您使用**指定线性约束**选项时，构成聚类算法基础的随机点在受约束的设计区域内随机分布。聚类算法使用这些点。

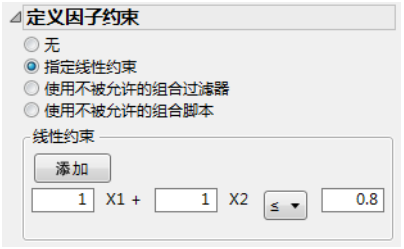
创建和查看受约束的快速灵活填充设计

本例创建和查看受约束的快速灵活填充设计。

构造设计

1. 选择**实验设计 > 特殊目的 > 空间填充设计**。
2. 添加 2 个连续因子。
3. 将因子水平值更改为 0 和 1，然后点击**继续**。
4. 在 “定义因子约束”分级显示项中，选择**指定线性约束**。
   请注意，快速灵活填充是唯一可用的空间填充设计方法。
5. 选择**添加**。
6. 输入以下系数和界限：
   X1 为 1
   X2 为 1
   界限为 0.8

图 21.23 线性约束



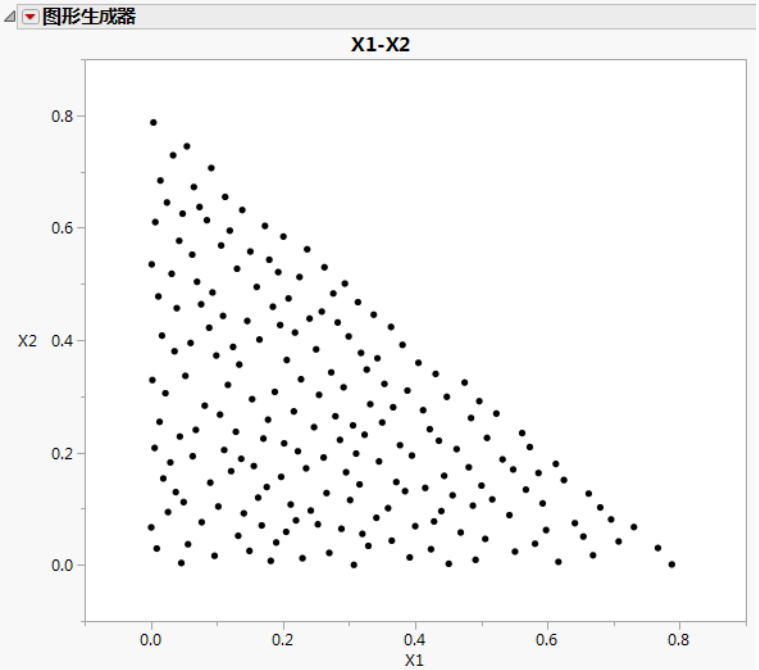
7. 在**试验次数**旁边键入 200。
8. 选择**快速灵活填充**。
   JMP 将创建一个满足这些约束的设计。打开 “设计”分级显示项以查看该设计。
9. 选择**制表**构造数据表。

构造图

1. 数据表处于活动状态时，选择**图形 > 图形生成器**。
2. 将 X1 拖到标记为 X 的拖放区。
3. 将 X2 拖到标记为 Y 的拖放区。
4. 通过点击平滑线图标删除**平滑线**。
5. 点击 “图形生成器”红色小三角，然后点击**显示控制面板**以取消选择它。

您看到的图形应类似于图 21.24 中的图。请注意这些点满足线性约束 $X1 + X2 \leq 0.8$。

图 21.24 具有一个线性约束的快速灵活填充设计



为地图形状创建空间填充设计

可以使用快速灵活填充 (FFF) 设计来设计要为跨地图形状之类的区域执行的实验。例如，若您有两种样本收集方法（可能用于空气或土壤样本），希望在美国某州分布的各个位置进行比较，则考虑 FFF 设计。

假定相关的美国州是佐治亚州。您需要 40 个样本，每种抽样方法 20 个样本，遍布全州。若要为地图形状创建 FFF 设计，您需要一个具有地图边界纬度和经度的文件（请参见《基本绘图》）。

在下例中，您将确定美国佐治亚州的经度和纬度，定义该州的形状，然后创建设计。该设计将为 40 个样本中的每一个样本提供位置和抽样方法。

查找佐治亚州的经度和纬度

1. 导航到 Maps 目录，通常会在此找到：

   Windows: C:/Program Files/SAS/JMP/< 版本 >/Maps

   macOS：/Library/Application Support/JMP/< 版本 >/Maps
2. 打开 US-State-XY.jmp 和 US-State-Name.jmp。

请注意，在 US-State-Name.jmp 中，Shape ID 11 与美国佐治亚州对应。在本例中，这就是相关州。

3. 在 US-State-XY.jmp 中，选择**行 > 行选择 > 选择符合条件的行**。
4. 选择**形状 ID** 并在文本框中输入 11，然后点击**确定**。

   这将选定形状 ID 为 11 （代表佐治亚州）的所有行。
5. 选择**表 > 子集**并点击**确定**以获取 US-State-XY.jmp 表中包含佐治亚州纬度和经度数据的子集 （选定行）。
6. 关闭 US-State-XY.jmp 和 US-State-Name.jmp。

获取值范围

要定义空间填充设计的空间，需要知道佐治亚州纬度和经度的值范围。然后，您可以使用这些值创建设计因子。

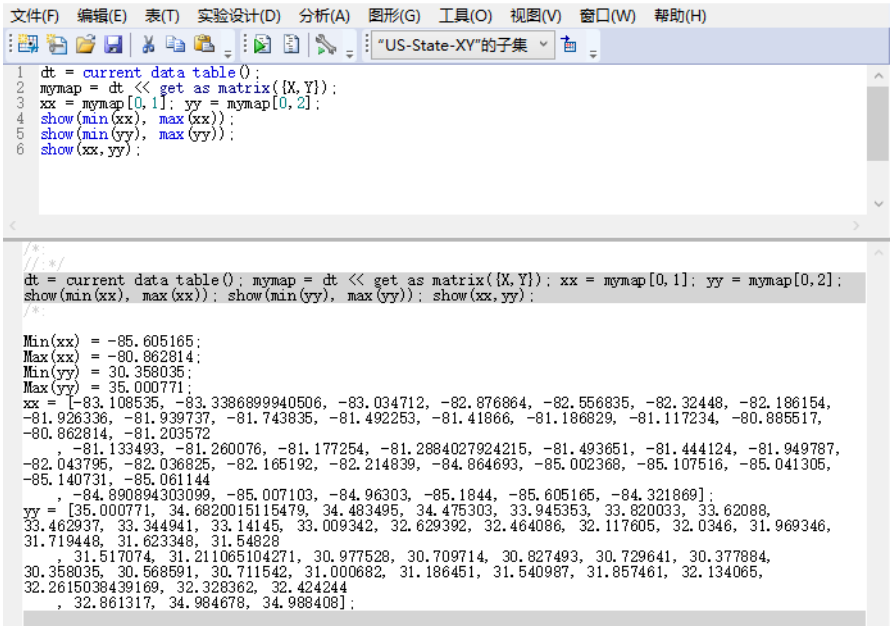
1. 选择**文件 > 新建 > 脚本**。在脚本窗口中，右击并选择**显示嵌入的日志**。
2. 复制以下内容并将其粘贴到新脚本中：

```
dt = current data table();
mymap = dt << get as matrix({X,Y});
xx = mymap[0,1];
yy = mymap[0,2];
show(min(xx), max(xx));
show(min(yy), max(yy));
show(xx,yy);
```

3. 在 US-State-XY.jmp 表的子集处于活动状态时，运行该脚本。

   佐治亚州纬度和经度的值范围以及 **xx** 和 **yy** 矩阵显示在日志中。

图 21.25 “范围查找”脚本和日志中的结果



添加因子

1. 选择**实验设计 > 特殊目的 > 空间填充设计**。
2. 向设计中添加两个连续因子。在**添加因子数**文本框中输入 2，然后点击**连续型**。
3. 重命名因子经度和纬度并在 “因子”分级显示项中输入其最小值和最大值。
   - 对于经度，输入 -85.6 和 -80.9 作为经度值。这些是日志中 x 的最小和最大经度值。
   - 对于纬度，输入 30.4 和 35.0 作为纬度值。这些是日志中 y 的最小和最大纬度值。
4. 添加一个分类因子。点击**分类 > 2 水平**。重命名因子方法。

图 21.26 因子

因子

名称	角色	值	
longitude	连续	-85.6	-80.7
latitude	连续	30.3	35
Method	分类	L1	L2

5. 点击**继续**。

定义形状

要进一步定义空间填充设计的空间，需要使用多边形定义州的形状。

1. 选择**使用不被允许的组合脚本**打开“不被允许的组合表达式”文本框。

经度和纬度范围定义一个正方形。佐治亚州是一个多边形，而不是正方形。您想排除落在佐治亚州多边形之外的点。

2. 转至脚本窗口并从脚本日志中复制 **xx** 和 **yy** 矩阵。将其粘贴到“不被允许的组合表达式”文本框。
3. 最后，添加以下行：

```
!In Polygon(longitude, latitude , xx, yy);
```

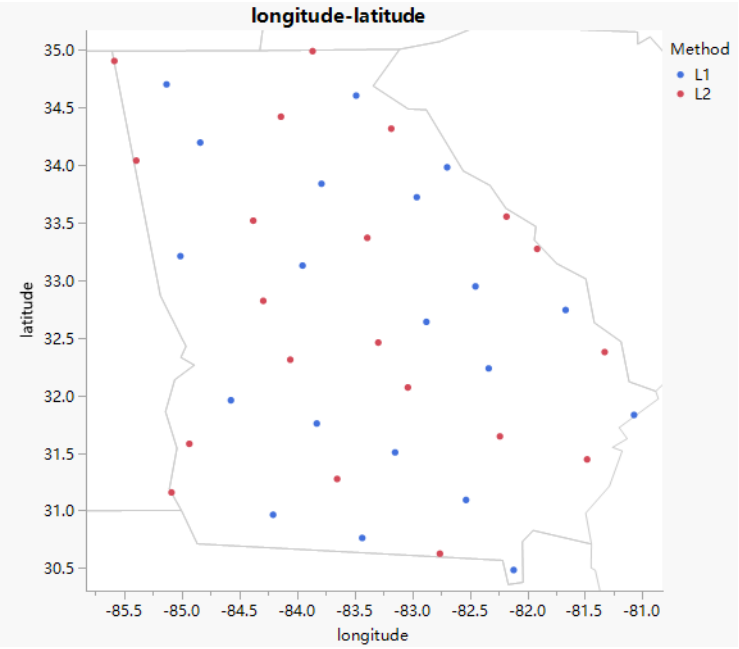
这将排除 **xx** 和 **yy** 坐标所定义的多边形中未包含的点。

图 21.27 不被允许的组合表达式

制作设计

1. 将**试验次数**改为 40 并点击**快速灵活填充**。
2. 点击**制表**。
3. 从您的设计表中，选择**图形 > 图形生成器**。
4. 将以下列拖入以下区域：
   - 将 longitude 拖入 X 区域
   - 将 latitude 拖入 Y 区域
   - 将 Method 拖入**颜色**区域
5. 在图形之上，点击 “平滑线”元素图标将其关闭。
6. 要添加州形状，右击该图形并选择**图形 > 背景地图**。
7. 在**边界**下，选择**美国各州**并点击**确定**。

图 21.28 美国佐治亚州的 FFF 设计示例



注意：由于 FFF 算法的起点随机性的原因，地图上点的准确位置可能与所示位置不同。

地图上标识出您应该收集样本的 40 个地点。抽样方法（L1 或 L2）由点的颜色指示。

“图形”>“背景地图”窗口包含以下附加选项：

- 要确定用于测试的郡县，请在**边界**下选择**美国各郡**。
- 要在街道一级绘制测试位置，请在**图像**下方选择**街道图服务**。

球堆积设计的示例

Worley (1987) 提出了一个通过钻孔 （从地表到两个蓄水层钻一个孔）的水流模型。响应变量 y 是通过钻孔的流速，单位为米 ^{3}/ 年，它的值通过以下等式计算：

$$y = \frac{2\pi T_u (H_u - H_l)}{\ln(r/r_w)\left[1 + \frac{2LT_u}{\ln(r/r_w) r_w^{\,2} K_w} + \frac{T_u}{T_l}\right]}$$

在该模型中有八个因子：

r_w = 钻孔的半径，为 0.05 到 0.15 米

r = 影响半径，为 100 到 50,000 米

T_u = 上部蓄水层的渗透系数，为 63,070 到 115,600 米 ^{2}/ 年

H_u = 上部蓄水层的电位头，为 990 到 1100 米

T_l = 下部蓄水层的渗透系数，为 63.1 到 116 米 ^{2}/ 年

H_l = 下部蓄水层的电位头，为 700 到 820 米

L = 钻孔长度，为 1120 到 1680 米

K_w = 钻孔导水系数，为 9855 到 12,045 米 / 年

您可以使用球堆积设计获取一组计算响应 y 的条件。然后您可以生成一个模型来估计在设计中使用的输入范围上的真实模型。评估估计的模型可以帮助了解八个因子各自对响应的影响。

为钻孔数据创建球堆积设计

要为钻孔模型创建球堆积设计，您可以使用已保存的因子设置的数据表。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Borehole Factors.jmp
2. 选择**实验设计** > **特殊目的** > **空间填充设计**
3. 点击 “空间填充设计”红色小三角菜单并选择**加载因子**。

图 21.29 “因子”面板 （已加载了钻孔示例的因子值）

因子

连续 | 分类 | 删除 | 添加因子数 1

名称	角色	值	
log10 Rw	连续	-1.3	-0.82
log10 R	连续	2	4.7
Tu	连续	63070	115600
Tl	连续	63.1	116
Hu	连续	990	1110
Hl	连续	700	820
L	连续	1120	1680
Kw	连续	9855	12045

注意：将 r 和 r_w 的对数作为因子。

4. 点击**继续**。
5. 将 “试验次数”设置为 32 并按 Enter。
6. 点击**球堆积**以生成设计。
7. 点击**制表**以生成一个表，其中显示实验的设计设置。

注意：设计表包含“模型”表脚本。该脚本运行响应 y 的高斯过程模型。

8. （可选）要查看该示例的已完成的数据表，请选择**帮助** > **样本数据库**，然后打开 Design Experiment/Borehole Sphere Packing.jmp。

   因为是从随机种子生成的设计，您获得的设置可能不同于完成的表中所示的那些设置。

确定性数据分析准则

切记确定性数据没有随机成分。相同的输入值生成相同的输出。因此，拟合的统计模型中的 p 值不具有通常的含义。大的 F 统计量（低 p 值）指示模型项带来的效应。但是，您不能构造效应或模型预测的有效置信区间。

对确定性数据的模型拟合的残差并不是噪声的测度。残差是模型偏倚的测度。偏倚是真实值与预测值的差值。残差中的非重复模式指示应为模型考虑其他项来减小偏倚。

分析钻孔球堆积设计

真实模型通常无法在简单解析表达式中提供。因此，仅知道观测的数据点处的预测偏倚。但是，在本示例中，真实模型的函数形式是已知的。在 Borehole Sphere Packing.jmp 数据表中，**真实模型**列包含已知函数的公式。通过该公式，您可以刻画因子输入区域上的预测偏倚。

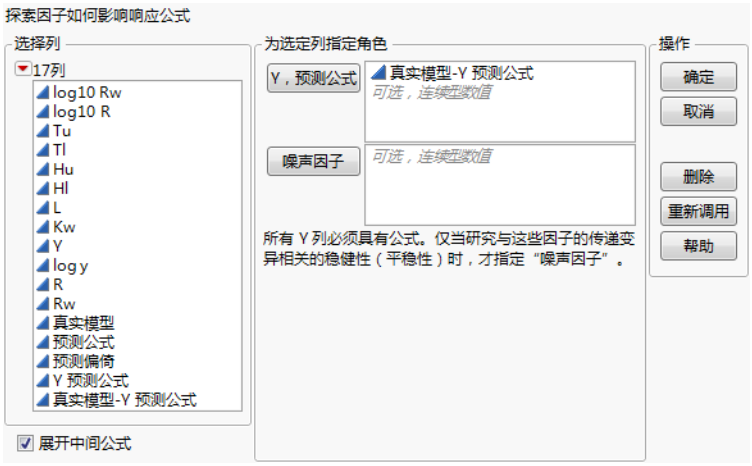
1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Borehole Sphere Packing.jmp。
2. 点击“模型 （来自实验设计的 GP）”脚本旁边的绿色小三角。

   使用“高斯过程模型”报表探索因子和结果 Y 之间的关系。
3. 点击“‘Y’的高斯过程模型”旁边的红色小三角，然后选择**保存预测公式**。
4. 返回到 Borehole Sphere Packing.jmp 数据表。
5. 在数据网格中，选择**真实模型**和 **Y 预测公式**的列标题。
6. 右击并选择**新建公式列** > **合并** > **差值**。

   这将创建包含偏倚的新列。
7. 从 Borehole Sphere Packing.jmp 数据表，选择**图形** > **刻画器**。
8. 选择**真实模型 -Y 预测公式**并点击 **Y，预测公式**。
9. 选择**展开中间公式**。

   该选项将偏倚显示为八个设计因子的函数。

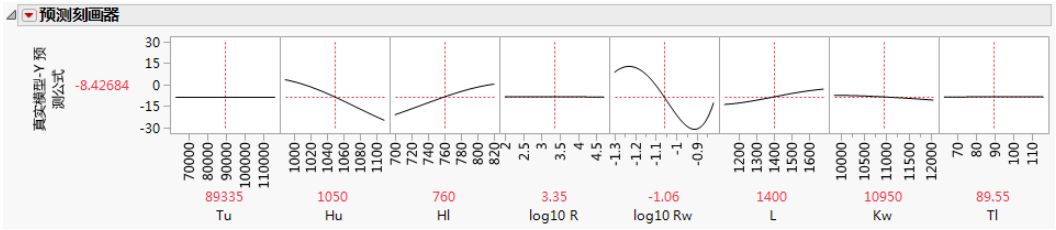
图 21.30 钻孔球堆积数据的刻画器对话框



10. 点击**确定**。

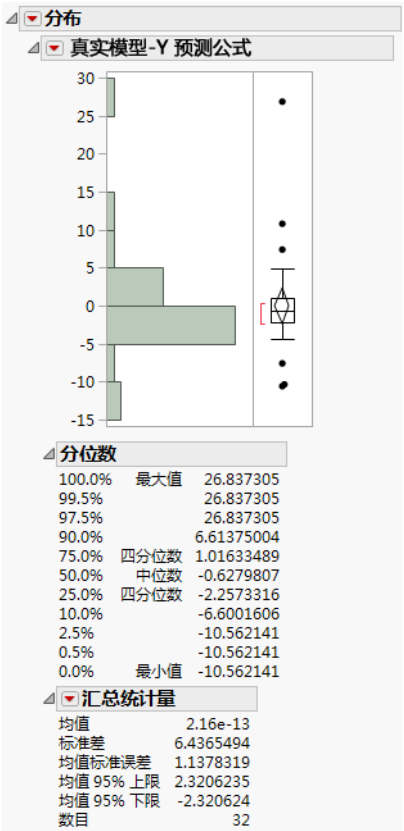
刻画器默认为设计区域的中心。若没有偏倚，则所有刻画迹线在每个因子的值范围之间将为常数。在本示例中，变量 logRw、 Hu 和 Hl 显示对偏倚的最大效应。

图 21.31 钻孔高斯过程模型 （Y 轴设置为 -30 到 30）的偏倚刻画器



您可以使用刻画器探索整个域上预测偏倚的范围。要找到最小和最大偏倚点，请从“预测刻画器”红色小三角菜单中选择**优化和意愿 > 意愿函数**。请参见《刻画器指南》。要评估设计点上的预测偏倚，请选择**分析 > 分布**来查看分布分析。

图 21.32 预测偏倚的分布



切记，在本示例中，真实模型是已知的。在很多应用中，任何因子设置的响应都是未知的。实验数据上的预测偏倚可能低估了整个设计域上的偏倚。

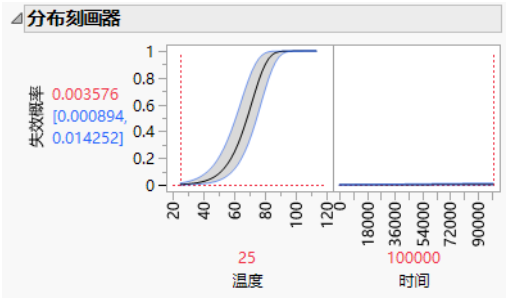
第 22 章

加速寿命试验设计

为加速寿命试验设计实验

使用“加速寿命试验 (ALT) 设计”平台可为加速寿命试验设计计划。处在正常使用条件下的产品的可靠性往往非常高，以至于检验产品一直到产品失效所需的时间过长而令检验不可行。所以不应在正常使用条件下检验产品，应改为在比正常使用条件更严重的条件下检验产品。这些严酷条件会导致产品更快退化和失效。您随后可以使用这一加速失效数据来预测产品在正常使用条件下的可靠性。您可以设计初始实验或扩充现有实验。

图 22.1　显示加速寿命试验的失效概率的刻画器



目录

加速寿命试验设计概述 583
加速寿命试验设计的示例 583
扩充加速寿命试验设计的示例 590
构建加速寿命试验设计 594
    指定设计结构 595
    指定加速因子 595
    指定设计详细信息 596
    检查和更新 ALT 计划规格 598
    创建和评估最优 599
    创建设计表 600
加速寿命试验选项 601
“ALT 设计”平台的统计详细信息 602
    ALT 设计的失效分布 602
    R 精度因子（95% 置信区间） 603
    加速寿命试验的最优性准则 604

加速寿命试验设计概述

当产品在正常使用条件下的可靠性非常高时使用加速寿命试验。您可以在比正常情况更严酷的条件下试验产品，以加快失效出现的时间。随后使用这一加速失效数据来预测产品在正常使用条件下的可靠性。

此类检验即称为**加速寿命检验** (ALT)。设定为可加速失效时间的水平的因子称为**加速因子**。加速寿命试验的模型通常为非线性模型。有关非线性模型的详细信息，请参见“非线性设计”一章中第 624 页的“非线性模型”。

“ALT 设计”平台可以为涉及一个或两个加速因子的情形创建和评估设计。您可以使用 D 最优性、分位数估计或失效概率优化设计。

创建 ALT 设计需要加速模型参数的初始估计值。使用主题知识定义初始估计值。若您不具备信息来说明您的初始估计值，您可以指定多元正态先验分布来说明其不确定性。

您还可以使用“加速寿命检验 (ALT) 设计”平台扩充当前实验。您可能想要扩充当前设计以减小估计值方差。

加速寿命试验设计的示例

使用 ALT 设计器构造失效概率最优 ALT 试验。在本例中，假定您需要为某个机械组件设计一个加速寿命试验。唯一的加速因子就是扭矩，正常使用强度为 35 Nm（牛顿米）。您想要估计正常使用强度下有 10% 的单元失效时的时间。您构造一个失效概率最优 ALT 试验。

您的试验计划具有以下特点：

- 总共有 100 个单元接受试验。
- 寿命分布假定采用 Weibull 分布。
- 通过对数变换给出寿命-应力关系。
- 您计划检验扭矩的三个应力水平：50、75 和 100 Nm。
- 可利用先验知识来预测这些试验水平下的初始失效时间。请参见第 583 页的“获取先验估计值”。
- 该试验运行 5,000 个周期。
- 您需要持续监控过程中出现的失效。

获取先验估计值

要创建加速寿命试验设计，您必须提供参数的先验估计值。下面提供了获取先验估计值的方式：

1. 利用您的流程知识创建一个包含少量应力水平下的失效时间的表，表中数据虽是假设的，但却是很可能的。

2. 使用 “以 X 拟合寿命”平台拟合模型并获取模型参数的估计值。
3. 将这些估计值用作您的先验值，以便使用 “ALT 设计”平台创建设计。

按照上述概括的方法创建一个数据表，其中包含用于平衡设计的特定失效周期数的估计值。表中包含要在设计中使用的单元（每个应力水平有五个单元，共三个应力水平）。您的数据表为 Torque Prior.jmp，该表位于 Sample Data Library 中。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Torque Prior.jmp。
2. 选择**分析** > **可靠性和生存** > **以 X 拟合寿命**。
3. 选择**周期数**并点击 **Y，事件时间**。
4. 选择**扭矩**并点击 **X**。
5. 从**关系**列表中，选择**对数**。
6. 为**使用条件**输入 35。回想一下， 35 Nm 是正常使用条件。
7. 从**分布**列表中，选择 Weibull。

图 22.2 “以 X 拟合寿命”启动窗口

8. 点击**确定**。
9. 向下滚动至 “Weibull 结果”分级显示项并打开 “相关性矩阵”分级显示项。

图 22.3 先验数据的以 X 拟合寿命模型

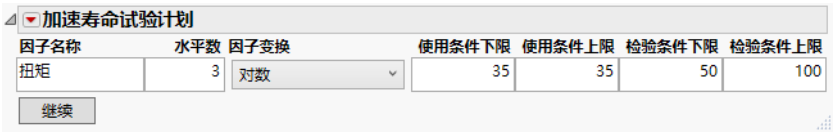
均值模型在“估计值”分级显示项中提供。“估计值”列包含截距 (β0)、线性系数 (β1) 和尺度 (σ) 的参数估计值。标准误差列包含估计值的标准误差。参数估计值的估计相关性在“相关性矩阵”分级显示项中给出。将这些参数估计值、标准误差和相关性用作构造 ALT 设计的先验值。

输入基本规格

要创建加速寿命试验设计，需要定义加速因子、使用条件和试验条件。

1. 选择**实验设计 > 特殊目的 > 加速寿命试验设计**。
   请注意，默认选定**一个加速因子**和**连续监控**。
2. 点击**继续**。
3. 在**因子名称**下，点击 X1 并键入**扭矩**。
   请注意**水平数**默认设置为 3。
4. 在“因子变换”下选择**对数**。
5. 为**使用条件下限**和**使用条件上限**均输入 35。
   将使用条件的上限和下限都设置为 35，这表示 35 是正常使用条件。
6. 为**检验条件下限**输入 50，为**检验条件上限**输入 100。

图 22.4 完成的 ALT 规格窗口



7. 点击**继续**。
8. 请注意，**扭矩水平值**设置为 50、 75 和 100。

注意：JMP 将各个水平的值设置为高低检验条件之间的等间距值。您可以更改这些水平。

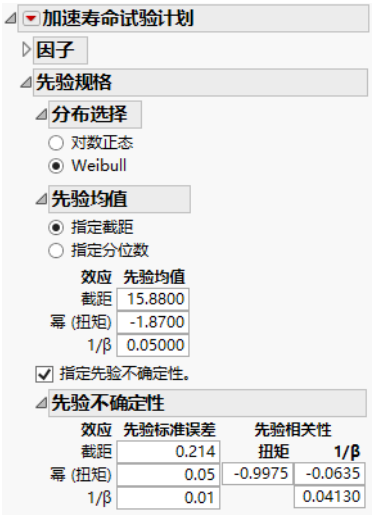
9. 确保将 Weibull 选作**分布选择**。

输入先验信息和其他规格

通过输入先验信息继续构造示例 ALT 设计。

1. 在**先验均值**下，选择**指定截距**并输入您使用“以 X 拟合寿命”平台和假设数据获得的先验加速模型参数估计值。
   - 为**截距**输入 15.88。
   - 为**功效（扭矩）**输入 -1.87。
   - 为 1/β（Weibull 模型的尺度参数）输入 0.05。
2. 选择**指定先验不确定性**。
3. 输入先验加速模型的估计标准误差和相关系数。图 22.3 显示完成的 “先验不确定性”分级显示项：

图 22.5 完成的 “先验规格”分级显示项



点击 “加速寿命试验计划”红色小三角，然后选择 **ALT 最优性准则 > 制作失效概率最优**。

您计划执行超过 5,000 个周期的试验。不过，您关注的是对多达 10,000 个周期的预测。

4. 在 “诊断选择”分级显示项中，为**关注的时间范围**的两个框均输入 10,000。

5. 在“设计选择”分级显示项中，为**试验长度**输入 5,000，为**试验单元的数量**输入 100。

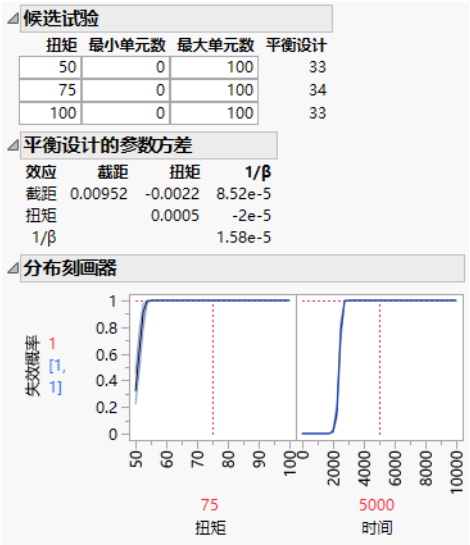
图 22.6 完成的“诊断选择”和“设计选择”分级显示项

制作设计

通过生成设计继续演示 ALT 示例。

1. 点击**继续**。

图 22.7 平衡设计诊断



平衡设计中的试验次数显示在“候选试验”分级显示项中。“平衡设计的参数方差”分级显示项显示该设计的参数的协方差矩阵。同时还会显示“分布刻画器”。获取最优设计时，您可以将“参数方差”和“分布刻画器”结果与平衡设计的这些结果作比较，查看不确定性是否降低。

2. 点击**制作设计**。

将显示最优实验设计以及其他结果。

图 22.8 最优设计

设计

扭矩	单元数	期望失效数	所有删失概率	失效概率
50	81	26.1	0.0	0.32
75	0	0.0	0.0	1
100	19	19.0	0.0	1

基于检验试验的水平、要试验的单元总数和您指定的先验信息计算最优设计。最优设计会在每个扭矩水平下试验指定的单元数：

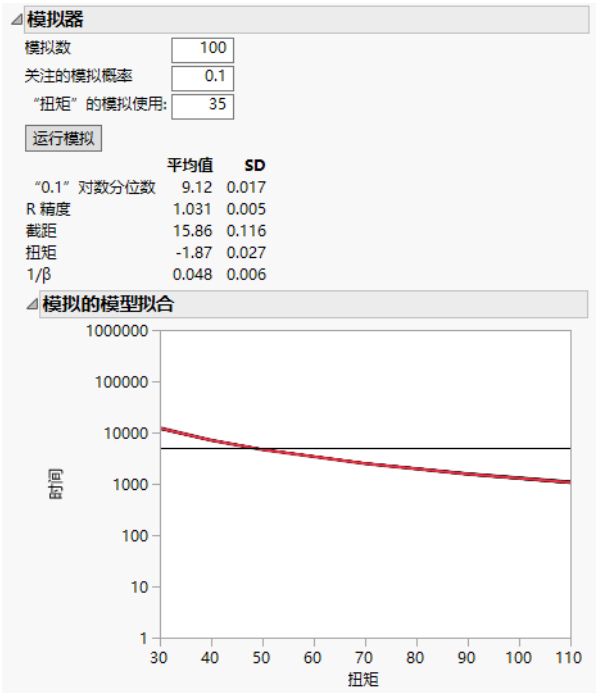
- 50 Nm 下 81 个单元
- 75 Nm 下 0 个单元
- 100 Nm 下 19 个单元

运行模拟

使用模拟器评估示例 ALT 设计。使用模拟器模拟基于用于构建设计的假设的试验计划的结果。这是一种在运行试验计划前评估试验计划是否充分的方法。

1. 为**模拟数**输入 100。
2. 为**关注的模拟概率**输入 0.1。
3. 为 **“扭矩”的模拟使用**输入 35。
4. 点击**运行模拟**运行 100 次模拟实验。这样可以检查针对设计的先验假设，您计划试验的 100 个单元是否充分。

图 22.9 模拟结果



模拟的模型拟合显示 100 次模型模拟均相似，因为它们全部落在红色曲线所示的平均值附近。 Y 轴位于对数尺度上。

5. 双击 Y 轴并从 “类型” 菜单中选择**线性**，然后点击**确定**。

   现在很容易看到围绕 35 次正常使用条件的模拟拟合中的变异性。您期望在正常使用条件的估计值中看到变异性。模拟可帮助您评估您的 ALT 计划是否可接受。若变异过多，您可能需要增加试验的单元数或调整试验条件。

保存设计

最后，通过制定试验计划并将其保存到数据表中，完成示例 ATL 设计。

1. 点击**制定试验计划**将设计保存至数据表。该表是设计的汇总。
2. 点击**制表**将设计保存至数据表。该表包含 100 行，每个检验试验一行。

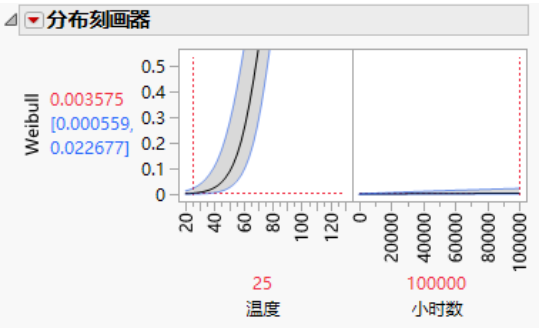
扩充加速寿命试验设计的示例

下面的示例说明如何使用“加速寿命试验设计”平台扩充现有设计。在本例中，在三个温度（85°、105° 和 125° 摄氏度）下对 150 个电容单元进行了 1500 小时的试验。结果记录在 Capacitor ALT.jmp 样本数据表中。生成的模型用于预测 25° 摄氏度的正常使用温度下，经过 100,000 小时后失效的总体比例。

检查当前预测

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Capacitor ALT.jmp。
2. 点击绿色小三角以运行**以 X 拟合寿命**表脚本。
3. 在“分布刻画器”中（位于“分布”选项卡上的“比较”分级显示项中），更改因子设置：
   - 点击**温度**上方的 105 并将其改为 25。
   - 点击**小时**上方的 750.5 并将其改为 100,000。

图 22.10 电容模型的分布刻画器



根据您当前的研究，预测的在 25° 下经过 100,000 个小时失效的电容总体比例为 0.003575，95% 置信区间为 0.00056 到 0.02268。您想要更精确地估计失效比例。要减小置信区间的宽度（测量您的估计值精度的一个测度），您可以添加额外的试验来扩充研究。

扩充设计

您想要扩充设计，以获取关于预测失效比例的更为精确的估计值。原始设计使用的温度设置为 85、105 和 125。在扩充设计中，您想要试验温度值 90、110 和 125。请注意，其中两个都是新设置。使用优选的试验扩充设计：

1. 选择**实验设计** > **特殊目的** > **加速寿命试验设计**。
2. 选择**一个加速因子**，再选择**连续监控**，然后点击**继续**。
3. 为**因子名称**输入**温度**。
4. 为**水平数**输入 5。

尽管您的扩充试验仅跨越三个水平 （90、 110 和 125），您还必须指定最初实验中使用的水平，所以总水平数是五个。“因子变换”默认设置为 “Arrhenius 摄氏温度”。

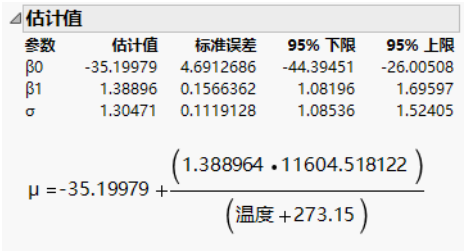
5. 为**使用条件下限**和**使用条件上限**均输入 25。
6. 为**检验条件下限**输入 85，为**检验条件上限**输入 125。
7. 点击**继续**。
8. 为**温度水平值**输入 85、 90、 105、 110 和 125。

   三个水平来自最初实验 （85、 105 和 125）。扩充设计新增两个水平 （90 和 110）以及来自第一个实验的水平之一 (125)。

   注意：所有水平都必须列出来。

9. 确保将 Weibull 选作**分布选择**。
10. 在**先验均值**下，选择**指定截距**。输入 “以 X 拟合寿命估计值”分级显示项中的当前加速模型参数估计值，该分级显示项位于 “统计量”选项卡上的 “Weibull 结果”分级显示项中。

图 22.11 “Weibull 结果”分级显示项中的参数估计值和拟合模型



估计值

参数	估计值	标准误差	95% 下限	95% 上限
β0	-35.19979	4.6912686	-44.39451	-26.00508
β1	1.38896	0.1566362	1.08196	1.69597
σ	1.30471	0.1119128	1.08536	1.52405

$$\mu = -35.19979 + \frac{(1.388964 \cdot 11604.518122)}{(\text{温度} + 273.15)}$$

   – 为**截距**输入 -35.200。
   – 为**活化能量（温度）**输入 1.389。

     这是活化能量的估计值，也是乘以 Boltzmann 常数后用开氏温度衡量的逆温度的系数。
   – 为 1/β 输入 1.305。

     对于 Weibull 分布， JMP 使用依赖于位置参数 μ 和尺度参数 σ 的参数化。依照常见的 α 和 β 参数化，尺度参数为 σ = 1/β。请参见第 602 页的 “Weibull”。

   在“加速寿命试验计划”窗口中，您可以通过选择**指定先验不确定性**选项来指定先验均值的不确定性。在本例中，您不指定先验不确定性。创建的设计假定在 “先验均值”分级显示项中指定的值是真实的参数值。
11. 在 “诊断选择”下，为**试验长度**输入 1500。

    该试验进行 1500 小时以上，这是原始设计的时长。
12. 为**试验单元的数量**输入 250。

    之前的实验试验了 150 个单元，扩充实验还要另外试验 100 个单元，总计 250 个单元。

图 22.12 完成的 “设计详细信息”窗口

13. 点击**继续**。
14. 在**候选试验**下，输入 50 作为温度 85、105 和 125 的最小单元数。这些表示已经完成的试验。
15. 点击 “加速寿命试验计划”红色小三角，然后选择 **ALT 最优性准则 > 制作失效概率最优**。

    有关该准则的详细信息，请参见第 601 页的 “制作失效概率最优”。
16. 点击**制作设计**。

    将显示最优实验设计以及其他结果。

图 22.13 最优设计

设计

温度	单元数	期望失效数	所有删失概率	失效概率
85	50	6.9	0.0	0.14
90	58	12.3	0.0	0.21
105	50	30.0	0.0	0.6
110	0	0.0	0.0	0.75
125	92	91.2	0.0	0.99

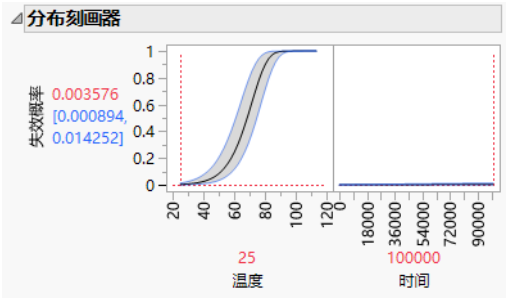
基于检验试验的水平、要试验的最小单元数、要试验的单元总数（即“候选试验”分级显示项中的信息）以及您指定的其他信息来计算最优设计。最优设计会在每个温度水平下试验指定的单元数：

- 在 85°C 下包含 50 个单元。由于上一个实验已经在 85°C 下试验了 50 个单元，所以无需更多单元。
- 在 90°C 下包含 58 个单元。
- 在 105°C 下包含 50 个单元。由于上一个实验已经在 105°C 下试验了 50 个单元，所以无需更多单元。
- 在 110°C 下包含 0 个单元。无需该水平。
- 在 125°C 下包含 92 个单元。由于上一个实验已经在 125°C 下试验了 50 个单元，所以还需要 42 个单元。

将扩充设计与最初研究作比较

1. 在“分布刻画器”中，为“温度”输入正常使用条件 25，为“时间”输入 100,000。失效的总体比例估计值为 0.00357，95% 置信区间为 0.00089 到 0.014525。该区间比最初实验中的区间（介于 0.00056 到 0.02268 之间，如图 22.10 所示）要窄。

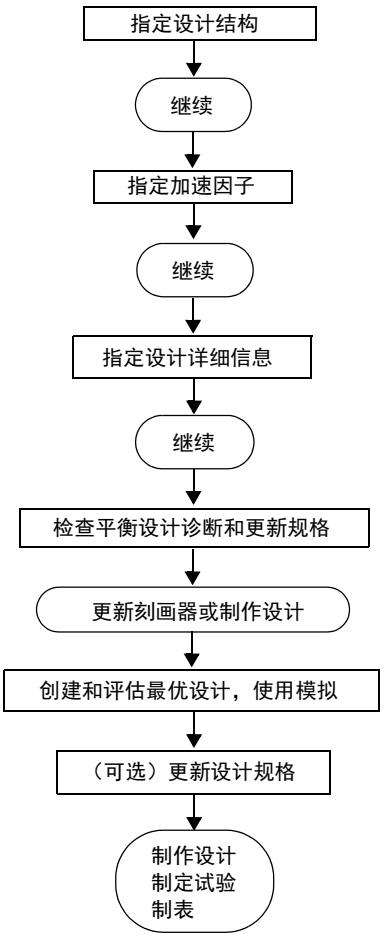
图 22.14 温度 = 25，时间 = 100000 时的分布刻画器



构建加速寿命试验设计

通过选择**实验设计 > 特殊目的 > 加速寿命试验设计**来构建加速寿命试验 (ALT) 设计。在设计过程中，“加速寿命试验计划”会随之更新。用于构建加速寿命试验设计的步骤遵循图 22.15 中的流程。

图 22.15 加速寿命试验计划流程

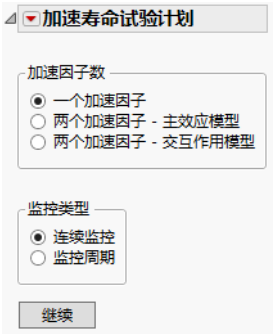


本节说明加速寿命试验计划流程中的分级显示项。

指定设计结构

使用 “加速寿命试验计划”分级显示项定义设计的模型结构和检查类型。

图 22.16 初始 “ALT 设计”窗口



加速因子数

一个加速因子 选择用于含有一个因子的设计。

两个加速因子 - 主效应模型 选择双因子设计，该设计针对主效应模型

两个加速因子 - 交互作用模型 选择双因子设计，该设计针对包含主效应和交互作用项的模型。

监控类型

连续监控 选择记录了精确失效时间的设计。超过检查长度的失效时间将被右删失。

监控周期 选择按间隔检查单元失效情况的设计。失效时间为区间删失。输入检查次数、第一次检查的时间和各次检查之间的时间。对于不规则的检查区间，您可以随后在 “设计选择”分级显示项中更改检查时间。

加速寿命试验计划 指定关于一个或多个加速因子的详细信息。

指定加速因子

指定关于一个或多个加速因子的详细信息。

图 22.17 ALT 规格窗口

因子名称 输入每个加速因子的名称。

水平数 对于每个加速因子，输入要在实验中包含的建议水平数。

因子变换 对于每个加速因子，选择一个变换函数。该变换说明寿命 - 应力关系，也就是寿命分布在应力水平之间更改的方式。可用的变换包括 “Arrhenius 摄氏温度”、“倒数”、“对数”、“平方根”和 “线性”。

使用条件下限 对于每个加速因子，输入典型使用情况下其值的下限。

使用条件上限 对于每个加速因子，输入典型使用情况下其值的上限。

注意：“使用条件下限”值可与 “使用条件上限”值相等。当使用条件为单个值时，使用相等值。

检验条件下限 对于每个加速因子，输入检验条件下限。

检验条件上限 对于每个加速因子，输入检验条件上限。

注意：检验条件下限和上限定义设计中加速因子的端点。初始候选项使用试验条件下限和上限之间等间距的水平。您可以在构造设计之前调整水平。

指定设计详细信息

指定因子水平、先验分布详细信息、时间范围和关注的概率，以及试验长度和要试验的单元数。

图 22.18 分布详细信息

因子 输入加速因子的水平。默认情况下，各个水平在检验条件下限和上限之间等间距分布。

分布选择 为每个加速因子选择寿命分布（对数正态或 Weibull）。请参见第 602 页的“‘ALT 设计”平台的统计详细信息”。

先验均值 输入加速模型参数的先验估计值。先验估计值是 Bayes 先验分布中的超参数。先验均值可以是基于主题知识的最佳猜测，也可以基于以前的研究。

若只有一个加速因子，可通过一个选择来指定截距或分位数。

指定截距 为包括截距在内的模型参数输入先验值。

指定分位数 输入加速因子的先验值、给定时间的期望失效比例以及加速因子值。该信息用于计算截距。

指定先验不确定性 选择以输入标准误差值和加速模型参数的先验分布相关系数。标准误差反映了相对于加速模型参数的先验估计值的不确定性。

注意：若未指定先验不确定性测度，则通过将“先验均值”下输入的值视为真实参数值来创建设计。若先验均值参数接近真值，则该设计接近最优。不过，若参数估计值指定有误，则该设计不稳健。若您不确定先验估计值，请使用“先验不确定性”规格反映您的不确定性。

诊断选择 可用选项取决于设计最优性准则。设计最优性准则位于“加速寿命试验计划”红色小三角菜单下。对于 D 最优设计，不存在诊断选择。

关注的时间范围 （仅适用于失效概率最优设计。）指定时间区间，您想要选择在这段时间内估计失效的总体比例。在左框中输入下限值，在右框中输入上限值。若您关注特定时间点，请在这两个框中都输入该值。

关注的概率 （仅适用于分位数估计最优设计。）指定用于估计所需时间的失效比例。例如，若您想要估计 10% 的单元失效的时间，则输入 0.10。

设计选择 指定与试验长度、检查间隔和要试验的单元数相关的值。

试验长度 （仅适用于“连续监控”。）指定单元接受试验的时长。制作设计表时，记录每个单元的失效时间或其是否右删失。

检查时间 （仅适用于“监控周期”。）指定执行检查的时间。制作设计表时，这些时间用于构造**开始时间**列和**结束时间**列。记录在每个间隔中失效的单元数。

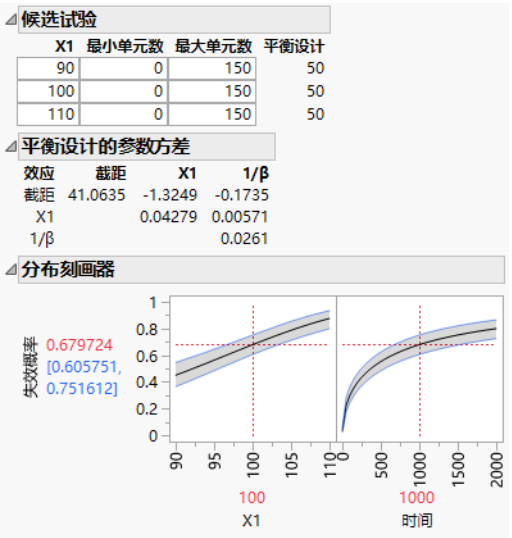
试验单元的数量 实验中的单元数。

- 若是在设计初始实验，则输入您计划试验的单元数。
- 若是在扩充上一个实验，则输入上一个实验中试验的单元数加上下一个实验中的单元数。

检查和更新 ALT 计划规格

指定加速寿命试验计划设计详细信息并点击**继续**后，有三个新的分级显示项会添加到窗口中。

图 22.19 新增的分级显示节点



候选试验 输入加速因子的每个水平所允许的最小单元数和最大单元数。若是扩充以前的实验，对于每个设置，都需包括在 “最小单元数”列中的每个水平上已运行的单元数。

平衡设计的参数方差 提供与平衡设计的加速模型参数估计值的协方差矩阵成比例的矩阵。

用 **X** 表示模型相对于参数 **θ** 的一阶偏导矩阵。用 σ^2 表示误差方差。在一般条件下， θ 的最小二乘估计量是渐近无偏的，渐近协方差矩阵如下所示：

$$Cov(\hat{\theta}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}$$

“平衡设计的参数方差”分级显示项提供 $(\mathbf{X}'\mathbf{X})^{-1}$，其中的导数要进行数值计算。计算假定在“先验均值”分级显示项中指定的值是真实的参数值。

请参见 “非线性设计”一章中第 624 页的 “非线性模型”。

分布刻画器 随着您更改加速因子和时间，支持您基于平衡设计探索失效概率。概率基于以下假设：在 “先验均值”中指定的值是真实的参数值。

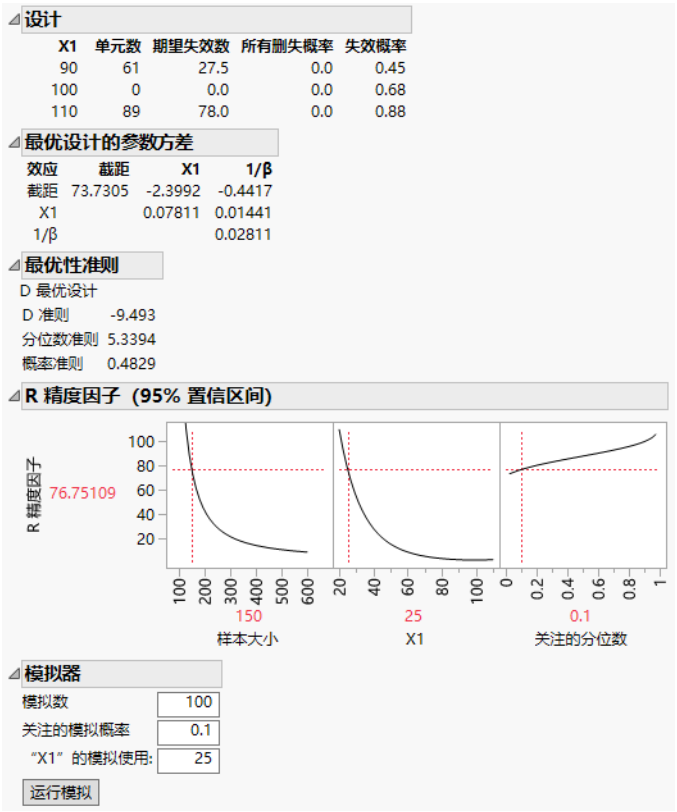
提示：若对 “分布选择”、“先验均值”或 “设计选择”做出更改，请点击**更新刻画器**更新刻画器。

创建和评估最优

点击**制作设计**创建最优 ALT 设计并查看与设计质量相关的结果。

注意：默认情况下，最优设计为 D 最优。要更改最优性准则，请点击“加速寿命试验计划”红色小三角并选择 **ALT 最优性准则**。

图 22.20 “设计”分级显示项



设计 显示要对加速因子的每一组合试验的单元数。对于单个因子，第一列提供该因子的水平。对于两个因子，前两列提供因子水平。

单元数 要对加速水平的每一组合试验的单元数。

期望失效数 设计设置的期望失效数。使用先验模型规格计算期望数。

所有删失概率 在加速水平组合下没有任何试验单元失效的概率。使用先验模型规格计算该概率。

失效概率 设计设置的预期失效概率。失效概率计算为预期失效数除以试验单元数。

注意：期望失效数和删失概率支持您判断您的先验规格是否合理。

最优设计的参数方差 显示与最优设计的加速模型参数估计值的协方差矩阵成比例的矩阵。计算假定在“先验均值”分级显示项中指定的值是真实的参数值。请参见第 598 页的“平衡设计的参数方差”。

注意：将“最优设计的参数方差”矩阵中的值与“平衡设计的参数方差”矩阵中的值进行比较，以确定最优设计减少参数方差的程度。

最优性准则 提供三个最优性准则的值：

D 准则 设计的 D 最优性。请参见第 601 页的“制作 D 最优设计”。

分位数准则 设计的时间 I 最优性。请参见第 601 页的“制作分位数估计值最优”。

概率准则 设计的概率 I 最优性。请参见第 601 页的“制作失效概率最优”。

R 精度因子（95% 置信区间） 用于计算样本大小的影响、因子水平和“R 精度因子”上的相关分位数的刻画器。“R 精度因子”（95% 置信区间）是用于所关注分位数的 95% CI 置信区间的精度的测度。

模拟器 使用模拟器在收集数据前探索加速模型。

模拟数 要运行的模拟试验数。

关注的模拟概率 关注的失效概率。

模拟使用 加速因子的使用条件值。

点击“运行模拟”后，会显示汇总统计量表和模型拟合图。

创建设计表

使用各个按钮制作加速寿命试验设计、试验计划和表格。

制作设计 若对分布选择、先验均值、先验方差矩阵、设计选择或候选试验进行了任何更改，则更新最优设计。

制定试验计划 创建一个数据表，其中每行都对应不同的设计设置。该表显示加速因子设计设置和要在这些设计设置下试验的单元数。

制表 创建可用于记录失效时间数据的表。

- 对于“连续监控”，该表中每一行包含一个要试验的单元以及该单元的设计设置。若某个单元在实验期间失效，则在**失效时间**列中记录该单元的失效时间，在**删失**列中记录 0。若某个单元没有在实验期间失效，则在**失效时间**列中记录最终检查时间，在**删失**列中记录 1。
- 对于“监控周期”，该表中每一行包含一个设计的设置以及时间间隔的组合。时间间隔在**开始时间**列和**结束时间**列中定义，这些指定基于“设计选择”分级显示项中的“检查时间”。对于每个设置和时间间隔，在**失效个数**列中记录失效单元数。

加速寿命试验选项

“加速寿命试验计划”红色小三角菜单包含以下选项：

模拟响应 当您点击**制表**时，会向表中添加模拟响应。模拟响应产生于从选定的分布的随机抽样，该分布使用在“先验均值”分级显示项中制定的参数值。若模拟响应超过指定试验长度，观测将在试验长度值处删失。

ALT 最优性准则

制作 D 最优设计 若 Monte Carlo 球数大于 0，则创建 Bayes D 最优设计。最优性准则是信息矩阵对先验分布取得的行列式的对数期望值。若 Monte Carlo 球数为 0，则设计为局部 D 最优设计。由此断定 D 最优性着重于对系数的精确估计。

制作分位数估计值最优 默认最优性准则。创建一种设计，使得指定使用条件下的指定百分位数的方差最小化。

制作失效概率最优 创建一种设计，使得指定使用条件下关注的时间范围内的失效概率估计值的方差最小化。

高级选项

Monte Carlo 球数 支持您设置在积分中使用的非零半径值的数目。要查找优化给定的最优性准则的非线性设计，JMP 最小化 Fisher 信息矩阵对参数的先验分布取得的行列式的对数积分。对这样的积分进行数值计算。有关如何求积分的详细信息，请参见“非线性设计”一章中第 623 页的“非线性设计选项”。

提示：默认情况下，“Monte Carlo 球数”设置为 4。较高的值会提高数值精度，但也会延长计算时间。

将脚本保存至脚本窗口 为您在“加速寿命试验计划”窗口中指定的设计创建脚本，并将其放入打开的脚本窗口。

“ALT 设计”平台的统计详细信息

ALT 设计的失效分布

在“加速寿命试验设计”平台中，您可以选择对数正态或 Weibull 失效分布。

本节包含每种分布的概率密度函数 (pdf) 和累积分布函数 (cdf) 的参数化。有关 Weibull 分布的详细信息，请参见《可靠性和生存方法》。

对数正态

对数正态分布通常用于失效时间数据范围是 10 的若干次幂的情形。该分布常常被视为许多较小的独立同分布的正数随机变量的乘积。当数据值的对数看起来呈正态分布时，该分布很适用。概率分布函数通常具有明显右偏的特征。

对数正态系列由位置参数 μ 和形状参数 σ 参数化。对数正态 pdf 和 cdf 如下所示，其中对数的底为 *e*：

$$f(x;\mu,\sigma) = \frac{1}{x\sigma}\phi_{nor}\left[\frac{\log(x)-\mu}{\sigma}\right], \quad x>0$$

$$F(x;\mu,\sigma) = \Phi_{nor}\left[\frac{\log(x)-\mu}{\sigma}\right],$$

函数

$$\phi_{nor}(z)= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{z^2}{2}\right)$$

和

$$\Phi_{nor}(z) = \int_{-\infty}^{z}\phi_{nor}(w)dw$$

分别为标准正态分布 (N(0,1)) 的 pdf 和 cdf。

Weibull

Weibull 分布可用于对具有升高或下降危险率的失效时间数据建模。由于该分布可对基于形状参数值的许多不同类型的数据极为灵活地建模，所以它常用在可靠性分析中。

Weibull pdf 和 cdf 通常表示如下：

$$f(x;\alpha,\beta) = \frac{\beta}{\alpha^{\beta}}x^{(\beta-1)}\exp\left[-\left(\frac{x}{\alpha}\right)^{\beta}\right]; \qquad x>0, \alpha>0, \beta>0$$

$$F(x;\alpha,\beta) = 1-\exp\left[-\left(\frac{x}{\alpha}\right)^{\beta}\right]$$

其中，α 是尺度参数，β 是形状参数。当 $\beta = 1$ 时，Weibull 分布简化为指数分布。

Weibull 参数化

在 JMP 中，Weibull 参数化将 σ 定义为尺度参数，将 μ 定义为位置参数。这些与 α 和 β Weibull 参数化的关系如下：

$$\alpha = \exp(\mu)$$

且

$$\beta = \frac{1}{\sigma}$$

使用这些参数，Weibull 分布的 pdf 和 cdf 表示为对数变换的最小极值分布 (SEV)，该分布使用位置尺度分布参数化 （$\mu = \log(\alpha)$， $\sigma = 1/\beta$）：

$$f(x;\mu,\sigma) = \frac{1}{x\sigma}\phi_{sev}\left[\frac{\log(x)-\mu}{\sigma}\right], \quad x > 0, \sigma > 0$$

$$F(x;\mu,\sigma) = \Phi_{sev}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

其中

$$\phi_{sev}(z) = \exp[z - \exp(z)]$$

和

$$\Phi_{sev}(z) = 1 - \exp[-\exp(z)]$$

分别为标准化最小极值（$\mu = 0$、$\sigma = 1$）分布的 pdf 和 cdf。

R 精度因子（95% 置信区间）

R 精度因子是一种测量 t_p（失效时间分布的第 p 分位数）的估计值精度的测度。$\log(t_p)$ 的近似 95% 置信区间为：

$$\log(\hat{t}_p) \pm z_{0.975}\sqrt{\widehat{Var}[\log(\hat{t}_p)]} = \log(\hat{t}_p) \pm \log(\hat{R})$$

生成 t_p 的置信区间；

$$[\hat{t}_p/\hat{R}, \hat{t}_p\hat{R}]$$

其中

$$\hat{R} = \exp\left[z_{0.975}\sqrt{\widehat{Var}[\log(\hat{t}_p)]}\right]$$

要得到 R，使用 $\log(t_p)$ 的方差的大样本近似值：

$$R = \exp z_{0.975}\sqrt{AVar[\log(\hat{t}_p)]}$$

加速寿命试验的最优性准则

考虑加速寿命试验 (ALT) 最多在两个加速因子 x_1 和 x_2（分别具有 J_1 和 J_2 个水平）之间进行。试验计划的目标是确定分配给每个因子水平组合的样本数。

最优性准则的计算依赖于来自单个水平特定信息矩阵的总体 Fisher 信息矩阵

$$I(\theta) = \sum_{i,j} n_{ij} I_{ij}(\theta)$$

以及在适当的情况下包含先验不确定性 S。

D 最优性

D 最优设计 $\mathbf{D}_d$ 是使下列表达式在 n_{jj} 的所有可能值上最小化的设计

$$\log\left|\int_{\theta}[\mathbf{S}^{-1} + \mathbf{I}(\theta)]^{-1}\pi(\theta)d\theta\right|$$

其中 $|\cdot|$ 是行列式运算，$\pi(\theta)$ 是参数的先验分布。该分布通常为多元正态分布。行列式的对数用于数值稳定性。

分位数最优性

分位数最优设计 $\mathbf{D}_q$ 是使以下表达式最小化的设计：

$$\int_{\theta}[\mathbf{c}'\mathbf{I}(\theta)c]\pi(\theta)d\theta$$

其中，向量 **c** 取决于使用条件。

对于单一使用条件

$$\mathbf{c}' = \{1, 0, 0, 0, z_p\}$$

其中，z_p 是失效分布的第 p 分位数。

对于使用条件范围，令

$$\varepsilon_i = \frac{G(x_{1iUh}) - G(x_{1iUl})}{g(x_{iUh}) - g(x_{iUl})}$$

其中，$G(\cdot)$ 是 $g(\cdot)$ 的反导数，x_{iUh} 是为第 i 个因子考虑的最高使用水平，x_{iUl} 是为同一因子考虑的最低使用水平。那么

$$\mathbf{c}' = \{1, \varepsilon_1, \varepsilon_2, \varepsilon_1, \varepsilon_2, z_p\}$$

其中，z_p 是失效分布的第 p 分位数。

失效概率最优性

失效概率最优设计 $\mathbf{D}_f$ 是使以下表达式最小化的设计：

$$\int_{\theta} \left\{ \frac{\frac{1}{\sigma^2}\left[\frac{\phi(z_u(\theta))}{\Phi(z_u(\theta))}\right]^2 c'_{fp} I(\theta) c_{fp}}{-\log(\Phi(z_u(\theta)))} \right\} (\pi(\theta) d\theta)$$

其中，$z_U(\boldsymbol{\theta})$ 是如“分位数最优性”中定义的使用条件下的标准化对数时间。$\mathbf{c}_{fp}$ 类似于“分位数最优性”中定义的 $\mathbf{c}$，只不过最后一项是 $z_U(\boldsymbol{\theta})$。分子中的量基于 $\log\Phi(z_U(\boldsymbol{\theta}))$ 的渐近方差，其中对数变换用于数值稳定性。随后按照原始 $\log\Phi(z_U(\boldsymbol{\theta}))$ 对渐进方差标准化以便保持一致，且越小越好。对于一定范围的使用条件，积分取在对数 $\log\Phi(\cdot)$ 内而不是对渐近方差取积分，以便于计算。作为一个最优性量度，它在性质上仍然是合适的，因为方差之和以总和方差为界限，并且对数函数和 cdf 都是单调函数。因此，上述表达式的最小化通过从上述限定失效概率来最小化失效概率的真实渐近方差。

第 23 章

非线性设计

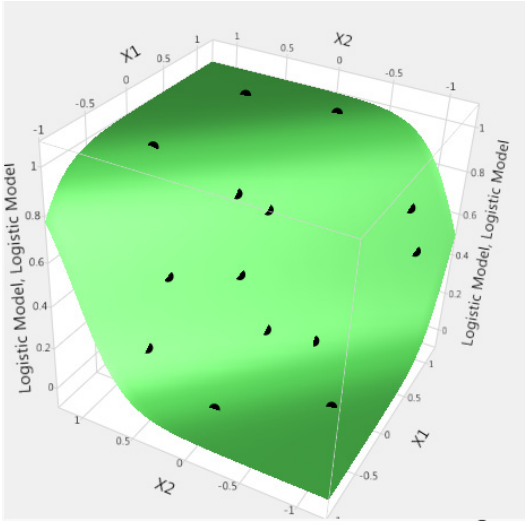
当您的实验目标是拟合未知参数关系为非线性的模型时，请使用非线性设计来将设计点放入对于拟合非线性模型很关键的区域。尽管您可能使用对线性模型最优的正交设计，但是这类设计通常不将设计点放入使拟合的参数估计值不确定性最小（或精度最高）的位置。

非线性设计的效率取决于未知参数的值。这导致了一个循环问题：为了找到最佳设计，您需要预先知道参数。JMP 使用 Bayesian 方法来构造一个非线性设计，它使得在指定的参数值范围内平均效率最高。要正确指定这些范围，您必须对关注的系统有一些了解。

非线性设计与线性模型的设计相比具有以下几个优势：

- 使用精选模型所得到的预测可能在较广的因子设置范围内都很准确。
- 可能对包含复杂曲率和渐近行为的响应曲面建模。

图 23.1 非线性模型的设计点



目录

非线性设计概述 609
非线性设计的示例 610
创建无先验数据的非线性设计 610
使用先验数据扩充设计 614
创建二项响应的设计 618
非线性设计启动窗口 621
“非线性设计”窗口 621
因子 622
参数 622
生成设计 622
设计 623
制表或扩充表 623
非线性设计选项 623
非线性设计的统计详细信息 624
非线性模型 624
最优性准则的径向球积分 624
查找最优设计 625

非线性设计概述

构造设计来使用“非线性设计”平台拟合参数关系为非线性的模型。对于非线性模型，您可以构造最优设计或以最优方式扩充现有数据。非线性设计基于基础过程的描述信息，它与多项式模型的标准设计相比，可获得模型参数和过程行为预测的更准确估计值。有关非线性模型的背景知识，请参见第 624 页的“非线性模型”。

非线性模型设计的效率取决于该设计所要估计的参数的未知值。因此，JMP 使用 Bayesian 方法来构造在很大范围的可能参数值上有效的设计。您可以指定未知参数的范围值和先验分布。先验分布选项包括“均匀”、“正态”、“对数正态”和“指数”。

“非线性设计”平台使用 Bayesian 方法，优化针对您指定的可能参数值的先验分布的设计。Bayesian D 最优性准则是信息矩阵对表示该先验概率分布的参数向量样本求得的行列式的对数期望值。信息矩阵条目取决于设计点的预测方差。少量信息由具有低方差的观测提供，其中响应几乎是确定的。它要求最优设计在高方差点放置一些设计设置。请参见 Gotwalt et al.(2009)。

最优设计原则是在实验的可行域中，最优设计在具有最高预测方差的位置放置点。尽管这可能与直觉不符，但是若备选设计在其他位置放置点，最优设计的设计点的预测方差将更高。对于参数关系为线性的模型，高方差的点通常位于实验区域的顶点。但这对于参数关系为非线性的模型不一定正确。

注意：使用随机起始设计来计算非线性设计。因此，对于相同规格获取的非线性设计通常有所不同。

要使用 “非线性设计”平台，您必须具有现有数据表。该数据表必须包含以下信息：

- 一个响应列。
- 每个因子对应一列。
- 有一列包含公式，公式显示因子与响应之间的关系。该公式必须包含未知参数。

注意：其格式与在用于建模的“非线性”平台中使用的数据表格式相同。

您的表可采用以下两种格式之一：

- 它可能是模板，只包含列信息，没有行。请参见第 610 页的“创建无先验数据的非线性设计”。
- 它可能包含具有预测变量信息的行。在这种情况下，在非线性设计中包含预测变量值。请参见第 614 页的“使用先验数据扩充设计”。

非线性设计的示例

- “创建无先验数据的非线性设计”
- “使用先验数据扩充设计”
- “创建二项响应的设计”

创建无先验数据的非线性设计

本例显示当您尚未收集数据但是有未知参数的猜测值时如何创建设计。在本例中，您对某化学反应的中间产品的相对产量（**观测产出**）建模。相对产量是反应时间和温度的函数。请参见 Box and Draper (1987)。

创建设计

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reaction Kinetics Start.jmp。

   请注意以下事项：

   – 该数据表不包含行，因为没有收集任何数据。

   – 预测变量的列**反应温度**和**反应时间**具有“编码”、“设计角色”和“因子更改”属性。要查看这些属性，请点击“列”面板中的 ✱。它们告知 JMP 在构造设计时如何处理这些预测变量。有关如何保存这些列属性的信息，请参见第 697 页的“列属性”一章。

   – **观测产出**列将包含通过执行实验获取的响应数据。

   – **产出模型**列包含将预测变量与响应**观测产出**关联的公式。点击“列”面板中的 ✙ 可查看公式。该公式对于参数 t1 和 t3 是非线性的。

2. 选择**实验设计** > **特殊目的** > **非线性设计**。
3. 选择**观测产出**并点击 **Y，响应**。
4. 选择**产出模型**并点击 **X，预测变量公式**。
5. 点击**确定**。

   在本例中，使用 “编码”列属性指定了**反应温度**为 510 和 540，**反应时间**为 0.1 和 0.3。或者，您可以直接在 “因子”分级显示项中指定值的合理范围。

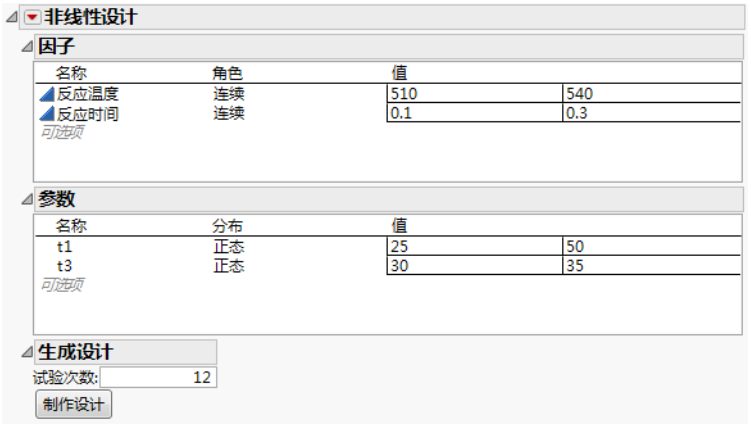
6. 将参数 t1 的值改为 25 和 50，将 t3 的值改为 30 和 35。

   这些新值表示针对实验情形的参数值的合理范围。根据在预测公式定义中指定的初始参数值构造默认值。有关构造公式的信息，请参见 《使用 JMP》。

   请注意，显示在 “分布”下的 t1 和 t3 的先验分布默认设置为 “正态”。

7. 在 “生成设计”面板中将试验次数更改为 12。

图 23.2 已完成的反应动力学实验的分级显示项



8. 点击**制作设计**。
9. 点击**制表**。

图 23.3 设计表

您的设计应与图 23.3 中所示的设计类似。试验顺序可能不同，**反应温度**和**反应时间**值以及为**产出模型**计算出的值可能略有不同。请注意，值显示在**产出模型**列中，因为该列包含模型的公式。还请注意，表包含一个**模型**脚本，您可以使用它来对观测值拟合非线性模型。

现在您已创建了设计表、执行了实验并在**观测产出**列中记录了响应数据。位于 Design Experiment 文件夹中的数据表 Reaction Kinetics.jmp 包含设计的观测结果。

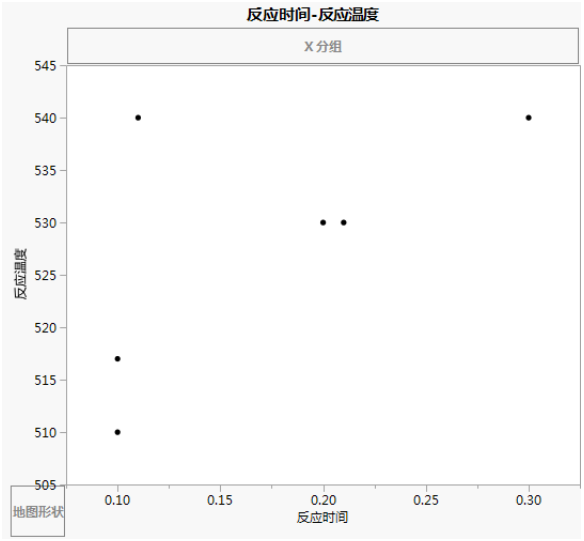
探索设计

在分析结果前，构造一个图来查看设计设置。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reaction Kinetics.jmp。
1. 选择**图形** > **图形生成器**。

2. 将**反应温度**拖放到 Y 区域。
3. 将**反应时间**拖放到 X 区域。
4. 点击图形上的第二个图标以取消选择平滑线。

图 23.4 设计设置

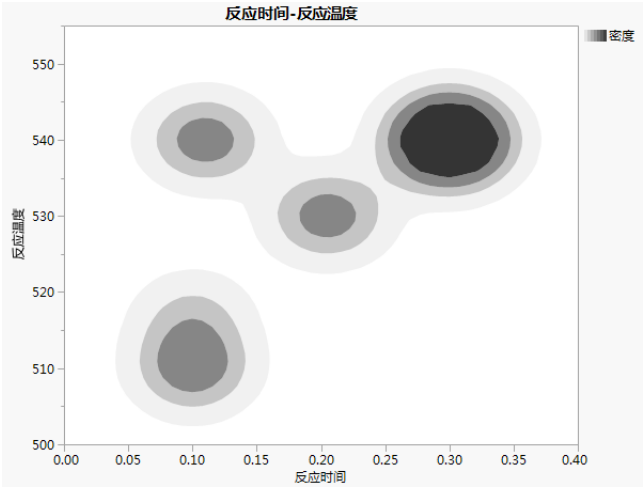


请注意点位于三个区域。没有点位于低温、时间长的区域（图形的右下角）。与正交设计不同，非线性设计不一定在设计区域的角落放置设计点。在本例中，位于低温、时间长的区域的设计点效率低下。

要查看其余三个角落中设计点的密度，请使用 “等高线”工具。

5. 点击 以启用 “等高线”工具。
6. 点击**完成**。

图 23.5 包含密度等高线的设计设置



请注意，有相对较少的点在时间短的高温区。从设计表，您可以看到只有两个这样的点。因为模型和参数规格，最优设计将更多设计点放置在时间长的高温区。

分析结果

您通过可视方式探索您的设计后，接着分析结果。

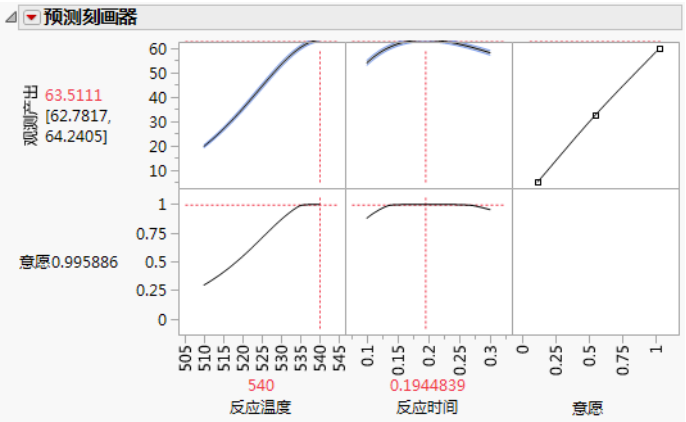
注意：若不执行第 1 步到第 4 步，您可以运行**模型**脚本。

1. 选择**分析 > 专业建模 > 非线性**。
2. 选择**观测产出**并点击 **Y，响应**。
3. 选择**产出模型**并点击 **X，预测变量公式**。

   请注意，模型显示在 “拟合定制公式的选项”面板中。
4. 点击**确定**。
5. 在控制面板中点击**执行**。

   迭代搜索解的过程会持续进行，直到达到停止限值之一。之后，会显示 “解”报表和 “估计值的相关性”报表。
6. 点击 “非线性拟合”红色小三角并选择**刻画器 > 刻画器**。
7. 要最大化产量，请点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 23.6 最大产量对应的时间和温度设置



估计的最大产量为 63.5%，此时反应温度为 540 度 （绝对温度），反应时间为 0.1945 分钟。

使用先验数据扩充设计

本例显示在您有先验数据时如何创建非线性设计。在本例中，数据与一个化学反应有关。您想对可用有机基质的吸收率（速度）作为该基质浓度的函数进行建模。请参见 Meyers (1986)。您已进行一个实验，但是想利用结果获得更精确的参数估计值。

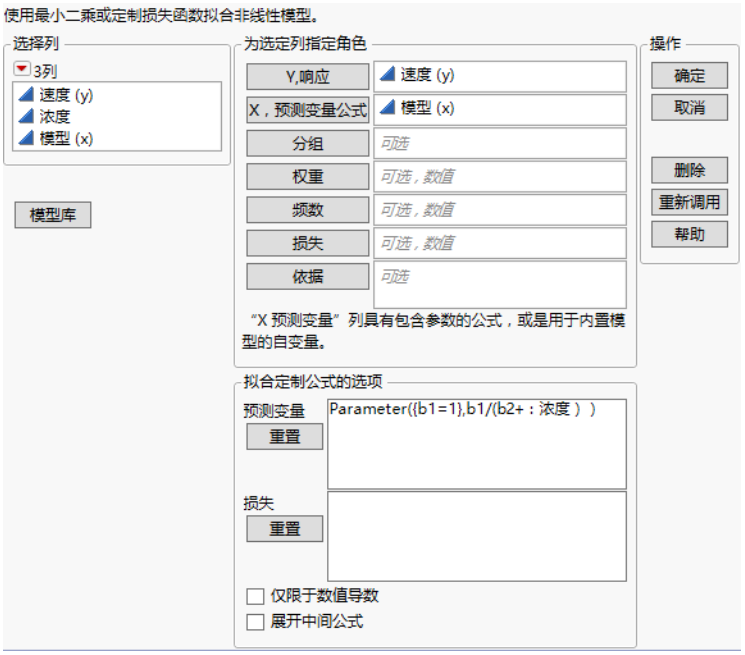
获取先验参数估计值

使用现有实验数据获取更好的参数估计值。

1. 选择**帮助** > **样本数据库**，然后打开 Nonlinear Examples/Chemical Kinetics.jmp。
2. 点击 “列” 面板中**模型 (x)** 旁边的加号。公式编辑器随即打开。
3. 公式编辑器底部中央的**参数**分级显示项显示模型参数的当前值。值 （b1 = 1 和 b2 = 1）是您的初始估计值。它们用于计算数据表中的**模型 (x)** 值。对于下一个实验，您想使用更好的估计值来替换它们。
4. 点击**取消**以关闭公式编辑器窗口。
5. 选择**分析** > **专业建模** > **非线性**。
6. 选择**速度 (y)** 并点击 **Y，响应**。
7. 选择**模型 (x)** 并点击 **X，预测变量公式**。

   请注意，**模型 (x)** 给出的公式显示在 “拟合定制公式的选项” 面板中。

图 23.7 非线性分析启动窗口

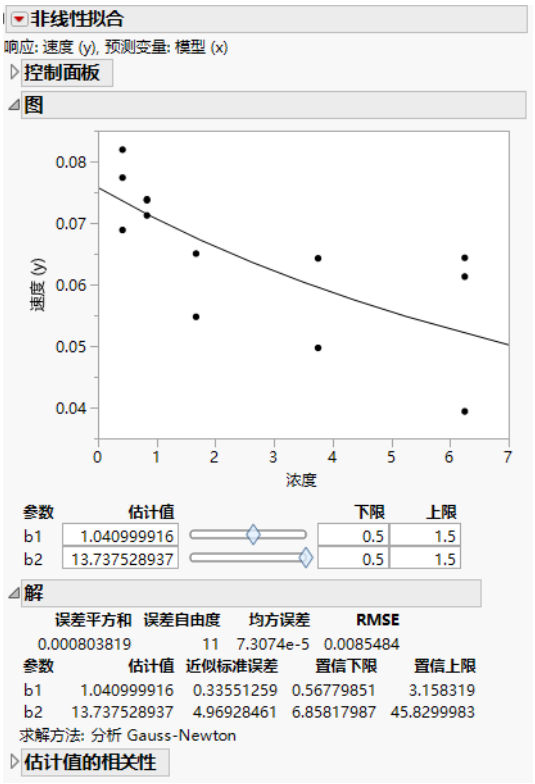


8. 点击**确定**。
9. 在控制面板中，点击**执行**。

   迭代搜索解的过程会持续进行，直到达到停止限值之一。之后，会显示 “解” 报表和 “估计值的相关性” 报表。此外，一个选项显示在 “控制面板” 中，使用它可以向 “解” 报表添加置信限。
10. 在 “控制面板” 中，点击**置信限**。

    参数 b1 和 b2 的置信区间显示在 “解” 报表中。

图 23.8 非线性拟合结果



b1 和 b2 的置信下限和置信上限值定义这些参数的值范围。接着，使用这些区间在扩充的非线性设计中定义先验值的范围。

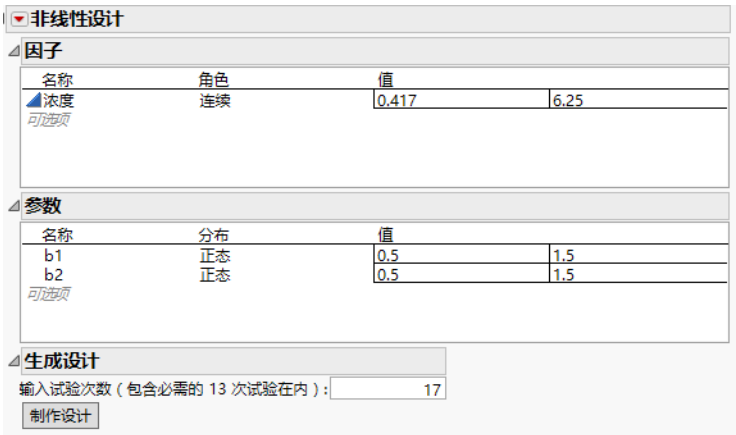
注意：请勿关闭“非线性拟合”报表，因为在后续步骤中需要使用这些结果。

扩充设计

现在，创建一个设计来更精确地估计非线性参数。

1. 激活 Chemical Kinetics.jmp 数据表后，选择**实验设计 > 特殊目的 > 非线性设计**。
2. 选择**速度 (y)** 并点击 **Y，响应**。
3. 选择**模型 (x)** 并点击 **X，预测变量公式**。
4. 点击**确定**。

图 23.9 因子和参数的非线性设计分级显示项

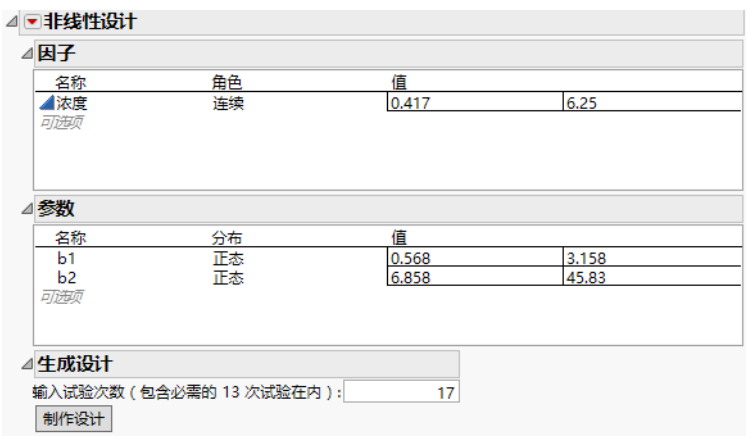


在 Chemical Kinetics.jmp 数据中，**浓度**的值范围为 0.417 到 6.25。因此，这些值最初显示为“因子”分级显示项中的低值和高值。您想更改这些值以包含更大的区间。

5. 点击 0.417 并键入 0.1。点击 6.25 并键入 7。
6. 仍使每个参数的先验分布设置为“正态”。

   参数值的范围反映了您对参数认识的不确定性。您可以指定一个范围，您认定该范围覆盖了 95% 的可能参数值。图 23.8 中所示的“非线性拟合”报表的置信限提供了这样一个范围。使用这些置信限替换“参数”分级显示项中的参数值（舍入到三个小数点）。
7. 在“实验设计 - 非线性设计”窗口中，将这些值输入 b1 和 b2 的参数：
   - b1：0.568 和 3.158
   - b2：6.858 和 45.830

图 23.10 因子和参数的已更新值



8. 在“生成设计”面板中为**试验次数**输入 40。

9. 点击**制作设计**。

   将打开“设计”分级显示项，其中显示原始 13 次试验的**浓度**和**速度** (y) 值以及新增 27 次试验的新**浓度**设置。

10. 点击**制表**。

    这创建了一个新 JMP 设计表，它包含原始 13 次试验设计的设置和结果以及新的 27 次试验的设置。无需创建新数据表，您可以通过点击 “扩充表”而非 “制表”将新试验添加到现有数据表。

新试验反映了**浓度**值的更宽区间以及从原始实验获取的 b1 和 b2 值范围，该范围用于定义先验分布。这两者都会得到更精确的 b1 和 b2 估计值。

创建二项响应的设计

在一些应用程序中，关注的测量值是通过/失败（二项）测量值。在本例中，您将为具有一个设计因子的二项响应构造一个非线性设计。有关非线性设置中的设计（尤其是这个二项情形）的详细信息，请参见 Gotwalt et al.(2009)。

您计划使用具有 Logistic 连结函数的广义线性模型并将 Y 的方差作为权重，

$$Y(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

对作为单个因子 x 的函数的二项响应的成功概率建模。

该模型对未知参数 β_0 和 β_1 是非线性关系。使用非线性设计器来计划实验设计。您的目标是针对 x 对二项响应 Y 的影响建模。

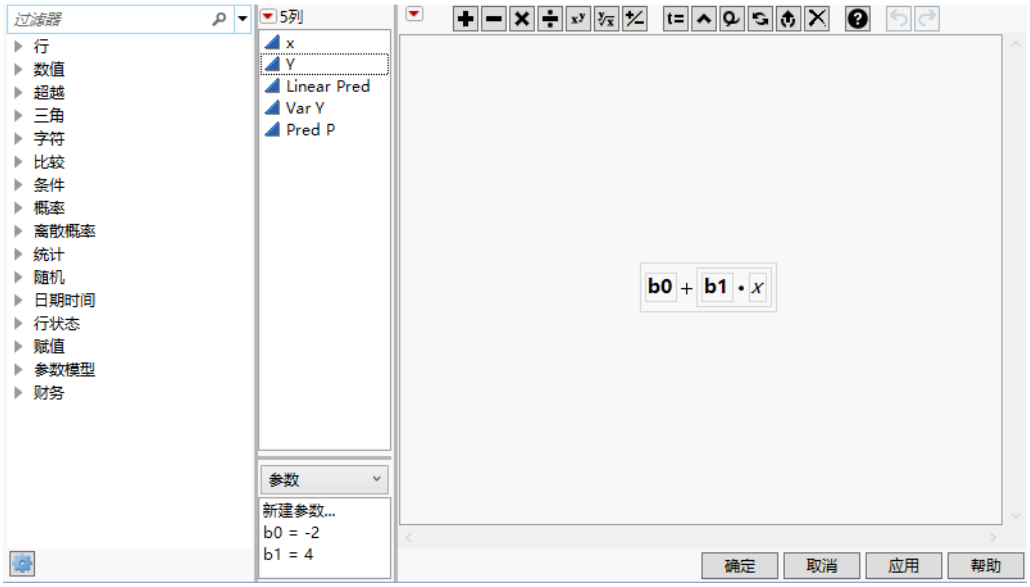
要生成所需的非线性设计，您必须有一个数据表，其中包含预测变量的列、包含表示您拟合的非线性响应模型公式的列以及包含权重公式的列。

用于设计构造的数据表

One Factor Logistic Design.jmp 数据表位于 Design Experiment 文件夹下，它包含以下内容：

- 用于预测变量的列 x。为该列定义的“编码”属性将低值设置为 0，高值设置为 1。
- 用于响应的列 Y。这是用于从您的试验中收集到的响应 (0/1) 的列。
- 包含连结函数线性部分的公式的列 Linear Pred。要查看该公式，请点击“列”面板中 Linear Pred 右侧的加号。模型公式包括参数 **b0** 和 **b1** 的初始估计值。初始参数值是在您定义公式时设置的。这些值显示在公式编辑器窗口中间靠下的公式元素面板中。

图 23.11 带有初始参数估计值的线性预测变量公式

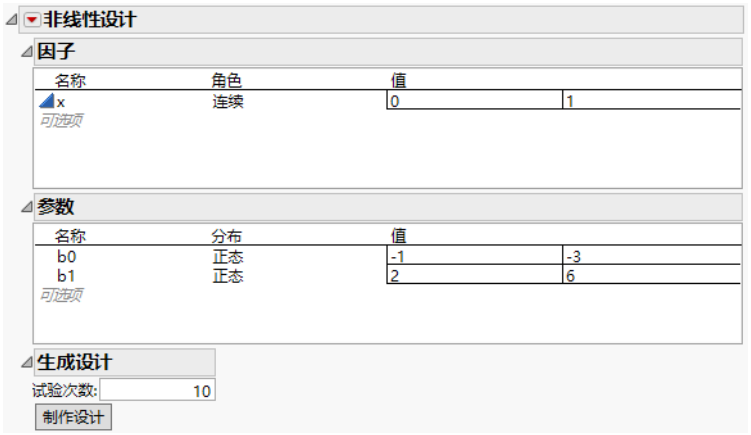


- 列 Var Y，其中包含基于假定的 Logistic 模型的响应的方差公式。即 *p(1-p)*，其中，*p* 是 Logistic 函数。该列用作权重列。
- 包含 Logistic 连结函数的列 Pred P。

创建设计

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/One Factor Logistic Design.jmp。
2. 选择**实验设计** > **特殊目的** > **非线性设计**。
3. 选择 Y 并点击 **Y，响应**。
4. 选择 Linear Pred 并点击 **X，预测变量公式**。
5. 选择 Var Y 并点击**权重**。
6. 点击**确定**。

图 23.12 “非线性设计”窗口

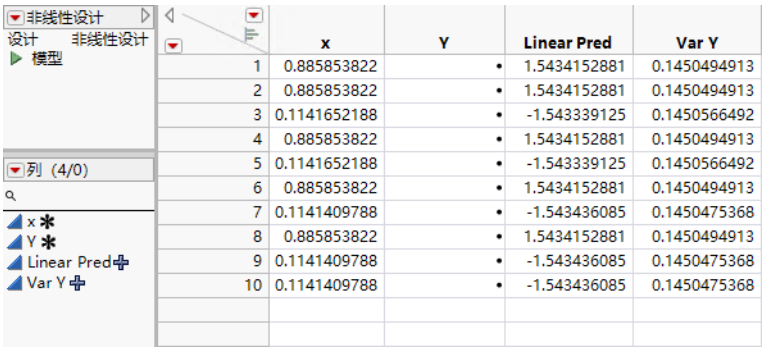


“因子”分级显示项显示带有 0 和 1 这两个指定值的因子 x。“参数”分级显示项显示其先验分布均设置为正态分布的两个模型参数。JMP 根据您对参数值的初始设置计算默认值。您可以更改该分级显示项中的参数的假设分布和范围。让所有设置保持原样。默认试验次数设置为 10。

7. 点击“非线性设计”红色小三角，然后选择**高级选项** > **Monte Carlo 球数**。将球数设置为 0 以获取局部最优设计。点击**确定**。
8. 点击**制作设计**。
9. 点击**扩充表**。

   这会向 One Factor Logistic Design.jmp （图 23.13）添加 10 次试验设计。因为优化算法具有随机成分，因此设计表将有所不同。

图 23.13 扩充 One Factor Logistic Design.jmp



	x	Y	Linear Pred	Var Y
1	0.885853822	•	1.5434152881	0.1450494913
2	0.885853822	•	1.5434152881	0.1450494913
3	0.1141652188	•	-1.543339125	0.1450566492
4	0.885853822	•	1.5434152881	0.1450494913
5	0.1141652188	•	-1.543339125	0.1450566492
6	0.885853822	•	1.5434152881	0.1450494913
7	0.1141409788	•	-1.543436085	0.1450475368
8	0.885853822	•	1.5434152881	0.1450494913
9	0.1141409788	•	-1.543436085	0.1450475368
10	0.1141409788	•	-1.543436085	0.1450475368

保存至数据表的模型是一个非线性模型，用来拟合创建设计时使用的“Linear Pred”。收集数据后，要为 x 对 Y 的影响建模，请使用带有 Logit 连结函数的广义线性模型 (GLM)。

非线性设计启动窗口

要使用 “非线性设计”平台，您必须具有包含以下信息的现有数据表：

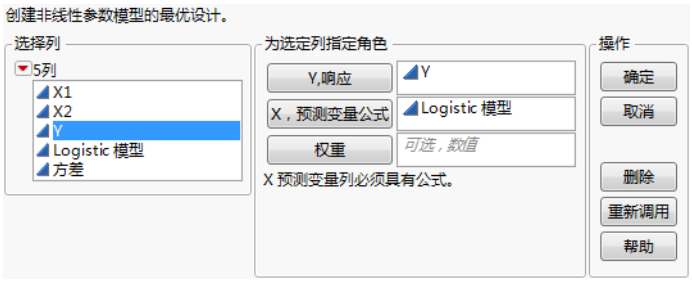
- 一个响应列。
- 每个因子对应一列。
- 有一列包含公式，公式显示因子与响应之间的关系。该公式必须包含未知参数。

有关公式的信息，请参见《使用 JMP》。

该表可包含预测变量和响应的值。若它包含，则您构造的设计扩充表中隐含的设计。可以没有包含缺失预测变量值的行。

激活起始数据表后，选择**实验设计 > 特殊目的 > 非线性设计**。

图 23.14 “非线性”启动窗口



Y，响应 响应值的数值列。

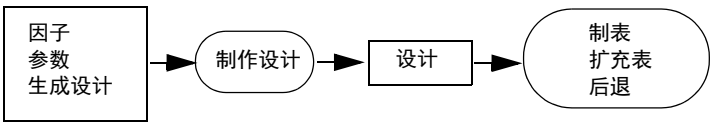
X，预测变量公式 包含非线性模型的公式的数值列。该公式必须包含参数。

权重 （可选）将权重分配给观测的数值列。

“非线性设计”窗口

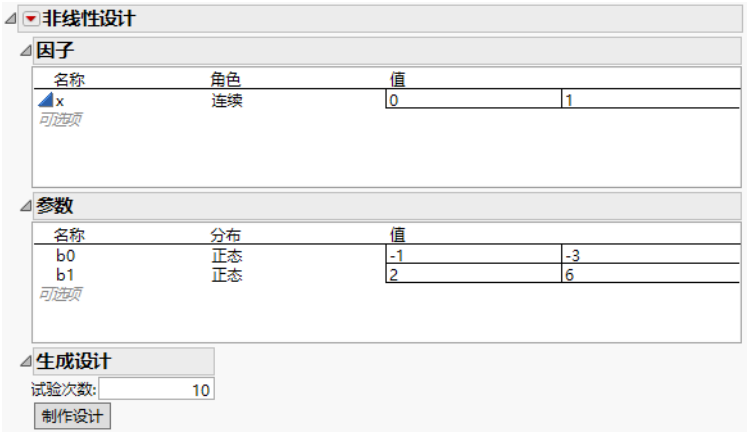
逐步执行设计步骤时，“非线性设计”窗口将相应更新。分级显示项遵循图 23.15 中的流程图，它们被一些按钮分开，点击这些按钮后便有不同的分级显示出现。

图 23.15 非线性设计流程



初始设计窗口显示“因子”、“参数”和“生成设计”分级显示项。

图 23.16 初始设计窗口



因子

在模型公式中使用的列名称将自动插入“因子”分级显示项的“名称”列。将每个因子的角色设置为“连续”。

对于每个因子，初始将“值”设置为 -1 和 1。或者，若您使用“编码”列属性定义了一个值，将使用这些值。您可以在“因子”分级显示项中更改这些值。

参数

在模型公式中使用的参数名称自动插入“参数”分级显示项的“名称”列。

对于每个参数，最初将“值”设置为围绕参数定义中指定的初始值的对称区间。按以下方式获取该区间：将初始值的距离设置为 0，然后以这个宽度围绕初始值构造一个区间。在定义模型参数的先验分布中使用这些值。

注意：根据选择的分布调整参数值，以反映模型参数的不确定性。

“分布”下列出了四个先验分布系列。您为参数指定的值确定了使用哪个先验分布系列的成员。用**低**表示低值，用**高**表示高值。然后按以下方式确定分布：

- 均匀：分布在区间 (**低**, **高**) 上是均匀的。
- 正态、对数正态、指数：分布是**低**为 0.025 分位数且**高**为 0.975 分位数的分布。

生成设计

JMP 提供建议的试验次数，按以下方式确定它们：

- 若您不是在扩充数据表，则试验次数为参数数目的四倍加上二。
- 若正在扩充数据表，则试验次数为数据表中的试验次数加上两倍的参数数目。

注意：若您正在扩充设计，则 JMP 建议的或您指定的试验次数将包括对应于数据表中观测的那些试验。相应调整试验次数。

设计

点击“制作设计”时，JMP 构造该设计并将一个“设计”分级显示项添加到“非线性设计”窗口。在“设计”分级显示项中，您可以查看因子水平设置。

制表或扩充表

“制表”按钮用于创建新设计表。若原始表包含现有试验次数，则新表也包含现有试验次数。**扩充表**按钮用于将新试验次数添加到现有表。

非线性设计选项

“非线性设计”平台中的红色小三角菜单包含以下选项：

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 不可用。

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

加载因子 不可用。

保存约束 不可用。

加载约束 不可用。

模拟响应 将响应值添加到设计表。先选择该选项，然后再点击“制表”。这样在得到的设计表中，响应列包含模拟值。

注意：要设置首选项以始终模拟响应，请选择**文件 > 首选项 > 平台 > 实验设计**，然后选择**模拟响应**。

开始数 设置使用求积分方法创建非线性设计的次数。在创建的设计中，平台选择使最优性准则最大化的设计。

高级选项 > Monte Carlo 样本数 设置在计算最优性准则中使用的每球八面体数。默认值为一个八面体。请参见第 624 页的“最优性准则的径向球积分”。

高级选项 > Monte Carlo 球数 设置在计算最优性准则中使用的非零半径值数。默认值为 2。请参见第 624 页的“最优性准则的径向球积分”。

注意：若将 “Monte Carlo 球数”（半径数）设置为零，则只在计算中使用中心点。这给出了对参数初始值最优的局部设计。对于某些情形，这足够了。

非线性设计的统计详细信息

- “非线性模型”
- “最优性准则的径向球积分”
- “查找最优设计”

非线性模型

用 $\mathbf{Y} = (Y_1, Y_2, ..., Y_n)$ 表示 n 个响应的向量。用以下属性定义非线性模型：

- Y_i 是独立的，是具有指数系列分布的相同分布。
- 每个 Y_i 的期望值（给定预测变量值的一个变量 $\mathbf{x}_i$）是参数 $\boldsymbol{\theta}$ 的非线性函数。该函数表示如下：

  $$E(Y_i(\mathbf{x})) = f(\theta, \mathbf{x}_i)$$

- 按以下方式表示每个 Y_i：

  $$Y_i = f(\theta, \mathbf{x}_i) + \varepsilon_i$$

- 误差向量 $\varepsilon = (\varepsilon_1, \varepsilon_2, ..., \varepsilon_n)'$ 具有均值 $\mathbf{0}$ 和协方差矩阵 $\sigma^2\mathbf{I}$，其中 $\mathbf{I}$ 是 n x n 单位矩阵。

使用 X 表示关于参数 θ 的函数 f 的一阶偏导矩阵。在一般情况下，θ 的最小二乘估计量是渐近无偏倚的，给出的渐近协方差矩阵如下：

$$Cov(\hat{\theta}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}$$

有关此结果的论证，请参见 Wu (1981) 和 Jennrich (1969)。

最优性准则的径向球积分

最优性准则是信息矩阵对先验分布取得的行列式的对数期望值。因此，找到最优非线性设计需要最大化 Fisher 信息矩阵行列式的对数积分（相对于参数的先验分布）。该积分必须进行数值计算。“非线性设计”平台中使用的方法基于 Gotwalt et al. (2009)。

对于先验正态分布，将该积分重新参数化到径向和很多角方向（角方向数等于参数数目减 1）。使用 Radau-Gauss-Laguerre 求积分法（对半径 = 0 求值）计算积分的径向部分。这通过构造一定数目的超八面体并随机轮换它们来做到这点。

若先验分布不是正态的，则将积分重新参数化以便新参数具有正态分布。然后应用径向球积分方法。

注意： 若参数的先验分布不适合求解且过程失败，会将一条消息添加到窗口，指出 Fisher 信息矩阵在参数空间区域是奇异的。若出现这样的情况，请考虑更改先验分布或参数范围。

查找最优设计

用于查找最优设计的方法类似于 Meyer 和 Nachtsheim (1995) 中所述的坐标交换算法。有关如何获取非线性最优设计的详细信息，请参见 Gotwalt et al.(2009)。一般采用以下步骤进行：

- 检验随机设计直到找到非奇异起始设计。
- 进行迭代，其中每次迭代包含遍历一次所有试验次数。
- 对于每次试验，一次优化一个因子。
- 目标函数是 Bayesian D 最优性准则。它是信息矩阵对先验分布取得的行列式的对数期望值。
- 目标函数的变化很小时终止迭代。

第 24 章

平衡不完整区组设计

若不可能将所有处理或因子组合都包括在每个区组中，则可以使用“平衡不完整区组设计(BIBD)”。仅当只能在任一个区组中运行 k 个处理（且 $k < a$）的情况下，才使用 BIBD 试验 b 个区组中的 a 个处理。例如，您可能遇到对可以同时试验的处理数有限制的情形。此时可以构造 BIBD 设计，使得每一处理和每对处理在区组中同时出现相同的次数。这会令设计**平衡**。未能在每个区组中包含所有处理将使设计**不完整**。

使用“平衡不完整区组”平台构造 BIBD。此外，您可以使用该平台构造每个处理在每个区组中都出现的完整区组设计。

图 24.1 大小为 3 的 4 个区组中的 4 个处理的 BIBD

设计

区组			
1	L2	L3	L4
2	L3	L4	L1
3	L4	L1	L2
4	L1	L2	L3

目录

不完整区组设计的示例 629
构建平衡不完整区组设计 631
指定处理、区组和制作设计 631
验证设计并制表 632
平衡不完整区组设计选项 632
平衡不完整区组设计的统计详细信息 633

不完整区组设计的示例

在本例中，您想要在工厂中试验三个产品配方，而且您想要按天分区组。每天最多试验两种配方。此外，您需要在一个工作周内完成您的实验。您需要一个至多有五个区组的设计。

1. 选择**实验设计** > **特殊目的** > **平衡不完整区组设计**。
2. 将 “处理大小” 设置为 3。

   提示：处理大小可指单个因子的水平数或是两个或更多因子的水平的组合。

3. 为**区组大小**选择 2。
4. 为**允许的区组**选择 3。

   您可以选择 3 或 6 个区组。在本例中，每个区组都是一天，您的天数或区组数不得超过 5。试验次数为 6，因为您有 3 个大小为 2 的区组。

注意：您可以输入 “处理名称”、“区组名称” 和 “处理标签” 以定制您的设计表。

5. 点击**制作设计**。

图 24.2 大小为 2 的 3 个区组中的 3 个处理的 BIBD

该设计在三个区组中有六个试验，每个区组有两个处理，如“区组设计”分级显示项所示。有关更多的设计详细信息，请打开“设计”和“关联矩阵”分级显示项。请参见第 632 页的“验证设计并制表”。

最初，您必须选择具有 3 或 6 个区组的设计。您选定了 3 个区组，因为您的工厂有时间限制。不过，假定您可以选择运行实验 6 天。那么您可以更新该设计以使用 6 个区组。

6. 在**允许的区组**菜单中选择 6。
7. 点击**制作设计**以更新该设计。
8. 打开**关联矩阵**分级显示项。

图 24.3 大小为 2 的 6 个区组中的 3 个处理的 BIBD 设计详细信息

关联矩阵

	处理		
区组	L1	L2	L3
1	1	0	1
2	1	0	1
3	0	1	1
4	0	1	1
5	1	1	0
6	1	1	0

配对处理频数

处理	L1	L2	L3
L1	4	2	2
L2		4	2
L3			4

位置频数

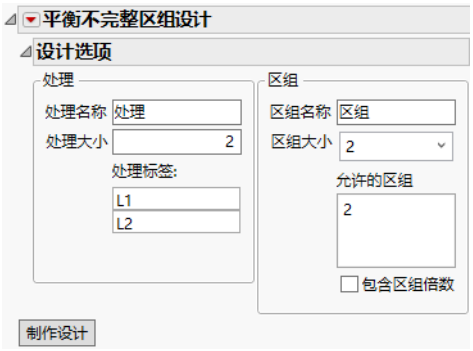
	位置	
处理	1	2
L1	4	0
L2	2	2
L3	0	4

现在设计中包含 6 个区组。关联矩阵显示每个区组中的处理。“配对处理频数”显示每个处理显示了四次（对角线元素），并且每个处理对在设计中出现了两次（非对角线元素）。每个区组都有两个位置。“位置频数”显示处理 L1 始终出现在位置 1 处，L2 分拆在两个位置之间，L3 出现在位置 2 处。

构建平衡不完整区组设计

通过选择**实验设计** > **特殊目的** > **平衡不完整区组设计**构建 “平衡不完整区组设计”。

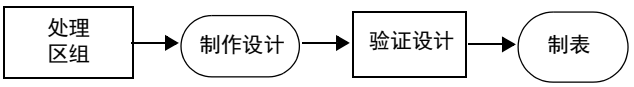
图 24.4 “平衡不完整区组设计”窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“平衡不完整区组设计”窗口随着您逐步执行设计步骤而更新。用于构建 BIBD 的步骤遵循图 24.5 中的流程。

图 24.5 平衡不完整区组设计流程



指定处理、区组和制作设计

以下选项包含在 “设计选项”分级显示项中：

处理名称 处理因子的名称。

处理大小 处理次数。

> **提示：**处理可以是因子水平组合。

处理标签 为处理添加标签。默认标签为 L1 到 L*a*，其中， *a* 是处理总数。

区组名称 分区组因子的名称。

区组大小 可以在任一个区组中运行的处理数量。

> **注意：**“区组大小”的值由处理数量确定。若某个值不在列表内，则是因为 BIBD 平台无法创建包含针对给定处理数量的区组大小的设计。

允许的区组 设计中允许的基于处理数量和区组大小的区组数量。

包含区组倍数 允许选择区组数量等于最初允许的区组大小的倍数 （由您指定）的设计。附加（倍数）区组是原始设计的循环置换。

指定乘数 （在选定 “包含区组倍数”后可用。）您想要应用到最初允许的区组大小的乘数。

制作设计 生成设计。

验证设计并制表

区组设计 以传统设计布局显示 BIBD。

制表 创建 BIBD 数据表。

包括顺序列 在设计表中包括显示每个区组内的试验顺序的列。

设计 通过以下区组布局显示 BIBD：每行都表示一个区组和该区组内的处理。

关联矩阵 显示出现在每个区组中的处理。矩阵中的每行都与设计中某个区组相对应；若处理位于区组内则为 1，否则为 0。

配对处理频数 显示每个处理出现在同一区组内的次数，每个其他处理出现在非对角线元素上。对角线元素是每个处理出现的总次数。

位置频数 显示每个处理出现在每个区组位置的次数。

平衡不完整区组设计选项

“平衡不完整区组设计”平台中的红色小三角菜单包含以下选项：

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击 “制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示“模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一：“正态”、“二项式”或“Poisson”。

注意：并非所有分布都可用于所有设计类型。

– 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在“模型”窗口中做出选择以控制模拟的响应值的分布。当您点击“应用”后，模拟的响应值的公式将保存在名为 **<Y> 模拟**的新列中，其中，Y 是响应的名称。再次点击“应用”将更新 **<Y> 模拟**中的公式和值。

有关其他详细信息，请参见“定制设计”一章中第 101 页的“模拟响应”。

注意: JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见《基本分析》中的“模拟”一章。

将脚本保存至脚本窗口 为您在“平衡不完整区组”窗口中指定的设计创建脚本，并将其放入打开的脚本窗口。

平衡不完整区组设计的统计详细信息

BIBD 中的总观测数由以下等式决定：

$$N = ar = bk$$

其中

a 是处理数量

r 是每个处理出现在设计中的次数

b 是区组数

k 是每个区组中的处理数量

每对处理出现在其中的区组数量由以下公式计算得出：

$$\lambda = \frac{r(k-1)}{a-1}$$

其中，λ 必须是整数。

这些条件是 BIBD 存在的必需条件，不是充分条件。BIBD 平台提供可为其生成 BIBD 的值。

若选定“包含区组倍数”，则会将原始 BIBD 复制指定的倍数。附加（倍数）区组是原始设计的循环置换。副本中处理的位置不同于原件中的位置。

有关 BIBD 的详细信息，请参见 SAS Institute Inc.(2013)。

第 25 章

MSA 设计

创建测量系统分析实验设计

测量系统分析实验（亦称**量具 R&R 研究**）基于完全析因设计。MSA 平台为测量系统分析 (MSA) 实验生成完全析因设计并提供设计诊断测度。有关 MSA 实验的分析，请参见《质量和过程方法》。

图 25.1 MSA 因子设置

名称	MSA 角色	水平数	方差	随机
部件	部分	5	1	是
操作员	运算符	4	1	是
量具	量具	2	1	是

目录

MSA 设计概述 637
MSA 设计示例 637
构建 MSA 设计 639
响应 639
因子 641
MSA 设计 642
MSA 设计诊断 642
MSA 设计表 642
MSA 设计选项 643
“MSA 设计”平台的统计详细信息 644

MSA 设计概述

测量系统分析 (MSA) 设计是因子为随机效应的完全析因设计。研究的重点是估计每个因子引起的响应中的变异。“MSA 设计”平台生成包含特定于 MSA 研究的诊断测度的完全析因设计。

MSA 设计示例

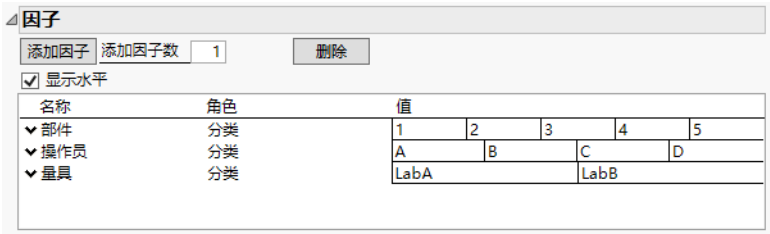
在本例中，您希望研究由于仪器和操作员而导致的测量系统中的变异性。您的实验室里有两个天平和四个有时间参与研究的操作员。您有一套五个标准部件来称重，每个操作员对每个仪器的每个部件进行六次测量。您没有以前的变异性估计值，所以您设计了一个基于方差大小为 1 的实验，该实验的结果是按真实变异性统一尺度的诊断。

1. 选择**实验设计** > **特殊目的** > **MSA 设计**。
2. 将响应 Y 的 “目标”设置为**匹配目标值**。
3. 将 “操作员”的**水平数**设置为 4。
4. 将 “部件”的**水平数**设置为 5。
5. 将 “量具”的**水平数**设置为 2。
6. 将**重复次数**设置为 5 以便为每个设计试验包含 6 个测量值。

图 25.2 因子设置

7. 选择**显示水平**。
8. 编辑图 25.3 中所示的因子 “值”。

图 25.3 因子值



注意：在第 9 步中设置“随机种子”可重现本例所示的设计和诊断。

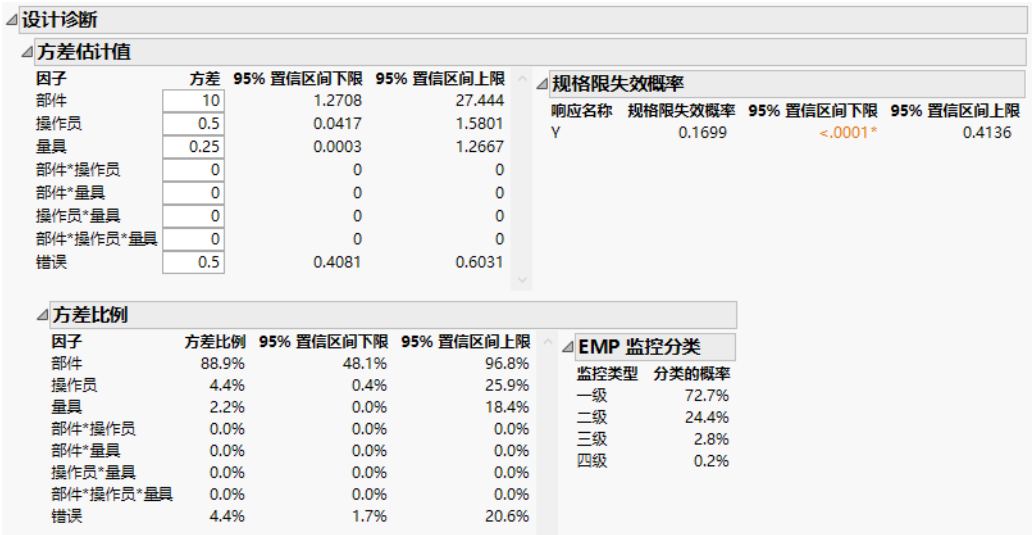
9. （可选）点击“MSA”红色小三角，选择**设置随机种子**，键入 123。
10. 点击**制作设计**。

注意：点击**制表**可生成数据收集表。有关设计表中包含的分析脚本的信息，请参见第 642 页的“MSA 设计表”。

11. 向下滚动到“设计诊断”分级显示项，并点击灰色展开图标将其打开。

    “设计诊断”报表包含假设方差的设置，假设方差随后用于估计置信区间、方差比例和“EMP 监控分类”。这些估计值用于估计 MSA 研究设计的强度。探索不同的方差假设，或使用**后退**按钮对您的设计进行调整。
12. 调整因子的预期“方差”，以探索对您的设计诊断的影响。将“操作员方差”设置为 0.5，将“部件方差”设置为 10，将“量具方差”设置为 0.25。

图 25.4 MSA 设计诊断

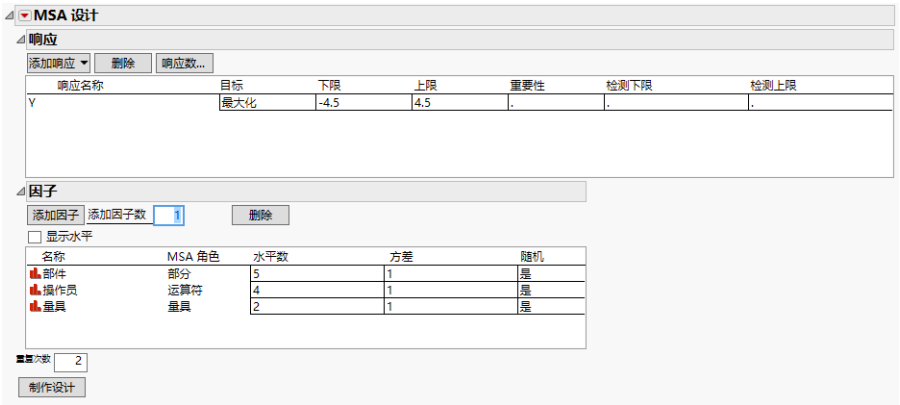


所有诊断都基于根据适当均方的抽样分布计算得出的模拟。基于方差估计假设，本研究得出一级过程的概率为 78%。

构建 MSA 设计

通过选择**实验设计** > **特殊目的** > **MSA 设计**构建 MSA 设计。

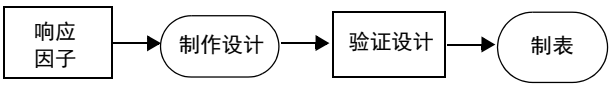
图 25.5 “MSA 设计”窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

逐步执行设计步骤时，“MSA 设计”窗口将相应更新。用于构建 MSA 设计的步骤遵循图 25.6 中的流程。

图 25.6 MSA 设计流程

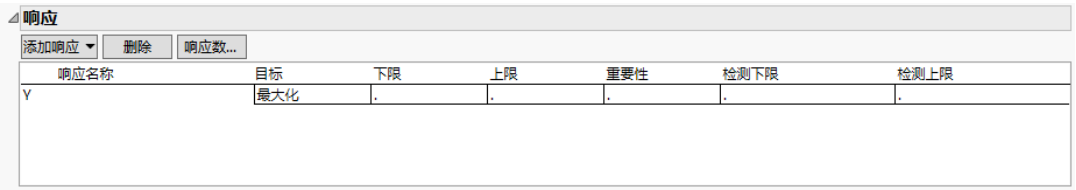


响应

使用“响应”分级显示项来指定一个或多个响应。

提示：填写完“响应”分级显示项后，考虑从红色小三角菜单中选择**保存响应**。该选项会将响应名称、目标、限值和重要性值保存到一个数据表中，以便将来在实验设计平台中重新加载。

图 25.7 “响应”分级显示项



添加响应 输入目标类型为“最大化”、“匹配目标值”、“最小化”、“无”或“函数”的单个响应。若您选择“匹配目标值”，则输入目标值的限值。若您选择“最大化”或“最小化”，输入限值不是必需的；若要使用意愿函数，则输入限值可能很有用。

JMP PRO 函数 （仅限于 JMP Pro。）向设计数据表添加具有 FDE 列特性的多个 Y 列。这些列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

删除 删除选定的响应。

响应数 输入更多响应以便输入的数值是总响应数。若输入了除默认值 Y 之外的响应，则其他每个响应的目标是与最后输入的响应对应的目标。否则，“目标”默认为“匹配目标值”。点击表中的“目标”类型可以更改它。

“响应”分级显示项包含以下列：

响应名称 响应的名称。在添加响应时，会给响应指定默认名称 Y、Y2 等。要更改该名称，双击它，然后输入所需名称。

目标、下限、上限 “目标”告知 JMP 您是要最大化响应、最小化响应、匹配目标值还是没有任何响应目标。JMP 基于这些规格为设计表中的每个响应列分配一个“响应限”列属性。它使用该信息定义每个响应的意愿函数。“刻画器”和“等高线刻画器”使用这些意愿函数找到最优因子设置。有关更多详细信息，请参见《刻画器指南》中的“刻画器”一章和“列属性”附录中第 700 页的 “响应限”。

- “目标”为“最大化”指示最佳值是可能的最大值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“最小化”指示最佳值是可能的最小值。若存在自然的下限或上限，您可以将其指定为“下限”或“上限”。
- “目标”为“匹配目标值”指示最佳值是特定的目标值。默认目标值假定为下限与上限之间的中间值。
- “目标”为“无”指示在最优化方面没有目标。不构造意愿函数。

注意：若目标响应不是下限与上限之间的中间值，您可以在生成设计表后更改该目标值。在数据表中，为响应列打开“列信息”窗口（**列 > 列信息**）并输入所需的目标值。

重要性 当您有几个响应时，指定的“重要性”值用于计算总意愿函数。这些值被视为响应的权重。若只有一个响应，则没有必要指定“重要性”，因为默认情况下它设置为 1。

JMP PRO **检测限值** 该检测上下限值是界限值，超过该界限值则无法测量响应。检测限值另存为列属性。您可以使用这些限值来指定 “广义回归”平台中的删失响应。请参见 《拟合线性模型》中的 “广义回归”一章。

编辑“响应”分级显示项

在 “响应”分级显示项中，请注意以下事项：

- 双击响应可以编辑响应名称。
- 点击目标可以更改它。
- 点击限值或重要性权重可以更改它。
- 对于多个响应，您可能要输入重要性权重的值。

“响应限”列属性

在输入响应时指定的“目标”、“下限”、“上限”和“重要性”将被用来查找最优因子设置。对于每个响应，信息在生成的设计数据表中作为“响应限”列属性保存。JMP 使用该信息定义意愿函数。在“预测刻画器”中使用意愿函数来查找最优因子设置。有关“响应限”列属性及其使用示例的更多详细信息，请参见“列属性”附录中第 700 页的“响应限”。

若您未指定上限和下限，JMP 将使用响应的观测数据范围来定义意愿函数的限值。指定下限和上限使您可以控制意愿函数的规格。有关构造意愿函数的更多详细信息，请参见《刻画器指南》中的“刻画器”一章。

因子

“因子”分级显示项默认显示包含 5 个操作员、3 个部件和 3 个量具的研究。

注意：MSA 设计器要求至少有一个因子具有部件角色。

添加因子 点击以添加一个或多个因子。

删除 删除选定的因子。

添加因子数 添加多个因子。输入要添加的因子数并点击**添加因子**。

显示水平 选择以显示因子水平。若选定，则以下选项可用。

名称 因子名称，可使用菜单来添加水平。

角色 固定为分类因子。MSA 因子默认为分类因子。

值 指定因子水平，您可以通过点击进行编辑。

名称 指定因子的名称，您可以通过点击进行编辑。最初，因子列表中有三个因子：“操作员”、“部件”和 “量具”。您可以在该列表中添加或删除因子。

MSA 角色 指定因子的 “MSA 角色”。点击可从 “操作员”、“部件”、“量具”和 “无”中进行选择。

水平数 指定因子的水平数。

方差 指定因子的先验方差。默认值为 1，结果是按真实变异性统一尺度的诊断。

随机 指定可对设计表中的因子进行随机化处理。

重复次数 指定要包含在设计中的 0 到 10 之间的重复次数。重复次数是除了初始测量之外的测量次数。默认为两次重复，所以每次试验都会生成三个测量值。要指定每次试验生成一个测量值，请将重复次数设置为 0。

MSA 设计

“设计”分级显示项显示 MSA 设计的试验。

MSA 设计诊断

“设计诊断”报表中的所有诊断都基于根据适当均方的抽样分布计算得出的模拟。您可以更改每个模型因子的方差估计值。设计诊断随着方差估计值的更改而更新。这样您就能够探索设计在不同假设模型下测量系统的能力。

MSA 设计方差估计值

“方差估计值”报表支持您为每个 MSA 模型因子指定方差估计值。诊断基于方差估计值并且包含模拟的置信区间 (CI) 界限。同时还提供了响应超出规格的概率的模拟估计值和 CI 界限。通过为响应设置的下限和上限定义规格。

MSA 设计方差比例

“方差比例”报表包含基于假设方差的估计方差比例的诊断。该报表还包括模拟的近似 CI 界限和测量系统 EMP 分类概率。有关 EMP 分类的详细信息，请参见《质量和过程方法》。

MSA 设计模拟结果和选项

“模拟”报表包含用于估计方差比例 CI 界限的模拟方差比例分布的箱线图。模拟基于大量试验。

“模拟结果”红色小三角菜单包含以下选项：

显示统计学 打开每个模型因子的汇总统计量。

显示点 隐藏或显示箱线图中落在须线之外的点。

制作数据表 打开模拟结果数据表。

MSA 设计表

使用各个按钮完成 MSA 设计构造。

制表 构造 MSA 设计数据表。该表包含用于分析的脚本：

EMP 测量系统分析 启动 MSA 分析平台。请参见《质量和过程方法》。

变异性图 启动“变异性量具图”平台。请参见《质量和过程方法》。

实验设计对话框 使用用于生成当前设计表的规格启动“MSA 设计”平台。

运算符工作表 为每个操作员打开一个数据表，用作数据收集的工作表。

后退 让您返回以便对设计规格进行调整。

MSA 设计选项

“MSA 设计”红色小三角菜单包含以下选项：

保存响应 将“响应”面板中的信息保存至新数据表。您可以随后将响应及其关联信息快速加载到多数实验设计窗口中。若您希望重新使用这些响应，该选项很有用。

加载响应 加载使用“保存响应”选项保存的响应。

保存因子 将“因子”面板中的信息保存至新数据表。每个因子的列都包含其水平。其他信息存储为列属性。您可以随后将因子及其关联信息快速加载到多数实验设计窗口中。

注意：可以通过将数据输入空表来创建因子表，但要记住为每个列分配合适的设计角色。要分配设计角色，在数据网格中右击列名，然后选择**列属性 > 设计角色**。在“设计角色”区域中，选择合适的角色。

加载因子 加载使用“保存因子”选项保存的因子。

保存约束 （对于一些平台不可用）将您在“定义因子约束”或“线性约束”分级显示项中定义的因子约束保存到数据表，每个约束对应该表的一列。您可以随后将约束快速加载到多数实验设计窗口中。

在约束表中，前几行包含每个因子的系数。最后一行包含不等式界限。每个约束的列中都包含一个名为**约束状态**的列属性，该属性可将约束标识为“小于”或“大于”约束。请参见“列属性”附录中第 730 页的 “约束状态”。

加载约束 （对于一些平台不可用）加载使用“保存约束”选项保存的因子约束。

设置随机种子 设置随机种子，JMP 用它来控制具有随机成分的操作。这些操作包括以下一项或多项：

- 初始化用于生成设计的搜索算法
- 随机化用于构造设计的“试验顺序”
- 基于随机开始数为设计选择一个初始设计

要重现某个设计，输入生成该设计的随机种子，然后点击**制作设计**。

注意：与设计关联的随机种子包含在保存至设计数据表的 “实验设计对话框”脚本中。

“MSA 设计”平台的统计详细信息

方差和方差比例模拟使用模拟均方项进行计算。均方项是由以相关的期望均方为尺度的卡方分布来模拟的。用矩量法估计量来计算模拟方差，然后用模拟方差来计算方差比例的置信限。

例如，考虑“部件”的方差。模拟计算按以下步骤执行：

1. 模拟部件的均方、涉及部件的每个双因子交互作用的均方，以及涉及部件的高阶交互作用的任何均方。
2. 使用包含 - 排除计算方差：

   “部件”方差 = “部件”均方 - 双因子交互作用的均方 + 三因子交互作用的均方 -…+/- 最高阶交互作用的均方
3. 然后将 “部件”的模拟方差比例计算为 “部件方差 / 总方差”。

注意：只有均方项的分布是直接就知道的。具体而言，均方 = 卡方分布(自由度)*期望均方。根据模拟值近似方差和方差比例的分布。因此，每次更新方差项或每次运行平台时，置信限都可能有所不同。

第 26 章

组正交超饱和设计

创建组正交超饱和设计

组正交超饱和设计是筛选设计的一个特例。这些设计适用于待调查因子数大于可行试验次数情况下的早期工作。超饱和设计是观测数少于模型参数个数的设计。也就是说，设计无法同时估计所有主效应。这让检测显著效应变得很困难。组正交超饱和设计是一类特殊的二水平超饱和设计，该设计具有模型选择所需的属性。

目录

组正交超饱和设计概述 647
组正交超饱和设计的示例 647
构建组正交超饱和设计 648
    指定因子、结构和制作设计 649
    验证设计并制表 649
组正交超饱和设计选项 650
组正交超饱和设计的统计详细信息 651

组正交超饱和设计概述

组正交超饱和设计 (GOSSD) 构造成具有特定结构以便于模型选择。设计因子被分成若干组，以使一组内的因子与同一组内的其他因子相关，而这些因子与其他组内的因子正交。一组包含三个或更多“假”因子。这些假因子未分配到设计因子。它们仅用在设计分析中。假设二阶和更高阶效应可以忽略不计，假因子提供了方差的无偏估计。

构造 GOSSD 的实际考虑因素包括：

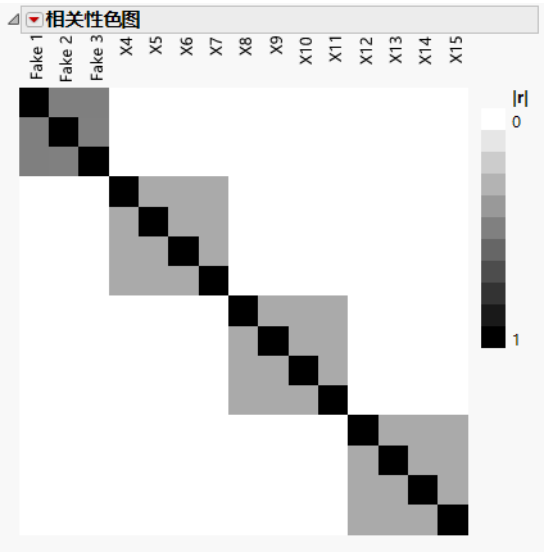
- 较多组好于较少组。
- 被认为活跃的因子应该分别放入不同组。
- 选择因子水平，以使预期系数为正。

组正交超饱和设计的示例

在本例中，您希望设计一个实验，在最多 12 次试验中研究 12 个因子。您想要一个至少包含 16 个参数的设计。截距需要一个参数，至少需要三个假因子。

1. 选择**实验设计 > 特殊目的 > 组正交超饱和 > 组正交超饱和设计**。
2. 选择**试验次数**。
3. 为**试验次数**输入 12，并点击离开文本框。
4. 选择包含 16 个参数 （位于 4 个组中，每组 4 个参数）的 “结构”窗口中的第二个选项。
5. 点击 X1 因子名称并将该名称改为 Fake 1。
6. 对 X2 和 X3 重复该过程，将其分别改为 Fake 2 和 Fake 3。
7. 点击**制作设计**。

图 26.1 GOSSD 相关性色图



设计结构在色图中显而易见。组中因子相互关联。不同组中的因子不相关，这表明它们之间正交。

构建组正交超饱和设计

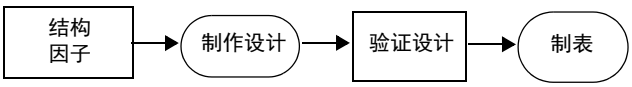
通过选择**实验设计** > **特殊目的** > **组正交超饱和** > **组正交超饱和设计**构建“组正交超饱和设计”。

图 26.2 “组正交超饱和设计”窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

逐步执行设计步骤时，GOSSD 窗口将随之更新。用于构建 GOSSD 的步骤遵循图 26.3 中的流程。

图 26.3 组正交超饱和设计流程



指定因子、结构和制作设计

“设计选项”分级显示项提供以下选项。

提示：提供试验次数或因子数以定义可能的设计结构。

试验次数 定义设计试验次数。该数字需为 2 的倍数，最好是 4 的倍数。

因子数 定义因子数。包括一个用于截距的因子和三到五个假因子。因子数必须是 8 的倍数。

结构 为您的设计提供一个可能结构列表。此列表中的列取决于您设置的是试验次数还是因子数。

- **组数** 设计中的组数。最小值为 2。
- **组大小** 每组中的因子数。
- **试验次数** （在设置因子数时可用）设计中的试验次数。
- **参数个数** （若设置了试验次数，则可用。）设计中的参数个数。这些参数包括一个截距、您的设计因子，而其余的参数则分配给假因子。

因子

- **因子名称** 截距的名称固定。可以重命名其余因子。
- **角色** 支持您将因子定义为连续或分类因子。
- **下限** 支持您设置每个因子的下限。
- **上限** 支持您设置每个因子的上限。

组结构 显示每组中出现的因子。

- **交换** 支持您在组间交换因子。

制作设计 生成设计。

验证设计并制表

设计 显示设计矩阵。

制表 创建 GOSSD 数据表。设计表包括以下脚本：

模型 （仅在填充“响应”列时执行。）运行**实验设计** > **特殊目的** > **组正交超饱和** > **拟合组正交超饱和**平台。

实验设计对话框 重新创建您用于生成设计表的“GOSSD 设计”窗口。

GOSSD 结构 显示设计结构（试验次数、参数个数、组数和组大小）。

多元 为设计因子运行**分析** > **多元**平台。

模型矩阵 包含设计矩阵。

实验设计模拟 （模拟响应时可用。）启动“模拟”窗口。

设计评估

相关性矩阵 显示效应之间的相关性矩阵。

相关性色图 使用强度尺度在图上显示两个效应之间的绝对相关性。包含一个红色三角形菜单，其中包含一个将相关性保存到数据表的选项。

组正交超饱和设计选项

“组正交超饱和设计”平台中的红色小三角菜单包含以下选项：

模拟响应 将响应值和包含模拟公式的列添加至设计表。先选择该选项，然后再点击“制表”。

点击“制表”时，将出现以下情况：

- 一组模拟的响应值添加到每个响应列。
- 对于每个响应，将一个包含模拟模型公式的新列添加到设计表。公式和值基于在设计窗口中指定的模型。
- 将显示“模型”窗口，您可以在此设置模型效应的系数值并指定三种分布之一：“正态”、“二项式”或“Poisson”。

注意：并非所有分布都可用于所有设计类型。

- 名为**实验设计模拟**的脚本保存至设计表。该脚本重新打开“模型”窗口，从而支持您重新模拟值或对模拟的响应分布做出更改。

在“模型”窗口中做出选择以控制模拟的响应值的分布。当您点击“应用”后，模拟的响应值的公式将保存在名为 <Y> **模拟**的新列中，其中，Y 是响应的名称。再次点击“应用”将更新 <Y> **模拟**中的公式和值。

有关其他详细信息，请参见“定制设计”一章中第 101 页的“模拟响应”。

注意： JMP PRO 您可以使用 JMP Pro 模拟功能用模拟响应来执行模拟分析。有关实验设计示例的详细信息，请参见《基本分析》中的“模拟”一章。

将脚本保存至脚本窗口 为您在“组正交超饱和设计”窗口中指定的设计创建脚本，并将其放入打开的脚本窗口。

组正交超饱和设计的统计详细信息

有关设计构造的统计详细信息，请参见 Jones, et.al.(2019)。

第 27 章

“拟合组正交超饱和”平台

分析组正交超饱和设计中的数据

使用“拟合组正交超饱和”平台分析组正交超饱和设计。该分析方法利用组正交超饱和设计的组正交结构。

图 27.1 拟合组正交超饱和结果

目录

“拟合组正交超饱和”平台概述 655
拟合组正交超饱和的示例 655
启动“拟合组正交超饱和”平台 657
“拟合组正交超饱和”报表 658
“拟合组正交超饱和”平台选项 659

“拟合组正交超饱和”平台概述

构造组正交超饱和设计 (GOSSD) 以帮助标识活跃效应。这些设计通过对因子分组，使得在组内因子彼此相关，但在组间因子是正交的。该结构导致两种分析功能：

- 使用分配给第一组的假因子或伪因子来估计方差，此时假定二阶或更高阶效应可忽略不计。
- 组间正交支持两步模型选择过程。

“拟合组正交超饱和”平台执行两步模型选择过程，支持您查看所有因子的回归图，然后为活跃效应提供一个模型，并且可以通过使用“拟合模型”平台来检查该模型。

拟合组正交超饱和的示例

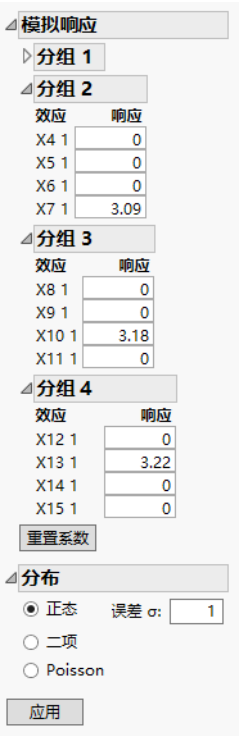
在本例中，您希望设计一个实验，在最多 12 次试验中研究 12 个因子。对于 GOSSD，您想要一个至少包含 16 个参数的设计。截距需要一个参数，假因子至少需要三个参数，设计因子需要 12 个参数。使用模拟响应来拟合模型。

提示：您可以使用不止三个假因子。可使用更多假因子。

生成设计

1. 选择**实验设计 > 特殊目的 > 组正交超饱和 > 组正交超饱和设计**。
2. 选择**试验次数**。
3. 为**试验次数**输入 12，并点击离开文本框。
4. 选择包含 16 个参数 （位于 4 个组中，每组 4 个参数）的 “结构”部分中的第二个选项。
5. 点击 X1 因子名称并将该名称改为 Fake 1。
6. 对 X2 和 X3 重复该过程，将其分别改为 Fake 2 和 Fake 3。
7. 点击 “组正交超饱和设计”红色小三角，然后选择**模拟响应**。
8. 点击**制作设计**。
9. 点击**制表**。

图 27.2 “模拟响应”窗口



“模拟响应”窗口显示用于生成设计表的设置。请注意，“分组 1”分级显示框已关闭。“分组 1”包含“假”因子。这些都设置为 0，因为它们被假定对模型没有任何影响。您的值可能不同于图 27.2 中显示的值。模拟中的每个组都有一个活跃效应。

提示：使用“模拟响应”窗口为不同的模拟输入新值。

分析模拟响应

1. 选择**实验设计 > 特殊目的 > 组正交超饱和 > 拟合组正交超饱和**。
2. 选择**响应**，然后点击 Y。
3. 从 Fake 1 一直选择到 X15，然后点击 X。
4. 点击**确定**。

提示：使用“组正交超饱和”平台生成的设计表包含名为“模型”的脚本。运行该脚本可直接运行该分析。

图 27.3 拟合组正交超饱和结果

您的模拟将 X7、X10 和 X13 设置为具有效应的因子。“拟合 GOSSD”过程将这三个因子标识为显著。此外，还标识了情形 X14。在实践中，您需要确定效应的大小和重要性是否值得进一步关注。

注意：由于生成的是随机结果，所以您的模拟结果可能不同于图 27.3 中所示的结果。您可能会看到一个警报，提示某个组具有一个或多个可能活跃的附加响应。这是超饱和设计的一个限制；您的发现可能不是决定性的。

5. 点击**运行模型**以运行标识的模型。

启动“拟合组正交超饱和”平台

通过选择**实验设计 > 特殊目的 > 组正交超饱和 > 拟合组正交超饱和**启动“拟合组正交超饱和设计”平台。

注意：若您使用“组正交超饱和”平台在 JMP 中创建了设计，则设计表包含名为“模型”的脚本。运行该脚本可直接运行分析。

图 27.4 “拟合组正交超饱和”启动窗口

Y 一个或多个数值响应变量。

X 两水平连续或分类因子。由于该平台使用 GOSSD 的独特结构执行分析，这些因子必须定义 GOSSD。

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“拟合组正交超饱和”报表

“拟合组正交超饱和”报表具有两部分。第一部分报告每个组中活跃效应的估计值，一次拟合一个组。仅显示活跃组。这是分析的 “阶段 1”。在 “阶段 2”中构造组合模型。

图 27.5 阶段 1 结果

项 标识为活跃的主效应。

估计值 针对主效应的 Y 回归拟合的参数估计值。

标准误差 使用组 1 中的假因子计算得出的估计值的标准误差。

t 比 估计值除以其标准误差。

概率 >|t| 使用 t 比和误差自由度 (DF) 计算所得的 *p* 值。

报表的第二部分包含阶段 2 分析的结果。阶段 2 构造用于估计设计结果的模型。

图 27.6 阶段 2 结果

项 标识为活跃的主效应。

估计值 针对主效应的 Y 回归拟合的参数估计值。

标准误差 使用完整数据集拟合指定的模型和截距所得的估计值的标准误差。

t 比 估计值除以其标准误差。

概率 >|t| 使用 t 比和误差自由度 (DF) 计算所得的 *p* 值。

RMSE 均方误差的平方根。

DF 与用于构造 RMSE 的误差估计值关联的自由度。

“拟合组正交超饱和”平台选项

“拟合组正交超饱和设计”平台中的红色小三角菜单包含以下选项：

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

“拟合组正交超饱和”平台的统计详细信息

采用两阶段策略拟合组正交超饱和设计。第一个阶段标识包含活跃效应的组。

- 从假因子中获取 σ 的估计值，然后在进一步分析中弃用假因子。

- 检验每组因子的显著性。

第二阶段考虑活跃组中各个因子的显著性。该阶段使用经指导的子集过程。请参见 Jones and Nachtsheim (2016)。有关完整的统计详细信息，请参见 Jones, et al.(2019)。

第 28 章

样本大小分析器

样本大小与功效计算器

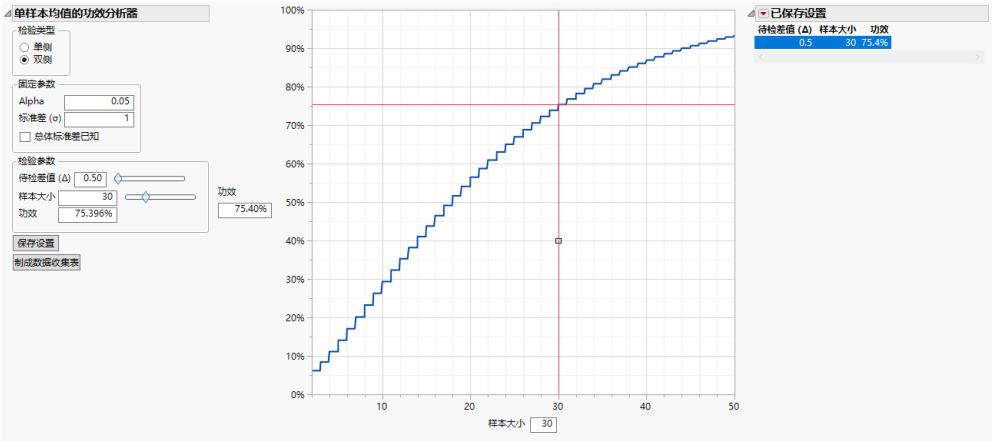
在计划实验或研究时，请使用“样本大小分析器”回答如下问题：

- 我应该检验多少个单元？
- 我是否能在我的处理方式中检测到差值？
- 构造具有指定宽度的区间需要多少个样本？
- 我必须检验多少个单元来估计失效时间？

“样本大小分析器”是一组交互式实用工具，这些实用工具支持您调整研究参数和假设，以探索不同的研究设计。这些工具按照用途来分组：

- 功效用于假设检验。
- 置信区间用于区间估计。
- 可靠性验证。

图 28.1 单样本均值的样本大小分析器



目录

“样本大小分析器”平台概述 663
样本大小分析器示例 663
启动“样本大小分析器” 664
假设检验功效分析器 664
单样本均值的功效 665
单样本比例的功效 667
单样本方差的功效 669
单样本等价性的功效 670
两个独立样本均值的功效 672
两个独立样本比例的功效 675
两个独立样本方差的功效 677
两个独立样本等价性的功效 679
方差分析的功效 681
置信区间计算器 682
单样本均值的误差幅度 683
单样本比例的误差幅度 685
单样本方差的误差幅度 686
两个独立样本均值的误差幅度 687
两个独立样本比例的误差幅度 689
两个独立样本方差的误差幅度 691
“可靠性验证”计算器 693
参数可靠性验证 693
非参数可靠性验证 694
样本大小分析器中的已保存设置 695

“样本大小分析器”平台概述

“样本大小分析器”平台用于样本大小与功效计算。“样本大小分析器”支持您评估和比较研究假设对研究所需的观测数（试验、实验单元或样本）的影响。提供用于假设检验、区间估计和可靠性验证的分析器。

有关功效和样本大小的详细信息，请参见 Barker (2011)。

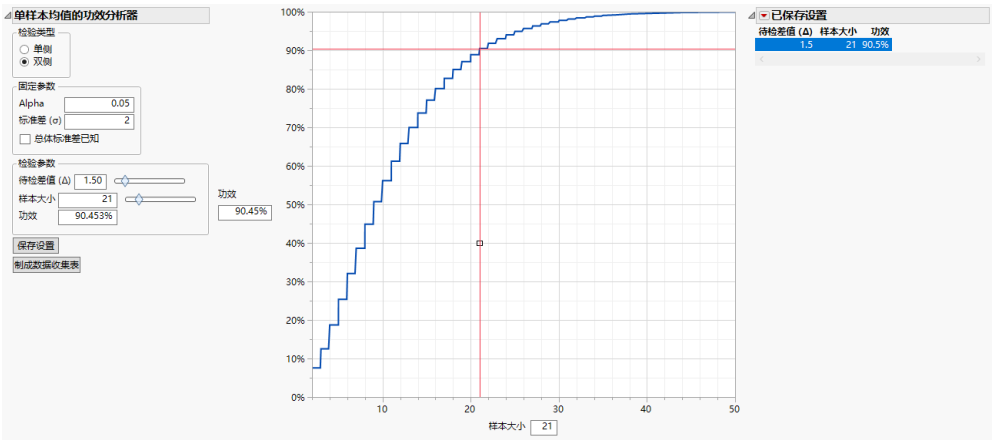
样本大小分析器示例

您可以使用 “样本大小分析器”来帮助确定新材料研究中待检验的单元数。假定您想验证您公司研发的新织物与当前材料相比可燃性有改进。以前的检验表明该织物的点燃时间的标准差为 2 秒。

使用 “单样本均值的功效分析器”可计算您需要检验的织物样本数。您想设计一个功效为 90% 的实验在显著性水平 $\alpha = 0.05$ 下检测 1.5 秒的差值。

1. 选择**实验设计** > **样本大小分析器** > **功效** > **单样本均值的功效**。
2. 保持 **Alpha** 设置为 0.05。
3. 为**标准差**输入 2。
4. 为**待检差值**输入 1.5。
5. 为**功效**输入 0.9。
6. 点击 “保存设置”可保存这些设置。
7. 使用滑块和 / 或文本框来为您的研究探索其他样本大小和功效可能。

图 28.2 单样本均值功效分析器



显著性水平为 0.05 时，要想有 90% 机会检测到显著的点燃时间差值 1.5 秒，需要 21 个织物样本。

启动“样本大小分析器”

通过选择**实验设计 > 样本大小分析器**启动“样本大小分析器”。可以从多个计算器中进行选择。

对于假设检验，选择**实验设计 > 样本大小分析器 > 功效**：

- 第 665 页的“单样本均值的功效”
- 第 667 页的“单样本比例的功效”
- 第 669 页的“单样本方差的功效”
- 第 670 页的“单样本等价性的功效”
- 第 672 页的“两个独立样本均值的功效”
- 第 672 页的“两个独立样本均值的功效”
- 第 677 页的“两个独立样本方差的功效”
- 第 679 页的“两个独立样本等价性的功效”
- 第 681 页的“方差分析的功效”

对于区间估计，选择**实验设计 > 样本大小分析器 > 置信区间**：

- 第 683 页的“单样本均值的误差幅度”
- 第 685 页的“单样本比例的误差幅度”
- 第 686 页的“单样本方差的误差幅度”
- 第 687 页的“两个独立样本均值的误差幅度”
- 第 689 页的“两个独立样本比例的误差幅度”
- 第 691 页的“两个独立样本方差的误差幅度”

对于可靠性验证，选择**实验设计 > 样本大小分析器 > 可靠性**：

- 第 693 页的“参数可靠性验证”
- 第 694 页的“非参数可靠性验证”

假设检验功效分析器

使用功效分析器计算假设检验的样本大小。探索样本大小对假设检验功效的影响，以检测指定的差值，其中的差值可能来自一个常数（单样本情况）或来自两组之间（双样本情况）。对于单样本情况，假设是关于单个参数（均值、比例或方差）。对于双样本情况，假设是关于两个组的比较。对于两个以上的组，您可以使用“方差分析功效分析器”，请参见“方差分析的功效”。

对于假设检验，选择**实验设计 > 样本大小分析器 > 功效**：

- 第 665 页的“单样本均值的功效”
- 第 667 页的“单样本比例的功效”

- 第 669 页的“单样本方差的功效”
- 第 670 页的“单样本等价性的功效”
- 第 672 页的“两个独立样本均值的功效”
- 第 672 页的“两个独立样本均值的功效”
- 第 677 页的“两个独立样本方差的功效”
- 第 679 页的“两个独立样本等价性的功效”
- 第 681 页的“方差分析的功效”

单样本均值的功效

使用“单样本均值的功效分析器”可确定关于一个均值的假设检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 单样本均值的功效**。探索在变异性假设、样本大小、功效、显著性和假设的待检差值之间如何权衡。样本大小与功效与以下假设检验相关联：

H_0: $\mu = \mu_0$

与双侧备择假设：

H_a: $\mu \neq \mu_0$

或与单侧备择假设：

H_a: $\mu < \mu_0$ 或者 H_a: $\mu > \mu_0$

其中 μ 是真均值，μ_0 是原假设均值或参考值。均值偏离 μ_0 的差值 Δ，是您认为重要而需要检测的量。对于相同的显著性水平和功效，检测小差值与检测大差值相比，前者需要更大的样本大小。假定关注的总体服从均值为 μ、标准差为 σ 的正态分布。

“单样本均值的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

检验类型 指定单侧或双侧假设检验。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的**显著性水平**。默认 alpha 水平为 0.05。

标准差 (σ) 指定假定的标准差。

提示：使用标准差 1 估计检测差值 （用标准差单位数表示）所需的样本大小。

总体标准差已知 指定使用基于已知总体标准差的计算。

检验参数 相互关联并在进行更改时随之更新的参数。

待检差值 (Δ) 指定真均值和假设均值（或参考均值）之间的最小差值，您希望能称为在统计上显著。

样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本功效的统计详细信息

当 σ 未知时，单样本均值计算基于传统 t 检验。对于 σ 已知的情况，计算使用 z 检验。对于 σ 未知的情况，根据备择假设计算功效。

对于单侧、上侧备择假设：

$$\Pr(\mu > \mu_0 \mid \mu = \mu_0 + \delta) = 1 - T\left(t_{1-\alpha,\, n-1}; n-1, \frac{\delta}{s/(\sqrt{n})}\right)$$

对于单侧、下侧备择假设：

$$\Pr(\mu < \mu_0 \mid \mu = \mu_0 - \delta) = T\left(-t_{1-\alpha,\, n-1}; n-1, \frac{\delta}{s/(\sqrt{n})}\right)$$

对于双侧备择假设：

$$\Pr(\mu \neq \mu_0 \mid \mu = \mu_0 \pm \delta) = 1 - T\left(t_{1-\alpha/2,\, n-1}; n-1, \frac{\delta}{s/(\sqrt{n})}\right) + T\left(-t_{1-\alpha/2,\, n-1}; n-1, \frac{\delta}{s/(\sqrt{n})}\right)$$

其中：

α 是显著性水平。

n 是样本大小。

s 是标准差

δ 是待检差值。

$t_{1-\alpha,\nu}$ 是自由度为 ν 的中心 t 分布的第 $(1 - \alpha)$ 分位数

$T(t; \nu, \lambda)$ 是自由度为 ν 且非中心参数为 λ 的非中心 t 分布的累积分布函数。

若 σ 已知，则在上述用于功效计算的公式中使用 z 分布和 σ。由于 δ 和 n 的闭合形式的解不存在，则使用数值例程对其求解。

单样本比例的功效

使用“单样本比例的功效分析器”可确定关于一个比例的假设检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 单样本比例的功效**。探索在样本大小、功效、显著性和假设的待检差值之间如何权衡。样本大小与功效与以下假设检验相关联：

$H_0: p = p_0$

与双侧备择假设：

$H_a: p \neq p_0$

或与单侧备择假设：

$H_a: p < p_0$ 或者 $H_a: p > p_0$

其中 p 是总体比例，p_0 是原假设比例。

“单样本比例的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

检验类型 指定单侧或双侧假设检验。

模型类型

精确检验 指定基于 Clopper-Pearson 方法的计算。

正态近似 指定基于正态近似方法的计算。

提示：因为二项分布是离散的，对于小样本或接近 0 或 1 的比例，实际检验大小可能与指定的 Alpha 水平有很大不同。为了确保 alpha 水平等于或大于指定的水平，请使用 “精确检验”。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

检验参数 相互关联并在进行更改时随之更新的参数。

假定比例 (p0) 指定您为研究预期或假设的比例，即原假设值。

备择比例 (pA) 指定要对照进行检验的比例，即备择假设值。

样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至 “已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的 “样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本比例分析器的统计详细信息

对于精确检验，基于备择假设的形式计算功效。

对于单侧、上侧备择假设：

$$\Pr(p > p_0 | p = p_a) = \sum_{x = x_{1-\alpha, n, p_o}}^{n} \binom{n}{x} p_a^x (1-p_a)^{n-x}$$

对于单侧、下侧备择假设：

$$\Pr(p < p_0 | p = p_a) = \sum_{x = x_0}^{x = x_{\alpha, n, p_o - 1}} \binom{n}{x} p_a^x (1-p_a)^{n-x}$$

对于双侧备择假设：

$$\Pr(p \neq p_0 | p = p_a) = \sum_{x = x_{1-\alpha/2, n, p_o}}^{n} \binom{n}{x} p_a^x (1-p_a)^{n-x} + \sum_{x = x_0}^{x = x_{\alpha/2, n, p_o - 1}} \binom{n}{x} p_a^x (1-p_a)^{n-x}$$

其中，$x_{q,n,p}$ 是有 n 次试验且概率为 p 的二项分布的第 q 分位数。

对于正态近似，基于备择假设的形式计算功效。

对于单侧、上侧备择假设：

$$\Pr(p > p_0 | p = p_a) = 1 - \Phi\left(z_{1-\alpha} \sqrt{\left(\frac{p_0(1-p_0)}{p_a(1-p_a)} \right)} - \frac{(p_a - p_0)}{\sqrt{\frac{p_a(1-p_a)}{n}}} \right)$$

对于单侧、下侧备择假设：

$$\Pr(p < p_0 | p = p_a) = \Phi\left(-z_{1-\alpha} \sqrt{\left(\frac{p_0(1-p_0)}{p_a(1-p_a)} \right)} - \frac{(p_a - p_0)}{\sqrt{\frac{p_a(1-p_a)}{n}}} \right)$$

对于双侧备择假设：

$$\Pr(p \neq p_0 \mid p = p_a) =$$

$$1-\Phi\left(z_{1-\alpha/2}\sqrt{\left(\frac{p_0(1-p_0)}{p_a(1-p_a)}\right)}-\frac{(p_a-p_0)}{\sqrt{\frac{p_a(1-p_a)}{n}}}\right)+\Phi\left(-z_{1-\alpha/2}\sqrt{\left(\frac{p_0(1-p_0)}{p_a(1-p_a)}\right)}-\frac{(p_a-p_0)}{\sqrt{\frac{p_a(1-p_a)}{n}}}\right)$$

单样本方差的功效

使用“单样本方差的功效分析器”可确定关于一个方差的假设检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 单样本方差的功效**。探索在样本大小、功效、显著性和假设的待检差值（定义为原假设值与备择假设值之间的比率）之间如何权衡。样本大小与功效与以下假设检验相关联：

H_0: $\sigma = \sigma_0$

与双侧备择假设：

H_a: $\sigma \neq \sigma_0$

或与单侧备择假设：

H_a: $\sigma < \sigma_0$ 或者 H_a: $\sigma > \sigma_0$

其中，σ 是真实方差，σ_0 是原方差或参考值。标准差偏离 σ_0 的差值，是您认为重要、需要检测的量（基于一组样本检测）。该差值表示为 σ_0/σ_a 的比值，或是备择假设下的原方差与假设方差的比值。对于相同的显著性水平和功效，检测小方差差值与检测大差值相比，前者需要更大的样本大小。假定关注的总体服从均值为 μ、标准差为 σ 的正态分布。

“单样本方差的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

检验类型 指定单侧或双侧假设检验。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

检验参数 相互关联并在进行更改时随之更新的参数。

方差比（原/备择） 指定原假设下的方差（参考方差）与备择假设下的方差（预期方差）之间的比值。

样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本方差分析器的统计详细信息

用于检验单样本组方差的功效计算基于 χ^2 检验。计算基于备择假设的形式。

对于单侧、上侧备择假设：

$$\Pr(\sigma > \sigma_0 \mid \sigma = \sigma_0/\rho) = 1 - \chi^2(\rho x_{1-\alpha,\, n-1}; n-1)$$

对于单侧、下侧备择假设：

$$\Pr(\sigma < \sigma_0 \mid \sigma = \sigma_0/\rho) = \chi^2(\rho x_{\alpha,\, n-1}; n-1)$$

对于双侧备择假设：

$$\Pr(\sigma \neq \sigma_0 \mid \sigma = \sigma_0/\rho) = 1 - \chi^2(\rho x_{1-\alpha/2,\, n-1}; n-1) + \chi^2(\rho x_{\alpha/2,\, n-1}; n-1)$$

其中：

α 是显著性水平。

n 是样本大小。

$\rho = \sigma_a/\sigma_0$

$x_{1-\alpha,\nu}$ 是自由度为 ν 的中心 χ^2 分布的第 $(1 - \alpha)$ 分位数

$\chi^2(x, \nu)$ 是自由度为 ν 的中心 χ^2 分布的累积分布函数。

单样本等价性的功效

使用“单样本等价性的功效分析器”可确定关于一个均值的等价性检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 单样本等价性的功效**。探索在变异性假设、样本大小、功效、显著性和等价性范围之间如何权衡。样本大小与功效与以下假设检验相关联：

$$H_0: \mu - \mu_0 \geq \delta_M \text{ 或者 } H_0: \mu - \mu_0 \leq \delta_m$$

对比备择假设：

$$H_a: \delta_m < \mu - \mu_0 < \delta_M$$

其中，μ 是真实均值，μ_0 是参考值，(δ_m, δ_M) 是等价性范围。对于相同的显著性水平和功效，检测小差值与检测大差值相比，前者需要更大的样本大小。假定关注的总体服从均值为 μ、标准差为 σ 的正态分布。

“单样本等价性的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

等价性范围

最大差值 指定最大值，高于该值的均值被视为不同于参考均值。

最小差值 指定最小值，低于该值的均值被视为不同于参考均值。

注意：通常，等价性范围是对称的。

固定参数

Alpha I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

标准差 (σ) 假定的标准差。

总体标准差已知 指定计算基于已知总体标准差。

检验参数 相互关联并在进行更改时随之更新的参数。

均值差值 指定真实均值与假设或参考均值之间的差值，以便将两个均值视为相等。

样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本等价性分析器的统计详细信息

在一个样本组中检验等价性的功效计算基于 Chow et al. (2008) 中所述的方法。

若 σ 未知，功效 (1-β) 计算如下：

$$1-\beta = T\left(-t_{1-\alpha,\,n-1};n-1,\ \frac{\delta-\delta_M}{s/(\sqrt{n})}\right) - T\left(t_{1-\alpha,\,n-1};n-1,\ \frac{\delta-\delta_m}{s/(\sqrt{n})}\right)$$

其中：

α 是显著性水平。

n 是样本大小。

s 是标准差

δ 是待检差值。

(δ_m, δ_M) 是等价性范围

$t_{1-\alpha,\nu}$ 是自由度为 ν 的中心 t 分布的第 $(1 - \alpha)$ 分位数

$T(t; \nu, \lambda)$ 是自由度为 ν 且非中心参数为 λ 的非中心 t 分布的累积分布函数。

若 σ 未知，则功效 $(1-\beta)$ 计算如下：

$$1 - \beta = \Phi\left(\frac{\delta - \delta_M}{\sigma/\sqrt{n}} - z_{1-\alpha}\right) + \Phi\left(\frac{\delta - \delta_m}{\sigma/\sqrt{n}} - z_{1-\alpha}\right) - 1$$

两个独立样本均值的功效

使用 “两个独立样本均值的功效分析器”可确定关于两个组的均值的假设检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 两个独立样本均值的功效**。探索在样本大小、功效、显著性和假设的待检差值之间如何权衡。样本大小与功效与以下假设检验相关联：

$$H_0: \mu_1 - \mu_2 = 0$$

与双侧备择假设：

$$H_a: \mu_1 - \mu_2 \neq 0$$

或与单侧备择假设：

$$H_a: \mu_1 - \mu_2 < 0 \text{ 或者 } H_a: \mu_1 - \mu_2 > 0$$

其中 μ_1 和 μ_2 是两个总体的真实均值。假定相关总体服从正态分布，且您要检测两个均值之间的差值 δ。

“两个独立样本均值的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

检验类型 指定单侧或双侧假设检验。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

组 1 标准差 (σ_1) 为其中一个组“组 1”指定假定的标准差。误差标准差的估计值可以是先前的模型拟合的均方根误差 (RMSE)。

组 2 标准差 (σ_2) 为第二个组“组 2”指定假定的标准差。误差标准差的估计值可以是先前的模型拟合的均方根误差 (RMSE)。

总体标准差已知 指定计算基于已知总体标准差。

检验参数 待检差值 (Δ) 指定组均值之间的最小差值，您希望能称为在统计上显著。

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。功效曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非其中一个组已被锁定。只能锁定一个组样本大小。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

两个独立均值的统计详细信息

检验两个样本组均值差值的功效计算基于传统的 t 检验；若 σ_1 和 σ_2 已知，则基于 *z* 检验。

对于 σ 未知的情况，假定 $\sigma_1=\sigma_2=\sigma$，则功效计算基于备择假设的形式。对于单侧、上侧备择假设：

$$\Pr(\mu_1 > \mu_2 \mid \mu_1 = \mu_2 + \delta) = 1 - T\left(t_{1-\alpha,\, n_1+n_2-2};\, n_1+n_2-2,\, \frac{\delta}{s\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}}\right)$$

对于单侧、下侧备择假设：

$$\Pr(\mu_1 < \mu_2 \mid \mu_1 = \mu_2 - \delta = T\left(-t_{1-\alpha,\, n_1+n_2-2};\, n_1+n_2-2,\, \frac{\delta}{s\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}}\right)$$

对于双侧备择假设：

$$\Pr(\mu_1 \neq \mu_2 \mid \mu_1 = \mu_2 \pm \delta) =$$

$$1 - T\left(t_{1-\frac{\alpha}{2},\, n_1+n_2-2};\, n_1+n_2-2,\, \frac{\delta}{s\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}}\right) + T\left(-t_{1-\frac{\alpha}{2},\, n_1+n_2-2};\, n_1+n_2-2,\, \frac{\delta}{s\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}}\right)$$

其中：

α 是显著性水平。

n_1 和 n_2 是组样本大小

s 是合并标准差

δ 是待检差值。

$t_{1-\alpha,\nu}$ 是自由度为 ν 的中心 t 分布的第 $(1 - \alpha)$ 分位数

$T(t;\, \nu,\, \lambda)$ 是自由度为 ν 且非中心参数为 λ 的非中心 t 分布的累积分布函数。

若未假定 σ_1 和 σ_2 相等，则上面等式中的自由度 n_1+ n_2-2 设置为：

$$\mathrm{DF} = \frac{\left(\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}\right)^2}{\frac{1}{n_1-1}\left(\frac{s_1^2}{n_1}\right)^2 + \frac{1}{n_2-1}\left(\frac{s_2^2}{n_2}\right)^2}$$

当 σ_1 和 σ_2 已知时，z 分布用于功效计算。基于备择假设的形式计算功效。对于单侧、上侧备择假设：

$$\Pr(\mu_1 > \mu_2 \mid \mu_1 = \mu_2 + \delta) = 1 - \Phi\left(z_{1-\alpha} - \frac{\delta}{\sqrt{\frac{\sigma_1}{n_1}+\frac{\sigma_2}{n_2}}}\right)$$

对于单侧、下侧备择假设：

$$\Pr(\mu_1 < \mu_2 \mid \mu_1 = \mu_2 - \delta = \Phi\left(-z_{1-\alpha} - \frac{\delta}{\sqrt{\frac{\sigma_1}{n_1}+\frac{\sigma_2}{n_2}}}\right)$$

对于双侧备择假设：

$$\Pr(\mu_1 \neq \mu_2 \mid \mu_1 = \mu_2 \pm \delta) = 1 - \Phi\left(z_{1-\frac{\alpha}{2}} - \frac{\delta}{\sqrt{\frac{\sigma_1}{n_1} + \frac{\sigma_2}{n_2}}}\right) + \Phi\left(-z_{1-\alpha} - \frac{\delta}{\sqrt{\frac{\sigma_1}{n_1} + \frac{\sigma_2}{n_2}}}\right)$$

其中：

α 是显著性水平。

n_1 和 n_2 是组样本大小

σ_1 和 σ_2 是已知的组标准差

δ 是待检差值。

$z_{1-\alpha}$ 是 z 分布的第 $(1 - \alpha)$ 分位数

$\Phi(x)$ 是正态分布的累积分布函数。

两个独立样本比例的功效

使用 “两个独立样本比例的功效分析器”可确定关于两个组的比例的假设检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 两个独立样本比例的功效**。探索在样本大小、功效、显著性和假设的待检差值之间如何权衡。样本大小与功效与以下假设检验相关联：

$$H_0: p_1 - p_2 = D_0$$

与双侧备择假设：

$$H_a: p_1 - p_2 \neq D_0$$

或与以下某个单侧备择假设：

$$H_a: (p_1 - p_2) < D_0 \text{ 或者 } H_a: (p_1 - p_2) > D_0$$

其中 p_1 和 p_2 是来自两个总体的总体比例，D_0 是假设的比例差值。

“两个独立样本比例的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

检验类型 指定单侧或双侧假设检验。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

检验参数 相互关联并在进行更改时随之更新的参数。

组 1 比例 (p1) 指定您期望或假设的组 1 比例。

组 2 比例 (p2) 指定您期望或假设的组 2 比例。

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。功效曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非锁定其中一个组。只能锁定一个组样本大小。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

两个独立样本比例的统计详细信息

用于检验两个样本组比例差值的功效计算基于正态近似。计算取决于备择假设的形式。对于单侧、上侧备择假设：

$$\Pr(p_1 > p_2 \mid p_1 = p_2 + \delta) = 1 - \Phi\left(z_{1-\alpha} - \frac{\delta}{\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}}\right)$$

对于单侧、下侧备择假设：

$$\Pr(p_1 > p_2 \mid p_1 = p_2 - \delta) = \Phi\left(-z_{1-\alpha} - \frac{\delta}{\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}}\right)$$

对于双侧备择假设：

$$\Pr(p_1 \neq p_2 \mid p_1 = p_2 \pm \delta) =$$

$$1 - \Phi\left(z_{1-\frac{\alpha}{2}} - \frac{\delta}{\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}}\right) + \Phi\left(-z_{1-\alpha} - \frac{\delta}{\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}}\right)$$

其中：

α 是显著性水平。

n_1 和 n_2 是组样本大小

p_1 和 p_2 是已知组标准差

δ 是待检差值。

$z_{1-\alpha}$ 是 z 分布的第 $(1 - \alpha)$ 分位数

$\Phi(x)$ 是正态分布的累积分布函数。

两个独立样本方差的功效

使用“两个独立样本方差的功效分析器”可确定关于两个组的方差的假设检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 两个独立样本方差的功效**。探索在样本大小、功效、显著性和假设的待检差值之间如何权衡。样本大小与功效与以下假设检验相关联：

$$H_0: \sigma_1 = \sigma_2$$

与双侧备择假设：

$$H_a: \sigma_1 \neq \sigma_2$$

或与单侧备择假设：

$$H_a: \sigma_1 < \sigma_2 \text{ 或者 } H_a: \sigma_1 > \sigma_2$$

其中，σ_1 是“组 1”的方差，σ_2 是“组 2”的方差。标准差偏离 σ_2 的差值，是您认为重要、需要检测的量（基于一组样本检测）。该差值表示为 σ_2/σ_1 的比值，或方差之比。对于相同的显著性水平和功效，检测小方差差值与检测大差值相比，前者需要更大的样本大小。假定关注的总体服从正态分布。

“两个独立样本方差的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

检验类型 指定单侧或双侧假设检验。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

检验参数 相互关联并在进行更改时随之更新的参数。

方差比（组 2/组 1） 指定方差的比值。

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。功效曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非锁定其中一个组。只能锁定一个组样本大小。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

两个独立样本方差分析器的统计详细信息

用于检验两个样本组方差比的功效计算基于标准 F 检验。计算取决于备择假设的形式。对于单侧、上侧备择假设：

$$\Pr(\sigma_0 > \sigma_1 \mid \sigma_1 = \sigma_0\rho) = 1 - F(\rho f_{1-\alpha,\, n_1-1,\, n_2-1}; n_1-1, n_2-1)$$

对于单侧、下侧备择假设：

$$\Pr(\sigma_0 < \sigma_1 \mid \sigma_1 = \sigma_0\rho) = F(\rho f_{\alpha,\, n_1-1,\, n_2-1}; n_1-1, n_2-1)$$

对于双侧备择假设：

$$\Pr(\sigma_0 \neq \sigma_1 \mid \sigma_1 = \sigma_0\rho) =$$

$$1 - F\left(\rho f_{1-\frac{\alpha}{2},\, n_1-1,\, n_2-1}; n_1-1, n_2-1\right) + F\left(\rho f_{\frac{\alpha}{2},\, n_1-1,\, n_2-1}; n_1-1, n_2-1\right)$$

其中：

α 是显著性水平。

n_1 和 n_2 是组样本大小

$\rho=\sigma_1/\sigma_0$

$f_{1-\alpha,\nu1,\nu2}$ 是自由度为 ν_1 和 ν_2 的 F 分布的第 $(1 - \alpha)$ 分位数

$F(x, \nu)$ 是自由度为 ν 的 F 分布的累积分布函数。

两个独立样本等价性的功效

使用 “两个样本等价性的功效分析器”可确定关于两个组的等价性检验的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 两个独立样本等价性的功效**。探索在变异性假设、样本大小、功效、显著性和等价性极差之间如何权衡。样本大小与功效与以下假设检验相关联：

H_0: $\mu_1 - \mu_2 \geq \delta_M$ 或者 H_0: $\mu_1 - \mu_2 \leq \delta_m$

对比备择假设：

H_a: $\delta_m < \mu_1 - \mu_2 < \delta_M$

其中，μ_1 和 μ_2 是真实组均值，(δ_m, δ_M) 是等价性范围。对于相同的显著性水平和功效，检测小差值与检测大差值相比，前者需要更大的样本大小。假定关注的总体服从正态分布。

“两个独立样本等价性的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

等价性范围

最大差值 指定幅度，超出它的均值被视为不同于参考均值。

最小差值 指定幅度，低于该值的均值被视为不同于参考均值。

注意：通常，等价性幅度是对称的，但并不一定必须对称。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

组 1 标准差 (σ_1) 为其中一个组 “组 1”指定假定的标准差。

组 2 标准差 (σ_2) 为第二个组 “组 2”指定假定的标准差。

总体标准差已知 指定计算基于已知总体标准差。

检验参数 相互关联并在进行更改时随之更新的参数。

均值差值 指定用来定义等价性范围的组均值之间的差值。

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。功效曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非锁定其中一个组。只能锁定一个组样本大小。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

两个独立样本等价性分析器的统计详细信息

检验两个组均值等价性的功效计算基于 Chow et al. (2008) 中所述的方法。

若 σ_1 和 σ_2 未知，功效 (1-β) 计算如下：

$$1-\beta = \left(-t_{1-\alpha,\, df};df,\ \frac{\delta-\delta_M}{\sqrt{\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}}} \right) - T\left(t_{1-\alpha,\, df};df,\ \frac{\delta-\delta_m}{\sqrt{\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}}} \right)$$

其中：

$$df = \begin{cases} n_1+n_2-2 & \text{若 } \sigma_1=\sigma_2 \\ \dfrac{\left(\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}\right)^2}{\frac{1}{n_1-1}\left(\frac{s_1^2}{n_1}\right)^2+\frac{1}{n_2-1}\left(\frac{s_2^2}{n_2}\right)^2} & \text{若 } \sigma_1\neq\sigma_2 \end{cases}$$

α 是显著性水平。

n_1 和 n_2 是组样本大小

s_1 和 s_2 是组标准差

δ 是待检差值。

(δ_m, δ_M) 是等价性极差

$t_{1-\alpha,\nu,}$ 是自由度为 ν 的中心 t 分布的第 (1 - α) 分位数

$T(t; \nu, \lambda)$ 是自由度为 ν 且非中心参数为 λ 的非中心 t 分布的累积分布函数。

若 σ 未知，则功效 (1-β) 计算如下：

$$1-\beta = \Phi\left(\left|\frac{\delta-\delta_M}{\sqrt{\frac{\sigma_1}{n_1}+\frac{\sigma_2}{n_2}}}\right| - z_{1-\alpha}\right) + \Phi\left(\left|\frac{\delta-\delta_m}{\sqrt{\frac{\sigma_1}{n_1}+\frac{\sigma_2}{n_2}}}\right| - z_{1-\alpha}\right) - 1$$

方差分析的功效

使用“方差分析的功效分析器”可确定要使用方差分析进行分析的 k 个组或处理的研究的样本大小。选择**实验设计 > 样本大小分析器 > 功效 > 方差分析的功效**。探索在变异性假设、样本大小、功效和显著性之间如何权衡。样本大小与功效与以下假设检验相关联：

H_0: $\mu_1 = \mu_2 = \ldots = \mu_k$

与双侧备择假设：

H_a：并非所有均值都相等

其中：

$X_{ij} \sim N(\mu_j, \sigma^2)$ for $i = 1, \ldots, n,\ j = 1, \ldots, k$

“方差分析的功效分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

组数 指定实验中的组或处理数。

组内方差 (σ^2) 指定假定各组间标准差相等时每组的假定方差。

检验参数 相互关联并在进行更改时随之更新的参数。

组间方差 指定围绕总均值的单个组均值的方差。

样本大小（每组） 指定实验中每组所需的观测数（试验次数、实验单元数或处理数）。

功效 指定当原假设为假时拒绝原假设的概率。在其他所有参数保持不变时，功效随着样本大小的增加而增加。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

方差分析分析器的统计详细信息

假定每个组的标准差相等，用于检验多个组的均值之间的差值的功效计算基于标准 F 检验。功效 (1-β) 计算如下：

$$1 - \beta = F\left(f_{1-\alpha, N-K, n-1}, \frac{\sigma^2_{BG}}{\sigma^2}\right)$$

其中

$F(df_1, df_2, \lambda)$ 是具有非中心参数 λ 的非中心 F 分布。

K 是组数

n 是每组内的样本数（假定所有组的样本数都相同）

$N = nK$

σ^2_{BG} 是组均值围绕总均值的变异

σ^2 是组内方差（假定所有组都相同）

置信区间计算器

使用置信区间计算器选择研究样本大小，以获得具有预先指定特征的区间估计值。探索样本大小对区间估计误差幅度的影响。探索一个均值的置信区间、预测区间或容差区间的样本大小。探索两个均值，或是一个或两个比例或方差的置信区间的样本大小。

对于区间估计，选择**实验设计 > 样本大小分析器 > 置信区间**：

- 第 683 页的“单样本均值的误差幅度”
- 第 685 页的“单样本比例的误差幅度”
- 第 686 页的“单样本方差的误差幅度”
- 第 687 页的“两个独立样本均值的误差幅度”
- 第 689 页的“两个独立样本比例的误差幅度”
- 第 691 页的“两个独立样本方差的误差幅度”

单样本均值的误差幅度

使用“单样本均值的区间分析器”可确定置信区间、预测区间或容差区间的样本大小。选择**实验设计 > 样本大小分析器 > 置信区间 > 单样本均值的误差幅度**。探索在变异性假设、样本大小、显著性和误差幅度之间如何权衡。

“单样本均值的区间分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

区间用途

置信 指定均值的置信区间。

预测 指定用于一个未来观测的预测区间。

容差 指定用于覆盖一定比例的总体的容差区间。

区间类型

界限 指定单侧区间（上限或下限）

区间 指定双侧区间。

固定参数

Alpha 指定置信水平为 1 - alpha。对于 95% 置信区间，默认 alpha 水平为 0.05。

标准差 (σ) 指定假定的标准差。

比例 （仅当为“区间用途”选择了“容差”时才可用。）指定容差区间要覆盖的总体比例。

总体标准差已知 指定计算基于已知总体标准差。

区间参数 相互关联并在进行更改时随之更新的参数。

样本大小 指定构造区间所需的总观测数（试验次数、实验单元数或样本数）。

误差幅度 （仅当为“区间类型”选定“区间”时才可用。）指定区间的半宽。在其他所有参数保持不变时，误差幅度随着样本大小的增加而减小。

界限大小 （仅当为“区间类型”选定“界限”时才可用。）指定从界限到估计值的距离。在其他所有参数保持不变时，界限随着样本大小的增加而减小。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本区间分析器的统计详细信息

若 σ 已知，则每种区间类型的计算都使用基于标准正态分布的过程，否则使用 *t* 分布过程或近似过程。

置信区间

置信区间的误差幅度 (MOE) 计算如下：

$$\text{MOE} = \begin{cases} z_{1-\alpha/2}\dfrac{\sigma}{\sqrt{n}} & \text{若 } \sigma \text{ 已知} \\ t_{1-\alpha/2,\, n-1}\dfrac{s}{\sqrt{n}} & \text{若 } \sigma \text{ 未知} \end{cases}$$

界限计算为：

$$\text{界限} = \begin{cases} \pm z_{1-\alpha}\dfrac{\sigma}{\sqrt{n}} & \text{若 } \sigma \text{ 已知} \\ t_{1-\alpha,\, n-1}\dfrac{s}{\sqrt{n}} & \text{若 } \sigma \text{ 未知} \end{cases}$$

预测区间

预测区间的误差幅度 (MOE) 计算如下：

$$\text{MOE} = \begin{cases} z_{1-\alpha/2}\sigma\sqrt{\left(1+\dfrac{1}{n}\right)} & \text{若 } \sigma \text{ 已知} \\ t_{1-\alpha/2,\, n-1}s\sqrt{\left(1+\dfrac{1}{n}\right)} & \text{若 } \sigma \text{ 未知} \end{cases}$$

界限计算为：

$$\text{界限} = \begin{cases} \pm z_{1-\alpha}\sigma\sqrt{\left(1+\dfrac{1}{n}\right)} & \text{若 } \sigma \text{ 已知} \\ \pm t_{1-\alpha,\, n-1}s\sqrt{\left(1+\dfrac{1}{n}\right)} & \text{若 } \sigma \text{ 未知} \end{cases}$$

容差区间

对于总体的比例 q 上的容差区间，基于 Krishnamoorthy and Mathew (2009) 中所述的近似过程计算误差幅度或界限。

容差区间的误差幅度 (MOE) 计算如下：

$$\text{MOE} = \begin{cases} \sigma\sqrt{\left(\dfrac{N\chi^2_{q,\,1,\,\frac{1}{n}}}{\chi^2_{\alpha,\,N}}\right)} & \text{若 } \sigma \text{ 已知} \\ s\sqrt{\left(\dfrac{(n-1)\chi^2_{q,\,1,\,\frac{1}{n}}}{\chi^2_{\alpha,\,n-1}}\right)} & \text{若 } \sigma \text{ 未知} \end{cases}$$

界限计算为：

$$\text{界限} = \begin{cases} \pm\dfrac{t_{1-\alpha,\,N,\,z_q\sqrt{n}}}{\sqrt{n}} & \text{若 } \sigma \text{ 已知} \\ \pm\dfrac{t_{1-\alpha,\,n-1,\,z_q\sqrt{n}}}{\sqrt{n}} & \text{若 } \sigma \text{ 未知} \end{cases}$$

其中 N 是表示渐近行为的适度大的整数值。JMP 将 N 的值设置为 2000。

单样本比例的误差幅度

使用“单样本比例的区间分析器”可确定置信区间的样本大小。选择**实验设计 > 样本大小分析器 > 置信区间 > 单样本比例的误差幅度**。探索在假设比例、样本大小、显著性和区间的误差幅度之间如何权衡。计算使用正态近似。

“单样本比例的区间分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

区间类型

界限 指定单侧区间（上限或下限）

区间 指定双侧区间。

固定参数

Alpha 指定置信水平，1 - alpha。对于 95% 置信区间，默认 alpha 水平为 0.05。

区间参数 相互关联并在进行更改时随之更新的参数。

比例 指定区间的假设比例。

样本大小 指定构造区间所需的总观测数（试验次数、实验单元数或样本数）。

误差幅度 指定区间的半宽。在其他所有参数保持不变时，误差幅度随着样本大小的增加而减小。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本比例区间分析器的统计详细信息

用于捕获总体比例的区间计算基于 Agresti-Coull 方法，请参见 Agresti and Coull(1998)。

置信区间的误差幅度 (MOE) 计算如下：

$$\text{MOE} = z_{1-\alpha/2}\sqrt{\frac{\tilde{p}(1-\tilde{p})}{\tilde{n}}}$$

界限计算为：

$$\text{界限} = \pm z_{1-\alpha}\sqrt{\frac{\tilde{p}(1-\tilde{p})}{\tilde{n}}}$$

其中

$$\tilde{n} = n + z_q^2$$

且

$$\tilde{p} = \frac{np + z_q^2/2}{\tilde{n}}$$

且 $q = 1 - \alpha/2$（对于区间），$q = 1 - \alpha$（对于界限）。

单样本方差的误差幅度

使用“单样本方差的区间分析器”可确定置信区间的样本大小。选择**实验设计 > 样本大小分析器 > 置信区间 > 单样本方差的误差幅度**。探索在样本大小、显著性和区间的误差幅度之间如何权衡。

“单样本方差的区间分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

区间类型

下限 指定单侧区间下限。

上限 指定单侧区间上限。

区间 指定双侧区间。

固定参数

Alpha 指定置信水平，1 - alpha。对于 95% 置信区间，默认 alpha 水平为 0.05。

区间参数 相互关联并在进行更改时随之更新的参数。

样本大小 指定构造区间所需的总观测数（试验次数、实验单元数或样本数）。

区间宽度 指定区间的完全宽度。在其他所有参数保持不变时，区间宽度随着样本大小的增加而减小。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

单样本方差区间分析器的统计详细信息

用于捕获总体方差的区间计算基于 X^2 分布。请注意，区间界限并非围绕样本估计值对称。

区间宽度计算为：

$$\text{宽度} = (n-1)\left(\frac{1}{\chi^2_{\alpha/2,\,n-1}} - \frac{1}{\chi^2_{1-\alpha/2,\,n-1}}\right)$$

下限计算为：

$$\text{下限} = \frac{n-1}{\chi^2_{1-\alpha,\,n-1}}$$

上限计算为：

$$\text{上限} = \frac{n-1}{\chi^2_{\alpha,\,n-1}}$$

两个独立样本均值的误差幅度

使用“两个独立样本均值的区间分析器”可确定置信区间的样本大小。选择**实验设计 > 样本大小分析器 > 置信区间 > 两个独立样本均值的误差幅度**。探索在变异性假设、样本大小、显著性和区间的误差幅度之间如何权衡。

“两个独立样本均值的区间分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

区间类型

界限 指定单侧区间（上限或下限）

区间 指定双侧区间

固定参数

Alpha 指定置信水平，1 - alpha。对于 95% 置信区间，默认 alpha 水平为 0.05。

组 1 标准差 (σ_1) 为其中一个组“组 1”指定假定的标准差。

组 2 标准差 ($\sigma 2$) 为第二个组“组 2”指定假定的标准差。

总体标准差已知 指定计算基于已知总体标准差。

区间参数 相互关联并在进行更改时随之更新的参数。

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。误差幅度曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非锁定其中一个组。只能锁定一个组样本大小。

误差幅度 指定区间的半宽。在其他所有参数保持不变时，误差幅度随着样本大小的增加而减小。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

两个独立样本区间分析器的统计详细信息

用于捕获总体均值差值的区间计算基于标准正态分布或 *t* 分布，这取决于 σ_1 和 σ_2 是已知还是未知。

误差幅度 (MOE) 计算为：

$$\text{MOE} = \begin{cases} z_{1-\alpha/2}\sqrt{\dfrac{\sigma_1^2}{n_1}+\dfrac{\sigma_2^2}{n_2}} & \text{若 } \sigma_1 \text{且 } \sigma_2 \text{ 已知} \\ t_{1-\alpha/2,\,n-1}\sqrt{\dfrac{s_1^2}{n_1}+\dfrac{s_2^2}{n_2}} & \text{若 } \sigma_1 \text{且 } \sigma_2 \text{ 未知} \end{cases}$$

当 σ_1 和 σ_2 未知时，界限计算为：

$$界限 = \pm t_{1-\alpha,\, df}\sqrt{\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}}$$

其中

$$df = \begin{cases} n_1 + n_2 - 2 & 若\ \sigma_1 = \sigma_2 \\ \dfrac{\left(\dfrac{s_1^2}{n_1}+\dfrac{s_2^2}{n_2}\right)^2}{\dfrac{1}{n_1-1}\left(\dfrac{s_1^2}{n_1}\right)^2 + \dfrac{1}{n_2-1}\left(\dfrac{s_2^2}{n_2}\right)^2} & 若\ \sigma_1 \neq \sigma_2 \end{cases}$$

当 σ 已知时，界限计算为：

$$界限 = \pm z_{1-\alpha}\sqrt{\frac{\sigma_1^2}{n_1}+\frac{\sigma_2^2}{n_2}}$$

两个独立样本比例的误差幅度

使用“两个独立样本比例的区间分析器”可确定两个比例的差值、对数相对风险或对数优势的置信区间的样本大小。选择**实验设计 > 样本大小分析器 > 置信区间 > 两个独立样本比例的误差幅度**。探索在变异性假设、样本大小、显著性和区间的误差幅度之间如何权衡。

“两个独立样本比例的区间分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

区间类型

界限 指定单侧区间（上限或下限）

区间 指定双侧区间。

比较方法

比例差值 指定两个比例差值 (p_1-p_2) 的置信区间。

对数相对风险 指定使用对数尺度的相对风险的置信区间 (p_1/p_2)。

对数优势比 指定使用对数尺度的优势的置信区间 ($p_1/(1-p_2)$)。

固定参数

Alpha 指定置信水平，1 - alpha。对于 95% 置信区间，默认 alpha 水平为 0.05。

区间参数 相互关联并在进行更改时随之更新的参数。

组 1 比例 指定组 1 的假设比例。

组 2 比例 指定组 2 的假设比例。

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。误差幅度曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非锁定其中一个组。只能锁定一个组样本大小。

误差幅度 指定区间的半宽。在其他所有参数保持不变时，误差幅度随着样本大小的增加而减小。

界限大小 指定区间界限。在其他所有参数保持不变时，误差幅度随着样本大小的增加而减小。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

两个独立比例区间分析器的统计详细信息

每种区间类型的计算都基于正态近似。

比例差值

对于有关比例差值的区间，误差幅度 (MOE) 计算如下：

$$\text{MOE} = z_{1-\alpha/2}\sqrt{\frac{p_1(1-p_1)}{n_1}+\frac{p_2(1-p_2)}{n_2}}$$

界限计算为：

$$\text{界限} = \pm z_{1-\alpha/2}\sqrt{\frac{p_1(1-p_1)}{n_1}+\frac{p_2(1-p_2)}{n_2}}$$

对数相对风险

对于相对风险的对数，误差幅度 (MOE) 计算为：

$$\text{MOE} = z_{1-\alpha/2}\sqrt{\frac{(1-p_1)}{n_1p_1}+\frac{(1-p_2)}{n_2p_2}}$$

界限计算为：

$$\text{界限} = \pm z_{1-\alpha}\sqrt{\frac{(1-p_1)}{n_1p_1}+\frac{(1-p_2)}{n_2p_2}}$$

对数优势

对于相对风险的对数，误差幅度计算为：

$$\text{MOE} = z_{1-\alpha/2}\sqrt{\frac{1}{n_1p_1}+\frac{1}{n_1(1-p_1)}+\frac{1}{n_2p_2}+\frac{1}{n_2(1-p_2)}}$$

界限计算为：

$$\text{界限} = \pm z_{1-\alpha}\sqrt{\frac{1}{n_1p_1}+\frac{1}{n_1(1-p_1)}+\frac{1}{n_2p_2}+\frac{1}{n_2(1-p_2)}}$$

两个独立样本方差的误差幅度

使用“两个独立样本方差的区间分析器”可确定置信区间的样本大小。选择**实验设计 > 样本大小分析器 > 置信区间 > 两个独立样本方差的误差幅度**。探索在样本大小、显著性和区间的误差幅度之间如何权衡。

“两个独立样本方差的区间分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

区间类型

下限 指定单侧区间下限。

上限 指定单侧区间上限。

区间 指定双侧区间。

固定参数

Alpha 指定置信水平，1 - alpha。对于 95% 置信区间，默认 alpha 水平为 0.05。

区间参数

组 1 样本大小 指定您实验中的“组 1”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 1”样本大小。

组 2 样本大小 指定您实验中的“组 2”需要的总观测数（试验次数、实验单元数或样本数）。选择“锁定”以锁定“组 2”样本大小。

总样本大小 指定您实验中需要的总观测数（试验次数、实验单元数或样本数）。误差幅度曲线基于总样本大小。

提示：调整一组的样本大小会影响第二组的样本大小，而总样本大小不变。调整总样本大小会按比例调整两个组，除非锁定其中一个组。只能锁定一个组样本大小。

区间宽度 指定区间的完全宽度。在其他所有参数保持不变时，区间宽度随着样本大小的增加而减小。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

两个独立方差区间分析器的统计详细信息

用于捕获两个总体方差的区间计算基于 *F* 分布。误差幅度 (MOE) 计算为：

$$\text{MOE} = F_{1-\alpha/2,\, n_1-1,\, n_2-1} - F_{\alpha/2,\, n_1-1,\, n_2-1}$$

下限计算为：

$$\text{下限} = F_{\alpha,\, n_1-1,\, n_2-1}$$

上限计算为：

$$\text{上限} = F_{1-\alpha,\, n_1-1,\, n_2-1}$$

“可靠性验证”计算器

使用 “可靠性验证”计算器在指定的时间段检验指定的单元数。若失效的单元少于 *k* 个，则通过验证，您可以认为产品可靠性满足或超过可靠性标准。

对于可靠性验证，选择**实验设计 > 样本大小分析器 > 可靠性**：

- 第 693 页的“参数可靠性验证”
- 第 694 页的“非参数可靠性验证”

参数可靠性验证

使用“参数可靠性的验证分析器”可确定在用于可靠性验证的检验中投入的单元数。选择**实验设计 > 样本大小分析器 > 可靠性 > 参数可靠性验证**。探索在时间、显著性、失效次数和待检单元数之间如何权衡。

“参数可靠性的验证分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

分布 指定假定的失效时间分布。可用的分布包括：Weibull 和对数正态。有关这些分布的详细信息，请参见《可靠性和生存方法》。

验证时间 指定待检项应在预定可靠性下生存的最短时间长度。

验证可靠性 指定待检项生存到定义的可靠性标准“验证时间”的概率。

尺度参数 指定失效时间分布的尺度参数。

注意：指定的 “分布”为 “Weibull”时，“尺度”字段用 Weibull β 表示。

验证参数 相互关联并在进行更改时随之更新的参数。

允许的最大失效次数 指定成功的检验验证允许的最大失效次数。

样本大小 指定验证所需的试验单元总数。

检验时间 指定验证检验的时间长度。

保存设置 将当前设置保存至 “已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的 “样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

参数可靠性验证分析器的统计详细信息

对于参数可靠性验证检验，使用二项分布计算验证产品达到验证时间 T_d 且可靠性为 R_d 所需的检验时间，该二项分布使用了从假定的寿命分布计算出的成功概率。

Weibull 分布的检验时间计算为：

$$\text{检验时间} = T_d\left(\frac{-\log(P)}{\log(R_d)}\right)^{1/\beta}$$

对数正态分布的检验时间计算为：

$$\text{检验时间} = \exp\left(\frac{z_{1-e^{-\log P}} - z_{1-R_d}}{\sigma}\right)$$

其中

$$\log P = -\log\left(\frac{Sf}{Sf+1}\right)$$

$$Sf = \frac{n-c}{c+1} f_{\alpha,\, 2(n-c),\, 2(c+1)}$$

且 c 是验证期间允许的最大失效次数。

非参数可靠性验证

使用“非参数可靠性的验证分析器”可确定：当不存在假设失效率分布时，在用于可靠性验证的检验中投入的单元数。选择**实验设计 > 样本大小分析器 > 可靠性 > 非参数可靠性验证**。探索在时间、显著性、失效次数和待检单元数之间如何权衡。

“非参数可靠性的验证分析器”选项

使用单选按钮、文本框和滑块设置研究假设并探索样本大小。对设置进行更改时，曲线随之更新。或者，通过拖动曲线上的十字准线或调整轴文本框中的值来更改设置。

固定参数

Alpha 指定 I 类误差的概率，即当原假设为真时拒绝原假设的概率。它通常是指检验的显著性水平。默认 alpha 水平为 0.05。

验证参数 相互关联并在进行更改时随之更新的参数。

允许的最大失效次数 指定成功的检验验证允许的最大失效次数。

样本大小 指定验证所需的试验单元总数。

验证可靠性 指定待检项生存到定义的可靠性标准“验证时间”的概率。

保存设置 将当前设置保存至“已保存设置”表。这样您就可以保存一套备选研究计划。请参见第 695 页的“样本大小分析器中的已保存设置”。

制成数据收集表 创建可用于数据收集的新数据表。

非参数可靠性验证分析器的统计详细信息

对于非参数可靠性验证检验，基于二项分布计算验证的可靠性。二项 cdf 的 α 分位数可通过从 *F* 分布变换为 α 分位数来计算，请参见 Jowett, (1963)。

可靠性计算为：

$$R = \frac{Sf}{Sf+1}$$

其中

$$Sf = \frac{n-c}{c+1} f_{\alpha, 2(n-c), 2(c+1)}$$

且 *c* 是验证期间允许的最大失效次数。

样本大小分析器中的已保存设置

“已保存设置”报表提供可支持您比较各个研究方案的已保存设置表。每次您点击**保存设置**按钮后，就会向表中添加新的一行。点击表中的某行可将分析器重置为这些设置。在表中右击并选择**列**可显示表中的附加参数值。

“已保存设置”红色小三角包含以下选项：

清除选定的行 从“已保存设置”表中删除选定行。

全部清除 从“已保存设置”表中删除所有行。

制表 创建一个新的数据表，该表包含所有“已保存设置”表列，包括隐藏的那些表列。

附录 A

列属性

了解实验设计分配的列属性

当您使用实验设计平台构造设计时，列属性会保存到包含所生成设计的数据表。本附录仅提供有关这些列属性（它们保存到实验设计平台构造的设计）的详细信息。示例说明实验设计平台如何分配和使用每个列属性。有关不由实验设计平台分配的列属性的说明，请参见《使用 JMP》。

本附录中所述的一些列属性在一般建模情形中很有用。要在更一般的情况下使用这些属性，您可以自行指定它们。当您的设计尚未由实验设计平台创建时，这个功能特别有用。本附录中的一些示例演示了您自行添加列属性的情形。

图 A.1 列属性星号和 “列信息” 窗口

目录

添加和查看列属性 699
响应限 700
响应限示例 701
编辑响应限 704
检测限值 706
函数响应 706
设计角色 706
设计角色示例 707
编码 709
低值和高值 709
“编码”列属性和中心多项式 710
编码示例 710
分配编码 713
混料 715
混料示例 716
因子更改 720
因子更改示例 721
值顺序 723
值顺序示例 724
指定值顺序 726
值标签 727
值标签示例 727
每区组试验次数 729
每区组试验次数示例 729
约束状态 730
约束状态示例 730

添加和查看列属性

实验设计平台自动将某些列属性保存到它们构造的设计表。但是，一些与设计实验相关的列属性在一般建模情形中很有用。要将这些列属性用于不是使用实验设计平台创建的数据表，您可以自行添加它们。

添加列属性

要将列属性分配给一个或多个列，请执行以下步骤：

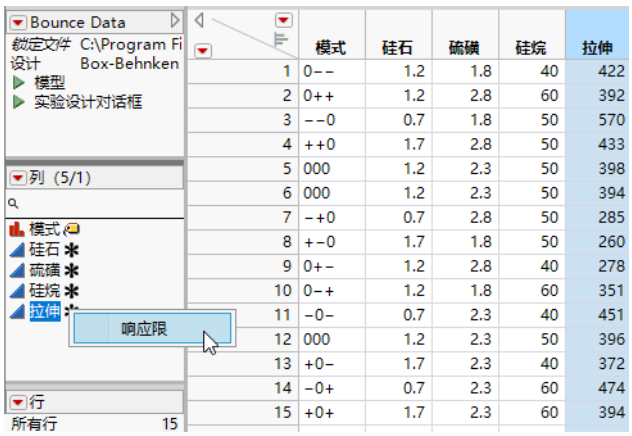
1. 选择要分配属性的一列或多列。
2. 执行以下操作之一：
   - 右击标题区域，选择**列属性**，然后选择属性。
   - 右击标题区域，选择**列信息**，然后从“列属性”菜单中选择属性。
   - 选择**列** > **列信息**，然后从“列属性”菜单中选择属性。
3. 在显示的列属性面板中，指定相应的值并选择合适的选项。
4. 点击**应用**添加列属性，或点击**确定**添加列属性并关闭列属性窗口。

提示：一个列可能已包含您要应用到其他列的属性。使用“标准化特性”命令可将该属性应用于其他列。请参见《使用 JMP》。

查看列属性

要查看分配给特定列的属性，请在列面板中点击列属性星号图标 ✱。点击一个属性以查看其设置或编辑它。图 A.2 显示了分配给 Bounce Data.jmp 样本数据表中**拉伸**的列属性，该数据表位于 Design Experiment 文件夹中。

图 A.2 “拉伸”的星号图标显示有两个列属性



您还可以通过访问“列信息”菜单中的“列属性”列表来查看列属性。选择要查看其列属性的一个或多个列，然后执行以下操作之一：

- 右击标题区域，选择**列信息**，然后从“列属性”列表中选择属性。
- 选择**列 > 列信息**，然后从“列属性”列表中选择属性。

响应限

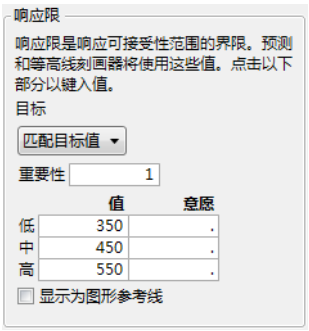
使用“响应限”列属性，您可以指定以下内容：

- 响应的变异范围的界限
- 意愿目标
- 响应的重要性的测度
- 意愿值

“响应限”列属性定义响应的意愿函数。“刻画器”和“等高线刻画器”使用意愿函数找到最优因子设置。请参见《刻画器指南》。

图 A.3 显示 Design Experiment 文件夹下 Bounce Data.jmp 样本数据表中**拉伸**响应的“列信息”窗口中的“响应限”面板。

图 A.3 “响应限”面板的示例



“响应限”面板包含以下区域：

目标 从菜单中选择您的响应目标。可用目标为：“最大化”、“匹配目标值”、“最小化”和“无”。JMP 会定义响应的意愿函数以匹配所选目标。若指定限值，则使用这些限值定义意愿函数。若未指定限值，JMP 使意愿函数基于从响应分布推导出来的保守限值。若选择“无”作为目标，则所有响应值被视为有相同程度的意愿。另见 “定制设计”一章中第 82 页的 “响应”。

重要性 输入在计算总意愿函数时每个响应的相对权重。“重要性”值可以为任意正数。未指定“重要性”值时，JMP 将给定分析中的所有响应视为具有相等的“重要性”值。若只有一个响应，则它的 “重要性”值为 1。

值 指定响应的上限、下限和“中”值。JMP 使用这些值构造响应的意愿函数。若未指定限值，JMP 使意愿函数基于保守的限值。若您的目标是 “匹配目标值”且未指定 “中”值，则目标定义为上限和下限的中点值。

意愿 指定反映您的 “低”、“中”和 “高”值意愿的值。“意愿”值应介于 0 到 1 之间。若您未指定 “意愿”值， JMP 将根据所选目标分配值。

显示为图形参考线 在 “预测值 - 实际值”图和 “预测刻画器”中为 “低”、“中”和 “高”值显示水平参考线。仅当指定限值时该选项才适用。

响应限示例

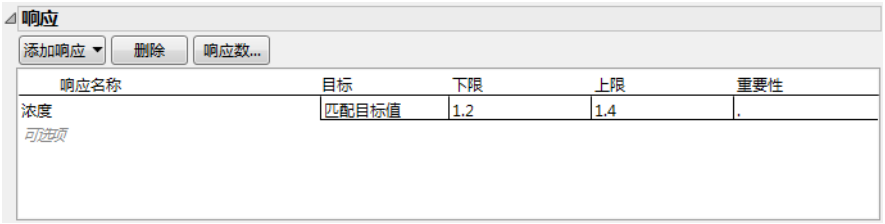
Coffee Data.jmp 样本数据表（位于 Design Experiment 文件夹中）包含执行的用于优化咖啡**浓度**的实验结果。有关该实验设计和分析的完整说明，请参见“从实验设计入手”一章中第 42 页的“咖啡浓度实验”。

您的目标是找到可使冲泡出的咖啡的目标浓度达到 1.3（这被视为具有最大意愿的值）的因子设置。小于 1.2 和大于 1.4 的值是完全不想要的值。介于 1.2 和 1.4 之间的值距离 1.3 越远，则越不想要。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Coffee Data.jmp。
2. 在 “表”面板中，点击**实验设计对话框**脚本旁边的绿色小三角。

   “实验设计对话框”脚本会重新创建当时在创建 Coffee Data.jmp 中的实验设计时所用的“定制设计”对话框。
3. 打开 “响应”分级显示项。

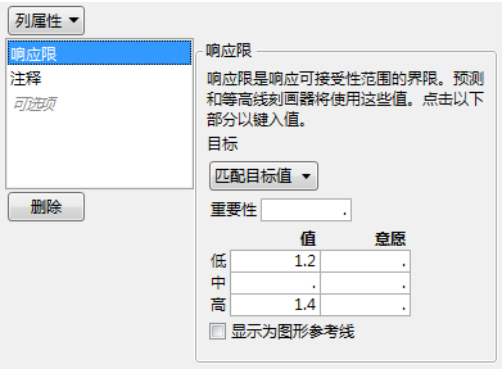
图 A.4 “定制设计”窗口中的 “响应”分级显示项



设计该实验时，指定了响应目标为“匹配目标值”，下限值为 1.2 且上限值为 1.4。因为只有一个响应，您未指定 “重要性”的值，因为它默认为 1。构造设计表时， JMP 会将 “响应限”列属性分配给**浓度**。

4. 关闭 “定制设计”窗口。
5. 在 Coffee Data.jmp 样本数据表中，选择**浓度**列，然后选择**列 > 列信息**。
6. 在 “列属性”列表中选择 “响应限”。

图 A.5 “浓度”的“响应限”列属性



请注意以下事项：

- “目标”设置为**匹配目标值**。
- 缺少“重要性”。缺少“重要性”时，JMP 将给定分析中的所有响应视为具有相等的“重要性”值。因此，JMP 为**浓度**分配“重要性”值 1。
- 下限为 1.2。
- 上限为 1.4。
- 未指定“中”值。

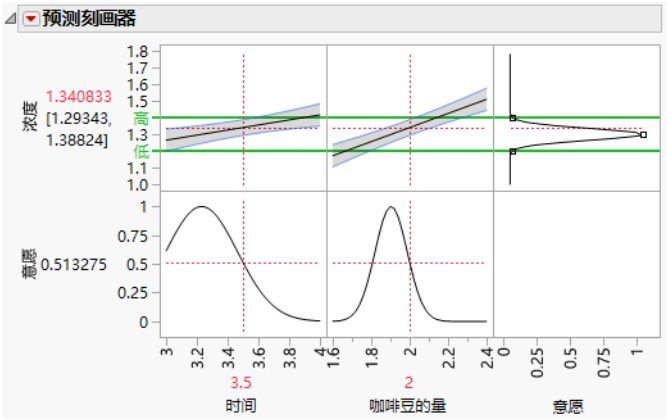
  因为未指定 “中”值，目标定义为上限和下限的中点值 （即 1.3）。
- 未指定“意愿”值。

7. 选择**显示为图形参考线**选项。

   该选项在 “预测值 - 实际值”图和 “预测刻画器”中为 “低”、“中”和 “高”值显示水平参考线。
8. 点击**确定**。
9. 在 Coffee Data.jmp 数据表中，点击**简化模型**脚本旁边的绿色小三角。
10. 点击**运行**。

    “预测刻画器”显示在报表底部。

图 A.6 显示 “浓度” 的意愿函数的刻画器



浓度的意愿函数显示在图中 “意愿” 的正上方。显示该图是因为数据表包含**浓度**的 “响应限” 列属性。“预测刻画器” 还为**浓度**的下限和上限显示参考线。

11. 按 Ctrl 键并点击 “意愿” 的**浓度**图。

图 A.7 “浓度” 的 “响应目标” 窗口

请注意以下事项：

- JMP 确定 “中” 值为您在 “响应限” 列属性中指定的 “高” 限值和 “低” 限值的中点值。
- 因为 “目标” 设置为 “匹配目标值”，JMP 将 “中” 值对应的 “意愿” 设置为 1。
- JMP 将 “高” 值和 “低” 值对应的 “意愿” 设置为很小的数，即 0.0183。
- 图 A.6 中的意愿图显示 JMP 如何使用图 A.7 中所示的意愿值。**浓度**的意愿函数在低值和高值范围之外时基本为 0，随着**浓度**接近目标值 1.3，意愿值逐渐增加到 1。“重要性” 值设置为 1，因为模型中只有一个响应。

12. 点击**取消**以退出窗口。
13. 点击 “预测刻画器” 红色小三角并选择**优化和意愿 > 最大化意愿**。

时间和**咖啡豆的量**的设置随即更新，以显示使**浓度**的意愿函数最大化的因子设置。但是，很多其他设置也可以使意愿函数最大化。有关如何标识使意愿函数最大化的其他设置的信息，请参见 《刻画器指南》。

编辑响应限

在 Vinyl Data.jmp 样本数据表中，已将 “响应限”列属性分配给响应**厚度**。该属性具有使**厚度**最大化的目标。假定您并不想使**厚度**最大化，而是想使塑料薄片的**厚度**介于 6 到 10 之间，目标**厚度**为 8.5。

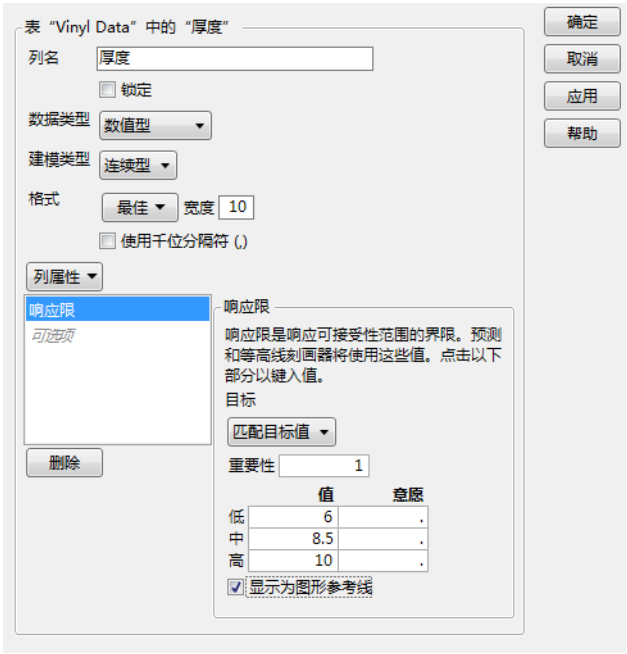
1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Vinyl Data.jmp。
2. 选择**厚度**列并选择**列** > **列信息**。

   “响应限”属性作为分配给**厚度**的唯一属性显示在 “列属性”列表中。“响应限”面板显示在列表右侧。
3. 点击**最大化**，然后选择**匹配目标值**。
4. 为**重要性**值键入 1。
5. 在 “值”下，为 “低”值键入 6、为 “中”值键入 8.5、为 “高”值键入 10。

   这是非对称响应限的示例。**厚度**像 6 这么小或像 10 这么大是可以接受的。但是**厚度**的目标值是 8.5。
6. 选择**显示为图形参考线**。

   该选项在 “预测值 - 实际值”图和 “预测刻画器”中为 “低”、“中”和 “高”值显示水平参考线。

图 A.8 已完成的 “响应限”面板



7. 点击**确定**。

8. 在 Vinyl Data.jmp 数据表中，点击**模型**脚本旁边的绿色小三角。

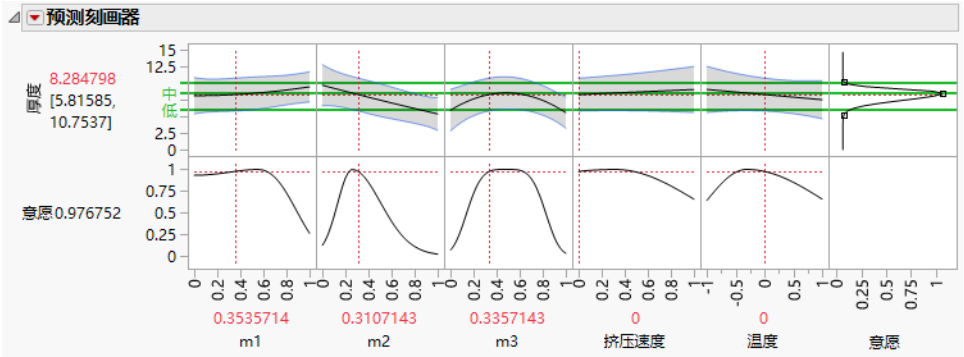
   请注意 m1、 m2 和 m3 是混料因子。此外，该设计涉及随机**整区**因子。因此，默认方法为 REML （推荐）。

9. 点击**运行**。
10. 点击 “响应 ‘厚度’” 红色小三角并选择**行诊断 > 标绘 “预测值 - 实际值”图**。

    下限、中值和上限对应的参考线显示在 “预测值 - 实际值”图上。

11. 点击 “响应 ‘厚度’” 红色小三角并选择**因子刻画 > 刻画器**。

图 A.9 显示非对称意愿函数的 “预测刻画器”



右边 “意愿”正上方的图显示 JMP 为**厚度**构造的意愿函数。中值 8.5 对应的意愿为 1。对于 6 以下和 10 以上的**厚度**值，意愿基本为 0。

12. 按 Ctrl 键并点击 “意愿”的**厚度**图。

图 A.10 “厚度”的 “响应目标”窗口

该窗口显示您的高值、中值和低值设置。它还基于您的 “匹配目标值”目标显示 JMP 分配的 “意愿”值。图 A.9 中所示的意愿函数是在高值、中值和低值处匹配图 A.10 中意愿设置的连续曲线。在其他值处，意愿函数分配与所选目标一致的意愿。

13. 点击**取消**。
14. 点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

因子的设置随即更新，以显示使**厚度**的意愿函数最大化的值。请记住很多其他设置也可以使意愿函数最大化。在这些最优设置处的预测响应为 8.5。回想一下您将 8.5 设置为目标设置，限值为 6 和 10。

15. 关闭 Vinyl Data.jmp 样本数据表而不保存更改。

JMP PRO 检测限值

“检测限值”列属性定义界限，超过该界限则无法测量响应。您可以使用这些限值来指定“广义回归”平台中的删失响应。

JMP PRO 函数响应

“FDE X”列属性指定某列作为函数响应。具有“FDE”列属性的多个 Y 列可以在“函数数据分析器”平台中使用，其中的实验设计因子作为补充因子。

设计角色

因子在设计的实验中以及更一般的模型中的行为可以是各种各样的。JMP 使用“设计角色”列属性来描述这些行为。以下是可能的设计角色：

- 连续
- 离散数值
- 分类
- 分区组
- 协变量
- 混料
- 常数
- 不受控
- 随机区组
- 信号
- 噪声

在很多 JMP 实验设计平台中，您可以指定具有不同设计角色的因子。在一些平台中，您的设计要求导致 JMP 去定义因子。例如，“整区”和“子区”就是当您指定极难更改和难以更改的因

子时 JMP 创建的因子。在可能出现各种设计角色的平台中，当 JMP 为您的设计创建设计表时，会为每个因子分配 “设计角色” 列属性。

- 有关 “随机区组”、“信号” 或 “噪声” 以外的其他设计角色的说明，请参见 “定制设计” 一章中第 85 页的 “因子类型”。
- 有关 “随机区组” 设计角色的说明，请参见 “定制设计” 一章中第 86 页的 “更改和随机区组”。
- 有关 “信号” 和 “噪声” 设计角色的说明，请参见 “田口设计” 一章中第 388 页的 “因子”。

设计角色示例

Odor Control Original.jmp 样本数据表中的实验研究了三个因子对**气味**的影响。您设计了 15 次试验并进行了实验。但是，向您报告实验结果时，您得知实验耗时三天。前 5 次试验在第 1 天执行，中间 5 次试验在第 2 天执行，最后 5 次试验在第 3 天执行。

因为温度和湿度的变异可能对响应有影响，您要把**日期**作为随机分区组因子。可以很容易向设计表添加**日期**列。但是您要使用 “评估设计” 平台将具有无法预料的区组的设计与原始设计比较。您还想可以使用 “扩充设计” 平台以防需要扩充设计。要使用 “评估设计” 和 “扩充设计” 平台，您需要将 “设计角色” 列属性添加到新的**日期**列。

1. 选择**帮助** > **样本数据库**，然后打开 Odor Control Original.jmp。
2. 选择第一列**试验**。
3. 选择**列** > **新建列**。
4. 在**列名**旁边，键入**天**。
5. 从**初始化数据**列表中，选择**序列数据**。
6. 输入以下内容：
   - 为**自**输入 1
   - 为**至**输入 3
   - 为**步长**输入 1
   - 为**每个值的重复次数**输入 5
7. 在**要添加的列数**旁边，键入 1。

图 A.11 完成的 “新建列” 窗口

8. 点击**确定**。
   日期列作为数据表中的第二列添加。
9. 选择**日期**列。
10. 选择**列** > **列信息**。
11. 从 “列属性” 列表中，选择**设计角色**。
12. 在 “设计角色” 面板中，点击**连续**，然后选择**随机区组**。
13. 点击**确定**。
    在列面板中，星号将出现在**日期**旁边。
14. 点击**日期**旁边的星号以验证分配了 “设计角色” 列属性。
15. 关闭 Odor Control Original.jmp 样本数据表而不保存更改。

编码

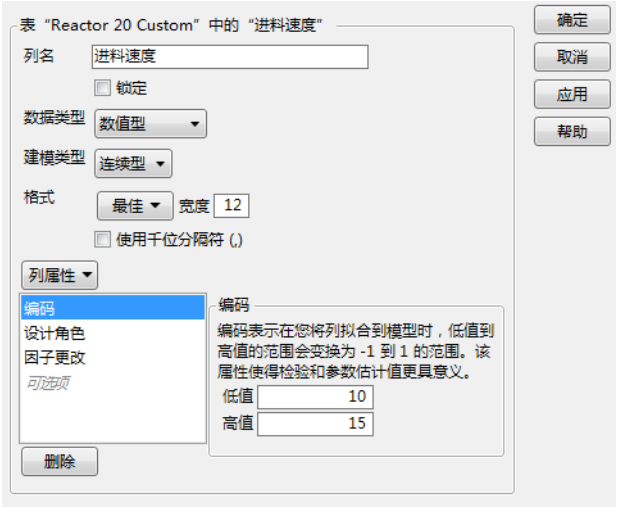
“编码”列属性仅应用到具有数值数据类型的列。它对列中的数据应用一个线性变换。在“编码”列属性窗口中，您指定低值和高值。原始数据中的低值和高值会变换为 –1 和 +1。每当该列作为“拟合模型”平台中的模型效应输入时，JMP 使用变换后的数据值。

出于以下原因，编码属性很有用：

- 编码后的预测变量使得可以更轻松地解释和比较参数估计值。
- 编码后的预测变量帮助降低具有交互作用和高阶项的模型的多重共线性。

除“加速寿命试验设计”之外的任何实验设计平台创建设计时，JMP 会为具有数值数据类型的每个非混料因子定义一个“编码”列属性。图 A.12 显示 Design Experiment 文件夹下 Reactor 20 Custom.jmp 样本数据表中**进料速度**列的“编码”列属性面板。

图 A.12 “进料速度”的“编码”属性面板



低值和高值

“编码”属性作为实验设计平台中设计构造的一部分应用时，“低值”最初设置为因子的最小设置，“高值”最初设置为最大设置。

您将“编码”属性应用到不包含该属性的列时，JMP 插入最小值作为“低值”，插入最大值作为“高值”。您可以根据需要更改这些值。

警告：将“编码”列属性分配给某列后，当您更改该列中的值时 JMP 不会自动更新它。若您更改具有“编码”列属性的列中的值，请查看“高值”和“低值”以确保它们仍是合适的。

“编码”列属性使列中的每个值中心化，方法是将每个值减去高值和低值的中点值，然后除以极差的一半。假定 H 是高值， L 是低值。然后将列中的每个 X 按以下公式进行变换：

$$\frac{X-(H+L)/2}{(H-L)/2}$$

对于每个因子，变换后的值具有中点值 0，范围从 -1 到 +1。

“编码”列属性和中心多项式

“中心多项式”选项位于“拟合模型”启动窗口的“模型规格”红色小三角菜单内。“中心多项式”通过减去列中每个值的均值，使多项式项中涉及的连续列中心化。请参见《拟合线性模型》。

若将“编码”列属性分配给某个列，则“中心多项式”选项对于该列没有影响。在涉及该列的多项式项中，按“编码”属性中的指定将值中心化和统一尺度。假定模型中的另一列没有“编码”属性且您选择了“中心多项式”，则该列通过它所在的任意多项式项中的均值中心化。

编码示例

Reactor 20 Custom.jmp 样本数据表包含来自 20 次试验设计的数据，该设计是使用“定制设计”平台构造的。实验针对某化学过程调查五个因子对产量响应（**反应百分比**）的影响。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Reactor 20 Custom.jmp。
2. 在“表”面板中，点击**实验设计对话框**脚本旁边的绿色小三角。
3. 打开**因子**分级显示项。

图 A.13 Reactor 20 Custom.jmp 中使用的设计的“因子”分级显示项

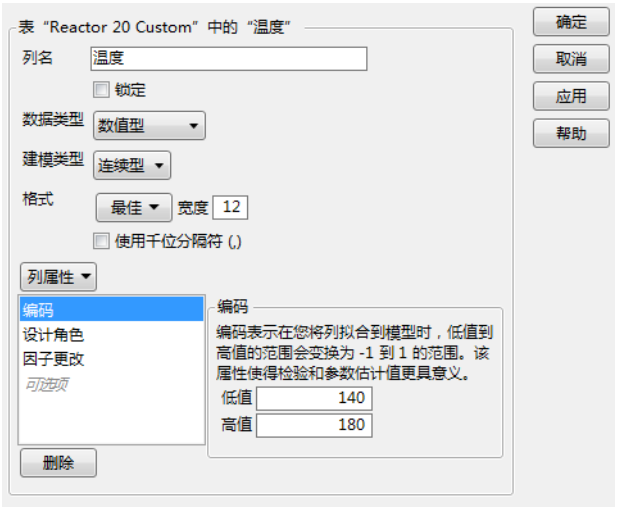
名称	角色	更改	值	
进料速度	连续	简单	10	15
催化剂	连续	简单	1	2
搅拌速度	连续	简单	100	120
温度	连续	简单	140	180
浓度	连续	简单	3	6

请注意**温度**的设置范围从 140 到 180。生成设计时，已将“编码”列属性分配给**温度**。“低值”设置为 140，“高值”设置为 180。

4. 关闭“定制设计”窗口。
5. 在 Reactor 20 Custom.jmp 样本数据表中，点击列面板中**温度**旁边的星号，然后选择**编码**。

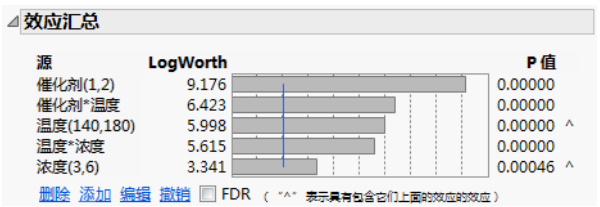
   随即显示“列信息”窗口并显示“编码”列属性面板。您可以看到 JMP 构造设计表时添加了列属性，指定了“低值”和“高值”。事实上，通过重复该步骤，您可以验证 JMP 为全部五个因子添加了“编码”属性。

图 A.14 “温度”的“编码”面板



6. 点击**取消**以关闭“列信息”窗口。
7. 在 Reactor 20 Custom.jmp 样本数据表中，点击**简化模型**脚本旁边的绿色小三角。
   该脚本拟合仅包含五个效应的模型，基于完整模型的分析这些效应被确定为显著。
8. 点击**运行**。

图 A.15 简化模型的“效应汇总”报表

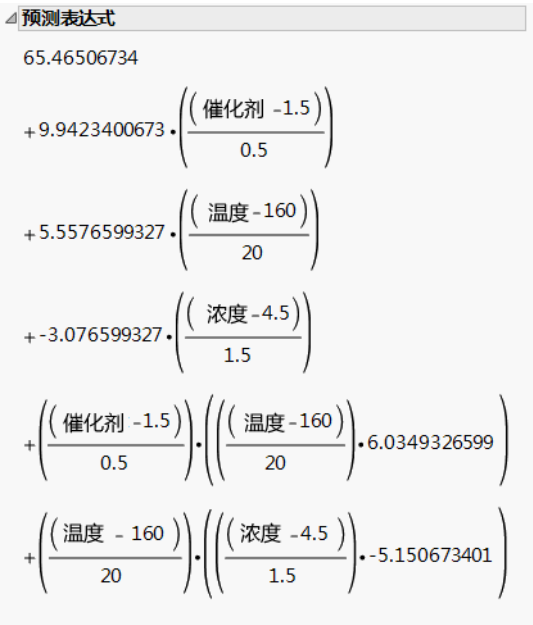


在“源”列表中，“编码”列属性中使用的高值和低值显示在主效应（**催化剂**、**温度**和**浓度**）右侧的括号中。没有为交互作用效应显示“编码”属性应用的范围。

提示：请注意**温度**和**浓度**对应 P 值右侧的“^”符号。这些符号指示这些主效应是具有更小 *p* 值的交互作用效应的分量效应。若模型中包含交互作用效应，则效应遗传原则要求所有分量效应也包括在模型中。请参见“从实验设计入手”一章中第 60 页的“效应遗传”。

9. 点击“响应 ‘反应百分比’”红色小三角并选择**估计值 > 显示预测表达式**。
   查看“预测表达式”分级显示项以了解编码如何影响预测公式。

图 A.16 简化模型的预测表达式



每个因子按 “编码”列属性的指定进行变换。例如，对于**温度**，注意以下事项：

- “编码”属性中的“低值”设置为 140。**温度**值 140 会变换为 -1。
- “编码”属性中的“高值”设置为 180。**温度**值 180 会变换为 +1。
- 低值和高值的中点值是 160。**温度**值 160 会变换为 0。

变换后的值可帮助您比较效应。**催化剂**的估计系数为 9.942，**浓度**的估计系数为 -3.077。由此得出结论：**催化剂**对**反应百分比**的预测效应是**浓度**对**反应百分比**的效应的三倍多。此外，这些系数指示预测的**反应百分比**随着**催化剂**的增加而增加，随着**浓度**的增加而降低。

变换后的值可帮助您解释系数：

- 所有因子处于中点值时，变换后的值为 0。预测的**反应百分比**是截距，即 65.465。
- **催化剂**和**浓度**处于中点值时，**温度**增加 20 个单位导致**反应百分比**增加 5.558 个单位。
- 假定**浓度**位于其中点值，因此变换后的值为 0：
  - **催化剂**位于中点值时，**温度**增加 20 个单位导致**反应百分比**增加 5.558 个单位。
  - **催化剂**采用高值设置时，**温度**增加 20 个单位导致**反应百分比**增加 5.558 + 6.035 = 11.593 个单位。

由此得出结论：交互作用项的系数 6.035 是对应**催化剂**变化 0.5 个单位，预测的**反应百分比**模型的斜率增量。

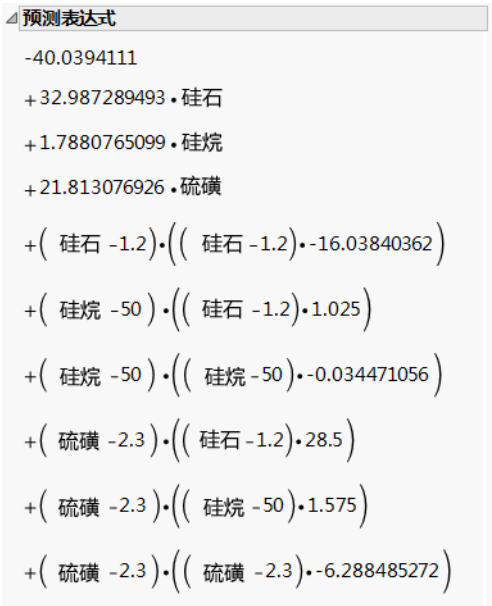
分配编码

Tiretread.jmp 样本数据表中的实验数据来自于以下实验: 研究**硅石**、**硅烷**和**硫磺**对轮胎胎面性能四个测度的影响。在本例中，您将只考虑其中一个响应（**磨损**）。

您首先使用未编码的因子拟合模型。然后将 “编码”属性分配给因子并重新运行模型以获取有意义的参数估计值。

1. 选择**帮助** > **样本数据库**，然后打开 Tiretread.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**磨损**，然后点击 Y。
4. 选择**硅石**、**硅烷**和**硫磺**，然后点击**宏** > **响应曲面**。
5. 选中**保持对话框打开**。
6. 点击**运行**。
7. 点击 “响应 ‘磨损’”红色小三角并选择**估计值** > **显示预测表达式**。

图 A.17 带未编码因子的模型的预测表达式



系数无法帮助您比较效应大小。系数的大小不反映效应在设置范围内对**磨损**的影响。此外，系数难以解释。例如，系数不便于您了解**硅石**位于范围的中点值时的预测响应。

将 “编码”列属性应用到这三个因子，以查看编码如何使系数更有意义。

8. 在 Tiretread.jmp 数据表中，在 “列”面板中选择**硅石**、**硅烷**和**硫磺**。点击 “列”红色小三角并选择**标准化特性**。

9. 在 “标准化属性” 面板中选择**列属性** > **编码**。
10. 点击**确定**。

    星号将显示在 “列” 面板中**硅石**、**硅烷**和**硫磺**的旁边，指示已为它们分配了列属性。
11. 在 “拟合模型” 窗口中，点击**运行**。
12. 点击 “响应 ‘磨损’” 红色小三角并选择**估计值** > **显示预测表达式**。

图 A.18 带编码因子的模型的预测表达式

使用编码因子的系数可以比较效应大小。在设计设置范围内，**硅烷**对**磨损**的影响最大。**硅石**和**硅烷 * 硫磺**交互作用的影响也很大。

编码因子的系数也更容易解释。例如，当所有因子处于范围中心时，**磨损**的预测值为截距，即 139.12。

13. 关闭 Tiretread.jmp 样本数据表而不保存更改。

混料

当数据表中的某个列表示混料的成分时，“混料”列属性很有用。混料成分之和被限制为一个常数。因此，它们与非混料因子不同。“混料”列属性有两个用途：

- 它将列标识为混料成分。

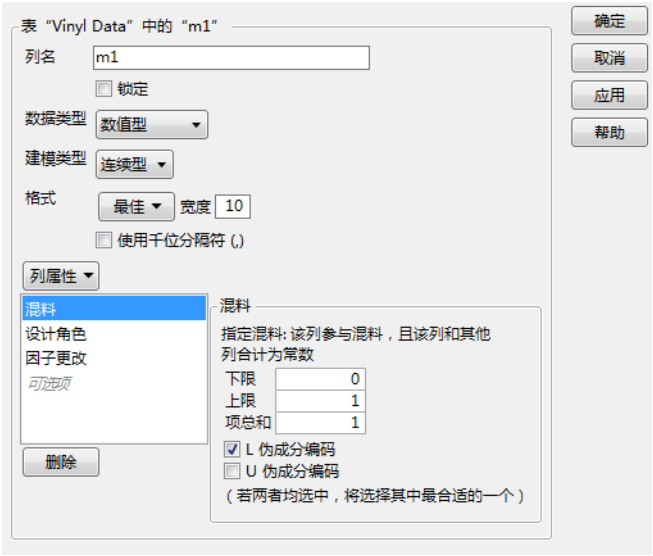
  若您在“分析 > 拟合模型”窗口中将具有“混料”列属性的列添加为模型效应，JMP 会自动生成一个没有截距的模型。

- 它定义混料成分的编码。

  混料成分的编码不同于非混料因子的编码。但是，和非混料因子一样，混料因子的编码可以帮助您解释参数估计值。请参见第 716 页的 “伪成分编码”。

图 A.19 显示 Design Experiment 文件夹下 Vinyl Data.jmp 样本数据表中因子 m1 的“混料”列属性面板。

图 A.19 “混料”列属性面板



在“混料”列属性面板中，可以指定以下内容：

下限 指定在伪成分编码中使用的低值。“混料”属性作为实验设计平台中设计构造的一部分应用时，“下限”设置为因子的最小设置。当您将“混料”属性应用到不包含该属性的列时，JMP 插入最小值作为“下限”。您可以按需更改该值。

上限 指定在伪成分编码中使用的高值。“混料”属性作为实验设计平台中设计构造的一部分应用时，“上限”设置为因子的最大设置。当您将 “混料”属性应用到不包含该属性的列时，JMP 插入最大值作为 “上限”。您可以按需更改该值。

项总和 指定混料成分之和。当您将“混料”属性应用到不包含该属性的列时，JMP 为“项总和”插入默认值 1。

L 伪成分编码 变换数据值使得 “下限”对应于 0。

U 伪成分编码 变换数据值使得 “上限”对应于 0。

伪成分编码

伪成分是线性变换。用 S 表示混料成分之和。假定给 i 个列 X_1、X_2、…、X_q 分配了“混料”列属性。假定列和根据这些列构造的效应在“拟合模型”窗口中作为效应输入。

定义以下公式：

$$L = \sum_{i=1}^{q} L_i$$，其中 L_i 是 X_i 的下限

$$U = \sum_{i=1}^{q} U_i$$，其中 U_i 是 X_i 的上限

用 x_i 表示列 X_i 的值。按以下方式定义 x_i 处的 L 伪成分：

$$x_i^L = (X_i - L_i)/(S - L)$$

按以下方式定义 x_i 处的 U 伪成分：

$$x_i^U = (U_i - X_i)/(U - S)$$

若您同时选择 **L 伪成分编码**和 **U 伪成分编码**，若 $(S - L) < (U - S)$，“拟合模型”平台将使用 L 编码 。否则使用 U 编码。

在“拟合模型”中，计算参数估计值前会使用伪成分变换混料因子。这可以使参数估计值更有意义。在处理参数估计值的报表中，通过编码变换给出混料主效应。其他报表（如刻画器）基于未编码的值。

混料示例

Design Experiment 文件夹下的 Donev Mixture Data.jmp 样本数据表中的数据基于 Atkinson 和 Donev (1992) 的示例。该设计包括三个混料因子和一个非混料因子。

- 响应是丙烯腈粉的电磁**阻尼**。
- 三个混料成分为硫酸铜 (CuSO4)、亚硫酸钠 (Na2S2O3) 和**乙二醛**。

- 关注的非混料环境因子为光的**波长**。

尽管**波长**在理论上是连续的，研究人员只关注三个离散波长处的预测值。因此，**波长**被视为具有三个水平的分类因子。

有关使用“定制设计”针对该情形构造设计的详细信息，请参见“定制设计示例”一章中第 160 页的 “混料实验”。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Donev Mixture Data.jmp。
2. 点击列面板中**硫酸铜 (CuSO4)** 旁边的星号，然后选择**混料**。

图 A.20 硫酸铜 (CuSO4) 的 “混料”列属性面板

请注意以下事项：

- “下限”为 0.2，即**硫酸铜 (CuSO4)** 的最小设计设置。
- “上限”为 0.8，即**硫酸铜 (CuSO4)** 的最大设计设置。
- “项总和”设置为 1。这是三个混料因子之和。
- 已选择 **L 伪成分编码**选项。请参见第 716 页的“伪成分编码”。

3. 点击**取消**。
4. 点击列面板中**乙二醛**旁边的星号，然后选择**混料**。

   对于该因子，请注意以下事项：
   - “下限”为 0，即**乙二醛**的最小设计设置。
   - “上限”为 0.6，即**乙二醛**的最大设计设置。
5. 点击**取消**。

6. 在 Donev Mixture Data.jmp 数据表中，点击**模型**脚本旁边的绿色小三角。

   模型包含混料因子的主效应和所有四个因子的双因子交互作用。

7. 点击**运行**。

   在 “参数估计值”报表中，混料因子以伪成分编码形式出现。混料因子出现在交互作用中时，它们不用编码形式表示。但是，基于伪成分拟合模型。“参数估计值”报表（图 A.21）中的前三项显示混料因子的编码形式。

图 A.21 “参数估计值”报表

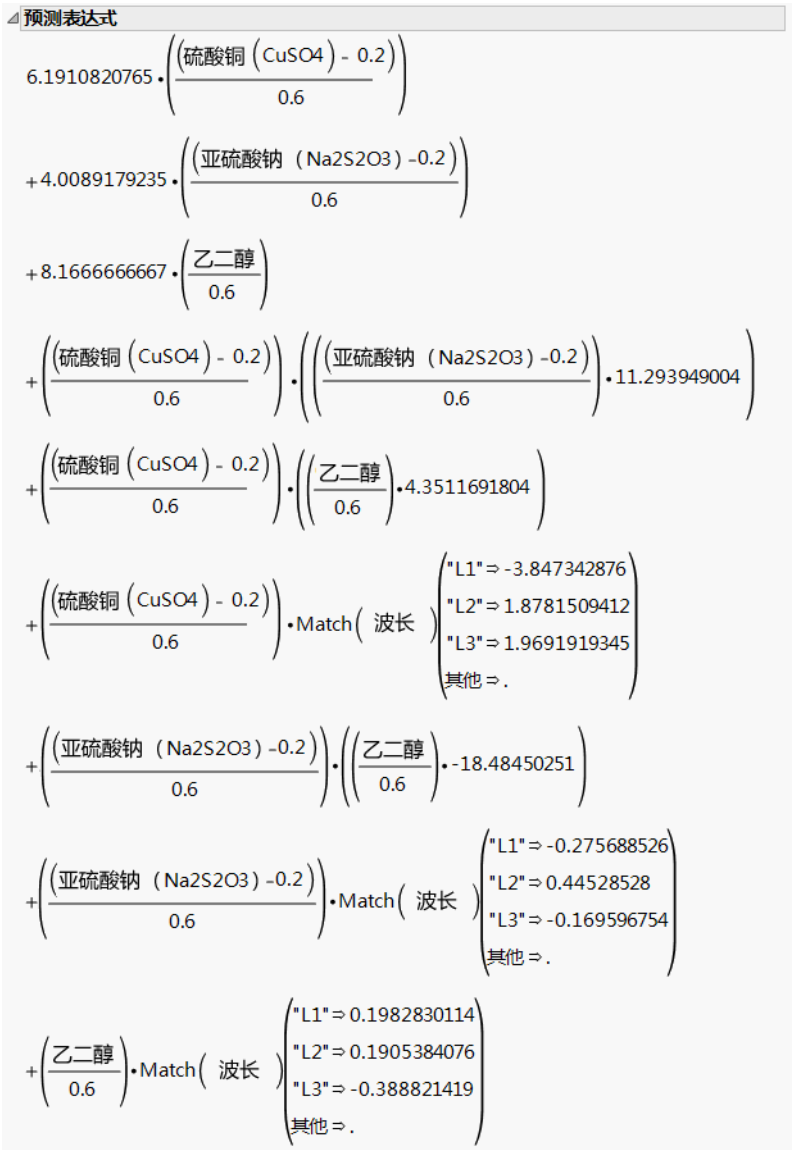
参数估计值

项	估计值	标准误差	t 比	概率>|t|
(硫酸铜 (CuSO4)-0.2)/0.6	6.1910821	0.918805	6.74	0.0005*
(亚硫酸钠 (Na2S2O3)-0.2)/0.6	4.0089179	0.918805	4.36	0.0048*
乙二醛/0.6	8.1666667	0.921638	8.86	0.0001*
硫酸铜 (CuSO4)*亚硫酸钠 (Na2S2O3)	11.293949	4.728922	2.39	0.0542
硫酸铜 (CuSO4)*乙二醛	4.3511692	4.512775	0.96	0.3722
硫酸铜 (CuSO4)*波长[L1]	-3.847343	1.113399	-3.46	0.0135*
硫酸铜 (CuSO4)*波长[L2]	1.8781509	1.079113	1.74	0.1324
亚硫酸钠 (Na2S2O3)*乙二醛	-18.4845	4.512775	-4.10	0.0064*
亚硫酸钠 (Na2S2O3)*波长[L1]	-0.275689	1.064004	-0.26	0.8042
亚硫酸钠 (Na2S2O3)*波长[L2]	0.4452853	1.103681	0.40	0.7006
乙二醛*波长[L1]	0.198283	1.090627	0.18	0.8617
乙二醛*波长[L2]	0.1905384	1.090647	0.17	0.8671

8. 点击 “响应 ‘阻尼’”红色小三角并选择**估计值 > 显示预测表达式**。

   “预测表达式”报表显示拟合的模型。请注意混料因子使用 L 伪成分编码进行了变换。

图 A.22 “阻尼”模型的预测表达式



假定您关注**波长** L2 处的预测值。还假定**亚硫酸钠** (Na2S2O3) 和**乙二醛**分别设置为其低值 0.2 和 0，**硫酸铜** (CuSO4) 设置为其高值 0.8。在这种情况下，预测的**阻尼**等于**硫酸铜** (CuSO4) 的参数估计值 (6.191) 加上**硫酸铜** (CuSO4)* **波长** [L2] 的参数估计值 (1.878)。您可以在“预测刻画器”中对此进行验证。

9. 点击“响应 ‘阻尼’”红色小三角并选择**保存列** > **保存编码表**。

图 A.23 显示出编码混料因子的编码表的前三列

	[硫酸铜 (CuSO4)-0.2]/0.6	[亚硫酸钠 (Na2S2O3)-0.2]/0.6	乙二醛/0.6
1	1	0	0
2	0.4	0.6	0
3	0	1	0
4	0.5	0	0.5
5	0	0.5	0.5
6	0	0	1
7	1	0	0
8	0.6	0.4	0
9	0	1	0
10	0.5	0	0.5
11	0	0.5	0.5
12	0	0	1
13	1	0	0
14	0.4	0.6	0
15	0	1	0
16	0.5	0	0.5
17	0	0.5	0.5
18	0	0	1

对于这个特定设计，“L 伪成分编码”将混料因子变换到 0-1 的范围。请注意这通常不会发生。

因子更改

“因子更改”列属性指示在设计的实验中更改因子设置的难度。“因子更改”的可能规格为“简单”、“困难”和“极难”。例如，图 A.24 显示 Design Experiment 文件夹下 Battery Data.jmp 样本数据表中因子 A1 的“因子更改”列属性面板。

图 A.24 “因子更改”列属性面板

- 设计包含难以更改和极难更改的因子时，还必须包含子区和整区因子：

  - 整区因子的水平定义将极难更改的因子水平保持为常数的试验组。
  - 子区因子的水平定义将难以更改的因子水平保持为常数的试验组。
- 设计仅包含难以更改的因子而不包含极难更改的因子时，应包括整区因子。整区因子的水平定义将极难更改的因子水平保持为常数的试验组。请参见“定制设计”一章中第 86 页的“更改和随机区组”。

扩充和评估设计

对于“评估设计”和“扩充设计”平台，“因子更改”列属性标识指定为“困难”或“极难”更改的因子。但是，这些平台还要求在启动窗口中将整区和子区因子作为模型效应输入。这是因为整区和子区因子是设计结构的一部分。

定制设计

“定制设计”平台允许您创建所有因子更改均为“简单”的设计。您还可以构造裂区、裂-裂区或双向裂区（带区）设计。构造这些设计时，您需要标识其值难以更改或极难更改的因子。在“因子”分级显示项中，您可以将因子标识为“简单”、“困难”或“极难”更改。“定制设计”平台构造设计表时，“因子更改”属性会分配给“因子”分级显示项中显示的每个因子。

“定制设计”平台是构造“困难”或“极难”更改因子的设计的唯一平台。其他实验设计平台也将“因子更改”列属性分配给自己构造的因子，但是因子的列属性值设置为“简单”。

若您使用因子表在“定制设计”窗口中加载因子，可以将“因子更改”列属性分配给该表中的列。使用该表加载因子时，“因子更改”规格会显示在“因子”分级显示项中。

因子更改示例

Design Experiment 文件夹下的 Battery Data.jmp 样本数据表包含研究电池的开路电压 (OCV) 的实验数据。该设计是双向裂区设计。有关更多背景知识，请参见第 181 页的“定制设计示例”一章。

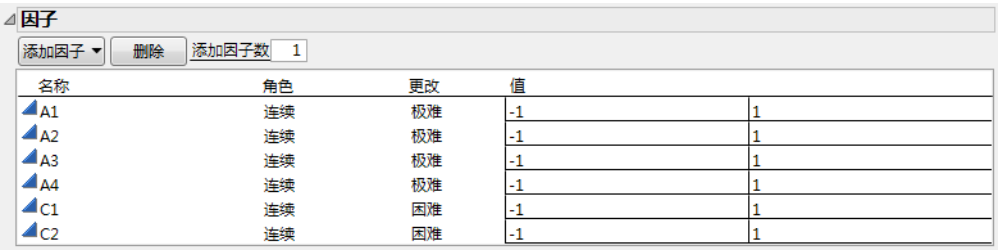
1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Battery Data.jmp。
2. 点击列面板中因子 C1 右侧的星号。
3. 选择**因子更改**。

图 A.25 C1 的 “因子更改” 面板

C1 的 “因子更改” 值为 “困难”。图 A.24 显示 A1 的 “因子更改” 值为 “极难”。

4. 点击**确定**。
5. 在数据表中，点击**实验设计对话框**脚本旁边的绿色小三角。
6. 打开**因子**分级显示项。

图 A.26 电池实验的 “因子” 分级显示项



名称	角色	更改	值	
A1	连续	极难	-1	1
A2	连续	极难	-1	1
A3	连续	极难	-1	1
A4	连续	极难	-1	1
C1	连续	困难	-1	1
C2	连续	困难	-1	1

因子 A1、A2、A3 和 A4 具有设置为 “极难” 的更改，因子 C1 和 C2 具有设置为 “困难” 的更改。“定制设计” 平台构造设计表时，它将这些规格作为 “因子更改” 列属性保存到相应的列。

在 “设计” 分级显示项中，注意**整区**和**子区**因子。

图 A.27 “设计”分级显示项部分视图

设计

试验	整区	子区	A1	A2	A3	A4	C1	C2
1	1	1	-1	-1	1	-1	-1	-1
2	1	2	-1	-1	1	-1	1	-1
3	1	3	-1	-1	1	-1	-1	1
4	2	4	-1	1	1	1	1	1
5	2	5	-1	1	1	1	-1	1
6	2	6	-1	1	1	1	1	-1
7	3	7	1	1	1	-1	-1	1
8	3	8	1	1	1	-1	1	1
9	3	9	1	1	1	-1	-1	-1
10	4	10	1	-1	1	1	1	1
11	4	11	1	-1	1	1	1	-1
12	4	12	1	-1	1	1	-1	-1

为了解释“困难”和“极难”因子更改，“定制设计”平台创建了两个因子。**整区**因子将“极难”因子水平为常数的试验分在一组，**子区**因子则将“困难”因子水平为常数的试验分在一组。当您在“评估设计”和“扩充设计”平台中输入具有“因子更改”列属性的列时，这些因子需要作为模型效应包括进来。

值顺序

“值顺序”列属性将顺序分配给列中的值。该顺序之后会在图和分析中使用。您可以指定希望值在报表中出现的顺序。

注意：对于具有自然排序的一些值（如星期几），JMP 会自动在报表中以合适的方式对它们排序。请参见《使用 JMP》。

图 A.28 显示 Car Physical Data.jmp 样本数据表中**车型**列的“值顺序”面板。涉及**车型**值的报表按以下顺序放置这些水平：Sporty、Small、Compact、Medium 和 Large。使用“值顺序”列表右侧的按钮可指定您希望的值顺序。

图 A.28 “车型”的“值顺序”列属性

在使用大多数实验设计平台创建的设计中，会为分类因子（包括构造的“整区”和“子区”因子）以及分区组因子分配“值顺序”属性。该属性根据水平在“因子”分级显示项中显示的顺序对水平排序。构造的因子水平是连续整数，“值顺序”属性指定该自然排序。您可以修改任何因子的“值顺序”规格来满足自己的需要。

“覆盖阵列”或“田口数组”平台不分配“值顺序”属性。“覆盖阵列”平台分配“值标签”列属性。请参见第 727 页的“值标签”。

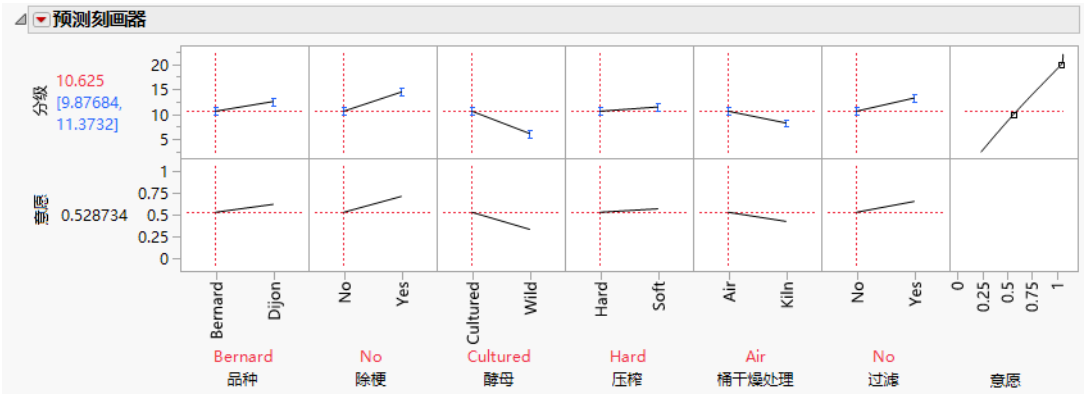
值顺序示例

假定您希望因子的值以不同顺序出现在“预测刻画器”中。考虑使用“定制设计”构造的葡萄酒品尝实验的示例。由五位专家对葡萄酒进行评级，每位专家作为 Wine Data.jmp 样本数据表中的**评级员**列出。**评级员**是固定分区组因子。研究了九个因子。**分级**是响应。

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Wine Data.jmp。
2. 在“表”面板中，点击**简化模型**脚本旁边的绿色小三角。
3. 点击**运行**。

   “预测刻画器”显示在报表底部。

图 A.29 具有原始值顺序的刻画器

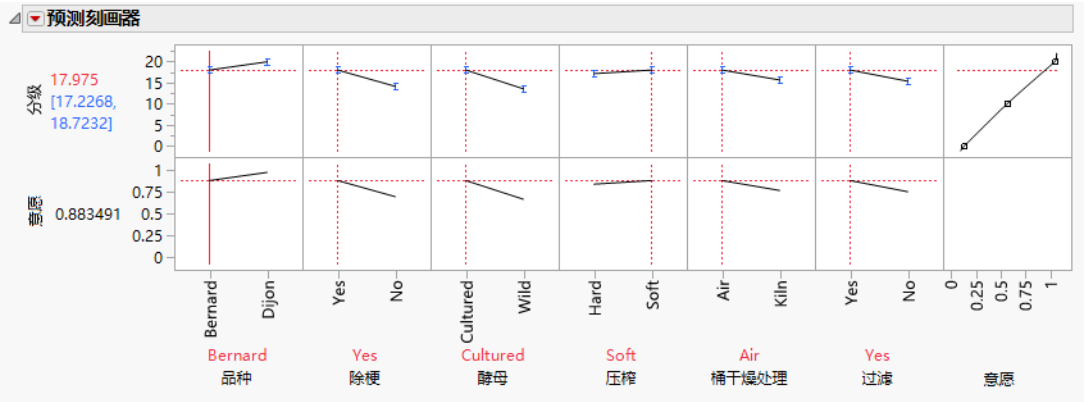


请注意**除梗**和**过滤**的值以“No”后跟“Yes”的顺序显示。您想反转顺序，以便“Yes”水平首先出现。

提示：要排列刻画器，请点击“预测刻画器”红色小三角并选择**外观 > 行内排列**。为本例输入 7 并点击**确定**。

4. 关闭“响应 ‘分级’”报表。
5. 在数据表中，点击列面板中**除梗**旁边的星号，然后选择“值顺序”。
6. 点击**反转**。
7. 点击**确定**。
8. 点击列面板中**过滤**旁边的星号，然后选择“值顺序”。
9. 点击**反转**。
10. 点击**确定**。
11. 再次点击**简化模型**脚本旁边的绿色小三角。
12. 点击**运行**。

图 A.30 具有新的值顺序的刻画器



除梗和**过滤**的水平现在以 “Yes” 后跟 “No” 的顺序显示。

13. 关闭 Wine Data.jmp 样本数据表而不保存更改。

指定值顺序

考虑 Candy Bars.jmp 样本数据表。在**品牌**下的 18 个品牌列表中，Hershey 和 M&M/Mars 的糖果品种数最多，如**名称**列中所列。您希望这两个品牌首先显示在报表中。

1. 选择**帮助** > **样本数据库**，然后打开 Candy Bars.jmp。
2. 选择**品牌**列。
3. 选择**列** > **列信息**。
4. 在 “列属性” 下，选择 “值顺序”。
5. 在 “值顺序” 列表中，选择 “Hershey”。
6. 点击**向上箭头**五次。

   Hershey 现在位于 “值顺序” 列表中的第一位。
7. 在 “值顺序” 列表中，选择 “M&M/Mars”。
8. 点击**向上箭头**七次。

   M&M/Mars 现在位于 “值顺序” 列表中的第二位。
9. 点击**确定**。

   指示 “值顺序” 列属性的星号出现在列面板中的**品牌**旁边。 JMP 现在在涉及**品牌**的报表中首先列出 Hershey 和 M&M/Mars。
10. 选择**分析** > **分布**。
11. 选择**卡路里**并点击 **Y，列**。
12. 选择**品牌**，然后点击**依据**。

13. 点击**确定**。
14. 按 Ctrl 键，点击 “卡路里”红色小三角，然后选择**显示选项 > 水平布局**。

    请注意， Hershey 和 M&M/Mars 对应的 “分布”报表在 18 个品牌中最先显示。
15. 关闭 Candy Bars.jmp 样本数据表而不保存更改。

值标签

“值标签”列属性代表列中具有指定标签的值。这些标签显示在数据表中，并用在图和报表中。在数据表中，您可以通过在单元格内双击来查看原始值。有关如何分配和使用“值标签”列属性的详细信息，请参见《使用 JMP》。

“覆盖阵列”平台是分配“值标签”列属性的唯一实验设计平台。“覆盖阵列”平台将因子保存到具有“名义型”建模类型的数据表。基础值是连续整数，介于 1 和您在覆盖阵列“因子”分级显示项中指定的水平数之间。您在“因子”分级显示项中指定的值是分配给基础整数的“值标签”。

值标签示例

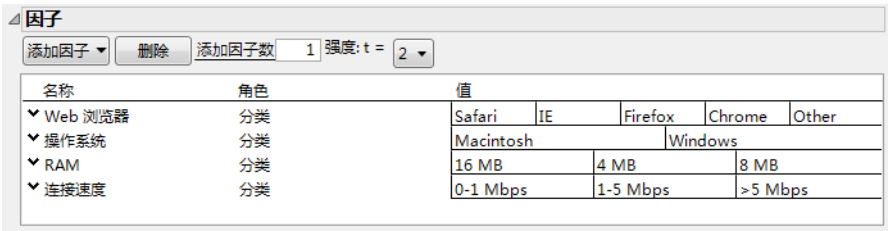
您要检验基于 Internet 的软件应用程序来检测操作环境中组件导致的问题。关注的四个组件为浏览器、操作系统、计算机 RAM 和连接速度。为了最小化检验时间，您限制自己只检验双因子交互作用。

构造强度为 2 的覆盖阵列来检验所需的因子水平组合。

1. 选择**实验设计 > 特殊目的 > 覆盖阵列**。
2. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Software Factors.jmp。
3. 点击 “覆盖阵列”红色小三角并选择**加载因子**。

   因子及其水平显示在 “因子”分级显示项中。

图 A.31 软件因子的 “因子”分级显示项



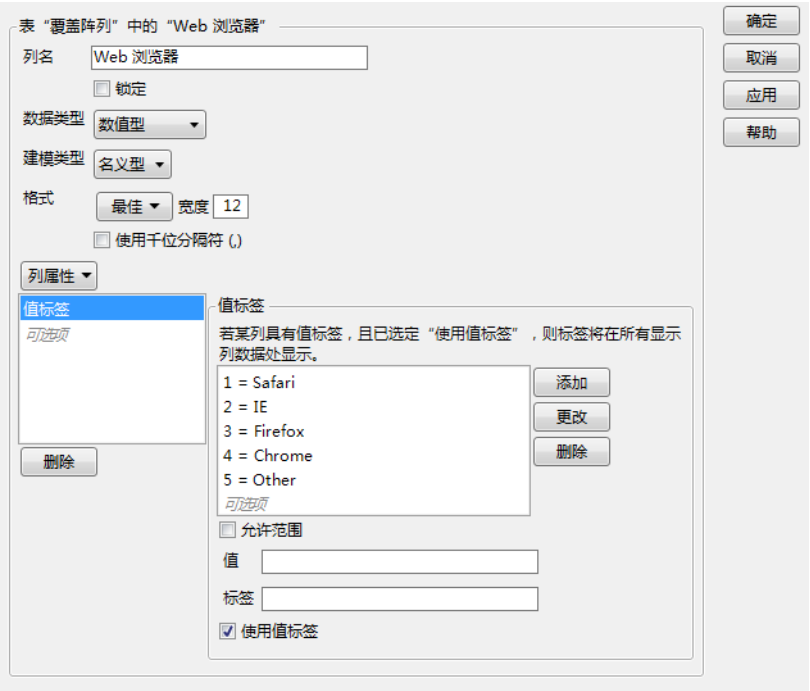
请注意四个因子的角色描述为 “分类”。

4. 点击**继续**。

“限制因子水平组合”分级显示项随即打开。因为设置的所有组合均有可能，请将该项设置为**无**。

5. 点击**制作设计**。
6. 点击**制表**。
7. 在列面板中，点击 Web 浏览器旁边的星号并选择**值标签**。

图 A.32 因子 A 的 “列信息”窗口

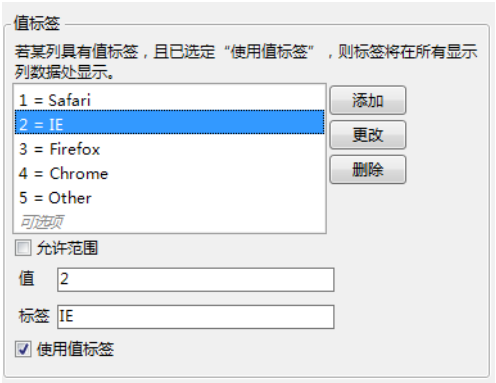


请注意 Web 浏览器具有 “数值型”数据类型和 “名义型”建模类型。 Web 浏览器的基础值是整数 1、 2、 3、 4 和 5。当您构造设计时，根据您输入的值为这些值分配值标签。

将 2 的值标签从 IE 改为 Internet Explorer。

8. 在 “值标签”列表中选择 2 = IE。

图 A.33 进行了选择的 “值标签” 面板



9. 在 “标签” 旁边键入 Internet Explorer。
10. 点击**更改**。

    更改将反映在数据表中。

注意：要使用数值而非标签，请取消选择**使用值标签**。

每区组试验次数

当您使用实验设计平台构造包含分区组因子的设计时，该因子会被分配带有 “分区组” 值的 “设计角色” 列属性。JMP 还给每个 “分区组” 因子分配 “每区组试验次数” 列属性。 “每区组试验次数” 属性指示每个区组中允许的最大试验次数。 “评估设计” 平台和 “扩充设计” 平台使用该属性来指示因子的分区组结构。请参见 “定制设计” 一章中第 85 页的 “分区组” 。

注意：JMP 分配 “每区组试验次数” 列属性作为设计构造的一部分。您无法直接分配该列属性。

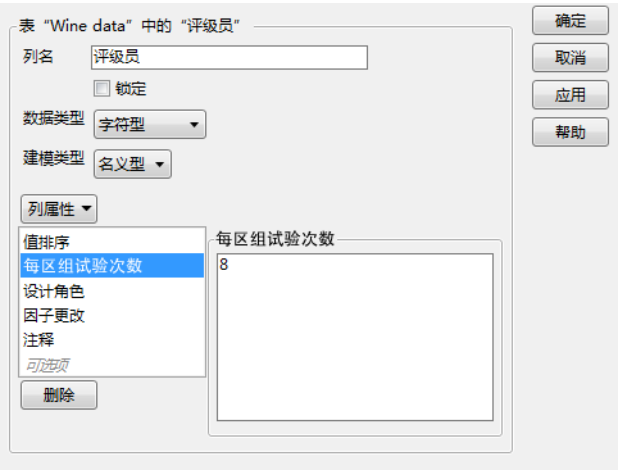
每区组试验次数示例

考虑 “定制设计” 一章中第 66 页的 “定制设计示例” 中所述的葡萄酒品尝实验。葡萄酒样本由五位评级员 （**评级员**）品尝，每位评级员品尝八个样本。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Wine Data.jmp。
2. 点击列面板中**评级员**旁边的星号，然后选择**设计角色**。

   请注意 “设计角色” 设置为 “分区组”。
3. 点击**取消**。
4. 点击列面板中**评级员**旁边的星号，然后选择**每区组试验次数**。

图 A.34 评级员的 “每区组试验次数”列属性面板



请注意 “每区组试验次数”的值为 8。“实验设计对话框”脚本构造的设计有 40 次试验。因为有五位评级员， JMP 为每位评级员构造了包含 40/5 = 8 次试验的设计。

约束状态

在 “定制设计”和 “混料设计”平台中，您可以保存为设计指定的约束。选择 “保存约束”时，每个线性约束的系数会显示在数据表的列中。在表的最后一行给出限制不等式的值。

“约束状态”列属性分配给每个约束列。该属性指定定义约束的不等式方向。从设计平台选择 “加载约束”时， “约束状态”列属性会告知 JMP 不等式的方向。

注意：JMP 分配 “约束状态”列属性作为设计构造的一部分。您无法直接分配该列属性。

约束状态示例

Design Experiment 文件夹中的样本数据表 Piepel.jmp 包含具有三个连续因子的混料设计。该设计基于 Snee (1979) 和 Piepel (1988) 提出的实验，其中每个因子有边界约束，还有三个其他线性约束。在本例中，您需要执行以下操作：

1. 更改三个其他约束之一
2. 将约束保存到表
3. 观察 “约束状态”列属性如何描述约束中不等式的方向

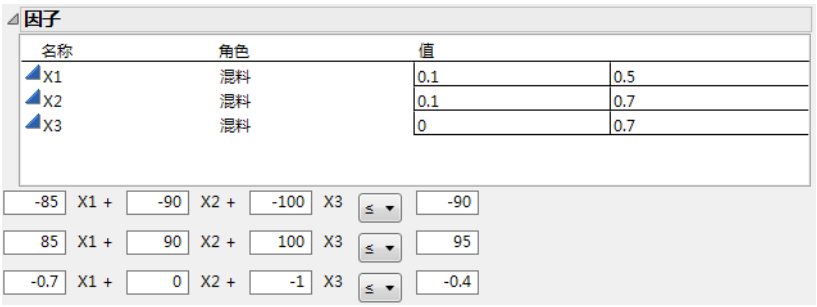
有关该示例的进一步发展，请参见 “混料设计”一章中第 368 页的 “包含线性约束的极端顶点示例”。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Piepel.jmp。

2. 在 “表”面板中，点击**实验设计对话框**脚本旁边的绿色小三角。

   请注意 “因子”分级显示项下的三个线性约束。要使约束更容易解释，您想根据 “大于或等于”不等式重新用公式表示第一个约束。

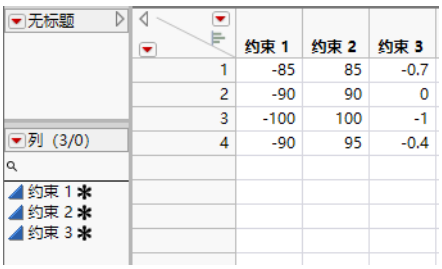
图 A.35 “因子”分级显示项下的线性约束



3. 在第一个约束中，执行以下操作：
   - 在 X1 旁边键入 85。
   - 在 X2 旁边键入 90。
   - 在 X3 旁边键入 100。
   - 从不等式菜单中选择 ≥。
   - 在不等式符号右侧键入 90。
4. 点击 “混料设计”红色小三角并选择**保存约束**。

   随即显示包含有关约束信息的表。

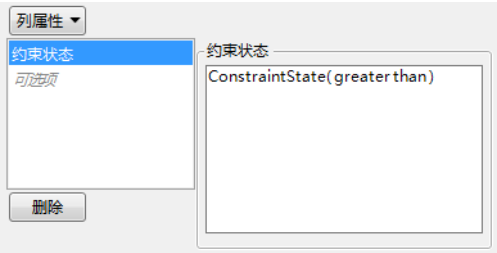
图 A.36 约束表



每列在第 1-3 行包含因子 X1、 X2 和 X3 的系数。第 4 行包含不等式符号右侧显示的值。

5. 点击**约束 1** 旁边的星号。
6. 点击**约束状态**。

图 A.37 “约束状态”列属性面板



X1 的“约束状态”面板指示不等式的方向为“大于”，即大于或等于 ≥。

7. 点击**取消**。

附录 B

技术细节

本附录包含有关别名矩阵、功效计算和相对预测方差的信息。

- 第 735 页的“别名矩阵”
- 第 736 页的“功效计算”
- 第 738 页的“相对预测方差”

目录

别名矩阵 735
具有“困难”或“极难”因子更改的设计 735
具有“若可能”效应的设计 735
功效计算 736
单个参数的功效 736
分类效应的功效 737
相对预测方差 738

别名矩阵

“别名矩阵”条目表示您在“别名项”分级显示项中指定的效应传导给模型参数的偏倚度。别名矩阵也用于定义别名最优性。请参见“定制设计”一章中第 118 页的“别名最优性”。

别名矩阵的计算基于模型矩阵。请参见“定制设计”一章中第 103 页的“模型矩阵”。

让 $\mathbf{X}_1$ 作为与“模型”分级显示框中的项对应的模型矩阵。用 $\mathbf{X}_2$ 表示“别名项”分级显示框中指定的效应的模型项矩阵。

按以下方式给定假设模型：

$$\mathbf{Y} = \mathbf{X}_1\beta_1 + \varepsilon$$

假定一些别名项是活跃的，并且按以下方式给定真实模型：

$$\mathbf{Y} = \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta_2 + \varepsilon$$

给定 β_1 的最小二乘估计量如下所示：

$$\hat{\beta}_1 = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{Y}$$

在通常的回归假设下，给定 $\hat{\beta}_1$ 的期望值如下所示：

$$E[\hat{\beta}_1] = \beta_1 + \mathbf{A}\beta_2$$

其中 $\mathbf{A} = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2$

矩阵 **A** 称为**别名矩阵**。

具有“困难”或“极难”因子更改的设计

对于具有难以更改的因子或极难更改的因子的设计，按以下方式给定别名矩阵：

$$\mathbf{A} = (\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'(\mathbf{V}^{-1}\mathbf{X}_2)$$

其中 **V** 是响应的区组对角协方差矩阵。

具有“若可能”效应的设计

对于具有 “若可能”效应的设计 （Bayesian D 和 I 最优设计），按以下方式给定别名矩阵：

$$\mathbf{A} = (\mathbf{X}_1'\mathbf{X}_1 + \mathbf{K}^2)^{-1}\mathbf{X}_1'\mathbf{X}_2$$

其中 **K** 是具有以下值的对角矩阵：

- $k = 0$ 代表“必需”项

- $k = 1$ 代表涉及含有两个以上水平的分类因子的“若可能”主效应、幂和交互作用
- $k = 4$ 代表其他所有“若可能”项

在 Bayesian 情形中，别名矩阵给出对应于带有先验方差 $\mathbf{K}^{-1}$ 的岭回归的效应别名程度。有关 Bayesian 设计的其他详细信息，请参见“定制设计”一章中第 115 页的“Bayesian D 最优性”和“定制设计”一章中第 117 页的“Bayesian I 最优性”。

在“定制设计”平台中，您可以通过从红色小三角菜单中选择**高级选项 > 先验参数方差**来控制用于“若可能”项的权重。在此处您可以通过指定 **K** 的对角线元素为所有模型项设置先验方差。当您点击“制作设计”时，该选项随之更新以显示默认权重。

功效计算

“功效分析”报表给出这些情况的功效计算：单个参数值以及整体效应（当设计包含具有三个或更多水平的分类因子时）。本节介绍以下两种情形的计算：

- 第 736 页的“单个参数的功效”
- 第 737 页的“分类效应的功效”

单个参数的功效

本节说明如何计算单个参数的检验功效。使用以下符号：

X 模型矩阵。有关名义型效应的编码的信息，请参见《拟合线性模型》。另请参见“定制设计”一章中第 103 页的“模型矩阵”。

注意：您可以通过运行“拟合模型”来查看模型矩阵。然后从主报表的红色小三角菜单选择**保存列 > 保存编码表**。

β_t 关注项所对应的参数。

$\hat{\beta}_i$ β_i 的最小二乘估计量

β_i^A 预期系数值。要检测的差值是 $2\beta_i^A$。

$\hat{\beta}_i$ 的方差由 $\sigma^2(\mathbf{X}'\mathbf{X})^{-1}$ 的第 i 个对角线元素给出，其中 σ^2 为误差方差。用 $(\mathbf{X}'\mathbf{X})_{ii}^{-1}$ 表示 $(\mathbf{X}'\mathbf{X})^{-1}$ 的第 i 个对角线元素。

误差方差 σ^2 由均方误差估计得出，具有 $n - p - 1$ 个自由度，其中：n 是观测数，p 是模型中除截距之外的项数。若 $n - p - 1 = 0$，则 JMP 将误差的自由度设置为 1。这允许在饱和设计中估计参数的功效。

H_0:$\beta_i = 0$ 检验表示为：

$$\frac{\hat{\beta}_i}{\sqrt{MSE(\mathbf{X}'\mathbf{X})^{-1}_{ii}}}$$

或等效表示为：

$$F_0 = \frac{\hat{\beta}_i^2}{MSE(\mathbf{X}'\mathbf{X})^{-1}_{ii}}$$

在原假设下，检验统计量 F_0 具有自由度为 *1* 和 $n-p-1$ 的 *F* 分布。

若 β_i 的真实值为 β_i^A，则 F_0 具有非中心 *F* 分布，其非中心参数由以下公式给出：

$$\lambda = \frac{(\beta_i^A)^2}{\sigma^2(\mathbf{X}'\mathbf{X})^{-1}_{ii}}$$

要计算检验功效，首先要求解 α 水平临界值 F_c：

$$\alpha = 1 - FDist(F_c, 1, n-p-1)$$

然后按照如下方式计算功效：

$$\text{功效} = 1 - FDist(F_c, 1, n-p-1, \lambda)$$

分类效应的功效

本节说明如何计算整个分类效应的检验功效。使用以下符号：

X 模型矩阵。请参见 “技术细节”附录中第 735 页的 “别名矩阵”。

β 参数的向量。

$\hat{\beta}$ β 的最小二乘估计值。

β^A 预期系数值的向量。

L 定义分类效应的检验的矩阵。矩阵 **L** 标识 β 中对应于分类效应的参数值并将它们设置为 0。分类效应检验的原假设为：

$$H_0: L\beta = \mathbf{0}$$

r **L** 的秩。或者， *r* 为分类效应的水平数减去 1。

注意：您可以通过运行 “拟合模型”来查看设计矩阵。然后从主报表的红色小三角菜单选择**保存列 > 保存编码表**。

$\hat{\beta}$ 的协方差矩阵由 $\sigma^2(\mathbf{X}'\mathbf{X})^{-1}$ 给出，其中 σ^2 为误差方差。

误差方差 σ^2 由均方误差估计得出，具有 $n-p-1$ 个自由度，其中：n 是观测数，p 是模型中除截距之外的项数。若 $n-p-1=0$，则 JMP 将误差的自由度设置为 1。这允许在饱和设计中估计参数的功效。

$H_0: \mathrm{L}\beta = \mathbf{0}$ 检验表示为：

$$F_0 = \left((\mathbf{L}\hat{\beta})'[\mathbf{L}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{L}']^{-1}(\mathbf{L}\hat{\beta})\right)/(rMSE)$$

在原假设下，检验统计量 F_0 具有自由度为 r 和 $n-p-1$ 的 F 分布。

若 β 的真实值为 β^A，则 F_0 具有非中心 F 分布，其非中心参数由以下公式给出：

$$\lambda = \left((\mathbf{L}\beta^A)'[\mathbf{L}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{L}']^{-1}(\mathbf{L}\beta^A)\right)/\sigma^2$$

要计算检验功效，首先要求解 α 水平临界值 F_c：

$$\alpha = 1 - FDist(F_c, r, n-p-1)$$

然后按照如下方式计算功效：

$$\text{功效} = 1 - FDist(F_c, r, n-p-1, \lambda)$$

相对预测方差

考虑以下符号：

X 模型矩阵。请参见“定制设计”一章中第 103 页的“模型矩阵”。定制设计提供显示该模型矩阵的脚本。请参见“定制设计”一章中第 103 页的“保存 X 矩阵”。

σ^2 误差方差。

$\hat{\beta}$ 参数的最小二乘估计值的向量。

$\mathbf{x}_i'$ **X** 的第 i 行。

使用该符号，**X** 的第 i 行的预测响应表示为：

$$\hat{\mathbf{Y}} = \mathbf{x}_i'\hat{\beta}$$

由 $\mathbf{x}_i$ 所定义的设置处的相对预测方差为：

$$\frac{\mathbf{x}_i' var(\hat{\mathbf{Y}})\mathbf{x}_i}{\sigma^2} = \frac{\mathbf{x}_i' var(\mathbf{x}_i'\hat{\beta})\mathbf{x}_i}{\sigma^2} = \mathbf{x}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i$$

附录 C

参考书目

Agresti, A., and Coull, B. A. (1998).“Approximate Is Better than ‘Exact’ for Interval Estimation of Binomial Proportions.”*American Statistician* 52:119–126.

Agresti, A., and Caffo, B. N. (2000).“Simple and Effective Confidence Intervals for Proportions and Differences of Proportions Result from Adding Two Successes and Two Failures.”*American Statistician* 54:280–288.

Atkinson, A. C., and Donev, A. N.(1992).*Optimum Experimental Designs*.New York: Oxford University Press.

Barker, C. (2011).“Power and Sample Size Calculations in JMP.”Technical Report, Cary NC: SAS Institute Inc. Accessed May 6, 2018. https://www.jmp.com/content/dam/jmp/documents/en/technical-reports/powerAndSampleSize.pdf.

Borror, C. M., and Montgomery, D. C. (2000).“Mixed Resolution Designs as Alternatives to Taguchi Inner-Outer Array Designs for Robust Design Problems.”Quality and Reliability Engineering International 16:117–128.

Box, G. E. P. (1988).“Signal–to–Noise Ratio, Performance Criteria, and Transformations.”*Technometrics* 30:1–40.

Box, G. E. P., and Behnken, D. W. (1960).“Some New Three-Level Designs for the Study of Quantitative Variables.”*Technometrics* 2:455–475.

Box, G. E. P., and Draper, N. R. (1987).*Empirical Model–Building and Response Surfaces*.New York: John Wiley & Sons.

Box, G. E. P., Hunter, W. G., and Hunter, J. S. (1978).*Statistics for Experimenters*.New York: John Wiley & Sons.

Box, G. E. P., Hunter, W. G., and Hunter, J. S. (2005).*Statistics for Experimenters: Design, Innovation, and Discovery*.2nd ed. New York: John Wiley & Sons.

Box, G. E. P., and Meyer, R. D. (1986).“An Analysis of Unreplicated Fractional Factorials.”*Technometrics* 28:11–18.

Box, G. E. P., and Wilson, K. B. (1951).“On the Experimental Attainment of Optimum Conditions.”*Journal of the Royal Statistical Society, Series B* 13:1–45.

Byrne, D. M., and Taguchi, G. (1986).“The Taguchi Approach to Parameter Designs.”*Quality Congress Transactions* 177:168–177.

Chow, S. -C., Shao, J., and Wang, H. (2008).*Sample Size Calculations in Clinical Research*.2nd ed. Boac Raton, FL: Chapman & Hall/CRC.

Clopper, C. J., and Pearson, E. S. (1934).“The Use of Confidence or Fiducial Limits Illustrated in the Case of the Binomial.”*Biometrika* 26:404–413.

Cochran, W. G., and Cox, G. M. (1957).*Experimental Designs*.2nd ed. New York: John Wiley & Sons.

Cohen, M. B., Gibbons, P. B., Mugridge, W. B., and Colbourn, C. J. (2003).“Constructing Test Suites for Interaction Testing.”In *Proceedings 25th International Conference on Software Engineering*, 38–48.

Cohen, M. B., Dwyer, M. B., and Shi, J. (2007).“Interaction Testing of Highly-Configurable Systems in the Presence of Constraints.”In *Proceedings of the 2007 International Symposium on Software Testing and Analysis*, 129–139.

Colbourn, C. (2004).“Combinatorial Aspects of Covering Arrays.”*Le Matematiche* 58:121–167.

Colbourn, C., and Dinitz, J., eds.(2010)?*Handbook of Combinatorial Designs*.2nd ed. Boca Raton, FL: CRC Press.

Cornell, J. A. (1990).*Experiments with Mixtures*.2nd ed. New York: John Wiley & Sons.

Dalal, S. R., and Mallows, C. L. (1998).“Factor-Covering Designs for Testing Software.”*Technometrics* 40:234–243.

DuMouchel, W., and Jones, B. (1994).“A Simple Bayesian Modification of D-Optimal Designs to Reduce Dependence on an Assumed Model.”*Technometrics* 36:37–47.

Errore, A., Jones, B., Li, W., and Nachtsheim, C. (2017).“Using Definitive Screening Designs to Identify Active First- and Second-Order Factor Effects.”Journal of Quality Technology 49:244–264.

Fries, A., and Hunter, W. G. (1980).“Minimum Aberration 2^{k-p} Designs.”*Technometrics* 22:601–608.

Gotwalt, C. M., Jones, B. A., and Steinberg, D. M., (2009).“Fast Computation of Designs Robust to Parameter Uncertainty for Nonlinear Settings.”*Technometrics* 51:88–95.

Goos, P. (2002).*The Optimal Design of Blocked and Split Plot Experiments*.New York: Springer.

Goos, P., and Jones, B. (2011).*Optimal Design of Experiments: A Case Study Approach*.New York: John Wiley & Sons.

Hamada, M., and Wu, C. F. J., (1992).“Analysis of Designed Experiments with Complex Aliasing.”*Technometrics* 24:130–137.

Hartman, A., and Raskin, L. (2004).“Problems and Algorithms for Covering Arrays.”*Discrete Mathematics* 284:149–156.

Horn, R., and Johnson, C. (2012).*Matrix Analysis*.2nd ed. Cambridge: Cambridge University Press.

Huber, J., and Zwerina, K. (1996).“The Importance of Utility Balance in Efficient Choice Designs.” *Journal of Marketing Research* 33:307–317.

Jennrich, R. I. (1969).“Asymptotic Properties of Non-Linear Least Squares Estimators.”*Annals of Mathematical Statistics* 40:633–643.

Johnson, M. E., and Nachtsheim, C. J. (1983).“Some Guidelines for Constructing Exact D–Optimal Designs on Convex Design Spaces.”*Technometrics* 25:271–277.

Jones, B. (1991).“An Interactive Graph For Exploring Multidimensional Response Surfaces.”In *Proceedings of the Annual Joint Statistical Meetings of the American Statistical Association*.Alexandria.VA: American Statistical Association.

Jones, B., and Goos, P. (2007).“A Candidate-set-free Algorithm for Generating D-optimal Designs.”*Applied Statistics* 56:1–18.Accessed January 23, 2017. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3001117/.

Jones, B., and Goos, P. (2015).“Optimal Design of Blocked Experiments in the Presence of Supplementary Information about the Blocks.”*Technometrics* 47:01–317.

Jones, B., and Johnson, R. T. (2009).“Design and Analysis for the Gaussian Process Model.”*Quality and Reliability Engineering International* 25:515–524.Accessed January 23, 2017. https://www.stat.osu.edu/~comp_exp/jour.club/Jones_Johnson_QREI09.pdf.

Jones, B., Lekivetz, R., Majumdar, D., Nachtsheim, C. J., and Stallings, J. W. (2019).“Construction, Properties, and Analysis of Group-Orthogonal Supersaturated Designs.”*Technometrics* 62: 403 - 414.Accessed December 15, 2020. https://doi.org/10.1080/00401706.2019.1654926

Jones, B., Lin, D. K. J., and Nachtsheim, C. J. (2008).“Bayesian D-Optimal Supersaturated Designs.” *Journal of Statistical Planning and Inference* 138:86–92.

Jones, B., and Nachtsheim, C. J. (2011a).“A Class of Three-Level Designs for Definitive Screening in the Presence of Second-Order Effects.”*Journal of Quality Technology* 43:1–15.Accessed January 23, 2017. https://www.jmp.com/en_us/whitepapers/jmp/class-three-level-definitive-screening.html.

Jones, B., and Nachtsheim, C. J. (2011b).“Efficient Designs with Minimal Aliasing.”*Technometrics* 53:62–71.

Jones, B., and Nachtsheim, C. J. (2013).“Definitive Screening Designs with Added Two-Level Categorical Factors.”*Journal of Quality Technology* 45:121–129.

Jones, B., and Nachtsheim, C. J. (2016a).“Blocking Schemes for Definitive Screening Designs.”*Technometrics* 58:74–83.

Jones, B., and Nachtsheim, C. J. (2016b).“Effective Model Selection for Definitive Screening Designs.”*Technometrics*.Accessed September 28, 2017. https://www.tandfonline.com/doi/full/10.1080/00401706.2016.1234979.

Joseph, V. R., Gul, E., and Ba, S. (2015).“Maximum Projection Designs for Computer Experiments.”*Biometrika* 102:371–380.

Jowett, G. (1963).“The Relationship Between the Binomial and F Distributions.”Journal of the Royal Statistical Society 13:55–57.

Kessels, R., Jones, B., Goos, P. (2011).“Bayesian Optimal Designs for Discrete Choice Experiments with Partial Profiles.”*Journal of Choice Modelling* 4:52–74.

Kowalski, S. M., Cornell, J. A., and Vining, G. G. (2002).“Split Plot Designs and Estimation Methods for Mixture Experiments with Process Variables.”*Technometrics* 44:72–79.

Krishnamoorthy, K., and Mathew, T. (2009).*Statistical Tolerance Regions: Theory, Applications, and Computations*.New Your: John Wiley & Sons.

Lenth, R. V. (1989).“Quick and Easy Analysis of Unreplicated Fractional Factorials.”*Technometrics* 31:469–473.

Lekivetz, R. (2014).“What is an Alias Matrix?”Accessed January 23, 2017. https://blogs.sas.com/content/jmp/2014/06/05/what-is-an-alias-matrix/.

Lekivetz, R., Sitter, R., Bingham, D., Hamada, M. S., Moore, L. M., and Wendelberger, J. R. (2015).“On Algorithms for Obtaining Orthogonal and Near-Orthogonal Arrays for Main-Effects Screening.”*Journal of Quality Technology* 47:2–13.

Lin, D. K. J. (1993).“A New Class of Supersaturated Design.”*Technometrics* 35:28–31.

Martirosyan, S. (2003).“Perfect Hash Families, Identifiable Parent Property Codes and Covering Arrays.”Ph.D. diss., Universitat Duisburg-Essen.

Meyer, R. K., and Nachtsheim, C. J. (1995).The Coordinate Exchange Algorithm for Constructing Exact Optimal Designs.”*Technometrics* 37:60–69.

Meyer, R. D., Steinberg, D. M., and Box, G.(1996).Follow-up Designs to Resolve Confounding in Multifactor Experiments.*Technometrics* 38:307.

Miller, A., and Sitter, R. R. (2005).“Using Folded-Over Nonorthogonal Designs.”Technometrics 47:502–513.

Montgomery, D. C. (2013).*Design and Analysis of Experiments*.8th ed. New York: John Wiley & Sons.

Morgan, J. (2009).“Constrained Covering Arrays: Resolving Invalid Level Combination Constraints.”Technical Report, Cary NC: SAS Institute Inc., also in *Supplemental Proceedings of the 20th International Symposium on Software Reliability Engineering*.Accessed May 6, 2018. https://www.jmp.com/content/dam/jmp/documents/en/technical-reports/ConstrainedCovArray09.pdf.

Morgan, J. (2014).“Computing Coverage and Diversity Metrics for Constrained Covering Arrays,” Technical Report, Cary NC: SAS Institute Inc. Accessed May 6, 2018. https://www.jmp.com/content/dam/jmp/documents/en/technical-reports/CoverageDiversity2014.pdf.

Morgan, J. P., and Stallings, J. (2017).“Optimal experimental design that targets meaningful information.”*WIREs Comput Stat* 2017, 9:e1393. doi: 10.1002/wics.1393.

Myers, R. H., Montgomery, D. C., and Anderson-Cook, C. M. (2009).*Response Surface Methodology, Process and Product Optimization Using Designed Experiments*.3rd ed. New York: John Wiley & Sons.

Mylona, K., Goos, P., and Jones, B. (2014).“Optimal Design of Blocked and Split-plot Experiments for Fixed Effects and Variance Component Estimation.”*Technometrics* 56:132–144.

Nayeri, P., Colbourn, C., and Konjevod, G. (2013).“Randomized Postoptimization of Covering Arrays.”*European Journal of Combinatorics* 34:91–103.

Piepel, G. F. (1988).“Programs for Generating Extreme Vertices and Centroids of Linearly Constrained Experimental Regions.”*Journal of Quality Technology* 20:125–139.

Sall, J. (2008).“Choice Experimental Designs Are Different.”Accessed January 22, 2016. https://blogs.sas.com/content/jmp/2008/12/16/choice-experimental-designs-are-different/.

SAS Institute Inc.(2013).“The %MktBIBD Macro.”Accessed September 27, 2017. https://support.sas.com/rnd/app/macros/mktbibd/mktbibd.htm.

Schoen, E. D. (1999).“Designing Fractional Two-level Experiments with Nested Error Structures.”*Journal of Applied Statistics* 26: 495–508.

Shewry, M. C., and Wynn, H. P. (1987).“Maximum Entropy Sampling.”*Journal of Applied Statistics* 14:165–170.

Snee, R. D., and Marquardt, D. W. (1974).“Extreme Vertices Designs for Linear Mixture Models.”*Technometrics* 16:391–408.

Snee, R. D. (1975).“Experimental Designs for Quadratic Models in Constrained Mixture Spaces.”*Technometrics* 17:149–159.

Snee, R. D. (1979).“Experimental Designs for Mixture Systems with Multicomponent Constraints.”*Communications in Statistics -Theory and Methods* 8:303–326.

Snee, R. D. (1985).“Computer Aided Design of Experiments - Some Practical Experiences.”*Journal of Quality Technology* 17:222–236.

Vivacqua, C. A., and Bisgaard, S. (2004).“Strip-Block Experiments for Process Improvement and Robustness.”*Quality Engineering* 16:495–500.

Worley, B. A. (1987).*Deterministic Uncertainty Analysis*.Technical Report ORN-0628, Oak Ridge National Laboratory.

Wu, C. F. J. (1981).“Asymptotic Theory of Nonlinear Least Squares Estimation.”*Annals of Mathematical Statistics* 9:501–513.

Wu, C. F. J., and Hamada, M. S. (2009).*Experiments: Planning, Analysis, and Optimization*.2nd ed. New York: John Wiley & Sons.

Xiao, L., Lin, D. K. J., and Bai, F. (2012).“Constructing Definitive Screening Designs Using Conference Matrices.”*Journal of Quality Technology* 44:2–8.

Ye, K. Q., and Hamada, M. (2000).“Critical Values of the Lenth Method for Unreplicated Factorial Designs.”*Journal of Quality Technology* 32:57–66.

Yilmaz, C., Fouche, S., Cohen, M. B., Porter, A., Demiroz, G., and Koc, U. (2014).“Moving Forward with Combinatorial Interaction Testing.”*Computer* 47:37–45.

Zahran, A., Anderson-Cook, C. M., and Myers, R. H. (2003).“Fraction of Design Space to Assess the Prediction Capability of Response Surface Designs.”*Journal of Quality Technology* 35:377–386.

Zhang, Z., and Zhang, J. (2011).“Characterizing Failure-Causing Parameter Interactions by Adaptive Testing.”In *Proceedings of the 2011 International Symposium on Software Testing and Analysis*, 331–341.

附录 D

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

拟合线性模型

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020–2021. JMP® 16 拟合线性模型。Cary, NC: SAS Institute Inc.

JMP® 16 拟合线性模型

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

拟合线性模型

1 了解 JMP 17

文档和其他资源

JMP 文档中的格式规范 19

JMP 帮助 20

JMP 文档库 20

学习 JMP 的其他资源 26

JMP 教程 26

样本数据表 26

学习统计和 JSL 术语 27

学习 JMP 的提示与技巧 27

JMP 工具提示 27

JMP 用户社区 27

免费在线统计思维课程 27

JMP 新用户欢迎套件 28

统计学知识门户 28

JMP 培训 28

用户编写的 JMP 手册 28

“JMP 起始页”窗口 28

JMP 技术支持 28

2 模型规格 29

指定线性模型

“拟合模型”平台概述 31

使用“拟合模型”的回归分析示例 32

启动“拟合模型”平台 34

“拟合模型”启动窗口元素 34

构造模型效应 37

拟合特质 42

“模型规格”选项 43
信息性缺失 45
合法性检查 46
模型规格及其模型拟合的示例 46
简单线性回归 46
X 的 k 次多项式 48
X 和 Z 的 k 次多项式 49
多重线性回归 49
单因子方差分析 50
双因子方差分析 50
包含交互作用的双因子方差分析 51
三因子完全析因 53
协方差分析，斜率相等 55
协方差分析，斜率不等 56
双因子嵌套随机效应模型 58
三因子完全嵌套随机效应模型 60
简单裂区或重复测量模型 61
双因子响应曲面模型 62
纽结样条效应 64

3 标准最小二乘法模型 67
分析常用种类的模型

使用标准最小二乘法的示例 70
启动“标准最小二乘法”特质 72
“拟合模型”启动窗口 72
“拟合模型”启动窗口中的“标准最小二乘法”选项 74
标准最小二乘法中的验证 75
缺失值 76
“拟合最小二乘法”报表 76
单个和多重响应的对比 76
与重点有关的报表结构 77
特殊报表 77
“最小二乘法拟合”选项 80
“拟合组”选项 81
响应选项 81

回归报表 82
拟合汇总 83
方差分析 83
参数估计值 84
效应检验 85
效应详细信息 86
失拟 100
估计值 101
显示预测表达式 103
排序后的估计值 103
扩展后的估计值 106
指标参数化估计值 108
序贯检验 109
定制检验 109
多重比较 112
比较斜率 124
联合因子检验 124
逆预测 125
Cox 混料 129
参数功效 130
估计值的相关性 131
名义型效应的编码 132
效应筛选 133
统一尺度估计值和连续项的编码 134
图选项 135
“正态图”报表 139
“Bayes 图”报表 140
“Pareto 图”报表 141
因子刻画 142
刻画器指南 143
交互作用图 144
等高线刻画器 145
混料刻画器 146
立方图 147

Box Cox Y 变换 148
曲面刻画器 150
行诊断 151
杠杆图 152
Press 155
保存列 155
预测公式 158
“效应汇总”报表 158
混合和随机效应模型报表和选项 162
混合模型和随机效应模型 162
限制最大似然 (REML) 方法 165
EMS（传统）模型拟合报表 169
模型项之间具有线性相依性的模型 171
奇异性详细信息 171
“参数估计值”报表 172
“效应检验”报表 173
示例 173
“标准最小二乘法”特质的统计详细信息 175
重点规则 175
“定制检验”示例的详细信息 176
估计值的相关性 176
杠杆图详细信息 177
Kackar-Harville 校正 179
功效分析 180

4 标准最小二乘法示例 191
分析常用种类的模型

单因子方差分析示例 193
斜率相等的协方差分析示例 195
斜率不等的协方差分析示例 197
响应曲面模型示例 200
拟合完全响应曲面模型 200
简化模型 201
检查“响应曲面”报表 202

使用预测刻画器查找临界点 204
使用等高线刻画器查看曲面 205
裂区设计示例 206
随机效应参数估计的示例 210
纽结样条效应示例 211
活跃因子的 Bayes 图示例 214

5 逐步回归模型 217
使用变量选择查找模型

逐步回归概述 219
使用逐步回归的示例 219
“逐步”报表 221
“逐步”平台选项 221
“逐步回归控制”面板 222
“当前估计值”报表 227
“步进历史记录”报表 228
具有交叉、交互作用或多项式项的模型 229
合并规则的示例 229
具有名义型和有序型效应的模型 231
分层项的构造 231
具有名义型项的模型示例 232
分层项的限制规则的示例 235
执行二值和有序型 Logistic 逐步回归 237
使用 Logistic 逐步回归的示例 238
“所有可能的模型”选项 239
使用“所有可能的模型”选项的示例 239
“模型平均”选项 241
使用“模型平均”选项的示例 242
逐步回归中的验证选项 243
逐步回归中具有两个或三个值的验证集 243
逐步回归中的 K 重交叉验证 245

6 广义回归模型 247
使用变量选择方法构建模型
“广义回归”特质概述 249
广义回归的示例 250
启动“广义回归”特质 253
分布 254
“广义回归”报表窗口 260
“广义回归”报表选项 261
“模型启动”控制面板 262
响应分布 262
“估计方法”选项 262
高级控件 266
“验证方法”选项 268
提前停止 269
执行 269
模型拟合报表 269
回归图 269
模型汇总 270
估计详细信息 272
解路径 273
中心化和统一尺度的预测变量的参数估计值 276
原始预测变量的参数估计值 277
活跃参数估计值 278
效应检验 278
模型拟合选项 278
“广义回归”特质的统计详细信息 285
估计方法的统计详细信息 285
高级控件的统计详细信息 287
分布的统计详细信息 288

7 广义回归示例 293
使用正则化方法构建模型
Poisson 广义回归的示例 295
二项广义回归的示例 297

零泛滥 Poisson 回归的示例 299
针对宽数据的广义回归示例 301

8 混合模型 305
对均值和协方差进行联合建模

“混合模型”特质概述 308
使用混合模型的示例 308
启动“混合模型”特质 313
“拟合模型”启动窗口 313
数据格式 318
“拟合混合模型”报表 318
拟合统计量 320
随机效应协方差参数估计值 321
固定效应参数估计值 322
重复效应协方差参数估计值 323
随机系数 323
随机效应预测 323
固定效应检验 324
序贯检验 324
多重比较 325
比较斜率 325
边缘模型推断 325
“预测值 - 实际值”图 326
残差图 326
刻画器 326
条件模型推断 327
“条件预测值 - 实际值”图 327
条件残差图 327
条件刻画器 328
变差图 328
协方差和相关性矩阵 329
保存列 329
“混合模型”特质的其他示例 330
重复测量示例 330

裂区示例 345
空间示例：均匀性试验 350
相关响应示例 358
“混合模型”特质的统计详细信息 363
收敛得分检验 364
随机系数模型 365
重复测量 366
重复的协方差结构 366
空间和时间变异性 371
Kackar-Harville 校正 374

9 多元响应模型 375

使用多元方差分析拟合关系

多重响应模型的示例 377
“多元方差分析”报表 379
“多元方差分析拟合”选项 379
响应规格 380
“选择响应”选项 380
“定制检验”选项 381
模型效应选项 381
多元检验 384
扩展的多元报表 385
多元检验的比较 386
一元检验和球形检验 386
具有重复测量的多元模型 387
重复测量示例 388
复合多元模型的示例 389
判别分析 391
“保存判别值”选项的示例 392
“多元方差分析”特质的统计详细信息 393
多元检验 393
近似 F 检验 394
典型详细信息 395

10 对数线性方差模型 397

对响应的方差和均值建模

对数线性方差模型概述 399
散度效应 399
模型规格 399
注意 399
启动“对数线性方差”特质 400
使用对数线性方差的示例 400
对数线性报表 403
“对数线性”平台选项 404
保存列 404
行诊断 405
查看残差 405
刻画拟合模型 406
刻画拟合模型的示例 406

11 Logistic 回归模型 409

拟合名义型或有序型响应的回归模型

“名义型 Logistic”和“有序型 Logistic”特质概述 411
名义型 Logistic 回归 411
有序型 Logistic 回归 411
拟合 Logistic 回归模型的其他 JMP 平台 411
Logistic 回归的示例 412
名义型 Logistic 回归的示例 412
有序型 Logistic 回归的示例 414
启动“名义型 Logistic”和“有序型 Logistic”特质 416
Logistic 回归模型中的验证 417
Logistic 拟合报表 417
整体模型检验 418
拟合详细信息 419
“失拟”检验 419
“Logistic 拟合”平台选项 420
名义型和有序型拟合的选项 420
名义型拟合的选项 421
有序型拟合的选项 422

Logistic 回归的其他示例 423
“逆预测”示例 423
将“效应汇总”用于名义型 Logistic 模型的示例 425
二次有序型 Logistic 模型的示例 427
堆叠多个列中的计数的示例 429
“名义型 Logistic”和“有序型 Logistic”特质的统计详细信息 430
Logistic 回归模型 430
优势比 431
统计检验的关系 432

12 广义线性模型 433
拟合非正态响应分布的模型

“广义线性模型”特质概述 435
广义线性模型的示例 435
启动“广义线性模型”特质 437
“广义线性模型拟合”报表 439
整体模型检验 439
“广义线性模型拟合”报表选项 440
“广义线性模型”特质的其他示例 442
使用对比比较变量水平的差值 442
带偏移的 Poisson 回归 443
具有对数连结的正态回归 444
“广义线性模型”特质的统计详细信息 446
模型选择和偏差 448

A 统计详细信息 451
拟合线性模型

响应模型 453
连续响应 453
名义型响应 453
有序型响应 455
因子模型 456
连续因子 456
名义型因子 456
有序型因子 467

频数 472
常用假设 472
假设模型 472
相对显著性 472
多个推断 472
合法性评估 473
替代方法 473
关键统计概念 473
不确定性，统一的概念 474
两个基本拟合装置 474
似然、AICc 和 BIC 477
功效计算 478
LSN 的计算 478
LSV 的计算 478
功效的计算 479
调整功效的计算 480
带置信限的逆预测 481
B 参考书目 483
C 技术许可声明 489

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 19
JMP 帮助 20
JMP 文档库 20
学习 JMP 的其他资源 26
JMP 教程 26
样本数据表 26
学习统计和 JSL 术语 27
学习 JMP 的提示与技巧 27
JMP 工具提示 27
JMP 用户社区 27
免费在线统计思维课程 27
JMP 新用户欢迎套件 28
统计学知识门户 28
JMP 培训 28
用户编写的 JMP 手册 28
“JMP 起始页”窗口 28
JMP 技术支持 28

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助 > JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助 > JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 20 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助 > JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

模型规格

指定线性模型

使用“拟合模型”平台，您可以有效地指定复杂模型。通过“宏”、“特性”和变换可简化任务。“拟合模型”是拟合众多模型和效应结构的途径。

这些模型和结构包括：

- 简单和多重线性回归
- 方差和协方差分析
- 随机效应、嵌套效应、混合效应、重复测量和裂区模型
- 名义型和有序型 Logistic 回归
- 多元方差分析 (MANOVA)
- 典型相关和判别分析
- 对数线性方差（对均值和方差建模）
- 广义线性模型 (GLM)
- 参数生存和比例风险
- 响应筛选，用于研究大量响应

JMP PRO 在 JMP Pro 中，您还可以拟合以下模型：

- 包含一系列协方差结构的混合模型
- 广义回归模型，包括弹性网络、Lasso 和岭回归
- 偏最小二乘

“拟合模型”平台允许您通过选择所需特质来拟合多种类型的模型。本章着重介绍“模型规格”窗口中大多数特质通用的元素。

目录

“拟合模型”平台概述 31
使用“拟合模型”的回归分析示例 32
启动“拟合模型”平台 34
“拟合模型”启动窗口元素 34
构造模型效应 37
拟合特质 42
“模型规格”选项 43
信息性缺失 45
合法性检查 46
模型规格及其模型拟合的示例 46
简单线性回归 46
X 的 k 次多项式 48
X 和 Z 的 k 次多项式 49
多重线性回归 49
单因子方差分析 50
双因子方差分析 50
包含交互作用的双因子方差分析 51
三因子完全析因 53
协方差分析，斜率相等 55
协方差分析，斜率不等 56
双因子嵌套随机效应模型 58
三因子完全嵌套随机效应模型 60
简单裂区或重复测量模型 61
双因子响应曲面模型 62
纽结样条效应 64

“拟合模型”平台概述

“拟合模型”平台为您提供了有效方式来指定具有复杂效应结构的模型。这些模型参数的效应结构是线性的。一旦指定了模型，您就可以从若干拟合**特质**中选择适当的拟合方法。在您选择某个特质之后，“拟合模型”窗口即提供与所选特质相关的选项。本章着重介绍“模型规格”窗口中大多数特质通用的元素。有关所有特质的说明，请参见第 34 页的““拟合模型”启动窗口元素”。

“拟合模型”可用于指定众多模型，可使用多种方法拟合这些模型。表 2.1 列出了可使用“拟合模型”定义的一些典型模型。在该表中，效应 X 和 Z 代表具有连续建模类型的列，而 A、B 和 C 则代表具有名义型或有序型建模类型的列。

请参见第 46 页的“模型规格及其模型拟合的示例”，了解生成这些模型效应的点击顺序、模型拟合图以及一些示例。

表 2.1 标准模型类型

模型类型	模型效应
简单线性回归	X
X 的 k 次多项式	X, X*X,..., X^k
X 和 Z 的 k 次多项式	X, X*X,..., X^k, Z, Z*Z,..., Z^k
多重线性回归	X、Z 和其他连续列
单因子方差分析	A
双因子方差分析	A, B
包含交互作用的双因子方差分析	A, B, A*B
三因子完全析因	A, B, C, A*B, A*C, B*C, A*B*C
协方差分析，斜率相等	A, X
协方差分析，斜率不等	A, X, A*X
双因子嵌套随机效应模型	A, B[A]&随机
三因子完全嵌套随机效应模型	A, B[A]&随机, C[A,B]&随机
简单裂区或重复测量模型	A, B[A]&随机, C, C*A
双因子响应曲面模型	X&RS, Z&RS, X*X, X*Z, Z*Z

使用“拟合模型”的回归分析示例

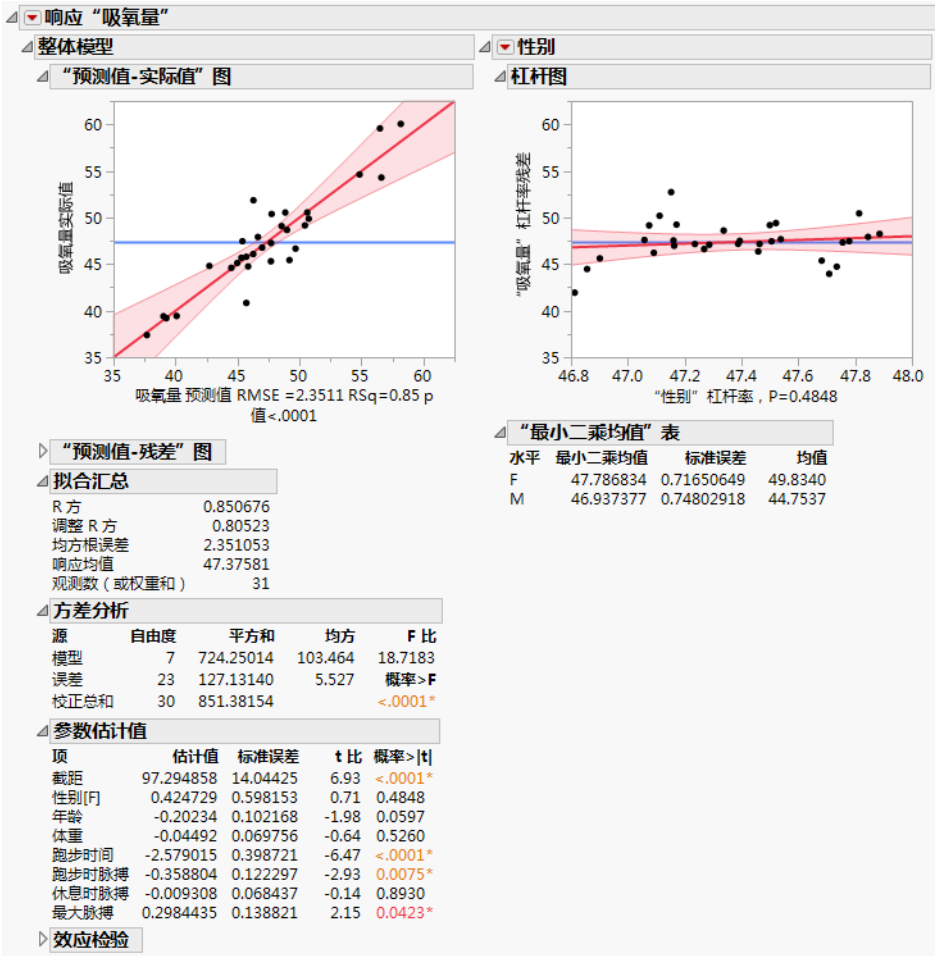
您的数据源自一项有氧健身研究，您想要从若干连续变量中预测吸氧量。

1. 选择**帮助** > **样本数据库**，然后打开 Fitness.jmp。
2. 选择**分析** > **拟合模型**。请注意，“特质”框为空。
3. 选择**吸氧量**并点击 Y。

   指定连续响应时，“特质”默认设置为“标准最小二乘法”，但您可以自由选择其他特质。此外，“重点”默认设置为“效应杠杆率”。
4. 按 Ctrl 键，然后选择**性别**、**年龄**、**体重**、**跑步时间**、**跑步时脉搏**、**休息时脉搏**和**最大脉搏**。点击**添加**将这些变量添加到“构造模型效应”列表。请注意，若想要让该窗口之后可用，您可以选择**保持对话框打开**。您的 “模型规格”窗口应如图 2.1 所示。
5. 点击**运行**。图 2.2 给出了报表的部分视图。

图 2.1 健身回归模型的 “模型规格”窗口

图 2.2 健身数据的 “标准最小二乘法”报表的部分视图



整体模型的图和报表显示在最左侧的报表列中。右侧列显示您在模型中指定的每个效应的杠杆图。受空间所限，图 2.2 仅显示了“**性别**”对应的列，但报表中还显示了其他六个效应对应的列。红色小三角菜单包含其他选项，可用于将报表和图添加至报表窗口。有关“标准最小二乘法”报表窗口的详细信息，请参见“标准最小二乘法模型”一章中第 76 页的““拟合最小二乘法”报表”。

检查“参数估计值”报表中的 *p* 值，您可以看到**跑步时间**、**跑步时脉搏**和**最大脉搏**似乎是对吸氧量贡献显著的预测变量。下一步可以通过删除不显著的预测变量来简化模型。请参见第 217 页的“逐步回归模型”一章。

启动“拟合模型”平台

通过选择**分析** > **拟合模型**启动“拟合模型”平台。图 2.3 显示了 Fitness.jmp 样本数据表的启动窗口的示例。“拟合模型”启动窗口是称为“特质”的许多平台的常见启动窗口。请参见第 42 页的“拟合特质”。

图 2.3 “拟合模型”启动窗口

注意：在包含名为**模型**的脚本的数据表中选择**分析** > **拟合模型**后，启动窗口会根据该脚本自动填充。

“拟合模型”启动窗口包含以下主要区域：

- “选择列”是当前数据表中所含列的列表。若当前数据表包含排除的列，则它们不会显示在列表中。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。
- “选择角色变量”包含标准的 JMP 启动窗口按钮。有关这些按钮的说明，请参见第 34 页的“‘拟合模型’启动窗口元素”。
- “构造模型效应”包含您在模型中输入效应所用的选项。请参见第 37 页的“构造模型效应”。
- “特质”是您可以选择的模型类型的列表。一旦选择某个特质，将根据您所选的特质显示不同的选项。请参见第 42 页的“拟合特质”。

“拟合模型”启动窗口元素

以下“拟合模型”启动窗口元素是大多数特质所共有的：

模型规格 “模型规格”菜单包含以下选项：

中心多项式 当模型中包含多项式时使效应中心化。

信息性缺失 创建可容纳效应中的缺失值的编码系统。

设置 Alpha 水平 为模型报表中的置信区间设置 alpha 水平。

保存到数据表 将模型规格保存至当前数据表中的脚本。

保存到脚本窗口 将模型规格保存至脚本窗口。

创建 SAS 作业 将当前模型规格的 SAS 代码保存至 “SAS 程序” 窗口。

提交至 SAS 将当前模型规格的 SAS 代码提交给 SAS 会话。

收敛设置 包含的选项用于设置某些特质的模型收敛中所用的最大迭代次数和收敛极限。

有关其中任何选项的详细信息，请参见第 43 页的 “‘模型规格’ 选项”。

选择列 列出当前数据表中的未排除列。

Y 标识模型的一个或多个响应变量 （因变量）。

注意：“拟合模型” 中不支持具有向量值的响应列。

权重 标识一列，该列的值为分析的每一行都分配一个权重。请参见第 36 页的 “权重”。

频数 标识一列，该列的值为分析的每一行都分配一个频数。通常来说，频数列的作用是扩展数据表，使得具有整数频数 *k* 的任何行都扩展为 *k* 行。允许您指定小数频数。请参见第 36 页的 “频数”。

JMP PRO **验证** 在 JMP Pro 中，对于某些特质，您可以输入 “验证” 列。有关每个特质如何处理 “验证” 列的详细信息，请参见相应的 “特质” 章节。若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见《预测和专业建模》。有关如何在 JMP 建模平台中使用 “验证” 列的详细信息，请参见 《预测和专业建模》。

依据 标识用于创建报表的一个列，该报表包括变量的每个水平的单独分析。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的分析。

添加 向模型添加效应。请参见第 37 页的 “添加”。

交叉 通过交叉两个或更多变量创建交互作用和多项式效应。请参见第 37 页的 “交叉”。

嵌套 创建嵌套效应。请参见第 37 页的 “嵌套”。

宏 为常用模型生成效应。请参见第 38 页的 “宏”。

次数 对包含使用 “宏” 生成的析因或多项式效应的模型应用指定次数。请参见第 38 页的 “宏” 中的**析因次数**和**多项式次数**。

特性 对模型效应应用特性。这些特性决定如何处理效应。请参见第 39 页的 “特性”。

变换 变换所选的连续效应或 *Y* 列。请参见第 40 页的 “变换”。

无截距 从模型中排除截距项。

特质 指定拟合方法。请参见第 34 页的 “‘拟合模型’启动窗口元素”。根据您所选的特质显示不同的选项。

目标水平 （仅在特定特质中并且 *Y* 为二值响应且建模类型为名义型时才可用。）指定您想要对其概率建模的水平。默认值是基于水平顺序的两个水平中的较高者。

帮助 将您引导到“拟合模型”启动窗口的“帮助”主题。

重新调用 使用您上一次运行的模型规格填充启动窗口。

删除 从分配的角色中删除选定的变量。也可以双击效应，或选择效应并按 Backspace 键。

运行 为指定的模型和特质生成报表窗口。

保持对话框打开 运行分析后保持启动窗口打开，以便您随时更改并重新运行分析。

频数

在**频数**文本框中输入的频数变量在大多数“拟合模型”特质中都受支持。通常，按以下方式解释频数。假定某行的频数为 *f*。那么计算结果等同于包含该行 *f* 个副本且每行频数为 1 的数据表的计算结果。

频数值为 0 或缺失的行将从分析中排除。包含负频数值的行只能用于删失观测，如若不然，就会从分析中排除这些行。与删失观测一同使用时，可以使用负频数值拟合截断的分布。

频数值不必为整数。有关说明如何处理频数列（包括具有非整数值的频数列）的技术细节，请参见第 472 页的“频数”。

权重

权重变量适用于观测具有不同方差的情形。例如，当用户针对每行包含预先汇总均值的数据执行回归建模时就会发生这种情况。此时，涉及大量观测（较小方差）的行对损失函数的贡献应该比涉及少量观测（较大方差）的行的贡献大。您可以通过使用正确定义的权重来确保这一点。

权重变量在许多“拟合模型”特质中都受支持。支持权重变量的每个特质都使用以下方法之一：

- 方差统一尺度
- 频数对称性

方差统一尺度

使用最小二乘法或正态理论最大似然执行估计时，给定行的权重 w 按照 $w^{-1/2}$ 来调整该行对损失函数的贡献。

权重变量会影响估计值和标准误差。不过，与频数变量不同，权重变量不影响假设检验中使用的自由度。

权重为负值或零的行将从分析中排除。

频数对称性

在 “名义型 Logistic” 和 “有序型 Logistic” 特质中，会像处理频数变量那样来处理权重变量。若同时指定了权重和频数变量，这些特质在处理每个观测时就像是有一个频数值等于权重与频数值的乘积一样。

构造模型效应

本节说明可用来协助在模型中输入效应的选项。第 46 页的 “模型规格及其模型拟合的示例” 中提供了如何使用这些选项来获取特定类型的模型的示例。

添加

向模型添加效应。可以从 “选择列” 列表中直接添加这些效应，也可以在该列表中选择这些效应并使用 “宏” 或 “特性” 加以修改。还可以使用 “交叉” 和 “嵌套” 创建和添加或修改效应。效应中涉及的变量的建模类型以及分配给该效应的任何 “特性” 共同决定如何在模型中处理该效应。

注意：要从 “构造模型效应” 列表中删除某个效应，请双击该效应，也可以选择它并点击**删除**或按 Backspace/Delete 键。

交叉

创建交互作用或多项式效应。在 “选择列” 列表中选择两个或更多变量，然后点击**交叉**。或者，在 “选择列” 列表中选择一个或多个变量，并且在 “构造模型效应” 列表中选择一个或多个效应，然后点击**交叉**。

有关如何对交叉效应进行参数化和编码的讨论，请参见第 451 页的 “统计详细信息”。

注意：您可以构造最多组合十个交叉和嵌套列的效应。

交叉效应示例

假定某种产品涂层要求涂某种染料。“染料 pH 值” 和 “染料浓度” 可能都对涂层颜色有影响。为了了解这两方面的影响，您需要设计一个实验，将 “染料 pH 值” 和 “染料浓度” 设置为各自的高水平和低水平。“染料 pH 值” 对颜色的影响在 “染料浓度” 的高水平上可能比在其低水平上更为显著。这称为**交互作用**。要对这个可能的交互作用建模，您需要在 “构造模型效应” 列表中包含交叉项 “染料 pH 值 * 染料浓度”。这样 JMP 就可以检验交互作用。

嵌套

创建嵌套效应。若一个效应 (B) 的所有水平仅在另一个效应 (A) 的一个水平中出现，则称 B **嵌套**在 A 中。通常表示为 B[A]，读作 “B 嵌套在 A 中”。请注意，嵌套定义了一种层次关系。A 称为**外部**效应，B 称为**内部**效应。嵌套项必须是分类项。

注意：嵌套项必须按照从外到内的顺序来指定。例如，若 B 嵌套在 A 中，C 嵌套在 B 中，则将模型指定为：A, B[A], C[B,A]（也可以表示为 A, B[A], C[A,B]）。您可以构造最多组合十个交叉和嵌套列的效应。

嵌套效应示例

为了演示嵌套，考虑两所学校中的数学老师。一所学校有三位数学老师；另一所学校有两位数学老师。每所学校中的每位老师都教两个或三个班，各班学生没有重叠。在本例中，班级 (C) 嵌套在教师 (B) 中，教师 (B) 嵌套在学校 (A) 中。在 “拟合模型” 启动窗口中输入这些效应：

1. 将 A 和 B 均添加到 “构造模型效应” 面板中。
2. 在 “构造模型效应” 面板中，选择 B。
3. 在 “选择列” 列表中，选择 A。
4. 点击**嵌套**。这会将 B 转换为效应 B[A]。
5. 将 C 添加至 “构造模型效应” 面板。
6. 在 “构造模型效应” 面板中，选择 C。
7. 在 “选择列” 列表中，选择 A 和 B。
8. 点击**嵌套**。这会将 C 转换为效应 C[A, B]。

宏

在 “宏” 列表中，选择选项以便为常用模型自动生成效应，并将其输入 “构造模型效应” 列表：

完全析因 为 “选择列” 列表中选定的列创建所有主效应和交互作用。在输入这些对象时采用 “选择列” 列表中所列的主效应的顺序。对于其他排序，请参见该表中的**析因排序**。

析因次数 创建所有主效应，但仅创建最高达到指定次数 （阶数） 的交互作用。在**宏**按钮下方的**次数**框中指定次数。

析因排序 创建与**完全析因**选项相同的一组效应，但按次数列出它们。先列出所有主效应，随后是所有双因子交互作用，然后是所有三因子交互作用，依此类推。

响应曲面 创建主效应、双因子交互作用和二次项。将为选定的主效应指定响应曲面特性，表示为 RS。对主效应应用 RS 特性并且选定 “标准最小二乘法” 特质后，将提供 “响应曲面” 报表。该报表提供有关临界值和响应曲面形状的信息。

另请参见第 39 页的 “特性” 中的 “响应曲面效应” 和 《实验设计指南》。

混料响应曲面 创建主效应和双因子交互作用。主效应具有响应曲面 (RS) 和混料 （**混料**）特性。在 “标准最小二乘法” 特质中，**混料**特性导致拟合混料模型。 RS 特性将创建特定于混料模型的 “响应曲面” 报表。

另请参见第 39 页的 “特性” 中的 “混料效应” 和 《实验设计指南》。

多项式次数 创建主效应和最高达指定次数的多项式项。在**宏**按钮下方的**次数**框中指定次数。

Scheffé 三次项 创建主效应、交互作用和 Scheffé 三次项，这在为混料实验指定响应曲面时很有用。该宏创建完整的三次模型。

若您使用三次多项式模型拟合混料，您不能引入偶数幂的项 （例如 X1*X1*X2），因为无法估计这样的项。不过，完整的曲面多项式表示形式却可以包含 X1*X2*(X1 – X2) 形式的项，它们称为 Scheffé 三次项。

若您在**次数**框中输入 3，而后选择**混料响应曲面**宏命令，此时也会包含 Scheffé 三次项。

JMP PRO **分组的回归变量** 为在报表不同部分中被视为一个效应的一组连续因子创建单个效应。您可以在 “效应汇总”中添加或删除分组效应。每个分组效应都有一个杠杆图和一个效应检验，而不是组中每个因子都有单独的图和检验。若数据表包含表示分类效应水平的指示符列，这会很有用。

注意：包括在一组回归变量中的非连续因子不包括在分析中。

特性

在 “特性”列表中，选择您可以为 “构造模型效应”列表中选定的效应分配的特性。

随机效应 为效应分配 “随机”特性。有关随机效应的详细信息，请参见 “标准最小二乘法模型”一章中第 164 页的 “指定随机效应和拟合方法”。

响应曲面效应 为效应分配 RS 特性。请注意，必须在 “构造模型效应”列表中包括相关模型项。“宏”列表中的 “响应曲面”选项自动生成这些项并将 RS 特性分配给主效应。要获取 “响应曲面”报表，无需为交互作用和多项式项分配 RS 特性。您只需将该特性分配给主效应。

对数方差效应 为效应分配 “对数方差”特性。该特性指示将在响应的方差模型中包含该效应。

要想**同时**为响应的均值和方差在模型中包含某个效应，您必须将该效应指定两次。在选项卡式界面中，它必须同时显示在 “均值效应”选项卡和 “方差效应”选项卡上。若不这样指定，您可以在 “均值效应”选项卡上将该效应输入两次，一次不包含 “对数方差效应”特性，另一次包含 “对数方差效应”特性。

混料效应 为主效应分配 “混料”特性。该特性用于指定混料中涉及的主效应。请注意，“宏”列表中的 “混料响应曲面”选项会为选定效应自动分配混料特性，并在可能的情况下提供 “响应曲面”报表。

排除效应 为效应分配 “排除”特性。这将从模型拟合中排除该效应。不过，该效应用于对观测分组，以执行失拟检验。在 “标准最小二乘法”特质中，为该效应提供最小二乘均值表。

纽结样条效应 为连续主效应分配 “纽结”特性。这会向模型规格隐式添加效应的三次样条。请参见第 39 页的 “纽结样条效应”。

纽结样条效应

纽结样条用于借助灵活的预测变量函数来拟合响应 Y 。考虑使用单个预测变量 X。若为 X 分配了 “纽结样条效应”并且指定了 k 个结点，则会向该组预测变量隐式添加 k - 2 个额外效应。其

中的每一个效应都是分段三次多项式样条，其中的各段由结点来定义。请参见 Stone and Koo (1985)。

样条数由结点数确定，结点数需要您来指定。根据特质使用的方法来估计与样条关联的系数。

结点按照文献提供的指导性信息来放置。特别是：若点数未超过 100 个，第一个结点和最后一个结点分别为第五个最小点和第五个最大点。另一方面，对于未超过 5 个结点的情况，第一个结点和最后一个结点分别位于第 0.05 和 0.95 分位数；对于 5 个以上结点的情况，第一个结点和最后一个结点分别位于第 0.025 和 0.975 分位数。观测数超过 30 时，默认结点数为 5；观测数不足 30 时，默认结点数为 3。

注意：仅针对主效应连续项应用纽结样条。

变换

使用 “变换”选项可变换在 “构造模型效应”文本框中选定的 Y 列或主效应。

注意：您还可以通过在“选择列”列表中右击某列并选择“变换”来变换该列。“选择列”列表中将出现变换列的引用。随后，您可以在“拟合模型”窗口中像使用任意数据表列那样使用该列。请参见《使用 JMP》。

无 删除已应用的所有 “变换”选项。

对数 对选定变量应用自然对数变换。

平方根 应用选定变量值的平方根。

平方 应用选定变量值的平方。

倒数 对变量 X 应用变换 1/X。

指数 对选定变量应用指数变换。

Arrhenius 对变量 T （摄氏度温度）应用 Arrhenius 变换：

$$X = \frac{11604.5181215503}{T + 273.15}$$

这是与活化能相乘的 Arrhenius 关系的成分。

ArrheniusInv 对变量 X 应用 Arrhenius 变换的逆变换：

$$T = \frac{11604.5181215503}{X} - 273.15$$

Logit 对选定列计算 Logistic 函数的反函数 （其中， p 介于 0 到 1 之间）：

$$\text{Logit}(p) = \log\left(\frac{p}{1-p}\right)$$

Logistic 计算选定列的 Logistic（亦称 Squish 和 Logist）函数（其中，结果介于 0 到 1 之间）：

$$\text{Logistic}(x) = \frac{1}{(1+\text{e}^{-x})}$$

LogitPct 以百分比形式计算选定列的 Logit （其中， *pct* 是介于 0 到 100 之间的百分比）：

$$\text{LogitPct}(pct) = \log\left(\frac{\left(\frac{pct}{100}\right)}{1-\left(\frac{pct}{100}\right)}\right)$$

LogisticPct 以百分比形式计算选定列的 Logistic（或 Logist，其中，结果介于 0 到 100 之间）：

$$\text{LogisticPct}(x) = \frac{100}{(1+e^{-x})}$$

无截距

若想要拟合不含截距项的模型，则选择“无截距”。某些建模结构要求无截距模型。对于这些结构，默认选中“无截距”框。

“构造模型效应”选项卡

对于以下特质，您可以使用选项卡式界面输入模型效应：

注意：若在第一个（主）选项卡上对效应应用了“特性”，这些特性确定如何在模型中处理这些效应。若运行该模型，然后从报表的红色小三角菜单中请求“模型对话框”，您会发现这些效应显示在相应的选项卡上。

标准最小二乘法 输入模型效应：

- **“固定效应”选项卡** 输入的效应要作为固定效应来建模。固定效应是要关注其特定处理水平的效应。您想要比较其处理水平之间的响应均值。
- **“随机效应”选项卡** 输入的效应要作为随机效应来建模。随机效应是其水平被视为较大总体中的随机样本的效应。您想要估计响应中可归因于该效应的变异。

混合模型 输入模型效应：

- **“固定效应”选项卡** 输入的效应要作为固定效应来建模。请参见该表中的“标准最小二乘法”。
- **“随机效应”选项卡** 输入的效应要作为随机效应来建模。用于方差分量模型和随机系数模型。
- **“重复结构”选项卡** 用于选择重复效应的协方差结构。

对数线性方差 输入模型效应：

- **“均值效应”选项卡** 输入要对其期望值建模的效应。

“方差效应”选项卡 输入要对其方差建模的效应。

若想同时对效应的期望值和方差建模，您必须在这两个选项卡上都输入该效应。

参数生存 输入模型效应：

“位置效应”选项卡 输入要在对位置参数 mu 或 Weibull 分布中的形状参数建模时使用的效应。

“尺度效应”选项卡 输入要在对尺度参数建模时使用的效应。

拟合特质

在“拟合模型”启动窗口中，您通过选择“特质”来选择拟合和分析方法。根据您输入的一个或多个响应和因子，JMP 会根据上下文作出关于所需特质的最初猜测，但您可以在特质菜单中更改该选择。

以下拟合特质可用：

标准最小二乘法 拟合响应为连续型的模型。拟合方法包括回归、方差分析、协方差分析、混合模型以及设计实验分析。请参见第 67 页的“标准最小二乘法模型”一章和第 75 页的“标准最小二乘法的“重点”选项”。

逐步 协助为标准最小二乘法和有序型 Logistic 分析（或具有二值响应的名义型 Logistic 分析）选择变量。对于连续响应，提供交叉验证、p 值、BIC 和 AICc 准则。同时还提供用于拟合所有可能模型和用于模型平均的选项。对于 Logistic 拟合，提供 *p* 值、BIC 和 AICc 准则。请参见第 217 页的“逐步回归模型”一章。

JMP PRO **广义回归** 使用正则（亦称惩罚）回归方法拟合广义线性模型。正则方法包括岭回归、Lasso、自适应 Lasso、弹性网络和自适应弹性网络。响应分布包括：正态、二项、Poisson、零泛滥 Poisson、负二项、零泛滥负二项和 gamma 分布。请参见第 247 页的“广义回归模型”一章和第 254 页的“分布”。

JMP PRO **混合模型** 针对具有复杂协方差结构的连续响应拟合众多线性模型。处理的情形包括：

- 裂区实验
- 随机系数模型
- 重复测量设计
- 空间数据
- 相关响应数据

请参见第 305 页的“混合模型”一章。

多元方差分析 拟合涉及多个连续 Y 变量的模型。拟合方法包括：多元方差分析、重复测量、判别分析和典型相关。请参见第 375 页的“多元响应模型”一章。

对数线性方差 对于连续 Y 变量，构造均值模型和方差模型。您可以为这两种模型指定不同的效应。请参见第 397 页的“对数线性方差模型”一章。

名义型 Logistic 拟合名义型响应的 Logistic 回归模型。请参见第 409 页的 “Logistic 回归模型”一章。

有序型 Logistic 拟合有序型响应的 Logistic 回归模型。请参见第 409 页的 “Logistic 回归模型”一章。

比例风险 拟合半参数化回归模型 （Cox 比例风险模型），以评估解释变量对生存时间的影响（将删失考虑在内）。

您还可以通过选择**分析 > 可靠性和生存 > 拟合比例风险**来启动该特质。请参见《可靠性和生存方法》。

参数生存 拟合生存时间的一般线性回归模型。若您的生存时间可表示为一个或多个解释变量的函数，则可以使用该选项。需要将多种生存分布和删失考虑在内。

您还可以通过选择**分析 > 可靠性和生存 > 拟合参数生存**来启动该特质。请参见《可靠性和生存方法》。

广义线性模型 使用多种分布和连结函数拟合广义线性模型。拟合方法包括：Logistic、Poisson 和指数回归。请参见第 433 页的 “广义线性模型”一章。

JMP PRO **偏最小二乘** 使用潜在因子拟合一个或多个 *Y* 变量的模型。这允许在解释变量 （*X* 变量）高度相关或是 *X* 变量的数量多于观测数时拟合模型。

您还可以通过选择**分析 > 多元方法 > 偏最小二乘**来启动偏最小二乘分析。请参见《多元方法》。

响应筛选 自动执行在大量响应之间进行线性模型效应检验的过程。检验结果和汇总统计量显示在数据表和图中。假发现率 (FDR) 方法可防止错误声明显著性。稳健估计方法可降低检验对于离群值的灵敏度。

注意：该特质仅允许连续响应。通过选择**分析 > 筛选 > 响应筛选**也可以提供针对单个因子的响应筛选。该平台支持分类响应，还提供等价性检验和具有实际意义的显著性检验。请参见 《预测和专业建模》。

“模型规格”选项

“模型规格”红色小三角菜单包含以下选项：

中心多项式 使得效应中涉及的次数大于 1 的所有连续项都以其均值为中心。默认选中该选项，在效应中涉及的项分配有 “混料效应”特性或具有 “混料”列属性的情况下除外。具有 “编码”列属性的项以其指定的 “高”值和 “低”值的中间值为中心。

中心化有助于使回归系数更容易解释，并且可降低低阶与高阶效应之间的共线性。

信息性缺失 提供缺失值的编码系统。该系统允许在存在缺失值的情况下估计预测模型。这在缺失数据可提供信息的情形中很有用。请参见第 45 页的 “信息性缺失”。

该选项可用于以下特质：标准最小二乘法、逐步、广义回归、多元方差分析、对数线性方差、名义型 Logistic、有序型 Logistic、比例风险、参数生存、广义线性模型和响应筛选。

设置 Alpha 水平 为“拟合模型”分析中的置信区间设置 alpha 水平。默认 alpha 水平为 0.05。

误差指定 （仅适用于没有随机效应时的“标准最小二乘法”特质。）指定用于“拟合最小二乘法”报表中的标准误差和检验的误差方差和误差自由度。请注意，更改“误差指定”不会影响“学生化残差”图和“Box Cox 变换”报表。若“误差指定”是“纯误差”或“指定”，则“方差分析”报表中将显示一个附加列。请参见第 83 页的“方差分析”。

默认估计值 使用模型的标准均方根误差和误差自由度计算所有检验和标准误差。

纯误差 使用“失拟”报表中的纯误差均方和相关的自由度计算所有检验和标准误差。请参见第 100 页的“失拟”。

警告：若纯误差自由度为 1，则会显示一条警告消息，指示检验较弱且置信限较大。

指定 使用用户指定的误差方差和误差自由度值计算所有检验和标准误差。

保存到数据表 将您的“拟合模型”启动窗口规格另存脚本附加到数据表中。该脚本名为“模型”。若表中包含名为“模型”的脚本，该脚本将在您选择**分析 > 拟合模型**时自动填充启动窗口。（若不想这样自动填充启动窗口，您可以重新命名脚本。）

有关 JSL 脚本的详细信息，请参见《Scripting Guide》。

保存到脚本窗口 将您的“拟合模型”启动窗口规格复制到脚本窗口中。您可以保存脚本窗口，之后通过运行该脚本随时重新创建该模型。

创建 SAS 作业 在脚本窗口中创建一个 SAS 程序，该程序可以在 SAS 中重新创建当前分析和数据表。一旦创建，就有若干选项供您向 SAS 提交代码。

1. 复制代码并将其粘贴到“SAS 程序编辑器”中。若您运行的是 SAS 旧版本（8.2 之前的版本），该方法很有用。
2. 选择**编辑 > 提交至 SAS**。
3. 保存该文件，并通过双击在 SAS 本地副本中打开该文件。若您想要利用 SAS ODS 选项（例如，从 SAS 代码生成 HTML 或 PDF 输出），该方法很有用。

请参见《使用 JMP》。

提交至 SAS 将代码提交至 SAS 并在 JMP 中显示结果。若您未连接 SAS 服务器，系统提示会引导您完成整个连接过程。

请参见《使用 JMP》。

收敛设置 “收敛设置”菜单包含以下选项：

最多迭代次数 指定用在模型拟合中的最大迭代次数。默认情况下，最大迭代次数为 100。若您的模型不容易收敛，您可能需要增加迭代次数。若有非常大的数据集或复杂模型，您可能需要限制迭代次数。

收敛极限 指定模型拟合的收敛极限。若您的模型不容易收敛，您可能需要增大收敛极限。默认情况下，收敛极限为 0.00000001。

注意：仅对特定特质显示“收敛设置”菜单。在“标准最小二乘法”特质中，仅当有一个随机效应并且将 REML 选作启动窗口中的“方法”时才显示“收敛设置”菜单。

信息性缺失

“信息性缺失”选项构造了编码系统，该系统允许在存在缺失值的情况下估计预测模型。该系统可同时支持对连续模型效应和分类模型效应编码。

连续效应

若连续主效应包含缺失值，将创建新的设计矩阵列。该列是一个指示变量；若主效应列缺失则变量值为 1，若主效应列不缺失则变量值为 0。此外，连续主效应的缺失值将替换为一个均值，该均值是在分析中所包含行的非缺失值的均值。该均值是一个中性值，用于保持参数估计值的可解释性。

与该指示变量关联的参数用来对缺失值分组所预测的响应与预测响应（若协变量设置为其均值）之间的差值进行估计。

对于高阶效应，协变量中的缺失值将替换为协变量均值。这会使包含缺失值的行的高阶效应为 0（假定选中默认设置“中心多项式”）。这是因为“中心多项式”以多项式中涉及的各项的均值为这些项的中心。

在“效应检验”报表中，包含缺失值的每个连续主效应的参数数目都为 2。在“参数估计值”报表中，包含缺失值的连续主效应的参数标记为**<列名>或均值（若缺失）**，且指标参数标记为**<列名>缺失**。保存到数据表的预测公式按照与这些模型参数对应的表达式来提供。

分类效应

若名义型或有序型主效应包含缺失值，这些缺失值将单独编码为该效应的一个水平。因此，在“效应检验”报表中，包含缺失值的每个分类主效应都包含一个额外参数。

在“参数估计值”报表中，名义型效应的参数标记为**<列名>[ ]**。对于有序型效应，参数标记为**<列名>[-x]**，其中的 x 表示具有最高值排序的水平。

和连续效应一样，保存到数据表的预测公式按照与这些模型参数对应的表达式来提供。

编码表

在使用“标准最小二乘法”特质时，您可以通过选择**保存列 > 保存编码表**来查看“信息性缺失”模型中使用的设计矩阵列。

合法性检查

拟合模型将检查模型中是否存在层次中的效应重复或效应缺失之类的错误。若您收到警报消息，您可以点击**继续**继续拟合，也可以点击**取消**停止拟合过程。

模型规格及其模型拟合的示例

本节提供一些模板，通过“拟合模型”平台中的“构造模型效应”面板，可以为您指定的各种模型类型输入效应。

- 模型效应 X 和 Z 表示连续列。
- 模型效应 A、B 和 C 表示名义型或有序型列。

对于多数模型，还提供其模型拟合的可视化视图。

- “简单线性回归”
- “X 的 k 次多项式”
- “X 和 Z 的 k 次多项式”
- “多重线性回归”
- “单因子方差分析”
- “双因子方差分析”
- “包含交互作用的双因子方差分析”
- “三因子完全析因”
- “协方差分析，斜率相等”
- “协方差分析，斜率不等”
- “双因子嵌套随机效应模型”
- “三因子完全嵌套随机效应模型”
- “简单裂区或重复测量模型”
- “双因子响应曲面模型”
- “纽结样条效应”

简单线性回归

要输入的效应：X

1. 在“选择列”列表中，选择 X。
2. 点击**添加**。

简单线性回归模型的示例

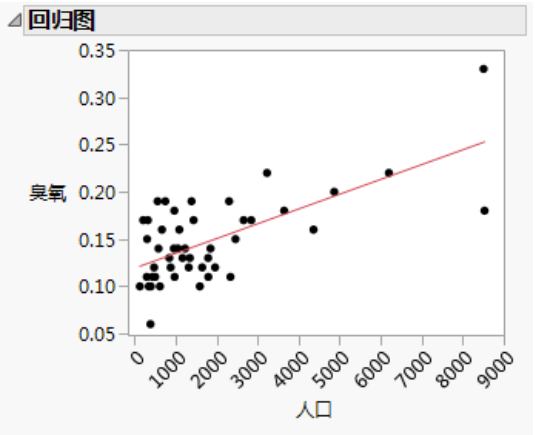
您关注的是人口（给定城市以千人计的人口数）与臭氧水平之间的关系。**臭氧**是要关注的响应，**人口**是连续模型效应。

1. 选择**帮助** > **样本数据库**，然后打开 Polycity.jmp。
2. 选择**分析** > **拟合模型**。
3. 在 “选择列”列表中，选择**臭氧**并点击 Y。
4. 在 “选择列”列表中，选择**人口**。
5. 点击**添加**。

图 2.4 简单线性回归的 “拟合模型”窗口

点击**运行**时，将出现 “拟合最小二乘法”报表，其中显示包括 “回归图”在内的各种结果。“回归图”显示数据和数据的简单线性回归模型拟合。

图 2.5 简单线性回归的模型拟合



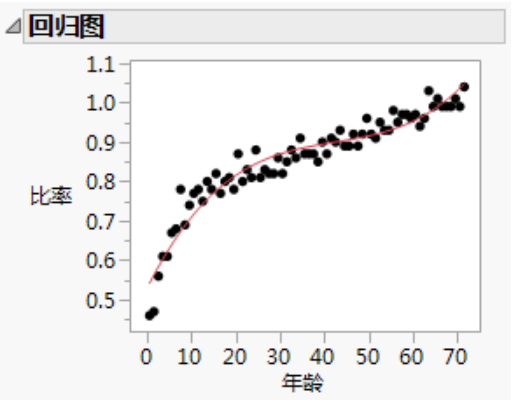
X 的 k 次多项式

要输入的效应：X, X*X,..., X^k

1. 在**次数**对应的文本框中键入 *k*。
2. 在 “选择列” 列表中，选择 X。
3. 选择**宏** > **多项式次数**。

图 2.6 显示了针对 Growth.jmp 样本数据表，数据和数据的三次多项式模型之间的关系图。这是在运行**二元**数据表脚本后生成的其中一个拟合。

图 2.6 单变量的三次多项式的模型拟合



X 和 Z 的 k 次多项式

要输入的效应：X, X*X,..., X^k, Z, Z*Z,..., Z^k

1. 在**次数**对应的文本框中键入 *k*。
2. 在“选择列”列表中，选择 X 和 Z。
3. 选择**宏** > **多项式次数**。

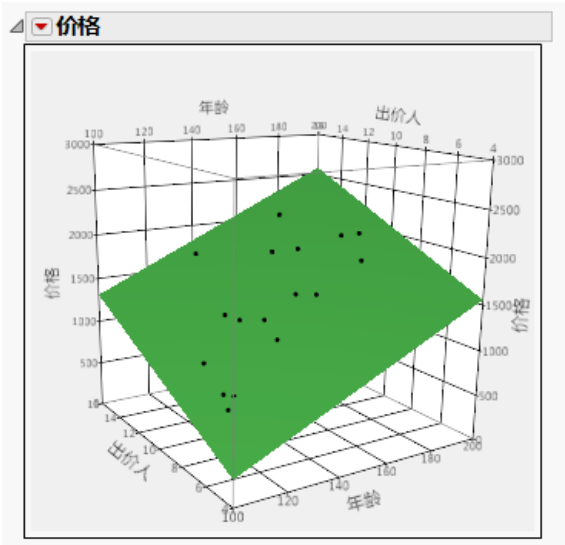
多重线性回归

要输入的效应：选定列

1. 在“选择列”列表中，选择要关注的连续效应。
2. 点击**添加**。

图 2.7 显示了针对 Grandfather Clocks.jmp 样本数据表，数据和数据的多重线性回归拟合的曲面刻画器图。模型效应为**年龄**和**出价人**。响应为**价格**。您可以通过运行数据表脚本**带曲面刻画器图的拟合模型**来获取该图。

图 2.7 包含两个预测变量的多重线性回归模型的模型拟合



多重线性回归模型的示例

另见第 32 页的“使用“拟合模型”的回归分析示例”，了解具有若干预测变量的示例。

单因子方差分析

要输入的效应：A

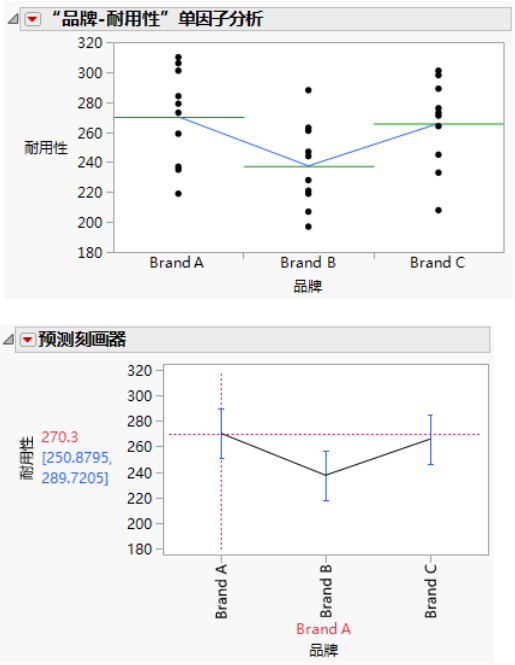
1. 在“选择列”列表中，选择一个名义型或有序型效应：A。
2. 点击**添加**。

考虑 Golf Balls.jmp 样本数据表。您关注的是**耐用性**是否随**品牌**而变化。图 2.8 显示了两个图。

第一个图使用“以 X 拟合 Y”获得，其中显示按品牌划分的数据。每个品牌的均值处都绘制了水平线，并通过线段将这些均值连接起来。要生成该图，请运行 Golf Balls.jmp 样本数据表中的脚本**单因子: 品牌 - 耐用性**。

第二个图是使用“拟合模型”获取的刻画器图。这第二个图显示每个品牌的预测响应，响应之间由线段连接。要生成该图，请运行 Golf Balls.jmp 样本数据表中的脚本**拟合模型: 品牌 - 耐用性**。将红色垂直虚线拖至关注的品牌。红色水平虚线更新，与垂直轴交叉在预测响应处。

图 2.8 单因子方差分析的模型拟合



双因子方差分析

要输入的效应：A, B

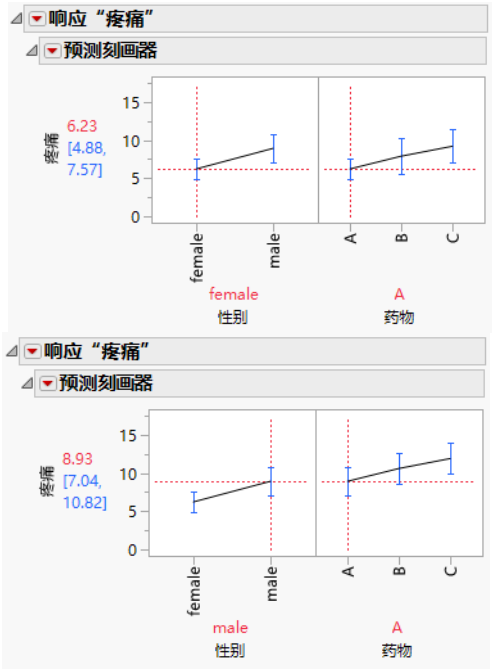
1. 在“选择列”列表中，选择两个名义型或有序型效应：A 和 B。
2. 点击**添加**。

图 2.9 显示 Analgesics.jmp 样本数据表的数据拟合的两个刻画器图。模型效应为**性别**和**药物**。响应为**疼痛**。要获取该图，请选择**分析 > 拟合模型**，将**疼痛**选作 Y，将**性别**和**药物**选作模型效应，然后点击**运行**。点击“响应‘疼痛’”红色小三角并选择**因子刻画 > 刻画器**。

每个图中的线段将红色垂直虚线定义的设置的预测值连接起来。移动它们可以查看其他设置处的预测。

图 2.9 中的上图显示女性预测，下图显示男性预测。请注意，三种药物的相对效应在**性别**各水平之间是一致的。这是因为模型中没有交互作用项。有关包含交互作用的示例，请参见图 2.11。

图 2.9 不含交互作用的双因子方差分析的模型拟合



包含交互作用的双因子方差分析

要输入的效应：A, B, A*B

1. 在“选择列”列表中，选择两个名义型或有序型效应：A 和 B。
2. 选择**宏 > 完全析因**。

或者：

1. 在“选择列”列表中，选择两个名义型或有序型效应：A 和 B。
2. 点击**添加**。
3. 在“选择列”列表中，再次选择 A 和 B，然后点击**交叉**。

包含交互作用的双因子方差分析的示例

您有关于爆米花产量的数据，而且您关注爆米花的类型和包型大小是否对产量有影响。

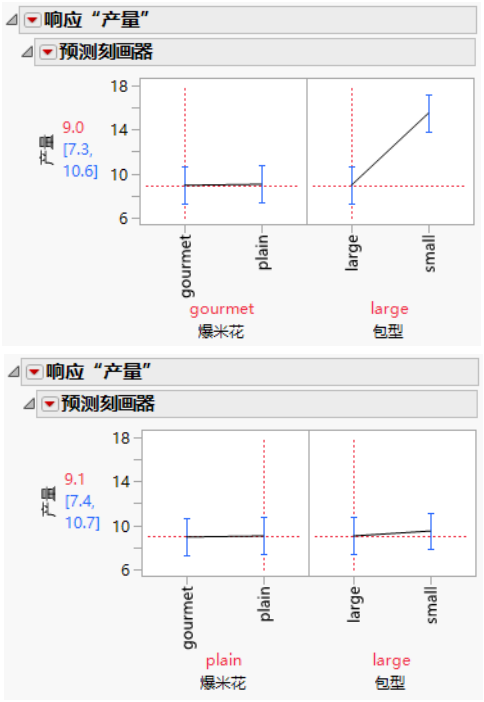
1. 选择**帮助** > **样本数据库**，然后打开 Popcorn.jmp。
2. 选择**分析** > **拟合模型**。
3. 在 “选择列”列表中，选择**产量**并点击 Y。
4. 在 “选择列”列表中，选择**爆米花**和**包型**。
5. 选择**宏** > **完全析因**。

图 2.10 包含交互作用的双因子方差分析的 “拟合模型”窗口

图 2.11 显示该示例的拟合刻画器图。要获取该图，请在图 2.10 所示的“拟合模型”窗口中点击**运行**。然后，点击“响应‘产量’”红色小三角并选择**因子刻画** > **刻画器**。

在上图中，**爆米花**设置为 gourmet；在下图中，爆米花设置为 plain。请注意**包型**设置的预测值如何依赖于**爆米花**的类型。这是**爆米花**与**包型**之间交互作用的结果。

图 2.11 包含交互作用的双因子方差分析的模型拟合



三因子完全析因

要输入的效应：A, B, C, A*B, A*C, B*C, A*B*C

1. 在 “选择列” 列表中，选择三个名义型或有序型效应：A、B 和 C。
2. 选择**宏** > **完全析因**。

三因子完全析因的示例

您关注的是**速度**、**角度**和**材料**或其交互作用是否对切割工具的**损耗**有影响。

1. 选择**帮助** > **样本数据库**，然后打开 Tool Wear.jmp。
2. 选择**分析** > **拟合模型**。
3. 在 “选择列” 列表中，选择**损耗**并点击 Y。
4. 在 “选择列” 列表中，选择**速度**、**角度**和**材料**。
5. 选择**宏** > **完全析因**。

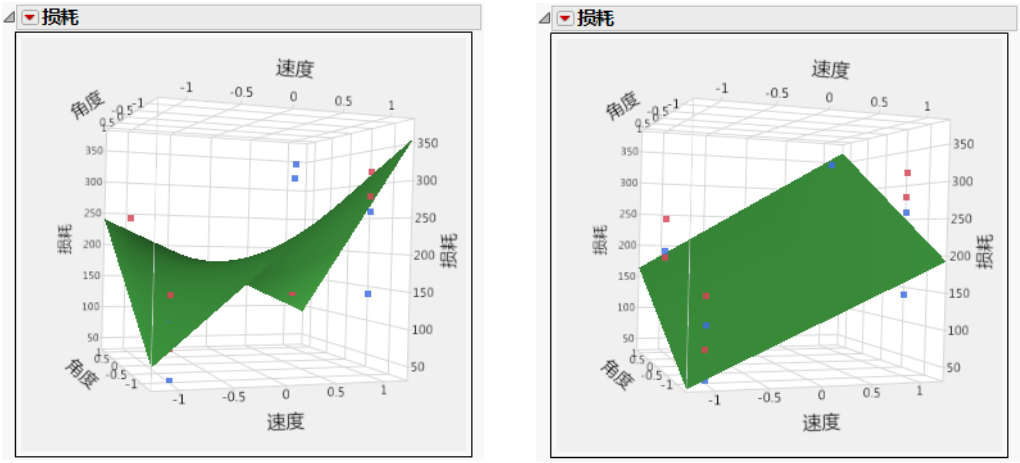
图 2.12 三因子完全析因的 “拟合模型” 窗口

图 2.13 中的 “曲面刻画器” 图根据两个连续效应**速度**和**角度**显示了**损耗**的预测响应。左图显示**材料**为 A 时的预测响应；右图显示**材料**为 B 时的预测响应。**材料**为 A 时对应的点显示为红色，**材料**为 B 时对应的点显示为蓝色。响应曲面形式在**材料**各水平之间的差异是三因子交互作用的结果。

要获取 “曲面刻画器” 图，请在图 2.12 所示的 “拟合模型” 窗口中点击**运行**。请点击 “响应‘损耗’” 红色小三角并选择**因子刻画 > 曲面刻画器**。要向图中添加点，请打开 “外观” 面板并点击**实际值**。若想要使点显得更大，请在图中右击，然后选择**设置**并调整**标记大小**。

要同时为材料 A 和材料 B 显示图，请使用 “自变量” 面板中标为**材料**的滑块，针对材料 A 将滑块设置为 0，针对材料 B 将滑块设置为 1。请注意，该表包含两个可生成 “曲面刻画器” 图的数据表脚本：**预测和曲面刻画器**和**两种材料的曲面刻画器**。

图 2.13 三因子完全析因设计的模型拟合 - 左图为材料 A，右图为材料 B



协方差分析，斜率相等

在此您关注的是：在 X 为协变量时检验 A 的影响。假定您有理由相信 X 对响应的影响不依赖于 A 的水平。

要输入的效应：A, X

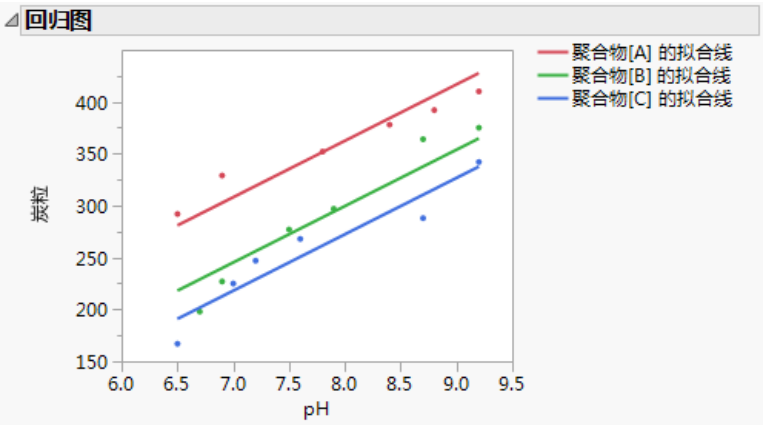
1. 在“选择列”列表中，选择一个名义型或有序型效应 A，以及一个连续效应 X。
2. 点击**添加**。

图 2.14 显示了 Cleansing.jmp 样本数据表中的数据。您关注的是哪一种**聚合物**能最有效地去除清洗油箱上的**炭粒**。不过，您怀疑油箱的 pH 对于去除炭粒也有影响。该图显示协方差拟合分析。在此假定与 pH 和**炭粒**相关的斜率在**聚合物**各水平之间是相等的。

您可以使用以下步骤获取图 2.14 中的图。在“拟合模型”窗口中，输入**炭粒**作为 Y，并在**构造模型效应**框中输入 pH 和**聚合物**。点击**运行**。“拟合最小二乘法”报表中会显示“回归图”。要对各点着色，请选择**行 > 按列设定颜色或标记**，再从**按列标记**列表中选择**聚合物**，然后点击**确定**。

不过，更完整的分析指明 pH 和**聚合物确实**彼此交互影响**炭粒**。正确的模型拟合显示在第 56 页的“协方差分析，斜率不等”中。

图 2.14 “协方差分析，斜率相等”的模型拟合



协方差分析，斜率不等

在此您同样关注的是：在 X 为协变量时检验 A 的影响。但您构造模型以便允许一种可能性：X 对响应的影响依赖于 A 的水平。

要输入的效应：A, X, A*X

1. 在“选择列”列表中，选择一个名义型或有序型效应 A，以及一个连续效应 X。
2. 选择**宏** > **完全析因**。

或者：

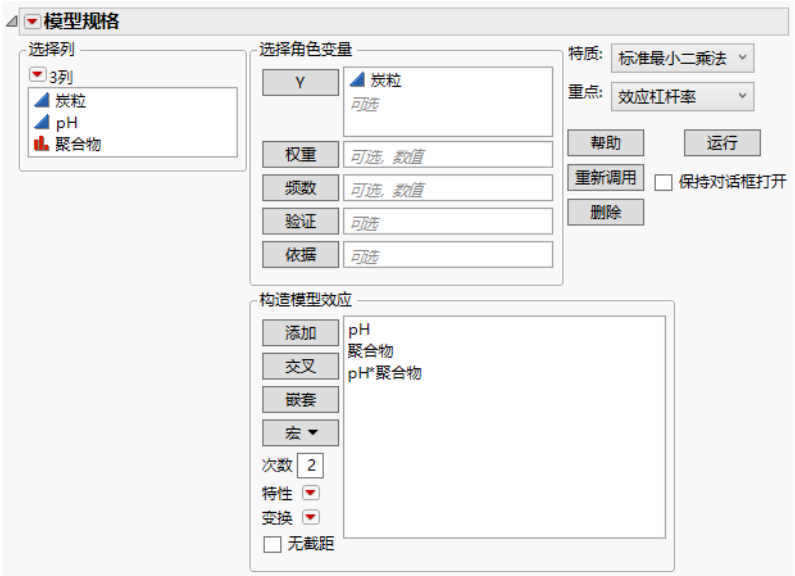
1. 在“选择列”列表中，选择一个名义型或有序型效应 A，以及一个连续效应 X。
2. 点击**添加**。
3. 在 “选择列”列表中，再次选择 A 和 X，然后点击**交叉**。

“协方差分析，斜率不等”的示例

您关注的是三种聚合物 （**聚合物**）中是否有任何聚合物影响清除炭粒 （**炭粒**）。油箱 pH 作为协变量包括在内，因为它可能影响聚合物清洗油箱的能力。针对三种**聚合物**类型对 pH 和**炭粒**之间的关系建模时，您允许存在可能不同的斜率。

1. 选择**帮助** > **样本数据库**，然后打开 Cleansing.jmp。
2. 选择**分析** > **拟合模型**。
3. 在 “选择列”列表中，选择**炭粒**并点击 Y。
4. 在 “选择列”列表中，选择 pH 和**聚合物**。
5. 选择**宏** > **完全析因**。

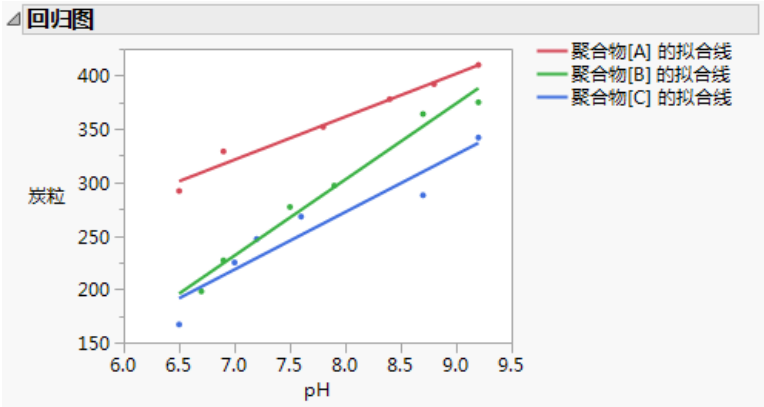
图 2.15 “协方差分析，斜率不等”的“拟合模型”窗口



当您点击**运行**时，“拟合最小二乘法”报表随即显示。“效应检验”报表指示 pH 与**聚合物**之间的交互作用显著，应包括在模型中。

报表中提供的“回归图”显示在图 2.16 中。该图显示点和模型拟合。交互作用允许表示 pH 与**炭粒**相关的线条的斜率依赖于**聚合物**。请注意，尽管存在该交互作用，在关注的范围内，“聚合物 A”始终保持最高的清除影响。若要如图 2.16 所示对各点着色，请选择**行 > 按列设定颜色或标记**，再从**按列标记**列表中选择**聚合物**，然后点击**确定**。

图 2.16 “协方差分析，斜率不等”的模型拟合



双因子嵌套随机效应模型

考虑包含 A 和 B 这两个因子的模型，但 B 嵌套在 A 中。尽管有时嵌套效应被视为固定效应，但多数情况下嵌套效应被视为随机效应。出于此原因，在如下所述的模型中，嵌套效应输入时作为随机效应。

要输入的效应：A, B[A]& 随机

1. 在“选择列”列表中，选择两个名义型或有序型效应：A 和 B。
2. 点击**添加**。
3. 要将 B 嵌套在 A 中：在“构造模型效应”列表中，选择 B。在“选择列”列表中，选择 A。这两个效应会突出显示。
4. 点击**嵌套**。
5. 当 B[A] 在 “构造模型效应”列表中突出显示时，选择**特性 > 随机效应**。

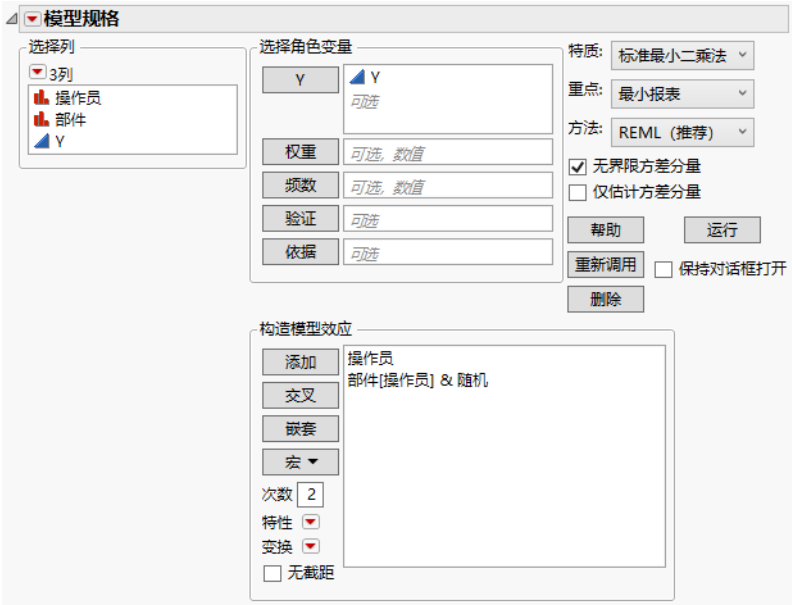
双因子嵌套随机效应模型的示例

作为测量系统分析研究的一部分，测量了 24 个随机选取的部件。这些部件平均分配给六个操作员，他们通常负责测量这些部件。针对分配的四个部件中的每一个部件，每个操作员都要进行三次独立的测量。

由于一个操作员测量的部件就只是这一特定操作员负责测量的，所以**部件**嵌套在**操作员**中。由于部件是随机生成的样本，**部件**被视为随机效应。由于这六个特定的操作员是关注对象，所以**操作员**被视为固定效应。指定适当的模型。

1. 选择**帮助 > 样本数据库**，然后打开 Variability Data/2 Factors Nested.jmp。
2. 选择**分析 > 拟合模型**。
3. 在 “选择列”列表中，选择 Y 并点击 Y。
4. 在 “选择列”列表中，选择**操作员**和**部件**。
5. 点击**添加**。
6. 要将**部件**嵌套在**操作员**中：在 “构造模型效应”列表中选择**部件**。在 “选择列”列表中，选择**操作员**。这两个效应应突出显示。
7. 点击**嵌套**。
8. 当 “部件 [操作员]”在 “构造模型效应”列表中突出显示时，选择**特性 > 随机效应**。

图 2.17 双因子嵌套随机效应模型的 “拟合模型”窗口



9. 点击**运行**获取 “拟合最小二乘法”报表。

图 2.18 显示了两个图。第一个图是“变异性图”，其中显示了每个**操作员**针对四个部件中的每一个部件给出的三个测量值。水平线段显示每个**操作员**的测量均值。

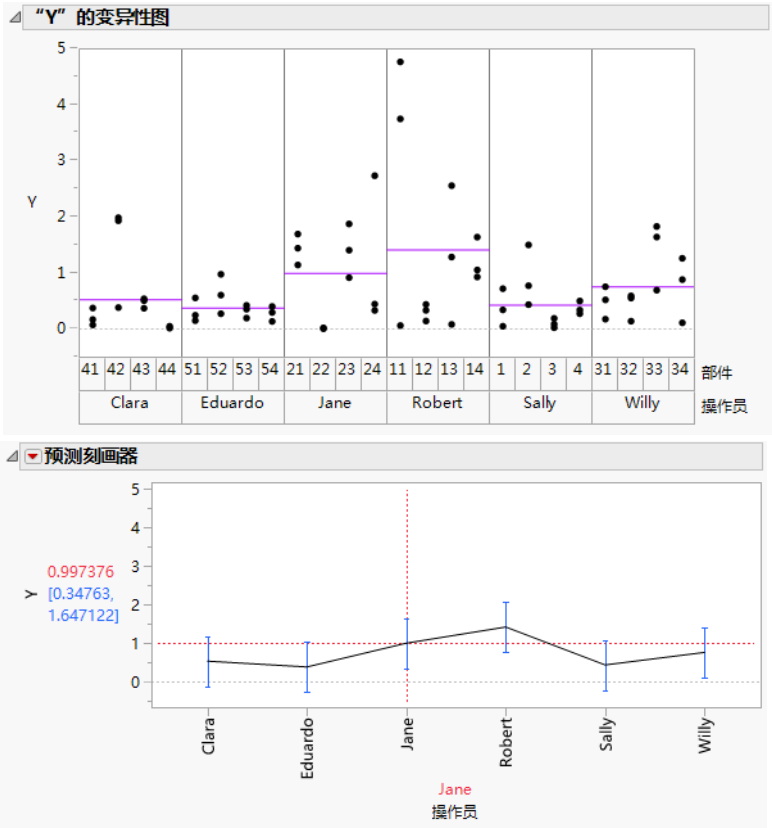
要构造图 2.18 中的“变异性图”，请在 2 Factors Nested.jmp 样本数据表中运行数据表脚本**变异性图 — 嵌套**。点击“变异性量具”红色小三角，撤销选择**显示极差条**，然后选择**显示组均值**。

第二个图是**操作员**的“拟合最小二乘法”报表“预测刻画器”图。该图显示每个操作员的预测响应。设置在 Jane 位置的红色垂直虚线指示 Jane 的预测响应为 0.997。您可以看到“预测刻画器”图中提供的模型预测与“变异性图”中的原始数据之间的对应关系。

要获取“预测刻画器”图，请点击“响应‘Y’”红色小三角并选择**因子刻画 > 刻画器**。

这些图显示如何对每个**操作员**的预测测量值建模。不过，请记住您关注的不仅仅是操作员测量部件的方式是否不同；您还同时关注部件测量值本身是否存在变异性，这要求估计与**部件**关联的方差分量。

图 2.18 双因子嵌套随机效应模型的模型拟合



三因子完全嵌套随机效应模型

考虑包含 A、B 和 C 这三个因子的模型，其中，B 嵌套在 A 中，C 嵌套在 A 和 B 中。同时要将 B 和 C 都视为随机效应。

要输入的效应：A, B[A]& 随机 , C[A,B]& 随机

1. 在“选择列”列表中，选择三个名义型或有序型效应：A、B 和 C。
2. 点击**添加**。
3. 要将 B 嵌套在 A 中：在“构造模型效应”列表中，选择 B。在“选择列”列表中，选择 A。这两个效应会突出显示。
4. 点击**嵌套**。
5. 要将 C 嵌套在 A 和 B 中：在“构造模型效应”列表中，选择 C。在“选择列”列表中，选择 A 和 B。这三个效应会突出显示。
6. 点击**嵌套**。
7. 当 B[A] 和 C[A,B] 在 “构造模型效应”列表中均突出显示时，选择**特性 > 随机效应**。

简单裂区或重复测量模型

在此，A 是整区变量，B[A] 是整区 ID，C 是裂区（或重复测量）变量。

要输入的效应：A, B[A]& 随机 , C, C*A

1. 在“选择列”列表中，选择两个名义型或有序型效应：A 和 B。
2. 点击**添加**。
3. 要将 B 嵌套在 A 中：在“构造模型效应”列表中，选择 B。在“选择列”列表中，选择 A。这两个效应会突出显示。
4. 点击**嵌套**。
5. 在 “构造模型效应”列表中，选择 B[A]。
6. 选择**特性 > 随机效应**。
7. 在 “选择列”列表中，选择第三个名义型或有序型效应 C。
8. 点击**添加**。
9. 在“构造模型效应”列表中，选择 C。在“选择列”列表中，点击 A。这两个效应应突出显示。
10. 点击**交叉**。

简单重复测量模型的示例

您有关于六种动物中的每种动物在四个季节中的每一个季节中迁徙的距离数据。共有两个物种并且动物标识符嵌套在**物种**中。由于这六种动物足以代表较大的种群，您决定将**研究课题**视为随机效应。您想要对响应**英里数**建模，将其作为**物种**和**季节**的函数，并考虑到每种动物都有重复测量这一事实。

1. 选择**帮助 > 样本数据库**，然后打开 Animals.jmp。
2. 选择**分析 > 拟合模型**。
3. 在 “选择列”列表中，选择**英里数**并点击 Y。
4. 在 “选择列”列表中，选择**物种**和**研究课题**。
5. 点击**添加**。
6. 要将研究课题嵌套在物种中：在 “构造模型效应”列表中，选择**研究课题**。在 “选择列”列表中，选择**物种**。这两个效应应突出显示。
7. 点击**嵌套**。
8. 在 “构造模型效应”列表中，选择**研究课题 [物种]**。
9. 选择**特性 > 随机效应**。
10. 在 “选择列”列表中，选择**季节**。
11. 点击**添加**。
12. 在 “构造模型效应”列表中，选择**季节**。在 “选择列”列表中，点击**物种**。这两个效应应突出显示。

13. 点击**交叉**。

图 2.19 简单重复测量模型的 “拟合模型” 窗口

双因子响应曲面模型

要输入的效应：X&RS, Z&RS, X*X, X*Z, Z*Z

1. 在 “选择列” 列表中，选择两个连续效应，X 和 Z。
2. 选择**宏** > **响应曲面**。

双因子响应曲面模型的示例

在 Odor Control Original.jmp 样本数据表中，您想要拟合响应曲面，以便对作为**温度**和**气 / 液比**的函数的**响应气味**建模。（您可以包括**高度**，如数据表脚本**响应曲面**所示，但在此演示中请不要包括。）

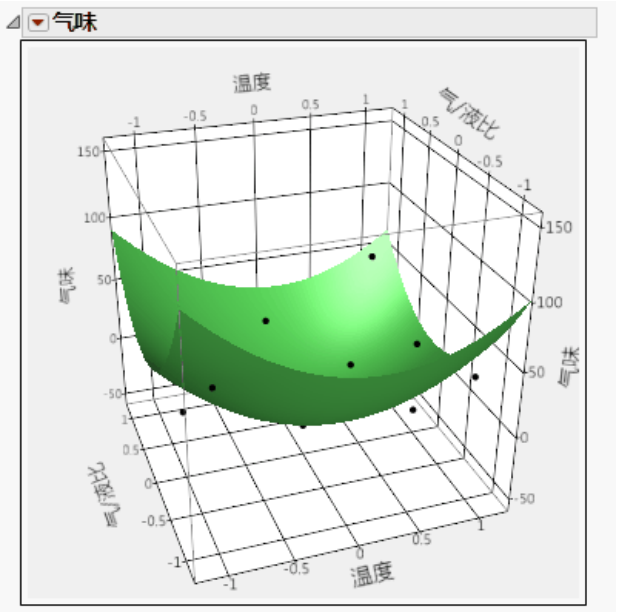
1. 选择**帮助** > **样本数据库**，然后打开 Odor Control Original.jmp。
2. 选择**分析** > **拟合模型**。
3. 在 “选择列” 列表中，选择**气味**并点击 Y。
4. 在 “选择列” 列表中，选择**温度**和**气 / 液比**。
5. 选择**宏** > **响应曲面**。

图 2.20 双因子响应曲面模型的“拟合模型”窗口

6. 点击**运行**。

图 2.21 显示了针对 Odor Control Original.jmp 样本数据表，数据和数据的二次响应曲面拟合的“曲面刻画器”图。要获得该图，请点击“响应‘气味’”红色小三角并选择**因子刻画 > 曲面刻画器**。要显示点，请点击展开图标以打开**外观**面板，然后点击**实际值**。若想要使点显得更大，请在图中右击，然后选择**设置**并调整**标记大小**。

图 2.21 两个变量的二次多项式的模型拟合



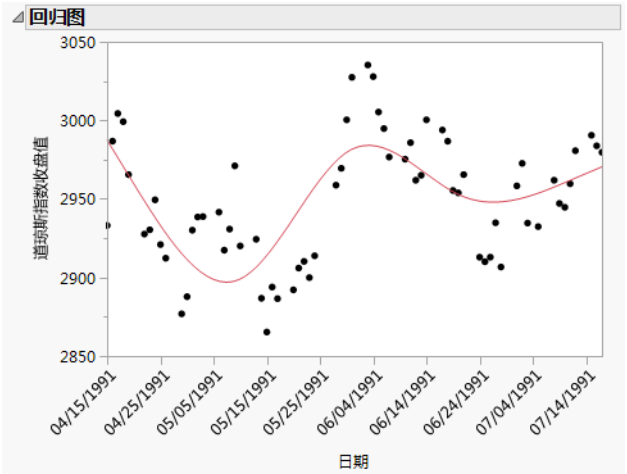
纽结样条效应

要输入的效应：X& 已纽结

1. 在“选择列”列表中，选择连续效应 X。
2. 点击**添加**。
3. 在 “构造模型效应”列表中选择 X。
4. 选择**特性 > 纽结样条效应**。
5. 在出现的菜单中，指定结点数或接受默认数值。
6. 点击**确定**。

图 2.22 显示了针对 XYZ Stock Averages (plots).jmp 样本数据表中数据的模型拟合的“回归图”。在此，向**日期**分配了“纽结样条效应”并指定了五个结点。**道琼斯指数收盘值**是响应。

图 2.22 带五个结点的纽结样条的模型拟合



第 3 章

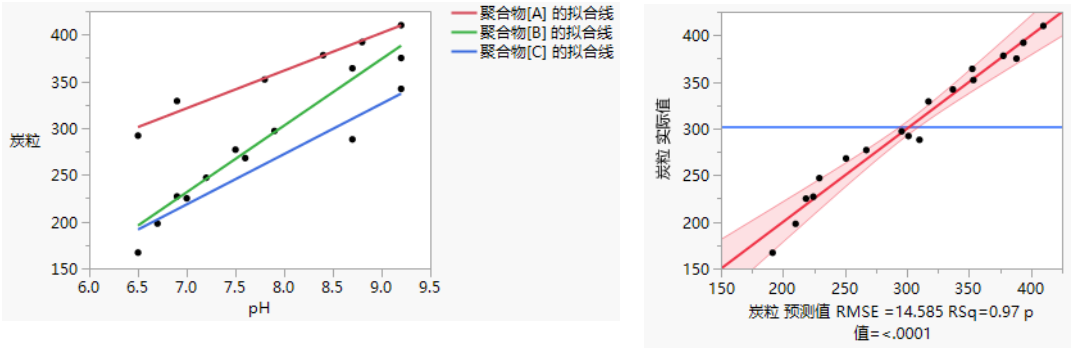
标准最小二乘法模型

分析常用种类的模型

“拟合模型”平台中的“标准最小二乘法”特质可以拟合大范围的标准模型。这些模型包括回归、方差分析、协方差分析和混合模型，以及通常用于分析设计实验的模型。使用“标准最小二乘法”特质可以通过最小二乘法为连续响应数据构造线性模型，或使用限制最大似然 (REML) 法为随机效应构造线性模型。

分析结果受到极具吸引力的动态可视化工具（如刻画器、等高线图和曲面图）的支持（请参见《刻画器指南》）。这些直观的演示可促进、补充和支持您对模型的理解。利用它们可同时优化若干响应并探索噪声的影响。

图 3.1 标准最小二乘法的图示例



目录

使用标准最小二乘法的示例 70
启动“标准最小二乘法”特质 72
“拟合模型”启动窗口 72
“拟合模型”启动窗口中的“标准最小二乘法”选项 74
标准最小二乘法中的验证 75
缺失值 76
“拟合最小二乘法”报表 76
单个和多重响应的对比 76
与重点有关的报表结构 77
特殊报表 77
“最小二乘法拟合”选项 80
“拟合组”选项 81
响应选项 81
回归报表 82
拟合汇总 83
方差分析 83
参数估计值 84
效应检验 85
效应详细信息 86
失拟 100
估计值 101
显示预测表达式 103
排序后的估计值 103
扩展后的估计值 106
指标参数化估计值 108
序贯检验 109
定制检验 109
多重比较 112
比较斜率 124
联合因子检验 124
逆预测 125
Cox 混料 129
参数功效 130
估计值的相关性 131
名义型效应的编码 132
效应筛选 133
统一尺度估计值和连续项的编码 134

图选项 135
“正态图”报表 139
“Bayes 图”报表 140
“Pareto 图”报表 141
因子刻画 142
刻画器指南 143
交互作用图 144
等高线刻画器 145
混料刻画器 146
立方图 147
Box Cox Y 变换 148
曲面刻画器 150
行诊断 151
杠杆图 152
Press 155
保存列 155
预测公式 158
“效应汇总”报表 158
混合和随机效应模型报表和选项 162
混合模型和随机效应模型 162
限制最大似然 (REML) 方法 165
EMS（传统）模型拟合报表 169
模型项之间具有线性相依性的模型 171
奇异性详细信息 171
“参数估计值”报表 172
“效应检验”报表 173
示例 173
“标准最小二乘法”特质的统计详细信息 175
重点规则 175
“定制检验”示例的详细信息 176
估计值的相关性 176
杠杆图详细信息 177
Kackar-Harville 校正 179
功效分析 180

使用标准最小二乘法的示例

在一项关于治疗某种疾病的药效研究中，三十位患者被随机划分到三个组中，每组 10 人。其中两个组给予了药物（**药物** a 和**药物** d），第三组给予了安慰剂（**药物** f）。对每个患者都进行一次治疗前的测量 x 和一次治疗后的测量 y。治疗前得分 x 作为协变量包括在内，以解释整个病程期间患者间的差异。（该示例选自 Snedecor and Cochran (1967, p. 422)。

您想要确定三个**药物**组之间是否有差异。使用响应 y、模型效应**药物**和 x，以及**药物**与 x 之间的交互作用来构造模型。该交互作用或许能解释各种药物因病程而异导致不同效应的情形。（有关“拟合模型”窗口和各种特质的背景信息，请参见第 29 页的 “模型规格”一章。）

1. 选择**帮助** > **样本数据库**，然后打开 Drug.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 y 并点击 Y。

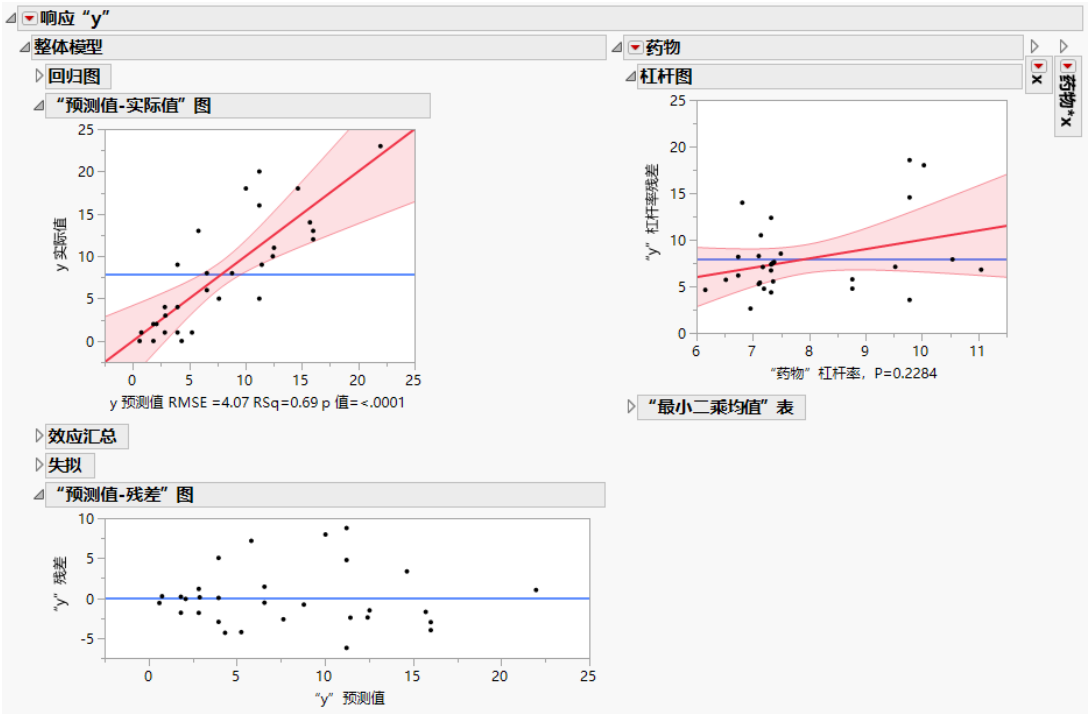
   当您将该列添加为 Y 后，拟合**特质**变为“标准最小二乘法”。随之会添加**重点**选项，该选项的值为 “效应杠杆率”，但您可以按需更改。
4. 选择**药物**和 x。当这两个效应在 “选择列”列表中突出显示时，点击**宏**并选择**完全析因**。该宏向 “构造模型效应”列表添加这两个效应及其双因子交互作用。

图 3.2 完成的 “拟合模型”启动窗口

5. 点击**运行**。

“拟合最小二乘法”报表显示在图 3.3 中。 请注意，由于空间有限，其中的有些报表已关闭。“预测值-实际值”图、“预测值-残差”图和杠杆图在模型拟合和基本假设方面未显示任何差异。

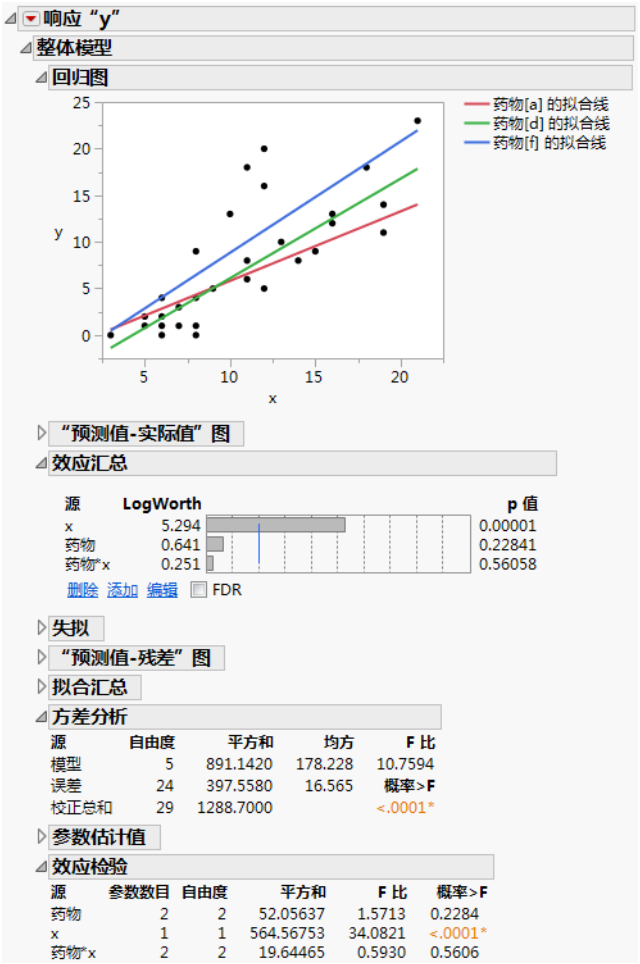
图 3.3 “拟合最小二乘法”报表，其中显示了用于评估模型拟合的图



由于模型拟合不存在任何明显问题，您现在可以解释统计检验。图 3.4 显示了相关报表。总体模型显著，如“方差分析”报表所示。

尽管“回归图”表明**药物**与治疗前测量 x 交互，但“效应检验”报表中的“概率>F”值不支持该结论。“效应检验”报表还显示 x 在解释 y 时显著，但**药物**不显著。该研究未检测到三个组之间的差异。不过，您不能推断出**药物**无效果的结论。各种药物可能有不同效果，但该研究规模不足以检测出这种差异。

图 3.4 “拟合最小二乘法”报表，其中显示了用于评估显著性的报表



启动“标准最小二乘法”特质

标准最小二乘法是您可以在“拟合模型”启动窗口中选择的若干分析方法之一。

本节说明您如何在“拟合模型”启动窗口中将选择标准最小二乘法作为拟合方法，同时还涵盖了特定于该选择的选项。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“拟合模型”启动窗口

您可以在“拟合模型”启动窗口中指定同时包含固定效应和随机效应的模型。选项根据您所指定的模型的性质而有所不同。

仅固定效应

要使用 “标准最小二乘法”特质拟合模型，请选择**分析 > 拟合模型**，然后从 “特质”列表中选择**标准最小二乘法**。当您在 Y 列表中输入一个或多个连续变量后，“特质”默认设置为 “标准最小二乘法”。但是请注意，还有其他选择可用于连续 Y 变量。在您为 “标准最小二乘法”拟合仅指定固定效应时，“拟合模型”启动窗口如图 3.5 所示。本例使用 Big Class.jmp 样本数据表演示了启动窗口。

图 3.5 固定效应模型的 “拟合模型”启动窗口

在 “特质”列表中选择 “标准最小二乘法”特质后，还会显示 “重点”选项。“重点”选项控制在初始报表窗口中提供的报表。根据包括的模型效应，JMP 推断出您可能需要的报表。不过，通过从默认报表的红色小三角菜单中选择适当的选项，可以显示未显示在初始报表中的任何报表。

有关可用于每个 “重点”选项的报表的详细信息，请参见第 75 页的 “标准最小二乘法的 “重点”选项”。

随机效应

若指定的模型包含一个或多个随机效应，“拟合模型”启动窗口会提供更多的选项。考虑 Machine.jmp 样本数据表。在六个随机挑选的工人中，每一个工人都在每台机器（共三台）上执行操作，他们的产出将被评级。您关注的是估计各工人的等级中的变异，而不是确定这六个特定的工人的等级是否存在差异。指定模型时，您需要将**工人**和**机器*工人**视为随机效应。

该模型的 “拟合模型”启动窗口显示在图 3.6 中。将 “随机效应”特性应用到**工人**时，“方法”选项以及与方差分量相关的两个选项将显示在 “拟合模型”启动窗口中。

图 3.6 包含随机效应的模型的 “拟合模型”启动窗口

“拟合模型”启动窗口中的“标准最小二乘法”选项

下面的 “拟合模型”启动窗口选项是 “标准最小二乘法”特质特有的。

重点 控制显示在初始报表窗口中的报表和图的类型。请参见第 75 页的 “标准最小二乘法的“重点”选项”。

方法 （仅当指定随机效应时才显示。）使用以下一种方法估计模型：

REML 请参见第 167 页的“REML 方差分量估计值”。

EMS 期望均方，亦称矩量法。请参见第 169 页的“EMS（传统）模型拟合报表”。

无界限方差分量 （仅在将 REML 选作 “方法”时才出现。）允许方差分量估计值为负数。默认选定该选项。若您关注的是固定效应，该选项应保持选定状态，因为将方差估计值的边界定为 0 将导致固定效应检验发生偏倚。请参见第 165 页的 “负方差”。

仅估计方差分量 （仅在将 REML 选作 “方法”时才出现。）提供仅显示方差分量估计值的报表。请参见第 165 页的 “仅估计方差分量”。

分别拟合 （仅对有多个 Y 变量且没有随机效应的模型显示。）使用非缺失的所有行为每个 Y 变量拟合单独模型。请参见第 76 页的 “缺失值”。

标准最小二乘法的“重点”选项

“重点”列表中的三个选项控制构成“标准最小二乘法”特质的初始报表中显示的图和报表的类型。请参见下面的说明。JMP 基于数据表中的行数、在“构造模型效应”列表中输入的效应数，以及应用于效应的特性来选择默认重点。您可以根据自身需要更改该重点选项。有关 JMP 如何选择重点的详细信息，请参见第 175 页的“重点规则”。

初始报表打开后，您可以从平台报表窗口的红色小三角菜单中添加其他报表和图。

提供以下重点选项：

效应杠杆率 显示杠杆图和残差图，以及包含关于模型拟合详细信息的报表。若您的主要关注点是模型拟合，该选项很有用。

“效应杠杆率”是最全面的选项。该重点将报表划分为与“整体模型”相关的报表和与单个模型效应相关的报表。“整体模型”报表位于报表窗口左角的“整体模型”标题之下，效应报表则位于右侧。

效应筛选 显示经过排序或统一尺度的参数估计值报表（适用时还显示图形）、“预测刻画器”以及包含有关模型拟合详细信息的报表。若您有许多效应而您最初的关注点是发现哪些效应是活跃的（就像在筛选设计中那样），该选项很有用。

若选定“效应筛选”，则计算 Box-Cox 变换。若估计的 λ 的置信区间不包含 1，则显示“Box-Cox 变换”报表。请参见第 148 页的“Box Cox Y 变换”。

最小报表 仅显示回归图和包含有关模型拟合详细信息的报表。将“随机效应”特性应用于任何模型效应时，该“重点”是默认选项。

该选项的信息最少，最简明。您可以从红色小三角菜单中请求您尤为关注的报表。

要更改对所有“重点”选项显示哪些报表或图，请使用平台首选项。转至**文件 > 首选项 > 平台 > 拟合最小二乘法**，然后使用**设置**复选框：

- 要防止某个选项出现在报表中，请在该选项旁边选择**设置**但不选择该选项。
- 要确保某个选项出现在报表中，请选择**设置**并选择该选项。

JMP PRO 标准最小二乘法中的验证

在 JMP Pro 中，您可以在“拟合模型”窗口中指定“验证”列。验证列必须具有数值数据类型，应至少包含两个非重复值。

- 若列包含两个值，则较小的值定义训练集，较大的值定义验证集。
- 若列包含三个值，则这些值按大小递增的顺序定义训练集、验证集和测试集。
- 若列包含四个或更多非重复值，则将忽略“验证”列。

JMP Pro 中的“标准最小二乘法”特质支持使用“验证”列。若输入某个“验证”列，则提供“交叉验证”报表。请参见第 79 页的“ “交叉验证”报表”。

若输入“验证”列，则在报表的图中，验证集和测试集中的观测将分别标记为“v”和“t”。

有关如何在 JMP 建模平台中使用“验证”列的详细信息，请参见《预测和专业建模》。

缺失值

默认情况下，Y 或任何模型效应具有缺失值的行将从分析中排除。

注意：JMP 在“拟合模型”窗口的“模型规格”红色小三角菜单中提供了“信息性缺失”选项。“信息性缺失”支持您使用缺失模型效应的行拟合模型。请参见“模型规格”一章中第 45 页的“信息性缺失”。

若您的模型包含随机效应，则默认情况下分别拟合 Y 值。各报表显示在“拟合组”报表中。

假定您的模型仅包含固定效应，则以下陈述成立：

- 您指定了不止一个 Y 响应。
- 其中某些 Y 响应具有缺失值。
- 您没有在“拟合模型”启动窗口中选择“分别拟合”选项。

之后， JMP 会提示您选择以下一个选项：

- “分别拟合”使用对该特定 Y 为非缺失的所有行来拟合每个 Y。
- “同时拟合”仅使用对所有 Y 变量都为非缺失的那些行来拟合每个 Y。

若选择“分别拟合”，“拟合组”报表会包含 Y 变量的各个报表。您可以从“拟合组”红色小三角菜单中选择刻画器，以便在同一刻画器中查看所有 Y 变量。或者，您可以从单个 Y 变量报表中选择某个刻画器，以便在该刻画器中仅查看该变量。

若选择“同时拟合”，“拟合最小二乘法”报表会包含每个 Y 变量的各个报表。不过，该报表某些部分是针对所有 Y 变量组合起来的形式：“效应汇总”和“刻画器”。

“拟合最小二乘法”报表

使用 “标准最小二乘法”特质拟合模型时，您会得到 “拟合最小二乘法”报表。该报表的内容由数据性质和您在 “拟合模型”启动窗口中的选择来决定。

提示：要始终查看默认不显示的报表，请使用**文件 > 首选项 > 平台 > 拟合最小二乘法**来选择这些报表。

单个和多重响应的对比

拟合单个响应变量 Y 时，“拟合最小二乘法”窗口将各个详细报表组织到一个名为“响应‘Y’”的报表中。拟合若干响应时，各个响应的报表通常组织到一个名为“最小二乘法拟合”的报表中。不过，若存在缺失响应数据，而您选择了“分别拟合”选项，则各个响应的报表将组织到一个名为“拟合组”的报表中。

与重点有关的报表结构

当您在“拟合模型”启动窗口中选择“效应杠杆率”重点时，给定响应的报表排列在若干列中。左列包含“整体模型”报表，其中包含与模型相关的其他报表。模型中每个效应的报表显示在“整体模型”报表右侧的列中。

当您在“拟合模型”启动窗口中选择“效应筛选”或“最小报表”重点时，每个响应的所有报表都排列在左列中。

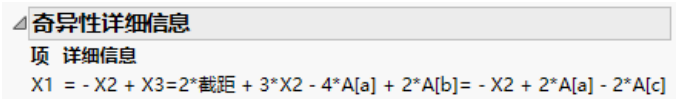
特殊报表

本节说明根据数据结构或您针对效应特性所作选择而提供的报表。

奇异性详细信息

当模型效应之间存在线性相依性时，“奇异性详细信息”报表将显示为“响应”报表标题下方的第一个报表。它包含模型项满足的线性函数的表。这些函数定义模型项之间的别名关系。图 3.7 显示了 Singularity.jmp 样本数据表的示例。

图 3.7 “奇异性详细信息”报表



若效应间存在线性相依性，则某些模型项的估计值不唯一。请参见第 171 页的“模型项之间具有线性相依性的模型”。

“响应曲面”报表

若模型中的某个效应具有响应曲面 (&RS) 或混料响应曲面（&RS& 混料）特性，此时将提供“响应曲面”报表。请参见图 3.8，了解有关 Tiretread.jmp 样本数据表的 “响应曲面”报表的示例。

图 3.8 “响应曲面”报表

响应“磨损”

响应曲面

系数

	硅石	硅烷	硫磺	磨损
硅石	-16.0384	1.025	28.5	32.987289
硅烷	.	-0.034471	1.575	1.7880765
硫磺	.	.	-6.288485	21.813077

解

变量	临界值
硅石	0.5982746
硅烷	36.317583
硫磺	0.9573951

解是鞍点
临界值超出数据范围
对应于解的预测值 102.31882

典型曲率

特征值和特征向量

特征值	4.1100	-0.2470	-26.2243
硅石	0.56690	-0.12971	0.81351
硅烷	0.22090	0.97530	0.00157
硫磺	0.79361	-0.17882	-0.58155

“系数”表

“系数”表显示为“响应曲面”报表的第一个部分，该表提供估计模型参数的简要汇总。前面几列提供二阶项的系数。最后一列提供线性项的系数。要查看完整的预测表达式，请从该报表的红色小三角菜单中选择**估计值** > **显示预测表达式**。

“解”报表

“解”报表提供了可能存在的临界值（最大值、最小值或鞍点）以及该点处的预测值。若解落在数据范围之外，它还会给出相应提醒。

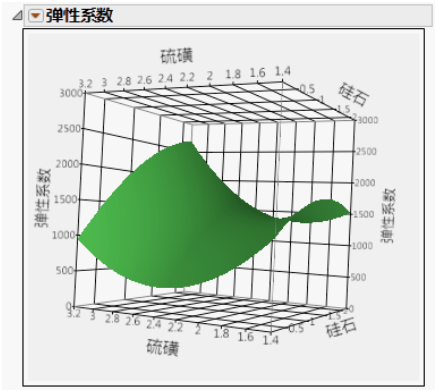
“典型曲率”报表

二阶参数估计值矩阵的特征值和特征向量确定曲率类型。特征向量显示曲面的主方向，包括最大曲率和最小曲率的方向。

特征值在 “典型曲率”表的第一行提供。

- 若特征值为负，响应曲面将从最大值向下弯曲。
- 若特征值为正，曲面形状将从最小值向上弯曲。
- 若既有正特征值也有负特征值，该曲面将呈鞍状，在一个方向上向上弯曲，另一个方向上向下弯曲。请参见图 3.9，了解使用 Tiretread.jmp 样本数据表的示例。

图 3.9 具有鞍状曲面的曲面刻画器图



特征值下列出的特征向量显示主轴的方向。特征值的绝对值越大，响应曲面在其关联方向上的曲率越大。有时会出现零特征值。该特征值意味着沿着相应特征向量描述的方向，拟合曲面是扁平的。

注意：对于包含 20 个以上因子的响应曲面模型，不显示“响应曲面”报表。不提供错误消息或警报。有关响应曲面设计的详细信息，请参见《实验设计指南》。

混合和随机效应模型报表

当您在“拟合模型”启动窗口中指定随机效应时，将显示“方法”列表。该列表提供两种拟合方法：REML（推荐）和 EMS（传统）。根据您选择的模型和方法，还可能显示其他报表以及“保存列”和“刻画器”选项。

有关 REML 方法报表的详细信息，请参见第 165 页的“限制最大似然 (REML) 方法”。有关 EMS 方法报表的详细信息，请参见第 169 页的“EMS（传统）模型拟合报表”。

“交叉验证”报表

JMP PRO 在“拟合模型”启动窗口中输入“验证”列后，将提供“交叉验证”报表。该报表为验证中使用的每个集提供以下信息：

源 将集标识为“训练”、“验证”或“测试”集。

R 方 给定集中的观测相对于使用训练集得出的模型计算出的 R 方值。对于该训练集，这是常规 R 方值。

对于每个训练集、验证集和测试集，该 R 方值计算如下：

- 对于给定集中的每个观测，计算预测误差。这是实际响应与训练集模型预测的响应之间的差值。
- 计算预测误差的平方和以得到 $SSE_{源}$，其中的下标源表示任意训练集、验证集或测试集。
- 对源集中的观测的实际响应与它们的均值的差值求平方和。用 $SST_{源}$ 表示该值。

– 源集的 R 方为：

$$R\ 方_{源} = 1 - \frac{SSE_{源}}{SST_{源}}$$

注意：验证集和测试集的 R 方值可能为负值。

RASE 均方预测误差的平方根。对于每个训练集、验证集和测试集， RASE 都计算如下：

– 对于给定集中的每个观测，计算预测误差。这是实际响应与训练集模型预测的响应之间的差值。

– 计算预测误差的平方和以得到 $SSE_{源}$，其中的下标源表示任意训练集、验证集或测试集。

– 用 n 表示观测数。

– RASE 是：

$$\text{RASE}_{源} = \sqrt{\frac{SSE_{源}}{n}}$$

频数 源集中的观测数。

“最小二乘法拟合”选项

您可能有不止一个 Y 且没有任何缺失响应值，也可能有不止一个 Y 但包含缺失值。若选择“同时拟合”，响应将组合到名为“最小二乘法拟合”的报表中。红色小三角菜单包括以下选项：

刻画器 在单个刻画器中显示所有响应。您可以同时查看所有响应的模型项的效应，并执行多重优化。请参见第 142 页的 “因子刻画”。

模型对话框 显示当前分析的已完成的启动窗口。

效应汇总 显示交互式“效应汇总”报表，该报表支持您在模型中添加或删除效应。请参见第 158 页的 ““效应汇总”报表”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“拟合组”选项

若您有不止一个 Y 且您选择 “分别拟合”，响应将分组在名为 “拟合组”的报表中。红色小三角菜单包括以下选项：

刻画器 在单个刻画器中显示所有响应。您可以同时查看所有响应的模型项的效应，并执行多重优化。请参见第 143 页的 “刻画器指南”。

等高线刻画器 在单个等高线刻画器中显示所有响应。您可以同时探索模型项对于所有响应的影响。

曲面刻画器 为每个响应显示单独的曲面刻画器报表。

行内排列 将平台分析报表重新排列在指定行数中。该选项多用于排列报表，以便在窗口或输出页中容纳更多报表。

按拟合优度排序 按拟合显著性 （R 方）对报表排序。该选项仅可用于将 R 方出现在平台级别的平台。例如，若您从一个启动窗口中生成了数百个单因子分析，这些分析将显示在**拟合组**中，您可以对其进行排序，使最强的关系首先显示。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

响应选项

响应的红色小三角菜单选项可支持您根据需要定制报表。

回归报表 提供基本报表和报表选项。请参见第 82 页的 “回归报表”。

估计值 提供的选项用于与参数估计值相关的进一步分析。请参见第 101 页的 “估计值”。

效应筛选 提供用于标识显著效应的报表和图。请参见第 133 页的 “效应筛选”。

因子刻画 提供刻画器、交互作用和立方图，用于检查响应与模型项有何关联。还提供用于拟合 Box-Cox 变换的图和报表。请参见第 142 页的 “因子刻画”。

行诊断 提供用于检查残差的图和报表。还报告 PRESS 统计量并提供 Durbin-Watson 检验。请参见第 151 页的 “行诊断”。

保存列 将模型结果另存为数据表中的列，但 “保存编码表”除外，该选项将结果保存在单独的数据表中。请参见第 155 页的 “保存列”。

模型对话框 为当前分析显示完成的 “拟合模型”启动窗口。

效应汇总 显示交互式“效应汇总”报表，该报表支持您在模型中添加或删除效应。请参见第 158 页的““效应汇总”报表”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

回归报表

“回归报表”菜单提供关于模型拟合、效应显著性和模型参数的汇总信息。

拟合汇总 显示或隐藏模型拟合的汇总。请参见第 83 页的“拟合汇总”。

方差分析 显示或隐藏用于将拟合模型与简单均值模型相比较的计算。请参见第 83 页的“方差分析”。

参数估计值 显示或隐藏一个报表，其中包含参数估计值和假设每个参数均为零的 *t* 检验。请参见第 84 页的“参数估计值”。

效应检验 显示或隐藏模型中固定效应的检验。请参见第 85 页的“效应检验”。

效应详细信息 显示或隐藏包含各个效应的详细信息、图和检验的报表。请参见第 86 页的“效应详细信息”。

当选定“效应杠杆率”重点选项时，每个效应自身的报表都显示在“拟合最小二乘法”报表窗口的顶部。该报表包括效应详细信息选项以及一个杠杆图。请参见第 152 页的“杠杆图”。

失拟 显示或隐藏一个检验，用来评估在可执行该检验时模型是否包含正确的效应。请参见第 100 页的“失拟”。

显示所有置信区间 显示或隐藏以下统计量的置信区间：

- “参数估计值”报表中的参数估计值
- “最小二乘均值”表中的最小二乘均值

AICc 在“拟合汇总”报表中显示或隐藏 AICc 和 BIC 值。请参见“统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

拟合汇总

“拟合汇总”报表提供 R 方计算以及 AICc 和 BIC 值之类的详细信息。

R 方 估计由模型而非随机误差解释的响应变量的变异的比例。使用相应的 “方差分析”表中的量， R 方 （亦称**多重判定系数**）计算如下：

$$\frac{\text{平方和（模型）}}{\text{平方和（校正总和）}}$$

与 R 方越接近 0 相比，R 方越接近 1 表示数据拟合效果更好。R 方接近 0 表示与响应均值相比，该模型并不是更好的响应预测变量。

注意：较低的 R 方值表明可能有一些模型中不包含的变量可以解释尚未解释的变异。不过，如果数据受大量的内在变异影响，即便是有用的回归模型也可能具有较低的 R 方值。阅读您的研究领域中的文献材料，了解有关典型 R 方值的信息。

调整 R 方 针对模型中的参数数目调整 R 方统计量。调整 R 方有利于包含不同数目参数的模型间进行比较。计算中使用自由度。使用相应 “方差分析”表中的量，调整 R 方计算如下：

$$1 - \frac{\text{均方（误差）}}{\text{平方和（校正总和）}/\text{自由度（校正总和）}}$$

均方根误差 估计随机误差的标准差。该数量是 “方差分析”报表中 “误差均方”的平方根。

注意：均方根误差通常称为 *RMSE*。

响应均值 显示响应值的总均值。

观测数 （或权重和） 显示模型中使用的观测数。

- 在以下条件下该值与数据表中的行数相同：没有缺失值、没有排除行，没有为任何列分配“权重”或“频数”角色。
- 该值是在为某列分配了“权重”角色时， “权重”列中正值的总和。
- 该值是在为某列分配了“频数”角色时， “频数”列中正值的总和。

AICc （仅当您在 “回归报表”菜单中选定了 “AICc”选项或将 “AICc”设置为 “拟合最小二乘法”首选项时才显示。）显示校正的 “Akaike 信息准则”值 (AICc)。请参见 “统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

BIC （仅当您在 “回归报表”菜单中选定了 “AICc”选项或将 “AICc”设置为 “拟合最小二乘法”首选项时才显示。 ）显示 “Bayes 信息准则”值 (BIC)。请参见 “统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

方差分析

“方差分析”报表提供用于将拟合模型与所有预测值都等于响应均值的模型进行比较的计算。

注意：若在“拟合模型”启动窗口中输入了“频数”或“权重”变量，“方差分析”报表中的条目将进行调整以符合第 36 页的“频数”和第 36 页的“权重”中的说明。

“方差分析”报表包含以下列：

源 变异的三个来源：模型、误差和校正总和。

自由度 每个变异来源关联的**自由度** (DF)。校正总和自由度始终为观测数减 1，它按以下方式分为模型自由度和误差自由度：

- 模型自由度是用于拟合模型的参数数目（截距除外）。
- 误差自由度是校正总和自由度与模型自由度之间的差值。

平方和 每个变异来源关联的平方和 (SS)。

- 总（校正总和）平方和是响应值与样本均值之间的差值平方和。它表示响应值中的总变异。
- 误差平方和是拟合值与实际值之间的差值平方和。它表示拟合模型尚未解释的变异。
- 模型平方和是校正总和平方和与误差平方和之间的差值。它表示模型解释的变异。

均方 变异的模型和误差源的均方统计量。每个均方都是平方和除以其相应自由度的结果。

注意：误差均方的平方根与 “拟合汇总”报表中的 RMSE 相同。

使用的 MSE （仅当“误差指定”为“纯误差”或“指定”时才显示。）在未选定默认误差指定时使用的均方误差。该值用于计算 “F 比”而不是 “均方”列中的均方误差。

使用的 DFE （仅当“误差指定”为“纯误差”或“指定”时才显示。）在未选定默认误差指定时使用的误差自由度。该值用于计算“概率 > F”而不是“自由度”列中的“误差自由度”。

F 比 模型均方除以误差均方。F 比是以下检验的检验统计量：检验模型是否与所有预测值都是响应均值的模型显著不同。

概率 > F 检验的 *p* 值。“概率 >F”值测量在除截距之外的所有参数均为零的情况下，获取与观测到的 F 比同样大的 F 比的概率。较小的 “概率 >F”值指示观测到的 *F* 比不太可能。这样的值被视为模型中至少有一个显著效应的证据。

参数估计值

“参数估计值”报表显示模型参数的估计值，并为每个参数提供假设该参数等于 0 的 *t* 检验。

注意：尽量获取并检验估计值，即便模型项之间存在线性相依性也不例外。此类估计值标记为“有偏”或“清零”。请参见第 171 页的“模型项之间具有线性相依性的模型”。

项 与估计参数对应的模型项。第一个项始终是截距，除非在 “拟合模型”启动窗口中选定了“无截距”选项。连续效应以数据表列的名称显示。请注意，属于高阶项一部分的连续列可能被中心化。名义型或有序型效应的水平值显示在括号中。有关名义型和有序型项的编码的

信息，请参见第 132 页的“名义型效应的编码”和“统计详细信息”附录中第 456 页的“因子模型”。

估计值 每个项的参数估计值。这些值是模型系数的估计值。若模型项之间存在线性相依性，这些估计值可能标记为“有偏”或“清零”。请参见第 171 页的“模型项之间具有线性相依性的模型”。

标准误差 每个估计参数的标准误差的估计值。

***t* 比** 检验每个参数的真实值是否为 0。*t* 比是估计值与其标准误差之比。在有关模型的一般假设下，*t* 比在原假设下服从 Student *t* 分布。

概率 >|t| 实际参数值为零（而不是双侧备择假设下它不为零）的检验的 *p* 值。

95% 下限 参数估计值的 95% 置信下限。仅当已选定“回归报表”>“显示所有置信区间”选项或是在报表中右击并选择“列”>“95% 下限”时，该列才显示。

95% 上限 参数估计值的 95% 置信上限。仅当已选定“回归报表”>“显示所有置信区间”选项或是在报表中右击并选择“列”>“95% 上限”时，该列才显示。

标准 Beta 回归模型的参数估计值，模型中所有项都标准化为均值等于 0 且标准差等于 1。仅当您在报表中右击并选择“列”>“标准 Beta”时才显示该列。

VIF 模型中每个项的方差膨胀因子。高 VIF 指示模型中的项存在共线性问题。

第 *i* 个项 x_i 的 VIF 定义如下：

$$VIF_i = \frac{1}{1 - R_i^2}$$

其中，R_i^2 是作为其他解释变量函数的 x_i 的回归的 R 方或**多重判定系数**。仅当您在报表中右击并选择“列”>“VIF”时才显示该列。

设计标准误差 参数估计值的**相对方差**的平方根 (Goos and Jones 2011, p. 25)：

$$\sqrt{diag(X'X)^{-1}}$$

这些值是标准误差除以 RMSE 的结果。仅当您在报表中右击并选择“列”>“设计标准误差”时才显示该列。

效应检验

仅当模型中包含固定效应时，“效应检验”报表才显示。给定效应的效应检验用来检验以下原假设：与该效应关联的所有参数均为零。对于单个连续解释变量，一个效应可能只有一个参数。在这种情况下，该检验等价于“参数估计值”报表中该项的 *t* 检验。根据名义型或有序型效应的水平数，此类效应可有若干关联参数。此类效应的效应检验用于检验所有关联参数是否均为零。

请注意以下事项：

- 对于项涉及线性相依性的效应，会尽量执行效应检验。请参见第 171 页的“模型项之间具有线性相依性的模型”。
- 参数化和奇异性处理不同于 SAS GLM 过程。有关参数化和奇异性处理的详细信息，请参见“统计详细信息”附录中第 456 页的“因子模型”。

“效应检验”报表包含以下列：

源 模型中的效应。

参数数目 与效应关联的参数数目。连续效应有一个参数。名义型或有序型效应的参数个数为其水平数减 1。一个交叉效应的参数个数是每个单独效应的参数个数的乘积。

自由度 效应检验的自由度。通常，“参数数目”和“自由度”是相同的。若预测变量间存在线性相依性，这两个值可能不同。在这种情况下，“自由度”可能小于“参数数目”，指示至少有一个与效应关联的参数无法检验。只要“自由度”小于“参数数目”，注释“损失自由度”即显示在报表中该行的右侧。若存在误差自由度，则会执行该检验。请参见第 173 页的“‘效应检验’报表”。

平方和 效应为零假设下的平方和。

F 比 用于检验效应为零的 F 统计量。F 比是效应均方与误差均方之比。效应均方是效应平方和除以效应自由度的结果。

概率 > F 效应检验的 *p* 值。

均方 效应均方，它是效应平方和除以效应自由度的结果。

注意：仅当您在报表中右击并选择**列 > 均方**时才显示。

效应详细信息

“效应详细信息”报表提供各个效应的详细信息、图和检验。该报表根据您在“拟合模型”启动窗口中选择的重点由不同的报表构成。

- “效应杠杆率”重点：每个效应自身的报表都显示在“拟合最小二乘法”报表窗口的顶部，“整体模型”报表的右侧。在这种情况下，该报表包括效应的“杠杆图”。
- “效应筛选”或“最小报表”重点：提供“效应详细信息”报表，但该报表最初处于关闭状态。点击展开图标可显示该报表。

报表初始内容是“最小二乘均值表”。有时该表并不适用，取决于该效应的性质。默认报表可能最初不显示任何内容。不过，某些红色小三角选项可用。

效应选项表

效应名称旁边的红色小三角菜单提供以下选项。对于特定建模类型，其中某些选项可能并不适用，所以不可用。

最小二乘均值表 显示检验效应时所比较的统计量。请参见第 87 页的 “最小二乘均值表”。

对于连续效应，未启用该选项。

最小二乘均值图 显示名义型和有序型效应的最小二乘均值图。若效应是交互作用，该选项显示 “最小二乘均值图选项”窗口。请参见第 89 页的 “最小二乘均值图”。

最小二乘均值对比 显示 “对比设置”窗口，该窗口支持您指定和检验对比，用于比较名义型和有序型效应及其交互作用的各个水平。请参见第 91 页的 “最小二乘均值对比”。

最小二乘均值 Student t 为使用 Student *t* 检验执行的最小二乘均值配对比较显示检验和置信区间。请参见第 94 页的 ““最小二乘均值 Student t”和 “最小二乘均值 Tukey HSD””。

注意：显著性水平应用于单个比较，而**不**共同应用于集体比较。集体比较的误差率大于单个检验的误差率。

最小二乘均值 Tukey HSD 为使用 Tukey-Kramer HSD （真实显著性差异）检验 (Tukey 1953; Kramer 1956) 执行的最小二乘均值配对比较显示检验和置信区间。请参见第 94 页的 ““最小二乘均值 Student t”和 “最小二乘均值 Tukey HSD””。

注意：显著性水平应用于集体配对比较。若样本大小相同，则显著性水平是精确的；若样本大小不同，则显著性水平是保守的 (Hayter 1984)。

最小二乘均值 Dunnett 针对您指定的控制水平显示配对比较的检验和置信区间。还提供检验结果图。请参见第 97 页的 “最小二乘均值 Dunnett”。

检验切片 对于交互作用中每列的每个水平，联合检验该交互作用中其他分类列的所有水平之间的配对比较。请参见第 98 页的 “检验切片”。

注意：仅可用于涉及名义型和有序型效应的交互作用。

功效分析 显示 “功效详细信息”报表，该报表支持您分析效应检验的功效。请参见第 99 页的 “功效分析”。

最小二乘均值表

最小二乘均值是在其他模型因子均设置为**中性**值的情况下，模型针对某个分类效应的水平预测的值。连续效应的中性值定义为其样本均值。关注的效应中未涉及的名义型效应的中性值是该效应的系数平均值。未涉及的有序型效应的中性值定义为值排序中的第一个效应水平。

最小二乘均值亦称**调整均值**或**总体边缘均值**。若模型中还有其他效应，最小二乘均值可能与简单均值不同。事实上，常常可以看到各最小二乘均值比起各样本均值来说，靠得更近。这种情形是由于做出预测处的中性值的性质导致的。

由于最小二乘均值是其他模型因子的特定值处的预测，所以您可以比较它们。检验效应时，将使用最小二乘均值进行比较。有关最小二乘均值的详细信息，请参见“统计详细信息”附录中第 460 页的“名义型因子的最小二乘均值”和第 469 页的“有序型最小二乘均值”。

对于主效应，“最小二乘均值表”还包括样本均值（图 3.10）。

“最小二乘均值表”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**体重**并点击 Y。
4. 选择**年龄**、**性别**和**身高**并点击**添加**。
5. 从**重点**列表中，选择**效应筛选**。
6. 点击**运行**。
7. “效应详细信息”报表显示在“拟合最小二乘法”报表的底部附近，最初处于关闭状态。点击“效应详细信息”报表标题旁边的展开图标可显示该报表。

“效应详细信息”报表（如图 3.10 所示）为三个效应中的每一个效应都显示了报表。为**年龄**和**性别**提供了最小二乘均值表，但没有为连续效应**身高**提供最小二乘均值表。请注意最小二乘均值与样本均值之间的差异。

图 3.10 最小二乘均值表

响应“体重”

效应详细信息

年龄

“最小二乘均值”表

水平	最小二乘均值	标准误差	均值
12	119.38671	5.5302002	99.000
13	105.70112	5.2695143	94.714
14	93.53945	3.9461891	100.833
15	99.61686	5.1729356	108.286
16	109.22769	7.7788438	118.333
17	121.90696	8.0768318	140.667

性别

“最小二乘均值”表

水平	最小二乘均值	标准误差	均值
F	121.43592	6.1844278	100.944
M	117.33750	5.7895918	108.318

身高

“最小二乘均值”报表包含以下列：

水平 分类水平或水平组合。

最小二乘均值 每个水平的最小二乘均值的估计值。

可估计性 （仅当最小二乘均值无法估计时显示。）若最小二乘均值无法估计会显示一个警告。

标准误差 显示每个水平的最小二乘均值的标准误差。

95% 下限 显示最小二乘均值的 95% 置信下限。仅当已选定“回归报表”>“显示所有置信区间”选项或是在报表中右击并选择“列”>“95% 下限”时，该列才显示。

95% 上限 显示最小二乘均值的 95% 置信上限。仅当已选定“回归报表”>“显示所有置信区间”选项或是在报表中右击并选择 “列”>“95% 上限”时，该列才显示。

均值 显示给定水平的响应样本均值。若模型中其他效应的值在该效应上不平衡，该均值则不同于最小二乘均值。

最小二乘均值图

“最小二乘均值图”选项为名义型和有序型主效应及其交互作用生成最小二乘均值图。若效应是交互作用，该选项显示“最小二乘均值图选项”窗口。请参见第 89 页的“最小二乘均值图选项”。

“最小二乘均值图”红色小三角菜单包含以下选项：

显示置信限 显示或隐藏图中每个估计值的置信限。

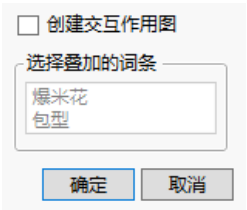
显示连接的点 显示或隐藏连接图中每个水平的最小二乘均值的一条或多条线。

删除 删除指定效应的 “最小二乘均值图”报表。

最小二乘均值图选项

当您从某个交互作用效应的红色小三角菜单选择 “最小二乘均值图”选项时，将显示 “最小二乘均值图选项”窗口。

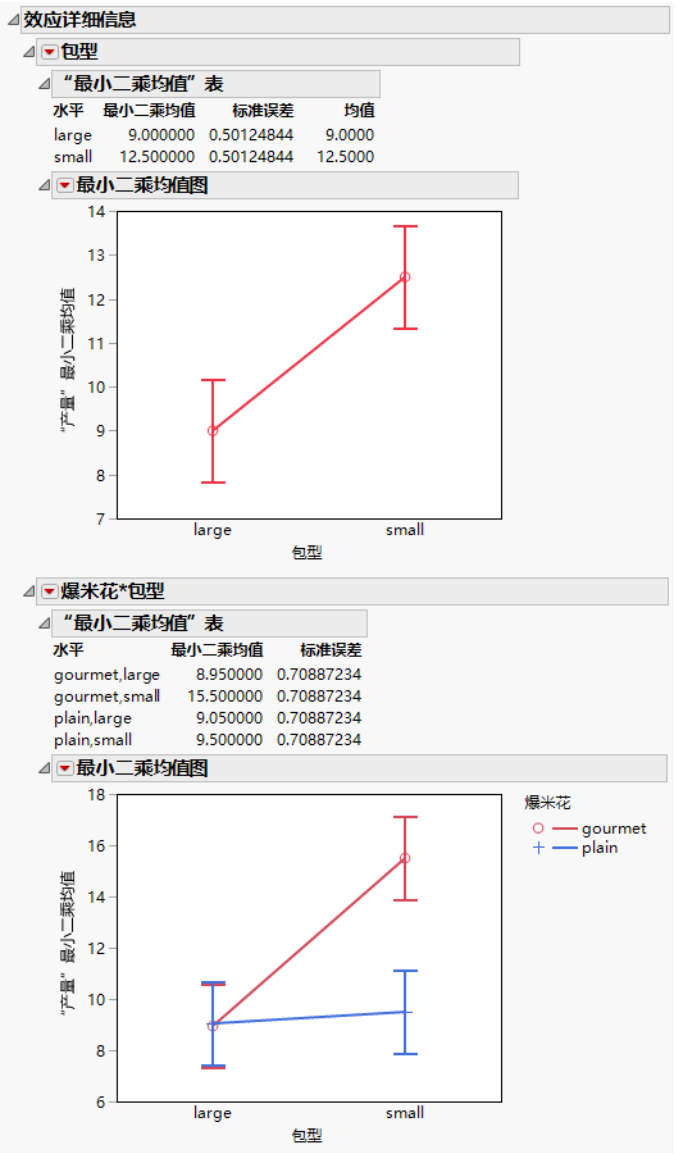
图 3.11 “最小二乘均值图选项”窗口



若您在窗口中未选择任何内容就点击“确定”，将显示一个最小二乘均值图。该图的水平轴由嵌套的因子水平组成，以获取每个组合的单独效应。要创建一个交互作用图，请选中“创建交互作用图”旁边的框。使用“选择叠加的词条”选项，您可以选择将哪个效应显示为交互作用图中的叠加变量。

对于三因子交互作用项，在您为交互作用图选择一个叠加变量后将显示选项的第二个面板。若您在第二个面板中未选择任何内容就点击“确定”，将显示一个交互作用图。或者，使用选项的第二个面板来为您在“为单独的图选择词条”下选择的效应的每个水平创建单独的交互作用图。

图 3.12 两个效应的最小二乘均值表和图



最小二乘均值图的示例

要创建图 3.12 中的报表，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Popcorn.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**产量**并点击 **Y**。

4. 选择**爆米花**、**油量**和**包型**，然后点击**宏** > **完全析因**。请注意，“重点”改为“效应筛选”。
5. 点击**运行**。
6. 点击“效应详细信息”展开图标以显示这七个模型效应的详细信息。
7. 点击“包型”红色小三角并选择**最小二乘均值图**。
8. 点击“爆米花 * 包型”红色小三角并选择**最小二乘均值图**。将显示“最小二乘均值图选项”窗口。
9. 在“最小二乘均值图选项”窗口中，点击“创建交互作用图”旁边的框。
10. 在“选择叠加的词条”下，选择**爆米花**。
11. 点击**确定**。
12. 要为**爆米花 * 包型**转置图中的因子，请重复第 8 步和第 9 步。
13. 在“选择叠加的词条”下，选择**包型**并点击**确定**。

图 3.13 转置了因子的交互作用的最小二乘均值图

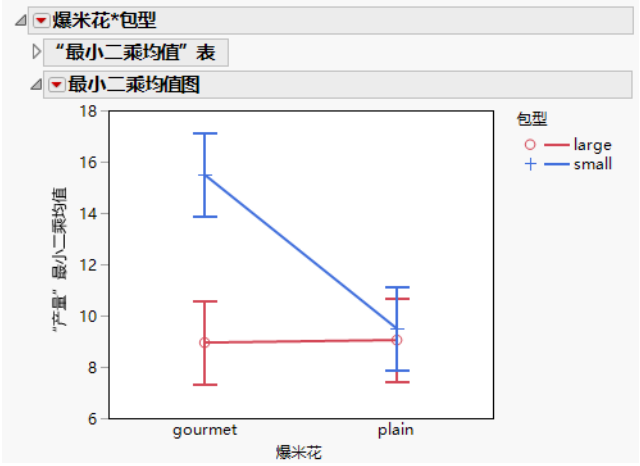


图 3.13 显示了已转置因子的**爆米花*包型**交互作用图。将其与图 3.12 中的图进行对比。这两个图描绘了相同的信息。一个图可能比另一个图更为直观，这取决于您的关注点。

最小二乘均值对比

对比是参数值的线性组合。在“对比设置”窗口中，您可以指定多个对比并联合检验它们是否为零（图 3.14）。

JMP 按照效应的最小二乘均值来构建对比。对比的每一列进行标准化使总和为 0，以便绝对值的总和等于 2。若对比涉及协变量，您可以指定要针对协变量的哪一个值来检验对比。

“对比设置”框显示了效应名称以及效应中各水平的名称。对比值最初设置为 0 并显示在包含 + 和 - 这两个符号的单元格旁边。点击这些按钮可比较水平。

每次您点击 + 或 - 按钮时，对比系数都进行标准化，尽可能使其总和为 0 且其绝对值总和等于 2。要比较其他水平，请点击**新建列**按钮。随即显示一个新列，您可以在其中定义新对比。完成操作后，点击**完成**。“对比”报表随即显示（图 3.15）。总体检验是一个针对所有对比的联合 *F* 检验。

注意：若尝试指定超过最大可能对比数的数值，检验将自动求值。

“对比”报表提供以下关于联合 *F* 检验的详细信息：

平方和 联合检验的平方和。

分子自由度 分子自由度。

分母自由度 分母自由度。

F 比 平方和除以分子自由度与均方误差之比。

概率 > F 显著性检验的 *p* 值。

“检验详细信息”报表

“检验详细信息”报表（图 3.15）为您检验的每个对比都显示一列。对于每个对比，该报表提供其估计值、其标准误差、该单一对比检验的 *t* 比，相应的 *p* 值、其平方和，以及对比估计值的置信区间。

“参数函数”报表

“参数函数”报表（图 3.15）显示您指定的对比，表示为模型项的线性组合。

最小二乘均值对比的示例

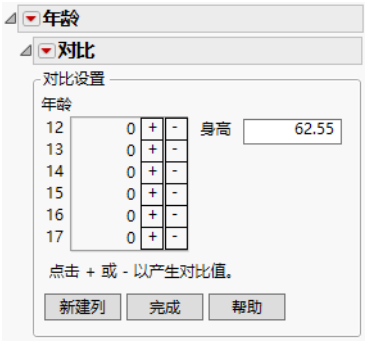
要演示“最小二乘均值对比”选项，请创建一个对比，用来比较前两个年龄水平与随后的两个水平。

请按照以下步骤创建图 3.14 所示的报表。

1. 选择**帮助 > 样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**体重**并点击 Y。
4. 选择**年龄**、**性别**和**身高**，然后点击**添加**。
5. 选择“选择列”列表中的**年龄**，然后选择“构造模型效应”列表中的**身高**，然后点击**交叉**。
6. 点击**运行**。

   “拟合最小二乘法”报表随即显示。
7. 点击“年龄”红色小三角并选择**最小二乘均值对比**。

图 3.14 “年龄”的最小二乘均值对比设置



8. 对年龄 12 和 13 点击 “+”。
9. 对年龄 14 和 15 点击 “-”。

   该对比根据身高 62.55 处的预测值，检验这两个年龄组的体重均值是否存在差异。
10. 请注意，连续效应**身高**旁边有一个文本框。默认值是连续效应的均值。
11. 点击**完成**。
12. 打开 “检验详细信息”和 “参数函数”报表。

“对比”报表显示在图 3.15 中。对比检验在 0.05 水平下显著。您可以得出结论：**身高**均值为 62.55 时，**年龄**为 12 岁和 13 岁的少年的**体重**预测值在统计上与**年龄**为 14 岁和 15 岁的少年的**体重**预测值存在差异。

图 3.15 最小二乘均值 “对比”报表

年龄

对比

检验详细信息

12	0.5
13	0.5
14	-0.5
15	-0.5
16	0
17	0
估计值	15.585
标准误差	6.5334
t 比	2.3854
概率>|t|	0.0243
平方和	1059.4
95% 下限	2.1792
95% 上限	28.99

平方和	分子自由度	分母自由度	F 比	概率>F
1059	1	27	5.6900	0.0243*

参数函数

参数	
截距	0
年龄[13-12]	-0.5
年龄[14-13]	-1
年龄[15-14]	-0.5
年龄[16-15]	0
年龄[17-16]	0
性别[F]	0
身高	0
(身高-62.55)*年龄[13-12]	0
(身高-62.55)*年龄[14-13]	0
(身高-62.55)*年龄[15-14]	0
(身高-62.55)*年龄[16-15]	0
(身高-62.55)*年龄[17-16]	0

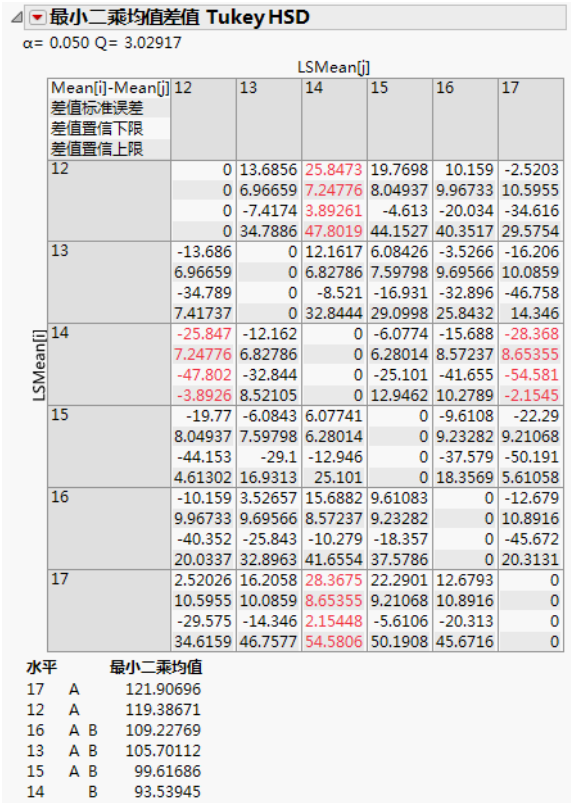
“最小二乘均值 Student t”和“最小二乘均值 Tukey HSD”

最小二乘均值 Student t 和最小二乘均值 Tukey HSD （**真实显著性差异**）选项检验模型效应的配对比较。

- “最小二乘均值 Student t”选项是基于通常的独立样本的等方差 *t* 检验。每个比较基于指定的显著性水平。执行多重比较得到的总误差率超过这个指定的显著性水平。
- “最小二乘均值 Tukey HSD”选项执行 Tukey HSD 检验。对于这些比较，显著性水平应用于整体配对比较。出于上述原因，最小二乘均值 Tukey HSD 的置信区间比最小二乘均值 Student t 的置信区间要宽。若样本大小相同，则显著性水平是精确的；若样本大小不同，则显著性水平是保守的 (Hayter 1984)。

图 3.16 显示了 Big Class.jmp 样本数据表中**年龄**效应的“最小二乘均值 Tukey”报表。（要获得该报表，请运行**拟合模型**数据表脚本，点击“年龄”红色小三角并选择**最小二乘均值** Tukey HSD。）默认情况下，该报表显示“交叉表报表”和“连接字母报表”。

图 3.16 “最小二乘均值 Tukey HSD”报表



最小二乘均值差值 Tukey HSD

α= 0.050 Q= 3.02917

LSMean[i]	Mean[i]-Mean[j] 差值标准误差 差值置信下限 差值置信上限	LSMean[j] 12	13	14	15	16	17
12		0	13.6856	25.8473	19.7698	10.159	-2.5203
		0	6.96659	7.24776	8.04937	9.96733	10.5955
		0	-7.4174	3.89261	-4.613	-20.034	-34.616
		0	34.7886	47.8019	44.1527	40.3517	29.5754
13		-13.686	0	12.1617	6.08426	-3.5266	-16.206
		6.96659	0	6.82786	7.59798	9.69566	10.0859
		-34.789	0	-8.521	-16.931	-32.896	-46.758
		7.41737	0	32.8444	29.0998	25.8432	14.346
14		-25.847	-12.162	0	-6.0774	-15.688	-28.368
		7.24776	6.82786	0	6.28014	8.57237	8.65355
		-47.802	-32.844	0	-25.101	-41.655	-54.581
		-3.8926	8.52105	0	12.9462	10.2789	-2.1545
15		-19.77	-6.0843	6.07741	0	-9.6108	-22.29
		8.04937	7.59798	6.28014	0	9.23282	9.21068
		-44.153	-29.1	-12.946	0	-37.579	-50.191
		4.61302	16.9313	25.101	0	18.3569	5.61058
16		-10.159	3.52657	15.6882	9.61083	0	-12.679
		9.96733	9.69566	8.57237	9.23282	0	10.8916
		-40.352	-25.843	-10.279	-18.357	0	-45.672
		20.0337	32.8963	41.6554	37.5786	0	20.3131
17		2.52026	16.2058	28.3675	22.2901	12.6793	0
		10.5955	10.0859	8.65355	9.21068	10.8916	0
		-29.575	-14.346	2.15448	-5.6106	-20.313	0
		34.6159	46.7577	54.5806	50.1908	45.6716	0

水平		最小二乘均值
17	A	121.90696
12	A	119.38671
16	A B	109.22769
13	A B	105.70112
15	A B	99.61686
14	B	93.53945

交叉表报表

两个选项都显示一个称为“交叉表报表”的矩阵，其中每个单元格都包含均值差值、差值标准误差，以及置信下限和上限。显著性水平和相应的临界值列在该矩阵上方。默认显著性水平为 0.05，但是您可以在“拟合模型”启动窗口中指定不同的显著性水平。在统计上存在差异的各对均值所对应的单元格显示为红色。

连接字母报表

“连接字母报表”默认显示在“交叉表”矩阵下。共享相同字母或由相同字母连接的水平在统计上无差异。不是由公共字母连接的水平在统计上有差异。

在图 3.16 中，水平 17、12、16、13 和 15 由字母 A 连接。该连接指示这些水平在 0.05 显著性水平下无差异。同样，水平 16、13、15 和 14 由字母 B 连接，指示它们在统计上无差异。但是，年龄 17 和 14 以及年龄 12 和 14 没有由公共字母连接，这指示这两对水平在统计上有差异。

提示：在连接字母报表中右击，然后选择“列”来添加包含连接字母（字母）、标准误差（标准误差）和置信区间限值（X% 下限和 X% 上限）的列。在“字母”列中，连接字母拼接到一列中。显著性和置信水平通过您在“拟合模型”启动窗口中使用“设置 Alpha”选项指定的显著性水平来确定。

“最小二乘均值 Student t”和“最小二乘均值 Tukey HSD”选项

每个报表窗口中显示的红色小三角选项用来显示或隐藏可选报表。以下所有选项都可用于 “最小二乘均值 Student t”。前四个选项可用于“最小二乘均值 Tukey HSD”。不管是“最小二乘均值 Student t”还是 “最小二乘均值 Tukey HSD”，默认均显示 “交叉表报表”和 “连接字母报表”。

交叉表报表 显示一个双向表，该表为每对水平提供均值差值、差值标准误差和差值置信限。包含显著性差异的单元格内容突出显示为红色。

连接字母报表 使用连接字母演示显著和不显著的比较。没有由相同字母连接的水平存在显著差异。由相同字母连接的水平没有显著差异。

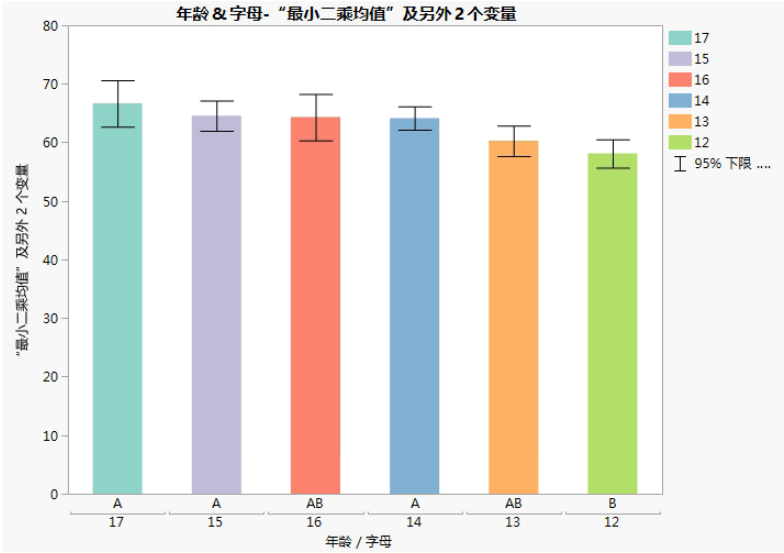
保存连接字母表 创建一个数据表，其中的列提供效应水平、连接字母、最小二乘均值、其标准误差和置信区间。该表包含名为“条形图”的脚本，该脚本可生成叠加了置信区间的最小二乘均值的着色条形图。各水平按照最小二乘均值的降序排列。

图 3.17 显示了基于 Big Class.jmp 的示例的条形图。运行**拟合模型**数据表脚本，点击 “年龄”红色小三角并选择**最小二乘均值** Tukey HSD。从 “最小二乘均值差值 Tukey HSD”报表中选择**保存连接字母表**。在显示的数据表中运行**条形图**脚本。

差值排序报表 按从大到小的顺序对差值排名，提供标准误差、置信限和 p 值。同时在条形图上标绘差值，且置信区间叠加在一起。

详细比较 为每个比较提供单个详细报表。对于给定比较，该报表显示估计差值、标准误差、置信区间、 t 比、自由度以及单侧和双侧检验的 p 值。同时还显示 t 分布图，该图演示比较的显著性检验。着色部分的区域是双侧检验的 p 值。

图 3.17 最小二乘均值差值 HSD 连接字母表中的条形图



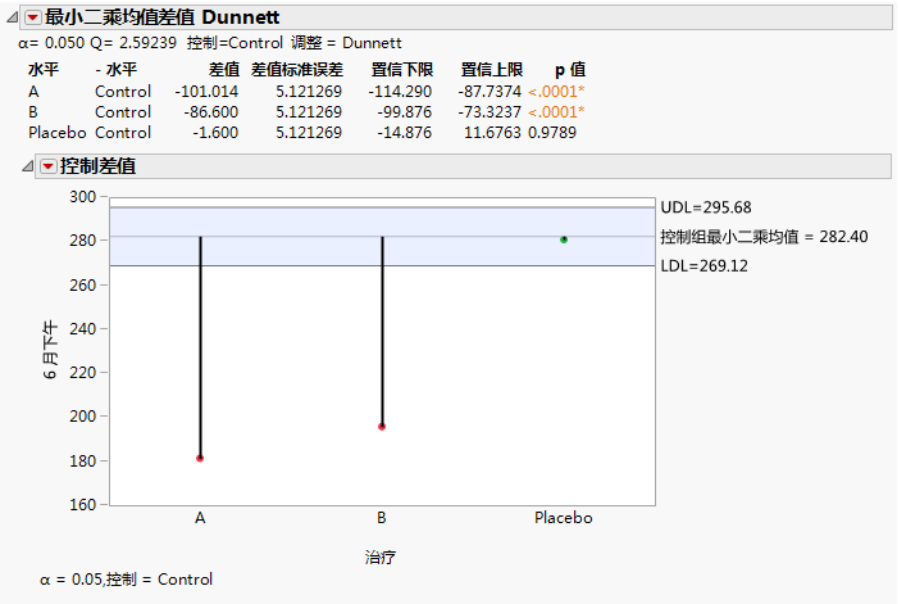
最小二乘均值 Dunnett

Dunnett 检验 (Dunnett 1955) 将一组均值与控制组的均值进行比较。误差率应用于配对比较集合。“最小二乘均值 Dunnett”选项针对给定效应的水平执行 Dunnett 检验。使用 Hsu 因子分析近似来计算 *p* 值和置信区间 (Hsu 1992)。

选择“最小二乘均值 Dunnett”后，系统会提示您为效应输入控制水平。“最小二乘均值差值 Hsu-Dunnett”报表显示显著性水平、检验统计量 (Q) 的值和控制水平。

图 3.18 显示了 Cholesterol.jmp 样本数据表中**治疗**效应的“最小二乘均值 Dunnett”选项所对应的报表。在此，响应为 **6 月下午**，**治疗**的“Control”水平被指定为控制水平。

图 3.18 “最小二乘均值 Dunnett”报表



该报表包含两个选项：

控制差值报表 默认显示“控制差值”报表。对于效应的每个水平，表中显示以下信息：与控制水平相比较的水平，估计差值、差值标准误差、置信区间和比较的 *p* 值。

控制差值图 对于控制水平之外的每个水平，都有一个点来显示该水平的最小二乘均值与控制水平的最小二乘均值之间的差值。图中标绘了上决策限和下决策限 (UDL, LDL)。报表包含“显示汇总报表”选项和“显示”选项。“显示汇总报表”选项提供图详细信息。“显示”选项支持您修改图外观。

检验切片

针对由名义型或有序型列构成的交互作用效应启用“检验切片”选项。对于交互作用中每个名义型或有序型列的每个水平，该选项都生成一个报表，用于对涉及该水平的设置的所有配对比较执行联合检验。该检验实际上是对交互作用的指定“切片”内的差值执行的检验。

假定您关注的是 A*B*C 交互作用，其中 A 的一个水平为“Small”。切片 A = Small 的“检验切片”报表将对 A = Small 时 B*C 诸水平的所有配对比较进行联合检验。它支持您检验交互作用内各水平的差值。

“检验切片”报表采用与“最小二乘均值对比”报表相同的格式。请参见第 91 页的“最小二乘均值对比”。

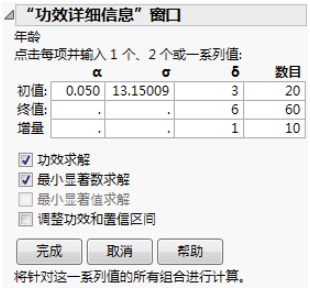
功效分析

打开“功效详细信息”窗口，您可以在该窗口中输入信息以获取特定效应的 *F* 检验的回顾或前瞻详细信息。

注意：要确保您的研究所包含的观测数足以检测所需的差值，请在**设计**实验时使用关于功效的信息。这种分析称为前瞻功效分析。请考虑使用“实验设计”平台来设计您的研究。“实验设计”>“样本大小与功效”和“实验设计”>“评估设计”都有助于前瞻功效分析。有关使用标准最小二乘法的前瞻功效分析的示例，请参见第 185 页的“前瞻功效分析”。

图 3.19 显示了 Big Class.jmp 样本数据表的“功效详细信息”窗口的示例。使用“功效详细信息”窗口，您可以探索 alpha (α)、sigma (σ)、delta (δ) 和“数目”（研究规模）值的功效。输入单个值（仅“初值”）、两个值（“初值”和“终值”），或是对一系列值输入起始值（初值）、终止值（终值）以及增加值（增量）。将为您指定的值的所有可能组合报告功效计算。

图 3.19 “功效详细信息”窗口



请参见第 180 页的“功效分析”。

“功效详细信息”窗口报表包含以下列和选项：

Alpha (α) 检验的显著性水平。该值介于 0 和 1 之间，通常为 0.05、0.01 或 0.10。Alpha 初始值（显示在第一行中）为 0.05，除非您在“拟合模型”启动窗口中选定“设置 Alpha 水平”并设置了不同的值。

Sigma (σ) 模型中剩余误差的估计值。显示在第一行中用以提供指导的初始值为 RMSE（均方误差的平方根）。

Delta (δ) 关注的效应大小。请参见第 180 页的“效应大小”。显示在第一行中的初始值是假设的平方和除以研究中的观测数之后的平方根（即 $\delta = \sqrt{SS/n}$）。

数目 (n) 样本大小。显示在第一行中的初始值是当前研究中的观测数。

功效求解 对作为 α、σ、δ 和 *n* 的函数的功效求解。功效是检测大小为 δ 的差值的概率，是在指定的 σ 和 *n* 的前提下看到在水平 α 下显著的检验结果的概率。请参见“统计详细信息”附录中第 479 页的“功效的计算”。

最小显著数求解 针对指定的 δ 和 σ，求解在水平 α 下获得显著的检验结果所需的最小观测数。请参见“统计详细信息”附录中第 478 页的“LSN 的计算”。

最小显著值求解 求解生成 p 值为 α 的参数或参数的线性函数的最小正值。最小显著值是 α、σ 和 n 的函数。该选项仅可用于自由度为 1 的检验。请参见“统计详细信息”附录中第 478 页的 “LSV 的计算”。

调整功效和置信区间 回顾功效计算使用标准误差估计值，以及估计 F 分布的非中心参数时使用的检验参数的估计值。**调整功效**是基于非中心参数的估计值（已去除正偏倚）的回顾功效计算 (Wright and O'Brien 1988)。

调整功效的置信区间基于非中心估计值的置信区间。

调整功效处理样本估计值，所以仅针对当前研究中估计的 δ 计算它及其置信限。请参见“统计详细信息”附录中第 480 页的 “调整功效的计算”。

失拟

“失拟”报表可为评估模型是否良好拟合数据的检验提供详细信息。仅当可以执行该检验时才会显示“失拟”报表。该检验依赖于使用独立于模型的估计值估计响应方差的能力。构造该估计值要求模型效应的重复值处有响应值。该检验涉及**纯误差**估计值的计算，它基于利用这些重复观测得到的平方和。

在以下情况下，由于无法计算检验统计量，“失拟”报表不会显示：

- X 变量没有重复点，所以无法计算纯误差平方和。
- 模型**饱和**，这表示估计参数的个数与观测数相同。这种模型完全拟合，所以不可能评估失拟。

模型中的误差平方和与纯误差平方和之间的差值称为**失拟**平方和。若模型不合适，失拟变异会显著大于纯误差变异。例如，您的预测变量的函数形式可能有误，或是您模型中的交互作用效应可能不足或不正确。

“失拟”报表包含以下列：

源 变异的三个来源：失拟、纯误差和总误差。

自由度 每个变异来源的自由度 (DF)：

- 总误差的自由度与“方差分析”表的“误差”行中的自由度值相同。根据平方和分解，总误差自由度可分为失拟自由度和纯误差自由度。
- 纯误差自由度是每个重复观测组的合并结果。通常情况下，若有 g 个组，而且每组中每个效应具有相同的设置，则纯误差自由度（表示为 $\mathrm{DF}_{\mathrm{PE}}$）定义如下：

$$\mathrm{DF}_{PE} = \sum_{i=1}^{g} (n_i - 1)$$

其中，n_i 是第 i 个组中的重复值数目。

- 失拟自由度是总误差自由度与纯误差自由度之间的差值。

平方和 每个误差来源关联的平方和 (SS)：

- 总误差平方和是相应的“方差分析”表的误差行中的平方和。

– 纯误差平方和是每个重复的观测组的平方和的总和。用纯误差平方和除以其自由度可估计给定的预测变量设置下的响应方差。该估计值不受模型影响。通常情况下，若有 g 个组，而且每组中每个效应具有相同的设置，则纯误差平方和（表示为 SS_{PE}）定义如下：

$$SS_{PE} = \sum_{i=1}^{g} SS_i$$

其中，SS_i 是每个观测响应与第 i 个组的响应均值之间差值的平方和。

– 失拟平方和是总误差平方和与纯误差平方和之间的差值。

均方 源的均方，它是平方和除以自由度的结果。与纯误差均方相比，较大的失拟均方表明模型拟合不佳。*F* 比提供正式检验。

F 比 失拟均方与纯误差均方之比。*F* 比检验的是以下假设：失拟估计的方差与纯误差均方相等，这可以解释为表示“无失拟”。

概率 > F 失拟检验的 *p* 值。较小的 *p* 值指示失拟显著。

最大 R 方 仅基于这些效应的模型可实现的最大 R 方。纯误差平方和不随模型形式变化而变化。所以包含这些重复效应的模型可以解释的最大变异量等于：

$$\frac{\text{SS（校正总和）} - \text{SS（纯误差）}}{\text{SS（校正总和）}} = 1 - \frac{\text{SS（纯误差）}}{\text{SS（校正总和）}}$$

该公式定义了最大 R 方。

估计值

“估计值”菜单提供有关模型参数的更多详细信息。为了更好地理解估计值，您可能需要查看 JMP 对名义型和有序型效应编码所用的方法。请参见第 176 页的““定制检验”示例的详细信息”、“统计详细信息”附录中第 456 页的“名义型因子”和“统计详细信息”附录中第 467 页的“有序型因子”。

若模型包含随机效应，则“估计值”菜单中仅提供以下适用的选项。

“估计值”菜单提供以下选项：

显示预测表达式 显示或隐藏包含估计模型的方程的“预测表达式”报表。相关示例，请参见第 103 页的“显示预测表达式”。

排序后的估计值 显示或隐藏“排序后的参数估计值”报表，该报表在筛选情形下很有用。若设计不饱和，该报表是“参数估计值”报表，其中除截距之外的项按照显著性的降序排列。若设计是饱和的，则提供伪 *t* 检验。请参见第 103 页的“排序后的估计值”。

扩展后的估计值 （仅当至少有一个效应非连续型时才可用。）显示或隐藏“扩展后的估计值”报表，该报表通过给出名义型效应的所有水平的参数估计值扩展了“参数估计值”报表。请参见第 106 页的“扩展后的估计值”。

指标参数化估计值 （仅当模型效应之间存在名义型列时可用。）显示或隐藏“指标函数参数化”报表，该报表包含使用经典指标函数将模型中的名义型效应参数化后的参数估计值。请参见第 108 页的“指标参数化估计值”。

序贯检验 显示或隐藏“序贯（1 型）检验”报表，该报表包含效应按顺序添加到模型后的平方和。基于序贯平方和执行 *F* 检验。请参见第 109 页的“序贯检验”。

定制检验 支持您检验定制假设。请参见第 109 页的“定制检验”。

多重比较 使您可以指定效应水平的比较。这些比较可涉及单个效应，您也可以定义灵活的定制比较。您可以与总均值、控制均值进行比较，也可以使用 Tukey HSD 或 Student *t* 获取所有配对比较。指定 Student *t* 方法时，还可以执行等价性检验来识别有实际意义的配对差值。请参见第 112 页的“多重比较”。

比较斜率 （仅当有一个名义型项、一个连续型项以及它们针对固定效应的交互作用效应时才可用。）生成一个报表，该报表支持您比较协方差分析 (ANCOVA) 模型中交互作用效应的每个水平的斜率。请参见第 124 页的“比较斜率”。

联合因子检验 （仅当模型包含交互作用时可用。）对于模型中的每个主效应，显示或隐藏对涉及该主效应的所有参数的联合检验。请参见第 124 页的“联合因子检验”。

逆预测 使您可以对一个或多个响应值预测解释变量的值。请参见第 125 页的“逆预测”。

Cox 混料 （仅当模型包含混料效应时可用。）为 Cox 混料模型生成参数估计值。使用这些估计值可导出因子效应，并相对于设计空间中的参考点估计响应曲面形状。请参见第 129 页的“Cox 混料”。

参数功效 向“参数估计值”报表添加列，这些列提供与相应的假设检验相关的功效和其他详细信息。请参见第 130 页的“参数功效”。

估计值的相关性 显示或隐藏模型中所有参数估计值的相关性矩阵。请参见第 131 页的“估计值的相关性”。

误差指定 （仅当不存在随机效应时才可用。）指定用于“拟合最小二乘法”报表中的标准误差和检验的误差方差和误差自由度。请注意，更改“误差指定”不会影响“学生化残差”图和“Box Cox 变换”报表。若“误差指定”是“纯误差”或“指定”，则“方差分析”报表中将显示一个附加列。请参见第 83 页的“方差分析”。

默认估计值 使用模型的标准均方根误差和误差自由度计算所有检验和标准误差。

纯误差 使用“失拟”报表中的纯误差均方和相关的自由度计算所有检验和标准误差。请参见第 100 页的“失拟”。

警告：若纯误差自由度为 1，则会显示一条警告消息，指示检验较弱且置信限较大。

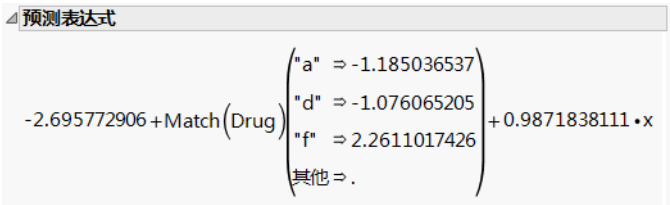
指定 使用用户指定的误差方差和误差自由度值计算所有检验和标准误差。

显示预测表达式

“显示预测表达式”选项显示用于预测响应的方程式。图 3.20 显示了 Drug.jmp 样本数据表的示例。该表达式作为典型 JMP 公式给出。例如，要预测服用药物 a 的某人在 x = 10 时的响应，您需要按如下方式计算（适当舍入）：–2.696 - 1.185 + 0.987(10) = 5.99。

提示：要在预测公式中指定位数，请转至**文件 > 首选项 > 表**，并更改**默认字段宽度**值。

图 3.20 预测表达式



要获取以上报表，请执行以下步骤：

预测表达式的示例

1. 选择**帮助 > 样本数据库**，然后打开 Drug.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择 y 并点击 Y。
4. 选择**药物**和 x，然后点击**添加**。
5. 点击**运行**。
6. 点击“响应‘Y’”红色小三角并选择**估计值 > 显示预测表达式**。随即显示图 3.20 中的报表。

排序后的估计值

“排序后的估计值”选项可生成在筛选情形下使用的“参数估计值”报表版本。若设计不饱和，“排序后的估计值”报表提供“参数估计值”报表所显示的信息，但其中除截距之外的所有项都按照显著性的降序排列（图 3.21 中的第二个报表）。若设计是饱和的，则提供伪 *t* 检验。这些检验基于 Lenth **伪标准误差** (Lenth 1989)。请参见第 105 页的“Lenth PSE”。

“排序后的估计值”报表的示例

1. 选择**帮助 > 样本数据库**，然后打开 Reactor.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择 Y 并点击 Y。
4. 确保 2 显示在靠近窗口底部附近的**次数**框中。
5. 选择 F、 Ct、 A、 T 和 Cn，然后点击**宏 > 析因次数**。

6. 点击**运行**。
7. 点击“响应 ‘Y’”红色小三角并选择**估计值 > 排序后的估计值**。

图 3.21 排序后的参数估计值

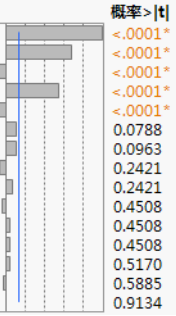
参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	65.5	0.565962	115.73	<.0001*
F	-0.6875	0.565962	-1.21	0.2421
Ct	9.75	0.565962	17.23	<.0001*
A	-0.3125	0.565962	-0.55	0.5885
T	5.375	0.565962	9.50	<.0001*
Cn	-3.125	0.565962	-5.52	<.0001*
F*Ct	0.6875	0.565962	1.21	0.2421
F*A	0.375	0.565962	0.66	0.5170
F*T	-0.4375	0.565962	-0.77	0.4508
F*Cn	0.0625	0.565962	0.11	0.9134
Ct*A	0.4375	0.565962	0.77	0.4508
Ct*T	6.625	0.565962	11.71	<.0001*
Ct*Cn	1	0.565962	1.77	0.0963
A*T	1.0625	0.565962	1.88	0.0788
A*Cn	0.4375	0.565962	0.77	0.4508
T*Cn	-5.5	0.565962	-9.72	<.0001*

效应检验

排序后的参数估计值

项	估计值	标准误差	t 比	概率>|t|
Ct	9.75	0.565962	17.23	<.0001*
Ct*T	6.625	0.565962	11.71	<.0001*
T*Cn	-5.5	0.565962	-9.72	<.0001*
T	5.375	0.565962	9.50	<.0001*
Cn	-3.125	0.565962	-5.52	<.0001*
A*T	1.0625	0.565962	1.88	0.0788
Ct*Cn	1	0.565962	1.77	0.0963
F	-0.6875	0.565962	-1.21	0.2421
F*Ct	0.6875	0.565962	1.21	0.2421
F*T	-0.4375	0.565962	-0.77	0.4508
Ct*A	0.4375	0.565962	0.77	0.4508
A*Cn	0.4375	0.565962	0.77	0.4508
F*A	0.375	0.565962	0.66	0.5170
A	-0.3125	0.565962	-0.55	0.5885
F*Cn	0.0625	0.565962	0.11	0.9134



若“重点”设置为“效应筛选”并且所有效应都只有一个参数，“排序后的参数估计值”报表也会自动显示。

请注意“参数估计值”报表与“排序后的参数估计值”报表之间的以下差异（两个报表均显示在图 3.21 中）：

- “排序后的参数估计值”报表不显示截距。
- 效应按照 *t* 比的绝对值排序，最显著的效应显示在顶部。
- 条形图显示 *t* 比，其中使用垂直线来显示 0.05 显著性水平的临界值。

饱和模型的“排序后的估计值”报表

筛选实验经常涉及完全饱和的模型，这种模型中没有足够的自由度来估计误差。在这些情况下，“排序后的估计值”报表（图 3.21）提供相对标准误差并使用 Lenth 伪标准误差 (PSE) 构造 *t* 比和 *p* 值。这些量的名称中标有**伪**。请参见第 105 页的“Lenth PSE”和第 105 页的“伪 t 比”。通过一条注释来解释更改并显示 PSE。

该报表包含以下列：

项 系数受关注的模型项。

估计值 参数估计值以排序后的顺序显示，最小的 *p* 值列在最前面。

相对标准误差 若剩余误差没有自由度，该报表将提供相对标准误差。通过将均方根误差设置为 1 来计算相对标准误差。

伪 t 比 估计值的 *t* 比，使用伪标准误差计算得到。Lenth PSE 的值显示在报表底部的注释中。

伪 p 值 使用误差自由度值 (DFE) 为 *m*/3 计算的 *p* 值，其中 *m* 是除截距之外的参数数目。DFE 的值显示在报表底部的注释中。

Lenth PSE

Lenth **伪标准误差** (PSE) 是 Lenth (1989) 提出的剩余误差的估计值。它基于效应稀疏原则：在筛选实验中，相对较少的效应是活跃的。不活跃效应表示随机噪声并构成 Lenth 估计值的基础。

按如下方式计算该值：

1. 考虑所有非截距参数的绝对值。
2. 删除绝对值超过绝对估计值中位数 3.75 倍的所有参数估计值。
3. 将剩余参数估计值的绝对值的中位数乘以 1.5。

伪 t 比

若相对标准误差相等， Lenth PSE 会显示在报表底部的注释中。伪 *t* 比计算如下：

$$\text{伪 } t \text{ 比} = \frac{\text{估计值}}{\text{PSE}}$$

若相对标准误差不等，则计算 TScale Lenth PSE。该值是估计值的 PSE 除以其相对标准误差的结果。伪 *t* 比计算如下：

$$\text{伪 } t \text{ 比} = \frac{\text{估计值}}{\text{TScale Lenth PSE} \times \text{相对标准误差}}$$

请注意，要估计给定估计值的标准误差，需要对 TScale Lenth PSE 进行调整，将其乘以估计值的相对标准误差。

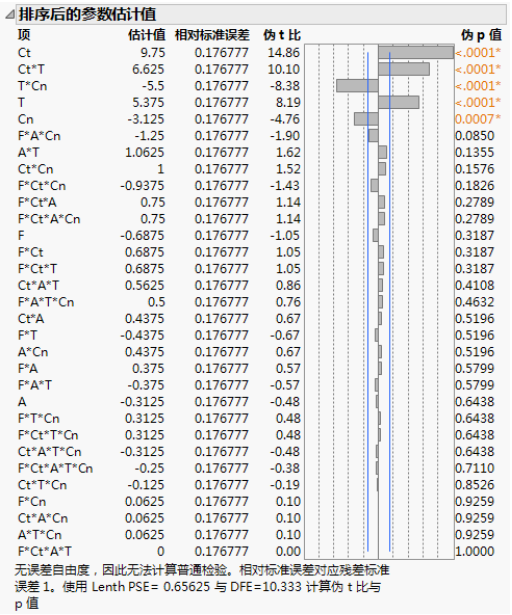
饱和模型示例

1. 选择**帮助 > 样本数据库**，然后打开 Reactor.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择 Y 并点击 Y。
4. 选择以下五列：F、 Ct、 A、 T 和 Cn。

5. 点击**宏**按钮并选择**完全析因**。
6. 点击**运行**。
7. 点击 “响应 ‘Y’” 红色小三角并选择**估计值 > 排序后的估计值**。

   请注意，使用的 Lenth PSE 和自由度在报表底部提供。该报表指示：基于效应的伪 p 值，效应 Ct、 Ct*T、 T*Cn、 T 和 Cn 高度显著。

图 3.22 饱和模型的 “排序后的参数估计值” 报表



排序后的参数估计值

项	估计值	相对标准误差	伪 t 比	伪 p 值
Ct	9.75	0.176777	14.86	<.0001*
Ct*T	6.625	0.176777	10.10	<.0001*
T*Cn	-5.5	0.176777	-8.38	<.0001*
T	5.375	0.176777	8.19	<.0001*
Cn	-3.125	0.176777	-4.76	0.0007*
F*A*Cn	-1.25	0.176777	-1.90	0.0850
A*T	1.0625	0.176777	1.62	0.1355
Ct*Cn	1	0.176777	1.52	0.1576
F*Ct*Cn	-0.9375	0.176777	-1.43	0.1826
F*Ct*A	0.75	0.176777	1.14	0.2789
F*Ct*A*Cn	0.75	0.176777	1.14	0.2789
F	-0.6875	0.176777	-1.05	0.3187
F*Ct	0.6875	0.176777	1.05	0.3187
F*Ct*T	0.6875	0.176777	1.05	0.3187
Ct*A*T	0.5625	0.176777	0.86	0.4108
F*A*T*Cn	0.5	0.176777	0.76	0.4632
Ct*A	0.4375	0.176777	0.67	0.5196
F*T	-0.4375	0.176777	-0.67	0.5196
A*Cn	0.4375	0.176777	0.67	0.5196
F*A	0.375	0.176777	0.57	0.5799
F*A*T	-0.375	0.176777	-0.57	0.5799
A	-0.3125	0.176777	-0.48	0.6438
F*T*Cn	0.3125	0.176777	0.48	0.6438
F*Ct*T*Cn	0.3125	0.176777	0.48	0.6438
Ct*A*T*Cn	-0.3125	0.176777	-0.48	0.6438
F*Ct*A*T*Cn	-0.25	0.176777	-0.38	0.7110
Ct*T*Cn	-0.125	0.176777	-0.19	0.8526
F*Cn	0.0625	0.176777	0.10	0.9259
Ct*A*Cn	0.0625	0.176777	0.10	0.9259
A*T*Cn	0.0625	0.176777	0.10	0.9259
F*Ct*A*T	0	0.176777	0.00	1.0000

无误差自由度，因此无法计算普通检验。相对标准误差对应残差标准误差 1。使用 Lenth PSE= 0.65625 与 DFE=10.333 计算伪 t 比与 p 值

扩展后的估计值

处理参数估计值时，您必须了解 JMP 如何对名义型和有序型列编码。有关如何对名义型列编码的详细信息，请参见第 176 页的 ““定制检验”示例的详细信息”。有关如何对有序型列编码和建模的详细信息，请参见 “统计详细信息” 附录中第 456 页的 “名义型因子” 和 “统计详细信息” 附录中第 467 页的 “有序型因子”。

若模型中有名义型项并且您想要查看完整的一组估计值的详细信息，此时可使用 “扩展后的估计值” 选项。“扩展后的估计值” 选项提供估计值、估计值的标准误差、*t* 比和 *p* 值。

“扩展后的估计值” 报表的示例

1. 选择**帮助 > 样本数据库**，然后打开 Drug.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择 y 并点击 Y。
4. 选择**药物**和 x，然后点击**添加**。

5. 点击**运行**。
6. 点击“响应 ‘Y’”红色小三角并选择**估计值 > 扩展后的估计值**。

“扩展后的估计值”报表随同“参数估计值”报表显示在图 3.23 中。请注意，项“药物[f]”的估计值显示在“扩展后的估计值”报表中。该检验的原假设为：药物 f 组的均值与总均值无差异。“药物[f]”的检验在 0.05 水平下显著，这表明药物 f 组的响应均值与总响应存在差异。请参见第 107 页的“‘扩展后的估计值’检验的解释”。

图 3.23 “参数估计值”与“扩展后的估计值”的比较

响应 “y”

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	-2.695773	1.911085	-1.41	0.1702
药物[a]	-1.185037	1.060822	-1.12	0.2742
药物[d]	-1.076065	1.041298	-1.03	0.3109
x	0.9871838	0.164498	6.00	<.0001*

扩展后的估计值

名义型因子 - 扩展至所有水平

项	估计值	标准误差	t 比	概率>|t|
截距	-2.695773	1.911085	-1.41	0.1702
药物[a]	-1.185037	1.060822	-1.12	0.2742
药物[d]	-1.076065	1.041298	-1.03	0.3109
药物[f]	2.2611017	1.093974	2.07	0.0488*
x	0.9871838	0.164498	6.00	<.0001*

“扩展后的估计值”检验的解释

假定模型由一个包含 *n* 个水平的名义型因子构成。该因子由 *n-1* 个指标变量表示，*n-1* 个水平中的每个水平都对应一个指标变量。与 *n-1* 个指标变量中的任意一个变量对应的参数估计值都是该水平的响应均值与所有水平的响应平均值之间的差值。这种表示形式是由于 JMP 对名义型变量编码的方式造成的（请参见第 176 页的“‘定制检验’示例的详细信息”）。参数估计值通常解释为该水平的**效应**。

例如，在 Cholesterol.jmp 样本数据表中，考虑单个因子**治疗**和响应 **6 月下午**。与治疗[A] 这个项（或指标变量）关联的参数估计值是**治疗** A 的 **6 月下午**的均值与 **6 月下午**的总均值之间的差值。

名义型变量**所有**水平的效应被限制为总和为零。考虑水平排序中的最后一个水平（也就是使用 –1s 编码的水平）的效应。该水平的效应是其他 *n-1* 个水平的效应之和的负数。由此判定：最后一个水平的效应是其他 *n-1* 个水平的参数估计值之和的负数。

“估计值”菜单中的“扩展后的估计值”选项计算缺失估计值、检验涉及名义型列的所有效应，并在文本报表中显示它们。您可以验证任意此类因子各水平的估计值的均值（或总和）为 0。特别是，该关系指示这些估计值及其关联检验彼此不独立。

在图 3.23 所示的 Drug.jmp 报表中，**药物**关联项的估计值基于包含协变量 **x** 的模型。

注意：

- “药物[a]”的估计值是**药物** a 的最小二乘均值与 y 的总均值之间的差值。
- “药物[f]”的估计值（在“扩展后的估计值”报表中提供）是“药物[a]”与“药物[d]”的估计值之和的负数。

- “扩展后的估计值”报表中提供的“药物 [f]”的 *t* 检验用来检验**药物** f 组的响应是否与响应总均值存在差异。
- 若高阶交互作用中涉及名义型因子，“扩展后的估计值”报表会很长。例如，二水平名义型因子的五因子交互作用仅生成一个参数估计值，但包含 $2^5 = 32$ 个扩展效应，所有效应都相同，直到符号改变。

指标参数化估计值

该选项显示“指标函数参数化”报表，该报表提供模型的参数估计值，该模型中的名义型列使用指标 (SAS GLM) 参数化而且被视为连续值。有序型列仍使用通常的 JMP 编码方案保持编码。“统计详细信息”附录中第 456 页的“因子模型”对 SAS GLM 和 JMP 编码方案进行了说明。

在 JMP 编码方案中，与名义型变量的某个水平的指标对应的估计值是该水平的响应均值与所有水平的响应平均值之间的差值。要查看 JMP 编码，请从“标准最小二乘法”红色小三角菜单中选择**保存列 > 保存编码表**。

在指标编码方案中，与名义型变量的某个水平的指标对应的估计值是该水平的响应均值与最后一个水平的响应平均值之间的差值。最后一个水平是具有最高值排序编码的水平；这个水平的指标函数未包含在模型中。

警告：“指标函数参数化”报表中提供的标准误差和 t 比不同于“参数估计值”报表中提供的标准误差和 t 比。这是因为这些估计值估计的是不同的参数。

要创建图 3.24 中的报表，请执行第 106 页的 “‘扩展后的估计值’报表的示例”中的步骤。不过，不要选择 “扩展后的估计值”，改为选择**指标参数化估计值**。

图 3.24 指标参数化估计值

响应 “y”

指标函数参数化

项	估计值	标准误差	分母自由度	t 比	概率>|t|
截距	-0.434671	2.471354	26.00	-0.18	0.8617
药物[a]	-3.446138	1.886781	26.00	-1.83	0.0793
药物[d]	-3.337167	1.853866	26.00	-1.80	0.0835
x	0.9871838	0.164498	26.00	6.00	<.0001*

在除“拟合模型”的“广义回归”特质之外的所有 JMP 环境中都会对名义型变量使用 JMP 编码方案。有关“广义回归”特质中名义型变量的编码方案的详细信息，请参见“广义回归模型”一章中第 253 页的“启动“广义回归”特质”。

请注意，使用 JMP 和 SAS GLM 参数化分别生成的模型可能存在差异。某些模型是等价的。其他模型（如：无截距模型、含有缺失单元格的模型、含有名义型或有序型效应的模型以及混料模型）可能显示差异。

序贯检验

“序贯检验”报表显示效应按顺序添加到模型后的平方和与检验。输入顺序由效应在“拟合模型”启动窗口的“构造模型效应”列表中的显示顺序来定义。图 3.25 中的报表针对 Drug.jmp 样本数据表。

图 3.25 “序贯检验”报表

响应 “y”

序贯 (1类) 检验

源	参数数目	自由度	序贯平方和	F 比	概率>F
药物	2	2	293.60000	9.1486	0.0010*
x	1	1	577.89740	36.0145	<.0001*

构成序贯检验基础的平方和亦称 **I 型平方和**。通过按照指定的效应输入顺序执行拟合模型的步骤来计算该平方和。考虑一个特定的效应。为包含在该效应**之前**输入的所有效应的模型计算模型平方和。然后为包含这些效应**和**指定效应的模型计算模型平方和。指定效应的序贯平方和是这两个模型平方和的增量。

请参考图 3.25，其中显示了 Drug.jmp 样本数据表的序贯平方和。在“拟合模型”启动窗口中，首先输入的是**药物**，然后输入的是 x。仅含**药物**的模型的模型平方和为 293.6。向模型添加 x 后，模型平方和变为 871.4974。增量 577.8974 是 x 的序贯平方和。

“序贯（1 型）检验”报表中显示的检验是基于序贯平方和的 *F* 检验，该检验亦称 **I 型检验**。“*F* 比”检验指定的效应，其中模型仅包含该效应以及在“源”列中列在该效应上方的效应。

序贯平方和的总和为模型平方和。另一个不错的功能就是：在一般模型假设下，值在统计上彼此独立。不过，这些值确实依赖于模型中项的顺序，因此在许多情况下不适用。

序贯检验在以下情况下适用：

- 以适当顺序指定的平衡方差分析模型（即：在效应列表中双因子交互作用跟在主效应之后，依此类推）
- 以适当顺序指定的纯嵌套模型
- 以适当顺序指定的多项式回归模型

“参数估计值”和“效应检验”报表中提供的检验基于 **III 型平方和**。在此，效应的平方和是在模型中输入所有其他效应之后，由该效应所解释的额外平方和。

定制检验

要检验涉及任意模型参数的一个或多个定制假设，请从“估计值”菜单中选择**定制检验**。在该窗口中，您可以指定模型参数的一个或多个线性函数或**对比**。

结果包括针对各个对比的单个检验以及针对所有对比的联合检验。请参见图 3.26。单个对比的报表提供参数的指定线性函数的估计值及其标准误差。同时还提供 *t* 比、其 *p* 值以及关联的平方和。在单个对比结果下方，针对所有对比的联合检验提供平方和、分子自由度、*F* 比及其 *p* 值。

警告：使用剩余误差执行这些检验。若您的模型中包含随机效应，并且您使用 EMS 而不是 REML，则这些检验可能不合适。

注意：若您要检验高阶效应中涉及的效应，请考虑使用最小二乘均值检验而不是定制检验。针对其他模型效应调整了最小二乘均值。您可以在“效应详细信息”下检验最小二乘均值对比。

“定制检验”报表的组成部分

“定制检验”规格窗口具有以下组成部分：

可编辑的文本框 “定制检验”标题栏下方的空间是用于输入检验名称的可编辑区域。

参数 列出模型项。在项列表右侧是对应参数对应为零值的列。在这些单元格中输入值，为检验指定线性函数。

“=”符号 “参数”列表中的最后一行标有“=”。在该单元格中输入常数，完成对每个对比的指定。

添加列 添加包含零值的列，以便您可以联合检验参数的若干线性函数。

完成 点击“完成”按钮可执行检验。报表随即更改，以便为每个检验列显示检验统计量值、标准误差和其他统计量。针对所有列的联合 *F* 检验在报表底部的框中提供。

“定制检验”报表选项

“定制检验”红色小三角菜单包含以下选项：

功效分析 提供联合检验的功效分析。仅当执行检验之后，该选项才可用。请参见第 130 页的“参数功效”。

删除 删除“定制检验”报表。

注意：重复选择**估计值** > **定制检验**，执行若干联合定制检验。

图 3.26 使用 Cholesterol.jmp 样本数据表显示了包含三个对比的规格窗口的示例。请注意，三个检验的常数全部设置为 0。这些检验的报表显示在图 3.27 中。

“定制检验”的示例

Cholesterol.jmp 样本数据表提供了 20 位患者在 6 个时间段的重复测量数据。研究了 4 个**治疗**组。通常，应该将所有重复测量都用作响应来正确分析该数据。本例仅考虑 **6 月下午**的响应。

假定您想要检验三个对比。您想要比较以下各项的响应均值：

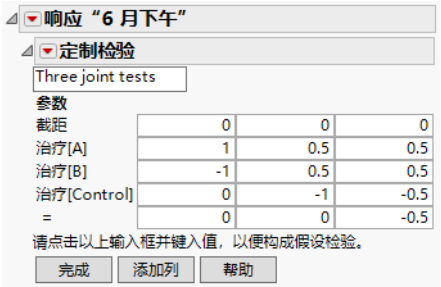
- 治疗 A 与治疗 B
- 治疗 A 和 B 与控制组
- 治疗 A 和 B 与控制组和安慰剂组

使用 “定制检验”来检验这些对比：

1. 选择**帮助 > 样本数据库**，然后打开 Cholesterol.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择 **6 月下午**并点击 **Y**。
4. 选择**治疗**，然后点击**添加**。
5. 点击**运行**。
6. 点击 “响应 ‘6 月下午’”旁边的红色小三角并选择**估计值 > 定制检验**。
7. 在 “定制检验”规格报表中，点击**添加列**两次以创建另外两列。
8. 在可编辑区域中填写检验名称，并在三列中输入值，如图 3.26 所示。

   要查看如何获取这些值，特别是第三列中的那些值，请参见 “统计详细信息”附录中第 457 页的 “参数的解释”。

图 3.26 三个对比的 “定制检验”规格报表



9. 点击**完成**。

图 3.27 中显示的结果指示三个假设无论是单个检验还是联合检验都是显著的。

图 3.27 显示三个对比的检验的 “定制检验”报表

多重比较

使用该选项可获取用来比较模型效应各水平所定义的均值的检验和置信水平。多重比较方法的目标是：确定组均值是否存在差异，同时控制得出不正确结论的概率。“多重比较”选项支持您将组均值与总平均值（均值分析）和控制组均值进行比较。您还可以使用 Tukey HSD 或 Student *t* 执行配对比较。指定 Student *t* 方法时，还可以执行等价性检验来识别有实际意义的配对差值。

Student *t* 方法仅控制单个比较的误差率。因此，它不是真正的多重比较过程。提供的所有其他方法控制关注的所有比较的总误差率。在这些方法中，每一种方法都使用多重比较**调整**来计算 *p* 值和置信限。

若您的模型包含名义型和有序型效应，您可以使用最小二乘均值估计值执行比较，也可以使用“用户定义的估计值”定义特定比较。若您的模型仅包含连续效应，您可以使用“用户定义的估计值”比较均值。

提示：假定连续效应包含相对较少的水平。若您关注的是使用“最小二乘均值估计值”进行比较，则考虑为该效应分配有序型（或名义型）建模类型。

启动选项

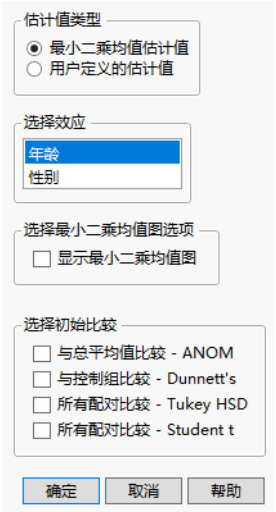
“多重比较”选项的控制窗口示例显示在图 3.28 中。本例基于 Big Class.jmp 数据表，其中将**体重**设置为 Y，**年龄**、**性别**和**身高**设置为模型效应。有两类估计值可用于比较：“最小二乘均值估计值”和“用户定义的估计值”。

最小二乘均值估计值

该选项比较最小二乘均值，仅在模型中有名义型或有序型效应时才可用。回想一下，最小二乘均值是在模型中其他效应设置为某个中性值时计算的均值。（有关最小二乘均值的定义，请参见第

87 页的 “最小二乘均值表”。）您必须选择相关的效应。在图 3.28 中，为**年龄**指定了 “最小二乘均值估计值”。提供显示最小二乘均值图的选项。请参见第 89 页的 “最小二乘均值图选项”。

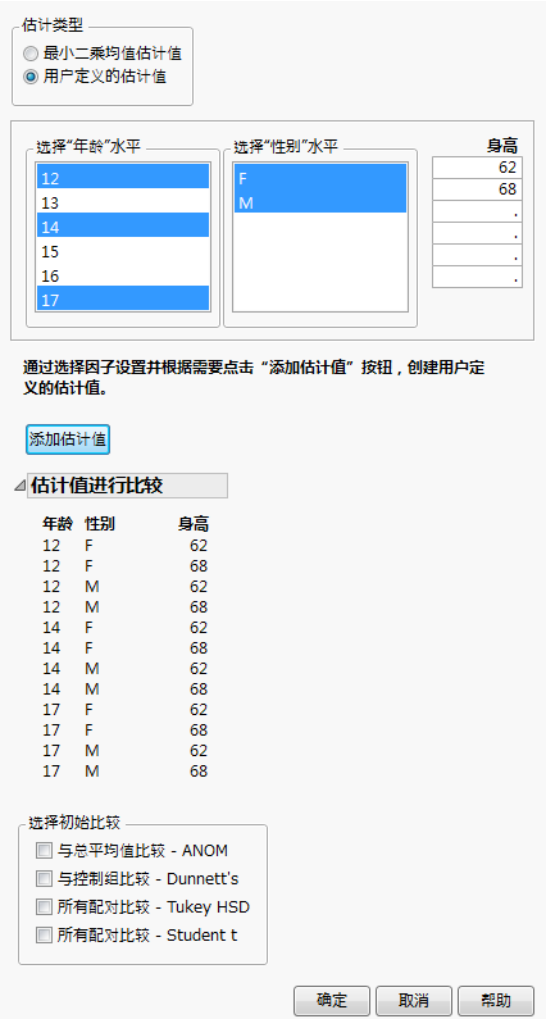
图 3.28 “最小二乘均值估计值”的启动窗口



用户定义的估计值

图 3.29 对如何指定“用户定义的估计值”进行了演示。已选定“年龄”的三个水平和“性别”的两个水平。此外，还手动输入了“身高”的两个值。已点击“添加估计值”按钮，生成了指定水平的所有可能组合的列表。此时，您可以指定更多估计值，并通过再次点击“估计值”按钮将其添加至“估计值进行比较”列表。

图 3.29 “用户定义的估计值”的启动窗口



使用 “用户定义的估计值”时，根据建模类型设置未指定水平的效应：

- 连续效应设置为效应的均值。
- 名义型和有序型效应设置为值排序中的第一个水平。

注意：在本节中，我们将用**均值**一词来指代最小二乘均值的估计值或用户定义的估计值。

选择最小二乘均值图选项

选择**显示最小二乘均值图**以获取最小二乘均值图。若您的效应是交互作用项，则提供创建交互作用图的选项。选择叠加的词条。若您未选择交互作用图，则最小二乘图将嵌套这些效应项。请参见第 89 页的“最小二乘均值图选项”。

选择初始比较

一旦指定了估计值，通过在 “选择初始比较”下作出选择，您可以选择希望在初始报表中查看的比较类型。或者点击 “确定”不作任何选择。

与总平均值比较 - ANOM 将每个效应的最小二乘均值与总最小二乘均值进行比较。（均值分析）。

与控制组比较 - Dunnett's 将每个效应的最小二乘均值与控制水平的最小二乘均值进行比较。

所有配对比较 - Tukey HSD 使用 Tukey HSD 多重性调整来检验效应最小二乘均值的所有配对比较。

所有配对比较 - Student t 检验效应最小二乘均值的所有配对比较，但不使用多重性调整。

通过以上每个选项都可以打开一个报表，顶部区域会显示特定于该报表的详细信息。该信息包括分位数或临界值。对于真正的多重比较过程，会显示用于多重比较调整的方法。若您指定的是“用户定义的估计值”，相对于指定的估计值和这些效应被设置的水平，报表会显示不会发生变化的效应列表。除非您另行指定，所有连续效应都设置为其均值。任何名义型和有序型效应都设置为其值排序中的第一个水平。

若点击“确定”而不从“选择初始比较”列表中进行任何选择，“多重比较”报表打开时会显示“最小二乘均值估计值”表或“用户定义的估计值”表。在“多重比较”红色小三角菜单中，以上列出的所有选项都可用。可用的报表和选项说明如下。

“最小二乘均值估计值”或“用户定义的估计值”报表

默认情况下，“多重比较”选项显示 “最小二乘均值估计值”报表或“用户定义的估计值”报表，具体取决于您在启动窗口中选择的估计值类型。对于关注的每种水平组合，该表提供均值的估计值，以及检验和置信区间。具体而言，该表包含以下信息：

分类效应的水平 报表的第一列标识关注的一个或多个效应。列中的值指定要分析的组。

估计值 每个组的均值的估计值。

标准误差 每个组的均值的标准误差。

自由度 均值是否为 0 的检验的自由度。

95% 下限 均值的置信下限。通过在 “拟合模型”窗口中选择 “设置 alpha 水平”，您可以更改置信水平。

95% 上限 均值的置信上限。

t 比 显著性检验的 *t* 比。仅当您在报表中右击并选择 “列” > “t 比”时才显示该列。

概率 >|t| 显著性检验的 p 值。仅当您在报表中右击并选择“列”>“概率 >|t|”时才显示该列。

算术均值估计值 （仅在“最小二乘均值估计值”报表中显示。）每个组的算术均值的估计值。

数目 （仅在“最小二乘均值估计值”报表中显示。）用于计算每组均值的观测数。

注意：通过在表中右击并选择“列”，您可以获取 t 比和 p 值。

与总平均值比较

该选项将指定水平的均值与这些水平的总均值进行比较。它显示一个表（其中显示总均值差值的置信区间）和一个图（其中显示决策限）。比较所用的方法称为均值分析(ANOM) (Nelson et al. 2005)。ANOM 是一种多重比较过程，用来控制针对总均值的所有配对比较的联合误差率。有关基于 Lipid Data.jmp 样本数据表的报表，请参见图 3.30。

ANOM 的显示可能与方差分析类似。不过，根本上的不同在于：ANOM 标识其均值与所有水平的总均值存在差异的水平。相反，方差分析检验均值自身的差值。

在“与总平均值比较”报表的顶部，您会发现：

分位数 用于构造决策限的 Nelson h 统计量的值。

调整的自由度 用于构造决策限的自由度。

平均值 平均均值。对于最小二乘估计值，平均均值是组最小二乘均值的加权平均值。该加权平均值表示中性设置 （用于计算组最小二乘均值）下的总均值。

具体而言，平均最小二乘均值是权重与矩阵 $L(X'X)^{-1}L'$ 的对角线元素成反比的加权平均值。其中， L 是用于计算组最小二乘均值的系数所构成的矩阵。有关最小二乘均值的技术定义，请参见 SAS Institute Inc.(2020b)。

对于用户定义的估计值，平均均值也按类似方式定义。不过，在这种情况下 L 是用于定义估计值的系数所构成的矩阵。

调整 说明用于获取临界值的方法：

Nelson 提供精确临界值和 p 值。尽可能使用，特别是在估计值不相关时。

Nelson-Hsu 基于使用 Hsu 因子分析近似 (Hsu 1992) 提供近似临界值和 p 值。在无法获取精确值时使用。

Sidak 在 Nelson 和 Nelson-Hsu 均失败时使用。

有关技术细节，请参见 SAS Institute Inc.(2020b)。

“与总平均值比较”报表菜单提供三个选项：

与总平均值差异

对于组均值与总均值的每次比较，该报表提供以下详细信息：

- 要比较的水平

- 差值 — 估计的差值
- 标准误差 — 差值的标准误差
- 置信区间的上下限
- t 比 - “差值”与“标准误差”列之比

与总平均值比较决策图

该决策图在每个组的均值处标绘一个点。在平均均值处标绘了一条水平线。同时标绘上决策限和下决策限。假定与组均值对应的点落在这些限值之外。这表示基于指定显著性水平下的均值分析检验，组均值与总均值存在差异。显著性水平显示在图的下方。

“与总平均值比较决策图”报表菜单包含以下选项：

显示汇总报表 生成显示每组的估计值、决策限和超出限值的表

显示选项 提供用于控制图显示的若干选项。

计算调整的 P 值

将包含 *p* 值（概率>|t|）的列添加到“与总平均值比较”报表。请注意，计算不平衡设计的精确临界值和 *p* 值要求复积分，计算起来可能较为困难。针对此类分位数的计算若失败，则计算 Sidak 分位数，但不提供 *p* 值。

“与总平均值比较”的示例

考虑 Lipid Data.jmp 样本数据表。您关注在控制饮酒状况和心脏病史的前提下四种**吸烟史**类别中是否有任何类别的**咖啡摄入量 （杯 / 天）**均值与咖啡摄入量的总平均值有明显不同。您指定包含**咖啡摄入量 （杯 / 天）**的模型作为响应，将**吸烟史**、**饮酒状况**和**心脏病史**作为模型效应。

1. 选择**帮助** > **样本数据库**，然后打开 Lipid Data.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**咖啡摄入量 （杯 / 天）**并点击 Y。
4. 选择**吸烟史**、**饮酒状况**和**心脏病史**，然后点击**添加**。
5. 点击**运行**。
6. 点击 “响应 ‘咖啡摄入量 （杯 / 天）’ ”旁边的红色小三角并选择**估计值** > **多重比较**。
7. 从 “选择效应”列表中，选择**吸烟史**。
8. 在 “选择初始比较”列表中，选择**与总平均值比较 - ANOM**。
9. 点击**确定**。

结果显示在图 3.30 中，它说明在咖啡摄入量方面非吸烟者和吸烟者的最小二乘均值与总平均值有显著不同。

图 3.30 与“分级”的总平均值比较

响应“咖啡摄入量（杯/天）”

“吸烟史”的多重比较

最小二乘均值估计值

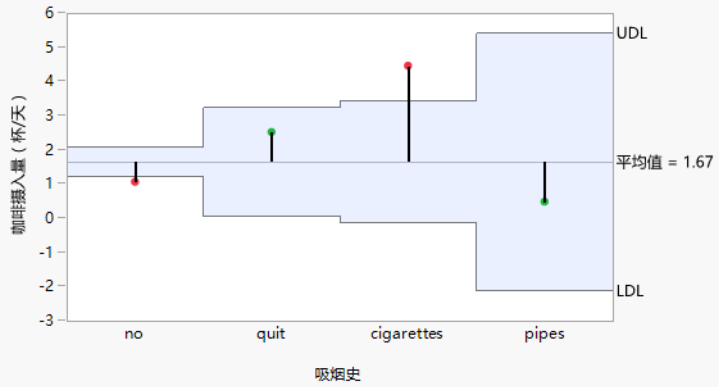
与总平均值比较

分位数 = 2.50786，调整的自由度 = 85.0，平均值 = 1.66719，调整 = Nelson-Hsu

与总平均值差异

吸烟史	-吸烟史	差值	标准误差	调整的下限 95%	调整的上限 95%	t 比
no	平均值	-0.59761	0.165469	-1.01259	-0.18264	-3.61
quit	平均值	0.85689	0.634351	-0.73397	2.44775	1.35
cigarettes	平均值	2.79463	0.704828	1.02702	4.56223	3.96
pipes	平均值	-1.18007	1.502993	-4.94936	2.58922	-0.79

与总平均值比较决策图



α = 0.05

与控制组比较

若选择“与控制组比较 - Dunnett's”，随即打开一个窗口，要求您指定控制组。若选定“最小二乘均值估计值”，该列表将包含您选定的效应的所有水平。若选定“用户定义的估计值”，该列表将包含您指定的效应水平组合。

选择控制组并点击“确定”后，“与控制组比较”报表随即显示在“拟合最小二乘法”报表中。该选项比较指定设置的均值与控制组均值。它显示一个表（其中显示与控制组差值的置信区间）和一个显示决策限的图。使用 Dunnett 方法进行比较。Dunnett 方法是一种多重比较过程，用于控制所有比较上的误差率 (Hsu 1996; Westfall et al. 2011)。

若无法精确计算 p 值和置信区间，则使用 Hsu 因子分析近似 (Hsu 1992)。请注意，计算不平衡设计的精确临界值和 p 值要求复积分，可能需要大量计算。针对此类分位数的计算若失败，则计算 Sidak 分位数。

除了对于指定估计值不会改变的效应列表之外，“与控制组比较”报表的顶部还显示：

分位数 Dunnett 检验的临界值。

调整的自由度 用于构造置信区间的自由度。

控制 定义控制组的设置。若已经选定单一效应，则这是单个水平；若指定了多个效应的用户定义组合，则这是多个水平的组合。

调整 用于获取临界值的方法：

Dunnett 提供精确临界值和 p 值。尽可能使用，特别是在估计值不相关时。

Dunnett-Hsu 基于使用 Hsu 因子分析近似 (Hsu 1992) 提供近似临界值和 p 值。在无法获取精确值时使用。

Sidak 在 Dunnett 和 Dunnett-Hsu 均失败时使用。

有关技术细节，请参见 SAS Institute Inc.(2020b)。

“与控制组比较”报表菜单提供三个选项：

与控制组差异

对于组均值与控制均值的每次比较，该报表都提供以下详细信息：

- 要比较的水平
- 差值 — 估计的差值
- 标准误差 — 差值的标准误差
- 置信区间的上下限
- t 比 - “差值”与“标准误差”列之比

与控制组比较决策图

该决策图在与控制组作比较的每个组的均值处标绘一个点。有一条水平线显示控制组的均值。同时标绘上决策限和下决策限。若某个点落在这些限值之外，则这点对应组的均值与控制组均值存在差异（基于指定显著性水平下的 Dunnett 检验）。该水平显示在图下方。

“与控制组比较决策图”报表菜单包含以下选项：

显示汇总报表 生成显示每组的估计值、决策限和超出限值的表

显示选项 提供用于控制图显示的若干选项。

计算调整的 P 值

将包含 p 值（概率>|t|）的列添加到“与总平均值比较”报表。请注意，计算不平衡设计的精确临界值和 p 值要求复积分，计算起来可能较为困难。针对此类分位数的计算若失败，则计算 Sidak 分位数，但不提供 p 值。

所有配对比较

“所有配对比较”选项可显示“Tukey HSD 所有配对比较”或“Student t 所有配对比较”报表 (Hsu 1996; Westfall et al. 2011)。构造 Tukey HSD 比较，以使显著性水平联合应用到所有配对比较。相比之下，对于 Student t 比较，显著性水平应用到每一单个比较。使用 Student t 检验执行若干配对比较时，某个比较被错误地判定存在差异的风险可能远远超出规定的显著性水平。

在“Tukey HSD 所有配对比较”报表的顶部，您会发现：

分位数 检验的临界值。请注意，对于 Tukey HSD，分位数为 $q/(\sqrt{2})$，其中 q 是适当的学生化范围统计量的百分点。

调整的自由度 用于构造置信区间的自由度。

调整 说明用于获取临界值的方法：

Tukey 提供精确临界值和 p 值。在均值不相关且具有相等方差时或设计的方差平衡时使用。

Tukey-Kramer 提供近似临界值和 p 值。在无法获取精确值时使用。

有关技术细节，请参见 SAS Institute Inc.(2020b)。

在“Student 所有配对比较”报表中，您可找到用于 t 检验的分位数（或临界值），以及用于 t 检验的自由度 DF。

“所有配对差异”报表

Tukey HSD 和 Student t 比较所有水平对。对于每个配对比较，“所有配对差异”报表都显示：

- 要比较的水平
- 差值 — 均值间的估计差值
- 标准误差 — 差值的标准误差
- t 比 — 确定差值是否为 0 的检验的 t 比
- 概率>|t| — 检验的 p 值
- 均值差值的置信区间的上下限

所有配对比较散点图

该图有时称为**差值图**或**均值-均值散点图**，其中显示所有均值配对差值的置信区间。（相关示例，请参见图 3.32。）颜色指示哪些差值显著。

该图将参考线显示为对角线上向上倾斜的线。这条线表示两个均值相等的点。每条线段都对应于某一配对比较的置信区间。线段上显示的点的坐标是对应组的均值。悬停在其中一个点上方可显示工具提示，该提示标识要比较的组并显示估计的差值。若线段与对角线相交，则均值可能相等，比较不显著。

“配对比较散点图”有以下选项：

显示参考线 显示散点图上点的参考网格线。若散点图中有很多点，则不推荐。若有很多点，则最好悬停在各点上方以查看工具提示标签。

所有配对差异连接字母

使用该选项来显示使用连接字母说明显著比较和非显著比较的报表。没有由相同字母连接的水平存在显著差异。由相同字母连接的水平没有显著差异。

保存所有配对差值连接字母表

该选项创建一个数据表，其中的列包含效应水平、连接字母、最小二乘均值、其标准误差和置信区间。该数据表包含名为“条形图”的脚本，该脚本可生成叠加了置信区间的最小二乘均值的着色条形图。各水平按照最小二乘均值的降序排列。

等价性检验

使用该选项可执行一个或多个等价性检验。若您想要检测有**实际**意义的差值，等价性检验很有用。您必须为组均值指定一个阈值差，比这更小的差值可被视为实际上等价。换言之，若两个组均值的差值未超过该数值，您愿意将其视为等价。

一旦指定该值，“等价性检验”报表随即显示。您指定的界限显示在该报表顶部。该报表包含一个提供等价性检验的表，以及一个显示这些检验的散点图。这些等价性检验和置信区间均基于 Student *t* 临界值。

注意：等价性检验仅适用于 Student *t* 方法。

等价性 TOST 检验

双单侧检验 (TOST) 方法用于检验均值之间的实际差值 (Schuirmann 1987)。针对实际差值超过阈值的原假设，构造双单侧合并方差 *t* 检验。若两个检验都拒绝，则均值差值在统计上未超过任一阈值。因此，这些组被视为实际上等价。若只有一个检验拒绝或两个检验均未拒绝，则各组可能实际上不等价。

对于每个比较，“等价性 TOST 检验”报表都提供以下信息：

- 差值 — 均值的估计差值
- t 比下限、t 比上限 — 双单侧合并方差显著性检验的 *t* 比的下限和上限
- p 值下限、p 值上限 — 与 *t* 比的下限和上限对应的 *p* 值
- 最大 p 值 — *p* 值下限和上限的最大值
- 均值差值的 1–2α 置信区间的上下限。

注意：等价性 TOST 检验仅适用于 Student *t* 方法。

等价性检验散点图

该散点图使用颜色指示哪些均值实际上是等价的，哪些实际上不等价，正如等价性检验确定的那样。该图有时称为**差值图**或均值-均值散点图。

该图在对角线上显示一条参考实线以及一个着色参考带。参考带的宽度是实际差值的两倍宽。线段上的点的坐标是对应组的均值。对角线上隐含第三个轴，其中每条线段都对应于某一配对比较的 1–2α 置信区间。悬停在其中一个点上方可显示工具提示，该提示指示要比较的组和估计的差值。若线段完全包含在对角带内，由此判定均值实际上等价。

注意：等价性检验散点图仅适用于 Student *t* 方法。

等价性检验散点图具有以下选项：

显示参考线 显示散点图上点的参考线。若散点图中有很多点，则不推荐。若有很多点，则最好悬停在各点上方以查看工具提示标签。

删除

该选项可从“Student t 所有配对比较”报表中删除“等价性检验”报表。

“Tukey HSD 所有配对比较”的示例

考虑 Lipid Data.jmp 样本数据表。您关注在两个年龄组 （25 和 35 岁）和平均身高保持不变的情况下**胆固醇**是否存在性别差异，是否在非吸烟者和曾吸烟者（其**吸烟史**分别等于 no 和 quit）方面存在差异。

1. 选择**帮助** > **样本数据库**，然后打开 Lipid Data.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**胆固醇**并点击 Y。
4. 选择**性别**、**年龄**、**身高**和**吸烟史**，然后点击**添加**。
5. 点击**运行**。
6. 点击 “响应 ‘胆固醇’”旁边的红色小三角并选择**估计值** > **多重比较**。
7. 从 “估计值类型”列表中，点击**用户定义的估计值**。
8. 从 “选择性别水平”列表中，选择 female （默认情况下应已选定）和 male。
9. 从 “选择吸烟史水平”列表中，选择 no 和 quit。
10. 在**年龄**列表中，在前两行输入年龄 25 和 35。

    不要在**身高**列表中输入任何值。因为未指定**身高**值，在 “多重比较”报表中将使用**身高**列的均值。
11. 点击**添加估计值**。

    请注意，并非您指定的水平的所有可能组合都显示在 “进行比较的估计值”报表中。
12. 在 “选择初始比较”列表中，选择**所有配对比较** - Tukey HSD。

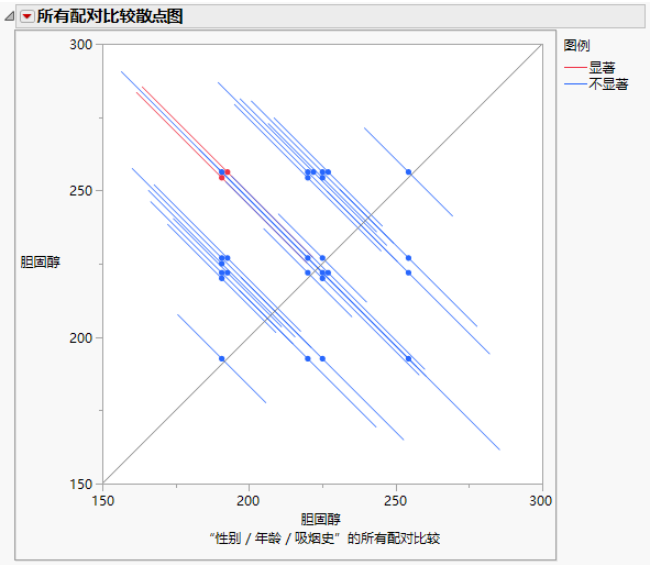
    检查您的窗口是否按图 3.31 中所示填写。

图 3.31 已填写的 “用户定义的估计值” 窗口

13. 点击**确定**。

“所有配对差值” 报表指示 28 个配对比较中的 2 个是显著的。图 3.32 中显示的 “所有配对比较散点图” 将这些比较的置信区间显示为红色。您可以悬停在任何点上方以确定该点代表哪个配对比较。工具提示还包含该比较中两个水平的差值。图 3.32 中的两个红色点表示对男性和女性比较 35 岁的曾吸烟者和 25 岁的非吸烟者的点。

图 3.32 针对用户定义的比较的 “所有配对比较散点图”



比较斜率

当有一个名义型项、一个连续型项以及它们针对固定效应的交互作用效应时，就会显示 “比较斜率” 选项。该选项生成一个报表，该报表支持您比较协方差分析 (ANCOVA) 模型中的斜率。该报表将交互作用效应每个水平的斜率与总体斜率作比较。该比较使用包含总平均值的均值分析 (ANOM)。有关均值分析 (ANOM) 报表的详细信息，请参见第 116 页的 “与总平均值比较”。

总平均值斜率是斜率的加权平均值，其中，权重与斜率估计值的方差成反比。这些方差是 “与总平均斜率的差异” 表中的 “标准误差” 列的平方值。

联合因子检验

若存在交互作用效应，则会显示 “联合因子检验” 选项。对于模型中的每个主效应，JMP 会生成联合检验，用于确定涉及该主效应的项的所有系数是否为 0。该检验以所有其他效应位于模型中为条件。具体而言，联合检验是受限模型的一般线性假设检验。在该模型中，与指定效应和包含该效应的交互作用对应的所有参数都设置为 0。

“联合因子检验” 报表的示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**体重**并点击 **Y**。
4. 验证**次数**框中显示的是 2。

5. 选择**年龄**、**性别**和**身高**，然后点击**宏** > **析因次数**。
6. 点击**运行**。
7. 点击“响应 ‘体重’”红色小三角并选择**估计值** > **联合因子检验**。

图 3.33 “联合因子检验”报表

联合因子检验

项	自由度	平方和	F 比	概率>F
年龄	15	6116.2127	2.8488	0.0139*
性别	7	2113.1080	2.1091	0.0879
身高	7	9217.8156	9.2002	<.0001*

请注意，**年龄**检验的自由度为 15。该检验涉及**年龄**的五个参数、**年龄*性别**的五个参数，以及**身高*年龄**的五个参数。该检验的原假设是全部 15 个参数都为 0。

逆预测

当您使用统计模型在给出响应变量的值的情况下推断解释变量的值时，就会出现**逆预测**。逆预测有时称为**校准**。

通过在“估计值”菜单上选择“逆预测”，您可以估计与响应的指定值对应的自变量 X 的值（图 3.37）。此外，您可以指定模型中其他解释变量的值（图 3.37）。逆预测计算为与指定的响应值对应的 X 的值提供置信限。您可以将响应值指定为响应均值，或只是单个响应。相关示例，请参见第 125 页的““逆预测”示例”。

分析多个解释变量

当模型包括多个解释变量时，您可以针对其他变量的指定值预测 X 的值。您可能希望预测当一个人的休息时脉搏为 60 时，吸氧量为 50 所需的跑步时间。您可能希望针对男性和女性分别进行逆预测。使用逆预测选项指定这些要求。

在逆预测窗口中，解释变量列表显示在左侧。（相关示例，请参见图 3.37。）每个连续变量最初都设置为其均值。每个名义型或有序型变量都设置为其最低水平（按照值排序）。您必须删除想要预测的变量的值，将其设置为缺失。此外，您还必须指定逆预测中希望固定的其他变量的值（若这些变量值与默认设置值不同）。在窗口右侧的列表中，您可以提供一个或多个关注的响应值。有关示例，请参见第 127 页的“在有多个模型效应的情况下预测单个 X 值的示例”。

注意：逆预测的置信限有时可生成单侧甚至是无穷区间。有关技术细节的信息，请参见“统计详细信息”附录中第 481 页的“带置信限的逆预测”。

“逆预测”示例

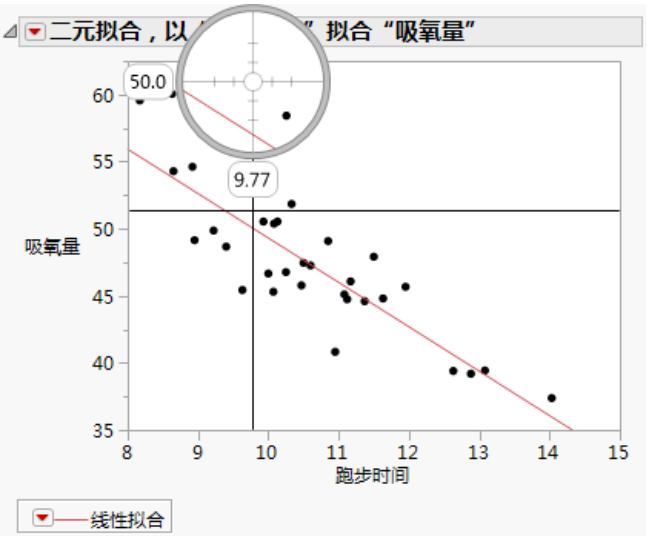
在本例中，您拟合的回归模型用于根据**跑步时间**预测吸氧量。然后您要估计导致指定的吸氧量值的**跑步时间**值。只有一个 X（即**跑步时间**），所以您可以先使用“以 X 拟合 Y”平台获取逆预测值的直观近似值。

1. 选择**帮助** > **样本数据库**，然后打开 Fitness.jmp。

2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**吸氧量**并点击 **Y，响应**。
4. 选择**跑步时间**并点击 **X，因子**。
5. 点击**确定**。
6. 点击 “二元拟合，以 ‘跑步时间’ 拟合 ‘吸氧量’” 红色小三角并选择**拟合线**。
   使用如下所述的十字准线工具近似计算导致**吸氧量**的均值为 50 的**跑步时间**值。
7. 选择**工具** > **十字准线**。
8. 点击预测线，然后拖动十字准线工具以查找**吸氧量** = 50 的逆预测。

图 3.34 显示**跑步时间**约为 9.77，它给出**吸氧量**约为 50 的逆预测。

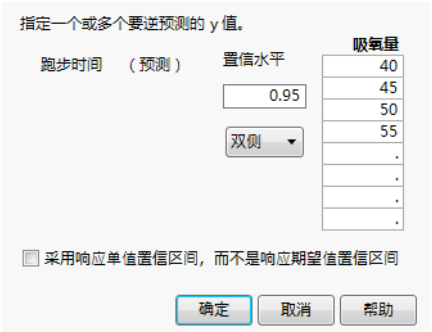
图 3.34 Fitness.jmp 的 “二元拟合”



要获取**跑步时间**的精确预测以及置信区间，请使用 “拟合模型” 启动窗口：

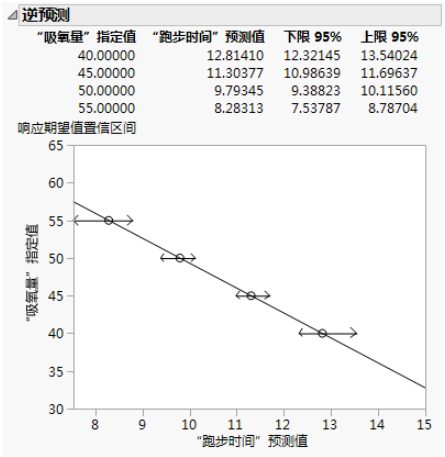
1. 从 Fitness.jmp 样本数据表中，选择**分析** > **拟合模型**。
2. 选择**吸氧量**并点击 **Y**。
3. 选择**跑步时间**，然后点击**添加**。
4. 点击**运行**。
5. 点击 “响应 ‘吸氧量’” 红色小三角并选择**估计值** > **逆预测**。
6. 按图 3.35 所示为**吸氧量**输入四个值。
7. 点击**确定**。

图 3.35 完成的 “逆预测规格” 窗口



“逆预测” 报表提供了与每个指定的**吸氧量**值对应的**跑步时间**预测值。该报表还显示了相对于获取响应均值，这些**跑步时间**值的 95% 置信下限和置信上限。

图 3.36 “逆预测” 报表



导致**吸氧量**值为 50 的**跑步时间**精确预测值为 9.7935。该值接近图 3.34 所示的 “二元拟合” 报表中显示的**跑步时间**近似值 9.77。“逆预测” 报表还提供了一个图，其中显示**吸氧量**与**跑步时间**之间的线性关系以及置信区间。

在有多个模型效应的情况下预测单个 X 值的示例

本例预测**休息时脉搏**为 60 时导致吸氧量为 50 的**跑步时间**。分别针对男性和女性预测**跑步时间**。

1. 从 Fitness.jmp 样本数据表中，选择**分析 > 拟合模型**。
2. 选择**吸氧量**并点击 Y。
3. 选择**性别**、**跑步时间**和**休息时脉搏**，然后选择**添加**。
4. 点击**运行**。

5. 点击“响应 ‘吸氧量’”红色小三角并选择**估计值** > **逆预测**。
6. 删除**跑步时间**的值，因为您想要预测该值。
7. 选择**性别**旁边的**全部**框，为**性别**的所有水平估计**跑步时间**。
8. 用 60 替换**休息时脉搏**的均值。
9. 为**吸氧量**输入值 50，如图 3.37 所示。
10. 点击**确定**。

图 3.37 多重回归模型的逆预测规格

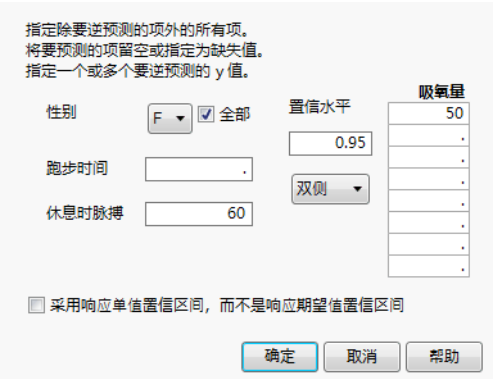
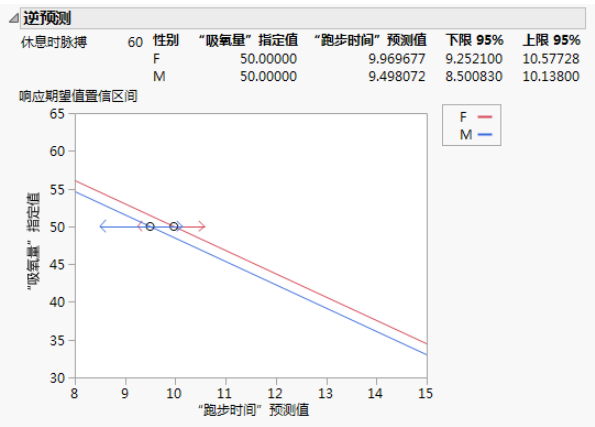


图 3.38 所示的报表为男性和女性提供了**跑步时间**的预测值。其中还包含响应均值为 50 的情况下**跑步时间**值的 95% 置信区间。

图 3.38 多重回归模型的 “逆预测” 报表



该图显示**休息时脉搏**为 60 时针对女性和男性的线性拟合。两个置信区间分别显示为红色和蓝色。

Cox 混料

注意：该选项仅可用于混料模型。

标准最小二乘法使用 Scheffé (1958) 中建议的参数化拟合混料模型。使用该模型的参数不太容易判断混料成分的影响。Cox 混料模型是 Scheffé 模型的重新参数化和受约束版本。使用其参数估计值，您可以导出因子效应和相对于设计空间中的参考点的响应曲面形状。有关完整讨论，请参见 Cornell (1990)。

使用“Cox 混料”选项可打开一个窗口，您可以在其中输入参考混料。若为参考混料输入的成分总和不为 1，则会按比例调整各个成分，使其总和为 1。重新调整的混料作为“参考混料”显示在报表中。成分效应也显示在报表中。Cox 成分效应是随着因子沿 Cox 效应方向从其最小值变为最大值所带来的预测响应中的差异。

“Cox 混料”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Five Factor Mixture.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y1 并点击 Y。
4. 选择 X1 到 X5。
5. 选择**宏** > **混料响应曲面**。
6. 点击**运行**。
7. 点击“响应 ‘Y1’”红色小三角并选择**估计值** > **Cox 混料**。
8. 在“参考混料”规格窗口中，输入以下值：
   - 在 X1 旁边，键入 0.15。
   - 在 X2 旁边，键入 0.25。
   - 在 X3 旁边，键入 0.4。
   - 在 X4 旁边，键入 0.1。
   - 在 X5 旁边，键入 0.1。
9. 点击**确定**。

图 3.39 Cox 混料

Cox 参考混料模型

Cox 参数估计值

参数	估计值	标准误差	t 比	概率>|t|
截距	2.071	1.2541	1.651	0.1035
X1	3.120	7.7538	0.402	0.6887
X2	-5.828	7.0104	-0.831	0.4089
X3	-0.696	3.5822	-0.194	0.8465
X4	3.935	12.6619	0.311	0.7570
X5	8.740	12.5405	0.697	0.4883
X1^2	-15.897	16.3627	-0.972	0.3349
X2^2	17.877	13.5932	1.315	0.1931
X3^2	5.234	8.2498	0.634	0.5280
X4^2	-91.239	159.9347	-0.570	0.5703
X5^2	-145.693	158.3991	-0.920	0.3611
X1*X2	-2.938	24.9012	-0.118	0.9065
X1*X3	-7.617	19.1697	-0.397	0.6924
X2*X3	-16.250	18.7489	-0.867	0.3893
X1*X4	53.690	50.5711	1.062	0.2923
X2*X4	-17.702	49.9082	-0.355	0.7240
X3*X4	-7.808	44.0229	-0.177	0.8598
X1*X5	31.813	50.6186	0.628	0.5319
X2*X5	-2.279	49.3761	-0.046	0.9633
X3*X5	17.983	43.5145	0.413	0.6808
X4*X5	177.431	97.3862	1.822	0.0731

Cox 参考混料

因子	参考混料
X1	0.1500000
X2	0.2500000
X3	0.4000000
X4	0.1000000
X5	0.1000000

成分效应

成分	Cox 效应	成分范围
X1	0.80746	0.2200000
X2	-2.33103	0.3000000
X3	-0.48747	0.4200000
X4	0.65585	0.1500000
X5	1.45663	0.1500000

该报表显示 Cox 混料模型的参数估计值，以及标准误差和假设检验。参考混料显示在右侧。成分效应随同成分范围显示在下方。

参数功效

统计检验的**功效**是在实际存在差异时检验将显示出显著的概率。检验功效指示您的研究声明真实效应显著的概率。“参数功效”选项用于处理**回顾**功效分析。

注意：要确保您的研究所包含的观测数足以检测所需的差值，请在**设计**实验时使用关于功效的信息。这种分析称为前瞻功效分析。请考虑使用“实验设计”平台来设计您的研究。“实验设计”>“样本大小与功效”和“实验设计”>“评估设计”都有助于前瞻功效分析。有关使用标准最小二乘法的前瞻功效分析的示例，请参见第 185 页的“前瞻功效分析”。

用来检测差值的检验的功效受以下因素影响：

- 样本大小
- 未知的剩余误差方差
- 检验的显著性水平
- 要检测的效应的大小

假定您已经进行了研究、分析了数据并发现关注的效应不显著。您可能会关注您有可能检测到的差值大小或是您执行的检验的功效。或许您可能想要了解以高概率检测到给定大小的差值所需的观测数。

“参数功效”选项在“参数估计值”报表中插入与回顾功效分析相关的三列值。提供最小显著值 (LSV0.05)、最小显著数 (LSN0.05) 和功效计算 (AdjPower0.05)。

“参数功效”计算应用于新样本，该样本具有与观测样本相同的变异性特征。

警告：LSV0.05、LSN 和 AdjPower0.05 提供的结果不应在前瞻功效分析中使用。它们不能反映未来研究固有的不确定性。

- LSV0.05 是**最小显著值**。该数值是可令该检验在显著性水平 0.05 下显著的估计值的最小绝对值。具体而言，假定该设计的观测数、均方误差以及平方和与叉积矩阵都保持不变。那么，若估计值的绝对值小于 LSV0.05，则“概率>|t|”值将超过 0.05。（请参见第 183 页的“最小显著值 (LSV)”。）
- LSN 是**最小显著数**。该数值是可令该检验在显著性水平 0.05 下显著的观测数。具体而言，假定该设计的参数估计值、均方误差以及平方和与叉积矩阵都保持不变。那么，若观测数小于 LSN，则“概率>|t|”值将超过 0.05。（请参见第 182 页的“最小显著数 (LSN)”。）
- AdjPower0.05 是**调整功效**值。该数字是该检验将为显著的概率的估计值。当前研究中的样本值将替换为功效计算中通常使用的参数值。**调整**功效计算对因以下替换而造成的偏倚进行调整：在非中心参数的公式中直接替换成样本估计值 (Wright and O’Brien 1988)。（请参见第 183 页的“调整功效和置信区间”。）

LSV、LSN 和调整功效有助于评估检验的灵敏度。这些回顾计算还提供一种启发式教学工具。不过，您必须谨慎解释这些值 (Hoenig and Heisey 2001)。

有关 LSV、LSN 和调整功效的详细信息，请参见第 180 页的“功效分析”。有关回顾分析的示例，请参见第 184 页的“回顾功效分析的示例”。

估计值的相关性

“估计值”菜单上的“估计值的相关性”命令计算参数估计值的相关性矩阵。这些相关性指示是否存在共线性。

若要了解该矩阵的构造，考虑典型的最小二乘回归方程。在此，响应 (Y) 是预测变量 (x) 以及误差 (ε) 的线性函数：

$$Y = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + \varepsilon$$

数据表的每一行都包含一个响应值和 p 个预测变量的值。对于每个观测，预测变量值被视为固定不变。不过，响应值被视为随机变量的实现。

考虑预测变量值为固定值，对于任意一组 Y 值，系数 β_0、β_1、…、β_p 均可估计。通常，不同组的 Y 值将导致不同的系数估计值。“估计值的相关性”选项计算这些参数估计值的理论相关性。（有关技术细节的信息，请参见第 176 页的“‘定制检验’示例的详细信息”。）

参数估计值的相关性仅依赖于预测变量值和表示截距的项。两个参数估计值之间的相关性不受响应值影响。

两个估计值之间强正相关表明两个对应预测变量之间可能存在共线性关系。但是要注意，您需要谨慎解释这些相关性 (Belsley et al. 1980, p. 185, 92–94)。此外，发生均值偏移的预测变量的尺度重新调整会更改其参数估计值与截距值的相关性。

“估计值的相关性”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Socioeconomic.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**房屋均值**并点击 Y。
4. 选择**总人口**、**平均受教育年数**、**总就业率**和**专业服务**，然后点击**添加**。
5. 在 “重点”列表中，选择**最小报表**。
6. 点击**运行**。
7. 点击 “响应 ‘房屋中位值’”旁边的红色小三角并选择**估计值** > **估计值的相关性**。

图 3.40 “估计值的相关性”报表

估计值的相关性

相关性

	截距	总人口	平均受教育年数	总就业率	专业服务
截距	1.0000	-0.5743	-0.9818	0.4719	0.6903
总人口	-0.5743	1.0000	0.5871	-0.9746	-0.2398
平均受教育年数	-0.9818	0.5871	1.0000	-0.5132	-0.7085
总就业率	0.4719	-0.9746	-0.5132	1.0000	0.1094
专业服务	0.6903	-0.2398	-0.7085	0.1094	1.0000

该报表显示“截距”与**受教育中位年数**的参数估计值之间强负相关 (–0.9818)。**总人口**与**总就业率**之间也是强负相关 (–0.9746)。

名义型效应的编码

在模型中输入名义型建模类型的列时，JMP 在内部将其表示为一组连续指标变量。每个变量仅采用值 –1、0 和 1。（请注意，该编码是使用指标变量对名义型变量编码的众多方法之一。）若名义型列包含 *n* 个水平，则需要 *n-1* 个这样的指标变量来表示该列。（需要 *n-1* 个指标变量的直接原因是：与名义型列关联的主效应具有 *n-1* 个自由度。）“统计详细信息”附录中第 456 页的“名义型因子”中提供了完整的详细信息。

提示：通过从主报表的红色小三角菜单中选择“保存列”>“保存编码表”，您可以查看编码。请参见第 157 页的“保存编码表”。

假定您有一个包含四个水平的名义型列。以 Cholesterol.jmp 样本数据表中的**治疗**列为例。**治疗**列包含四个水平：A、B、Control 和 Placebo。前三个水平中分别由一个指标变量表示。这些指标变量名为“治疗[A]”、“治疗[B]”和“治疗[Control]”。

给定水平的指标变量为该水平分配值 1，为最后一个水平分配 –1，为其余水平分配 0。表 3.1 显示了本例中的 “治疗 [A]”、“治疗 [B]” 和 “治疗 [Control]” 指标变量的定义。例如，考虑指标变量 “治疗 [A]”。如表 3.1 中所示，该变量分配了以下值：

- 值 1 分配给**治疗** = A 的行
- 值 0 分配给**治疗** = B 或 = Control 的行
- 值 -1 分配给**治疗** = Placebo 的行

表 3.1 演示 Cholesterol.jmp 中 “治疗” 的指标变量

分配给行的治疗	治疗[A]	治疗[B]	治疗[Control]
A	1	0	0
B	0	1	0
Control	0	0	1
Placebo	-1	-1	-1

水平的顺序由 “值顺序” 列属性决定（若已经分配该属性），或由 JMP 指定的默认排序决定。通常，默认排序对数字而言即数值排序顺序，对字符数据而言即字母数字排序顺序。不过，特定的分类值（如月份名称）会默认以适当顺序排序。有关值排序的详细信息，请参见《使用 JMP》。

这些变量用于对模型进行参数化。它们一般不显示在数据表中，但这些变量的估计系数在 “参数估计值” 和其他报表中提供。尽管也可以使用其他很多编码方法，但该编码已证实是一种实用且可解释的方法。

有关名义型效应的编码的信息，请参见 “统计详细信息” 附录中第 467 页的 “有序型因子”。

效应筛选

筛选设计通常不提供误差自由度，因此，经典效应检验不可用。在这些情形下，“效应筛选” 尤其有用。

对于这些设计，大多数关于效应大小的推断都假定非截距参数的估计值不相关且具有相等方差。这些假设对于与许多经典实验设计关联的模型都成立。不过，也有些情形下这些假设并不成立。对于上述两种情形，“效应筛选” 平台会指导您确定哪些效应显著。

“效应筛选” 平台使用**效应稀疏**原则 (Box and Meyer 1986)。该原则主张在筛选设计所研究的效应中只有相对较少的效应是活跃的。多数效应是非活跃的，这表示其真正效应为零，并且其估计值是随机误差。

提供以下“效应筛选”选项：

统一尺度估计值 提供与尺度统一后均值为 0 且极差为 2 的因子对应的参数估计值。请参见第 134 页的“统一尺度估计值和连续项的编码”。

正态图 标识偏离正态性的参数估计值，可帮助您确定哪些效应是活跃的。请参见第 139 页的“‘正态图’报表”。

Bayes 图 使用 Bayesian 方法为所有模型项计算后验概率。请参见第 140 页的“‘Bayes 图’报表”。

Pareto 图 标绘正交化和标准化参数估计值的绝对值，将其关联到其绝对值的总和。请参见第 141 页的“‘Pareto 图’报表”。

统一尺度估计值和连续项的编码

参数估计值高度依赖于其相应因子的尺度。将因子从克转换为千克后，其参数估计值将变化一千倍。将同样的变化应用于平方（二次）项时，其参数估计值将变化一百万倍。

为了更好地理解和比较效应大小，您应该以尺度不变的方式检查参数估计值。应该使用一个尺度，将参数估计值的大小与其相应项对响应所产生的效应大小关联起来。可通过许多方法来执行该操作。

报表红色小三角菜单上的“效应筛选”>“统一尺度估计值”选项提供与统一尺度因子对应的系数。这些因子尺度统一后均值为 0 且极差为 2。图 3.41 显示了 Drug.jmp 的报表。

若因子样本值的最大值和最小值与样本均值的距离相等，则统一尺度因子的范围为 –1 到 1。这种统一尺度对应于实验设计中使用的传统编码。在包含单个因子的回归的情况下，随着因子遍历其整个范围时响应预测值会发生变化，统一尺度参数估计值等于该变化的一半。

在数据涉及未编码值时，统一尺度估计值对于评估模型项的影响很重要。对于正交设计，统一尺度估计值与未编码数据的估计值相同。

注意：“编码”列属性线性统一因子值的尺度，以便其编码值介于 –1 到 1 之间。参数估计值以这些编码值的形式提供，这样在此情形下就不再需要统一尺度估计值。（与用于获取统一尺度估计值的变换不同，编码值的均值不一定为 0。）

“统一尺度估计值”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Drug.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 y 并点击 Y。
4. 选择**药物**和 x，并将它们添加到**构造模型效应**列表。
5. 将“重点”改为**最小报表**。
6. 点击**运行**。

7. 点击“响应 ‘Y’”红色小三角并选择**效应筛选** > **统一尺度估计值**。

该报表指示连续因子 x 以其均值为中心，以其极差的一半为尺度。

图 3.41 “统一尺度估计值”报表

统一尺度估计值

名义型因子 - 扩展至所有水平
连续因子 - 以均值为中心，以极差的一半为尺度

项	统一尺度估计值	标准误差	t 比	概率>|t|
截距	7.9	0.731352	10.80	<.0001*
药物[a]	-1.185037	1.060822	-1.12	0.2742
药物[d]	-1.076065	1.041298	-1.03	0.3109
药物[f]	2.2611017	1.093974	2.07	0.0488*
x	8.8846543	1.480478	6.00	<.0001*

该模型拟合三条平行线，每个**药物**组对应一条线。x 值的范围介于 3 到 21 之间。x 的统一尺度估计值为 8.8846543，是任一**药物**组在 x = 21 时的预测值与 x = 3 时的预测值之间差值的一半。通过从报表的红色小三角菜单中选择**保存列** > **预测公式**，您可以验证这一事实。然后，添加行以获取每个**药物**组在 x = 21 和 x = 3 时的预测值。

因此，在这一特定数据集中的整个 x 值范围内，x 的影响是在大约 17.8 个单位的范围内改变响应。请注意，基于原始数据的 x 的参数估计值为 0.9871838；该估计值不允许用响应来进行直接解释。

图选项

“正态图”、“Bayes 图”和“Pareto 图”选项提供作为“效应筛选”报表一部分显示的报表。可以构造这些报表，使其能够针对估计值之间的不等方差和相关性进行校正。

注意：“正态图”、“Bayes 图”和“Pareto 图”选项要求您的模型至少涉及四个参数。其中一个可以是截距。

变换

选择任意图选项后，以下信息将显示在“效应筛选”报表标题正下方：

- 若估计值具有相等方差且不相关，将显示以下两条注释：
  - 参数估计值的方差相等。
  - 参数估计值彼此不相关。
- 若估计值具有不等方差或相关，则选项列表将取代有关注释。默认选定的列表项显示 JMP 已经变换估计值。若您想要撤销其中一个变换或两个变换全部撤销，请选择相应的列表项。

Lenth PSE 值

Lenth PSE（**伪标准误差**）表显示在注释或选项列表的正下方。（有关 PSE 的说明，请参见第 105 页的“Lenth PSE”。）Lenth 表中所显示的统计量取决于参数估计值的方差和相关性。

当参数估计值具有不等方差并且不相关时， Lenth 表提供以下统计量：

Lenth PSE 估计值的 Lenth 伪标准误差。

当参数估计值具有不等方差并且/或者相关时，Lenth 表提供以下统计量：

t 检验尺度 Lenth PSE 将为变换参数估计值计算的 Lenth 伪标准误差除以其采用变换尺度的标准误差计算得出。

编码尺度 Lenth PSE 变换参数估计值的 Lenth 伪标准误差。

“参数估计值总体”报表

“参数估计值总体”报表提供参数估计值的检验。这些检验基于选项列表中指定的变换。

- 选项**使用标准化成方差相等的估计值**应用归一化变换以使方差标准化。方差不等时默认选定该选项。
- 选项**使用正交化成不相关的估计值**应用变换以删除相关性。估计值相关时默认选定该选项。应用的变换等同于计算序贯平方和时所用的变换。在一个变量之前的所有变量输入模型后，这些估计值可测量这个变量的额外贡献。
- 若注释指示估计值具有相等方差且不相关，则不应用任何变换。

表中显示的列取决于最初在注释或选项列表中说明的选项。该报表突出显示与 p 值小于等于 0.20 的估计值对应的所有行。报表的所有版本都提供“项”、“估计值”，以及“t 比”和“概率 >|t|”或“伪 t 比”和“伪 p 值”。

项 提供其参数估计值受关注的模型项。

估计值 提供参数的估计值。仅当模型效应具有相同的尺度时，才可以比较估计值大小。

t 比 当存在误差自由度时显示。该值是参数估计值除以其标准误差之后的结果。

概率 >|t| 提供检验的 p 值。若应用了变换，该选项将提供使用变换数据进行检验的 p 值。

伪 t 比 当不存在误差自由度时显示。若参数的相对标准误差相等，“伪 t 比”是参数估计值除以 Lenth PSE 之后的结果。若相对标准误差不等，则按第 105 页的 “伪 t 比”所示计算。

伪 p 值 当不存在误差自由度时显示。p 值使用 t 分布生成。自由度为 $m/3$，舍入到最接近的整数，其中 m 是参数个数。

若选定**使用标准化成方差相等的估计值**并且有注释表明参数估计值不相关，该报表将显示名为“标准化估计值”的一列。该列提供的参数估计值来自于使估计值具有相等方差所用的变换。

若同时选定**使用标准化成方差相等的估计值**和**使用正交化成不相关的估计值**，该报表将提供名为“正交编码”的列。提供以下信息：

正交编码 提供使估计值正交化所用的变换生成的参数估计值。

正交 t 比 当存在误差自由度时显示。提供变换估计值的 t 比。

伪正交 t 比 当不存在误差自由度时显示。它是用正交化 （即通过正交编码）后的估计值除以“编码尺度 Lenth PSE”计算得出的 t 比。

“效应筛选”报表

图 3.42 显示了您通过运行 Bicycle.jmp 样本数据表中的“拟合模型”脚本创建的“效应筛选”报表。请注意，您需要选择**效应筛选 > 正态图**才能获取该格式的报表。报表标题正下方的注释指示不需要任何变换。因此，显示了 Lenth PSE。由于没有误差自由度，所以无法构造剩余误差的估计值。出于上述原因，Lenth PSE 用作剩余误差估计值以获取伪 *t* 比。为这些 *t* 比提供了伪 *p* 值。与 *p* 值小于等于 0.20 的三个估计值对应的非截距项行将突出显示。

图 3.42 针对相等方差和不相关估计值的 “效应筛选”报表

效应筛选

参数估计值的方差相等。
参数估计值彼此不相关。

Lenth PSE
0.75

参数估计值总体

项	估计值	伪 t 比	伪 p 值
截距	66.5000	88.6667	<.0001*
横把[down]	0.5000	0.6667	0.5649
发电机[off]	-6.0000	-8.0000	0.0096*
车座[down]	1.7500	2.3333	0.1266
轮胎[hard]	-1.2500	-1.6667	0.2195
齿轮[low]	-11.2500	-15.0000	0.0023*
雨披[off]	0.2500	0.3333	0.7665
快速制动[no]	0.5000	0.6667	0.5649

正交 t 检验使用了伪标准误差

正态图

针对不等方差和相关估计值的“效应筛选”报表

在 Odor.jmp 样本数据表中，运行“模型”脚本并点击**运行**。要创建图 3.43 所示的报表，点击“响应‘Y’”红色小三角并选择**效应筛选 > 正态图**。您也可以通过点击“响应‘Y’”红色小三角并选择“Bayes 图”或“Pareto 图”选项创建该报表。

报表标题正下方的注释指示需要通过变换来标准化和正交化估计值。相关性矩阵显示在“估计值的相关性”报表中。

该报表显示“*t* 检验尺度”和“编码尺度”Lenth PSE。但是，由于存在误差自由度，“参数估计值总体”报表中的检验未使用 Lenth PSE。与 *p* 值小于等于 0.20 的三个估计值对应的非截距项行将突出显示。“参数估计值总体”报表底部的注释指示正交化估计值依赖于其输入模型的顺序。

图 3.43 针对不等方差和相关估计值的 “效应筛选” 报表

效应筛选

使用标准化的方差相等估计值

使用正交化的不相关估计值

	Lenth PSE
t 检验尺度	1.7815489
编码尺度	14.253752

参数估计值总体

项	估计值	t 比	正交编码	正交 t 比	概率>|t|
截距	-4.3333	-0.2422	15.2000	1.8998	0.1159
温度	-25.8750	-2.3618	-18.8964	-2.3618	0.0646
气/液比	-23.5000	-2.1450	-17.1620	-2.1450	0.0848
高度	-10.6250	-0.9698	-7.7594	-0.9698	0.3767
温度*气/液比	-22.0000	-1.4200	-11.3608	-1.4200	0.2149
温度*高度	-5.2500	-0.3389	-2.7111	-0.3389	0.7485
气/液比*高度	-5.0000	-0.3227	-2.5820	-0.3227	0.7600
温度*温度	25.2917	1.5684	12.2138	1.5266	0.1874
气/液比*气/液比	18.5417	1.1498	9.5025	1.1877	0.2883
高度*高度	-7.2083	-0.4470	-3.5763	-0.4470	0.6736

各正交估计值受它之前的效应制约

估计值的相关性

相关性										
截距	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	-0.5547	-0.5547	-0.5547
温度	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
气/液比	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
高度	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
温度*气/液比	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
温度*高度	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000
气/液比*高度	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
温度*温度	-0.5547	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0769	0.0769
气/液比*气/液比	-0.5547	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	1.0000	0.0769
高度*高度	-0.5547	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	0.0769	1.0000

使估计值不相关的变换

“估计值的相关性” 报表

仅当估计值相关时，才会显示 “估计值的相关性” 报表（图 3.43）。该报表提供参数估计值的相关性矩阵。该矩阵类似于您通过选择 “估计值” > “估计值的相关性” 红色小三角选项获取的矩阵。不过，为提供更紧凑的表示，该报表未显示列标题。请参见第 131 页的 “估计值的相关性”。

“使估计值不相关的变换” 报表

仅当估计值相关时，“使估计值不相关的变换” 报表才显示。该报表提供用于变换设计矩阵以生成不相关参数估计值的矩阵。通过用该矩阵预先乘以原始估计值并将结果乘以尺度因子，可以获取变换的或**正交编码**的估计值。尺度因子是均方根误差 (RMSE) 和行数的函数。其定义如下：

RMSE / sqrt(行数)

可以使用 Cholesky 分解获取变换矩阵。将 X′X 表示为 LL′，其中 L 是 Cholesky 分解中的下三角矩阵。随后可通过 L′ 提供变换矩阵。

该变换按照模型项的顺序，使每个回归变量与其之前的回归变量标准化正交。变换生成包含相等对角线元素的对角协方差矩阵。编码估计值是该迭代过程的结果。

注意：正交编码估计值依赖于模型项的顺序。仅在每个效应与其之前输入的效应正交后，才会估计每个效应的贡献。考虑先输入主效应，随后是双因子交互作用，然后是三因子交互作用，依此类推。

“正态图”报表

在“正态图”报表标题下方，选择正态图或半正态图 (Daniel 1959)。两个图都基于效应稀疏原则（即：相对较少的效应是活跃的）进行预测。不活跃的那些效应表示随机噪声。它们的估计值可认定为服从均值为 0 且方差为 σ^2 的正态分布，其中 σ^2 表示剩余误差方差。由此判定：在正态概率图上，表示不活跃效应的估计值落在斜率为 σ 的线条附近。

正态图

若不需要任何变换，正态图的垂直坐标表示估计值，水平坐标表示其正态分位数。表示不活跃效应的点应沿着斜率为 σ 的线条分布。Lenth PSE 用于估计 σ，具有该斜率的蓝线显示在图中。

若已经应用正交性变换，则垂直轴表示“标准化估计值”。这些值就是在“参数估计值总体”报表中提供的“正交 *t* 比”值。（“正交 *t* 比”值是“正交编码”估计值除以“编码尺度 Lenth PSE”之后的结果。）

由于估计值通过 σ 的估计值进行了标准化，与不活跃效应对应的点应落在斜率为 1 的线条附近。斜率为 1 的红线显示在图中，同时还显示了一条斜率等于“t 检验尺度 Lenth PSE”的蓝线。

在所有情况下，基于“参数估计值总体”报表中的 *p* 值，在 0.20 水平下偏离正态性的估计值都在图中进行了标记。

半正态图

半正态图显示效应的绝对值。轴的构造与显示的线条反映了正态图所具有的那些特征。

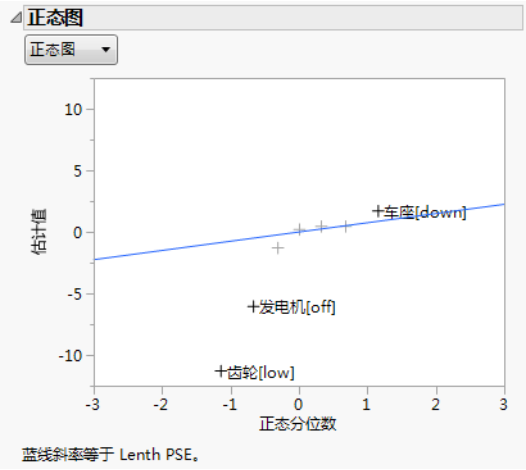
图 3.44 显示了 Bicycle.jmp 样本数据表的“正态图”报表。该模型不需要变换，所以图中显示了针对其正态分位数标绘的原始估计值。图中显示了斜率等于 Lenth PSE 的线条。该图表明**齿轮**、**发电机**和**车座**是活跃因子。

“正态图”示例

1. 选择**帮助** > **样本数据库**，然后打开 Bicycle.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 从**横把**一直选到**快速制动**，然后点击**添加**。
5. 点击**运行**。
6. 点击“响应 ‘Y’”红色小三角并选择**效应筛选** > **正态图**。

   随即显示下面的正态图。

图 3.44 正态图



“Bayes 图”报表

“Bayes 图”报表为确定活跃的效应提供了另一种方法。该报表帮助您使用 Bayesian 方法计算后验概率。根据 Box and Meyer (1986)，该方法假定估计值是具有两种分布的混合。大部分对应于不活跃效应的估计值被认定为方差为 σ^2 的纯随机正态噪声。其余估计值（即活跃效应）被认定来自**污染**分布，其方差是 σ^2 的 *k* 倍。

项 提供与参数估计值对应的模型项。

估计值 提供参数估计值。Bayes 图相对于估计标准差等于 1 的估计值构造。若估计值不相关，则使用 *t* 比。若估计值相关，则使用正交 *t* 比。

先验概率 支持您指定估计值为非零 （相当于估计值处于污染分布中）的概率。估计值的先验概率通常设置为相等值。最初显示的是一般建议的值 0.2，但之后您可以更改该值。

K 污染项 污染系数的值，表示污染分布方差与误差方差之比。*K* 一般设置为 10，这是默认值。

标准误差尺度 若标准误差的估计值存在自由度，该值将设置为 1。JMP 使用该值是因为报表中使用的估计值经过变换，尺度统一为单位方差。对于不含误差估计值的饱和模型，该值设置为 0。若指定不同值，请从 RMSE 估计值的缩放因子的角度考虑它。

误差自由度 提供误差自由度。

图 3.45 中显示了规格窗口，其中显示 Bicycle.jmp 样本数据表的“Bayes 图”的默认设置。点击该窗口中的“执行”将更新报表，以显示每项的后验概率和条形图（图 3.46）。

“Bayes 图”示例

1. 选择**帮助 > 样本数据库**，然后打开 Bicycle.jmp。
2. 选择**分析 > 拟合模型**。

3. 选择 Y 并点击 Y。
4. 从**横把**一直选到**快速制动**，然后点击**添加**。
5. 点击**运行**。
6. 点击 “响应 ‘Y’ ” 红色小三角并选择**效应筛选** > **Bayes 图**。

图 3.45 “Bayes 图”规格

7. 点击**执行**计算后验概率。

图 3.46 “Bayes 图”报表

“Bayes 图”报表中的图下方的注释提供了样本未被污染的后验概率。后验概率是基于先验和数据给出的关于不存在活跃效应的概率。概率很小 (0.0144)，指示可能存在活跃效应。“后验概率”列表明至少**发电机**和**齿轮**是活跃效应。

“Pareto 图”报表

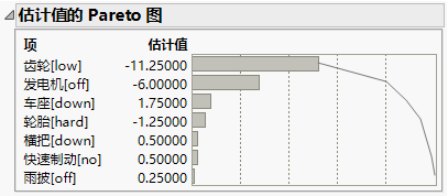
“Pareto 图”报表提供了关于估计值的绝对值的 Pareto 图。图 3.47 显示了 Bicycle.jmp 样本数据表的 “Pareto 图”报表。

- 若原始估计值具有相等方差且不相关，则标绘原始估计值。
- 若原始估计值具有不等方差且不相关，则标绘 *t* 比。
- 若原始估计值具有不等方差且相关，则标绘“正交编码”估计值。

图中的累积和线条对估计值的绝对值进行了求和。请记住：正交估计值依赖于在模型中输入项的顺序。

注意：图中显示的估计值经过标准化，具有相等方差并且变换为正交。您可以选择撤销这些变换。请参见第 135 页的“变换”。

图 3.47 Pareto 图



因子刻画

“因子刻画”菜单帮助您探索和直观演示估计的模型。您可以探索响应曲面的形状、查找最佳设置、基于指定的噪声假设模拟响应数据，以及按需变换响应。

“刻画器”、“等高线刻画器”、“混料刻画器”和“曲面刻画器”是极为通用的工具，其适用范围已不限于标准最小二乘法模型。有关这些刻画器的说明和用法的详细信息，请参见《刻画器指南》。

若模型中的响应列包含一个变量的逆变换，则刻画器和交互作用图均显示采用未变换尺度的响应。

提供以下“因子刻画”选项：

注意：若模型包含表达式或向量作为效应，则其中多数选项都不可用。

刻画器 显示每个 X 变量的预测迹线。支持您查找一个或多个响应的最佳设置，并使用模拟探索响应分布。请参见第 143 页的“刻画器指南”和《刻画器指南》。

交互作用图 当模型中包含交互作用效应时，显示交互作用图矩阵。请参见第 144 页的“交互作用图”。

等高线刻画器 提供交互式等高线刻画器，该刻画器适用于以图形方式优化响应曲面。请参见第 145 页的“等高线刻画器”和《刻画器指南》。

混料刻画器 在三元图上显示混料实验模型的响应等高线。请参见第 146 页的“混料刻画器”和《刻画器指南》。

注意：仅当对三个或更多效应应用“混料效应”特性或是对三个或更多列应用“混料”属性时，该选项才显示。

立方图 为落在立方图顶点上的因子范围的极值显示预测值。请参见第 147 页的 “立方图”。

Box Cox Y 变换 查找最能满足正态性和方差齐性假设的响应的 Box-Cox 幂变换。请参见第 148 页的 “Box Cox Y 变换”。

曲面刻画器 显示响应曲面的三维曲面图。请参见第 150 页的 “曲面刻画器”和《刻画器指南》。

刻画器指南

注意：有关刻画器的详细信息，请参见《刻画器指南》。

刻画器（或 “预测刻画器”）显示每个 X 变量的预测迹线。

图 3.48 演示了 Reactor.jmp 样本数据表的刻画器的一部分。每个 X 变量的垂直虚线显示其**当前值**或**当前设置**。使用 “刻画器”可一次更改一个变量，以便检查对预测响应的影响。

图 3.48 预测迹线演示

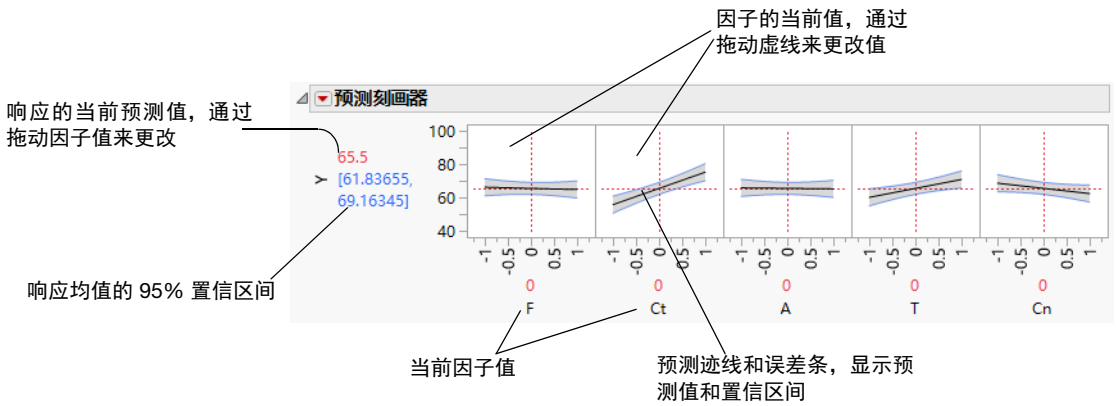


图 3.48 中的因子 F 和 Ct 是连续的。若因子是名义型，则在水平轴上显示水平。

对于每个 X 变量，因子名称上方的值是其当前值。通过在图形中点击或通过将虚线拖至您希望新的当前值所处的位置，您可以更改当前值。

- 水平虚线显示针对 X 变量的当前值，每个 Y 变量的**当前预测值**。
- 图内黑线显示当您更改单个 X 变量的当前值时预测值如何随之更改。预测值的 95% 置信区间通过环绕预测迹线的虚曲线（对于连续变量）或误差条（对于分类变量）来显示。

交互作用图

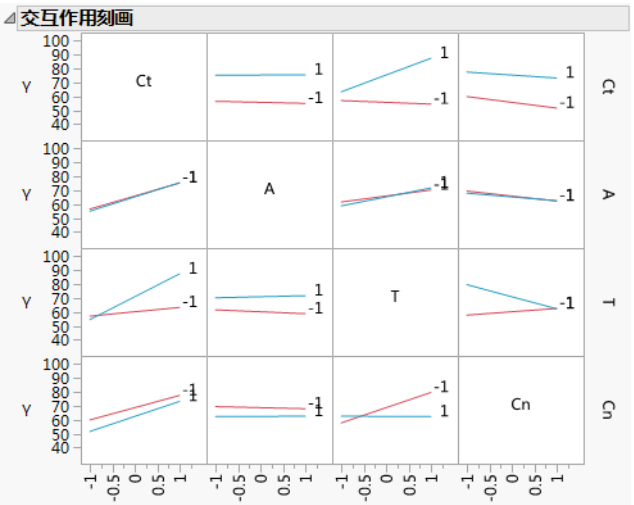
当模型中包含交互作用效应时，“交互作用图”选项显示交互作用图矩阵。矩阵的每个单元格都包含一个图，图中水平轴针对图所在的列中显示的效应进行了尺度统一。针对该效应与相应行中显示的效应的交互作用标绘了线段。所以，交互作用图显示行效应与列效应的交互作用。

为行效应的每个水平都标绘了一条线段。模型预测的响应值通过线段来连接。非平行线段为可能的交互作用提供了直观证明。不过，在推断出存在交互作用的结论之前，应对提出的交互作用的 p 值进行检查。图 3.49 提供了 Reactor.jmp 样本数据表的交互作用图矩阵。

“交互作用图”示例

1. 选择**帮助 > 样本数据库**，然后打开 Reactor.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择 Y 并点击 Y。
4. 确保**次数**框的值为 2。
5. 选择 Ct、 A、 T 和 Cn，然后点击**宏 > 析因次数**。
6. 点击**运行**。
7. 点击 “响应 ‘Y’ ” 红色小三角并选择**因子刻画 > 交互作用图**。

图 3.49 交互作用图



与 T*Cn 交互作用对应的图是底部一行图中的第三个图，或等效于最后一列图中的第三个图。这两个图均显示：在 T 的低水平上，Cn 对 Y 的影响基本保持不变，不论 Cn 是设置在高水平还是低水平。不过，在 T 的高水平上，Cn 对 Y 的影响根据其水平有所不同。Cn 为 –1 比 Cn 为 1 时的预测 Y 值要高。请注意，p 值 < 0.0001，表明该交互作用显著。

在某些设计中，双因子交互作用与其他双因子交互作用具有别名关系。若存在这种别名关系，“交互作用刻画”图中与这些交互作用对应的单元格将灰显。

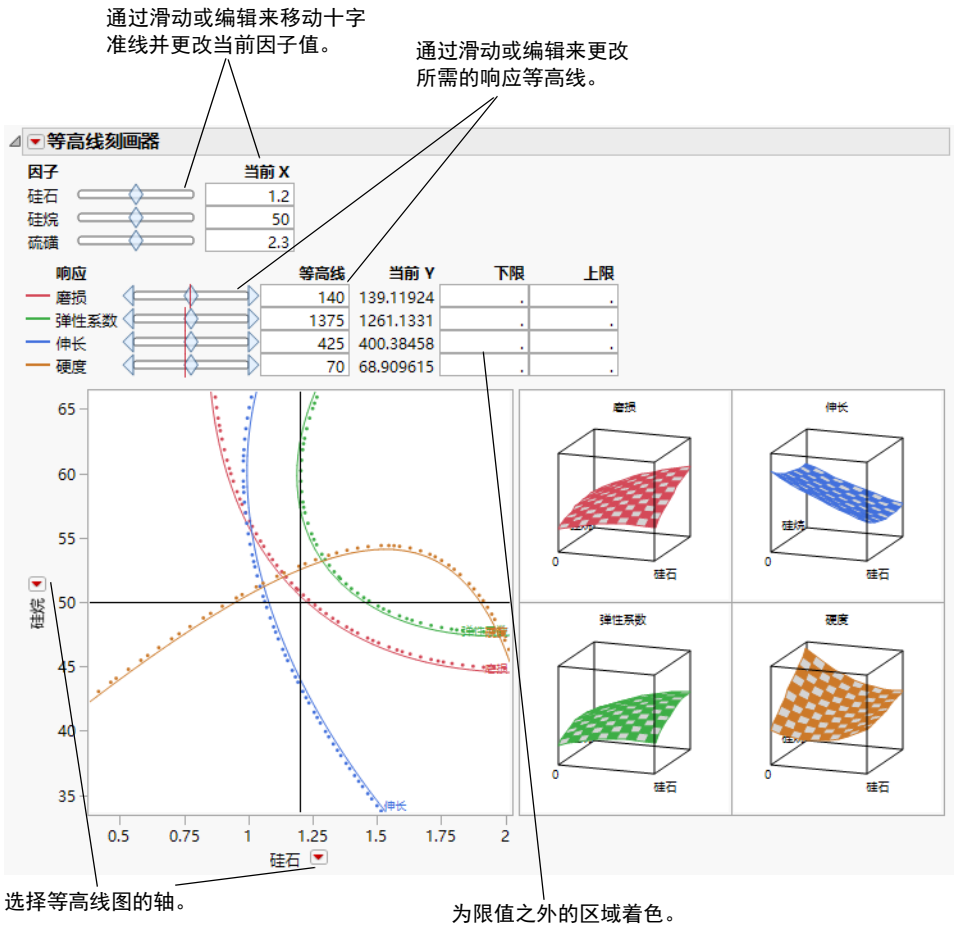
等高线刻画器

注意：有关“等高线刻画器”的详细信息，请参见《刻画器指南》。

使用交互式“等高线刻画器”以图形方式优化响应曲面。“等高线刻画器”一次显示两个因子的拟合模型的等高线。该报表还包含一个曲面图。

图 3.50 显示了 Tiretread.jmp 样本数据表的等高线刻画器视图。运行数据表脚本 **4 种响应的响应曲面设计**，然后从“最小二乘法拟合”报表菜单中选择**刻画器 > 等高线刻画器**。

图 3.50 等高线刻画器



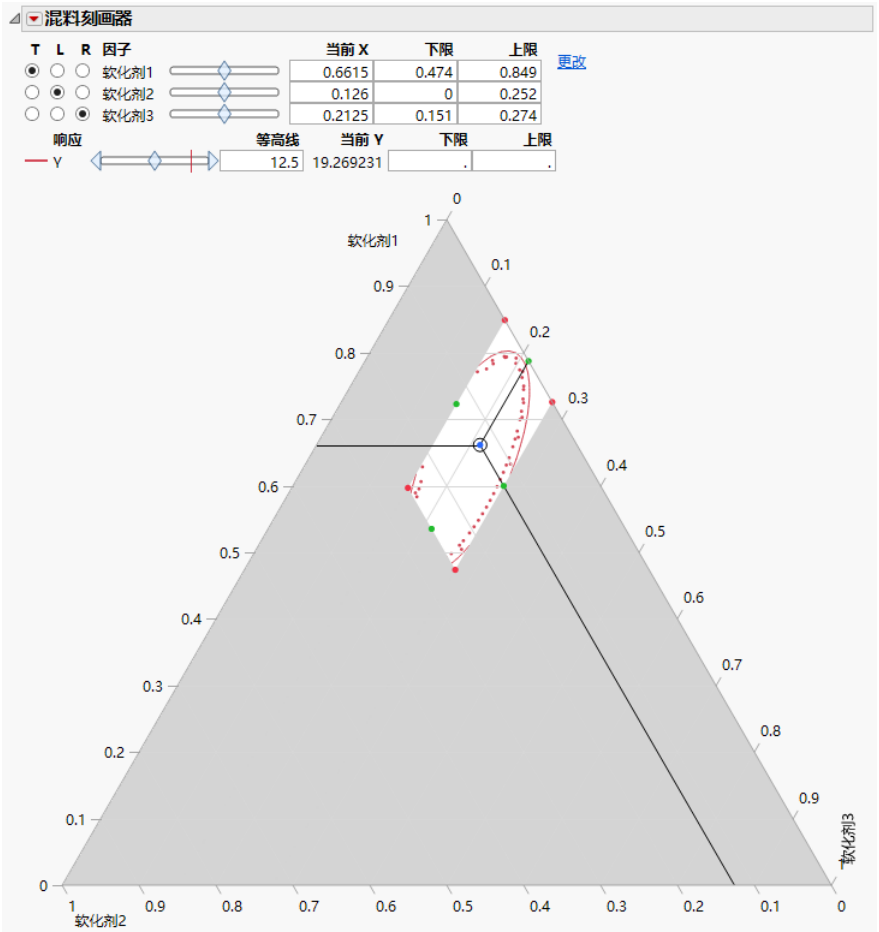
混料刻画器

注意：仅当您为效应指定**宏 > 混料响应曲面**选项时，才显示该选项。有关“混料刻画器”的详细信息，请参见《刻画器指南》。

“混料刻画器”在三元图上显示混料实验模型的响应等高线。当实验中的三个或更多因子是混料成分时可以使用“混料刻画器”。“混料刻画器”可帮助您直观演示和优化实验的响应曲面。

图 3.51 显示了 Plasticizer.jmp 样本数据表中模型的“混料刻画器”。运行**模型**数据表脚本，然后从报表的红色小三角菜单中选择**因子刻画 > 混料刻画器**。通过在图中左上角选择不同的单选按钮，可以修改因子的图轴。图右上角的“下限”列和“上限”列支持您为因子和响应输入限值。

图 3.51 混料刻画器



立方图

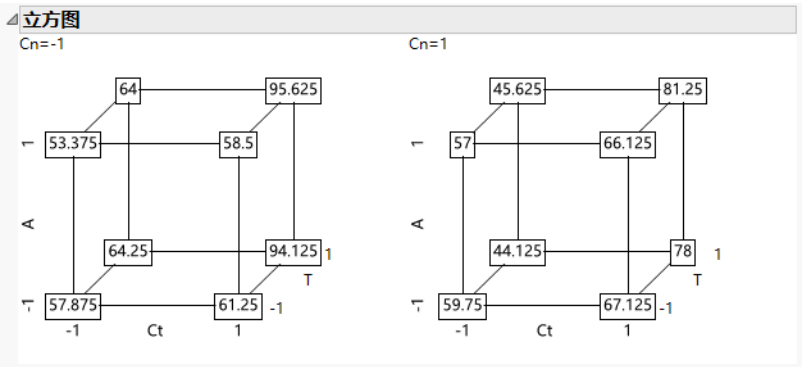
“立方图”选项显示因子范围极值的预测值。这些值显示在立方体的顶点上（图 3.52）。顶点由因子的最小和最大观测值定义。若有多重响应，多重响应在每个顶点处堆叠显示。

“立方图”示例

1. 选择**帮助** > **样本数据库**，然后打开 Reactor.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 确保**次数**框的值为 2。
5. 选择 Ct、 A、 T 和 Cn，然后点击**宏** > **析因次数**。

6. 点击**运行**。
7. 点击“响应 ‘Y’”红色小三角并选择**因子刻画 > 立方图**。

图 3.52 立方图



请注意，有一个立方体对应 Cn = –1，一个立方体对应 Cn = 1。要更改布局，以便因子映射到不同的立方体坐标，可以点击第一个立方体中的因子名称。拖动它使其覆盖所需轴的因子名称。例如，在图 3.52 中，若您点击 T 并将其拖至 Ct 上方，则 T 和 Ct（以及它们相应的坐标）会交换位置。要在单个立方体中查看 Cn 的水平，可以使其与第一个立方体中的另一个因子交换位置，只需将其拖放到该因子上方即可。

Box Cox Y 变换

Box-Cox 幂变换用于变换响应，以便能够更接近于满足通常的正态性和方差齐性回归假设。变换响应随后可以使用回归模型进行拟合。不过，您还可能出于其他原因使用 Box-Cox 幂变换来变换变量。仅当响应 Y 为严格正数时，这种变换才适用。

常用的变换是将响应升高到特定次幂。 Box and Cox (1964) 对这一系列的幂变换进行了规范化和说明。构造该变换的公式以便按照参数 λ 提供连续定义，这样即可比较误差平方和。具体而言，以下方程提供了一系列的变换：

$$Y_\lambda = \begin{cases} \dfrac{y^\lambda - 1}{\lambda \dot{y}^{\lambda-1}} & \text{若 } \lambda \neq 0 \\ \dot{y}\ln(y) & \text{若 } \lambda = 0 \end{cases}$$

在此，$\dot{y}$ 表示几何均值。

“Box-Cox Y 变换”选项对从 λ = –2 到 2（增量为 0.2）的变换进行拟合。要选择 λ 的某个值，将为这些变换中的每一个变换计算似然函数。计算的假设前提是：误差是独立的，服从均值为 0 且方差为 σ^2 的正态分布。将选择使似然函数最大化的 λ 值。该值还使误差平方和在 λ 值中最小。使用二次插值找到围绕具有最小误差平方和的网格点的两个增量网格点之间使误差平方和

最小化的 λ 值。若该插值导致误差平方和的值为负，则使误差平方和最小化的网格值 λ 将报告为最佳 λ。

“Box-Cox 变换”报表显示一个图，其中显示 λ 的值与误差平方和 (SSE) 值的对应关系。该图上的水平红线表示 λ 的单侧 95% 置信区间。该置信区间基于 Box and Cox (1964, p. 216) 中定义的置信区域。置信区域用以下不等式定义：

$SSE(\lambda) < SSE(\lambda_{best}) * \exp(ChiSquareQuantile(0.95,1) / dfe)$

其中

$SSE(\lambda_{best})$ 是使用报告的最佳 λ 计算的误差平方和

ChiSquareQuantile(0.95,1) 是自由度为 1 的 χ^2 分布的第 0.95 分位数

dfe 是回归模型的 “方差分析”表中的误差自由度

“Box-Cox 变换”报表提供以下选项：

使用变换重新拟合 支持您指定 lambda 值以定义变换的 Y 变量，之后提供变换后的变量的最小二乘拟合。

使用变换替换 支持您指定 lambda 值以定义变换的 Y 变量，之后使用变换后变量的拟合替换当前的最小二乘拟合。若有多重响应，“使用变换替换”仅替换您所变换的响应的报表。

保存最佳变换 在数据表中创建一个新列并保存最佳变换的公式。

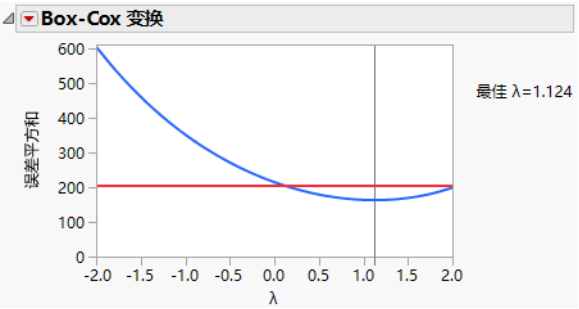
保存特定变换 支持您指定 lambda 值，并使用您指定的变换的公式在数据表中创建列。

估计值表 创建一个新的数据表，其中包含从 –2 到 2 （以 0.2 为增量）的所有 λ 的参数估计值和误差平方和值。

“Box-Cox Y 变换”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Reactor.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 确保**次数**框的值为 2。
5. 选择 F、 Ct、 A、 T 和 Cn，然后点击**宏** > **析因次数**。
6. 点击**运行**。
7. 点击 “响应 Y”红色小三角并选择**因子刻画** > **Box-Cox Y 变换**。

图 3.53 Box Cox Y 变换



该图显示 λ 的最佳值介于 0.1 和 2.0 之间。JMP 使用两个最佳值之间的插值（采用 λ 值的 0.2 单位网格）选择的值为 1.124。

8. （可选）要查看用于构造图形的误差平方和值，请选择**估计值表**。

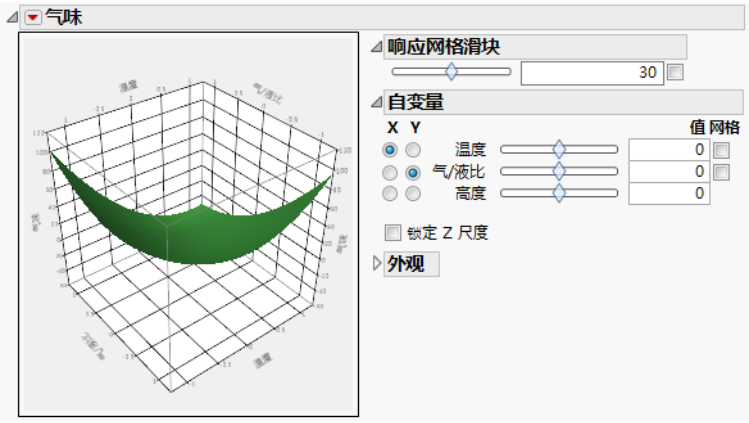
曲面刻画器

注意：有关“曲面刻画器”的详细信息，请参见《刻画器指南》。

“曲面刻画器”显示响应曲面的三维曲面图。

图 3.54 显示了 Odor.jmp 样本数据表中模型的“曲面刻画器”。运行**模型**数据表脚本，然后从报表的红色小三角菜单中选择**因子刻画 > 曲面刻画器**。您可以使用“自变量”下的单选按钮更改轴上的变量。此外，您可以通过点击“外观”下的“实际值”来标绘点。

图 3.54 曲面图



行诊断

行诊断菜单处理特定于行或观测的问题。

标绘回归图 显示 “回归图” 报表，其中显示分类效应每个水平的数据和回归线的散点图。

仅当模型中包含恰好一个连续效应且分类效应不超过一个时，该选项才显示。在这种情况下，默认提供 “回归图” 报表。

标绘 “预测值 - 实际值” 图 显示 “预测值 - 实际值” 图，其中针对 Y 的预测值标绘 Y 的观测值。该图是整体模型的杠杆图。请参见第 152 页的 “杠杆图”。

注意：若在 “拟合模型” 启动窗口中将 “效应杠杆率” 或 “效应筛选” 选作 “重点” 并且 R 方值小于 0.999，此时默认显示 “预测值 - 实际值” 图。

标绘效应杠杆图 显示模型中每个效应的 “杠杆图” 报表。该图显示观测如何影响该效应的检验，并提供对多重共线性的领悟。请参见第 152 页的 “杠杆图”。

注意：若在 “拟合模型” 启动窗口中将 “效应杠杆率” 选作 “重点” 并且 R 方值小于 0.999，此时默认显示 “效应杠杆图”。它们显示在 “整体模型” 报表的右侧。若选择另一个 “重点”，“效应杠杆图” 会显示在 “效应详细信息” 报表中。在所有情况下，必须选择 “回归报表” > “效应详细信息” 选项，“效应杠杆图” 才能显示。

标绘 “预测值 - 残差” 图 显示 “预测值 - 残差图” 报表。该图显示针对 Y 预测值标绘的残差。您通常希望残差值在零附近随机分散。

注意：若在 “拟合模型” 启动窗口中将 “效应杠杆率” 或 “效应筛选” 选作 “重点” 并且 R 方值小于 0.999，此时默认显示 “预测值 - 残差” 图。

标绘 “行号 - 残差” 图 显示 “行号 - 残差图” 报表。针对行号标绘残差值。该图可帮助您检测观测值相对行顺序变化的模式。

标绘学生化残差 显示 “学生化残差” 图。图上每个点都使用其标准差的估计值计算得出 （获取估计值时删除了当前观测）。这些残差亦称 RStudent 或**外部学生化**残差。

该图包含两组限值：

- 显示为红色的外部限值为 95% Bonferroni 限值。这些限值位于 ± tQuantile(0.025/*n*, *n*–*p*–1) 的位置，其中 *n* 是观测数，*p* 是预测变量数。
- 图上显示为绿色的内部限值为 95% 单值 *t* 分布限值。这些限值位于 ± tQuantile(0.025, *n*–*p*–1) 的位置，其中 *n* 是观测数，*p* 是预测变量数。

落在红色限值之外的点应视为可能的离群值。落在绿色限值之外但在红色限值之内的点应视为可能的离群值，但更不太确定一些。这些限值的 95% 置信水平不受您在 “模型规格” 窗口的 “设置 Alpha 水平” 选项中的选择影响。

警告：使用 “保存列” > “学生化残差”保存的残差**不是**外部学生化残差。

注意：若模型包含随机效应并且 REML 是启动窗口中的指定方法，则学生化残差图不包含限值，而且标绘的各点**不是**外部学生化残差。

标绘 “正态分位数 - 残差”图 （当 REML 是启动窗口中指定的 “方法”时不可用。）显示残差正态分位数图。针对正态分布的分位数标绘这些残差值。该图可帮助您评估残差正态性的假设是否正确。

Press 显示提供 Press 统计量及其均方根误差 (RMSE) 的 Press 报表。Press 统计量对于比较多个模型很有用。包含较低 Press 统计量的模型为优良模型。（请参见第 155 页的 “Press”。）

Durbin-Watson 检验 （指定 “频数”列时不可用）。显示 Durbin-Watson 报表，其中提供统计量来检验残差是否具有一阶自相关。该报表还显示残差的自相关以及 “概率 <DW”，这是与统计量相关的精确概率。该选项仅适用于时间序列数据，并假定您的观测按时间排序。

杠杆图

针对 X 的效应杠杆图在以下方面很有用：

- 您可以看到哪些点可能对 X 的假设检验施加影响。
- 您可以发现异常模式和违反模型假设情况。
- 您可以发现多重共线性问题。

构造

效应的杠杆图显示在模型中已存在其他效应的情况下向模型添加该效应的影响。为了方便演示，考虑为单个连续效应 X 构造效应杠杆图。有关在更一般情况下水平轴的尺度的信息，请参见第 153 页的 “水平轴尺度”。

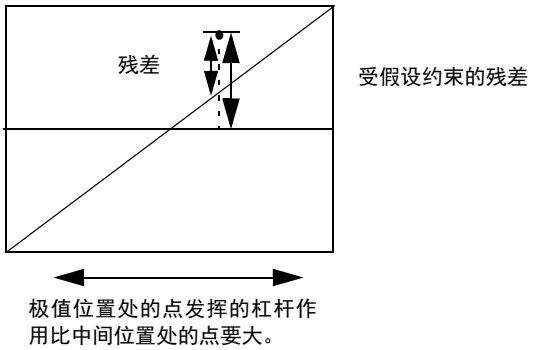
响应 Y 对 X 以外的所有预测变量上进行回归，并获取残差。将这些残差称为 Y 残差。然后，X 对模型中其他所有预测变量上进行回归，并计算残差。将这些残差称为 X 残差。X 残差可能包含 Y 残差不包含的信息，Y 残差是在模型不含 X 的情况下获得的。

X 的效应杠杆图实际上是一个 X 残差对 Y 残差的散点图（图 3.57）。为了帮助解释和比较您可能构造的其他图，JMP 向 Y 残差添加了 Y 均值，向 X 残差添加了 X 均值。转换的 Y 残差称为 Y 杠杆率残差，转换的 X 残差称为 X 杠杆率值。效应杠杆图上的点是这些 X 杠杆率与 Y 杠杆率残差配对而成的点。

JMP 拟合这些点的最小二乘线并拟合均值置信带；拟合线为红色实线，置信带着红色。最小二乘线的斜率恰好是模型中 X 的系数估计值；在该模型中，Y 对 X 和其他预测变量上进行回归。蓝色水平虚线设置在 “Y 杠杆率残差”的均值位置。这条线描述 X 残差与 Y 残差不线性相关的情形。若拟合线具有非零斜率，则向模型添加 X 可帮助解释变异。

图 3.55 显示如何在杠杆图中描述残差。从某个点到拟合线的距离是包含该效应的模型的残差。从该点到水平线的距离是模型中不含该效应时的剩余误差。换言之，杠杆图中的均值线表示参数（效应）的假设值被限制为 0 时的模型。

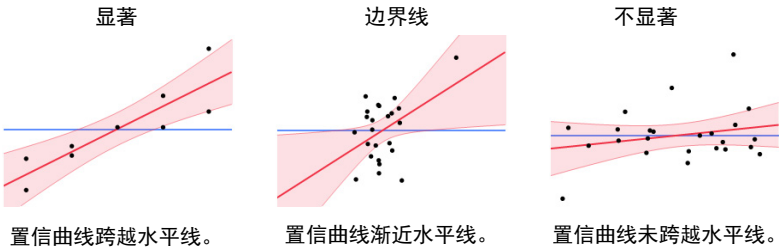
图 3.55 一般杠杆图的演示



置信曲线

拟合线的置信曲线显示在杠杆图中。这些曲线直观地表明所关注的检验在 5% 水平下（或是您在“拟合模型”启动窗口中指定的“设置 Alpha 水平”下）是否显著。若曲线之间的置信区域包含表示假设的水平线，则效应不显著。若曲线跨越水平线，则效应显著。请参见图 3.56 中的示例。

图 3.56 比较杠杆图中显示的显著性



水平轴尺度

若预测变量 X 的建模类型为连续，则水平轴将根据 X 的单位来统一尺度。水平轴范围反映 X 值的范围。杠杆图中拟合线的斜率是 X 的参数估计值。请参见图 3.57 中的左图。

若效应为名义型或有序型，或效应是交互作用之类的复杂效应，则水平轴无法直接表示效应值。在这种情况下，水平轴将按照响应的单位统一尺度，且拟合线是斜率为 1 的对角线。“整体模型”杠杆图（其中所关注的假设为“所有参数值均为 0”）使用该尺度。（请参见第 177

页的“杠杆图详细信息”。）对于该图，水平轴按照整体模型的预测响应值来统一尺度，如图 3.57 中的右图所示。

简单回归中线性效应的杠杆图与“预测变量-实际响应值”的传统图相同。

杠杆率

之所以使用**杠杆率**一词，是因为这些图可帮助您直观演示在模型中添加效应后各点对检验的影响。在水平方向距离图中心较远的点与接近中心的点相比，对效应检验发挥的影响更大。回想一下，效应检验涉及比较模型在包含和不含该效应时的残差平方和。在极值处，受假设约束之前和之后的残差的差值相对较大。因此，这些残差往往对该效应的假设检验的平方和有更大的贡献。

多重共线性

多重共线性是指这样一种状况：两个或更多预测变量高度相关，或从技术上更准确地说，涉及接近线性相关的关系。存在多重共线性时，标准误差可能会膨胀，参数估计值可能不稳定。若某个效应与其他预测变量共线，则点的水平值往往朝着图中央聚拢。这种情形指示拟合线的斜率不稳定。

整体模型“预测值 - 实际值”图

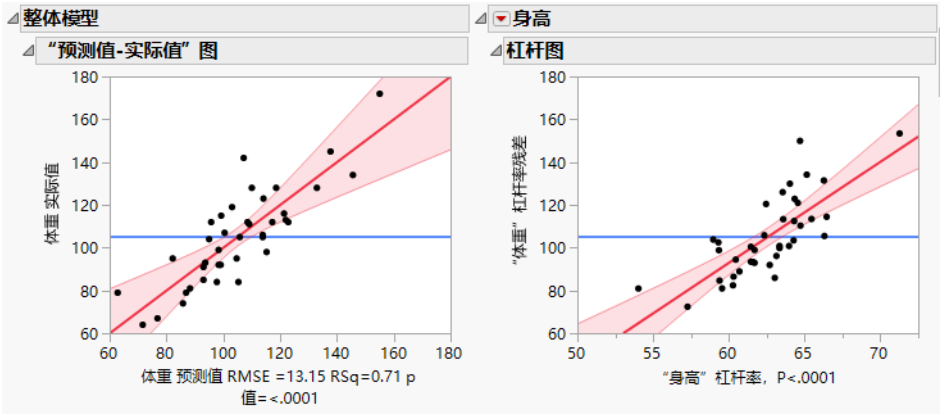
“标绘效应杠杆图”选项为模型中的每个效应生成一个杠杆图。此外，“预测值-实际值”图也可被视为杠杆图。该图支持您直观演示关于模型中的所有参数（截距除外）均为 0 的检验。在“方差分析”报表中也会通过分析方法执行相同的检验。（有关该图的详细信息，请参见第 177 页的“杠杆图详细信息”。）

线性效应的杠杆图示例

1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**体重**并点击 Y。
4. 选择**身高**、**年龄**和**性别**，然后点击**添加**。
5. 点击**运行**。

整体模型“预测值-实际值”图以及针对身高的效应“杠杆图”显示在图 3.57 中。左侧的整体模型图针对所有效应进行检验。您可以推断出该模型显著，因为置信曲线跨越响应**体重**均值处的水平线。右侧的**身高**“杠杆图”也显示**身高**是显著的，即便是模型中包含**年龄**和**性别**也是如此。这两个图均未表明需要担心与影响点或多重共线性相关的问题。

图 3.57 整体模型和效应杠杆图



Press

Press（即**预测误差平方和**）统计量是使用留一法交叉验证计算的预测误差估计值。在留一法交叉验证中，轮流删除每个观测。考虑一个特定的观测。在保留那个观测的情况下拟合模型，然后为该观测获取预测值。计算该观测的残差。该过程应用到所有观测，并且对残差进行平方再求和计算来给出 Press 值。

具体而言，按以下方式给出 Press 统计量：

$$Press = \sum_{i=1}^{n} (\hat{y}_{(i)} - y_i)^2$$

其中，n 是观测数，y_i 是第 i 个观测的观测响应值，$\hat{y}_{(i)}$ 是第 i 个观测的预测响应值。这些值基于不含该观测的模型拟合。

Press RMSE 定义为 $\sqrt{Press/n}$。

Press R 方定义为 1 - Press/SS$_{总和}$。

保存列

每个“保存列”选项都向当前数据表添加一个或多个新列。当拟合方法为“REML”时还会显示其他“保存列”选项。第 169 页的“REML“保存列”选项”中对此进行了详细说明。

请注意以下事项：

- 创建公式时，这些公式将作为“公式”列属性输入。
- 对于许多新列，将添加“注释”列属性，用于描述该列并指示“拟合最小二乘法”创建了该列。

- 对于“预测公式”和“预测值”选项，将添加“预测”列属性。在 JMP 内部执行模型比较（分析 > 预测建模 > 模型比较）时会使用该属性。拟合许多模型时，该属性也会对您有所帮助，因为它记录列的来源。

提供以下 “保存列”选项：

预测公式 创建名为**预测公式 : < 列名 >** 的新列，该列包含公式和预测值。将添加 “预测”列属性，指出预测的源。

注意：预测公式 : < 列名 > 继承 **< 列名 >** 的特定属性。这些包括 “响应限”、“规格限”和“控制限”。若在保存**预测公式 : < 列名 >** 后更改 **< 列名 >** 的这些属性，这些属性不会在**预测公式 : < 列名 >** 中更新。

请参见第 158 页的 “预测公式”。

预测值 创建名为**预测值 : < 列名 >** 的新列，该列包含由指定模型计算的预测值。将同时添加“注释”和 “预测”列属性，指出预测的源。

注意：预测值 : < 列名 > 继承 **< 列名 >** 的特定属性。这些包括 “响应限”、“规格限”和“控制限”。若在保存**预测值 : < 列名 >** 后更改 **< 列名 >** 的这些属性，这些属性不会在**预测值 : < 列名 >** 中更新。

残差 创建名为**残差 : < 列名 >** 的新列，该列包含观测响应值减去其预测值所得的值。

均值置信区间 创建名为 **95% 均值下限 : < 列名 >** 和 **95% 均值上限 : < 列名 >** 的两个新列。这些列包含响应均值的 95% 置信下限和上限。

注意：若在选择该选项时按住 Shift 键，系统会提示您为计算输入一个 α 水平。

单值置信区间 创建名为 **95% 单值下限 : < 列名 >** 和 **95% 单值上限 : < 列名 >** 的两个新列。这些列包含单个响应值的 95% 置信下限和上限。

注意：若在选择该选项时按住 Shift 键，系统会提示您为计算输入一个 α 水平。

学生化残差 创建名为**学生化残差 : < 列名 >** 的新列，该列包含残差除以其标准误差所得的值。

外部学生化残差 （当拟合方法为 REML 时不可用。）创建名为**外部学生化残差 : < 列名 >** 的新列，该列包含残差除以排除当前行的标准误差估计值的结果。请参见第 151 页的 “标绘学生化残差”。

帽子值 创建名为**帽子值: <列名>**的新列。列值为矩阵 $X(X'X)^{-1}X'$ 的对角线值，有时称为帽子值。

预测值标准误差 创建名为**预测值标准误差: <列名>**的新列，该列包含预测响应均值的标准误差。

残差标准误差 创建名为**残差标准误差 : < 列名 >** 的新列，该列包含残差值的标准误差。

单值标准误差 创建名为**单值标准误差: <列名>**的新列，该列包含单个预测响应值的标准误差。

效应杠杆率对 创建一组新列，其中包含每个杠杆图的 X 杠杆率值和 Y 杠杆率残差。对于模型中的每个效应，都添加两列。若响应列名为 R，效应为 X，则新列名为：

- X 杠杆率: X-R
- Y 杠杆率: X-R

在列面板中，这些列组织到名为**杠杆率**的列组中。

Cook 距离影响 创建名为 Cook 距离影响 : < 列名 > 的新列，该列包含 Cook 距离影响统计量的值。

标准误差预测公式 创建名为**预测值标准误差** : < 列名 > 的新列，该列包含预测值标准误差的公式和值。

注意：保存的公式可能很大。若您不需要公式，请使用 “预测值标准误差”选项。

均值置信限公式 创建名为 95% 均值下限 : < 列名 > 和 95% 均值上限 : < 列名 > 的两个新列。这些列包含响应均值的 95% 置信下限和上限的公式和值。

注意：若在选择该选项时按住 Shift 键，系统会提示您为计算输入一个 α 水平。

单值置信限公式 创建名为 95% 单值下限 : < 列名 > 和 95% 单值上限 : < 列名 > 的两个新列。这些列包含单个响应值的 95% 置信下限和上限的公式和值。

注意：若在选择该选项时按住 Shift 键，系统会提示您为计算输入一个 α 水平。

保存编码表 创建一个新的数据表，其中第一列显示所有模型参数的 JMP 编码。最后一列提供响应的值。若输入了多个响应列，则所有这些列都显示为编码表中的最后几列。

注意：编码数据表包含一个称为 “原始数据”的表变量，提供用于分析的数据表的名称。在已指定 “依据”变量的情况下，“原始数据”表变量提供 “依据”变量及其水平。

JMP PRO **发布预测公式** 创建预测公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

JMP PRO **发布标准误差公式** 创建标准误差公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

JMP PRO **发布均值置信限公式** 创建响应均值的置信限公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

JMP PRO **发布单值置信限公式** 创建单个响应值的置信限公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

预测公式

预测公式: <列名>与**预测值: <列名>**的不同之处在于它包含预测公式。右击**预测公式: <列名>**列标题，然后选择**公式**可查看预测公式。若模型很大，预测公式可能需要相当大的空间。若您不需要在预测值列中包含公式，可以使用**保存列 > 预测值**选项。有关公式的信息，请参见《使用 JMP》。

“预测公式”选项可用于预测新行中的值或与刻画器一同使用。刻画器在“拟合最小二乘法”报表（因子刻画）中提供。不过，若您的数据表包括公式列，您还可以使用**图形**菜单中提供的刻画器。当您分析多重响应时，访问**图形**菜单中的刻画器会很有帮助。

注意：若选择**图形 > 刻画器**访问刻画器，首先使用“预测公式”和“标准误差预测公式”将公式列保存到数据表中。然后，将这些公式放入“刻画器”窗口中的“Y，预测公式”角色。点击**确定**后，指定您是否想要使用“预测值标准误差: <**列名**>”为“预测公式: <**列名**>”构造置信区间。若不这样操作，JMP 将为“预测值标准误差: <**列名**>”创建单独的刻画器图。

“效应汇总”报表

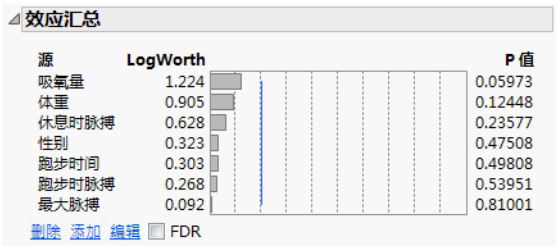
“效应汇总”选项显示一个交互式报表。它提供一个图，包含模型效应的 LogWorth（或 FDR LogWorth）值。该报表还提供一些控件，支持您在模型中添加或删除效应。模型拟合报表会基于在“效应汇总”报表中所做的更改自动更新。

“效应汇总”报表在以下特质中提供：

- 标准最小二乘法
- 名义型 Logistic
- 有序型 Logistic
- 比例风险
- 参数生存
- 广义线性模型

图 3.58 显示了 Fitness.jmp 数据表的“效应汇总”报表的初始视图。标记为 FDR 的复选框控制显示在汇总表中的列。

图 3.58 “效应汇总”报表



“效应汇总”表列

“效应汇总”表包含以下列：

源 列出按 *p* 值升序排序的模型效应。

LogWorth 显示每个模型效应的 LogWorth，该值定义为 $-\log_{10}(p \text{ 值})$。该变换调整 *p* 值以提供适用于绘图的尺度。超过 2 的值在 0.01 水平下显著 （因为 $-\log_{10}(0.01) = 2$）。

FDR LogWorth 显示每个模型效应的假发现率 LogWorth，该值定义为 $-\log_{10}(\text{FDR P 值})$。这是用于标绘和评估显著性的最佳统计量。不过，该统计量高度依赖于显著性排序，对于正相关检验很保守，并且不提供 alpha 水平下实验方面的保护。选中 FDR 复选框可将 LogWorth 列替换为 FDR LogWorth 列。

条形图 显示 LogWorth（或 FDR LogWorth）值的条形图。该图在整数值处标绘垂直虚线，在值 2 处标绘一条蓝色参考线。

p 值 显示每个模型效应的 *p* 值。这通常是模型报表的 “效应检验”表或 “效应似然比检验”表中显示的显著性检验所对应的 *p* 值。

FDR p 值 显示使用 Benjamini-Hochberg 方法计算的每个模型效应的假发现率 *p* 值。该方法调整 *p* 值以控制多重检验的假发现率。选中 FDR 复选框可将 *p* **值**列替换为 FDR *p* **值**列。

有关 FDR 校正的详细信息，请参见 Benjamini and Hochberg (1995)。有关假发现率的详细信息，请参见 《预测和专业建模》或 Westfall et al.(2011)。

“效应遗传”列 标识构成更显著的高阶效应的低阶效应。低阶效应使用补注符号来标识。请参见第 161 页的 “效应遗传”。

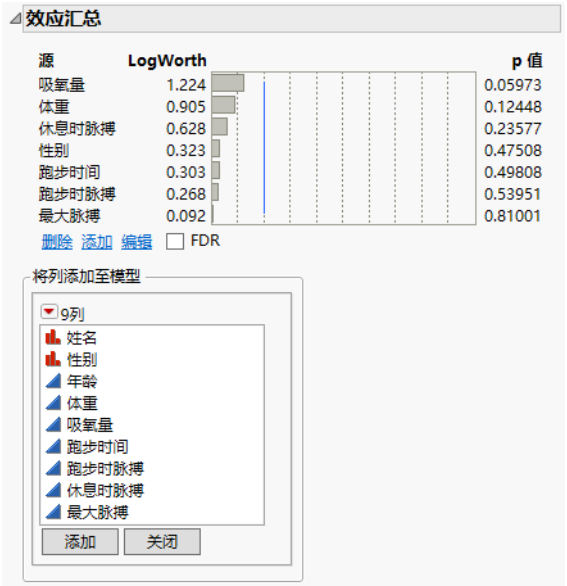
“效应汇总”表选项

汇总表下方的选项支持您添加和删除效应：

删除 从模型中删除选定的效应。要删除一个或多个效应，请选择与效应对应的行并点击**删除**按钮。

添加 打开一个面板，其中包含数据表中所有列的列表。选择您要添加到模型中的列，然后点击列选择列表下方的**添加**，将这些列添加到模型中。点击**关闭**以关闭该面板。图 3.59 显示了“添加列”面板。

图 3.59 “效应汇总”的“添加列”面板



编辑 打开“编辑模型”面板，其中包含“选择列”列表和“效应”规格面板。“效应”面板类似于“拟合模型”启动窗口中的“构造模型效应”面板。“编辑模型”面板支持您添加单个、交叉、嵌套和变换的效应。您还可以使用“宏”菜单添加多重效应。有关如何使用“添加”、“交叉”、“嵌套”、“宏”和“变换”构造效应的详细信息，请参见第 37 页的“构造模型效应”。

“效应”面板右侧提供以下选项：

提交 将您的更新应用到模型中。

关闭 关闭面板而不对模型进行任何更改。

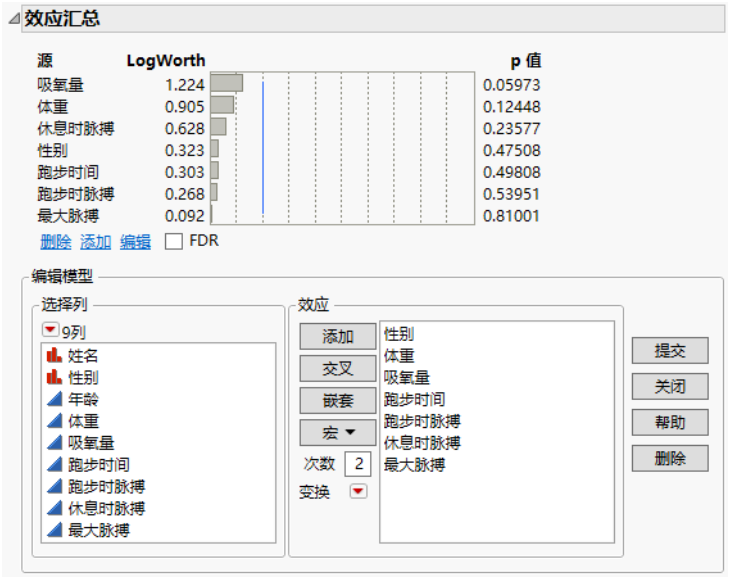
删除 从“效应”列表中删除一个或多个选定效应。

图 3.60 显示了“编辑模型”面板。

提示：“编辑”按钮支持您最大限度地控制对模型的更新。它包含“删除”和“添加”按钮的功能，并支持您构造效应以添加到模型中。

撤销 支持您撤销对模型中效应的更改。

图 3.60 “效应汇总”的“编辑模型”面板



效应遗传

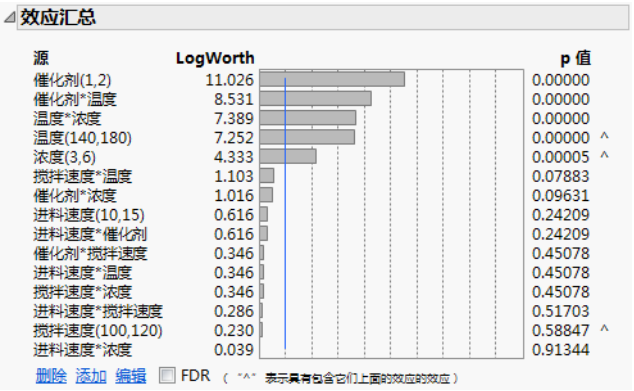
若模型包含显著的高阶效应，您可能想保留其部分或全部低阶成分，尽管这些成分并不显著。**强效应遗传**原则表明：若模型中包含高阶效应，其全部低阶成分也都应该包含在模型中。**弱效应遗传**原则表明应包含一连串的成分。

若某个高阶效应的低阶成分显示在“效应汇总”表中该高阶效应之下，则最右侧的列中会显示一个补注符号。该补注符号指示包含高阶效应比低阶效应更显著。若包含某个低阶效应的所有高阶效应都没有该低阶效应显著，则该低阶效应所在的行中不会出现补注符号。

若删除标有补注符号的效应，您可以选择两种方法之一来删除效应。选择**删除所有选定效应**删除所有选定效应，包括标有补注符号的效应。选择**仅删除未包含的效应**仅删除选定的效应，这些效应没有更高阶的效应留在模型当中。

图 3.61 显示了“效应汇总”表的示例，其中有三个低阶效应显示在包含这些低阶效应的高阶效应的下方。例如，**搅拌速度(100,120)** 显示在**搅拌速度*温度**下方。

图 3.61 针对 Reactor 32 Runs.jmp 的含效应遗传的 “效应汇总”表



多重响应

对于多重响应的情况，每个效应显示在每个响应模型中，但仅显示一个“效应汇总”报表。对于每个效应，该表显示该效应所有 p 值中的最小 p 值。添加或删除效应将应用于所有响应的模型。

混合和随机效应模型报表和选项

可以在 “拟合模型”启动窗口中指定混合和随机效应模型。“标准最小二乘法”特质拟合方差分量协方差结构。提供了两种用于拟合此类模型的方法：REML 和 EMS。

注意： JMP PRO 建议 JMP Pro 用户使用“拟合模型”窗口中的“混合模型”特质。“混合模型”特质提供比“标准最小二乘法”更宽泛的协方差结构。

- “混合模型和随机效应模型”
- “限制最大似然 (REML) 方法”
- “EMS（传统）模型拟合报表”

混合模型和随机效应模型

随机效应模型是所有因子都表示随机效应的模型。（请参见第 163 页的“随机效应”。）此类模型亦称**方差分量模型**。随机效应模型往往是分层模型。同时包含固定和随机效应的模型称为**混合模型**。重复测量和裂区模型是混合模型的特例。通常使用**混合模型**一词来包含随机效应模型。

要拟合混合模型，您必须在“拟合模型”启动窗口中指定随机效应。不过，若所有模型效应都是随机的，您还可以在“变异性 / 计数量具图”平台中拟合模型。只能在该平台中拟合特定模型。请注意，在“变异性 / 计数量具图”平台中使用的拟合方法不允许方差分量估计值为负。有关“变异性/计数量具图”平台如何拟合方差分量模型的详细信息，请参见《质量和过程方法》。

随机效应

随机效应是其水平被视为某个总体中的随机样本的因子。通常，随机效应的精确水平并不受关注，而水平反映出的变异才受关注（**方差分量**）。不过，也有些时候您想要预测随机效应给定水平的响应。更严格地来讲，随机效应被视为服从均值为零且方差非零的正态分布。

假定您关注的是两种特定的烤炉对模具收缩的影响是否不同。一个烤炉一次仅处理一批 50 个模具。您设计这样一个研究方案：先后分别为每一个烤炉随机选三个批次（每批 50 个模具）放进烤炉。这些批次的模具处理完毕后，对每批中随机选择的五个部件测量收缩。

请注意，“批次”是包含六个水平的因子，每批次对应一个水平。所以，在您的模型中包括两个因子 —“烤炉”和“批次”。由于您特别关注比较每个烤炉对于收缩的影响，所以“烤炉”是一个固定效应。但您并不关注这六个特定批次对于收缩均值的影响。这些批次代表了一个都有可能被这个实验选上的批次总体，而且其分析结果必须推广到的整个批次总体。“批次”被视为随机效应。在该实验中，“批次”因子因为在所有可能批次中的收缩变异而受到关注。您想要估计它所解释的收缩变异量。（请注意，“批次”还嵌套在“烤炉”中，因为在一个烤炉中一次只能处理一个批次。）

现在假定您关注的是在两种不同喂养制度下喂养出来的母鸡的蛋重。为十只母鸡随机分配喂养制度：为五只分配了喂养制度 A，另五只分配了喂养制度 B。不过，在本研究方案的设计中并未考虑这十只母鸡存在的一些基因上的差异。在这种情况下，您关注的是某些特定母鸡的预测蛋重以及母鸡蛋重的方差。

经典线性混合模型

JMP 拟合经典线性混合效应模型：

$$
\begin{aligned}
Y &= X\beta + Z\gamma + \varepsilon \\
\gamma &\sim N(0, G) \\
\varepsilon &\sim N(0, \sigma^2 I_n)
\end{aligned}
$$

在此，

- Y 是响应的 $n \times 1$ 向量
- $\mathbf{X}$ 是固定效应的 $n \times p$ 设计矩阵
- β 是设计矩阵 X 的未知固定效应的 $p \times 1$ 向量
- $\mathbf{Z}$ 是随机效应的 $n \times s$ 设计矩阵
- γ 是设计矩阵 Z 的未知随机效应的 $s \times 1$ 向量
- ε 是未知随机误差的 $n \times 1$ 向量
- $\mathbf{G}$ 是 $s \times s$ 对角矩阵（针对随机效应的每个水平包含相同的条目）
- I_n 是 $n \times n$ 单位矩阵
- γ 和 ε 不相关

G 的对角线元素以及 σ^2 称为**方差分量**。这些方差分量以及固定效应的向量 β 和随机效应的向量 γ 都是必须要估计的模型参数。

该模型的协方差结构有时称为**方差分量**结构 (SAS Institute Inc. 2018d, ch.83)。这种协方差结构仅在“标准最小二乘法”特质中提供。

JMP PRO “混合模型”特质拟合众多协方差结构，包括残差、一阶自回归（或 AR(1)）、非结构化和空间。请参见“混合模型”一章中第 316 页的““重复结构”选项卡”。

用于拟合包含随机效应的模型的 REML 与 EMS

JMP 提供了两种方法来拟合包含随机效应的模型：

- REML 表示**限制最大似然**（始终为推荐方法）
- EMS 表示**期望均方**（仅适用于旧版教科书的教学）

REML 方法现已取代传统 EMS 方法成为主流的拟合方法。与 EMS 方法相比，REML 在适用性方面要广泛得多。REML 方法最早是由 Patterson and Thompson (1974) 提出的。另见 Wolfinger et al. (1994) 和 Searle et al. (1992)。

EMS 方法，亦称**矩量法**，是在功能强大的计算机面世之前开发出来的。研究人员将自己限定在平衡环境中，并使用 EMS 方法（提供了计算快捷方式）来计算随机效应的估计值和混合模型。由于当今仍在使用的许多教科书仍使用 EMS 方法来介绍包含随机效应的模型，所以 JMP 提供了 EMS 选项。（请参见，例如 McCulloch et al., 2008; Poduri, 1997; Searle et al., 1992。）

REML 方法执行限制似然函数的最大似然估计，它不依赖于固定效应参数。这将生成方差分量的估计值，该估计值随后用于获取固定效应的估计值。精确估计值基于参数的协方差矩阵的估计值。即便数据不平衡，REML 也可提供有用的估计值、检验和置信区间。

EMS 方法通过使观测均方与期望均方相等对方差分量的估计值求解。对于平衡设计，通过一组复杂的规则指定如何获取估计值。对不平衡数据应用该方法存在问题。

对于平衡数据，REML 估计值与 EMS 估计值相同。但是，与 EMS 不同的是，REML 善于处理不平衡数据。

指定随机效应和拟合方法

在“拟合模型”启动窗口中指定包含随机效应的模型。要指定随机效应，在“构造模型效应”列表中将其突出显示，然后选择**特性 > 随机效应**。这将向模型效应列表中的效应名称后附加上“&随机”。（有关随机效应的定义，请参见第 163 页的“随机效应”。）还可以在单独的效应选项卡中指定随机效应。（请参见“模型规格”一章中第 41 页的““构造模型效应”选项卡”。）

在“拟合模型”启动窗口中，一旦向效应添加了“&随机”特性，您就可以在 REML（推荐）或 EMS（传统）这两种拟合方法中任选其一。

警告：您必须显式声明交叉和嵌套关系。例如，当每个对象仅位于一个组内时，对象 ID 也可能标识包含该对象的组。在这种情形下，对象 ID 必须仍声明为嵌套在组内。务必要小心谨慎地明确定义设计结构。

不受限的方差分量参数化

将方差分量参数化可使用两种不同的方法：**不受限**方法和**受限**方法。当模型中存在混合效应时（比如固定效应与随机效应的交互作用），就会出现问题。这种交互作用项被视为随机效应。

在受限方法中，对于随机效应的每个水平，在固定效应各个水平上的交互作用效应的总和假设为 0。在不受限方法中，只是简单地将混合项假设为具有均值 0 和公共方差的正态分布的独立随机实现。（该假设类似于通常应用于剩余误差的假设。）

JMP 和 SAS 使用不受限方法。该区别很重要，因为很多统计学教科书都使用受限方法。这两种方法都已广泛教授有 60 年之久。有关这两种方法的讨论，请参见 Cobb（1998，第 13.3 节）。

负方差

尽管方差始终为正，但在某些情况下方差的无偏估计值也可能为负。以下实验情况可能出现负估计值：某个效应非常微弱，或是只有极少的水平与方差分量对应。因此，观测数据有可能导致估计值为负。

无界限方差分量

JMP 可以为 REML 和 EMS 生成负估计值。对于 REML，“拟合模型”启动窗口中有两个选项：“无界限方差分量”和“仅估计方差分量”。默认选定“无界限方差分量”选项。取消选择该选项将限制方差分量估计值为非负。

若您关注的是固定效应，则应保持选定“无界限方差分量”选项。**将方差估计值限制为非负将导致固定效应检验发生偏倚。**

仅估计方差分量

若您只想查看“REML 方差分量估计值”报表，则选择该选项。若您仅关注方差分量，则可能想限制方差分量为非负。适当的操作可能是取消选择“无界限方差分量”选项，然后选择“仅估计方差分量”选项。

限制最大似然 (REML) 方法

基于所选的拟合方法，“拟合最小二乘法”报表提供不同的分析结果并提供“保存列”和“刻画器”的更多菜单选项。尤其需要指出，“方差分析”报表并不显示，因为方差和自由度未按照通常方式分割。您可以从“REML 方差分量估计值”报表获取残差方差估计值。（请参见第 167 页的“REML 方差分量估计值”。）对于检验固定效应的情况，“效应检验”报表被“固定效应检验”报表取代。其他报表提供随机效应的预测值以及关于方差分量的详细信息。

图 3.62 显示了使用 REML 方法拟合 Investment Castings.jmp 样本数据所获取的报表。运行脚本**模型 - REML**，然后拟合该模型。请注意，**铸件**是嵌套在**温度**内的随机效应。

图 3.62 REML 方法的 “拟合最小二乘法” 报表

响应 “收缩”

参数估计值

随机效应预测

项	BLUP	标准误差	分母自由度	t 比	概率>|t|
温度[1]:铸件[1]	-0.353959	0.337993	7.939	-1.05	0.3258
温度[1]:铸件[3]	0.0524938	0.337993	7.939	0.16	0.8805
温度[1]:铸件[5]	0.5139011	0.337993	7.939	1.52	0.1672
温度[1]:铸件[7]	-0.212436	0.337993	7.939	-0.63	0.5473
温度[2]:铸件[2]	0.4367848	0.337993	7.939	1.29	0.2326
温度[2]:铸件[4]	-0.094099	0.337993	7.939	-0.28	0.7878
温度[2]:铸件[6]	-0.862231	0.337993	7.939	-2.55	0.0343*
温度[2]:铸件[8]	0.519546	0.337993	7.939	1.54	0.1631

REML 方差分量的估计值

随机效应	方差比	方差分量	标准误差	95% 下限	95% 上限	Wald P 值	占合计的百分比
铸件[温度]	1.511973	0.3205234	0.2161944	-0.10321	0.7442566	0.1382	60.191
残差		0.2119901	0.0611963	0.1292489	0.4102654		39.809
合计		0.5325135	0.2204825	0.2716915	1.4771758		100.000

-2 对数似然 = 57.247690262
注意: 合计是正方差分量的总和。
包括负估计值的合计 = 0.5325135

方差分量估计值的协方差矩阵

迭代

固定效应检验

源	参数数目	自由度	分母自由度	F 比	概率>F
温度	1	1	6	6.5130	0.0434*

随机效应预测

对于模型中的每个项，该报表提供其**最佳线性无偏预测变量** (BLUP) 的经验估计值，以及针对相应系数是否为 0 的检验。

注意：必须选定 “回归报表” > “参数估计值” 选项，才能显示 “随机效应预测” 报表。

项 模型中与随机效应对应的项。

BLUP 每个随机效应的**最佳线性无偏预测变量** (BLUP) 的经验估计值。请参见第 167 页的 “最佳线性无偏预测变量”。

标准误差 BLUP 的标准误差。

分母自由度 针对效应为 0 的检验的分母自由度。多数情况下，*t* 检验的自由度是小数。

t 比 用于检验效应为 0 的 *t* 比。通过将 BLUP 除以其标准误差可以得出 *t* 比。

概率 >|t| 检验的 *p* 值。

95% 下限 BLUP 的 95% 置信下限。仅当已选定 “回归报表” > “显示所有置信区间” 选项或是在报表中右击并选择 “列” > “95% 下限” 时，该列才显示。

95% 上限 BLUP 的 95% 置信上限。仅当已选定 “回归报表” > “显示所有置信区间” 选项或是在报表中右击并选择 “列” > “95% 上限” 时，该列才显示。

最佳线性无偏预测变量

最佳线性无偏预测变量 (BLUP) 一词指的是随机效应的估计量。具体而言，它是所有无偏估计量中可最小化均方预测误差的估计量。“随机效应预测”报表提供 BLUP 的估计值或**经验** BLUP。这些之所以是经验值是因为 BLUP 依赖方差分量的值，而这些值是未知的。方差分量的估计值被代入 BLUP 的公式中，生成了报表中所示的估计值。

REML 方差分量估计值

在“拟合模型”启动窗口中将 REML 选作拟合方法后，会提供“REML 方差分量估计值”报表。该报表包含以下列：

随机效应 模型中的随机效应。

方差比 效应的方差分量与残差的方差分量之比。它比较的是效应的估计方差与模型的估计误差方差。

方差分量 效应的估计方差分量。请注意，“合计”的方差分量只是正方差分量之和。所有方差分量的总和在表的下方提供。

标准误差 方差分量估计值的标准误差。

95% 下限 方差分量的 95% 置信下限。请参见第 167 页的“方差分量的置信区间”。

95% 上限 方差分量的 95% 置信上限。请参见第 167 页的“方差分量的置信区间”。

Wald P 值 用于检验协方差参数是否等于零的 p 值。仅当您在“拟合模型”启动窗口中选定了“无界限方差分量”时，该列才显示。

占合计的百分比 效应的方差分量与合计的方差分量之比（用百分比表示）。

方差分量平方根 相应方差分量的平方根。它是效应的标准差的估计值。仅当您在报表中右击并选择“列”>“方差分量平方根”时才显示该列。

变异系数 方差分量的变异系数。它是将方差分量的平方根乘以 100 后再除以响应均值的结果。仅当您在报表中右击并选择“列”>“变异系数”时才显示该列。

范数 KHC Kackar-Harville 校正。请参见第 168 页的“Kackar-Harville 校正”。仅当您在报表中右击并选择“列”>“范数 KHC”时才显示该列。

方差分量的置信区间

用于计算置信限的方法取决于您是否在“拟合模型”启动窗口中选定了“无界限方差分量”。请注意，默认情况下选定“无界限方差分量”选项。

- 若选定了“无界限方差分量”，则计算基于 Wald 的置信区间。这些区间渐近有效，但请注意样本较小时它们并不可靠。
- 若未选定“无界限方差分量”，这表示参数的下边界约束为 0，此时将使用 Satterthwaite 近似 (Satterthwaite 1946)。

Kackar-Harville 校正

在 REML 方法中，使用方差分量估计值来估计固定效应的标准误差。不过，若不考虑这些估计值中的变异性，则会低估标准误差。要考虑增大的变异性，可以使用 Kackar-Harville 校正（Kackar and Harville 1984；Kenward and Roger 1997）来调整固定效应的协方差矩阵。涉及固定效应的协方差矩阵的所有计算都使用该校正。这些计算包括最小二乘均值、固定效应检验、置信区间和预测方差。有关统计详细信息，请参见第 179 页的“Kackar-Harville 校正”。

“范数 KHC”是 Kackar-Harville 校正的 Frobenius（矩阵）范数。在设计颇为平衡的情况下，“范数 KHC”往往较小。

方差分量估计值的协方差矩阵

该报表提供方差分量的渐近协方差矩阵的估计值。它是观测的 Fisher 信息矩阵的逆矩阵。

迭代

通过将仅依赖于这些参数的残差对数似然函数最大化，可获得 σ^2 的估计值和 *G* 中的方差分量。迭代过程尝试将残差对数似然函数最大化，或等效于将负残差对数似然的两倍 （–2 对数似然）最小化。“迭代”报表提供有关该过程的详细信息。

迭代 迭代次数。

-2 对数似然 负对数似然的两倍。它是目标函数。请参见 “统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

范数梯度 目标函数的梯度 （一阶导数）范数。

参数 在标记为“参数”的列和其余列中，每一列都对应一个随机效应。列的顺序遵循随机效应在 “REML 方差分量估计值”报表中所列的顺序。在每次迭代中，列中的值是该阶段方差分量的估计值。

收敛准则基于梯度，默认容差为 10^{-8}。您可以在“拟合模型”启动窗口中更改该准则，只需选择 “收敛设置” > “收敛极限”选项并指定所需容差即可。

固定效应检验

使用 REML 时，“效应检验”报表提供固定效应检验。该报表包含以下列：

源 模型中的固定效应。

参数数目 与效应关联的参数数目。

自由度 与效应关联的自由度。

分母自由度 分母自由度。这些基于使用 Kenward-Roger 校正调整协方差矩阵时获得的统计量的近似分布。请参见第 168 页的 “Kackar-Harville 校正”和第 73 页的 “随机效应”。

F 比 计算的 *F* 比。

概率 > F 效应检验的 p 值。

REML“保存列”选项

使用 REML 方法时，“保存列”菜单中将额外显示六个选项。这些选项的名称以**条件**一词开头。该前缀指示这些列的计算使用与随机效应关联的项的预测值，而不是其期望值 0。

条件预测公式 将预测公式保存至数据表中的新列。

条件预测值 将预测值保存至数据表中的新列。

条件残差 将残差保存至数据表中的新列。

条件均值置信区间 保存均值的置信区间。

条件单值置信区间 保存单值的置信区间。

JMP PRO **发布条件公式** 创建条件预测公式并将其另存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见《预测和专业建模》。

REML“刻画器”选项

当您使用 REML 方法并选择“因子刻画”>“刻画器”时，新选项“条件预测”将显示在“预测刻画器”旁边的红色小三角菜单中。请注意，条件值使用随机效应的预测值，而不是其期望值 0。

注意：刻画器为因子水平的所有组合显示条件预测值和条件均值置信区间。某些组合可能因为嵌套而无意义。

EMS（传统）模型拟合报表

警告：不推荐使用 EMS。REML 是推荐的方法。

将 EMS 选作拟合方法时，将显示四个新报表。“效应检验”报表不显示，因为在“关于随机效应的检验”报表中执行同时针对固定和随机效应的检验。

期望均方

模型效应的期望均方是方差分量和固定效应值的线性组合，其中包括剩余误差方差。该表提供用于定义每个模型效应的期望均方的系数。矩阵行对应于左侧列出的效应。列对应于顶部标识的方差分量。每个期望均方都包括系数为 1 的残差方差。该信息在表下方提供。

图 3.63 显示了 Investment Castings.jmp 样本数据表的“期望均方”报表。运行“模型 - EMS”脚本，然后运行该模型。

图 3.63 “期望均方”报表

期望均方

每行的均方乘以每列的方差分量

EMS

	截距	温度	铸件[温度]&随机
截距	0	0	0
温度	0	16	4
铸件[温度]&随机	0	0	4

加上 1.0 倍的剩余误差方差

正如表中所指明的那样，**铸件 [温度]** 的期望均方为

$$4\sigma^2_{\text{铸件[温度]}} + \sigma^2_{\text{误差}}$$

方差分量估计值

通过让期望均方等于相应的观测均方并求解，可以获得方差分量的估计值。“方差分量估计值”报表提供估计的方差分量。

成分 列出随机效应。

方差分量估计值 提供方差分量的估计值。

占合计的百分比 提供方差分量与方差分量总和的比值。

变异系数 提供方差分量的变异系数。它是将方差分量的平方根乘以 100 后再除以响应均值的结果。

注意：仅当您在报表中右击并选择 “列” > “变异系数”时才显示。

检验分母合成

对于要检验的每个效应，都会构造一个 F 统计量。该统计量的分母是一个均方值，其期望值是原假设下的分子均方的期望值。该分母是通过固定效应所关联的方差分量和值构造或**合成**的。

源 显示要检验的效应。

分母均方 提供 *F* 检验的分母的估计均方。

分母自由度 提供合成分母的自由度。它们是使用 Satterthwaite 方法 (Satterthwaite 1946) 构造的。

分母均方的合成 提供分母合成中使用的方差分量。剩余误差方差始终是该合成的一部分。

关于随机效应的检验

该报表中提供有关固定和随机效应的检验。

源 列出要检验的效应。这些效应包括固定和随机效应。

平方和 提供效应的平方和。

分子均方 提供分子均方。

分子自由度 提供分子自由度。

F 比 提供检验的 *F* 比。这是分子均方与分母均方的比值。分母均方的值可从“检验分母合成”报表中获得。

概率 > F 提供效应检验的 *p* 值。

警告：最小二乘均值的标准误差和对比 *F* 检验的分母都使用合成的分母。在特定情形下（比如检验涉及在公共水平上比较的交叉效应），这些检验可能不合适。执行定制检验要使用剩余误差，构造杠杆图要使用剩余误差，所以这些检验可能同样不合适。

EMS 刻画器

使用 EMS 方法并选择“因子刻画”>“刻画器”后，刻画器会基于固定效应模型提供预测和条件均值置信区间。这些值不基于随机效应的预测值。

模型项之间具有线性相依性的模型

若预测变量矩阵各列之间存在线性相依性，这将影响若干标准最小二乘法报表。

- “奇异性详细信息”
- ““参数估计值”报表”
- ““效应检验”报表”
- “示例”

奇异性详细信息

线性回归模型的公式为 $Y = X\beta + \varepsilon$。在此，X 是这样一个矩阵：其第一列由 *1* 构成，其余列都是模型中非截距项的值。若模型包含 p 个项（包括截距），则 X 为一个 $n \times p$ 矩阵，其中 n 是观测数。参数估计值 （用向量 b 表示）通常通过以下方式提供：

$$b = (X'X)^{-1}X'Y$$

不过，该公式假定 $X'X^{-1}$ 存在，换言之，$p \times p$ 矩阵 $X'X$ 是可逆的（等效于满秩）。若 $X'X$ 不可逆则经常会出现问题，原因在于 X 各列之间存在线性相依性。

在这些情况下，矩阵 $X'X$ 是奇异的，“拟合最小二乘法”报表包含“奇异性详细信息”报表。该报表包含描述线性相依性的表达式的表。这些线性相依性中涉及的项存在别名关系（混杂）。

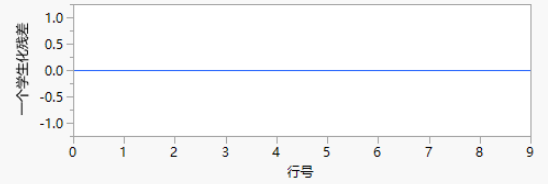
图 3.64 显示了 Reactor 8 Runs.jmp 样本数据表的报表。要获取这些报表，请以**反应百分比**作为 Y 来拟合模型。输入**进料速度**、**催化剂**、**搅拌速度**、**温度**、**浓度**、**催化剂*搅拌速度**、**催化剂*浓度**以及**进料速度*催化剂**作为模型效应。

图 3.64 具有线性相依性的模型的 “奇异性”和 “参数估计值”报表

奇异性详细信息

项	详细信息
浓度(3,6)	=催化剂*搅拌速度
搅拌速度(100,120)	=催化剂*浓度

多个学生化残差



具有 95% 联合限值的外部学生化残差 (Bonferroni)

参数估计值

项		估计值	标准误差	t 比	概率>|t|
截距		65.875	6.125	10.76	0.0590
进料速度(10,15)		-4.875	6.125	-0.80	0.5720
催化剂(1,2)		10.125	6.125	1.65	0.3463
搅拌速度(100,120)	有偏	1.875	6.125	0.31	0.8109
温度(140,180)		5.875	6.125	0.96	0.5133
浓度(3,6)	有偏	-3.375	6.125	-0.55	0.6794
催化剂*搅拌速度	清零	0	0	.	.
催化剂*浓度	清零	0	0	.	.
进料速度*催化剂		0.375	6.125	0.06	0.9611

效应检验

源	参数数目	自由度	平方和	F 比	概率>F	
进料速度(10,15)	1	1	190.12500	0.6335	0.5720	
催化剂(1,2)	1	1	820.12500	2.7326	0.3463	
搅拌速度(100,120)	1	0	0.00000	.	.	损失自由度
温度(140,180)	1	1	276.12500	0.9200	0.5133	
浓度(3,6)	1	0	0.00000	.	.	损失自由度
催化剂*搅拌速度	1	0	0.00000	.	.	损失自由度
催化剂*浓度	1	0	0.00000	.	.	损失自由度
进料速度*催化剂	1	1	1.12500	0.0037	0.9611	

“参数估计值”报表

若 $X'X$ 是奇异的，则使用广义逆来获取估计值。该方法允许估计涉及线性相依性的部分参数而不是全部参数。基于参数关联项输入模型的顺序来估计这些参数，这样最后输入的项就是不估计其参数的项。估计值在“参数估计值”报表中提供，指定无法估计的参数的估计值为 0。

不过，涉及线性相依性的项的参数估计值并不唯一。由于关联项存在别名关系，所以有无限多的估计值向量都满足最小二乘准则。在这些情况下，“有偏”显示在“参数估计值”报表中这些估计值的左侧。对于涉及线性相依性但其参数无法估计的项，“清零”将显示在“参数估计值”报表中的估计值 0 的左侧。有关示例，请参见图 3.64。

若有可用于误差估计值的自由度，则执行使用有偏估计值估计的参数的 *t* 检验。但考虑到这些估计值并不唯一，应该谨慎解释这些检验。

“效应检验”报表

在标准最小二乘法拟合中，有多少个模型自由度就可以估计多少个参数。执行 “效应检验”报表中的检验时，每个效应都被视为在模型中输入的最后一个效应。

- 一个效应的自由度在所有模型自由度都已被其他效应用掉时将显示为 0。自由度等于 0 时，无法计算平方和。因此，无法检验该效应。
- 若并不是所有模型自由度都已被其他效应使用，那么该效应具有非零自由度。不过，其自由度可能少于其参数个数（参数数目），这指示只能检验其部分关联参数。

若效应的自由度非零，则执行 *F* 检验，假定存在误差自由度。只要自由度少于参数数目，与该效应对应的行最右侧就会显示说明“损失自由度”（图 3.64）。这些效应有机会只解释尚未归因于吸收了其损失自由度的别名效应的模型平方和。由此判定：“效应检验”报表中给出的平方和很可能不足以代表与该效应关联的“实际”平方和。若该检验显著，则其显著性有意义。但应谨慎解释显著性的缺乏。

有关统计详细信息，请参见 SAS Institute Inc.(2020c)。

示例

打开 Singularity.jmp 样本数据表。其中包含一个响应 Y，四个预测变量 X1、X2、X3 和 A 以及五个观测。除了 A 是具有四个水平的名义型预测变量之外，其余预测变量都是连续的。还请注意：这些连续效应之间具有线性相依性，也就是说，X3 = X1 + X2。

估计值的不唯一性

要查看存在线性相依性时不唯一的估计值，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Singularity.jmp。
2. 运行脚本**模型 1**。该脚本打开“拟合模型”启动窗口，其中的效应按照 X1、X2、X3 的顺序输入。
3. 点击**运行**并保持报表窗口打开。
4. 运行脚本**模型 2**。该脚本打开“拟合模型”启动窗口，其中的效应按照 X1、X3、X2 的顺序输入。
5. 点击**运行**并保持报表窗口打开。

比较这两个报表（图 3.65）。两个报表顶部的“奇异性详细信息”报表显示线性相依性，指示 X1 = X3 - X2。

现在比较两个模型的“参数估计值”报表。举例来说，请注意“模型 1”的 X1 的估计值为 –1.25，而“模型 2”的该值为 2.75。在这两个模型中，仅估计了与效应关联的两个项，因为只有两个模型自由度。请参见“方差分析”报表。估计的两个项的估计值标记为“有偏”，而其余估计值设置为 0 并标记为“清零”。

“效应检验”报表显示未执行任何检验。每一行都标记为“损失自由度”。发生这种情况的原因是：这些效应中任何一个的效应检验都要求在模型中最后输入该效应。不过，另外两个效应

完全可以解释与这两个模型自由度关联的模型平方和。所以，对于关注的效应，不存在自由度或关联的平方和。

图 3.65 模型 1 （左）和模型 2 （右）的 “拟合最小二乘法”报表

损失自由度

要获取关于损失自由度的更多深入分析，请按照以下步骤执行或运行数据表脚本**拟合模型报表**:

1. 选择**帮助** > **样本数据库**，然后打开 Singularity.jmp。
2. 点击**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 选择 X1 和 A，然后点击**添加**。
5. 将**重点**设置为**最小报表**。
6. 点击**运行**。

报表各部分显示在图 3.66 中。“奇异性详细信息”报表显示存在涉及到 X1 和效应 A 所关联的三个项的线性相依性。（有关如何对名义型效应编码的详细信息，请参见第 176 页的 ““定制检验”示例的详细信息”）。“方差分析”报表显示有三个模型自由度。“参数估计值”报表为 X1、A[a] 和 A[b] 这三个项显示了“有偏”估计值，为第四个项 A[c] 显示了“清零”估计值。

“效应检验”报表显示无法检验 X1，因为必须首先输入 A，A 解释三个模型自由度。不过，可以检验 A，但仅有两个自由度。（必须首先输入 X1，它解释其中一个模型自由度。）针对 A 的检验是不完整的，所以必须谨慎解释它。

图 3.66 包含 X1 和 A 的模型的 “拟合最小二乘法”报表

响应 “Y”

奇异性详细信息

项	详细信息
截距	=2*X1 - A[a] - A[b] + 3*A[c]

回归图

拟合汇总

方差分析

源	自由度	平方和	均方	F 比
模型	3	98.300000	32.7667	65.5333
误差	1	0.500000	0.5000	概率>F
校正总和	4	98.800000		0.0905

参数估计值

项		估计值	标准误差	t 比	概率>|t|
截距	有偏	14.416667	0.399653	36.07	0.0176*
X1	有偏	-2.583333	0.399653	-6.46	0.0977
A[a]	有偏	-5.333333	0.471405	-11.31	0.0561
A[b]	有偏	3.1666667	0.552771	5.73	0.1100
A[c]	清零	0	0	.	.

效应检验

源	参数数目	自由度	平方和	F 比	概率>F	
X1	1	0	0.000000	.	.	损失自由度
A	3	2	64.500000	64.5000	0.0877	损失自由度

效应详细信息

“标准最小二乘法”特质的统计详细信息

- “重点规则”
- “‘定制检验’示例的详细信息”
- “估计值的相关性”
- “杠杆图详细信息”
- “Kackar-Harville 校正”
- “功效分析”

重点规则

“拟合模型”启动窗口中的默认重点基于行数 n、在“构造模型效应”列表中输入的效应数 (k)，以及应用于效应的特性。

- 若 $n > 1000$，则“重点”设置为“最小报表”。
- 若 $n \leq 1000$ 且 $k \leq 4$，则“重点”设置为“效应杠杆率”。
- 若 $n \leq 1000$ 且 $k \geq 10$，则“重点”设置为“效应筛选”。
- 若 $n \leq 1000$，$4 < k < 10$ 且 $n - k > 20$，则“重点”设置为“效应杠杆率”。
- 若有任何效应具有“随机效应”特性，则“重点”设置为“最小报表”。
- 若以上任何条件都不成立，则“重点”设置为“效应筛选”。

“定制检验”示例的详细信息

在第 110 页的 “‘定制检验’的示例”中，您关注的是使用 Cholesterol.jmp 样本数据表检验三个对比。具体而言，您想要比较：

- 治疗 A 和 B 的响应均值
- 治疗 A 和 B 的合并响应均值与控制组的响应均值
- 治疗 A 和 B 的合并响应均值与控制组和安慰剂组的合并响应均值

要获得您在“定制检验”列中输入的对比系数，请执行以下操作。将这四个治疗组的理论效应表示为：α_A、α_B、$\alpha_{Control}$ 和 $\alpha_{Placebo}$。这些都是治疗效应，所以它们的总和限制为 0。由于与指标变量关联的参数仅表示前三个效应，所以您需要按照这前三个效应用公式表示对比。请参见第 176 页的 “‘定制检验’示例的详细信息”和“统计详细信息”附录中第 457 页的 “参数的解释”。

您想要检验的假设可以按照模型效应写成如下形式：

- 比较治疗 A 与治疗 B：

  $$\alpha_A - \alpha_B = 0$$

- 将治疗 A 和 B 与控制组作比较：

  $$0.5(\alpha_A + \alpha_B) - \alpha_{Control} = 0$$

- 将治疗 A 和 B 与控制组和安慰剂组作比较：

  $$0.5(\alpha_A + \alpha_B) - 0.5(\alpha_{Control} + \alpha_{Placebo}) = \alpha_A + \alpha_B = 0$$

  要获取该对比的对比系数，您需要按照模型效应编写安慰剂效应。具体而言，使用事实 $\alpha_A + \alpha_B + \alpha_{Control} + \alpha_{Placebo} = 0$。于是 $\alpha_{Placebo} = -\alpha_A - \alpha_B - \alpha_{Control}$。由此判定：

$$\begin{aligned} 0.5(\alpha_A + \alpha_B) - 0.5(\alpha_{Control} + \alpha_{Placebo}) &= 0.5(\alpha_A + \alpha_B) - 0.5(\alpha_{Control} - \alpha_A - \alpha_B - \alpha_{Control}) \\ &= 0.5(\alpha_A + \alpha_B) - 0.5\alpha_{Control} + 0.5(\alpha_A + \alpha_B + \alpha_{Control}) \\ &= \alpha_A + \alpha_B \\ &= 0 \end{aligned}$$

估计值的相关性

考虑包含 n 个观测和 $p - 1$ 个预测变量的数据集。定义矩阵 **X** 为设计矩阵。即：**X** 是 $n \times p$ 矩阵，其中的第一列包含 *1*，其余 $p - 1$ 列包含 $p - 1$ 个预测变量值。（名义型列按照指标预测变量来编码。每一列都是矩阵 X 中的一列。）

回归系数的向量的估计值为

$$\hat{\beta} = (X'X)^{-1}X'Y$$

其中，Y 表示响应值的向量。

在通常的回归假设下，$\hat{\beta}$ 的协方差矩阵为

$$Cov(\hat{\beta}) = \sigma^2(X'X)^{-1}$$

其中，σ^2 表示响应的方差。

通过将协方差矩阵中的每个元素除以对角线元素平方根的乘积，可获得估计值的相关性矩阵。将 V 定义为对角矩阵，其元素是协方差矩阵对角线元素的平方根：

$$V = Sqrt(Diag(Cov(\hat{\beta})))$$

于是参数估计值的相关性矩阵可通过以下公式得出：

$$Corr(\hat{\beta}) = \sigma^2 V^{-1}(X'X)^{-1}V^{-1}$$

杠杆图详细信息

效应杠杆图亦称**偏回归残差杠杆图** (Belsley et al. 1980) 或**增加变量图** (Cook and Weisberg 1982)。Sall (1990) 推广了这些图，可应用于任何线性假设。

JMP 提供两类杠杆图：

- 效应杠杆图显示相对于以下假设的观测：关注的效应不在模型中，而其他所有效应都包含在模型中。
- 整体模型杠杆图（在“预测值 - 实际值”图报表中提供）显示相对于没有因子效应的假设下的观测。

在效应杠杆图中，只有一个效应假定为 0。但在整体模型“预测值-实际值”图中，所有效应都假定为 0。Sall (1990) 发表的论文将杠杆图理念推广到了任意线性假设，整体模型杠杆图就是其中的一个例子。该论文中的详细信息（在本节中进行了总结）具体表示为 JMP 中的上述两类图。

构造

假定关注的可估计假设为

$$L\beta = 0$$

杠杆图通过以下方式刻画了该检验的特征：标绘各个点，让每个点与斜率回归线的距离显示无约束残差。与位于 0 处的水平线的距离显示拟合受到假设约束时的残差。这两组残差的平方和之间的差值是因假设导致的平方和。该值成为 F 检验的主成分。

受假设约束的参数估计值可写为

$$b_0 = b - (X'X)^{-1}L'\lambda$$

在此， b 是最小二乘估计值

$$b = (X'X)^{-1}X'y$$

lambda 是假设约束的 Lagrangian 乘数，通过以下公式计算得出

$$\lambda = (L(X'X)^{-1}L')^{-1}Lb)$$

无约束和受假设约束的残差分别为

$$r = y - Xb$$

$$r_0 = r + X(X'X)^{-1}L'\lambda$$

对于每个观测，考虑水平轴值为 v_x 且垂直轴值为 v_y 的点，其中：

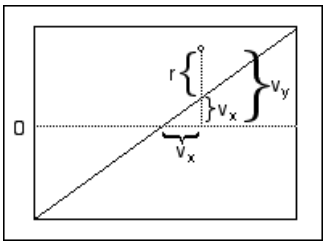
- v_x 是受约束残差减去无约束残差 $r_0 - r$，反映出应用约束后留下的信息
- v_y 是水平轴值加上无约束残差

因此，这些点具有以下 x 坐标和 y 坐标

$$v_x = X(X'X)^{-1}\mathbf{L}'\lambda \text{ 且 } v_y = r + v_x$$

这些点构成杠杆图的基础。图 3.67 对此构造进行了演示，其中，响应均值为 0，实线的斜率为 1。JMP 中的杠杆图在响应均值 $\bar{y}$ 处有一条水平虚线。标绘的点的坐标为：$(v_x + \bar{y}, v_y)$。

图 3.67 杠杆图的构造



在杠杆图上叠加检验

在简单线性回归中，您可以将响应期望值的置信限绘制为预测变量 **x** 的平滑函数

$$\text{上限 (x)} = \mathbf{xb} + t_{\alpha/2}s\sqrt{x(X'X)^{-1}x'}$$

$$\text{下限 (x)} = \mathbf{xb} - t_{\alpha/2}s\sqrt{x(X'X)^{-1}x'}$$

其中，$\mathbf{x} = [1\ x]$ 是双向量的预测变量。

这些置信曲线给出了相应假设检验的显著性的直观估计，如图 3.56 所示：

- 显著：若斜率参数与 0 存在显著差异，置信曲线将在响应均值处跨越水平线。
- 边界线：若斜率参数的 *t* 检验恰好落在显著性边缘上，则置信曲线在响应均值处渐近水平线。
- 不显著：若斜率参数与 0 没有显著差异，置信曲线不会在响应均值处跨越水平线。

杠杆图通过显示置信曲线来体现这种思路。对这些曲线进行了调整以使图适当中心化。用 *z* 来表示水平轴上的点。定义函数

$$\text{上限}(z) = z + \sqrt{s^2 t^2_{\alpha/2} \bar{h} + (F_\alpha / F) z^2}$$

且

$$\text{下限}(z) = z - \sqrt{s^2 t^2_{\alpha/2} \bar{h} + (F_\alpha / F) z^2}$$

其中， F 是假设的 F 统计量， F_α 是显著性水平 α 的参考值。

并且 $\bar{h} = \bar{x}(X'X)^{-1}\bar{x}'$，其中 $\bar{x}$ 是由预测变量的合适中间值 （如其均值）构成的行向量。

这些函数的行为方式与简单线性回归的置信曲线相同：

- 若 F 统计量大于参考值，则置信函数与水平轴交叉。
- 若 F 统计量等于参考值，则置信函数以水平轴为渐近线。
- 若 F 统计量小于参考值，则置信函数不交叉。

此外，重要的是上限(*z*) - 下限(*z*) 对于 z 处的预测值而言为有效的置信区间。

Kackar-Harville 校正

固定效应的方差矩阵始终修改为包括 Kackar-Harville 校正。BLUP 的方差矩阵以及 BLUP 与固定效应之间的协方差都不做 Kackar-Harville 校正。这种方法的原理是：当随机效应包含许多水平时，BLUP 校正可能占用较多的计算和内存资源。

在 SAS 中，仅当设置了 DDFM=KENWARDROGER 时，才为固定效应和 BLUP 执行 Kackar-Harville 校正。

- 仅涉及固定效应参数的线性组合的标准误差与 PROC MIXED DDFM=KENWARDROGER 相符。这种情况假定用户已经在 PROC MIXED 和 JMP 使用的不同参数化之间谨慎变换。
- 仅涉及 BLUP 参数的线性组合的标准误差与 PROC MIXED DDFM=SATTERTHWAITE 相符。
- 若数据不平衡，则同时涉及固定效应和 BLUPS 的线性组合的标准误差与任何 DDFM 选项的 PROC MIXED 都不相符。不过，这些标准误差介于使用 DDFM=SATTERTHWAITE 选项与 DDFM=KENWARDROGER 选项分别获得的结果之间。若数据平衡，不论如何设置 DDFM 选项，平衡数据的 JMP 结果始终与 SAS 结果相符，因为 Kackar-Harville 校正为空。

自由度

仅涉及固定效应参数的线性组合的检验的自由度使用 Kenward-Roger 校正来计算。因此，JMP 中这些检验的结果与使用 DDFM=KENWARDROGER 选项的 PROC MIXED 相符。若线性组合中存在 BLUP，JMP 将使用 Satterthwaite 近似来获取自由度。结果将遵循与上一段中为标准误差描述的模式类似的模式。

有关 Kackar-Harville 校正和 Kenward-Roger 自由度方法的详细信息，请参见 Kenward and Roger (1997)。Satterthwaite 方法在 SAS Institute Inc.(2020d, ch.83)。

功效分析

与功效计算有关的选项仅适用于连续响应模型。以下内容是可用的功效和相关检验详细信息：

参数估计值

要获取每个参数估计值的回顾检验详细信息，请从报表的红色小三角菜单中选择**估计值 > 参数功效**。该选项基于当前研究数据，为每个参数显示 0.05 显著性水平检验的最小显著值、最小显著数和调整功效。

效应或效应详细信息

要获取特定效应的 *F* 检验的前瞻或回顾详细信息，请从效应的红色小三角菜单中选择**功效分析**。谨记：对于“效应筛选”和“最小报表”特质，每个效应的报表位于“效应详细信息”下方。对于“效应杠杆率”特质，效应的报表位于报表中第一列（整体模型）右侧。

最小二乘均值对比

要获取一个或多个对比的检验的前瞻或回顾详细信息，请从效应的红色小三角菜单中选择**最小二乘均值对比**。定义所关注的对比并点击“完成”。从“对比”红色小三角菜单中，选择**功效分析**。

定制检验

要获取定制检验的前瞻或回顾详细信息，请从响应的红色小三角菜单中选择**估计值 > 定制检验**。定义所关注的对比并点击**完成**。从“定制检验”红色小三角菜单中，选择**功效分析**。

在除第一种情况外的所有情况中，选择“功效分析”都可以打开“功效详细信息”窗口。随后您可以在“功效详细信息”窗口中输入信息，根据您的需要修改计算。

效应大小

效应大小（表示为 δ）是表示原假设与涉及参数的真实值之间差值的测度。可根据设置为等于 0 的单个线性对比或根据具有该设置的若干对比来用公式表示原假设。δ 的值反映对比的实际值与其假设值 0 之间的差值。

概括来说，效应大小通过以下公式得出：

$$\delta = \sqrt{SS_{Hyp(Pop)}/n}$$

其中，$SS_{Hyp(Pop)}$ 是按照总体参数指定的所检验假设的平方和，n 是观测总数。

若提供了观测数，则将针对假设计算的平方和代入 δ 的公式来计算估计的效应大小。

平衡单因子布局

例如，对于包含 k 个水平的平衡单因子布局的特例，其中第 i 个组具有响应均值 α_i，

$$\delta^2 = \frac{\sum(\alpha_i - \bar{\alpha})^2}{k}$$

回想一下 JMP 对参数编码，因此，对于 i =1, 2,..., k-1

$$\beta_i = (\alpha_i - \bar{\alpha})$$

且

$$\beta_k = -\sum_{m=1}^{k-1} \alpha_m$$

所以，按照这些参数，两水平平衡布局的 δ 通过以下公式得出：

$$\delta^2 = \frac{\beta_1^2 + (-\beta_1)^2}{2} = \beta_1^2$$

或者 $\delta = |\beta_1|$

不平衡单因子布局

对于包含 k 个水平的不平衡单因子布局的情况，其中第 i 个组具有响应均值 α_i 和 n_i 个观测，且 $n = \sum n_i$：

$$\delta^2 = \sum \frac{n_i}{n}(\alpha_i - \bar{\alpha})^2$$

效应大小和功效

功效就是当真实效应大小是一个指定值时某个假设的 F 检验在 α 显著性水平下显著的概率。若真实效应大小等于 δ，则检验统计量服从非中心参数为以下值的非中心 F 分布

$$\lambda = (n\delta^2)/\sigma^2$$

若原假设成立（即：效应大小为 0），则非中心参数为 0 且检验统计量服从 F 分布。

检验的功效随 λ 增大。特别是功效随着样本大小 n 和效应大小 δ, 增大，并随着误差方差 σ^2 减小。

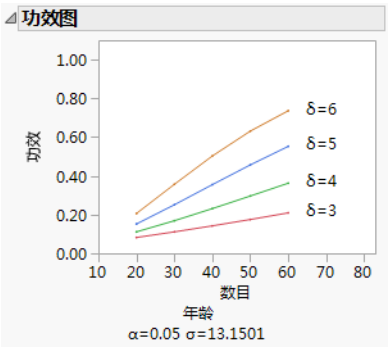
某些书目（例如，Cohen (1977)）使用标准化效应大小 $\Delta = \delta/\sigma$，而不是 JMP 所用的原始效应大小。对于标准化效应大小，非中心参数等于 $\lambda = n\Delta^2$。

在“功效详细信息”窗口中，δ 最初设置为 $\sqrt{SS_{Hyp}/n}$。SS_{Hyp} 是假设的平方和，n 是当前研究中的观测数。SS_{Hyp} 是基于数据计算的 δ 的估计值，但这种估计值是有偏的 (Wright and O'Brien 1988)。要使用 δ 的样本估计值计算功效，您可能需要使用“调整功效和置信区间”计算而不是“功效求解”计算。调整功效计算使用 δ 的估计值，该值已对偏倚进行了部分校正。请参见“统计详细信息”附录中第 480 页的“调整功效的计算”。

样本大小 - 功效图

要查看样本大小-功效图，请从“功效”报表底部的红色小三角菜单中选择“功效图”选项。JMP 标绘“功效”表中的“功效”和“数目”列。图 3.68 所示的图是通过标绘第 184 页的“回顾功效分析的示例”中获取的“功效”表得到的。

图 3.68 样本大小 - 功效图



最小显著数 (LSN)

最小显著数 (LSN) 是在指定了 delta、sigma 和 alpha 值的前提下，得到显著检验结果的最小观测数。回想一下，delta、sigma 和 alpha 分别代表效应大小、误差标准差和显著性水平。

注意：LSN 不适于推荐样本大小，因为它未考虑到显著性概率。其计算基于 delta 和 sigma 的指定值。

LSN 具有以下特征：

- 若 LSN 小于实际样本大小 n，则效应显著。
- 若 LSN 大于 n，则效应不显著。若您相信更多数据实际上会显示与当前样本相同的结构化结果，则 LSN 可建议实现显著性所需的数据量。
- 若 LSN 等于 n，则 p 值等于显著性水平 alpha。检验落在显著的分界线上。

- 当 n = LSN 时计算的效应大小的检验功效始终大于等于 0.5。但请注意，该功效可能接近 0.5，该值被视为较低，无法满足计划目的。

最小显著值 (LSV)

针对单个自由度的假设检验计算 LSV 或**最小显著值**。这些检验包括针对各个模型参数的显著性检验，以及更一般化的线性对比。LSV 是在水平 alpha 下显著的绝对值最小的效应大小。LSV 在参数尺度而不是概率尺度上提供了关于检验灵敏度的测度。

LSV 具有以下特征：

- 若参数估计值或对比的绝对值大于等于 LSV，则显著性检验的 p 值小于等于 alpha。
- 当且仅当其显著性检验的 p 值等于 alpha 时，参数估计值或对比的绝对值才等于 LSV。
- LSV 是参数或参数线性组合的 1 – α 置信区间的半径。1 – α 置信区间以参数或对比的估计值为中心。

功效

检验的功效是该检验给出显著结果的概率。功效是效应大小 δ, 显著性水平 α、误差标准差 σ 和样本大小 n 的函数。功效是您在给定的显著性水平下检测出指定效应大小的概率。通常，您想要让设计的研究具有高功效，以检测出具有实际意义或科学意义的差值。

功效具有以下特征：

- 若参数真实值实际上就是假设值，则功效等于检验的显著性水平。显著性水平通常是一个较小的值，比如 0.05。较小的值之所以合理是因为您希望当假设成立时有较低的概率看到显著结果。
- 若参数真实值**不是**假设值，通常情况下，您希望功效越大越好。
- 功效随着以下值变化而增大：样本大小增大、误差方差减小、真实参数值与假设值之间的差值增大。

调整功效和置信区间

在回顾功效分析中，您通常将样本估计值替换为功效计算中涉及的总体参数。该替换导致非中心参数估计值具有正偏倚 (Wright and O’Brien 1988)。调整功效计算基于估计的非中心参数的形式，该参数已针对偏倚进行部分校正。

您还可以为调整功效构造置信区间。这种置信区间往往较宽。请参见 Wright and O’Brien (1988)。

请注意，调整功效和置信区间计算仅对从数据中估计的 δ 的值（默认提供的值）有意义。对于其他 delta 值，不提供调整功效和置信区间。

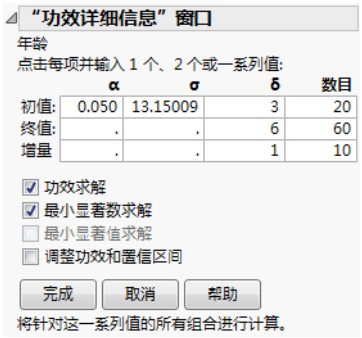
请参见“统计详细信息”附录中第 480 页的“调整功效的计算”。

回顾功效分析的示例

本例使用 Big Class.jmp 样本数据表演示了回顾功效分析。“功效详细信息”窗口（图 3.69）允许在 α、σ、δ 和“数目”或研究大小的范围内探索各种量。点击“完成”，该窗口将替换为计算结果。

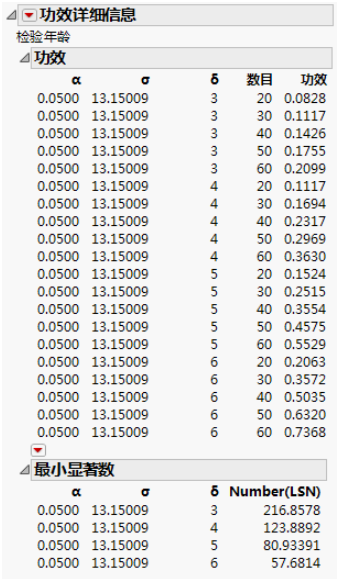
1. 选择**帮助** > **样本数据库**，然后打开 Big Class.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**体重**并点击 Y。
4. 添加**年龄**、**性别**和**身高**作为效应。
5. 点击**运行**。
6. 点击“年龄”红色小三角并选择**功效分析**。

图 3.69 “年龄”的“功效详细信息”窗口



7. 将**初值**框中的 δ 值替换为 3，在**终值**框和**增量**框中分别输入 6 和 1，如图 3.69 所示。
8. 将**初值**框中的**数目**值替换为 20，在**终值**框和**增量**框中分别输入 60 和 10，如图 3.69 所示。
9. 选择**功效求解**和**最小显著数求解**。
10. 点击**完成**。
11. “功效详细信息”报表取代“功效详细信息”窗口。

图 3.70 “年龄”的“功效详细信息”报表



功效详细信息

检验年龄

功效

α	σ	δ	数目	功效
0.0500	13.15009	3	20	0.0828
0.0500	13.15009	3	30	0.1117
0.0500	13.15009	3	40	0.1426
0.0500	13.15009	3	50	0.1755
0.0500	13.15009	3	60	0.2099
0.0500	13.15009	4	20	0.1117
0.0500	13.15009	4	30	0.1694
0.0500	13.15009	4	40	0.2317
0.0500	13.15009	4	50	0.2969
0.0500	13.15009	4	60	0.3630
0.0500	13.15009	5	20	0.1524
0.0500	13.15009	5	30	0.2515
0.0500	13.15009	5	40	0.3554
0.0500	13.15009	5	50	0.4575
0.0500	13.15009	5	60	0.5529
0.0500	13.15009	6	20	0.2063
0.0500	13.15009	6	30	0.3572
0.0500	13.15009	6	40	0.5035
0.0500	13.15009	6	50	0.6320
0.0500	13.15009	6	60	0.7368

最小显著数

α	σ	δ	Number(LSN)
0.0500	13.15009	3	216.8578
0.0500	13.15009	4	123.8892
0.0500	13.15009	5	80.93391
0.0500	13.15009	6	57.6814

该分析是回顾功效分析，因为计算中假设研究具有与 Big Class.jmp 样本数据表相同的结构。例如，本例中的功效计算依赖于模型中输入的效应以及每个年龄和性别分组中的参与者数。此外，σ 的值从当前研究中获得，尽管您可以用代表将来研究的值替换该值。

有关图 3.70 中显示的功效结果的详细信息，请参见第 183 页的“功效”。有关最小显著数 (LSN) 的详细信息，请参见第 182 页的“最小显著数 (LSN)”。

前瞻功效分析

前瞻分析帮助您回答这样的问题：“若存在指定大小的差异，我是否能在建议的样本大小、alpha 水平和误差方差估计值的条件下检测到这些差异？”在前瞻功效分析中，您必须在数据表中提供组均值的估计值和样本大小。您还必须在“功效详细信息”窗口中提供误差标准差 σ 的估计值。

相等组大小

考虑您准备比较三个独立组的均值的情形。要获取可实现给定功效的样本大小，请选择**实验设计 > 样本大小与功效**，然后选择 **k 个样本均值**。在“标准差”旁边，输入您的误差标准差估计值。在“前瞻均值”列表中，输入反映您想要检测的最小差值的均值。举例来说，若您想要检测任意两个均值之间 8 个单位的差值，请输入均值的极值（比如 40、40 和 48）。由于功效基于总均值的偏差，您可以仅输入反映所需差值的值（比如 0、0 和 8）。

若点击**继续**，即可获取样本大小-功效图。若在“样本大小”窗口中指定功效或样本大小，则计算其他量并显示在“样本大小”窗口中。特别是，若您指定功效，提供的样本大小是所需的总样本大小。k 个样本均值计算假定使用相等组大小。对于三个组，您需要将样本大小除以 3 以获取各个组大小。有关 k 个样本均值的详细信息，请参见《实验设计指南》。

不等组大小

假定您想要设计一个使用大小不等的组的研究。您需要计划一个实验，研究据报道可减少细菌计数的两种治疗方案。您想要将这些治疗方案的效果与未接受任何治疗的控制组的结果进行比较。还想要检测两个治疗组中任一个组的均值与控制组均值之间至少 8 个单位的差值。但是控制组必须是任一个治疗组的两倍大。两个治疗组的大小还必须相等。之前的研究表明误差标准差大约为 5 或 6。

要获取该情形的前瞻功效分析，请创建包含一些基本信息的数据表，如 Bacteria.jmp 样本数据表所示。

图 3.71 Bacteria.jmp 数据表

3/0 列	组	均值	相对大小
1	Control	40	2
2	Treatment A	40	1
3	Treatment B	48	1

- **组**列标识这些组。
- **均值**列反映有必要检测的列间最小差值。在此，假定控制组的均值大约为 40。您希望在任一治疗组的均值至少比控制组均值高出 8 个单位时，检验是显著的。出于此原因，您为两个治疗组中的一个组分配了均值 48，将另一个治疗组的均值设置为等于控制组均值。（或者，您可以为控制组和其中一个治疗组分配均值 0，为剩下的治疗组分配均值 8。）请注意，组均值之间的差值是总体值。
- **相对大小**列显示所需的治疗组相对大小。该列指示控制组需为每个治疗组的两倍大。（或者，您可以根据相对大小准则，为治疗组最初猜想一个大小。）

注意：必须为**相对大小**列分配 “频数”角色。请查看列面板中列名右侧的符号。

下一步，使用“拟合模型”拟合单因子方差分析模型（图 3.72）。请注意，**相对大小**在启动窗口中声明为**频数**。此外，选定“最小报表”重点选项。

图 3.72 细菌研究的 “拟合模型”启动窗口

点击**运行**获取“拟合最小二乘法”报表。该报表显示“均方根误差”和“误差平方和”为 0.0，因为您指定的数据表在各组内没有误差变异。您必须为误差变异输入建议的值范围才能获取功效分析。具体而言，您了解到误差变异大约为 5，但可能大到 6。

1. 点击“效应详细信息”旁边的展开图标，打开该报表。
2. 点击 “组”红色小三角并选择**功效分析**。
3. 要探索科研人员怀疑的误差变异范围，在 σ 下的第一个框中输入 5，在第二个框中输入 6（图 3.73）。
4. 请注意，δ 输入为 3.464102。这是与组均值中的指定差值对应的效应大小。该数据表包含三个隐藏列，用来演示效应大小计算。（请参见第 181 页的 “不平衡单因子布局”。）
5. 要探索一定范围内的研究大小的功效，请在**数目**下的第一个框中输入 16，在第二个框中输入 64，在第三个框中输入增量 4 （图 3.73）。
6. 选择**功效求解**。
7. 点击**完成**。

图 3.73 细菌研究的 “功效详细信息”窗口

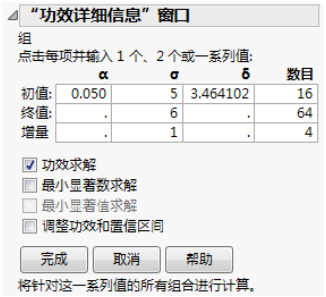


图 3.74 所示的“功效详细信息”报表取代“功效详细信息”窗口。该报表针对 α = 0.05，σ = 5 和 6 的所有组合，以及以 4 为增量从 16 到 64 的样本大小提供功效计算。σ 为 5 时，要获取大约 90% 的功效，您需要总样本大小大约为 32。您需要控制组中有 16 个参与者，每个治疗组中有 8 个参与者。另一方面，若 σ 为 6，则总共需要 44 个参与者。

图 3.74 细菌研究的 “功效详细信息”报表

功效详细信息

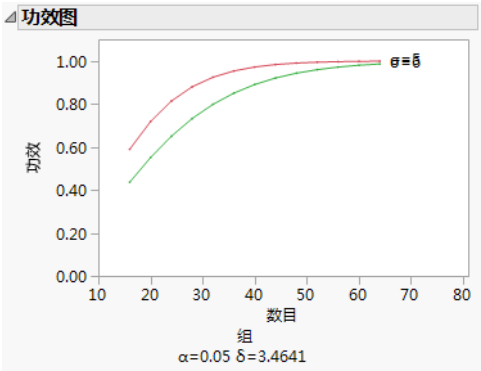
检验组

功效

α	σ	δ	数目	功效
0.0500	5	3.464102	16	0.5894
0.0500	5	3.464102	20	0.7184
0.0500	5	3.464102	24	0.8134
0.0500	5	3.464102	28	0.8799
0.0500	5	3.464102	32	0.9246
0.0500	5	3.464102	36	0.9537
0.0500	5	3.464102	40	0.9721
0.0500	5	3.464102	44	0.9835
0.0500	5	3.464102	48	0.9903
0.0500	5	3.464102	52	0.9944
0.0500	5	3.464102	56	0.9968
0.0500	5	3.464102	60	0.9982
0.0500	5	3.464102	64	0.9990
0.0500	6	3.464102	16	0.4361
0.0500	6	3.464102	20	0.5510
0.0500	6	3.464102	24	0.6501
0.0500	6	3.464102	28	0.7323
0.0500	6	3.464102	32	0.7986
0.0500	6	3.464102	36	0.8506
0.0500	6	3.464102	40	0.8906
0.0500	6	3.464102	44	0.9208
0.0500	6	3.464102	48	0.9433
0.0500	6	3.464102	52	0.9598
0.0500	6	3.464102	56	0.9717
0.0500	6	3.464102	60	0.9803
0.0500	6	3.464102	64	0.9864

点击“功效详细信息”报表中表底部的箭头，获取针对 σ 的两个值的样本大小-功效图，如图 3.75 所示。 在此，红色标记对应于 σ = 5，绿色标记对应于 σ = 6。

图 3.75 细菌研究的 “功效图”



第 4 章

标准最小二乘法示例

分析常用种类的模型

本章提供有关若干标准最小二乘法主题的教学材料的示例。其中包括方差分析、协方差分析、响应曲面模型、裂区设计、随机效应参数估计、纽结样条拟合，以及使用提供 Bayesian 方法的脚本标识活跃因子。

目录

单因子方差分析示例 193
斜率相等的协方差分析示例 195
斜率不等的协方差分析示例 197
响应曲面模型示例 200
    拟合完全响应曲面模型 200
    简化模型 201
    检查“响应曲面”报表 202
    使用预测刻画器查找临界点 204
    使用等高线刻画器查看曲面 205
裂区设计示例 206
随机效应参数估计的示例 210
纽结样条效应示例 211
活跃因子的 Bayes 图示例 214

单因子方差分析示例

在单因子方差分析中，由名义型变量标识的不同组分别拟合不同的均值。要为 JMP 指定模型，选择一个连续响应列和一个名义型效应列。本例使用 Drug.jmp 中的数据。

1. 选择**帮助** > **样本数据库**，然后打开 Drug.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 y 并点击 Y。
4. 选择**药物**，然后点击**添加**。
5. 点击**运行**。

在本例中，**药物**包含三个水平：a、d 和 f。标准最小二乘法拟合方法将该规格转换为线性模型：名义型变量定义一系列指标变量，这些变量的值假定只包含 1、 0 和 –1。线性模型指定如下：

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i$$

其中：

- y_i 是第 i 个观测的观测响应
- x_{1i} 是第 i 个观测对应的第一个指标变量的值
- x_{2i} 是第 i 个观测对应的第二个指标变量的值
- β_0、β_1 和 β_2 分别为截距、第一个指标变量和第二个指标变量的参数。
- ε_i 是相互独立的服从正态分布的误差项

第一个指标变量 x_1 定义如下。请注意，**药物** = a 为指标变量提供了值 1，**药物** = d 提供了值 0，**药物** = f 提供了值 –1：

$$x_{1i} = \begin{cases} 1, & \text{若“药物”} = \text{a} \\ 0, & \text{若“药物”} = \text{d} \\ -1, & \text{若“药物”} = \text{f} \end{cases}$$

第二个指标变量 x_2 具有以下值：

$$x_{2i} = \begin{cases} 0, & \text{若“药物”} = \text{a} \\ 1 & \text{若“药物”} = \text{d} \\ -1, & \text{若“药物”} = \text{f} \end{cases}$$

按照这种参数化，三个水平对应的均值估计值定义如下：

$$\mu_a = \beta_0 + \beta_1$$

$$\mu_d = \beta_0 + \beta_2$$

$$\mu_f = \beta_0 - \beta_1 - \beta_2$$

对 β_i 求解将生成以下结果：

$$\beta_0 = \frac{(\mu_a + \mu_d + \mu_f)}{3} = \mu \quad \text{（各水平下的平均值）}$$

$$\beta_1 = \mu_a - \mu$$

$$\beta_2 = \mu_d - \mu$$

因此，若回归变量编码为每个水平的指标变量减去最后一个水平的指标变量，那么可以将某个水平的参数解释为该水平的响应与所有水平下的平均响应之间的差值。请参见附录第 451 页的“统计详细信息”，获取有关名义型因子的参数解释的更多信息。

图 4.1 显示**药物**效应的“杠杆图”和“最小二乘均值表”。图 4.2 显示药物数据的单因子分析的“参数估计值”和“效应检验”报表。

图 4.1 “药物”的“杠杆图”和“最小二乘均值表”

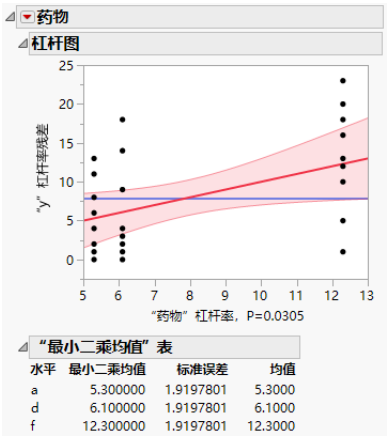


图 4.2 Drug.jmp 的“参数估计值”和“效应检验”

使用最小二乘均值的对比可以更详尽地研究**药物**效应：

1. 点击“药物”红色小三角并选择**最小二乘均值对比**。

2. 点击药物 a 和药物 d 对应的 + 框，并点击药物 f 对应的 - 框，定义药物 a 和药物 d 的平均值与药物 f 之间的对比 （如图 4.3 所示）。
3. 点击**完成**。

图 4.3 药物实验的对比示例

“对比”报表显示药物 f 的最小二乘均值与其他两种药物的最小二乘均值的平均值存在显著差异。

斜率相等的协方差分析示例

包含连续回归量项的方差分析模型称为**协方差分析**。在 Drug.jmp 样本数据表中，列 x 是一个协变量。

协变量向模型方程添加了附加项 x_{3i}。协方差模型的模型分析表示为：

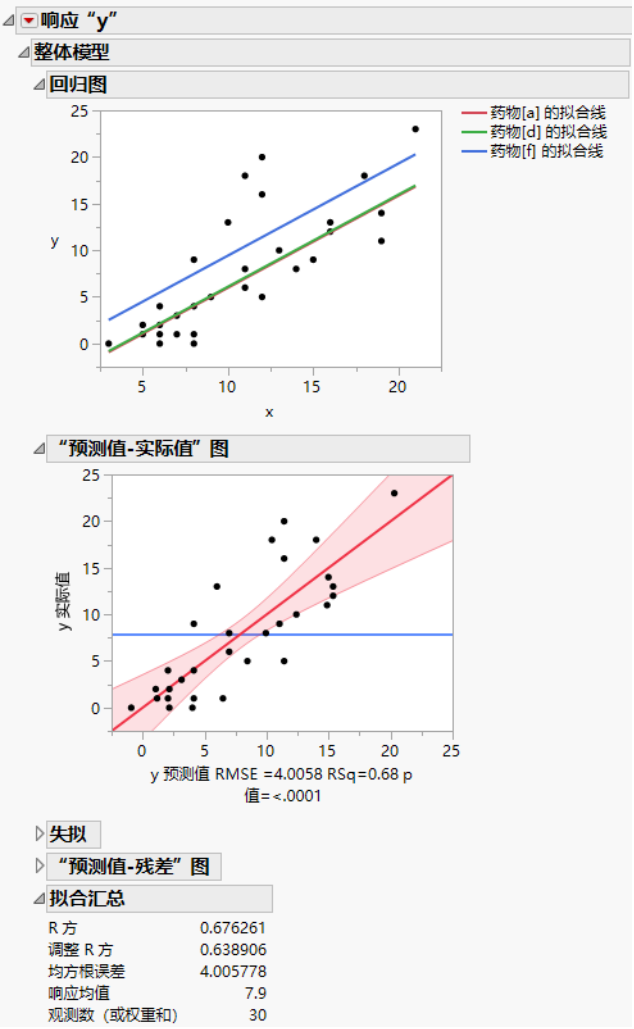
$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \varepsilon_i$$

有两个模型效应：一个是涉及两个参数的名义型主效应，另一个是与一个参数关联的连续协变量。

1. 选择**帮助** > **样本数据库**，然后打开 Drug.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 y 并点击 Y。
4. 选择**药物**和 x，然后点击**添加**。
5. 点击**运行**。

报表中的“回归图”显示您拟合了一个斜率相等的模型（图 4.4）。与主效应模型（仅含**药物**效应）相比，R 方从 22.8% 增长至 67.6%。均方根误差从 6.07 降至 4.0。如图 4.4 所示，整体模型的 *F* 检验显著性概率从 0.03 降至不足 0.0001。

图 4.4 药物数据的协方差分析



药物数据表包含重复的观测。例如，第 1 行和第 9 行都包含**药物** = a 且 x = 11。在拟合模型时，重复的观测可用于构造变异的**纯误差**估计值。可以为未指定函数形式的协变量或名义型效应的交互作用构造另一种误差估计值。这些估计值构成失拟检验的基础。若失拟误差显著，这指示模型未能解释数据中的某个效应。

“失拟”报表显示针对药物数据的这一检验的结果。失拟误差不显著，“概率>F”值 0.7507 证实了这一点。

协变量 x 解释了之前由**药物**变量所解释的响应中的多数变异。因此，尽管该模型的拟合误差少得多，但**药物**效应不再显著。在主效应模型中观测到的**药物**的显著性看起来由协变量解释。

协方差模型中的最小二乘均值不同于普通均值。这是因为这些均值已针对协变量 x 的效应在响应 y 上进行了调整。最小二乘均值是当协变量 x 具有某个中性值时，为**药物**三个水平中的每个水平预测的值。中性值被选作协变量的均值，即 10.7333。

使用 “参数估计值”报表中给出的参数估计值计算最小二乘均值：

预测表达式： -2.696 - 1.185* 药物 [a] - 1.0761* 药物 [d] + 0.98718*x

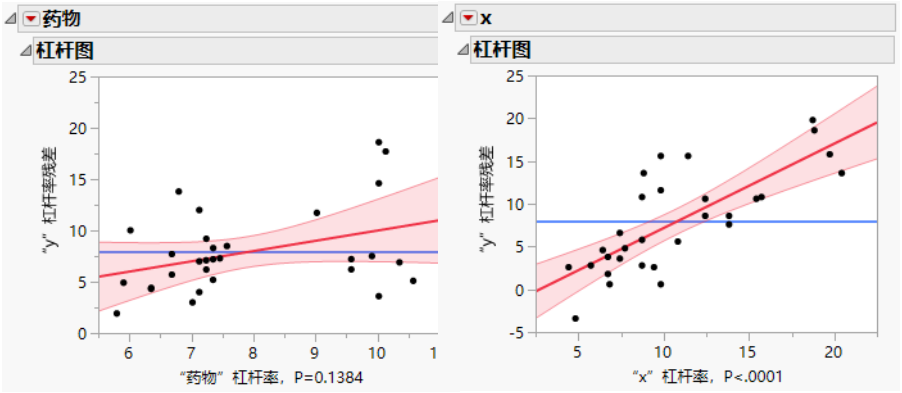
对于 a： -2.696 - 1.185*(1) -1.0761*(0) + 0.98718*(10.7333) = 6.71

对于 d： -2.696 - 1.185*(0) -1.0761*(1) + 0.98718*(10.7333) = 6.82

对于 f： -2.696 - 1.185*(-1) -1.0761*(-1) + 0.98718*(10.7333) = 10.16

图 4.5 显示每个效应的杠杆图。由于协变量显著，**药物**的杠杆率值距离其最小二乘均值有些分散。

图 4.5 比较药物检验数据的杠杆图



斜率不等的协方差分析示例

继续使用 Drug.jmp 样本数据表，本例拟合的模型其协变量的斜率依赖于**药物**的水平。拟合模型后，本例比较在协变量 x 的特定值处**药物**的各个水平的最小二乘均值。

1. 选择**帮助** > **样本数据库**，然后打开 Drug.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 y 并点击 Y。
4. 选择**药物**和 x，然后点击**宏** > **析因次数**。

   这会将最高达到**次数**框中指定的次数的项添加到模型中。**次数**的默认值为 2。因此会将主效应**药物**和 x 及其交互作用**药物** *x 添加到模型效应列表中。

5. 点击**运行**。

这样指定会将两列（分别为 x_{4i} 和 x_{5i}）添加到线性模型，这允许协变量斜率在**药物**不同水平下有所不同。通过将**药物**的指标变量乘以协变量值得到新变量，公式如下：

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \beta_4 x_{4i} + \beta_5 x_{5i} + \varepsilon_i$$

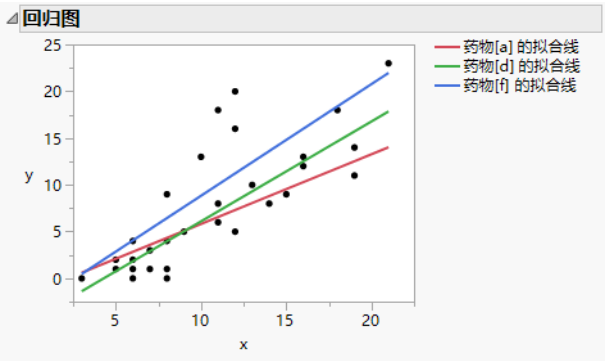
表 4.1 显示该模型的编码。X 的均值为 10.7333。该均值用于使连续项中心化。

表 4.1 具有不同斜率的协方差分析的编码

回归变量	效应	值
X_1	药物[a]	+1（若为 a）、0（若为 d）、–1（若为 f）
X_2	药物[d]	0（若为 a）、+1（若为 d）、–1（若为 f）
X_3	X	X 的值
X_4	药物[a]*(X - 10.733)	X – 10.7333（若为 a）、0（若为 d）、–(X – 10.7333)（若为 f）
X_5	药物[d]*(X - 10.733)	0（若为 a）、X – 10.7333（若为 d）、–(X – 10.7333)（若为 f）

该报表的一部分显示在图 4.6 中。“回归图”显示具有不同斜率的拟合线。“效应检验”报表给出交互作用的 p 值为 0.56。该值并不显著，这指示该模型不需要包含不同斜率。

图 4.6 包含交互作用的图



效应检验

源	参数数目	自由度	平方和	F 比	概率>F
药物	2	2	52.05637	1.5713	0.2284
x	1	1	564.56753	34.0821	<.0001*
药物*x	2	2	19.64465	0.5930	0.5606

执行聚光灯分析

您现在想比较协变量 x 的特定值处**药物**水平的最小二乘均值。在协方差模型的分析中，此类比较有时称为**聚光灯分析**。有关聚光灯分析的详细信息，请参见 Spiller et al. (2013)。

1. 点击“响应‘Y’”红色小三角并选择**估计值 > 多重比较**。
2. 在“多重比较”窗口中，选择**用户定义的估计值**。
3. 在**选择药物水平**下选择所有三个值。
4. 在 x 下的第一个框中，输入 12.5。
5. 点击**添加估计值**。

   这将添加 x = 12.5 处**药物**三个水平的比较。
6. 点击**确定**。

   “用户定义的估计值”报表显示协变量 x 设置为 12.5 时**药物**每个水平的最小二乘均值估计值。“用户定义估计值的多重比较”旁边的红色小三角包含允许您检验估计值差值的选项。

图 4.7 “用户定义的估计值”报表

用户定义估计值的多重比较

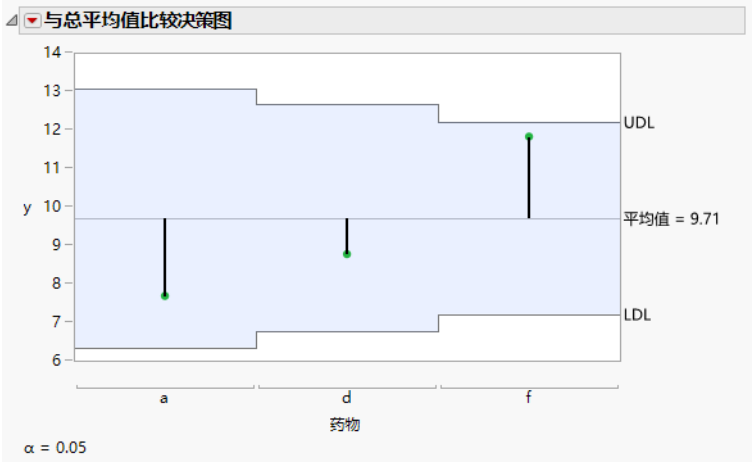
用户定义的估计值

x = 12.5

药物	估计	标准误差	自由度	95% 下限	95% 上限
a	7.684713	1.5772057	24	4.4295208	10.939906
d	8.771371	1.4401229	24	5.7991033	11.743639
f	11.822214	1.2943345	24	9.1508392	14.493590

7. 点击“用户定义估计值的多重比较”旁边的红色小三角并选择**与总平均值比较**。

图 4.8 与总平均值比较决策图



“与总平均值比较”选项为总平均值与三个最小二乘均值的差值创建均值分析 (ANOM) 图。从 ANOM 图，您得出以下结论：在 x = 12.5 处，**药物**对响应没有显著影响。

响应曲面模型示例

本例拟合响应曲面模型。您的目标是将响应最小化。

- “拟合完全响应曲面模型”
- “简化模型”
- “检查“响应曲面”报表”
- “使用预测刻画器查找临界点”
- “使用等高线刻画器查看曲面”

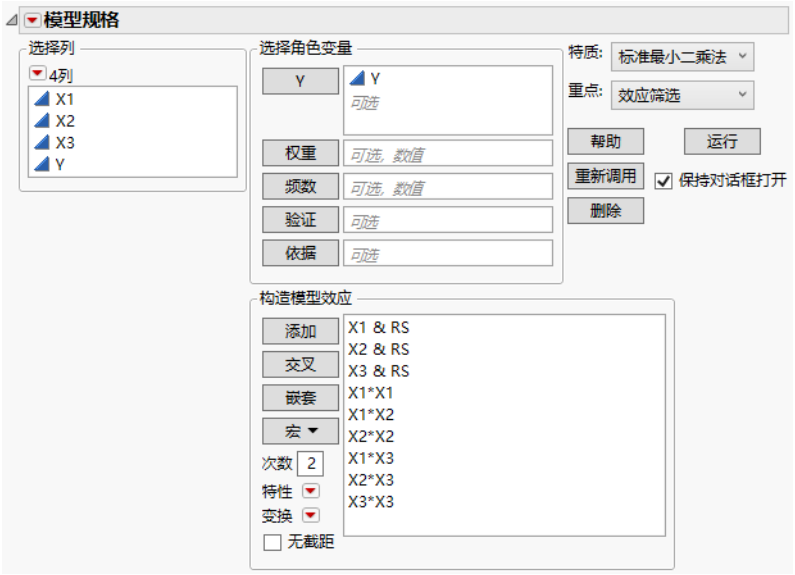
拟合完全响应曲面模型

1. 选择**帮助** > **样本数据库**，然后打开 Design Experiment/Custom RSM.jmp。
2. 选择**分析** > **拟合模型**。

   由于该数据表包含**模型**脚本，因此“模型规格”窗口已按照**模型**脚本中的指定填充。请注意以下几点：

   – 主效应显示在**构造模型效应**列表中，并带有 &RS 后缀，指示已应用“响应曲面”宏。
   – 效应是针对三个预测变量 X1、X2 和 X3 中的完全响应曲面的效应。
   – 由于该模型包含带有 &RS 后缀的项，所以分析结果将包含“响应曲面”报表。

图 4.9 响应曲面分析的 “拟合模型” 启动窗口

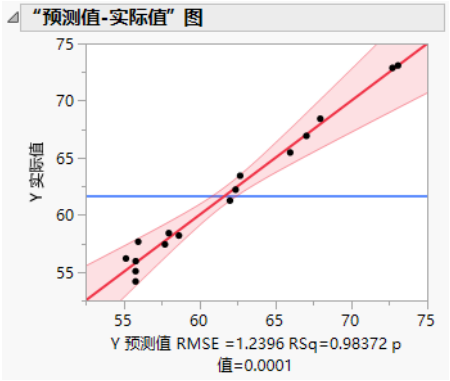


3. 点击运行。

简化模型

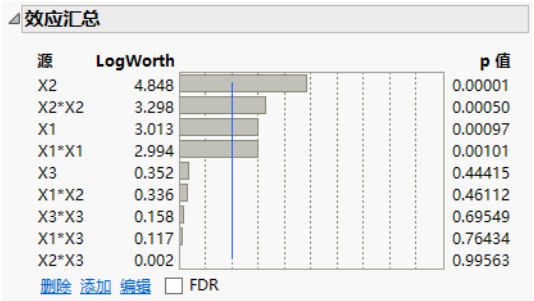
“预测值 - 实际值” 图显示该模型显著。没有证据表明失拟。

图 4.10 完全模型的 “预测值 - 实际值” 图



“效应汇总” 报表表明有一些效应不显著。特别是，X2*X3 是最不显著的效应，其 P 值为 0.99563。接下来，您需要以交互方式使用 “效应汇总” 报表来系统化地简化该模型。

图 4.11 “效应汇总”报表



1. 在“效应汇总”报表中，点击 X2*X3，然后点击**删除**。

   模型随之更新。

   “效应汇总”报表中的 “p 值”列指示 X1*X3 不显著。

2. 点击 X1*X3，然后点击**删除**。

   X3*X3 的 p 值指示其不显著。

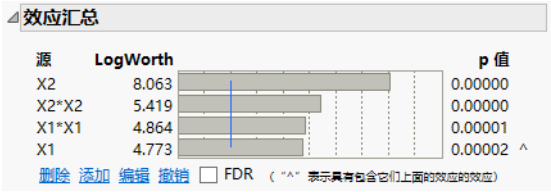
3. 点击 X3*X3，然后点击**删除**。

4. 点击 X1*X2，然后点击**删除**。

   请注意，X3 不显著。它未包含在任何更高阶的效应中，所以您可以删除它，而不会破坏“效应遗传”原则。请参见 “标准最小二乘法模型”一章中第 161 页的 “效应遗传”。

5. 点击 X3，然后点击**删除**。

图 4.12 简化模型后的 “效应汇总”报表

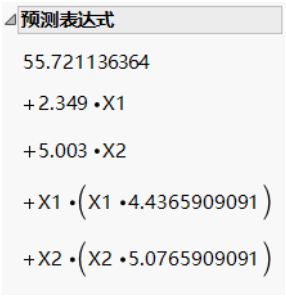


余下的所有效应都显著。

检查“响应曲面”报表

1. 点击“响应‘Y’”红色小三角并选择**估计值 > 显示预测表达式**。

图 4.13 预测表达式



提示：“预测表达式”是预测公式。您还可以通过选择**保存列 > 预测公式**来获取该公式。

在以下步骤中，您可以查阅预测表达式来查看模型系数。

2. 打开 “响应曲面”报表，然后打开 “典型曲率”报表。

图 4.14 “响应曲面”报表

第一个表提供矩阵格式的二阶模型系数。X1*X1 的系数为 4.4365909，X2*X2 的系数为 5.0765909，X1*X2 的系数为 0。线性效应的系数（X1 为 2.349，X2 为 5.003）在标记为 Y 的列中提供。

“解”报表显示临界值。这些值指的是最大值、最小值或鞍点值。在本例中，“解”报表指示响应曲面在临界点处实现了最小值 54.18，其中，X1 = -0.265，X2 = -0.493。

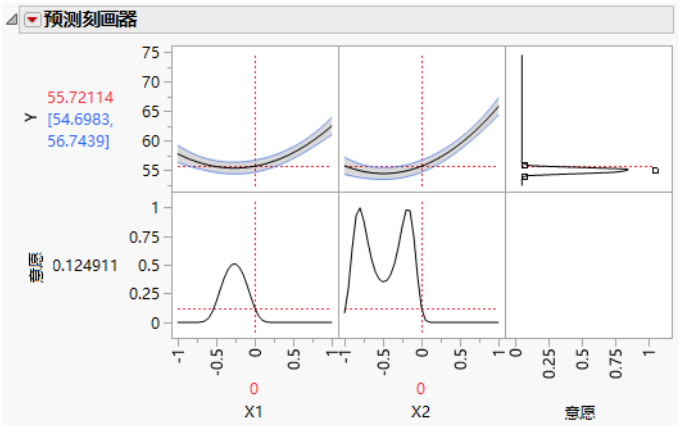
“典型曲率”报表显示二阶参数估计值矩阵的特征结构。特征结构有助于标识曲率的形状和方向。请参见 “标准最小二乘法模型”一章中第 78 页的 “‘典型曲率’报表”。

在本例中，两个特征值均为正，指示曲面达到最小值。最大曲率的方向对应于最大特征值 (5.0766)。该方向由相应的特征向量成分定义。对于第一个方向，X2 （特征向量值为 1.00）确定方向。第二个方向由 X1 确定 （X1 的特征向量值也为 1.00）。

使用预测刻画器查找临界点

“预测刻画器”报表沿着 X1 和 X2 的迹线显示响应曲面的二次行为。由于为 Y 设置了“响应限”列属性，该刻画器还会显示意愿函数。

图 4.15 将“匹配目标值”设置为“目标”的预测刻画器

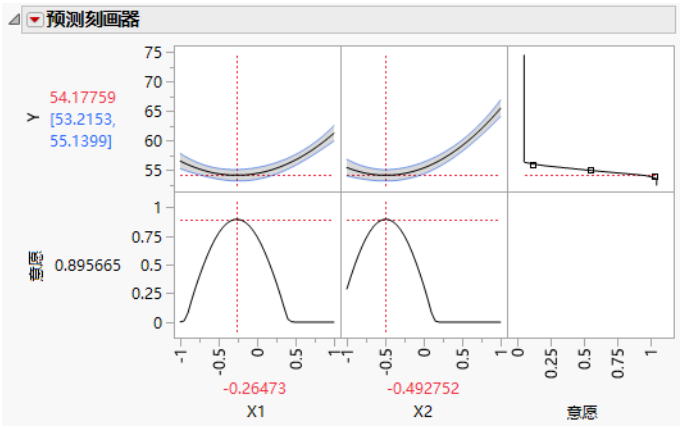


“响应限”列属性设置为“匹配目标值”。但是对于本例，您关注的是最小化 Y 而不是匹配目标。因此，您必须更改设置：

1. 按 Ctrl 键，然后点击预测刻画器的右上单元格。
2. 在“响应目标”对话框中，从选项列表中选择**最小化**。
3. 点击**确定**。

   意愿函数现在即可反映您的目标，即：最小化 Y。
4. 点击“预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 4.16 将“最小化”设置为“目标”并且已最大化意愿的预测刻画器



最小化 Y 的设计区域中的设置显示在该刻画器之下。请注意，这些设置就是 “解” 报表中提供的临界值。

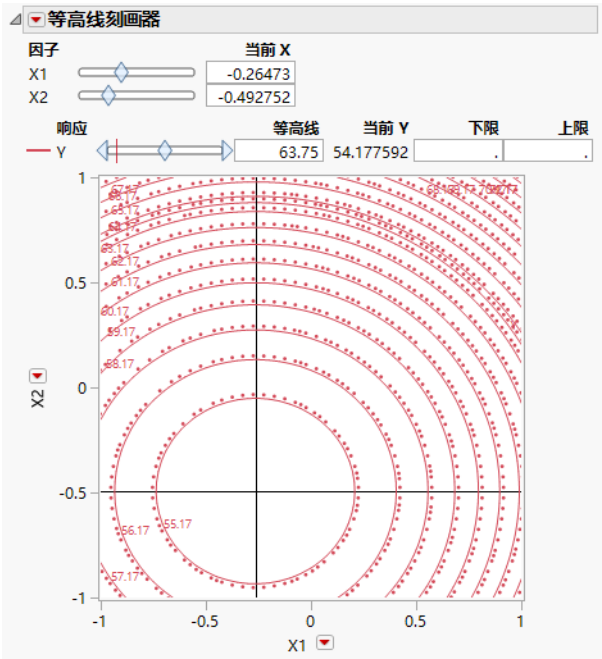
使用等高线刻画器查看曲面

“等高线刻画器” 显示响应曲面的等高线。它提供预测响应在临界点区域内的备选刻画器可视化视图。

1. 点击 “响应 Y” 红色小三角并选择**因子刻画 > 等高线刻画器**。
2. 点击 “等高线刻画器” 红色小三角并选择**等高线网格**。
3. 对于**增量**，键入 1。
4. 点击**确定**。

   按照一个单位的间隔标绘等高线。请参见图 4.17。
5. 点击 “预测刻画器” 红色小三角并选择**因子设置 > 关联刻画器**。

图 4.17 临界点处有十字准线的等高线刻画器

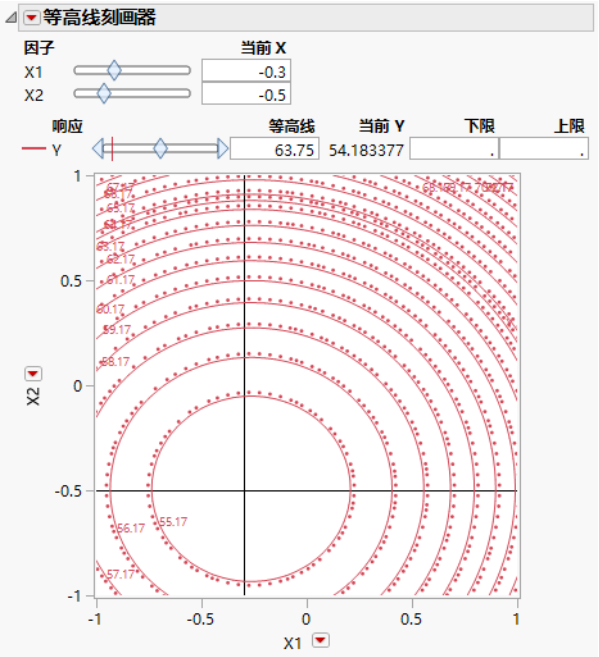


将 “等高线刻画器” 关联到 “预测刻画器” 即可将 “等高线刻画器” 中的**当前 X** 值关联到 “预测刻画器” 中显示的 X 值。“预测刻画器” 中的 X 值提供 Y 最小化时的临界点。“等高线刻画器” 中的十字准线显示临界点。请注意，**当前 Y** 值为 54.177592，即根据 “预测刻画器” 得出的预测最小值。

通常不太可能将过程因子恰好设置为将响应最优化的值。“等高线刻画器”可帮助您标识过程因子的备选设置。在接下来的步骤中，假定您可将过程中的 X1 和 X2 值设置为仅具有一个小数位的精度，那么您的过程设置可以在以上设置中的任一个方向上变动一个小数位。

6. 在 “等高线刻画器”报表中的**当前 X** 下，在 X1 旁边键入 -0.3，在 X2 旁边键入 -0.5。

图 4.18 显示 X1 = -0.3 和 X2 = -0.5 的等高线刻画器



十字准线位于最内层等高线之内，“当前 Y”（“当前 X”设置下的预测 Y 值）为 54.183377，只是略微不同于预测最小值 54.177592。

7. 在 “等高线刻画器”中，点击并拖动十字准线，探索以 X1 = -0.3 和 X2 = -0.5 为中心 0.1 单位半径内的 “当前 X”值。

预测 Y 值全部低于 54.4。实际上，若设置徘徊在最内层等高线内的任意点位置，则预测 Y 会小于等高线值 55.17。

裂区设计示例

随机效应的水平从较大总体的水平中随机选取。为了进行推断，假定随机效应服从均值为零并有一定方差 （称为**方差分量**）的正态分布。

在某种意义上，每个模型都至少有一个随机效应，即构成剩余误差的效应。假定各个观测是从大得多的总体中随机选取的，并且假定误差项的均值为 0 且方差为 σ^2。

最常见的随机效应模型是重复测量或裂区模型。表 4.2 列出了裂区模型中的效应类型。在这些模型中，实验具有两个层次。某些效应应用于实验的整区或对象。那么这些整区会进行划分或对象在不同时间进行测量，而其他效应则应用在这些子单元之内。描述整区或对象的效应为整区效应，子区或重复测量则为子区效应。通常，子单元效应在模型中省略，并被吸收为剩余误差。

表 4.2 裂区模型中的效应类型

裂区模型	效应类型	重复测量模型
整区处理	固定效应	对象间处理
整区 ID	随机效应	对象 ID
子区处理	固定效应	对象内处理
子区 ID	随机效应	重复测量 ID

以上每种情况都可以被视为分层模型，可通过若干传统方法以合理方式拟合它们。可以将该情形视为两个不同的实验：

1. 整区实验以整区或对象为实验单元来构成其误差项。
2. 子区处理以单个测量值为实验单元来构成其误差项 （留作剩余误差）。

较早的检验整区的传统方法是执行以下任一操作：

- 取各测量值的均值，并使用整区效应拟合这些均值。
- 通过将整区均方除以整区 ID 均方得出 *F* 比。
- 组织数据以使裂区或重复测量构成不同列。拟合多元方差分析模型，并使用一元统计方法。

这些方法适用于结构简单并且数据完整平衡的情形。不过，有更为通用的模型适用于任何结构的随机效应。这个更普遍的模型称为**混合模型**，因为它同时包含固定效应和随机效应。

最常见的分层设计类型是平衡裂区，通常采用随时间重复测量的形式。针对某些效应的一个实验单元被细分（有时按时间期间细分），而其他效应则应用到这些子单元。

考虑 Animals.jmp 样本数据表中的数据 （数据是虚构的）。该研究收集了关于狐狸和草原狼的季节性狩猎习惯的差异信息。在某一年的每个季节，阶段性地对三只狐狸和三只草原狼进行了标记和观测。研究中记录了这些动物在一年中的不同季节从其巢穴外出活动的平均英里数 （舍入到最近的英里数）。根据以下方面定义该模型：

- 称为**英里数**的连续响应变量
- 包含值 fox 或 coyote 的**物种**效应
- 包含值 fall、winter、spring 和 summer 的**季节**效应
- 称为**研究课题**的动物标识代码，对 fox（狐狸）和 coyote（草原狼）均使用名义型值 1、2、3

模型包含两个层次：

1. 顶层是对象间层，在该层中相对于对象间变异对作为 fox 或 coyote（**物种**效应）的效应进行检验。
2. 底层是对象内层，在该层中相对于对象内季节变异对四个季节的重复测量因子（**季节**效应）进行检验。对象内变异性反映在剩余误差中。

季节效应可将剩余误差用作其 *F* 统计量的分母。不过，对象间变异性无法通过剩余误差进行测量，必须使用模型中的“研究课题嵌套在物种内”（**研究课题[物种]**）效应捕获该变异性。对象间效应**物种**的 *F* 统计量使用该嵌套效应而不是剩余误差（作为其 *F* 比的分母）。

注意：JMP Pro 用户可以使用“混合模型”特质构造该模型。

要为该数据指定裂区模型，请执行以下步骤：

1. 选择**帮助 > 样本数据库**，然后打开 Animals.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**英里数**并点击 **Y**。
4. 选择**物种**和**研究课题**，然后点击**添加**。
5. 在 “选择列”列表中，选择**物种**。
6. 在 “构造模型效应”列表中，选择**研究课题**。
7. 点击**嵌套**。

   这会向模型添加 “研究课题嵌套在物种内”（**研究课题 [物种]**）效应。
8. 选择嵌套效应**研究课题 [物种]**。
9. 选择**特性 > 随机效应**。

   该嵌套效应现在标识为**物种**效应的误差项，并显示为**研究课题 [物种]**& 随机。
10. 在 “选择列”列表中，选择**季节**并点击**添加**。

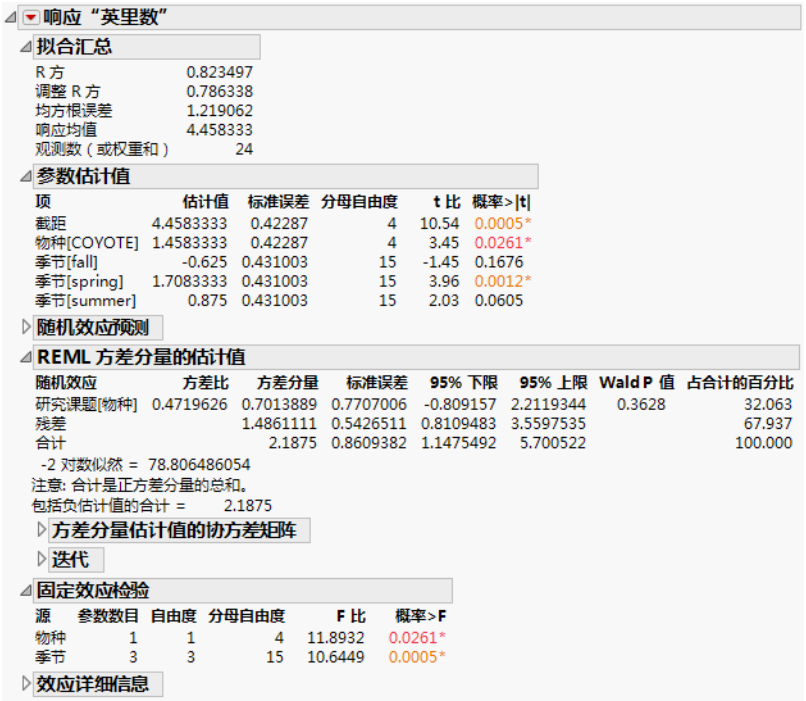
    使用**特性**菜单将某个效应定义为随机效应时，“方法”选项 （REML 和 EMS）会出现在该对话框的右上方。 REML 选项被选作默认选项。填写的启动窗口如图 4.19 中所示。

图 4.19 “拟合模型”对话框

11. 点击**运行**。

该报表如图 4.20 中所示。两个固定效应**物种**和**季节**都显著。“REML 方差分量的估计值”报表提供 “**研究课题**嵌套在**物种**内”和残差方差的估计值。

图 4.20 REML 分析的部分报表



响应“英里数”

拟合汇总

R 方	0.823497
调整 R 方	0.786338
均方根误差	1.219062
响应均值	4.458333
观测数（或权重和）	24

参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|
截距	4.4583333	0.42287	4	10.54	0.0005*
物种[COYOTE]	1.4583333	0.42287	4	3.45	0.0261*
季节[fall]	-0.625	0.431003	15	-1.45	0.1676
季节[spring]	1.7083333	0.431003	15	3.96	0.0012*
季节[summer]	0.875	0.431003	15	2.03	0.0605

随机效应预测

REML 方差分量的估计值

随机效应	方差比	方差分量	标准误差	95% 下限	95% 上限	Wald P 值	占合计的百分比
研究课题[物种]	0.4719626	0.7013889	0.7707006	-0.809157	2.2119344	0.3628	32.063
残差		1.4861111	0.5426511	0.8109483	3.5597535		67.937
合计		2.1875	0.8609382	1.1475492	5.700522		100.000

-2 对数似然 = 78.806486054

注意: 合计是正方差分量的总和。

包括负估计值的合计 = 2.1875

方差分量估计值的协方差矩阵

迭代

固定效应检验

源	参数数目	自由度	分母自由度	F 比	概率>F
物种	1	1	4	11.8932	0.0261*
季节	3	3	15	10.6449	0.0005*

效应详细信息

随机效应参数估计的示例

随机效应具有双重特征。其中一个特征就是它们代表剩余误差，比如与整区实验单元关联的误差。另一个特征则是：和固定效应一样，随机效应的每个水平都关联一个参数。您掌握关于参数的额外信息，这些参数来自具有 0 均值和方差分量估计的方差的正态分布。该额外信息的影响在于：参数的估计值朝着 0 收缩。

与随机效应关联的参数估计值称为 BLUP（最佳线性无偏预测变量）。某些研究人员将这些 BLUP 视为关注的参数，而其他研究人员则将其视为该方法的无关副产品。

BLUP 参数估计值用来估计随机效应的最小二乘均值，因此它们也朝着总均值收缩。收缩度取决于效应方差和效应中每个水平的观测数。对于较大的方差估计值，收缩很小。若方差分量较小，则会出现较多的收缩。若方差分量为 0，效应水平将收缩到恰好为 0。甚至还有可能获得很大的负方差分量，此时的收缩已反转。您可以将固定效应视为随机效应的特例，这种情况下的方差分量非常大。

REML 方法可将每一单个水平的信息与水平间方差的信息进行平衡。若每个水平的观测数较大，则估计值收缩较小。若每个水平的观测数极少，则估计值收缩较大。若有无穷多的观测，则无收缩，估计值与固定效应相同。

假定您有不同棒球选手的击球平均值。不同选手击球表现的方差分量描述不同选手之间在击球平均值上有多大的变异是典型的。假定某个选手只击球几次，并且击球平均值异常小或异常大。那么您往往不会相信该估计值，因为该值基于的打数很少。但若您将该估计值与总均值混合，即：朝着总均值收缩该估计值，那么您会较为信任该估计值。对于有着较长击球记录的选手而言，其收缩比记录较短的选手的收缩要小得多。

您可以自行探索该行为。

1. 选择**帮助 > 样本数据库**，然后打开 Baseball.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**击球**，然后点击 **Y，响应**。
4. 选择**选手**，然后点击**添加**。
5. 选择 “构造模型效应”框中的 “选手”，然后从 “特性”列表中选择**随机效应**。
6. 点击**运行**。

表 4.3 显示使用“REML（推荐）”拟合时“选手”报表中的“最小二乘均值”。同时显示的还有使用 EMS 方法获取的“矩量法”估计值。“矩量法”估计值是普通的**选手**均值。请注意，与有更多打数的其他选手的 REML 估计值相比，仅有三次打数的 Suarez 的 REML 估计值朝着总均值收缩的幅度更大。

表 4.3 矩量法与 REML 之间的估计值比较

	矩量法	REML	数目
方差分量	0.01765	0.019648	
Anderson	0.29500000	0.29640407	6
Jones	0.20227273	0.20389793	11
Mitchell	0.32333333	0.32426295	6
Rodriguez	0.55000000	0.54713393	6
Smith	0.35681818	0.35702094	11
Suarez	0.55000000	0.54436227	3
最小二乘均值	与普通均值相同	从均值收缩	

纽结样条效应示例

使用**纽结样条效应**选项可通过平滑多项式的一段来拟合一个指定的效应。选择该特性后将显示一个窗口，该窗口支持您指定结点数。（仅针对主效应连续项应用纽结样条。）

JMP 遵循文献中关于如何放置点的建议。纽结样条亦称 Stone 样条或 Stone-Koo 样条。请参见 Stone and Koo (1985)。若点数未超过 100 个，第一个结点和最后一个结点分别为第五个最小点和第五个最大点。另一方面，对于未超过 5 个结点的情况，第一个结点和最后一个结点分别位于第 0.05 和 0.95 分位数；对于 5 个以上结点的情况，第一个结点和最后一个结点分别位于第 0.025 和 0.975 分位数。默认结点数为 5，除非有不到 30 个点，此时默认为 3 个结点。

与平滑样条相比，纽结样条具有以下属性：

- 纽结样条适用于包含很多项的一般模型，而平滑样条仅适用于二元回归。
- 回归算法不是基于响应的函数。
- 纽结样条很简化，对于 k 个结点的曲率仅添加 $k-2$ 个项。
- 纽结样条比纯多项式更为保守，这是因为纽结样条在数据范围之外的外推为直线而不是多项式。
- 有一个简单的曲率检验。

使用纽结样条效应检验曲率的示例

要检验曲率，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Growth.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**比率**列并点击 **Y**。
4. 选择**年龄**列并点击**添加**。
5. 选择**效应**窗格中的**年龄**，然后选择**特性** > **纽结样条效应**。
6. 对于结点数，请键入 5 并点击**确定**。

图 4.21 “拟合模型”启动窗口

7. 点击**运行**。
8. 点击 “响应 ‘比率’”红色小三角并选择**估计值 > 定制检验**。
   在 “定制检验”报表中，请注意只有一列，而您需要三列。
9. 点击两次**添加列**按钮，生成总共三列。
10. 在第一列中，为**年龄 & 已纽结 @4.5** 键入 1。
11. 在第二列中，为**年龄 & 已纽结 @20.25** 键入 1。
12. 在第三列中，为**年龄 & 已纽结 @36** 键入 1。

图 4.22 曲率定制检验的值

13. 点击**完成**。
    “定制检验”报表中的较小 “概率 > F”值指示存在曲率。

图 4.23 “曲率”报表

定制检验

参数			
截距	0	0	0
年龄&已纽结	0	0	0
年龄&已纽结@4.5	1	0	0
年龄&已纽结@20.25	0	1	0
年龄&已纽结@36	0	0	1
=	0	0	0
值	-1.436874e-5	0.0000365148	-0.000033071
标准误差	1.9803969e-6	6.0647971e-6	8.0723366e-6
t 比	-7.255483783	6.020782207	-4.096860562
概率>|t|	5.275629e-10	8.1625874e-8	0.0001152566
平方和	0.0583271224	0.0401646174	0.0185968833

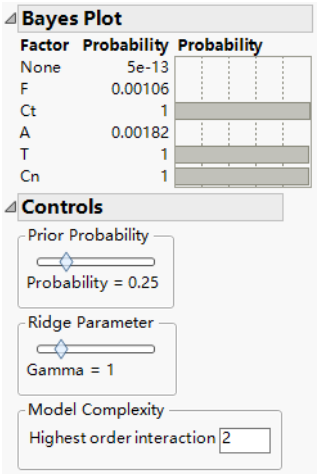
平方和	0.112524874
分子自由度	3
F 比	33.852401355
概率>F	1.955866e-13

活跃因子的 Bayes 图示例

假定您执行了一项实验设计并且想要确定哪些因子是活跃的。您可以使用 JMP 通过多种方式来解决该问题。本例演示了一个脚本，可使用该脚本通过 Bayes 方法来标识活跃因子。

1. 选择**帮助** > **样本数据库**，然后打开 Reactor.jmp。
2. 在 Samples > Scripts 文件夹中，打开 BayesPlotforFactors.jsl 样本脚本。
3. 选择**编辑** > **运行脚本**。
4. 选择 Y 并点击 **Y，响应**。
5. 选择 F、 Ct、 A、 T 和 Cn，然后点击 **X，因子**。
6. 点击**确定**。

图 4.24 因子活动的 Bayes 图



“模型复杂性”指示要考虑的最高阶交互作用为 2。因此，因此构造了包含最高达到二阶的交互作用的所有可能模型。根据为“先验概率”分配的值，为每个可能模型计算后验概率。某个因子的概率就是涉及该因子的每个模型的概率之和。

该方法将 Ct、T 和 Cn 标识为活跃因子，将 A 和 F 标识为不活跃因子。

若岭参数为 0（不允许），则所有模型都可以用最小二乘法拟合。随着岭参数增大，任何模型的参数估计值都朝着 0 收缩。有关岭参数以及该参数为何不能为零的详细信息，请参见 Box and Meyer (1993)。

第 5 章

逐步回归模型

使用变量选择查找模型

逐步回归是为回归模型选择效应子集的方法。它在以下情况下很有用：

- 用于指导选择模型项的理论很少。
- 您想以交互方式探索哪些预测变量可提供良好的拟合效果。
- 您想减小由于估计不必要的项导致的方差，以此提高模型的预测性能。

对于分类预测变量，您可以执行以下操作：

- 从各种规则中进行选择，以确定相关项如何进入模型。
- 强制效应遗传。

使用“逐步”平台还可以探索所有可能的模型以及执行模型平均。

目录

逐步回归概述 219
使用逐步回归的示例 219
“逐步”报表 221
“逐步”平台选项 221
“逐步回归控制”面板 222
“当前估计值”报表 227
“步进历史记录”报表 228
具有交叉、交互作用或多项式项的模型 229
合并规则的示例 229
具有名义型和有序型效应的模型 231
分层项的构造 231
具有名义型项的模型示例 232
分层项的限制规则的示例 235
执行二值和有序型 Logistic 逐步回归 237
使用 Logistic 逐步回归的示例 238
“所有可能的模型”选项 239
使用“所有可能的模型”选项的示例 239
“模型平均”选项 241
使用“模型平均”选项的示例 242
逐步回归中的验证选项 243
逐步回归中具有两个或三个值的验证集 243
逐步回归中的 K 重交叉验证 245

逐步回归概述

在 JMP 中，逐步回归是“拟合模型”平台的特质。**逐步**功能计算出与其他最小二乘法平台相同的估计值，但是它便于在很多模型中搜索和选择。

该方法有一些副作用，您需要注意。所选模型的统计量的显著性水平违反了标准统计假设，因为只是在一个固定模型中选择了该模型但是没有检验该模型。该方法值得肯定的一点是30 年来它一直被人们用于减少项数。Subset Selection in Regression (Miller 1990) 一书为模型选择统计量带来统计意义。

在承认真实的显著性概率可能与报告的概率相差甚远的情形下，本章暂且以机械方式使用术语**显著性概率**以表示计算在固定模型中是有效的。

使用逐步回归的示例

Fitness.jmp 数据表包含一个有氧健身研究的结果。可以使用特殊的测试来评估有氧健身的效果，该测试对在跑步机上跑完规定距离的人员测量吸氧量。但是若能找到一个使用更简单的测量方法来评估健身效果和预测吸氧量，那就更为经济实惠。为了确定这样的方程，我们对跑了 1.5 英里的 31 名参与者进行了测量，记录了他们的年龄、体重、跑步时间和脉搏。

注意：为了便于演示，我们更改了某些**最大脉搏**和**跑步时脉搏**的值，它们和 Rawlings (1988, p.105) 采用的数据有所不同。

1. 选择**帮助** > **样本数据库**，然后打开 Fitness.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**吸氧量**并点击 Y。
4. 选择**体重**、**跑步时间**、**跑步时脉搏**、**休息时脉搏**、**最大脉搏**，然后点击**添加**。
5. 对于**特质**，选择**逐步**。

图 5.1 完成的 “拟合模型”启动窗口

JMP PRO 验证仅在 JMP Pro 中可用。

6. 点击运行。

图 5.2 “逐步”报表窗口

要找到好的吸氧量预测方程，您需要比较很多不同的回归模型。使用“逐步”报表窗口中的选项来搜索具有不同效应组合的模型并选择所需的模型。

“逐步”报表

“逐步”报表窗口包含以下元素：

平台选项 “逐步拟合”红色小三角菜单包含影响所有变量的选项。请参见第 221 页的 “ “逐步”平台选项”。

逐步回归控制 一些控件，用于限制回归变量效应概率、确定选择效应的方法、开始或停止选择过程以及创建模型。请参见第 222 页的 ““逐步回归控制”面板”。

当前估计值 一个报表，该报表支持您在模型中添加、删除和锁定效应。请参见第 227 页的 ““当前估计值”报表”。

步进历史记录 一个报表，用于记录将项添加到模型的影响。请参见第 228 页的 ““步进历史记录”报表”。

“逐步”平台选项

“逐步拟合”红色小三角菜单包含以下选项：

K 重交叉验证 （仅可用于连续响应。）在选择过程中执行 K 重交叉验证。选中该选项时，该选项将启用 “最大 K 重 R 方”停止规则 （第 222 页的 ““逐步回归控制”面板”）。有关验证的详细信息，请参见第 243 页的 “逐步回归中的验证选项”。

所有可能的模型 （仅可用于连续响应。）拟合最多达到指定限值的所有可能的模型，并显示每个项数对应的最佳模型。输入要在任意一个模型中拟合的最大项数的值。还输入对应每个项数要显示所得到的最佳模型数目的最大值。使用指标变量表示分类变量。请参见第 231 页的 “具有名义型和有序型效应的模型”。您可以将显示的模型限制为满足强效应遗传的那些模型。请参见第 239 页的 ““所有可能的模型”选项”。

模型平均 （仅可用于连续响应。）支持您平均若干模型的拟合结果，而非选择一个最佳模型。请参见第 241 页的 ““模型平均”选项”。

标绘准则历史 创建 AICc 和 BIC 随参数数目变化的图。“准则历史”图包含两个着色区域。将最小 AICc 值定义为 $V^{最佳}$。绿色区域通过范围 [$V^{最佳}$, $V^{最佳}$ +4] 来定义。黄色区域通过范围 [$V^{最佳}$ +4, $V^{最佳}$ +10] 来定义。

标绘 R 方历史 （仅可用于连续响应。）创建训练和验证 R 方随参数数目变化的图。

清除历史记录 清除并重置步进历史记录。

JMP PRO **带验证导出模型** （仅当您在“逐步”启动窗口中输入了“验证”列时才可用。）选择“构建模型”时向“模型规格”窗口添加“验证”列。选择“运行模型”时运行带“验证”列的模型。

默认选定该选项。

模型对话框 为当前窗口显示完成的 “拟合模型”启动窗口。

“逐步回归控制”面板

使用“逐步回归控制”面板可限制回归变量效应概率，确定选择效应的方法，开始或停止选择过程以及运行模型。“执行”按钮下会显示一条注释，指示您是否有排除或缺失行。

图 5.3 “逐步回归控制”面板

停止规则

“停止规则”确定选择哪个模型。对于除“P 值阈值”之外的所有停止规则，只允许“前进”和“后退”方向。使用验证的停止规则只有“最大验证 R 方”和“最大 K 重 R 方”。请参见第 243 页的“逐步回归中的验证选项”。

p 值阈值 使用 *p* 值（显著性水平）使效应进入模型和从模型中删除效应。选择“P 值阈值”时将显示其他两个选项：

进入的概率 指定在前进步中效应必须进入模型的最大 *p* 值。

剔除的概率 指定在后退步中必须从模型删除效应的最小 *p* 值。

最小 AICc 使用最小校正 Akaike 信息准则来选择最佳模型。有关更多详细信息，请参见“统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

最小 BIC 使用最小 Bayesian 信息准则来选择最佳模型。有关更多详细信息，请参见“统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

JMP PRO **最大验证 R 方** 使用验证集中的最大 R 方来选择最佳模型。仅当您使用具有两个或三个非重复值的验证列时，该选项才可用。有关验证的详细信息，请参见第 243 页的“逐步回归中具有两个或三个值的验证集”。

最大 K 重 R 方 使用 K 重交叉验证中的最大 R 方来选择最佳模型。您可以通过从“逐步”红色小三角菜单中选择该选项来访问“最大 K 重 R 方”停止规则。 JMP Pro 用户可使用具有四个或更多值的验证集来访问该选项。选择该选项时，系统要求您指定折数。有关验证的详细信息，请参见第 245 页的“逐步回归中的 K 重交叉验证”。

方向

您选择的方向控制效应如何进入模型和从模型中剔除。选择以下选项之一：

前进 使 *p* 值最小的项进入。若选择了 “p 值阈值”停止规则，该项在**进入的概率**所指定的水平下必须是显著的。请参见第 225 页的 “向前选择示例”。

后退 删除 *p* 值最大的项。若选择了 “p 值阈值”停止规则，该项在**剔除的概率**所指定的水平下不能是显著的。请参见第 226 页的 “向后选择示例”。

注意：将“后退”选作“方向”时，您必须在点击“执行”或“步进”之前点击“全部进入”。

混合 仅当选择了 “p 值停止规则”时才可用。它使前进步和后退步交替出现。它包含满足**进入的概率**的最显著的项，删除满足**剔除的概率**的最不显著的项。它继续删除项，直到其余项是显著的，然后变为前进方向。

“执行”、“停止”、“步进”按钮

使用“执行”、“停止”和“步进”按钮，您可以控制如何使项进入模型或从模型中删除项。

注意：所有停止规则仅考虑由 *p* 值进入（“前进”方向）或删除（“后退”方向）定义的模型。停止规则不考虑所有可能的模型。

执行 自动执行进入（“前进”方向）或删除（“后退”方向）项的过程。在拟合的模型中，基于所选停止规则认定是最佳的模型最后列出。除了 “p 值阈值”停止规则外，选择为 “最佳”的模型是在停止规则统计量行为中忽略局部下沉的模型。“最佳”模型选择右侧的按钮选择该模型提供给 “构建模型”和 “运行模型”选项，但是您可以自由更改该选择。

- 对于“p 值阈值”，最佳模型基于“进入的概率”和“剔除的概率”准则。请参见第 222 页的 “p 值阈值”。
- 对于“最小 AICc”和“最小 BIC”，继续执行自动拟合，直到找到最佳模型。最佳模型是具有最小 AICc 或 BIC 的模型，该模型可能分别后跟具有更大 AICc 或 BIC 值的多达 10 个模型。该模型通过“参数”列中的“最佳”和“操作”列中的“特定”这两项来指定。
- 对于“最大验证 R 方”（仅限 JMP Pro）和“最大 K 重 R 方”，继续执行自动拟合，直到找到最佳模型。最佳模型是具有“最大验证 R 方”或“最大 K 重 R 方”值的模型，该模型可能分别后跟具有更小“验证 R 方”或“K 重 R 方”值的 10 个模型。该模型通过“参数”列中的“最佳”和“操作”列中的“特定”这两项来指定。

停止 停止由 “执行”按钮开始的自动选择过程。

步进 在 “前进”方向中逐个进入项或在 “后退”方向中逐个删除项。在任何点，您均可以通过点击“步进历史记录”报表中的右侧按钮来选择模型。模型项的选择将在“当前估计值”报表中更新。这是您一点击 “构建模型”或 “运行模型”即使用的模型。

规则

注意：仅当您的模型包含相关项时才显示。当您具有名义型或有序型变量时，会构造相关项并且它们显示在“当前估计值”表中。

使用 “规则”来更改模型中有项的层次结构时应用的规则。在以下情况下会出现层次结构：

- 一个变量是另一变量的成分时出现层次结构。例如，若您的模型包含变量 A、B 和 A*B，则 A 和 B 在层次结构中是 A*B 的**前**项。
- 包含名义型或有序型变量时也出现层次结构。在树结构中位于另一项之上的项是**前**项。请参见第 231 页的“分层项的构造”。

选择以下选项之一：

合并 在考虑输入具有前项的项时，计算两个不同检验的 p 值。第一个 p 值 p_1 的计算方式是：将该项与其前项分组在一起并以联合 F 检验计算该组进入的显著性概率。第二个 p 值 p_2 是在前项已经进入模型后，检验该项进入的显著性概率的结果。具有前项的项进入的最终显著性概率为 $\max(p_1, p_2)$。

提示：“合并”规则避免包括非显著交互作用项，其前项可能具有特别强烈的效应。在这种情况下，强主效应可能会使该组进入的显著性概率 p_1 非常小。不过，第二个检验发现交互作用自身不显著。最终，p_2 很大，用作最终的进入显著性概率。

警告：具有前项的项的自由度值取决于两个进入显著性概率中哪一个比较大。用于最终的进入显著性概率的检验确定 “当前估计值”表中的自由度 nDF。因此，若使用 p_1，则 nDF 为联合检验的组中的项数；若使用 p_2，则 nDF 等于 1。

“合并”选项是默认规则。请参见第 229 页的 “具有交叉、交互作用或多项式项的模型”。

限制 限制具有前项的项，使它们在其前项进入前无法进入。请参见第 231 页的 “具有名义型和有序型效应的模型”和第 235 页的 “分层项的限制规则的示例”。

无规则 使选择例程能自由选择项，无论该例程是否破坏层次结构。

整体效应 当涉及该效应的项显著时，仅使整体效应进入。仅当具有两个以上水平的分类变量作为可能的模型效应进入时该规则才适用。请参见第 235 页的 “规则”。

按钮

“逐步控制”面板包含以下按钮：

执行 将选择过程自动执行完成。

停止 停止选择过程。

步进 一次使选择过程增加一步。

箭头按钮 在选择过程中前进和后退一步。

全部进入 使所有未锁定的项进入模型。

全部删除 从模型中删除所有未锁定的项。

构建模型 根据当前在 “当前估计值”表中显示的模型为 “拟合模型”窗口创建模型。有名义型或有序型项时，**构建模型**将创建临时变换列以包含模型所需的项。

运行模型 运行当前在 “当前估计值”表中显示的模型。有名义型或有序型项时，**运行模型**将创建临时变换列以包含模型所需的项。

统计量

以下统计量显示在 “逐步回归控制”面板下方。

误差平方和 当前模型的误差平方和。

误差自由度 当前模型的误差自由度。

RMSE 当前模型的均方根误差 （残差）。

R 方 响应中可归因于模型中的项而非随机误差的变异比例。

调整 R 方 通过在计算中使用自由度，调整 R^2 使该值在带有不同数量参数的各个模型之间更具可比性。在逐步过程中调整 R^2 很有用，因为您正在查看很多不同模型并且想要根据模型中的项数进行调整。

Cp 用于选择模型的 Mallow C_p 准则。它是误差平方和的替代测度，可以按以下方式定义：

$$C_p = \left(\frac{\text{SSE}_p}{s^2}\right) - (N - 2p)$$

其中 s^2 是完全模型的 MSE，SSE_p 是具有 p 个变量（包括截距）的模型的误差平方和。请注意，p 是 x 变量数 +1。若使用 p 对 C_p 绘图，Mallows (1973) 建议选择 C_p 首次接近 p 的模型。

p 模型中的参数数目，包括截距。

AICc 校正 Akaike 信息准则。有关更多详细信息，请参见 “统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

BIC Bayesian 信息准则。有关更多详细信息，请参见“统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

向前选择示例

在向前选择中，项进入模型并添加最显著的项，直到所有项都显著后结束。

1. 完成第 219 页的 “使用逐步回归的示例”中的步骤。

   请注意**方向**的默认选择是 “前进”。

2. 点击**步进**。

   在图 5.4 中，您可以看到在执行一步后，最显著的项**跑步时间**进入模型。

3. 点击**执行**。

   在图 5.5 中，显示已添加除**休息时脉搏**和**体重**之外的所有项。

图 5.4 向前选择一步后的“当前估计值”表

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	82.4217727	1	0	0.000	1
☐	☐	体重	0	1	1.323628	0.171	0.68267
☐	☑	跑步时间	-3.3105554	1	632.9001	84.008	4.6e-10
☐	☐	跑步时脉搏	0	1	15.36208	2.118	0.15673
☐	☐	休息时脉搏	0	1	0.130138	0.017	0.89814
☐	☐	最大脉搏	0	1	1.567361	0.202	0.65632

图 5.5 向前选择三步后的“当前估计值”表

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	80.9007896	1	0	0.000	1
☐	☐	体重	0	1	4.989591	0.827	0.37137
☐	☑	跑步时间	-2.9701867	1	443.2028	73.971	3.25e-9
☐	☑	跑步时脉搏	-0.3751142	1	55.14175	9.203	0.00529
☐	☐	休息时脉搏	0	1	0.350744	0.056	0.81399
☐	☑	最大脉搏	0.35421891	1	41.34703	6.901	0.01403

向后选择示例

在向后选择中，所有项进入模型，之后删除最不显著的项，直到剩余的所有项都显著后结束。

1. 完成第 219 页的“使用逐步回归的示例”中的步骤。
2. 点击**全部进入**。

图 5.6 所有效应都进入模型

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	82.3936054	1	0	0.000	1
☐	☑	体重	-0.0509071	1	4.83788	0.772	0.38784
☐	☑	跑步时间	-2.9518165	1	366.3375	58.489	5.29e-8
☐	☑	跑步时脉搏	-0.3970425	1	59.51519	9.502	0.00495
☐	☑	休息时脉搏	0.01239004	1	0.199033	0.032	0.85995
☐	☑	最大脉搏	0.38479281	1	45.83023	7.317	0.01212

3. 对于**方向**，选择“后退”。
4. 点击**步进**两次。

   第一个后退步删除**休息时脉搏**，第二个后退步删除**体重**。

图 5.7 已删除项的 “当前估计值”和 “步进历史记录”表

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	80.9007896	1	0	0.000	1
☐	☐	体重	0	1	4.989591	0.827	0.37137
☐	☑	跑步时间	-2.9701867	1	443.2028	73.971	3.25e-9
☐	☑	跑步时脉搏	-0.3751142	1	55.14175	9.203	0.00529
☐	☐	休息时脉搏	0	1	0.350744	0.056	0.81399
☐	☑	最大脉搏	0.35421891	1	41.34703	6.901	0.01403

步进历史记录

步进	参数	操作	"显著性概率"	序贯平方和	R 方	Cp	p	AICc	BIC	
1	全部	已进入	.	.	0.8161	6	6	157.051	162.22	○
2	休息时脉搏	已删除	0.8600	0.199033	0.8158	4.0318	5	153.721	158.825	○
3	体重	已删除	0.3714	4.989591	0.8100	2.8284	4	151.592	156.362	◉

图 5.7 中所示的“当前估计值”和“步进历史记录”表汇总了后退逐步选择过程。请注意“步进历史记录”表中第三步的 BIC 值 156.362。若再次点击“步进”从模型中删除另一个参数，BIC 值将增至 159.984。出于此原因，您可以选择步骤 3 模型。这也是按下“执行”按钮后生成的模型。

“当前估计值”报表

使用 “当前估计值”报表使效应进入模型、删除效应或在模型中锁定效应。（截距永远锁定到模型中。）该报表包含以下列：

图 5.8 “当前估计值”表

误差平方和	误差自由度	RMSE	R 方	调整 R 方	Cp	p	AICc	BIC
851.38154	30	5.3272305	0.0000	0.0000	106.93073	1	195.1018	197.5412

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	47.3758065	1	0	0.000	1
☐	☐	体重	0	1	22.55181	0.789	0.38169
☐	☐	跑步时间	0	1	632.9001	84.008	4.6e-10
☐	☐	跑步时脉搏	0	1	134.8447	5.457	0.0266
☐	☐	休息时脉搏	0	1	135.7828	5.503	0.02604
☐	☐	最大脉搏	0	1	47.71646	1.722	0.19975

锁定 锁定一个项在模型之内或之外。选中的项不能进入模型或从模型中删除。

已进入 指示某个项当前是否在模型中。您可以点击项的复选框来手动使效应进入或离开模型。

参数 效应名称。

估计值 当前参数估计值 （若效应当前不在模型中，则为零）。

自由度 项的自由度数。若某个项进入模型也强制其他项进入模型，则该项的自由度大于 1。

Wald/ 得分卡方 （仅在响应为分类响应时才显示。）模型中每个项的检验统计量。对于模型中尚未包含的项，检验统计量基于将该项包含在模型中的得分检验。对于已经包含在模型中的项，检验统计量基于将该项从模型中移除的 Wald 检验。

“显著性概率” （仅在响应为分类响应时才显示。）与基于 nDF 自由度的 Wald/ 得分卡方检验统计量关联的 *p* 值。“显著性概率”用于确定将包含在模型中的下一项。

平方和 （仅在响应为连续响应时才显示。）若项进入模型则导致误差 （残差）平方和 (SS) 减小，若从模型中删除项则导致误差 SS 增加。若以某种方式限制项，则可能报告其 SS 为零。

“F 比” （仅在响应为连续响应时才显示。）要检验项效应为零的传统检验统计量。它是 *t* 比的平方。它包含在引号中，因为它没有用于检验项的 *F* 分布 （因为在拟合时已选择模型）。

“概率 >F” （仅在响应为连续响应时才显示。）与 *F* 统计量关联的 *p* 值。与“F 比”一样，它包含在引号中，因为不相信它是真实的显著性概率。

R 与模型中的其他效应多重相关。仅当您在报表中右击并选择“列”>“R”时才显示该列。

“步进历史记录”报表

随着每步的执行，“步进历史记录”报表记录了将某个项添加到模型的影响。例如，Fitness.jmp 示例的“步进历史记录”报表显示项进入模型的顺序以及显示每个模型的统计量。请参见第 219 页的“使用逐步回归的示例”。

使用右侧的单选按钮选择模型。

图 5.9 “步进历史记录”报表

步进历史记录

步进	参数	操作	"显著性概率"	序贯平方和	R 方	Cp	p	AICc	BIC	
1	跑步时间	已进入	0.0000	632.9001	0.7434	7.8825	2	155.397	158.81	○
2	跑步时脉搏	已进入	0.1567	15.36208	0.7614	7.4298	3	155.787	159.984	○
3	最大脉搏	已进入	0.0140	41.34703	0.8100	2.8284	4	151.592	156.362	◉

分类响应的步进历史记录

具有分类响应的模型的“步进历史记录”报表包含额外的四列：L-R 卡方、“显著性概率”、条目卡方、条目“显著性概率”。

“L-R 卡方”列和“显著性概率”列包含完全似然比与简化似然比检验统计量以及 *p* 值。在此处，完全模型是包含指定项的模型，简化模型则不包含指定项。

“条目卡方”和“条目‘显著性概率’”列包含“Wald/得分卡方”和“显著性概率”值，这些值用于选择要包含在模型中的最近的项。

图 5.10 第一步前分类响应的步进历史记录

当前估计值

锁定	已进入	参数	估计值	自由度	Wald/得分卡方	"显著性概率"
☑	☑	截距[M]	-0.0645385	1	0	1
☐	☐	吸氧量	0	1	7.275453	0.00699
☐	☐	体重	0	1	10.13425	0.00146
☐	☐	跑步时间	0	1	5.764797	0.01635
☐	☐	跑步时脉搏	0	1	1.865024	0.17205
☐	☐	休息时脉搏	0	1	2.693096	0.10078
☐	☐	最大脉搏	0	1	1.467541	0.22573

步进历史记录

步进	参数	操作	似然比卡方	"显著性概率"	条目卡方	条目 "显著性概率"	R 方	p	AICc	BIC

图 5.11 第一步后分类响应的步进历史记录

当前估计值

锁定	已进入	参数	估计值	自由度	Wald/得分卡方	"显著性概率"
☑	☑	截距[M]	-16.228294	1	0	1
☐	☐	吸氧量	0	1	8.750599	0.0031
☐	☑	体重	0.20792272	1	6.994297	0.00818
☐	☐	跑步时间	0	1	6.371395	0.0116
☐	☐	跑步时脉搏	0	1	1.428737	0.23197
☐	☐	休息时脉搏	0	1	3.510878	0.06097
☐	☐	最大脉搏	0	1	0.365726	0.54534

步进历史记录

步进	参数	操作	似然比卡方	"显著性概率"	条目卡方	条目 "显著性概率"	R 方	p	AICc	BIC
1	体重	已进入	12.16669	0.0005	10.1342	0.00146	0.2833	2	35.2047	37.6441

请注意，图 5.11 中“步进历史记录”表中的**体重**的“条目卡方”和“条目‘显著性概率’”值与图 5.10 中**体重**的“Wald/得分卡方”和“显著性概率”值相符。

具有交叉、交互作用或多项式项的模型

一些模型（特别是与实验设计关联的那些模型）涉及交互作用项。对于连续因子，这些是表示效应的列之积。对于名义型和有序型因子，交互作用由涉及表示分类水平的项之积的模型项定义。

有交互作用项时，您通常想要对模型选择过程进行限制，以便在模型中包含高阶效应的低阶成分。这由“效应遗传”原理建议。请参见《实验设计指南》。例如，若模型中包含双因子交互作用，也应包含其成分主效应（**前项**）。

合并规则的示例

1. 选择**帮助** > **样本数据库**，然后打开 Reactor.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 在 “次数”框中，键入 2。
5. 选择 F、 Ct、 A、 T 和 Cn，然后点击**宏** > **析因次数**。
6. 对于**特质**，选择**逐步**。

7. 点击**运行**。

图 5.12 使用合并规则的初始 “当前估计值” 报表

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	65.5	1	0	0.000	1
☐	☐	F	0	1	15.125	0.066	0.79972
☐	☐	Ct	0	1	3042	23.412	3.67e-5
☐	☐	A	0	1	3.125	0.014	0.90823
☐	☐	T	0	1	924.5	4.611	0.03998
☐	☐	Cn	0	1	312.5	1.415	0.24363
☐	☐	F*Ct	0	1	15.125	0.066	0.79972
☐	☐	F*A	0	3	22.75	0.031	0.9926
☐	☐	F*T	0	1	6.125	0.027	0.87178
☐	☐	F*Cn	0	1	0.125	0.001	0.98161
☐	☐	Ct*A	0	1	6.125	0.027	0.87178
☐	☐	Ct*T	0	1	1404.5	7.612	0.00979
☐	☐	Ct*Cn	0	1	32	0.139	0.71193
☐	☐	A*T	0	1	36.125	0.157	0.69476
☐	☐	A*Cn	0	1	6.125	0.027	0.87178
☐	☐	T*Cn	0	1	968	4.863	0.03525

图 5.12 中的模型包含五个连续因子的最高达到双因子交互作用的所有项。通过第 224 页的 “规则” 中所述的 “合并”、“限制” 和 “整体效应” 规则，您可以控制交互作用项的进入。

“合并” 规则确定基于两个检验的交互作用项的输入。请参见第 224 页的 “合并”。您可以根据自由度 nDF 来确定将哪两个检验用于 p 值。例如，交互作用项 F*A 的自由度值为 3。这意味着 F*A 与其前项 F 和 A 组合在一起，并且基于自由度为 3 的联合 F 检验进入。相比之下，交互作用项 Ct*T 的自由度值为 1。这意味着 Ct*T 基于自由度为 1 的 F 检验进入（该检验用于在 Ct*T 的前项 Ct 和 T 已经进入模型后检验 Ct*T 的显著性）。点击**步进**一次可看到 Ct 独自进入。

8. 再次点击**步进**可看到 Ct*T 已随同 T 进入 （Ct 已在模型中）。

图 5.13 使用 “合并” 规则执行一步后的 “当前估计值” 报表

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	65.5	1	0	0.000	1
☐	☐	F	0	1	15.125	0.263	0.61236
☐	☑	Ct	9.75	2	4446.5	39.676	6.74e-9
☐	☐	A	0	1	3.125	0.054	0.81819
☐	☑	T	5.375	2	2329	20.781	2.93e-6
☐	☐	Cn	0	1	312.5	6.715	0.01524
☐	☐	F*Ct	0	2	30.25	0.256	0.7764
☐	☐	F*A	0	3	22.75	0.123	0.9459
☐	☐	F*T	0	2	21.25	0.178	0.83755
☐	☐	F*Cn	0	1	0.125	0.002	0.96335
☐	☐	Ct*A	0	2	9.25	0.077	0.92601
☐	☑	Ct*T	6.625	1	1404.5	25.064	2.72e-5
☐	☐	Ct*Cn	0	1	32	0.562	0.45989
☐	☐	A*T	0	2	39.25	0.334	0.7194
☐	☐	A*Cn	0	1	6.125	0.106	0.74747
☐	☐	T*Cn	0	1	968	43.488	4.49e-7

有显著交互作用项时，几个项可以在同一步进入。若点击了**步进**按钮两次，Ct*T 将随它的两个包含的效应 Ct 和 T 一起进入。但是，后退一步不是对称的，因为可以在不删除其两个成分项的情况下删除交叉项。请注意现在 Ct 和 T 各自的自由度均为 2。这是因为若 “逐步” 删除 Ct

或 T，它必须也删除 Ct*T。若您将“方向”更改为**后退**并点击**步进**，将删除 Ct*T 且 Ct 和 T 的自由度变为 1。

具有名义型和有序型效应的模型

传统上，逐步回归不适用于模型中有分类效应的情形。请注意以下几点：

- 回归模型包含名义型或有序型效应时，这些效应用一组指示符列表示。
- 分类效应只有两个水平时，该效应用单个列表示。
- 分类效应有 *k* 个水平（其中 *k* > 2）时，它必须用 k-1 个列表示。

JMP 中用于标准平台的惯例是使用其参数估计值在所有水平上平均为零的项表示名义型变量。

在“逐步”平台中，以**分层**方式对分类变量（名义型和有序型）进行编码。这与其他最小二乘法拟合平台中的编码方式不同。在分层编码中，将分类变量的水平逐个拆分为最能分隔响应均值的水平组。该拆分过程实现了用 *k* - 1 个项表示 *k* 个水平的分类变量的目标。

注意：在分层编码中，构造的初始项表示实现最大分隔的组。该编码方案的优势是这些信息性项有可能先进入模型。

分层项的构造

分层项使用类似于“分割”分析的树结构进行构造。但是，最大化的准则为组间平方和 (SSB)。

对于具有 *k* 个水平的名义型变量，将 *k* 个水平拆分为具有最大 SSB 的两组水平。将这两组水平称为 A1 和 A2，其中 A1 具有较小的均值，A2 具有较大的均值。A1 和 A2 中的两组水平用于定义一个指标变量，它对于 A1 中的水平具有值 1，对于 A2 中的水平具有值 -1。该变量是名义型变量的第一个分层项。

对于初始组 A1 和 A2 每个组中的各个水平，找到一个具有最大 SSB 的两组水平的拆分。假定具有最大 SSB 的水平组是 A1 中的水平。将这两个组称为 B1 和 B2，其中 B1 具有较小的均值，B2 具有较大的均值。B1 和 B2 中的两组水平用于定义一个分层变量，它对于 B1 中的水平具有值 1，对于 B2 中的水平具有值 -1，对于 A2 中的水平具有值 0。要构造下一个变量，请拆分 B1、B2 和 A2 中的水平。使 SSB 最大化的拆分定义下一个分层变量。继续执行该过程，直到构造了 *k-1* 个分层项。

对于有序型变量，在拆分中考虑的水平组仅包含在顺序中是相邻的水平。这确保构造的项遵守水平顺序。

规则和分层项

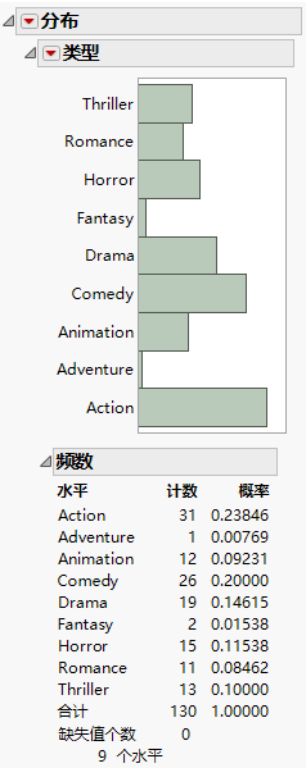
当您使用**合并**规则或**限制**规则时，项无法进入模型，除非层次中该项上层的所有项都进入了模型。使用**整体效应**规则并使分类变量的项进入时，它的所有关联项也会进入。有关示例，请参见第 234 页的“示例中分层项的构建”。

具有名义型项的模型示例

该示例使用有关在 2011 年发行的电影的数据。您对**全球总收入**值特别感兴趣，该值表示总收入。您的潜在预测变量为**烂番茄分数**、**观众打分**和**类型**。两个得分变量为连续的，但是**类型**是名义型的。在尝试使用 “逐步”简化模型前，您想探索相关变量。

1. 选择**帮助** > **样本数据库**，然后打开 Hollywood Movies.jmp。
2. 选择**分析** > **分布**。
3. 选择**类型**，然后点击 **Y，列**。
4. 点击**确定**。

图 5.14 “类型”的分布



请注意，**类型**有 9 个水平，因此将用 8 个模型项来表示。进一步的数据探索将揭示：由于缺失数据，“逐步”只考虑 8 个水平。

5. 在数据表的列面板中，选择关注的列：**烂番茄分数**、**观众打分**和**全球总收入**。
6. 选择**分析** > **筛选** > **探索缺失值**。
7. 点击 **Y，列**，然后点击**确定**。

图 5.15 “缺失列”报表

缺失列

仅显示带缺失值的列
关闭
选择列并选择一项操作。
选择行 为单元格着色
排除行 为行着色

列	缺失值数
烂番茄分数	2
观众打分	1
全球总收入	2

请注意，**烂番茄分数**在 2 行中有缺失，**观众打分**在 1 行中有缺失，**全球总收入**在 2 行中有缺失。

8. 在 “缺失列”报表中，选择**列**下面列出的三列。
9. 点击**选择行**。

   在数据表的行面板中，您可以看到选择了三行。因为这三行包含有关预测变量或响应的缺失数据，它们将自动从 “逐步”分析中排除。请注意，第 128 行是 “Adventure”类别中的唯一一个条目，这意味着将从分析中彻底删除该类别。为了进行 “逐步”分析，因此**类型**只有 8 个类别。既然您已了解缺失数据的影响，接下来将执行 “逐步”分析。
10. 选择**分析 > 拟合模型**。
11. 选择**烂番茄分数**、**观众打分**和**类型**，然后点击**添加**。

    若您使用**烂番茄分数**、**观众打分**和**类型**作为预测变量，使用标准最小二乘法模型拟合**全球总收入**，残差将具有很高的异方差性。（这是财务数据的典型特点）。使用对数变换可以更好地满足方差相等的回归假设。
12. 在 “选择列”列表中右击**全球总收入**，然后选择**变换 > 对数**。

    变换后的变量**对数 [全球总收入]** 显示在 “选择列”列表的底部。
13. 选择**对数 [全球总收入]**，然后点击 **Y**。
14. 从 “特质”列表中选择**逐步**。
15. 点击**运行**。

图 5.16 显示了模型项列表的 “当前估计值”表

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率> F"
☑	☑	截距	4.09998703	1	0	0.000	1
☐	☐	烂番茄分数	0	1	10.78976	3.303	0.07154
☐	☐	观众打分	0	1	8.715114	2.655	0.10577
☐	☐	类型{Drama&Horror&Thriller&Fantasy&Romance&Comedy-Action&Animation}	0	1	49.78148	16.850	7.25e-5
☐	☐	类型{Drama&Horror&Thriller-Fantasy&Romance&Comedy}	0	1	9.560063	2.918	0.09008
☐	☐	类型{Drama-Horror&Thriller}	0	1	2.027141	0.608	0.43718
☐	☐	类型{Horror-Thriller}	0	1	0.013011	0.004	0.95043
☐	☐	类型{Fantasy&Romance-Comedy}	0	1	1.428612	0.428	0.51439
☐	☐	类型{Fantasy-Romance}	0	1	0.000376	0.000	0.99157
☐	☐	类型{Action-Animation}	0	1	1.362765	0.408	0.52426

在“当前估计值”表中，请注意**类型**用 7 个项表示。您将使用其中的两个项构造模型以了解这些项是如何定义的。

16. 选中**类型**前两个项旁边的**已进入**框：
    - 类型{Drama&Horror&Thriller&Fantasy&Romance&Comedy-Action&Animation}
    - 类型{Drama&Horror&Thriller-Fantasy&Romance&Comedy}
17. 点击**构建模型**。

    请注意，已将两个项作为临时变换列添加到 “模型规格”窗口中的 “模型效应”列表。将在下一节中讨论这些列。

示例中分层项的构建

回想一下，由于存在缺失值，**类型**是具有 8 个水平的名义型变量。在“当前估计值”表中，**类型**用 7 个项表示。这是合适的，因为**类型**有 8 个水平。下面说明了表示**类型**的前两个项。后面的项按类似方式定义。

第一个项

显示的第一个项是类型{Drama&Horror&Thriller&Fantasy&Romance&Comedy-Action&Animation}。该变量的格式为**类型** {A1 - A2}，其中 A1 和 A2 用减号分隔。该符号指示根据组间平方和确定的最大化分隔发生在以下两组水平之间：

- Drama、Horror、Thriller、Fantasy、Romance 和 Comedy（用 A1 表示）
- Action 和 Animation（用 A2 表示）

若您在模型中包含项**类型** {Drama&Horror&Thriller&Fantasy&Romance&Comedy-Action&Animation}，则在模型中使用表示该项的临时变换列。该列包含以下值：

- 对于 Drama、Horror、Thriller、Fantasy、Romance 和 Comedy，值为 1
- 对于 Action 和 Animation，值为 -1

第二个项

显示的第二个项是**类型** {Drama&Horror&Thriller-Fantasy&Romance&Comedy}。这组水平完全包含在第一个项 (A1) 的第一次拆分中。该符号对比水平：

- Drama、Horror 和 Thriller
- Fantasy、Romance 和 Comedy。

在 Drama、Horror、Thriller、Fantasy、Romance 和 Comedy (A1) 水平与 Action 和 Animation (A2) 水平的所有拆分中，算法确定该拆分具有组间的最大平方和。

若您在模型中包含该项，则模型中使用表示该项的临时变换列。该列包含以下值：

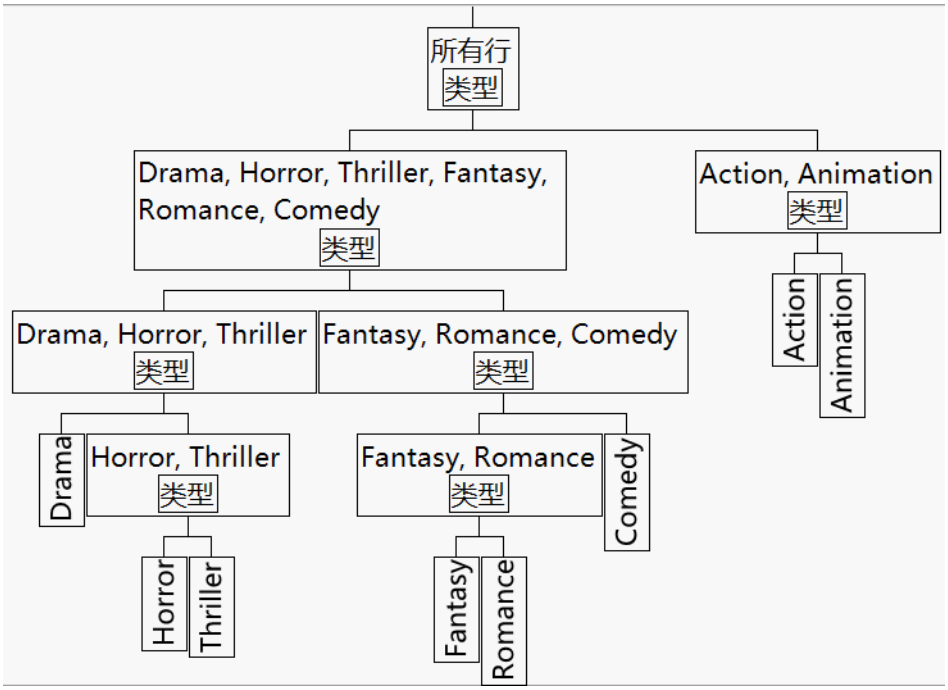
- 对于 Drama、Horror 和 Thriller，值为 1
- 对于 Fantasy、Romance 和 Comedy，值为 -1

- 对于 Action 和 Animation，值为 0

项的层次结构

基于组间平方和准则继续拆分项。导致项的定义的层次结构显示在图 5.17 中。

图 5.17 显示了分层编码中使用的拆分的树



规则

当您使用**合并**规则或**限制**规则时，项无法进入模型，除非层次中该项上层的所有项都进入了模型。例如，若您输入**类型**{Action-Animation}，则 JMP 还会输入**类型**{Drama&Horror&Thriller&Fantasy&Romance&Comedy-Action&Animation}。

当您使用**整体效应**规则并使任何一个**类型**项进入时，所有**类型**项都将进入。

分层项的限制规则的示例

若您具有包含名义型或有序型项的模型，当构建或运行模型时，包含模型中涉及的分层项的临时变换列将在模型拟合中使用。模型本身显示在新的“拟合模型”窗口中。该示例进一步说明了**逐步**是如何构造具有分层效应的模型的。

用一个简单模型来检查热狗的每盎司成本（**美元 / 盎司**）与热狗的**种类**（Met、Beef、Poultry）和热狗的**大小**（Jumbo、Regular、Hors d'oeuvre）的函数关系。

1. 选择**帮助 > 样本数据库**，然后打开 Hot Dogs2.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**美元 / 盎司**，然后点击 **Y**。
4. 选择**种类**和**大小**，然后点击**添加**。
5. 对于**特质**，选择**逐步**。
6. 点击**运行**。
7. 对于**停止规则**，选择 **p 值阈值**。
8. 对于**规则**，选择**限制**。

图 5.18 使用 “p 值阈值” 和 “限制” 规则的 “逐步控制” 面板

请注意，将默认规则 “合并” 更改为 “限制” 时，两个项的 “F 比” 和 “概率 >F” 值显示为缺失。这是项**种类** {Poultry-Meat} 和**大小** {Regular-Jumbo}。这是因为在其前项进入模型前，这两个项无法进入模型。

9. 点击**步进**。

项**种类** {Poultry&Meat-Beef} 进入模型。该项具有最小的 “概率 >F” 值，该值小于 “进入的概率” 阈值 0.25。

图 5.19 已进入一项的 “逐步控制” 面板

当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	0.11864706	1	0	0.000	1
☐	☑	种类{Poultry&Meat-Beef}	-0.0283529	1	0.040492	26.954	3.51e-6
☐	☐	种类{Poultry-Meat}	0	1	0.012426	9.648	0.00309
☐	☐	大小{Hors d'oeuvre-Regular&Jumbo}	0	1	0.001038	0.687	0.4112
☐	☐	大小{Regular-Jumbo}	0	0	0	.	.

显示项**种类** {Poultry-Meat} 的 “F 比” 和 “概率 >F” 值。因为它的前项已进入模型，现在允许**种类** {Poultry-Meat} 进入。

10. 点击**步进**。

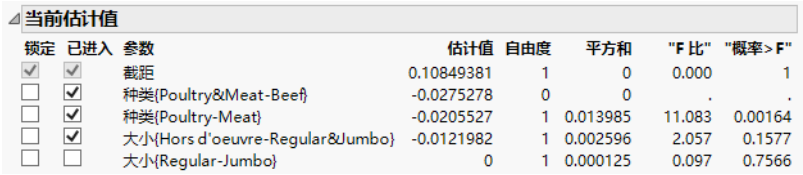
    因为**种类** {Poultry-Meat} 在剩余项中具有最小的 “概率 >F” 值且该值小于 “进入的概率” 阈值，它是要进入模型的下一项。

11. 点击**步进**。

    项**大小** {Hors d'oeuvre-Regular&Jumbo} 进入模型，因为它的 “概率 >F” 值为 0.1577。因为它的前项现在在模型中，因此允许项**大小**{Regular-Jumbo}进入模型且显示它的“概率>F”值。

    但是，项**大小** {Regular-Jumbo} 的 “概率 >F” 值为 0.7566，它超过了 “进入的概率” 值 0.25。因此，若您再次点击 “步进”，它不进入模型。

图 5.20 最终模型的 “当前估计值” 报表



当前估计值

锁定	已进入	参数	估计值	自由度	平方和	"F 比"	"概率>F"
☑	☑	截距	0.10849381	1	0	0.000	1
☐	☑	种类{Poultry&Meat-Beef}	-0.0275278	0	0	.	.
☐	☑	种类{Poultry-Meat}	-0.0205527	1	0.013985	11.083	0.00164
☐	☑	大小{Hors d'oeuvre-Regular&Jumbo}	-0.0121982	1	0.002596	2.057	0.1577
☐	☐	大小{Regular-Jumbo}	0	1	0.000125	0.097	0.7566

提示：使用 “执行” 按钮可自动运行整个逐步过程。要查看它是如何执行的，请点击**全部删除**。然后点击**执行**。

12. 点击**构建模型**。

    点击 “构建模型” 后，将显示 “拟合模型” 启动窗口，其中只包含在逐步过程中选择的三个模型效应。在启动窗口中，定义了进入模型的三个分层效应的临时变换列包含在模型中。

执行二值和有序型 Logistic 逐步回归

当响应为有序型或名义型时，“拟合模型” 的 “逐步” 特质将执行有序型 Logistic 逐步回归。名义型响应在 Logistic 逐步回归拟合过程中被视为有序型响应。响应只有两个水平时，有序型 Logistic 回归模型与名义型 Logistic 回归模型等价。要运行 Logistic 逐步回归，请指定一个有序型或名义型响应，像往常那样向模型添加项，然后从 “特质” 菜单中选择 “逐步” 。

Logistic 模型的逐步报表类似于响应为连续型时提供的那些报表。以下元素是 Logistic 回归结果特有的：

- 若响应为分类响应，则通过其负对数似然（- 对数似然）来提供模型的整体拟合。该值基于完整的迭代最大似然拟合进行计算。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。
- “当前估计值”报表显示卡方统计量（Wald/得分卡方）及其 *p* 值（显著性概率）。检验统计量列显示当前模型未包含的参数的得分统计量，并且显示当前模型包含的参数的 Wald 统计量。回归估计值（估计值）基于完整的迭代最大似然拟合。
- “步进历史记录”报表显示 L-R 卡方。该值是在给定模型的其他项时，假设相应回归参数为零的似然比检验的检验统计量。“显著性概率”是该检验的 *p* 值。

注意：若响应为名义型，您可以通过点击“构建模型”按钮使用“拟合模型”的“名义型 Logistic”特质拟合当前模型。在出现的“拟合模型”启动窗口中，点击“运行”。

使用 Logistic 逐步回归的示例

1. 选择**帮助** > **样本数据库**，然后打开 Fitness.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**性别**，然后点击 Y。
4. 选择**体重**、**跑步时间**、**跑步时脉搏**、**休息时脉搏**和**最大脉搏**，然后点击**添加**。
5. 对于**特质**，选择**逐步**。
6. 点击**运行**。
7. 点击**执行**。

图 5.21 Logistic 逐步报表

当前估计值

锁定	已进入	参数	估计值	自由度	Wald/得分卡方	"显著性概率"
☑	☑	截距[M]	-35.807846	1	0	1
☐	☑	体重	0.28229655	1	5.471429	0.01933
☐	☑	跑步时间	1.32680366	1	2.545732	0.11059
☐	☐	跑步时脉搏	0	1	0.470408	0.4928
☐	☐	休息时脉搏	0	1	0.007216	0.9323
☐	☐	最大脉搏	0	1	0.076497	0.7821

步进历史记录

步进	参数	操作	似然比卡方	"显著性概率"	条目卡方	条目 "显著性概率"	R 方	p	AICc	BIC
1	体重	已进入	12.16669	0.0005	10.1342	0.00146	0.2833	2	35.2047	37.6441
2	跑步时间	已进入	8.017826	0.0046	6.37139	0.0116	0.4700	3	29.6472	33.0603
3	跑步时脉搏	已进入	1.440088	0.2301	1.44684	0.22903	0.5036	4	30.8567	35.0542
4	最大脉搏	已进入	0.071809	0.7887	0.06994	0.79142	0.5052	5	33.6464	38.4164
5	休息时脉搏	已进入	0.007156	0.9326	0.00722	0.93228	0.5054	6	36.7393	41.8432
6	最佳	特定	.	.	0.00722	0.93228	0.4700	3	29.6472	33.0603

两个变量**体重**和**跑步时间**基于 “停止规则”进入模型。

8. 点击**构建模型**。

图 5.22 简化模型的 “模型规格”窗口

显示 “模型规格”窗口，其中包含两个变量作为模型效应。请注意 “特质”为 “名义型 Logistic”。若响应是有序型，“特质”将为 “有序型 Logistic”。

“所有可能的模型”选项

使用“所有可能的模型”选项调查可使用您的预测变量构造的所有模型。可从“逐步”红色小三角中访问该选项。

请注意以下事项：

- 该选项不适用于大问题，此时模型数将超过 5 百万。
- 分类预测变量用指标变量表示。请参见第 231 页的“具有名义型和有序型效应的模型”。

以下选项限制显示的模型数：

模型中的最大项数 为模型中的最大项数输入一个值。

要查看的最佳模型数 输入要显示的每个大小的最大模型数。根据 R 方值显示最佳模型。

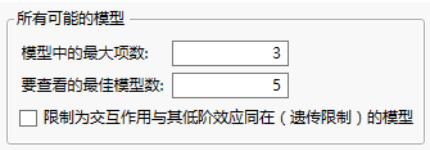
限制为交互作用与其低阶效应同在 （遗传限制）的模型 仅显示这样的模型，即：当高阶效应被加进时，该模型包含其所有低阶效应。这些模型满足强效应遗传。当您的预测变量包含交互作用或多项式项时，该选项很有用。

使用“所有可能的模型”选项的示例

1. 选择帮助 > 样本数据库，然后打开 Fitness.jmp。

2. 选择**分析 > 拟合模型**。
3. 选择**吸氧量**并点击 **Y**。
4. 选择**跑步时间**、**跑步时脉搏**、**休息时脉搏**和**最大脉搏**，然后点击**添加**。
5. 对于**特质**，选择**逐步**。
6. 点击**运行**。
7. 点击“逐步”红色小三角菜单并选择**所有可能的模型**。
8. 输入 3 作为最大项数，输入 5 作为最佳模型数。

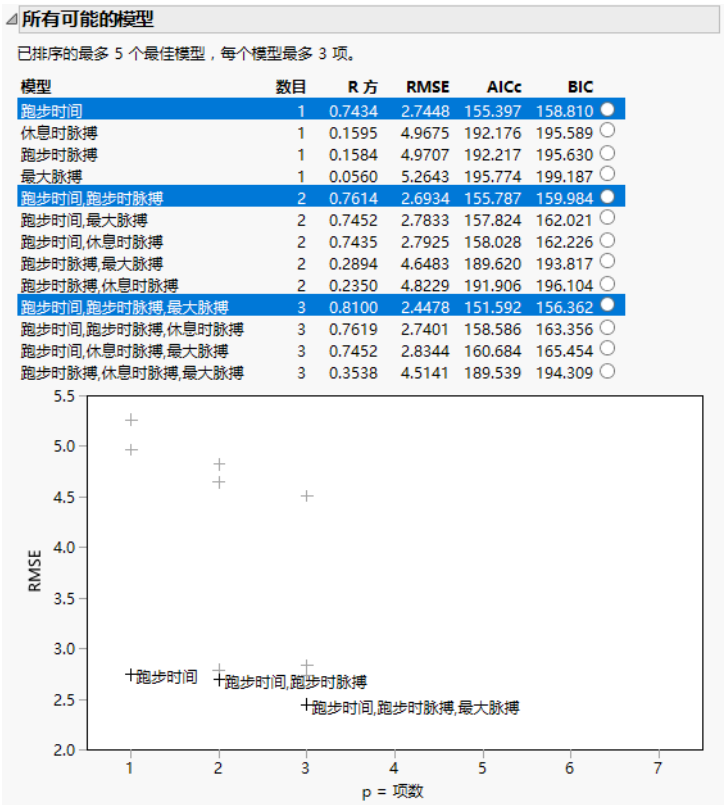
图 5.23 所有可能的模型弹出对话框



9. 点击**确定**。

拟合所有可能的模型（一个模型中最多有三项）。

图 5.24 “所有可能的模型”报表



以模型包含的参数数目递增的顺序列出这些模型。对于每个参数数目，突出显示具有最大 R^2 的模型。通过表右侧的单选按钮列，您可以一次选择一个模型并检查结果。

注意：用于选择模型的建议准则是选择对应于最小 BIC 或 AICc 值的模型。有些分析人员还想查看 C_p 统计量。计算 Mallow C_p 统计量，但是初始时它隐藏在表中。要使它可见，可以在表中右击并选择**列 > Cp**。

“模型平均”选项

模型平均方法支持您平均若干模型的拟合结果，而非选择一个模型。生成的平均模型往往具有优越于被平均的每一个单个模型的预测能力。“模型平均”功能有助于避免模型过度拟合数据。将很多项选入模型时，拟合往往会增大参数估计值。模型平均倾向于收缩较弱项的估计值，这会得到更好的预测。根据每个模型的 AICc 权重来平均各个模型，其计算如下所示：

AICc 权重 = exp[-0.5(AICc - AICc 最佳)]

“AICc 最佳”是拟合模型中最小的 AICc 值。为每个模型计算 AICc 权重值，按降序排序，再统一尺度使总和为 1。总和小于 1 减去指定的累积 AICc 权重截止值的统一尺度的 AICc 权重值设置为零。这样可以避免在平均模型中使用弱模型。平均模型的参数是具有非零 AICc 权重值的各模型的参数估计值的加权平均值。

使用“模型平均”选项的示例

1. 选择**帮助 > 样本数据库**，然后打开 Fitness.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**吸氧量**并点击 Y。
4. 选择**跑步时间**、**跑步时脉搏**、**休息时脉搏**和**最大脉搏**，然后点击**添加**。
5. 对于**特质**，选择**逐步**。
6. 点击**运行**。
7. 点击“逐步”红色小三角菜单并选择**模型平均**。
8. 为模型中的最大项数输入 3。
9. 为“累积 AICc 权重截止”保持 0.95。

图 5.25 “模型平均”窗口

10. 点击**确定**。

图 5.26 “模型平均”报表

在“模型平均”报表中，显示每个参数的平均估计值和标准误差。显示的标准误差反映朝向零的估计值偏倚。

11. 点击**保存预测公式**以在原始数据表中保存预测公式。

逐步回归中的验证选项

要在 JMP 中为逐步回归模型执行交叉验证，请点击“逐步拟合”红色小三角并选择 **K 重交叉验证**。请参见第 245 页的“逐步回归中的 K 重交叉验证”。

JMP PRO 在 JMP Pro 中，您可以在“拟合模型”窗口中指定“验证”列。验证列必须具有数值数据类型，应至少包含两个非重复值。

- 若列包含两个值，则较小的值定义训练集，较大的值定义验证集。
- 若列包含三个值，值按大小递增的顺序定义训练、验证和测试集。
- 若列包含四个或更多非重复值且响应是连续的，则这些值定义 k 重验证的折数。

有关使用“验证”列的更多信息，请参见第 243 页的“逐步回归中具有两个或三个值的验证集”。

逐步回归中具有两个或三个值的验证集

JMP PRO 若您指定具有两个或三个值的“验证”列，“逐步”基于训练集拟合模型。为验证集和测试集报告模型拟合统计量。有关如何定义这些统计量的详细信息，请参见第 244 页的“验证集和测试集统计量定义”。

若响应是连续的，以下统计量显示在 “逐步回归控制”面板中：

- R 方验证（也显示在“步进历史记录”报表中）
- RASE 验证
- R 方检验（若有测试集）
- RASE 检验（若有测试集）

若响应是二值名义型或有序型，以下统计量显示在 “逐步回归控制”面板中：

- R 方验证（也显示在“步进历史记录”报表中）
- 验证平均对数误差
- R 方检验（若有测试集）
- 平均对数误差检验（若有测试集）

最大验证 R 方

若您在“拟合模型”窗口中指定具有两个或三个值的验证列，则“停止规则”默认为“最大验证 R 方”。该规则尝试找到使验证集的 R 方统计量最大的模型。该规则可在“方向”设置为“前进”或“后退”时应用。

注意：“最大验证 R 方”仅考虑由 *p* 值进入（“前进”方向）或删除（“后退”方向）定义的模型。它不考虑所有可能模型。

您可以使用“步进”按钮在“前进”方向中使项逐个进入，也可以在“后退”方向中逐个删除项。在任何点，您都可以通过点击“步进历史记录”报表中“R 方验证”右侧的按钮来选择模型。模型项的选择将在“当前估计值”报表中更新。这是您一点击“构建模型”或“运行模型”即使用的模型。

前进方向

在“前进”方向中，“逐步”基于下一个最小的 *p* 值添加项来构造逐次模型。

若您点击“执行”而非“步进”，则自动执行项进入过程。在拟合的模型中，被视为最佳的模型最后列出。通过忽略“R 方验证”中的局部下沉来获得该模型。具体而言，它是具有最大“R 方验证”的模型，后面可能跟着具有更小“R 方验证”值的 10 个模型。该模型通过“参数”列中的“最佳”和“操作”列中的“特定”这两项来指定。使用“R 方验证”右侧的按钮选择这个“最佳”模型，尽管您可以自由更改该选择。

后退方向

在“后退”方向中，“逐步”基于下一个最大的 *p* 值删除项来构造逐次模型。

要使用“后退”方向，您必须首先点击“全部进入”使所有项进入模型。“后退”方向以类似于“前进”方向的方式运行。若您点击“执行”而非“步进”，则自动执行项删除过程。指定为“最佳”的模型是具有最大“R 方验证”的模型，它后面可能跟着具有更小“R 方验证”值的 10 个模型。

验证集和测试集统计量定义

“R 方验证”和“RASE 验证”在本节中定义。以完全类似的方式为测试集计算“R 方检验”和“RASE 检验”。

连续响应

R 方验证 按以下方式计算验证集的 R 方测度：

- 对于验证集中的每个观测，计算预测误差。这是实际响应与训练集模型预测的响应之间的差值。
- 计算预测误差的平方和以得到 $SSE_{\text{验证}}$。
- 对验证集中的实际响应与它们的均值的差值求平方和。这是 $SST_{\text{验证}}$。
- R 方验证计算如下：

$$\text{R方验证} = 1 - \frac{SSE_{\text{验证}}}{SST_{\text{验证}}}$$

注意：R 方验证有可能为负值。

RASE 验证 验证集的均方预测误差的平方根。它按以下方式计算：

- 对于验证集中的每个观测，计算预测误差。这是实际响应与训练集模型预测的响应之间的差值。
- 计算预测误差的平方和以得到 $SSE_{验证}$。
- 用 $n_{验证}$ 表示验证集中的观测数。
- RASE 验证表示如下：

$$\text{RASE 验证} = \sqrt{\frac{SSE_{验证}}{n_{验证}}}$$

二值名义型或有序型响应

R 方验证 按以下方式计算验证集的 “熵 R 方” 测度 （也称为 McFadden R^2）：

- 使用训练集拟合模型。
- 获得所有观测的预测概率。
- 使用基于训练集模型的预测概率，为验证集中的观测计算模型的似然。该量被称为似然_完全$_{验证}$。
- 使用验证集中的数据，计算简化模型（无预测变量）的似然。该量被称为似然_简化$_{验证}$。
- R 方验证计算如下：

$$\text{R方验证} = 1 - \frac{\log(\text{似然_完全}_{验证})}{\log(\text{似然_简化}_{验证})}$$

注意：R 方验证有可能为负值。

验证平均对数误差 按以下方式计算验证集的平均对数误差：

- 对于验证集中的每个观测值，计算由模型根据训练集确定的其预测概率的对数。
- 将这些对数求和，除以验证集中的观测数，然后对所得值求负数。

提示：验证平均对数误差的值越小越好。

逐步回归中的 K 重交叉验证

K 重交叉验证随机将数据分为 k 个子集。依次将 k 个子集中的每一个用作验证集，而将其余数据用作训练集来拟合模型。总共拟合 k 个模型，获得 k 个验证统计量。给出最佳验证统计量的模型将被选为最终模型。该方法适合小数据集，因为它能充分利用有限的数据。

注意：K 重交叉验证仅适用于连续响应。

在 JMP 中，点击“逐步拟合”红色小三角并选择 K 重交叉验证。

在 JMP Pro 中，您可以通过两种方式访问 *k* 重交叉验证：

- 点击“逐步拟合”红色小三角并选择 K 重交叉验证。
- 指定具有四个或更多非重复值的验证列。

R 方 K 重统计量

若您执行 k 重交叉验证，R 方 K 重统计量显示在“逐步回归控制”面板中其他统计量的右侧。R 方 K 重是 *k* 重的 R 方验证值的平均值。

最大 K 重 R 方

使用 *k* 重交叉验证时，“停止规则”默认为“最大 K 重 R 方”。该规则尝试使 R 方 K 重统计量最大化。

注意：“最大 K 重 R 方”仅考虑由 *p* 值进入（“前进”方向）或删除（“后退”方向）定义的模型。它不考虑所有可能模型。

“最大 K 重 R 方”停止规则以类似于“最大验证 R 方”停止规则的方式运行。请参见第 243 页的“最大验证 R 方”。使用“R 方 K 重”替换对“R 方验证”的引用。

第 6 章

JMP PRO 广义回归模型

使用变量选择方法构建模型

“拟合模型”平台的 “广义回归”特质仅在 JMP Pro 中可用。

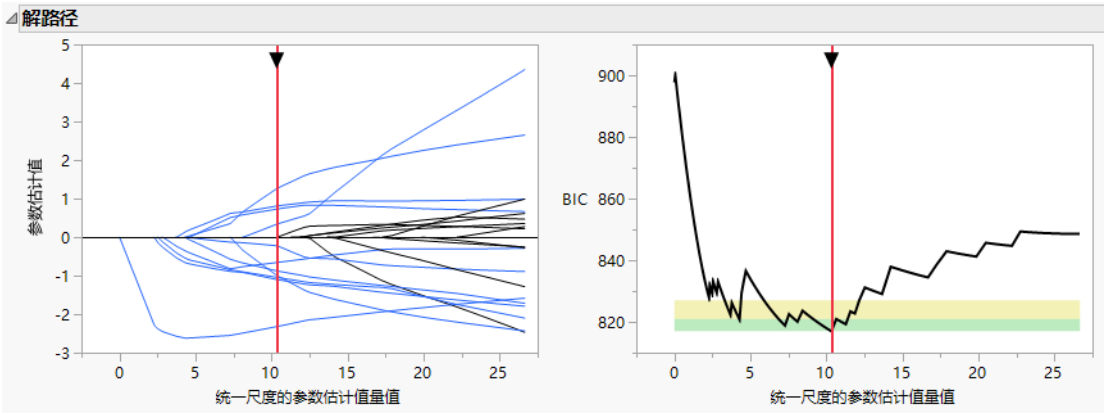
在 JMP Pro 中，“拟合模型”平台的“广义回归”特质提供了变量选择方法，其中包括专门用于解决高维相关数据建模的收缩方法。其中的两种方法（Lasso 和弹性网络）会将变量选择作为建模过程的一部分。

包含很多变量的大数据集通常显示多重共线性问题。如今的数据集可包含比观测值更多的变量，若使用传统建模方法，需要选择变量。多重共线性问题和存在大量预测变量凸显了传统方法的不足。

即使对于相关性不强或不相关的小数据集（包括设计实验），Lasso 和弹性网络也很有用。可利用它们来构建预测模型，或者选择用于模型简化或未来研究的变量。

“广义回归”特质对于很多建模情形都很有用。该特质允许您为响应变量指定各种分布。当您的响应是连续的、二项、计数或零泛滥时可以使用它。当您想选择变量或怀疑预测变量中存在共线性时可以使用它。更常见的是使用它来拟合模型，以将该模型与使用其他方法获得的模型进行比较。

图 6.1 弹性网络拟合的解路径



目录

“广义回归”特质概述 249
广义回归的示例 250
启动“广义回归”特质 253
分布 254
“广义回归”报表窗口 260
“广义回归”报表选项 261
“模型启动”控制面板 262
响应分布 262
“估计方法”选项 262
高级控件 266
“验证方法”选项 268
提前停止 269
执行 269
模型拟合报表 269
回归图 269
模型汇总 270
估计详细信息 272
解路径 273
中心化和统一尺度的预测变量的参数估计值 276
原始预测变量的参数估计值 277
活跃参数估计值 278
效应检验 278
模型拟合选项 278
“广义回归”特质的统计详细信息 285
估计方法的统计详细信息 285
高级控件的统计详细信息 287
分布的统计详细信息 288

JMP PRO “广义回归”特质概述

“广义回归”特质采用正则或惩罚回归方法。这类方法通过向零收缩模型系数来尝试拟合更好的模型。得到的估计值有偏倚。这种偏倚增大可能导致预测方差减小，从而与非惩罚模型相比会降低总预测误差。其中的两种方法（弹性网络和 Lasso）会将变量选择作为建模过程的一部分。

弹性网络和 Lasso 这类建模方法特别适用于大数据集，这些数据集中通常存在共线性问题。此外，如今的数据集通常包含比观测值更多的变量。这种情形有时被称为 $p > n$ 问题，其中 n 为观测数，p 为预测变量数。若使用传统建模方法，这类数据集需要进行变量选择。

弹性网络和 Lasso 也可用于相关性不强的小数据集（包括设计实验）。可利用它们来构建预测模型，或者选择用于模型简化或未来研究的变量。

该特质提供以下几类建模方法：

- 最大似然
- 基于步骤的估计
- 惩罚回归

弹性网络和 Lasso 是相对较新的方法（Tibshirani 1996；Zou and Hastie 2005）。这两种方法都对模型系数的大小有惩罚，从而导致连续收缩。收缩量由**调节参数**决定。最优收缩水平由几种验证方法中的一种决定。这两种方法都可以将系数收缩到零。这样就可将变量选择内置到建模过程中。弹性网络模型将 Lasso 和岭回归作为特例纳入。请参见第 285 页的“估计方法的统计详细信息”。

有关广义回归建模方法的详细信息

- 最大似然方法是经典方法。它提供用于与其他方法进行比较的基线，并且它是最适用于假设检验等传统推理方法的位置。
- 向前选择是一种逐步回归方法。在向前选择中，项进入模型。添加最为显著的项，直到所有项都添加到模型中，或不再有任何自由度。
- Lasso 有两个缺点。当几个变量高度相关时，它倾向于只从该组中选择一个变量。当变量数 p 超过观测数 n 时，Lasso 最多选择 n 个预测变量。
- 另一方面，弹性网络倾向于从相关的组中选择所有变量，拟合合适的系数。当 $p > n$ 时，它还可选择 n 个以上的预测变量。
- 岭回归是最早提出的惩罚回归方法之一（Hoerl 1962；Hoerl and Kennard 1970）。岭回归不将系数收缩到零，因此它不执行变量选择。
- “双 Lasso”试图通过使用初始 Lasso 模型执行变量选择来分离选择步骤和收缩步骤。初始模型中选定的变量随后会用作第二个 Lasso 模型的输入变量。
- “两阶段向前选择”执行两个阶段的向前逐步回归。它会在第一个阶段中对主效应执行变量选择。然后，在第二个阶段中允许高阶效应进入该模型。

“广义回归”特质还拟合**自适应**版本的 Lasso 和弹性网络。这些自适应版本尝试对实际活跃集中的变量执行惩罚的力度小于对未包含在实际活跃集中的变量的惩罚力度。**实际活跃集**是指模型中对响应有实际影响的一组项。开发 Lasso 和弹性网络的自适应版本是为了确保 oracle 性质得以保留。假如您拟合预测变量实际活跃集的模型，oracle 性质将确保以下内容：您的估计值渐进地成为它们本来的值。更具体而言，您的模型可以正确标识应具有零系数的预测变量。您的估计值将收敛到仅使用实际活跃集建模时获得的那些值。请参见第 287 页的“自适应方法”。

“广义回归”特质允许您为响应变量指定各种分布。这些分布拟合包括正态、Cauchy、Student *t*、指数、gamma、Weibull、对数正态、beta、二项、beta 二项、Poisson、负二项、零泛滥二项、零泛滥 beta 二项、零泛滥 Poisson、零泛滥负二项和零泛滥 gamma。这种灵活性使您可以拟合分类和计数响应以及连续响应，特别是拟合向右偏斜的连续响应。您还可以拟合分位数回归和 Cox 比例危险模型。对于某些分布，您可以拟合删失数据的模型。该特质提供了各种用于模型选择的验证准则并支持训练列、验证列和测试列。请参见第 254 页的“分布”。

JMP PRO 广义回归的示例

Diabetes.jmp 样本数据表中的数据包括 442 名糖尿病患者的测量值。关注的响应是 Y，即在进行基线测量一年后测量的疾病进展数据。对十个被认定为与疾病进展有关的变量也测量了基线值。本例说明如何使用广义回归方法来生成预测模型。

1. 选择**帮助 > 样本数据库**，然后打开 Diabetes.jmp。
2. 选择**分析 > 拟合模型**。
3. 从“选择列”列表中选择 Y，然后点击 Y。
4. 从**年龄**一直选到**葡萄糖**，然后点击**宏 > 析因次数**。

   这会将最高达到 2 次（**次数**框中的默认值）的所有项都添加到模型中。
5. 从“选择列”列表中选择**验证**，然后点击**验证**。
6. 从“特质”列表中选择**广义回归**。
7. 点击**运行**。

   显示的“广义回归”报表包含“模型比较”报表、“模型启动”控制面板和一个“使用‘验证列’验证的‘正态标准最小二乘法’”报表。

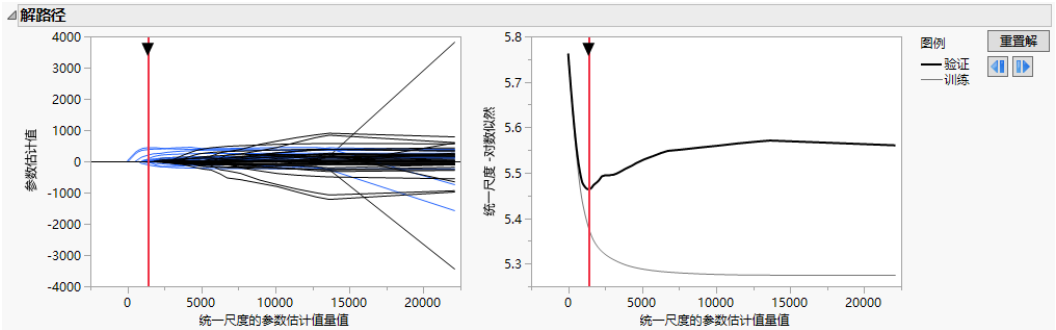
   在“模型启动”控制面板中，请注意以下事项：

   – “响应分布”设置为“正态”，因为您在“拟合模型”启动窗口中将“正态”指定为分布。
   – 默认“估计方法”是“Lasso”。
   – 因为您在“拟合模型”窗口中指定了验证列，所以“验证方法”设置为“验证列”。
8. 点击**执行**。

   随即显示“使用‘验证列’验证的‘正态’‘Lasso’”报表。“解路径”报表（图 6.2）显示参数估计值图和统一尺度的负对数似然图。当统一尺度的参数估计值量值减小时收缩量增

加。图最右侧的估计值是最大似然估计值。红色的垂直线指示验证准则选择的那些参数值，在该示例中的保留样本由**验证**列定义。

图 6.2 解路径图



9. 点击“使用‘验证列’验证的‘正态’‘Lasso’”旁边的红色小三角，然后选择**选择非零项**。

   该选项突出显示“原始预测变量的参数估计值”报表（图 6.3）中的非零项和它们在“解路径”图中的路径。同时选定数据表中的对应列。请注意，在 55 个参数估计值中，只有 11 个是非零的。同时还估计了正态分布的尺度参数 (sigma)，它显示在“原始数据的参数估计值”报表底部的单独表中。请注意，这 55 个参数估计值并不是全部都显示在图 6.3 中。

图 6.3 “原始预测变量的参数估计值”报表的一部分

原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	-247.8422	42.926975	33.334183	<.0001*	-331.9775	-163.7068
年龄	0	0	0	1.0000	0	0
性别[1-2]	8.7286488	6.9678967	1.5692446	0.2103	-4.928178	22.385475
BMI	5.702121	0.9303784	37.562422	<.0001*	3.8786128	7.5256293
BP	0.8156913	0.2526706	10.421787	0.0012*	0.320466	1.3109166
总胆固醇	0	0	0	1.0000	0	0
LDL	0	0	0	1.0000	0	0
HDL	-0.47425	0.3986141	1.4155011	0.2341	-1.25552	0.3070188
TCH	0	0	0	1.0000	0	0
LTG	41.286654	7.0844177	33.963392	<.0001*	27.401451	55.171858
葡萄糖	0	0	0	1.0000	0	0
(年龄-48.5181)*性别[1-2]	-0.082735	0.3621512	0.0521915	0.8193	-0.792538	0.6270682
(年龄-48.5181)*(BMI-26.3758)	0	0	0	1.0000	0	0
(年龄-48.5181)*(BP-94.647)	0.0043544	0.0171735	0.0642885	0.7998	-0.029305	0.0380137
(年龄-48.5181)*(总胆固醇-189.14)	0	0	0	1.0000	0	0
(年龄-48.5181)*(LDL-115.439)	0	0	0	1.0000	0	0
(年龄-48.5181)*(HDL-49.7885)	0	0	0	1.0000	0	0
(年龄-48.5181)*(TCH-4.07025)	0	0	0	1.0000	0	0
(年龄-48.5181)*(LTG-4.64141)	0	0	0	1.0000	0	0
(年龄-48.5181)*(葡萄糖-91.2602)	0.0231484	0.0233144	0.9858084	0.3208	-0.022547	0.0688439
性别[1-2]*(BMI-26.3758)	0	0	0	1.0000	0	0
性别[1-2]*(BP-94.647)	0	0	0	1.0000	0	0
性别[1-2]*(总胆固醇-189.14)	0	0	0	1.0000	0	0
性别[1-2]*(LDL-115.439)	0	0	0	1.0000	0	0
性别[1-2]*(HDL-49.7885)	-0.221009	0.4728172	0.2184913	0.6402	-1.147714	0.7056955
性别[1-2]*(TCH-4.07025)	0	0	0	1.0000	0	0
性别[1-2]*(LTG-4.64141)	0	0	0	1.0000	0	0
性别[1-2]*(葡萄糖-91.2602)	0	0	0	1.0000	0	0
(BMI-26.3758)*(BP-94.647)	0.0254032	0.0525392	0.2337819	0.6287	-0.077572	0.1283782
(BMI-26.3758)*(总胆固醇-189.14)	0	0	0	1.0000	0	0
(BMI-26.3758)*(LDL-115.439)	0	0	0	1.0000	0	0
(BMI-26.3758)*(HDL-49.7885)	0	0	0	1.0000	0	0
(BMI-26.3758)*(TCH-4.07025)	0	0	0	1.0000	0	0
(BMI-26.3758)*(LTG-4.64141)	0	0	0	1.0000	0	0
(BMI-26.3758)*(葡萄糖-91.2602)	0.0210214	0.0597655	0.1237153	0.7250	-0.096117	0.1381596
(BP-94.647)*(总胆固醇-189.14)	0	0	0	1.0000	0	0

“正态” 分布参数	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
尺度	52.512586	2.3230199	511.00007	<.0001*	47.959551	57.065621

要保存预测公式，请点击“使用 ‘验证列’验证的 ‘正态’‘Lasso’”报表旁边的红色小三角，然后选择**保存列 > 保存预测公式**。

JMP PRO 启动“广义回归”特质

通过以下方式来启动 “广义回归”特质：选择**分析 > 拟合模型**，输入一个或多个列作为 Y，然后从**特质**菜单中选择**广义回归**。

图 6.4 选择了 “广义回归”的 “拟合模型”启动窗口

有关对所有特质通用的“拟合模型”窗口各方面的详细信息，请参见第 29 页的“模型规格”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。此处给出了针对“广义回归”特质的信息。

“广义回归”特质中使用的名义型变量的参数化不同于这些变量在使用其他“拟合模型”特质时的参数化。“广义回归”特质使用指标函数参数化。在该参数化中，与名义型变量的某个水平的指标对应的估计值是该水平的响应均值与最后一个水平的响应平均值之间的差值。最后一个水平是具有最高值排序编码的水平；这个水平的指标函数未包含在模型中。

若您的模型效应有缺失值，可以将这些缺失值视为信息性类别。从“模型规格”红色小三角菜单中选择“信息性缺失”选项。

要指定没有截距项的模型，请在 “拟合模型”窗口的 “构造模型效应”面板中选择 “无截距”选项。若选择该选项，请注意以下事项：

- 预测变量未中心化也未统一尺度。
- 优势比、危险率和发生率等比率在该报表窗口中不可用。
- “无截距”选项不可用于“有序型 Logistic 分布”。

警告：不建议将“无截距”选项用于 Lasso 或弹性网络，因为结果对于模型效应的尺度很敏感。建议使用这些估计方法的自适应版本。

删失

可采用以下方法之一为响应变量指定删失：

- 对于区间删失、右删失和左删失响应，指定包含“检测限值”列属性的列。列属性中指定的限值定义被视为未删失的响应值的范围：
  - 对于区间删失响应，请在“检测限值”列属性中为“检测下限”和“检测上限”指定非缺失值。
  - 对于右删失响应，请在“检测限值”列属性中为“检测下限”指定缺失值，为“检测上限”指定非缺失值。
  - 对于左删失响应，请在“检测限值”列属性中为“检测下限”指定非缺失值，为“检测上限”指定缺失值。
- 对于右删失响应，在启动窗口中将包含右删失观测的指示符的列指定为“删失”列。从“删失代码”列表中选择该列中指定右删失观测的值。
- 对于区间删失和左删失响应，在 Y 列角色中指定定义删失区间的两列：
  - 对于区间删失响应，第一个 Y 变量提供下限。
  - 对于左删失响应，第一个 Y 变量包含一个缺失值。
  - 对于区间删失和左删失响应，第二个 Y 变量给出每个响应的上限。

  若为 Y 指定了两列并且指定了支持删失的分布，则会显示一条提醒，询问这些列是否表示删失。若您选择 “否”，则这些列将被视为单独的响应。

  注意：您可以使用 “广义回归”平台首选项中的 “两个响应列的处理”首选项指定两个响应的默认行为。

仅当指定的分布为“正态”、“指数”、“Gamma”、“Weibull”、“对数正态”或“Cox 比例风险”时，删失才可用。

JMP PRO 分布

当您从**特质**菜单中选择**广义回归**时，将显示**分布**选项。您可以在此处指定 Y 的分布。缩写形式 ZI 表示**零泛滥**。根据响应将分布分为三个类别：连续、离散和零泛滥。在下面介绍了这些选项。

注意：若您在“模型规格”窗口中指定了多个 Y 变量，则必须对指定的所有 Y 变量使用相同的响应分布。若想要在同一个“广义回归”报表中对不同的响应变量拟合不同的分布，则必须使用脚本。

JMP PRO 连续型

正态 Y 服从均值为 μ 且标准差为 σ 的正态分布。正态分布是对称的，若样本足够大，很多其他分布可以使用中心极限定理将其近似为正态分布。μ 的连结函数是恒等。即，Y 的均值表示为线性模型。

注意：若指定的“分布”为“正态”，则“标准最小二乘法”将取代“最大似然估计”方法。

正态分布的尺度参数为 σ。若估计方法中无罚值，则尺度参数 σ 的估计值为均方根误差 (RMSE)。RMSE 是 σ^2 的通常无偏估计量的平方根。除非涉及删失观测，否则所显示的结果等同于标准最小二乘法拟合。

注意：“广义回归”特质中使用的名义型变量的参数化不同于这些变量在使用“标准最小二乘法”特质时的参数化。由于存在这种差异，参数估计值不同于包含名义型或有序型效应的模型。

请参见第 288 页的 “分布的统计详细信息”。

Cauchy Y 服从位置参数为 μ 且尺度参数为 σ 的 Cauchy 分布。Cauchy 分布具有未定义的均值和标准差。中位数和众数都是 μ。大多数数据本质上都不服从 Cauchy 分布。不过，它对包含较大比例的离群值（高达 50%）的数据执行稳健回归很有用。μ 的连结函数是恒等。请参见第 288 页的 “分布的统计详细信息”。

t(5) Y 服从自由度为 5、位置参数为 μ 且尺度参数为 σ 的 Student *t* 分布。 Student *t* 分布是对称的而且是一个稳健的选项，它跨越正态分布与 Cauchy 分布之间的空间。随着 Student *t* 分布中的自由度接近无穷大，该分布等价于正态分布。当 Student *t* 分布中的自由度等于 1 时，该分布等价于 Cauchy 分布。μ 的连结函数是恒等。请参见第 288 页的 “分布的统计详细信息”。

指数 Y 服从均值参数为 μ 的指数分布。指数分布是向右偏斜的，通常用于对寿命或两个相继事件之间的时间建模。 μ 的连结函数为对数。请参见第 288 页的 “分布的统计详细信息”。

Gamma Y 服从均值参数为 μ 且离散参数为 σ 的 gamma 分布。 gamma 分布很灵活，它包含一系列其他广泛使用的分布。例如，指数分布是 gamma 分布的特例，其中 σ = μ。卡方分布也可以从 gamma 分布导出。μ 的连结函数为对数。请参见第 288 页的 “分布的统计详细信息”。

Weibull Y 服从均值参数为 μ 且尺度参数为 σ 的 Weibull 分布。 Weibull 分布是较为灵活的分布，经常用于对寿命或事件发生前的时间建模。 μ 的连结函数是恒等。请参见第 288 页的 “分布的统计详细信息”。

对数正态 Y 服从均值参数为 μ 且尺度参数为 σ 的对数正态分布。对数正态分布是向右偏斜的，通常用于对寿命或事件发生前的时间建模。 μ 的连结函数是恒等。请参见第 288 页的 “分布的统计详细信息”。

Beta Y 服从均值参数为 μ 且离散参数为 σ 的 beta 分布。 beta 的响应介于 0 到 1 之间 （不包括 0 和 1），通常用于比例或比率建模。μ 的连结函数为 Logit。请参见第 288 页的 “分布的统计详细信息”。

分位数回归 分位数回归对响应的指定条件分位数建模。没有对所基于的分布的形式做任何假设。当您选择 “分位数回归” 时，在 “分布” 菜单下将显示一个 “分位数” 框。指定所需的分位数。

若您为 “模型对话框” 窗口上的 “分位数” 指定 0.5 （默认值），则分位数回归将对响应的条件中位数建模。当用回归系数表示的条件分位数的变化率取决于分位数时，分位数回归特别有用。分位数回归相对于最小二乘回归的一个优势是：它对具有非纯一条件分布的数据建模很灵活。

通过使用迭代方法使目标函数最小化来拟合分位数回归。有关分位数回归的详细信息，请参见 Koenker and Hallock (2001) 和 Portnoy and Koenker (1997)。

当您选择 “分位数回归” 时，“最大似然” 是唯一可用的估计方法，“无” 是唯一可用的验证方法。

注意：若分位数回归拟合要花很长时间才完成，将显示一个进度条。该进度条显示目标函数的相对变化。当您点击 “接受当前估计值” 时，计算将停止，报告的参数估计值将对应于此时的最佳模型拟合。

Cox 比例风险 Cox 比例危险模型是含有预测变量的事件时间数据的回归模型。它基于预测变量与危险率函数之间的相乘关系，可用于检查预测变量对于生存时间的效应。该模型涉及一个任意基线危险率函数，该函数通过预测变量统一尺度以提供常规危险率函数。比例风险模型为每个预测变量生成参数估计值和标准误差。 Cox 比例危险模型最初是由 D. R. Cox (1972) 提出的。有关比例风险模型的详细信息，请参见 Kalbfleisch and Prentice (2002)。

选择 Cox 比例风险时，可用的验证方法只有 BIC 和 AICc。此外，岭估计方法也不可用。

注意：若响应中有结值，则使用 Efron 似然。请参见 Efron (1977)。这种处理结值的方法不同于在 “拟合模型” 平台的 “比例风险” 特质或 “拟合比例风险” 平台中使用的方法。

JMP PRO 离散

二项 Y 服从参数为 p 和 n 的二项分布。响应 Y 指示所有试验中具有固定概率 p 的 n 个独立试验中的总成功次数。该分布允许使用样本大小列。若未列出任何列，则假定样本大小为 1。p 的连结函数为 Logit。若选择建模类型为 “名义型” 的二值响应变量，则只有 “二项” 是可用的响应分布。请参见第 288 页的 “分布的统计详细信息”。

当您选择 “二项” 作为 “分布” 时，必须使用以下方式之一指定响应变量。

– 未汇总：若您的数据未汇总为事件频数，指定一个二值列作为响应。若该列的建模类型为名义型，那么您可将其中一个水平指定为 “目标水平”。默认 “目标水平” 值是基于水平顺序的两个水平中的较高者。

– 已使用“频数”列汇总：若您的数据已汇总为成功和失败的频数，则指定一个二值列作为响应。若该列的建模类型为名义型，那么您可将其中一个水平指定为“目标水平”。默认“目标水平”值是基于水平顺序的两个水平中的较高者。将频数列分配给**频数**角色。

– 已汇总且将样本大小列输入为第二个 Y：若您的数据已汇总为事件（成功）和试验的频数，则按如下顺序将两个连续列指定为 Y：成功次数的计数和试验次数的计数。

注意：若指定的“分布”为“二项”，则“Logistic 回归”将取代“最大似然估计”方法。

Beta 二项 Y 服从成功概率为 p、试验次数为 n 且过度离散参数为 δ 的 beta 二项分布。该分布是过度离散版本的二项分布。

运行 JMP Samples/Scripts 文件夹中的 demoBetaBinomial.jsl，将具有离散参数 δ 的 beta 二项分布与具有参数 p 且 $n = 20$ 的二项分布进行比较。

beta 二项分布对于每个观测要求样本大小大于 1。因此，用户必须指定一个样本大小列。要插入样本大小列，请按如下顺序将两个连续列指定为 Y：成功次数和试验次数。p 的连结函数为 Logit。请参见第 288 页的 “分布的统计详细信息”。

多项式 Y 服从有三个或更多离散水平的多项式分布。响应变量的建模类型必须为名义型或有序型。该模型为响应变量的每个水平拟合单独的截距和效应参数。若响应变量有 k 个水平，则模型包含 k - 1 个截距和效应参数。多项式分布的连结函数是多项式 Logit。请参见“统计详细信息”一章中第 453 页的 “名义型响应”。

有序型 Logistic Y 服从包含有序型水平的多项式分布。响应变量的建模类型必须为有序型。该模型为响应变量的每个水平拟合截距。效应参数对于响应变量的所有水平都是相同的。有序型 Logistic 分布的连结函数是有序的 Logit。请参见 “统计详细信息”一章中第 455 页的 “有序型响应”。

注意：“广义回归”中的“有序型 Logistic”的截距参数化不同于“拟合模型”的“有序型 Logistic”特质的截距参数化。“广义回归”中的第一个截距项对应于“有序型 Logistic”特质中的第一个截距项。“广义回归”中的后续截距项是响应变量的有序水平的截距项之间的相邻差值。

Poisson Y 服从均值为 λ 的 Poisson 分布。 Poisson 分布通常对给定时间间隔中的事件数建模，经常表示为计数数据。λ 的连结函数为对数。即使 Y 采用非整数值， Poisson 回归也是允许的。请参见第 288 页的 “分布的统计详细信息”。

负二项 Y 服从均值为 μ 且离散参数为 σ 的负二项分布。负二项分布通常对指定失败次数之前的成功次数建模。负二项分布也等价于某些条件下的 Gamma Poisson 分布。有关负二项和 Gamma Poisson 之间关系的详细信息，请参见 《基本分析》。

运行 JMP Samples/Scripts 文件夹中的 demoGammaPoisson.jsl，将具有均值 λ 和离散参数 σ 的 Gamma Poisson 分布与具有均值 λ 的 Poisson 分布进行比较。

μ 的连结函数为对数。即使 Y 采用非整数值，负二项回归也是允许的。请参见第 288 页的 “分布的统计详细信息”。

JMP PRO 零泛滥

零泛滥二项 Y 服从参数为 p、n 且零泛滥参数为 π 的零泛滥二项分布。响应 Y 指示所有试验中具有固定概率 p 的 n 个独立试验中的总成功次数。该分布允许使用样本大小列。若未列出任何列，则假定样本大小为 1。p 的连结函数为 Logit。请参见第 288 页的“分布的统计详细信息”。

零泛滥 Beta 二项 Y 服从成功概率为 p、试验次数为 n、过度离散参数为 δ 且零泛滥参数为 π 的 beta 二项分布。该分布是过度离散版本的零泛滥二项分布。零泛滥 beta 二项分布对于每个观测要求样本大小大于 1。因此，用户必须指定一个样本大小列。要插入样本大小列，请按如下顺序将两个连续列指定为 Y：成功次数和试验次数。p 的连结函数为 Logit。请参见第 288 页的“分布的统计详细信息”。

零泛滥 Poisson Y 服从均值参数为 λ 且零泛滥参数为 π 的零泛滥 Poisson 分布。参数 λ 是基于来自 Poisson 分布的观测值（而非来自于泛滥的零值）的条件均值。λ 的连结函数为对数。即使 Y 采用无观测零值或非整数值，零泛滥 Poisson 回归也是允许的。请参见第 288 页的“分布的统计详细信息”。

零泛滥负二项 Y 服从位置参数为 μ、离散参数为 σ、且零泛滥参数为 π 的零泛滥负二项分布。参数 μ 是基于来自负二项分布的观测值（而非来自于泛滥的零值）的条件均值。μ 的连结函数为对数。即使 Y 采用无观测零值或非整数值，零泛滥负二项回归也是允许的。请参见第 288 页的“分布的统计详细信息”。

零泛滥 Gamma Y 服从均值参数为 μ 且零泛滥参数为 π 的零泛滥 gamma 分布。在很多时候，我们可能认为非零响应服从 gamma 分布。这对于保险索赔是成立的：索赔值近似服从 gamma 分布，但是对于没有任何索赔的保险单，数据中也有零。零泛滥 gamma 可以直接处理这类数据而不必将数据分为零和非零响应。参数 μ 是基于来自 gamma 分布的观测值（而非来自于泛滥的零值）的条件均值。μ 的连结函数为对数。请参见第 288 页的“分布的统计详细信息”。

表 6.1 列出了被分配各种分布的 Y 变量的数据类型、建模类型和其他要求。

表 6.1 服从各种分布的 Y 的要求

分布	数据类型	建模类型	其他
正态	数值	连续	
Cauchy	数值	连续	
t(5)	数值	连续	
指数	数值	连续	正值
Gamma	数值	连续	正值
Weibull	数值	连续	正值

表 6.1 服从各种分布的 Y 的要求 (续)

分布	数据类型	建模类型	其他
对数正态	数值	连续	正值
Beta	数值	连续	介于 0 到 1 之间
分位数回归	数值	连续	
Cox 比例风险	数值	连续	非负
二项，未汇总	任意	任意	二值
二项，已使用“频数”列汇总	任意	任意	二值
二项，已汇总且将计数列输入为第二个 Y	数值	连续	非负
Beta 二项	数值	连续	非负
多项式	任意	有序型或名义型	
有序型 Logistic	任意	有序型	
Poisson	数值	任意	非负
负二项	数值	任意	非负
零泛滥二项	数值	任意	非负
零泛滥 Beta 二项	数值	任意	非负
零泛滥 Poisson	数值	任意	非负
零泛滥负二项	数值	任意	非负
零泛滥 Gamma	数值	连续	非负

有关这些分布如何参数化的详细信息，请参见第 288 页的“分布的统计详细信息”。表 6.2 汇总了这些详细信息。

表 6.2 分布、参数和连结函数

分布	参数	均值模型连结函数
正态	μ, σ	恒等 (μ)
Cauchy	μ, σ	恒等 (μ)
t(5)	μ, σ	恒等 (μ)

表 6.2 分布、参数和连结函数 (续)

分布	参数	均值模型连结函数
指数	μ	*Log*(μ)
Gamma	μ, σ	*Log*(μ)
Weibull	μ, σ	恒等 (μ)
对数正态	μ, σ	恒等 (μ)
Beta	μ	*Logit*(μ)
二项	*n*, *p*	*Logit*(*p*)
Beta 二项	*n*, *p*, δ	*Logit*(*p*)
Poisson	Λ	*Log*(μ)
负二项	μ, σ	*Log*(μ)
零泛滥二项	*n*, *p*, π（零泛滥）	*Logit*(*p*)
零泛滥 Beta 二项	*n*, *p*, δ, π（零泛滥）	*Logit*(*p*)
零泛滥 Poisson	λ, π（零泛滥）	*Log*(μ)
零泛滥负二项	μ, σ, π （零泛滥）	*Log*(μ)
零泛滥 Gamma	μ, σ, π （零泛滥）	*Log*(μ)

在选择合适的分布后，点击**运行**。随即显示“广义回归”报表窗口。

JMP PRO “广义回归”报表窗口

在“拟合模型”启动窗口中点击**运行**后，出现的“广义回归”报表窗口将包含以下项：

- “模型比较”报表，支持您比较报表中拟合的所有模型。每次使用“模型启动”控制面板拟合新模型时，都会将其添加到“模型比较”报表中。可以使用“显示”列中的复选框显示或隐藏拟合模型的报表。“模型比较”报表中的其他列包含有关拟合的每个模型的信息，以及每个模型的拟合统计量。点击列标题可按“模型比较”报表中的任何列对模型进行排序。第一次点击按升序排序；第二次点击列标题按降序排序。

  注意：拟合一个或多个模型后，“模型比较”报表随即显示。

- 用于拟合模型的“模型启动”控制面板。请参见第 262 页的““模型启动”控制面板”。拟合模型时，将添加各个分级显示项，这些项附带标题，用来描述您已经拟合的模型的类型。请参见第 269 页的“模型拟合报表”和第 278 页的“模型拟合选项”。

  - 若模型项之间存在线性相依性，则“模型启动”控制面板会包含一个“奇异性详细信息”报表，其中显示模型项所满足的线性函数。请参见“标准最小二乘法模型”一章中第 171 页的“模型项之间具有线性相依性的模型”。
- “最大似然”报表，其中显示使用最大似然估计拟合的模型的结果。请参见第 263 页的“最大似然”。只有满足以下条件时，才显示“最大似然”报表：
  - 预测变量中不存在线性相依性。
  - 观测数多于预测变量数。
  - 预测变量数不超过 250 个。

注意：当指定的“分布”为“正态”分布时，此报表标记为“标准最小二乘”。当指定的“分布”为“二项”时，此报表标记为“Logistic 回归”。

JMP PRO “广义回归”报表选项

模型对话框 为当前分析显示完成的“拟合模型”启动窗口。

设置随机种子 为用于“K 重”和“保留”验证的随机化过程设置种子。若您要重新生成分析，该选项会很有用。将种子设置为正值然后保存脚本，该种子将自动保存在脚本中。运行该脚本可以始终生成相同的交叉验证分析。

保存编码表 创建一个新的数据表，其中第一列显示所有模型参数的 JMP 编码。最后一列显示响应变量的值。若为二项响应指定了样本大小列，则响应和样本大小列都会显示在表中。若您使用两个响应列指定删失，这两个响应列将显示在表中。若使用包含“检测限值”列属性的响应列来指定删失，则编码表中的响应列包含“检测限值”列属性。

注意：编码数据表包含一个称为“原始数据”的表变量，提供用于分析的数据表的名称。在已指定“依据”变量的情况下，“原始数据”表变量还提供“依据”变量及其水平。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO “模型启动”控制面板

“模型启动”控制面板提供以下方面的选项：

- 第 262 页的“响应分布”
- 第 262 页的“‘估计方法’选项”
- 第 266 页的“高级控件”
- 第 268 页的“‘验证方法’选项”
- 第 269 页的“提前停止”
- 第 269 页的“执行”

JMP PRO 响应分布

控制面板中可用的响应分布取决于“拟合模型”启动窗口中指定的响应变量的属性。“响应分布”选项中提供了适当的响应分布。有关响应分布的详细信息，请参见第 254 页的“分布”。

JMP PRO “估计方法”选项

可用的估计方法可归入几下几类：无选择和无惩罚的方法、基于步骤的模型选择方法和惩罚回归方法。

“最大似然”、“标准最小二乘法”和“Logistic 回归”等方法拟合“拟合模型”启动窗口中指定的整个模型。不执行变量选择。这些模型可充当与其他方法比较时的基线。

注意：“最大似然”、“标准最小二乘法”和“Logistic 回归”中只有一个办法可用于给定报表。该估计方法的名称取决于“拟合模型”启动窗口中指定的分布。

“向后消除”、“向前选择”、“剪除的向前选择”、“最佳子集”和“两阶段向前选择”等方法均基于每一步进入或离开模型的变量。不过，它们不会对回归系数施加惩罚。

Dantzig 选择器、Lasso、弹性网络、岭和双 Lasso 等方法均为惩罚回归方法。它们收缩回归系数的大小并减小估计值方差，以便改进模型的预测能力。

注意：您的数据具有高度共线性时，自适应版本的 Lasso 和弹性网络可能无法提供好的解。这是因为自适应版本假定 MLE 提供好的估计值。这些情况下不推荐使用“自适应”选项。

在这些方法中使用两种罚值：

- l_1 罚值，它惩罚回归系数的**绝对值**之和
- l_2 罚值，它惩罚回归系数的**平方**和

默认的观测数据估计方法是 Lasso。若数据表包含实验设计脚本但不具有奇异性，则默认估计方法是“向前选择”，且启用“效应遗传”选项。若数据表包含实验设计脚本且设计矩阵中存在奇异性，则默认估计方法是“两阶段向前选择”，且启用“效应遗传”选项。

以下方法可用于模型拟合：

无选择和无惩罚的估计方法

最大似然 计算模型参数的最大似然估计值 (MLE)。不应用罚值。“最大似然”是唯一可用于分位数回归的估计方法。若在 “拟合模型”启动窗口中指定了 “验证”列，则拟合训练集的最大似然模型。默认情况下，只要满足以下条件，就会出现最大似然模型报表：

- 预测变量中不存在线性相依性。
- 观测数多于预测变量数。
- 预测变量数不超过 250 个。

“最大似然”选项为您提供了一种针对 “广义回归”特质支持的响应分布构造经典模型的方式。此外，基于最大似然的模型可以作为模型比较的基线。

当指定的 “分布”为 “正态”或 “二项”分布时，“最大似然”方法分别称为 “标准最小二乘”或 “Logistic 回归”。

标准最小二乘法 若指定了“正态”分布，则使用“标准最小二乘法”估计方法取代“最大似然”估计方法。默认报表为“标准最小二乘法”报表，该报表提供通常的标准最小二乘法结果。

Logistic 回归 若指定了“二项”分布，则使用“Logistic 回归”估计方法取代“最大似然”估计方法。默认报表为“Logistic 回归”报表。Logistic 结果同于最大似然结果。

基于步骤的估计方法

注意：当指定的“分布”为“多项式”时，基于步的估计方法不可用。

向后消除 使用向后消除回归计算参数估计值。所选的模型提供相对所选的验证方法而言是最好的解。向后消除最先将所有参数包括在模型中，然后每一步去除一个效应，直到得到仅含截距的模型。在每一步，都使用每个参数的 Wald 检验来确定删除哪个参数。

警告：“向后消除的解路径”的水平轴是其他估计方法中同一轴的相反轴。因此，当您在“向后消除的解路径”估计方法中从左向右移动时，将从模型中删除各个项而不是添加各个项。

向前选择 使用向前逐步回归计算参数估计值。在每一步，具有最显著得分检验的效应添加到模型中。所选的模型就是相对所选的验证方法而言提供最佳解的模型。

若存在交互作用并且启用了“效应遗传”选项，则通过以下方式处理复合效应。若在给定步骤具有最显著得分检验的效应是将要违反效应遗传的效应，则创建复合效应。复合效应包含具有最显著得分检验的效应以及满足效应遗传所需的任何其他不活跃效应。若复合效应具有最显著得分检验，则该复合效应中的所有效应都添加到模型中。

剪除向前选择 使用混合的向前和向后步骤计算参数估计值。算法从一个仅含截距的模型开始。第一步，具有最显著得分检验的效应添加到模型中。第一步之后，算法会在每一步考虑以下三种可能：

1. 从模型未包含的效应中，添加具有最显著得分检验的效应。
2. 从模型中的效应，删除具有最不显著 Wald 检验的效应。
3. 在一步中执行以上两个操作。

要选择每一步执行的操作，算法使用指定的验证方法。例如，若验证方法为 BIC，则算法选择生成最小 BIC 值的操作。若存在交互作用并且启用了 “效应遗传”选项，则添加效应时会考虑复合效应，但删除效应时不会考虑复合效应。

当模型变得饱和时，算法尝试使用后退步骤来检查这是否会改进模型。算法中的最大步骤数是参数数目的 5 倍。所选的模型就是相对所选的验证方法而言提供最佳解的模型。

“剪除向前选择”是 “逐步回归”特质中的 “混合步骤”选项的替换选项。不过，它不使用 p 值来确定哪些变量进入或离开模型。

提示：不推荐对 “剪除向前选择”估计方法使用 “提前停止”选项。

最佳子集 通过在每一步中增加活跃效应数计算参数估计值。在每一步中，从所有可能的模型（具有由步骤号指定的一些效应）中选择所需模型。“解路径”图的水平轴上的值表示模型中活跃效应的数目。步骤 0 对应于仅含截距的模型。步骤 1 对应于仅包含一个活跃效应的模型中的最佳模型。步骤会继续增加，直到达到在 “模型启动”报表的 “高级控件”中指定的 “最大效应数”的值。请参见第 266 页的 “高级控件”。

提示：“最佳子集”估计方法要占用大量计算资源。不推荐用于大型问题。

两阶段向前选择 （仅当模型中有二阶或高阶效应时才可用。）在两个阶段中计算参数估计值。在第一阶段中，针对主效应运行向前逐步回归模型以确定保留在模型中的项。在第二阶段中，针对完全由第一阶段中选定的主效应构成的所有高阶效应运行向前逐步回归模型。该方法假定存在强效应遗传。

未保留在第一阶段中的项仍作为清零项显示在“参数估计值”报表中；不过，在第二阶段模型的拟合中会忽略这些项。在第一阶段中选定的项不强制进入第二阶段；这些项可供在第二阶段中选择。

惩罚估计方法

Dantzig 选择器 （仅当指定的分布为 “正态”且未选定 “无截距”选项时才可用。）通过使用线性规划法应用 l_1 罚值来计算参数估计值。请参见 Candes and Tao (2007)。Dantzig 选择器适用于分析设计实验的结果。对于正交问题，Dantzig 选择器和 Lasso 提供相同结果。请参见第 286 页的 “Dantzig 选择器”。

Lasso 通过应用 l_1 罚值来计算参数估计值。由于 l_1 罚值，一些系数可能估计为零。这样，变量选择作为拟合过程的一部分来执行。在普通 Lasso 中，所有系数被同等惩罚。

自适应 Lasso 通过惩罚回归系数的绝对值加权之和来计算参数估计值。l_1 罚值权重由数据决定以保证 Oracle 性质 (Zou, 2006)。该选项使用 MLE 来衡量 l_1 惩罚的权重。若预测变量数超过观测数或者预测变量之间存在严格的线性依赖关系，则无法计算 MLE。若无法计算回归参数的 MLE，则将广义逆矩阵解或岭解用于 l_1 罚值权重。请参见第 287 页的“自适应方法”。

若预测变量高度相关，Lasso 和自适应 Lasso 选项通常选择更简化的模型。这些方法倾向于在一组相关的预测变量中只选择一个变量。高维数据倾向于具有高度相关的预测变量。对于这类数据，选择弹性网络可能比选择 Lasso 更好。请参见第 286 页的 “Lasso 回归”。

弹性网络 通过同时应用 l_1 罚值和 l_2 罚值来计算参数估计值。l_1 罚值确保执行变量选择。l_2 罚值通过像岭方法那样收缩系数来改进预测能力。

自适应弹性网络 使用自适应 l_1 罚值以及 l_2 罚值来计算参数估计值。该选项使用 MLE 来衡量 l_1 惩罚的权重。若预测变量数超过观测数或者预测变量之间存在严格的线性依赖关系，则无法计算 MLE。若无法计算回归参数的 MLE，则将广义逆矩阵解或岭解用于 l_1 罚值权重。您可以在“高级控件”面板中设置“弹性网络 Alpha”的值。请参见第 287 页的“自适应方法”。

当预测变量高度相关时，与 Lasso 相比，弹性网络能提供更好的预测准确性。（实际上，岭和 Lasso 是弹性网络的特例。）从预测能力来看，自适应弹性网络的性能通常要比弹性网络和自适应 Lasso 的性能高。弹性网络可以选择有相关性的预测变量组并为涉及的预测变量分配合适的参数估计值。请参见第 286 页的 “弹性网络”。

注意：若您选择弹性网络拟合并将“弹性网络 Alpha”设置为缺失，则算法按如下顺序计算拟合：Lasso、弹性网络和岭拟合。若拟合要用很长时间才完成，将显示一个进度条。当您点击“接受当前估计值”时，计算将停止，报告的参数估计值将对应于此时的最佳模型拟合。进度条指示算法拟合 Lasso、弹性网络和岭的时间。您可以使用该信息来决定何时点击“接受当前估计值”。

岭 使用岭回归计算参数估计值。岭回归是应用 l_2 罚值的有偏回归方法，它不会生成为 0 的参数估计值。当您要保留模型中的所有预测变量时，它很有用。请参见第 286 页的“岭回归”。

双 Lasso 在两个阶段中计算参数估计值。在第一阶段中，拟合 Lasso 模型以确定要用在第二阶段中的项。在第二阶段中，使用第一阶段中的项拟合 Lasso 模型。显示的“解路径”结果和参数估计值报表均针对第二阶段拟合。若在第一阶段中没有任何变量进入模型，则不会出现第二阶段，报表中将显示第一阶段的结果。

双 Lasso 在观测数小于预测变量数时尤其有用。通过将变量选择和收缩运算细分为两个阶段，第二阶段中的 Lasso 就不太可能过度惩罚应包含在模型中的项。双 lasso 类似于松弛 lasso。松弛 lasso 在 Hastie et al.(2009, p. 91) 中说明。

自适应双 Lasso 在两个阶段中计算参数估计值。在第一阶段中，拟合自适应 Lasso 模型以确定要用在第二阶段中的项。在第二阶段中，使用第一阶段中的项拟合自适应 Lasso 模型。第二阶段仅考虑包含在第一阶段模型中的项，并使用基于第一阶段中的参数估计值的权重。您可以使用“高级控件”中的“自适应惩罚权重”选项选择计算权重的方法。请参见第 267 页的“ “高级控件”选项”。显示的结果针对第二阶段拟合。若在第

一阶段中没有任何变量进入模型，则不会出现第二阶段，报表中将显示第一阶段的结果。请参见第 287 页的“自适应方法”。

JMP PRO 高级控件

使用“高级控件”选项可以调整模型拟合过程的各个方面。很多控件与调节参数的网格有关。

调节参数

Lasso 和岭估计方法的解路径取决于一个调节参数。弹性网络的解路径取决于针对似然罚值以及弹性网络 Alpha 的调节参数。弹性网络的似然罚值是与 Lasso 和岭估计方法关联的罚值的加权求和。“弹性网络 Alpha”决定这两个罚值的权重。请参见第 285 页的“估计方法的统计详细信息”和第 287 页的“高级控件的统计详细信息”。

调节参数为零时，不惩罚解，得到最大似然估计值。调节参数增大时，罚值增大。

解为一组参数估计值，它使惩罚负对数似然函数相对所选验证方法最小化。当前解在“解路径”图中用红色垂直实线标明。

注意：当“解路径”图中的“统一尺度的参数估计值量值”减小时，调节参数的值增大。接近 MLE 的估计值与大的量值关联，被严重惩罚的估计值则与小的量值关联。

请注意以下事项：

- 调节参数太小时，数据通常过拟合，生成方差很高的模型。
- 若调节参数太大，则数据通常拟合不足。

调节参数网格

要得到一个解，请在细网格上增大调节参数。

- 对于 Lasso、指定了“弹性网络 Alpha”的弹性网络以及岭，给出解的调节参数值是在调节参数网格上提供最佳拟合的值。

  注意：默认情况下，“弹性网络 Alpha”设置为 0.99。

- 若您未设置“弹性网络 Alpha”的值，则 alpha 的值也将在细网格上增大。对于固定的调节参数值，alpha 将不断变化，直到连续 10 个 alpha 值无法改进由验证方法决定的最佳拟合为止。对调节参数值的整个网格重复该过程。调节参数和 alpha 的最终值是在调节参数网格上提供最佳拟合的值。

调节参数值的网格范围从零（大多数情况）到使所有非截距项为零的最小值。定义使所有非截距项为零的调节参数的最小值为其**上限**。调节参数的**下限**为零，但是在以下两种情况下它设置为 0.0001：

- 若设计矩阵是奇异的，则无法计算最大似然估计值。下限 0.0001 允许计算接近 MLE 的估计值。

- 若选定的分布为二项或多项分布，则下限 0.0001 帮助阻止分离。

“高级控件”选项

强制效应遗传 要求低阶效应在相关的高阶效应前进入模型。在多数情况下，这意味着 X^2 不在模型中，除非 X 在模型中。对于“向前选择”之外的估计方法，有可能 X^2 进入模型而在同一步中 X 离开模型。若数据表包含实验设计脚本，则启用该选项，但是默认情况下它是关闭的。

弹性网络 Alpha 设置弹性网络的 α 参数。该 α 参数确定用于估计弹性网络系数的 l_1 和 l_2 罚值调节参数的组合值。默认值为 $\alpha = 0.99$，它设置 l_1 罚值的系数为 0.99，设置 l_2 罚值的系数为 0.01。仅当选择“弹性网络”作为“估计方法”时，该选项才可用。请参见第 285 页的“估计方法的统计详细信息”。

网格点数目 指定调节参数的下限和上限之间的网格点数目。在每个网格点值处，都获得这一调节参数值的参数估计值。默认值为 150 个网格点。

最小罚值率 指示调节参数的下限与上限之比的最小值。调节参数的下限为 0 时，解提供 MLE。在您不想包括 MLE 或很接近它的解的情况下，可以将“最小罚值率”设置为非零值。对于双 Lasso 估计方法，该选项的指定值仅用在拟合的第一阶段中。若设计矩阵中存在奇异性，则默认值为 0.0001。否则，默认值为 0。

网格尺度 提供用于选择网格尺度的分布的选项。您可以选择线性、平方根或对数尺度。与指定的网格点数数量相等的网格点将根据调节参数下限和上限之间的选定尺度来分布。默认网格尺度是平方根。请参见第 287 页的“高级控件的统计详细信息”。

第一阶段解 提供选项，用于选择双 Lasso 和两阶段向前选择的第一阶段中的解。默认情况下，根据指定的验证方法被视为最佳拟合的解将被选定，并且是最初显示的解（最佳拟合）。您可以选择最初显示一些模型，这些模型位于绿色或黄色区域中具有更大或更小 l_1 范数值。例如，若您选择“黄色区域中最小”，最初显示的解将是黄色区域中具有最小 l_1 范数的模型。请参见第 275 页的“可比模型区域”。

最大效应数 指定要在模型中为“最佳子集”估计方法考虑的最大效应数。您可以使用该选项限制拟合模型所需的计算次数。默认值为 10。

最初显示的解 提供的选项用于选择解，该解最初显示为“解路径”报表中的当前模型。当前模型通过一条垂直实线来标识。请参见第 273 页的“当前模型指示符”。最佳拟合解通过垂直虚线来标识。默认情况下，显示的解是根据指定的验证方法被视为最佳拟合的解。

您可以选择最初显示一些模型，这些模型位于绿色或黄色区域中具有更大或更小 l_1 范数值。例如，若您选择“黄色区域中最小”，最初显示的解将是黄色区域中具有最小 l_1 范数的模型。请参见第 275 页的“可比模型区域”。

自适应惩罚权重 提供用于计算自适应双 Lasso 的第二阶段中使用的惩罚权重的选项。默认情况下，选定“逆矩阵解”选项。该选项使用第一阶段拟合中的参数估计值计算惩罚权重。

“逆模型平均值”选项使用作为 AICc 或 BIC 模型加权平均值的解中的参数估计值计算惩罚权重。若选定 AICc 验证方法，则使用 AICc 模型。否则使用 BIC 模型。若使用“逆模型平均值”选项，最大似然解（若存在）则显示为“自适应双 Lasso”模型的解路径中的最右侧点。

强制项 允许您选择要将哪些项 （若有）强制包括在模型中。强制包括在模型中的项不包括在罚值中。

JMP PRO “验证方法”选项

以下方法可用于模型拟合的验证：

注意：唯一可用于“分位数回归”的验证方法为“无”。可用于“最大似然”估计方法的验证方法只有“无”和“验证列”。可用于“Cox 比例风险”的验证方法只有 BIC、AICc 和“无”。Dantzig 选择器估计方法只允许使用 BIC 和 AICc 验证方法。

K 重 对于调节参数的每个值，都将执行以下步骤：

- 观测划分到 *k* 个子集或**重**中。
- 每一重都轮流用作一个验证集。针对**不**含在“重”中的观测拟合模型。针对“重”**中**的观测计算基于该模型的对数似然，同时提供**验证**对数似然。
- 计算 *k* 重的验证对数似然的均值。该值充当调节参数值的验证对数似然。

具有最大验证对数似然的调节参数的值用于构造最终解。要获取最终模型，将对整个数据集拟合从调节参数最优值得到的全部 *k* 个模型。在这些模型中，具有最高的验证对数似然的模型将被选为最终模型。用于该最终模型的训练集将被指定为训练集，并且该模型的“维持”重是验证集。这些就是在最终解的图中和报表结果中使用的训练集和验证集。

保留 随机选择指定比例的数据作为验证集，并使用除验证集外的数据拟合模型。最终解是将验证集的负对数似然最小化的解。该方法对于大数据集很有用。随机选择基于对模型因子的分层抽样，以尝试创建比基于简单随机抽样的训练集和验证集更平衡的训练集和验证集。

留一法 执行留一法交叉验证。这等价于 K 重，其折数等于行数。该选项不应在中等或大数据集中使用。即使中等数量的观测值，它也可能需要很长的处理时间。将按照对 K 重验证那样来确定最终解的图中和报表结果中使用的训练集和 （一行）验证集。

BIC 使解路径上的 Bayesian 信息准则 (BIC) 最小化。请参见 “统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

AICc 使解路径上的校正的 Akaike 信息准则 (AICc) 最小化。 AICc 是 “验证方法”的默认设置。请参见 “统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

注意：当参数数目接近或超过样本大小时不定义 AICc。

ERIC 使解路径上的扩展正则信息准则 (ERIC) 最小化。请参见第 270 页的 “模型拟合详细信息”。仅可用于指数系列分布以及 Lasso 和自适应 Lasso 估计方法。

无 不使用验证。仅可用于 “最大似然估计方法”和 “分位数回归”。

验证列 使用在 “拟合模型”窗口中指定的列作为具有 “验证”角色。最终解是将验证集的负对数似然最小化的解。当指定的估计方法为 Dantzig 选择器或是指定的分布为 “分位数回归”或 “Cox 比例风险”时，该选项不可用。

JMP PRO 提前停止

“提前停止”添加一个提前停止规则：

- 对于“向前选择”，当在模型中添加变量的 10 个连续步骤无法改进验证测度时，该算法终止。解是这 10 个连续步骤的前一个步骤所得的模型。
- 对于 Lasso、弹性网络和岭，当调节参数的 10 个连续值无法改进由验证方法决定的最佳拟合时，该算法终止。解是对应于在 10 个连续值之前的调节参数值的估计值。

注意：对于 AICc 和 BIC 验证方法，在至少四个预测变量进入模型前不会提前停止。

JMP PRO 执行

点击**执行**时，将打开一个报表。报表的标题指定您选择的响应分布、估计方法和验证方法。您可以返回到“模型启动”控制面板来执行更多分析以及选择其他响应分布、估计方法和验证方法。

JMP PRO 模型拟合报表

对于您在 “模型启动”面板中指定的每种估计方法和验证方法，都会生成一个报表。报表在其标题中指定您选择的分布、估计方法和验证方法。

默认提供以下报表：

- 回归图
- 模型汇总
- 估计详细信息（仅为“Lasso”、“弹性网络”和“岭”显示）
- 解路径（对除“最大似然估计方法”和“分位数回归”之外的所有方法显示）
- 原始预测变量的参数估计值
- 效应检验

JMP PRO 回归图

注意：仅当有一个连续预测变量且分类预测变量不超过一个时，“回归图”报表才显示。若“分布”选项设置为“多项式”、“有序型 Logistic”或“Cox 比例风险”，该选项不可用。响应必须是连续的。

“回归图”报表显示垂直轴上的响应值和水平轴上的连续预测变量图。点的上方显示一条回归线。若模型中有分类预测变量，则该分类预测变量的每个水平都有一条单独的回归线，而且图的旁边还会显示一个图例。若响应指定为成功次数的计数和试验次数的计数，则在垂直轴上绘制成功次数除以试验次数的结果。

JMP PRO 模型汇总

“模型汇总”报表说明您拟合的模型并提供有关拟合本身的汇总信息。

JMP PRO 模型描述详细信息

“模型汇总”报表的第一部分提供有关描述您拟合的模型的信息。

响应 “拟合模型”窗口中分配给 Y 角色的列。若使用两列指定区间删失，则会列出这两列的名称。

分布 在“拟合模型”窗口中选择的分布。对于“分位数回归”，还显示响应的指定分位数值。

估计方法 在“模型启动”面板中选择的估计方法。

验证方法 在“模型启动”面板中选择的验证方法。

“均值”模型连结 均值模型的连结函数（基于在“拟合模型”窗口中选择的分布）。

“位置”模型连结 位置参数模型的连结函数，当在“拟合模型”窗口中选择 Cauchy 或 t(5) 作为分布时显示。

“尺度”模型连结 尺度参数模型的连结函数（基于在“拟合模型”窗口中选择的分布）。

“概率”模型连结 概率模型的连结函数（基于在“拟合模型”窗口中选择的分布）。

“离散”模型连结 离散参数模型的连结函数（基于在“拟合模型”窗口中选择的分布）。

“零泛滥”模型连结 零泛滥参数模型的连结函数（基于在“拟合模型”窗口中选择的分布）。

检测下限 在分配给响应列的“检测限值”列属性中指定的检测下限。

检测上限 在分配给响应列的“检测限值”列属性中指定的检测上限。

删失列 在“拟合模型”窗口中分配给“删失”角色的列。

删失代码 “删失”列中指定右删失观测的值。这是在“拟合模型”窗口的“删失代码”列表中指定的值。

JMP PRO 模型拟合详细信息

“模型汇总”报表的第二部分显示与模型拟合有关的统计量。若选择“保留”或“验证列”作为“验证方法”，则为训练集和验证集单独计算这些统计量。若选择“K 重”或“留一法”作为“验证方法”，则“模型汇总”报表的这个部分不可用。

行数 行数。

频数和 “拟合模型”窗口中分配给“频数”或“权重”角色的列值之和。

注意：对于 “- 对数似然”、“BIC”、“AICc”和 “ERIC”，值越小越好。请参见 “统计详细信息”附录中第 477 页的 “似然、 AICc 和 BIC”。

- 对数似然 当前模型的似然函数的自然对数的负数。请参见“统计详细信息”一章中第 477 页的 “似然、 AICc 和 BIC”。

注意：“- 对数似然”对于 “分位数回归”不可用。

目标函数 （仅可用于 “分位数回归”。）该函数值被最小化用以拟合指定的分位数回归模型。最小化的函数是检查损失函数。

参数个数 当前模型中的非零参数数目。

BIC Bayes 信息准则，定义如下：

$$\text{BIC} = -2LogLikelihood + k\ln(n)$$

请参见 “统计详细信息”一章中第 477 页的 “似然、 AICc 和 BIC”。

AICc 校正的 Akaike 信息准则，定义如下：

$$\text{AICc} = -2LogLikelihood + 2k + 2k(k+1)/(n-k-1)$$

请参见 “统计详细信息”一章中第 477 页的 “似然、 AICc 和 BIC”。

ERIC （仅可用于指数系列分布以及指定了 Lasso 或自适应 Lasso 估计方法的情况。）扩展正则信息准则。请参见 Hui et al. (2015)。 ERIC 定义如下：

$$\text{ERIC} = -2LogLikelihood + (k-2)\ln(n\phi/\lambda)$$

其中， λ 是调节参数的值， ϕ 是冗余参数。

广义 R 方 （不可用于 “分位数回归”。）可应用到一般回归模型的一种扩展的 R 方测度。广义 R 方将拟合模型的似然 (L_M) 与仅截距（常数）模型的似然 (L_0) 进行比较。它统一尺度后具有最大值 1。对于 “二项”之外的分布，广义 R 方按以下方式定义：

$$\text{广义R方} = 1 - (L_0/L_M)^{2/n}$$

若 “二项”为指定的分布，广义 R 方按以下方式定义：

$$\text{广义R方} = \frac{1 - (L_0/L_M)^{2/n}}{1 - L_0^{2/n}}$$

广义 R 方值为 1 表示完美模型，值为 0 表示模型不比常数模型更好。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。

注意：指定 “正态”分布时，“广义 R 方”将被 “R 方”取代。

警告：对于使用不同响应分布的模型，不应比较广义 R 方值。只能比较具有给定响应分布的仅截距模型。

R 方 （仅当指定了 “正态”分布时可用。）估计响应中可归因于模型而不是随机误差的变异比例。 R 方值为 1 表示完美模型；值为 0 表示模型不比常数模型更好。 R 方值计算如下：

$$1 - \frac{\sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{N} (Y_i - \bar{Y})^2}$$

调整 R 方 （仅当指定 “正态”分布且估计方法不涉及惩罚时才可用。）针对模型中的参数数目调整 R 方统计量。调整 R 方有利于包含不同数目参数的模型间进行比较。计算中使用自由度。调整 R 方值计算如下：

$$1 - \frac{\frac{1}{N-(p-1)} \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2}{\frac{1}{N-1} \sum_{i=1}^{N} (Y_i - \bar{Y})^2}$$

其中， N 是观测数， p 是参数个数。

注意：若有验证集，仅为训练集报告调整的 R 方统计量。

RASE （仅当指定了 “正态”分布时可用。）平均平方根误差 (RASE) 是当前模型中均方预测误差的平方根。请参见 “标准最小二乘法模型”一章中第 80 页的 “RASE”。

Lambda 惩罚 （仅适用于 Dantzig 选择器、 Lasso、弹性网络、岭和双 Lasso 等估计方法。）当前模型的调节参数 λ 的值。请参见第 285 页的 “估计方法的统计详细信息”。

JMP PRO 估计详细信息

“估计详细信息”报表显示 “Dantzig 选择器”、“Lasso”、“弹性网络”、“岭”和 “双 Lasso”估计方法的 “高级控件”设置。有关这些控件的详细信息，请参见第 266 页的 “高级控件”。

JMP PRO 解路径

注意：为除“最大似然”之外的所有估计方法和除“分位数回归”之外的所有分布显示“解路径”报表。

“解路径”报表显示以下两个图：

- “解路径”图显示估计参数的值。
- 验证图显示对应于选定验证方法的验证统计量的值。

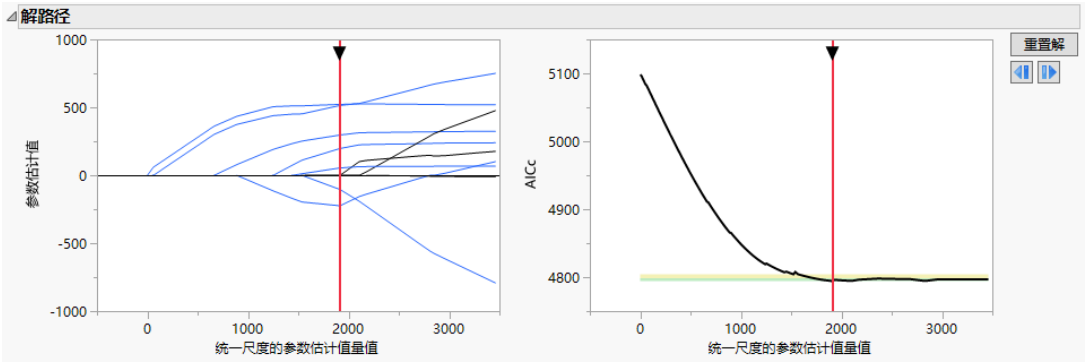
这两个图的水平尺度以统一尺度的参数估计值量值形式给出。这是 l_1 范数，定义为统一尺度的参数估计值的绝对值之和，这些估计值是针对均值的模型估计值。（会从 l_1 范数的计算中排除对应于截距、离散参数和零泛滥参数的估计值。）请注意以下几点：

- 具有大的 l_1 范数值的估计值接近 MLE。
- 具有小的 l_1 范数值的估计值被严重惩罚。
- 当 l_1 范数减小时，调节参数的值增大。

JMP PRO 当前模型指示符

两个图中的红色垂直实线都对应同一个 l_1 范数值，该值是“原始预测变量的参数估计值”报表中显示的解的值。您可以在任意一个图中拖动红色垂直线顶部的箭头来更改罚值的大小，以指示新的当前模型。在验证图中，您还可以点击图中的任意位置来更改模型。拖动红色的垂直线指示新模型时，报表中的结果将相应更新以反映当前选择的模型。垂直虚线仍在最佳拟合模型处。您可以点击“验证图”旁边的**重置解**按钮，返回初始解的垂直红线和相应结果。对于某些验证方法，验证图提供用于标识可比模型的区域。请参见第 275 页的“可比模型区域”。

图 6.5 Diabetes.jmp 的 “解路径”报表，使用 AICc 验证的 Lasso



有关“解路径”图的详细信息，请参见第 274 页的“解路径图”。有关验证图的详细信息，请参见第 274 页的“验证图”。

JMP PRO 解路径图

您可以选择“解路径”图中的路径以突出显示“参数估计值”报表中的相应项。该操作还选择数据表中的相应列。在任意一个报表中选择行将突出显示另一报表中的相应行和“解路径”图中的相应路径。按 Shift 键并点击可以选择多个路径或行。

使用“解路径”图的垂直轴标绘“参数估计值”。它们是**统一尺度的**参数估计值。推导出它们是为了用在使用中心化和统一尺度的预测变量表示的模型中（请参见第 276 页的“中心化和统一尺度的预测变量的参数估计值”）。

预测变量数小于观测值数时，“解路径”图通常显示估计值的整个范围（从零到 MLE 所给出的未惩罚拟合）。否则，该图扩展到接近未惩罚解的大小。从倒数第二个网格点跳到 MLE 解的幅度过大，导致直到倒数第二个网格点的解的详细信息模糊，这时会发生上述情况。出现这样的情况时，只要 MLE 不是最终解，将对“解路径”图重新调整尺度以便轴仅延伸到倒数第二个网格点。

JMP PRO 解 ID

“解路径”中的每个解都会在内部分配有一个解 ID。若您调节参数以选择最初显示的解之外的解，相应的解 ID 将显示在“保存脚本”选项所创建的脚本中。解 ID 是**设置解 ID(N)**命令中的值 N。保存解 ID 可确保能够在运行脚本时重新创建选定的解。

JMP PRO 验证图

验证图显示统计量图，这些图描绘模型拟合在调节参数各值间的优劣程度，换言之，即在“统一尺度的参数估计值量值”各值间的优劣程度。标绘的统计图取决于选定的验证方法。对于每种验证方法，表 6.3 都列出了所标绘的统计量。所有验证方法都是值越小拟合效果越好。对于 K 重和留一法验证方法以及包含三个以上值的验证列，标绘的统计量是各重之间统一尺度的负对数似然值的均值。

表 6.3 中的**统一尺度 -对数似然**是负对数似然除以为其计算负对数似然的集内的观测数所生成的结果。

表 6.3 验证方法以及相应的验证统计量和区域

验证方法	验证统计量	调节参数区域
K 重	*K* 重统一尺度 -对数似然值的均值	2
保留	统一尺度 -对数似然	无
留一法	所有重的统一尺度 -对数似然值的均值	2
BIC	训练数据的 BIC	2
AICc	训练数据的 AICc	2

表 6.3 验证方法以及相应的验证统计量和区域 (续)

验证方法	验证统计量	调节参数区域
ERIC	训练数据的 ERIC	2
具有两个或三个值的验证列	统一尺度 -对数似然	无
K > 3 的值的验证列	*K* 重统一尺度 -对数似然值的均值	2

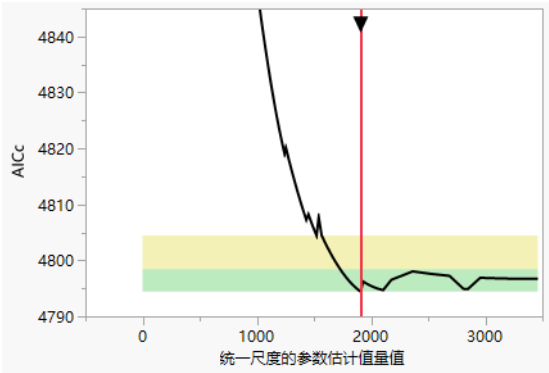
JMP PRO 可比模型区域

尽管估计某个模型是最佳模型，但也可能存在与该选择相关的不确定性。竞争模型可能拟合得几乎一样好，并且可能包含有用信息。对于 AICc、BIC、K 重和留一法验证方法以及包含三个以上值的验证列，验证图提供标识可能值得考虑的竞争模型的区域。不推荐使用落在这些区域之外的模型。请参见 Burnham and Anderson (2004) 和 Burnham et al. (2011)。

区域是验证统计量值的区间。区域标绘为跨整个水平轴的绿色或黄色矩形。若模型的验证统计量的值落在某区域内，则该模型会落在该区域内。您可以拖动红色垂直实线来探索各区域内的解。请参见第 273 页的“当前模型指示符”。

图 6.6 显示了 Diabetes.jmp 的验证图，其垂直轴展开以显示这两个区域。

图 6.6 Diabetes.jmp 的验证图，使用 AICc 验证的 Lasso



针对 BIC、AICc 和 ERIC 验证的区域

对于这些验证方法，图中显示两个区域。用 $V^{最佳}$来表示最佳解的验证 BIC、AICc 和 ERIC 值。

- 绿色区域标识出这样一些模型，有强有力的证据表明会有一个模型与最佳模型一样好。绿色区域是区间 [$V^{最佳}$, $V^{最佳}$+4]。
- 黄色区域标识出这样一些模型，没有什么证据表明会有一个模型与最佳模型一样好。黄色区域是区间 [$V^{最佳}$+4, $V^{最佳}$+10]。

针对 K 重验证、留一法验证和包含三个以上值的验证列的区域

对于这些验证方法，图中显示两个区域。在最佳模型解的位置，对每个验证集都计算统一尺度的负对数似然函数。将这些值的标准误差表示为 L^{SE}。用 $L^{最佳}$表示最佳解的统一尺度的负对数似然。

- 绿色区域标识出这样一些模型，有强有力的证据表明会有一个模型与最佳模型一样好。绿色区域是区间 $[L^{最佳}, L^{最佳}+L^{SE}]$。
- 黄色区域标识出这样一些模型，没有什么证据表明会有一个模型与最佳模型一样好。黄色区域是区间 $(L^{最佳}+L^{SE}, L^{最佳}+2.5*L^{SE}]$。

JMP PRO 中心化和统一尺度的预测变量的参数估计值

“中心化和统一尺度的预测变量的参数估计值”报表提供模型中所有参数的估计值和其他结果。初始表包括模型中的预测变量的系数。另有一个附加表包括尺度、散度或零泛滥参数之类的其他模型参数。请参见第 254 页的“分布”。两个表都包括相同的结果列。

提示：您可以点击“中心化和统一尺度的预测变量的参数估计值”报表中的项来突出显示“解路径”图中的相应路径。同时选定数据表中的对应列。这对于运行进一步的分析很有用。按 Shift 键并点击各项以选择多行。

对于 “广义回归”特质中的所有拟合，将每个预测变量中心化以具有均值零并统一尺度以使平方和为 1：

- 从每个观测值中减去均值。
- 然后将差值除以与每个均值的差值的平方和的平方根。

这使得所有预测变量相对于应用的罚值具有平等的地位。

注意：若在启动窗口中选定了“无截距”选项，则不会对预测变量执行中心化和统一尺度。

“中心化和统一尺度的预测变量的参数估计值”报表给出用中心化和统一尺度的预测变量表示的模型的参数估计值。这些估计值由您指定的验证方法决定。这些估计值在“解路径”图中用红色的垂直线描绘。

该报表提供以下信息：

项 模型项的列表。对于使用 “高级控件”选项强制进入模型的任何项，其旁边会显示 “强制进入”。

估计值 对应于中心化和统一尺度的模型项的参数估计值。

标准误差 估计值的标准误差。这是使用 M 估计和夹层公式（Zou 2006； Huber and Ronchetti 2009）得到的。

Wald 卡方 用于检验参数是否为 0 的 Wald 检验的卡方值。

概率 > 卡方 Wald 检验的 *p* 值。

95% 下限 参数的 95% 置信区间的下限。通过从 “模型规格” 红色小三角菜单中选择 “设置 Alpha 水平”，您可以在 “拟合模型” 窗口中更改 α 水平。

95% 上限 参数的 95% 置信区间的上限。通过从 “模型规格” 红色小三角菜单中选择 “设置 Alpha 水平”，您可以在 “拟合模型” 窗口中更改 α 水平。

奇异性详细信息 （仅当模型项之间存在线性相依性时可用。）模型项所满足的线性函数。

JMP PRO 原始预测变量的参数估计值

“原始预测变量的参数估计值” 报表提供模型中所有参数的估计值和其他结果。初始表包括模型中的预测变量的系数。另有一个附加表包括尺度、散度或零泛滥参数之类的其他模型参数。请参见第 254 页的 “分布”。两个表都包括相同的结果列。

提示：您可以点击 “原始预测变量的参数估计值” 报表中的项来突出显示 “解路径” 图中的相应路径。同时选定数据表中的对应列。这在运行进一步的分析时很有用。按 Shift 键并点击各项以选择多行。

“原始预测变量的参数估计值” 报表给出用原始（未中心化和统一尺度的）预测变量表示的模型的参数估计值。

该报表提供以下信息：

项 模型项的列表。对于使用 “高级控件” 选项强制进入模型的任何项，其旁边会显示 “强制进入”。

估计值 对应于用原始测量值表示的模型项的参数估计值。

标准误差 估计值的标准误差。这是使用 M 估计和夹层公式（Zou 2006 和 Huber and Ronchetti 2009）得到的。

Wald 卡方 用于检验参数是否为 0 的 Wald 检验的卡方值。

概率 > 卡方 Wald 检验的 *p* 值。

95% 下限 参数的 95% 置信区间的下限。通过从 “模型规格” 红色小三角菜单中选择 “设置 Alpha 水平”，您可以在 “拟合模型” 窗口中更改 α 水平。

95% 上限 参数的 95% 置信区间的上限。通过从 “模型规格” 红色小三角菜单中选择 “设置 Alpha 水平”，您可以在 “拟合模型” 窗口中更改 α 水平。

奇异性详细信息 （仅当模型项之间存在线性相依性时可用。）模型项所满足的线性函数。

VIF （仅当分布为 “正态” 时才可用。）仅当您在参数估计值表中右击并选择 “列” > “VIF” 时才显示。）显示模型中每个项的方差膨胀因子 (VIF)。高 VIF 值指示模型中的项存在共线性问题。

第 i 个项 x_i 的 VIF 计算如下：

$$VIF_i = \frac{1}{1 - R_i^2}$$

其中， R_i^2 是作为其他解释变量函数的 x_i 的回归的 R 方。

JMP PRO 活跃参数估计值

“活跃参数估计值”报表提供“原始预测变量的参数估计值”报表的子集。“活跃参数估计值”报表仅显示非零参数。

JMP PRO 效应检验

“效应检验”报表给出以下信息：

源 模型中效应的列表。

参数数目 与效应关联的参数数目。

自由度 Wald 卡方检验的自由度。这是与模型中的效应关联的非零参数估计值的个数。

Wald 卡方 Wald 检验的卡方值，该检验检测与效应关联的所有参数是否为零。

概率 > 卡方 Wald 卡方检验的 *p* 值。

若分布为“正态”而且估计方法中没有罚值，则以下列会代替“Wald 卡方”和“概率 > 卡方”显示在报表中：

平方和 效应为零假设下的平方和。

F 比 用于检验效应为零的 F 统计量。F 比是效应均方与误差均方之比。效应均方是效应平方和除以效应自由度的结果。

概率 > F 效应检验的 *p* 值。

若某个效应的系数已估计为零，则：

- 若该效应有一个自由度，则在该效应对应的行的最右侧出现“已删除”字样。
- 若该效应有多个自由度，则显示短语“删除的水平数”，后跟对应于参数估计值为零的项的水平数。

JMP PRO 模型拟合选项

每个模型拟合报表具有带以下选项的红色小三角菜单：

警告：若在启动窗口中指定的列具有“表达式”数据类型或“向量”建模类型，则该平台中的许多选项都不可用。

回归报表 支持您定制为指定的模型拟合所显示的报表。除了 “中心化和统一尺度的预测变量的参数估计值”报表和 “活跃参数估计值”报表，默认显示以下所有报表。

模型汇总 显示或隐藏包括有关模型的规格和拟合优度统计量信息的“模型汇总”报表。该选项还为适用的模型显示“估计详细信息”报表。请参见第 270 页的“模型汇总”和第 272 页的“估计详细信息”。

解路径 （不可用于最大似然模型。）显示或隐藏“解路径”图和“验证路径”图。请参见第 273 页的“解路径”。

中心化和统一尺度的预测变量的参数估计值 显示或隐藏中心化和统一尺度的参数估计值表。请参见第 276 页的“中心化和统一尺度的预测变量的参数估计值”。

原始预测变量的参数估计值 显示或隐藏采用数据原始尺度的参数估计值表。请参见第 277 页的“原始预测变量的参数估计值”。

活跃参数估计值 （不可用于最大似然或岭回归模型。）显示或隐藏当前选定模型的活跃或非零参数估计值的表。

显示解路径汇总 （不可用于最大似然或岭回归模型。）显示或隐藏一个报表，其中包含解路径上的点的拟合统计量的表以及活跃集发生更改的验证路径图。可用的统计量取决于估计方法。有关可用于“使用‘BIC 验证’的‘正态’‘Lasso’”模型的条件模型概率的详细信息，请参见 Hu et al.(2019)。当指定 BIC、AICc 或 ERIC 验证时，BIC 和 AICc 列中的单元格按照与“验证图”相同的方式着色。请参见第 275 页的“可比模型区域”。

效应检验 显示或隐藏每个效应的检验。给定效应的效应检验用来检验以下原假设：与该效应关联的所有参数均为零。根据名义型或有序型效应的水平数，此类效应可有若干关联参数。此类效应的效应检验用于检验所有关联参数是否均为零。当分布为“多项式”时，将针对响应的各个水平合并效应。请参见第 278 页的“效应检验”。

显示预测表达式 显示或隐藏包含估计模型的方程的 “预测表达式”报表。相关示例，请参见“标准最小二乘法模型”一章中第 103 页的 “显示预测表达式”。

选择非零项 在报表中突出显示具有非零系数的项。同时选定数据表中的所有关联列。将 “岭回归”选作估计方法时，该选项不可用。

选择清零项 在报表中突出显示具有零系数的项。同时选定数据表中的所有关联列。将 “岭回归”选作估计方法时，该选项不可用。

重新启动活跃集 （不适用于包含具有 “矢量”建模类型的预测变量的模型。）包含的选项用于打开 “拟合模型”启动窗口，其中的 “构造模型效应”列表包含基于具有非零参数估计值的项的一组项。这些项即**活跃**效应。启动窗口中的所有其他规格都是在原始分析中使用过的。

注意：若您在包含 “依据”变量的报表中选择任何 “重新启动活跃集”选项，该 “依据”变量不会添加至 “拟合模型”启动窗口。

使用活跃效应重新启动 仅使用活跃效应填充“构造模型效应”列表。

重新启动活跃的主效应和二阶析因 使用用活跃效应构造的二阶析因填充“构造模型效应”列表。

重新启动活跃的主效应和三阶析因 使用用活跃效应构造的三阶析因填充“构造模型效应”列表。

重新启动活跃的主效应和完全析因 使用用活跃效应构造的完全析因填充“构造模型效应”列表。

重新启动活跃的主效应和二阶多项式 使用用活跃效应构造的二阶多项式填充“构造模型效应”列表。

重新启动活跃的主效应和三阶多项式 使用用活跃效应构造的三阶多项式填充“构造模型效应”列表。

重新启动活跃的主效应和响应曲面模型 使用用活跃效应构造的响应曲面模型填充“构造模型效应”列表。

隐藏不活动的路径 调整“解路径参数估计值”图中不活动路径的透明度以使当前不活动的路径变淡显示。

优势比 （仅当指定的分布为“二项”并且模型包含截距时才可用。不适用于包含具有“矢量”建模类型的预测变量的模型。）显示一个报表，其中包含分类预测变量的优势比，以及连续预测变量的单位优势比和范围优势比。**优势比**是两个事件的优势之比。事件的**优势**是指相关事件发生的概率与该事件不发生的概率的对比情况。相关事件由“拟合模型”启动窗口中的“目标水平”来定义。

对于每个分类预测变量，都显示一个“优势比”报表。为分类模型项水平的所有组合都显示优势比。

若存在连续预测变量，还会显示另外两个报表：

– “单位优势比”报表。针对连续模型项中一个单位的更改计算**单位优势比**。

– “范围优势比”报表。针对连续模型项中整个范围的更改计算**范围优势比**。

“优势比”报表中的置信区间为基于 Wald 的区间。请注意，仅当模型项未涉及任何更高阶效应时该模型项的优势比才有意义。

注意：若模型中包含交互作用，您可以使用“多重比较”选项获取优势比。请参见第 281 页的 “多重比较”。

发生率比 （仅当指定的分布为“Poisson”或“负二项”并且模型包含截距时才可用。）显示一个报表，其中包含分类预测变量的发生率比，以及连续预测变量的单位发生率比和范围发生率比。**发生率比**是两个事件的发生率之比。模型项的**发生率**是发生在给定时间期间的新事件数。

对于每个分类预测变量，都显示一个“发生率比”报表。为分类模型项水平的所有组合都显示发生率比。

若存在连续预测变量，还会显示另外两个报表：

– “单位发生率比”报表。针对连续模型项中一个单位的更改计算**单位发生率比**。

– “范围发生率比”报表。针对连续模型项中整个范围的更改计算**范围发生率比**。

“发生率比”报表中的置信区间为基于 Wald 的区间。请注意，仅当模型项未涉及任何更高阶效应时该模型项的发生率比才有意义。

危险比 （仅当指定的“分布”为“Cox 比例风险”时才可用。）显示一个报表，其中包含分类预测变量的危险比，以及连续预测变量的单位危险比和范围危险比。**危险比**是两个事件的危险比率之比。某事件在时间 *t* 时的危险率是该事件不会额外存续一段时间的条件概率，前提是该事件已经存续到时间 *t*。

对于每个分类预测变量，都显示一个“危险比”报表。为分类模型项水平的所有组合都显示危险比。

若存在连续预测变量，还会显示另外两个报表：

– “单位危险比”报表。针对连续模型项中一个单位的更改计算**单位危险比**。

– “范围危险比”报表。针对连续模型项中整个范围的更改计算**范围危险比**。

“危险比”报表中的置信区间为基于 Wald 的区间。请注意，仅当模型项未涉及任何更高阶效应时该模型项的危险比才有意义。

估计值的协方差 显示一个矩阵，其中显示参数估计值的协方差。这些值是使用 M 估计和夹层公式（Zou 2006 和 Huber and Ronchetti 2009）计算得到的。协方差矩阵不包含零项。

估计值的相关性 显示一个矩阵，其中显示参数估计值的相关性。这些值是使用 M 估计和夹层公式（Zou 2006 和 Huber and Ronchetti 2009）计算得到的。相关性矩阵不包含零项。

逆预测 （不适用于包含具有“矢量”建模类型的预测变量的模型。）在给定 Y 和其他 X 变量的特定值的情况下预测某个 X 值。只能用于预测连续变量。有关“逆预测”的详细信息，请参见“标准最小二乘法模型”一章中第 125 页的“逆预测”。

多重比较 （不适用于包含具有“矢量”建模类型的预测变量的模型，或不包含任何分类预测变量的模型。）显示“多重比较”启动窗口。有关“多重比较”启动窗口和报表的详细信息，请参见“标准最小二乘法模型”一章中第 112 页的“多重比较”。请注意多重比较是在线性预测变量尺度上执行的。若指定的分布为“二项式”，则针对优势比执行多重比较。若指定的分布为 Poisson，则针对发生率比执行多重比较。若指定的分布为“Cox 比例风险”，则针对危险比执行多重比较。

混淆矩阵 （仅当指定的分布为“二项式”、“多项式”或“有序型 Logistic”时才可用。）显示一个矩阵，该矩阵是实际响应水平与预测响应水平的表格。对于好的模型，预测响应水平应与实际响应水平相同。混淆矩阵支持您评估预测响应与实际响应的相符程度。若使用了验证，将为每一个训练集、验证集和测试集都显示一个混淆矩阵。

设置概率阈值 （仅当指定的分布为“二项”时才可用。）指定用于对响应分类的截止概率。默认情况下，若某个观测的预测概率超过 0.5，则该观测将分类到“目标水平”中。更改该阈值，指定 0.5 之外的值作为分类到“目标水平”的截止值。混淆矩阵中的“预测比率”随之更新，根据指定的阈值以反映分类。

若响应具有“收益矩阵”列属性，则概率阈值的初始值由收益矩阵确定。

刻画器 （不适用于包含具有“矢量”建模类型的预测变量的模型。）提供支持您探索拟合模型的各种刻画器。

注意：若行数小于等于 500 而预测变量数小于等于 30，当您在任一个 “解路径”图中拖动当前模型指示符时，刻画器图将随之不断更新。如若不然，刻画器图将在您释放鼠标按钮时更新。

刻画器 显示“预测刻画器”。参数估计值为零且未涉及任何带有非零系数的交互作用项的预测变量不显示在该刻画器中。有关“预测刻画器”的详细信息，请参见《刻画器指南》。

“分布”刻画器 （当指定的分布为“二项”或“分位数回归”时不可用。）显示预测变量与响应的累积分布函数的刻画器。该响应显示在最右侧的单元格中。

分位数刻画器 （当指定的分布为“二项”或“分位数回归”时不可用。）显示一个刻画器，该刻画器显示预测响应（作为预测变量的函数）以及累积分布函数的分位数。该分位数称为“概率”，显示在最右侧的单元格中。

生存刻画器 （仅当指定的分布为“正态”、“指数”、“Weibull”、“对数正态”或“Cox 比例风险”时才可用。）显示一个刻画器，该刻画器显示作为预测变量和响应的函数的生存函数。该响应显示在最右侧的单元格中。

危险率刻画器 （仅当指定的分布为“正态”、“指数”、“Weibull”、“对数正态”或“Cox 比例风险”时才可用。）显示一个刻画器，该刻画器显示作为预测变量和响应的函数的危险率。该响应显示在最右侧的单元格中。

定制检验 显示支持您检验定制假设的 “定制检验”报表。若模型有 “解路径”，定制检验结果将随着您更新解而更新。有关定制检验的详细信息，请参见“标准最小二乘法模型”一章中第 109 页的 “定制检验”。“定制检验”红色小三角菜单包含一个用于删除 “定制检验”报表的选项。

诊断图 提供各种图，用来帮助评估当前模型拟合的好坏。若指定了“验证”列，或者选择“K 重”、“保留”或 “留一法”作为 “验证方法”，下面的选项会支持您查看训练集、验证集以及测试集 （若适用），或是为这些集构造相应的图。若选择 “K 重”或 “留一法”，则这些图对应于优化预测误差的验证集及其相应的训练集。请参见第 268 页的 “K 重”。

注意：所有诊断图都会随着您在任一个 “解路径”图中拖动当前模型指示符时不断更新。

诊断包 （当指定的分布为“二项式”、“多项式”、“有序型 Logistic”或“Cox 比例风险”时不可用。）显示一组四个图形：“预测值-残差”图、“行号-残差”图、残差直方图以及观测响应大于观测到的响应的概率直方图。

这些图形都是使用所有观测构造的。若使用了“验证”列，或者选择“K 重”、“保留”或“留一法”作为“验证方法”，则可通过复选框选择训练集、验证集以及测试集（若适用）。与这些集对应的行在数据表中会被相应地选定，而且相应的点和区域会在图形中突出显示。使用该选项可确定模型拟合在各个集之间是否相似。

“观测较大响应的拟合概率”直方图帮助您评估模型的拟合优度。基于分布应用不同准则：

– 对于零泛滥分布和分位数回归之外的分布，“正确”模型应显示近似均匀的值分布。
– 对于零泛滥分布，该直方图应在零处显示点质量，在其他位置显示近似均匀分布。
– 对于分位数回归，该直方图应在指定的分位数左侧显示近似均匀的值分布，在指定的分位数右侧显示稍微高一些的值的近似均匀分布。

标绘基线生存和危险率 （仅当指定的“分布”为“Cox 比例风险”时才可用。）显示“基线生存和危险率”图，这些图为基线比例风险函数标绘随响应变量变化的生存和危险率函数。在这些图下方，有一个包含标绘值的表。

注意：若指定的分布为“Cox 比例风险”，则“标绘基线生存和危险率”选项是唯一可用的诊断图。

ROC 曲线 （仅当指定的分布为“二项式”、“多项式”或“有序型 Logistic”时才可用。）显示“受试者操作特征 (ROC)”曲线。若使用了验证，将为每一个训练集、验证集和测试集都显示一条 ROC 曲线。

ROC 曲线测量拟合概率正确地对响应水平分类的能力。曲线距离对角线越远，拟合效果越好。有关 ROC 曲线的简介，请参见《基本分析》。

若响应的水平数超过两个，则“ROC 曲线”图会为每个响应水平显示一条 ROC 曲线。对于给定的响应水平，该曲线是正确分类到该水平的 ROC 曲线。有关 ROC 曲线的详细信息，请参见《预测和专业建模》。

提升曲线 （仅当指定的分布为“二项式”、“多项式”或“有序型 Logistic”时才可用。）显示模型的提升曲线。若使用了验证，将为每一个训练集、验证集和测试集都显示一条 ROC 曲线。

提升曲线显示在拟合概率降低时响应水平被分类的有效程度。拟合概率沿水平轴以降序标绘。拟合概率的垂直坐标是该概率（或更高）的正确分类的比例除以总正确分类率的结果。使用提升曲线可查看在您仅选择拟合概率超过阈值的那些观测的情况下，您是否能够对较大比例的观测正确分类。

若响应的水平数超过两个，则“提升曲线”图会为每个响应水平显示一条提升曲线。对于给定的响应水平，该曲线是正确分类到该水平的提升曲线。有关提升曲线的详细信息，请参见《预测和专业建模》。

标绘“预测值-实际值”图 （当指定的分布为“二项式”、“多项式”、“有序型 Logistic”或“Cox 比例风险”时不可用。）在垂直轴上标绘实际 Y 值，在水平轴上标绘预测的 Y 值。若使用了验证，将为每一个训练集、验证集和测试集都显示一个图。

标绘“预测值-残差”图 （当指定的分布为“二项式”、“多项式”、“有序型 Logistic”或“Cox 比例风险”时不可用。）在垂直轴上标绘残差，在水平轴上标绘预测的 Y 值。若使用了验证，将为每一个训练集、验证集和测试集都显示一个图。

标绘“预测变量-残差”图 （当指定的分布为“二项式”、“多项式”、“有序型 Logistic”或“Cox 比例风险”时不可用。不适用于包含具有“矢量”建模类型的预测变量的模型。）对于模型中的每个预测变量，在垂直轴上标绘残差，在水平轴上标绘预测变量值。模型中的每个预测变量都对应一个图。若使用了验证，将为每一个训练集、验证集和测试集都显示一组图。

正态分位数图 （仅当指定的分布为“正态”且无删失时才可用。）在垂直轴上绘制正态分位数，在水平轴上绘制标准化残差。若使用了验证，将为每一个训练集、验证集和测试集都显示一个图。

保存列 支持您将基于拟合模型的列保存到数据表中。有关在将“Cox 比例风险”选作分布时可用的选项的详细信息，请参见第 285 页的“保存用于 Cox 比例危险模型的列选项”。对于其他所有分布，可以将以下列保存到数据表中：

保存函数预测公式 （仅当响应列包含“FDE FPC 数”列属性时才可用。）将每个 FDE 主成分响应对应的列保存到包含每个函数主成分的预测公式的数据表。该选项还添加模型预测公式，该公式是预测公式与“函数数据分析器”平台中的特征函数列的线性组合。将向数据表中添加一个脚本，该脚本支持您使用模型预测公式刻画原始响应，该响应在 FDE 主成分响应列的“FDE 输出”列属性中指定。有关函数主成分的详细信息，请参见《预测和专业建模》。

注意：“保存函数预测公式”选项用于保存报表窗口中所有 FDE 主成分响应的公式列。若为单个响应拟合多个模型，则使用每个响应的最终模型来创建该响应的预测公式。

保存预测公式 将包含预测公式（用观测的（非标准化）数据值形式给出）的列保存到数据表中。预测公式不包含清零项。有关均值公式，请参见第 288 页的“分布的统计详细信息”。

若响应列为分类响应，该选项将为每个响应水平创建一个概率列，并创建一个包含最可能响应的列。最可能的响应列包含具有基于模型的最高概率的水平。若“概率阈值”是 0.5 之外的值，该选项可创建一个附加列，其中包含基于概率阈值的最可能的响应。

均值置信区间 将包含响应均值的 95% 置信下限和上限的两列保存到数据表中。

注意：通过从“模型规格”红色小三角菜单中选择“设置 Alpha 水平”，您可以在“拟合模型”窗口中更改置信区间的 α 水平。

预测值标准误差 将包含预测响应均值的标准误差的列保存到数据表中。

保存残差公式 将包含残差公式（用 Y 减去预测公式形式给出）的列保存到数据表中。残差公式不包含清零项。若将“二项”选作分布，则不可用。

保存方差公式 将包含预测方差公式的列保存到数据表中。使用选定分布的方差公式计算预测的方差。通过将连结函数的反函数应用到估计的线性成分来估计连结函数中涉及的参数值。其他参数用其估计值替代。有关方差公式，请参见第 288 页的“分布的统计详细信息”。若将“二项”选作分布，则不可用。

保存线性预测变量 将包含设计矩阵与参数估计值向量乘积公式的列保存到数据表中。这通常称为 $\mathbf{X}\beta$。该公式不包含清零项。

保存验证列 （仅当指定的“验证方法”为“K 重”、“保留”或“留一法”时才可用。）保存一个列，该列说明如何将行分配给重。对于“K 重”，该列列出各行被分配到第几重。对于“保留”，将每行标识为属于训练集或验证集。对于“留一法”，行的值指示它被剔除的顺序。

注意：若您在启动窗口中选择了“验证”列，将不显示“保存验证列”选项。

保存分布公式 （当指定的分布为“二项”或“分位数回归”时不可用。）将包含累积分布函数公式的列保存到数据表中。

保存生存公式 （仅当指定的分布为连续时才可用。）将包含观测时的生存概率公式的列保存到数据表中。生存函数等于 1 减累积分布函数。

保存模拟公式 将使用估计参数为您拟合的模型生成模拟值的公式所在的列保存到数据表中。该列可用在模拟实用工具中，作为“换入列”。请参见《基本分析》。

Cook 距离影响 （仅当指定的分布为“正态”且指定的估计方法为“标准最小二乘法”时才可用。）将包含“Cook 距离影响”统计量值的列保存到数据表中。

帽子值 （仅当指定的分布为“正态”且指定的估计方法为“标准最小二乘法”时才可用。）将包含 $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ 的对角线元素的列保存到数据表中。这些值有时称为**帽子值**。

发布预测公式 创建预测公式并将它们保存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见《预测和专业建模》。

保存用于 Cox 比例危险模型的列选项

保存生存公式 将包含观测时的生存概率公式的列保存到数据表中。

保存 Cox Snell 残差公式 将包含 Cox-Snell 残差公式的列保存到数据表中。Cox-Snell 残差为严格正数。有关 Cox-Snell 残差的说明，请参见 Meeker and Escobar (1998, sec. 17.6.1)。

保存 Martingale 残差公式 将包含 Martingale 残差公式的列保存到数据表中。Martingale 残差定义为某个个体的观测事件数与有条件预期事件数之间的差值。Martingale 残差的均值为零，范围介于负无穷大和 1 之间。请参见 Fleming and Harrington (1991)。

保存线性预测变量 将包含设计矩阵与参数估计值向量乘积公式的列保存到数据表中。这通常称为 $\mathbf{X}\beta$。该公式不包含清零项。

删除拟合 删除拟合的报表。

JMP PRO “广义回归”特质的统计详细信息

- “估计方法的统计详细信息”
- “高级控件的统计详细信息”
- “分布的统计详细信息”

JMP PRO 估计方法的统计详细信息

惩罚回归方法通过赋予回归系数罚值引入了偏倚。

JMP PRO 岭回归

在岭回归期间，回归系数应用了 l_2 罚值。按以下公式给出岭回归系数的估计值：

$$\hat{\beta}^{岭} = \text{argmin}_{\beta}\left\{\sum_{i=1}^{N} -LogLikelihood(\beta;y_i) + \frac{\lambda}{2}\sum_{j=1}^{p}\beta_j^2\right\},$$

其中，$\sum_{j=1}^{p}\beta_j^2$ 是 l_2 罚值，λ 是调节参数，N 是行数，p 是变量数。

JMP PRO Dantzig 选择器

在 Dantzig 选择器计算期间，回归系数应用了 l_∞ 罚值。Dantzig 选择器的系数估计值满足以下准则：

$$\min_{\beta}\left\|X^T(y - X\beta)\right\|_\infty \qquad \text{受限于 } \|\beta\|_1 \le t$$

其中，$\|v\|_\infty$ 表示 l_∞ 范数，该范数是向量 v 的分量的最大绝对值。

JMP PRO Lasso 回归

在 Lasso 期间，回归系数应用了 l_1 罚值。按以下公式给出 Lasso 的系数估计值：

$$\hat{\beta}^{lasso} = \text{argmin}_{\beta}\left\{\frac{1}{2}\sum_{i=1}^{N} -LogLikelihood(\beta;y_i) + \lambda\sum_{j=1}^{p}|\beta_j|\right\},$$

其中，$\sum_{j=1}^{p}|\beta_j|$ 是 l_1 罚值，λ 是调节参数，N 是行数，p 是变量数

JMP PRO 弹性网络

弹性网络组合 l_1 和 l_2 罚值。按以下公式给出弹性网络的系数估计值：

$$\hat{\beta}^{enet} = \text{argmin}_{\beta}\left\{\sum_{i=1}^{N} -LogLikelihood(\beta;y_i) + \lambda\sum_{j=1}^{p}\left(\alpha|\beta_j| + \frac{(1-\alpha)}{2}\beta_j^2\right)\right\},$$

这是等式中使用的符号：

- $\sum_{j=1}^{p}|\beta_j|$ 是 l_1 罚值
- $\sum_{j=1}^{p}\beta_j^2$ 是 l_2 罚值

- λ 是调节参数
- α 是一个参数，用于确定 l_1 和 l_2 罚值的组合值
- N 是行数
- p 是变量数

提示：有两个样本脚本，这两个脚本演示变动单个预测变量的弹性网络中的 α 和 λ 的收缩影响。选择**帮助 > 样本数据**，点击**打开样本脚本目录**，然后选择 demoElasticNetAlphaLambda.jsl 或 demoElasticNetAlphaLambda2.jsl。每个脚本都包含关于使用方法和演示内容的说明。

JMP PRO 自适应方法

自适应 Lasso 方法使用加权罚值来提供一致的系数估计值。l_1 罚值的加权形式为

$$\sum_{j=1}^{p} \frac{|\beta_j|}{\tilde{\beta}_j},$$

其中，若 MLE 存在，则 $\tilde{\beta}_j$ 是 MLE。若 MLE 不存在并且响应分布为正态分布，则使用最小二乘法进行估计，而 $\tilde{\beta}_j$ 是使用广义逆矩阵求得的解。若响应分布非正态分布，则 $\tilde{\beta}_j$ 是岭解。

对于自适应 Lasso，这个加权形式的 l_1 罚值用于确定 $\hat{\beta}^{lasso}$ 系数。

自适应弹性网络使用这个加权形式的 l_1 罚值，也应用 l_2 罚值的加权形式。自适应弹性网络的 l_2 罚值的加权形式为

$$\sum_{j=1}^{p} \left(\frac{\beta_j}{\tilde{\beta}_j}\right)^2,$$

其中，若 MLE 存在，则 $\tilde{\beta}_j$ 是 MLE。若 MLE 不存在并且响应分布为正态分布，则使用最小二乘法进行估计，而 $\tilde{\beta}_j$ 是使用广义逆矩阵求得的解。若响应分布非正态分布，则 $\tilde{\beta}_j$ 是岭解。

JMP PRO 高级控件的统计详细信息

JMP PRO 网格

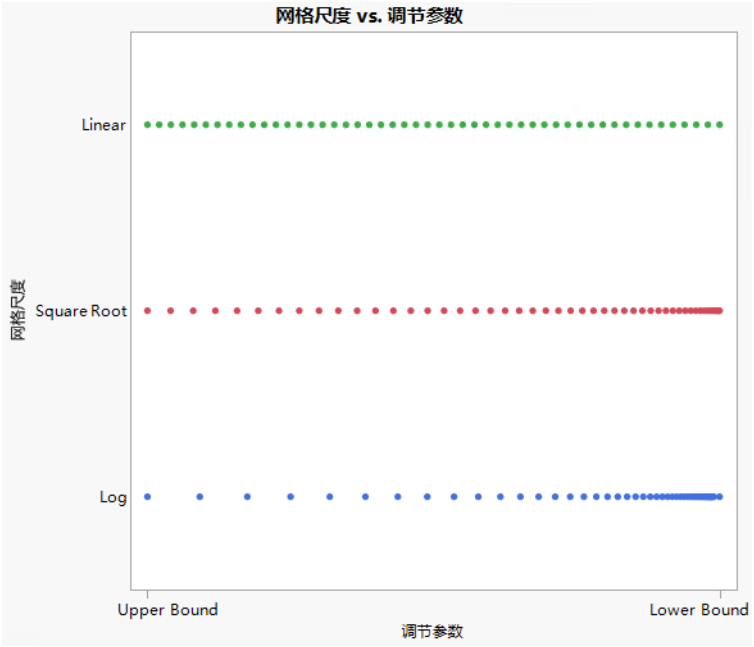
通过搜索调节参数值的网格来找到最大限度使惩罚似然的岭回归和 Lasso 最小化的调节参数。这个网格值位于调节参数的下限和上限之间。您可以在“高级控件”下指定网格点数目。

下限为零，但是在特殊情况下它被设置为 0.0001。请参见第 266 页的“调节参数”。调节参数的下限为零时，不惩罚解，系数为 MLE。上限是使所有非截距项为零的最小值。

以迭代方式搜索介于上限和下限之间的网格值，以确定调节参数的最佳值。可以用三个不同的尺度设置可能的调节参数网格：线性、对数和平方根。默认网格尺度是平方根。

在某些情况下，未惩罚的估计值和上一步的估计值相差很大。这么大的差异可能使解路径失真。对数尺度侧重搜索大值不多的小调节参数值，而线性尺度则均匀分散搜索，从最小值搜索到最大值。平方根尺度是这两种尺度的折衷。图 6.7 显示不同的网格尺度。

图 6.7 调节参数网格尺度的选项



JMP PRO 分布的统计详细信息

下面根据模型拟合中使用的参数给出了按“广义回归”特质拟合的分布。尽管没有专门在说明中指出，但是“广义回归”特质允许您对离散分布指定非整数值。

JMP PRO 连续分布

正态分布

$$f(y|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left[-\frac{1}{2\sigma^2}(y-\mu)^2\right], -\infty < y < \infty$$

$$E(Y) = \mu$$

$$Var(Y) = \sigma^2$$

Cauchy 分布

$$f(y|\mu,\sigma) = \left\{\pi\sigma\left[1+\left(\frac{y-\mu}{\sigma}\right)^2\right]\right\}^{-1}, -\infty < y < \infty$$

$$E(Y) = \text{未定义}$$

$$Var(Y) = \text{未定义}$$

t(5) 分布

$$f(y|\mu,\sigma) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)}\frac{1}{\sqrt{\nu\pi\sigma^2}}\left[1+\frac{(x-\mu)^2}{\nu\sigma^2}\right]^{-\left(\frac{\nu+1}{2}\right)}, -\infty < y < \infty, \nu = 5$$

$$E(Y) = \mu$$

$$Var(Y) = \sigma^2\frac{\nu}{\nu-2}$$

指数分布

$$f(y|\theta) = \frac{1}{\mu}\exp\left[-\frac{y}{\mu}\right], y > 0$$

$$E(Y) = \mu$$

$$Var(Y) = \mu^2$$

Gamma 分布

$$f(y|\mu,\sigma) = \frac{y^{(\mu/\sigma)-1}\exp[-y/\sigma]}{\Gamma[\mu/\sigma]\sigma^{\mu/\sigma}}, y > 0$$

$$E(Y) = \mu$$

$$Var(Y) = \mu\sigma$$

Weibull 分布

$$f(y|\mu,\sigma) = \frac{1}{y\sigma}\exp\left[\frac{\log(y)-\mu}{\sigma}\right]\exp\left\{-\exp\left[\frac{\log(y)-\mu}{\sigma}\right]\right\}, y > 0$$

$$E(Y) = \exp(\mu)\Gamma[1+\sigma]$$

$$Var(Y) = \exp(2\mu)\{\Gamma[1+2\sigma] - (\Gamma[1+\sigma])^2\}$$

对数正态分布

$$f(y|\mu,\sigma) = \frac{1}{y\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma^2}[\log(y)-\mu]^2\right\},\ y>0$$

$$E(Y) = \exp\left(\mu+\frac{\sigma^2}{2}\right)$$

$$Var(Y) = [\exp(\sigma^2)-1]\exp(2\mu+\sigma^2)$$

Beta 分布

$$f(y|\mu,\sigma) = \frac{\Gamma[1/\sigma]}{\Gamma[\mu/\sigma]\Gamma[(1-\mu)/\sigma]}y^{\mu/\sigma-1}(1-y)^{\frac{(1-\mu)}{\sigma}-1},\ y\in(0,1)$$

$$E(Y) = \mu$$

$$Var(Y) = \mu(1-\mu)\frac{\sigma}{\sigma+1}$$

JMP PRO 离散分布

二项分布

$$f(y|n,p) = \binom{n}{y}p^y(1-p)^{n-y},\ y = 0,1,2,\ldots,n$$

$$E(Y) = np$$

$$Var(Y) = np(1-p)$$

Beta 二项分布

$$f(y|n,p,\delta) = \binom{n}{y}\frac{\Gamma\left[\frac{1}{\delta}-1\right]\Gamma\left[y+p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-y+(1-p)\left(\frac{1}{\delta}-1\right)\right]}{\Gamma\left[p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[(1-p)\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-1+\frac{1}{\delta}\right]},\ y = 0,1,2,\ldots,n$$

$$E(Y) = np$$

$$Var(Y) = np(1-p)[1+(n-1)\delta]$$

Poisson 分布

$$f(y|\lambda) = \frac{\lambda^y}{y!}\exp(-\lambda),\ y = 0, 1, 2, \ldots$$

$$E(Y) = \lambda$$

$$Var(Y) = \lambda$$

负二项分布

$$f(y|\mu, \sigma) = \frac{\Gamma[y+(1/\sigma)]}{\Gamma[y+1]\Gamma[1/\sigma]}\left[\frac{(\mu\sigma)^y}{(1+\mu\sigma)^{y+(1/\sigma)}}\right],\ y = 0, 1, 2, \ldots$$

$$E(Y) = \mu$$

$$Var(Y) = \mu + \sigma\mu^2$$

JMP PRO 零泛滥分布

零泛滥二项分布

$$f(y|n, p, \pi) = \begin{cases} \pi + (1-\pi)(1-p)^n, & \text{对于 } y = 0 \\ (1-\pi)\binom{n}{y}p^y(1-p)^{n-y}, & \text{对于 } y = 1, 2, \ldots, n \end{cases}$$

$$E(Y) = np(1-\pi)$$

$$Var(Y) = (1-\pi)np(1-p) + n^2p^2(1-\pi)^2$$

零泛滥 Beta 二项分布

$$f(y|n, p, \delta, \pi) = \begin{cases} \pi + (1-\pi)\dfrac{\Gamma\left[\frac{1}{\delta}-1\right]\Gamma\left[p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n+(1-p)\left(\frac{1}{\delta}-1\right)\right]}{\Gamma\left[p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[(1-p)\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-1+\frac{1}{\delta}\right]}, & \text{对于 } y = 0 \\ (1-\pi)\binom{n}{y}\dfrac{\Gamma\left[\frac{1}{\delta}-1\right]\Gamma\left[y+p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-y+(1-p)\left(\frac{1}{\delta}-1\right)\right]}{\Gamma\left[p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[(1-p)\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-1+\frac{1}{\delta}\right]}, & y = 1, 2, \ldots, n \end{cases}$$

$$E(Y) = np(1-\pi)$$

$$Var(Y) = (1-\pi)np\{1+(1-p)[1+(n-1)\delta]\} - [np(1-\pi)]^2$$

零泛滥 Poisson 分布

$$f(y|\lambda,\pi) = \begin{cases} \pi + (1-\pi)\exp[-\lambda] \text{ ，对于 } y = 0 \\ (1-\pi)\dfrac{\lambda^y}{y!}\exp[-\lambda] \text{ ，对于 } y = 1, 2, \ldots \end{cases}$$

$$E(Y) = (1-\pi)\lambda$$

$$Var(Y) = \lambda(1-\pi)(1+\lambda\pi)$$

零泛滥负二项分布

$$f(y|\mu,\sigma,\pi) = \begin{cases} \pi + (1-\pi)(1+\mu\sigma)^{-(1/\sigma)} \text{ ，对于 } y = 0 \\ (1-\pi)\dfrac{\Gamma[y+(1/\sigma)]}{\Gamma[y+1]\Gamma[1/\sigma]}\left[\dfrac{(\mu\sigma)^y}{(1+\mu\sigma)^{y+(1/\sigma)}}\right] \text{ ，对于 } y = 1, 2, \ldots \end{cases}$$

$$E(Y) = (1-\pi)\mu$$

$$Var(Y) = \mu(1-\pi)[1+\mu(\sigma+\pi)]$$

零泛滥 Gamma 分布

$$f(y|\mu,\sigma,\pi) = \begin{cases} \pi \text{ ，对于} y = 0 \\ (1-\pi)\dfrac{\exp(-y/\sigma)}{\Gamma[\mu/\sigma]\sigma^{\mu/\sigma}y^{1-\mu/\sigma}} \text{ ，对于 } y > 0 \end{cases}$$

$$E(Y) = \mu(1-\mu)$$

$$Var(Y) = \mu(1-\pi)(\sigma+\mu) - (1-\pi)^2\mu^2$$

第 7 章

广义回归示例

使用正则化方法构建模型

本章提供有关使用“广义回归”平台的若干模型拟合的带有说明性材料的示例。

目录

Poisson 广义回归的示例 295
二项广义回归的示例 297
零泛滥 Poisson 回归的示例 299
针对宽数据的广义回归示例 301

Poisson 广义回归的示例

Liver Cancer.jmp 样本数据表包含 136 名患者的肝癌**结节计数**值。它还包含以下 6 个有可能相关的变量的测量值：**BMI**、**年龄**、**时间**、**标记**、**肝炎**和**黄疸**。这些变量在数据表每一列中的“注释”列属性中说明。要查看“注释”列属性，请右击列名，选择“列信息”，然后在“列属性”下选择“注释”。

本例使用 6 个预测变量构建**结节计数**的预测模型。使用 Poisson 分布对**结节计数**建模。

1. 选择**帮助** > **样本数据库**，然后打开 Liver Cancer.jmp。
2. 选择**分析** > **拟合模型**。
3. 从“选择列”列表中选择**结节计数**，然后点击 **Y**。
4. 从 **BMI** 一直选到**黄疸**，然后点击**宏** > **析因次数**。

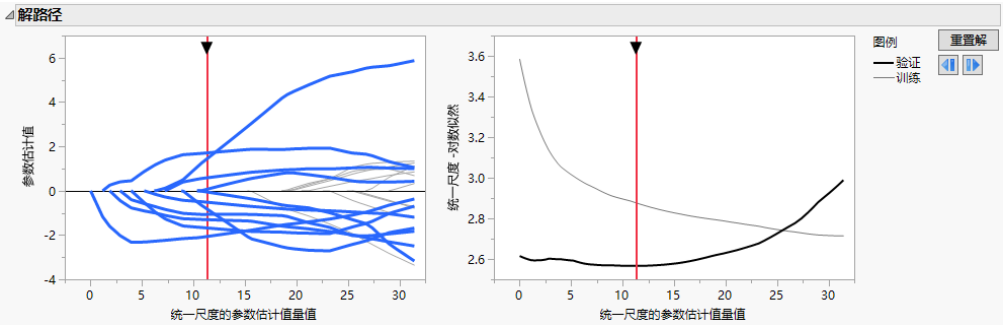
   这会将最高达到 2 次（**次数**框中的默认值）的所有项都添加到模型中。
5. 从“选择列”列表中选择**验证**，然后点击**验证**。
6. 从“特质”列表中选择**广义回归**。
7. 从“分布”列表中选择 **Poisson**。
8. 点击**运行**。

   显示的“广义回归”报表包含“模型比较”报表、“模型启动”控制面板和一个“使用‘验证列’验证的‘Poisson 最大似然’”报表。请注意，默认估计方法是“Lasso”。
9. 选择**自适应**框。
10. 点击**执行**。
11. 点击“使用‘验证列’验证的‘Poisson’自适应‘Lasso’”旁边的红色小三角，然后选择**选择非零项**。

    “解路径”如图 7.1 所示。突出显示具有非零系数的项的路径。想象一下解路径在图上从右向左移动，解在逐渐远离 MLE 的位置收缩。很多项具有很早就收缩到零的路径。

    “解路径”图中的垂直轴表示标准化的预测变量的参数估计值。红色的垂直线指示在最佳收缩处的值，该值由交叉验证决定。在此点上，11 个项具有非零系数。请注意，红色的垂直线指示“验证集”中最小的统一尺度 – 对数似然值。

图 7.1 突出显示了非零项的自适应 Lasso 拟合的解路径



“原始预测变量的参数估计值”报表（图 7.2）显示非中心化和未统一尺度的数据的参数估计值。突出显示具有非零参数估计值的 11 项。其中包括交互作用效应。在数据表中，选择全部 6 个预测变量列，因为每个预测变量列显示在具有非零系数的项中。

在“效应检验”报表中，具有零系数估计值的 10 个效应标明为“已删除”。“效应检验”报表指示在 0.05 水平下只有一个效应是显著的：**年龄 * 标记**交互作用。

图 7.2 突出显示非零项的参数估计值报表

原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	1.9658064	1.5268652	1.6576014	0.1979	-1.026794	4.9584071
BMI	-0.014557	0.0655906	0.0492589	0.8244	-0.143113	0.1139977
年龄	-0.000344	0.0052439	0.0043119	0.9476	-0.010622	0.0099335
时间	0	0	0	1.0000	0	0
标记[0-1]	0.3853241	0.3347727	1.3248061	0.2497	-0.270818	1.0414665
肝炎[0-1]	0	0	0	1.0000	0	0
黄疸[0-1]	-0.220764	0.1645347	1.8002793	0.1797	-0.543246	0.1017185
(BMI-23.1737)*(年龄-56.3994)	0.0007336	0.0014188	0.267356	0.6051	-0.002047	0.0035144
(BMI-23.1737)*(时间-9.00945)	0	0	0	1.0000	0	0
(BMI-23.1737)*标记[0-1]	-0.066574	0.0511526	1.6938495	0.1931	-0.166831	0.0336832
(BMI-23.1737)*肝炎[0-1]	0.0269878	0.0827496	0.1063662	0.7443	-0.135198	0.1891741
(BMI-23.1737)*黄疸[0-1]	0	0	0	1.0000	0	0
(年龄-56.3994)*(时间-9.00945)	0	0	0	1.0000	0	0
(年龄-56.3994)*标记[0-1]	-0.023498	0.0083023	8.0106599	0.0047*	-0.03977	-0.007226
(年龄-56.3994)*肝炎[0-1]	0	0	0	1.0000	0	0
(年龄-56.3994)*黄疸[0-1]	0	0	0	1.0000	0	0
(时间-9.00945)*标记[0-1]	-0.007529	0.0104633	0.5178189	0.4718	-0.028037	0.0129784
(时间-9.00945)*肝炎[0-1]	0	0	0	1.0000	0	0
(时间-9.00945)*黄疸[0-1]	0.0009619	0.0093476	0.0105895	0.9180	-0.017359	0.0192828
标记[0-1]*肝炎[0-1]	-0.422619	0.397399	1.1309549	0.2876	-1.201507	0.3562684
标记[0-1]*黄疸[0-1]	0	0	0	1.0000	0	0
肝炎[0-1]*黄疸[0-1]	0	0	0	1.0000	0	0

12. 在“原始预测变量的参数估计值”报表中，点击 (**年龄** - 56.3994)* **标记** [0-1] 所对应的行。

    该操作将在“解路径”图中突出显示该效应的路径并在数据表中选择**年龄**和**标记**两列。

13. 点击“使用‘验证列’验证的‘Poisson’自适应‘Lasso’”旁边的红色小三角，然后选择**保存列 > 保存预测公式**和**保存列 > 保存方差公式**。

    以下两列添加到数据表：**节点计数预测公式**和**节点计数方差**。

14. 在数据表中，右击列标题并选择**公式**以查看公式。或者，点击列面板中列名称右侧的加号。

保存预测公式列中的预测公式将指数函数应用到模型的估计线性部分。使用相同公式给出“**结节计数**”**方差**中的预测方差公式，因为 Poisson 分布的方差等于其均值。

二项广义回归的示例

本例说明如何在 Liver Cancer.jmp 样本数据表中构建二项响应**严重性**的预测模型。

1. 选择**帮助** > **样本数据库**，然后打开 Liver Cancer.jmp。
2. 选择**分析** > **拟合模型**。
3. 从“选择列”列表中选择**严重性**，然后点击 Y。
4. 从 BMI 一直选到**黄疸**，然后点击**宏** > **析因次数**。

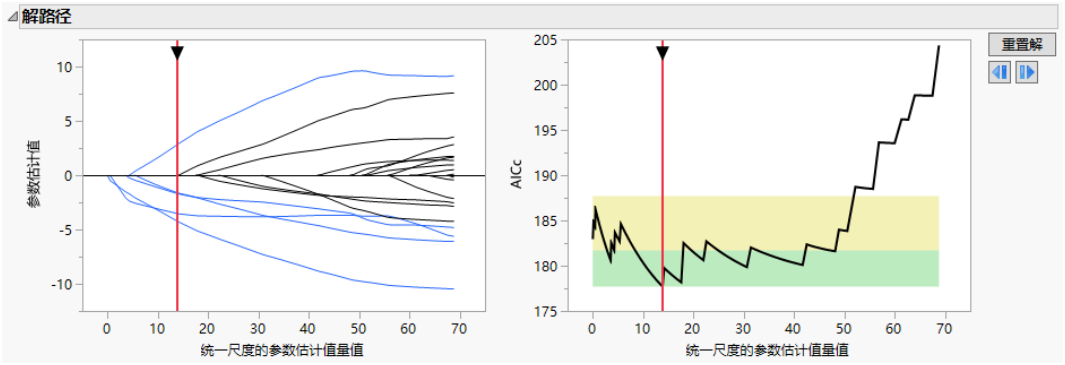
   这会将最高达到 2 次（**次数**框中的默认值）的所有项都添加到模型中。
5. 从“特质”列表中选择**广义回归**。

   “分布”列表自动显示“二项”分布。这是在 Y 为二值响应并且建模类型为“名义型”时唯一可用的分布。
6. 点击**运行**。

   显示的“广义回归”报表包含“模型比较”报表、“模型启动”控制面板和“Logistic 回归”报表。请注意，默认估计方法是“Lasso”。
7. 选择**弹性网络**作为“估计方法”。
8. 选择**自适应**框。
9. 点击**执行**。

   随即显示“使用‘AICc 验证’的‘二项’自适应‘弹性网络’”报表。“解路径”如图 7.3 所示。

图 7.3 解路径图



具有非零系数的项的路径用蓝色显示。最佳参数值大幅收缩，远小于 MLE。右侧的验证图指示有几个模型可以认定为与最佳模型一样好。要查看这些模型，请在黑线位于绿色区域中的区域周围滑动红色的垂直条。

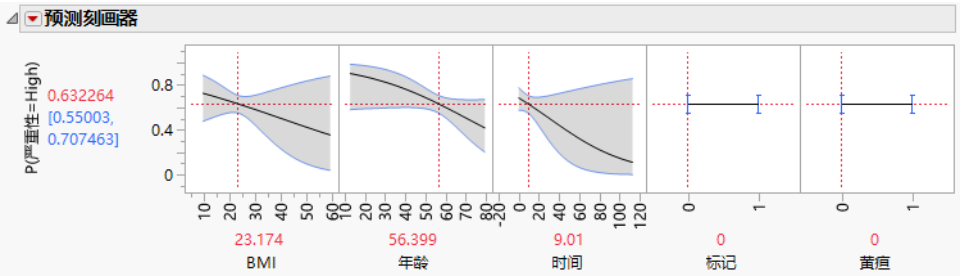
10. 点击“使用‘AICc 验证’的‘二项’自适应‘弹性网络’”旁边的红色小三角，然后选择**选择清零项**选项。

    系数估计值为零的 16 个项在“原始预测变量的参数估计值”报表中突出显示。“效应检验”报表将这些项标明为“已删除”。

    “效应检验”报表还显示在 0.05 水平下没有显著的项。但是，**时间 * 标记**交互作用具有很小的 p 值 (0.0626)，**时间**效应具有很小的 p 值 (0.1458)。

11. 点击“使用‘AICc 验证’的‘二项’自适应‘弹性网络’”旁边的红色小三角，然后选择**刻画器 > 刻画器**。

图 7.4 时间短，严重性 = High 的概率对应的刻画器

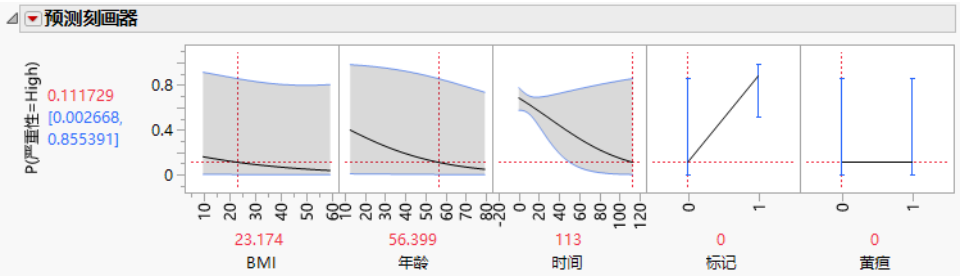


检查“预测刻画器”以查看**时间**和**时间 * 标记**交互作用如何影响**严重性**。

注意：预测变量**肝炎**不显示在该刻画器中，因为它未显示在任何活跃（非零）项中。由于**标记**和**黄疸**均显示在活跃的交互作用项中，所以它们会显示在该刻画器中，尽管作为主效应它们并不活跃。

12. 从左向右移动**时间**的红色虚线以查看它与**标记**的交互作用（图 7.4 和图 7.5）。对于具有小的确诊**时间**值的进入研究的患者，**标记**对**严重性**的影响很小。但是对于确诊时间更长的进入研究的患者，**标记**很重要。对于那些患者，常规标记指示高**严重性**的概率更低。

图 7.5 时间长，严重性 = High 的概率对应的刻画器



零泛滥 Poisson 回归的示例

Fishing.jmp 样本数据表包含虚拟数据，用于研究哪些因素会影响成组游客在公园里钓到的鱼的数目。该数据表包含结伴而来的家庭或团体的 250 个响应。本例将**钓鱼数**建模为**活鱼饵**、**鱼竿**、**露营者**、**人**和**儿童**的函数。这些变量在数据表每一列中的“注释”列属性中说明。要查看“注释”列属性，请右击列名，选择“列信息”，然后在“列属性”下选择“注释”。

该数据表包含名为**钓过鱼**的隐藏列。在数据收集期间，它从不确定组中的任何人是否实际钓鱼了。但是，将**钓过鱼**列包括在表中以强调在以下两种情况下钓鱼数可以为零：组中没有一个人在钓鱼或组中钓鱼的人运气都很不好。

因此，零响应可能有两个来源。要解决该问题，您可以拟合零泛滥分布。因为 Poisson 分布适用于来自钓鱼的人的计数数据，您拟合零泛滥 Poisson 分布。

1. 选择**帮助** > **样本数据库**，然后打开 Fishing.jmp。
2. 选择**分析** > **拟合模型**。
3. 从“选择列”列表中选择**钓鱼数**，然后点击 Y。
4. 从**活鱼饵**一直选到**儿童**，然后点击**宏** > **析因次数**。

   这会将最高达到 2 次（**次数**框中的默认值）的所有项都添加到模型中。
5. 从“选择列”列表中选择验证，然后点击**验证**。
6. 从“特质”列表中选择**广义回归**。
7. 从“分布”列表中选择**零泛滥 Poisson**。
8. 点击**运行**。

   显示的“广义回归”报表包含“模型比较”报表、“模型启动”控制面板和一个“使用‘验证列’验证的‘ZI Poisson 最大似然’”报表。请注意，默认估计方法是“Lasso”。
9. 从“估计方法”列表中，选择**弹性网络**。
10. 点击**执行**。

    随即显示“使用‘验证列’验证的‘ZI Poisson 弹性网络’”报表。“解路径”、“原始预测变量的参数估计值”报表和“效应检验”报表都指示有一项已清零。零泛滥参数（其估计值显示在两个参数估计值报表的最后一行）高度显著。这指示响应**钓鱼数**中的一些变异可能是一些组实际不钓鱼造成的。

图 7.6 “原始预测变量的参数估计值”报表

原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	1.3599776	0.3237335	17.647701	<.0001*	0.7254716	1.9944837
活鱼饵[0-1]	-0.379555	0.1919937	3.9081951	0.0481*	-0.755856	-0.003255
鱼竿	0.3034685	0.0465239	42.547553	<.0001*	0.2122833	0.3946537
露营者[0-1]	-0.363094	0.1899392	3.654349	0.0559	-0.735368	0.0091795
人	0.0776017	0.121377	0.4087617	0.5226	-0.160293	0.3154962
儿童	-0.192656	0.1028943	3.5057658	0.0612	-0.394325	0.009013
活鱼饵[0-1]*(鱼竿-0.856)	-0.026766	0.0668764	0.160184	0.6890	-0.157841	0.1043095
活鱼饵[0-1]*露营者[0-1]	-0.336382	0.48759	0.4759452	0.4903	-1.292041	0.6192764
活鱼饵[0-1]*(人-2.516)	-0.164164	0.1994702	0.6773258	0.4105	-0.555118	0.2267908
活鱼饵[0-1]*(儿童-0.936)	0.1025463	0.2362161	0.1884607	0.6642	-0.360429	0.5655213
(鱼竿-0.856)*露营者[0-1]	-0.446199	0.1120088	15.869111	<.0001*	-0.665732	-0.226666
(鱼竿-0.856)*(人-2.516)	-0.00611	0.0853344	0.0051267	0.9429	-0.173362	0.1611423
(鱼竿-0.856)*(儿童-0.936)	-0.17587	0.080167	4.8127458	0.0282*	-0.332995	-0.018746
露营者[0-1]*(人-2.516)	-0.195261	0.3035369	0.4138148	0.5200	-0.790182	0.3996608
露营者[0-1]*(儿童-0.936)	0.1899805	0.2998548	0.4014171	0.5264	-0.397724	0.7776852
(人-2.516)*(儿童-0.936)	0	0	0	1.0000	0	0

“零泛滥 Poisson”分布参数

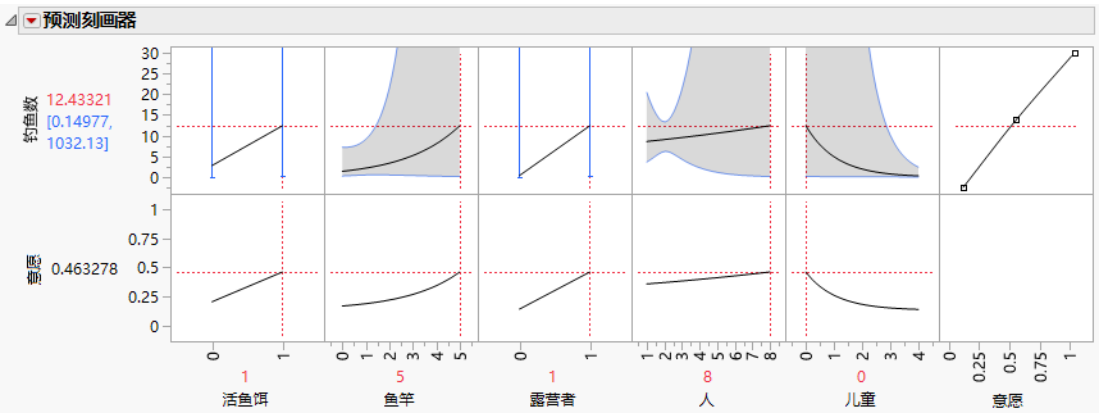
估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
零泛滥 0.7815222	0.0354433	486.19905	<.0001*	0.7120545	0.8509898

“效应检验”报表指示以下四项在 0.05 水平下显著：**活鱼饵**、**鱼竿**、**鱼竿 * 露营者**和**鱼竿 * 儿童**。

11. 点击“使用‘验证列’验证的‘ZI Poisson 弹性网络’”旁边的红色小三角，然后选择**刻画器 > 刻画器**。
12. 点击“预测刻画器”红色小三角，然后选择**优化和意愿 > 意愿函数**。

    响应应用了一个函数，它指示最大化**钓鱼数**是您想要的。有关意愿函数的详细信息，请参见《刻画器指南》。
13. 点击“预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。

图 7.7 最大化钓鱼数的“预测刻画器”



您可以更改预测变量的设置来查看以下显著效应的影响：**活鱼饵**、**鱼竿**、**鱼竿 * 露营者**和**鱼竿 * 儿童**。例如，**活鱼饵**与钓到更多的鱼关联；**露营者**与不露营的人相比带来的鱼竿更多，因此钓的鱼更多。

14. 点击“使用‘验证列’验证的‘ZI Poisson 弹性网络’”旁边的红色小三角，然后选择**保存列 > 保存预测公式**和**保存列 > 保存方差公式**。

    以下两列添加到数据表：**“钓鱼数”预测公式**和**“钓鱼数”方差**。

15. 在数据表中，右击列标题并选择**公式**以查看公式。或者，点击列面板中列名称右侧的加号。请注意，估计的零泛滥参数 0.781522155 在这两个公式中都出现了。

针对宽数据的广义回归示例

当预测因子多于观测值时，传统的回归方法是不实用的。包含变量选择的回归方法支持您在这些情况下拟合回归模型。在本例中，您可以比较三个变量选择程度各不相同的模型。这三个模型都位于“验证图”上的绿色区域内，因此有强有力的证据表明，其中任何一个模型都与最佳模型一样好。

Prostate Cancer.jmp 样本数据表包含从 165 名男性中采集的血清样本的结果，其中大约一半患有前列腺癌。血清样本中测量了 667 种蛋白质。

1. 选择**帮助 > 样本数据库**，然后打开 Prostate Cancer.jmp。
2. 选择**分析 > 拟合模型**。
3. 从“选择列”列表中选择**状态**，然后点击 Y。

   由于这是一个“名义型”响应列，“特质”更改为“名义型 Logistic”并且显示“目标水平”选项。该选项的默认值为 CCD，因为这是在数据表的“目标水平”列属性中指定的值。

4. 从“特质”列表中选择**广义回归**。

   “分布”列表自动显示“二项”分布。这是在 Y 为二值响应并且建模类型为“名义型”时唯一可用的分布。

5. 从“选择列”列表中选择**蛋白质**列组并点击**添加**。

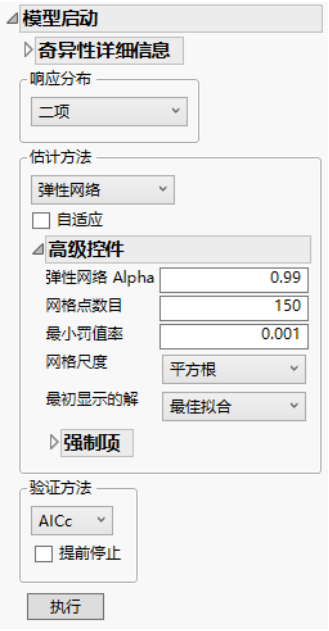
   这会将列组中的所有 667 列添加到模型中。

6. 点击**运行**。

   显示的“广义回归”报表包含“模型启动”控制面板。由于预测变量数大于观测数，因此没有初始“Logistic 回归”模型拟合。

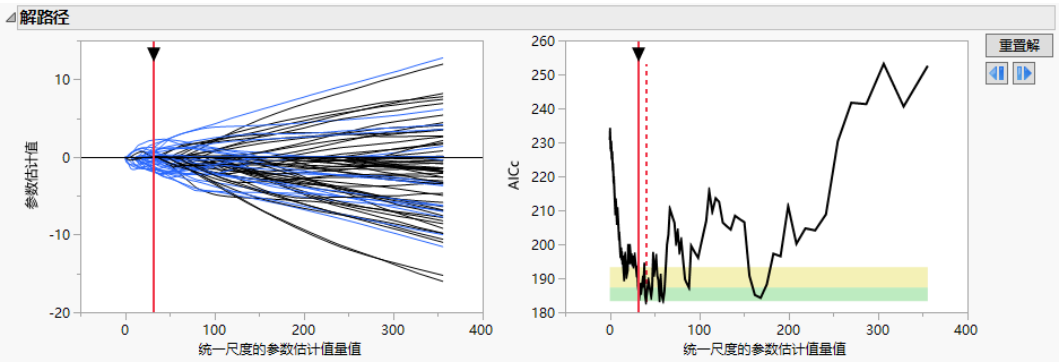
7. 选择**弹性网络**作为“估计方法”。
8. 点击“高级控件”旁边的灰色展开图标。

图 7.8 高级控件



9. 选择**绿色区域中最小**作为“最初显示的解”。
10. 点击**执行**。

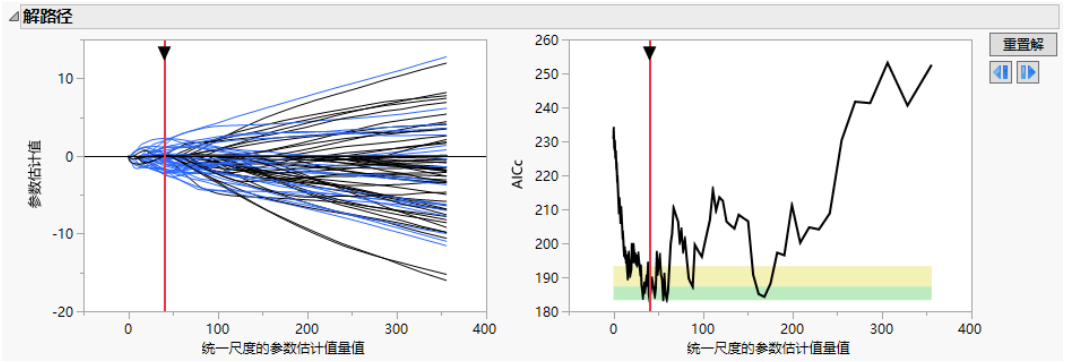
图 7.9 “绿色区域中最小”模型



“解路径”显示被视为与最小 AICc 模型相当的最小模型，其中最小模型是指参数最少的模型。

11. 点击“使用‘AICc 验证’的‘二项弹性网络’”旁边的灰色展开图标。
12. 点击“模型启动”旁边的灰色展开图标。
13. 选择**最佳拟合**作为“最初显示的解”。
14. 点击**执行**。

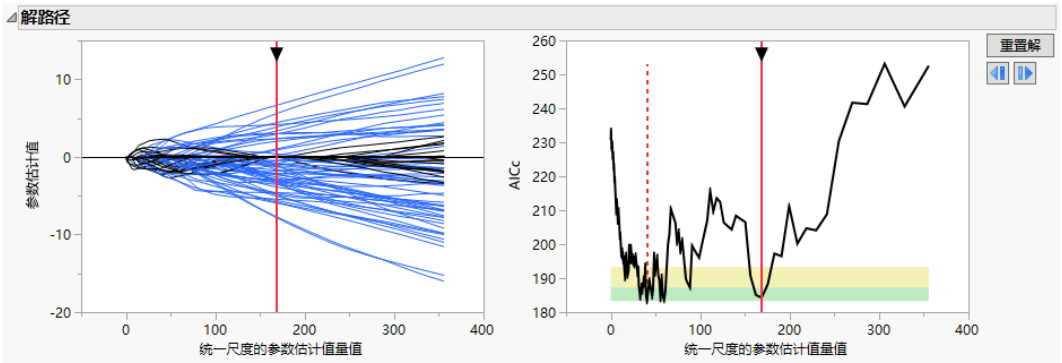
图 7.10 “最佳拟合”模型



“解路径”显示最佳拟合模型，其中最佳拟合表示具有最小 AICc 值的拟合。

15. 点击“使用 ‘AICc 验证’ 的 ‘二项弹性网络’”旁边的灰色展开图标。
16. 点击“模型启动”旁边的灰色展开图标。
17. 选择**绿色区域中最大**作为“最初显示的解”。
18. 点击**执行**。

图 7.11 “绿色区域中最大”模型



“解路径”显示被视为与最小 AICc 模型相当的最大模型，其中最大模型是指参数最多的模型。

19. 点击“使用 ‘AICc 验证’ 的 ‘二项弹性网络’”旁边的灰色展开图标。

图 7.12 “模型比较”表

模型比较

显示	响应分布	估计方法	验证方法	非零参数	AICc	BIC	广义 R 方
☑	二项	弹性网络	AICc	36	186.07102	277.07256	0.7466199
☑	二项	弹性网络	AICc	40	183.29544	281.08164	0.8021925
☑	二项	弹性网络	AICc	57	184.25399	299.49849	0.9824565

“模型比较”报表显示三个模型。您可以使用“非零参数”列标识每个模型的大小。随着模型中参数个数的增加，“广义 R 方”值增加。由于这些模型全部位于绿色区域，因此有强有力的证据表明，其中任何一个模型都与最佳模型一样好。

第 8 章

JMP PRO 混合模型

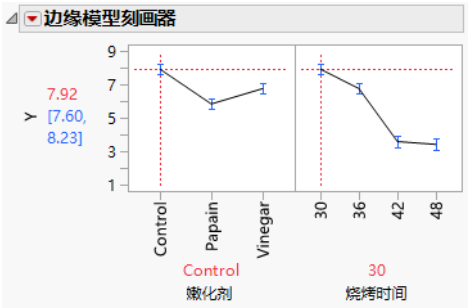
对均值和协方差进行联合建模

“拟合模型”平台的 “混合模型”特质仅在 JMP Pro 中可用。

在 JMP Pro 中，“拟合模型”平台的“混合模型”特质为具有复杂协方差结构的连续响应拟合各种线性模型。这些模型包含随机系数、重复测量、空间数据以及具有多个相关响应的数据。使用“混合模型”特质通过直观的界面来方便地指定线性混合模型及其协方差结构，并使用最大似然方法来拟合这些模型。

通过交互式可视化工具（如刻画器、曲面图和等高线图）支持建模结果。您可以使用这些工具来补充您对模型的理解。

图 8.1 裂区实验的边缘模型刻画器



目录

“混合模型”特质概述 308
使用混合模型的示例 308
启动“混合模型”特质 313
“拟合模型”启动窗口 313
数据格式 318
“拟合混合模型”报表 318
拟合统计量 320
随机效应协方差参数估计值 321
固定效应参数估计值 322
重复效应协方差参数估计值 323
随机系数 323
随机效应预测 323
固定效应检验 324
序贯检验 324
多重比较 325
比较斜率 325
边缘模型推断 325
“预测值 - 实际值”图 326
残差图 326
刻画器 326
条件模型推断 327
“条件预测值 - 实际值”图 327
条件残差图 327
条件刻画器 328
变差图 328
协方差和相关性矩阵 329
保存列 329
“混合模型”特质的其他示例 330
重复测量示例 330
裂区示例 345
空间示例：均匀性试验 350
相关响应示例 358
“混合模型”特质的统计详细信息 363
收敛得分检验 364
随机系数模型 365
重复测量 366
重复的协方差结构 366

空间和时间变异性 371
Kackar-Harville 校正 374

JMP PRO “混合模型”特质概述

在 JMP Pro 中，“混合模型”特质允许您分析具有复杂协方差结构的模型。可以分析的情形包括：

- 裂区实验
- 随机系数模型
- 重复测量设计
- 空间数据
- 相关响应数据

裂区实验是包含实验单元的两个或更多水平（或大小）而导致多个误差项的实验。当一些因子易于改变而另一些因子难以改变时，通常需要进行这样的设计。请参见《实验设计指南》。

随机系数模型也称为层次或多水平模型 (Singer 1998；Sullivan et al. 1999)。当认为批次或对象的截距和斜率随机变化时，使用这些模型。制药行业的药物稳定性试验和教育研究领域的个人成长研究通常需要随机系数模型。

重复测量设计、空间数据和相关响应数据都具有观测值不独立的特点，要求您对相关性结构进行建模。

- 重复测量设计也称为对象内设计，它对响应在时间或空间上的变化建模并允许误差有相关性。
- 空间数据是在二维或多维中进行的测量，通常为纬度和经度。空间测量值的相关性通常表示为其空间邻近度的函数。
- 相关响应数据来自对同一实验单元的几次测量。例如，医学研究中对个体所测的身高、体重和血压读数可能是相关的，制造业中对产品所测的硬度、强度和弹性值可能是相关的。尽管可以单独研究这些测量值，将它们作为相关响应处理可以得到有用的信息。

未考虑观测值之间的相关性可能导致得到有关处理效应的不正确结论。但是，估计协方差结构参数需要使用数据中的信息。要估计的参数数目影响功效和第一类错误率。因此，您必须正确选择协方差模型。请参见第 330 页的“重复测量示例”。

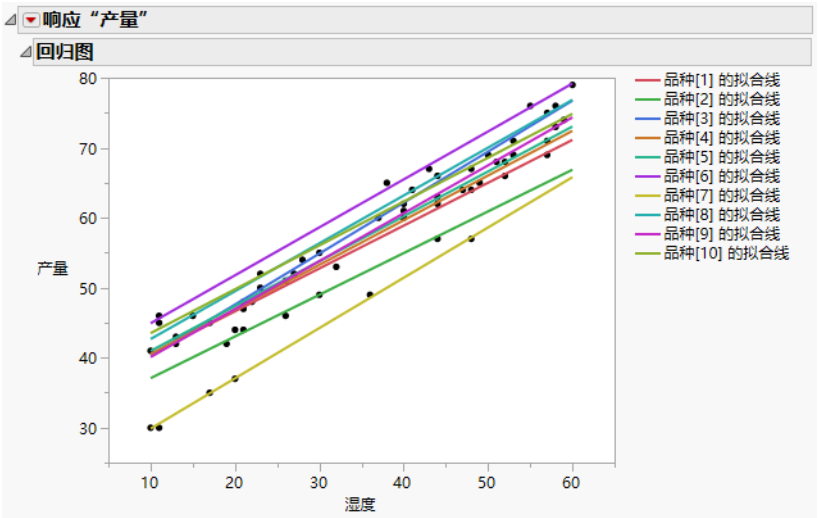
JMP PRO 使用混合模型的示例

在小麦产量的研究中，从适应干燥气候条件的硬红冬麦品种的总体中随机选择 10 个小麦品种。将这些品种随机分配给 6 个一英亩地块。种植前地块的含水量可能影响发芽率，从而影响地块的最终产量。因此，为每个地块确定土壤表层 36 英寸内的种植前水分含量。您想知道含水量是否影响产量。

因为是随机选择品种，每个品种的回归模型是从品种模型的总体中选择的随机模型。截距和斜率对于每个品种是随机的，并且可能相关。随机系数以固定效应为中心。固定效应是总体截距和斜率，它们是品种的截距和斜率总体的期望值。本例摘自 Littell et al. (2006, p. 320)。

在“标准最小二乘法”特质中使用 REML 拟合模型，这使您可以查看截距和斜率中的变异（图 8.2）。请注意斜率没有很大的变异性，但是截距有很大的变异性。截距和斜率可能是负相关的，具有较小截距的品种似乎有更大的斜率。

图 8.2 标准最小二乘法回归



要对截距和斜率之间的相关性建模，可以使用 “混合模型”特质。您想确定总体回归方程以及特定品种的方程。

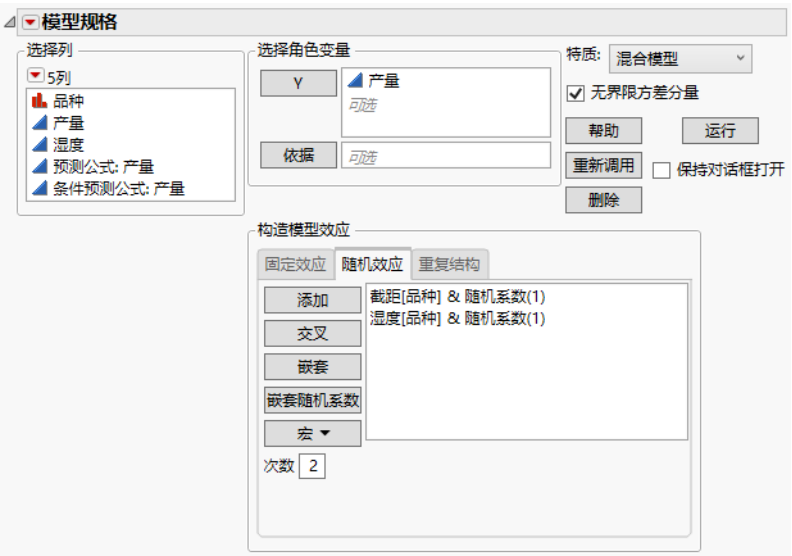
1. 选择**帮助 > 样本数据库**，然后打开 Wheat.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**产量**并点击 Y。

   当您将该列添加为 Y 后，拟合特质变为 “标准最小二乘法”。
4. 从 “特质”列表中选择**混合模型**。或者，您可以先选择 “混合模型”特质，然后点击 Y 以添加**产量**。
5. 选择**湿度**，然后点击 “固定效应”选项卡上的**添加**。

图 8.3 显示 “固定效应” 的完成的 “拟合模型” 启动窗口

6. 选择**随机效应**选项卡。
7. 选择**湿度**，然后点击**添加**。
8. 从 “选择列” 列表中选择**品种**，从 “随机效应” 选项卡中选择**湿度**，然后点击**嵌套随机系数**。

图 8.4 显示 “随机效应” 选项卡的完成的 “拟合模型” 启动窗口



随机效应按品种分组，并且截距作为随机成分包括在内。

9. （可选。）点击“模型规格”红色小三角并选中**中心多项式**选项的设置。

   即使选择了“中心多项式”选项，**湿度**效应也不会以其均值为中心，因为它涉及到随机效应。

10. 点击**运行**。

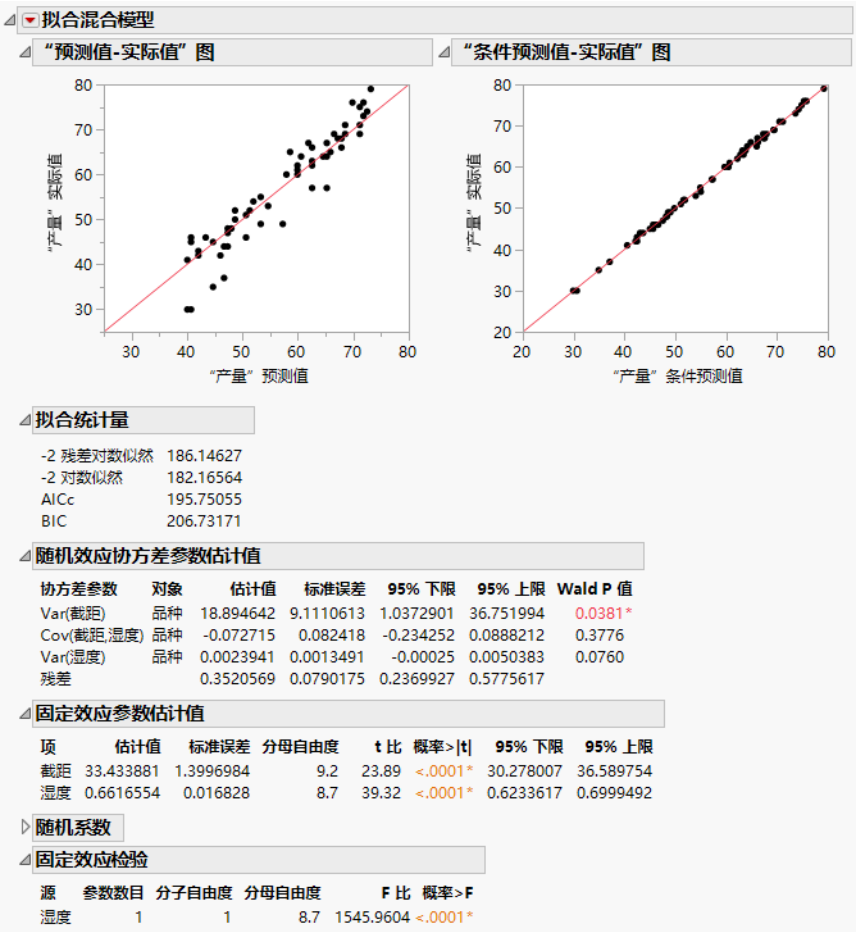
    “拟合混合模型”报表显示在图 8.5 中。“预测值 - 实际值”图显示模型拟合和所基于的假设之间没有不一致之处。

    因为模型拟合没有明显的问题，您现在可以解释统计检验并获得回归方程。湿度对产量的影响是显著的，如“固定效应检验”报表中所示。“固定效应参数估计值”中给出的估计值指示以下方程是估计的总体回归方程：

    产量 = 33.43 + 0.66 * 湿度

    “随机效应协方差参数估计值”报表给出品种的截距方差估计值“Var(截距)”、品种的斜率 (即湿度) 方差估计值“Var(湿度)”以及它们的协方差估计值“Cov(湿度 , 截距)”。在该示例中，截距和斜率不显著相关，因为估计值的置信区间包括零。该报表还给出残差方差的估计值。

图 8.5 “拟合混合模型”报表



尽管您有总体回归方程的估计值，还想知道品种 2 的估计产量。

11. 打开 “随机系数”报表以查看 “截距”和 “湿度”的品种效应估计值。这些系数估计每个品种与总体的差异有多大。

图 8.6 “随机系数”报表

随机系数

品种

品种	截距	湿度
1	0.9577924	-0.049211
2	-2.284289	-0.066697
3	-0.408109	0.0672225
4	0.6960205	-0.023306
5	1.115904	-0.019904
6	4.639151	0.0238887
7	-10.73004	0.0564233
8	2.4011709	0.0224336
9	-0.176207	0.0233566
10	3.7886051	-0.034207

协方差矩阵

随机效应	截距	湿度
截距	18.89464	-0.07272
湿度	-0.07272	0.002394

从“固定效应参数估计值”和“随机系数”报表，您得到**品种** 2 的以下预测方程：

产量 = 33.433 + 0.662 * 湿度 – 2.284 – 0.067 * 湿度

产量 = 31.149 + 0.595 * 湿度

品种 2 开始的产量低于总体平均值，且随**湿度**增加的速率慢于总体平均值。

JMP PRO 启动“混合模型”特质

“混合模型”是您可以在“拟合模型”启动窗口中选择的几个特质之一。本节说明如何在“拟合模型”启动窗口中选择“混合模型”作为拟合方法，同时还涵盖了特定于该选择的选项。

JMP PRO “拟合模型”启动窗口

您可以指定具有固定效应、随机效应、重复结构或这三项组合的模型。选项根据您所指定的模型的性质而有所不同。

要使用“混合模型”特质拟合模型，选择**分析 > 拟合模型**，然后从“特质”列表中选择**混合模型**。请注意，当您在选择特质**前**在“Y”列表中输入一个连续变量时，“特质”默认为“标准最小二乘法”。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

使用“混合模型”特质拟合模型时，您可以允许无界限方差分量。这意味着具有负估计值的方差分量不被报告为零。默认选定该选项。若您关注的是固定效应，则它应保持选定状态，因为将方差估计值的边界定为零将导致固定效应检验发生偏倚。有关“无界限方差分量”选项的详细信息，请参见“标准最小二乘法模型”一章中第 165 页的“负方差”。

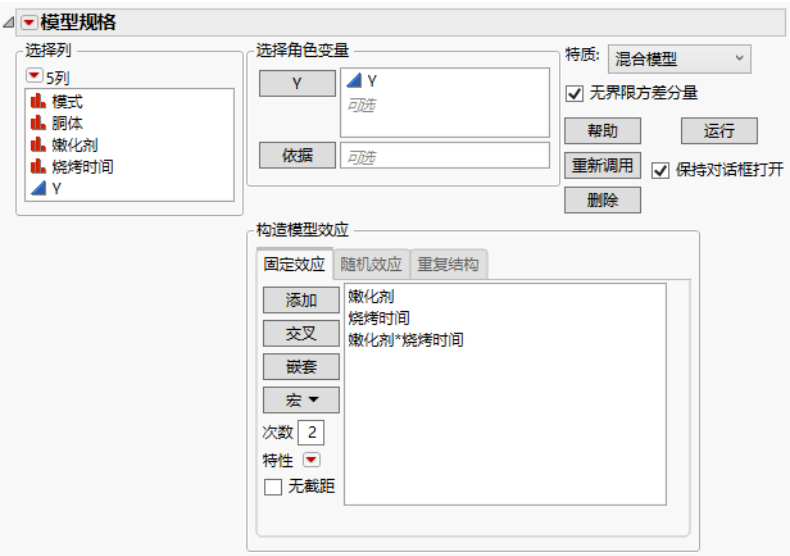
JMP PRO “固定效应”选项卡

在“固定效应”选项卡上添加所有固定效应。根据需要使用“添加”、“交叉”、“嵌套”、“宏”和“特性”选项。有关这些选项的详细信息，请参见第 29 页的“模型规格”一章。

注意：若连续列涉及随机效应，则该列不会位于中心，即使选择了“模型规格”红色小三角菜单中的“中心多项式”选项也是如此。

Split Plot.jmp 样本数据表分析的固定效应显示在图 8.7 中。请注意，模型中可能没有固定效应。相关示例，请参见第 350 页的“空间示例：均匀性试验”。

图 8.7 显示完成的 “固定效应”的 “拟合模型”启动窗口



JMP PRO “随机效应”选项卡

使用“随机效应”选项卡指定传统方差分量模型和随机系数模型。

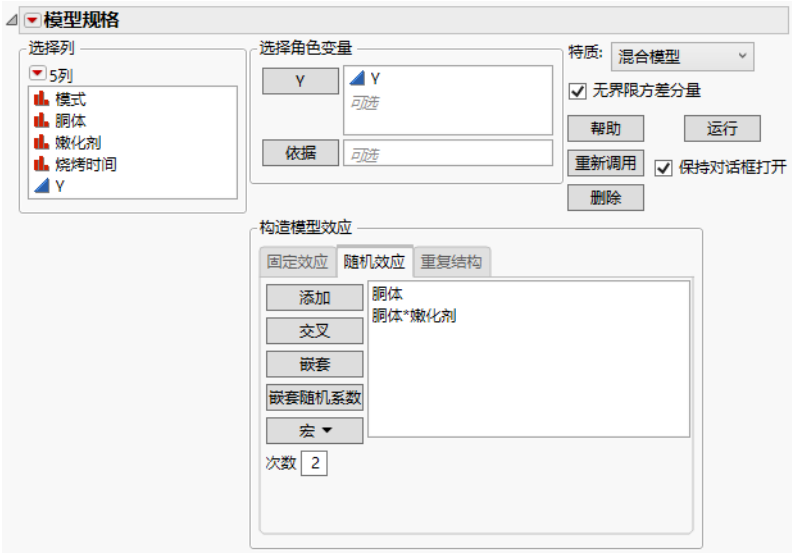
注意：若连续列涉及随机效应，则该列不会位于中心，即使选择了“模型规格”红色小三角菜单中的“中心多项式”选项也是如此。

方差分量

对于传统方差分量模型，使用“添加”、“交叉”或“嵌套”选项指定如随机区组、整区误差项和子区误差项之类的项。有关这些选项的详细信息，请参见第 29 页的“模型规格”一章。

图 8.8 显示 Split Plot.jmp 样本数据的随机效应规格，其中“胴体”为随机区组。第 345 页的“裂区示例”详细描述了该示例。

图 8.8 显示完成的“随机效应”选项卡的“拟合模型”启动窗口



随机系数

要构造随机系数模型，使用“嵌套随机系数”按钮来创建随机系数组。

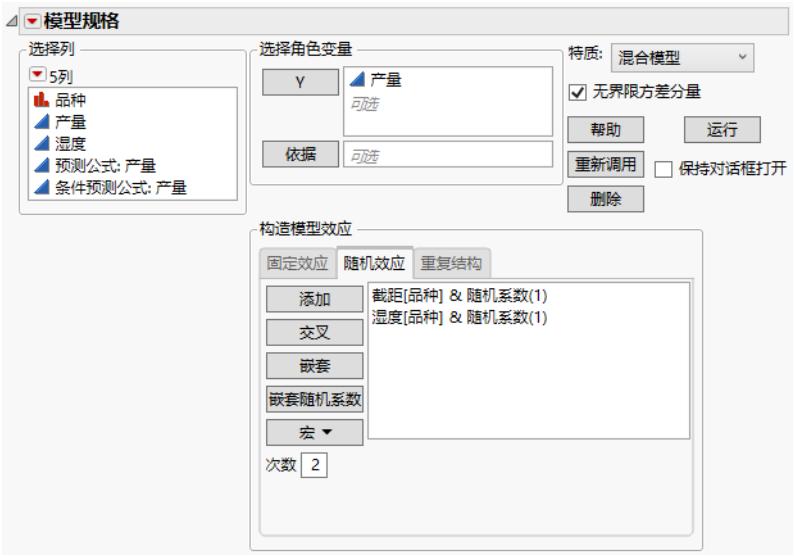
1. 从“选择列”列表中选择作为预测变量的连续列。
2. 选择**随机效应**选项卡，然后点击**添加**。
3. 在“随机效应”选项卡中选择这些效应。还选择包含随机效应的列，该效应的水平定义各回归模型。该列本质上是 SAS PROC MIXED 的随机语句中的对象。
4. 点击**嵌套随机系数**按钮。

最后一步创建在随机效应的各个水平内有相关性的随机截距和随机斜率效应。由于对象间的变异性，对象嵌套在其他效应内。若您相信截距可能对所有组是固定的，可以选择**截距[<组>]&随机系数(1)**，然后点击**删除**。

您可以按该方式定义多个随机系数组，如在层次线性模型中那样。当斜率和截距系数具有随机批次效应和处理*随机批次效应时，这可能是必要的。在以下层次线性模型中这也可能是必要的：学习成绩具有随机学生效应和随机学校效应，并且学生嵌套在学校内。

使用非结构化协方差结构对随机系数建模。图 8.9 显示 Wheat.jmp 样本数据的随机系数规格。（另见第 308 页的“使用混合模型的示例”。）

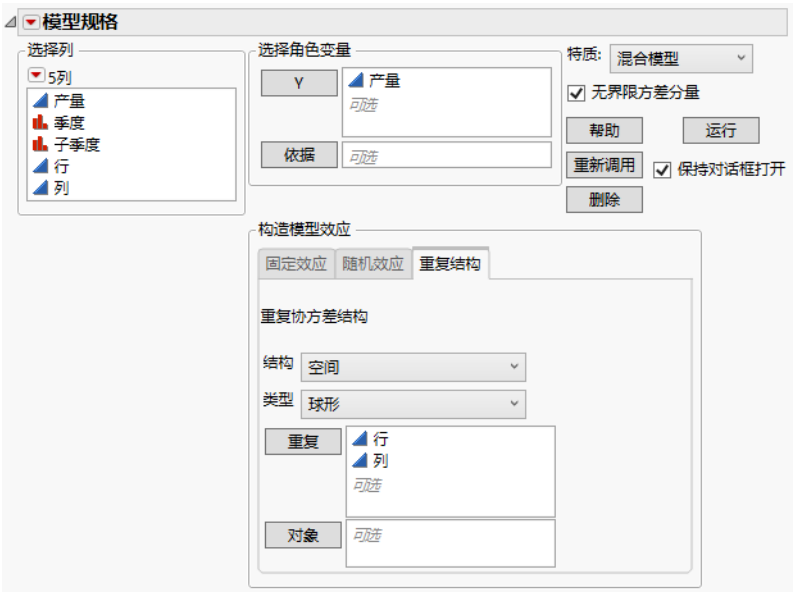
图 8.9 显示随机系数的完成的 “拟合模型”启动窗口



JMP PRO “重复结构”选项卡

使用 “重复结构”选项卡来选择模型中重复效应的协方差结构。

图 8.10 显示 “重复结构”选项卡的完成的 “拟合模型”启动窗口



结构

重复结构默认设置为“残差”。“残差”结构指定观测值之间没有协方差，即误差是独立的。除了“残差”和“不等方差”结构，所有其他协方差结构都对观测之间的协方差建模。有关这些结构的详细信息，请参见“统计详细信息”一节中的第 366 页的“重复测量”和第 371 页的“空间和时间变异性”。

表 8.1 列出了可用的协方差结构、使用每个结构的要求和给定结构的协方差参数个数。观测次数用 *J* 表示。

表 8.1 重复协方差结构要求

结构	重复列类型	所需的重复列数目	对象	参数个数
残差	不适用	0	不适用	0
不等方差	分类	1	可选	*J*
非结构化	分类	1	必需	*J*(*J*+1)/2
AR(1)	连续	1	可选	2
复合对称性	分类	1	必需	2
事前相关相等方差	分类		必需	*J*
Toeplitz	分类	1	必需	*J*
复合对称性不等方差	分类	1	必需	*J*+1
事前相关	分类		必需	2*J*-1
Toeplitz 不等方差	分类	1	必需	2*J*-1
空间	连续	2+	可选	
空间各向异性	连续	2+	可选	
带 Nugget 的空间	连续	2+	可选	
带 Nugget 的空间各向异性	连续	2+	可选	

若您输入具有残差结构的重复列或对象列，将忽略这些列。显示以下警报：“当选择残差协方差结构时忽略重复列和对象列。”

类型

当您选择一个空间协方差结构时，将显示一个“类型”列表，您可以在其中选择空间结构的类型。提供四种类型：“乘方”、“指数”、“高斯”和“球形”。图 8.10 显示为 Uniformity Trial.jmp 样本数据选择的空间球形。

重复

输入定义重复测量结构的列。“重复”列的建模类型取决于协方差结构。有关每个重复测量协方差结构的要求的详细信息，请参见表 8.1。

对象

输入定义对象的一个或多个列。“对象”列必须是分类的。

JMP PRO 数据格式

“拟合模型”平台中的“混合模型”特质要求所有响应测量值都包含在一个响应列中。重复测量数据有时记录在多个列中，其中，每行都是一个对象，重复的测量值记录在单独的响应列中。必须在运行“混合模型”特质之前堆叠该格式的数据。Cholesterol.jmp 和 Cholesterol Stacked.jmp 样本数据表分别演示了宽格式和堆叠格式。请注意，宽表中的每行都对应于堆叠表中**患者**的一个水平。

JMP PRO “拟合混合模型”报表

“拟合混合模型”红色小三角菜单包含以下选项：

模型报表 生成与混合模型拟合有关的报表。这些报表提供模型参数的估计值和检验，还列出拟合统计量。请参见第 319 页的 “模型报表”。

方差分量的线性组合 显示一个报表，该报表支持您计算方差分量线性组合的置信区间。最初，报表包含一个可编辑的文本框和模型中的方差分量表。使用文本框可为线性组合添加标签。在表格右列的单元格中输入值，以指定置信区间的线性函数。指定参数的线性组合并点击“完成”后，将显示一个表，其中包含指定线性组合的置信区间。

该表包含估计和标准误差，以及两种类型的置信区间 （Satterthwaite 和 Wald）和 Wald *p* 值。 Wald *p* 值对应估计值不同于零的假设检验。

提示：Satterthwaite 置信区间仅限于正值，因此不建议用于指定系数为负值的情况。若估计值为负，则无法构造 Satterthwaite 置信区间并报告为缺失。

多重比较 打开“多重比较”对话框窗口，您可以在其中选择一个或多个效应以及初始比较。该报表适用于分类固定效应。请参见第 325 页的 “多重比较”。

比较斜率 （仅当有一个名义型项、一个连续型项以及它们针对固定效应的交互作用效应时才可用。）生成一个报表，该报表支持您比较协方差分析 (ANCOVA) 模型中交互作用效应的每个水平的斜率。请参见第 325 页的 “比较斜率”。

逆预测 对于一个或多个响应值，预测解释变量的值。请参见 “标准最小二乘法模型”一章中第 125 页的 “逆预测”。

边缘模型推断 生成基于边缘预测值和边缘残差的图。这些图显示随机效应导致的变异。请参见第 325 页的 “边缘模型推断”。

条件模型推断 生成基于条件预测值和条件残差的图。这些图显示在剔除随机效应所解释的变异后剩下的变异。请参见第 327 页的 “条件模型推断”。

保存列 包含用于将各种模型结果保存为数据表中的列的选项。请参见第 329 页的“保存列”。

模型对话框 为当前分析打开完成的“拟合模型”启动窗口。请参见第 313 页的 ““拟合模型”启动窗口”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

模型报表

“模型报表”下提供的报表由您执行的分析类型决定。默认情况下显示其中的几个报表。

拟合统计量 显示模型拟合统计量的报表。请参见第 320 页的 “拟合统计量”。

随机效应协方差参数估计值 显示随机效应协方差参数估计值的报表。当您在启动窗口中指定随机效应时显示该报表。请参见第 321 页的 “随机效应协方差参数估计值”。

固定效应参数估计值 显示固定效应参数估计值的报表。请参见第 322 页的 “固定效应参数估计值”。

重复效应协方差参数估计值 显示重复效应协方差参数估计值的报表。当您在启动窗口中指定重复效应时显示该报表。请参见第 323 页的 “重复效应协方差参数估计值”。

指标参数化估计值 （仅当固定效应之间存在名义型列时可用。）显示 “指标函数参数化”报表。该报表基于一个模型给出固定效应的参数估计值，在该模型中，名义型固定效应列使用指标 (SAS GLM) 参数化编码并作为连续列处理。有序型列仍使用通常的 JMP 编码方案保持编码。“统计详细信息”附录中第 456 页的“因子模型”对 SAS GLM 和 JMP 编码方案进行了说明。

警告：“指标函数参数化”报表中给出的标准误差、 t 比和其他结果不同于 “参数估计值”报表中的相应数据。这是因为这些估计值估计的是不同的参数。

随机系数 显示随机系数的报表。当您在启动窗口中指定随机效应时显示该报表。请参见第 323 页的 “随机系数”。

随机效应预测 显示随机效应预测的报表。当您在启动窗口中指定随机效应时显示该报表。请参见第 323 页的 “随机效应预测”。

固定效应检验 （仅可用于至少包含一个固定效应的模型。）显示固定效应的检验。当您在启动窗口中指定固定效应时显示该报表。请参见第 324 页的 “固定效应检验”。

序贯检验 （仅可用于至少包含一个固定效应的模型。）显示“序贯（1 型）检验”报表，该报表包含效应按顺序添加到模型后的平方和。基于序贯平方和执行 *F* 检验。请参见第 324 页的 “序贯检验”。

JMP PRO 拟合统计量

“拟合统计量”报表提供用于模型比较的统计量。对于所有拟合统计量，值越小越好。若一个模型包含在另一模型中，两个模型之间可以执行似然比检验。若不是包含关系，谨慎比较似然可以提供有用的信息。相关示例，请参见第 350 页的“拟合空间结构模型”。

“‘拟合统计量’报表的说明”使用以下符号：

- 指定混合模型：

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\gamma + \varepsilon$$

其中 $\mathbf{y}$ 是观测值的 *n*x*1* 向量，β 是固定效应参数的向量，$\boldsymbol{\gamma}$ 是随机效应参数的向量，$\boldsymbol{\varepsilon}$ 是误差的向量。

- 假定 γ 和 ε 具有多元正态分布，其中

$$E\begin{bmatrix}\gamma\\ \varepsilon\end{bmatrix} = \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}$$

且

$$Var\begin{bmatrix}\gamma\\ \varepsilon\end{bmatrix} = \begin{bmatrix}\mathbf{G} & \mathbf{0}\\ \mathbf{0} & \mathbf{R}\end{bmatrix}$$

- 在这些假设下，按以下方式计算 $\mathbf{y}$ 的方差：

$$\mathbf{V} = \mathbf{Z}\mathbf{G}\mathbf{Z}' + \mathbf{R}$$

“拟合统计量”报表的说明

-2 残差对数似然 最终求值为两倍的负残差对数似然，即目标函数。

$$-2log likelihood_{\mathbf{R}}(\mathbf{G},\mathbf{R}) = \log|\mathbf{V}| + \log\left|\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right| + \mathbf{r}'\mathbf{V}^{-1}\mathbf{r} + (n-p)\log(2\pi)$$

其中

$$\mathbf{r} = \mathbf{y} - \mathbf{X}(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}(\mathbf{X}'\mathbf{V}^{-1}\mathbf{y})$$

p 是 X 的秩。仅在比较固定效应部分相同的模型时使用残差似然。请参见“统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

-2 对数似然 求值为两倍的负残差对数似然。请参见“统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

将对数似然用于比较固定、随机和重复效应不同的模型。

AICc 校正 Akaike 信息准则。请参见“统计详细信息”附录中第 477 页的“似然、 AICc 和 BIC”。

BIC Bayesian 信息准则。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。

JMP PRO 收敛得分检验

若模型收敛有问题，将在拟合统计量下显示一条警告消息。图 8.11 显示该警告，指出收敛问题的原因和可能的解决方法。它还包括最终迭代的相对梯度检验。若该检验不显著，则模型可能是正确的但是未完全满足收敛准则。在这种情况下，需慎用模型和结果。请参见第 364 页的“收敛得分检验”。

图 8.11 收敛得分检验

JMP PRO 随机效应协方差参数估计值

“随机效应协方差参数估计值”报表提供您在模型中指定的随机效应协方差参数的详细信息。

协方差参数 列出您在模型中指定的随机效应的所有协方差参数。

注意：当模型仅包含方差分量时，该标记为“方差分量”。

对象 列出生成区组对角协方差矩阵所依据的对象。

估计值 给出效应的估计方差或协方差分量。

注意：当模型等价于 REML 模型时，将 “合计”协方差参数的行添加到表。“合计”协方差分量的估计值是仅为正的方差分量之和。

标准误差 给出协方差分量估计值的标准误差。

95% 下限 给出协方差分量的 95% 置信下限。请参见第 322 页的 “方差分量的置信区间”。

95% 上限 给出协方差分量的 95% 置信上限。请参见第 322 页的 “方差分量的置信区间”。

Wald P 值 给出用于检验协方差参数是否等于零的 *p* 值。仅当您在“拟合模型”启动窗口中选定了 “无界限方差分量”时，该列才显示。

JMP PRO 方差分量的置信区间

用于计算置信限的方法取决于您是否在 “拟合模型”启动窗口中选定了 “无界限方差分量” 。请注意，默认情况下选定 “无界限方差分量”。

- 若选定了“无界限方差分量”，则计算基于 Wald 的置信区间。这些区间渐近有效，但请注意样本较小时它们并不可靠。区间较宽，可能导致您误信某个估计值与零没有显著差异。
- 若未选定“无界限方差分量”，这表示参数的下边界约束为零，此时将使用 Satterthwaite 近似 (Satterthwaite 1946)。置信区间也以零为边界。

JMP PRO 固定效应参数估计值

“固定效应参数估计值”报表提供在模型中指定的固定效应参数的详细信息。对于每个参数，该报表提供以下详细信息：

- 估计值
- 标准误差 (Std Error)
- 假设估计值等于零的 *t* 检验
- 估计值的 95% 置信区间

“固定效应参数估计值”报表包含以下列：

项 提供与估计参数对应的模型项。第一个项始终是截距，除非在 “拟合模型”启动窗口中选择了“无截距”选项。默认情况下，属于高阶项一部分的连续列被中心化。名义型或有序型效应的水平值显示在括号中。有关名义型和有序型项的编码的信息，请参见第 456 页的 “因子模型”。

注意：若连续列涉及随机效应，则该列不会位于中心，即使选择了“模型规格”红色小三角菜单中的 “中心多项式”选项也是如此。

估计值 给出每个项的参数估计值。这是模型中项的系数的估计值。

标准误差 给出参数估计值的标准误差的估计值。

分母自由度 给出效应检验的分母自由度，即误差的自由度。使用 Kenward-Roger 一阶近似计算分母自由度。请参见第 374 页的 “Kackar-Harville 校正”。

t 比 检验参数的真实值是否为 0。*t* 比是估计值与其标准误差之比。在有关模型的一般假设下，*t* 比在原假设下服从 Student *t* 分布。

概率 >|t| 列出 *t* 比的双侧检验的 *p* 值。

95% 下限 显示参数的 95% 置信下限。

95% 上限 显示参数的 95% 置信上限。

JMP PRO 重复效应协方差参数估计值

“重复效应协方差参数估计值”报表提供您在模型中指定的重复效应协方差参数的详细信息。它包括每个参数的估计值、标准误差和 95% 置信界限。对于各向同性空间模型，协方差参数估计值具有关于极差、nugget 和 sill 的解释。请参见第 372 页的“变差图”。

注意：方差是效应与自身的协方差。

“重复效应协方差参数估计值”报表包含以下列：

协方差参数 列出模型中重复效应的所有协方差参数。

估计值 给出效应的估计方差或协方差分量。

标准误差 给出方差或协方差分量估计值的标准误差。

95% 下限 给出协方差分量的 95% 置信下限。请参见第 322 页的 “方差分量的置信区间”。

95% 上限 给出协方差分量的 95% 置信上限。请参见第 322 页的 “方差分量的置信区间”。

JMP PRO 随机系数

对于每个随机效应，“混合模型”特质提供显示估计系数的报表和显示协方差估计值矩阵的报表。系数报表的每行对应于随机效应的一个水平。该行显示与随机效应的水平关联的所有系数估计值。将随机系数估计值与固定效应估计值一起使用来创建随机效应的任何特定水平的预测。

JMP PRO 随机效应预测

对于模型中的每个随机效应，该报表给出称为**最佳线性无偏预测变量** (BLUP) 的估计值、它的标准误差、自由度和基于 Satterthwaite 的置信区间。

标准误差的估计需要计算 BLUP 协方差矩阵，这可能需要很长时间。若计算时间很长，将显示一个进度条。

JMP PRO 固定效应检验

该报表为模型中的每个固定效应显示显著性检验。给定效应的检验用于检验与该效应关联的所有参数为零这个原假设。对于单个连续解释变量，一个效应可能只有一个参数。在这种情况下，该检验等价于“固定效应参数估计值”报表中该项的 *t* 检验。根据名义型或有序型效应的水平数，此类效应可有若干关联参数。此类效应的效应检验用于检验所有关联参数是否均为零。

“固定效应检验”报表包含以下列：

源 列出模型中的固定效应。

参数数目 显示与效应关联的参数个数。连续效应有一个参数。名义型或有序型效应的参数个数为其水平数减 1。一个交叉效应的参数个数是每个单独效应的参数个数的乘积。

分子自由度 显示效应检验的分子自由度。

分母自由度 显示效应检验的分母自由度（误差的自由度）。使用 Kenward-Roger 一阶近似计算分母自由度。请参见第 374 页的 “Kackar-Harville 校正”。

F 比 给出用于检验效应为零的计算得到的 *F* 比。

概率 > F 提供效应检验的 *p* 值。

JMP PRO 序贯检验

该报表显示固定效应的序贯（类型 I）检验。当效应按顺序添加到模型时，报表包含平方和。输入顺序由效应在“拟合模型”启动窗口的“构造模型效应”列表中的显示顺序来定义。

构成序贯检验基础的平方和亦称 **I 型平方和**。通过按照指定的效应输入顺序执行拟合模型的步骤来计算该平方和。考虑一个特定的效应。为包含在该效应**之前**输入的所有效应的模型计算模型平方和。然后为包含这些效应**和**指定效应的模型计算模型平方和。指定效应的序贯平方和是这两个模型平方和的增量。

序贯检验在以下情况下适用：

- 以适当顺序指定的平衡方差分析模型（即：在效应列表中双因子交互作用跟在主效应之后，依此类推）
- 以适当顺序指定的纯嵌套模型
- 以适当顺序指定的多项式回归模型

“序贯 （类型 I）检验”报表包含以下列：

源 列出模型中的固定效应。

参数数目 显示与效应关联的参数个数。连续效应有一个参数。名义型或有序型效应的参数个数为其水平数减 1。一个交叉效应的参数个数是每个单独效应的参数个数的乘积。

分子自由度 显示效应检验的分子自由度。

分母自由度 显示效应检验的分母自由度（误差的自由度）。使用 Kenward-Roger 一阶近似计算分母自由度。请参见第 374 页的 “Kackar-Harville 校正”。

F 比 给出用于检验效应为零的计算得到的 *F* 比。

概率 > F 提供效应检验的 *p* 值。

JMP PRO 多重比较

“多重比较”选项提供各种方法来比较主效应和交互作用效应的最小二乘均值。有关多重比较选项的详细信息，请参见“标准最小二乘法模型”一章中第 112 页的“多重比较”。有关混合模型示例，请参见第 341 页的“比较六月的所有治疗”和第 345 页的“裂区示例”。

在多重比较中仅使用模型的固定效应部分。“多重比较”报表显示最小二乘均值的估计值、标准误差、无效应的 *t* 检验以及 95% 置信区间。该报表后跟您选择的多重比较检验。“所有配对比较”报表提供等价性检验。

JMP PRO 比较斜率

当有一个名义型项、一个连续型项以及它们针对固定效应的交互作用效应时，就会显示“比较斜率”选项。该选项生成一个报表，该报表支持您比较协方差分析 (ANCOVA) 模型中的斜率。该报表将交互作用效应每个水平的斜率与总体斜率作比较。该比较使用包含总平均值的均值分析 (ANOM)。有关均值分析 (ANOM) 报表的详细信息，请参见“标准最小二乘法模型”一章中第 116 页的“与总平均值比较”。

总平均值斜率是斜率的加权平均值，其中，权重与斜率估计值的方差成反比。这些方差是“与总平均斜率的差异”表中的“标准误差”列的平方值。

JMP PRO 边缘模型推断

边缘模型图基于边缘预测值和边缘残差。

“预测值 - 实际值”图 标绘实际值与模型预测的值，但是不考虑随机效应。默认显示“预测值 - 实际值”图。请参见第 326 页的 ““预测值 - 实际值”图”。

残差图 提供评估模型拟合的残差图，不考虑随机效应。请参见第 326 页的 “残差图”。

刻画器 , 等高线刻画器 , 混料刻画器 , 曲面刻画器 提供刻画器来检查响应和模型项之间的关系，不考虑随机效应。请参见第 326 页的 “刻画器”。

JMP PRO “预测值 - 实际值”图

默认显示“预测值-实际值”图。提供一种方式来直观评估模型拟合，它反映随机效应导致的变异。它针对 Y 的边缘预测值标绘 Y 的观测值。这是在您选择“保存列”>“预测公式”时获取的预测值。

将线性混合模型表示为 $E[\mathbf{Y}|\gamma] = \mathbf{X}\beta + \mathbf{Z}\gamma$。其中，β 是固定效应系数的向量，γ 是随机效应系数的向量。**边缘预测**是从预测模型的固定效应部分得到的预测，通过 $\mathbf{X}\hat{\beta}$ 给出。

JMP PRO 残差图

边缘残差反映仅基于固定效应拟合的预测误差。边缘残差是实际值与您选择“保存列”>“预测公式”时获取的预测值之间的差值。

将线性混合模型表示为 $E[\mathbf{Y}|\gamma] = \mathbf{X}\beta + \mathbf{Z}\gamma$。其中，β 是固定效应系数的向量，γ 是随机效应系数的向量。**边缘残差**是来自预测模型的固定效应部分的残差：

$$\mathbf{r} = \mathbf{Y} - \mathbf{X}\hat{\beta}$$

“残差图”选项提供三种直观方法来评估模型拟合：

“预测值 - 残差”图 显示针对 Y 的预测值标绘的残差。您通常希望残差值在零附近随机分散。

残差分位数图 显示针对标准正态分布的分位数标绘的残差分位数。还显示残差的条形图。若残差服从正态分布，正态分位数图上的点应近似落在红色对角线上。此类图也称为分位数 - 分位数图，或 Q-Q 图。正态分位数图还显示 Lilliefors 置信界限 (Conover 1999)。

“行号 - 残差”图 显示针对行号标绘的残差。该图可帮助您检测观测值相对行顺序变化的模式。

JMP PRO 刻画器

边缘模型刻画器中的图基于边缘预测值。这是在您选择“保存列”>“预测公式”时获取的预测值。

将线性混合模型表示为 $E[\mathbf{Y}|\gamma] = \mathbf{X}\beta + \mathbf{Z}\gamma$。其中，β 是固定效应系数的向量，γ 是随机效应系数的向量。**边缘预测**是从预测模型的固定效应部分得到的预测，通过 $\mathbf{X}\hat{\beta}$ 给出。

注意：边缘模型刻画器图仅显示特殊设置的固定效应的预测。该刻画器仅显示对应于固定效应的单元格。在可以显示随机效应的其他刻画器中，仅固定效应的设置决定预测值。

提供四种刻画器：

- 刻画器
- 等高线刻画器
- 混料刻画器

- 曲面刻画器

将启用适用于您所拟合模型的选项。有关刻画器的示例，请参见图 8.21。有关“曲面刻画器”的示例，请参见图 8.42。有关“曲面刻画器”的详细信息，请参见《刻画器指南》。

JMP PRO 条件模型推断

条件模型诊断图基于条件残差。条件残差反映同时拟合固定效应和随机效应时的预测误差。

“条件预测值 - 实际值”图 标绘实际值与模型预测的值，考虑随机效应。有随机效应时，默认显示“条件预测值 - 实际值”图。请参见第 327 页的 “条件预测值 - 实际值”图。

条件残差图 提供评估模型拟合的残差图，考虑随机效应。请参见第 327 页的“条件残差图”。

条件刻画器 , 条件等高线刻画器 , 条件混料刻画器 , 条件曲面刻画器 提供刻画器来检查响应和模型项之间的关系，考虑随机效应。请参见第 328 页的 “条件刻画器”。

变差图 提供变差图，该图显示观测值之间的距离增加时协方差的变化。选择残差结构时，您可以选择要用作时间或空间坐标的列。请参见第 328 页的 “变差图”。

JMP PRO “条件预测值 - 实际值”图

默认显示“条件预测值-实际值”图。提供一种方式来直观评估模型拟合，它解释随机效应导致的变异。它针对 Y 的条件预测值标绘 Y 的观测值。这是在您选择“保存列”>“条件预测公式”时获取的预测值。

将线性混合模型表示为 $E[\mathbf{Y}|\gamma] = \mathbf{X}\beta + \mathbf{Z}\gamma$。其中，$\beta$ 是固定效应系数的向量，γ 是随机效应系数的向量。**条件预测**是从模型中获取的预测，通过 $\mathbf{X}\hat{\beta} + \mathbf{Z}\hat{\gamma}$ 给出。

JMP PRO 条件残差图

条件残差反映同时拟合固定和随机效应时的预测误差。条件残差是实际值与您选择“保存列”>“条件预测公式”时获取的条件预测值之间的差值。

将线性混合模型表示为 $E[\mathbf{Y}|\gamma] = \mathbf{X}\beta + \mathbf{Z}\gamma$。其中，$\beta$ 是固定效应系数的向量，γ 是随机效应系数的向量。**条件残差**计算如下：

$$\mathbf{r} = \mathbf{Y} - (\mathbf{X}\hat{\beta} + \mathbf{Z}\hat{\gamma})$$

“条件残差图”选项提供三种直观方法来评估模型拟合：

“预测值 - 条件残差”图 显示针对 Y 的条件预测值标绘的条件残差。您通常希望条件残差在零附近随机分散。

条件残差分位数图 显示针对标准正态分布的分位数标绘的条件残差分位数。还显示条件残差的条形图。若条件残差服从正态分布，正态分位数图上的点应近似落在红色对角线上。此类

图也称为分位数 - 分位数图，或 Q-Q 图。正态分位数图还显示 Lilliefors 置信界限 (Conover 1999)。

“行号 - 条件残差”图 显示针对行号标绘的条件残差。该图可帮助您检测观测值相对行顺序变化的模式。

JMP PRO 条件刻画器

条件模型刻画器图基于条件预测值。这是在您选择“保存列”>“条件预测公式”时获取的预测值。

将线性混合模型表示为 $E[\mathbf{Y}|\gamma] = \mathbf{X}\beta + \mathbf{Z}\gamma$。其中，$\beta$ 是固定效应系数的向量，γ 是随机效应系数的向量。**条件预测**是从模型中获取的预测，通过 $\mathbf{X}\hat{\beta} + \mathbf{Z}\hat{\gamma}$ 给出。

提供四种刻画器：

- 条件刻画器
- 条件等高线刻画器
- 条件混料刻画器
- 条件曲面刻画器

将启用适用于您所拟合模型的选项。有关刻画器的示例，请参见图 8.21。有关“曲面刻画器”的示例，请参见图 8.42。有关刻画器的详细信息，请参见《刻画器指南》。

JMP PRO 变差图

变差图根据观测值的距离描述观测值之间的空间或时间相关性。该图将半方差显示为距离或时间的函数。理论半方差是两个响应值之间差值方差的一半，这两个响应值所在的位置相距给定的距离。请注意，给定距离的半方差和相关性是逆相关的。若给定距离的两个值之间的相关性小，则该距离上两个观测值之间的半方差大。

当您在“拟合模型”窗口中指定任意各向同性重复结构（“AR(1)”、“空间”或“带 Nugget 的空间”）时，默认显示“变差图”。若您指定“残差”结构，通过在红色小三角菜单中选择“变差图”选项，您可以选择要在计算变差图时使用的连续列。您可以包含描述您数据的空间或时间结构的任意数目的列。

初始“变差图”报表显示距离-经验半方差图。有关更多背景和详细信息，请参见第 370 页的“事前相关协方差结构”。

各向同性结构的半方差曲线

针对以下各向同性协方差结构提供半方差曲线：AR(1)、乘方、指数、高斯和球形。对于空间结构，针对带 nugget 和不带 nugget 的模型提供曲线。

使用协方差参数估计值拟合理论模型的曲线。有关所基于的公式，请参见 Chilès and Delfiner (2012) 以及 Cressie (1993)。

使用理论模型来确定您的数据是否符合所选的各向同性结构。若您选择了“残差”结构，可以使用经验变差图来确定您的数据是否显示某个时间或空间结构。若这些点看上去在一条水平线上，这表示相关性不随距离变化，“残差”结构是合适的。若这些点显示一个模式，则拟合各种各向同性模型可能建议一个合适的“重复”结构，使用该结构来重新拟合您的模型。

JMP PRO Nugget

nugget 是从变差图原点处的 0 值处的垂直跳跃，该跳跃幅度为一个很小分离距离的半方差值。带 nugget 的变差图模型在原点处中断。略大于 0 的距离的理论曲线值为 nugget。

JMP PRO “变差图”选项

AR(1) 标绘 “AR(1)”协方差结构的变差图。

空间 标绘 “指数”、“高斯”、“乘方”或 “球形”协方差结构的变差图。

带 Nugget 的空间 标绘带 nugget 的 “指数”、“高斯”、“乘方”或 “球形”协方差结构的变差图。

JMP PRO 协方差和相关性矩阵

“协方差和相关性矩阵”菜单包含用于查看与模型关联的协方差矩阵的选项。

固定效应的协方差 显示或隐藏模型中的固定效应的协方差矩阵。

协方差参数的协方差 显示或隐藏模型中的随机效应的协方差矩阵。矩阵中的效应按以下方式排序：G 侧随机效应、 R 侧随机效应和残余效应。

全部参数的协方差 显示或隐藏模型中的所有效应的协方差矩阵。矩阵中的效应按以下方式排序：固定效应、 G 侧随机效应、 R 侧随机效应和残余效应。

固定效应的相关性 显示或隐藏模型中的固定效应的相关性矩阵。

JMP PRO 保存列

“保存列”菜单中的每个选项将一个或多个新列添加到数据表。

预测公式 创建名为**预测公式 “< 列名 >”**的新列，该列包含公式和边缘均值预测值。将添加“预测”列属性，指出预测的源。请参见第 325 页的 “边缘模型推断”。

预测值标准误差 创建名为**预测值标准误差 “< 列名 >”**的新列，该列包含预测的边缘均值响应的标准误差。

均值置信区间 创建名为 **95% 均值下限 : < 列名 >** 和 **95% 均值上限 : < 列名 >** 的两个新列。这些列包含响应均值的 95% 置信下限和上限。这些区间包含估计值变异，但不包含响应变异。

单值置信区间 （可用于仅包含 G 侧效应的模型。）创建名为 95% 单值下限 : < 列名 > 和 95% 单值上限 : < 列名 > 的两个新列。这些列包含单个响应值的 95% 置信下限和上限。这些区间包括响应及其估计值的变异。

残差 创建名为残差 “< 列名 >”的新列，该列包含观测的响应值减去边缘均值预测值所得的值。请参见第 325 页的 “边缘模型推断”。

条件预测公式 创建名为条件预测公式 “< 列名 >”的新列，该列包含公式和条件均值预测值。将添加 “预测”列属性，指出预测的源。请参见第 328 页的 “条件刻画器”。

条件预测值标准误差 创建名为条件预测值标准误差 “< 列名 >”的新列，该列包含预测的条件均值响应的标准误差。

条件均值置信区间 （可用于包含 G 侧效应的模型。）创建名为 95% 条件均值下限 : < 列名 > 和 95% 条件均值上限 : < 列名 > 的两个新列。这些列包含条件预测期望值的 95% 置信下限和上限。置信区间包括带随机效应的模型的随机效应估计值。请参见第 327 页的 “条件模型推断”。

条件残差 创建名为条件残差 “< 列名 >”的新列，该列包含观测的响应值减去条件均值预测值所得的值。请参见第 327 页的 “条件模型推断”。

保存模拟公式 （仅可用于方差分量和随机系数模型。）将使用估计参数为您拟合的模型生成模拟值的公式所在的列保存到数据表中。该列可用在模拟实用工具中，作为 “换入列”。请参见 《基本分析》。

JMP PRO “混合模型”特质的其他示例

- “重复测量示例”
- “裂区示例”
- “空间示例：均匀性试验”
- “相关响应示例”

JMP PRO 重复测量示例

考虑 Cholesterol Stacked.jmp 样本数据表。执行了一项研究来检验两种新胆固醇药物相对于对照药物的效果。将具有高胆固醇的二十名患者随机分配给四个治疗方案之一（两个实验药物、对照药物和安慰剂）。在研究期间分别在以下六个时间测量每名患者的总胆固醇：四月、五月和六月的第一天的早上和下午。您想了解这两种新药物对于降低胆固醇是否有效以及时间和治疗是否有交互作用。

JMP PRO 背景

过去一直使用以下两种方法来分析这类设计：

- 多元方差分析 (MANOVA)
- 进行 Huynh-Feldt (1976) 或 Greenhouse-Geisser (1959) 校正的时间裂区一元方差分析 (ANOVA)

在“拟合模型”中使用“多元方差分析”特质可提供这两个选项。这两个选项是对协方差结构建模的两个极端。MANOVA 分析假定一个非结构化协方差结构，其中单独估计所有方差和协方差。独立的时间裂区分析假定所有误差是独立的。在高斯数据案例中，这等价于假定复合对称性协方差结构。

这两个模型可能得到有关治疗效应的截然不同的结论。当您假定复杂的协方差结构时，使用数据中的信息来估计协方差参数。若您拟合过多的协方差参数，可能会过拟合模型。对重复测量数据建模时，必须找到平衡处理这些问题的协方差结构。

- 模型过拟合时，检验差值的功效小于您假定相对简单的协方差结构时的功效。
- 模型拟合不足时，将失去对第一类错误的控制。在一些情况下，这导致拒绝率增大。在另一些情况下，拒绝率可能由于方差增大而减小。

JMP PRO 协方差结构

“混合模型”特质拟合各种协方差结构。对于时间上的重复测量，Toeplitz 协方差结构和一阶自回归 (AR(1)) 协方差结构通常提供合适的相关性结构。这些结构允许有相关性的观测而不会过拟合模型。AR(1) 假定一个公共方差参数，而具有不等方差的 Toeplitz 协方差矩阵则为每个重复测量变量单元估计唯一的方差。请参见第 317 页的“重复协方差结构要求”。

在该示例中，您拟合四个协方差结构。观测次数 *J* 为 6。

- 第 332 页的“协方差结构：非结构化”。非结构化模型一共拟合的协方差参数数目为 *J*(*J*+1)/2。在该示例中，模型拟合 21 个方差。
- 第 334 页的“协方差结构：残差”。“残差”模型等价于通常的方差分量结构。在该示例中，模型拟合两个方差。
- 第 336 页的“协方差结构：Toeplitz”。Toeplitz 模型拟合 2*J*-1 个协方差参数。在该示例中，模型拟合 11 个方差。
- 第 338 页的“协方差结构：AR(1)”。该模型拟合两个协方差参数。一个参数确定方差，另一个参数确定协方差如何随时间变化。

您使用 AICc 评估模型拟合效果。还可以使用 BIC 准则。在该示例中，这两个准则选择同一模型。您选择一个最佳协方差结构，然后继续执行其他分析：

- 第 340 页的“使用 AR(1) 结构进行进一步分析”
- 第 343 页的“AR(1) 模型的回归模型示例”

提示：完成该示例中的步骤时，请保持“拟合模型”启动窗口打开。

JMP PRO 数据结构

Cholesterol.jmp 数据表采用通常用于记录重复测量数据的格式。要使用“混合模型”特质分析这些数据，每个胆固醇测量值需要在自己的行中，如 Cholesterol Stacked.jmp 中所示。要构造 Cholesterol Stacked.jmp，请使用“表”>“堆叠”来堆叠 Cholesterol.jmp 中的数据。

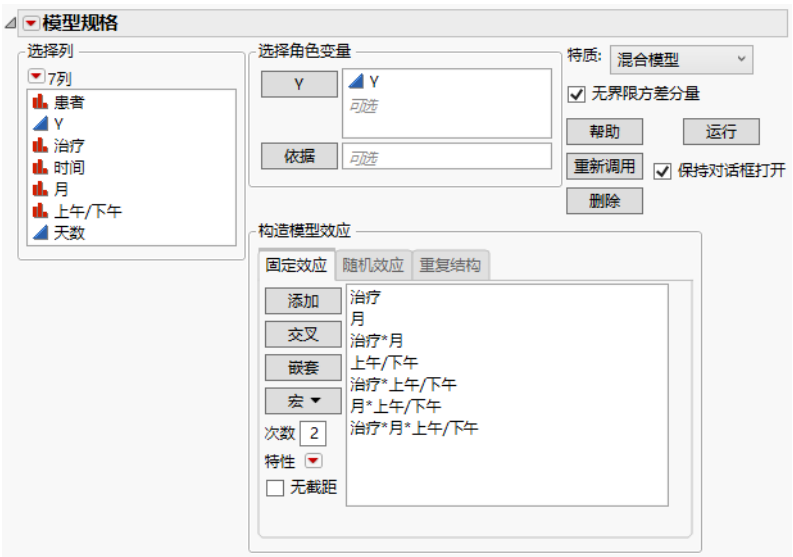
使用公式构造的堆叠表中的**天数**列。**天数**列给出测量胆固醇时已进行研究的天数。建模类型为连续。这是必要的，因为“AR(1)”协方差结构要求重复效应是连续的。

JMP PRO 协方差结构：非结构化

首先使用 “非结构化”协方差结构来拟合模型。

1. 选择**帮助** > **样本数据库**，然后打开 Cholesterol Stacked.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**保持对话框打开**，以便您可以在下一示例中返回到启动窗口。
4. 选择 Y 并点击 Y。
5. 从 “特质”列表中选择**混合模型**。
6. 选择**治疗**、**月**和**上午 / 下午**，然后选择**宏** > **完全析因**。

图 8.12 显示完成的 “固定效应”选项卡的 “拟合模型”启动窗口



7. 选择**重复结构**选项卡。

8. 从“结构”列表中选择**非结构化**。
9. 选择**时间**，然后点击**重复**。**重复**列定义对象内的重复测量值。
10. 选择**患者**并点击**对象**。

> **注意：**“非结构化”协方差模型不允许重复结构变量采用重复值。在该示例中，假定对象在治疗内嵌套且在每个治疗内使用值 1、2、3、4 和 5 对患者进行了编号。运行该分析时将显示一个警告。您需要重新对患者编号以使重复变量的每个值具有不同的标识符。或者，您需要在数据表中创建一列表示治疗内嵌套并将该效应作为“对象”输入。

图 8.13 显示完成的“重复结构”选项卡的“拟合模型”启动窗口

11. 点击**运行**。

“拟合混合模型”报表显示在图 8.14 中。因为要使用 AICc 或 BIC 来比较三个模型，您关注“拟合统计量”报表。非结构化模型的 AICc 为 703.84。

“重复效应协方差参数估计值”报表显示所有 21 个协方差参数的估计值。如您所希望的，在时间上挨得更近的观测值比时间相隔较远的观测值具有更高的协方差。此外，方差随时间增加。

图 8.14 “拟合混合模型”报表 - “非结构化”协方差结构

拟合混合模型

▷ **“预测值-实际值”图**

拟合统计量

-2 残差对数似然	554.98818
-2 对数似然	557.89101
AICc	703.83696
BIC	773.32814

重复效应协方差参数估计值

重复效应: 时间
对象: 患者

协方差参数	估计值	标准误差	95% 下限	95% 上限
Var(April AM)	18.725	6.6202872	5.7494754	31.700525
Cov(April PM,April AM)	18.354884	6.6035621	5.4121399	31.297628
Var(April PM)	19.268932	6.8125963	5.9164888	32.621376
Cov(May AM,April AM)	9.2756709	8.4629182	-7.311344	25.862686
Cov(May AM,April PM)	5.5074058	8.3704001	-10.89828	21.913089
Var(May AM)	56.603347	20.012305	17.379949	95.826744
Cov(May PM,April AM)	9.4147226	8.5038584	-7.252534	26.081979
Cov(May PM,April PM)	6.6230805	8.4532303	-9.944946	23.191108
Cov(May PM,May AM)	55.365523	19.83529	16.489068	94.241978
Var(May PM)	57.058089	20.173081	17.519577	96.596602
Cov(June AM,April AM)	1.1945478	8.6266098	-15.7133	18.102392
Cov(June AM,April PM)	0.3183447	8.7461245	-16.82374	17.460434
Cov(June AM,May AM)	1.106725	14.992149	-28.27735	30.490796
Cov(June AM,May PM)	0.6455101	15.050552	-28.85303	30.14405
Var(June AM)	63.512277	22.454981	19.501323	107.52323
Cov(June PM,April AM)	1.5810194	8.7687993	-15.60551	18.76755
Cov(June PM,April PM)	0.7647113	8.8882627	-16.65596	18.185386
Cov(June PM,May AM)	0.9262595	15.232066	-28.92804	30.780561
Cov(June PM,May PM)	0.6543895	15.292238	-29.31785	30.626625
Cov(June PM,June AM)	63.878686	22.700344	19.386829	108.37054
Var(June PM)	65.568482	23.181959	20.132677	111.00429

▷ **固定效应参数估计值**

固定效应检验

源	参数数目	分子自由度	分母自由度	F 比	概率>F
治疗	3	3	16.0	274.96713	<.0001*
月	2	2	15.0	340.48166	<.0001*
治疗*月	6	6	18.3	123.47461	<.0001*
上午/下午	1	1	16.0	360.93593	<.0001*
治疗*上午/下午	3	3	16.0	0.6339843	0.6038
月*上午/下午	2	2	15.0	1.1988247	0.3289
治疗*月*上午/下午	6	6	18.3	1.1642781	0.3671

JMP PRO 协方差结构：残差

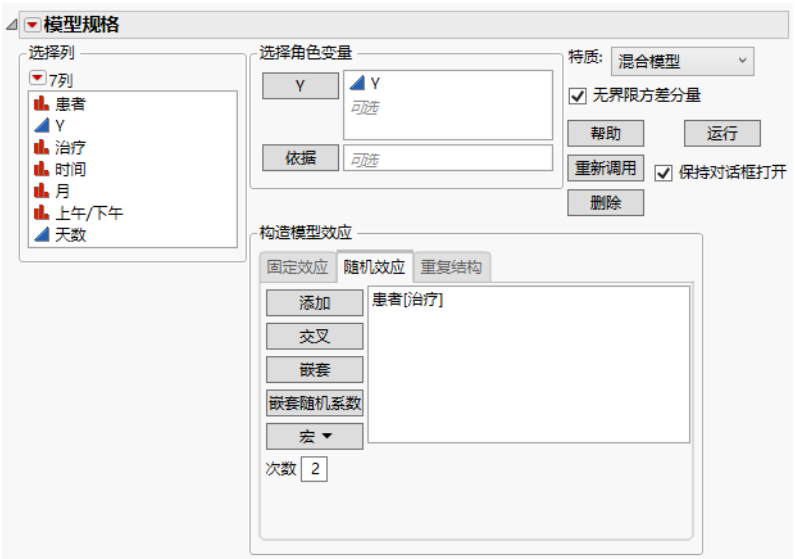
拟合裂区模型时，“残差”协方差结构是合适的。

1. 完成第 332 页的“协方差结构：非结构化”中的第 1 步到第 7 步。
2. 在**重复结构**选项卡上，从“结构”列表中选择**残差**。
3. 若您接上例继续操作，请删除**时间**和**患者**。

   否则将显示一条警告：“当选择残差协方差结构时忽略重复列和对象列。”系统提供您选项来点击**确定**以继续分析。
4. 选择**随机效应**选项卡。

5. 选择**患者**，然后点击**添加**。
6. 选择 “随机效应”区域中的**患者**，选择**治疗**列，然后点击**嵌套**。

图 8.15 显示完成的 “随机效应”选项卡的 “拟合模型”启动窗口



7. 点击**运行**。

“拟合混合模型”报表显示在图 8.16 中。“拟合统计量”报表显示 “残差”模型的 AICc 是 832.55，而 “非结构化”模型的 AICc 为 703.84。

两个协方差参数的估计值显示在 “随机效应协方差参数估计值”报表中。这些是 “‘患者’嵌套在 ‘治疗’内”方差的估计值和 “残差”方差的估计值。

图 8.16 “拟合混合模型”报表 - “剩余误差”协方差结构

拟合混合模型

“预测值-实际值”图 “条件预测值-实际值”图

拟合统计量

-2 残差对数似然	721.03949
-2 对数似然	765.45515
AICc	832.55192
BIC	889.92993

随机效应协方差参数估计值

方差分量	估计值	标准误差	95% 下限	95% 上限	Wald P 值
患者[治疗]	11.707432	6.2749012	-0.591148	24.006012	0.0621
残差	35.081923	5.546939	26.320843	49.105827	
合计	46.789355	7.7386523	34.678063	66.607248	

固定效应参数估计值

随机系数

固定效应检验

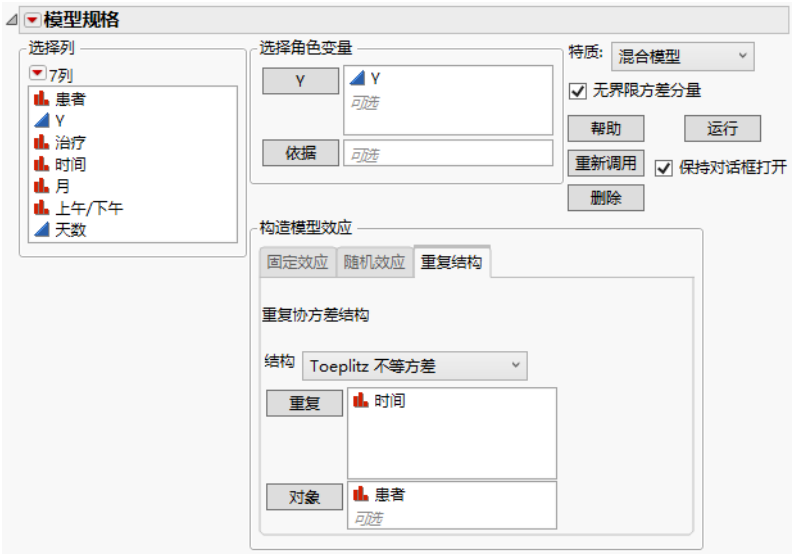
源	参数数目	分子自由度	分母自由度	F 比	概率>F
治疗	3	3	16.0	274.96713	<.0001*
月	2	2	80.0	622.88066	<.0001*
治疗*月	6	6	80.0	226.49464	<.0001*
上午/下午	1	1	80.0	13.700114	0.0004*
治疗*上午/下午	3	3	80.0	0.0240643	0.9949
月*上午/下午	2	2	80.0	0.0324977	0.9680
治疗*月*上午/下午	6	6	80.0	0.030708	0.9999

JMP PRO 协方差结构：Toeplitz

使用 “Toeplitz 不等方差”结构拟合模型。

1. 完成第 332 页的“协方差结构：非结构化”中的第 1 步到第 6 步。
2. 若您接上例继续操作，请选择**随机效应**选项卡上的**患者 [治疗]**，然后点击**删除**。
   若您同时包含随机效应和重复效应，估计这两个效应的数据通常不足。
3. 选择**重复结构**选项卡。
4. 从 “结构”列表中选择 **Toeplitz 不等方差**。
5. 选择**时间**，然后点击**重复**。
6. 选择**患者**并点击**对象**。

图 8.17 显示完成的 “重复结构”选项卡的 “拟合模型”启动窗口



7. 点击运行。

图 8.18 “拟合混合模型”报表 - Toeplitz 不等方差结构

拟合混合模型

“预测值-实际值”图

拟合统计量

-2 残差对数似然	659.09915
-2 对数似然	688.02972
AICc	788.02972
BIC	855.59193

重复效应协方差参数估计值

对象: 患者

协方差参数	估计值	标准误差	95% 下限	95% 上限
Toeplitz 相关性(1)	0.6230225	0.0856377	0.4551756	0.7908693
Toeplitz 相关性(2)	0.2555783	0.1463264	-0.031216	0.5423728
Toeplitz 相关性(3)	0.0962006	0.1887883	-0.273818	0.466219
Toeplitz 相关性(4)	-0.1112	0.2248683	-0.551934	0.3295338
Toeplitz 相关性(5)	0.2427218	0.1955609	-0.140571	0.6260141
方差(April AM)	17.552396	6.9782256	3.8753253	31.229467
方差(April PM)	24.652754	8.3215651	8.3427857	40.962721
方差(May AM)	73.950898	25.930509	23.128035	124.77376
方差(May PM)	63.668022	21.766899	21.005684	106.33036
方差(June AM)	69.45679	22.578117	25.204495	113.70909
方差(June PM)	52.484909	19.171565	14.909332	90.060486

固定效应参数估计值

固定效应检验

源	参数数目	分子自由度	分母自由度	F 比	概率>F
治疗	3	3	14.1	230.25344	<.0001*
月	2	2	23.7	363.8599	<.0001*
治疗*月	6	6	26.2	134.01838	<.0001*
上午/下午	1	1	5.2	106.73459	0.0001*
治疗*上午/下午	3	3	5.2	0.1874794	0.9007
月*上午/下午	2	2	26.9	0.036706	0.9640
治疗*月*上午/下午	6	6	34.6	0.0347301	0.9998

注意：JMP 中的“混合模型”特质报告相关性，而 SAS 中的 PROC MIXED 报告协方差。

“拟合统计量”报表显示具有不等方差的 Toeplitz 模型的 AICc 为 788.03。将该数字与残差模型的 832.55 以及非结构化模型的 703.84 进行比较。

“Toeplitz 不等方差”结构需要估计 11 个协方差参数。这些估计值显示在“重复效应协方差参数估计值”报表中。显示 Toeplitz 相关性估计值，后跟每个时间点的方差估计值。有关如何参数化该矩阵的信息，请参见第 366 页的“重复测量”。

JMP PRO 协方差结构：AR(1)

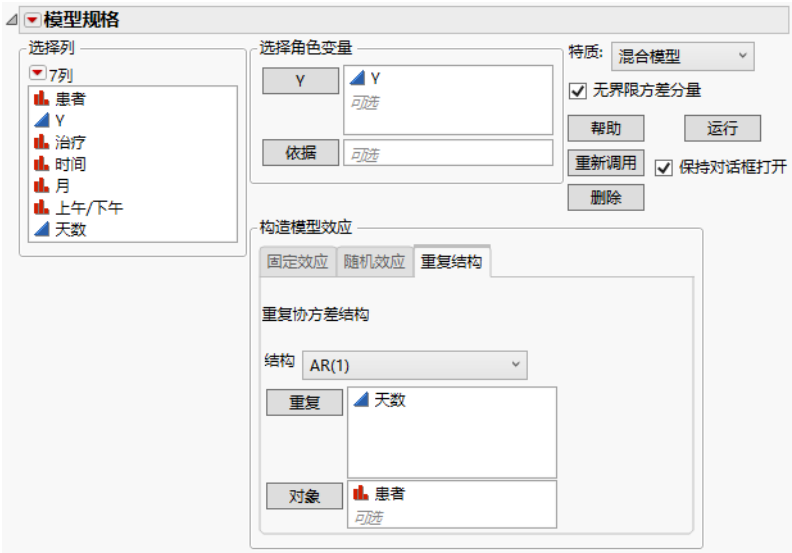
最后，拟合“AR(1)”结构。

1. 完成第 332 页的“协方差结构：非结构化”中的第 1 步到第 6 步。
2. 若是继续前一示例，请在**重复**框中选择**时间**，然后点击**删除**。

   AR(1) 需要重复值的连续变量。
3. 从“结构”列表中选择 AR(1)。

4. 选择**天数**，然后点击**重复**。

图 8.19 显示完成的 “重复结构”选项卡的 “拟合模型”启动窗口



5. 点击**运行**。

“拟合混合模型”报表显示在图 8.20 中。“拟合统计量”报表显示 AR(1) 模型的 AICc 为 652.63。将该数字与残差模型的 832.55、非结构化模型的 703.84、 Toeplitz 不等方差模型的 788.03 进行比较。根据 AICc 准则， AR(1) 模型在这四个模型中是最佳的。

“AR(1)”结构需要估计两个协方差参数。这些估计值显示在“重复效应协方差参数估计值”报表中。“AR(1) 天数”参数估计值是 “AR(1)”结构中相关性参数 ρ 的估计值。

“变差图”显示经验半方差和 AR(1) 模型的曲线。因为**天数**只有五个非零值，只可能有四个距离分类，因此只显示四个点。“AR(1)”结构看起来是合适的。要探索其他结构，请从“变差图”红色小三角菜单中选择相应选项。有关“变差图”选项的详细信息，请参见第 328 页的 “变差图”。

图 8.20 “拟合混合模型”报表 - “AR(1)”协方差结构

拟合混合模型

“预测值-实际值”图

拟合统计量

-2 残差对数似然	577.09954
-2 对数似然	585.53021
AICc	652.62698
BIC	710.00499

重复效应协方差参数估计值

对象: 患者

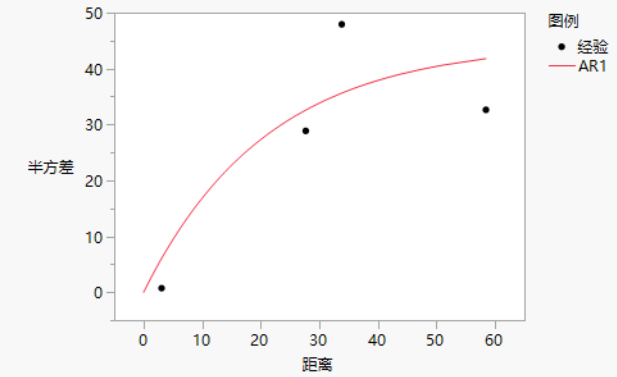
协方差参数	估计值	标准误差	95% 下限	95% 上限
AR(1) 天数	0.9536532	0.0120225	0.9300895	0.9772168
残差	44.579921	9.1281274	30.970215	69.685214

固定效应参数估计值

固定效应检验

源	参数数目	分子自由度	分母自由度	F 比	概率>F
治疗	3	3	22.8	242.61165	<.0001*
月	2	2	80.8	266.42271	<.0001*
治疗*月	6	6	82.7	98.224978	<.0001*
上午/下午	1	1	57.6	461.72978	<.0001*
治疗*上午/下午	3	3	57.6	0.8110288	0.4930
月*上午/下午	2	2	58.0	1.0871135	0.3440
治疗*月*上午/下午	6	6	58.0	1.0275594	0.4168

变差图



JMP PRO 使用 AR(1) 结构进行进一步分析

因为“AR(1)”模型给出最佳拟合，您采用它作为模型并继续分析。“固定效应检验”报表指示“治疗”和“月”之间有显著交互作用以及存在“上午 / 下午”主效应。下面我们来探索这些显著效应。

1. 点击“拟合混合模型”红色小三角菜单，然后选择**边缘模型推断 > 刻画器**。

   通过“边缘模型刻画器”报表（图 8.21），您可以查看**治疗**、**月**和**上午 / 下午**的各种设置对胆固醇水平 (Y) 的效应。

2. 在**月**图中，将红色的垂直虚线从“April”拖到“May”，然后拖到“June”。

请注意预测的 Y 上午测量值在三个月内是下降的，从四月的均值 277.4 降到六月的均值 177.7。

3. 在**治疗**图中，将红色的垂直虚线从 A 拖到 B。

   通过在**月**图中将线从 “April” 拖到 “June”，您看到对于治疗 B，预测的 Y 上午测量值从四月的 276.8 降到六月的 191.2。

4. 在**治疗**图中，将垂直虚线拖到 “Control”，再拖到 “Placebo”。

   请注意，当您将**治疗**设置为 “Control” 或 “Placebo” 时，发现胆固醇水平在这三个月中没有什么变化 （图 8.22）。

   接着，您探索**上午 / 下午**的效应。

5. 通过拖动红色的垂直线，将**治疗**和**月**设置为其水平的全部 12 个组合。

   对于这 12 个组合，预测的胆固醇水平在下午都高于早上的值，这演示了主效应。

请注意，治疗 A 在五月得到的胆固醇读数比治疗 B 低。若该效应是显著的，这可能指示治疗 A 的效果比 B 快。下一节（第 341 页的“比较六月的所有治疗”）说明如何评估治疗。

图 8.21 治疗 A 的边缘刻画器图

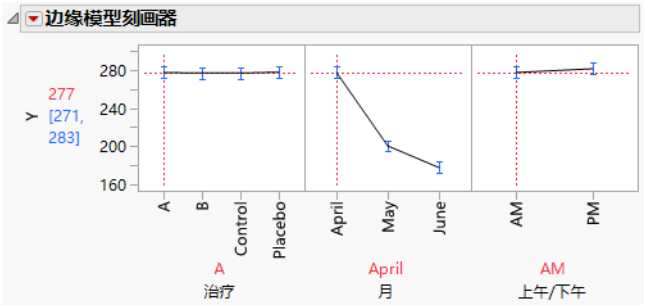
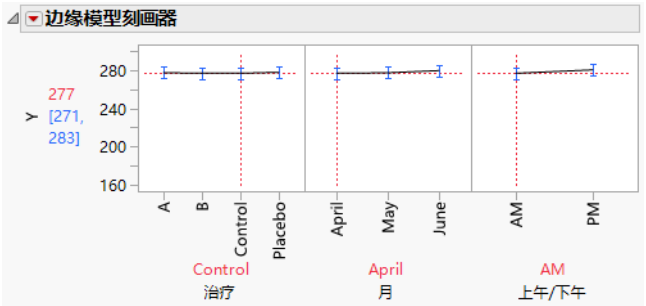


图 8.22 Control 的边缘刻画器图



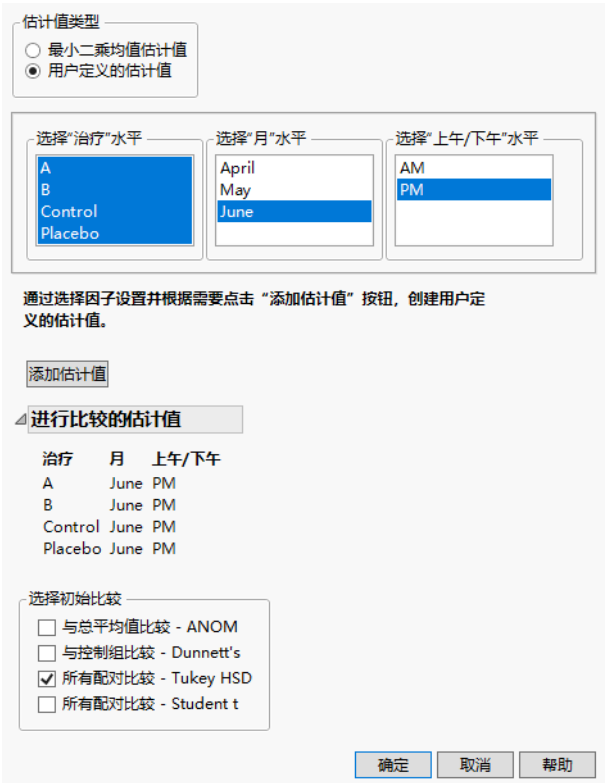
比较六月的所有治疗

在四月、五月和六月这三个月进行研究。您想知道哪些治疗在六月下午的测量值存在差异。

1. 点击“拟合混合模型”红色小三角并选择**多重比较**。

2. 在“估计值类型”下，选择**用户定义的估计值**。
3. 从“选择‘治疗’水平”面板，选择所有四种治疗类型。
4. 从“选择‘月’水平”面板，选择“June”。
5. 从“选择‘上午 / 下午’水平”面板，选择“PM”。
6. 点击**添加估计值**。
7. 从“选择初始比较”列表，选择**所有配对比较 - Tukey HSD**。

图 8.23 完成的“多重比较”窗口



8. 点击**确定**。

图 8.24 六月下午所有治疗的 “Tukey HSD 所有配对比较”报表

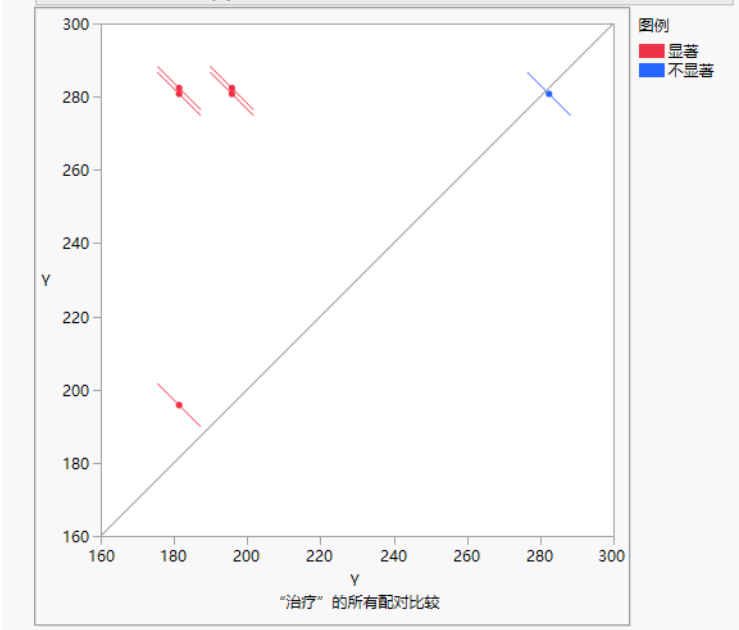
Tukey HSD 所有配对比较

月 = June, 上午/下午 = PM
分位数 = 2.76931，调整的自由度 = 22.8，调整 = Tukey

所有配对差值

治疗	-治疗	差值	标准误差	t 比	概率>|t|	95% 下限	95% 上限
A	B	-14.414	4.222792	-3.41	0.0120*	-26.108	-2.7196
A	Control	-101.014	4.222792	-23.92	<.0001*	-112.708	-89.3196
A	Placebo	-99.414	4.222792	-23.54	<.0001*	-111.108	-87.7196
B	Control	-86.600	4.222792	-20.51	<.0001*	-98.294	-74.9058
B	Placebo	-85.000	4.222792	-20.13	<.0001*	-96.694	-73.3058
Control	Placebo	1.600	4.222792	0.38	0.9810	-10.094	13.2942

所有配对比较散点图



“Tukey HSD 所有配对比较”报表显示“所有配对差值”报表和“所有配对比较散点图”。在六月下午的测量值中，除了“Control”和“Placebo”之外的所有治疗均存在显著差异。

考虑治疗 A 和 B 之间的差值。均值差值为 -14.414，置信区间范围为 -26.108 到 -2.7196。您得到结论：治疗 A 导致的胆固醇测量值的减少超过治疗 B 导致的减少大约在 2.7 点到 26.1 点之间。治疗 A 和 B 与“Control”以及“Placebo”相比都是高度有效的。

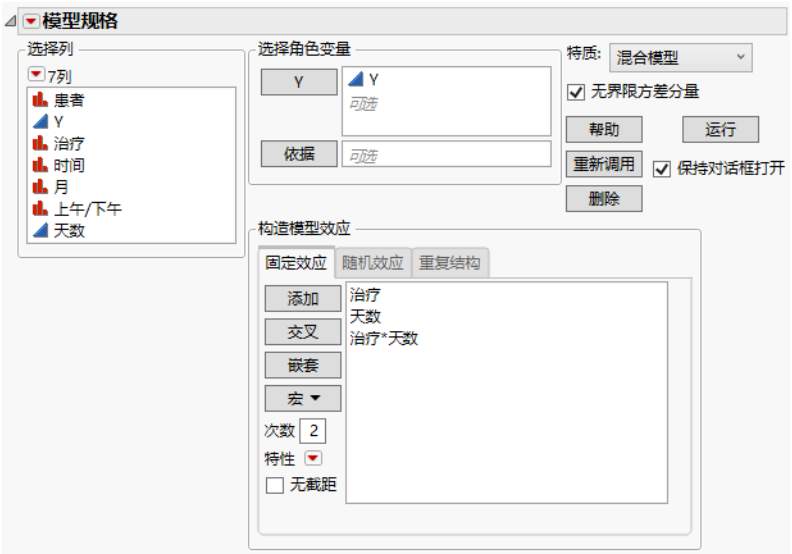
JMP PRO AR(1) 模型的回归模型示例

使用“月”和“上午/下午”分类效应，您比较了胆固醇数据的四个协方差结构。（请注意分类效应对于“非结构化”拟合是必需的。）您已决定使用“AR(1)”协方差结构。

现在假定您要根据连续效应“天数”而非分类效应对治疗效应建模。然后您可以预测治疗期间内的任意时间的胆固醇水平。

1. 在执行第 338 页的“协方差结构：AR(1)”中的第 1 步到第 4 步后，返回到“拟合模型”启动窗口。
2. 在“固定效应”选项卡上，选择现有的固定效应，然后点击**删除**。
3. 选择**治疗**和**天数**，然后选择**宏 > 完全析因**。

图 8.25 显示“固定效应”选项卡的“拟合模型”启动窗口



4. 点击“模型规格”红色小三角并取消选择**中心多项式**。

注意：在默认设置中，交互作用项中使用的连续效应被中心化。通过禁用“中心多项式”选项，交互作用项中使用的连续效应不会被中心化。

5. 点击**运行**。

“拟合混合模型”报表显示在图 8.26 中。您看到“治疗”和“天数”的交互作用是高度显著的，这指示药物的回归存在差异。

注意：要预测不同天数后药物的效果，请使用刻画器。请参见《刻画器指南》。

图 8.26 “拟合混合模型”报表 - 具有连续固定效应的 “AR(1)”协方差结构

拟合混合模型

拟合统计量

-2 残差对数似然	791.54856
-2 对数似然	789.76821
AICc	811.78655
BIC	837.64312

重复效应协方差参数估计值

对象: 患者

协方差参数	估计值	标准误差	95% 下限	95% 上限
AR(1) 天数	0.7983061	0.0473103	0.7055796	0.8910327
残差	100.5539	18.519406	72.239307	149.60244

固定效应参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|	95% 下限	95% 上限
截距	276.17698	1.9946144	52.7	138.46	<.0001*	272.17582	280.17815
治疗[A]	-6.13599	3.4547735	52.7	-1.78	0.0815	-13.06621	0.7942322
治疗[B]	-0.352827	3.4547735	52.7	-0.10	0.9190	-7.283049	6.5773954
治疗[Control]	1.9453265	3.4547735	52.7	0.56	0.5758	-4.984896	8.8755487
天数	-0.735803	0.0505395	53.3	-14.56	<.0001*	-0.83716	-0.634446
治疗[A]*天数	-0.877856	0.087537	53.3	-10.03	<.0001*	-1.053411	-0.7023
治疗[B]*天数	-0.643666	0.087537	53.3	-7.35	<.0001*	-0.819221	-0.46811
治疗[Control]*天数	0.788161	0.087537	53.3	9.00	<.0001*	0.6126056	0.9637164

固定效应检验

源	参数数目	分子自由度	分母自由度	F 比	概率>F
治疗	3	3	52.7	1.302892	0.2832
天数	1	1	53.3	211.9633	<.0001*
治疗*天数	3	3	53.3	76.472494	<.0001*

JMP PRO 裂区示例

“混合模型”特质提供直接简单的方法来指定和分析裂区实验。“混合模型”提供指定效应的选项卡。最终分析针对随机效应。还请注意，可以使用“标准最小二乘法”特质来分析裂区实验。

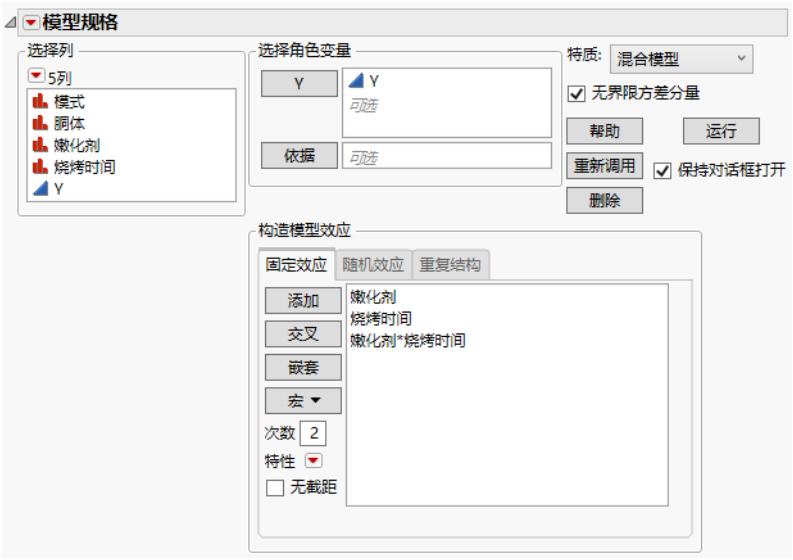
Split Plot.jmp 样本数据表中的数据来自嫩化剂和烹调时间长度对肉的影响的研究。从肉包装厂的胴体中随机选择六个牛肉胴体。选择每个胴体的右肋眼肉，在一致条件下准备三块卷烤肉。给每个烤肉随机分配一个嫩化处理。处理后，使用取心设备标记靠近每块肉中心的四个肉心。

在预热炉中将来自同一酮体的三块烤肉放在一起并准备烹调。30 分钟后，随机从每块烤肉中取一个肉心。按此方式在 36 分钟、42 分钟和 48 分钟后再取肉心。每组冷却到饮用温度时，使用 Warner-Bratzler 设备测量肉心的嫩度。测量值越大指示肉越硬。

您想了解嫩化剂、烧烤时间的影响，特别是了解嫩化剂和烧烤时间之间是否有交互作用。该设计可以达到这个目的。

1. 选择**帮助** > **样本数据库**，然后打开 Split Plot.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 从 “特质”列表中选择**混合模型**。
5. 选择**嫩化剂**和**烧烤时间**，然后选择**宏** > **完全析因**。

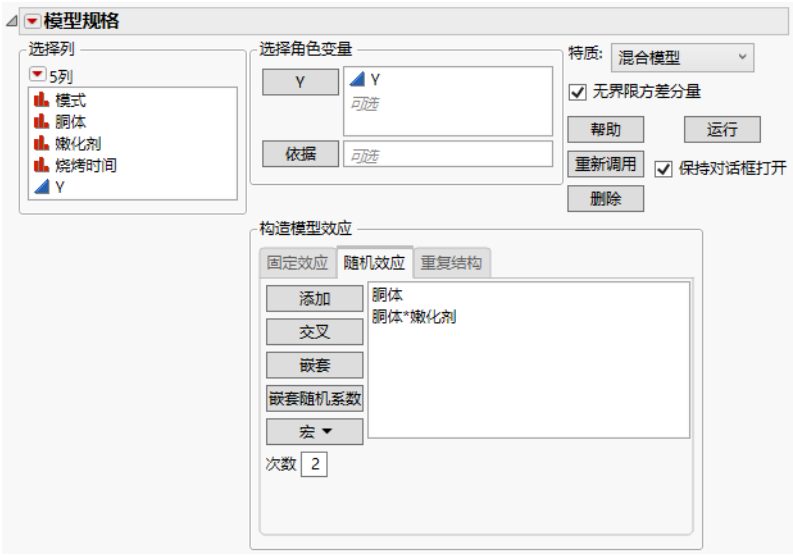
图 8.27 显示完成的“固定效应”选项卡的“拟合模型”启动窗口



6. 选择**随机效应**选项卡。
7. 选择**胴体**，然后点击**添加**以创建随机胴体效应。
8. 选择**胴体**和**嫩化剂**，然后点击**交叉**。

   “胴体 * 嫩化剂”交互作用是整区因子“嫩化剂”的误差项。这等价于“标准最小二乘法”中的“胴体 * 嫩化剂 & 随机”项。

图 8.28 显示完成的“随机效应”选项卡的“拟合模型”启动窗口

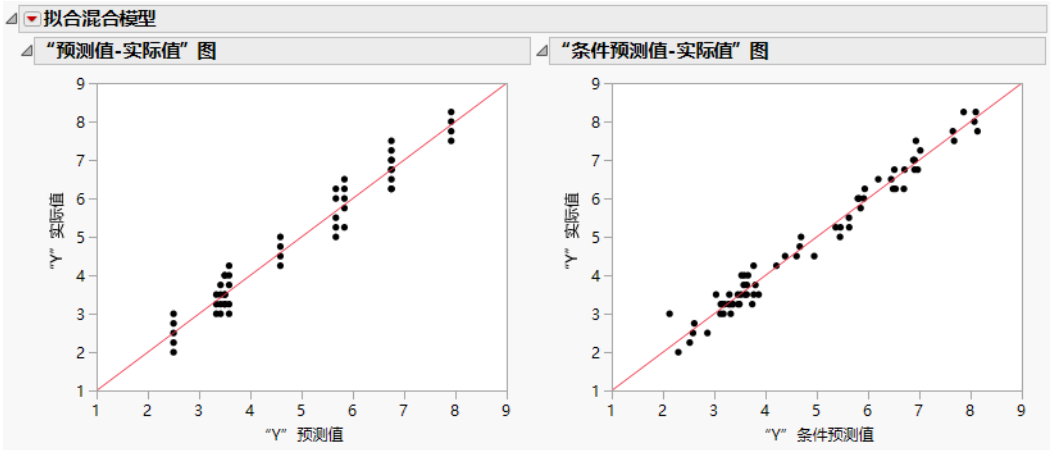


9. 点击运行。

“拟合混合模型”报表显示在图 8.29 中。

“预测值 - 实际值”图和 “条件预测值 - 实际值”图显示模型拟合没有问题，因此您可以继续解释结果。“固定效应检验”报表指示嫩化剂和烧烤时间之间存在显著的交互作用。

图 8.29 “拟合混合模型”报表



拟合统计量

-2 残差对数似然	68.324206
-2 对数似然	23.659713
AICc	62.231142
BIC	87.809705

随机效应协方差参数估计值

方差分量	估计值	标准误差	95% 下限	95% 上限	Wald P 值
胴体	0.0605042	0.0411897	-0.020226	0.1412346	0.1419
胴体*嫩化剂	-0.007958	0.0076058	-0.022865	0.0069492	0.2954
残差	0.0863433	0.0182028	0.0594013	0.1369749	
合计	0.1468476	0.0450325	0.0871689	0.2983101	

固定效应参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|	95% 下限	95% 上限
截距	4.7777778	0.1041208	5.0	45.89	<.0001*	4.5101253	5.0454302
嫩化剂[Control]	0.6388889	0.0389128	10.0	16.42	<.0001*	0.5521867	0.7255911
嫩化剂[Papain]	-0.673611	0.0389128	10.0	-17.31	<.0001*	-0.760313	-0.586909
烧烤时间[30]	2.0555556	0.0599803	45.0	34.27	<.0001*	1.934749	2.1763621
烧烤时间[36]	0.8888889	0.0599803	45.0	14.82	<.0001*	0.7680823	1.0096955
烧烤时间[42]	-1.25	0.0599803	45.0	-20.84	<.0001*	-1.370807	-1.129193
嫩化剂[Control]*烧烤时间[30]	0.4444444	0.084825	45.0	5.24	<.0001*	0.2735982	0.6152907
嫩化剂[Control]*烧烤时间[36]	0.4444444	0.084825	45.0	5.24	<.0001*	0.2735982	0.6152907
嫩化剂[Control]*烧烤时间[42]	-0.583333	0.084825	45.0	-6.88	<.0001*	-0.75418	-0.412487
嫩化剂[Papain]*烧烤时间[30]	-0.326389	0.084825	45.0	-3.85	0.0004*	-0.497235	-0.155543
嫩化剂[Papain]*烧烤时间[36]	-0.409722	0.084825	45.0	-4.83	<.0001*	-0.580569	-0.238876
嫩化剂[Papain]*烧烤时间[42]	0.6458333	0.084825	45.0	7.61	<.0001*	0.474987	0.8166796

随机系数

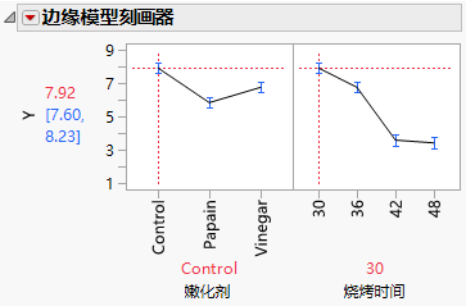
固定效应检验

源	参数数目	分子自由度	分母自由度	F 比	概率>F
嫩化剂	2	2	10.0	190.00818	<.0001*
烧烤时间	3	3	45.0	656.61634	<.0001*
嫩化剂*烧烤时间	6	6	45.0	18.458286	<.0001*

探索嫩化剂和烧烤时间之间的交互作用

1. 点击“拟合混合模型”红色小三角菜单，然后选择**边缘模型推断 > 刻画器**。

图 8.30 将烧烤时间设置为 30 分钟的 “边缘模型刻画器”



2. 在 “烧烤时间”面板中将红色的垂直虚线移动到 36、 42 和 48。

   在图 8.30 中，请注意，当烧烤时间为 30 或 36 分钟时，与 “control”相比，“papain”和“vinegar”嫩化剂可得到显著较低的嫩度得分。但是，烧烤时间为 42 分钟时，没有显著差异。烧烤时间为 48 分钟时，使用“papain”得到比“control”更低的值，但是使用“vinegar”不是这样。除了烧烤时间为 42 分钟外，使用 “papain”得到的嫩度得分比使用 “vinegar”得到的评分都低。

3. 点击 “拟合混合模型”红色小三角并选择**多重比较**。
4. 选择**嫩化剂 * 烧烤时间**。
5. 选择**所有配对比较 - Tukey HSD**，然后点击**确定**。

   图 8.31 显示配对比较列表的部分内容。在刻画器中观测到的“papain”和 “vinegar”之间的大多数差异在统计上是显著的。因此，“papain”似乎是更好的嫩化剂。

图 8.31 多重比较，部分视图

拟合混合模型

“嫩化剂*烧烤时间”的多重比较

最小二乘均值估计值

嫩化剂	烧烤时间	估计	标准误差	自由度	95% 下限	95% 上限
Control	30	7.9166667	0.15214556	20.355	7.5996515	8.2336819
Control	36	6.7500000	0.15214556	20.355	6.4329848	7.0670152
Control	42	3.5833333	0.15214556	20.355	3.2663181	3.9003485
Control	48	3.4166667	0.15214556	20.355	3.0996515	3.7336819
Papain	30	5.8333333	0.15214556	20.355	5.5163181	6.1503485
Papain	36	4.5833333	0.15214556	20.355	4.2663181	4.9003485
Papain	42	3.5000000	0.15214556	20.355	3.1829848	3.8170152
Papain	48	2.5000000	0.15214556	20.355	2.1829848	2.8170152
Vinegar	30	6.7500000	0.15214556	20.355	6.4329848	7.0670152
Vinegar	36	5.6666667	0.15214556	20.355	5.3496515	5.9836819
Vinegar	42	3.5000000	0.15214556	20.355	3.1829848	3.8170152
Vinegar	48	3.3333333	0.15214556	20.355	3.0163181	3.6503485

Tukey HSD 所有配对比较

分位数 = 3.44501，调整的自由度 = 45.0，调整 = Tukey-Kramer

所有配对差值

嫩化剂	烧烤时间	-嫩化剂	-烧烤时间	差值	标准误差	t 比	概率>|t|	95% 下限	95% 上限
Control	30	Control	36	1.16667	0.1696500	6.88	<.0001*	0.58222	1.75111
Control	30	Control	42	4.33333	0.1696500	25.54	<.0001*	3.74889	4.91778
Control	30	Control	48	4.50000	0.1696500	26.53	<.0001*	3.91555	5.08445
Control	30	Papain	30	2.08333	0.1616430	12.89	<.0001*	1.52647	2.64019
Control	30	Papain	36	3.33333	0.1616430	20.62	<.0001*	2.77647	3.89019
Control	30	Papain	42	4.41667	0.1616430	27.32	<.0001*	3.85981	4.97353
Control	30	Papain	48	5.41667	0.1616430	33.51	<.0001*	4.85981	5.97353
Control	30	Vinegar	30	1.16667	0.1616430	7.22	<.0001*	0.60981	1.72353
Control	30	Vinegar	36	2.25000	0.1616430	13.92	<.0001*	1.69314	2.80686
Control	30	Vinegar	42	4.41667	0.1616430	27.32	<.0001*	3.85981	4.97353
Control	30	Vinegar	48	4.58333	0.1616430	28.35	<.0001*	4.02647	5.14019
Control	36	Control	42	3.16667	0.1696500	18.67	<.0001*	2.58222	3.75111
Control	36	Control	48	3.33333	0.1696500	19.65	<.0001*	2.74889	3.91778
Control	36	Papain	30	0.91667	0.1616430	5.67	<.0001*	0.35981	1.47353
Control	36	Papain	36	2.16667	0.1616430	13.40	<.0001*	1.60981	2.72353
Control	36	Papain	42	3.25000	0.1616430	20.11	<.0001*	2.69314	3.80686
Control	36	Papain	48	4.25000	0.1616430	26.29	<.0001*	3.69314	4.80686
Control	36	Vinegar	30	8.882e-16	0.1616430	0.00	1.0000	-0.55686	0.55686
Control	36	Vinegar	36	1.08333	0.1616430	6.70	<.0001*	0.52647	1.64019
Control	36	Vinegar	42	3.25000	0.1616430	20.11	<.0001*	2.69314	3.80686
Control	36	Vinegar	48	3.41667	0.1616430	21.14	<.0001*	2.85981	3.97353
Control	42	Control	48	0.16667	0.1696500	0.98	0.9974	-0.41778	0.75111
Control	42	Papain	30	-2.25000	0.1616430	-13.92	<.0001*	-2.80686	-1.69314
Control	42	Papain	36	-1.00000	0.1616430	-6.19	<.0001*	-1.55686	-0.44314
Control	42	Papain	42	0.08333	0.1616430	0.52	1.0000	-0.47353	0.64019
Control	42	Papain	48	1.08333	0.1616430	6.70	<.0001*	0.52647	1.64019
Control	42	Vinegar	30	-3.16667	0.1616430	-19.59	<.0001*	-3.72353	-2.60981
Control	42	Vinegar	36	-2.08333	0.1616430	-12.89	<.0001*	-2.64019	-1.52647
Control	42	Vinegar	42	0.08333	0.1616430	0.52	1.0000	-0.47353	0.64019
Control	42	Vinegar	48	0.25000	0.1616430	1.55	0.9187	-0.30686	0.80686
Control	48	Papain	30	-2.41667	0.1616430	-14.95	<.0001*	-2.97353	-1.85981
Control	48	Papain	36	-1.16667	0.1616430	-7.22	<.0001*	-1.72353	-0.60981
Control	48	Papain	42	-0.08333	0.1616430	-0.52	1.0000	-0.64019	0.47353
Control	48	Papain	48	0.91667	0.1616430	5.67	<.0001*	0.35981	1.47353
Control	48	Vinegar	30	-3.33333	0.1616430	-20.62	<.0001*	-3.89019	-2.77647
Control	48	Vinegar	36	-2.25000	0.1616430	-13.92	<.0001*	-2.80686	-1.69314
Control	48	Vinegar	42	-0.08333	0.1616430	-0.52	1.0000	-0.64019	0.47353
Control	48	Vinegar	48	0.08333	0.1616430	0.52	1.0000	-0.47353	0.64019

JMP PRO 空间示例：均匀性试验

考虑 Uniformity Trial.jmp 样本数据表。对 8x8 的田块网格执行农事均匀性试验。在均匀性试验中，在没有进行实验处理的一块田地上种植检验作物。测量响应变量（通常为产量）。目标是找出田地变异性特征，以便为在该田地上进行设计实验提供参考。（请参见 Littell et al. 2006, p. 447。）

您的目标是使用这些数据中的信息来设计具有 16 个处理的产量试验。具体而言，您想知道是否要在田地上执行将来的实验：

- 具有 4 个区组（在数据中表示为“季度”）的完整区组设计
- 具有 16 个区组（在数据中表示为“子季度”）的不完整区组设计
- 具有空间相关误差的完全随机化的设计

在这个目标下，空间数据可以视为重复测量，它们具有二维或多维作为重复效应。因此，您可以使用“拟合统计量”报表中的值比较并选择合适的模型。您首先确定是否有显著的空间变异性，然后确定是否有 nugget 效应。

确定是否有 nugget 效应后，接着确定如何拟合最佳空间协方差结构。最后，您拟合分区组模型并将它们与最佳空间结构进行比较。在本例中，使用 AICc 和 BIC 来选择最佳模型。第 372 页的“空间相关性结构”提供了有关 nugget 效应和其他空间术语的详细信息。

提示：本节将引导您完成拟合空间数据的很多方面（从拟合模型到确定最佳协方差结构）。完成各示例中的步骤时，请保持“拟合模型”启动窗口打开。

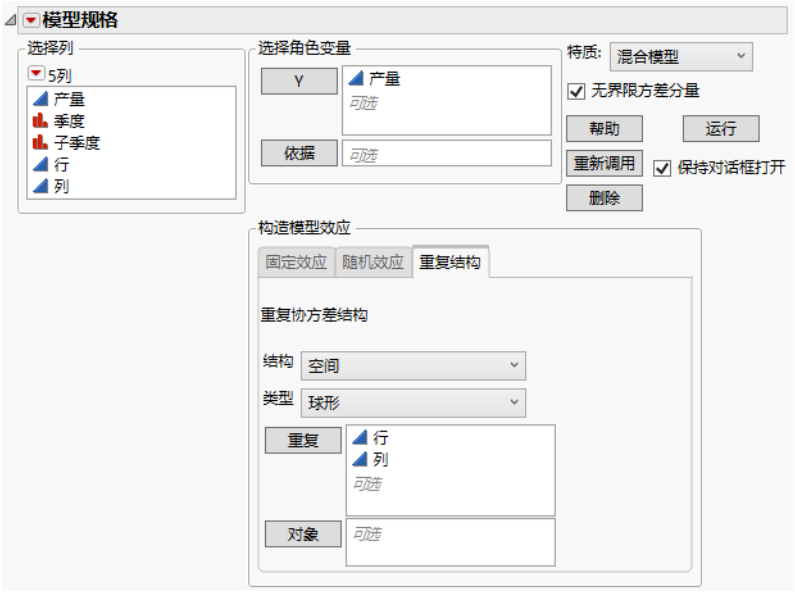
JMP PRO 拟合空间结构模型

要确定是否有显著的空间变异性，您可以拟合考虑空间变异性的模型。然后将该空间模型的似然与不考虑空间变异性的模型的似然进行比较。您可以这样做是因为独立的误差模型嵌套在空间模型系列中：独立误差模型是空间相关性 ρ 等于 0 的空间模型。这意味着您可以对这两个模型执行正式的似然比检验。

首先，拟合考虑空间结构的模型。

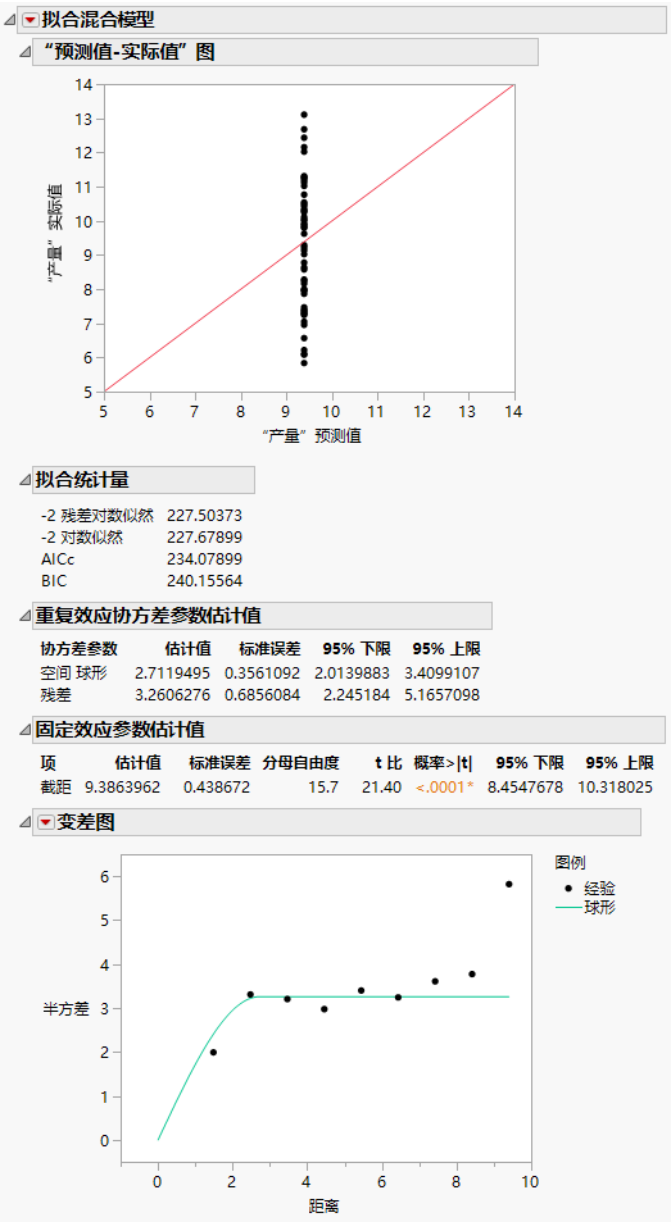
1. 选择**帮助** > **样本数据库**，然后打开 Uniformity Trial.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**保持对话框打开**，以便您可以在下一示例中返回到启动窗口。
4. 选择**产量**并点击 **Y**。
5. 从“特质”列表中选择**混合模型**。
6. 选择**重复结构**选项卡。
7. 从“结构”旁边的列表中选择**空间**。
8. 从“类型”旁边的列表中选择**球形**。
9. 选择**行**和**列**，然后点击**重复**。

图 8.32 显示 “重复结构”选项卡的完成的 “拟合模型”启动窗口



10. 点击运行。

图 8.33 “拟合混合模型”报表 - 空间球形



在“拟合混合模型”报表中，“预测值-实际值”图显示预测产量为单个值。这是因为只拟合了空间协方差。“拟合统计量”报表显示“-2 对数似然”为 227.68，“AICc”为 234.08。

因为拟合了各向同性空间结构，将显示“变差图”。因为是在 8x8 网格设计这些试验的，距离近的地方将比距离很远的地方有更多对点。请参见图 8.37 以了解布局。“变差图”显示球形空间结构对于最大约为 8.4 的距离是极好的拟合。最终距离的距离分类只包括两对对角线点。

“重复效应协方差参数估计值”报表给出极差的估计值（空间球形 = 2.71）和 sill 的估计值（残差 = 3.26）。请参见第 372 页的“变差图”。

JMP PRO 拟合独立误差模型

接下来，拟合独立误差模型。

1. 返回到“拟合模型”启动窗口。
2. 选择**重复结构**选项卡。
3. 从“结构”列表中选择**残差**。
4. 从“重复”效应列表中删除**行**和**列**。

   否则将显示一条警告：“当选择残差协方差结构时忽略重复列和对象列。”
5. 点击**运行**。

   独立误差模型的拟合统计量为：-2 对数似然 = 254.22， AICc = 258.41。这两个值分别超过了空间相关性模型的相应值，后者的“-2 对数似然”为 227.68，“AICc”为 234.08。因为这些统计量的值越小说明拟合效果越好，因此空间模型可能提供更好的拟合。

JMP PRO 执行似然比检验（可选）

正式的似然比检验说明空间相关性模型是否解释了显著变异。一个模型必须嵌套在另一模型中才能创建有效的似然比检验。

通常，使用 AICc 或 BIC 而非正式似然比检验来比较空间模型。计算 AICc 或 BIC 的值更快，并且很多空间模型不嵌套。

您可以在本例中执行似然比检验，因为独立误差模型嵌套在空间模型系列中。独立误差模型是空间相关性 ρ 等于 0 的空间模型。这意味着您可以对这两个模型执行正式的似然比检验。

在本例中，似然比检验统计量为 254.22-227.68=26.54。将它与一个自由度的卡方分布进行比较，当 p 值 < 0.0001 时拒绝无空间相关性这个原假设。您可以得出这些数据包含显著空间变异性的结论。

JMP PRO 选择空间协方差的类型

接下来，您确定哪个空间协方差结构最好地拟合数据：

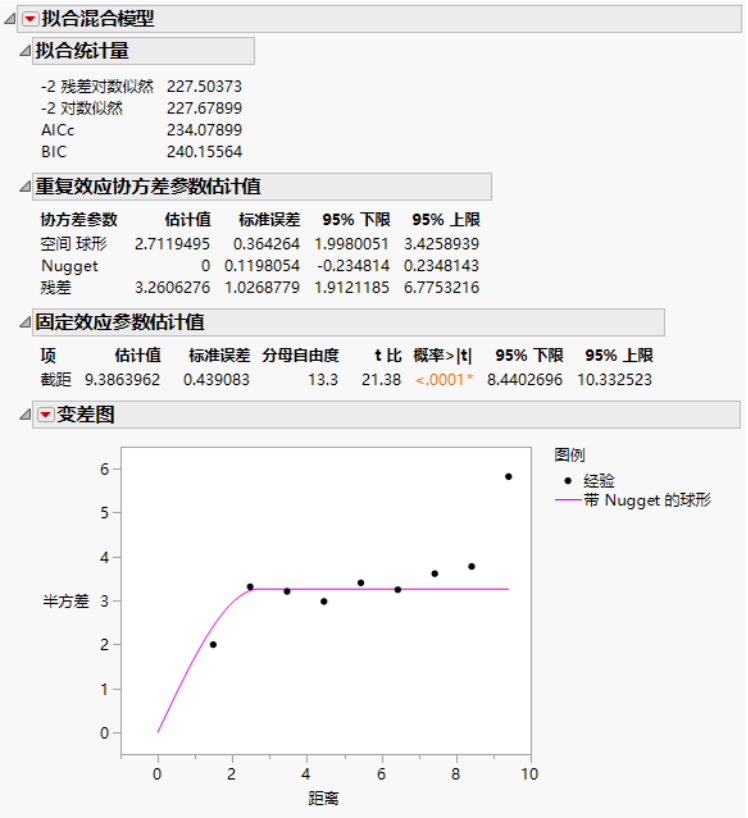
- 带或不带 nugget 效应（相对小距离上的变异）
- 各向同性（所有方向上的空间相关性相等）或各向异性（两个方向上的空间相关性存在差异）
- 结构的类型：球形、高斯、指数或乘方。

1. 返回到“拟合模型”启动窗口。
2. 选择**重复结构**选项卡。

3. 选择**行**和**列**，然后点击**重复**。
4. 从“结构”列表中选择**带 Nugget 的空间**。
5. 从“类型”列表中选择**球形**。
6. 点击**运行**。

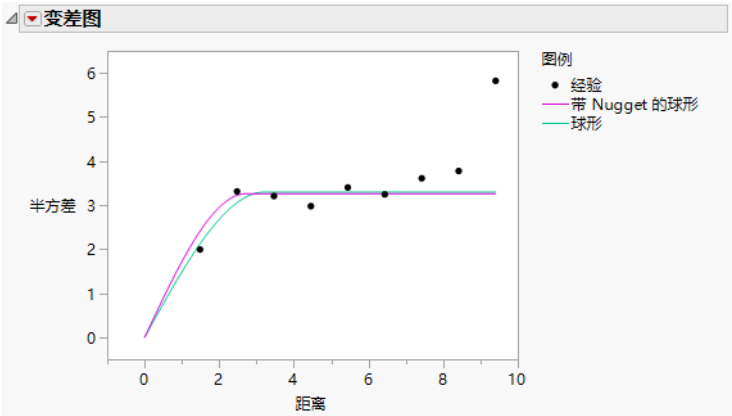
   “拟合混合模型”报表显示在图 8.34 中。请注意，该对数似然基本上等于不带 nugget 的球形模型的对数似然值， AICc 略高一点（236.36 对比 234.08）。“重复效应协方差参数估计值”报表显示 Nugget 协方差参数的估计值为零。这说明不存在任何 nugget 效应。

图 8.34 “拟合混合模型”报表 - 带 Nugget 的空间球形



7. 点击“变差图”红色小三角并选择**空间 > 球形**。

图 8.35 “拟合混合模型”报表 - 变差图



这两个变差图实际上是相同的。这也暗示不存在 nugget 效应。

8. 返回到 “拟合模型”启动窗口。
9. 选择**重复结构**选项卡。
10. 要检验各向异性，请从 “结构”列表中选择**空间各向异性**。
11. 从 “类型”列表中选择**球形**。
12. 点击**运行**。

“拟合混合模型”报表显示在图 8.36 中。拟合统计量指示该模型的拟合效果没有各向同性（空间结构）球形模型的拟合效果好（AICc 240.54 对比 234.08）。“重复效应协方差参数估计值”报表显示 “行”（空间球形行）和 “列”（空间球形列）协方差的估计值很接近。没有证据表明网格的行内和列内的空间相关性存在差异。

图 8.36 “拟合混合模型”报表 - 空间各向异性球形

拟合混合模型

拟合统计量

-2 残差对数似然	232.09009
-2 对数似然	231.86638
AICc	240.54434
BIC	248.50191

重复效应协方差参数估计值

协方差参数	估计值	标准误差	95% 下限	95% 上限
空间 球形 行	2.2453516	0.4114883	1.4388492	3.0518539
空间 球形 列	2.2285839	0.3316821	1.5784989	2.8786689
残差	3.0050408	0.6112702	2.0921756	4.6820483

固定效应参数估计值

项	估计值	标准误差	分母自由度	t 比	概率>|t|	95% 下限	95% 上限
截距	9.4039776	0.3585807	21.3	26.23	<.0001*	8.658977	10.148978

JMP PRO 确定空间结构的类型

不带 nugget 的各向同性空间结构是合适的结构。为了确定空间结构的类型，可以比较可用类型。

1. 返回到“拟合模型”启动窗口。
2. 选择**重复结构**选项卡。
3. 选择**行**和**列**，然后点击**重复**。
4. 从“结构”列表中选择**空间**。
5. 从“类型”列表中选择**乘方**。
6. 点击**运行**。

   请注意模型拟合的 AICc 和 BIC 值。
7. 返回到“拟合模型”启动窗口。
8. 从“类型”列表中选择**指数**。
9. 点击**运行**。

   请注意模型拟合的 AICc 和 BIC 值。
10. 返回到“拟合模型”启动窗口。
11. 从“类型”列表中选择**高斯**。
12. 点击**运行**。

    请注意模型拟合的 AICc 和 BIC 值。

表 8.2 汇总了这些类型以及您执行的其他拟合的观测 AICc 值。

表 8.2 空间模型拟合的拟合统计量

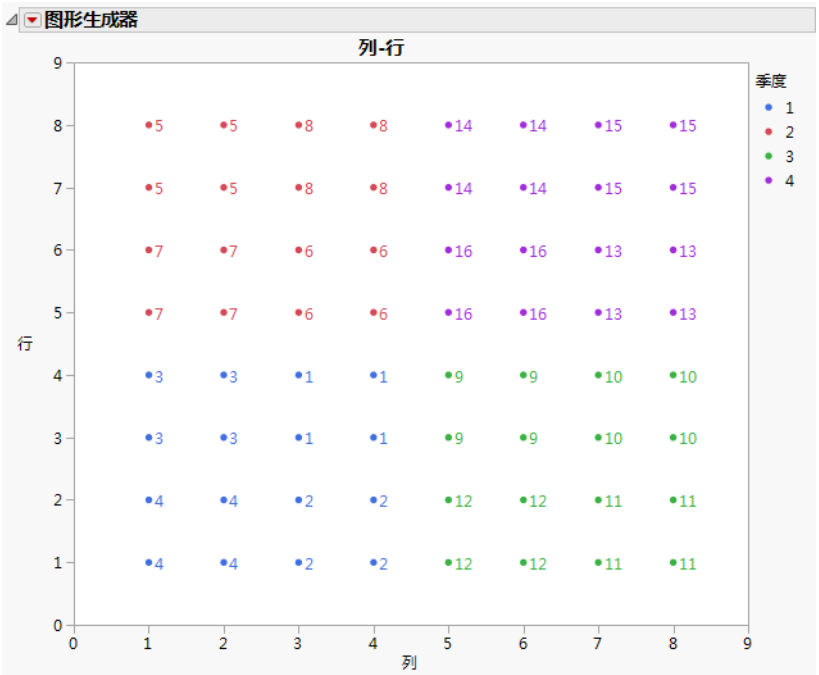
结构	类型	AICc	BIC
空间	球形	234.08	240.16
残差		258.41	262.53
带 Nugget 的空间	球形	236.36	244.31
空间各向异性	球形	240.54	248.50
空间	乘方	240.24	246.32
空间	指数	240.24	246.32
空间	高斯	238.37	244.44

最佳拟合模型（AICc 值最小）是具有“球形”协方差结构的“空间”结构模型。现在您要将该模型与完整和不完整的区组模型进行比较，才能实现均匀性试验的目标。

JMP PRO 将模型与区组设计比较

1. 返回到 Uniformity Trial 数据表。
2. 在 “表”面板中，运行 “图形生成器”脚本。

图 8.37 建议的完整和不完整区组设计的图形生成器图



该图显示为田地建议的完整和不完整区组设计。颜色指示将作为完整区组的季度田地。编号的点表示将作为不完整区组的子季度田地。

要拟合完整区组模型，请执行以下步骤：

1. 返回到“拟合模型”启动窗口。
2. 选择**重复结构**选项卡。
3. 从 “结构”列表中选择**残差**。
4. 从效应中删除**行**和**列**。

   否则将显示一个弹出窗口，它指出：“当选择残差协方差结构时忽略重复列和对象列。”
5. 选择**随机效应**选项卡。
6. 选择**季度**，然后点击**添加**。
7. 点击**运行**。

要拟合不完整区组模型，请执行以下步骤：

1. 返回到“拟合模型”启动窗口。
2. 选择**随机效应**选项卡。
3. 选择**季度**并点击**删除**。
4. 选择**子季度**并点击**添加**。
5. 点击**运行**。

以下列表显示竞争模型的 AICc 和 BIC 值。使用球形协方差结构可以得到最佳模型拟合效果。这指示在使用该田地的将来的研究中，首选具有空间相关误差的完全随机化的设计。

- 球形模型（请参见第 356 页的“确定空间结构的类型”）
  - AICc: 234.08
  - BICc: 240.16
- 完全区组 (RCBD) 模型
  - AICc: 259.90
  - BICc: 265.97
- 不完全区组模型
  - AICc: 248.77
  - BICc: 254.85

JMP PRO 相关响应示例

在该示例中，研究了与晶片生产有关的两个布局对所关注的特征研究的影响。将每 50 个晶片划分为四个象限，在这些象限中测量该特征。此类数据通常以以下格式提供：其中每行包含某个关注单元的所有重复测量值。通常使用每个响应的单独模型来分析此类数据。但是，在单个单元取重复测量值时，可能存在单元内相关性。不考虑该相关性将得不到好的决策和预测。您可以使用“混合模型”特质来考虑可能的相关性并对其建模。

要对重复的测量数据进行“混合模型”分析，每个重复的测量值需要位于自己的行中。若您的数据采用典型格式（所有重复测量值位于同一行中），可以通过使用“表”>“堆叠”为“混合模型”分析构造合适的数据表。请参见《使用 JMP》。

在该示例中，您首先使用通常的数据表格式拟合一元模型。然后使用 “混合模型”特质拟合四个响应的模型，同时考虑响应之间可能的相关性。

JMP PRO 拟合一元模型

使用 “标准最小二乘法”拟合四个象限的一元模型。

1. 选择**帮助 > 样本数据库**，然后打开 Wafer Quadrants.jmp。

   该数据表已针对 “混合模型”分析重组结构，每个**象限**上的每个 Y 测量值占据一行。为了构造一元分析，您使用保存的脚本拆分该表。

2. 在 Wafer Quadrants.jmp 数据表中，点击**按 “象限”拆分** Y 脚本旁边的绿色小三角。

   新数据表采用通常用于记录重复测量数据的格式。**晶片 ID** 的每个值定义一行，在单独的行中给出该晶片的四个测量值。

3. 选择**分析 > 拟合模型**。

   因为数据表中有 “模型”脚本，因此已填写 “模型规格”窗口。请注意，列 High, High 到 Low, Low 被输入为 Y，**布局**为单个模型效应。

4. 点击**运行**。

图 8.38 四个一元模型

报表指示**布局**对于除 “High, Low”象限之外的所有象限在统计上具有显著效应。

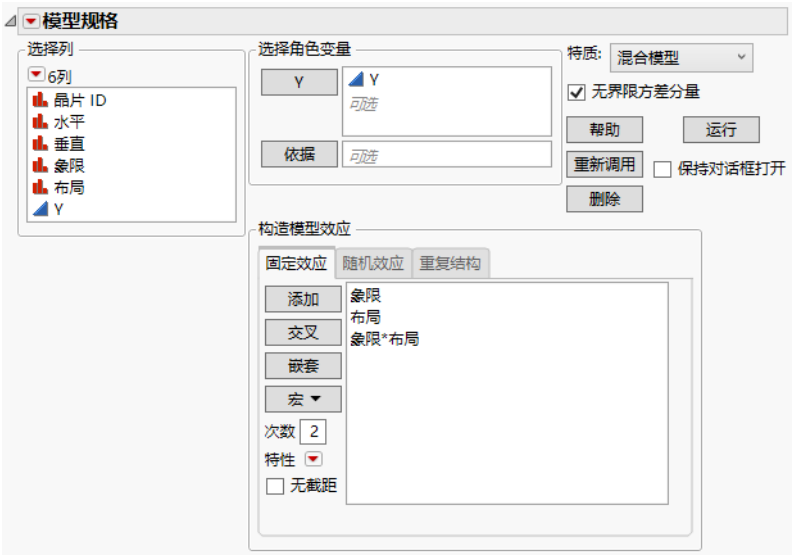
JMP PRO 执行“混合模型”分析

使用 “混合模型”分析，您可以获取更多信息。

1. 返回到 Wafer Quadrants.jmp。或者，若您关闭了它，请选择**帮助** > **样本数据库**，然后打开 Wafer Quadrants.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 Y 并点击 Y。
4. 从**特质**列表中选择**混合模型**。
5. 从**选择列**列表中选择**象限**和**布局**，然后点击**宏** > **完全析因**。

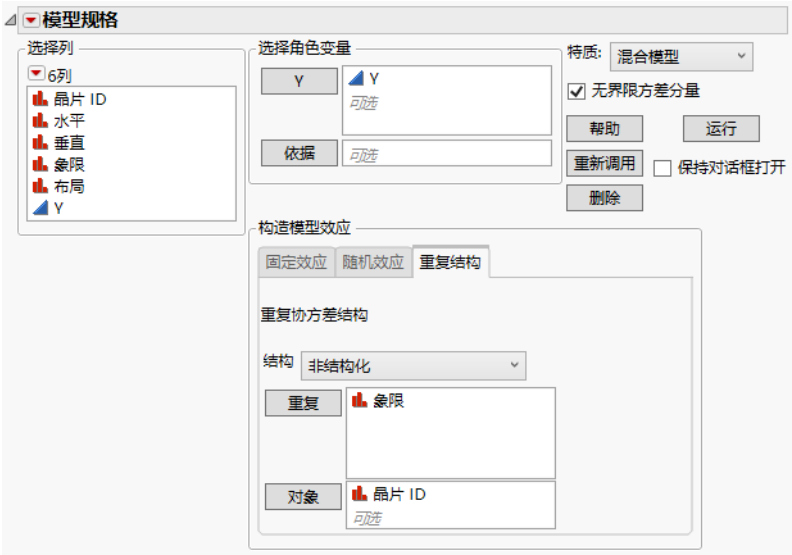
   该模型规格允许您探索**布局**对重复测量值的效应以及可能的**布局**与**象限**之间的交互作用。

图 8.39 显示完成的 “固定效应”选项卡的 “拟合模型”启动窗口



6. 选择**重复结构**选项卡。
7. 从 “结构”列表中选择**非结构化**。
8. 选择**象限**并点击**重复**。
9. 选择**晶片 ID** 并点击**对象**。

图 8.40 显示 “重复结构”选项卡的 “拟合模型”启动窗口



10. 点击**运行**。

图 8.41 关闭了 “固定效应参数估计值”报表的 “拟合混合模型”报表

拟合混合模型

重复效应协方差参数估计值

重复效应: 象限
对象: 晶片 ID

协方差参数	估计值	标准误差	95% 下限	95% 上限
Var(High, High)	19.835702	4.0489457	11.899914	27.77149
Cov(High, Low,High, High)	0.0111321	2.5624804	-5.011237	5.0335013
Var(High, Low)	15.889659	3.2434631	9.5325884	22.24673
Cov(Low, High,High, High)	15.893138	6.3915311	3.3659676	28.420309
Cov(Low, High,High, Low)	1.0055411	5.3413847	-9.463381	11.474463
Var(Low, High)	86.121899	17.579559	51.666597	120.5772
Cov(Low, Low,High, High)	14.723606	7.7996739	-0.563474	30.010686
Cov(Low, Low,High, Low)	8.0424897	6.8163252	-5.317262	21.402242
Cov(Low, Low,Low, High)	2.1512894	15.640276	-28.50309	32.805667
Var(Low, Low)	136.28412	27.81888	81.76012	190.80813

固定效应检验

源	参数数目	分子自由度	分母自由度	F 比	概率>F
象限	3	3	46.0	149.21352	<.0001*
布局	1	1	48.0	51.757797	<.0001*
象限*布局	3	3	46.0	22.379307	<.0001*

“重复效应协方差参数估计值”报表给出四个响应的估计方差和协方差。请注意，Low, High 与 High, High 的协方差的置信区间不包括零。这暗示这两个象限中的测量值之间的协方差为正。这是 “混合模型”分析使用的信息，当独立对响应建模时，该信息不可用。

“固定效应检验”报表指示**布局** * **特征**交互作用是显著的。

JMP PRO 使用刻画器探索“布局 * 特征”交互作用

1. 点击“拟合混合模型”红色小三角菜单，然后选择**边缘模型推断** > **刻画器**。
2. 在“刻画器”图中，通过首先在“Layout A”处设置红色的垂直虚线，然后在“Layout B”处设置该线，比较所有象限上 Y 的预测值。

图 8.42 “Layout A”的象限的刻画

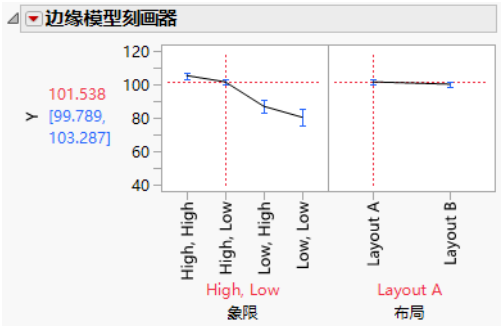
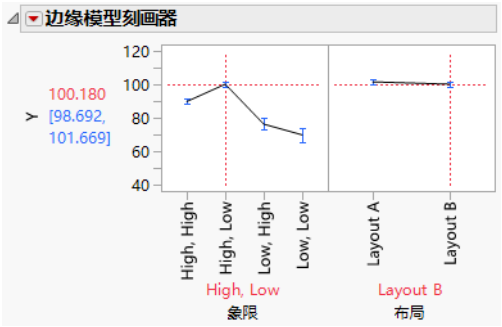


图 8.43 “Layout B”的象限的刻画



通过**布局**的每个设置处的刻画差值，您可以知道存在显著的交互作用。似乎该交互作用部分来自于“High, High”象限的差值。

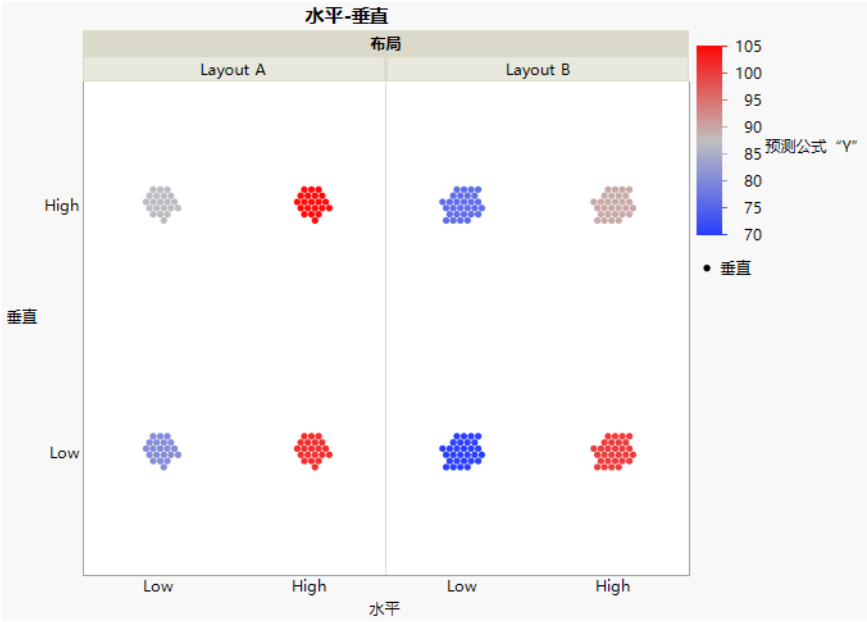
JMP PRO “布局 -Y”和“象限 -Y”图

使用图形生成器探索交互作用的性质。

1. 点击“拟合混合模型”红色小三角菜单，然后选择**保存列** > **预测公式**。
   将预测公式保存到数据表的**预测公式** Y 列中。
2. 选择**图形** > **图形生成器**。
3. 将**水平**拖到 X 区域。
4. 将**垂直**拖到 Y 区域。

5. 将**预测公式 Y** 拖到**颜色**区域。
6. 将**布局**拖到**重叠**区域。
7. 点击**完成**以隐藏控制面板。

图 8.44 已完成的图形生成器图



注意：因为图形生成器中的点是随机散布的，您的图可能与图 8.44 不完全相符。

该图使用色彩强度尺度显示八个**布局**和**象限**组合的预测差值。“High, Low”象限的预测值位于右下方。颜色梯度显示这些预测值的差值相对较小。其他差值被清楚地指示。

JMP PRO “混合模型”特质的统计详细信息

- “收敛得分检验”
- “随机系数模型”
- “重复测量”
- “重复的协方差结构”
- “空间和时间变异性”
- “Kackar-Harville 校正”

JMP PRO 收敛得分检验

收敛失败警告显示以下假设的得分检验：未知最大似然估计值 (MLE) 与模型拟合算法的最后一次迭代中给出的参数一致。该假设检验是可能实现的，因为相对梯度准则在代数上等价于得分检验统计量。显然得分检验不需要知道真实 MLE。

JMP PRO 得分检验

首先考虑单个参数 θ 的情况。假定 l 是 θ 的对数似然函数，**x** 是数据。得分是对数似然函数对 θ 的导数：

$$U(\theta) = \frac{\partial l(\theta|\mathbf{x})}{\partial \theta}$$

观测信息是：

$$I(\theta) = -\frac{\partial^2}{\partial \theta^2} l(\theta|\mathbf{x})$$

$H_0: \theta = \theta_0$ 的得分检验的统计量为：

$$S(\theta_0) = \frac{U(\theta_0)^2}{I(\theta_0)}$$

该统计量在原假设下服从自由度为 1 的渐近卡方分布。

得分检验可以推广到多个参数。考虑参数 θ 的向量。那么 $H_0: \theta = \theta_0$ 的得分检验的检验统计量为：

$$S(\theta_0) = \mathbf{U}'(\theta_0)\mathbf{I}^{-1}(\theta_0)\mathbf{U}(\theta_0)$$

其中

$$\mathbf{U}(\theta) = \frac{\partial l(\theta|\mathbf{x})}{\partial \theta}$$

且

$$\mathbf{I}(\theta) = -\frac{\partial^2 l(\theta|\mathbf{x})}{\partial \theta(\partial \theta')}$$

$\mathbf{U}'$ 表示矩阵 **U** 的转置矩阵。

检验统计量服从自由度为 k 的渐近卡方分布。其中 k 是无界参数的数目。

JMP PRO 相对梯度

“混合模型”拟合过程的收敛准则基于相对梯度 $g'\mathbf{H}^{-1}g$。在此，$g(\theta) = \mathbf{U}(\theta)$ 是对数似然函数的梯度，$\mathbf{H}(\theta) = -\mathbf{I}(\theta)$ 是它的 Hessian。

用 θ_0 表示算法终止时的 θ 值。请注意，在 θ_0 处计算出的相对梯度为得分检验统计量。使用具有 k 个自由度的卡方分布计算 p 值。该 p 值指示未知的 MLE 值是否与 θ_0 一致。“随机效应协方差参数估计值”报表中列出的无界参数数目等于 k。

JMP PRO 随机系数模型

标准随机系数模型指定每个对象的随机截距和斜率。用 y_{ij} 表示第 i 个对象的第 j 个观测的测量值。这样，随机系数模型可以指定如下：

$$y_{ij} = a_i + b_i x_{ij} + e_{ij}$$

其中

$$i = 1, 2, \ldots, t$$

$$j = 1, 2, \ldots, n_i$$

$$\begin{bmatrix} a_i \\ b_i \end{bmatrix} \sim iid\ N\left(\begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \mathbf{G} \right)$$

$$\mathbf{G} = \begin{bmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{bmatrix}$$

且

$$e_{ij} \sim iid\ N(0, \sigma^2)$$

您可以重构模型来反映估计的固定成分和随机成分：

$$y_{ij} = (\alpha + a_i^*) + (\beta + b_i^*) x_{ij} + e_{ij}$$

其中

$$a_i^* = \alpha_i - \alpha$$

$$b_i^* = \beta_i - \beta$$

且

$$\begin{bmatrix} a_i^* \\ b_i^* \end{bmatrix} \sim iid\ N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \mathbf{G}\right)$$

按上文所述定义 **G** 和 e_{ij}。

JMP PRO 重复测量

重复测量模型采用以下形式：$y_{ijk} = \alpha_{ij} + s_{ik} + e_{ijk}$，其中

α_{ij} 可以表示为处理和时间析因

s_{ik} 是分配给第 i 个处理的第 k 个对象的随机效应

$j = 1,\ldots,m$ 表示随时间变化的重复测量值。

假定 s_{ik} 是独立同分布 $N(0, \sigma_s^2)$ 的变量。用 t 表示处理因子数，用 s 表示对象数。则 e_{ijk} 的分布为 $N(0, \Sigma)$，其中

$$\Sigma = \mathbf{I}_{ts} \otimes Var(\mathbf{y}_{ik} | s_{ik})$$

且

$$\mathbf{y}'_{ik} | s_{ik} = \left(\left[y_{i1k}\ y_{i2k}\ \ldots\ y_{imk} \right] \middle| s_{ik} \right)$$

用 Σ_{ik} 表示对应于处理内第 ik 个对象的协方差矩阵 Σ 的区组对角成分。换句话说，$\Sigma_{ik} = Var(\mathbf{y}_{ik} | s_{ik})$。因为对象内随时间变化的观测值通常不是独立的，有必要估计 $y_{ijk} | s_{ik}$ 的方差。未考虑相关性将导致推断失真。

有关可用于 Σ_{ik} 的协方差结构的详细信息，请参见第 366 页的“重复的协方差结构”和第 371 页的“空间和时间变异性”。

JMP PRO 重复的协方差结构

本节给出以下协方差结构的参数化：

- 第 367 页的“不等方差协方差结构”
- 第 367 页的“非结构化协方差结构”
- 第 367 页的“复合对称性协方差结构”
- 第 369 页的“AR(1) 协方差结构”
- 第 370 页的“Toeplitz 协方差结构”
- 第 370 页的“事前相关协方差结构”

JMP PRO 不等方差协方差结构

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0 & 0 & \dots & 0 \\ & \sigma_2^2 & 0 & \dots & 0 \\ & & \sigma_3^2 & \dots & 0 \\ & & & \ddots & \vdots \\ & & & & \sigma_m^2 \end{bmatrix}$$

在此，在时间 j 获取的观测值之间的方差为：

$$\sigma_j^2 = Var(y_{ijk} | s_{ik})$$

允许重复列各水平的方差有所不同。不同水平的观测之间的协方差为零。

JMP PRO 非结构化协方差结构

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} & \dots & \sigma_{1[m-1]} & \sigma_{1m} \\ & \sigma_2^2 & \sigma_{23} & \dots & \sigma_{2[m-1]} & \sigma_{2m} \\ & & \sigma_3^2 & \dots & \sigma_{3[m-1]} & \sigma_{3m} \\ & & & \ddots & \vdots & \vdots \\ & & & & \sigma_{m-1}^2 & \sigma_{[m-1]m} \\ & & & & & \sigma_m^2 \end{bmatrix}$$

在此，在时间 j 获取的观测值之间的方差为：

$$\sigma_j^2 = Var(y_{ijk} | s_{ik})$$

在时间 j 和 j' 获取的观测值之间的协方差为：

$$\sigma_{jj'} = Cov(y_{ijk}, y_{ij'k} | s_{ik})$$

允许重复列各水平的方差有所不同。不同水平的观测之间的协方差是唯一的。

JMP PRO 复合对称性协方差结构

在 JMP 中使用具有独立误差的混合模型方法来实现复合对称性协方差结构。随机效应分为两个类别：G 侧或 R 侧。详细信息，请参见 Searle et al.(1992)。

G 侧随机效应与随机效应的设计矩阵关联。 R 侧随机效应与剩余误差关联。对象内方差是设计结构的一部分，在 G 侧进行建模。对象间方差属于残差结构，在 R 侧进行建模。独立结构中的方差按如下方式建模：

- 通过 $s_{ik} \sim \text{iid } N(0, \sigma_s^2)$ 对随机效应 G 侧方差建模。
- 通过 $e_{ijk} \sim \text{iid } N(0, \sigma^2)$ 对 R 侧方差建模。

则协方差矩阵定义如下：

$$\Sigma_{ik} = \sigma_s^2\mathbf{J} + \sigma^2\mathbf{I} = \begin{bmatrix} \sigma_s^2+\sigma^2 & \sigma_s^2 & \cdots & \sigma_s^2 \\ & \sigma_s^2+\sigma^2 & \cdots & \sigma_s^2 \\ & & \ddots & \vdots \\ & & & \sigma_s^2+\sigma^2 \end{bmatrix}$$

其中 **J** 是包含 *1* 的矩阵， **I** 是单位矩阵。

或者，可以在 R 侧对所有方差建模。在高斯假定下，该复合对称性协方差结构等价于独立模型（SAS 中的 Type=CS ）。在 JMP 中，通过使用“重复结构”选项卡中的“复合对称性”结构来提供该结构。在这，重复观测对之间的相关性相同，而与观测之间的时间差值无关。因此，相关性矩阵可以指定如下：

$$C = \begin{bmatrix} 1 & \rho & \cdots & \rho \\ & 1 & \cdots & \rho \\ & & \ddots & \vdots \\ & & & 1 \end{bmatrix}$$

在 JMP 中使用“复合对称性”结构也假定对于任何时间点取的观测之间存在公共方差 σ_e^2 。则协方差结构为 $\Sigma = \sigma_e^2 C$，其中

$$\sigma_e^2 = \sigma_s^2 + \sigma^2$$

且

$$\rho = \frac{\sigma_s^2}{\sigma_s^2+\sigma^2}.$$

在这里，ρ 是组内相关系数，σ_e^2 为残差方差。另一选项是在 JMP 中使用 “复合对称性不等方差”结构，该结构允许方差随时间点变化。这导致以下协方差矩阵：

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 & \rho\sigma_1\sigma_3 & \cdots & \rho\sigma_1\sigma_{m-1} & \rho\sigma_1\sigma_m \\ & \sigma_2^2 & \rho\sigma_2\sigma_3 & \cdots & \rho\sigma_2\sigma_{m-1} & \rho\sigma_2\sigma_m \\ & & \sigma_3^2 & \cdots & \rho\sigma_3\sigma_{m-1} & \rho\sigma_3\sigma_m \\ & & & \ddots & \vdots & \vdots \\ & & & & \sigma_{m-1}^2 & \rho\sigma_{m-1}\sigma_m \\ & & & & & \sigma_m^2 \end{bmatrix}$$

JMP PRO AR(1) 协方差结构

$$\Sigma = \begin{bmatrix} \sigma^2 & \rho^{|t_1-t_2|}\sigma^2 & \rho^{|t_1-t_3|}\sigma^2 & \cdots & \rho^{|t_1-t_{m-1}|}\sigma^2 & \rho^{|t_1-t_m|}\sigma^2 \\ & \sigma^2 & \rho^{|t_2-t_3|}\sigma^2 & \cdots & \rho^{|t_2-t_{m-1}|}\sigma^2 & \rho^{|t_2-t_m|}\sigma^2 \\ & & \sigma^2 & \cdots & \rho^{|t_3-t_{m-1}|}\sigma^2 & \rho^{|t_3-t_m|}\sigma^2 \\ & & & \ddots & \vdots & \vdots \\ & & & & \sigma^2 & \rho^{|t_{m-1}-t_m|}\sigma^2 \\ & & & & & \sigma^2 \end{bmatrix}$$

在此，t_j 是观测值 j 的时间。在该结构中，在任意给定时间取的观测具有相同方差 σ^2。参数 ρ（其中 $-1 < \rho < 1$）是相隔一个时间单元的两个观测之间的相关性。随着观测值之间的时间差异增大，其协方差会下降，因为 ρ 上升到更高次幂。在很多应用中，AR(1) 提供对象内相关性的适当模型，提供更高功效而不会降低对第一类错误的控制。

JMP PRO Toeplitz 协方差结构

在 Toeplitz 结构中，用固定的时间单位数分隔的观测具有相同的相关性。对比 AR(1) 相关性结构，固定时间差值处的 Toeplitz 相关性是任意的。用 ρ_d 表示相隔 d 个单元的观测的相关性。相关性矩阵定义如下：

$$C = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{m-1} & \rho_m \\ & 1 & \rho_1 & \cdots & \rho_{m-2} & \rho_{m-1} \\ & & 1 & \cdots & \rho_{m-3} & \rho_{m-2} \\ & & & \ddots & \vdots & \vdots \\ & & & & 1 & \rho_1 \\ & & & & & 1 \end{bmatrix}$$

JMP 中的两个选项使用该相关性矩阵：

- Toeplitz 结构对于来自任意时间点的观测假定有公共方差 σ^2。协方差结构为 $\Sigma = \sigma^2 C$。
- 或者，“Toeplitz 不等方差”结构允许方差随时间点变化：

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho_1\sigma_1\sigma_2 & \rho_2\sigma_1\sigma_3 & \cdots & \rho_{m-1}\sigma_1\sigma_{m-1} & \rho_m\sigma_1\sigma_m \\ & \sigma_2^2 & \rho_1\sigma_2\sigma_3 & \cdots & \rho_{m-2}\sigma_2\sigma_{m-1} & \rho_{m-1}\sigma_2\sigma_m \\ & & \sigma_3^2 & \cdots & \rho_{m-3}\sigma_3\sigma_{m-1} & \rho_{m-2}\sigma_3\sigma_m \\ & & & \ddots & \vdots & \vdots \\ & & & & \sigma_{m-1}^2 & \rho_1\sigma_{m-1}\sigma_m \\ & & & & & \sigma_m^2 \end{bmatrix}$$

JMP PRO 事前相关协方差结构

事前相关模型是一种灵活的常规模型，允许相关性结构随时间变化。在该模型中，相邻时间点 j - 1 和 j 处的两个观测之间的相关性是唯一的，将其表示为 $\rho_{j[j-1]}$。

非相邻时间点 j 和 j' 处的观测对之间的相关性为这两点之间所有相邻相关性之积。

$$Corr(y_{ijk}, y_{ij'k} | s_{ik}) = \prod_{k=j}^{j'-1} \rho_{k[k-1]}$$

例如，时间点 j=2 和 j'=6 处的观测对之间的相关性将为 $\rho_{21}\rho_{32}\rho_{43}\rho_{54}$。

相关性矩阵定义如下：

$$C = \begin{bmatrix} 1 & \rho_{10} & \rho_{10}\rho_{21} & \cdots & \rho_{10}\cdots\rho_{[m-1][m-2]} & \rho_{10}\cdots\rho_{m[m-1]} \\ & 1 & \rho_{21} & \cdots & \rho_{21}\cdots\rho_{[m-1][m-2]} & \rho_{21}\cdots\rho_{m[m-1]} \\ & & 1 & \cdots & \rho_{32}\cdots\rho_{[m-1][m-2]} & \rho_{32}\cdots\rho_{m[m-1]} \\ & & & \ddots & \vdots & \vdots \\ & & & & 1 & \rho_{m[m-1]} \\ & & & & & 1 \end{bmatrix}$$

JMP 中的两个选项使用该相关性矩阵：

- “事前相关相等方差”结构假定观测时间上的方差相等，但仍允许相关性有变化。任意时间处的观测之间的方差为 σ^2，协方差矩阵为 $\Sigma = \sigma^2 C$。
- 事前相关结构允许任何给定时间处的观测之间的方差各不相同。将时间 j 处的观测之间的方差表示为 σ_j^2，则协方差矩阵定义如下：

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho_{10}\sigma_1\sigma_2 & \rho_{10}\rho_{21}\sigma_1\sigma_3 & \cdots & \rho_{10}\cdots\rho_{[m-1][m-2]}\sigma_1\sigma_{m-1} & \rho_{10}\cdots\rho_{m[m-1]}\sigma_1\sigma_m \\ & \sigma_2^2 & \rho_{21}\sigma_2\sigma_3 & \cdots & \rho_{21}\cdots\rho_{[m-1][m-2]}\sigma_2\sigma_{m-1} & \rho_{21}\cdots\rho_{m[m-1]}\sigma_2\sigma_m \\ & & \sigma_3^2 & \cdots & \rho_{32}\cdots\rho_{[m-1][m-2]}\sigma_3\sigma_{m-1} & \rho_{32}\cdots\rho_{m[m-1]}\sigma_3\sigma_m \\ & & & \ddots & \vdots & \vdots \\ & & & & \sigma_{m-1}^2 & \rho_{m[m-1]}\sigma_{m-1}\sigma_m \\ & & & & & \sigma_m^2 \end{bmatrix}$$

JMP PRO 空间和时间变异性

考虑简单模型 $y_i = \mu + e_i$。通过误差项 e_i 对空间或时间结构建模。通常，空间相关性模型可以定义为 $Var(e_i) = \sigma_i^2$ 且 $Cov(e_i, e_j) = \sigma_{ij}$。

用 $\mathbf{s}_i$ 表示 y_i 的位置，其中通过反映空间或时间的坐标指定 $\mathbf{s}_i$。通常通过假定协方差是欧几里得距离 d_{ij}（$\mathbf{s}_i$ 和 $\mathbf{s}_j$ 之间）的函数，限制空间或时间结构。该协方差可以表示为 $Cov(e_i, e_j) = \sigma^2[f(d_{ij})]$，其中 $f(d_{ij})$ 表示观测值 y_i 和 y_j 之间的相关性。

在两个或更多位置坐标的情况下，若 $f(d_{ij})$ 不依赖于方向，则协方差结构为**各向同性**。若依赖于方向，则该结构为**各向异性**。

JMP PRO 空间相关性结构

下面显示 JMP 中可用的空间模型的相关性结构。这些结构由参数 ρ 表示，除非另行约束，否则该参数值为正数。

- 球形

$$f(d_{ij}) = [1 - 1.5(d_{ij}/\rho) + 0.5(d_{ij}/\rho)^3] \times 1_{\{d_{ij} < \rho\}}$$

其中，$1_{\{d_{ij} < \rho\}} = \begin{cases} 1, & \text{若 } d_{ij} < \rho \\ 0, & \text{若 } d_{ij} \geq \rho \end{cases}$

- 指数

$$f(d_{ij}) = \exp(-d_{ij}/\rho)$$

- 高斯

$$f(d_{ij}) = \exp(-d_{ij}^2/\rho^2)$$

- 功效

$$f(d_{ij}) = \rho^{d_{ij}}$$

对于各向异性模型，相关性函数针对每个方向包含一个参数 ρ_κ。

JMP PRO 变差图

当空间过程为二阶平稳时，第 372 页的“空间相关性结构”中列出的结构定义**变差图**。从地质统计学借鉴而来，变差图是用于描述和估计空间变异性的标准工具。它使用半方差测量作为两个观测值之间的距离 d_{ij} 的函数的空间变异性。

用 $Z(\mathbf{s})$ 表示位置 $\mathbf{s}$ 处的响应值。按以下方式定义 $\mathbf{s}_i$ 和 $\mathbf{s}_j$ 处的两个观测值之间的**半方差**：

$$\gamma(\mathbf{s}_i, \mathbf{s}_j) = (Var(Z(\mathbf{s}_i) - Z(\mathbf{s}_j)))/2$$

若响应具有常数均值，则表达式可以简化为：

$$\gamma(\mathbf{s}_i, \mathbf{s}_j) = E[(Z(\mathbf{s}_i) - Z(\mathbf{s}_j))^2]/2$$

若过程是各向同性的，则半方差仅依赖于点之间的距离 h 且函数可以指定如下：

$$\gamma(h) = E[(Z(\mathbf{s}_i) - Z((\mathbf{s}_i + h)))^2]/2$$

以下项与变差图关联：

Nugget 定义为截距。它表示 $h = 0$ 处的跳跃不连续性。

Sill 定义为在达到更大距离的平稳值处的半变差图的值。它对应于观测值的方差。在不带 nugget 效应的模型中，sill 为 σ^2。在带 nugget 效应的模型中，sill 为 $\sigma^2 + c_1$，其中 c_1 表示 nugget。偏 sill 定义为 σ^2。

极差 定义为半变差图达到 sill 处的距离。在小于极差的距离处，观测值在空间上是相关的。对于大于等于极差的距离，空间相关性实际上为零。在球形模型中，ρ 为极差。在指数模型中，3ρ 为实际极差。在高斯模型中，$\rho\sqrt{3}$ 为实际极差。实际极差定义为协方差下降到 95% 的 sill 处的距离。

在第 354 页上的图 8.34 中，重复效应协方差参数估计值表示各种半变差图功能：

空间球形 极差 ρ 的估计值。

Nugget c_1 的统一尺度的估计值。残差与 Nugget 之积为 c_1。

残差 偏 sill 或无 nugget 模型中的 sill。

JMP PRO 变差图估计值

对于给定的各向同性空间结构，使用非线性最小二乘法将观测数据拟合为第 372 页的“空间相关性结构”中的合适函数，以得到估计的变差图。

JMP PRO 经验半方差

要计算**经验半方差**，需要计算为变差图协方差选择的变量的所有点对之间的距离。距离的极差分为 10 个相等的区间。若数据不允许 10 个区间，则构造尽可能多的区间。

构造包含点对的距离分类。第 h 个距离分类包含其距离位于第 h 个区间的所有点对。

考虑以下符号：

n 点对总数

C_h 包含其距离位于第 h 个最大区间的点的距离分类

$Z(\mathbf{x})$ $\mathbf{x}$ 处的响应值，其中 $\mathbf{x}$ 是时间或空间坐标的向量

$\gamma(h)$ 距离分类 C_h 的半方差

按以下方式定义半方差函数 γ：

$$\gamma(h) = \begin{cases} \dfrac{1}{2n}\left\{ \displaystyle\sum_{(x,y)\ \text{in}\ C_h} [Z(\mathbf{x}) - Z(\mathbf{y})]^2 \right\}, & \text{对于 } h > 0 \\ \hat{c}_1 & \text{对于 } h = 0 \end{cases}$$

在此 $\hat{c}_1$ 是 nugget 效应的估计值。

JMP PRO Kackar-Harville 校正

固定效应的方差矩阵始终修改为包括 Kackar-Harville 校正。BLUP 的方差矩阵以及 BLUP 与固定效应之间的协方差都不做 Kackar-Harville 校正。这种方法的原理是：当随机效应包含许多水平时，BLUP 校正可能占用较多的计算和内存资源。在 SAS 中，仅当设置了 DDFM=KENWARDROGER 时，才为固定效应和 BLUP 执行 Kackar-Harville 校正。

对于具有相对协方差参数的非零二阶导数的协方差结构，Kenward-Roger 协方差矩阵调整包括一个二阶项。该项会导致标准误差收缩。此外，得到的调整协方差矩阵可能是无限的，在重新参数化时会变化。一阶 Kenward- Roger 协方差矩阵调整消除了二阶导数的计算。所有空间结构和 AR(1) 结构是通常导致非零二阶导数的协方差结构。

因为 JMP 执行 Kenward-Roger 一阶调整

- 仅涉及固定效应参数的线性组合的标准误差与 PROC MIXED DDFM=KENWARDROGER(FIRSTORDER) 相符。这种情况假定用户已经在 PROC MIXED 和 JMP 使用的不同参数化之间谨慎变换。
- 仅涉及 BLUP 参数的线性组合的标准误差与 PROC MIXED DDFM=SATTERTHWAITE 相符。
- 若数据不平衡，则同时涉及固定效应和 BLUPS 的线性组合的标准误差与任何 DDFM 选项的 PROC MIXED 都不相符。但是，这些标准误差介于使用 DDFM=SATTERTHWAITE 选项和 DDFM=KENWARDROGER 选项分别获得的结果之间。若数据平衡，不论如何设置 DDFM 选项，JMP 结果始终与 SAS 结果相符，因为 Kackar-Harville 校正为空。

JMP PRO 自由度

仅涉及固定效应参数的线性组合的检验的自由度使用一阶 Kenward-Roger 校正来计算。因此，JMP 中这些检验的结果与使用 DDFM=KENWARDROGER(FIRSTORDER) 选项的 PROC MIXED 相符。若线性组合中存在 BLUP，JMP 将使用 Satterthwaite 近似来获取自由度。结果将遵循与上一段中为标准误差描述的模式类似的模式。

有关 Kackar-Harville 校正和 Kenward-Roger 自由度方法的详细信息，请参见 Kenward and Roger (1997)。Satterthwaite 方法在 SAS Institute Inc.(2020d, ch.83)。

第 9 章

多元响应模型

使用多元方差分析拟合关系

多元模型用一组效应拟合几个响应（Y 变量）。可以使用适当的响应设计检验 Y 变量上的函数。

除了创建标准 MANOVA（多元方差分析）模型外，您还可以使用以下方法：

- 重复测量分析，用于以下情况：对每个对象进行重复测量并且您要分析各测量值中对象间和对象内的效应。当测量值上的相关性结构具有任意性时，该多元方法特别重要。
- 典型相关性，用于查找 X 和 Y 变量具有最高相关性的线性组合。
- 判别分析，用于找到点和各组的多元均值之间的距离公式，以便可以将点分类到最有可能位于其中的组。更完整的判别分析是在**判别**平台中执行的。

多元拟合从显示参数估计值和最小二乘均值的初步分析开始。然后您可以指定 Y 变量上的响应设计并执行多元检验。

目录

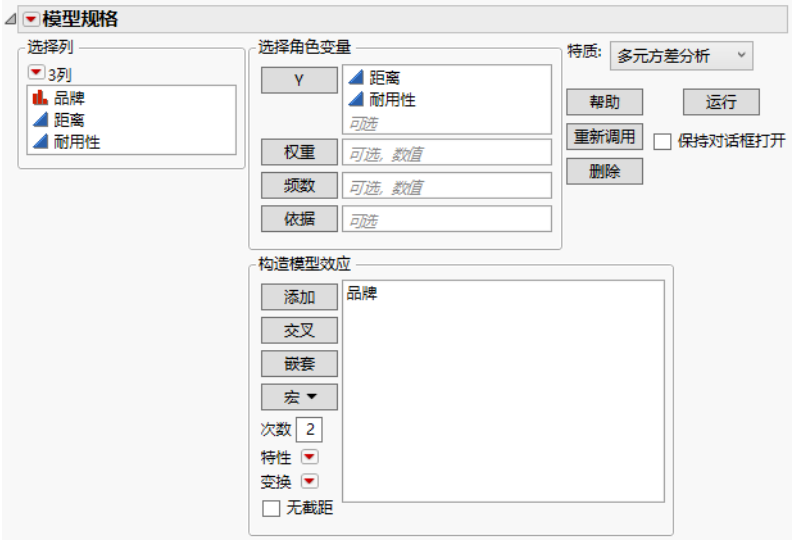
多重响应模型的示例 377
“多元方差分析”报表 379
“多元方差分析拟合”选项 379
响应规格 380
“选择响应”选项 380
“定制检验”选项 381
模型效应选项 381
多元检验 384
扩展的多元报表 385
多元检验的比较 386
一元检验和球形检验 386
具有重复测量的多元模型 387
重复测量示例 388
复合多元模型的示例 389
判别分析 391
“保存判别值”选项的示例 392
“多元方差分析”特质的统计详细信息 393
多元检验 393
近似 F 检验 394
典型详细信息 395

多重响应模型的示例

本例使用 Golf Balls.jmp 样本数据表 (McClave and Dietrich 1988)。数据是有关三种不同品牌的高尔夫球飞行距离比较和耐用性测量的。高尔夫发射机以随机顺序为每种品牌掷出 10 个球的随机样本。要检验的假设是距离和耐用性对于这三种高尔夫球品牌是相同的。

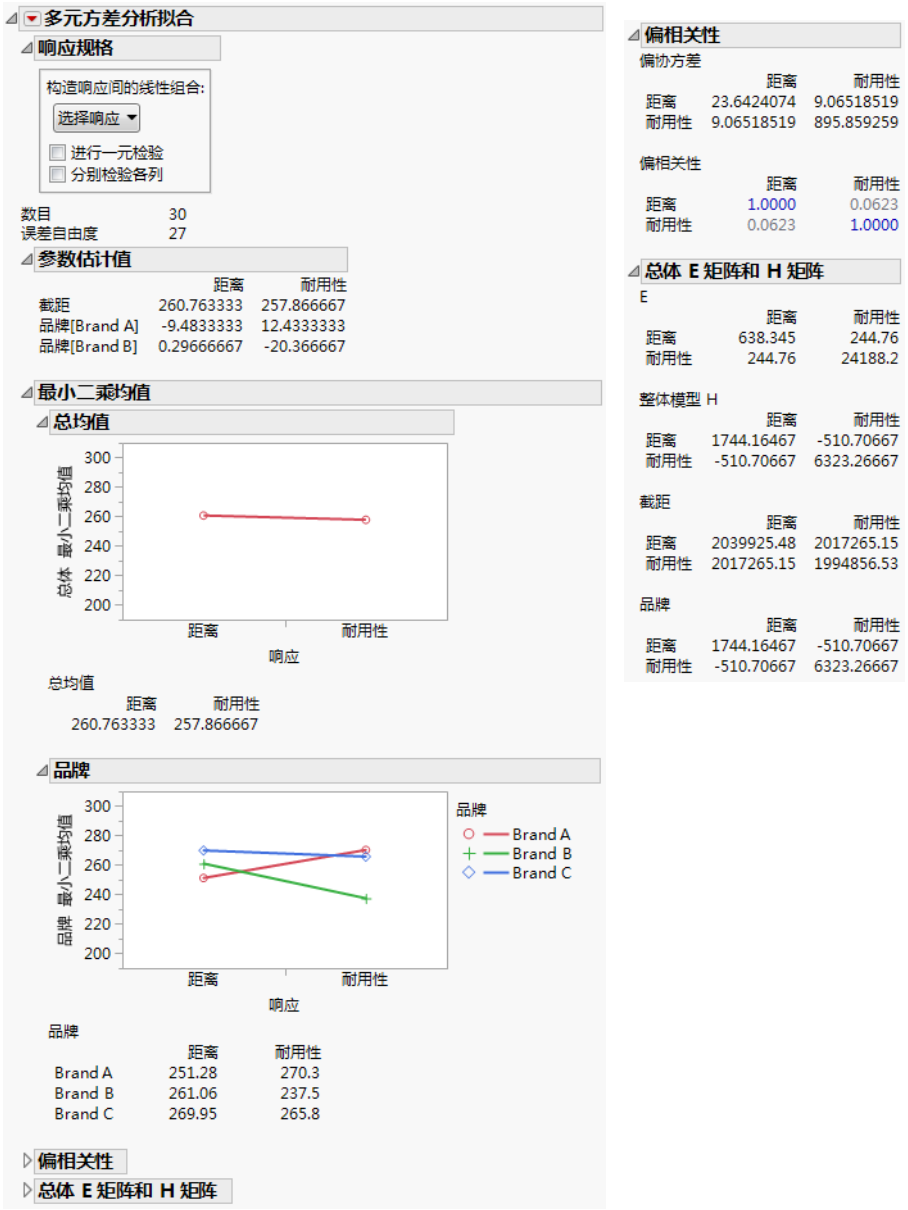
1. 选择**帮助** > **样本数据库**，然后打开 Golf Balls.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**距离**和**耐用性**，然后点击 Y。
4. 选择**品牌**，然后点击**添加**。
5. 对于**特质**，请选择**多元方差分析**。

图 9.1 多元方差分析设置



6. 点击**运行**。

图 9.2 “多元方差分析”报表窗口



初始结果本身可能不是很有趣，因为尚未指定响应设计。在指定响应设计后，多元平台显示多元估计值和检验的表。有关指定响应设计的详细信息，请参见第 380 页的“响应规格”。

“多元方差分析”报表

“多元方差分析”报表窗口包含以下元素：

“多元方差分析拟合”红色小三角菜单 包含保存选项。请参见第 379 页的““多元方差分析拟合”选项”。

响应规格 允许您指定各种检验的响应设计。请参见第 380 页的 “响应规格”。

参数估计值 包含每个响应变量的参数估计值（不含标准误差或 *t* 检验这类详细信息）。每个响应变量对应一列。

最小二乘均值 报告所有响应列的总最小二乘均值、每个名义型水平的最小二乘均值以及均值的最小二乘均值图。

偏相关性 显示初始拟合的残差的协方差矩阵和偏相关性矩阵，它们针对 *X* 效应进行了调整。

总体 E 矩阵和 H 矩阵 显示 E 和 H 矩阵：

- **E** 矩阵的元素是残差的叉积。
- **H** 矩阵对应于假设平方和以及叉积。

针对整体模型和模型中每个效应分别有一个 **H** 矩阵。 **E** 和 **H** 矩阵的对角线元素对应于一元 *F* 检验的假设 （分子）和误差 （分母）平方和。任何给定响应设计的新的 **E** 和 **H** 矩阵是从这些初始矩阵生成的，并依据它们计算多元检验统计量。

“多元方差分析拟合”选项

提供以下 “多元方差分析拟合”选项：

保存判别值 执行判别分析并将结果保存到数据表。请参见第 391 页的 “判别分析”。

保存预测值 将预测响应保存到数据表。

保存残差 将残差保存到数据表。

模型对话框 显示当前分析的已完成的启动窗口。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

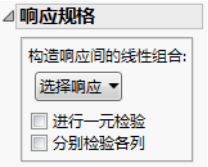
保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

响应规格

使用“响应规格”面板指定各种检验的响应设计。

图 9.3 “响应规格”面板



选择响应 提供 M 矩阵的选项。请参见第 380 页的 ““选择响应”选项”。

进行一元检验 获取调整和未调整的一元重复测量检验以及多元检验。在重复测量模型中使用。

分别检验各列 获取针对每个响应的一元方差分析检验和多元检验。

当您选择了一个响应选项后，以下按钮才可用：

运行 执行分析并显示多元估计值和检验。请参见第 384 页的 “多元检验”。

帮助 显示有关“响应规格”面板的帮助。

正交化 正交化矩阵。在列对除 “总和”之外的所有响应类型进行对比 （总和为零）后执行正交化。

删除最后一列 降低变换的维数。

“选择响应”选项

响应设计生成 M 矩阵。 M 矩阵的列定义多元分析的一组变换变量。**选择响应**按钮包含 M 矩阵的以下选项：

重复测量 构造并运行 “总和”和 “对比”响应。

总和 给出单个值的响应之和。

恒等 使用每个单独的响应 （单位矩阵）。

对比 将每个响应与第一个响应比较。

多项式 构造正交多项式矩阵。

Helmert 将每个响应与它下面所列的组合响应比较。

刻画 将每个响应与后面的响应比较。

均值 将每个响应与其他响应的均值比较。

复合 若响应是由两个效应复合生成的，则创建并运行几个合适的响应函数。

定制 使用您输入的任意定制 M 矩阵。

多元回归的最常见响应设计是**重复测量**和**恒等**。由**对比**、Helmert、**刻画**和**均值**选项所给出的检验差别不大，因为它们跨越相同的空间。但是，它们的“最小二乘均值”和“参数估计值”表中的检验和详细信息相应地显示不同的重要信息。

重复测量和**复合**选项显示对话框来指定响应效应名称。然后它们拟合几个响应函数而不用等待进一步的用户输入。否则，选定内容会展开控制面板，为您提供更多机会来细化规格。

“定制检验”选项

使用**定制检验**选项设置效应水平的定制检验。

注意：有关如何创建定制检验的说明，请参见“标准最小二乘法模型”一章中第 109 页的“定制检验”。

模型效应选项

每个效应名称旁边的菜单图标为您提供以下选项，以请求有关多元拟合的其他信息：

检验详细信息 显示用于构造多元检验统计量的 $\mathbf{E}^{-1}\mathbf{H}$ 矩阵的特征值和特征向量。请参见第 381 页的“检验详细信息”。

重心图 在从检验空间生成的前两个典型变量上标绘重心（多元最小二乘均值）。请参见第 382 页的“重心图”。

保存典型得分 将名为 Canon[1]、Canon[2] 等的变量保存为当前数据表中的列。这些列具有值和公式。相关示例，请参见第 383 页的“保存典型得分”。有关技术细节的信息，请参见第 395 页的“典型详细信息”。

对比 执行您在对比对话框中指定的处理水平的统计对比。

注意：对比命令与具有单个响应的回归相同。有关**最小二乘均值对比**命令的说明和示例，请参见“标准最小二乘法模型”一章中第 91 页的“最小二乘均值对比”。

检验详细信息

“检验详细信息”报表提供有关整体模型或指定效应的检验的典型详细信息。

特征值 在计算多元检验统计量中使用的 $\mathbf{E}^{-1}\mathbf{H}$ 矩阵的特征值。

典型相关性 与每个特征值关联的典型相关性。这是变换的响应与效应的典型相关性（已针对模型中的所有其他效应进行了校正）。

特征向量 $\mathbf{E}^{-1}\mathbf{H}$ 矩阵或等价的 $(\mathbf{E}+\mathbf{H})^{-1}\mathbf{H}$ 的特征向量。

“检验详细信息”的示例

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。

   Iris 数据 (Mardia et al. 1979) 具有**物种**的三个水平，名为 Virginica、 Setosa 和 Versicolor。对每个样本取四个测量值 （**花瓣长度**、**花瓣宽度**、**萼片长度**和**萼片宽度**）。
2. 选择**分析** > **拟合模型**。
3. 选择**花瓣长度**、**花瓣宽度**、**萼片长度**和**萼片宽度**，然后点击 Y。
4. 选择**物种**并点击**添加**。
5. 对于**特质**，请选择**多元方差分析**。
6. 点击**运行**。
7. 点击**选择响应**按钮并选择**恒等**。
8. 点击**运行**。
9. 点击 “物种”红色小三角并选择**检验详细信息**。

   将显示特征值、特征向量和典型相关性。

图 9.4 检验详细信息

物种

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.0234386	199.1453	8	288	<.0001*
Pillai 迹	1.1918988	53.4665	8	290	<.0001*
Hotelling-Lawley	32.47732	582.1970	8	203.4	<.0001*
Roy 最大根	32.191929	1166.9574	4	145	<.0001*

特征值	典型相关性
32.1919292	0.98482089
0.28539104	0.47119702
1.235e-15	0
-6.174e-16	0

特征向量

萼片长度	-0.0684059	0.00198791	-0.2350196	0.1176771
萼片宽度	-0.1265612	0.1785267	0.21657608	0.04510419
花瓣长度	0.18155288	-0.0768636	0.23964446	0.06563465
花瓣宽度	0.23180286	0.23417227	-0.2865277	-0.2438953

重心图

重心图命令（从“物种”红色小三角访问）在从检验空间生成的前两个典型变量上标绘重心（多元最小二乘均值），如图 9.5 中所示。第一个典型轴是垂直轴，以便在检验空间只有一个维时，重心在垂直轴上排成一条直线。重心点用对应于 95% 置信区域的圆圈显示 (Mardia et al. 1979)。在效应检验下创建重心图时，对应于要检验的效应的圆圈显示为红色。其他圆圈显示为蓝色。双标图射线显示检验空间中原始响应变量的方向。请参见第 395 页的“ “重心图”选项的详细信息”。

点击**重心值**展开图标以显示其他信息，如图 9.5 中所示。

具有特征值的第一个典型轴比第二个轴解释更多分离。均值被很好地分离（判别），第一组比其他两组隔得更远。第一个典型变量似乎针对花瓣宽度变量加载花瓣长度变量。可以使用双标图射线和相关的特征向量验证各组变量之间的关系。

图 9.5 重心图和重心值

保存典型得分

将名为 Canon[i] 的列保存到数据表，其中 *i* 表示 Y 变量的第 *i* 个典型得分。典型得分基于用于构造多元检验统计量的 $\mathbf{E}^{-1}\mathbf{H}$ 矩阵来计算。保存对应于非零特征值的特征向量的典型得分。

典型相关性

典型相关性分析不是特定命令，但是可以通过多元拟合平台中的一系列命令来执行该分析。

1. 执行第 382 页的““检验详细信息”的示例”中的第 1 步到第 8 步。
2. 点击“整体模型”红色小三角并选择**检验详细信息**。
3. 点击“整体模型”红色小三角并选择**保存典型得分**。

详细信息在特征值旁边列出典型相关性（**典型相关性**）。保存的变量称为 Canon[1]、Canon[2] 等等。这些列同时包含值和公式。

要获得 *X* 侧的典型变量，请重复相同步骤，但是将 *X* 和 *Y* 变量互换。若您已有附加到数据表的列 Canon[n]，则新列名为使名称唯一的 Canon[n] 2（或另一数字）。

这是典型相关性分析的另一个示例。

1. 选择**帮助** > **样本数据库**，然后打开 Exercise.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**引体向上**、**仰卧起坐**和**向上跳**，然后点击 Y。
4. 选择**体重**、**腰围**和**脉搏**，然后点击**添加**。
5. 对于**特质**，请选择**多元方差分析**。

6. 点击**运行**。
7. 点击**选择响应**按钮并选择**恒等**。
8. 点击**运行**。
9. 点击 “整体模型”红色小三角并选择**检验详细信息**。
10. 点击 “整体模型”红色小三角并选择**保存典型得分**。

图 9.6 典型相关性

整体模型

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.3503905	2.0482	9	34.223	0.0635
Pillai 迹	0.6784815	1.5587	9	48	0.1551
Hotelling-Lawley	1.7719415	2.6397	9	19.053	0.0357*
Roy 最大根	1.7247387	9.1986	3	16	0.0009*

特征值	典型相关性
1.72473874	0.79560815
0.0419084	0.20055604
0.00529433	0.07257029

特征向量

引体向上	0.02503681	-0.016636	0.05641878
仰卧起坐	0.00637953	0.0004622	-0.004547
向上跳	-0.0052909	0.0048507	0.0018787

输出典型变量使用显示为 *Y* 变量的线性组合的特征向量。例如， Canon[1] 计算如下：

```
0.02503681* 引体向上数 + 0.00637953* 仰卧起坐数 + -0.0052909* 向上跳数
```

该典型分析不生成均值为 0 且标准差为 1 的标准化变量，但是很容易使用具有这些功能的计算器定义新的标准化变量。

多元检验

选择响应的示例

1. 完成第 377 页的“多重响应模型的示例”中的第 1 步到第 6 步。
2. 点击**选择响应**按钮并选择**恒等**。
3. 点击**运行**。

图 9.7 多元检验报表

恒等

M 矩阵

M 变换参数估计值

整体模型

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.2117852	15.2485	4	52	<.0001*
Pillai 迹	0.936487	11.8876	4	54	<.0001*
Hotelling-Lawley	3.0216575	19.4162	4	30.19	<.0001*
Roy 最大根	2.7688021	37.3788	2	27	<.0001*

截距

检验	值	精确的 F 值	分子自由度	分母自由度	概率>F
F 检验	3226.6839	41946.890	2	26	<.0001*

品牌

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.2117852	15.2485	4	52	<.0001*
Pillai 迹	0.936487	11.8876	4	54	<.0001*
Hotelling-Lawley	3.0216575	19.4162	4	30.19	<.0001*
Roy 最大根	2.7688021	37.3788	2	27	<.0001*

M 矩阵报表提供您指定的响应设计。**M 变换参数估计值**报表提供原始参数估计值矩阵与 M 矩阵的转置矩阵之积。

注意：最初在本章中，矩阵名称 **E** 和 **H** 表示误差和假设叉积。指定响应设计后，**E** 和 **H** 表示用响应设计变换的那些矩阵，实际上为 **M′EM** 和 **M′HM**。

扩展的多元报表

在多元拟合中，归因于假设和误差的平方和是平方与叉积的矩阵（而非单个数值）。有很多方法来衡量假设平方和以及叉积的矩阵（称为 **H** 或 **SSCP**）值与残差的矩阵（称为 **E**）值相比有多大。JMP 报告文献中常见的四种多元检验。若您在寻找精确显著性水平下的检验，可能需要在参考书中查找表。幸运的是，这四种检验都可以变换为近似 *F* 检验。若响应设计生成单个值，或假设是单个自由度，则多元检验是等价的，生成相同的精确 *F* 检验。JMP 用**精确的 F 值**标记检验。否则 JMP 用**近似的 F 值**标记该检验。

在高尔夫球示例中，只有一个效应，因此整体模型检验和**品牌**的检验是相同的，它们将使用近似 *F* 值检验显示四种多元检验。只有自由度为 2（每个响应各一个）的单个截距，因此它的 *F* 检验是精确的，被标记为**精确的 F 值**。

“整体模型”、“截距”和“品牌”报表上的红色小三角菜单包含生成其他信息的选项，这些信息包括特征值、典型相关性、重心值列表、重心图和用于保存典型变量的**保存**命令。

效应（该示例中为**品牌**）弹出菜单也包含用于指定对比的选项。

定制检验和对比功能与具有单个响应的回归的这些功能相同。请参见第 67 页的“标准最小二乘法模型”一章。

要查看多元方差分析表检验的公式，请参见第 393 页的“多元检验”。

扩展的 “多元”报表包含以下列：

检验 标记表中的每个统计检验。若响应函数值 （**M** 矩阵中指定的列）的数目为 1 或效应对于每个响应函数只有一个自由度，则提供精确 *F* 检验。否则，使用近似 *F* 检验给出四个标准多元检验统计量：Wilks Lambda (Λ)、 Pillai 迹、 Hotelling-Lawley 迹和 Roy 最大根。

值 报表中每个多元统计检验的值。

近似的 F 值 （或精确的 F 值） 对应于多元检验的 *F* 值。若响应设计生成单个值或检验只有一个自由度，则是精确 *F* 检验。

分子自由度 分子自由度。

分母自由度 分母自由度。

概率 > F 对应于 *F* 值的显著性概率。

注意：有关“球形检验”表的详细信息，请参见第 386 页的“一元检验和球形检验”。

多元检验的比较

尽管四种标准多元检验通常给出相似的结果，但是在一些情形下它们的结果是不同的，一种检验可能比另一种检验更有优势。很遗憾，它们没有明确的优劣之分。通常，按功效给出以下优先顺序：

1. Pillai 迹
2. Wilks Lambda
3. Hotelling-Lawley 迹
4. Roy 最大根

在与原假设有很大偏差且特征值相差很大时，优先顺序正好倒过来 (Seber 1984)。

一元检验和球形检验

在一些情形下 （如重复测量模型），允许将多元问题变换为一元问题 (Huynh and Feldt 1970)。在以下情形下在多元环境中使用一元检验是有效的：

- 若响应设计矩阵 **M** 是正交的（**M**′**M** = 单位）。
- 若 **M** 生成多个响应且每个变换的系数之和为零。
- 若满足**球形**条件。球形条件意味着经过 M 变换后的响应不相关且具有相同方差。**M**′Σ**M** 与单位矩阵成比例，其中 Σ 是 *Y* 变量的协方差。

若保持这些条件，则对 **E** 和 **H** 检验矩阵的对角线元素求和，以生成 *F* 检验的分母和分子的一元平方和。请注意，若不保持上述条件，将显示错误消息。在 Golf Balls.jmp 的案例中，单位矩阵指定为 **M** 矩阵。在列向量中心化和正交化后，单位矩阵不能变换为满秩矩阵。因此忽略一元请求。

一元和球形检验的示例

1. 选择**帮助** > **样本数据库**，然后打开 Dogs.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**组织胺 0 的对数**、**组织胺 1 的对数**、**组织胺 3 的对数**和**组织胺 5 的对数**，然后点击 Y。
4. 选择**药物**和**消耗**，然后点击**添加**。
5. 在 “构造模型效应”面板中，选择**药物**。在 “选择列”面板中，选择**消耗**。点击**交叉**。
6. 对于 “特质”，请选择**多元方差分析**。
7. 点击**运行**。
8. 选中**进行一元检验**旁边的复选框。
9. 在**选择响应**菜单中，选择**重复测量**。

   应为 “Y 名称”输入 “时间”，应选中**进行一元检验**。
10. 点击**确定**。

图 9.8 球形检验

球形检验	
Mauchly 准则	0.1752641
卡方	16.930873
自由度	5
概率>卡方	0.0046328

球形检验使用 Mauchly 准则检查对象内效应的未调整的一元 *F* 检验是否合适，以检验球形假设 (Anderson 1958)。始终使用正交化的 **M** 矩阵进行球形检验和一元检验。使用以下指南解释球形检验：

- 若真正的协方差结构为球形，您可以使用未调整的一元 *F* 检验。
- 若球形检验显著，该检验表明真正的协方差结构不是球形的。因此，您可以使用多元或调整的一元检验。

一元 *F* 统计量具有近似 *F* 分布（即使不含球形），但分子和分母的自由度大约减少了一个分数，大小为 epsilon (ε)。Box (1954)、Greenhouse and Geisser (1959) 以及 Huynh-Feldt (1976) 提供估计 epsilon 自由度调整的方法。Muller and Barton (1989) 基于功效的研究建议采用 Greenhouse-Geisser 版本。

在多元报表中 epsilon 调整的检验标记为 G-G (Greenhouse-Geisser) 或 H-F (Huynh-Feldt)。epsilon 调整量显示在**值**列中。

具有重复测量的多元模型

多元拟合的一个常见用法是分析具有重复测量的数据（也称为**纵向数据**）。在一段时间上重复测量某个对象，排列数据使每个时间测量值构成一个变量。因为测量值之间的相关性，不应将

数据堆叠为单个列，也不应作为一元模型进行分析，除非相关性形成了称为**球形**的模式。有关该主题的更多详细信息，请参见上一节第 386 页的“一元检验和球形检验”。

使用重复测量，可以将分析分为两个层次：

- 通过使用模型效应拟合重复测量列之和为来对对象间效应建模。这对应于使用**总和**响应函数。该响应函数是作为 1 的单一向量的 **M** 矩阵。
- 使用拟合重复测量列差值的响应函数对对象内效应（重复效应或时间效应）建模。可以使用**对比**响应函数或以下任意类似差分函数执行该分析：**多项式**、Helmert、**刻画**或**均值**。当对重复测量间的差值建模时，将差值视为新的对象内效应（通常为时间）来处理。在模型中拟合效应时，将它们解释为与对象内效应的交互作用。例如，截距的效应变为“时间”（对象内）效应，显示重复测量间的总差值。若您有效应 A，则将对象内检验解释为 A*时间交互作用的检验，它用模型说明重复测量差值如何随 A 效应变化。

第 389 页上的表 9.1 显示响应函数和模型效应（与一元模型规格相比）之间的关系。使用**总和**（对象间）和**对比**（对象内）模型，您应可以重新构造源自将响应堆叠为单个列并得到标准一元拟合的检验。

可以通过直接和间接方式来执行重复测量分析：

- 直接方式是使用弹出菜单项“重复测量”。这提示您命名表示重复测量间对象内效应的效应。然后它拟合**对比**和**总和**响应函数。这种方式的优点是使用对象内效应名称合理标记效应。
- 间接方式是单独指定两个响应函数。首先，指定**总和**响应函数；其次，指定**对比**或对差值建模的其他某个函数。您需要记住在对比拟合中将对象内效应与模型效应关联。

重复测量示例

例如，考虑 Cole and Grizzle (1966) 所做的研究。结果位于样本数据文件夹中的 Dogs.jmp 表。将 16 条狗分配给由变量**药物**和**消耗**定义的四个组，每个变量各具有两个水平。因变量是注射药物 0、 1、 3 和 5 分钟后组胺的血药浓度。使用浓度的对数将数据的均值和方差之间的相关性最小化。

1. 选择**帮助** > **样本数据库**，然后打开 Dogs.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**组织胺 0 的对数**、**组织胺 1 的对数**、**组织胺 3 的对数**和**组织胺 5 的对数**，然后点击 Y。
4. 选择**药物**和**消耗**，然后从**宏**菜单选择**完全析因**。
5. 对于 “特质”，请选择**多元方差分析**。
6. 点击**运行**。
7. 在**选择响应**菜单中，选择**重复测量**。

   应为“Y 名称”输入“时间”。若您选中**进行一元检验**复选框，则报表包含一元检验，计算该检验时就仿佛将响应堆叠成了单个列。
8. 点击**确定**。

图 9.9 “重复测量”窗口

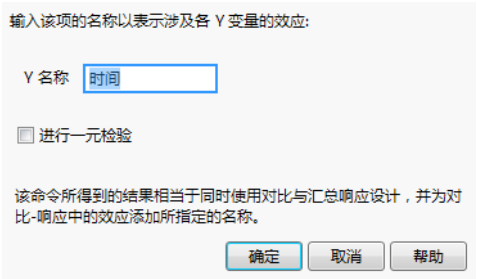


表 9.1 显示**总和**和**对比**响应设计的多元检验如何对应于一元检验标记方式（此时列**组织胺 0 的对数**、**组织胺 1 的对数**、**组织胺 3 的对数**和**组织胺 5 的对数**的数据堆叠为单个 *Y* 列）。使用名义型分组变量**时间**标识新行。

表 9.1 对应的多元和一元检验

总和 M 矩阵 对象间		对比 M 矩阵 对象内	
多元检验	一元检验	多元检验	一元检验
截距	截距	截距	时间
药物	药物	药物	时间*药物
消耗	消耗	消耗	时间*消耗

首先进行对象间分析。该分析与在弹出菜单上选择了**总和**时执行的分析相同（标题不同）。

接着生成对象内分析。该分析与在弹出菜单上选择了**对比**时执行的分析相同（标题不同），尽管对象内效应名称（**时间**）已添加到报表中的效应名称。请注意以前由**截距**占据的位置为**时间**，因为截距项估计的是重复测量值上的总差异。

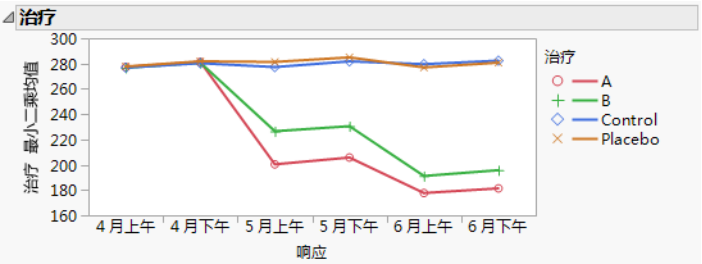
复合多元模型的示例

JMP 可以处理具有多个重复测量层次的数据。例如，请查看 Cholesterol.jmp 数据表。五个参与者构成的组分别属于名为 A、B、Control 和 Placebo 的四个**治疗**组中的一个。在三个月中每个月的早上和下午各测量一次胆固醇 （数据是虚构的） 。在本例中，按月内当天的时间这样的时间顺序排列响应列。

1. 选择**帮助** > **样本数据库**，然后打开 Cholesterol.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择 4 月上午、 4 月下午、 5 月上午、 5 月下午、 6 月上午和 6 月下午，然后点击 Y。

4. 选择**治疗**，然后点击**添加**。
5. 在“特质”旁边，选择**多元方差分析**。
6. 点击**运行**。

图 9.10 “治疗”图



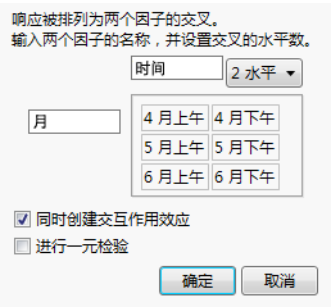
在**治疗**图中，您可以看到开始研究的四个治疗组具有很相似的胆固醇均值。A 和 B 治疗组在试验期结束后看上去具有更低的胆固醇值。 control 和 placebo 组没有什么变化。

7. 点击**选择响应**菜单，然后选择**复合**。

   完成该窗口以告知 JMP 如何在数据表中排列响应以及每个响应的水平数。在胆固醇示例中，在月内排列当天的时间列。因此，您将当天的时间命名为一个因子，将月效应命名为另一个因子。检验交互作用效应是可选的。

8. 使用图 9.11 中的选项来完成窗口。

图 9.11 “复合”窗口



9. 点击**确定**。

   将显示每个效应的检验。图 9.12 中显示了部分报表。请注意以下事项：

   – “时间”报表显示“时间”与**治疗**之间的交互作用的 *p* 值为 0.6038，这指示交互作用不显著。这意味着没有证据表明“上午”和“下午”的**治疗**之间存在差异。因为“时间”有两个水平（“上午”和“下午”），所以将显示精确 *F* 检验。

   – “月”报表显示“月”与**治疗**之间的交互作用的 *p* 值 <.0001，这指示交互作用显著。这暗示治疗组之间的差异变化取决于月份。图 9.10 中的**治疗**图指示在四月组之间没有差

异，但是在五月治疗类型（A、B、Control 和 Placebo）之间的差异变大，在六月变得更大。

– “时间*月”的报表显示**治疗的** *p* 值不显著。这表明涉及“时间”、“月”和**治疗**的三因子交互作用效应在统计上不显著。

图 9.12 胆固醇研究结果

复合

时间

M 矩阵

M 变换参数估计值

整体模型

截距

治疗

检验	值	精确的 F 值	分子自由度	分母自由度	概率>F
F 检验	0.1188721	0.6340	3	16	0.6038

复合

月

M 矩阵

M 变换参数估计值

整体模型

截距

治疗

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.013025	38.8109	6	30	<.0001*
Pillai 迹	1.3128917	10.1907	6	32	<.0001*
Hotelling-Lawley	50.753209	123.4368	6	18.326	<.0001*
Roy 最大根	50.255302	268.0283	3	16	<.0001*

复合

时间*月

M 矩阵

M 变换参数估计值

整体模型

截距

治疗

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.6623742	1.1435	6	30	0.3619
Pillai 迹	0.3582613	1.1638	6	32	0.3498
Hotelling-Lawley	0.4785668	1.1639	6	18.326	0.3671
Roy 最大根	0.4008469	2.1378	3	16	0.1355

判别分析

判别分析是一种基于响应的已知值预测单因子分类的某个水平的方法。该方法基于测量值变量与要预测的多元水平均值的接近程度。使用“判别”平台可更全面地执行判别分析。请参见《多元方法》。

在 JMP 中，您指定测量值变量为 *Y* 效应，指定分类变量为单个 *X* 效应。多元拟合平台给出数据的均值估计值和协方差矩阵（假定协方差对于每个组是相同的）。您使用“多元方差分析”平台名称旁边的弹出菜单中的**保存判别值**选项获取判别信息。该命令使用初始 **E** 和 **H** 矩阵将距离和概率保存为当前数据表中的列。

对于具有 k 个水平的分类变量，JMP 向当前数据表添加 k 个距离列、k 个分类概率列、一个预测的分类列以及两列其他计算信息。

“保存判别值”选项的示例

查看 Mardia et al. (1979) 中的 Fisher 的 Iris 数据。物种有 $k = 3$ 个水平，每个样本取四个测量值。

1. 选择**帮助 > 样本数据库**，然后打开 Iris.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y**。
4. 选择**物种**并点击**添加**。
5. 在“特质”旁边，选择**多元方差分析**。
6. 点击**运行**。
7. 点击“多元方差分析拟合”红色小三角并选择**保存判别值**。

以下列添加到 Iris.jmp 样本数据表：

SqDist[0] Mahalanobis 距离计算中所需的二次型。

SqDist[setosa] 观测值与 Setosa 重心的 Mahalanobis 距离。

SqDist[versicolor] 观测值与 Versicolor 重心的 Mahalanobis 距离。

SqDist[virginica] 观测值与 Virginica 重心的 Mahalanobis 距离。

Prob[0] 在下面使用的 Mahalanobis 距离的负指数之和。

Prob[setosa] 属于 Setosa 类别的概率。

Prob[versicolor] 属于 Versicolor 类别的概率。

Prob[virginica] 属于 Virginica 类别的概率。

预测的 “物种” 根据概率得到的最可能的物种。

现在您可以将数据表中的新列用于其他 JMP 平台来使用报表和图形汇总判别分析。例如：

1. 从更新的 Iris.jmp 数据表（包含新列）中，选择**分析 > 以 X 拟合 Y**。
2. 选择**物种**，然后点击 **Y，响应**。
3. 选择 **“物种”预测值**，然后点击 **X，因子**。
4. 点击**确定**。

列联表汇总了判别分类。标识了三个误分类。

图 9.13 预测和实际物种的列联表

列联表

		物种			
	计数 合计百分比 列百分比 行百分比	setosa	versicolor	virginica	合计
Pred 物种	setosa	50 33.33 100.00 100.00	0 0.00 0.00 0.00	0 0.00 0.00 0.00	50 33.33
	versicolor	0 0.00 0.00 0.00	48 32.00 96.00 97.96	1 0.67 2.00 2.04	49 32.67
	virginica	0 0.00 0.00 0.00	2 1.33 4.00 3.92	49 32.67 98.00 96.08	51 34.00
	合计	50 33.33	50 33.33	50 33.33	150

“多元方差分析”特质的统计详细信息

- “多元检验”
- “近似 F 检验”
- “典型详细信息”

多元检验

在下文中，**E** 是残差叉积矩阵，**H** 是模型叉积矩阵。**E** 的对角线元素是每个变量的残差平方和。**H** 的对角线元素是每个变量的模型平方和。在判别分析文献中，**E** 通常称为 **W**，其中 **W** 表示**组内**。

多元结果表中的检验统计量是 $\mathbf{E}^{-1}\mathbf{H}$ 的特征值 λ 的函数。以下列表说明了每个检验统计量的计算。

注意：在指定响应设计后，将初始 **E** 和 **H** 矩阵用 **M**′ 自左乘，然后用 **M** 自右乘。

- Wilks Lambda

$$\Lambda = \frac{\det(\mathbf{E})}{\det(\mathbf{H}+\mathbf{E})} = \prod_{i=1}^{n}\left(\frac{1}{1+\lambda_i}\right)$$

- Pillai 迹

$$V = \text{迹}[\mathbf{H}(\mathbf{H}+\mathbf{E})^{-1}] = \sum_{i=1}^{n}\frac{\lambda_i}{1+\lambda_i}$$

- Hotelling-Lawley 迹

$$U = \text{迹}(\mathbf{E}^{-1}\mathbf{H}) = \sum_{i=1}^{n} \lambda_i$$

- Roy 最大根

$\Theta = \lambda_1$，$\mathbf{E}^{-1}\mathbf{H}$ 的最大特征值。

E 和 **H** 定义如下：

$$\mathbf{E} = \mathbf{Y}'\mathbf{Y} - \mathbf{b}'(\mathbf{X}'\mathbf{X})\mathbf{b}$$

$$\mathbf{H} = (\mathbf{L}\mathbf{b})'(\mathbf{L}(\mathbf{X}'\mathbf{X})^{-}\mathbf{L}')^{-1}(\mathbf{L}\mathbf{b})$$

其中 **b** 是模型系数的估计向量，$\mathbf{A}^{-}$ 表示矩阵 **A** 的广义逆。

整体模型 **L** 是与以下单位矩阵拼接而成的零列（用于截距）：该矩阵的行数和列数与模型中的参数数目相等。效应的 **L** 矩阵是整体模型 **L** 矩阵的行的子集。

近似 F 检验

为了计算 F 值和自由度，用 p 表示 $\mathbf{H}+\mathbf{E}$ 的秩。用 q 表示 $\mathbf{L}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{L}'$ 的秩，其中 **L** 矩阵标识与要检验的效应关联的 $\mathbf{X}'\mathbf{X}$ 的元素。用 v 表示误差自由度，用 s 表示 p 和 q 中的最小项。还假定 $m = 0.5(|p-q|-1)$ 和 $n = 0.5(v-p-1)$。

第 394 页上的表 9.2 列出了从相应的检验统计量如何计算每个近似 F 值。

表 9.2 近似 F 统计量

检验	近似 F	分子自由度	分母自由度
Wilks Lambda	$F = \left(\frac{1-\Lambda^{1/t}}{\Lambda^{1/t}}\right)\left(\frac{rt-2u}{pq}\right)$	pq	$rt-2u$
Pillai 迹	$F = \left(\frac{V}{s-V}\right)\left(\frac{2n+s+1}{2m+s+1}\right)$	$s(2m+s+1)$	$s(2n+s+1)$
Hotelling-Lawley 迹	$F = \frac{2(sn+1)U}{s^2(2m+s+1)}$	$s(2m+s+1)$	$2(sn+1)$
Roy 最大根	$F = \frac{\Theta(v-\max(p,q)+q)}{\max(p,q)}$	$\max(p,q)$	$v-\max(p,q)+q$

典型详细信息

“检验详细信息”选项的详细信息

当您为给定检验选择“检验详细信息”选项后，特征值、典型相关性和特征向量都会显示在报表中。

“检验详细信息”选项所生成的典型相关性计算如下：

$$\rho_i = \sqrt{\frac{\lambda_i}{1+\lambda_i}}$$

其中，λ_i 是计算多元检验统计量所使用的 $\mathbf{E}^{-1}\mathbf{H}$ 矩阵的第 i 个特征值

标记为“特征向量”的矩阵是 **V** 矩阵，该矩阵是给定检验的 $\mathbf{E}^{-1}\mathbf{H}$ 的特征向量矩阵。

注意：给定检验的 **E** 和 **H** 矩阵指的是初始 **E** 和 **H** 矩阵中的 **M′EM** 和 **M′HM**。**M** 矩阵由响应设计定义。本节中使用的 **E** 和 **H** 在第 393 页的“多元检验”中定义。

“重心图”选项的详细信息

效应的总样本重心和重心值计算如下：

总体 = $(c'_1\bar{y}, c'_2\bar{y}, \ldots, c'_g\bar{y})$

效应 $_j$ = $(c'_1\bar{x}_j, c'_2\bar{x}_j, \ldots, c'_g\bar{x}_j)$

其中

$$c_i = \left(v'_i\left(\frac{E}{N-r}\right)v_i\right)^{-1/2} v_i$$

N 是观测数

v_i 是 **V**（即给定检验的 $\mathbf{E}^{-1}\mathbf{H}$ 特征向量矩阵）的第 i 列。

$\bar{x}_j$ 是第 j 个效应的多元最小二乘均值

$\bar{y}$ 是响应的总均值

g 是大于 0 的 $\mathbf{E}^{-1}\mathbf{H}$ 的特征值数

r 是 **X** 矩阵的秩

注意：给定检验的 **E** 和 **H** 矩阵指的是初始 **E** 和 **H** 矩阵中的 **M′EM** 和 **M′HM**。**M** 矩阵由响应设计定义。本节中使用的 **E** 和 **H** 在第 393 页的“多元检验”中定义。

效应的重心半径计算如下：

$$d = \sqrt{\frac{\chi^2_g(0.95)}{L(X'X)^{-1}L'}}$$

其中，g 是大于 0 的 $\mathbf{E}^{-1}\mathbf{H}$ 的特征值数，分母中的 **L** 矩阵来自多元最小二乘均值计算。

“保存典型评分”选项的详细信息

典型 Y 值计算如下：

$$\tilde{Y} = \mathbf{YM'V}$$

其中

Y 是响应变量矩阵

M′ 是转置的响应设计矩阵

V 是给定检验的 $\mathbf{E}^{-1}\mathbf{H}$ 的特征向量矩阵

注意：给定检验的 **E** 和 **H** 矩阵指的是初始 **E** 和 **H** 矩阵中的 **M′EM** 和 **M′HM**。**M** 矩阵由响应设计定义。本节中使用的 **E** 和 **H** 在第 393 页的 “多元检验”中定义。

为与大于零的特征值对应的特征向量保存典型 Y 值。

第 10 章

对数线性方差模型

对响应的方差和均值建模

“拟合模型”平台的“对数线性方差”特质允许您使用回归模型对响应的期望值和方差建模。使用一个线性模型拟合方差的对数，同时使用其他线性模型拟合期望响应。

注意：估计值要求很多精心设计的良好拟合数据。相对于拟合均值，您需要更多数据来拟合方差。

对于很多工程师来说，实验目标不是最大化或最小化响应本身，而是达到目标响应并实现最小的变异性。对数线性方差模型提供一个非常普遍和有效的方法来对方差建模，可用于无重复数据以及具有重复的数据。

目录

对数线性方差模型概述 399
散度效应 399
模型规格 399
注意 399
启动“对数线性方差”特质 400
使用对数线性方差的示例 400
对数线性报表 403
“对数线性”平台选项 404
保存列 404
行诊断 405
查看残差 405
刻画拟合模型 406
刻画拟合模型的示例 406

对数线性方差模型概述

对数线性方差模型提供一种仅通过线性模型对方差建模的方法。请参见 Harvey (1976)、 Cook and Weisberg (1983)、 Aitken (1987) 和 Carroll and Ruppert (1988)。除了使用回归变量项对响应均值建模之外，在线性模型中还使用回归变量项对方差对数建模：

均值模型：$E(y) = \mathbf{X}\beta$

方差模型：log(**方差** (y)) = $\mathbf{Z}\,\lambda$，

或等价于

方差 (y) = exp($\mathbf{Z}\,\lambda$)

其中 X 的列是响应均值的回归变量，Z 的列是响应方差的回归变量。常规线性模型参数用 β 表示，λ 表示方差模型的参数。

使用 REML 估计对数线性方差模型。

离散或**对数方差**效应可以对响应方差的变化建模。该模型是通过“拟合模型”平台中的一个名为“对数线性方差”的拟合特质来实现。

散度效应

对散度效应建模并未在教科书中广泛涉及，只有田口方法框架中有所提及。在田口样式的实验中，通过以下方式处理散度效应：在外表的不同设置下进行多次测量，构造新响应来测量这个外表上偏离目标的变异性，然后拟合模型来找到实现最小变异性的因子。这类建模需要一个作为两个设计的完整笛卡儿积专门的设计。本章采用更灵活的基于模型的方法对方差建模。田口建议用于变异性建模的特殊性能统计量为 $STD = -\log(s)$。在 JMP 使用的方法中，对 $\log(s^2)$ 建模并将它与具有均值的模型合并。这两个方法基本上是等价的，因为 $\log(s^2)=2\log(s)$。

模型规格

通过突出显示并在**特性**下拉菜单中选择**对数方差效应**，在“拟合模型”对话框中指定对数线性方差效应。**&对数方差**显示在效应末尾。使用该特性时，它还将顶部的拟合**特质**更改为**对数线性方差**。若您希望一个效应同时用于响应的均值和方差，必须指定它两次，其中使用**对数方差**选项指定一次。

使用对数方差特性指定的效应成为在模型中生成 Z 的效应，其他效应在模型中成为 X。

注意

每当为均值模型估计另一参数时，至少需要再多一个观测，越多越好。但是，使用方差参数时，每个方差参数需要多几个观测来获得合理的估计值。估计方差所需的数据比估计均值所需的数据多。

对数线性方差模型是拟合散度效应的很灵活的方法，该方法应该引起我们更多的注意，尽管当前文献中涉及的不多。

启动“对数线性方差”特质

通过以下方式来启动“对数线性方差”特质：选择**分析** > **拟合模型**，输入一个或多个列作为 **Y**，然后从**特质**菜单中选择**对数线性方差**。

图 10.1 已选定 “对数线性方差”的 “拟合模型”启动窗口

有关对所有特质通用的“拟合模型”窗口各方面的详细信息，请参见第 29 页的“模型规格”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。在此提供特定于“对数线性方差”特质的信息。

若您的模型效应有缺失值，可以将这些缺失值视为信息性类别。从“模型规格”红色小三角菜单中选择“信息性缺失”选项。

使用对数线性方差的示例

数据表 InjectionMolding.jmp 包含来自 7 因子 2^{7-3} 部分析因设计的实验结果，该设计添加了四个中心点 (Montgomery 1991)。初步调查已确定响应均值似乎只随前两个因子**铸模温度**和**转速**变化，方差则受**持续时间**影响。

图 10.2 注塑成型数据

	模式	铸模温度	转速	持续时间	喷嘴大小	循环时间	湿度	压力	收缩
1	-------	-1	-1	-1	-1	-1	-1	-1	6
2	+----++	1	-1	-1	-1	1	-1	1	10
3	-+--+-+	-1	1	-1	-1	1	1	-1	32
4	++--++-	1	1	-1	-1	-1	1	1	60
5	--+-++-	-1	-1	1	-1	1	1	1	4
6	+-+-+-+	1	-1	1	-1	-1	1	-1	15
7	-++--++	-1	1	1	-1	-1	-1	1	26
8	+++----	1	1	1	-1	1	-1	-1	60
9	---++++	-1	-1	-1	1	-1	1	1	8
10	+--++--	1	-1	-1	1	1	1	-1	12
11	-+-+-+-	-1	1	-1	1	1	-1	1	34
12	++-+--+	1	1	-1	1	-1	-1	-1	60
13	--++--+	-1	-1	1	1	1	-1	-1	16
14	+-++-+-	1	-1	1	1	-1	-1	1	5
15	-++++--	-1	1	1	1	-1	1	-1	37
16	+++++++	1	1	1	1	1	1	1	52
17	0000000	0	0	0	0	0	0	0	25
18	0000000	0	0	0	0	0	0	0	29
19	0000000	0	0	0	0	0	0	0	24
20	0000000	0	0	0	0	0	0	0	27

1. 选择**帮助** > **样本数据库**，然后打开 InjectionMolding.jmp。
2. 选择**分析** > **拟合模型**。

   由于数据表中的变量已经分配有预选角色，所以分析会自动运行。

图 10.3 “对数线性方差”报表窗口

对数线性方差拟合

拟合统计量

-2 残差对数似然	90.712809
-2 对数似然	87.777035
AICc	106.23857
BIC	105.75143

“收缩”的均值模型

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	27.582516	0.380697	72.45	<.0001*
铸模温度	7.683713	0.392907	19.56	<.0001*
转速	18.673515	0.392907	47.53	<.0001*
铸模温度*转速	5.765297	0.392907	14.67	<.0001*

固定效应检验

“收缩”的方差模型

方差相等的似然比检验

源	-2 残差对数似然
最初方差相等	101.76083063
方差拟合之后	90.712808548
差值: 卡方	11.048022086
自由度	1
概率>卡方	0.0008878186

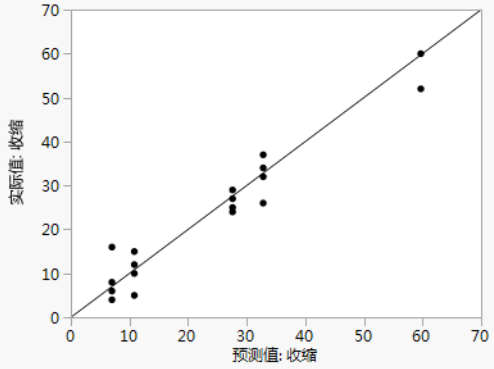
方差参数估计值

项	估计值	标准误差	卡方	概率>卡方	95% 下限	95% 上限
持续时间	1.5788108	0.412603	11.0480	0.0009*	0.7057476	2.37325
残差	6.6912718	2.445524	7.4864	0.0062*	3.2689762	13.696373

方差效应似然比检验

源	检验类型	自由度	卡方	概率>卡方
持续时间	似然	1	11.0480	0.0009*

“预测值-实际值”图



“‘收缩’的均值模型”报表给出均值模型的参数，“‘收缩’的方差模型”报表给出方差模型的参数。

对数线性报表

生成的报表顶部显示期望响应的拟合。第 320 页的““拟合统计量”报表的说明” 说明如何计算拟合统计量。“参数估计值”和“固定效应检验”类似于在标准最小二乘法特质中的报表，尽管它们来自限制最大似然 (REML)。

图 10.4 均值模型输出

对数线性方差拟合

拟合统计量

-2 残差对数似然	90.712809
-2 对数似然	87.777035
AICc	106.23857
BIC	105.75143

“收缩”的均值模型

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	27.582516	0.380697	72.45	<.0001*
铸模温度	7.683713	0.392907	19.56	<.0001*
转速	18.673515	0.392907	47.53	<.0001*
铸模温度*转速	5.765297	0.392907	14.67	<.0001*

固定效应检验

源	参数数目	自由度	分母自由度	F 比	概率>F
铸模温度	1	1	5.758	382.4387	<.0001*
转速	1	1	5.758	2258.768	<.0001*
铸模温度*转速	1	1	5.758	215.3094	<.0001*

图 10.5 方差模型输出

“收缩”的方差模型

方差相等的似然比检验

源	-2 残差对数似然
最初方差相等	101.76083063
方差拟合之后	90.712808548
差值: 卡方	11.048022086
自由度	1
概率>卡方	0.0008878186

方差参数估计值

项	估计值	标准误差	卡方	概率>卡方	95% 下限	95% 上限
持续时间	1.5788108	0.412603	11.0480	0.0009*	0.7057476	2.37325
残差	6.6912718	2.445524	7.4864	0.0062*	3.2689762	13.696373

方差效应似然比检验

源	检验类型	自由度	卡方	概率>卡方
持续时间	似然	1	11.0480	0.0009*

报表的第二部分显示方差模型的拟合。**方差参数估计值**报表显示估计值和相关统计量。提供两个隐藏的列：

- 隐藏的列 exp(Estimate) 是估计值的指数。因此，若将这些因子编码为具有 +1 和 -1 值，则因子的 +1 水平将具有方差与 exp(Estimate) 值相乘所得的值。同样，-1 水平将具有方差与该列倒数相乘所得的值。要查看隐藏的列，可右击报表并从显示的**列**菜单中选择列的名称。
- 若回归变量的范围为 -1 到 +1，则标记为 exp(2|Estimate|) 的隐藏列是较大方差与较小方差之比。

该报表还显示每个估计值的标准误差、卡方、*p* 值和刻画似然置信限。残差参数是方差的总估计值，假设所有其他回归变量都是零。

方差模型拟合是否显著好于原始模型？针对该问题的似然比检验比较了拟合模型和除截距外所有参数均为零的模型（即相等方差的模型）。在本例中，*p* 值是高度显著的。**持续时间**的变化导致方差变化。

方差效应似然比检验通过重新拟合依次不含每项的模型来创建似然比检验。这些检验通常比 Wald 检验更可信。

“对数线性”平台选项

“对数线性方差拟合”菜单包含以下选项：

保存列 在数据表中创建一个或多个列。请参见第 404 页的 “保存列”。

行诊断 标绘行诊断。请参见第 405 页的 “行诊断”。

刻画器 打开刻画器、等高线刻画器或曲面刻画器。请参见 “标准最小二乘法模型”一章中第 142 页的 “因子刻画”。

模型对话框 显示当前分析的已完成的启动窗口。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

保存列

提供以下 “保存列”选项：

预测公式 创建名为**均值**的新列。新列包含由指定模型计算出的均值的预测值。

方差公式 创建名为**方差**的新列。新列包含由指定模型计算出的方差的预测值。

标准差公式 创建名为**标准差**的新列。新列包含由指定模型计算出的标准差的预测值。

残差 创建名为**残差**的新列，它包含残差，即观测的响应值减去预测值所得的值。请参见第 405 页的 “查看残差”。

学生化残差 创建名为**学生化残差**的新列。新列值是残差除以标准误差所得的值。

预测值标准误差 创建名为**预测值标准误差**的新列。新列包含预测值的标准误差。

单值标准误差 创建名为**单值标准误差**的新列。新列包含单个预测值的标准误差。

均值置信区间 创建两个新列：95% **均值下限**和 95% **均值上限**。新列包含预测均值的置信区间的界限。

单值置信区间 创建两个新列：95% **单值下限**和 95% **单值上限**。新列包含单个响应值的置信限。

行诊断

提供以下 “行诊断”选项：

标绘 “预测值 - 实际值”图 针对 *Y* 的预测值标绘其观测值。这是整体模型的杠杆图。

标绘 “预测值 - 学生化残差”图 针对 *Y* 的预测值标绘其学生化残差。

标绘 “行 - 学生化残差”图 针对行标绘学生化残差。

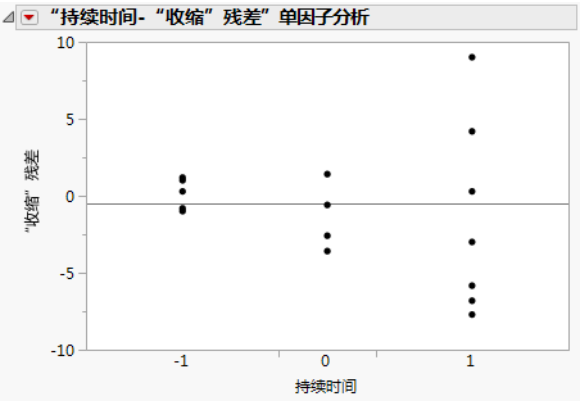
查看残差

要查看散度效应，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 InjectionMolding.jmp。
2. 选择**分析** > **拟合模型**。

   由于数据表中的变量已经分配有预选角色，所以分析会自动运行。
3. 点击 “对数线性方差拟合”红色小三角并选择**保存列** > **残差**。
4. 在 InjectionMolding.jmp 样本数据表中，右击列面板中**持续时间**旁边的连续型图标，然后选择**名义型**。
5. 选择**分析** > **以 X 拟合 Y**。
6. 选择 **“收缩”残差**，然后点击 **Y，响应**。
7. 选择**持续时间**，然后点击 **X，因子**。
8. 点击**确定**。

图 10.6 散度效应 - 残差



在该图中，很容易看出方差随着**持续时间**的增加而增加。这通过将**持续时间**视为名义型因子来实现。

刻画拟合模型

使用**刻画器**、**等高线刻画器**或**曲面刻画器**可进一步了解拟合模型。要选择刻画器选项，可点击“对数线性方差拟合”红色小三角并选择**刻画器**菜单下的一个选项。

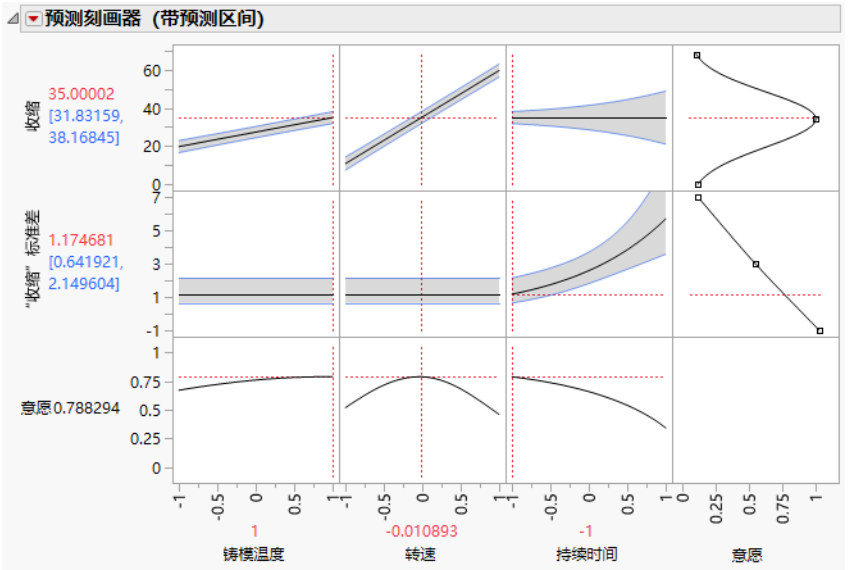
刻画拟合模型的示例

例如，假定目标是找到实现响应的目标值为 35 同时使方差最小的因子设置。拟合模型并从报表菜单中选择 “刻画器”。例如，图 10.7 显示匹配均值的目标值并使方差最小化的刻画器设置。

1. 选择**帮助 > 样本数据库**，然后打开 InjectionMolding.jmp。
2. 选择**分析 > 拟合模型**。

   由于数据表中的变量已经分配有预选角色，所以分析会自动运行。
3. 点击 “对数线性方差拟合”红色小三角并选择**刻画器 > 刻画器**。
4. 点击 “预测刻画器”红色小三角并选择**最优化和意愿 > 设置意愿**。
5. 在显示的 “响应目标”窗口中，将 “最大化”更改为 “匹配目标值”。
6. 点击**确定**。
7. 在第二个 “响应目标”窗口中，点击**确定**。
8. 点击 “预测刻画器”红色小三角并选择**优化和意愿 > 最大化意愿**。
9. 点击 “预测刻画器”红色小三角并选择**预测区间**。

图 10.7 具有预测区间的 “匹配目标值”和 “最小化方差”的刻画器



了解均值和方差（都使用“对数方差”特质建模）之间关系的一种最佳方法是查看有关均值的单值预测置信区间。常规置信区间（在“刻画器”中默认显示）不像单值预测置信区间那样显示有关方差模型的信息。预测区间在一个图中同时显示均值和方差模型。

若 Y 是建模的响应且您希望知道 x_n 处新观测值的预测区间，则：

$$s^2\left|x_n = s^2_Y\right|x_n + s^2_{\hat{Y}}\left|x_n\right.$$

其中：

$s^2\left|x_n\right.$ 是 x_n 处单个预测值的方差

$s^2_Y\left|x_n\right.$ 是 x_n 处 Y 的分布的方差

$s^2_{\hat{Y}}\left|x_n\right.$ 是 $\hat{Y}$ 的抽样分布的方差，也是均值的方差。

因为单个预测值的方差包含 Y 的分布的方差，可以看到 Y 的方差变化的效应。不仅单值预测区间变宽，而且它们可能随着方差效应的变化而更改形状。

第 11 章

Logistic 回归模型

拟合名义型或有序型响应的回归模型

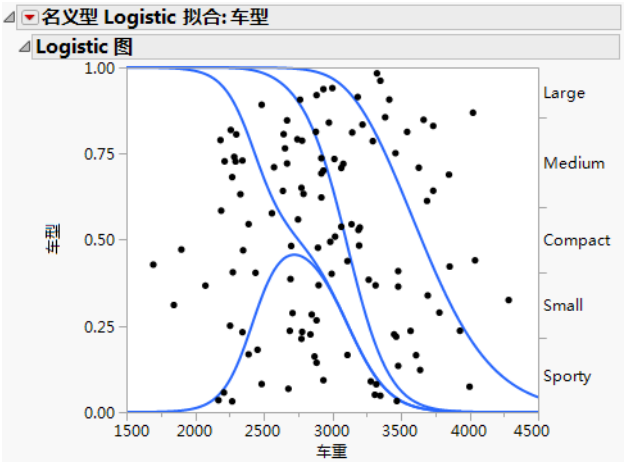
当您的响应变量具有离散值时，您可以使用“拟合模型”平台来拟合 Logistic 回归模型。“拟合模型”平台提供用于拟合 Logistic 回归模型的两个特质。您使用的特质取决于响应列的建模类型（名义型或有序型）。

对于名义型响应变量，“名义型 Logistic”特质通过多水平 Logistic 响应函数来拟合线性模型。

对于有序型响应变量，“有序型 Logistic”特质通过线性模型的 Logistic 分布函数拟合累积响应概率。

这两个特质都提供模型的似然比检验、混杂矩阵、ROC 和提升曲线。响应为二值响应时，“名义型 Logistic”特质还提供优势比（具有相应的置信区间）。

图 11.1 名义型 Logistic 回归模型的 Logistic 图



目录

“名义型 Logistic”和“有序型 Logistic”特质概述 411
名义型 Logistic 回归 411
有序型 Logistic 回归 411
拟合 Logistic 回归模型的其他 JMP 平台 411
Logistic 回归的示例 412
名义型 Logistic 回归的示例 412
有序型 Logistic 回归的示例 414
启动“名义型 Logistic”和“有序型 Logistic”特质 416
Logistic 回归模型中的验证 417
Logistic 拟合报表 417
整体模型检验 418
拟合详细信息 419
“失拟”检验 419
“Logistic 拟合”平台选项 420
名义型和有序型拟合的选项 420
名义型拟合的选项 421
有序型拟合的选项 422
Logistic 回归的其他示例 423
“逆预测”示例 423
将“效应汇总”用于名义型 Logistic 模型的示例 425
二次有序型 Logistic 模型的示例 427
堆叠多个列中的计数的示例 429
“名义型 Logistic”和“有序型 Logistic”特质的统计详细信息 430
Logistic 回归模型 430
优势比 431
统计检验的关系 432

“名义型 Logistic”和“有序型 Logistic”特质概述

Logistic 回归将分类 *Y* 响应变量的各个水平概率建模为一个或多个 *X* 效应的函数。“拟合模型”平台提供用于拟合 Logistic 回归模型的两个特质。您使用的特质取决于响应列的建模类型（名义型或有序型）。

有关拟合 Logistic 回归模型的详细信息，请参见 Walker and Duncan (1967)、Nelson (1976)、Harrell (1986) 和 McCullagh and Nelder (1989)。

有关 Logistic 回归模型的参数化的详细信息，请参见第 430 页的“Logistic 回归模型”。

名义型 Logistic 回归

当响应变量具有名义型建模类型时，平台使用最大似然通过多水平 Logistic 响应函数来拟合线性模型。因此，除一个响应水平外，所有响应水平均由一个 Logistic 曲线建模，该曲线表示给定 *X* 效应值时响应水平的概率。最终响应水平的概率为 1 减去其他拟合概率之和。因此，在所有 *X* 效应值处，响应水平的拟合概率之和为 1。

若响应变量为二值响应，可以在“拟合模型”窗口中设置“目标水平”来指定您要对其概率建模的水平。默认情况下，模型估计响应变量的较高水平的概率。

有关为名义型响应变量拟合模型的详细信息，请参见“统计详细信息”一章中第 453 页的“名义型响应”。

有序型 Logistic 回归

当响应变量具有有序型建模类型时，平台使用最大似然通过线性模型的 Logistic 函数拟合累积响应概率。因此，由曲线对每个响应水平处或低于该水平的累积概率建模。这些曲线对于每个水平是相同的，只是向右或向左平移了一段距离。

提示：若有很多响应水平，则有序型模型与名义型模型相比，拟合速度更快，占用的内存更少。

有关使用有序型响应变量拟合模型的详细信息，请参见“统计详细信息”一章中第 455 页的“有序型响应”。

拟合 Logistic 回归模型的其他 JMP 平台

JMP 中有很多其他平台可用于拟合 Logistic 回归模型：

- 要拟合具有单个连续主效应的 Logistic 回归模型，您可以使用“以 X 拟合 Y”平台查看每个效应的累积 Logistic 概率图。请参见《基本分析》。
- 要执行 Logistic 回归模型中的变量选择，您可以使用“拟合模型”平台的“逐步”特质。请参见第 217 页的“逐步回归模型”一章。

- 要拟合使用连结函数而不是 Logit 连结的 Logistic 回归模型，您可以使用“拟合模型”平台的“广义线性模型”特质。请参见第 433 页的“广义线性模型”一章。
- JMP PRO 要执行 Logistic 回归模型中的变量选择并拟合惩罚 Logistic 回归模型，您可以使用“拟合模型”平台的“广义回归”特质。请参见第 247 页的“广义回归模型”一章。

Logistic 回归的示例

本节包含两个示例，分别针对“拟合模型”平台中的两个 Logistic 回归特质（“名义型”和“有序型”）：

- 第 412 页的“名义型 Logistic 回归的示例”
- 第 414 页的“有序型 Logistic 回归的示例”

名义型 Logistic 回归的示例

针对金属锭执行一个实验，采用不同加热时间和浸泡时间准备这些金属锭，然后检验金属锭是否可以滚压。请参见 Cox and Snell (1989)。数据包含在 Ingots.jmp 样本数据表中。在本示例中，“拟合模型”平台使用 Logistic 回归模型（回归变量为**加热时间**和**浸泡时间**）拟合“可用性”响应的概率。

1. 选择**帮助 > 样本数据库**，然后打开 Ingots.jmp。

   分类变量**可用性**的值 Ready 和 Not Ready 指示金属模具是否已准备好滚压。
2. 选择**分析 > 拟合模型**。
3. 选择**可用性**，然后点击 Y。

   因为您选择了具有“名义型”建模类型的列，“拟合模型”特质更新为“名义型 Logistic”。

   因为**可用性**是只有两个水平的名义型列，将显示“目标水平”选项。该选项支持您指定您要为其概率建模的响应水平。
4. 从“目标水平”列表中，选择“可用性”。

   在该模型中，“目标水平”为“可用性”，因此您要对“可用性”响应的概率建模。
5. 选择**加热时间**和**浸泡时间**，然后点击**添加**。
6. 选择**计数**并点击**频数**。
7. 点击**运行**。

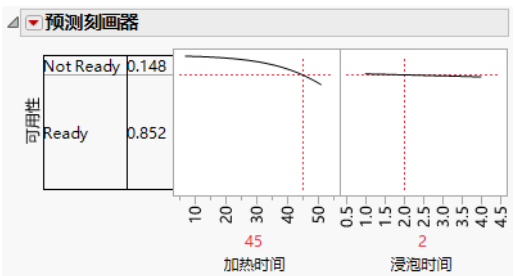
   拟合过程收敛时，将显示名义型回归报表。

图 11.2 名义型 Logistic 拟合报表

在“整体模型检验”报表中，卡方统计量 (11.64) 的 p 值很小 (0.0030)，这表示整体模型是显著的。但是，**浸泡时间**的参数估计值的 p 值为 0.8639，这表示浸泡时间在统计上不显著。

8. 点击“‘可用性’的名义型 Logistic 拟合”旁边的红色小三角并选择**刻画器**。

图 11.3 预测刻画器



当**加热时间**设置为 45 并且**浸泡时间**设置为 2 时，**可用性**的概率为 0.85。

现在，您还可能想使用逆预测查找在特定浸泡时间和指定可以滚压的特定概率时的加热时间。有关本示例的继续，请参见第 423 页的“ “逆预测”示例”。

有序型 Logistic 回归的示例

执行了一个实验来检验各种奶酪添加剂 （A 到 D）是否对奶酪口味有影响。通过品尝小组评定口味并采用从 1 （很不喜欢）到 9 （很喜欢）的顺序尺度记录结果。请参见 McCullagh and Nelder (1989)。数据位于 Cheese.jmp 样本数据表中。

1. 选择**帮助** > **样本数据库**，然后打开 Cheese.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**响应**，然后点击 Y。

   因为您选择了具有“有序型”建模类型的列，“拟合模型”特质更新为“有序型 Logistic”。
4. 选择**奶酪**，然后点击**添加**。
5. 选择**计数**并点击**频数**。
6. 点击**运行**。

图 11.4 有序型 Logistic 拟合报表

有序型 Logistic 拟合: 响应

频数: 计数

整体模型检验

模型	-对数似然	自由度	卡方	概率>卡方
差值	74.22695	3	148.4539	<.0001*
完全	355.67395			
简化	429.90090			

R 方 (U)	0.1727
AICc	734.695
BIC	770.061
观测数（或权重和）	208

拟合详细信息

失拟

源	自由度	-对数似然	卡方
失拟	21	10.15410	20.30819
饱和	24	345.51986	概率>卡方
拟合	3	355.67395	0.5018

参数估计值

项	估计值	标准误差	卡方	概率>卡方
截距[1]	-4.6051335	0.4267239	116.46	<.0001*
截距[2]	-3.5499488	0.3073868	133.37	<.0001*
截距[3]	-2.4503882	0.2398702	104.36	<.0001*
截距[4]	-1.381774	0.2001699	47.65	<.0001*
截距[5]	-0.0455233	0.1803299	0.06	0.8007
截距[6]	0.90648953	0.1886609	23.09	<.0001*
截距[7]	2.408151	0.2380517	102.34	<.0001*
截距[8]	3.96800057	0.3466551	131.02	<.0001*
奶酪[A]	-0.8622328	0.2289236	14.19	0.0002*
奶酪[B]	2.48960592	0.2703151	84.82	<.0001*
奶酪[C]	0.8476542	0.2280359	13.82	0.0002*

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
奶酪	3	3	148.453899	<.0001*

本示例中的模型拟合将仅包含截距的 **–对数似然** 429.9 减小到完整模型时的 355.67。这个减小导致整体模型的似然比卡方统计量为 148.45（自由度为 3）。因此，观测到的奶酪口味差异非常显著。

最喜欢的奶酪添加剂为具有最大负参数估计值的奶酪添加剂。奶酪 [D] 未显示在“参数估计值”报表中，因为它没有自己的设计矩阵列。但是，奶酪 D 的效应可以计算为其他奶酪效应之和的负数，如表 11.1 中所示。

表 11.1 Cheese.jmp 中奶酪添加剂的偏好程度

奶酪	估计值	偏好程度
A	-0.8622	第 2 喜欢
B	2.4896	最不喜欢
C	0.8477	第 3 喜欢
D	-2.4750	最喜欢

与名义型 Logistic 模型的比较

“失拟”报表显示模型是否与数据很好拟合的一个检验。

作为一个有序型问题，前八个响应水平各具有一个截距，以及**奶酪**的四个水平只有三个参数。因此，在该有序型模型中自由度为 3。该有序型模型为“失拟”检验中的拟合模型。

作为一个名义型问题，前八个响应水平具有截距，但是**奶酪**的四个水平只有三个参数。因此，在名义型模型中的自由度为 8 x 3 = 24。因此，名义型模型是“失拟”检验中的“饱和”模型。

在本示例中，有序型模型的“失拟”检验恰好针对名义型模型检验有序型响应模型。失拟的非显著性导致认定有序型模型是合理的。

启动“名义型 Logistic”和“有序型 Logistic”特质

通过选择**分析 > 拟合模型**并为 Y 输入一个或多个非连续列，启动 “名义型 Logistic”和 “有序型 Logistic”特质。若为 Y 输入多个列，则拟合每个响应的模型。

图 11.5 选择了 “名义型 Logistic”的 “拟合模型”启动窗口

有关对所有特质通用的“拟合模型”窗口各方面的详细信息，请参见第 29 页的“模型规格”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。这里提供了有关“名义型 Logistic”和“有序型 Logistic”特质的特有信息。

若您的模型效应有缺失值，可以将这些缺失值视为信息性类别。从“模型规格”红色小三角菜单中选择“信息性缺失”选项。

要指定没有截距项的模型，请在“拟合模型”窗口的“构造模型效应”面板中选择“无截距”选项。“无截距”选项不可用于“有序型 Logistic”特质。

Logistic 回归模型中所关注的事件由“拟合模型”窗口中的“目标水平”选项定义。仅当在“名义型 Logistic”特质中指定二值响应列时该选项才可用。

注意：“拟合模型”平台中的“Logistic”特质要求您的数据采用堆叠格式，以便所有响应位于一列中。有时，您的数据采用多列格式。有关将多列中的响应转换为单列响应的示例，请参见第 429 页的“堆叠多个列中的计数的示例”。

JMP PRO Logistic 回归模型中的验证

验证是使用数据集的一部分估计模型参数并使用另一部分评估模型预测能力的过程。

- **训练**集用于估计模型参数。
- **验证**集用在模型拟合中，用于评估或验证模型的预测能力。
- **测试**集是对模型预测能力的最终独立评估。仅当使用验证列时才能使用测试集。

将训练集、验证集和测试集创建为原始数据的子集。通过使用“拟合模型”启动窗口中的验证列来完成这项工作。

验证列的值决定如何拆分数据以及使用什么方法进行验证：

- 若列有两个非重复值，则创建训练集和验证集。
- 若列有三个非重复值，则创建训练集、验证集和测试集。
- 若列有三个以上的非重复值或仅有一个非重复值，则不执行验证。

使用验证列时，针对“拟合详细信息”报表中的训练集、验证集和测试集给出模型拟合统计量。还为每个训练集、验证集和测试集提供单独的 ROC 曲线、提升曲线和混杂矩阵。

有关如何在 JMP 建模平台中使用“验证”列的详细信息，请参见《预测和专业建模》。

Logistic 拟合报表

当您使用“名义型 Logistic”或“有序型 Logistic”特质拟合模型时，分别得到“名义型 Logistic 拟合”报表或“有序型 Logistic 拟合”报表。默认情况下，这些报表包含以下报表：

效应汇总 一个交互式报表，该报表支持您在模型中添加或删除效应。请参见“标准最小二乘法模型”一章中第 158 页的 “‘效应汇总’报表”。

Logistic 图 （仅当您的模型包含单个连续效应时才可用。）Logistic 概率图说明了 Logistic 模型的拟合程度。在水平轴上的每个值处，将垂直方向上的概率尺度划分为每个响应类别的概率。用两个曲线之间的垂直距离表示这些概率大小，且所有响应类别的概率之和为 1。

Logistic 图中的点表示数据表中的观测。每个点的水平位置由其连续因子的值决定。每个点的垂直位置被随机选择在对应于其响应类别值的曲线之间。因为使用了固定随机种子，所以在同一模型的多个拟合中，垂直位置没有差异。

迭代 （仅在“名义型 Logistic”特质中可用。）启动“拟合模型”后，开始执行迭代估计过程，并按迭代顺序进行报告。拟合过程完成后，您可以打开“迭代历史记录”报表并查看迭代步骤。

整体模型检验 显示用于比较整体模型拟合与忽略除截距参数之外的所有回归变量效应的模型的检验。该检验类似于连续响应的“方差分析”表。有关“整体模型检验”报表的详细信息，请参见第 418 页的 “整体模型检验”。

拟合详细信息 显示模型的各种拟合测度。请参见第 419 页的 “拟合详细信息”。

失拟 （仅当有关于 *X* 效应的重复点且模型未饱和时可用。）显示失拟检验（也称为拟合优度检验），用于解决模型中是否需要更多项的问题。请参见第 419 页的 “失拟”检验”。

参数估计值 显示参数估计值、标准误差和相关的假设检验。“估计值的协方差”报表给出参数估计值的方差和协方差。

注意：仅对名义型响应变量显示“估计值的协方差”报表，有序型响应变量不显示该报表。

奇异性详细信息 （仅当模型项之间存在线性相依性时才显示。）显示包含模型项满足的线性函数的报表。

效应似然比检验 似然比卡方检验计算为完全模型和受检验假设限定的模型的对数似然差值的两倍。限定的模型是不包含该效应的模型。这些检验可能需要一些时间，因为每个检验需要进行单独的一组迭代。

注意：若预计可在 20 秒之内完成似然比检验，则似然比检验为平台默认值。否则，默认效应检验为 Wald 检验。

整体模型检验

“整体模型检验”表显示用于比较整体模型拟合与忽略除截距参数之外的所有回归参数的模型的检验。该检验类似于连续响应的“方差分析”表。负对数似然对应于平方和，卡方检验对应于 *F* 检验。

“整体模型检验”表显示以下量：

模型 列出模型标签。

差值 **完全**模型与**简化**模型之间的差值。该模型用于测量回归变量作为一个整体对拟合的显著性。

完全 完整的模型包括截距和所有效应。

简化 仅包含截距参数的模型。

– 对数似然 相应模型的负对数似然。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。

自由度 完全与简化模型的差值的自由度 (DF)。

卡方 假设所有回归参数均为零的似然比卡方检验统计量。该检验统计量的值是拟合模型和仅包含截距的简化模型之间的负对数似然差值的两倍。

概率 > 卡方 若指定的模型拟合不比仅包含截距的模型好，此时获得更大卡方值的概率。

R 方 (U) 归因于模型拟合的总不确定性的比例，被定义为差值与简化负对数似然值之比。R 方的范围为零（拟合无改进）到 1（完全拟合）。R 方 (U) 值为 1 指示预测的事件发生的概率

等于 1：预测概率不存在不确定性。因为预测概率的确定性对于 Logistic 模型很少见，R 方 (U) 通常很小。

R 方 (U) 有时表示为 U（不确定性系数）或 McFadden 的伪 R^2。

AICc 校正的 Akaike 信息准则。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。

BIC Bayesian 信息准则。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。

观测数（或权重和） 样本中的总观测数。若在“拟合模型”窗口中指定了“频数”或“权重”列，则该值是分配给“频数”或“权重”角色的列值之和。

拟合详细信息

“拟合详细信息”报表包含以下统计量：

熵 R 方 等价于 R 方 (U)。请参见第 418 页的“整体模型检验”。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。

-Log p 均值 -log(p) 的平均值，其中 p 是与发生的事件关联的拟合概率。

RASE 平均平方根误差，其中差值为响应和 p（实际发生事件的拟合概率）之间的差值。

绝对偏差的均值 响应和 p（实际发生事件的拟合概率）的差值绝对值的平均值。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。

数目 观测数。

对于熵 R 方和广义 R 方，值越接近 1 表示拟合效果越好。对于 -Log p 均值、RASE、绝对偏差的均值和误分类率，值越小表示拟合效果越好。

要检验效应作为整体是显著的（“整体模型”检验），计算卡方统计量，它是拟合模型与仅包含截距的简化模型的负对数似然差值的两倍。

若指定了验证列，则“拟合详细信息”报表将包含每个训练集、验证集和测试集的列。

“失拟”检验

“失拟”检验解决当前模型中是否有足够信息或是否需要更复杂的项的问题。该检验有时称为拟合优度检验。失拟检验通过为数据中的模型效应值的每个组合构造类别来计算纯误差负对数似然。“失拟”表中的**饱和**行包含该对数似然。“失拟”报表还包含**饱和**对数似然是否显著好于**拟合**模型的检验。

饱和自由度为 m–1，其中 m 是唯一总体数。**拟合**自由度是不包括截距的参数数目。

“失拟”表包含由于**失拟**导致的误差、**饱和**模型中的误差（纯误差）以及**拟合**模型中的总误差的负对数似然。卡方统计量用于检验失拟。

“Logistic 拟合”平台选项

- “名义型和有序型拟合的选项”
- “名义型拟合的选项”
- “有序型拟合的选项”

名义型和有序型拟合的选项

以下选项在 “名义型 Logistic 拟合”和 “有序型 Logistic 拟合”红色小三角菜单中均可用：

Logistic 图 （仅当您的模型包含单个连续效应时才可用。）显示或隐藏 “Logistic 图”报表。请参见第 417 页的 “Logistic 拟合报表”。

似然比检验 显示或隐藏 “效应似然比检验”报表。似然比卡方检验计算为完全模型和受检验假设限定的模型的对数似然差值的两倍。限定的模型是不包含该效应的模型。这些检验可能需要一些时间，因为每个检验需要进行单独的一组迭代。因此，对于大问题可能需要很长时间检验。

注意：若预计可在 20 秒之内完成似然比检验，则似然比检验为平台默认值。强烈建议使用该默认选项。

Wald 检验 显示或隐藏 “效应 Wald 检验”报表。 Wald 卡方是似然比检验的二次近似，它是计算的副产品。尽管 Wald 检验被认为可信度较低，但是它们提供了筛选效应的合适显著性指示符。Wald 检验显示每个参数估计值和效应。若预计似然比检验需要 20 秒以上的时间完成，则这是默认检验。

置信区间 显示或隐藏模型参数的刻画似然置信区间。您可以通过在 “拟合模型”窗口的 “模型规格”红色小三角菜单中选择 “设置 Alpha 水平”来更改置信水平。在模型拟合中每个置信限需要进行一组迭代，可能需要很长时间来计算。此外，有时并不能成功找到置信限。

ROC 曲线 显示或隐藏模型的 ROC 曲线。受试者操作特征 (ROC) 曲线测量模型的拟合概率的排序效率以排序响应水平。 ROC 曲线还可以帮助设置诊断检验中的准则点。曲线越高于对角线，拟合效果越好。有关 ROC 曲线的简介，请参见 《基本分析》。

若 Logistic 拟合具有两个以上的响应水平，它生成广义 ROC 曲线（与 “分割”平台中的曲线相同）。在此类图中，每个响应水平有一条曲线，它是该水平相对所有其他水平的 ROC 曲线。请参见 《预测和专业建模》。

若指定了验证列，则为每个训练集、验证集和测试集都显示一条 ROC 曲线。

提升曲线 显示或隐藏模型的提升曲线。提升曲线显示与 ROC 曲线相同的信息，但是它在开头就突出显示排序的丰富程度。垂直轴显示总体比例在所选响应水平中的丰富程度与该响应水

平作为整体的比率之比。若指定了验证列，则为每个训练集、验证集和测试集都显示一条提升曲线。有关提升曲线的详细信息，请参见 《预测和专业建模》。

混杂矩阵 显示或隐藏混淆统计量报表，其中包含“混淆矩阵”报表和“混淆率”报表。两个报表显示实际响应水平和预测响应水平的双向分类。预测响应水平是在启动窗口中指定的“目标水平”。“混淆率”报表等同于“混淆矩阵”报表，只不过其中的数字要除以行合计。

对于好的模型，预测响应水平应与实际响应水平相同。“混淆矩阵”报表提供一种评估预测响应与实际响应相符程度的方法。若指定了验证列，则为每个训练集、验证集和测试集都显示一个混杂矩阵。

若响应为名义型并且具有“收益矩阵”列属性，选定该选项时也会显示“决策矩阵”报表。有关“决策矩阵”报表的详细信息，请参见 《预测和专业建模》。

刻画器 显示或隐藏预测刻画器，显示当模型中因子的值变化时指定响应概率的拟合值。该功能可用于名义型和有序型响应。有关红色小三角菜单中选项的详细信息，请参见 《刻画器指南》。

等高线刻画器 （仅当模型包含不止一个连续因子时才可用。）显示或隐藏 “等高线刻画器”。有关红色小三角菜单中选项的详细信息，请参见 《刻画器指南》。

模型对话框 显示当前分析的已完成的启动窗口。

效应汇总 显示或隐藏 “效应汇总”报表，该报表支持您以交互方式更新模型中的效应。请参见第 417 页的 “Logistic 拟合报表”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

名义型拟合的选项

以下选项在 “名义型 Logistic 拟合”红色小三角菜单中可用：

图选项 （仅当您的模型包含单个连续效应时才可用。）“图选项”菜单包含以下选项：

显示点 将“Logistic 图”中的点切换为开或关。

显示比率曲线 允许您将效应变量各个值处的比率与拟合的 Logistic 曲线进行比较。仅当每个效应值有几个点时该选项才有用。在这种情况下，您获取每个值处的比率的合理估计值，然后将该比率与拟合的 Logistic 曲线进行比较。为了防止出现过多的退化点（通常在 0 或 1 处），JMP 仅在 x 值处至少有三个点时才显示该比率值。

线条颜色 指定图曲线的颜色。

优势比 显示或隐藏“优势比”报表，它包含“单位优势比”和“范围优势比”。

图 11.6 优势比

优势比

用于 Ready 对 Not Ready 的“可用性”优势比

单位优势比

回归变量中的单位变化

项	优势比	下限 95%	上限 95%	倒数
加热时间	0.921244	0.87937	0.965111	1.0854892
浸泡时间	0.94481	0.493646	1.808313	1.0584137

范围优势比

整个范围上回归变量中的变化

项	优势比	下限 95%	上限 95%	倒数
加热时间	0.027069	0.003496	0.209605	36.942229
浸泡时间	0.8434	0.120295	5.913179	1.185677

优势比的检验及其置信区间基于 Wald。

逆预测 （仅适用于二水平名义型响应。）查找具有指定概率的 x 值。有关逆预测的详细信息，请参见附录第 451 页的“统计详细信息”。

保存概率公式 在当前数据表中创建列，这些列包含响应水平线性组合的公式、响应水平的预测公式以及给出最可能的响应的预测公式。

对于具有 r 个水平的名义型响应模型， JMP 创建以下列：

- 名为 Lin[j] 的列，这些列包含针对响应水平 $j = 1, 2, \ldots r - 1$ 的回归变量的线性组合
- 名为 Prob[r] 的列，该列包含拟合到最后水平 r 的公式
- 名为 Prob[j]（对于 $j < r$）的列，该列包含拟合到水平 j 的公式
- 名为**最可能的“响应名称”**的列，该列基于计算的概率选择每行最可能的水平。

JMP PRO **发布概率公式** 创建概率公式并在“公式存储库”平台中将它们保存为公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见《预测和专业建模》。

指标参数化估计值 （仅当模型效应之间存在名义型列时可用。）显示或隐藏“指标函数参数化”报表，该报表列出模型的参数估计值，而模型的名义型列使用指标 (SAS GLM) 参数化进行编码，当作连续列来处理。有序型列仍使用通常的 JMP 编码方案保持编码。“统计详细信息”附录中第 456 页的“因子模型”对 SAS GLM 和 JMP 编码方案进行了说明。

警告：“指标函数参数化”报表中给出的标准误差和卡方值不同于“参数估计值”报表中给出的相应值。这是因为这些估计值所估计的模型参数不同。

有序型拟合的选项

以下选项在“有序型 Logistic 拟合”红色小三角菜单中可用：

保存 包含以下“保存”选项：

保存概率公式 在当前数据表中创建列，这些列包含响应水平线性组合的公式、响应水平的预测公式以及给出最可能的响应的预测公式。

对于具有 r 个水平的有序型响应模型，JMP 创建以下列：

- 名为**线性**的列，该列包含不带截距项的回归变量的线性组合公式
- 名为 Cum[j] 的列，每列包含响应小于等于水平 j 的累积概率公式（对于水平 $j = 1, 2, \ldots r - 1$）。没有对于所有行均为 1 的 Cum[j = 1, 2, ... r - 1]
- 名为 Prob[j = 1, 2, ... r - 1]（对于 $1 < j < r$）的列，每列包含响应为水平 j 的概率公式。Prob[j] 是 Cum[j] 与 Cum[j –1] 的差值。Prob[1] 是 Cum[1]，Prob[r] 是 1 – Cum[r – 1]。
- 名为**最可能的“响应名称”**的列，该列基于计算的概率选择每行最可能的水平。

JMP PRO **发布概率公式** 创建概率公式并在“公式存储库”平台中将它们保存为公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见《预测和专业建模》。

保存分位数 （仅当响应为数值并且具有有序型建模类型时才可用。）在当前数据表中创建名为 **5 分位数（顺序）**、**50 分位数（顺序）**和 **95 分位数（顺序）**的列，它们拟合这三个概率的分位数。

保存期望值 （仅当响应为数值并且具有有序型建模类型时才可用。）在当前数据表中创建名为**期望值（顺序）**的列。该列包含具有每行的拟合响应概率的响应值的线性组合，并给出期望值。

Logistic 回归的其他示例

本节包含使用“拟合模型”平台中的 Logistic 回归的更多示例。

- “逆预测”示例”
- “将“效应汇总”用于名义型 Logistic 模型的示例”
- “二次有序型 Logistic 模型的示例”
- “堆叠多个列中的计数的示例”

“逆预测”示例

逆预测允许您预测在响应的指定概率处一个自变量的值。若有多个自变量，则必须指定除您要预测的自变量之外的所有自变量值。

针对金属锭执行一个实验。采用不同加热时间和浸泡时间准备这些金属锭，然后检验金属锭是否可以滚压。请参见 Cox and Snell (1989)。在 JMP 中，数据包含在 Ingots.jmp 样本数据表中。

您想预测在可以滚压的概率分别为 0.8 和 0.9、浸泡时间为 2 时的加热时间。

1. 要进行分析，执行第 412 页的“名义型 Logistic 回归的示例”中的步骤。
2. 点击“‘可用性’的名义型 Logistic 拟合”旁边的红色小三角并选择**逆预测**。
3. 删除**加热时间**的值。

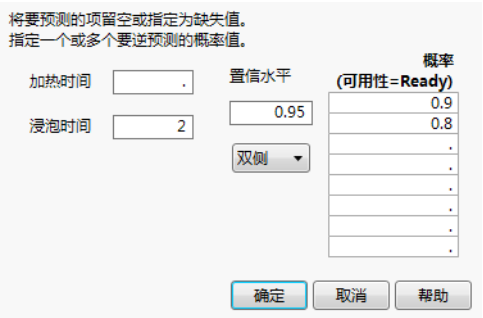
   因为您**要找到加热时间**的预测值，因此将**加热时间**值留空。

4. 为**浸泡时间**输入 2。

   您要预测**浸泡时间**为 2 时的加热时间。

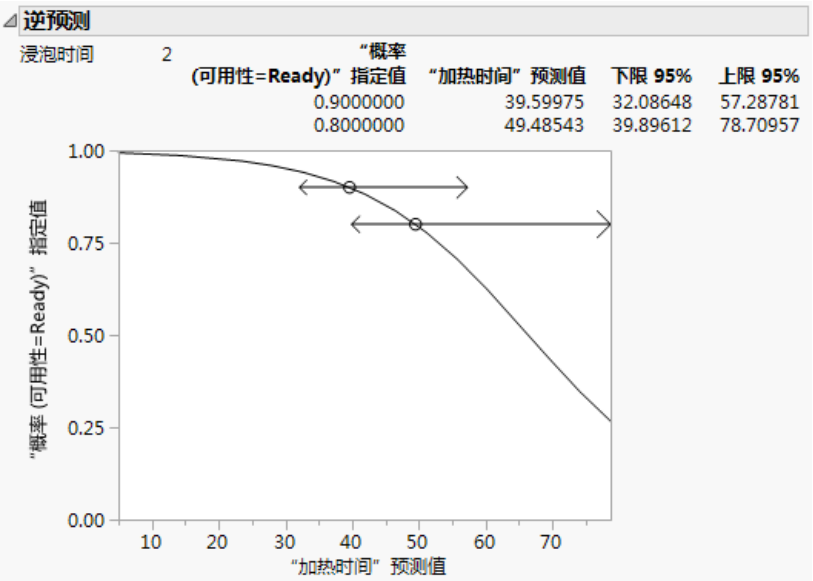
5. 在“概率”下，在前两行中输入 0.9 和 0.8。

图 11.7 “逆预测规格”窗口



6. 点击**确定**。

图 11.8 “逆预测”报表



当**浸泡时间**为 2 时，若金属锭有 90% 的机会可以滚压，**加热时间**的预测值为 39.60，其 95% 置信区间为 32.09 到 57.29。当**浸泡时间**为 2 时，若金属锭有 80% 的机会可以滚压，**加热时间**的预测值为 49.49，其 95% 置信区间为 39.90 到 78.71。

将“效应汇总”用于名义型 Logistic 模型的示例

执行一个市场研究来评估对某品牌洗涤剂的偏好程度。请参见 Ries and Smith (1963)。研究结果位于 Detergent.jmp 样本数据表中。根据以下项定义模型：

- 响应变量**品牌**，具有值 m 和 x
- 名为**柔软度**（水柔软度）的效应，具有值“soft”、“medium”和“hard”
- 名为**先前使用过**的效应，具有值“yes”和“no”
- 名为**温度**的效应，具有值“high”和“low”
- 计数变量**计数**，它指定每个效应类别组合的频数统计。

该研究首先指定完全三因子析因模型。

1. 选择**帮助** > **样本数据库**，然后打开 Detergent.jmp。
2. 选择**分析** > **拟合模型**。
3. 从“选择列”列表中选择**品牌**，然后点击 Y。

   因为您指定了一个名义型响应变量，因此“特质”变为“名义型 Logistic”。

   因为**品牌**是只有两个水平的“名义型”列，因此显示“目标水平”选项。该选项支持您指定您要为其概率建模的响应水平。
4. 从“目标水平”列表中，选择“m”。
5. 选择**计数**并点击**频数**。
6. 选择**柔软度**到**温度**，然后点击**宏** > **完全析因**。
7. 点击**运行**。

图 11.9 三因子析因模型的名义型 Logistic 拟合

名义型 Logistic 拟合: 品牌

效应汇总

梯度收敛,3 次迭代

频数: 计数

迭代

整体模型检验

模型	-对数似然	自由度	卡方	概率>卡方
差值	16.41281	11	32.82562	0.0006*
完全	682.24780			
简化	698.66061			

R 方 (U)	0.0235
AICc	1388.81
BIC	1447.48
观测数（或权重和）	1008

拟合详细信息

参数估计值

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
柔软度	2	2	0.09804239	0.9522
先前使用过	1	1	22.1316677	<.0001*
柔软度*先前使用过	2	2	3.78609668	0.1506
温度	1	1	3.63914017	0.0564
柔软度*温度	2	2	0.19617686	0.9066
先前使用过*温度	1	1	2.26089203	0.1327
柔软度*先前使用过*温度	2	2	0.73731691	0.6917

“整体模型检验”报表显示三因子完全析因模型在整体上是显著的（概率 > 卡方 = 0.0006）。

“效应似然比检验”报表显示包含**柔软度**的效应对模型拟合贡献不显著。这导致您考虑从模型中删除**柔软度**。可以从“效应汇总”报表中执行该操作。

8. 在“效应汇总”报表中，在“源”下选择**柔软度 * 先前使用过**到**柔软度**，然后点击**删除**。

   将更新报表以显示两因子析因模型（图 11.10）。“整体模型检验”报表显示两因子模型在整体上也是显著的。

图 11.10 两因子析因模型的名义型 Logistic 拟合

名义型 Logistic 拟合: 品牌

梯度收敛,3 次迭代

频数: 计数

迭代

整体模型检验

模型	-对数似然	自由度	卡方	概率>卡方
差值	13.58479	3	27.16958	<.0001*
完全	685.07582			
简化	698.66061			

R 方 (U)	0.0194
AICc	1378.19
BIC	1397.81
观测数（或权重和）	1008

拟合详细信息

参数估计值

项	估计值	标准误差	卡方	概率>卡方
截距	0.03043265	0.0666827	0.21	0.6481
先前使用过[no]	-0.3148498	0.0666827	22.29	<.0001*
温度[low]	-0.1331196	0.0666827	3.99	0.0459*
先前使用过[no]*温度[low]	0.11106673	0.0666827	2.77	0.0958

对数优势比: m/x

估计值的协方差

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
先前使用过	1	1	22.6511053	<.0001*
温度	1	1	3.99625478	0.0456*
先前使用过*温度	1	1	2.78794934	0.0950

您得出结论：先前使用过洗涤剂品牌和水温对洗涤剂偏好有影响。您还发现“温度”和“先前使用过”之间的交互作用在统计上不显著，因此没有证据表明温度依赖于“先前使用过”。

二次有序型 Logistic 模型的示例

有序型 Logistic 特质可以拟合二次曲面来优化更高或更低水平的有序型响应的概率。

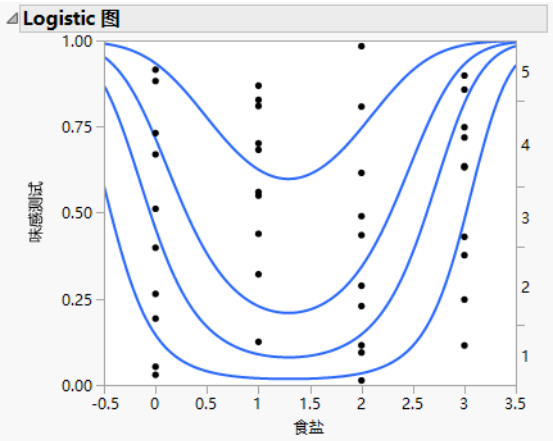
在本示例中，某微波爆米花制造商想知道消费者希望爆米花的盐含量是多少。为此，该厂商查找作为爆米花包装中添加的盐量函数的满意响应的最大概率。在本实验中，将盐含量控制为 0、1、2 和 3 茶匙。评估者按 1（低）到 5（高）的量表评定口味。最佳盐含量是使更满意响应的概率最大化的量。每个盐水平对应的 10 个观测值显示在表 11.2 中。

表 11.2 爆米花的盐含量

盐含量	盐量评定响应
无盐	1, 3, 2, 4, 2, 2, 1, 4, 3, 4
1 茶匙	4, 5, 3, 4, 5, 4, 5, 5, 4, 5
2 茶匙	4, 3, 5, 1, 4, 2, 5, 4, 3, 2
3 茶匙	3, 1, 2, 3, 1, 2, 1, 2, 1, 2

1. 选择**帮助** > **样本数据库**，然后打开 Salt in Popcorn.jmp。
2. 选择**分析** > **拟合模型**。
3. 从 “选择列”列表中选择**味感测试**，然后点击 Y。
4. 从 “选择列”列表中选择**食盐**，然后点击**宏** > **响应曲面**。
5. 点击**运行**。
6. 点击 “响应曲面”旁边的展开图标以打开报表。

图 11.11 Salt in Popcorn.jmp 的有序型 Logistic 拟合



Logistic 图

整体模型检验

拟合详细信息

失拟

参数估计值

项	估计值	标准误差	卡方	概率>卡方
截距[1]	-4.8055768	1.0159898	22.37	<.0001*
截距[2]	-3.2214091	0.8058945	15.98	<.0001*
截距[3]	-2.1154113	0.6922969	9.34	0.0022*
截距[4]	-0.3879895	0.6531433	0.35	0.5525
食盐	0.56374559	0.2751412	4.20	0.0405*
(食盐-1.5)*(食盐-1.5)	1.34994322	0.3514837	14.75	0.0001*

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
食盐	1	1	4.60818105	0.0318*
食盐*食盐	1	1	17.9473211	<.0001*

响应曲面

系数

	食盐	味感测试
食盐	1.3499432	0.5637456

解

变量	临界值
食盐	1.2911966

解是 最小值

典型曲率

该报表显示二次模型如何拟合响应概率。曲线（而非偏移的 Logistic 曲线）变为叠堆的 U 形曲线，其中每条曲线在同一点实现最小值。临界值在 Mean(*X*) – 0.5 * (b1/b2) 处，其中 b1 为线性系数，b2 为二次系数。该公式针对中心化的 *X*。从“参数估计值”表，您可以计算最佳含盐量为 1.5 - 0.5 * (0.5637/1.3499) = 1.29 茶匙。

每条曲线之间特定含盐量处的垂直距离指示特定含盐量的五个响应水平各自的概率。最高响应水平的概率是从顶部曲线到图矩形顶部的距离。当盐含量约为 1.3 茶匙时，该距离达到最大值。所有曲线共享同一临界点。

食盐和**食盐*食盐**的参数估计值是用于查找临界值的系数。尽管临界值在 Logistic 图上显示为最小值，但是对于使最高响应的概率最大化而言，它是最大值。报表的“解”部分显示在图 11.11 中的“响应曲面”下，其中 1.29 显示在**临界值**下。

堆叠多个列中的计数的示例

当作为频数（计数）的数据在数据表中的几个列中列出时，您必须将数据变换为 Logistic 回归所需的格式。例如，Ingots2.jmp 数据表（图 11.12）具有列**可用数量**和**不可用数量**。这些列对于**加热时间**和**浸泡时间**值的每个组合给出可以滚压和不可以滚压的金属锭数。

图 11.12 Ingots2.jmp 样本数据表

	加热时间	浸泡时间	可用数量	不可用数量	总数	P	损失
1	7	1	10	0	10	0.5000	6.9315
2	7	1.7	17	0	17	0.5000	11.7835
3	7	2.2	7	0	7	0.5000	4.8520
4	7	2.8	12	0	12	0.5000	8.3178
5	7	4	9	0	9	0.5000	6.2383
6	14	1	31	0	31	0.5000	21.4876
7	14	1.7	43	0	43	0.5000	29.8053
8	14	2.2	31	2	33	0.5000	22.8739
9	14	2.8	31	0	31	0.5000	21.4876
10	14	4	19	0	19	0.5000	13.1698
11	27	1	55	1	56	0.5000	38.8162
12	27	1.7	40	4	44	0.5000	30.4985
13	27	2.2	21	0	21	0.5000	14.5561
14	27	2.8	21	1	22	0.5000	15.2492
15	27	4	15	1	16	0.5000	11.0904

在拟合 Logistic 回归模型前，使用以下步骤将**可用数量**和**不可用数量**列堆叠为单个列：

1. 选择**帮助** > **样本数据库**，然后打开 Ingots2.jmp。
2. 选择**表** > **堆叠**。
3. 从“选择列”列表中选择**可用数量**和**不可用数量**，然后点击**堆叠列**。
4. 点击**确定**。

   这将创建图 11.13 中的新表。**标签**是响应 (*Y*) 列，**数据**是频数列。

这个堆叠的数据表等价于第 412 页的“名义型 Logistic 回归的示例”中使用的 Ingots.jmp 样本数据表。

图 11.13 堆叠的数据表

	加热时间	浸泡时间	总数	P	损失	Label	Data
1	7	1	10	0.5000	6.9315	可用数量	10
2	7	1	10	0.5000	6.9315	不可用数量	0
3	7	1.7	17	0.5000	11.7835	可用数量	17
4	7	1.7	17	0.5000	11.7835	不可用数量	0
5	7	2.2	7	0.5000	4.8520	可用数量	7
6	7	2.2	7	0.5000	4.8520	不可用数量	0
7	7	2.8	12	0.5000	8.3178	可用数量	12
8	7	2.8	12	0.5000	8.3178	不可用数量	0
9	7	4	9	0.5000	6.2383	可用数量	9
10	7	4	9	0.5000	6.2383	不可用数量	0
11	14	1	31	0.5000	21.4876	可用数量	31
12	14	1	31	0.5000	21.4876	不可用数量	0
13	14	1.7	43	0.5000	29.8053	可用数量	43
14	14	1.7	43	0.5000	29.8053	不可用数量	0
15	14	2.2	33	0.5000	22.8739	可用数量	31
16	14	2.2	33	0.5000	22.8739	不可用数量	2
17	14	2.8	31	0.5000	21.4876	可用数量	31
18	14	2.8	31	0.5000	21.4876	不可用数量	0

“名义型 Logistic”和“有序型 Logistic”特质的统计详细信息

- “Logistic 回归模型”
- “优势比”
- “统计检验的关系”

Logistic 回归模型

Logistic 回归使用 X 项的线性模型拟合名义型 Y 响应。更确切地说，它使用 Logistic 函数拟合响应水平的概率。对于两个响应水平，该函数为：

$$P(Y = r_1) = (1 + e^{-Xb})^{-1}$$ 其中 r_1 是第一个响应水平

或等价于：

$$\log\left(\frac{P(Y = r_1)}{P(Y = r_2)}\right) = Xb$$ 其中 r_1 和 r_2 分别是两个响应水平

注意：当 Y 为二值且具有名义型建模类型时，可以在“拟合模型”窗口中设置“目标水平”来指定您要对其概率建模的水平。在本节中，将目标水平指定为 r_1。

对于 r 个名义型响应水平 （其中 $r > 2$），模型由以下形式的 $r - 1$ 个线性模型参数定义：

$$\log\left(\frac{P(Y = j)}{P(Y = r)}\right) = Xb_j$$

最大似然的拟合原理是选择 β 通过模型使归因于发生响应的联合概率最大。该拟合原理等价于使负对数似然 （**– 对数似然**）最小化：

$$\text{损失} = -\text{对数似然} = \sum_{i=1}^{n} -\log(\text{Prob}(\text{第i行具有第 } y_j \text{ 个响应}))$$

优势比

对于两个响应水平， Logistic 回归模型指定如下：

$$\log\left(\frac{\text{Prob}(Y = r_1)}{\text{Prob}(Y = r_2)}\right) = Xb \text{ ，其中 } r_1 \text{ 和 } r_1 \text{ 是两个响应水平}$$

因此，**优势比**定义如下：

$$\frac{\text{Prob}(Y = r_1)}{\text{Prob}(Y = r_2)} = \exp(X\beta) = \exp(\beta_0) \cdot \exp(\beta_1 X_1) \cdots \exp(\beta_i X_i)$$

请注意 $\exp(\beta_i(X_i + 1)) = \exp(\beta_i X_i)\exp(\beta_i)$。这说明若 X_i 变化一个单位量，则优势乘以 $\exp(\beta_i)$，我们将其标记为**单位优势比**。当 X_i 在整个范围内变化时，优势乘以 $\exp((X_{高} - X_{低})\beta_i)$，我们将其标记为**范围优势比**。对于二值响应，响应水平交换后的对数优势比仅涉及参数符号的变化。因此，您可能需要报告值的倒数，以关注最后一个响应水平而非第一个响应水平。

二水平名义型效应的第一个和第二个水平分别编码为 1 和 -1，因此将关注范围优势比或其倒数。

若您选择“优势比”选项时未计算任何参数估计值的置信区间，则选择“优势比”选项将为优势比生成基于 Wald 的置信区间。

满足以下所有条件时选择 “优势比”选项将为优势比生成基于刻画似然的置信区间：

- 选择了“似然比检验”选项
- 选择了“置信区间”选项
- 每个响应的参数数目少于 8
- 行数少于 1000。

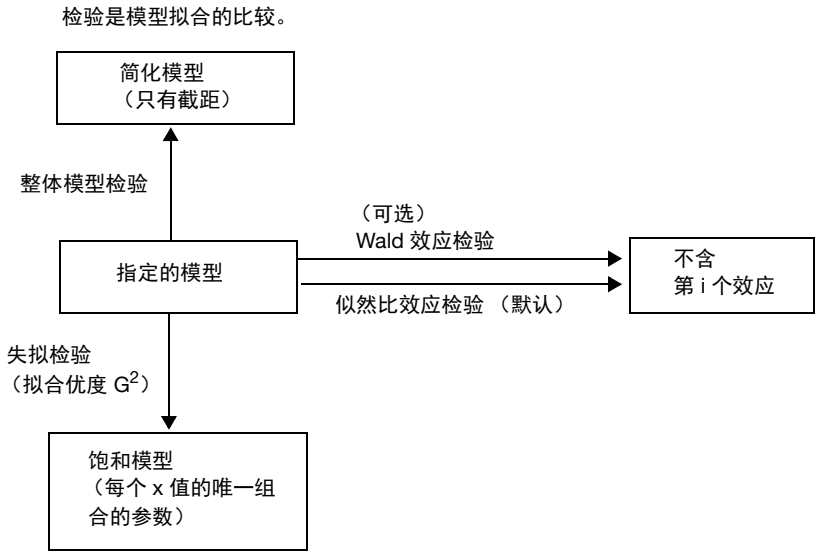
在“优势比”报表底部说明用于计算优势比的置信区间的方法。

统计检验的关系

“Logistic 回归”报表中的所有统计检验比较指定模型与子集或超集模型的拟合效果，如图 11.14 中所示。若检验显示显著性，则高阶模型是合理的。

- 整体模型检验：指定的模型是否显著好于简化模型（该模型除截距外没有任何效应）。
- 失拟检验：饱和模型是否显著好于指定的模型。
- 效应检验：指定的模型是否显著好于不含给定效应的模型。

图 11.14 统计检验的关系



第 12 章

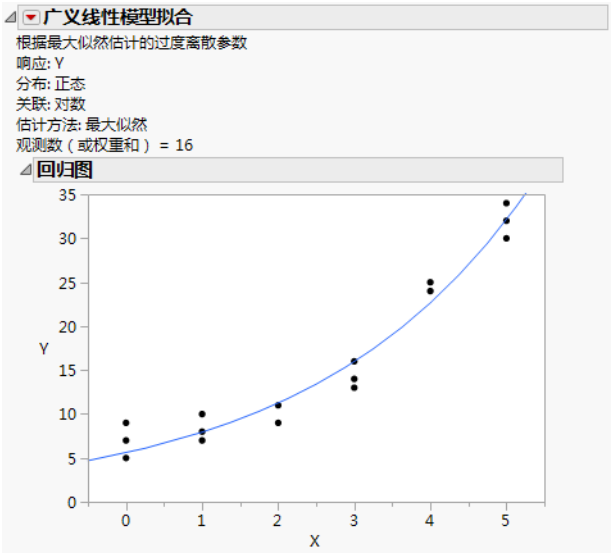
广义线性模型

拟合非正态响应分布的模型

广义线性模型提供统一方式来拟合不符合传统线性模型常规要求的响应。例如，频数统计通常用 Poisson 分布来表征，它使用广义线性模型来拟合。

“拟合模型”平台的“广义线性模型”特质使您可以针对具有二项、正态、Poisson 或指数分布的响应拟合广义线性模型。该平台提供的报表类似于为传统线性模型提供的那些报表。该平台还可以使用 Firth 校正处理 Logistic 回归模型的分离问题。

图 12.1 广义线性模型拟合的示例



目录

“广义线性模型”特质概述 435
广义线性模型的示例 435
启动“广义线性模型”特质 437
“广义线性模型拟合”报表 439
整体模型检验 439
“广义线性模型拟合”报表选项 440
“广义线性模型”特质的其他示例 442
使用对比比较变量水平的差值 442
带偏移的 Poisson 回归 443
具有对数连结的正态回归 444
“广义线性模型”特质的统计详细信息 446
模型选择和偏差 448

“广义线性模型”特质概述

传统线性模型广泛用于统计数据分析中。但是，有些情况不满足传统线性模型的假设。在这些情况下，传统线性模型不合适。传统线性模型假定响应是连续的、服从正态分布，且所有观测的方差为常数。这些假定可能不合理。例如，若您要对计数建模或观测的响应方差随响应增大而增大，则这些假设不合理。不满足传统线性模型假设的另一示例是当响应均值被限制为特定值范围时的情况，例如比例位于 0 到 1 之间。

对于这类属于更广范围的数据分析问题的情况，可以采用广义线性模型。广义线性模型是对传统线性模型的扩展。广义线性模型包含一个线性成分、一个连结函数和一个方差函数。连结函数 $g(\mu_i) = x'_i\beta$ 是单调的可微函数，它描述 Y_i 的期望值如何与线性预测变量相关。广义线性回归的一个示例是 Poisson 回归，其中 $\log(\mu_i)$ 是连结函数。有关使用“拟合模型”平台的“广义线性模型”特质的可用广义线性回归模型的完整列表，请参见第 446 页的““广义线性模型”特质的统计详细信息”。

可使用与传统线性模型相同的统计量，对拟合的广义线性模型进行汇总和评估。“拟合模型”平台为广义线性模型提供参数估计值、标准误差、拟合优度统计量、置信区间和假设检验。应该注意的是，精确的分布原理对于广义线性模型并不是始终可用或切合实际的。因此，一些推断过程基于渐近结果。

拟合广义线性模型的一个重要方面是模型中解释变量的选择。拟合优度统计量的变化通常用于评估解释变量的子集对特定模型的贡献。**偏差**被定义为可达到的最大对数似然函数值与回归参数的最大似然估计值处的对数似然函数值的差值的两倍。偏差通常用作拟合优度的测度。使用每个观测对应有参数的模型能够实现可达到的最大对数似然。

JMP PRO 对于广义线性建模中的变量选择和惩罚方法，您可以使用 JMP Pro 中“拟合模型”平台的“广义回归”特质。请参见第 247 页的“广义回归模型”一章。

广义线性模型的示例

本示例使用 Poisson 回归对来自筑巢的马蹄蟹研究的计数数据建模。每个雌性螃蟹的巢里住着一个雄性螃蟹。该研究调查是否有其他雄性（称为伴侣）住在附近。数据表 CrabSatellites.jmp 包含一个列出雄性伴侣数的响应变量，以及说明雌性螃蟹的颜色、脊情况、重量和壳宽的变量。您想关注的是伴侣数和描述雌性螃蟹的变量之间的关系。

要拟合 Poisson 回归模型，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 CrabSatellites.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**伴侣数**并点击 Y。
4. 选择**颜色**、**脊**、**宽度**和**重量**，然后点击**添加**。
5. 从“特质”列表中选择**广义线性模型**。
6. 从“分布”列表中选择 Poisson。

在“连结函数”列表中，应自动为您选择**对数**。

7. 点击**运行**。

图 12.2 螃蟹伴侣结果

广义线性模型拟合

响应: 伴侣数
分布: Poisson
关联: 对数
估计方法: 最大似然
观测数（或权重和）= 173

整体模型检验

模型	-对数似然	似然比卡方	自由度	概率>卡方
差值	41.6030357	83.2061	7	<.0001*
完全	452.44163			
简化	494.044666			

拟合优度统计量	卡方	自由度	概率>卡方
Pearson	533.8165	165	<.0001*
偏差	549.5856	165	<.0001*

AICc
921.7613

效应检验

源	自由度	似然比卡方	概率>卡方
颜色	3	9.2405455	0.0263*
脊	2	1.794722	0.4076
宽度	1	0.1168819	0.7324
重量	1	9.0438942	0.0026*

参数估计值

项	估计值	标准误差	似然比卡方	概率>卡方	置信下限	置信上限
截距	-0.710184	0.9339406	0.5761557	0.4478	-2.532002	1.1287428
颜色[Light Med]	0.3273541	0.132167	5.7773064	0.0162*	0.0619978	0.5810457
颜色[Medium]	0.0625029	0.0737267	0.7238945	0.3949	-0.080871	0.2083795
颜色[Dark Med]	-0.186351	0.0963747	3.8096887	0.0510	-0.377505	0.0007691
脊[Both Good]	0.0210298	0.0947416	0.049346	0.8242	-0.163737	0.2084429
脊[One Worn/Broken]	-0.129342	0.1313469	1.0163566	0.3134	-0.399791	0.1168764
宽度	0.0167487	0.0489197	0.1168819	0.7324	-0.07991	0.1118218
重量	0.0004965	0.0001663	9.0438942	0.0026*	0.000172	0.0008234

“整体模型检验”显示完整模型和简化模型的对数似然差值为 41.6。“概率 > 卡方”值等价于整体模型 *F* 检验的 *p* 值。 *p* 值小于 0.0001 指示模型在整体上是显著的。报表还包含校正的 Akaike 信息准则 (AICc) 值 921.7613。该值可以与其他模型进行比较以确定数据的最佳拟合模型。 AICc 值越小说明拟合模型越好。

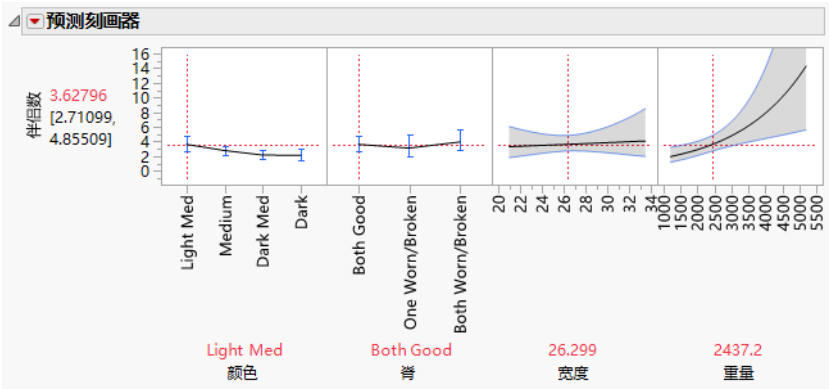
“效应检验”报表显示**重量**在模型中是显著因子。请注意，**重量**的 *p* 值 0.0026 在“参数估计值”表中具有相同的值，因为**重量**是连续变量。

“效应检验”报表还显示分类变量**颜色**是显著的。**颜色**具有四个水平：Light Med、Medium、Dark Med 和 Dark。前三个水平的参数估计值显示在“参数估计值”表中。Dark 的参数估计值是前三个水平的参数估计值之和的负数。

您可以使用“预测刻画器”进一步探索该模型。

8. 点击“广义线性模型拟合”旁边的红色小三角并选择**刻画器 > 刻画器**。

图 12.3 伴侣数的 “预测刻画器”



重量的置信带指示：较小的**重量**值在模型中的变异比起较大的**重量**值在模型中的变异要小。该刻画器支持您轻松探索分类变量的水平。从该刻画器，您可以看到伴侣螃蟹的预测数随着螃蟹颜色从浅变深而降低。

启动“广义线性模型”特质

通过以下方式启动 “广义线性模型”特质：选择**分析 > 拟合模型**，输入一个或多个 Y 列，然后从**特质**菜单选择**广义线性模型**。

图 12.4 选择了 “广义线性模型”的 “拟合模型”启动窗口

有关对所有特质通用的“拟合模型”窗口各方面的详细信息，请参见第 29 页的“模型规格”一章。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。这里提供“广义线性模型”特质特有的信息。

若您的模型效应有缺失值，可以将这些缺失值视为信息性类别。从“模型规格”红色小三角菜单中选择“信息性缺失”选项。

提示：“无截距”选项在“拟合模型”平台的“广义线性模型”特质中不可用。

当您为“特质”选择“广义线性模型”时，“拟合模型”启动窗口会有所改变，以包含其他选项。在“广义线性模型”特质中提供以下其他选项：

分布 指定响应变量的概率分布。

连结函数 指定将线性模型与响应变量关联的连结函数。

过度离散检验和区间 指定应在模型中包含过度离散参数。当响应方差大于响应分布的理论方差期望值时，发生过度离散。Poisson 和二项响应模型中都可能出现这种情况。McCullagh and Nelder (1989) 指出过度离散实际上并不少见。

注意：该选项在“整体模型检验”报表中的“拟合优度统计量”表中添加对应“过度离散”的列。

Firth 偏倚调整估计值 指定使用 Firth 偏倚调整方法来拟合模型。这个基于最大似然的方法与不使用偏倚修正的基于最大似然的模型相比，已被证实可以得到更好的估计值和检验值。此外，修正偏倚的 MLE 可缓解 Logistic 类型的模型中常见的分离问题。有关 Logistic 回归中的分离问题的详细信息，请参见 Firth (1993) 和 Heinze and Schemper (2002)。

偏移 （显示为“选择角色变量”按钮。）指定偏移变量。将偏移变量作为其参数固定为 1.0 的回归协变量处理。偏移变量最常用于在对数连结的 Poisson 回归情形中对均值建模统一尺度。

二项响应分布的响应规格

选择“二项”作为“分布”时，必须按以下方式之一指定响应变量：

- 若您的数据未汇总为事件频数，指定一个二值列作为响应。响应列必须是名义型。
- 若您的数据已汇总为事件频数，指定一个二值列作为响应以及一个频数变量作为“频数”角色。响应列必须是名义型，频数变量包含每个响应水平的计数。
- 若您的数据已汇总为事件频数和试验次数，以该顺序指定两个连续列：成功次数的计数和试验次数的计数。或者，您可以指定失败次数而非成功次数。

“广义线性模型拟合”报表

默认情况下，“广义线性模型拟合”报表包含有关模型规格和以下报表的详细信息：

奇异性详细信息 （仅当模型项之间存在线性相依性时才显示。）显示包含模型项满足的线性函数的报表。

回归图 （仅当有一个连续预测变量且分类预测变量不超过一个时显示。）显示垂直轴上的响应图和水平轴上的连续预测变量。点的上方显示一条回归线。若模型中有分类预测变量，则该分类预测变量的每个水平都有一条单独的回归线，而且图的旁边还会显示一个图例。

整体模型检验 显示用于比较整体模型拟合与忽略除截距参数之外的所有效应的模型的检验。该报表还包含拟合优度统计量和校正的 Akaike 信息准则 (AICc) 值。请参见第 439 页的“整体模型检验”。

效应汇总 一个交互式报表，该报表支持您在模型中添加或删除效应。请参见“标准最小二乘法模型”一章中第 158 页的“‘效应汇总’报表”。

效应检验 “效应检验”是单个效应的所有参数均为零的联合检验。若效应只有一个参数 （与连续效应一样），则这些检验与“参数估计值”表中的检验相同。

注意：即便使用 Firth 调整，效应检验也基于非惩罚似然函数。

参数估计值 显示参数估计值、标准误差、相关的假设检验和置信限。简单连续效应只有一个参数。具有复杂分类效应的模型对于每个预期自由度有一个参数。

注意: 若有超过 1,000 个观测，则显示基于 Wald 的置信区间。否则，显示轮廓似然置信区间。

预测值 - 学生化偏差残差 显示垂直轴上为学生化偏差残差、水平轴上为预测响应值的图。

整体模型检验

“整体模型检验”表显示用于比较整体模型拟合与忽略除截距参数之外的所有回归参数的模型的检验。它还包含两个拟合优度统计量和 AICc 值来评估模型适当性。

“整体模型检验”表显示以下量：

模型 列出模型标签。

差值 **完全**模型与**简化**模型之间的差值。该模型用于测量回归变量作为一个整体对拟合的显著性。

完全 完整的模型包括截距和所有效应。

简化 仅包含截距参数的模型。

– 对数似然 相应模型的负对数似然。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。

似然比卡方 假设所有回归参数均为零的似然比卡方检验统计量。该检验统计量的值是拟合模型和仅包含截距的简化模型之间的负对数似然差值的两倍。

自由度 完全与简化模型的差值的自由度 (DF)。

概率 > 卡方 若指定的模型拟合不比仅包含截距的模型好，此时获得更大卡方值的概率。

拟合优度统计量 列出两个拟合优度统计量：Pearson 和偏差。

卡方 相应拟合优度统计量的卡方检验统计量。

自由度 相应拟合优度统计量的自由度。

概率 > 卡方 相应拟合优度统计量的 *p* 值。

过度离散 （仅当在启动窗口中选择了“过度离散检验和区间”选项时显示。）过度离散参数的估计值。请参见第 446 页的 “‘广义线性模型’特质的统计详细信息”。

AICc 校正的 Akaike 信息准则。请参见“统计详细信息”附录中第 477 页的“似然、AICc 和 BIC”。

“广义线性模型拟合”报表选项

“广义线性模型拟合”红色小三角菜单包含以下选项：

定制检验 支持您检验定制假设。有关定制检验的详细信息，请参见 “标准最小二乘法模型”一章中第 109 页的 “定制检验”。

对比 支持您检验一个变量各水平之间是否存在差异。若对比涉及协变量，您可以指定要针对协变量的哪一个值来检验对比。有关 “对比”选项的示例，请参见第 442 页的 “使用对比比较变量水平的差值”。

逆预测 （仅对于连续 *X* 变量可用。）可以在给定 *Y* 的特定值以及其他 *X* 变量值的情况下预测某个 *X* 值。有关“逆预测”选项的详细信息，请参见“标准最小二乘法模型”一章中第 125 页的 “逆预测”。

估计值的协方差 显示或隐藏模型中所有效应的协方差矩阵。参数估计量的估计协方差矩阵定义如下：

$$\Sigma = -\mathbf{H}^{-1}$$

其中 **H** 是使用最后一次迭代中参数估计值求得的 Hessian（或二阶导数）矩阵。请注意，离散参数 （无论是估计还是指定的）将并入 **H**。对应于别名参数的行和列不包括在 Σ 中。

估计值的相关性 显示或隐藏模型中所有效应的相关性矩阵。相关性矩阵是标准化的协方差矩阵。对于 Σ 的每个 σ_{ij} 元素，相应的相关性矩阵元素是 $\sigma_{ij}/\sigma_i\sigma_j$，其中 $\sigma_i = \sqrt{\sigma_{ii}}$。

刻画器 显示以下刻画器的子菜单：

刻画器 显示或隐藏一个预测刻画器，它用于检查每个 X 变量的预测迹线。有关预测刻画器的详细信息，请参见“标准最小二乘法模型”一章中第 143 页的“刻画器指南”。

等高线刻画器 显示或隐藏交互式等高线刻画器。有关“等高线刻画器”的详细信息，请参见《刻画器指南》。

曲面刻画器 显示或隐藏 3D 曲面刻画器。有关“曲面刻画器”的详细信息，请参见《刻画器指南》。

诊断图 显示残差、预测值和实际值的图的子菜单。这些图允许您搜索离群值和确定模型的适当性。有关偏差的详细信息，请参见第 448 页的 “模型选择和偏差”。提供以下图：

预测值-学生化偏差残差 显示或隐藏垂直轴上为学生化偏差残差、水平轴上为预测响应值的图。

预测值-学生化 Pearson 残差 显示或隐藏垂直轴上为学生化 Pearson 残差、水平轴上为预测响应值的图。

预测值-偏差残差 显示或隐藏垂直轴上为偏差残差、水平轴上为预测响应值的图。

预测值- Pearson 残差 显示或隐藏垂直轴上为 Pearson 残差、水平轴上为预测响应值的图。

回归图 （仅当有一个连续预测变量且分类预测变量不超过一个时才可用。）显示或隐藏垂直轴上的响应图和水平轴上的连续预测变量。点的上方显示一条回归线。若模型中有分类预测变量，则该分类预测变量的每个水平都有一条单独的回归线，而且图的旁边还会显示一个图例。

线性预测变量图 （仅当有一个连续预测变量且分类预测变量不超过一个时才可用。）显示或隐藏垂直轴上为通过反连结函数变换的响应值、水平轴上为连续预测变量的图。点的上方显示一条已变换的回归线。若模型中有分类预测变量，则该分类预测变量的每个水平都有一条单独的已变换回归线，而且图的旁边还会显示一个图例。

保存列 显示选项的子菜单，您可以使用它们将某些数量保存为当前数据表中的新列。有关残差公式的详细信息，请参见第 449 页的 “残差公式”。

预测公式 在当前数据表中创建预测模型的公式列。

预测值 在当前数据表中创建包含模型所预测的值的列。

均值置信区间 在当前数据表中创建包含模型的预测方程的 95% 置信限的列。这些置信限反映参数估计值的变异。

注意：通过从“模型规格”红色小三角菜单中选择“设置 Alpha 水平”，您可以在“拟合模型”窗口中更改置信限的 α 水平。

保存单值置信限 在当前数据表中创建包含模型的给定单值的 95% 置信限的列。这些置信限反映误差变异和参数估计值的变异。

注意：通过从“模型规格”红色小三角菜单中选择“设置 Alpha 水平”，您可以在“拟合模型”窗口中更改置信限的 α 水平。

偏差残差 在当前数据表中创建包含偏差残差的列。

Pearson 残差 在当前数据表中创建包含 Pearson 残差的列。

学生化偏差残差 在当前数据表中创建包含学生化偏差残差的列。

学生化 Pearson 残差 在当前数据表中创建包含学生化 Pearson 残差的列。

模型对话框 显示当前分析的已完成的启动窗口。

效应汇总 显示或隐藏“效应汇总”报表，该报表支持您以交互方式更新模型中的效应。请参见“标准最小二乘法模型”一章中第 158 页的 “‘效应汇总’报表”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“广义线性模型”特质的其他示例

- “使用对比比较变量水平的差值”
- “带偏移的 Poisson 回归”
- “具有对数连结的正态回归”

使用对比比较变量水平的差值

本示例是第 435 页的“广义线性模型的示例”中螃蟹伴侣示例的继续。假定您要检验具有好脊的雌性螃蟹是否比具有差脊或断脊的雌性螃蟹吸引更多数目的雄性螃蟹（伴侣）。

1. 完成第 435 页的“广义线性模型的示例”的第 1 步到第 7 步。
2. 点击“广义线性模型拟合”旁边的红色小三角菜单，然后选择**对比**。“选择效应进行对比”窗口随即显示。
3. 选择**脊**（要关注的变量），然后点击**执行**。
4. 要比较具有好脊的螃蟹和具有差脊或断脊的螃蟹，请点击“Both Good”旁边的 + 按钮以及“One Worn/Broken”和“Both Worn/Broken”旁边的 - 按钮。

   这将创建一个对比设置，它比较具有好脊的雌性螃蟹和具有差脊或断脊的雌性螃蟹。

图 12.5 “对比设置”窗口

5. 点击**完成**。

图 12.6 “对比”报表

因为“概率>卡方”0.8242 远大于 0.05，我们无法得到基于脊情况的伴侣吸引力有差异这个结论。

带偏移的 Poisson 回归

样本数据表 Ship Damage.JMP 改编自 McCullagh and Nelder (1989) 中的数据。该数据表包含有关波浪对船体前部造成的某类损坏的信息。船体建造工程师想了解与以下三个变量关联的损坏风险：船种类（**种类**）、造船的年份（**制造年份**）和船投入使用的年份（**投入使用年份**）。

在这个分析中，我们使用变量**服役期**（投入使用总月数的对数）作为**偏移变量**。将偏移变量作为其参数固定为 1.0 的回归协变量处理。偏移变量最常用于在对数连结的 Poisson 回归情形中对均值建模统一尺度。在本例中，我们使用 log(服役月数)，因为工程师期望维修次数与投入使用的月数成比例。

为了了解如何使用偏移变量，假定 GLM 的线性成分称为 η。然后使用对数连结函数，带有偏移均值的模型指定如下：

$$\exp[\text{Log}(\text{服役月数}) + \eta] = [(\text{服役月数}) * \exp(\eta)].$$

要运行该示例，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Ship Damage.jmp。

2. 选择**分析 > 拟合模型**。
3. 从 “特质”列表中选择**广义线性模型**。
4. 从 “分布”列表中选择 Poisson。
   在 “连结函数”列表中，应自动为您选择**对数**。
5. 选择 N 并点击 Y。
6. 选择**服役期**并点击**偏移**。
7. 选择**种类**、**制造年份**、**投入使用年份**并点击**添加**。
8. 点击**过度离散检验和区间**的复选标记框。
9. 点击**运行**。

从该报表中，我们看到三个效应 （**种类**、**制造年份**、**投入使用年份**）都是显著的。

图 12.7 具有偏移的 Poisson 模型的部分报表

广义线性模型拟合

偏移: 服役年数
根据 “Pearson 卡方/自由度” 估计的过度离散参数
响应: 数量
分布: Poisson
关联: 对数
估计方法: 最大似然
观测数（或权重和） = 34

整体模型检验

模型	-对数似然	似然比卡方	自由度	概率>卡方
差值	31.8247288	63.6495	8	<.0001*
完全	40.3773439			
简化	72.2020727			

拟合优度统计量	卡方	自由度	概率>卡方	过度离散
Pearson	42.2769	25	0.0168*	1.6911
偏差	38.6960	25	0.0395*	

AICc
110.3199

效应汇总

源	LogWorth	p 值
制造年份	3.475	0.00034
种类	2.137	0.00730
投入使用年份	1.919	0.01204

删除 添加 编辑 FDR

效应检验

源	自由度	似然比卡方	概率>卡方
种类	4	13.997527	0.0073*
制造年份	3	18.572934	0.0003*
投入使用年份	1	6.3044662	0.0120*

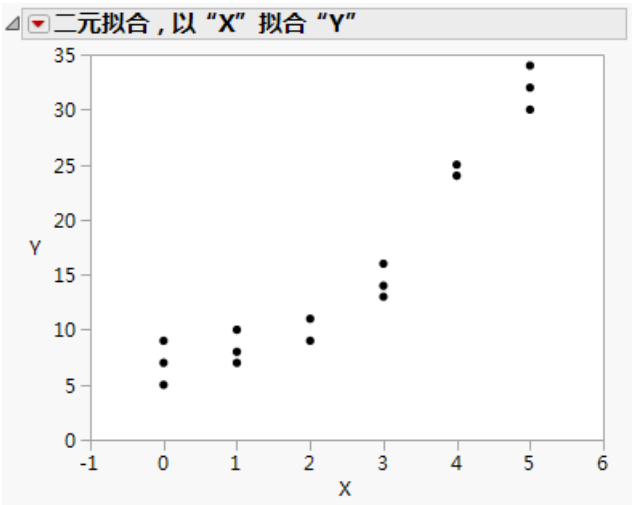
具有对数连结的正态回归

本示例使用 Nor.jmp 数据表，其中 X 是解释变量， Y 是响应变量。首先，您探索 X 和 Y 之间的关系来确定要在 “拟合模型”平台的 “广义线性模型”特质中使用的合适连结函数。

1. 选择**帮助 > 样本数据库**，然后打开 Nor.jmp。
2. 选择**分析 > 以 X 拟合 Y**。

3. 选择 Y 并点击 Y，**响应**。
4. 选择 X，点击 X，**因子**，然后点击**确定**。

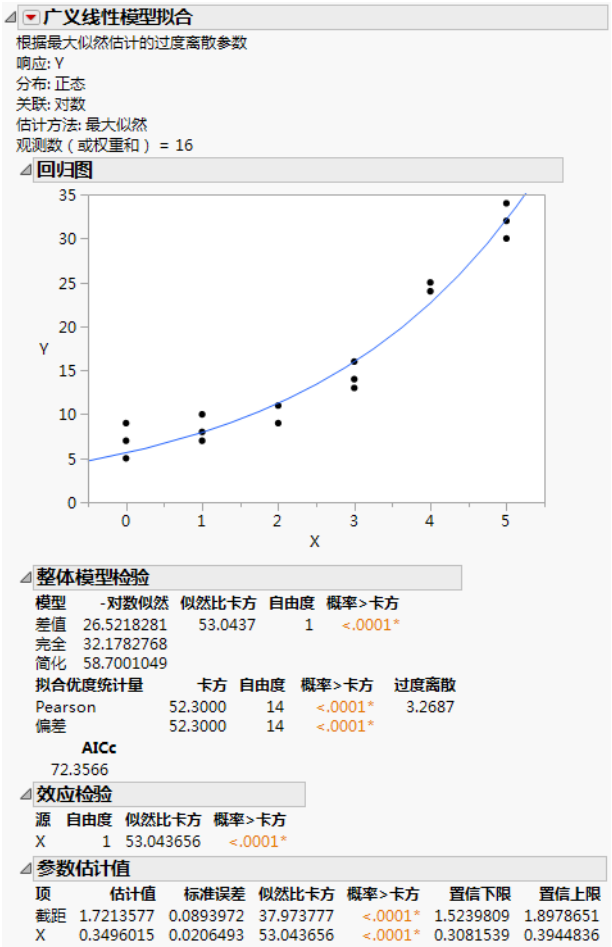
图 12.8 Nor.jmp 的 “以 X 拟合 Y” 结果



您可以看到 Y 随着 X 的变化呈现非线性变化，而方差基本为常数。因此，具有对数连结函数的正态分布适合对这些数据建模，即 $\log(\mu_i) = \mathbf{x}_i'\beta$ 使得 $\mu_i = \exp(\mathbf{x}_i'\beta)$。

5. 选择**分析 > 拟合模型**。
6. 在 “特质” 列表中，选择**广义线性模型**。
7. 在 “分布” 列表中，选择**正态**。
8. 在 “连结函数” 列表中，选择**对数**。
9. 选择 Y 并点击 Y。
10. 选择 X 并点击**添加**。
11. 点击**运行**。

图 12.9 Nor 结果



“广义线性模型”特质的统计详细信息

要构造广义线性模型，您必须为数据选择响应变量和解释变量。然后为响应选择合适的连结函数和概率分布。解释变量可以是连续变量、分类变量和交互作用的任意组合。表 12.1 中列出了一些常见的广义线性模型示例。

表 12.1 广义线性模型的示例

模型	响应变量	分布	默认连结函数
传统线性模型	连续	正态	恒等，$g(\mu) = \mu$

表 12.1 广义线性模型的示例 (续)

模型	响应变量	分布	默认连结函数
Logistic 回归	计数或二值随机变量	二项	logit，$g(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$
对数线性模型中的 Poisson 回归	计数	Poisson	对数，$g(\mu) = \log(\mu)$
指数回归	连续（正数）	指数	$\frac{1}{\mu}$

平台通过参数向量的最大似然估计用广义线性模型来拟合数据。通常对于参数的最大似然估计值没有闭合形式的解。因此，平台使用 Nelder and Wedderburn (1972) 最早提出的一个方法通过迭代拟合过程在数字上估计模型的参数。通过用 Pearson 拟合优度统计量除以其自由度来估计过度离散参数 ϕ。基于最大似然估计量的渐近正态性计算估计参数的协方差、标准误差和置信限。

在“拟合模型”平台的“广义线性模型”特质中提供很多连结函数和概率分布。表 12.2 列出了内置连结函数。

表 12.2 内置连结函数

连结函数名称	连结函数公式
恒等	$g(\mu) = \mu$
Logit	$g(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$
Probit	$g(\mu) = \Phi^{-1}(\mu)$，其中 Φ 是标准正态累积分布函数
对数	$g(\mu) = \log(\mu)$
倒数	$g(\mu) = \frac{1}{\mu}$
幂	$g(\mu) = \begin{cases} \mu^{\lambda} & \text{若 } (\lambda \neq 0) \\ \log(\mu) & \text{若 } \lambda = 0 \end{cases}$
互补重对数	$g(m) = \log(-\log(1-\mu))$

选择“幂”连结函数时，显示一个数字框，允许您输入所需的幂。

表 12.3 列出与该响应变量的可用分布关联的方差函数。

表 12.3 响应分布的方差函数

分布	方差函数
正态	$V(\mu) = 1$
二项	$V(\mu) = \mu(1 - \mu)$
Poisson	$V(\mu) = \mu$
指数	$V(\mu) = \mu^2$

模型选择和偏差

广义线性建模的一个重要方面是模型中解释变量的选择。拟合优度统计量的变化通常用于评估解释变量的子集对特定模型的贡献。将**偏差**定义为可达到的最大对数似然与回归参数的最大似然估计值处的对数似然的差值的两倍。偏差通常用作拟合优度的测度。使用每个观测对应有参数的模型能够实现可达到的最大对数似然。表 12.4 列出响应变量的每个可用分布的偏差公式。

表 12.4 响应分布的偏差公式

分布	偏差公式
正态	$\sum_i w_i(y_i - \mu_i)^2$
二项	$2\sum_i w_i m_i \left[y_i \log\left(\frac{y_i}{\mu_i}\right) + (1 - y_i)\log\left(\frac{1 - y_i}{1 - \mu_i}\right) \right]$
Poisson	$2\sum_i w_i \left[y_i \log\left(\frac{y_i}{\mu_i}\right) - (y_i - \mu_i) \right]$
指数	$2\sum_i w_i \left[-\log\left(\frac{y_i}{\mu_i}\right) + \left(\frac{y_i - \mu_i}{\mu_i}\right) \right]$

Pearson 卡方统计量定义如下：

$$X^2 = \sum_i \frac{w_i(y_i - \mu_i)^2}{V(\mu_i)}$$

其中

y_i 是第 i 个响应

μ_i 是相应的预测均值

$V(\mu_i)$ 是方差函数

w_i 是第 i 个观测的已知权重

注意：若未指定权重，则对于所有观测 $w_i = 1$。

用于变量选择的一个策略是拟合一系列模型。最开始是只有一个截距项的简单模型，然后在每个后续模型中多包含一个解释变量。您可以按两个相邻模型之间的偏差或拟合的对数似然差值测量增加的解释变量的重要性。渐近检验允许您评估增加项的统计显著性。

分布为非正态时，使用正态临界值来替代逆预测的 t 分布临界值。

残差公式

偏差

$$r_{Di} = \sqrt{d_i}(sign(y_i - \mu_i))$$

学生化偏差

$$r_{Di} = \frac{sign(y_i - \mu_i)\sqrt{d_i}}{\sqrt{\phi(1 - h_i)}}$$

Pearson

$$r_{Pi} = \frac{(y_i - \mu_i)}{\sqrt{V(\mu_i)}}$$

学生化 Pearson

$$r_{Pi} = \frac{y_i - \mu_i}{\sqrt{V(\mu_i)(1 - h_i)}}$$

其中

$(y_i - \mu_i)$ 是原始残差

若 $(y_i - \mu_i)$ 为正数，则 $\text{sign}(y_i - \mu_i)$ 为 1；若 $(y_i - \mu_i)$ 为负数，则它为 -1。

d_i 是观测 i 对总偏差的贡献

ϕ 是离散参数

$V(\mu_i)$ 是方差函数

h_i 是矩阵 $W_e^{(1/2)}X(X'W_eX)^{-1}X'W_e^{(1/2)}$ 的第 i 个对角线元素，其中 W_e 是在计算期望信息矩阵时使用的权重矩阵。

有关残差和广义线性模型的详细信息，请参见 SAS Institute Inc.(2020a)。

附录 A

统计详细信息

拟合线性模型

本附录讨论不同类型的响应模型、模型因子、模型设计编码和参数化。它还包含本手册正文所述方法的很多其他详细信息。

JMP 系统将线性模型分为三种不同的响应模型进行拟合，即连续型、有序型和名义型。因子方面的很多详细信息对于不同响应模型是相同的，但是 JMP 仅支持连续响应的图形和边缘刻画，不支持有序型和名义型响应的图形和边缘刻画。

不同的计算机程序使用不同的设计矩阵编码，因而通过不同的参数化来拟合效应和构造假设检验。JMP 使用不用于 SAS 系统中的 GLM 过程的编码方式，尽管在多数情况下 JMP 和 SAS GLM 过程生成相同的结果。以下章节说明 JMP 编码的详细信息，当它不同于 SAS GLM 过程的结果时将突出显示这些情况。

目录

响应模型 453
连续响应 453
名义型响应 453
有序型响应 455
因子模型 456
连续因子 456
名义型因子 456
有序型因子 467
频数 472
常用假设 472
假设模型 472
相对显著性 472
多个推断 472
合法性评估 473
替代方法 473
关键统计概念 473
不确定性，统一的概念 474
两个基本拟合装置 474
似然、AICc 和 BIC 477
功效计算 478
LSN 的计算 478
LSV 的计算 478
功效的计算 479
调整功效的计算 480
带置信限的逆预测 481

响应模型

JMP 根据三种不同的响应拟合线性模型：连续型、名义型和有序型。JMP 中提供的模型和方法很实用，得到广泛使用，满足了统计软件工具中对一般方法的需求。与所有统计软件一样，您需要了解选用的模型的假设以及不满足假设造成的后果。请参见本章中的第 472 页的 “常用假设”。

- “连续响应”
- “名义型响应”
- “有序型响应”

连续响应

当响应列（被分配了 Y 角色的列）为连续型时，JMP 直接拟合响应的值。基本模型是用于每个观测的模型，如下所示：

Y = (*X* 和参数组成的某个函数) + 误差

统计检验基于以下假设：模型中的误差项服从正态分布。

连续响应的拟合原理

拟合原理被称为**最小二乘法**。最小二乘法估计模型中的参数以使误差平方和最小。拟合模型中的误差称为**残差**，它是每个观测的实际值与拟合模型所预测的值之间的差值。

若误差具有正态分布，则最小二乘法等价于估计的最大似然法。这意味着分析估计的模型给出最有可能的残差。对数似然是正态分布的误差平方和的尺度倍数。

连续响应的基本模型

连续测量值的最简单模型只拟合一个值来预测所有响应值。该值是**均值**的估计值。均值仅是响应值的算术平均值。其他所有模型都与该基本模型进行比较。

名义型响应

使用 logit 模型的简单扩展来分析名义型响应。对于二值 （二水平）响应， logit 响应模型指定如下：

$$\log\left(\frac{P(y = 1)}{P(y = 2)}\right) = X\beta$$

上面的等式也可以写为：

$$P(y = 1) = F(X\beta)$$

其中 $F(x)$ 是标准 Logistic 分布的累积分布函数：

$$F(x) = \frac{1}{1+e^{-x}} = \frac{e^{x}}{1+e^{x}}$$

对于 r 个响应水平， JMP 拟合响应是数据值所给定的 r 个不同响应水平之一的概率。概率估计值必须均为正值。对于给定的 Xs 配置，响应水平上的概率估计值之和必须为 1。JMP 用于预测概率的函数是线性模型和多响应 Logistic 函数的组合。这有时称为**对数线性**模型，因为概率比的对数是线性模型。JMP 将每个响应概率与第 r 个概率关联并对这 r - 1 个模型分别拟合一组设计参数。

$$\log\left(\frac{\mathrm{P}(y=j)}{\mathrm{P}(y=r)}\right) = \mathrm{X}\beta_{(j)} \text{ 对于 } j = 1, \ldots, r-1$$

名义型响应的拟合原理

该拟合原理称为**最大似然**。它估计参数以使数据给定的所有响应的联合概率为可从模型获得的最大概率。不直接报告联合概率（似然），报告似然的负对数合计更方便。

不确定性 （负对数似然）是概率的负对数之和，该概率通过模型归因于在样本数据中实际发生的响应。对于大小为 n 的样本，它通常表示为 H，计算公式如下：

$$H = \sum_{i=1}^{n} -\log(\mathrm{P}(y = y_i))$$

若将概率 1 归因于已发生的每个事件，则对于完全拟合，负对数之和为零。

名义型模型的拟合会需要很长时间并且占用很多内存，有很多响应水平时更是如此。JMP 使用**迭代历史记录**跟踪计算进度，该记录显示负对数似然值在收敛到估计值的过程中逐渐变小。

名义型响应的基本模型

名义型响应的最简单模型是一组常数响应概率，它们拟合为整个数据表上每个响应水平的发生率。换句话说， y 是响应水平 j 的概率用每个响应水平的总计 n_j 除以总样本计数 n 来估计。该概率指定如下：

$$p_j = \frac{n_j}{n}$$

其他所有模型都与该基本模型进行比较。基本模型对于名义型响应的作用就像样本均值对于连续模型的作用一样。

R^2 统计量测量模型所解释的不确定性部分，公式为：

$$1 - \frac{H(\text{完全模型})}{H(\text{基本模型})}$$

但是，实际上对于分类模型 R^2 接近 1 的情况很少见。

有序型响应

具有有序型响应 (Y) 和具有名义型响应一样，JMP 拟合响应是数据给出的 *r* 个不同响应水平之一的概率。

有序型数据和连续数据一样具有顺序。在分析中使用该顺序但是不使用有序水平之间的间距或距离。若您有数值响应但是希望模型忽略值的间距，可以将有序型水平分配给该响应列。若您有分类变量且水平以某种自然顺序（如低、中和高）排列，可以使用有序型建模类型。

通过为累积概率拟合一系列平行 Logistic 曲线来对有序型响应建模。每条曲线具有相同的设计参数但是具有不同的截距，其公式指定如下：

$$\mathrm{P}(y \le j) = F(\alpha_j + \mathrm{X}\beta) \text{ 对于 } j = 1, \ldots, r-1$$

其中，有 *r* 个响应水平且 *F(x)* 是标准 Logistic 累积分布函数

$$F(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{1+e^x}$$

还可以根据非观测的连续变量 *z* 表示该函数，当该变量超过各种阈值时导致有序型响应变化。

$$y = \begin{cases} r & \alpha_{r-1} \le z \\ j & \alpha_{j-1} \le z < \alpha_j \\ 1 & z \le \alpha_1 \end{cases}$$

其中 *z* 是线性模型和误差的无法观测函数

$$z = \mathrm{X}\beta + \varepsilon$$

ε 具有 Logistic 分布。

这些模型具有吸引力的地方在于它们可识别响应的有序特征，所需的参数数目比名义型模型少得多，而且计算速度很快。

可以通过在数学上等价的另一方式看待有序型模型：设想一个名义型模型，它不是对优势建模而是对累积概率建模。不是针对除最后一个水平外的所有水平拟合函数，而是只拟合一个函数并滑动它来拟合每个累积响应概率。

有序型响应的拟合原理

有序型响应模型的最大似然拟合原理与名义型响应模型的拟合原理相同。它估计参数以使发生的所有响应的联合概率为可从模型获得的最大概率。与名义型拟合相比，它使用的迭代方法速度更快并且占用的内存更少。

基本模型

像名义型响应一样，有序型响应的最简单模型是拟合为整个数据表中响应的发生率的一组响应概率。

因子模型

模型的因子方面涉及如何对 x 变量 （因子）建模以预测期望值或概率。

因子进入作为 x 值和要估计的参数的线性组合的预测方程。对于连续响应模型，其中 i 表示观测索引， j 表示参数索引，典型观测值 y_i 的假设模型公式表示为：

$$y_i = \beta_0 + \beta_1 x_{1i} + \ldots + \beta_k x_{ki} + \varepsilon_i$$

其中

y_i 是响应

x_{ij} 是数据的函数

ε_i 是无法观测的随机误差实现值

b_j 是要估计的未知参数。

从因子项生成线性模型中 x 的方式对于每个建模类型各不相同。线性模型 x 也可以是复杂效应，如交互作用或嵌套效应。将在后面详细讨论复杂效应。

连续因子

连续因子作为回归变量直接放入设计矩阵。若一个列是其他列的线性函数，则该列的参数标记为**清零**或**不可估计**。连续因子与其他因子（交互作用和多项式项）交叉时，连续因子以其均值为中心。若因子具有**混料**或**编码**列属性或在指定模型时禁用中心化多项式选项，则禁用中心化。若有编码列属性，则在拟合前对因子编码。

名义型因子

在设计矩阵中，名义型因子变换为指标变量。SAS GLM 为每个名义型水平构造一个指示符列。JMP 为每个名义型水平构造相同的指示符列，但不包括最后一个水平。最后一个名义型水平出现时，从因子的所有其他列减去 1。例如，考虑针对 GLM 和 JMP 进行编码具有三个水平的名义型因子 A，如下所示。

表 A.1 名义型因子 A

	GLM			JMP	
A	A1	A2	A3	A13	A23
A1	1	0	0	1	0
A2	0	1	0	0	1
A3	0	0	1	-1	-1

在 GLM 中，线性模型设计矩阵具有列之间的线性相依性，最小二乘法解使用广义逆。选择的解发生在 A3 参数设置为零时。

在 JMP 中，将线性模型设计矩阵编码以便它达到满秩，除非存在缺失单元格或其他偶然的共线性。A 效应最后一个水平的参数是其他水平对应参数之和的负数，这使得在所有效应水平上参数之和为零。

参数的解释

注意：名义型水平的参数解释为该水平的预测响应与所有水平上的平均预测响应之间的差值。

一个因子水平的设计列构造为该因子水平的 0-1 指示符减去最后一个水平的指示符。这是得到上述参数解释的编码。

表 A.2 解释参数

JMP 参数报表	如何解释	设计列编码
截距	所有水平上的均值	**1**′
A[1]	$\alpha_1 - 1/3(\alpha_1 + \alpha_2 + \alpha_3)$	(A==1) – (A==3)
A[2]	$\alpha_2 - 1/3(\alpha_1 + \alpha_2 + \alpha_3)$	(A==2) – (A==3)

交互作用和交叉效应

GLM 和 JMP 均通过计算要交叉的因子的设计列所在行的直积来构造交互作用效应。例如，GLM 代码

```
PROC GLM;
  CLASS A B;
  MODEL A B A*B;
```

生成以下设计矩阵：

表 A.3 设计矩阵

		A			B			AB								
A	B	1	2	3	1	2	3	11	12	13	21	22	23	31	32	33
A1	B1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
A1	B2	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0
A1	B3	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0
A2	B1	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0
A2	B2	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0
A2	B3	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0
A3	B1	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
A3	B2	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0
A3	B3	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1

使用 JMP **拟合模型**命令并请求列 A 和 B 的析因模型将得到以下设计矩阵。请注意该矩阵中的 A13 是上一矩阵中的 A1–A3。但是，当前矩阵中 A13B13 是 A13*B13。

表 A.4 当前矩阵

		A		B					
A	B	13	23	13	23	A13 B13	A13 B23	A23 B13	A23 B23
A1	B1	1	0	1	0	1	0	0	0
A1	B2	1	0	0	1	0	1	0	0
A1	B3	1	0	-1	-1	-1	-1	0	0
A2	B1	0	1	1	0	0	0	1	0
A2	B2	0	1	0	1	0	0	0	1
A2	B3	0	1	-1	-1	0	0	-1	-1
A3	B1	-1	-1	1	0	-1	0	-1	0
A3	B2	-1	-1	0	1	0	-1	0	-1
A3	B3	-1	-1	-1	-1	1	1	1	1

当问题涉及具有少数几个水平的因子的交互作用时，JMP 编码可以节省内存和一些计算时间。依据 3*3 交叉模型的参数，单元格的期望值为：

表 A.5 3*3 交叉模型

	B1	B2	B3
A1	$\mu + \alpha_1 + \beta_1 + \alpha\beta_{11}$	$\mu + \alpha_1 + \beta_2 + \alpha\beta_{12}$	$\mu + \alpha_1 - \beta_1 - \beta_2 - \alpha\beta_{11} - \alpha\beta_{12}$
A2	$\mu + \alpha_2 + \beta_1 + \alpha\beta_{21}$	$\mu + \alpha_2 + \beta_2 + \alpha\beta_{22}$	$\mu + \alpha_2 - \beta_1 - \beta_2 - \alpha\beta_{21} - \alpha\beta_{22}$
A3	$\mu - \alpha_1 - \alpha_2 + \beta_1 - \alpha\beta_{11} - \alpha\beta_{21}$	$\mu - \alpha_1 - \alpha_2 + \beta_2 - \alpha\beta_{12} - \alpha\beta_{22}$	$\mu - \alpha_1 - \alpha_2 - \beta_1 - \beta_2 + \alpha\beta_{11} + \alpha\beta_{12} + \alpha\beta_{21} + \alpha\beta_{22}$

嵌套效应

GLM 中的嵌套效应采用与交互作用效应相同的编码方式，因为 GLM 通过不在模型中的项确定正确的检验。不包括在模型中的任何效应可通过包含的交互作用（或等价的嵌套）效应来吸收。

在 JMP 中嵌套效应的编码方式有所不同。JMP 使用括号里的项作为每个组的分组项。对于嵌套项的每个水平组合，JMP 在括号外构造效应。外部项的水平不需要在嵌套项的水平上排列。构造设计列和参数时单独处理嵌套的每个水平。

表 A.6 嵌套效应

				B(A)					
				A1	A1	A2	A2	A3	A3
A	B	A13	A23	B13	B23	B13	B23	B13	B23
A1	B1	1	0	1	0	0	0	0	0
A1	B2	1	0	0	1	0	0	0	0
A1	B3	1	0	-1	-1	0	0	0	0
A2	B1	0	1	0	0	1	0	0	0
A2	B2	0	1	0	0	0	1	0	0
A2	B3	0	1	0	0	-1	-1	0	0
A3	B1	-1	-1	0	0	0	0	1	0
A3	B2	-1	-1	0	0	0	0	0	1
A3	B3	-1	-1	0	0	0	0	-1	-1

名义型因子的最小二乘均值

在将所有其他因子设置为某个中性值后，最小二乘均值是对应于某个水平组合的预测值。直接连续回归变量的中性值定义为样本均值。具有未涉及的名义型因子的效应的中性值定义为水平上所取的平均效应（它导致我们的编码中全部为零）。在第 469 页的“有序型最小二乘均值”中，有序型因子使用另一中性值。最小二乘均值可能无法估计，若无法估计，将它们标记为无法估计。在不使用权重的情况下，JMP 中的最小二乘均值与 SAS PROC GLM (Goodnight and Harvey 1978) 中的最小二乘均值一致。使用权重变量时，JMP 对中性值使用加权均值而 SAS PROC GLM 使用未加权均值。

有效假设检验

通常，JMP 生成的假设检验与大多数其他可信程序（如 SAS PROC GLM，它使用假设类型 III 和 IV）的假设检验是一致的。以下两节说明它们的区别。

在 SAS PROC GLM 中，通过查看一般形式的可估计函数并找到仅涉及关注的效应和关注效应包含的效应的函数，以构造 III 和 IV 类假设检验 (Goodnight 1978)。

在 JMP 中构造相同的检验。但是因为参数化方式不同，可以通过对某效应的所有参数执行联合检验来检验该效应（假定现在为满秩）。检验不涉及包含交互作用参数，因为编码已使它们不涉及对包含的效应的检验。

若有缺失单元格或其他奇异性， JMP 检验与 GLM 检验不同。有几种描述它们的方式：

- JMP 检验等价于检验最小二乘均值存在差异，至少对于主效应是这样。若无法估计最小二乘均值，则检验不能包括一些比较，因此会损失自由度。对于交互作用，JMP 检验最小二乘均值差异是否刚好超过模型中所包含效应描述的边缘模式。
- JMP 通过以下方式检验效应：比较具有该效应的模型的 SSE 和没有该效应的模型的 SSE。JMP 参数化模型以使该方法变得有意义。
- JMP 实现**有效假设检验**（如 Hocking (1985, pp. 80–89, 163–166) 中所述），尽管 JMP 使用结构参数化而非单元格均值参数化。有效假设检验从效应所需的假设开始检验，并包括该检验的“尽量多的内容”。当然，若存在具有缺失单元格的包含效应，该检验必须去掉部分假设，因为完整的假设将是无法估计的。有效假设是从完整假设中删除尽量少的内容得来的。
- JMP 和 GLM（以及其他程序）中与是否存在缺失单元格有关的假设检验的差异不再被视为关注的检验。若交互作用是显著的，则不需要执行被包含的主效应的检验。若交互作用不是显著的，则随时可以从模型中删除它。一些检验甚至不是唯一的。若您重新标记缺失单元格设计中的水平，GLM IV 类检验会发生变化。

下一节详细说明该主题。

名义型效应中的奇异性和缺失单元格

考虑设计列中存在线性相依性的情况。使用 JMP 编码时，这种情况不会发生，除非没有足够的数据来填充需要估计的组合，或除非存在效应的某种混杂或共线性。

存在线性相依性时，参数的最小二乘解可能不是唯一的，无法执行一些假设检验。当发现参数的设计列与模型中前面的效应是线性相依时，JMP 选择的策略是按顺序将参数估计值设置为零。报表中的特殊列显示哪些参数估计值清零以及可以估计哪些参数估计值。单独的**奇异性**报表显示存在哪些线性相依性。

若存在奇异性，JMP 所检验的假设会不同于 GLM 选择的假设。与 GLM 相比，通常 JMP 找到更少的自由度进行检验，因为它将检验保持为更高的边缘标准。换句话说，JMP 检验始终对应于该效应的最小二乘均值的检验，但是 GLM 检验有时并不具有该特性。

例如，考虑具有交互作用和一个缺失单元格的双因子模型，其中 A 有三个水平，B 有两个水平，A3B2 单元格缺失。

表 A.7 具有交互作用的双因子模型

A B	A1	A2	B1	A1B1	A2B1	
A1 B1	1	0	1	1	0	
A2 B1	0	1	1	0	1	
A3 B1	-1	-1	1	-1	-1	
A1 B2	1	0	-1	-1	0	
A2 B2	0	1	-1	0	-1	
A3 B2	-1	-1	-1	1	1	假定该交互作用缺失。

每个单元格的期望值为：

表 A.8 期望值

	B1	B2
A1	$\mu + \alpha_1 + \beta_1 + \alpha\beta_{11}$	$\mu + \alpha_1 - \beta_1 - \alpha\beta_{11}$
A2	$\mu + \alpha_2 + \beta_1 + \alpha\beta_{21}$	$\mu + \alpha_2 - \beta_1 - \alpha\beta_{21}$
A3	$\mu - \alpha_1 - \alpha_2 + \beta_1 - \alpha\beta_{11} - \alpha\beta_{21}$	$\mu - \alpha_1 - \alpha_2 - \beta_1 + \alpha\beta_{11} + \alpha\beta_{21}$

显然，具有数据的任何单元格具有可估计的期望值。缺失的单元格具有无法估计的期望值。实际上，单元格的期望值恰好是奇异性报表中的设计列的以下线性组合

$$\mu - \alpha_1 - \alpha_2 - \beta_1 + \alpha\beta_{11} + \alpha\beta_{21}$$

假定您要构造比较 B1 和 B2 的最小二乘均值的检验。在本例中，上表中的行平均值给出这些最小二乘均值。

$$\begin{aligned}\text{LSM(B1)} = &(1/3)(\mu + \alpha_1 + \beta_1 + \alpha\beta_{11} + \\ &\mu + \alpha_2 + \beta_1 + \alpha\beta_{21} + \\ &\mu - \alpha_1 - \alpha_2 + \beta_1 - \alpha\beta_{11} - \alpha\beta_{21}) \\ = &\ \mu + \beta_1\end{aligned}$$

$$\begin{aligned}\text{LSM(B2)} = &(1/3)(\mu + \alpha_1 + -\beta_1 - \alpha\beta_{11} + \\ &\mu + \alpha_2 + -\beta_1 - \alpha\beta_{21} + \\ &\mu - \alpha_1 - \alpha_2 - \beta_1 + \alpha\beta_{11} + \alpha\beta_{21}) \\ = &\ \mu - \beta_1\end{aligned}$$

$$\text{LSM(B1)} - \text{LSM(B2)} = 2\beta_1$$

请注意，这说明针对 β_1 参数的检验等价于检验最小二乘均值是相同的。但是因为 β_1 无法估计，该检验无法进行，这意味着没有用于它的自由度。

现在，构造 A 水平上的最小二乘均值检验。

$$\begin{aligned}\text{LSM(A1)} = &(1/2)(\mu + \alpha_1 + \beta_1 + \alpha\beta_{11} + \mu + \alpha_1 - \beta_1 - \alpha\beta_{11}) \\ = &\ \mu + \alpha_1\end{aligned}$$

$$\begin{aligned}\text{LSM(A2)} = &(1/2)(\mu + \alpha_2 + \beta_1 + \alpha\beta_{21} + \mu + \alpha_2 - \beta_1 - \alpha\beta_{21}) \\ = &\ \mu + \alpha_2\end{aligned}$$

$$\begin{aligned}\text{LSM(A3)} = &(1/2)(\mu - \alpha_1 - \alpha_2 + \beta_1 - \alpha\beta_{11} - \alpha\beta_{21} + \\ &\mu - \alpha_1 - \alpha_2 - \beta_1 + \alpha\beta_{11} + \alpha\beta_{21}) \\ = &\ \mu - \alpha_1 - \alpha_2\end{aligned}$$

$$\text{LSM(A1)} - \text{LSM(A3)} = 2\alpha_1 + \alpha_2$$

$$\text{LSM(A2)} - \text{LSM(A3)} = 2\alpha_2 + \alpha_1$$

这些都是无法估计的，但是有另一个可以估计的比较，即比较没有缺失单元格的两个 A 列。

$$\text{LSM(A1)} - \text{LSM(A2)} = \alpha_1 - \alpha_2$$

该组合确实由 JMP 检验的（使用自由度为 1 的检验），尽管在效应中有两个参数。

通过计算奇异性组合的内积并检查它是否为零，可以验证可估性：

表 A.9 验证

	奇异性	组合
参数	组合	组合
m	1	0
a_1	-1	1
a_2	-1	-1
b_1	-1	0
ab_{11}	1	0
ab_{21}	1	0

结果是任意交互作用的缺失单元格的设计列始终剔除主效应的自由度（对于名义型因子）。因此，最小二乘均值的不可估性和检验对应于这些最小二乘均值的效应的自由度损失之间有直接关系。

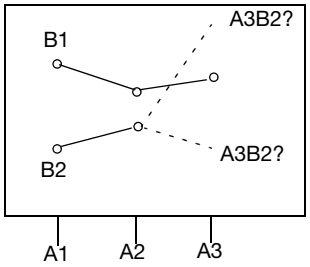
这如何与 GLM 所做的进行比较？当没有缺失单元格时，GLM 和 JMP 执行相同的检验。即，它们有效地检验最小二乘均值是相等的。但是当 GLM 遇到奇异性时，它以不同方式不再关注这些单元格，具体取决于它们属于类型 III 还是类型 IV。对于类型 IV，它查找可以找到的可估计组合。这些组合可能不是唯一的，若您重新排序水平，可能得到不同的结果。对于类型 III，它执行可估计函数的一些正交化来获得唯一检验。但是，根据单元格均值，检验可能不是很好解释的。

JMP 方法具有几个优点，尽管起初它与使用 GLM 相比可能损失更多自由度：

1. 这些检验在原理上与 LSM 关联。
2. 这些检验在计算上很简单，使用重新参数化的模型的简化平方和。
3. 这些检验与 Hocking 的 “有效假设检验”一致。
4. 这些检验是**整体边缘检验**，即它们始终完全涵盖交互作用中的其他效应。

最后一点需要更详尽阐述：我们来看上一示例（A3B2 具有缺失单元格）中单元格均值的期望值图。

图 A.1 单元格均值的期望值



该图显示具有缺失单元格的期望单元格均值。在 B 水平上刻画 A1 和 A2 单元格的均值。JMP 方法指示您不能检验具有缺失 A3B2 单元格的 B 主效应，因为缺失单元格的均值可能是交互作用项允许的任何值。若缺失单元格的均值是显示的更高值，B 效应可能是检验显著的。若它是更低的值，则可能是检验不显著的。该点是您不知道的点。它是声明最小二乘均值不可估计时它们要表达的内容。它也是效应的假设所说的那样 — 您不知道。

若您要检验涉及单元格子集的边缘的假设，则这是 GLM 类型 IV 要做的事。在 JMP 中，您必须通过使用很多计算分割效应或使用对比来自行构造这些检验。

JMP 和 GLM 假设

有缺失单元格时，GLM 工作方式不同于 JMP，它生成不同的假设检验。特别是，GLM 无法识别效应中嵌套和交叉的任何差异，但是 JMP 可以识别。假定您进行嵌套设计时，具有带不同水平数的三层嵌套：A、B(A) 和 C(A B)。

图 A.10 显示采用 GLM 参数的主效应 A 的检验。第一个列集是 JMP 执行的检验。第二个列集是 GLM 类型 IV 执行的检验。第三个列集等价于 JMP 执行的检验，它是前两列与以下矩阵相乘得到的：

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

它可与 GLM 检验进行比较。最后一个列集是 GLM 类型 III 检验。区别是检验在包含效应上的分布不同。在 JMP 中，它似乎更倾向于自上而下分层。在 GLM 类型 IV 中，检验似乎更倾向于由下而上。实际上，检验统计量通常是相似的。

表 A.10 GLM 和 JMP 假设的比较

参数	A 的 JMP 检验		A 的 GLM-IV 检验		JMP 旋转检验		A 的 GLM-III 检验	
u	0	0	0	0	0	0	0	0
a1	0.6667	-0.3333	1	0	1	0	1	0
a2	-0.3333	0.6667	0	1	0	1	0	1

表 A.10 GLM 和 JMP 假设的比较 (续)

a3	-0.3333	-0.3333	-1	-1	-1	-1	-1	-1
a1b1	0.1667	-0.0833	0.2222	0	0.25	0	0.2424	0
a1b2	0.1667	-0.0833	0.3333	0	0.25	0	0.2727	0
a1b3	0.1667	-0.0833	0.2222	0	0.25	0	0.2424	0
a1b4	0.1667	-0.0833	0.2222	0	0.25	0	0.2424	0
a2b1	-0.1667	0.3333	0	0.5	0	0.5	0	5
a2b2	-0.1667	0.3333	0	0.5	0	0.5	0	5
a3b1	-0.1111	-0.1111	-0.3333	-0.3333	-0.3333	-0.3333	-0.3333	-0.3333
a3b2	-0.1111	-0.1111	-0.3333	-0.3333	-0.3333	-0.3333	-0.3333	-0.3333
a3b3	-0.1111	-0.1111	-0.3333	-0.3333	-0.3333	-0.3333	-0.3333	-0.3333
a1b1c1	0.0833	-0.0417	0.1111	0	0.125	0	0.1212	0
a1b1c2	0.0833	-0.0417	0.1111	0	0.125	0	0.1212	0
a1b2c1	0.0556	-0.0278	0.1111	0	0.0833	0	0.0909	0
a1b2c2	0.0556	-0.0278	0.1111	0	0.0833	0	0.0909	0
a1b2c3	0.0556	-0.0278	0.1111	0	0.0833	0	0.0909	0
a1b3c1	0.0833	-0.0417	0.1111	0	0.125	0	0.1212	0
a1b3c2	0.0833	-0.0417	0.1111	0	0.125	0	0.1212	0
a1b4c1	0.0833	-0.0417	0.1111	0	0.125	0	0.1212	0
a1b4c2	0.0833	-0.0417	0.1111	0	0.125	0	0.1212	0
a2b1c1	-0.0833	0.1667	0	0.25	0	0.25	0	0.25
a2b1c2	-0.0833	0.1667	0	0.25	0	0.25	0	0.25
a2b2c1	-0.0833	0.1667	0	0.25	0	0.25	0	0.25

表 A.10 GLM 和 JMP 假设的比较 (续)

a2b2c2	-0.0833	0.1667	0	0.25	0	0.25	0	0.25
a3b1c1	-0.0556	-0.0556	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667
a3b1c2	-0.0556	-0.0556	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667
a3b2c1	-0.0556	-0.0556	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667
a3b2c2	-0.0556	-0.0556	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667
a3b3c1	-0.0556	-0.0556	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667
a3b3c2	-0.0556	-0.0556	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667	-0.1667

从 JMP 参数化的角度， A 的检验为：

表 A.11 A 的检验

参数	GLM–IV 检验		JMP 检验	
m	0	0	0	0
a13	2	1	1	0
a23	1	2	0	1
a1:b14	0	0	0	0
a1:b24	0.11111	0	0	0
a1:b34	0	0	0	0
a2:b12	0	0	0	0
a3:b13	0	0	0	0
a3:b23	0	0	0	0
a1b1:c12	0	0	0	0
a1b2:c13	0	0	0	0
a1b2:c23	0	0	0	0
a1b3:c12	0	0	0	0
a1b4:c12	0	0	0	0

表 A.11 A 的检验 (续)

a2b1:c13	0	0	0	0
a2b2:c12	0	0	0	0
a3b1:c12	0	0	0	0
a3b2:c12	0	0	0	0
a3b3:c12	0	0	0	0

因此，从 JMP 的角度看，GLM 检验有点奇怪，它对 a1b24 参数设置系数。

有序型因子

使用有序型建模类型标记的因子的编码方式不同于名义型因子的编码方式。这些参数估计值的解释不一样，检验不同并且最小二乘均值不同。

对于有序型因子，因子的第一个水平是控制或基线水平，参数测量将有序型因子设置为每个后续水平时对响应的影响。有序型因子编码适用于包含表示各种剂量水平的因子，其中第一个剂量为零。下表显示了一个三水平有序型因子的示例：

表 A.12 有序型因子

项	编码的列		
A	a2	a3	
A1	0	0	控制水平，零剂量
A2	1	0	低剂量
A3	1	1	更高剂量

设计的模式为：下三角处是 1，在别处为 0。

对于简单的主效应模型，这可以指定如下：

$$y = \mu + \alpha_2 X_{(a \le 2)} + \alpha_3 X_{(a \le 3)} + \varepsilon$$

请注意，μ 是 $A = 1$ 处的期望响应，$\mu + \alpha_2$ 是 $A = 2$ 处的期望响应，$\mu + \alpha_2 + \alpha_3$ 是 $A = 3$ 处的期望响应。因此，α_2 估计从 $A = 1$ 移到 $A = 2$ 的效应，α_3 估计从 $A = 2$ 移到 $A = 3$ 的效应。

若有序型主效应的所有参数具有相同符号，则响应效应在有序型水平上是单调的。

有序型交互作用

有序型交互作用与名义型效应一样，从因子各列的水平方向的直积得来。考虑一个示例，它有两个有序型因子 A 和 B，每个因子具有三个水平。JMP 的有序型编码生成下面所示的设计矩阵。交互作用的模式是下三角矩阵为 1 的分块下三角矩阵。

表 A.13 有序型交互作用

						A*B			
						A2		A3	
A	B	A2	A3	B2	B3	B2	B3	B2	B3
A1	B1	0	0	0	0	0	0	0	0
A1	B2	0	0	1	0	0	0	0	0
A1	B3	0	0	1	1	0	0	0	0
A2	B1	1	0	0	0	0	0	0	0
A2	B2	1	0	1	0	1	0	0	0
A2	B3	1	0	1	1	1	1	0	0
A3	B1	1	1	0	0	0	0	0	0
A3	B2	1	1	1	0	1	0	1	0
A3	B3	1	1	1	1	1	1	1	1

注意：当您检验是否没有效应时，对于简单模型的名义型和有序型因子没有太大的差异。但是，指定交互作用时，就有很大差异了。对于大多数模型，我们建议您使用名义型因子而非有序型因子。

有序型交叉模型的假设检验

要了解参数的含义，请查看这个使用参数的期望单元格均值的表，其中 μ 是截距，α_2 是水平 A2 的参数等等。

表 A.14 期望单元格均值

	B1	B2	B3
A1	μ	$\mu + \beta_2$	$\mu + \beta_2 + \beta_3$
A2	$\mu + \alpha_2$	$\mu + \alpha_2 + \beta_2 + \alpha\beta_{22}$	$\mu + \alpha_2 + \beta_2 + \beta_3 + \alpha\beta_{22} + \alpha\beta_{23}$
A3	$\mu + \alpha_2 + \alpha_3$	$\mu + \alpha_2 + \alpha_3 + \beta_2 + \alpha\beta_{22} + \alpha\beta_{32}$	$\mu + \alpha_2 + \alpha_3 + \beta_2 + \beta_3 + \alpha\beta_{22} + \alpha\beta_{23} + \alpha\beta_{32} + \alpha\beta_{33}$

请注意，A 的主效应检验实际上检验 B 固定在第一个水平时的 A 水平。类似地，B 的主效应检验将 A 固定在第一个水平下在 B 的各个水平的顶部行之间检验。该检验适用于两个因子是不同治疗的剂量的实验。主要问题是每个治疗本身的效力，当采用两种药物的剂量时，很少有人关注**药物交互作用**。在某些情况下，每种药物采用大剂量还可能是危险的。

请注意，可以通过添加与自身左上方的每个单元格（包括当前行和列）关联的所有参数，得到每个单元格的期望值。最后一个单元格的期望值是所有参数之和。

尽管包含在其他效应中的效应的假设检验在有序型和名义型编码方面不同，但是不包含在其他效应中的效应的检验是相同的。在上述交叉设计中，无论 A 和 B 是名义拟合还是有序拟合，交互作用的检验都是相同的。

有序型最小二乘均值

如前文所述，在将所有其他因子设置为某个中性值后，最小二乘均值是对应于某个水平组合的预测值。JMP 将具有未涉及的有序型因子的效应的中性值定义为在第一个水平（即“控制”或**基线**水平）下的效应。

这个有序型因子的最小二乘均值定义主张该观点：被包含效应的假设检验等价于检验最小二乘均值是相等的。

有序型效应中的奇异性和缺失单元格

使用有序型编码，您将指定有序型效应的第一个水平是基线。因此可以获得对主效应的良好检验，即使当交互作用中有缺失单元格时（即使您没有交互作用的数据时）也是如此。

具有缺失单元格的示例

该示例与上例相同，每个单元格有两个观测值，但是 A3B2 单元格没有数据。您现在可以比较以名义方式对因子编码时的结果和以有序方式对因子编码时的结果。模型拟合是相同的，如图 A.2 中所示。

表 A.15 观测值

Y	A	B
12	1	1
14	1	1
15	1	2
16	1	2
17	2	1
17	2	1
18	2	2
19	2	2
20	3	1
24	3	1

图 A.2 名义型拟合 （左）和有序型拟合 （右）的汇总信息

拟合汇总

R 方	0.891732
调整 R 方	0.805118
均方根误差	1.48324
响应均值	17.2
观测数（或权重和）	10

方差分析

源	自由度	平方和	均方	F 比
模型	4	90.60000	22.6500	10.2955
误差	5	11.00000	2.2000	概率>F
校正总和	9	101.60000		0.0125*

拟合汇总

R 方	0.891732
调整 R 方	0.805118
均方根误差	1.48324
响应均值	17.2
观测数（或权重和）	10

方差分析

源	自由度	平方和	均方	F 比
模型	4	90.60000	22.6500	10.2955
误差	5	11.00000	2.2000	概率>F
校正总和	9	101.60000		0.0125*

因为编码方式不同，参数估计值相差很大。请注意，缺失单元格影响某些名义型参数的可估性但是不影响任何有序型参数。

图 A.3 名义型拟合 （左）和有序型拟合 （右）的参数估计值

参数估计值

项		估计值	标准误差	t 比	概率>|t|
截距		17.333333	0.60553	28.63	<.0001*
A[1]		-4.333333	0.856349	-5.06	0.0039*
A[2]		-0.333333	0.856349	-0.39	0.7131
B[2-1]	有偏	1.5	1.48324	1.01	0.3583
A[1]*B[2-1]	有偏	1	2.097618	0.48	0.6537
A[2]*B[2-1]	清零	0	0	.	.

参数估计值

项		估计值	标准误差	t 比	概率>|t|
截距		17.333333	0.60553	28.63	<.0001*
A[1]		-4.333333	0.856349	-5.06	0.0039*
A[2]		-0.333333	0.856349	-0.39	0.7131
B[2-1]	有偏	1.5	1.48324	1.01	0.3583
A[1]*B[2-1]	有偏	1	2.097618	0.48	0.6537
A[2]*B[2-1]	清零	0	0	.	.

奇异性详细信息显示线性相依性（也通过检查这些值来标识缺失单元格）。

图 A.4 名义型拟合 （左）和有序型拟合 （右）的奇异性详细信息

奇异性详细信息

项	详细信息
B[2-1]	=A[1]*B[2-1] + A[2]*B[2-1]

奇异性详细信息

项	详细信息
A[3-2]*B[2-1]	= 0

名义型的效应检验损失自由度。在 B 中，没有检验。对于有序型，没有损失，因为对于基本的第一个水平没有缺失单元格。

图 A.5 名义型拟合 （左）和有序型拟合 （右）的效应检验

效应检验

源	参数数目	自由度	平方和	F 比	概率>F	
A	2	2	81.333333	18.4848	0.0049*	
B	1	0	0.000000	.	.	损失自由度
A*B	2	1	0.500000	0.2273	0.6537	损失自由度

效应检验

源	参数数目	自由度	平方和	F 比	概率>F	
A	2	2	81.333333	18.4848	0.0049*	
B	1	1	6.250000	2.8409	0.1527	
A*B	2	1	0.500000	0.2273	0.6537	损失自由度

最小二乘均值也不同。有些名义型 LSM 不可估计，但是有序型 LSM 都可以估计。您可以通过查看单元格均值来验证这些值。请注意 A*B LSM 对于名义型和有序型是相同的。图 A.6 显示名义型和有序型拟合的最小二乘均值。

图 A.6 名义型拟合 （左）和有序型拟合 （右）的最小二乘均值

效应详细信息

A

“最小二乘均值”表

水平	最小二乘均值	标准误差	均值
1	13.000000	1.0488088	14.2500
2	17.000000	1.0488088	17.7500
3	22.000000	1.0488088	22.0000

B

“最小二乘均值”表

水平	最小二乘均值		标准误差	均值
1	17.333333		0.60553007	17.3333
2	0.000000	不可估计	.	17.0000

A*B

“最小二乘均值”表

水平	最小二乘均值		标准误差
1,1	13.000000		1.0488088
1,2	15.500000		1.0488088
2,1	17.000000		1.0488088
2,2	18.500000		1.0488088
3,1	22.000000		1.0488088
3,2	0.000000	不可估计	.

效应详细信息

A

“最小二乘均值”表

水平	最小二乘均值	标准误差	均值
1	13.000000	1.0488088	14.2500
2	17.000000	1.0488088	17.7500
3	22.000000	1.0488088	22.0000

B

“最小二乘均值”表

水平	最小二乘均值	标准误差	均值
1	13.000000	1.0488088	17.3333
2	15.500000	1.0488088	17.0000

A*B

“最小二乘均值”表

水平	最小二乘均值		标准误差
1,1	13.000000		1.0488088
1,2	15.500000		1.0488088
2,1	17.000000		1.0488088
2,2	18.500000		1.0488088
3,1	22.000000		1.0488088
3,2	0.000000	不可估计	.

频数

频数 （包括具有非整数值的那些频数）对分析的影响由它们对损失函数的影响来解释。假定您要使用响应值 y_i 和预测变量 $x_{i1}, x_{i2},..., x_{in}$ 来估计参数 θ。假定损失函数 （没有频数变量）满足以下方程：

$$L(\theta|\underset{\sim}{y}) = \sum_{i=1}^{n} L(\theta|y_i, x_{i1}, x_{i2},\ldots,x_{in})$$

若定义频数 f_i，则损失函数为：

$$L(\theta|\underset{\sim}{y},\underset{\sim}{f}) = \sum_{i=1}^{n} f_i L(\theta|y_i, x_{i1}, x_{i2},\ldots,x_{in})$$

所有基于推断的量计算（如参数估计值、标准误差、假设检验和置信区间）都基于该形式的损失函数。

常用假设

在相信统计学前，再次确认自己了解所用方法的价值和局限性。统计方法仅仅是工具，它们无法保证您不采用不正确的理论知识（无效的统计假设）或错误的数据。

假设模型

大多数统计量基于模型是正确的这个假设。在某种程度上，您的模型可能不是正确的，不要盲目相信源自该模型的统计报表。

相对显著性

很多统计检验无法绝对评估模型。显著性检验统计量可能仅说明该模型拟合效果比某些简化模型（如均值）好。该模型可以拟合数据但是可能不能很好描述基础的物理模型。

多个推断

通常您不能直接相信统计结果的值，它们只是提供了帮助发现的钥匙。要确认这些发现，您可能需要进一步研究。否则，您可能只是筛查了一下数据。

例如，若您做过详尽的分析，您会发现，即使是因子没有预测价值，您的研究中 5% 会有显著效应（5% 水平上的显著性）。同样，当您是使用自己的数据来构造模型（而非检验数据的正确模型）时，在这个意义上您侵蚀腐败了报表中的显著性水平。还有随机误差会影响您的模型选择并导致您相信自己的模型比它的实际效果好。

合法性评估

评估模型合法性的方法和模式有多种：

- 可以使用失拟检验针对因子的饱和版本检查模型合法性。若您在非饱和模型中具有重复的 x 值，“拟合模型”平台会自动提供这些检验。
- 可以通过从“拟合模型”平台查看残差图和学生化残差图检查连续响应的分布假设。或者使用平台弹出菜单中的**保存**命令在数据表列中保存残差。然后对这些列使用**分析 > 分布**以查看直方图以及它的正态曲线和正态分位数图。残差不是很独立的，但是您可以大致识别严重的非正态分布。
- 连续响应的最佳综合性诊断工具是杠杆图，因为它显示每个点对每个假设检验的影响。若您怀疑数据中有错误的值，该图可以帮助确定单个点是否严重影响统计检验。
- 建议扫描您的数据来查找离群值并检查它们是否是有效的观测值。可以在“分布”平台报表和图中识别一元离群值。二元离群值显示在“以 X 拟合 Y”散点图和多元散点图矩阵中。您可以在**图形 > 三维散点图**生成的三维图中查看三元离群值。可以使用“主成分”或“三维散点图”以及在“多元”平台中计算和标绘的 Mahalanobis 和 jack-knifed 距离来查找高维的离群值。

替代方法

统计文献介绍了特殊的非参数和稳健的方法，但是 JMP 目前仅实现了其中的一部分。这些方法需要更少的分布假设（非参数），因此更抗污染（稳健）。但是，它们更难以成为一般性方法，而且针对检验统计量的小样本概率可能需要很长时间完成计算。

若您想进行线性秩检验且只要求满足正态大样本显著性近似条件，可以分析数据的秩以执行 Wilcoxon 秩和或 Kruskal-Wallis 单因子检验的等价检验。

若您不确定连续响应是否充分满足正态假设条件，可以将建模类型从连续型更改为有序型，然后安全地进行分析，即使这个方法会牺牲一些展示方面的丰富性以及降低一些统计功效。

关键统计概念

有两个关键概念，它们统一经典统计量并将统计属性和拟合原理包装为可进行可视化的形式：

- 统一的不确定性概念
- 两个基本拟合装置

这两个思想有助于使用建立在数学统计学基础上的直观概念来理解统计学。

统计对科学的意义就像会计对企业的意义一样。它是权衡观测数据的技能。统计检验就像信用审核一样。但是统计工具还可以做更多工作。通过它们，可以揭示有关数据的意外模式并促使提出新见解。在使用这些功能强大的工具前，您需要了解一下它们的工作机制。

不确定性，统一的概念

做会计核算时，您合计所有金额来获得汇总信息。在查看具有不确定性或噪声的科学观测值时，需要采用一些统计测度来汇总数据。像金额是可以累加的一样，若您选择了正确的测度，不确定性也可以累加。

最好的测度不是直接概率，因为要得到联合概率，您必须假定观测是独立的，然后将概率相乘而非相加。取每个概率的对数更简单，因为您可以对它们求和，总和是联合概率的对数。

但是，概率的对数是负数，因为它是介于 0 到 1 的数值的对数。为了使数值为正，JMP 使用概率的负对数。随着概率变得更小，其负对数变得更大。该测度称为不确定性，它与概率的测量方式相反。

在商业上，您想使收入最大化而使成本最小化。在科学上，您想最大程度降低不确定性。不确定性在科学中的作用就像成本在商业中的作用一样。所有统计方法拟合模型以使不确定性最小。

将不确定性可视化并不困难。请设想抛一系列硬币的情形：每次抛是独立的。抛出正面的概率为 0.5，其以 2 为底的对数 -log(0.5) 为 1。连续抛出 h 次正面的概率定义如下：

$$p = \left(\frac{1}{2}\right)^h$$

对 h 求解将生成以下结果：

$$h = -\log_2 p$$

您可以将某个事件的不确定性设想为，为了得到同等罕见的事件，您在抛掷中必须连续得到“正面”的次数。

在统计学上我们做的每件事几乎都有不确定性，-log$p$。统计文献将不确定性表示为**负对数似然**。

两个基本拟合装置

有关统计拟合的一个令人吃惊的事实是大多数经典方法可以简化为使用两个简单装置：弹簧和压力缸。

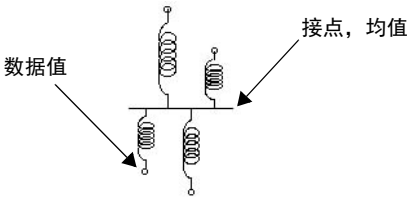
弹簧

首先，弹簧是连续响应模型的拟合装置 (Farebrother 1987)。假定您有 n 个点，想知道这些点的期望值（均值）。设想您将点排列在标尺上并使用弹簧将它们连接到公共接头（图 A.7）。当您放开弹簧时，弹簧上下摆动接点，然后在均值处停下来。根据物理学原理，这是必然发生的事。

若数据服从正态分布且均值在连接弹簧的接点处，则每个点的弹簧的物理能量与数据点的不确定性成比例。您为计算弹簧的能量（不确定性）所要做的是计算每个点到均值的距离平方和。

为了选择将最小不确定性归因于观测数据的估计值，选择弹簧停摆点为均值的估计值。这个点要求拉伸弹簧的能量最小，等价于最小二乘拟合。

图 A.7 将弹簧连接到数据点

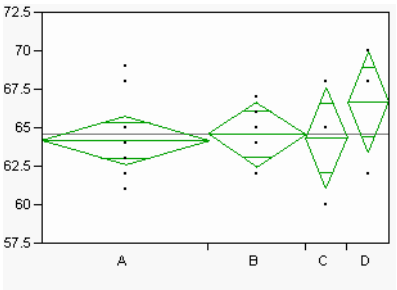


那就是您如何拟合一个均值或拟合几个均值的。那就是您如何拟合一条线、平面或超平面的。那就是您如何拟合连续数据的几乎任何模型的。通过弹簧必须延伸的距离的平方和来测量能量或不确定性。

统计学家信任正态分布，因为它对其信仰的要求是最低的分布。它在某种意义上来说是最随机的。它是具有最无信息形状的分布。它是具有给定方差的最期望的不确定性的分布。它是用平方距离测量不确定性的分布。在很多情况下，当您具有混合分布或独立量之和时，它是极限分布。它是导致可以轻松测量检验统计量的分布。

拟合受假设约束时，您通过测量这个相同的弹簧能量来检验假设。假定您在实验中有来自四个不同处理的响应，想检验均值是否有显著差异。首先，假定您的数据已分组标绘（如图 A.8 中所示），但是使用弹簧连接每个处理的单独均值。然后施加压力来抗拒弹力，使各个均值向公共均值移动。移好了！限制均值成为相同的能量大小就是所需的检验统计量。该能量是检验均值是否相同的假设的 *F* 检验的主要组成部分。

图 A.8 连续响应变量的单因子图



压力缸

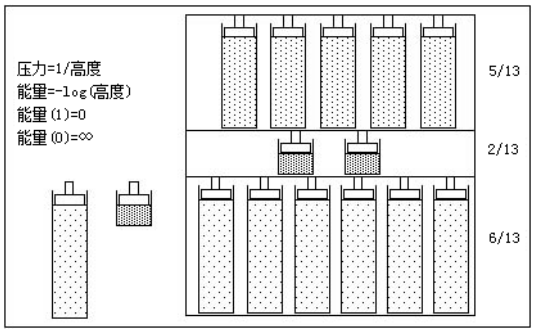
您的响应是分类型而非连续型时怎么办？例如，假定响应是小汽车样本的原产国。对于您的样本，有针对三个响应水平 （美国、欧洲和日本）的概率。您可以将原产国的这些概率设置为某个估计值，然后计算数据的不确定性。通过求数据所给响应的概率的负对数之和来计算这个不确定性。其定义如下：

$$H = \sum h_{y(i)} = -\sum \log p_{y(i)}$$

弹簧的构想说明了如何拟合连续数据的均值。响应为分类型时，统计方法直接估计响应概率并选择使数据的总不确定性最小的估计值。概率估计值必须非负且总和为 1。您可以将响应概率绘图为总长度为 1 的标尺上的一个组成部分。对于每个响应观测值，把一个气压缸（例如轮胎气泵）加载到其响应区域。改变响应水平之间的分割方式，直到达到最低势能的平衡。根据分割区的大小来估计响应概率。

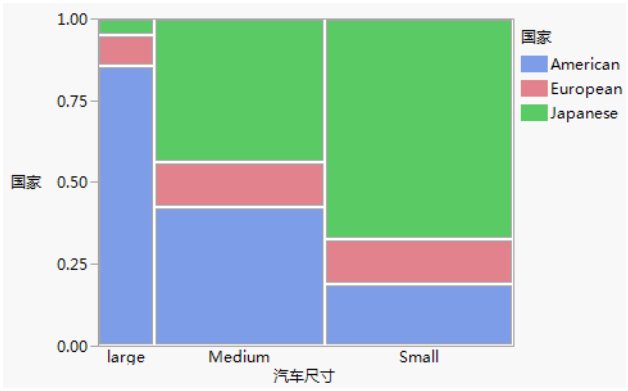
图 A.9 显示诸如 medium 类型车的单个类别的情形（请参见图 A.10 中Carpoll.jmp 中标记为 medium 的马赛克列）。假定有 13 个响应（小汽车）。第一个水平 (American) 具有 6 个响应，第二个水平具有 2 个响应，最后一个水平具有 5 个响应。响应概率分别变为 6/13、2/13 和 5/13，因为针对响应分割区的压力相抵以使总能量最小。

图 A.9 分割区中的压力缸效应



像弹簧用于连续数据一样，您可以通过一些因子来分割样本并拟合单独的分割集。然后通过测量需要多少额外能量来促使分割区相等，检验响应率在各组上是否相同。设想按车尺寸对车原产国概率分组的压力缸。强制每个组中的分割区水平对齐所需的能量检验这些变量是否有相同的概率。图 A.10 显示了这些分割区。

图 A.10 分类数据的马赛克图



似然、AICc 和 BIC

JMP 中的很多统计模型使用称为**最大似然**的方法进行拟合。该方法通过最大化似然函数设法估计模型的参数（它通常用 β 表示）。似然函数（用 L(β) 表示）是在观测的数据值处求得的概率密度函数 （或用于离散分布的概率质量函数）之积。给定观测的数据，最大似然估计设法找到使 L(β) 最大化的参数 β 的值。

不必使似然函数 L(β) 最大化，使用似然函数的自然对数的负数 *-Log* L(β) 更方便。使 L(β) 最大化的问题转换为最小化问题，此时您设法使负对数似然（-对数似然 = *-Log* L(β)）最小化。因此，较小的负对数似然值或负对数似然值的两倍（-2对数似然）指示更好的模型拟合。

通过使用 JMP 中不同的平台，您可以使用负对数似然的值选择模型并执行比较模型拟合效果的定制假设检验。这通过使用似然比检验来完成。在很多 JMP 平台中报告 -2 对数似然的一个原因是完全模型和简化模型 -2 对数似然值的差值是逐近的卡方分布。与这个似然比检验关联的自由度等于两个模型中参数数目的差值 (Wilks 1938)。

校正的 Akaike 信息准则 (AICc) 和 Bayesian 信息准则 (BIC) 是基于信息的评估模型拟合的准则。两者均基于 -2 对数似然。

按以下方式定义 AICc：

$$\text{AICc} = -2LogLikelihood + 2k + 2k(k+1)/(n-k-1)$$

其中，k 是模型中估计参数的个数，n 是模型中使用的观测数。可以使用该值比较同一数据集的各种模型以确定最佳拟合模型。具有最小值的模型，如 Akaike (1974) 中所述，通常是首选模型。

按以下方式定义 BIC：

$$\text{BIC} = -2LogLikelihood + k\ln(n)$$

其中，k 是模型中估计参数的个数，n 是模型中使用的观测数。比较两个模型的 BIC 值时，BIC 值更小的模型被认定为拟合效果更好。

对使用更多参数的模型，BIC 通常比 AICc 对模型惩罚更多。因此，这导致与 AICc 相比，BIC 选择更简化的模型，即具有更少参数的模型。有关 AICc 和 BIC 的详细比较，请参见 Burnham and Anderson (2004)。

最小二乘回归中 AICc 和 BIC 的简化公式

在最小二乘回归的情况下，也可以基于误差平方和 (SSE) 计算 AICc 和 BIC。根据 SSE，AICc 和 BIC 定义如下：

$$\text{AICc} = n\ln\left(\frac{SSE}{n}\right) + 2k + 2k(k+1)/(n-k-1) + n\ln(2\pi) + n$$

$$\text{BIC} = n\ln\left(\frac{SSE}{n}\right) + k\ln(n) + n\ln(2\pi) + n$$

其中，*k* 是模型中估计参数的个数，*n* 是模型中使用的观测数，*SSE* 是模型中的误差平方和。

功效计算

以下各节给出计算最小显著数 (LSN)、最小显著值 (LSV)、功效和调整功效的公式。除 LSV 以外，为每个效应和用户指定的对比集合（在“定制检验”和“最小二乘均值对比”下）提供这些计算。仅为单个线性对比计算 LSV。在下面的详细介绍中，**假设**是指相关的对比集合。

- “LSN 的计算”
- “LSV 的计算”
- “功效的计算”
- “调整功效的计算”

LSN 的计算

最小显著数 (LSN) 求解方程中的 *N*：

$$\alpha = 1 - FDist\left[\frac{N\delta^2}{df_{Hyp}\sigma^2}, df_{Hyp}, N - df_{Hyp} - 1\right]$$

其中

FDist 是中心 *F* 分布的累积分布函数

df_{Hyp} 表示假设的自由度

σ^2 是误差方差

δ^2 是效应大小的平方

对于回顾分析，通过用假设的平方和除以当前样本大小 *n* 来估计 δ^2。若检验针对一个效应，则用该效应的平方和除以当前研究中的观测数来估计 δ^2。对于回顾研究，用均方误差来估计误差方差 σ^2。将这些估计值以及 α 值 0.05 作为默认值输入“功效详细信息”窗口。

当您执行前瞻分析来规划将来的研究时，考虑确定将实现指定功效的样本大小（请参见第 479 页的“功效的计算”。）

LSV 的计算

仅为单个线性对比计算最小显著值 (LSV)。

单个线性对比的检验

考虑自由度为 1 的检验 $L\beta = 0$，其中 L 是常数的行向量。该假设的 t 检验的检验统计量为：

$$\frac{Lb}{s\sqrt{L(X'X)^{-1}L'}}$$

其中 s 是均方根误差。若检验统计量的绝对值超过 t 分布的 1 - α/2 分位数 $t_{1-\alpha/2}$（自由度等于误差自由度），则在显著性水平 α 上拒绝该假设。

要查找最小显著值 （用 $(Lb)^{LSV}$ 表示），我们求 Lb 的解：

$$(Lb)^{LSV} = t_{1-\alpha/2}s\sqrt{L(X'X)^{-1}L'}$$

单个参数的检验

在线性对比检验一个假设 （设置单个 β_i 等于 0）的特例中，这简化为：

$$b_i^{LSV} = t_{1-\alpha/2}s\sqrt{(X'X)_{ii}^{-1}} = t_{1-\alpha/2}StdError(b_i)$$

均值差异的检验

在相关检验是比较两组均值的情形中，文献提到了**最小显著性差异 (LSD)**。在模型只包含一个名义型变量的特例中，检验单个线性对比的公式简化为 LSD 的公式：

$$\text{LSD} = t_{1-\alpha/2}s\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}$$

但是，在 JMP 中，与名义型效应的水平关联的参数测量该水平的均值和所有水平的均值的差异。因此，用于这类比较的 LSV 是用于均值差异的 LSD 的一半。

注意：若您在检验名义型效应水平上的对比，请记住 JMP 是如何编码名义型效应的。即，与给定水平关联的参数测量针对所有水平平均值的差异。

功效的计算

假定您想计算某线性假设的检验功效，指定了显著性水平为 α，样本大小为 N。您要检测大小 δ 的效应。

要计算功效，首先求得线性假设的 α 水平 F 检验的临界值。这通过求解以下方程中的 F_C 得到：

$$\alpha = 1 - FDist[F_C, df_{Hyp}, N - df_{Model} - 1]$$

其中，df_{Hyp} 表示假设的自由度，df_{Model} 表示模型的自由度，N 是建议的（或实际的）样本大小。

然后计算与所需的效应大小关联的非中心参数。非中心参数定义如下：

$$\lambda = (N\delta^2)/\sigma^2$$

其中 σ^2 是误差方差的建议（估计）值。

给定大小 δ 的效应，检验统计量具有非中心 *F* 分布（在下面用 *FDist* 表示分布函数），它具有非中心参数 λ。要获得检验的功效，计算检验统计量超过临界值的概率：

$$\text{功效} = 1 - FDist\left[F_C, df_{Hyp}, N - df_{Model} - 1, \frac{N\delta^2}{\sigma^2}\right]$$

为了获得具有 *n* 个观测值的研究的回顾功效，JMP 根据 $\lambda = SS_{Hyp}/\hat{\sigma}^2$ 估计非中心参数 $\lambda = (n\delta^2)/\sigma^2$，其中 SS_{Hyp} 表示因假设产生的平方和。

调整功效的计算

调整功效的计算 (Wright and O'Brien 1988) 仅与回顾功效分析有关。执行调整功效的计算时，使用非中心参数估计值，该值已经过调整，消除了在简单地用样本估计值替代参数时导致的正偏倚。

非中心参数 λ 的估计值通过用样本估计值来估计 δ 和 σ 得到，它的公式计算如下：

$$\hat{\lambda} = SS_{Hyp}/MSE$$

Wright and O'Brien (1988) 说明了如何用以下公式给出非中心参数的无偏估计值：

$$[\hat{\lambda}(df_{Error} - 2)/df_{Error}] - df_{Hyp} = \frac{\hat{\lambda}(N - df_{Model} - 1 - 2)}{N - df_{Model} - 1} - df_{Hyp}$$

右侧的表达式说明当样本大小 *N*（与研究大小 *n* 不同时）被建议用于回顾功效分析时，如何计算无偏非中心参数。其中 df_{Hyp} 表示假设的自由度，df_{Model} 表示整体模型的自由度。

遗憾的是，这样调整非中心估计值可能导致负值，而将负值设置为零又会带来轻微的偏倚。调整的非中心估计值为

$$\hat{\lambda}_{\text{调整}} = \text{Max}\left[0, \frac{\hat{\lambda}(N - df_{Model} - 1 - 2)}{N - df_{Model} - 1} - df_{Hyp}\right]$$

调整的功效为

$$\text{功效}_{\text{调整}} = 1 - FDist[F_C, df_{Hyp}, N - df_{Model} - 1, \hat{\lambda}_{\text{adj}}]$$

按 Dwass (1955) 所述的方法来构造非中心参数的置信限：

$$\lambda \text{ 的置信下限} = \text{Max}\left[0, [(\sqrt{SS_{\text{Hyp}}/MSE}) - \sqrt{df_{Hyp}F_C}]^2\right]$$

$$\lambda \text{ 的置信上限} = [(\sqrt{SS_{\text{Hyp}}/MSE}) - \sqrt{df_{Hyp}F_C}]^2$$

通过将 λ 的这些置信限替换为以下等式来获取功效的置信限：

$$\text{功效} = 1 - FDist[F_C, df_{Hyp}, N - df_{Model} - 1, \lambda]$$

带置信限的逆预测

逆预测从响应变量估计自变量的值。在生物鉴定问题中，带置信限的逆预测特别有用。在 JMP 中，您可以请求提供连续和二值响应模型的逆预测估计值。若响应是连续的，您可以请求提供单个响应或期望响应的置信限。

使用 Fieller 定理 (1954) 计算置信限，该定理基于以下逻辑。目标是在给定其他回归变量值和响应值的情况下预测单个回归变量的值以及它的置信限。

- 设 **b** 估计参数 β，因此 **b** 分布服从 N(β,V)。
- 设 **x** 为相关的回归变量值，要估计第 *i* 个值。
- 设 **y** 为响应值。

我们需要知道 **x**[i] 值的置信区域，以便在给出 **x** 的所有其他值时 $\beta' x = y$。

逆预测为

$$x[i] = \frac{y - \beta'_{(i)}x_{(i)}}{\beta[i]}$$

其中带括号的下标 (*i*) 指示忽略第 *i* 个分量。可以从以下关系等式得到置信区间：

$$(y - b'x)^2 < t^2 x'Vx$$

其中 t 是指定置信水平的 t 值。

方程

$$(y - b'x)^2 - t^2 x'Vx = 0$$

可以表示为 $z = x[i]$ 的二次形式：

$$gz^2 + hz + f = 0$$

其中

$$g = b[i]^2 - t^2 V[i,i]$$

$$h = -2yb[i] + 2b[i]b'_{(i)}x_{(i)} - 2t^2 V[i,(i)]'x_{(i)}$$

$$f = y^2 - 2yb'_{(i)}x_{(i)} + (b'_{(i)}x_{(i)})^2 - t^2 x_{(i)}'V_{(i)}x_{(i)}$$

根据 g、 h 和 f 的值 （满足不等式的一组值），逆预测的置信区间有很多形式：

- (ϕ_1, ϕ_2) 形式的一个区间，其中 $\phi_1 < \phi_2$
- 形式为 $(-\infty, \phi_1) \cup (\phi_2, \infty)$ 的两个不相交区间，其中 $\phi_1 < \phi_2$
- 整个实线 $(-\infty, \infty)$
- 只能是 $-(\infty, \phi)$ 或 (ϕ, ∞) 之一

Fieller 置信区间为整个实线时，提供 Wald 区间。

注意：计算逆预测的置信区间时，“以 X 拟合 Y”Logistic 平台和“拟合模型名义型 Logistic”特质使用 *t* 值。“拟合模型广义线性模型”特质以及 SAS/STAT 中的 PROC PROBIT 使用 *z* 值，它们给出不同结果。

附录 B

参考书目

Aitken, M. (1987).“Modelling Variance Heterogeneity in Normal Regression Using GLIM.”Journal of the Royal Statistical Society, Series C 36:332–339.

Akaike, H. (1974).“A New Look at the Statistical Model Identification.”IEEE Transactions on Automatic Control AC-19:716–723.

Anderson, T. W. (1958).An Introduction to Multivariate Statistical Analysis.New York: John Wiley & Sons.

Belsley, D. A., Kuh, E., and Welsch, R. E. (1980).Regression Diagnostics: Identifying Influential Data and Sources of Collinearity.New York: John Wiley & Sons.

Benjamini, Y., and Hochberg, Y. (1995).“Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.”Journal of the Royal Statistical Society, Series B 57:289–300.

Box, G. E. P. (1954).“Some Theorems on Quadratic Forms Applied in the Study of Analysis of Variance Problems, Part 2: Effects of Inequality of Variance and of Correlation between Errors in the Two-Way Classification.”Annals of Mathematical Statistics 25:484–498.

Box, G. E. P., and Cox, D. R. (1964).“An Analysis of Transformations.”Journal of the Royal Statistical Society, Series B 26:211–243.

Box, G. E. P., and Meyer, R. D. (1986).“An Analysis for Unreplicated Fractional Factorials.”Technometrics 28:11–18.

Box, G. E. P., and Meyer, R. D. (1993).“Finding the Active Factors in Fractionated Screening Experiments.”Journal of Quality Technology 25:94–105.

Burnham, K. P., and Anderson, D. R. (2004).“Multimodel Inference: Understanding AIC and BIC in Model Selection.”Sociological Methods and Research 33:261–304.

Burnham, K. P., Andersen, D. R., and Huyvaert, K. P. (2011).“AIC Model Selection and Multimodel Inference in Behavioral Ecology: Some Background, Observations, and Comparisons.”Behavioral Ecology and Sociobiology 65:23–35.

Candes, E., and Tao, T. (2007).“The Dantzig Selector: Statistical Estimation when p is Much Larger than n.”The Annals of Statistics 35:2313–2351.

Carroll, R. J., and Ruppert, D. (1988).Transformation and Weighting in Regression.London: Chapman & Hall.

Chilès, J.-P., and Delfiner, P. (2012).Geostatistics: Modeling Spatial Uncertainty.2nd ed. New York: John Wiley & Sons.

Cobb, G. W. (1998).Introduction to Design and Analysis of Experiments.New York: Springer-Verlag.

Cohen, J. (1977).Statistical Power Analysis for the Behavioral Sciences.New York: Academic Press.

Cole, J. W. L., and Grizzle, J. E. (1966).“Applications of Multivariate Analysis of Variance to Repeated Measures Experiments.”Biometrics 22:810–828.

Conover, W. J. (1999).Practical Nonparametric Statistics.3rd ed. New York: John Wiley & Sons.

Cook, R. D., and Weisberg, S. (1982).Residuals and Influence in Regression.New York: Chapman & Hall.

Cook, R. D., and Weisberg, S. (1983).“Diagnostics for Heteroscedasticity in Regression.”Biometrika 70:1–10.

Cornell, J. A. (1990).Experiments with Mixtures.2nd ed. New York: John Wiley & Sons.

Cox, D. R. (1972).“Regression Models and Life-Tables.” Journal of the Royal Statistical Society, Series B 34:187–220.

Cox, D. R., and Snell, E. J. (1989).The Analysis of Binary Data.2nd ed. London: Chapman & Hall.

Cressie, N. A. C. (1993).Statistics for Spatial Data.Rev. ed. New York: John Wiley & Sons.

Daniel, C. (1959).“Use of Half-Normal Plots in Interpreting Factorial Two-Level Experiments.”Technometrics 1:311–341.

Dunnett, C. W. (1955).“A Multiple Comparisons Procedure for Comparing Several Treatments with a Control.”Journal of the American Statistical Association 50:1096–1121.

Dwass, M. (1955).“A Note on Simultaneous Confidence Intervals.”Annals of Mathematical Statistics 26:146–147.

Efron, B. (1977).“The Efficiency of Cox’s Likelihood Function for Censored Data.”Journal of the American Statistical Association 72:557–565.

Farebrother, R. W. (1987).“Mechanical Representations of the L1 and L2 Estimation Problems.”In Statistical Data Analysis Based on L_1 Norm and Related Methods, edited by Y. Dodge, 455–464.Amsterdam: North-Holland.

Fieller, E. C. (1954).“Some Problems in Interval Estimation.”Journal of the Royal Statistical Society, Series B 16:175–185.

Firth, D. (1993).“Bias Reduction of Maximum Likelihood Estimates.”Biometrika 80:27–38.

Fleming, T. R., and Harrington, D. P. (1991).Counting Processes and Survival Analysis.New York: John & Sons.

Goodnight, J. H. (1978).Tests of Hypotheses in Fixed Effects Linear Models.Technical Report R–101, SAS Institute Inc., Cary, NC.

Goodnight, J. H., and Harvey, W. R. (1978).Least-Squares Means in the Fixed-Effects General Linear Models.Technical Report R-103, SAS Institute Inc., Cary, NC.

Goos, P., and Jones, B. (2011).Optimal Design of Experiments: A Case Study Approach.Chichester, UK: John Wiley & Sons.

Greenhouse, S. W., and Geisser, S. (1959).“On Methods in the Analysis of Profile Data.”Psychometrika 32:95–112.

Harrell, F. E. (1986).“The LOGIST Procedure.”In SUGI Supplemental Library Guide, Version 5 Edition.Cary, NC: SAS Institute Inc.

Harvey, A. C. (1976).“Estimating Regression Models with Multiplicative Heteroscedasticity.”Econometrica 44:461–465.

Hastie, T. J., Tibshirani, R. J., and Friedman, J. H. (2009).The Elements of Statistical Learning: Data Mining, Inference, and Prediction.2nd ed. New York: Springer-Verlag.

Hayter, A. J. (1984).“A Proof of the Conjecture That the Tukey-Kramer Method Is Conservative.”Annals of Statistics 12:61–75.

Heinze, G., and Schemper, M. (2002).“A Solution to the Problem of Separation in Logistic Regression.”Statistics in Medicine 21:2409–2419.

Hocking, R. R. (1985).The Analysis of Linear Models.Monterey, CA: Brooks/Cole.

Hoenig, J. M., and Heisey, D. M. (2001).“The Abuse of Power: The Pervasive Fallacy of Power Calculations for Data Analysis.”American Statistician 55:19–24.

Hoerl, A. (1962).“Application of Ridge Analysis to Regression Problems.”Chemical Engineering Progress 58:54–59.

Hoerl, A., and Kennard, R. (1970).“Ridge Regression: Biased Estimation for Non-orthogonal Problems.”Technometrics 12:55–67.

Hsu, J. C. (1992).“The Factor Analytic Approach to Simultaneous Inference in the General Linear Model.”Journal of Computational and Graphical Statistics 1:151–168.

Hsu, J. C. (1996).Multiple Comparisons: Theory and Methods.London: Chapman & Hall.

Hu, W., Laber, E. B., Barker, C., and Stefanski, L. A. (2019.“Assessing Tuning Parameter Selection Variability in Penalized Regression.”Technometrics 61:154–164.

Huber, P. J., and Ronchetti, E. M. (2009).Robust Statistics.2nd ed. John Wiley & Sons.

Hui, F., Warton, D., and Foster, S. (2015).“Tuning Parameter Selection for the Adaptive Lasso Using ERIC.”Journal of the American Statistical Association 110:262–269.

Huynh, H., and Feldt, L. S. (1970).“Conditions Under Which Mean Square Ratios in Repeated Measurements Designs Have Exact F-Distributions.”Journal of the American Statistical Association 65:1582–1589.

Huynh, H., and Feldt, L. S. (1976).“Estimation of the Box Correction for Degrees of Freedom from Sample Data in the Randomized Block and Split Plot Designs.”Journal of Educational Statistics 1:69–82.

Kackar, R. N., and Harville, D. A. (1984).“Approximations for Standard Errors of Estimators of Fixed and Random Effects in Mixed Linear Models.”Journal of the American Statistical Association 79:853–862.

Kalbfleisch, J. D., and Prentice, R. L. (2002).The Statistical Analysis of Failure Time Data.2nd ed. Hoboken, NJ: John Wiley & Sons.

Kenward, M. G., and Roger, J. H. (1997).“Small Sample Inference for Fixed Effects from Restricted Maximum Likelihood.”Biometrics 53:983–997.

Koenker, R., and Hallock, K. (2001).“Quantile Regression: An Introduction.”Journal of Economic Perspectives 15:143–156.

Kramer, C. Y. (1956).“Extension of Multiple Range Tests to Group Means with Unequal Numbers of Replications.”Biometrics 12:307–310.

Lenth, R. V. (1989).“Quick and Easy Analysis of Unreplicated Factorials.”Technometrics 31:469–473.

Littell, R. C., Milliken, G. A., Stroup, W. W., Wolfinger, R. D., and Schabenberger, O. (2006).SAS for Mixed Models.2nd ed. Cary, NC: SAS Institute Inc.

Mallows, C. L. (1973).“Some Comments on C_p.”Technometrics 15:661–675.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979).Multivariate Analysis.London: Academic Press.

McClave, J. T., and Dietrich, F. H. (1988).Statistics.San Francisco: Dellen.

McCullagh, P., and Nelder, J. A. (1989).Generalized Linear Models.2nd ed. London: Chapman & Hall.

McCulloch, C. E., Searle, S. R., and Neuhaus, J. M. (2008).Generalized, Linear, and Mixed Models.New York: John Wiley & Sons.

Meeker, W. Q., and Escobar, L. A. (1998).Statistical Methods for Reliability Data.New York: John Wiley & Sons.

Miller, A. J. (1990).Subset Selection in Regression.New York: Chapman & Hall.

Montgomery, D. C. (1991).“Using Fractional Factorial Designs for Robust Process Development.”Quality Engineering 3:193–205.

Muller, K. E., and Barton, C. N. (1989).“Approximate Power for Repeated-Measures ANOVA Lacking Sphericity.”Journal of the American Statistical Association 84:549–555.Also see “Correction to ‘Approximate Power for Repeated-Measures ANOVA Lacking Sphericity’,” Journal of the American Statistical Association 86 (1991): 255–256.

Nagelkerke, N. J. D. (1991).“A Note on a General Definition of the Coefficient of Determination.”Biometrika 78:691–692.

Nelder, J. A., and Wedderburn, R. W. M. (1972).“Generalized Linear Models.”Journal of the Royal Statistical Society, Series A 135:370–384.

Nelson, F. D. (1976).“On a General Computer Algorithm for the Analysis of Models with Limited Dependent Variables.”Annals of Economic and Social Measurement 5:493–509.

Nelson, P. R., Wludyka, P. S., and Copeland, K. A. F. (2005).The Analysis of Means: A Graphical Method for Comparing Means, Rates, and Proportions.Philadelphia: SIAM.

Patterson, H. D., and Thompson, R. (1974).“Maximum Likelihood Estimation of Components of Variance.”In Proceedings of the Eighth International Biometric Conference, 197–207.Washington, DC: International Biometric Society.

Portnoy, S., and Koenker, R. (1997).“The Gaussian Hare and the Laplacian Tortoise: Computation of Squared-Error vs. Absolute-Error Estimators.”Statistical Science 12:279–300.

Rawlings, J. O. (1988).Applied Regression Analysis: A Research Tool.Pacific Grove, CA: Wadsworth & Brooks/Cole Advanced Books & Software.

Ries, P. N., and Smith, H. (1963).“The Use of Chi-Square for Preference Testing in Multidimensional Problems.”Chemical Engineering Progress 59:39–43.

Sall, J. P. (1990).“Leverage Plots for General Linear Hypotheses.”American Statistician 44:308–315.

SAS Institute Inc.(2020a).“The GENMOD Procedure.”In SAS/STAT 15.2 User’s Guide.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/genmod.pdf.

SAS Institute Inc.(2020b).“The GLM Procedure.”In SAS/STAT 15.2 User’s Guide.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/glm.pdf.

SAS Institute Inc.(2020c).“Introduction to Statistical Modeling with SAS/STAT Software.”In SAS/STAT 15.2 User’s Guide.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/intromod.pdf.

SAS Institute Inc.(2020d).“The MIXED Procedure.”In SAS/STAT 15.2 User’s Guide.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/mixed.pdf.

Satterthwaite, F. E. (1946).“An Approximate Distribution of Estimates of Variance Components.”Biometrics Bulletin 2:110–114.

Scheffé, H. (1958).“Experiments with Mixtures.”Journal of the Royal Statistical Society, Series B 20:344–360.

Schuirmann, D. J. (1987).“A Comparison of the Two One-Sided Tests Procedure and the Power Approach for Assessing the Equivalence of Average Bioavailability.”Journal of Pharmacokinetics and Biopharmaceutics 15:657–680.

Searle, S. R., Casella, G., and McCulloch, C. E. (1992).Variance Components.New York: John Wiley & Sons.

Seber, G. A. F. (1984).Multivariate Observations.New York: John Wiley & Sons.

Singer, J. D. (1998).“Using SAS PROC MIXED to Fit Multilevel Models, Hierarchical Models, and Individual Growth Models.”Journal of Educational and Behavioral Statistics 23:323–355.

Snedecor, G. W., and Cochran, W. G. (1967).Statistical Methods.6th ed. Ames: Iowa State University Press.

Spiller, S. A., Fitzsimons, G. J., Lynch, J. G., Jr., and McClelland, G. (2013).“Spotlights, Floodlights, and the Magic Number Zero: Simple Effects Tests in Moderated Regression.”Journal of Marketing Research 50:277–288.

Stone, C., and Koo, C. Y. (1985).“Additive Splines in Statistics.”In Proceedings of the Statistical Computing Section, 45–48.Alexandria, VA: American Statistical Association.

Sullivan, L. M., Dukes, K. A., and Losina, E. (1999).“An Introduction to Hierarchical Linear Modelling.”Statistics in Medicine 18:855–888.

Tibshirani, R. (1996).“Regression Shrinkage and Selection via the Lasso.”Journal of the Royal Statistical Society, Series B 58:267–288.

Tukey, J. W. (1953).“The Problem of Multiple Comparisons.”In Multiple Comparisons, 1948–1983, edited by H. I. Braun, vol. 8 of The Collected Works of John W. Tukey (published 1994), 1–300.London: Chapman & Hall.Unpublished manuscript.

Walker, S. H., and Duncan, D. B. (1967).“Estimation of the Probability of an Event as a Function of Several Independent Variables.”Biometrika 54:167–179.

Westfall, P. H., Tobias, R. D., and Wolfinger, R. D. (2011).Multiple Comparisons and Multiple Tests Using SAS.2nd ed. Cary, NC: SAS Institute Inc.

Wilks, S. S. (1938).“The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses.”Annals of Mathematical Statistics.9:60–62.

Wolfinger, R. D., Tobias, R. D., and Sall, J. (1994).“Computing Gaussian Likelihoods and Their Derivatives for General Linear Mixed Models.”SIAM Journal on Scientific Computing 15:1294–1310.

Wright, S. P., and O’Brien, R. G. (1988).“Power Analysis in an Enhanced GLM Procedure: What it Might Look Like.”In Proceedings of the Thirteenth Annual SAS Users Group International Conference, 1097–1102.Cary, NC: SAS Institute Inc. http://www.sascommunity.org/sugi/SUGI88/Sugi-13-220%20Wright%20OBrien.pdf.

Zou, H. (2006).“The Adaptive Lasso and Its Oracle Properties.”Journal of the American Statistical Association 101:1418–1429.

Zou, H., and Hastie, T. (2005).“Regularization and Variable Selection via the Elastic Net.”Journal of the Royal Statistical Society, Series B 67:301–320.

附录 C

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

预测和专业建模

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020–2021。JMP® 16 预测和专业建模. Cary, NC: SAS Institute Inc.

JMP® 16 预测和专业建模

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

预测和专业建模

1 了解 JMP 19

文档和其他资源

JMP 文档中的格式规范 21

JMP 帮助 22

JMP 文档库 22

学习 JMP 的其他资源 28

JMP 教程 28

样本数据表 28

学习统计和 JSL 术语 29

学习 JMP 的提示与技巧 29

JMP 工具提示 29

JMP 用户社区 29

免费在线统计思维课程 29

JMP 新用户欢迎套件 30

统计学知识门户 30

JMP 培训 30

用户编写的 JMP 手册 30

“JMP 起始页”窗口 30

JMP 技术支持 30

2 预测和专业建模简介 31

建模方法概述

3 神经网络 33

使用节点和层拟合非线性模型

“神经”平台概述 35

启动“神经”平台 35

“神经”启动窗口 36

“模型启动”控制面板 37

用于神经的验证方法 38
隐藏层结构 38
提升 39
拟合选项 40
模型报表 41
训练和验证拟合测度 41
混淆统计量 42
“神经”平台选项 42
“神经模型”选项 42
神经网络示例 43

4 分割模型 47
使用决策树进行数据探索和建模

“分割”平台概述 49
“分割”平台的示例 49
启动“分割”平台 52
“分割”报表 53
控制按钮 53
分类响应的报表 54
连续响应的报表 56
“分割”平台选项 58
显示拟合详细信息 61
指定收益矩阵 62
“决策矩阵”报表 63
信息性缺失 64
“预测值 - 实际值”图 65
ROC 曲线 65
提升曲线 66
节点选项 67
分割中的验证 68
分割的更多示例 69
连续响应的示例 69
信息性缺失的示例 72
“收益矩阵”报表和“决策矩阵”报表的示例 75

“分割”平台的统计详细信息 78
响应和因子 78
拆分准则 78
“决策树”和“Bootstrap 森林法”中的预测概率 79

5 Bootstrap 森林法 81
通过对许多树求平均值来拟合模型

“Bootstrap 森林法”平台概述 83
包含分类响应的 Bootstrap 森林法的示例 83
Bootstrap 森林法模型 83
缺失值 85
包含连续响应的 Bootstrap 森林法的示例 86
启动“Bootstrap 森林法”平台 88
启动窗口 88
规格窗口 89
“Bootstrap 森林法”报表 91
模型验证集汇总 93
规格 93
总体统计量 93
累积验证 94
按树汇总 95
“Bootstrap 森林法”平台选项 95

6 提升树 97
拟合树的许多层，每层都基于前一层

“提升树”平台概述 99
包含分类响应的提升树的示例 99
包含连续响应的提升树的示例 101
启动“提升树”平台 102
规格窗口 103
“提升树”报表 106
模型验证集汇总 108
规格 108
总体统计量 108
累积验证 109

“提升树”平台选项 109
“提升树”平台的统计详细信息 111
过拟合惩罚 111

7 K 最近邻 113
使用邻近观测预测响应值

“K 最近邻”平台概述 115
包含分类响应的 K 最近邻的示例 115
包含连续响应的 K 最近邻的示例 117
启动“K 最近邻”平台 119
“K 最近邻”报表 120
连续响应 120
分类响应 121
“K 最近邻”平台选项 122
模型拟合选项 122

8 朴素 Bayes 125
使用概率假设将观测分类

“朴素 Bayes”平台概述 127
朴素 Bayes 的示例 127
启动“朴素 Bayes”平台 129
“朴素 Bayes”报表 131
“拟合详细信息”报表 131
响应列报表 131
“混淆矩阵”报表 131
ROC 曲线 132
“朴素 Bayes”平台选项 132
朴素 Bayes 的更多示例 133
“朴素 Bayes”平台的统计详细信息 134
算法 134
已保存的概率公式 135

9 支持向量机 137
使用支持向量预测响应类别

支持向量机概述 139
支持向量机示例 139

启动“支持向量机”平台 141
“支持向量机”启动窗口 141
“模型启动”控制面板 143
“支持向量机”报表 144
“模型比较”报表 145
“支持向量机模型”报表 146
“支持向量机”平台选项 147
“支持向量机模型”报表选项 147
SVM 平台的更多示例 149
连续响应的支持向量回归示例 149

10 模型筛选 151
拟合和比较多重预测模型
连续响应的模型筛选示例 153
启动“模型筛选”平台 154
“模型筛选”报表 157
“跨越折汇总”报表 157
训练、验证和测试拟合测度 158
“模型筛选”平台选项 159
“决策阈值”报表 160

11 模型比较 163
比较拟合模型的预测能力
模型比较的示例 165
启动“模型比较”平台 168
“模型比较”报表 169
“模型比较”平台选项 170
连续和分类响应 170
连续响应 170
分类响应 171
模型比较的更多示例 171

12 生成验证列 175
将数据分为训练集、验证集和测试集
“生成验证列”平台概述 177

“生成验证列”示例 177
启动“生成验证列”平台 179
“生成验证列”报表 181
指定比率或相对比率 181
设置割点 181
选项 182
执行 182
“生成验证列”平台选项 182
“生成验证列”平台的更多示例 182

13 公式存储库 185
管理模型和生成得分代码
“公式存储库”平台概述 187
“公式存储库”示例 187
启动“公式存储库”平台 188
将预测公式发布到“公式存储库”的平台 188
“公式存储库”平台选项 189
公式存储库模型选项 190
从“公式存储库”平台生成得分代码 190

14 拟合曲线 193
拟合数据的内置非线性模型
“拟合曲线”平台概述 195
使用“拟合曲线”平台的示例 195
启动“拟合曲线”平台 198
“拟合曲线”报表 199
“拟合曲线”选项 199
“模型比较”报表 200
“模型拟合”报表 201
“模型拟合”选项 201
检验平行性 203
比较参数估计值 205
等价性检验 206
“拟合曲线”平台的统计详细信息 207

模型公式 207

15 非线性回归 215
将数据拟合为定制的非线性模型

拟合定制模型的示例 217
启动“非线性”平台 219
“非线性拟合”报表 220
“非线性”平台选项 224
使用模型库创建公式 226
更多示例 230
最大似然的示例：Logistic 回归 230
具有二项误差的 Probit 模型示例：数值导数 232
Poisson 损失函数的示例 233
设置参数限值的示例 235
“非线性”平台的统计详细信息 238
边侧似然置信限 239
定制损失函数的工作机制 239
关于导数的注释 240
关于有效的非线性建模的注释 242

16 函数数据分析器 243
执行函数数据分析

“函数数据分析器”平台概述 245
函数数据分析器的示例 245
启动“函数数据分析器”平台 249
“函数数据分析器”报表 251
模型报表 252
“函数数据分析器”平台选项 255
“函数数据分析器组”选项 256
“数据处理”报表选项 257
模型报表选项 259
“函数数据分析器”平台的更多示例 260
多个函数过程的示例 260
函数实验设计示例 263

“函数数据分析器”平台的统计详细信息 266
函数模型拟合 266
函数汇总详细信息 267

17 高斯过程 269
使用平滑模型拟合数据

高斯过程示例 271
启动“高斯过程”平台 273
“高斯过程”报表 274
“预测值 - 实际值”图 274
模型报表 274
边缘模型图 274
“高斯过程”平台选项 275
“高斯过程”平台的更多示例 275
高斯过程模型的示例 276
具有分类预测变量的高斯过程模型的示例 277
“高斯过程”平台的统计详细信息 278
包含连续预测变量的模型 279
包含分类预测变量的模型 280
方差公式参数化 280
模型拟合详细信息 280

18 时间序列分析 283
拟合时间序列模型和转换函数

“时间序列”平台概述 285
“时间序列”平台示例 285
启动“时间序列”平台 287
时间序列分析报表 289
时间序列图 289
时间序列基本诊断图 289
“时间序列”平台选项 291
时间序列诊断 291
差分和分解 292
ARIMA 和季节性 ARIMA 模型 294
平滑模型 295

状态空间平滑模型 296
转换函数模型 297
平滑模型规格窗口 299
报表 301
差分报表 301
分解报表 301
“模型比较”报表 303
ARIMA 和季节性 ARIMA 模型报表 305
“状态空间平滑”报表 308
转换函数报表 311
“谱密度”报表 312
“时间序列”平台的更多示例 312
创建时间 ID 列的示例 313
使用保留集的示例 318
“时间序列”平台的统计详细信息 321
谱密度的统计详细信息 321
X-11 分解的统计详细信息 321
平滑模型的统计详细信息 322
ARIMA 模型的统计详细信息 326
转换函数的统计详细信息 327

19 时间序列预测 329
预测多个时间序列

时间序列预测示例 331
启动“时间序列预测”平台 333
数据格式 333
“时间序列预测”报表 334
时间模式分析 335
时间的分组汇总 335
Y 的分组汇总 335
建模规格 336
“时间序列预测”平台选项 338
模型报表 339

20 配对分析 341

比较关于同一对象的测量值

“配对”平台概述 343
关于比较配对的示例 343
启动“配对”平台 344
多个 Y 列 345
“配对”报表 345
差值图和报表 346
组间 346
“配对”平台选项 347
比较组间配对的示例 348
“配对”平台的统计详细信息 349
配对图 349
响应的相关性 350

21 建模工具 353

探索离群值、缺失值和模式

“探索离群值”实用工具 355
“探索离群值”实用工具的示例 355
启动“探索离群值”实用工具 358
分位数范围离群值 359
稳健拟合离群值 362
稳健 PCA 离群值 364
K 最近邻离群值 366
“探索离群值”实用工具选项 368
“探索离群值”实用工具的更多示例 368
“探索缺失值”实用工具 370
“探索缺失值”实用工具的示例 370
缺失值报表 372
“探索缺失值”实用工具选项 377
“探索模式”实用工具 377
“探索模式”实用工具的示例 377
启动“探索模式”实用工具 379
“探索模式”报表 380

“探索模式”实用工具选项 . . . 381
“探索模式”实用工具的更多示例 . . . 383
“探索模式”实用工具的统计详细信息 . . . 385

22 响应筛选 . . . 387
在大型数据中检验很多响应

“响应筛选”平台概述 . . . 389
响应筛选的示例 . . . 390
启动“响应筛选”平台 . . . 392
“响应筛选”报表 . . . 394
结果表 . . . 394
FDR p 值图 . . . 395
FDR LogWorth 依据 : 效应大小 . . . 396
FDR LogWorth 依据 : R 方 . . . 397
“p 值”数据表 . . . 397
“p 值”数据表脚本 . . . 398
“响应筛选”平台选项 . . . 398
“均值”数据表 . . . 399
“比较均值”数据表 . . . 400
“拟合模型”中的“响应筛选”特质 . . . 402
在“拟合模型”中启动“响应筛选” . . . 402
“拟合响应筛选”报表 . . . 403
“拟合响应筛选”选项 . . . 404
“PValues”数据表 . . . 404
“Y Fits”数据表 . . . 405
响应筛选的更多示例 . . . 406
实际显著性检验和等价性检验的示例 . . . 406
“LogWorth 最大值”选项的示例 . . . 408
稳健拟合的示例 . . . 410
“响应筛选”特质 . . . 412
“响应筛选”平台的统计详细信息 . . . 413
假发现率 . . . 413

23 过程筛选 415
根据稳定性和能力筛选很多过程
“过程筛选”平台概述 417
过程筛选的示例 417
启动“过程筛选”平台 420
启动窗口角色 420
启动窗口选项 421
限值表 422
“过程筛选”报表 424
“过程筛选”平台选项 428
过程能力图 432
过程筛选的更多示例 433
“过程筛选”平台的统计详细信息 435
使用中位数估计 Sigma 的尺度因子 435
24 预测变量筛选 437
针对显著效应筛选很多预测变量
“预测变量筛选”平台概述 439
预测变量筛选的示例 439
启动“预测变量筛选”平台 441
“预测变量筛选”报表 441
“预测变量筛选”平台选项 442
25 关联分析 443
执行市场购物篮分析
“关联分析”平台概述 445
“关联分析”平台示例 445
启动“关联分析”平台 448
数据格式 449
“关联分析”报表 449
频繁项集 449
规则 450
“关联分析”平台选项 450
奇异值分解 451

SVD 报表 451
SVD 报表选项 452
主题分析 453
“主题分析”报表 453
“主题分析”报表选项 454
更多示例：SVD 分析 454
“关联分析”平台的统计详细信息 456
生成常用项集 456
关联分析性能测度 456

26 过程历史分析器 459
标识复杂过程历史中的问题构成
过程历史分析器的示例 461
启动“过程历史分析器”平台 463
数据格式 464
“过程历史分析器”报表 464
“过程历史分析器”平台选项 464

A 统计详细信息 467
预测和专业建模
拟合测度 469
熵 R 方 469
JMP 建模中的验证 469
“验证列”角色 469
K 折和保留验证 471

B 参考书目 475

C 技术许可声明 479

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 21
JMP 帮助 22
JMP 文档库 22
学习 JMP 的其他资源 28
JMP 教程 28
样本数据表 28
学习统计和 JSL 术语 29
学习 JMP 的提示与技巧 29
JMP 工具提示 29
JMP 用户社区 29
免费在线统计思维课程 29
JMP 新用户欢迎套件 30
统计学知识门户 30
JMP 培训 30
用户编写的 JMP 手册 30
“JMP 起始页”窗口 30
JMP 技术支持 30

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 22 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

预测和专业建模简介

建模方法概述

《预测和专业建模》提供有关更专业化的建模方法（如响应筛选、分割和神经网络）的详细信息。

- “神经”平台构建了一层或两层的完全连接的多层感知器。使用神经网络可借助灵活的输入变量函数预测一个或多个响应变量。请参见第 3 章，“神经网络”。
- “分割”平台根据 *X* 值与 *Y* 值之间的关系将数据递归分割，从而创建决策树。请参见第 4 章，“分割模型”。
- JMP PRO “Bootstrap 森林法”平台支持您通过对许多决策树（每棵树都拟合到训练数据的随机子集）求平均值来拟合组合模型。请参见第 5 章，“Bootstrap 森林法”。
- JMP PRO “提升树”平台可生成加法决策树模型，该模型由在层中构造的许多小决策树构成。每层中的树都包含少量拆分，拆分数通常不超过五个。每层都使用递归拟合方法拟合。请参见第 6 章，“提升树”。
- JMP PRO “K 最近邻”平台使用给定观测的本地邻域中的观测的响应预测该观测的响应值。它可与分类响应配合使用以便于分类，也可与连续响应配合使用以便于预测。请参见第 7 章，“K 最近邻”。
- JMP PRO “朴素 Bayes”平台将观测分类到由分类响应变量的水平定义的各组中。用于分类的变量（或因子）在数据挖掘文档中通常称为特征。请参见第 8 章，“朴素 Bayes”。
- JMP PRO “支持向量机”平台将观测分类到由分类响应变量的水平定义的各组中。该模型通过优化用于分离类的超平面来对数据进行分类。请参见第 9 章，“支持向量机”。
- JMP PRO “模型筛选”平台支持您快速运行多个预测模型并比较结果。为每个模型都提供了拟合测度以及叠加诊断图。请参见第 10 章，“模型筛选”。
- JMP PRO “模型比较”平台支持您比较不同模型的预测能力。为每个模型都提供了拟合测度以及叠加诊断图。请参见第 11 章，“模型比较”。
- JMP PRO “生成验证列”平台允许您将数据划分为两个或三个集合，使用五种不同的方法之一创建这些划分。请参见第 12 章，“生成验证列”。
- JMP PRO “公式存储库”平台支持您对要部署的模型进行组织、比较、刻画和评分。对于模型探索任务，您可以使用“公式存储库”在 JMP 数据表之外存储候选模型。请参见第 13 章，“公式存储库”。
- “拟合曲线”平台提供预定义的模型，如多项式、Logistic、Gompertz、指数、峰值和药代动力学模型。可以使用各种分析和图形方法比较不同的分组或对象。请参见第 14 章，“拟合曲线”。

- “非线性”平台支持您拟合定制的非线性模型，模型中包含模型公式和要估计的参数。请参见第 15 章，“非线性回归”。
- “高斯过程”平台对某个连续响应与一个或多个连续预测变量之间的关系进行建模。这些模型在计算机模拟实验（比如，有限元程序输出）之类的领域很常见，而且这些模型经常能够进行完美的数据插值。请参见第 17 章，“高斯过程”。
- JMP PRO “函数数据分析器”平台支持您将函数数据转换为可在其他 JMP 平台中分析的形式。请参见第 16 章，“函数数据分析器”。
- “时间序列”平台支持您探索、分析和预测一元时间序列。请参见第 18 章，“时间序列分析”。
- “时间序列预测”平台允许您对多个时间序列进行建模和预测。从多达 30 个指数平滑模型中自动选择最佳拟合模型。请参见第 19 章，“时间序列预测”。
- “配对”平台比较两个或更多相关变量之间的均值并评估差值。请参见第 20 章，“配对分析”。
- 建模工具可在数据分析的数据清理和预处理阶段中提供协助。每个实用工具都提供探索性工具，支持您更深入彻底地了解数据。请参见第 21 章，“建模工具”。
- “响应筛选”平台自动执行对大量的响应进行检验的过程。为了支持数据探索，您的检验结果和汇总统计量显示在数据表而不是报表中。请参见第 22 章，“响应筛选”。
- “过程筛选”平台支持您探索随时间变化的大量过程。该平台计算控制图、过程稳定性和过程能力指标，并检测较大的过程偏移。请参见第 23 章，“过程筛选”。
- “预测变量筛选”平台支持您为显著预测变量筛选数据集。请参见第 24 章，“预测变量筛选”。
- JMP PRO “关联分析”平台支持您标识彼此有密切关联的项。该分析经常用于分析交易数据（亦称“市场购物篮”），用以识别在交易中经常一起出现的项。请参见第 25 章，“关联分析”。
- “过程历史分析器”平台支持您在复杂的过程历史中标识问题构成。请参见第 26 章，“过程历史分析器”。

第 3 章

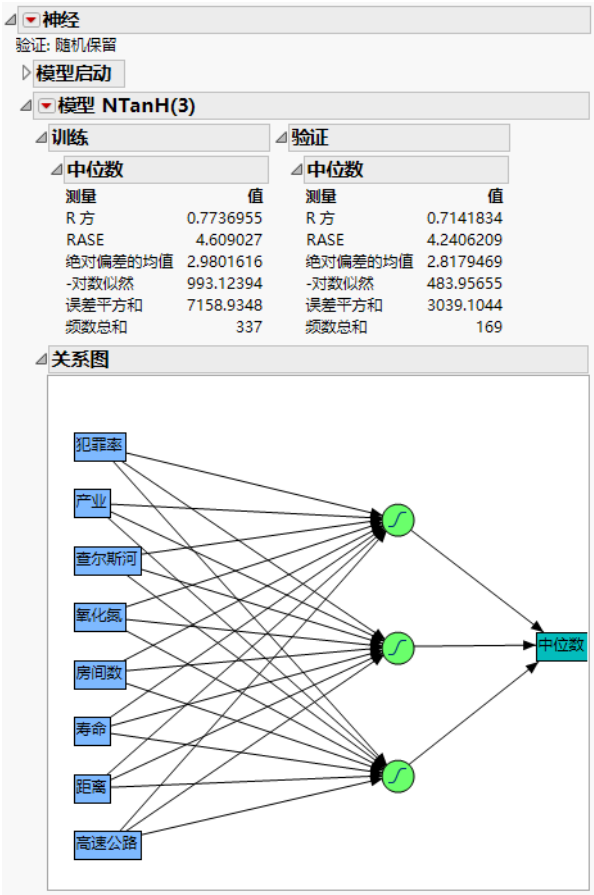
神经网络

使用节点和层拟合非线性模型

JMP PRO 该平台中的很多功能只能在 JMP Pro 中使用，并采用该图标进行标注。

“神经”平台构建了一层或两层的完全连接的多层感知器。使用神经网络可借助灵活的输入变量函数预测一个或多个响应变量。若不必描述响应曲面的函数形式或输入与响应之间的关系，那么神经网络是极佳的预测工具。

图 3.1 神经网络示例



目录

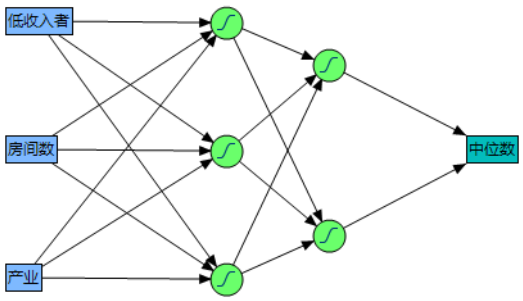
“神经”平台概述 35
启动“神经”平台 35
“神经”启动窗口 36
“模型启动”控制面板 37
用于神经的验证方法 38
隐藏层结构 38
提升 39
拟合选项 40
模型报表 41
训练和验证拟合测度 41
混淆统计量 42
“神经”平台选项 42
“神经模型”选项 42
神经网络示例 43

“神经”平台概述

您可以将神经网络视为一组派生输入（称为隐藏节点）的函数。隐藏节点是原始输入的非线性函数。您最多可以指定两层隐藏节点，每层您可以根据需要，指定包含任意数目的隐藏节点。

图 3.2 显示了包含三个 X 变量和一个 Y 变量的两层神经网络。在本例中，第一层包含两个节点，每个节点是第二层中全部三个节点的函数。第二层包含三个节点，所有节点是三个 X 变量的函数。预测 Y 变量是第一层中两个节点的函数。

图 3.2 神经网络图



在隐藏层的节点处应用的函数称为激活函数。激活函数是 X 变量线性组合的变换。有关激活函数的详细信息，请参见第 38 页的“隐藏层结构”。

应用于响应的函数是一个线性组合（对于连续响应）或 Logistic 变换（对于名义型或有序型响应）。

神经网络模型的主要优点是它能够有效地对不同的响应曲面建模。只要有足够多的隐藏节点和层，任何曲面可以达到任意精度的近似。神经网络模型的主要缺点是结果不容易解释。这是因为它在 X 变量和 Y 变量之间存在中间层，并不像一般的回归模型中那样在两个变量之间是直接关联的。

启动“神经”平台

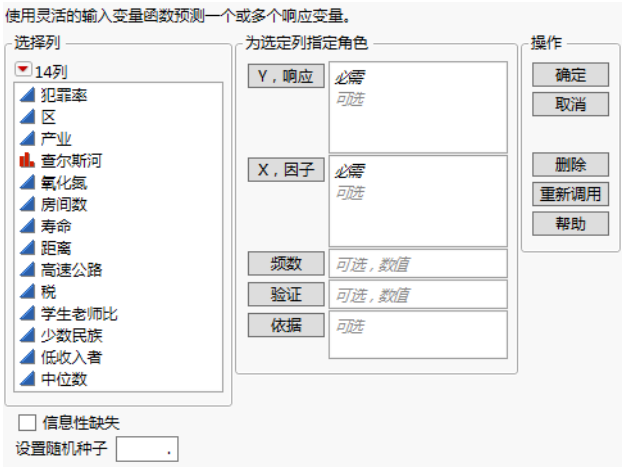
要启动“神经”平台，请选择**分析 > 预测建模 > 神经**。

启动“神经”平台需要执行两个步骤。第一步，在“神经”启动窗口中输入变量。第二步，在“模型启动”控制面板中指定选项。

JMP PRO “神经” 启动窗口

使用 “神经” 启动窗口可指定 X 和 Y 变量、验证列，以及是否启用信息性缺失值编码。

图 3.3 “神经” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，响应 您想要分析的一个或多个响应变量。指定多个响应时，这些响应的模型共享隐藏层中的所有参数 （未连接到响应的那些参数）。

X，因子 预测变量。

频数 一列，该列的数值为分析中的每一行都分配一个频数。

JMP PRO **验证** 用于定义验证集的数值列。请参见第 38 页的 “用于神经的验证方法”。若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见第 175 页的 “生成验证列” 一章。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的报表。

JMP PRO **信息性缺失** 选中该框可启用缺失值的信息编码。该编码允许在包含缺失值的情况下估计预测模型。这在缺失数据可提供信息的情形中很有用。若未选中该选项，则忽略包含缺失值的行。

对于连续变量，缺失值将被该变量的均值取代。同时还会创建名为 < **列名** > **缺失**的缺失值指示变量，并将其包括在模型中。若使用 “模型启动” 控制面板中的 “变换协变量” 拟合选项对变量进行了变换，缺失值将被变换后的变量的均值取代。

对于分类变量，缺失值被视为该变量的单独水平。

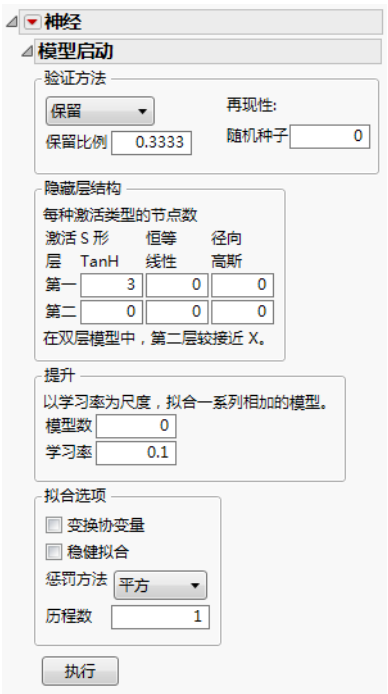
设置随机种子 设置拟合过程中使用的起始值的种子。若您要重现分析，“设置随机种子”很有用。若您设置一个随机种子并保存脚本，该种子将自动保存在脚本中。

“模型启动”控制面板

使用“模型启动”控制面板可指定验证方法、隐藏层的结构、是否使用梯度提升以及其他拟合选项。

图 3.4 “模型启动”控制面板

JMP Pro



JMP

验证方法 选择您想要用于模型验证的方法。请参见第 38 页的“用于神经的验证方法”。

随机种子 若您要在将来启动“神经”平台时重现验证分配，请指定非零数字随机种子。默认情况下，“随机种子”设置为零，即不生成可重现的结果。将分析保存到脚本中时，您输入的随机种子将保存到该脚本中。

隐藏层结构或隐藏节点数 指定每层中的每种类型的隐藏节点数。请参见第 38 页的“隐藏层结构”。

注意：JMP 的标准版本仅使用 TanH 激活函数，只能拟合包含一个隐藏层的神经网络。

JMP PRO **提升** 指定有关梯度提升的选项。请参见第 39 页的“提升”。

JMP PRO **拟合选项** 指定有关变量变换和模型拟合的选项。请参见第 40 页的“拟合选项”。

执行 拟合神经网络模型并显示模型报表。

在点击**执行**拟合模型后，您可以重新打开“模型启动”控制面板并更改设置以便拟合其他模型。

用于神经的验证方法

神经网络会在其拟合例程中使用某种验证方法。可选的验证方法包括保留、K 折或（在 JMP Pro 中）使用验证列。要拟合模型，“神经”平台执行以下操作：

- 对模型参数应用罚值
- 使用验证集来调节参数罚值。

选择以下验证方法之一：

保留排除行 使用行状态对数据取子集。未排除的行用作训练集，排除的行用作验证集。

有关使用行状态和如何排除行的详细信息，请参见 《使用 JMP》。

保留 将原始数据随机划分为训练集和验证集。指定要用作验证集 （保留）的数据占原始数据的比例。随机选择基于对模型因子的分层抽样，尝试创建比基于简单随机抽样的训练集和验证集更平衡的训练集和验证集。

K 重 将数据分为 K 个子集或折。这 K 折中的每一折依次用于验证其余数据的模型拟合，总共拟合 K 个模型。最佳模型是得到具有最佳验证统计量的那一折模型。

该方法适合小数据集，因为它能充分利用有限的数据。

JMP PRO **验证列** 使用用来定义验证集的某个数值列。验证列的值将确定如何拆分数据：

- 若验证列有两个水平，则较小的值定义训练集，较大的值定义验证集。
- 若验证列包含三个水平，将按值由小到大的顺序相应定义训练集、验证集和测试集。
- 若验证列包含三个以上唯一值，则执行 K 折验证。折数由验证列中的值的数量决定。

“神经”平台使用验证列来训练和调优模型，或者训练、调优和评估模型。有关验证的详细信息，请参见 “统计详细信息”附录中第 469 页的 “JMP 建模中的验证”。

隐藏层结构

注意：JMP 的标准版本仅使用 TanH 激活函数，只能拟合包含一个隐藏层的神经网络。

“神经”平台可拟合一层或两层神经网络。增加第一层中的节点数或添加第二层可使神经网络更加灵活。每一层添加的节点数不受限制。第二层节点是 X 变量的函数。第一层节点是第二层节点的函数。Y 变量是第一层节点的函数。

警告：您无法在两层神经网络中使用提升。若在第二层中指定任何非零值并且还指定了提升，则忽略第二层。

在隐藏层的节点处应用的函数称为激活函数。激活函数是 X 变量线性组合的变换。提供以下激活函数：

TanH 双曲正切函数是一个 S 形函数。TanH 将值变换到 -1 和 1 之间，是经过中心化和统一尺度变换的 Logistic 函数。双曲正切函数为：

$$\frac{e^{2x}-1}{e^{2x}+1}$$

其中， x 是 X 变量的线性组合。

线性 恒等函数。不变换 X 变量的线性组合。

线性激活函数经常与一个非线性激活函数一起使用。在这种情况下，线性激活函数置于第二层中，非线性激活函数置于第一层中。若您想要先降低 X 变量的维度，然后建立 Y 变量的非线性模型，那么这样设置激活函数会很有用。

对于连续 Y 变量，若仅使用线性激活函数，Y 变量的模型会简化为 X 变量的线性组合。对于名义型或有序型 Y 变量，模型会简化为 Logistic 回归。

高斯 高斯函数。对径向基函数行为使用该选项，或是在响应曲面在形状上为高斯 （正态）分布时使用该选项。高斯函数为：

$$e^{-x^2}$$

其中， x 是 X 变量的线性组合。

提升

JMP PRO 提升是通过拟合一系列小模型来构建较大的加法神经网络模型的过程。每个小模型在前一个模型经过尺度统一的残差的基础上进行拟合。然后所有模型组合在一起构成一个较大的最终模型。该过程可使用验证来评估要拟合多少个组件模型，但不超过指定的数目。

提升往往比拟合单个大模型要快。但是，基本模型应该为 1 到 2 个节点的单层模型，否则若是指定了大量模型，拟合较快的优势就会丧失。

使用“模型启动”控制面板中的“提升”面板可指定组件模型数和学习率。使用“模型启动”控制面板中的“隐藏层结构”面板可指定基本模型的结构。

学习率必须为 $0 < r \leq 1$。学习率接近 1 会导致最终模型收敛更快，但是也更容易过度拟合数据。当指定小的“模型数”时，可以使用接近 1 的学习率。

接下来我们举例说明提升的工作原理：假定您指定的基本模型包含一层和两个节点，并且模型数等于 8。第一步先要拟合一个单层两节点模型。将该模型的预测值以学习率来统一尺度，然后从实际值中减去预测值以构成尺度统一的残差。第二步则是拟合另一个单层两节点模型，其中的响应值是上一步中模型的尺度统一的残差。该过程会一直持续到拟合了 8 个模型为止，或一直持续到增加另一个模型却不能使验证统计量更好为止。最后，所有组件模型组合在一起构成一个大的最终模型。在本例中，若停止之前拟合了 6 个模型，则最终模型包含一层和 2 x 6 = 12 个节点。

警告：您无法在两层神经网络中使用提升。若在第二层中指定任何非零值并且还指定了提升，则忽略第二层。

拟合选项

JMP PRO 提供以下模型拟合选项：

变换协变量 使用 Johnson Su 或 Johnson Sb 分布将所有连续变量变换为接近正态性。变换连续变量有助于减轻离群值或高偏斜分布的不利影响。请参见第 42 页的 ““神经模型”选项”中的 “保存变换的协变量”选项。

稳健拟合 使用最小绝对偏差而不是最小二乘来训练模型。若您想要将离群值的影响降到最小，该选项会很有用。该选项仅适用于连续响应。

惩罚方法 选择惩罚方法。为了降低神经网络过拟合数据的趋势，拟合过程引入了似然惩罚。请参见第 40 页的 “惩罚方法”。

历程数 指定重新开始拟合过程的次数，每次迭代都使用不同的随机起始点进行参数估计。具有最佳验证统计量的迭代将被选为最终模型。

惩罚方法

罚值为 $\lambda p(\beta_i)$，其中 λ 是惩罚参数，$p(\,)$ 是参数估计值的函数，称为惩罚函数。验证用于找到惩罚参数的最优值。

表 3.1 惩罚方法说明

方法	惩罚函数	说明
平方	$\sum\beta_i^2$	若您认为大多数 X 变量都对模型的预测能力有贡献，则使用该方法。
绝对值	$\sum\lvert\beta_i\rvert$	若 X 变量数量很多并且一小部分的贡献超过其他部分，则使用这两种方法之一。
加权衰减	$\sum\frac{\beta_i^2}{1+\beta_i^2}$	
无惩罚	无	不使用惩罚。若您有大量数据同时希望拟合过程快速完成，则可以使用该选项。不过，该选项会导致模型的预测能力不如使用惩罚的模型。

模型报表

JMP 会为每个神经网络模型创建模型报表，显示训练集和验证集的拟合测度。此外，若响应为名义型或有序型，还会显示混淆统计量。

图 3.5 神经模型报表的示例

训练和验证拟合测度

显示训练集和验证集的拟合测度 （图 3.5）。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。值为 1 表示完美模型，值为 0 表示并不比常数模型好的模型。对于标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。

熵 R 方 （仅当响应为名义型或有序型时才显示。 ）比较拟合模型和恒定概率模型的对数似然的一种拟合测度。熵 R 方的范围介于 0 到 1 之间，其中接近 1 的值指示拟合更佳。请参见“统计详细信息”一章中第 469 页的 “熵 R 方”。

R 方 给出模型的 R 方。

RASE 提供均方根误差。响应为名义型或有序型时，差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

绝对偏差的均值 响应的真实值与预测值的差值绝对值的平均值。响应为名义型或有序型时，差值介于 1 和 p （实际发生的响应水平的拟合概率）之间。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。仅当响应为名义型或有序型时才显示。

- 对数似然 给出负对数似然。请参见 《拟合线性模型》。

误差平方和 给出误差平方和。仅当响应为连续变量时才可用。

频数总和 给出使用的观测数。若您在“神经”启动窗口中指定了“频数”变量，“频数总和”将给出频数列的总和。

若有多个响应，将给出每个响应的拟合统计量，并且给出总体广义 R 方和负对数似然。

混淆统计量

对于名义型或有序型响应，将给出“混淆矩阵”报表和“混淆率”报表（图 3.5）。“混淆矩阵”报表显示实际响应水平和预测响应水平的双向分类。对于分类响应，预测水平是具有最高预测概率的水平。“混淆率”报表等同于“混淆矩阵”报表，只不过其中的数字要除以行合计。

“神经”平台选项

“神经”红色小三角菜单包含以下选项：

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“神经模型”选项

每个模型报表都有一个红色小三角菜单，其中包含用于生成其他输出或保存结果的选项。“模型报表”红色小三角菜单包含以下选项：

关系图 显示表示隐藏层结构的图。

显示估计值 在报表中显示参数估计值。

刻画器 启动 “预测刻画器”。对于名义型或有序型响应，每个响应水平在 “预测刻画器”中分别由单独的行来表示。有关红色小三角菜单中选项的详细信息，请参见 《刻画器指南》。

分类刻画器 启动 “预测刻画器”。类似于 “刻画器”选项，只不过所有分类概率合并为一个刻画器行。仅适用于名义型或有序型响应。有关红色小三角菜单中选项的详细信息，请参见《刻画器指南》。

等高线刻画器 启动 “等高线刻画器” 。仅当模型包含不止一个连续因子时才可用。有关红色小三角菜单中选项的详细信息，请参见 《刻画器指南》。

曲面刻画器 启动 “曲面刻画器” 。仅当模型包含不止一个连续因子时才可用。有关红色小三角菜单中选项的详细信息，请参见 《刻画器指南》。

ROC 曲线 创建 ROC 曲线。仅适用于名义型或有序型响应。有关 ROC 曲线的详细信息，请参见 “分割模型” 一章中第 65 页的 “ROC 曲线”。

提升曲线 创建提升曲线。仅适用于名义型或有序型响应。有关提升曲线的详细信息，请参见 “分割模型” 一章中第 66 页的 “提升曲线”。

标绘 “预测值 - 实际值” 图 标绘 “预测值 - 实际值” 图。仅适用于连续响应。

标绘 “预测值 - 残差” 图 标绘 “预测值 - 残差” 图。仅适用于连续响应。

保存公式 在数据表中创建新列，用于包含预测响应和隐藏层节点的公式。

保存刻画公式 在数据表中创建包含预测响应公式的新列。隐藏层节点的公式嵌入在该公式中。该选项生成的公式可供交互式 HTML 版本的刻画器使用。

保存快速公式 在数据表中创建包含预测响应公式的新列。隐藏层节点的公式嵌入在该公式中。该选项生成的公式提供比其他选项更快的求值速度，但不能用在交互式版本的刻画器中。

JMP PRO **发布预测公式** 创建预测公式并将它们保存为 “公式存储库” 平台中的公式列脚本。若未打开 “公式存储库” 报表，该选项将创建 “公式存储库” 报表。请参见第 185 页的 “公式存储库” 一章。

生成 SAS DATA 步 创建您可以用来对新数据集评分的 SAS 代码。

保存验证 在数据表中创建新列，用于标识在训练集和验证集中分别使用了哪些行。在 “神经” 启动窗口中指定 “验证” 列时，该选项不可用。请参见第 36 页的 ““神经” 启动窗口”。

JMP PRO **保存变换的协变量** 在数据表中创建显示变换的协变量的新列。这些列包含显示变换的公式。仅当在 “模型启动” 控制面板中选中 “变换协变量” 选项时，该选项才可用。请参见第 40 页的 “拟合选项”。

删除拟合 删除整个模型报表。

神经网络示例

本例使用 Boston Housing.jmp 数据表。假定您想要创建一个模型来预测住房价格中位数，它可以表示为几项人口统计特征的函数。按照以下步骤来构建神经网络模型：

1. 选择**帮助** > **样本数据库**，然后打开 Boston Housing.jmp。
2. 通过选择**分析** > **预测建模** > **神经**来启动 “神经” 平台。
3. 将**中位数**分配给 **Y，响应**角色。
4. 将其他列 （从**犯罪率**到**低收入者**）分配给 **X，因子**角色。
5. 点击**确定**。
6. 为**保留比例**输入 0.2。

7. 为**随机种子**输入 1234。

注意：一般来说，由于验证集是随机选择的，所以结果会有所不同。输入上面的种子可以重现本示例中所示的结果。

8. 为第一层中的 TanH 节点数输入 3。
9. 选中**变换协变量**选项。
10. 点击**执行**。

图 3.6 神经报表

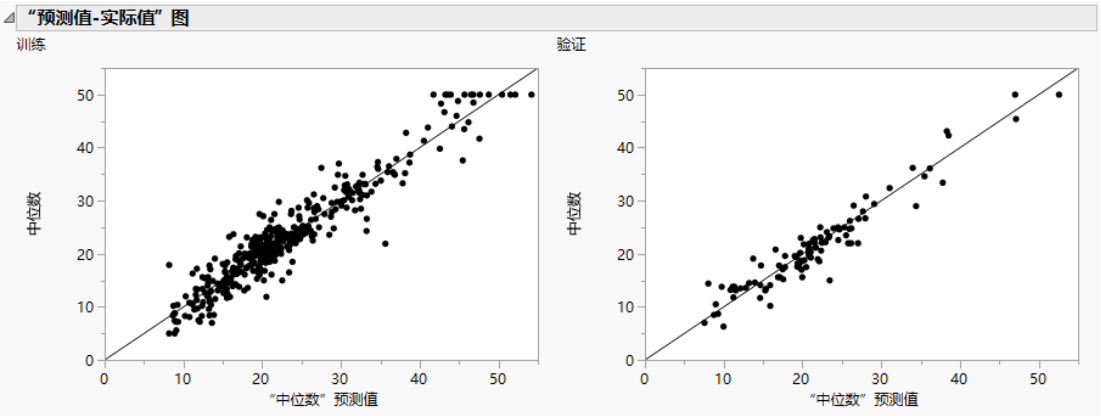
模型 NTanH(3)

训练		验证	
中位数		中位数	
测量	值	测量	值
R 方	0.9074857	R 方	0.9125961
RASE	2.8561483	RASE	2.4304529
绝对偏差的均值	2.1278671	绝对偏差的均值	1.8285015
-对数似然	997.23865	-对数似然	235.31565
误差平方和	3295.6636	误差平方和	602.52435
频数总和	404	频数总和	102

报表中同时为训练集和验证集提供了结果。您可以使用验证集的结果来表示模型对于将来观测的预测能力。

验证集的 R 方统计量为 0.913，表明该模型可以很好地预测未用于训练模型的数据。作为对模型拟合的进一步评估，点击“模型 NTanH(3)”旁边的红色小三角并选择**标绘“预测值 - 实际值”图**。

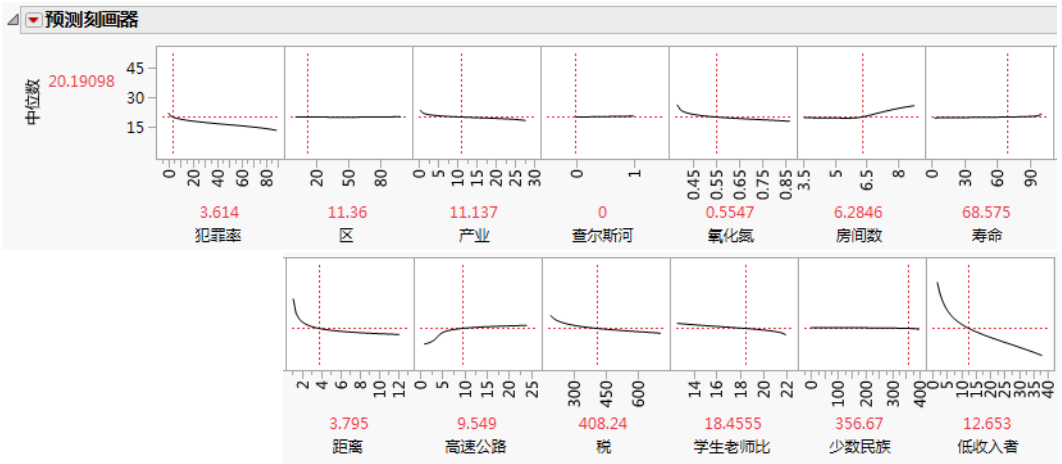
图 3.7 “预测值 - 实际值”图



图上的点沿着线条分布，表明预测值与实际值相差不大。

要对 X 变量如何影响预测值有总体了解，请点击“模型 NTanH(3)”旁边的红色小三角并选择**刻画器**。

图 3.8 刻画器



有些变量的刻画迹线具有正斜率，有些则具有负斜率。例如，变量**房间数**具有正斜率。这表明住房中的房间数越多，中位数预测值就越高。变量**学生老师比**是按城镇划分的师生比。该变量的斜率为负数，指示师生比越高，中位数值越小。

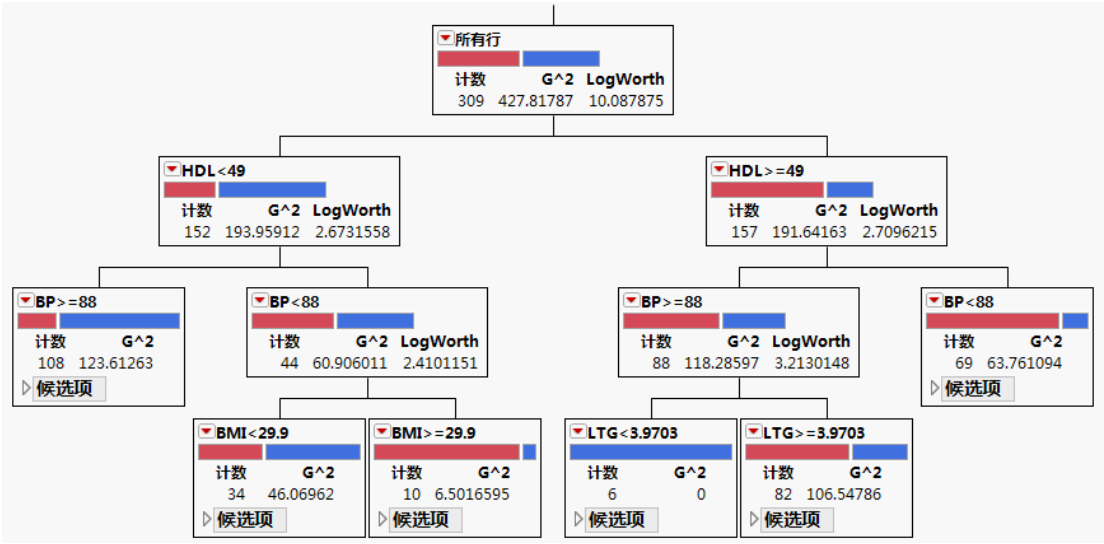
第 4 章

分割模型

使用决策树进行数据探索和建模

“分割”平台根据预测变量和响应值之间的关系将数据递归分割，从而创建决策树。分割算法搜索预测变量的所有可能的拆分以最好地预测响应。以递归方式进行数据的这些拆分（或**分割**）以形成决策规则树。继续拆分直到得到所需的拟合。分割算法从大量可能的拆分方式中选择最佳拆分，使之成为强大的建模和数据发现工具。

图 4.1 决策树的示例



目录

“分割”平台概述 49
“分割”平台的示例 49
启动“分割”平台 52
“分割”报表 53
控制按钮 53
分类响应的报表 54
连续响应的报表 56
“分割”平台选项 58
显示拟合详细信息 61
指定收益矩阵 62
“决策矩阵”报表 63
信息性缺失 64
“预测值 - 实际值”图 65
ROC 曲线 65
提升曲线 66
节点选项 67
分割中的验证 68
分割的更多示例 69
连续响应的示例 69
信息性缺失的示例 72
“收益矩阵”报表和“决策矩阵”报表的示例 75
“分割”平台的统计详细信息 78
响应和因子 78
拆分准则 78
“决策树”和“Bootstrap 森林法”中的预测概率 79

“分割”平台概述

“分割”平台根据预测变量和响应值之间的关系将数据递归分割，从而创建决策树。分割的发展变化很快，基本涵盖了以下几种算法：决策树、CART™、CHAID™、C4.5、C5 等等。因为以下原因，该方法通常被视为一种数据挖掘方法：

- 它适用于在不具备良好的先验模型的情况下探索变量间的关系
- 它可以轻松处理大型问题
- 结果可以解释

分割的经典应用是创建某疾病的诊断启发法。给出许多对象的症状和结果后，可以使用分割来生成问题的层次结构以帮助诊断新患者。

预测变量可以是连续或分类变量（名义型或有序型）。若预测变量是连续变量，则按**切割值**创建拆分。样本根据其值是低于还是高于该切割值进行划分。若预测变量是分类变量，则将样本分为两个水平组。

响应也可以是连续或分类变量（名义型或有序型）。若响应是连续变量，则该平台拟合响应值的均值，并且选择拆分以最小化误差平方和。若响应是分类变量，则拟合的值是响应水平的概率，并且选择拆分以最小化残差对数似然卡方。

有关拆分准则的详细信息，请参见第 78 页的““分割”平台的统计详细信息”。

有关递归分割的详细信息，请参见 Hawkins and Kass (1982) 和 Kass (1980)。

“分割”平台的示例

在该示例中，您使用 “分割”平台构造一个决策树，用于预测糖尿病患者的一年疾病发展情况（“低”或 “高”）。

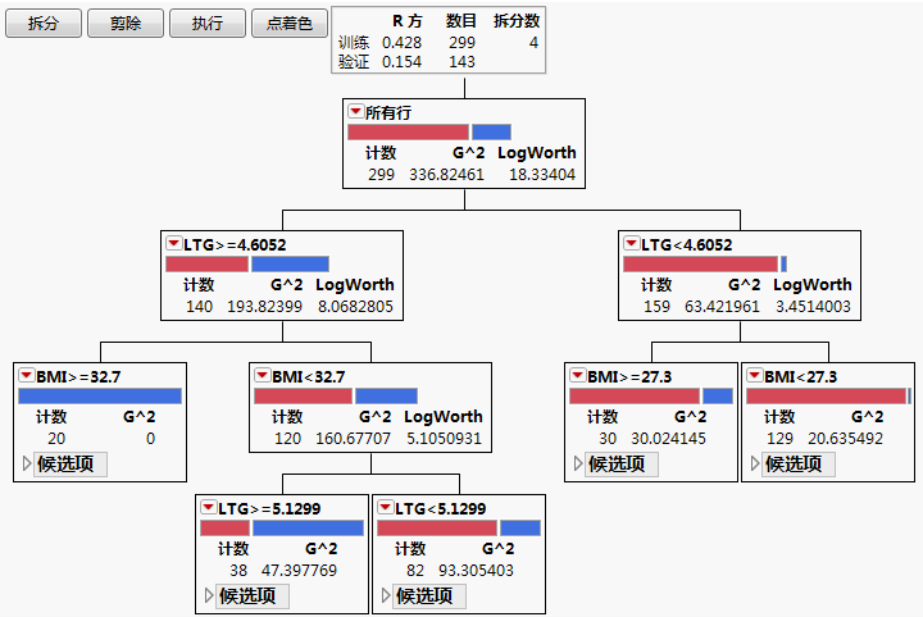
1. 选择**帮助** > **样本数据库**和 Diabetes.jmp。
2. 选择**分析** > **预测建模** > **分割**。
3. 选择 **Y 二值型**并点击 **Y，响应**。
4. 从**年龄**一直选到**葡萄糖**，然后点击 **X，因子**。
5. 为**验证部分**输入 0.33。

   注意：在 JMP Pro 中，可以使用一个验证列进行验证。选择**验证**并点击**验证**。将**验证部分**设置为 0。

6. 点击**确定**。
7. 在平台报表窗口中，点击**执行**以执行自动拆分。

   注意：由于 “验证部分”会随机抽取，您的结果有可能不同于图 4.2 中的结果。

图 4.2 糖尿病的分割报表



自动拆分得到四个拆分。验证集的最终 R 方为 0.154。决策树显示四个拆分以及每个拆分中的观测计数。

8. 点击“‘Y 二值型’分割”旁边的红色小三角菜单，然后选择**列贡献**。

图 4.3 “列贡献”报表

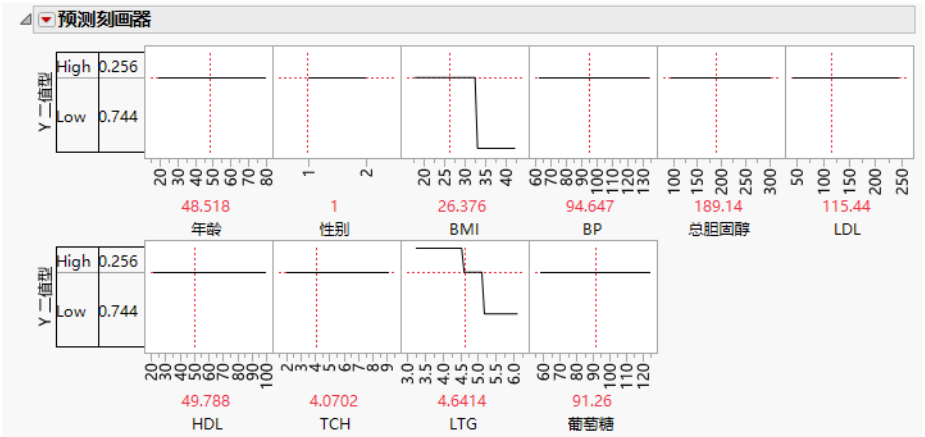
列贡献

项	拆分数	G^2	对应部分
LTG	2	99.5525528	0.6844
BMI	2	45.9092475	0.3156
年龄	0	0	0.0000
性别	0	0	0.0000
BP	0	0	0.0000
总胆固醇	0	0	0.0000
LDL	0	0	0.0000
HDL	0	0	0.0000
TCH	0	0	0.0000
葡萄糖	0	0	0.0000

“列贡献”报表显示 LTG 和 BMI 是决策树模型中仅有的两个预测变量。在两个拆分中使用每个列。您的结果可能不同。使用“验证部分”时，从数据表中随机选择验证集。若您重做分析，将选择新的随机验证集，并且您的结果可能不同于第一次分析的结果。

9. 点击“‘Y 二值型’分割”旁边的红色小三角，然后选择**保存列 > 刻画器**。

图 4.4 分割模型的刻画器



该刻画器支持您更改 BMI 和 LTG 的值以获取预测的 Y 二值型结果。其他任何因子在分割模型中都没有拆分。它们的特征是平直线条。

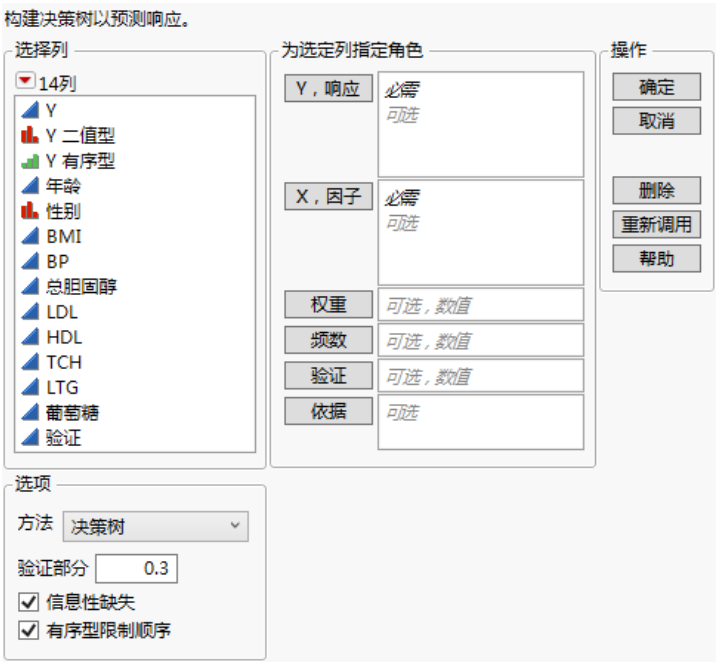
10. 点击“‘Y 二值型’分割”旁边的红色小三角菜单，然后选择**保存列 > 保存预测公式**。

在 Diabetes.jmp 数据表中，添加名为 Prob(Y 二值型 ==Low)、Prob(Y 二值型 ==High) 和**最可能的 “Y 二值型”**的列。要查看如何计算这些响应概率，在“列”面板中，在每列旁边双击“公式”图标 。

启动 “分割” 平台

通过选择**分析** > **预测建模** > **分割**来启动 “分割” 平台。

图 4.5 “分割” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，响应 您想要分析的一个或多个响应变量。

X，因子 预测变量。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，该列的数值为分析中的每一行都分配一个频数。

JMP PRO **验证** 用于定义验证集的数值列。请参见第 68 页的 “分割中的验证”。若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见第 175 页的 “生成验证列” 一章。

依据 一个或多个列，其水平定义不同的分析。对于列的每个水平，都使用您指定的其他变量分析相应行。结果显示在单独报表中。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的报表。

JMP PRO **方法** 支持您选择分割方法 （Bootstrap 森林法、提升树、 K 最近邻或朴素 Bayes）。

有关这些方法的详细信息，请参见第 5 章，“Bootstrap 森林法”、第 6 章，“提升树”、第 7 章，“K 最近邻”和第 8 章，“朴素 Bayes”。

验证部分 要用作验证集的数据部分。请参见第 68 页的 “分割中的验证”。

信息性缺失 若选定，则对于分类预测变量，将其缺失值单独作为一类；对于连续预测变量，则对其缺失值采取信息性处理。请参见第 64 页的 “信息性缺失”。

有序型限制顺序 若选定，对于有序变量，在拆分时则会进行限定以维持有序变量水平的排序。

“分割”报表

初始 “分割”报表显示一个分割图、一些控制按钮、一个汇总面板和一个决策树。该分割图和决策树在没有任何拆分的情况下初始化。报表详细信息对于分类响应和连续响应是不同的。

- “控制按钮”
- “分类响应的报表”
- “连续响应的报表”

控制按钮

使用控制按钮来与决策树交互。

拆分 使用最优拆分创建数据的一个分割。要指定多次拆分，请在点击**拆分**时按住 Shift 键。

剪除 删除最差拆分。

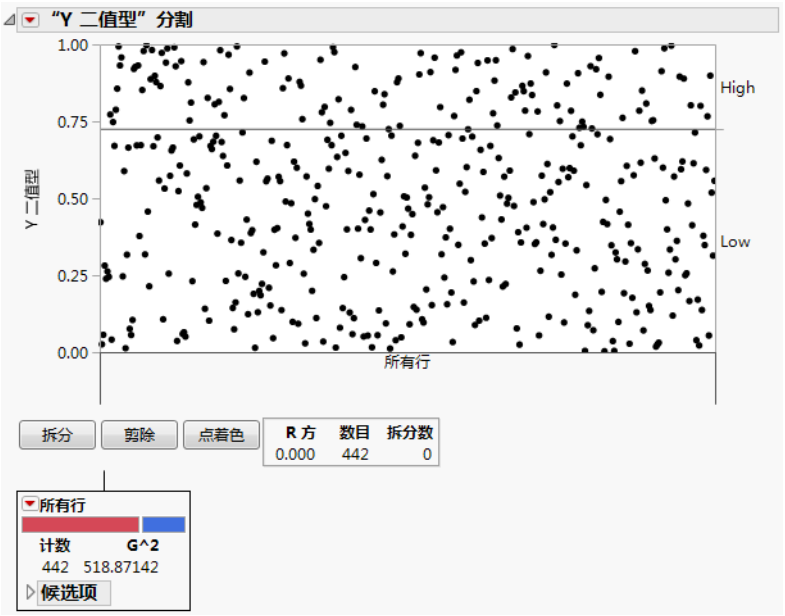
执行 （使用验证时可用。）自动将拆分添加到决策树，直到优化了验证统计量。请参见第 68 页的 “分割中的验证”。没有验证时，您只是决定要在分割模型中使用的拆分数。

点着色 对于分类响应，根据响应水平对观测着色。将这些颜色添加到数据表。

分类响应的报表

样本数据表 Diabetes.jmp 用于创建分类响应 Y 二值型的报表。

图 4.6 分类响应的分割报表



分割图

分割图中的每个点表示数据表中的一个观测。若使用了验证，该图仅用于训练数据。初始分割图不显示拆分。

请注意以下情况：

- 左侧的垂直轴是每个响应结果的比例。
- 右侧的垂直轴显示标绘响应水平的顺序。
- 水平线按响应变量划分每个拆分。初始水平线显示数据集中第一个标绘的响应的总体比例。
- 拆分显示在 X 轴下（它具有文本说明和在图中拆分观测的垂直线）。该垂直线延伸到图并指示每个节点的边界。最近的拆分直接显示在水平轴下，且在现有拆分之上。使用决策树的每个拆分或剪除更新该图。

汇总报表

图 4.7 分类响应的汇总报表

	R 方	数目	拆分数
训练	0.428	299	4
验证	0.154	143	

“汇总报表”提供训练数据、验证数据和测试数据（若使用）的拟合统计量。“汇总面板”中的拟合统计量随着您添加拆分或剪除决策树而更新。

R 方 R^2 的当前值。

数目 观测数。

拆分数 决策树中的当前拆分数。

节点报表

树中的每个节点都有一个报表和一个包含附加选项的红色小三角菜单。末端节点也有“候选项”报表。

图 4.8 分类响应的末端节点报表

所有行

计数	G^2
442	518.87142

候选项

项	候选项 G^2		LogWorth	截断点
年龄	10.5000264		1.71376465	51
性别	1.8302510		0.75424581	2
BMI	92.8760803		31.25572705	27.3
BP	64.8300680		18.98689929	100
总胆固醇	20.3048623		4.01316712	194
LDL	12.5858490		0.82750128	122.2
HDL	44.2535587		11.48909721	46
TCH	64.3516993		17.86426783	4
LTG	102.8078418	*	35.97159929	4.8203
葡萄糖	43.2683018		11.05809993	99

计数 由节点表征的训练观测数。

G^2 用于分类响应的拟合统计量（而非用于连续响应的平方和）。较小的值指示更好的拟合。请参见第 78 页的 “‘分割’平台的统计详细信息”。

候选项 对于每列，“候选项”报表都提供有关该列的最优拆分的详细信息。用星号标记所有项上的最优拆分。

项 显示候选项列。

候选项 G^2 最佳拆分的似然比卡方。在具有最大 G^2 的预测变量上拆分将最大程度简化模型 G^2。

LogWorth LogWorth 统计量，定义为 $-\log_{10}(p \text{ 值})$。最优拆分是使 LogWorth 得到最大值的拆分。请参见第 78 页的““分割”平台的统计详细信息”。

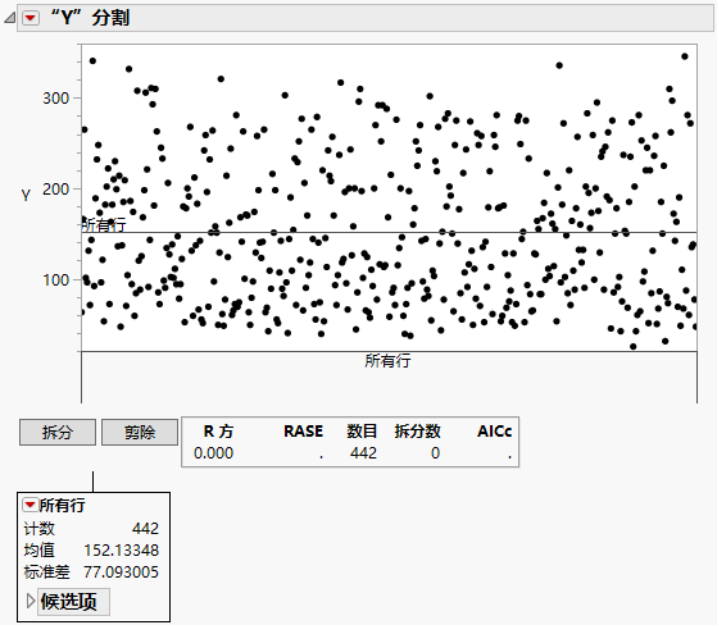
截断点 确定拆分的预测变量的值。对于分类项，列出最左侧拆分中的水平。

最优拆分通过一个星号来标注。不过在某些情况下，“候选项 G^2”最大值与“Logworth”最大值对应的不是同一个变量。在这种情况下，使用 > 和 < 为每个变量指出最佳方向。星号对应于两者一致的状况。请参见第 78 页的““分割”平台的统计详细信息”。

连续响应的报表

样本数据表 Diabetes.jmp 用于为连续响应 Y 创建一个报表。

图 4.9 连续响应的分割报表



分割图

初始化分割图，不进行任何分割。每个点表示数据表中的一个观测。若使用了验证，该图仅用于训练数据。

请注意以下情况：

- 垂直轴表示观测的响应值。
- 水平线显示决策树的每个节点的响应均值。初始水平线位于响应的总体均值。
- 垂直轴划分表示决策树中的拆分。最近拆分的文本说明显示在水平轴下。将观测重新组织为创建或删除拆分时的相应节点。

提示：要查看狭窄分割的工具提示，请悬停在分割图水平轴上的标签上方。

汇总报表

图 4.10 连续响应的汇总报表

	R 方	RASE	数目	拆分数	AICc
训练	0.490	54.690663	299	4	3253.83
验证	0.366	61.16064	143		

“汇总报表”提供训练数据、验证数据和测试数据（若使用）的拟合统计量。“汇总面板”中的拟合统计量随着您添加拆分或剪除决策树而更新。

R 方 R^2 的当前值。

RASE 均方根误差。请参见 《拟合线性模型》。

数目 观测数。

拆分数 决策树中的当前拆分数。

AICc 校正的 Akaike 信息准则。请参见 《拟合线性模型》。

节点报表

树中的每个节点都有一个报表和一个包含附加选项的红色小三角菜单。末端节点也有“候选项”报表。

图 4.11 连续响应的末端节点报表

拆分 剪除

R 方	RASE	数目	拆分数	AICc
0.000	.	442	0	.

所有行

计数	442
均值	152.13348
标准差	77.093005

候选项

项	候选项平方和		LogWorth	截断点
年龄	101593.5850		3.61765555	51
性别	4860.2308		0.40460420	1
BMI	729618.2987		46.76510648	27.3
BP	446708.6991		23.89250262	102
总胆固醇	157877.7151		6.39470855	194
LDL	120014.5872		4.49760240	126.6
HDL	390514.6338		20.07867675	46
TCH	470204.7358		25.55525064	3.73
LTG	764133.3264	*	50.04935713	4.6052
葡萄糖	341244.3856		16.90517792	100

计数 分支中的观测（行）数。

均值 该分支中所有观测的平均响应。

标准差 该分支中所有观测的响应的标准差。

候选项 对于每列，“候选项”报表都提供有关该列的最优拆分的详细信息。用星号标记所有列上的最优拆分。

项 显示候选项列。

候选项平方和 最佳拆分的平方和。

LogWorth LogWorth 统计量，定义为 $-\log_{10}(p \text{ 值})$。最优拆分是使 LogWorth 得到最大值的拆分。请参见第 78 页的“‘分割’平台的统计详细信息”。

截断点 确定拆分的预测变量的值。对于分类项，列出最左侧拆分中的水平。

最佳拆分通过一个星号来标注。不过在有些情况下，“候选项平方和”最大值与“Logworth”最大值对应的不是同一个变量。在这种情况下，使用 > 和 < 为每个变量指出最佳方向。星号对应于两者一致的状况。请参见第 78 页的“‘分割’平台的统计详细信息”。

“分割”平台选项

“分割”红色小三角菜单选项使您可以根据需要定制报表。可用选项由您用于分析的数据类型决定。

显示选项 包含显示或隐藏报表元素的选项。

显示点 显示点。对于分类响应，该选项显示点或颜色面板。

显示树 显示分割的大树。

显示图形 显示分割图形。

显示拆分条 （仅可用于分类响应。）显示指示每个叶中的拆分比例的着色条。

显示拆分统计量 显示拆分统计量。有关分类拆分统计量 G^2 的详细信息，请参见第 78 页的“‘分割’平台的统计详细信息”。

显示拆分概率 （仅可用于分类响应。）在节点报表中显示“比率”和“概率”统计量。

JMP 会在您选择**显示拆分计数**时自动显示“比率”和“概率”统计量。有关“比率”和“概率”的详细信息，请参见第 78 页的“‘分割’平台的统计详细信息”。

显示拆分计数 （仅可用于分类响应。）在节点报表中显示频数统计。当您选择该选项时，JMP 会自动选择**显示拆分概率**。当您取消选择**显示拆分概率**时，计数也会消失。

显示拆分候选项 显示“候选项”报表。

对拆分候选项进行排序 按统计量或 log(worth)（取合适的）对“候选项”报表进行排序。

最佳拆分 在最优拆分点拆分树。这等价于点击**拆分**按钮。

剪除最差 删除判别能力最低的末端拆分。这等价于点击**剪除**按钮。

最小拆分大小 通过输入一个数或占总样本大小的比例来定义允许的最小拆分大小。要指定一个数字，请输入大于等于 1 的值。要指定占样本大小的比例，请输入小于 1 的值。默认值设置为最大值 5 或行数除以 10,000 后的整数部分。

锁定列 以交互方式锁定列以便不考虑拆分它们。您可以关闭显示，然后重新打开而不影响单个锁定。

标绘 “预测值 - 实际值”图 （仅可用于连续响应。）显示“预测值 - 实际值”图。请参见第 65 页的 ““预测值 - 实际值”图”。

小树形视图 在分割图右侧显示一个较小版的分割树。

三维树 显示三维树结构图。要访问该选项，请按 Shift 键并点击红色小三角菜单。

叶报表 显示报表底层叶的均值和计数或比率。

列贡献 显示一个报表，它指示每个输入列对拟合的贡献。该报表还显示对应列用于拆分的次数，以及归因于该列的总 G^2 或平方和。

拆分历史记录 显示 “拆分数 -R 方 “图。若您使用排除行验证、保留验证或验证列，将为训练和验证 R 方值绘制单独的曲线。训练集的 R 方曲线是蓝色的，验证集的 R 方曲线是红色的。若您选择 “K 重交叉验证”，则所有数据的 R 方曲线均为蓝色，交叉验证 R 方的曲线为绿色。

ROC 曲线 （仅可用于分类响应。）受试者操作特征 (ROC) 曲线显示模型的拟合概率在排序响应水平方面的效率。请参见第 65 页的 “ROC 曲线”。

提升曲线 （仅可用于分类响应。）提升曲线显示分割模型的预测能力。请参见第 66 页的 “提升曲线”。

显示拟合详细信息 （仅对分类响应显示。）“拟合详细信息”报表显示几个拟合测度并提供“混淆矩阵”报表。请参见第 61 页的 “显示拟合详细信息”。

保存列 包含用于保存模型和树结果以及创建 SAS 代码的选项。

保存残差 将模型的残差值保存到数据表。

保存预测值 将模型的预测值保存至数据表。

保存叶数 将树的叶数保存到数据表中的一列。

保存叶标签 将树的叶标签保存到数据表。标签记录了该行沿着树经过每个分支的轨迹。每个分支都用“&”分隔。例如，标签可能是“汽车尺寸(Small,Medium)& 汽车尺寸(Small)”。但是，JMP 在类别标签形式中不包含重复的冗余信息。叶的类别标签可能指向更高层次的树节点中的完整类别列表。若树节点包含冗余标签，则显示一个插入符号（“^”）。因此，“汽车尺寸(Small,Medium)&汽车尺寸(Small)”显示为 ^&汽车尺寸(Small)。

保存预测公式 将预测公式保存到数据表中的一个或多个列。公式由描述树形结构的嵌套条件子句构成。若响应是连续的，则添加包含“预测”属性的一列。若响应是分类的，则为响应的每个水平都添加包含“响应概率”属性的列。此外，还添加一个**最可能的**列，其中包含具有每个观测的最高出现概率的响应水平。

保存容缺失预测公式 保存即便存在缺失值并且未选定“信息性缺失”的情况下也能进行预测的公式。该预测公式通过以下方式容许缺失值：它为拆分随机分配缺失预测变量的响应值。若响应是连续的，则该列包含“预测”属性。若响应是分类的，则该列包含“响应概率”属性。若已选中“信息性缺失”，您可以在按住 Shift 键的同时点击报表的红色小三角来保存“容缺失预测公式”。

保存叶数公式 在数据表中保存一列，其中包含计算叶数的公式。

保存叶标签公式 在数据表中保存一列，其中包含计算叶标签的公式。

生成 SAS DATA 步 创建用于对新数据集评分的 SAS 代码。

JMP PRO **发布预测公式** 创建预测公式并将其另存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见第 185 页的“公式存储库”一章。

JMP PRO **发布容缺失预测公式** 创建容缺失预测公式并将其另存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见第 185 页的“公式存储库”一章。若已选定“信息性缺失”，您可以在按住 Shift 键的同时点击报表的红色小三角来使用该选项。

指定收益矩阵 （仅可用于分类响应。）支持您指定与正确或不正确的分类决策分别关联的收益或成本。对于名义型响应，您可以使用概率阈值指定收益矩阵项。请参见第 61 页的“显示拟合详细信息”。

刻画器 显示交互式刻画器报表。因子值的更改体现在估计的分类概率中。请参见《刻画器指南》。

点着色 （仅可用于分类响应。）基于响应水平对点着色。这等价于点击“点着色”按钮。请参见第 54 页的“分类响应的报表”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

显示拟合详细信息

图 4.12 分类响应 （Diabetes.jmp 中的 “Y 二值型”）的拟合详细信息

拟合详细信息

测度	训练	验证	定义
熵 R 方	0.4276	0.1543	1-Loglike(model)/Loglike(0)
广义 R 方	0.5656	0.2464	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.3224	0.5312	∑ -Log(ρ[j])/n
RASE	0.3201	0.4080	√ ∑(y[j]-ρ[j])²/n
绝对偏差的均值	0.2083	0.2917	∑ |y[j]-ρ[j]|/n
误分类率	0.1371	0.2238	∑ (ρ[j]≠ρMax)/n
数目	299	143	n

混淆矩阵

训练

实际分类 Y 二值型	预测计数 Low	预测计数 High
Low	212	12
High	29	46

实际分类 Y 二值型	预测率 Low	预测率 High
Low	0.946	0.054
High	0.387	0.613

验证

实际分类 Y 二值型	预测计数 Low	预测计数 High
Low	84	13
High	19	27

实际分类 Y 二值型	预测率 Low	预测率 High
Low	0.866	0.134
High	0.413	0.587

熵 R 方 比较拟合模型和恒定概率模型的对数似然的一种拟合测度。熵 R 方的范围介于 0 到 1 之间，其中接近 1 的值指示拟合更佳。请参见“统计详细信息”一章中第 469 页的“熵 R 方”。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。值越接近 1 指示拟合效果越好。

-Log p 均值 -log(p) 的平均值，其中 p 是与发生的事件有关的拟合概率。值越小指示拟合效果越好。

RASE 均方根误差，其中差值为响应和 p（实际发生事件的拟合概率）之间的差值。值越小指示拟合效果越好。

绝对偏差的均值 响应和 p （实际发生事件的拟合概率）的差值绝对值的平均值。值越小指示拟合效果越好。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。值越小指示拟合效果越好。

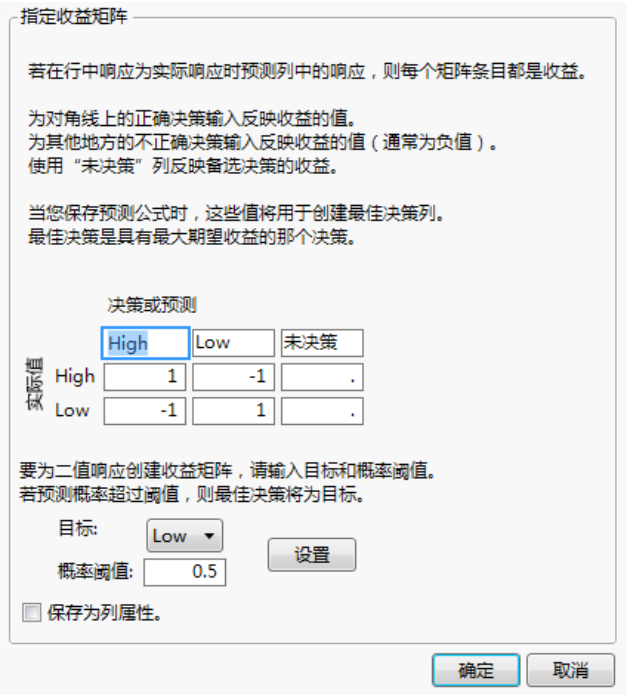
“混淆矩阵”报表显示针对训练集以及验证集和测试集（若定义）的矩阵。“混淆矩阵”报表包含混淆矩阵和混淆率矩阵。混淆矩阵是实际响应和预测响应的双向分类。混淆率矩阵等同于混淆矩阵，只不过其中的数字要除以行合计。

若响应具有“收益矩阵”列属性，或者您使用“指定收益矩阵”选项指定了成本，则会显示“决策矩阵”报表。请参见第 63 页的“ “决策矩阵”报表”。

指定收益矩阵

收益矩阵可以用于分类响应。使用收益矩阵来分配成本到不希望的结果，分配收益到希望的结果。

图 4.13 “指定收益矩阵”窗口



您可以为实际响应类别和预测响应类别的每种组合分配收益和成本值。要指定归入某个备选类别的成本，请在“未决策”列中输入值。要将您的分配作为属性保存到响应列，请选中**保存为列属性**。若不选中该选项，则仅将收益矩阵应用到当前“分割”报表。

指定收益矩阵的概率阈值

响应为二值时，不必将权重输入到收益矩阵，您可以在“收益矩阵”窗口中指定概率阈值。有关如何为收益矩阵计算值的详细信息，请参见《使用 JMP》。

目标 为其建模概率的水平。

概率阈值 目标水平概率的阈值。若某观测落入目标水平的概率超过概率阈值，则将该观测归入该水平。

使用“指定收益矩阵”选项定义成本并选择“显示拟合详细信息”后，“决策矩阵”报表才会显示。请参见第 63 页的“‘决策矩阵’报表”。

指定收益矩阵并保存模型预测公式时，保存到数据表的公式列包含以下信息：

- **<水平> 的收益**：对于每个响应水平，会有一列给出用于将每个观测分类到该水平中的期望收益。
- **<列名> 的最佳收益预测**：对于每个观测，给出具有最高期望收益的响应的水平。
- **<列名> 的期望收益**：对于每个观测，给出“最佳收益预测”列定义的分类的期望收益。
- **<列名> 的实际收益**：对于每个观测，给出用于将该观测分类到“最佳收益预测”列所指定的水平中的实际收益。

“决策矩阵”报表

图 4.14 具有 “决策矩阵”报表的 “拟合详细信息”报表

拟合详细信息

测度	训练	验证	定义
熵 R 方	0.3788	0.0367	1-Loglike(model)/Loglike(0)
广义 R 方	0.5382	0.0647	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.4134	0.6402	∑ -Log(ρ[j])/n
RASE	0.3649	0.4694	√ ∑(y[j]-ρ[j])²/n
绝对偏差的均值	0.2854	0.3861	∑ |y[j]-ρ[j]|/n
误分类率	0.1915	0.3810	∑ (ρ[j]≠ρMax)/n
数目	94	42	n

混淆矩阵

训练

实际分类 严重性	预测计数 Low	High
Low	30	6
High	12	46

实际分类 严重性	预测率 Low	High
Low	0.833	0.167
High	0.207	0.793

验证

实际分类 严重性	预测计数 Low	High
Low	11	5
High	11	15

实际分类 严重性	预测率 Low	High
Low	0.688	0.313
High	0.423	0.577

决策矩阵

训练

实际分类 严重性	决策计数 Low	High
Low	33	3
High	16	42

实际分类 严重性	决策率 Low	High
Low	0.917	0.083
High	0.276	0.724

误分类率 0.2021

验证

实际分类 严重性	决策计数 Low	High
Low	12	4
High	12	14

实际分类 严重性	决策率 Low	High
Low	0.750	0.250
High	0.462	0.538

误分类率 0.3810

指定的收益矩阵

实际值	决策 Low	High
Low	1	-5
High	-3	1

注意：仅当响应具有“收益矩阵”列属性，或者您使用“指定收益矩阵”选项指定了成本，该报表才可用。该报表是“拟合详细信息”报表的一部分。

定义收益矩阵后，分割算法会使用矩阵中的值来计算每个决策的收益。选择“显示拟合详细信息”时，显示“决策矩阵”报表。

在“决策矩阵”报表中，决策计数反映了基于收益矩阵中的权重所产生的最佳收益预测决策。该报表为训练集以及验证集和测试集（若定义）提供“决策计数”矩阵和“决策率”矩阵。为了参考，还显示收益矩阵。

注意：若您使用“指定收益矩阵”选项更改收益矩阵中的权重，“决策矩阵”报表会自动更新以反映您的更改。

决策计数矩阵 显示一个双向分类矩阵，行中显示实际的响应，列中显示分类计数。

指定的收益矩阵 给出用于定义收益矩阵的权重。

决策率矩阵 显示与给定行的观测划分到每个类别的比例相对应的比率值。若所有观测都分类正确，则对角线上的比率均为 1。

提示：您可以使用成本为 1 和 -1 的默认收益矩阵，获取响应的决策率矩阵。从红色小三角菜单选择**指定收益矩阵**，保持默认值不变，然后点击**确定**。

矩阵排列为两行：

- “决策计数”矩阵位于第一行中。
- “指定的收益矩阵”位于第一行中的右侧。
- “决策率”矩阵位于第二行中。

信息性缺失

“信息性缺失”选项支持对预测变量的缺失值进行信息性处理。拟合的模型是确定性模型。“信息性缺失”选项位于启动窗口中，默认情况下选定它。选择 “信息性缺失”选项时，分类预测变量与连续预测变量的处理方式有所不同：

- 对于分类预测变量，若行包含缺失值，则作为该变量的一个单独水平进行分析。
- 按以下方式将包含连续预测变量的缺失值的行分配给拆分：对连续预测变量的值排序。先考虑将缺失行放置在排序值的低端，构造所有拆分。然后考虑将缺失行放置在排序值的高端，再次构造所有拆分。最后通过使用 LogWorth 准则确定出最佳拆分。要对给定的预测变量进一步拆分，算法会按照对该预测变量进行第一次拆分时确定好的高值或者低值来计算缺失值是高值还是低值。

若未选定 “信息性缺失”选项，按以下方式处理缺失值：

- 将具有缺失值的预测变量作为拆分变量时，将该预测变量上具有缺失值的每行随机分配给拆分两侧中的一侧。
- 首次将具有缺失值的预测变量作为拆分变量时，将一个**插补**列添加到显示插补数的“汇总报表”。进行其他插补时，更新“插补”列（图 4.15），其中执行了五次插补。

注意：插补数可能大于包含缺失值的行数。在每次拆分时进行插补。可以随机多次分配具有缺失值的行。每次将行随机分配时，它递增插补计数。

图 4.15 汇总报表中的插补消息

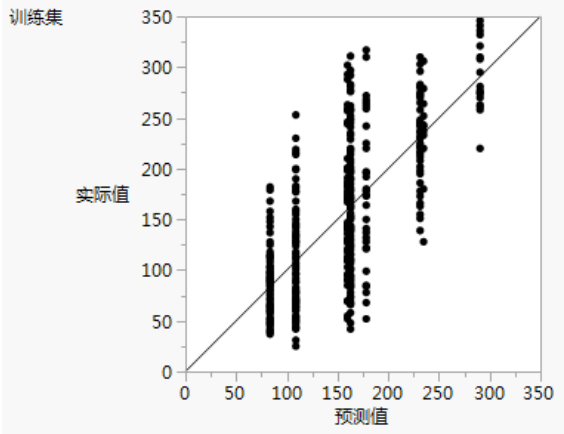
R 方	RASE	数目	拆分数	AICc
0.448	6.8246278	501	1	3352.2

“预测值 - 实际值”图

对于连续响应，“预测值-实际值”图是实际响应对预测响应的典型图。拟合决策树时，叶中的所有观测具有相同的预测值。若有 *n* 个叶子，则“预测值-实际值”图最多显示 *n* 个非重复的预测值。实际值形成围绕 *n* 个垂直线上每个叶均值的散布点。

对角线为 Y = X 线。对于完全拟合，所有点都会落在该对角线上。使用验证时，会为训练集和验证集同时显示图（图 4.16）。

图 4.16 连续响应的 “预测值 - 实际值”图



ROC 曲线

“ROC 曲线”选项仅对分类响应可用。受试者操作特征 (ROC) 曲线显示模型的拟合概率在排序响应水平方面的效率。有关 ROC 曲线的简介，请参见《基本分析》。

分割模型中每个观测的预测响应为介于 0 到 1 的值。要使用预测响应来将观测分类为正数或负数，将使用一个**截断点**。例如，若截断点是 0.5，则具有预测响应 0.5 或之上的观测被分类为正数，在 0.5 之下的观测则为负数。分类可能有权衡，因为截断点变了。

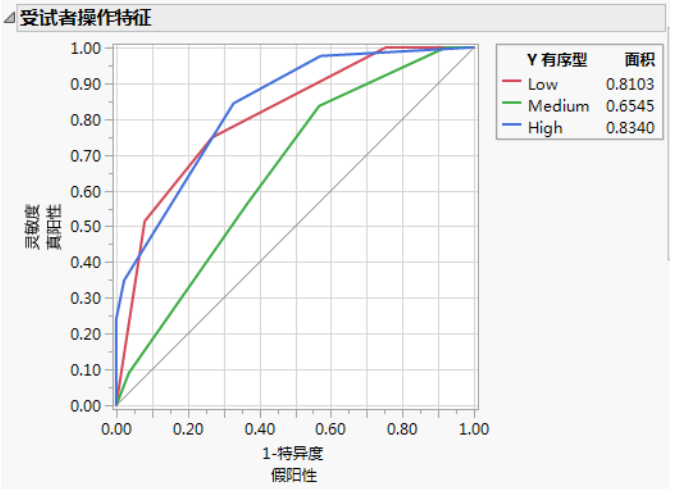
为了生成 ROC 曲线，将每个预测响应水平视为可能的截断点，并为每个可能的截断点计算以下值：

- **灵敏度**是真阳率或具有预测响应的正观测大于截断点的百分比。
- **特异度**是真阴率或具有预测响应的负观测小于截断点的比例。

ROC 曲线标绘灵敏度与 (1 - 特异度)。具有 *n* 个拆分的分割模型具有 *n*+1 个预测值。分割模型的 ROC 曲线具有 *n*+1 个线段。

若您的响应具有两个以上的水平，对于每个响应水平与其他水平，“分割”报表包含单独的 ROC 曲线。每个曲线是阳性响应水平的表示。若只有两个水平，一个曲线是另一曲线的反射。

图 4.17 三水平响应的 ROC 曲线



若模型可以对响应值实现完美的等级排序，则排序后的数据首先包含所有正值，后跟所有其他值。在这种情况下，曲线先一直移到顶部，然后再完全移至右侧。若模型预测效果不好，则曲线沿着对角线从图左下角延伸到右上角。

实际上，ROC 曲线位于对角线之上。曲线下的面积指示模型的拟合优度。值 1 指示完美拟合，接近 0.5 的值指示模型无法区分各组。

响应有两个以上的水平时，ROC 曲线图支持您查看哪个响应类别具有曲线下的最大面积。

提升曲线

“提升曲线”选项提供另一个图来显示分割模型的预测能力。提升曲线标绘针对观测部分的提升。每个非重复预测概率值都对应一个点。响应水平的每个预测概率都定义预测概率大于或等于非重复预测概率值的观测部分。对于特定的响应水平，**提升**值是该部分中观测响应的比例与观测响应的总体比例之比。

注意：对于较小的模型，可能有大部分点都具有相同的预测概率值。若该概率是响应水平的最高预测概率，则提升曲线不会从“对应部分 =0”开始。例如，图 4.18 的“Low”提升曲线中对此进行了演示。

图 4.18 提升曲线

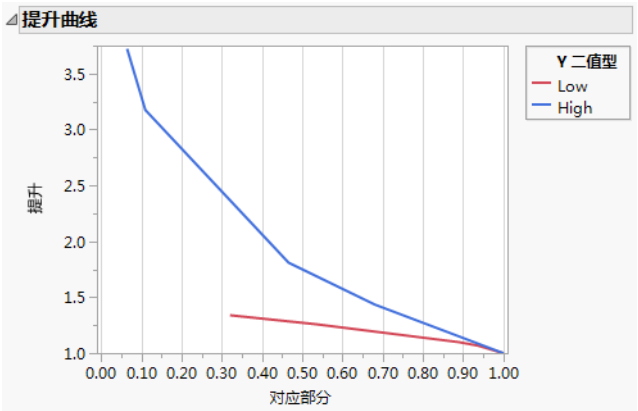


图 4.19 提升曲线的提升表

概率 “High”	数目 > 概率 “High”	对应部分	对应部分中的 “High” 数目	High” 对应部分	提升：“High” 对应部分除以 0.27
0.97	20	0.06	20	1.00	3.72
0.77	44	0.14	39	0.89	3.30
0.33	68	0.22	47	0.69	2.57
0.31	168	0.54	78	0.46	1.73
0.04	309	1.00	83	0.27	1.00

图 4.19 提供一个值表来演示如何计算用于图 4.18 中所示的“High”提升曲线的“提升”和“对应部分”。构建了具有五个拆分的一个分割模型来预测响应 **Y 二值型**。**Y 二值型**具有两个水平：“Low”和 “High”。提升曲线基于 309 个观测。有 83 个观测到的 “High”响应，总体比率为 0.27。

- **概率 “High”**：分割模型用于“High”响应水平的五个非重复预测概率值。
- **数目 > 概率 “High”**：预测概率值等于或大于**概率 “High”**中值的观测数。
- **对应部分**：**数目 > 概率 “High”**除以 309（观测总数）。
- **对应部分中的 “High” 数目**：每个对应部分中具有观测到的“High”响应的观测数。
- **“High” 对应部分**：对应部分中**“High” 的数目**除以**数目 > 概率 “High”**。
- **提升**：**“High” 对应部分**除以 0.27（观测到的“High”响应的总比率）。

提升量测量与每个部分的期望“High”响应数相比，有多少“High”响应落入该部分。对于数据集的前 6%，提升量为 3.72。使用该模型选择具有最高预测值的 6% 的观测所得的“High”响应比随机选择这 6% 所得的“High”响应多 3.72。

节点选项

本节说明每个节点的红色小三角菜单中的选项。

最佳拆分 查找并执行该节点或该节点下方的最佳拆分。

拆分此处 在选定节点处对拆分所依据的最佳列进行拆分。

特定拆分 支持您指定在何处进行拆分。该选项可以用于显示基于分割点的拆分准则是如何构成的，同时也有助于您确定定制分割点。指定拆分列时，您可以为如何执行拆分选择以下选项：

最优值 在选定变量的最优值处拆分。

指定值 支持您指定在哪个水平进行拆分。

输出拆分表 生成显示所有可能的拆分及其关联的拆分值的数据表。

剪除下方 删除选定节点下方的拆分。

剪除最差 查找并删除选定节点下方的最差拆分。

选择行 选择与该叶对应的数据表行。您可以通过按住 Shift 键并在另一个节点中选择该命令来扩展选择。

显示详细信息 生成一个数据表来显示选定变量的拆分准则。该数据表由拆分区间及其关联的准则值组成，它还附加了一个脚本，运行脚本可对准则值进行绘图。

锁定 防止某个节点或其子节点被选定进行拆分。若选中该选项，节点标题中会显示一个锁图标。

分割中的验证

由于分割模型很容易过度拟合，因此对分割模型进行验证是非常重要的。发生这种情况时，模型可以很好地预测用于构建模型的数据，但不能很好地预测将来观测。验证是使用数据集的一部分估计模型参数并使用另一部分评估模型预测能力的过程。有关验证的详细信息，请参见“统计详细信息”附录中第 469 页的 “JMP 建模中的验证”。

在“分割”中，使用验证方法时会显示**执行**按钮。使用**执行**按钮可反复拆分而不必重复点击**拆分**按钮。点击**执行**按钮后，将反复执行拆分，直到验证 R 方值大于接下来的 10 个拆分所获得的 R 方值时结束。该规则可能会生成不太容易解释但预测能力很强的复杂树。

使用**执行**按钮会启用**拆分历史记录**命令。若使用**执行**按钮生成的树包含 40 个以上的节点，则会禁用**显示树状结构**命令。

选择以下验证方法之一：

排除行 使用行状态对数据取子集。未排除的行用作训练集，排除的行用作验证集。

有关使用行状态和如何排除行的详细信息，请参见 《使用 JMP》。

保留 将原始数据随机划分为训练数据集和验证数据集。平台启动窗口中的 “验证部分”用于指定要用作验证数据集（保留）的数据占原始数据的比例。有关“验证部分”的详细信息，请参见第 52 页的 “启动 “分割” 平台”。

JMP PRO **验证列** 使用用来定义验证集的某个数值列。该列应包含至多三个非重复值。

– 若验证列有两个水平，则较小的值定义训练集，较大的值定义验证集。
– 若验证列包含三个水平，将按值由小到大的顺序相应定义训练集、验证集和测试集。
– 若验证列有三个以上的水平，则包含最小三个值的行定义验证集。其他所有行都从分析中排除。

若在“选择列”列表中没有选择任何列的情况下点击“验证”按钮，您可以向数据表添加一个验证列。有关“生成验证列”实用工具的详细信息，请参见第 175 页的“生成验证列”一章。

提示：要使用“K 折交叉验证”或“嵌套 K 折交叉验证”，请通过“模型筛选”平台拟合分割模型。请参见第 151 页的“模型筛选”一章。

“交叉验证”报表

“交叉验证”报表显示以下内容：

k 重 重数。

-2LogLike 或 SSE 响应为分类变量时给出负对数似然值的两倍 （-2 对数似然）。响应是连续变量时，给出误差平方和 (SSE)。第一行给出重上的平均结果。第二行给出单个模型拟合所有观测的结果。有关对数似然的详细信息，请参见 《拟合线性模型》。

R 方 第一行给出重上的平均 R 方值。第二行给出单个模型拟合所有观测的 R 方值。

分割的更多示例

- “连续响应的示例”
- “信息性缺失的示例”
- ““收益矩阵”报表和“决策矩阵”报表的示例”

连续响应的示例

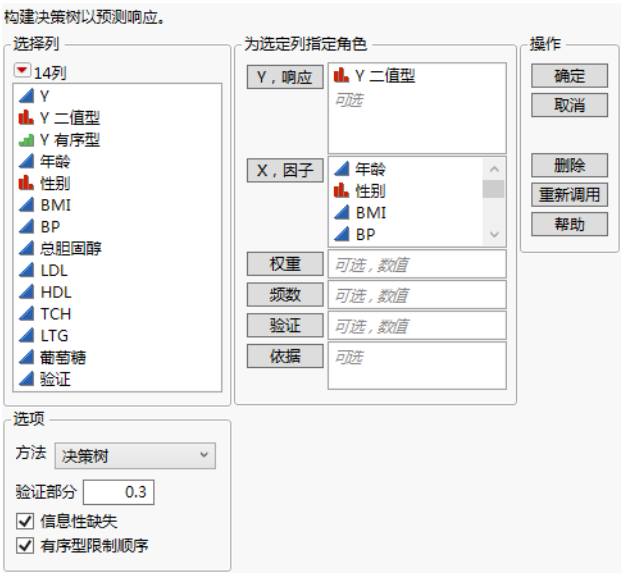
在该示例中，您使用 “分割”平台构造一个决策树，用于预测糖尿病患者的一年疾病发展情况（用定量等级表示）。

1. 选择**帮助** > **样本数据库**和 Diabetes.jmp。
2. 选择**分析** > **预测建模** > **分割**。
3. 选择 Y 并点击 **Y，响应**。
4. 从**年龄**一直选到**葡萄糖**，然后点击 **X，因子**。
5. 基于您的 JMP 安装选择一个验证过程：
   – 对于 JMP Pro，选择**验证**并点击**验证**。

– 对于 JMP，输入 0.3 作为**验证部分**。

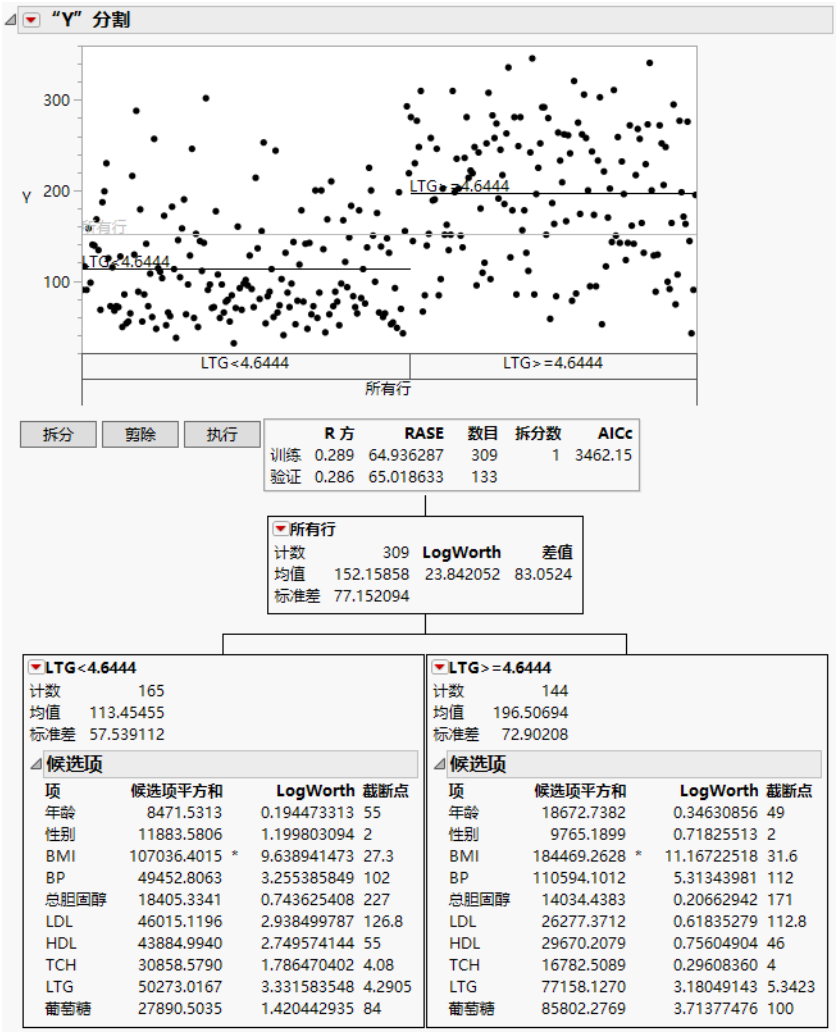
注意：因为验证行是随机选择的，使用验证比例的结果可能不同于此处所示的结果。

图 4.20 验证部分 = 0.3 的已完成启动窗口



6. 点击**确定**。
7. 在平台报表窗口中，点击一次**拆分**以执行一次拆分。

图 4.21 第一次拆分后的报表 （隐藏了决策树）



训练数据集中的原始 309 个值现在拆分为两部分：

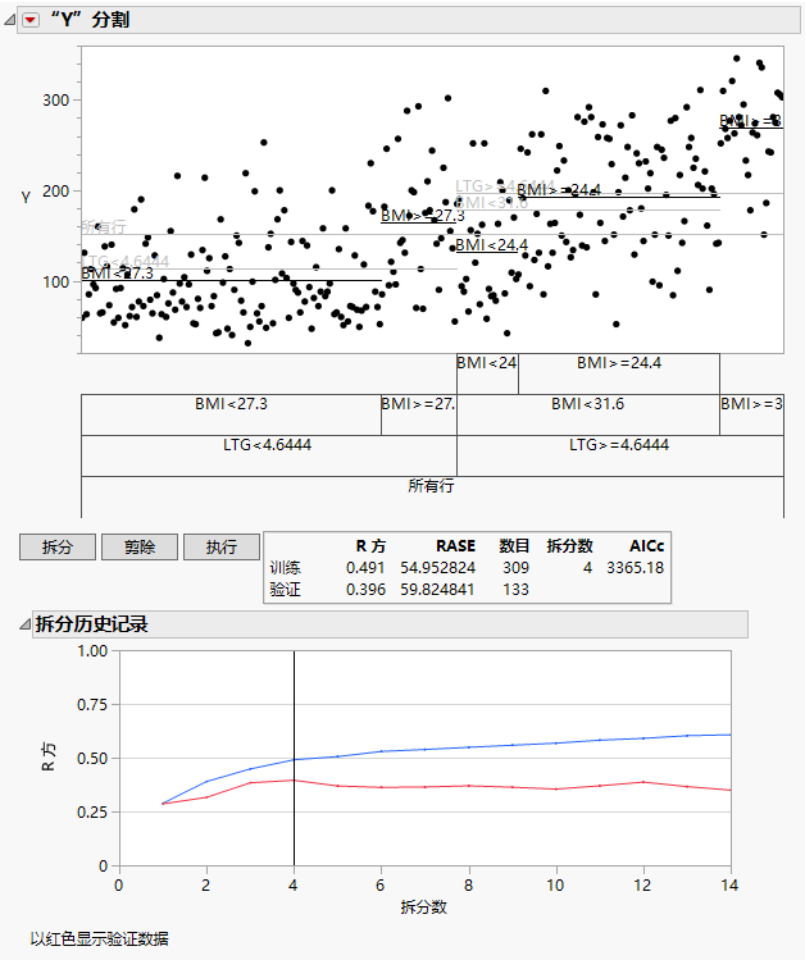
– 左叶（对应于 LTG < 4.6444）有 165 个观测。

– 右叶（对应于 LTG >= 4.6444）有 144 个观测。

对于左叶和右叶，下次拆分将针对 BMI。右叶上 BMI 的候选项 SS 高于左叶上 BMI 的候选项 SS。因此，下一次拆分在右叶上进行。

8. 点击**执行**以使用自动拆分。

图 4.22 带验证、自动拆分后的报表



解找到四次拆分。“拆分历史记录”图显示在四次拆分后，验证数据集没有进一步改进。验证数据上的 R 方值 0.39 不支持该模型作为疾病发展的强预测变量。分割图中各部分的散布进一步指示该模型不能很好地分离 Y 水平。

信息性缺失的示例

在该示例中，您构建一个决策树模型来预测客户是否有信用风险。因为您的数据集包含缺失值，您也探索了“信息性缺失”选项的效用。

启动“分割”平台

1. 选择**帮助 > 样本数据库**，然后打开 Equity.jmp。
2. 选择**分析 > 预测建模 > 分割**。

3. 选择**不良**并点击 **Y，响应**。
4. 从**贷款**一直选到**负债收入比**，然后点击 **X，因子**。
5. 点击**确定**。

使用“信息性缺失”创建决策树和 ROC 曲线

1. 按 Shift 键并点击**拆分**。
2. 为拆分数输入 5，然后点击**确定**。
3. 点击“‘不良’分割”旁边的红色小三角菜单，然后选择 **ROC 曲线**。
4. 点击“‘不良’分割”旁边的红色小三角菜单，然后选择**保存列 > 保存预测公式**。

   列 Prob(不良 == Good Risk) 和 Prob(不良 == Bad Risk) 包含“信息性缺失”实用工具用来将未来贷款申请者的信用风险分类的公式。您想比较该模型与不使用“信息性缺失”的模型的分类效果。

不使用“信息性缺失”创建决策树和 ROC 曲线

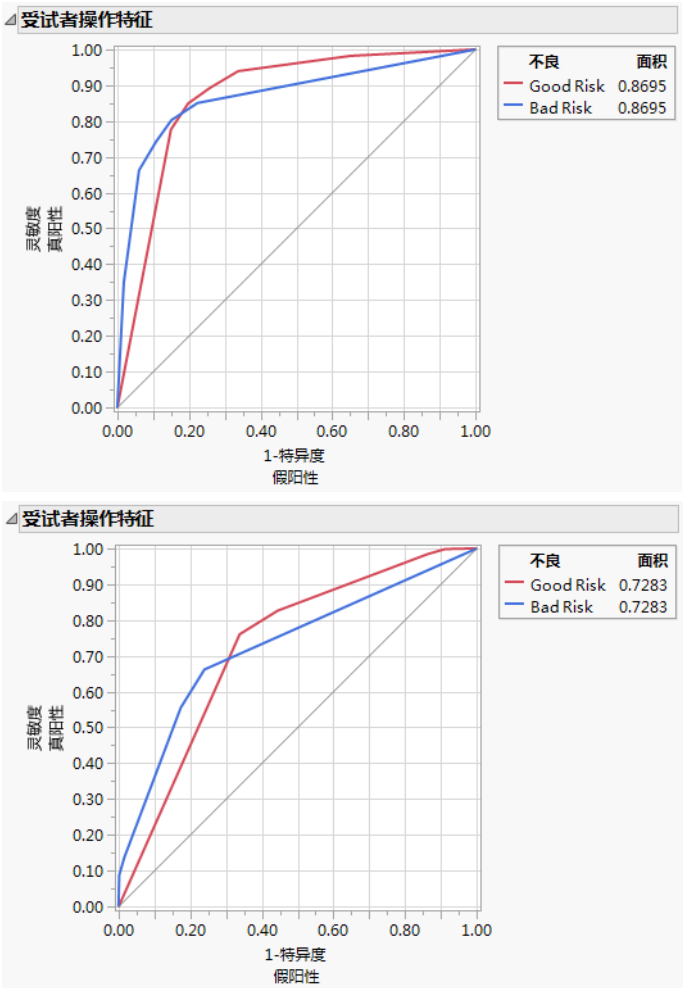
1. 点击“‘不良’分割”旁边的红色小三角菜单，然后选择**重新运行 > 重新启动分析**。
2. 取消选择**信息性缺失**。
3. 点击“确定”并重复 “使用“信息性缺失”创建决策树和 ROC 曲线”中的步骤。

   列 Prob(不良 == Good Risk) 2 和 Prob(不良 == Bad Risk) 2 包含不使用“信息性缺失”实用工具的公式。

比较 ROC 曲线

直观比较两个模型的 ROC 曲线。左侧的模型具有“信息性缺失”，右侧的模型没有“信息性缺失”。

图 4.23 具有 （左）和没有 （右）“信息性缺失”的模型的 ROC 曲线



具有 “信息性缺失”的模型的曲线下面积 (AUC) (0.8695) 大于没有 “信息性缺失”的模型的 AUC (0.7283)。因为响应只有两个水平，每个模型的 ROC 曲线是另一个模型曲线的反射，且 AUC 相等。

注意：您的 AUC 可能不同于此处所示没有 “信息性缺失”的模型的 AUC。不使用 “信息性缺失”时，随机将缺失行分配到拆分两侧。重新运行分析可能导致结果略有不同。

使用“模型比较”平台

接着，使用 “模型比较”平台比较模型，以比较您在第 4 步和第 3 步中创建的两组公式。

1. 选择**分析 > 预测建模 > 模型比较**。

2. 选择 Prob(不良 == Good Risk)、Prob(不良 == Bad Risk)、Prob(不良 == Good Risk) 2 和 Prob(不良 == Bad Risk) 2，然后点击 Y，预测变量。

   第一对公式列包含具有 “信息性缺失” 的模型的公式。第二对公式列包含没有 “信息性缺失” 的模型的公式。

3. 点击**确定**。

图 4.24 模型比较的拟合测度

“不良” 的拟合测度

创建方式	.2 .4 .6 .8	熵 R 方	广义 R 方	-Log p 均值	RMSE	绝对偏差的均值	误分类率	数目
分割		0.3813	0.5015	0.3092	0.3013	0.1817	0.1158	5960
分割		0.1214	0.1808	0.4391	0.3691	0.2765	0.1768	5960

“拟合测度” 报表显示第一个模型（使用 “信息性缺失” 拟合）的表现好于第二个模型（未使用 “信息性缺失” 拟合）。第一个模型具有更高的 R 方值以及更低的 RMSE 值和误分类率。这些结果与 ROC 曲线比较的结果相符。

注意：同样，由于未使用 “信息性缺失” 时的随机差异，您的结果可能不同。

“收益矩阵” 报表和 “决策矩阵” 报表的示例

在该示例中，考察肝癌患者的一个研究。基于各种测量值和标志物，您想要根据患者的疾病严重程度（高或低）对患者进行分类。为患者分类时存在两类错误：将严重程度高的对象划分到低组，或将严重程度低的对象划分到高组。临床上，将严重程度高的患者错误划分到低组是代价很高的错误，因为患者可能无法获得所需的积极治疗。将严重程度低的患者划分到严重程度高的组是一个代价相对较低的错误。该患者可能获得比所需治疗更积极的治疗，但是这问题不大。

在本例中，您以肝癌研究为背景定义了收益矩阵，得到 “决策矩阵” 报表。“决策矩阵” 报表可帮助您评估相对于您在收益矩阵中所指定的成本的分类率。

1. 选择**帮助 > 样本数据库**，然后打开 Liver Cancer.jmp。
2. 选择**分析 > 预测建模 > 分割**。
3. 选择**严重性**并点击 **Y，响应**。
4. 从 BMI 一直选到**黄疸**，然后点击 **X，因子**。
5. 基于您的 JMP 安装选择一个验证过程：
   - 对于 JMP Pro，选择**验证**并点击**验证**。
   - 对于 JMP，输入 0.3 作为**验证比例**。

注意：因为验证行是随机选择的，使用验证比例的结果可能不同于此处所示的结果。

图 4.25 验证部分 = 0.3 的已完成启动窗口

6. 点击**确定**。
7. 按 Shift 键并点击**拆分**。
8. 为拆分数输入 10，然后点击**确定**。
   确认图下方的面板中显示的 “拆分数”为 10。
9. 点击 “‘严重性’分割”旁边的红色小三角菜单，选择**指定收益矩阵**。
10. 将条目更改为以下值：
    - 在“High，High”框中输入 1。
    - 在“High，Low”框中输入 -5。
    - 在“Low，High”框中输入 -3。
    - 在“Low，Low”框中输入 1。

图 4.26 完成的收益矩阵

提示：您可以将该收益矩阵另存为列属性，以供将来分析时使用。选中“收益矩阵”窗口底部的 “保存为列属性”复选框。

请注意以下事项：

- 值 1 反映您作出正确决策时所得到的收益。

– -3 值表示若您将严重性为“Low”的患者分类为严重性“High”，您的损失是正确决策收益的 3 倍。

– -5 值指示若您将严重性为“High”的患者分类为严重性“Low”，您的损失是正确决策收益的 5 倍。

11. 点击**确定**。

12. 点击“‘严重性’分割”旁边的红色小三角菜单，选择**显示拟合详细信息**。

图 4.27 “混淆矩阵”报表和“决策矩阵”报表

拟合详细信息

测度	训练	验证	定义
熵 R 方	0.3788	0.0367	1-Loglike(model)/Loglike(0)
广义 R 方	0.5382	0.0647	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.4134	0.6402	∑ -Log(ρ[j])/n
RASE	0.3649	0.4694	√ ∑(y[j]-ρ[j])²/n
绝对偏差的均值	0.2854	0.3861	∑ |y[j]-ρ[j]|/n
误分类率	0.1915	0.3810	∑ (ρ[j]≠ρMax)/n
数目	94	42	n

混淆矩阵

训练

实际分类 严重性	预测计数 Low	High
Low	30	6
High	12	46

实际分类 严重性	预测率 Low	High
Low	0.833	0.167
High	0.207	0.793

验证

实际分类 严重性	预测计数 Low	High
Low	11	5
High	11	15

实际分类 严重性	预测率 Low	High
Low	0.688	0.313
High	0.423	0.577

决策矩阵

训练

实际分类 严重性	决策计数 Low	High
Low	33	3
High	16	42

实际分类 严重性	决策率 Low	High
Low	0.917	0.083
High	0.276	0.724

误分类率 0.2021

验证

实际分类 严重性	决策计数 Low	High
Low	12	4
High	12	14

实际分类 严重性	决策率 Low	High
Low	0.750	0.250
High	0.462	0.538

误分类率 0.3810

指定的收益矩阵

实际值	决策 Low	High
Low	1	-5
High	-3	1

“混淆矩阵”报表和“决策矩阵”报表位于“拟合详细信息”报表中的“测度”列表的后面。注意到“混淆矩阵”报表和“决策矩阵”报表中的混淆矩阵显示不同的计数。这是因为与不带权重的预测概率相比，收益矩阵中的权重会生成不同的决策。

验证集的混淆矩阵基于预测概率单独显示分类。基于这些概率可以看出，11 个“High”严重性的患者被分类为“Low”严重性，5 个“Low”严重性的患者被分类为“High”严重性。

“决策矩阵”报表包含收益矩阵权重。使用这些权重，只有 6 个“High”严重性的患者被分类为“Low”严重性。不过，这样做的代价是有 6 个“Low”严重性的患者被误分类到“High”严重性组（1 个另外的患者）。

13. 点击“‘严重性’分割”旁边的红色小三角菜单，选择**保存列 > 保存预测公式**。

数据表中添加了八列。

提示：要快速返回数据表，请点击报表窗口 (Windows) 右下角的 “查看相关的数据”图标，或是工具栏菜单 (macOS) 上的 “显示数据表”图标。

- 前三列仅涉及预测概率。混淆矩阵计数基于**最可能的“严重性”**列，该列将患者分类到预测概率最高的那个水平。预测概率在 Prob(**严重性**==High) 和 Prob(**严重性**==Low) 列中给出。
- 后五列涉及收益矩阵权重。名为**“严重性”的最佳收益预测**的列包含基于收益矩阵所作的决策。针对患者所作的决策是具有最大收益的那个水平。收益在**“High”的收益**和**“Low”的收益**列中给出。

“分割”平台的统计详细信息

- “响应和因子”
- “拆分准则”
- ““决策树”和“Bootstrap 森林法”中的预测概率”

响应和因子

响应可以是连续变量，也可以是分类变量 （名义型或有序型）：

- 若响应为分类变量，则平台拟合的是为响应水平估计的概率，目标是最小化残差对数似然卡方 [2*熵] 的值。
- 若响应为连续变量，则平台拟合的是均值，目标是最小化误差平方和的值。

因子可以是连续变量，也可以是分类变量 （名义型或有序型）：

- 若因子为连续因子，则根据因子的拆分“切割”值进行分割。
- 若因子为分类因子，则将 *X* 类别划分为两组水平，并考虑将所有可能分组归为两个水平。

拆分准则

节点拆分基于 LogWorth，该统计量出现在节点的“候选项”报表中。LogWorth 按如下方式计算：

$$-\log_{10}(p \text{ 值})$$

其中，调整的 *p* 值计算方式较为复杂，这种方式将各种可能的拆分方式都考虑在内。相对于未调整的 *p* 值，这种计算极为合理，它支持 *X* 包含许多水平，而 Bonferroni *p* 值仅支持 *X* 包含少量水平。有关该方法的详细信息在 Sall (2002) 中讨论。

对于连续响应，节点报表中会显示“平方和”(SS)，它是拆分引起的误差平方和的变化。

已经选取的候选项平方和如下：

$SS_{测试} = SS_{父级} - (SS_{右} + SS_{左})$，其中，节点中的 SS 为 $s^2(n - 1)$。

针对连续响应的报表中还会显示“差值”统计量，它是父节点的两个子节点的预测值之间的差值。

对于分类响应，报表中会显示 G^2（似然比卡方）。它实际上是 [自然对数] 熵的两倍，或是熵的变化的两倍。对于每个观测，熵为 $\Sigma -\log(p)$，其中，p 是对应的响应水平发生的概率。

已经选取的候选项 G^2 如下：

$G^2_{测试} = G^2_{父级} - (G^2_{左} + G^2_{右})$。

分割实际上有两个比率；一个比率用于训练模型，即为一般定义上的计数占总数的比例，另一个比率稍微偏离 0。因为使用的概率永不为 0，因此我们能够基于验证集或其他排除在训练集之外的数据集计算概率的对数（用于熵 R 方的计算）。

“决策树”和“Bootstrap 森林法”中的预测概率

“决策树”和“Bootstrap 森林法”方法的预测概率通过如下所述的“概率”统计量来计算。

对于 “决策树”中的分类响应，使用 “显示拆分概率”命令可以显示以下统计量：

比率 在相应的节点处每个响应水平的观测比例。

概率 树中该节点的预测概率。用于计算第 i 个响应水平在给定节点处的 “概率”的方法定义如下：

$$Prob_i = \frac{n_i + prior_i}{\sum(n_i + prior_i)}$$

其中，求和计算针对的是所有响应水平； n_i 是第 i 个响应水平在节点处的观测数， $prior_i$ 是第 i 个响应水平的先验概率，它采用以下方式计算：

$prior_i = \lambda p_i + (1-\lambda)P_i$

其中， p_i 是父节点的 $prior_i$， P_i 是父节点的 $Prob_i$， λ 是权重因子，它当前设置为 0.9。

计算 “概率”的方法可确保预测概率始终为非零值。

第 5 章

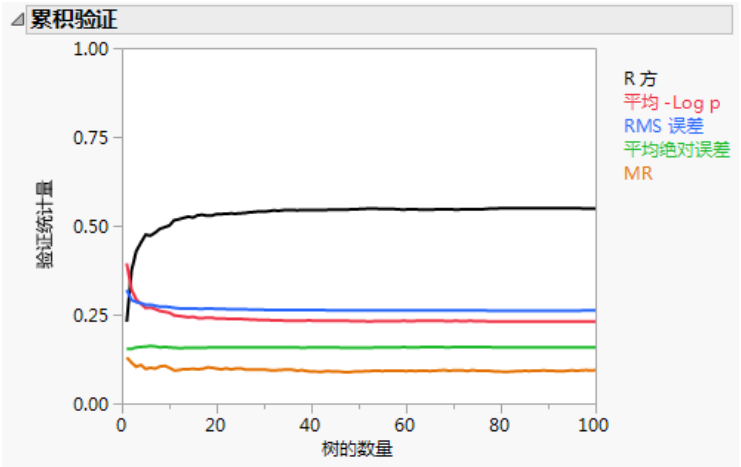
JMP PRO Bootstrap 森林法

通过对许多树求平均值来拟合模型

仅在 JMP Pro 中提供 “Bootstrap 森林法”平台。

“Bootstrap 森林法”平台通过对许多决策树（每棵树都拟合到训练数据的 bootstrap 样本）求平均值来拟合组合模型。每棵树中的每个拆分都考虑随机的预测变量子集。通过这种方式，许多弱模型可组合起来生成更为强大的模型。某个观测的最终预测是该观测在所有决策树上的预测值的平均值。

图 5.1 “累积验证”报表的示例



目录

“Bootstrap 森林法”平台概述 83
包含分类响应的 Bootstrap 森林法的示例 83
Bootstrap 森林法模型 83
缺失值 85
包含连续响应的 Bootstrap 森林法的示例 86
启动“Bootstrap 森林法”平台 88
启动窗口 88
规格窗口 89
“Bootstrap 森林法”报表 91
模型验证集汇总 93
规格 93
总体统计量 93
累积验证 94
按树汇总 95
“Bootstrap 森林法”平台选项 95

JMP PRO “Bootstrap 森林法”平台概述

“Bootstrap 森林法”平台通过对许多决策树的预测响应值求平均值来预测响应值。每棵树都在训练数据的 Bootstrap 样本上增长。Bootstrap 样本是有放回的随机观测样本。此外，在决策树的每个拆分中都对预测变量抽样。使用第 47 页的“分割模型”一章中所述的递归分割方法拟合决策树。

以下是训练集的拟合过程：

1. 对于每棵树，选择观测的 Bootstrap 样本。
2. 使用递归分割拟合各个决策树。
   - 为每个拆分选择随机的一组预测变量。
   - 继续拆分，直到满足在“Bootstrap 森林法规格”窗口中指定的停止规则。
3. 重复执行第 1 步和第 2 步，直到达到在 “Bootstrap 森林法规格”窗口中指定的树个数或直到出现 “提前停止”。

对于单个树，在抽取用于拟合树的观测的 Bootstrap 样本时是有回放的。您可以指定要抽样的观测比例。若指定抽取 100% 的观测，由于抽样有回放，未使用的观测的预期比例为 1/e，即大约 36.8%。对于每一棵树，这些未使用的观测都称为**袋外**观测。拟合树中使用的观测称为**袋内**观测。对于连续响应，“Bootstrap 森林法”平台提供用于袋外观测误差率的测度，称为**袋外误差**。

对于连续响应，某观测的预测值是其在单个树集合中的预测值的平均值。对于分类响应，某观测的预测概率是其在单个树集合中的预测概率的平均值。该观测分类到其预测概率最高的水平中。

有关 Bootstrap 森林法的详细信息，请参见 Hastie et al. (2009)。

JMP PRO 包含分类响应的 Bootstrap 森林法的示例

在本例中，您通过构造 Bootstrap 森林法模型来预测客户是否具有不良信用风险。但您清楚您的数据集包含缺失值，所以您还要探索值缺失的程度。

JMP PRO Bootstrap 森林法模型

1. 选择**帮助** > **样本数据库**，然后打开 Equity.jmp。
2. 选择**分析** > **预测建模** > **Bootstrap 森林法**。
3. 选择**不良**并点击 **Y，响应**。
4. 从**贷款**一直选到**负债收入比**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。

6. 点击**确定**。
7. 在 “每个拆分抽取的项数”旁边输入 4。
8. 在 “每树最大拆分数”旁边输入 30。
9. 选择**在数个项数下进行多个拟合**并在 “最大项数”旁边输入 10。
10. （可选）选择**禁止多线程**并在 “随机种子”旁边输入 123。

    由于 Bootstrap 森林法涉及随机样本，这些操作可确保您获取如下所示的精确结果。
11. 点击**确定**。

图 5.2 “总体统计量”报表

“不良”的 Bootstrap 森林法

规格

总体统计量

测度	训练	验证	测试	定义
熵 R 方	0.5124	0.4828	0.4824	1-Loglike(model)/Loglike(0)
广义 R 方	0.6302	0.6153	0.6068	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.2354	0.2793	0.2618	∑ -Log(p[j])/n
RASE	0.2596	0.2886	0.2771	√ ∑(y[j]-p[j])²/n
绝对偏差的均值	0.1580	0.1779	0.1688	∑ |y[j]-p[j]|/n
误分类率	0.0898	0.1133	0.0990	∑ (p[j]≠pMax)/n
数目	3576	1192	1192	n

混淆矩阵

训练

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	2819	86
Bad Risk	235	436

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.970	0.030
Bad Risk	0.350	0.650

验证

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	884	33
Bad Risk	102	173

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.964	0.036
Bad Risk	0.371	0.629

测试

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	921	28
Bad Risk	90	153

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.970	0.030
Bad Risk	0.370	0.630

由于指定了“在数个项数下进行多个拟合”选项，则在每个拆分中将 4、5、6、8 和 10 用作预测变量数来创建模型。“模型验证集汇总”报表显示验证集有最高的“熵 R 方”的模型是五项模型。这也是误分类率最小的模型。确定该模型为最佳模型，总体报表中的结果是针对该模型的结果。

总体报表显示验证集和测试集的误分类率分别约为 11.3% 和 9.9%。混淆矩阵表明最大的误分类来源是将不良风险客户分类为良性风险客户。

测试集的结果提示您模型扩展到独立观测的能力有多强。验证集用于选择 “Bootstrap 森林法”模型。出于该原因，验证集的结果会有一定偏差地指出模型如何推广到独立数据。

您关注的是确定哪些预测变量对模型的贡献最多。

12. 点击 “‘不良’的 Bootstrap 森林法”旁边的红色小三角，然后选择**列贡献**。

图 5.3 “列贡献”报表

列贡献

项	拆分数	G^2	对应部分
负债收入比	96	567.182771	0.5456
价值	80	104.83095	0.1008
拖欠交易记录数	83	102.545902	0.0986
不良报告数	77	73.3253058	0.0705
最早交易记录时限	68	54.1092672	0.0520
交易记录数	55	30.2200294	0.0291
工作	50	26.1866129	0.0252
最近信用调查次数	51	24.5238489	0.0236
贷款	41	19.7652555	0.0190
未偿抵押贷款	42	17.0533251	0.0164
工作年限	40	16.3836362	0.0158
原因	13	3.45680616	0.0033

“列贡献”报表表明客户信用风险的最强预测变量为**负债收入比**，即负债与收入的比率。对模型贡献第二高的变量包括**拖欠交易记录数**（拖欠债务的贷方行）和**价值**（客户的评估值）。

JMP PRO 缺失值

接下来，您要探索预测变量值缺失的程度。

1. 选择**分析 > 筛选 > 探索缺失值**。
2. 从**不良**一直选到**负债收入比**，然后点击 **Y，列**。
3. 在出现的 “警示”中点击**确定**。

   由于列**原因**和**工作**的数据类型为 “字符”，所以不会将其添加到 “Y，列”列表。您可以使用 “分布”（本示例未显示）查看这两列有多少缺失值。
4. 点击**确定**。

图 5.4 “缺失值”报表

负债收入比列包含 1267 个缺失值，达到观测数的 21%。“Bootstrap 森林法”分析中涉及的大多数其他列也包含缺失值。启动窗口中的“信息性缺失”选项确保在处理缺失值时认可它们所传达的任何信息。请参见 “分割模型”一章中第 64 页的 “信息性缺失”。

JMP PRO 包含连续响应的 Bootstrap 森林法的示例

在本例中，您通过构造 Bootstrap 森林法模型来预测男性对象的体脂肪百分比。

1. 选择**帮助** > **样本数据库**，然后打开 Body Fat.jmp。
2. 选择**分析** > **预测建模** > **Bootstrap 森林法**。
3. 选择**体脂肪百分比**并点击 **Y，响应**。
4. 从**年龄** （年）一直选择到**腕围 (cm)**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。
7. （可选）选择**禁止多线程**并在 “随机种子”旁边输入 123。

   由于 Bootstrap 森林法涉及随机样本，这些操作可确保您获取如下所示的精确结果。
8. 点击**确定**。

图 5.5 总体统计量

“体脂肪百分比” 的 Bootstrap 森林法

规格

目标	体脂肪百分比	训练行:	180
验证列:	验证	验证行:	72
		测试行:	0
森林中树的数量:	100	项数:	13
每个拆分抽取的项数:	10	Bootstrap 抽样次数:	180
		每树最小拆分数:	10
		最小拆分大小:	5

总体统计量

单个树	RASE
袋内	2.025251
袋外	5.995064

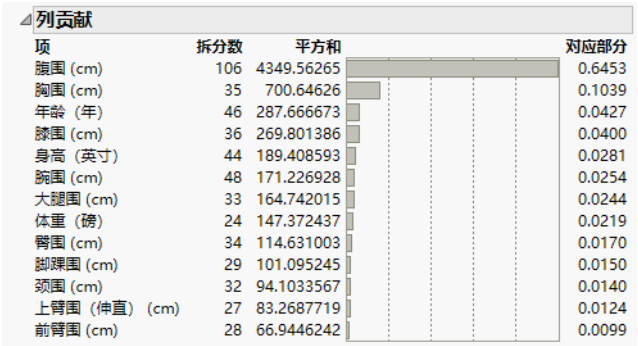
	R 方	RASE	数目
训练	0.892	2.6941446	180
验证	0.675	4.9577438	72

“总体统计量”报表显示“验证 R 方”为 0.675。

您关注的是获取最重要的预测变量的独立于模型的指示。

9. 点击“‘体脂肪百分比’的 Bootstrap 森林法”旁边的红色小三角，然后选择**列贡献**。

图 5.6 列贡献



列贡献

项	拆分数	平方和	对应部分
腹围 (cm)	106	4349.56265	0.6453
胸围 (cm)	35	700.64626	0.1039
年龄 (年)	46	287.666673	0.0427
膝围 (cm)	36	269.801386	0.0400
身高 (英寸)	44	189.408593	0.0281
腕围 (cm)	48	171.226928	0.0254
大腿围 (cm)	33	164.742015	0.0244
体重 (磅)	24	147.372437	0.0219
臀围 (cm)	34	114.631003	0.0170
脚踝围 (cm)	29	101.095245	0.0150
颈围 (cm)	32	94.1033567	0.0140
上臂围 (伸直) (cm)	27	83.2687719	0.0124
前臂围 (cm)	28	66.9446242	0.0099

“列贡献”报表表明**腹围 (cm)**、**胸围 (cm)** 和**年龄** （年）是**体脂肪百分比**的最强预测变量。

JMP PRO 启动 “Bootstrap 森林法” 平台

通过选择**分析** > **预测建模** > **Bootstrap 森林法**启动 “Bootstrap 森林法” 平台。

JMP PRO 启动窗口

图 5.7 “Bootstrap 森林法” 启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“Bootstrap 森林法” 平台启动窗口提供以下选项：

Y，响应 您想要分析的一个或多个响应变量。

X，因子 预测变量。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，该列的数值为分析中的每一行都分配一个频数。

验证 用于定义验证集的数值列。该列应包含至多三个非重复值。

- 若验证列有两个水平，则较小的值定义训练集，较大的值定义验证集。
- 若验证列包含三个水平，将按值由小到大的顺序相应定义训练集、验证集和测试集。

– 若验证列有三个以上的水平，则包含最小三个值的行定义验证集。其他所有行都从分析中排除。

“Bootstrap 森林法” 平台使用验证列来训练和调优模型，或者训练、调优和评估模型。有关验证的详细信息，请参见 “统计详细信息” 附录中第 469 页的 “JMP 建模中的验证”。

若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见第 175 页的 “生成验证列” 一章。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的报表。

方法 支持您选择分割方法 （决策树、 Bootstrap 森林法、提升树、 K 最近邻或朴素 Bayes）。这些备选方法 （决策树除外）均在 JMP Pro 中提供。

有关这些方法的详细信息，请参见第 4 章，“分割模型”、第 6 章，“提升树”、第 7 章，“K 最近邻” 和第 8 章，“朴素 Bayes”。

验证部分 要用作验证集的数据部分。

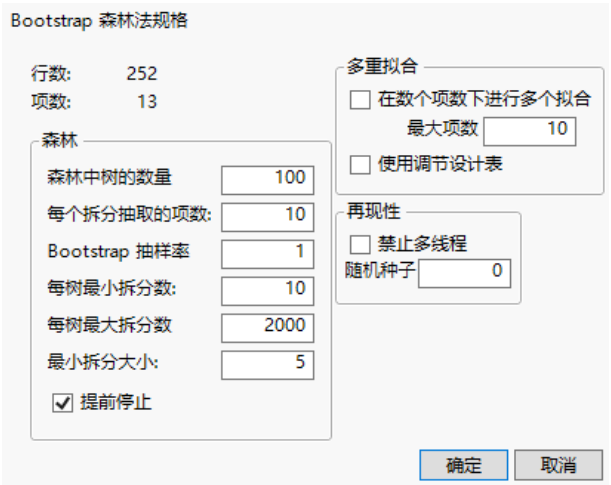
信息性缺失 若选定，则对于分类预测变量，将其缺失值单独作为一类；对于连续预测变量，则对其缺失值采取信息性处理。请参见 “分割模型” 一章中第 64 页的 “信息性缺失”。

有序型限制顺序 若选定，对于有序变量，在拆分时则会进行限定以维持有序变量水平的排序。

JMP PRO 规格窗口

在启动窗口中选择**确定**后，“Bootstrap 森林法规格” 窗口随即显示。

图 5.8 “Bootstrap 森林法规格” 窗口



JMP PRO “规格” 面板

行数 数据表中的行数。

项数 指定为预测变量的列数。

JMP PRO “森林” 面板

森林中树的数量 要增长及随后求平均值的树个数。

每个拆分抽取的项数 要在每个拆分中视为拆分候选项的预测变量数。对于每个拆分，都会选新的随机预测变量样本作为候选集。

Bootstrap 抽样率 要为增长每棵树而抽取的观测比例 （有放回） 。会为每棵树生成新的随机样本。

每树最小拆分数 每棵树的最小拆分数目。

每树最大拆分数 每棵树的最大拆分数目。

最小拆分大小 候选拆分所需的最小观测数。

提前停止 （仅在使用验证时才可用。）在选定时，若更多树不会改善验证统计量，则过程会停止增长更多树。验证统计量是该验证集的熵 R 方（对于分类响应）或该验证集的 R 方值（对于连续响应）。若未选定，过程会继续，直到达到指定数量的树。

JMP PRO “多重拟合” 面板

在数个项数下进行多个拟合 若选定，则可以对 “每个拆分抽取的项数” 的多个可能的不同取值创建 Bootstrap 森林法。显示结果的模型就是验证集的熵 R 方值（对于分类响应）或 R 方（对于连续响应）最大的模型。

下限是 “每个拆分抽取的项数” 规格。上限由下面的选项指定：

最大项数 要考虑进行拆分的最大项数。

使用调节设计表 打开一个窗口，您可以在其中选择包含 “森林” 面板调节参数值的数据表（称为**调节设计表**）。调节设计表针对您想要指定的每个选项都有一列，并且包含一行或多行，每行都表示一个 Bootstrap 森林法模型设计。若未在调节设计表中指定任何选项，则使用默认值。

对于表中的每行， JMP 都使用指定的调节参数创建一个 Bootstrap 森林法模型。若在调节设计表中指定了不止一个模型，则 “模型验证集汇总” 报表列出每个模型的 R 方值。“Bootstrap 森林法” 报表显示具有最大 R 方值的模型的拟合统计量。

您可以使用 “实验设计” 工具创建调节设计表。Bootstrap 森林法调节设计表可以包含以任意顺序排列的不区分大小写的以下列：

– 树的数量

- 项数
- 部分 Bootstrap
- 每树最小拆分数
- 每树最大拆分数
- 最小拆分大小

JMP PRO “再现性”面板

禁止多线程 若选定，则在单个线程上执行所有计算。

随机种子 指定非零数值随机种子，以便将来启动该平台时重现结果。默认情况下，“随机种子”设置为零，即不生成可重现的结果。将分析保存到脚本中时，您输入的随机种子将保存到该脚本中。

JMP PRO “Bootstrap 森林法”报表

在“Bootstrap 森林法规格”窗口中点击**确定**后会出现一个窗口，告知您 JMP 大约需要多长时间来拟合模型。若估计的时间过长，您可以通过选择**接受当前模型**选项查看当前模型。在拟合完整模型后或是在接受当前模型后，“Bootstrap 森林法”报表随即显示。

图 5.9 “接受当前模型”对话框

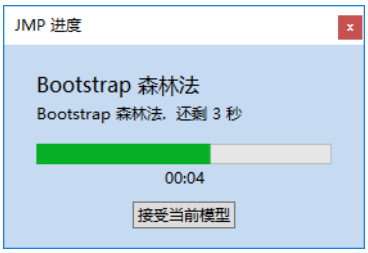
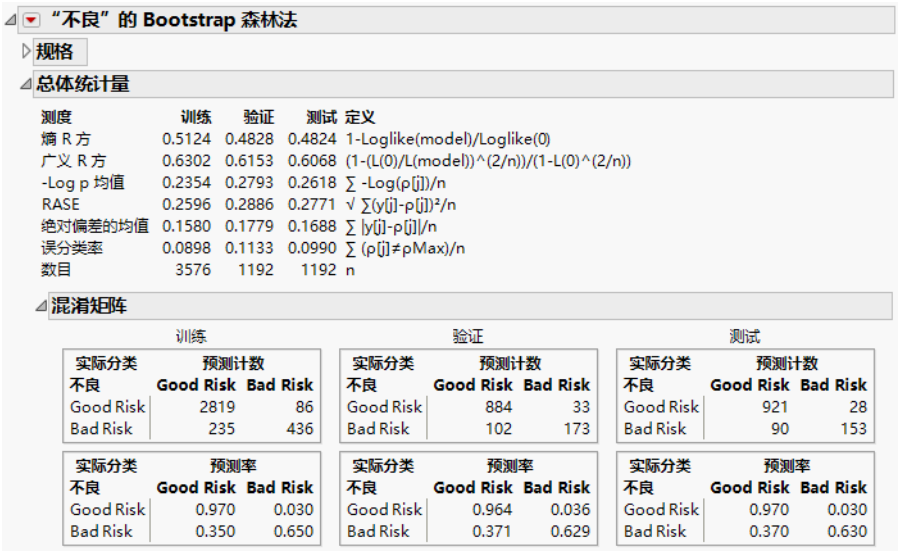


图 5.10 分类响应的 “Bootstrap 森林法”报表



“不良”的 Bootstrap 森林法

规格

总体统计量

测度	训练	验证	测试	定义
熵 R 方	0.5124	0.4828	0.4824	1-Loglike(model)/Loglike(0)
广义 R 方	0.6302	0.6153	0.6068	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.2354	0.2793	0.2618	∑ -Log(ρ[j])/n
RASE	0.2596	0.2886	0.2771	√ ∑(y[j]-ρ[j])²/n
绝对偏差的均值	0.1580	0.1779	0.1688	∑ |y[j]-ρ[j]|/n
误分类率	0.0898	0.1133	0.0990	∑ (ρ[j]≠ρMax)/n
数目	3576	1192	1192	n

混淆矩阵

训练

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	2819	86
Bad Risk	235	436

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.970	0.030
Bad Risk	0.350	0.650

验证

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	884	33
Bad Risk	102	173

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.964	0.036
Bad Risk	0.371	0.629

测试

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	921	28
Bad Risk	90	153

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.970	0.030
Bad Risk	0.370	0.630

图 5.11 连续响应的 “Bootstrap 森林法”报表

“体脂肪百分比”的 Bootstrap 森林法

规格

目标	体脂肪百分比	训练行:	180
验证列:	验证	验证行:	72
		测试行:	0
森林中树的数量:	100	项数:	13
每个拆分抽取的项数:	10	Bootstrap 抽样次数:	180
		每树最小拆分数:	10
		最小拆分大小:	5

总体统计量

单个树	RASE
袋内	2.025251
袋外	5.995064

	R 方	RASE	数目
训练	0.892	2.6941446	180
验证	0.675	4.9577438	72

根据响应是分类响应还是连续响应提供以下报表：

- 第 93 页的“模型验证集汇总”
- 第 93 页的“规格”
- 第 93 页的“总体统计量”
- 第 94 页的“累积验证”
- 第 95 页的“按树汇总”

JMP PRO 模型验证集汇总

（仅当您选择了“Bootstrap 森林法规格”窗口中的“在数个项数下进行多个拟合”选项时才可用。）提供所有模型拟合的拟合统计量。请参见图 5.10 和第 90 页的“‘多重拟合”面板”。

JMP PRO 规格

显示拟合模型时使用的设置。

JMP PRO 总体统计量

提供训练集以及验证集和测试集（若已指定）的拟合统计量。报表的具体形式取决于响应的建模类型。

假定您使用“Bootstrap 森林法规格”窗口中的“在数个项数下进行多个拟合”选项拟合多个模型。之后，结果显示在“总体统计量”报表和“累积验证”报表中的模型就是验证集的熵 R 方值（对于分类响应）或 R 方（对于连续响应）最大的模型。

JMP PRO 分类响应

“测度”报表

提供训练集以及验证集和测试集（若已指定）的以下统计量。

注意：对于熵 R 方和广义 R 方，值越接近 1 表示拟合效果越好。对于 -Log p 均值、RASE、绝对偏差的均值和误分类率，值越小表示拟合效果越好。

熵 R 方 比较拟合模型和恒定概率模型的对数似然的一种拟合测度。它的范围介于 0 到 1 之间。请参见 “统计详细信息”一章中第 469 页的 “熵 R 方”。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。值为 1 表示完美模型，值为 0 表示并不比常数模型好的模型。对于标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R2，它是 Cox and Snell 伪 R2 的标准化版本。

-Log p 均值 -log(p) 的平均值，其中 p 是与发生的事件有关的拟合概率。

RASE 均方根预测误差。差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

绝对偏差的均值 响应的真实值与预测值的差值绝对值的平均值。差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。

数目 观测数。

“混淆矩阵”报表

（仅可用于分类响应。）显示训练集以及验证集和测试集（若已指定）的分类统计量。“混淆矩阵”报表包含混淆矩阵和混淆率矩阵。混淆矩阵是实际响应和预测响应的双向分类。混淆率矩阵等同于混淆矩阵，只不过其中的数字要除以行合计。

决策矩阵

（仅适用于分类响应，并且响应具有“收益矩阵”列属性或者您使用“指定收益矩阵”选项指定了成本。）为训练集以及验证集和测试集（若已指定）提供“决策计数”矩阵和“决策率”矩阵。请参见“分割模型”一章中第 69 页的“分割的更多示例”。

JMP PRO 连续响应

“单个树”报表

提供“袋内”和“袋外”观测在所有树中经过平均的 RASE 值。用于构造树的训练集观测称为**袋内**观测。未用于构造树的训练观测称为**袋外** (OOB) 观测。

对于每棵树，袋外 RASE 都计算为除以 OOB 观测数的误差平方和的平方根。每棵树的袋外 RASE 的平方值在“按树汇总”报表中作为 OOB SSE/N 提供。

R 方和 RASE 报表

提供针对训练集以及验证集和测试集（若定义）的 R 方、均方根预测误差和观测数。

JMP PRO 累积验证

（仅在使用验证时才可用。）显示树的数量-验证集的拟合统计量图。

对于连续响应，唯一的拟合统计量为 “R 方” 。对于分类响应，拟合统计量列在下面，并且在第 93 页的 “‘测度’报表”中进行了说明。

- R 方（熵 R 方）
- 平均 -Log p（-Log p 均值）
- 均方根预测误差 (RASE)
- 平均绝对误差（绝对偏差的均值）)
- MR（误分类率）

“累积验证”图下的“累积详细信息”报表提供图中使用的值。

JMP PRO 按树汇总

“按树汇总”报表涉及袋内和袋外观测的概念。对于单个树，在抽取用于拟合树的观测的 Bootstrap 样本时是有回放的。即便指定对 100% 的观测抽样，但因为抽样时有回放，所以未使用的观测数的预期比例为 1/*e*。对于每一棵树，这些未使用的观测都称为**袋外**观测。拟合树中使用的观测称为**袋内**观测。

“按树汇总”报表显示每棵树的以下汇总统计量：

拆分 决策树中的拆分数。

排名 以升序排列的树的 OOB 损失的排名。 OOB 损失最小的数字树的排名为 1。

OOB 损失 一个测度，用于衡量当应用到“袋外”行时树的总预测误差。较低的值指示较高的预测准确性。

OOB 平均损失 除以 OOB 行数 (OOB 个数) 后的 OOB 损失。

R 方 （仅可用于连续响应。）树的 R 方值。

IB 误差平方和 （仅可用于连续响应。）“袋内”行的误差平方和。

IB 平均误差平方和 （仅可用于连续响应。 ）“袋内”行的误差平方和除以 “袋内”观测数。“袋内”观测数等于训练集中的观测数乘以您在 “Bootstrap 森林法规格”窗口中指定的 Bootstrap 抽样率。

OOB 个数 （仅可用于连续响应。）“袋外”行数。

OOB 误差平方和 （仅可用于连续响应。）当树应用到 “袋外”行时的误差平方和。

OOB 平均误差平方和 （仅可用于连续响应。）除以 OOB 行数 (OOB 个数) 后的 OOB 误差平方和。

JMP PRO “Bootstrap 森林法”平台选项

“Bootstrap 森林法”红色小三角菜单包含以下选项：

标绘 “预测值 - 实际值”图 （仅可用于连续响应。）提供 “预测值 - 实际值”图。

列贡献 显示一个报表，其中显示每个输入列对于拟合的贡献。该报表还显示：

- 指定的列用于拆分数据时，所有树上的实例总数。
- 归因于该列的总 G^2（对于分类响应）或 SS（平方和，对于连续响应）。
- G^2 或 SS 的条形图。
- 归因于该列的 G^2 或 SS 的比例。

显示树状结构 提供用于显示 “树视图”报表中的树的各种选项。该报表提供在提升过程的每一层拟合的树视图。有关 “显示名称类别估计值”选项所显示的 “概率”列的说明，请参见 “分割模型”一章中第 79 页的 “决策树”和 “Bootstrap 森林法”中的预测概率”。

ROC 曲线 （仅可用于分类响应。）请参见 “分割模型”一章中第 65 页的 “ROC 曲线”。

提升曲线 （仅可用于分类响应。）请参见 “分割模型”一章中第 66 页的 “提升曲线”。

保存列 包含用于保存模型和树结果以及创建 SAS 代码的选项。

保存预测值 将模型的预测值保存至数据表。

保存预测公式 将预测公式保存至数据表中的列。该公式中包含描述树结构的嵌套条件子句。若响应是连续的，则该列包含 “预测”属性。若响应是分类的，则该列包含 “响应概率”属性。

保存容缺失预测公式 （应该用 “保存预测公式”选项代替该选项。仅当 “保存预测公式”不可用时才使用该选项。）保存即便存在缺失值并且未选定 “信息性缺失”的情况下也能进行预测的公式。该预测公式通过以下方式容许缺失值：它为拆分随机分配缺失预测变量的响应值。若响应是连续的，则该列包含 “预测”属性。若响应是分类的，则该列包含 “响应概率”属性。若已选定 “信息性缺失”，您可以在按住 Shift 键的同时点击报表的红色小三角来保存 “容缺失预测公式”。

保存残差 （仅可用于连续响应。）将残差保存到数据表。

保存累积详细信息 （仅在使用验证时才可用。）创建包含每棵树的拟合统计量的数据表。

发布预测公式 创建预测公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见第 185 页的 “公式存储库”一章。

发布容缺失预测公式 （应该用 “发布预测公式”选项代替该选项。仅当 “发布预测公式”不可用时才使用该选项。）创建容缺失预测公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见第 185 页的 “公式存储库”一章。若已选定 “信息性缺失”，您可以在按住 Shift 键的同时点击报表的红色小三角来使用该选项。

生成 SAS DATA 步 创建用于对新数据集评分的 SAS 代码。

指定收益矩阵 （仅可用于分类响应。）支持您指定与正确或不正确的分类决策分别关联的收益或成本。请参见 “分割模型”一章中第 61 页的 “显示拟合详细信息”。

刻画器 显示 “预测刻画器”。请参见 《刻画器指南》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

第 6 章

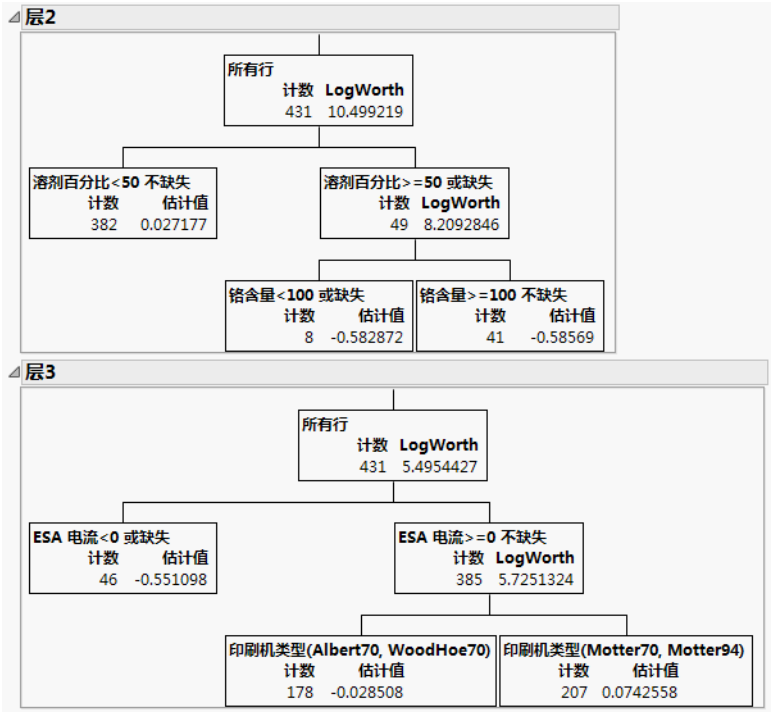
JMP PRO 提升树

拟合树的许多层，每层都基于前一层

仅在 JMP Pro 中提供 “提升树”平台。

提升是通过拟合一系列称为**层**的小决策树来构建较大的加法决策树的过程。每层的树都由少量拆分构成。在之前层的残差的基础上拟合树，这样允许每层都可以更正之前层中拟合较差数据的拟合。某个观测的最终预测是该观测在所有层上的预测的总和。

图 6.1 提升树层的示例



目录

“提升树”平台概述 99
包含分类响应的提升树的示例 99
包含连续响应的提升树的示例 101
启动“提升树”平台 102
规格窗口 103
“提升树”报表 106
模型验证集汇总 108
规格 108
总体统计量 108
累积验证 109
“提升树”平台选项 109
“提升树”平台的统计详细信息 111
过拟合惩罚 111

JMP PRO “提升树”平台概述

“提升树”平台可生成加法决策树模型，该模型基于在**层**中构造的许多小决策树。每层中的树都包含少量拆分，拆分数通常不超过五个。每层都使用第 47 页的“分割模型”一章中所述的递归拟合方法拟合。唯一的区别在于拟合会在指定的拆分数停止。对于给定的树，叶中某观测的预测值是该叶中所有观测的均值。

以下是拟合过程：

1. 拟合初始层。
2. 计算残差。通过从观测的实际值中减去这些观测在叶中的预测均值得到残差。
3. 拟合残差层。
4. 构造加法树。对于给定的观测，将各层上的观测预测值加总。
5. 重复执行第 2 步到第 4 步，一直持续到达到指定的层数；若使用了验证，该过程会一直持续到拟合更多层也不能改善验证统计量为止。

最终的预测是某个预测在所有层上的预测值之和。

通过基于之前层的残差连续拟合各层，每层都可以改善拟合。

对于分类响应，仅支持包含两个响应水平的响应。对于分类响应，每层拟合的残差是线性 Logit 的偏移量。最终的预测是所有层的线性 Logit 总和的 Logistic 变换。

有关提升树的详细信息，请参见 Hastie et al. (2009)。

JMP PRO 包含分类响应的提升树的示例

在本例中，您构造一个提升树模型以预测哪些打印作业受到名为**纹印**的缺陷的影响。

1. 选择**帮助** > **样本数据**，然后打开 Bands Data.jmp。
2. 选择**分析** > **预测建模** > **提升树**。
3. 选择**纹印？**并点击 **Y，响应**。
4. 选择 Predictors 列组并点击 **X，因子**。
5. 为**验证部分**输入 0.2。
6. 点击**确定**。

   随即显示“提升树规格”窗口。
7. （可选）在“再现性”面板中，选择**禁止多线程**并为“随机种子”输入 123。

   由于提升树拟合涉及随机成分，这些操作可确保您获取如下所示的精确结果。
8. 点击**确定**。

图 6.2 名义型响应的总体统计量

总体统计量

测度	训练	验证	定义
熵 R 方	0.7103	0.3299	1-Loglike(model)/Loglike(0)
广义 R 方	0.8340	0.4803	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.1985	0.4383	∑ -Log(ρ[j])/n
RASE	0.2196	0.3801	√ ∑(y[j]-ρ[j])²/n
绝对偏差的均值	0.1650	0.2828	∑ |y[j]-ρ[j]|/n
误分类率	0.0232	0.1852	∑ (ρ[j]≠ρMax)/n
数目	431	108	n

混淆矩阵

训练

实际分类 纹印?	预测计数 band	noband
band	179	9
noband	1	242

实际分类 纹印?	预测率 band	noband
band	0.952	0.048
noband	0.004	0.996

验证

实际分类 纹印?	预测计数 band	noband
band	22	17
noband	3	66

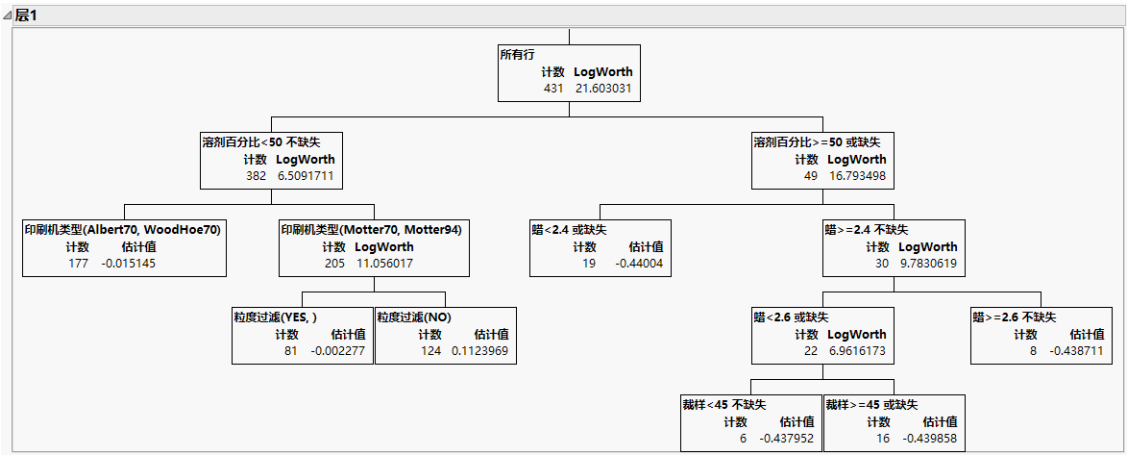
实际分类 纹印?	预测率 band	noband
band	0.564	0.436
noband	0.043	0.957

由于响应**纹印？**是分类响应，提升树分析在“测度”下提供“误分类率”并提供一个“混淆矩阵”报表。验证集的误分类率为 0.1852，或约为 19%。

9. 点击“‘纹印？’的提升树”旁边的红色小三角，然后选择**显示树状结构 > 显示名称类别估计值**。

   “树视图”报表随即显示，并且为各层都显示概要图。您可以检查各层以查看拟合的树和预测值。

图 6.3 提升树的第 1 层



10. 点击“‘纹印？’的提升树”旁边的红色小三角，然后选择**保存列 > 保存预测公式**。

    名为 Prob(纹印？ ==noband)、Prob(纹印？ ==band) 和**最可能的纹印？**的列添加到数据表中。检查 Prob(纹印？ ==noband) 列以查看如何从各层计算模型预测。

JMP PRO 包含连续响应的提升树的示例

在本例中，您构造一个提升树模型，以便预测在名义型和连续因子的特定组合下的体脂肪百分比。

1. 选择**帮助** > **样本数据**并打开 Body Fat.jmp 样本数据表。
2. 选择**分析** > **预测建模** > **提升树**。
3. 选择**体脂肪百分比**并点击 **Y，响应**。
4. 从**年龄 （年）**一直选择到**腕围 (cm)**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。
7. 点击**确定**。

图 6.4 连续响应的总体统计量

总体统计量

	R 方	RASE	数目
训练	0.847	3.2116281	180
验证	0.611	5.4301412	72

“总体统计量”报表提供提升树模型的 R 方和 RASE。验证集的 R 方为 0.611。验证集的 RASE 约为 5.43。

您关注的是获取**体脂肪百分比**的重要预测变量的独立于模型的指示。

8. 点击“‘体脂肪百分比’的提升树”旁边的红色小三角，然后选择**刻画器**。
9. 点击“预测刻画器”旁边的红色小三角，然后选择**评估变量重要性** > **独立均匀输入**。

注意：由于“评估变量重要性”使用随机处理，您的结果可能不会与图 6.5 中的结果完全匹配。

图 6.5 变量重要性的汇总报表

变量重要性: 独立均匀输入

汇总报表

列	主效应	总效应
腹围 (cm)	0.859	0.883
年龄 （年）	0.065	0.089
腕围 (cm)	0.006	0.013
上臂围 （伸直） (cm)	0.003	0.007
臀围 (cm)	0.003	0.007
胸围 (cm)	0.002	0.004
脚踝围 (cm)	0.002	0.004
大腿围 (cm)	0.001	0.002
身高 （英寸）	0.001	0.002
体重 （磅）	0.001	0.002
膝围 (cm)	0.001	0.001
颈围 (cm)	9e-18	2e-17
前臂围 (cm)	9e-18	2e-17

“汇总报表”显示**腹围 (cm)** 是**体脂肪百分比**最重要的预测变量。

JMP PRO 启动“提升树”平台

通过选择**分析 > 预测建模 > 提升树**启动“提升树”平台。

图 6.6 使用 Body Fat.jmp 的“提升树”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“提升树”平台启动窗口提供以下选项：

Y，响应 您想要分析的一个或多个响应变量。

X，因子 预测变量。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，该列的数值为分析中的每一行都分配一个频数。

验证 用于定义验证集的数值列。该列应包含至多三个非重复值。

- 若验证列有两个水平，则较小的值定义训练集，较大的值定义验证集。
- 若验证列包含三个水平，将按值由小到大的顺序相应定义训练集、验证集和测试集。

– 若验证列有三个以上的水平，则包含最小三个值的行定义验证集。其他所有行都从分析中排除。

“提升树” 平台使用验证列来训练和调优模型，或者训练、调优和评估模型。有关验证的详细信息，请参见 “统计详细信息” 附录中第 469 页的 “JMP 建模中的验证”。

若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见第 175 页的 “生成验证列” 一章。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的报表。

方法 支持您选择分割方法 （决策树、 Bootstrap 森林法、提升树、 K 最近邻或朴素 Bayes）。这些备选方法 （决策树除外）均在 JMP Pro 中提供。

有关这些方法的详细信息，请参见第 4 章，“分割模型”、第 5 章，“Bootstrap 森林法”、第 7 章，“K 最近邻” 和第 8 章，“朴素 Bayes”。

验证部分 要用作验证集的数据部分。

信息性缺失 若选定，则对于分类预测变量，将其缺失值单独作为一类；对于连续预测变量，则对其缺失值采取信息性处理。请参见 “分割模型” 一章中第 64 页的 “信息性缺失”。

有序型限制顺序 若选定，对于有序变量，在拆分时则会进行限定以维持有序变量水平的排序。

JMP PRO 规格窗口

在启动窗口中选择**确定**后，“梯度提升树规格” 窗口随即显示。

图 6.7 “提升树规格” 窗口

JMP PRO “提升”面板

层数 要包含在最终树中的最大层数。

每树拆分数 每层的拆分数。

学习率 $0 < r \leq 1$ 之类的数字。学习率接近 1 会导致最终树收敛更快，但也更容易过度拟合数据。当指定小的“层数”时，可以使用接近 1 的学习率。学习率通常是介于 0.01 和 0.1 之间的一个小数，它可以减慢模型收敛的速度。这样可以保留机会，以便之后的层可以使用与之前层不同的拆分。

过拟合惩罚 （仅可用于分类响应。）一个偏倚参数，该参数可帮助防止拟合概率等于 0。请参见第 111 页的“过拟合惩罚”。

最小拆分大小 候选拆分所需的最小观测数。

JMP PRO “多重拟合”面板

在数个拆分和学习率下进行多个拟合 若选定，则可以为“每树拆分数”（以整数递增）与“学习率”（以 0.1 递增）的每个组合创建一个提升树。

组合的下限由“每树拆分数”与“学习率”选项指定。组合的上限由以下选项指定：

每树最大拆分数 每树拆分数的上限。

最大学习率 学习率的下限。

使用调节设计表 打开一个窗口，您可以在其中选择包含某些调节参数的数据表（称为**调节设计表**）。调节设计表针对您想要指定的每个选项都有一列，并且包含一行或多行，每行都表示一个提升树模型设计。若未在调节设计表中指定任何选项，则使用默认值。

对于表中的每行，JMP 都使用指定的调节参数创建一个提升树模型。若在调节设计表中指定了不止一个模型，则“模型验证集汇总”报表列出每个模型的 R 方值。“提升树”报表显示具有最大 R 方值的模型的拟合统计量。

您可以使用“实验设计”工具创建调节设计表。提升树调节设计表可以包含以下以任意顺序排列的不区分大小写的列：

- 层数
- 每树拆分数
- 学习率
- 最小拆分大小
- 行抽样率
- 列抽样率

JMP PRO “随机梯度提升” 面板

行抽样率 为每层抽样的训练行的比例。

> **注意：**若响应为分类响应，则使用分层随机抽样对训练行抽样。

列抽样率 为每层抽样的预测变量列的比例。

JMP PRO “再现性” 面板

禁止多线程 若选定，则在单个线程上执行所有计算。

随机种子 指定非零数值随机种子，以便将来启动该平台时重现结果。默认情况下，“随机种子” 设置为零，即不生成可重现的结果。将分析保存到脚本中时，您输入的随机种子将保存到该脚本中。

JMP PRO 提前停止

提前停止 若选定，则提升过程会在更多层不会改善验证统计量时停止拟合更多层。若未选定，则提升过程会继续，直到达到指定的层数。仅当使用验证时，该选项才显示。

JMP PRO “提升树”报表

在“梯度提升树规格”窗口中点击“确定”后，“提升树”报表随即打开。

图 6.8 连续响应的“提升树”报表

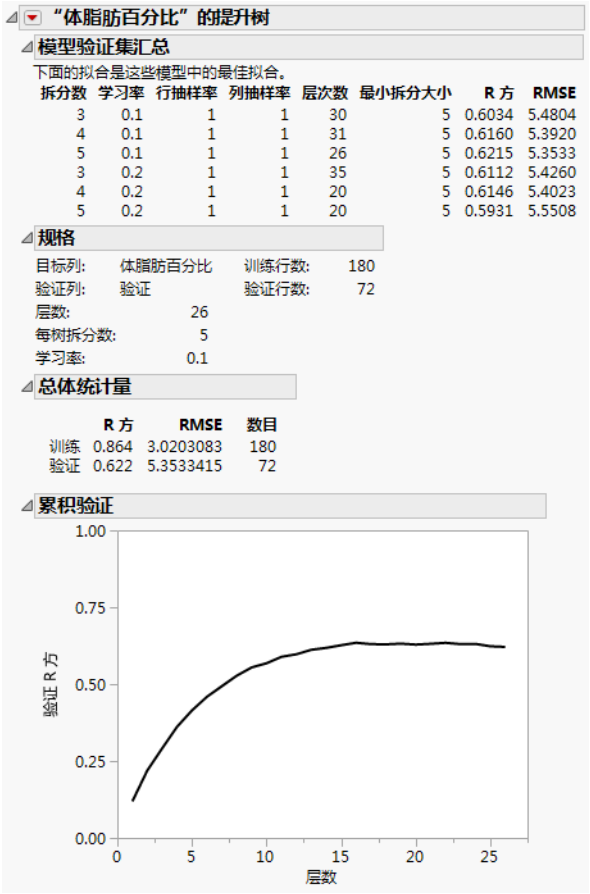
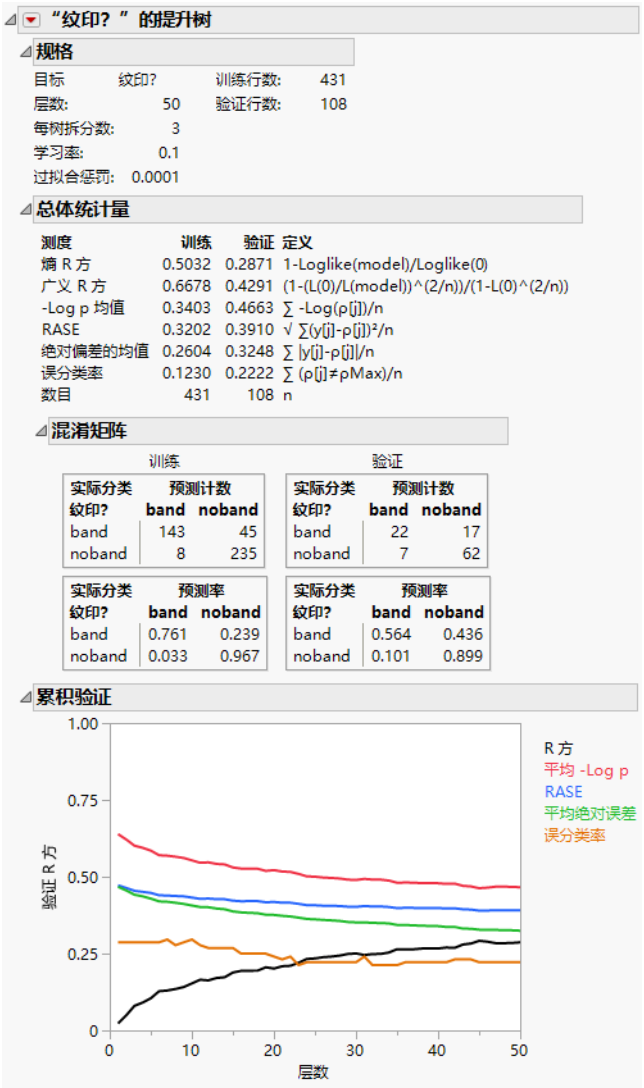


图 6.9 分类响应的 “提升树”报表



根据响应是分类响应还是连续响应提供以下报表：

- 第 108 页的 “模型验证集汇总”
- 第 108 页的 “规格”
- 第 108 页的 “总体统计量”
- 第 109 页的 “累积验证”

JMP PRO 模型验证集汇总

若在“规格”窗口中选定“在数个拆分和学习率下进行多个拟合”选项，则显示所有模型拟合的拟合统计量。请参见图 6.8和第 104 页的““多重拟合”面板”。

JMP PRO 规格

显示拟合模型时使用的设置。

JMP PRO 总体统计量

显示训练集以及验证集和测试集（若已指定）的拟合统计量。

假定您使用“提升树规格”窗口中的“在数个拆分和学习率下进行多个拟合”选项拟合多个模型。之后，结果显示在“总体统计量”报表和“累积验证”报表中的模型就是验证集的熵 R 方值（对于分类响应）或 R 方（对于连续响应）最大的模型。

JMP PRO “测度”报表

（仅可用于分类响应。）提供训练集以及验证集和测试集（若已指定）的以下统计量。

注意：对于熵 R 方和广义 R 方，值越接近 1 表示拟合效果越好。对于 -Log p 均值、RASE、绝对偏差的均值和误分类率，值越小表示拟合效果越好。

熵 R 方 比较拟合模型和恒定概率模型的对数似然的一种拟合测度。它的范围介于 0 到 1 之间。请参见“统计详细信息”一章中第 469 页的“熵 R 方”。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。值为 1 表示完美模型，值为 0 表示并不比常数模型好的模型。对于标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。

-Log p 均值 -log(p) 的平均值，其中 p 是与发生的事件有关的拟合概率。

RASE 均方根预测误差。差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

绝对偏差的均值 响应的真实值与预测值的差值绝对值的平均值。差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。

数目 观测数。

JMP PRO “混淆矩阵”报表

（仅可用于分类响应。）显示训练集以及验证集和测试集（若已指定）的分类统计量。“混淆矩阵”报表包含混淆矩阵和混淆率矩阵。混淆矩阵是实际响应和预测响应的双向分类。混淆率矩阵等同于混淆矩阵，只不过其中的数字要除以行合计。

JMP PRO 决策矩阵

（仅适用于分类响应，并且响应具有“收益矩阵”列属性或者您使用“指定收益矩阵”选项指定了成本。）为训练集以及验证集和测试集（若已指定）提供“决策计数”矩阵和“决策率”矩阵。请参见“分割模型”一章中第 69 页的“分割的更多示例”。

JMP PRO 累积验证

（仅在使用验证时才可用。）显示层数-验证集的拟合统计量图。

对于连续响应，唯一的拟合统计量为“R 方”。对于分类响应，拟合统计量列在下面，并且在第 108 页的“‘测度’报表”中进行了说明。

- R 方（熵 R 方）
- 平均 -Log p（-Log p 均值）
- 均方根预测误差 (RASE)
- 平均绝对误差（绝对偏差的均值）)
- MR（误分类率）

“累积验证”图下的“累积详细信息”报表提供图中使用的值。

JMP PRO “提升树”平台选项

“提升树”红色小三角菜单包含以下选项：

显示树状结构 提供用于显示“树视图”报表中的树的选项。该报表提供在提升过程的每一层拟合的树视图。

标绘“预测值 - 实际值”图 （仅可用于连续响应。）提供“预测值 - 实际值”图。

列贡献 显示一个报表，其中显示每个输入列对于拟合的贡献。该报表还显示：

- 指定的列用于拆分数据时，所有树上的实例总数。
- 归因于该列的总 G^2（对于分类响应）或 SS（平方和，对于连续响应）。
- G^2 或 SS 的条形图。
- 归因于该列的 G^2 或 SS 的比例。

ROC 曲线 （仅可用于分类响应。）请参见 “分割模型”一章中第 65 页的 “ROC 曲线”。

提升曲线 （仅可用于分类响应。）请参见 “分割模型”一章中第 66 页的 “提升曲线”。

保存列 包含用于保存模型和树结果以及创建 SAS 代码的选项。

保存预测值 将模型的预测值保存至数据表。

保存预测公式 将预测公式保存至数据表中的列。该公式中包含描述树结构的嵌套条件子句。若响应是连续的，则该列包含 “预测”列属性。若响应是分类的，则该列包含 “响应概率”列属性。

保存容缺失预测公式 （应该用 “保存预测公式”选项代替该选项。仅当 “保存预测公式”不可用时才使用该选项。）保存即便存在缺失值并且未选定 “信息性缺失”的情况下也能进行预测的公式。该预测公式通过以下方式容许缺失值：它为拆分随机分配缺失预测变量的响应值。若响应是连续的，则该列包含 “预测”列属性。若响应是分类的，则该列包含 “响应概率”列属性。若已选定 “信息性缺失”，您可以在按住 Shift 键的同时点击报表的红色小三角来保存 “容缺失预测公式”。

保存残差 （仅可用于连续响应。）将残差保存到数据表。

保存剩余偏差估计值 （仅可用于分类响应。）保存线性成分的总和。这些估计值是拟合概率的 Logit 变换。

保存树状结构详细信息 创建包含拆分详细信息和每层估计值的数据表。

保存累积详细信息 （仅在使用验证时才可用。）创建包含每层的拟合统计量的数据表。

发布预测公式 创建预测公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见第 185 页的 “公式存储库”一章。

发布容缺失预测公式 （应该用 “发布预测公式”选项代替该选项。仅当 “发布预测公式”不可用时才使用该选项。）创建容缺失预测公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见第 185 页的 “公式存储库”一章。若已选定 “信息性缺失”，您可以在按住 Shift 键的同时点击报表的红色小三角来使用该选项。

生成 SAS DATA 步 创建用于对新数据集评分的 SAS 代码。

指定收益矩阵 （仅可用于分类响应。）支持您指定与正确或不正确的分类决策分别关联的收益或成本。请参见 “分割模型”一章中第 61 页的 “显示拟合详细信息”。

刻画器 显示 “预测刻画器”。请参见 《刻画器指南》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

JMP PRO “提升树”平台的统计详细信息

本节说明特定于“提升树”平台的详细信息。有关递归决策树的统计详细信息，请参见“分割模型”一章中第 78 页的““分割”平台的统计详细信息”。

JMP PRO 过拟合惩罚

若响应为分类响应，则施加参数惩罚。对于每一层，针对负对数似然加上乘以每个观测估计值的平方和之后的罚值的结果，估计值都会将其最小化。该罚值可防止每个新层过拟合训练数据。

第 7 章

JMP PRO K 最近邻

使用邻近观测预测响应值

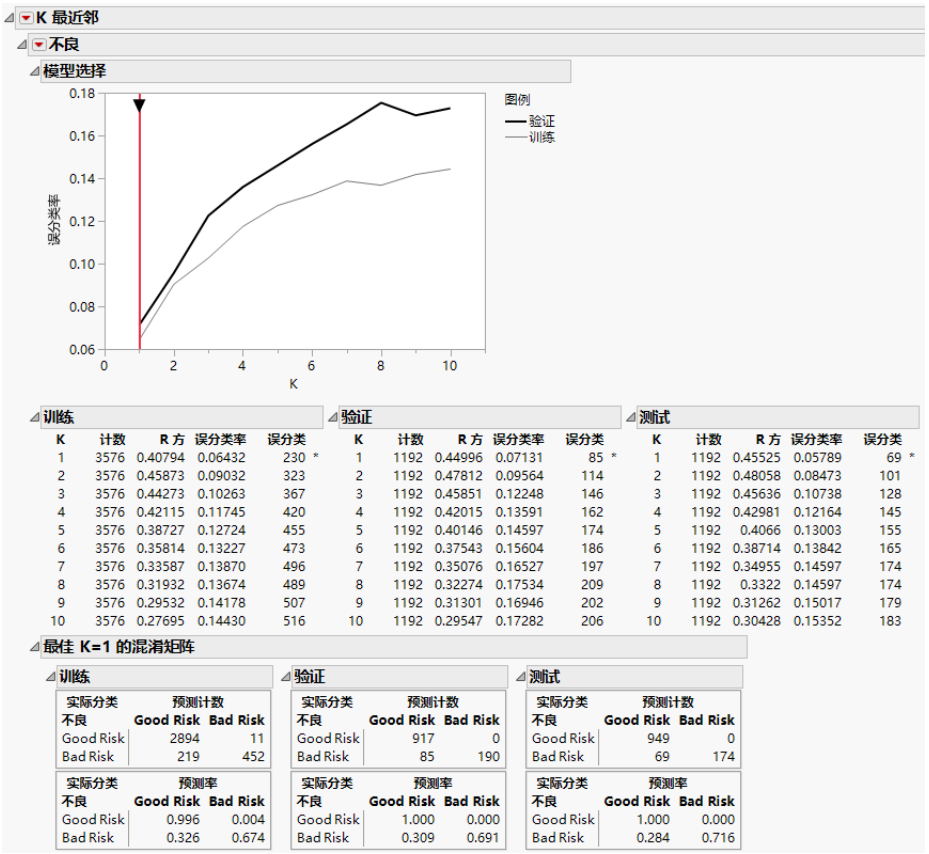
仅在 JMP Pro 中提供 “K 最近邻”平台。

“K 最近邻”平台使用给定观测的本地邻域中的观测的响应预测该观测的响应值。它可用于分类响应的分类以及连续响应的预测。

“K 最近邻”是一种非参数方法，该方法基于与邻近观测的距离。鉴于以上事实，“K 最近邻”能够使用不规则的预测变量值边界对观测分类。但是，因为该算法对不相关的预测变量很敏感，预测变量的选择会影响您的结果。

“K 最近邻”已经成功地应用于许多应用，比如：对卫星影像和心电图 (EKG) 形态进行分类。

图 7.1 “K 最近邻”平台示例



目录

“K 最近邻”平台概述 115
包含分类响应的 K 最近邻的示例 115
包含连续响应的 K 最近邻的示例 117
启动“K 最近邻”平台 119
“K 最近邻”报表 120
连续响应 120
分类响应 121
“K 最近邻”平台选项 122
模型拟合选项 122

JMP PRO “K 最近邻”平台概述

“K 最近邻”平台基于 *k* 个最近邻的响应预测响应值。给定观测的 *k* 个最近邻通过以下方式来确定：标识该观测的预测变量值与其他每个观测的预测变量值之间的 *k* 个最小欧氏距离。“K 最近邻”平台可对连续响应和分类响应建模。

k 最近邻方法的一个潜在缺点是对于大规模问题，预测公式通常很复杂，难于解释，这限制了它的使用。此外，“K 最近邻”不计算分类响应的概率。有关 *k* 最近邻方法的详细信息，请参见 Hastie et al. (2009)、Hand et al. (2001) 和 Shmueli et al.(2017) 中的表 J.1a、J.1b、J.6a 和 J.6b。

连续响应

对于连续响应，预测值是 *k* 个最近邻的响应的平均值。每个连续预测变量都按照其标准差来统一尺度。借助这种统一尺度，范围很大的单个预测变量不会过度影响距离计算。连续预测变量的缺失值将被该预测变量的均值取代。请参见第 117 页的“包含连续响应的 K 最近邻的示例”。

分类响应

对于分类响应，预测值是 *k* 个最近邻出现次数最多的响应水平。若最频繁的水平同时有两个或两个以上，则随机从这些水平中选择一个来分配预测响应。

注意：对于分类响应，由于最频繁的水平的结值是随机断开的，所以每次运行平台的结果可能有所不同。要获得可重现的结果，请使用启动窗口中的“设置随机种子”选项或在 JSL 脚本中包含 `Set Random Seed()` 函数。

在分类预测模型中，每个分类预测变量用指标变量的形式表示，用一个指标变量表示每个水平。包含分类预测变量缺失值的行在该预测变量的所有指标变量上都用零值来表示。

JMP PRO 包含分类响应的 K 最近邻的示例

您有关于已申请房屋净值贷款的 5,960 个客户的历史财务数据。每个客户都被归入“良性风险”或“不良风险”两类。很多预测变量有缺失值。您想要构造模型，用于对将来客户的信用风险分类。

1. 选择**帮助** > **样本数据库**，然后打开 Equity.jmp。
2. 选择**分析** > **预测建模** > **K 最近邻**。
3. 选择**不良**并点击 **Y，响应**。
4. 从**贷款**一直选到**交易记录数**，然后点击 **X，因子**。

由于其中一个潜在预测变量**负债收入比**有很多缺失值，所以您没有将其包含在模型内。连续预测变量的缺失值用该预测变量的平均值替代。该过程有时对于随机缺失的值很有效。尽管负债收入比的高缺失率指示缺失可能是信息性的，但是我们在本例中不调查它。

5. 选择**验证**并点击**验证**。
6. 点击**确定**。

图 7.2 “K 最近邻”报表

训练				验证				测试			
K	计数	误分类率	误分类	K	计数	误分类率	误分类	K	计数	误分类率	误分类
1	3576	0.06432	230 *	1	1192	0.07131	85 *	1	1192	0.05789	69 *
2	3576	0.08389	300	2	1192	0.09228	110	2	1192	0.09060	108
3	3576	0.10263	367	3	1192	0.12248	146	3	1192	0.10738	128
4	3576	0.11633	416	4	1192	0.14178	169	4	1192	0.11913	142
5	3576	0.12724	455	5	1192	0.14597	174	5	1192	0.13003	155
6	3576	0.13311	476	6	1192	0.16275	194	6	1192	0.13926	166
7	3576	0.13870	496	7	1192	0.16527	197	7	1192	0.14597	174
8	3576	0.13954	499	8	1192	0.17030	203	8	1192	0.15017	179
9	3576	0.14178	507	9	1192	0.16946	202	9	1192	0.15017	179
10	3576	0.14206	508	10	1192	0.17701	211	10	1192	0.15268	182

训练		
实际分类	预测计数	
不良	良性风险	不良风险
良性风险	2894	11
不良风险	219	452

验证		
实际分类	预测计数	
不良	良性风险	不良风险
良性风险	917	0
不良风险	85	190

测试		
实际分类	预测计数	
不良	良性风险	不良风险
良性风险	949	0
不良风险	69	174

对于 *K* 的每个值，JMP 仅使用训练集观测构造模型。这些模型均用于对验证集观测分类。使用验证集结果来选择最佳模型。在本例中，基于单个最近邻 (*K* = 1) 的模型的误分类率最小。测试集确认这一单个最近邻模型对于独立数据的效果最佳。

7. 点击**不良**红色小三角并选择**发布预测公式**。
8. 在**近邻数，** K 旁边，保留默认值 1。
9. 点击**确定**。

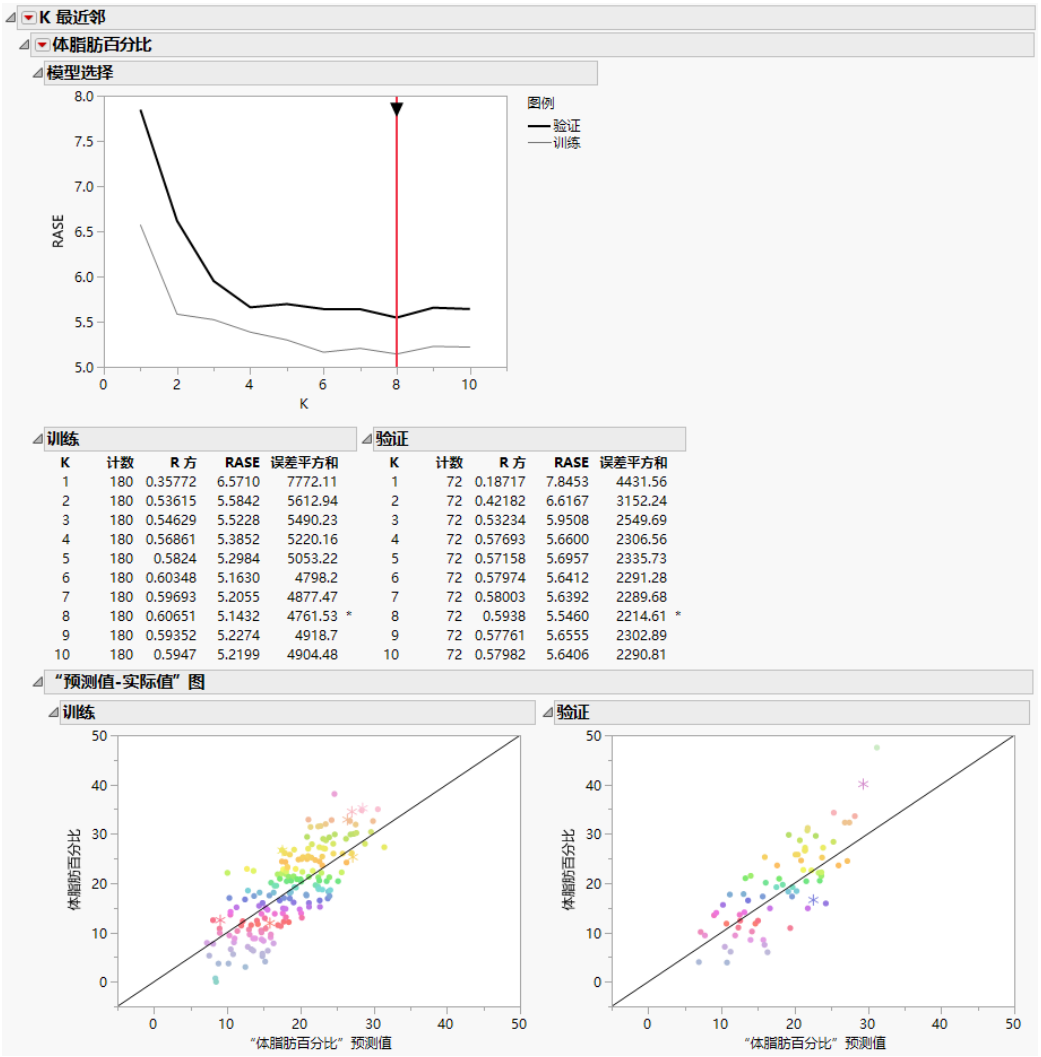
预测方程保存在 “公式存储库”中。您可以使用 “公式存储库”中的 “模型比较”选项比较发布到 “公式存储库”中的备择模型和 $K = 1$ 最近邻模型的效果。请参见第 185 页的 “公式存储库”一章。

JMP PRO 包含连续响应的 K 最近邻的示例

在本例中，您想要使用 13 个预测变量预测男性体脂肪百分比。Body Fat.jmp 样本数据表包含基于水下称重和多种体围测量值的体脂肪百分比估计值。

1. 选择**帮助** > **样本数据库**，然后打开 Body Fat.jmp。
2. 选择**分析** > **预测建模** > **K 最近邻**。
3. 选择**体脂肪百分比**并点击 **Y，响应**。
4. 从**年龄** （年）一直选择到**腕围 (cm)**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。
7. 点击**体脂肪百分比**红色小三角并选择**标绘** “**预测值 - 实际值**”图。

图 7.3 “K 最近邻”报表

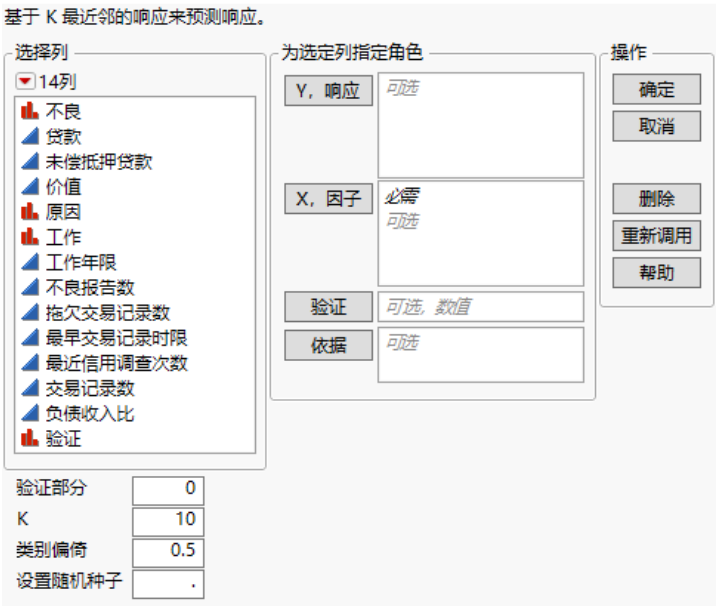


$K = 8$ 模型针对验证集具有最低的 RASE。在 k 最近邻模型之间，基于 8 个最近邻的模型看起来效果最佳。训练集的“预测值 - 实际值”图显示点落在线上，指示这些预测值与实际值相似。验证集的图上的大多数点落在线上，只有上端的几个观测离线远一点。

JMP PRO 启动 “K 最近邻” 平台

通过选择**分析** > **预测建模** > **K 最近邻**启动 “K 最近邻” 平台。

图 7.4 “K 最近邻” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“K 最近邻” 启动窗口提供以下选项：

Y，响应 您想要分析的一个或多个响应变量。

> **注意：** 即使不存在响应变量，“K 最近邻” 平台也可用作确定相邻观测之间距离的实用工具。若未指定响应变量，则会出现空白报表。不过，可以使用红色小三角菜单选项 “保存近邻行” 和 “保存近邻距离”。

X，因子 预测变量。

验证 用于定义验证集的数值列。该列应包含至多三个非重复值。

- 若验证列有两个水平，则较小的值定义训练集，较大的值定义验证集。
- 若验证列包含三个水平，将按值由小到大的顺序相应定义训练集、验证集和测试集。
- 若验证列有三个以上的水平，则包含最小三个值的行定义验证集。其他所有行都从分析中排除。

“K 最近邻” 平台使用验证列来训练和调优模型，或者训练、调优和评估模型。有关验证的详细信息，请参见 “统计详细信息” 附录中第 469 页的 “JMP 建模中的验证”。

若在“选择列”列表中没有选择任何列的情况下点击“验证”按钮，您可以向数据表添加一个验证列。有关“生成验证列”实用工具的详细信息，请参见第 175 页的“生成验证列”一章。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

验证部分 要用作验证集的数据部分。

近邻数，K 要分析的最大最近邻数目。针对一个最近邻拟合模型，最多可使用您为 *K* 指定的值进行拟合。

注意：最大近邻数 *K* 比训练数据表中的行数大不能超过 1。若指定的 *K* 大于允许的最大 *K*，则会显示警告。

类别偏倚 指定一个确保分类响应的拟合概率始终为正的调节参数。默认情况下，“类别偏倚”为 0.5。

设置随机种子 设置随机化过程的种子，用于断开名义型响应和有序型响应的结值。若您指定“验证部分”，该选项还设置用于验证的行的种子。若您要重现分析，“设置随机种子”很有用。若您设置一个随机种子并保存脚本，该种子将自动保存在脚本中。

JMP PRO “K 最近邻”报表

“K 最近邻”报表包含用于每个响应变量的单独报表。每个响应变量报表包含有关该响应的拟合模型的信息。该信息包含一个“模型选择”报表和拟合的 *k* 个模型各自的汇总信息。该报表显示针对训练集的表以及针对验证集和测试集的表（若已使用验证定义这些数据集）。

“模型选择”报表基于误分类率（对于分类响应）或 RASE（对于连续响应）显示跨越 *K* 的解路径图。默认情况下，将滑块置于对应于最佳效果模型的 *K* 的值。您可以拖动滑块来更改报表中 *K* 的值。

报告的统计量取决于响应的建模类型。汇总表中的每行对应于 *k* 个最近邻所定义的模型，其中 *K* 的范围为 1 到您在启动窗口中为“近邻数，K”指定的值。

JMP PRO 连续响应

默认情况下，除了“模型选择”图之外，连续响应的报表还包含“汇总表”报表。

汇总表

用星号来标记具有最小 RASE 的 *K* 值所属的模型。连续响应的报表包含以下列：

K 模型中使用的最近邻的数目。*K* 的范围介于 1 到您在启动窗口中指定的“近邻数，K”之间。

计数 观测数。

R 方 模型的 R 方值。

RASE 模型的均方根预测误差。用星号来标记具有最小 RASE 的模型。若有多个相同的 RASE 值，具有最小 *K* 的模型将标记有星号。

误差平方和 模型的误差平方和。

JMP PRO 分类响应

默认情况下，除了“模型选择”图之外，用于分类响应的报表还包含“汇总表”、“混淆矩阵”和“马赛克图”报表。

汇总表

用星号来标记具有最小误分类率的 *K* 值所属的模型。分类响应的报表包含以下列：

K 模型中使用的最近邻的数目。*K* 的范围介于 1 到您在启动窗口中指定的“近邻数，K”之间。

计数 观测数。

误分类率 模型误分类的观测比例。该值用误分类数除以计数计算得出。用星号来标记具有最小误分类率的模型。若有多个相同的误分类率，具有最小 *K* 的模型将标记有星号。

误分类 模型预测有误的观测数。

混淆矩阵

默认情况下，为具有最小“误分类率”的模型显示“混淆矩阵”报表。若有多个相同的最小误分类率，则显示具有最小 *K* 的模型的报表。“混淆矩阵”报表包含混淆矩阵和混淆率矩阵。混淆矩阵是实际响应和预测响应的双向分类。混淆率矩阵等同于混淆矩阵，只不过其中的数字要除以行合计。若使用验证，将显示验证集和测试集的混淆矩阵和混淆率矩阵。使用“混淆矩阵”报表和误分类率来评估您的模型。

提示：若您在解路径图中更改滑块的位置，将为所选的 *K* 值显示一个额外的“混淆矩阵”报表。使用这一额外的报表来比较备择模型和默认最佳模型。

马赛克图

默认情况下，为具有最小误分类率的模型显示马赛克图。若有多个相同的最小误分类率，则为具有最小 *K* 值的模型显示马赛克图。马赛克图是一种堆叠条形图，其中每段与组的频数统计成比例。有关马赛克图的详细信息，请参见《基本分析》。若您使用验证，则显示验证集和测试集的马赛克图。

提示：若您在解路径图中更改滑块的位置，将更新马赛克图以便为所选的 *K* 值显示结果。

JMP PRO “K 最近邻”平台选项

“K 最近邻”红色小三角菜单包含以下选项：

保存近邻行 用于将 *K* 个列保存至数据表。这些列的名称为**行近邻 : <k>**。对于给定行，第 *k* 列包含其第 *k* 个最近邻的行号。

警告：当您对数据表中的行重新排序时，列**行近邻 : <k>** 中的行号不会更新。若您对行重新排序，这些列中的值将不正确。

保存近邻距离 用于将 *K* 个列保存至数据表。这些列命名为**距离 <k>**。对于给定行，第 *k* 列包含了到该行的第 *k* 个最近邻的距离。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

模型拟合选项

马赛克图 （仅适用于名义型或有序型响应。）显示或隐藏马赛克图。请参见第 121 页的“马赛克图”。

标绘 “预测值 - 实际值”图 （仅可用于连续响应。）针对具有最小 RASE 的模型标绘预测响应 - 实际值的图。若有多个相同的最小 RASE，则该图的绘制基于具有最小 *K* 的模型。

提示：若您将解路径图上的滑块位置更改为另一个 *K*，将更新“预测值 - 实际值”图以反映对应所选的 *K* 值的模型。

标绘 “预测值 - 残差”图 （仅可用于连续响应。）针对具有最小 RASE 的模型标绘预测响应 - 实际值的图。若有多个相同的最小 RASE，则该图的绘制基于具有最小 *K* 的模型。

提示：若您将解路径图上的滑块位置更改为另一个 *K*，将更新“预测值 - 残差”图以反映对应所选的 *K* 值的模型。

保存预测值 用于将 *K* 个预测值列保存至数据表。这些列的名称为**预测值 : < 响应 > <k>**。第 *k* 个列包含基于 *k* 个最近邻的模型的预测，其中**响应**是响应列的名称。

保存预测公式 用于保存包含特定 *k* 最近邻模型的预测公式的列。在出现提示时输入 *K* 的值。预测公式包含所有训练数据，所以该选项可能对大型数据表不实用。

警告：从“保存预测公式”和“保存预测值”获得的值不一定匹配。从“保存预测公式”选项获得的值使用训练集中的所有行，包括预测值所在的行。来自“保存预测值”选项的值不使用训练集中预测值所在的行，仅使用其他所有的行。

发布预测公式 为指定的 *k* 最近邻模型创建预测公式，并将其另存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见第 185 页的“公式存储库”一章。

第 8 章

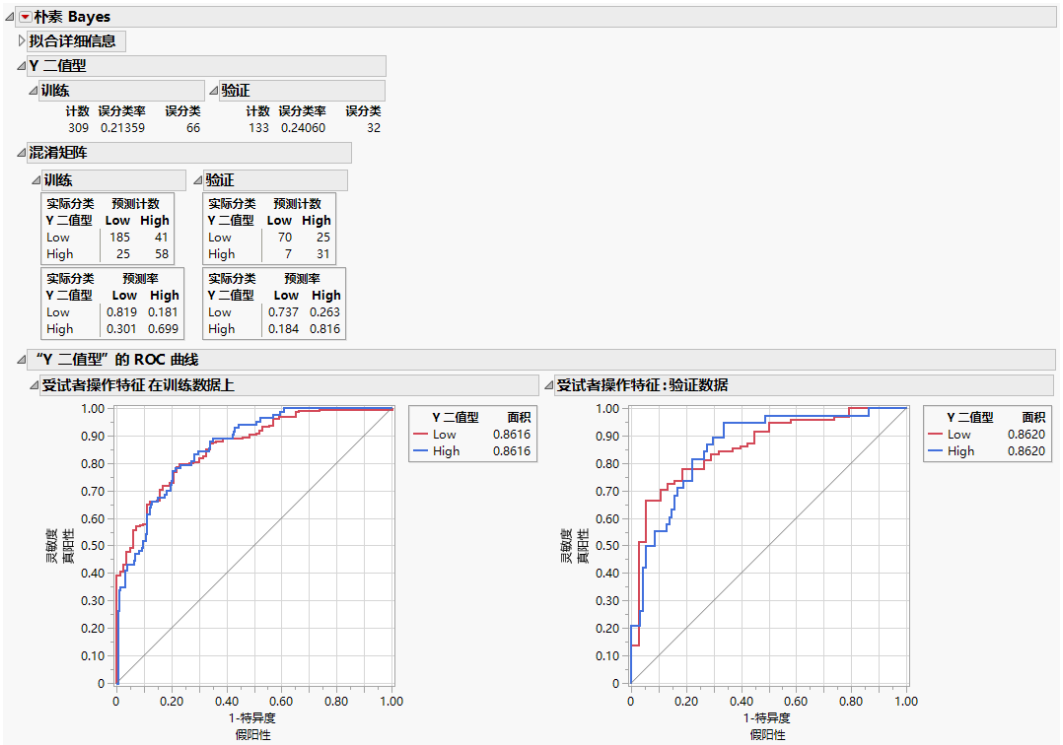
JMP PRO 朴素 Bayes

使用概率假设将观测分类

仅在 JMP Pro 中提供 “朴素 Bayes”平台。

“朴素 Bayes”平台拟合模型来预测分类变量的值。朴素 Bayes 是速度快、计算简单的方法。它特别适合有很多预测变量的情形。

图 8.1 朴素 Bayes 分析的示例



目录

“朴素 Bayes”平台概述 127
朴素 Bayes 的示例 127
启动“朴素 Bayes”平台 129
“朴素 Bayes”报表 131
“拟合详细信息”报表 131
响应列报表 131
“混淆矩阵”报表 131
ROC 曲线 132
“朴素 Bayes”平台选项 132
朴素 Bayes 的更多示例 133
“朴素 Bayes”平台的统计详细信息 134
算法 134
已保存的概率公式 135

JMP PRO “朴素 Bayes”平台概述

“朴素 Bayes”平台将观测分为由分类响应变量的水平定义的几个**类**。用于分类的变量（或因子）在数据挖掘文档中通常称为**特征**。

对于每个类，朴素 Bayes 算法计算每个特征值出现的条件概率。若某特征是连续的，则估计其条件边缘密度。朴素 Bayes 方法假定在一个类中各个特征是相互独立的。（这是该方法被称为“朴素”的原因。）分类基于这样的原理：其特征值在某一类中具有很高的条件概率的观测有很高的概率属于该类。请参见 Hastie 等人 (2009)。

因为该算法只估计一维密度或分布，因此速度很快。这使它适合大数据集，特别是具有大量特征的数据集。使用观测的所有非缺失特征值来计算条件概率。

对于每个类，给每个观测分配一个**朴素得分**。一个观测的给定类的朴素得分是属于该类的训练观测的比例乘以观测的条件概率之积。观测属于某类的**朴素概率**是该类的朴素得分除以它在所有类上的朴素得分之和。将观测分配给它具有最高朴素概率的类。

警告：因为假定类成员之间的条件概率是相互独立的，因此朴素 Bayes 估计的概率效率低下。

朴素 Bayes 需要大量训练观测来确保所有预测变量值和类的表示。若正在将验证集中的观测分类且它具有在训练集中缺失的分类预测变量值，则平台将使用非缺失特征来预测。若某个观测缺失所有预测值，则预测响应为最常见的响应。预测公式处理缺失值的方式是：令其不对观测得分作出任何贡献。

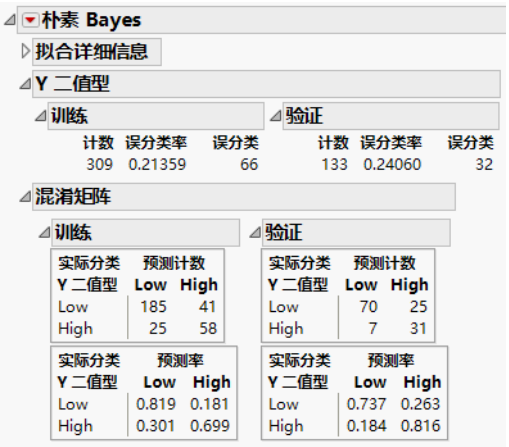
有关朴素 Bayes 方法的详细信息，请参见 Hand 等人 (2001) 和 Shmueli et al.(2010)。

JMP PRO 朴素 Bayes 的示例

您想构造一个分类模型以便在预测未来患者的疾病发展为“High”或“Low”时使用。您有 442 名糖尿病患者的基准医疗数据，还有在每名患者首次就诊一年后得到的糖尿病疾病发展的二值型测度。该测度将疾病发展量化为“Low”或“High”。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 选择**分析** > **预测建模** > **朴素 Bayes**。
3. 选择 **Y 二值型**并点击 **Y，响应**。
4. 从**年龄**一直选到**葡萄糖**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。

图 8.2 “朴素 Bayes”报表

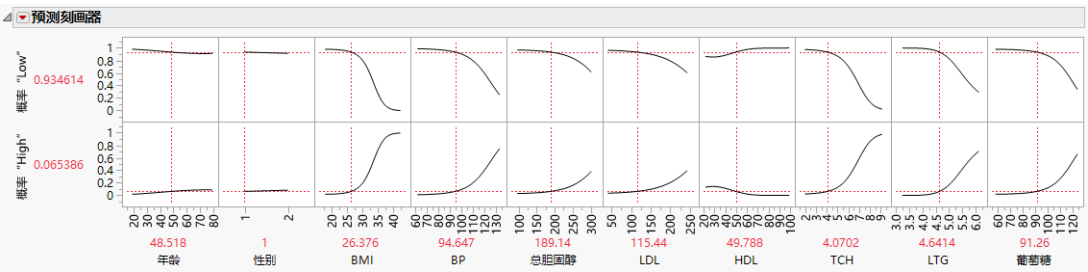


训练集大约有 21% 的误分类率，验证集大约有 24% 的误分类率。混淆矩阵指示对于训练集和验证集，较大的误分类源来自将具有 “Low”疾病发展的患者归入具有 “High”疾病发展一类。验证集结果指示模型如何扩展到独立观测。

您想了解哪些预测变量对朴素 Bayes 分类的影响最大。

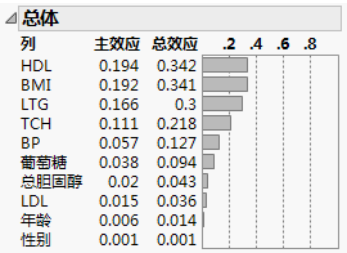
7. 点击 “朴素 Bayes”红色小三角并选择**刻画器**。

图 8.3 疾病发展的预测刻画器



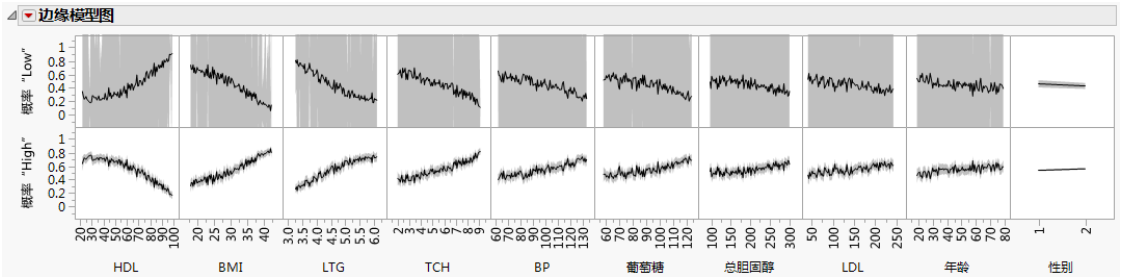
8. 点击 “预测刻画器”红色小三角并选择**评估变量重要性 > 独立均匀输入**。

图 8.4 变量重要性



“汇总报表”指示 HDL、 BMI 和 LTG 对估计的概率具有最大影响。

图 8.5 “边缘模型图”报表

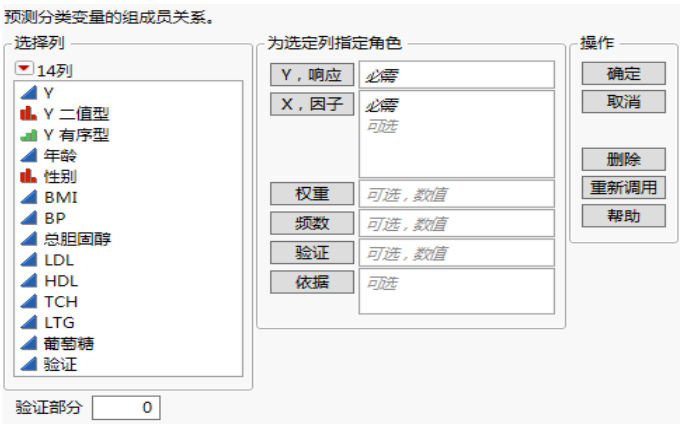


“边缘模型图”报表中图的第二行显示较高的 HDL 值与将患者归类为 “High”的较小概率关联。而且，较高的 BMI 和 LTG 值与将患者归类为 “High”的较大概率关联。

JMP PRO 启动“朴素 Bayes”平台

通过选择**分析** > **预测建模** > **朴素 Bayes** 来启动 “朴素 Bayes”平台。

图 8.6 “朴素 Bayes”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“朴素 Bayes”启动窗口提供以下选项：

Y，响应 其值为相关类的分类响应列。

X，因子 分类或连续预测变量列。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，该列的数值为分析中的每一行都分配一个频数。

验证 用于定义验证集的数值列。该列应包含至多三个非重复值。

- 若该列有三个唯一值，则：

  具有最小值的行用于训练集。

  具有中间值的行用于验证集。

  具有最大值的行用于测试集。
- 若该列有两个唯一值，则仅使用训练集和验证集。
- 若验证列有三个以上的水平，则包含最小三个值的行定义验证集。其他所有行都从分析中排除。

“朴素 Bayes”平台使用验证列来训练和评估模型。有关验证的详细信息，请参见 “统计详细信息”附录中第 469 页的 “JMP 建模中的验证”。

若在 “选择列”列表中没有选择任何列的情况下点击 “验证”按钮，您可以向数据表添加一个验证列。有关“生成验证列”实用工具的详细信息，请参见第 175 页的 “生成验证列”一章。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在单独的报表中。若指定了多个 “依据”变量，将为 “依据”变量水平的每种可能组合生成单独的分析。

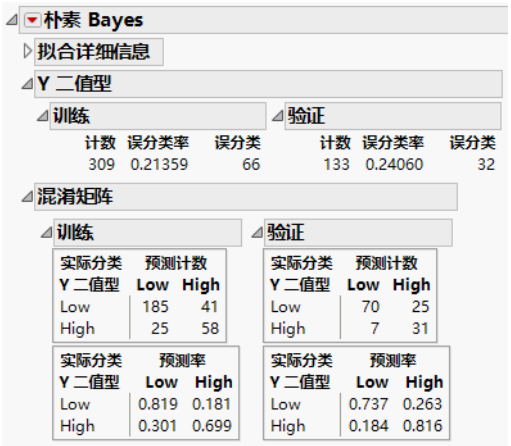
验证部分 要用作验证集的数据部分。

注意：若未在启动窗口中指定一个“验证”列或“验证部分”且有排除的列，则将这些行作为验证集处理。有关验证的详细信息，请参见“统计详细信息”一章中第 469 页的 “JMP 建模中的验证”。

JMP PRO “朴素 Bayes”报表

在启动窗口中点击**确定**后，显示“朴素 Bayes”报表。默认情况下，“朴素 Bayes”报表包含针对拟合详细信息、响应列、混淆矩阵以及 ROC 和提升曲线的报表。

图 8.7 “朴素 Bayes”报表



JMP PRO “拟合详细信息”报表

“拟合详细信息”报表为训练集以及验证集和测试集（若已指定）显示模型的各种拟合测度。“测度”列列出不同的拟合统计量，“定义”列显示相应拟合统计量的公式。请参见“模型比较”一章中第 170 页的“分类响应的“拟合测度””。默认情况下，“朴素 Bayes”报表窗口中的“拟合详细信息”报表处于关闭状态。

JMP PRO 响应列报表

响应列报表显示汇总表中训练集、验证集和测试集 （若指定它们）的朴素 Bayes 分类的性能统计信息。汇总表包含以下列：

计数 集内与表 （训练集、验证或测试集）对应的观测数。

误分类率 被模型误分类的相应集中的观测比例。该值用误分类数除以计数计算得出。

误分类 被错误分类的相应集内的观测数。

JMP PRO “混淆矩阵”报表

“混淆矩阵”报表显示训练集的混淆矩阵和混淆率，若指定了验证集和测试集，也显示这个两个集的混淆矩阵和混淆率。混淆矩阵是实际响应和预测响应的双向分类。混淆率矩阵等同于混淆矩阵，只不过其中的数字要除以行合计。

JMP PRO ROC 曲线

报表为训练集以及验证集和测试集（若已指定）显示“受试者操作特征 (ROC)”曲线。ROC 曲线测量拟合概率对响应水平正确分类的能力。曲线距离对角线越远，拟合效果越好。有关 ROC 曲线的简介，请参见《基本分析》。

有关 ROC 曲线的详细信息，请参见“分割模型”一章中第 65 页的“ROC 曲线”。

JMP PRO “朴素 Bayes”平台选项

“朴素 Bayes”红色小三角菜单包含以下选项：

ROC 曲线 显示或隐藏 ROC 图。默认显示 ROC 图。请参见第 132 页的 “ROC 曲线”。

提升曲线 显示或隐藏模型的提升曲线。若使用验证，则会为每个训练集、验证集和测试集（若已指定）都显示提升曲线。

提升曲线显示响应水平如何随着拟合概率的下降进行有效分类。拟合概率沿水平轴以降序标绘。拟合概率的垂直坐标是该概率（或更高）的正确分类的比例除以总正确分类率的结果。使用提升曲线可查看：当您仅选择拟合概率超出阈值的那些观测时，您是否能对较大比例的观测正确分类。

若响应具有两个水平，提升曲线图只会为该响应的第一个水平显示提升曲线。若响应有两个以上水平，则提升曲线图会为每个响应水平都显示曲线的子概图。有关提升曲线的详细信息，请参见 “分割模型”一章中第 66 页的 “提升曲线”。

保存预测值 将预测分类保存到数据表中名为**朴素预测的 “<Y，响应 >”**的列。

保存预测公式 将名为**朴素预测公式 “<Y，响应 >”**的列保存到数据表。该列包含分类的预测公式。

保存概率公式 将列（包含用于将每个观测分类的公式）保存到数据表。保存三组列：

朴素得分“<类>”，朴素得分和 对于表示类的每个列，该列给出衡量给定类中成员身份强度的得分公式。在“朴素得分和”列中，将各个类上的这些得分累加。请参见第 135 页的“已保存的概率公式”。

朴素概率“<类>” 对于每个类，该列给出某观测属于该类的条件概率的公式。请参见第 135 页的“已保存的概率公式”。

朴素预测公式“<Y，响应>” 给出预测类的公式。

发布概率公式 创建概率公式并将它们保存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将打开“公式存储库”窗口。请参见第 185 页的 “公式存储库”一章。

刻画器 显示或隐藏交互式刻画器报表。因子值的更改体现在估计的分类概率中。请参见 《刻画器指南》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO 朴素 Bayes 的更多示例

您有关于已申请房屋净值贷款的 5,960 个客户的历史财务数据。每个客户都被归入“良性风险”或“不良风险”两类。多数预测变量都有缺失数据。您想要构造模型，用于对将来客户的信用风险分类。

1. 选择**帮助** > **样本数据库**，然后打开 Equity.jmp。
2. 选择**分析** > **预测建模** > **朴素 Bayes**。
3. 选择**不良**并点击 **Y，响应**。

   潜在预测变量之一**负债收入比**具有很多可能是信息性的缺失值。但是，朴素 Bayes 不能很好处理大量缺失值，因此您不能在模型中包括**负债收入比**。
4. 从**贷款**一直选到**交易记录数**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。

图 8.8 “不良”的朴素 Bayes 报表

不良

训练

计数	误分类率	误分类
3576	0.18009	644

验证

计数	误分类率	误分类
1192	0.19128	228

测试

计数	误分类率	误分类
1192	0.18960	226

混淆矩阵

训练

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	2691	214
Bad Risk	430	241

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.926	0.074
Bad Risk	0.641	0.359

验证

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	859	58
Bad Risk	170	105

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.937	0.063
Bad Risk	0.618	0.382

测试

实际分类	预测计数	
不良	Good Risk	Bad Risk
Good Risk	885	64
Bad Risk	162	81

实际分类	预测率	
不良	Good Risk	Bad Risk
Good Risk	0.933	0.067
Bad Risk	0.667	0.333

训练集、验证集和测试集显示误分类率介于 18% 到 19% 之间。所有这些集的混淆矩阵指示最大的误分类来源是将 “不良风险”客户归入 “良性风险”客户。

您关注具有特定财务背景值的客户被归入 “高风险”一类的概率。

7. 点击“朴素 Bayes”红色小三角并选择**保存概率公式**。

   将对应三个集的列添加到数据表。

   - 三个“朴素得分”列包含“良性风险”、“不良风险”以及这两者之和的朴素得分公式。
   - 两个“朴素概率”列包含“良性风险”和“不良风险”的概率公式。
   - **朴素预测公式“不良”**列包含将观测分配给它具有最高朴素概率的类的公式。

   使用这些公式获得新客户的得分。有关公式列的详细信息，请参见第 135 页的“已保存的概率公式”。

JMP PRO “朴素 Bayes”平台的统计详细信息

- “算法”
- “已保存的概率公式”

JMP PRO 算法

朴素 Bayes 方法将某观测归入在给出特征值的情况下它是其成员的概率最高的类。该方法假定特征在每个类中有条件独立。

用 $C_1, \ldots, C_K$ 表示可能的分类。用 $X_1, X_2, \ldots, X_p$ 表示特征或预测变量。

按以下方式计算具有预测变量值 $x_1, x_2, \ldots, x_p$ 的观测属于 C_k 类的条件概率：

$$P(C_k|(x_1, \ldots, x_p)) = \frac{\left(P(C_k)\prod_{j=1}^{p}[P(x_j|C_k)]\right)/(\mathbf{R})}{\sum_{k=1}^{K}\left(P(C_k)\left(\prod_{j=1}^{p}[P(x_j|C_k)]\right)/\mathbf{R}\right)},$$

其中，**R** 是正则化常数。在上面的公式中，按以下方式确定 $X_j = x_j$ 的观测属于 C_k，$P(x_j|C_k)$ 类的条件概率：

- 若 X_j 是分类变量：

$$P(x_j|C_k) = \frac{\text{观测数 } x_j|C_k}{\text{观测数 } C_k}$$

- 若 X_j 是连续变量：

$$P(x_j|C_k) = \frac{1}{s_{jk}}\phi((x_j - m_{jk})/s_{jk})$$

其中，ϕ 是标准正态密度函数，m 和 s 分别是 C_k 类中预测变量值的均值和标准差。

某观测属于 C_k，$P(C_k)$ 类的非条件概率按以下方式计算：

$$P(C_k) = \frac{\text{观测数 } C_k + (0.5/K)}{\text{总观测数} + 0.5}$$

将某观测归入其条件概率最大的类。

注意：在 $P(C_k)$ 的公式中，0.5 是先验偏倚因子。该值是默认值。要更改先验默认因子，请转至“文件”>“首选项”>“平台”>“朴素 Bayes”，然后选中“先验偏倚”复选框，再更改该值。

JMP PRO 已保存的概率公式

本节说明使用“保存概率公式”选项保存的公式。具有预测变量值 $x_1, x_2, \ldots, x_p$ 的观测属于 C_k 类的条件概率与 $P(C_k|(x_1,\ldots, x_p))$ 给出的概率（显示在第 134 页的“算法”一节中）稍有不同。这样做是为了提高计算效率。

朴素得分公式

给定类 C_k，$S(C_k)$ 的朴素得分公式是 $P(C_k|(x_1,\ldots, x_p))$ 的表达式中分子的变化形式，该公式计算如下：

$$S(C_k) = \exp\left(\sum_{j=1}^{p} \ln(1/s_{jk}) + P(C_k) + \sum_{j=1}^{p} \text{正态对数密度}\left(\frac{x_j - m_{jk}}{s_{jk}}\right) - \ln(\mathbf{R}) \right)$$

朴素得分和公式

朴素得分和公式 S 对所有类的朴素得分公式求和。这是 $P(C_k|(x_1,\ldots, x_p))$ 的表达式中分母的变化形式。

$$S = \sum_{k=1}^{K} S(C_k)$$

朴素概率公式

给定类 C_k 的朴素概率公式等于 $P(C_k|(x_1,\ldots, x_p))$。在 JMP 公式中，

$$P(C_k|(x_1, \ldots, x_p)) = \frac{S(C_k)}{S}$$

朴素预测公式

某观测的朴素预测公式将该观测归入 $P(C_k|(x_1,..., x_p))$ 最大的类。这等价于将观测归入其朴素得分公式最大的类。

第 9 章

JMP PRO 支持向量机

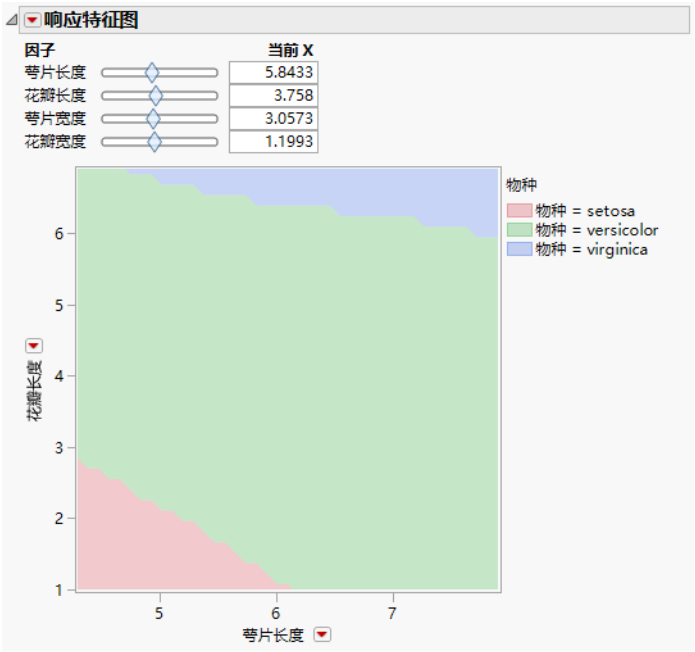
使用支持向量预测响应类别

仅在 JMP Pro 中提供 “支持向量机”平台。

在 JMP Pro 中，“支持向量机 (SVM)”平台为非线性回归和分类问题提供机器学习工具。SVM 算法是一种监督学习算法，它利用训练数据建立模型对新观测进行预测或分类。

若您有分类响应，您可以在“响应特征图”中可视化模型分类。SVM 平台还提供误分类率、混淆矩阵、ROC 和提升曲线，用来帮助您确定模型拟合数据的良好程度。若您有连续响应，则使用拟合统计量和“预测值-实际值”图来确定模型拟合数据的良好程度。

图 9.1 响应特征图



目录

支持向量机概述 139
支持向量机示例 139
启动“支持向量机”平台 141
“支持向量机”启动窗口 141
“模型启动”控制面板 143
“支持向量机”报表 144
“模型比较”报表 145
“支持向量机模型”报表 146
“支持向量机”平台选项 147
“支持向量机模型”报表选项 147
SVM 平台的更多示例 149
连续响应的支持向量回归示例 149

JMP PRO 支持向量机概述

支持向量机 (SVM) 模型是一种监督学习算法，用于对新观测进行预测或分类。基于响应已知的一组训练数据拟合模型。然后，该模型用于预测新观测的响应。

若响应为分类响应，SVM 模型通过优化用于分离类的超平面来对数据进行分类。这也可以看作是找到最大化类之间边距的超平面。在简单的问题中，这个超平面是线性的。然而，更复杂的数据往往无法线性分离。对于这些情形，SVM 平台提供了使用径向基函数内核将点映射到非线性维度的选项，这样可以使类更容易分离。

若响应为连续响应，则拟合的模型称为支持向量回归 (SVR) 模型。在典型的回归问题中，目标是拟合一个模型，使预测响应和实际响应之间的误差最小化。在 SVR 问题中，目标是拟合一个模型，使预测响应和实际响应之间的误差落在 -ε 到 ε 的范围内。这会提供一个更灵活的拟合。在 JMP Pro 中，ε 等于 0.1。SVR 算法通过创建两个类 Y + ε 和 Y - ε 使数据增加一倍。然后，用于分类问题的同一算法也用于预测 (SVR) 问题。

SVM 算法的最大化是通过求解二次规划问题来实现的。在 JMP Pro 中，SVM 平台使用的算法基于 John Platt 于 1998 年提出的序列最小优化 (SMO) 算法。通常，SVM 二次规划问题非常大。SMO 算法将整体二次规划问题分解为一系列较小的二次规划问题。较小的二次规划问题是解析解而不是数值解，这意味着它们生成闭式解。因此，SMO 算法比求解整体二次规划问题更有效 (Platt 1998)。

JMP PRO 支持向量机示例

您有 442 名糖尿病患者的基准医疗数据，还有在每名患者首次就诊一年后得到的糖尿病疾病发展的二值型测度。该测度将疾病发展量化为 “Low” 或 “High”。您想构造一个分类模型以便预测未来患者的疾病发展为 “High” 还是 “Low”。您想要探索核函数选项。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 选择**分析** > **预测建模** > **支持向量机**。
3. 选择 **Y 二值型**并点击 **Y，响应**。
4. 从**年龄**一直选到**葡萄糖**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。
7. 在 “模型启动” 控制面板中，检查核函数是否为 “径向基函数” 并且选择 “调节设计”。
8. 在 “运行次数” 旁边输入 10。
9. 点击**执行**。
10. 点击 “模型启动” 旁边的灰色小三角以打开 “模型启动” 控制面板。
11. 将核函数改为线性函数并选择 “调节设计”。
12. 在 “运行次数” 旁边输入 10。

13. 点击**执行**。

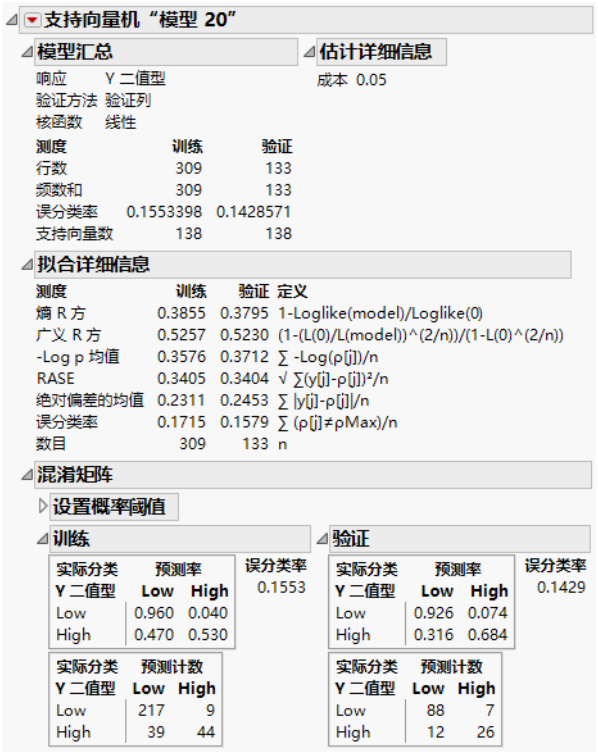
图 9.2 “模型比较”报表

模型比较

显示	方法	核函数	成本	Gamma	# SV	训练误分类率	验证误分类率	验证广义 R 方	概率阈值	阈值验证误分类率	最佳
☐	模型 1	径向基函数	2.57061	0.43877	251	0.00324	0.25564	0.00838	0.39079	0.25564	
☐	模型 2	径向基函数	3.04271	0.31899	218	0.01294	0.28571	-0.4206	0.23223	0.28571	
☐	模型 3	径向基函数	3.93906	0.46899	255	0.00324	0.28571	-0.0081	0.53372	0.28571	
☐	模型 4	径向基函数	1.073	0.27968	208	0.0712	0.21805	0.22158	0.17372	0.21805	
☐	模型 5	径向基函数	0.41677	0.0281	156	0.15534	0.18797	0.50485	0.36015	0.18797	
☐	模型 6	径向基函数	0.23369	0.39898	246	0.26861	0.28571	0.22323	6e-8	0.28571	
☐	模型 7	径向基函数	4.8013	0.15144	167	0.05178	0.2782	-0.4245	0.26219	0.2782	
☐	模型 8	径向基函数	3.51202	0.20295	181	0.04207	0.28571	-0.454	0.22753	0.28571	
☐	模型 9	径向基函数	4.2693	0.06093	140	0.11974	0.19549	0.42841	0.32812	0.19549	
☐	模型 10	径向基函数	1.87641	0.0919	150	0.13269	0.21053	0.43499	0.30598	0.21053	
☐	模型 11	线性	4.42735	.	122	0.17476	0.17293	0.51944	0.47722	0.17293	
☐	模型 12	线性	4.99046	.	123	0.16828	0.17293	0.52037	0.47551	0.17293	
☐	模型 13	线性	3.8272	.	123	0.17476	0.17293	0.51925	0.47763	0.17293	
☐	模型 14	线性	2.74788	.	124	0.17152	0.17293	0.51723	0.48019	0.17293	
☐	模型 15	线性	3.22207	.	123	0.17476	0.17293	0.5185	0.47842	0.17293	
☐	模型 16	线性	1.29304	.	124	0.17152	0.17293	0.51523	0.4819	0.17293	
☐	模型 17	线性	1.74671	.	124	0.16828	0.17293	0.51556	0.48143	0.17293	
☐	模型 18	线性	0.83575	.	123	0.16505	0.15789	0.51498	0.48207	0.15789	
☐	模型 19	线性	2.22	.	125	0.17152	0.17293	0.5158	0.48255	0.17293	
☑	模型 20	线性	0.04975	.	138	0.15534	0.14286	0.52302	0.41946	0.14286	最小误分类率 最大 R 方

“模型比较”报表显示：依据误分类率和 R 方，最佳模型为模型 20。该模型具有成本参数为 0.04975 的线性核函数。这是要进一步分析的模型。

图 9.3 “最佳拟合模型”的模型报表



支持向量机 “模型 20”

模型汇总

响应	Y 二值型
验证方法	验证列
核函数	线性

测度	训练	验证
行数	309	133
频数和	309	133
误分类率	0.1553398	0.1428571
支持向量数	138	138

估计详细信息

成本 0.05

拟合详细信息

测度	训练	验证	定义
熵 R 方	0.3855	0.3795	1-Loglike(model)/Loglike(0)
广义 R 方	0.5257	0.5230	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
-Log p 均值	0.3576	0.3712	∑ -Log(ρ[j])/n
RASE	0.3405	0.3404	√ ∑(y[j]-ρ[j])²/n
绝对偏差的均值	0.2311	0.2453	∑ |y[j]-ρ[j]|/n
误分类率	0.1715	0.1579	∑ (ρ[j]≠ρMax)/n
数目	309	133	n

混淆矩阵

设置概率阈值

训练

实际分类 Y 二值型	预测率 Low	预测率 High
Low	0.960	0.040
High	0.470	0.530

误分类率 0.1553

实际分类 Y 二值型	预测计数 Low	预测计数 High
Low	217	9
High	39	44

验证

实际分类 Y 二值型	预测率 Low	预测率 High
Low	0.926	0.074
High	0.316	0.684

误分类率 0.1429

实际分类 Y 二值型	预测计数 Low	预测计数 High
Low	88	7
High	12	26

“模型汇总”报表显示，训练集和验证集的误分类率非常相似。这是一个很好的迹象，表明模型并没有过度拟合数据。混淆矩阵提供了关于模型误分类的观测类型的详细信息。在混淆矩阵中，左上角显示：该模型在大多数情况下正确分类了 Low 响应 （训练集中为 92%，验证集中为 89.5%）。然而，正确分类的 “High” 响应较少（训练集中为 57.8%，验证集中为 65.8%）。因此，大多数误分类是被误分类为 Low 的 High 响应。

JMP PRO 启动“支持向量机”平台

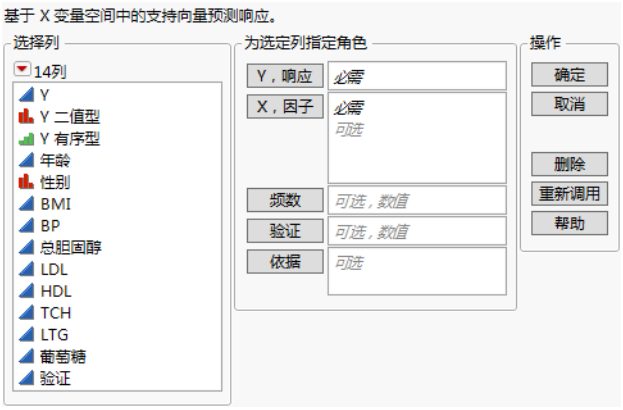
通过选择**分析 > 预测建模 > 支持向量机**启动 “支持向量机 (SVM)”平台。

启动 SVM 平台需要执行两个步骤。第一步，在 SVM 启动窗口中输入变量。第二步，在“模型启动”控制面板中指定选项。

JMP PRO “支持向量机”启动窗口

使用 SVM 启动窗口指定响应和模型因子。若使用“验证”和“依据”列，也是在启动窗口中指定这些列。

图 9.4 “支持向量机”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“支持向量机”启动窗口提供以下选项：

Y，响应 您想要分析的响应变量。

X，因子 预测变量。

频数 一列，该列的数值为分析中的每一行都分配一个频数。

验证 用于定义验证集的数值列。请参见第 144 页的 “验证方法”。若在 “选择列”列表中没有选择任何列的情况下点击 “验证”按钮，您可以向数据表添加一个验证列。有关 “生成验证列”实用工具的详细信息，请参见第 175 页的 “生成验证列”一章。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据”变量，将为 “依据”变量水平的每种可能组合生成单独的报表。

JMP PRO “模型启动”控制面板

当您在启动窗口中点击 “确定”时，会出现 “支持向量机”报表窗口，该窗口显示用于拟合模型的 “模型启动”控制面板。使用 “模型启动”控制面板指定核函数和相关参数值以及验证方法。

图 9.5 “模型启动”控制面板

“模型启动”控制面板包含以下选项：

核函数 指定模型中使用的核函数。从以下核函数中进行选择：

径向基函数 选择径向基函数内核，以创建用于分离各个类的非线性超平面。

- 成本参数是在训练集中与观测误分类相关联的罚值。成本较高的参数实现的算法不太可能误分类训练集中的点，而成本较低的参数生成更宽的范围。成本参数必须大于 0，且默认值为 1。
- Gamma 参数是核函数中的参数。该参数确定决策线的曲率；Gamma 值越高，表示曲率越大。非线性决策线提供了更灵活的拟合，但曲率过大会导致过度拟合。Gamma 参数必须大于 0，且默认值为 1/（预测变量数）。

线性 选择线性核函数以创建用于分离各个类的线性超平面。

- 成本参数是在训练集中与观测误分类相关联的罚值。成本较高的参数实现的算法不太可能误分类训练集中的点，而成本较低的参数生成更宽的范围。成本参数必须大于 0，且默认值为 1。

注意：若指定的参数值超出范围，则使用默认值。

提示：要查找最佳拟合模型，需拟合一系列核函数和参数值并使用 “模型比较”报表。

调节设计 支持您拟合指定内核的一系列参数值。具有最大 R 方和最小误分类率或 RASE 的模型会在“模型比较”报表中标识出来。选择“调节设计”后，必须为参数指定最小值和最大值。默认值基于您的数据，且最小值必须大于零。您还必须为“运行次数”指定一个值。SVM 平台在由最小值和最大值确定的参数值网格上拟合许多模型。

验证方法 指定模型验证方法。首次点击执行按钮后，将使用指定的验证方法拟合第一个 SVM 模型。“验证方法”随后用于 SVM 窗口内的所有 SVM 模型拟合。这样可以确保报表窗口中的所有模型都使用相同的验证方法和验证集进行拟合。

保留 将原始数据随机划分为训练集和验证集。您可以指定要用作验证集（保留）的数据占原始数据的比例。

K 重 （仅当 *Y* 为连续型或名义型时才可用。）将原始数据随机划分为 *K* 个子集。这 *K* 个子集依次对基于剩余 k-1 个子集的数据构建的模型进行验证，总共拟合 *K* 个模型。若 *Y* 为连续型，具有最佳验证 RASE 统计量的模型将被选为最终模型。若 *Y* 为名义型，具有最佳验证误分类率的模型将被选为最终模型。

验证列 （仅当在启动窗口中指定了验证列时才可用。）使用指定的验证列中的值将数据分成多个部分。该列的值决定如何拆分数据，以及使用什么方法进行验证：

- 若有两个值，较小的值定义训练集，较大的值定义验证集。
- 若有三个值，这些值按大小递增的顺序定义训练集、验证集和测试集。
- 若验证列有三个以上水平，则使用“验证列 K 折”。

SVM 平台使用验证列来训练和评估模型，除非使用了“调节设计”。若选定“调节设计”选项，SVM 平台使用验证列来训练和调优模型，或者训练、调优和评估模型。有关验证的详细信息，请参见“统计详细信息”附录中第 469 页的“JMP 建模中的验证”。

注意：若验证列未能导致有效地分割数据，则改用“保留”验证方法。

验证列 K 折 （仅当“Y，响应”列恰好有两个水平并且在启动窗口中指定了验证列时才可用。）使用指定验证列中的值将数据划分为 *K* 个集，其中 *K* 是列中唯一值的数目。然后，执行 K 重验证。

无 未使用验证。

执行 拟合指定的 SVM 模型并显示模型报表。

注意：若有大型数据表，则会为拟合数据的每个模型显示一个进度条。模型拟合的总数为 $k!/2(k\text{-}2)!$，其中 *k* 是响应变量的水平数。每个进度条都有一个**接受当前估计值**按钮。若您想要提前停止拟合算法并接受当前估计值，点击该按钮。由于预测计算是在点击该按钮后执行的，因此可能需要一些时间才能显示报表。

JMP PRO “支持向量机”报表

点击“模型启动”控制面板中的**执行**后，“模型比较”报表出现在控制面板上方，同时控制面板下方会显示一个模型报表。模型报表按顺序编号。每次拟合新模型时，都会将其添加到“模型比较”报表中，并将新模型报表添加到窗口中。请参见第 145 页的“‘模型比较’报表”和第 146 页的“‘支持向量机模型’报表”。点击列标题可按“模型比较”报表中的任何列对模型进行排序。第一次点击列标题按升序排序；第二次点击按降序排序。

JMP PRO “模型比较”报表

“模型比较”报表包含以下信息：

显示 一个复选框，指示是否应在报表窗口中显示相应的拟合的 “支持向量机模型”报表。

方法 模型编号。

核函数 相应模型的核函数。

成本 相应模型的成本参数的值。

Gamma 相应模型的 gamma 参数的值。若模型使用线性内核，则该值缺失。

 # SV 相应模型中使用的支持向量数。

训练误分类率 （仅在响应为分类响应时才显示。）训练集中的观测的误分类率。该比率基于平台计算的分类决策规则。

验证误分类率 （仅在响应为分类响应并且使用验证时才显示。）验证集中的观测的误分类率。该比率基于平台计算的分类决策规则。

测试误分类率 （仅在响应为分类响应并且使用验证列指定了测试集时才显示。）测试集中的观测的误分类率。该比率基于平台计算的分类决策规则。

训练 RASE （仅在响应为连续响应时才显示。）训练集中观测的均方预测误差的平方根（均方根误差）。请参见第 146 页的 “RASE”。

验证 RASE （仅在响应为连续响应并且使用验证时才显示。）验证集中观测的均方预测误差的平方根 （均方根误差）。请参见第 146 页的 “RASE”。

检验 RASE （仅在响应为连续响应并且使用验证列指定了测试集时才显示。）测试集中观测的均方预测误差的平方根 （均方根误差）。请参见第 146 页的 “RASE”。

验证广义 R 方 （仅当使用验证时才显示。）验证集中观测的广义 R 方值。请参见第 146 页的 “拟合详细信息”。

概率阈值 （仅在响应为二值响应并且使用验证时才显示。）由平台计算的分类决策规则确定的截止概率。若某个观测的预测概率超过该值，则该观测将分类到目标水平中。

提示：可以使用 “混淆矩阵”报表中的 “设置概率阈值”选项更改该值。请参见第 147 页的 “混淆矩阵”。

条件验证误分类率 （仅在响应为二值响应并且使用验证时才显示。）在概率阈值条件下，验证集中的观测的误分类率。

最佳 指示哪种模型拟合具有最小误分类率。若在没有测试集的情况下使用验证，则这是验证误分类率最小的模型。若与测试集一起使用验证，则这是测试误分类率最小的模型。

模型比较图

显示参数值网格上验证集的模型性能等高线图。仅当在“模型启动控制面板”中指定了“调节设计”时，该报表才可用。

JMP PRO “支持向量机模型”报表

响应特征图 （仅当响应为分类响应时才可用。）给出分类模型的可视化表示。图上的点是实际的数据观测，仅当恰好有两个连续变量时才会默认处于启用状态。对于绘制的两个变量，阴影轮廓表示由其余模型因子的固定值确定的预测空间的平面。预测基于平台计算的分类决策规则。固定值控件位于图上方。使用滑块或数字框更改因子的固定值时，绘制的变量的预测空间将自动更新。也可以使用每个轴上的红色小三角更改绘图轴以显示任何连续因子。

“响应特征图”红色小三角菜单包含以下选项：

网格密度 确定阴影轮廓下预测网格的精细度。较高的网格密度提供较平滑的决策线。

显示点 显示或隐藏图中的点。仅当有两个变量时才会默认处于启用状态。

模型汇总 给出响应列的名称、验证方法以及模型拟合中使用的核函数的类型。“模型汇总”表还包含有关训练集、验证集和测试集的模型拟合的信息。报告每个集的观测数和支持向量数。若响应为分类响应，则为每个集报告误分类率。误分类率是模型误分类的观测比例。该比率通过用误分类数除以观测总数计算得出。若响应为连续响应，则为每个集报告 RASE 和 R 方值。

估计详细信息 给出模型中使用的参数值。

拟合详细信息 （仅当响应为分类响应时才可用。）提供训练集以及验证集和测试集（若已指定）的以下统计量：

熵 R 方 比较拟合模型和恒定概率模型的对数似然的一种拟合测度。熵 R 方的范围介于 0 到 1 之间，其中接近 1 的值指示拟合更佳。请参见“统计详细信息”一章中第 469 页的“熵 R 方”。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。值越接近 1 指示拟合效果越好。

-Log p 均值 -log(p) 的平均值，其中 p 是与发生的事件有关的拟合概率。值越小指示拟合效果越好。

RASE 均方预测误差的平方根（平均平方根误差）。RASE 计算如下，其中**源**指示训练集、验证集或测试集。

$$\text{RASE}_{\text{源}} = \sqrt{\frac{SSE_{\text{源}}}{n}}$$

绝对偏差的均值 响应的真实值与预测值的差值绝对值的平均值。差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。值越小指示拟合效果越好。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。

注意：“拟合详细信息”报表中的误分类率可能与“混淆矩阵”报表中的误分类率不一致。当响应为二值响应时，“拟合详细信息”报表中的比率使用概率截止值 0.5，但是“混淆矩阵”报表中的比率使用概率阈值作为截止值。

数目 观测数。

混淆矩阵 （仅当响应为分类响应时才可用。）为模型中指定的每个集（训练集、验证集和测试集）显示一个混淆矩阵。混淆矩阵是实际响应和预测响应的双向分类。使用混淆矩阵和误分类率来评估您的模型。

混淆矩阵和误分类率使用概率阈值框中的值作为截止值。默认情况下，该值基于平台计算的分类决策规则。可以通过拖动滑块或在“概率阈值”旁边的框中输入新值来更改截止值。若更改概率阈值，混淆矩阵和误分类率将自动更新。“模型比较”报表中的概率阈值和条件验证误分类率列也随之更新。

“预测值 - 实际值”图 （仅当响应为连续响应时才可用。）标绘 “预测响应 - 实际响应”图。对角线为 Y = X 线。点越靠近这条线，模型拟合数据的效果就越好。使用验证时，为训练、验证和测试集 （若指定）都显示图。

JMP PRO “支持向量机”平台选项

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO “支持向量机模型”报表选项

每个 “支持向量机模型”报表都包含以下选项：

响应特征图 （仅当响应为分类响应时才可用。）显示或隐藏 “响应特征图”。

支持向量系数 显示或隐藏支持向量系数表。选择表中的行以突出显示原始数据表和 “响应特征图”中的支持向量。

混淆矩阵 （仅当响应为分类响应时才可用。）显示或隐藏混淆矩阵。请参见第 147 页的“混淆矩阵”。

ROC 曲线 （仅当响应为分类响应时才可用。）显示或隐藏 ROC 曲线。若使用验证，则在显示训练集的 ROC 曲线之外，还会显示验证集和测试集（若指定）的 ROC 曲线。详细信息，请参见“分割模型”一章中第 65 页的“ROC 曲线”。

提升曲线 （仅当响应为分类响应时才可用。）显示或隐藏提升曲线。若使用验证，则在显示训练集的提升曲线之外，还会显示验证集和测试集（若指定）的提升曲线。详细信息，请参见“分割模型”一章中第 66 页的“提升曲线”。

标绘“预测值 - 实际值”图 （仅当响应为连续响应时才可用。）显示或隐藏预测值与实际值之间的关系图。

标绘“预测值 - 残差”图 （仅当响应为连续响应时才可用。）显示或隐藏预测值与残差之间的关系图。

刻画器 显示或隐藏“预测刻画器”。每个响应水平在“预测刻画器”中分别由单独的行来表示。有关红色小三角菜单中选项的详细信息，请参见《刻画器指南》。

等高线刻画器 （仅当模型包含不止一个连续因子时才可用。）显示或隐藏“等高线刻画器”。有关红色小三角菜单中选项的详细信息，请参见《刻画器指南》。

曲面刻画器 （仅当模型包含至少一个连续因子时才可用。）显示或隐藏曲面刻画器。有关红色小三角菜单中选项的详细信息，请参见《刻画器指南》。

保存预测值 将基于阈值概率截止值的预测分类保存到数据表中名为**预测 <Y，响应 >** 的列。

保存预测公式 将基于阈值概率截止值的预测公式保存到数据表中。若响应为二值响应，则向数据表中添加一个预测公式列。若响应有两个以上的水平，则除了用于最终预测的公式列外，还将为每个成对模型向数据表中添加一个预测公式列。

发布预测公式 基于阈值概率截止值创建概率公式并将它们保存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将打开“公式存储库”窗口。请参见第 185 页的“公式存储库”一章。

保存概率 将每个响应水平的概率保存为数据表中的单独一列，以及一个名为**最可能的 <Y，响应 >** 列。

保存概率公式 （仅当响应为二值响应时才可用。）将以下公式列保存至数据表：

决策值 一个包含决策值公式的隐藏列。

概率 两个概率公式列，每个二值响应水平对应一列。

最可能的 较高概率值所基于的预测分类。

阈值预测 基于从混淆矩阵得到的阈值概率截止值的预测分类。默认情况下，该值基于平台计算的分类决策规则。若阈值更改为 0.5，则**阈值预测**列不会保存到数据表中，因为它相当于**最可能的**列。

发布概率公式 （仅当响应为二值响应时才可用。）创建概率公式并将它们保存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将打开“公式存储库”窗口。请参见第 185 页的“公式存储库”一章。

保存验证 （仅当使用 “保留” 或 “K 重” 验证方法时可用。）在数据表中创建新列，用于标识在训练集和验证集中分别使用了哪些行。

删除拟合 删除整个模型报表。

SVM 平台的更多示例

连续响应的支持向量回归示例

支持向量回归 (SVR) 模型使用 SVM 算法来预测连续响应数据。本例使用第 139 页的 “支持向量机示例” 中所用的关于糖尿病患者的相同医疗数据。您现在有了一个连续测度，而不是糖尿病发展的二值型测度。值越高，疾病进展越快。改变径向基函数的成本参数以便找到最佳拟合模型。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 选择**分析** > **预测建模** > **支持向量机**。
3. 选择 Y 并点击 **Y，响应**。
4. 从**年龄**一直选到**葡萄糖**，然后点击 **X，因子**。
5. 选择**验证**并点击**验证**。
6. 点击**确定**。
7. 在 “模型启动” 控制面板中，检查核函数是否为成本参数为 1 且 Gamma 参数为 0.1 的 “径向基函数”。
8. 点击**执行**。
9. 点击 “模型启动” 旁边的灰色小三角以打开 “模型启动” 控制面板。
10. 将成本参数改为 0.1。
11. 点击**执行**。
12. 点击 “模型启动” 旁边的灰色小三角以打开 “模型启动” 控制面板。
13. 将成本参数改为 2。
14. 点击**执行**。

图 9.6 “模型比较” 报表

模型比较

显示	方法	核函数	成本	Gamma	# SV	训练 RASE	验证 RASE	验证广义 R 方	最佳
☑	模型 1	径向基函数	1	0.1	266	43.8147	58.3285	0.42552	
☑	模型 2	径向基函数	0.1	0.1	280	52.5804	57.5025	0.44167	最小 RASE 最大 R 方
☑	模型 3	径向基函数	2	0.1	265	40.553	60.7801	0.37621	

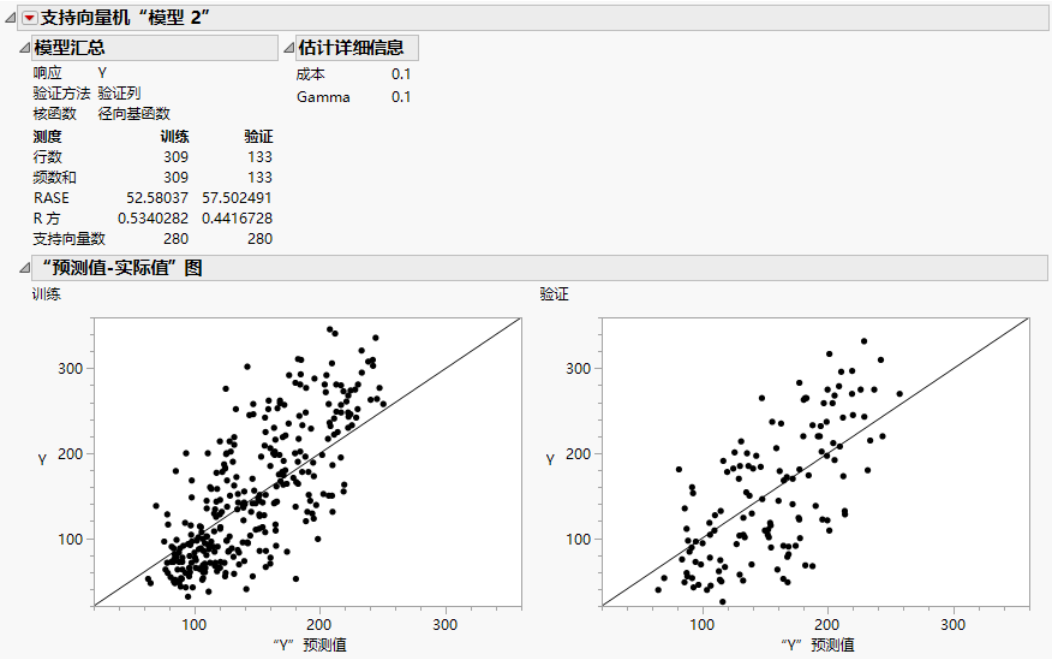
模型比较图

验证列: 验证

“模型比较”报表包含使用不同成本参数的模型结果。回想一下，较高的成本参数实现的算法不太可能误分类某个点，而较低的成本参数实现的算法更灵活。在这种情况下，更灵活的模型（成本参数等于 0.1）会生成最佳模型拟合。

15. 在“模型比较”报表的“显示”列中，取消选择“模型 1”和“模型 3”。

图 9.7 连续响应的模型报表



连续响应的模型报表包含“模型汇总”、“估计详细信息”和“预测值 - 实际值”图。在“模型汇总”报表中，训练集的 RASE 是 52.58，验证集的 RASE 是 57.50。

第 10 章

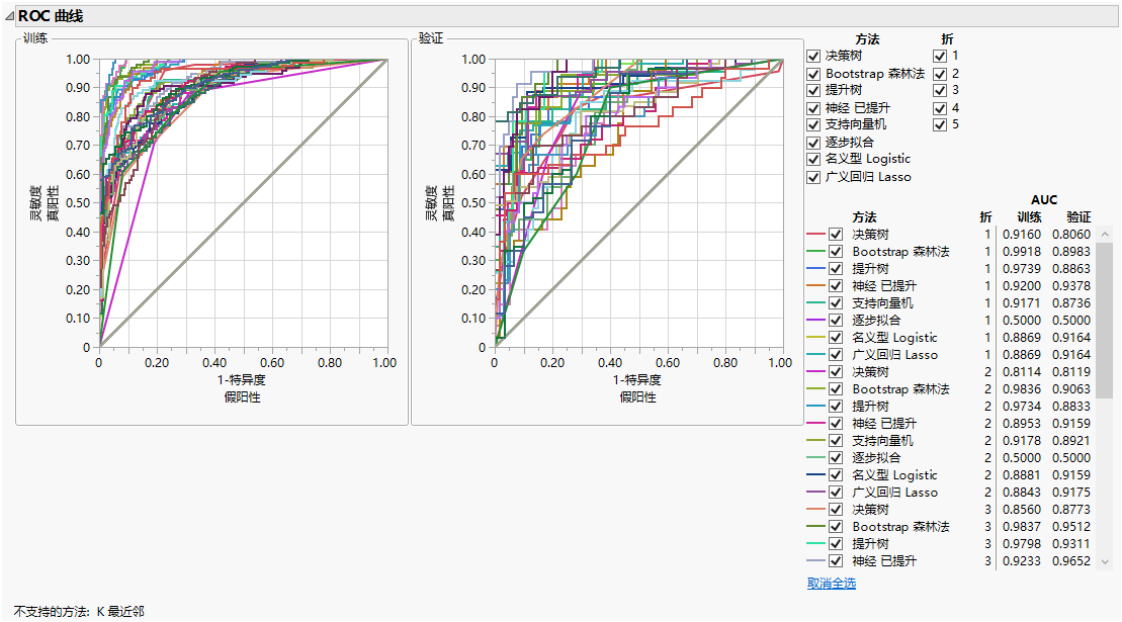
JMP PRO 模型筛选

拟合和比较多重预测模型

仅在 JMP Pro 中提供 “模型筛选”平台。

使用 JMP Pro 中的“模型筛选”平台，您可以从一个启动窗口运行多个预测建模平台并集合来自不同方法的汇总。这有助于您选择性能最佳的模型，而无需启动每个单独的模型平台。您还可以轻松地从“模型筛选”报表窗口为任何模型启动单个平台报表，以便进一步细化。

图 10.1 “模型筛选”报表



目录

连续响应的模型筛选示例 153
启动“模型筛选”平台 154
“模型筛选”报表 157
“跨越折汇总”报表 157
训练、验证和测试拟合测度 158
“模型筛选”平台选项 159
“决策阈值”报表 160

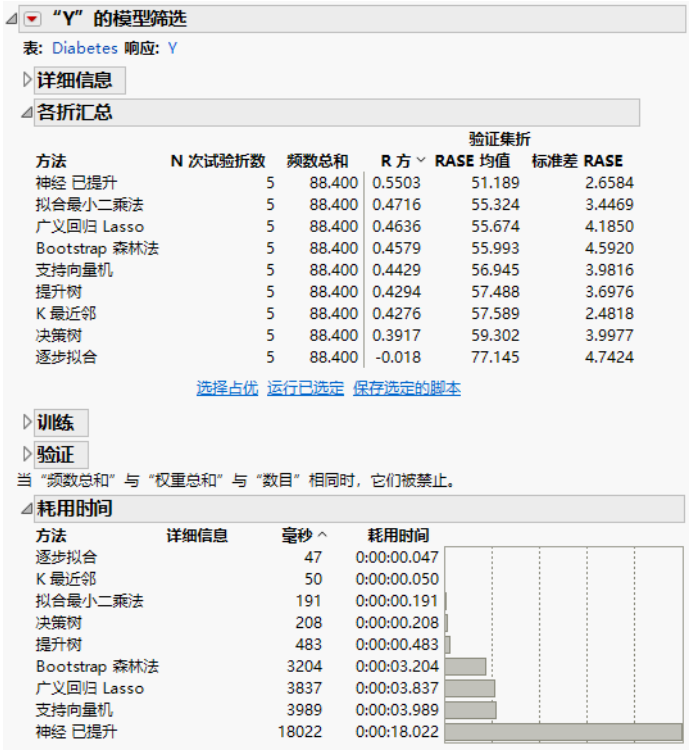
JMP PRO 连续响应的模型筛选示例

您有 442 名糖尿病患者的基准医疗数据，还有在每名患者首次就诊一年后得到的糖尿病疾病发展的连续测度。值越高，疾病进展越快。您需要评估各种模型预测糖尿病进展的能力。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 选择**分析** > **预测建模** > **模型筛选**。
3. 选择 Y 并点击 **Y，响应**。
4. 从**年龄**一直选到**葡萄糖**，然后点击 **X，因子**。
5. 在 “已折叠交叉验证”部分中，选择 **K 折交叉验证**旁边的框。
6. （可选）在 “选项”部分中的**设置随机种子**旁边键入 42920。

   使用随机种子以匹配示例输出。
7. 点击**确定**。
8. 点击 “‘Y’的模型筛选”旁边的红色小三角并选择**可选报表** > **耗用时间**。

图 10.2 连续响应的模型报表



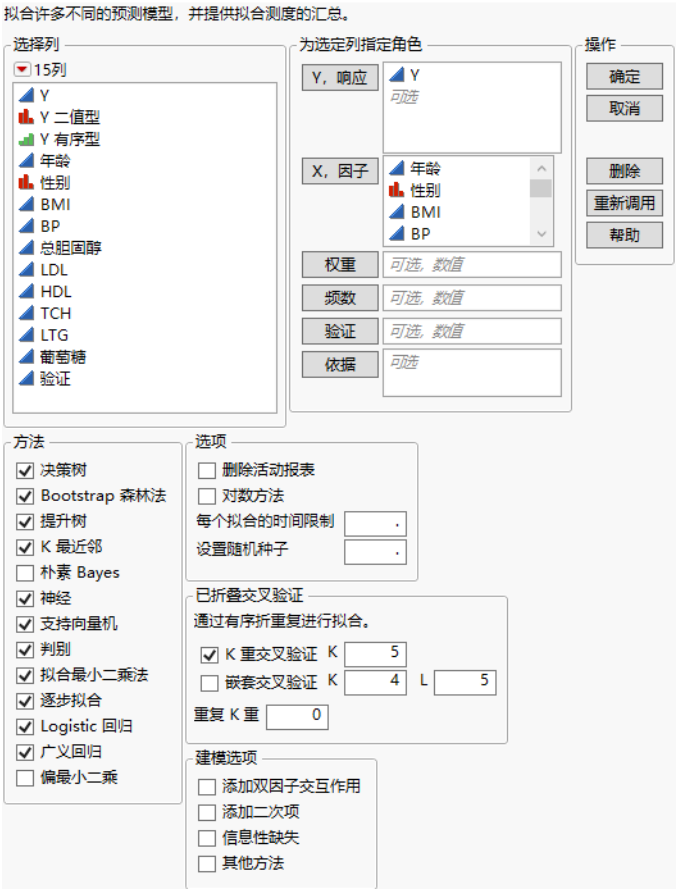
基于跨越折计算的平均验证 R 方的最佳效果模型是“提升神经”。“提升神经”的验证 R 方平均值为 0.5503。“耗用时间”报表显示“提升神经”模型的拟合时间也最长。在这种情况下，经过的时间只有 5 秒，但是对于更大的数据集或更复杂的模型，这些信息可能很重要。

提示：要查看各个模型的性能量度，请点击“验证”旁边的灰色展开图标。

JMP PRO 启动“模型筛选”平台

要启动 “模型筛选”平台，请选择**分析 > 预测建模 > 模型筛选**。

图 10.3 “模型筛选”启动窗口



有关 “选择列”红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。

Y，响应 您想要分析的一个或多个响应变量。

X，因子 预测变量。

权重 （不适用于 “K 最近邻”、“支持向量机” 或 “神经” 等建模平台。）一列，该列的数值为分析中的每一行都分配一个权重。

频数 （不适用于 “K 最近邻” 建模平台。）一列，列中的数值为分析中的每行分配一个频数。

验证 （若在启动窗口中选择了任何 “交叉验证” 选项，则不适用。）用于定义验证集的数值列。若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见第 175 页的 “生成验证列”一章。

注意：若指定的验证列具有三个以上的水平，则该列用于执行 “K 折” 交叉验证。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的报表。

方法 支持您选择所需的建模平台。默认情况下，拟合的建模平台包括 “决策树 （分割）”、“Bootstrap 森林法”、“提升树”、“K 最近邻”、“神经”、“支持向量机”、“判别”、“拟合最小二乘法”、“逐步拟合”、“Logistic 回归” 和 “广义回归”。还提供 “朴素 Bayes”、“偏最小二乘” 和 “XGBoost” 等平台。

注意：

- JMP 不支持 XGBoost，只有在安装了 XGBoost 插件后才可用。有关 XGBoost 的详细信息，请参见 https://community.jmp.com。
- “决策树（分割）”、“判别” 和 “偏最小二乘” 都需要某种类型的验证集来拟合模型。
- 若一个验证集中的观测数少于 20 个，则无法拟合 “决策树（分割）” 模型。
- 建模平台在模型拟合中使用默认选项。您可以尝试通过直接调用平台并选择不同选项来改善拟合效果，从而绕过默认选项生成的效果。

选项

提供其他选项。

删除活动报表 不包括 “模型筛选” 报表窗口中的单个模型平台报表。

提示：选择该选项可在许多方法和拟合出现较大问题时释放内存。

对数方法 在每次调用拟合平台时，向日志中写入进度消息。

每个拟合的时间限制 指定每个拟合的时间限值（以秒为单位）。对于支持提前停止的平台，可以提供至该点为止的最佳估计。对于不支持提前停止的平台，不提供任何结果。

设置随机种子 设置用于模型拟合例程的任何随机成分的随机种子。这样您就可以重新运行平台并获得相同的模型拟合。

已折叠交叉验证

提供用于多种交叉验证的选项。

K 折交叉验证 将数据随机分为 *K* 个部分或折。每个模型拟合数据 *K* 次，每次都有不同的折保留为交叉验证集。总共拟合 *K* 个模型。 *K* 的默认值为 5。

– *K* 指定“K 折交叉验证”的折数。默认值为 5，且 *K* 必须大于 1。

嵌套交叉验证 将数据划分为嵌套的多折以进行交叉验证。首先，数据划分为 $k = 1, \ldots$， *K* 等分或 K 折。对于每折，第 *k* 折用作测试集，其余数据进一步划分为 *L* 等分。这些 *L* 子部分称为**内折**。然后，一个模型拟合数据 *L* 次，每次都有不同的内部折保留为交叉验证集。*L* 个模型之后使用第 *k* 折作为共同测试数据集。总之，一共拟合 *K***L* 个模型。 *K* 的默认值为 4，且 *L* 的默认值为 5。

例如，设置 K=2 且 L =3。数据最初分为两折。第一折作为一个测试集保留，第二折分为 3 个内折。拟合三个数据模型，每次都有一个不同的内部折保留为交叉验证集。然后，在第一折上检验全部三个模型。

接着将第二折作为一个测试集保留，第一折分为 3 个内折。拟合三个数据模型，每次都有一个不同的内部折保留为交叉验证集。然后，在第二折上检验全部三个模型。

– *K* 指定“嵌套交叉验证”的折数。默认值为 4，且 *K* 必须大于 1。

– *L* 指定“嵌套交叉验证”的内折数。默认值为 5，且 *K* 必须大于 1。

注意：若同时选定 “K 折交叉验证”和 “嵌套交叉验证”，则执行 “嵌套交叉验证”。

重复 K 折 指定 “K 折交叉验证”或 “嵌套交叉验证”过程重复的次数。

建模选项

提供用于平台建模的其他选项。

添加双因子交互作用 向线性模型添加双因子交互作用效应。

添加二次项 将连续变量平方的效应添加到线性建模。

信息性缺失 为所有平台启用信息性缺失。

其他方法 在“广义回归”平台中调用其他几种方法，如、“岭”、“弹性网络”和“Lasso”。请参见 《拟合线性模型》。

警告：这会导致其他模型拟合。

点击“确定”后，拟合指定模型并显示一组进度条。上部进度条报告所有拟合的进度。下部进度条报告当前单个模型拟合的进度。您可以停止下部进度条实施提前停止，而上部进度条将继续运行。

JMP PRO “模型筛选”报表

“模型筛选”报表的内容取决于实施的交叉验证类型和响应变量的建模类型。在“模型筛选”报表的顶部，显示了数据表的名称和响应变量的列名。若指定了验证列，则还会显示该列的名称。

默认情况下，接下来将显示“详细信息”报表。该报表包含每折和试验的所有模型的单个模型报表。这些报表是每个平台的实际输出，并且包含所有报表和命令功能。例如，若您已对最佳模型作出决定，则可以转到“详细信息”报表中相应的平台报表，并保存所需的预测公式列。您可以通过在启动窗口中选择“删除活动报表”选项来隐藏“详细信息”报表。

若在启动窗口中指定了“K 折交叉验证”选项或“嵌套交叉验证”选项，则将显示“跨越折汇总”报表。该报表汇总了跨越测试折的平均拟合测度。

为训练集、验证集和测试集（如适用）的每个单独的模型拟合显示拟合度量。若通过在启动窗口中指定验证列来使用交叉验证，则报表中仅显示单个模型拟合测度。

“跨越折汇总”报表

“跨越折汇总”报表包含跨越折的拟合测度汇总，若指定了 “重复 K 折”则还包含针对试验的拟合测度汇总。若指定了 “K 折交叉验证”选项，则针对各个验证集汇总拟合测度。若指定了“嵌套 K 折交叉验证”选项，则针对各个测试集汇总拟合测度。该报表还包含具有以下列的表：

方法 用于拟合模型的方法的名称。

N 次试验折数 对所有折和试验 （若适用）拟合的模型总数。

频数总和 验证或测试集中观测的平均数。测试集用于估计模型性能。

R 方 所有验证或测试集折的 R 方均值。若响应为分类响应，该列包含 “熵 R 方”。

RASE 均值 所有验证或测试集折的 RASE （均方根误差）的均值。

标准差 RASE 跨越所有验证或测试集的 RASE 的标准差。

AUC 均值 （仅可用于分类响应。）跨越所有验证或测试集的 ROC 曲线 (AUC) 下的均值区域。

均值 MR （仅可用于分类响应。）所有验证或测试集的误分类率 (MR) 均值。

表下方提供了以下选项：

选择占优 根据模型拟合准则的组合，选择优于或等同于其他所有模型的每个模型。这也称为选择 Pareto 边界。对于连续响应，在确定占优模型时考虑了 “R 方”和 “频数总和”。对于分类响应，在确定占优模型时考虑了“熵 R 方”、“误分类率”、“AUC”和“频数总和”。

运行已选定 为每个选定行运行单个模型。点击 “运行已选定”后，系统会提示您为每个选定行选择一个验证列。这将指定用于拟合模型的验证集。

保存选定的脚本 将每个选定行的模型脚本保存至脚本窗口。点击“保存选定的脚本”后，系统会提示您为每个选定行选择一个验证列。这将指定保存在脚本中的验证列。

训练、验证和测试拟合测度

对于指定的每个模型数据集，都有一个拟合测度报表。模型数据集可以是训练集、训练和验证集，或是训练、验证和测试集。每个报表都包含具有以下列的表：

方法 用于拟合模型的方法的名称。

数目 集中的观测数。

权重和 权重总和。

R 方 （仅可用于连续响应。）拟合模型的 R 方值。

熵 R 方 （仅可用于分类响应。）比较拟合模型和恒定概率模型的对数似然的一种拟合测度。熵 R 方的范围介于 0 到 1 之间，其中接近 1 的值指示拟合更佳。请参见“统计详细信息”一章中第 469 页的 “熵 R 方”。

误分类率 （仅可用于分类响应。）模型误分类的观测比例。值越小指示拟合效果越好。

注意：在这些表中，误分类率总是使用概率阈值 0.5 来计算。

AUC （仅可用于分类响应。） ROC 曲线下面积。值越接近 1 指示拟合效果越好。

RASE 均方预测误差的平方根（平均平方根误差）。RASE 计算如下，其中**源**指示训练集、验证集或测试集。

$$\text{RASE}_{\text{源}} = \sqrt{\frac{SSE_{\text{源}}}{n}}$$

广义 R 方 （仅可用于分类响应。）可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。值为 1 表示完美模型，值为 0 表示并不比常数模型好的模型。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。

折 （仅当在启动窗口中指定了“K 折交叉验证”选项或“嵌套交叉验证”选项时才可用。）标识为该行中的模型拟合保留的折。

内折 （仅当在启动窗口中指定了“嵌套交叉验证”选项时才可用。）标识为该行中的模型拟合保留的内部折。

试验 （仅当在启动窗口中指定了 “重复 K 折”选项时才可用。 ）标识该行中的模型拟合的试验数目。

在每个表下提供了以下选项：

选择占优 根据模型拟合准则的组合，选择优于或等同于其他所有模型的每个模型。对于连续响应，在确定占优模型时考虑了 “R 方”和 “频数总和”。对于分类响应，在确定占优模型时考虑了 “熵 R 方”、“误分类率”、“AUC”和 “频数总和”。

运行已选定 为每个选定行运行单个模型。

警告：若指定了任何形式的 “K 折交叉验证”，“运行已选定”将针对最后一折确定的训练集和验证集运行选定的模型。

保存选定的脚本 将每个选定行的模型脚本保存至脚本窗口。

警告：若指定了任何形式的 “K 折交叉验证”，“保存选定的脚本”将为最后一折确定的训练集和验证集上运行的选定模型保存脚本。

JMP PRO “模型筛选”平台选项

标绘 “预测值 - 实际值”图 （仅可用于连续响应。）显示或隐藏叠加所有模型拟合的预测点 - 实际点的图报表。训练、验证和测试集分别对应不同图。该报表包含一个带有复选框的“方法”图例，这些复选框支持您指定基于方法、折叠、内部折和试验组合在图上显示哪些拟合。点击图例底部的 “取消全选”按钮可从图中删除所有模型拟合。

注意：

- 若从“训练”、“验证”或“测试”报表中选择模型子集，然后选择“标绘‘预测值-实际值’图”，则仅显示选定模型的图。
- 并非所有建模平台都支持“预测值-实际值”图。不受支持的平台列表显示在图下方。

ROC 曲线 （仅可用于分类响应。）显示或隐藏为所有模型拟合叠加 ROC 曲线的图报表。训练、验证和测试集分别对应不同图。该报表包含两个带有复选框的图例，这些图例支持您指定哪些拟合显示在图上。点击图例底部的“取消全选”按钮可从图中删除所有模型拟合。有关 ROC 曲线的详细信息，请参见 “分割模型”一章中第 65 页的 “ROC 曲线”。

注意：

- 若从“训练”、“验证”或“测试”报表中选择模型子集，然后选择“ROC 曲线”，则仅显示选定模型的图。
- 并非所有建模平台都支持 ROC 图。不受支持的平台列表显示在图下方。

显示收益 （仅可用于具有“收益矩阵”列属性的二值分类响应。）显示或隐藏每个模型的期望收益。

决策阈值 （仅可用于二值分类响应。）显示或隐藏用于训练、验证和测试集（若指定）的“决策阈值”报表。每个报表包含每个模型的拟合概率分布图、每个模型的混淆矩阵以及用于调整阈值和比较模型拟合的分类图。请参见第 160 页的 “决策阈值”报表”。

注意：若从“训练”、“验证”或“测试”报表中选择模型子集，然后选择“决策阈值”，则仅显示所选定模型的信息。

保存结果表 将“验证”报表中的信息保存到名为“模型筛选统计量验证集”的新数据表中。若有一个测试集，例如指定“嵌套交叉验证”选项时，“测试”报表中的信息也会保存到名为“模型筛选统计量测试集”的新数据表中。

保存 K 重结果表 （仅当在启动窗口中指定了“交叉验证”选项时才可用。）将“跨越折汇总”报表中的信息保存到名为“K 折结果”的新数据表中。

可选报表 显示其他报表选项的子菜单。

耗用时间 显示或隐藏包含拟合每种模型所花费总运行时间的报表。

预测变量的基数 显示或隐藏每个分类预测变量的水平数以及线性模型拟合中使用了多少参数的报表。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“决策阈值”报表

“决策阈值”报表支持您探索二值分类模型的阈值。对于验证方法指定的每个模型数据集，都有一个“决策阈值”报表。模型数据集可以是训练集、训练和验证集，或是训练、验证和测试集。每个“决策阈值”报表都包含拟合概率分布图、分类条形图和混淆矩阵。所有这些都按照模型适合，折、试验和分类水平来排列。该报表还包含一个关于分类准确性测度的选项卡式部分和一个用于设置收益矩阵的选项。该报表随着您调整概率阈值而更新。

拟合概率的分布

拟合概率的分布或模型得分支持您查看每个单独的模型拟合如何在两个类之间进行区分。图上的垂直线表示概率阈值，该阈值确定每个观测的分类。默认情况下，概率阈值为 0.5。可以通过拖动垂直线或点击概率阈值并输入新值来更改概率阈值。这将更改整个“决策阈值”报表中的概率阈值。概率阈值的值必须介于 0 和 1 之间。

分类计数

分类条形图显示当前阈值下响应变量的每个水平的分类计数。绿条表示分类正确的观测；红条表示分类错误的观测。

混淆矩阵

混淆矩阵（也称为列联表）显示了每个单独的模型拟合的实际和预测响应的双向分类。同时还显示混淆率矩阵。比率等于混淆矩阵中的值除以行合计。

分类准确度测度

阈值 - 错误分类 显示了按概率阈值划分的误分类计数图和按概率阈值划分的误分类率图。对于每个单独的模型拟合，每个图都包含两条曲线。低响应类别的曲线为实线，高响应类别的曲线为虚线。两条曲线在阈值处相交，从而为每个响应水平生成等误分类，计数或比率。每个图上还有一条垂直线，表示当前概率阈值。可以通过拖动垂直线来更改概率阈值。这将更改整个报表中的概率阈值。

对应部分 - 错误分类 显示对应部分 （按等级排名的得分） - 误分类计数或比率图。对于每个单独的模型拟合，每个图都包含两条曲线。低响应类别的曲线为实线，高响应类别的曲线为虚线。

阈值 - 正确分类 显示概率阈值 - 正确计数图和概率阈值 - 真实比率图。对于每个单独的模型拟合，每个图都包含两条曲线。低响应类别的曲线为虚线，高响应水平的曲线为实线。两条曲线在阈值处相交，从而为每个响应生成等正确分类，计数或比率。每个图上还有一条垂直线，表示当前概率阈值。可以通过拖动垂直线来更改概率阈值。这将更改整个报表中的概率阈值。

对应部分 - 正确分类 显示对应部分（按等级排名的得分）- 真实分类计数或比率图。对于每个单独的模型拟合，每个图都包含两条曲线。低响应类别的曲线为虚线，高响应类别的曲线为实线。

阈值 - 收益 （仅当指定收益矩阵时才可用。）显示概率阈值 - 平均收益图。每个单独的模型拟合都有一条曲线，还有一条表示当前概率阈值的垂直线。图旁边还会显示指定的收益矩阵。

量度 显示每个模型的分类准确性量度表。提供图例，用来描述如何计算每列中的量度。

设置收益矩阵

支持您将成本分配到不希望的结果，将收益分配到希望的结果。请参见“分割模型”一章中第 62 页的“指定收益矩阵”。若在“收益矩阵”窗口中更改概率阈值并点击“确定”，则会使用该值作为概率阈值更新“决策阈值”报表。

第 11 章

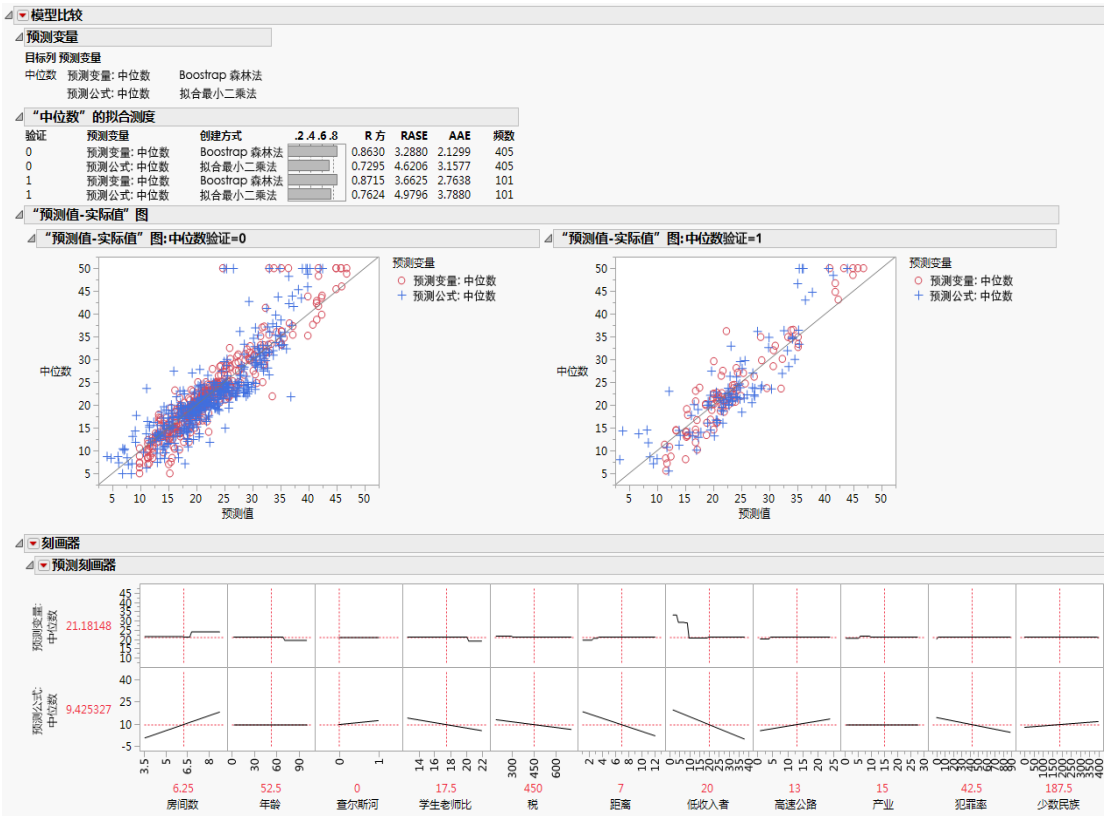
JMP PRO 模型比较

比较拟合模型的预测能力

仅在 JMP Pro 中提供 “模型比较”平台。

JMP Pro 中的“模型比较”平台支持您比较不同模型的预测能力。为每个模型都提供了拟合测度以及叠加诊断图。

图 11.1 模型比较示例



目录

模型比较的示例 165
启动“模型比较”平台 168
“模型比较”报表 169
“模型比较”平台选项 170
连续和分类响应 170
连续响应 170
分类响应 171
模型比较的更多示例 171

JMP PRO 模型比较的示例

本节提供了一个使用“模型比较”平台的示例。该示例使用人口统计数据来构建住房价格中位数的模型，同时比较回归模型和 Bootstrap 森林法模型。

首先选择**帮助** > **样本数据库**并打开 Boston Housing.jmp。

创建验证列

1. 选择**分析** > **预测建模** > **生成验证列**。
2. 不要选择启动窗口中的任何列。

   这指示该平台将创建简单随机验证列。
3. 点击**确定**。
4. 在 “新建列名”旁边的框中，键入 “创建验证”。
5. 在 “随机种子”旁边的框中，输入 1234。
6. 点击**执行**。

   随即创建新的 “验证列”。指定了 0 值的行是训练集。指定了 1 值的行是验证集。

创建回归模型并将预测公式保存至列

1. 选择**分析** > **拟合模型**。
2. 选择**中位数**并点击 Y。
3. 从**犯罪率**一直选择到**低收入者**，然后点击**添加**。
4. 选择**创建验证**并点击**验证**。
5. 在 “特质”列表中选择**逐步**。
6. 点击**运行**按钮。
7. 从 “停止规则”列表中选择 **p 值阈值**。
8. 点击**执行**按钮。
9. 点击**运行模型**按钮。

图 11.2 “拟合模型”报表

响应“中位数”

验证: 创建验证

效应汇总

拟合汇总

R 方	0.753984
调整 R 方	0.74663
均方根误差	4.787344
响应均值	23.02868
观测数（或权重和）	380

方差分析

源	自由度	平方和	均方	F 比
模型	11	25848.508	2349.86	102.5306
误差	368	8434.069	22.92	概率>F
校正总和	379	34282.577		<.0001*

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	38.500819	5.676983	6.78	<.0001*
犯罪率	-0.128236	0.03545	-3.62	0.0003*
区	0.0434609	0.015555	2.79	0.0055*
查尔斯河[0]	-1.215423	0.490413	-2.48	0.0136*
氧化氮	-15.38954	4.180964	-3.68	0.0003*
房间数	3.8274522	0.445164	8.60	<.0001*
距离	-1.501159	0.218407	-6.87	<.0001*
高速公路	0.3364509	0.073112	4.60	<.0001*
税	-0.012471	0.003826	-3.26	0.0012*
学生老师比	-0.994528	0.146259	-6.80	<.0001*
少数民族	0.0088057	0.003206	2.75	0.0063*
低收入者	-0.59558	0.054166	-11.00	<.0001*

10. 要将预测公式保存至列，请点击 “响应”红色小三角并选择**保存列 > 预测公式**。

创建 Bootstrap 森林法模型并将预测公式保存至列

1. 选择**分析 > 预测建模 > Bootstrap 森林法**。
2. 选择**中位数**并点击 **Y，响应**。
3. 从**犯罪率**一直选择到**低收入者**，然后点击 **X，因子**。
4. 选择**创建验证**并点击**验证**。
5. 点击**确定**。
6. 选中**提前停止**复选框。
7. 选中**在数个项数下进行多个拟合**复选框。
8. 在**随机种子**旁边的框中输入 617。
9. 点击**确定**。

图 11.3 Bootstrap 森林法模型

10. 要将预测公式保存至列，请点击“Bootstrap 森林法”红色小三角并选择**保存列 > 保存预测公式**。

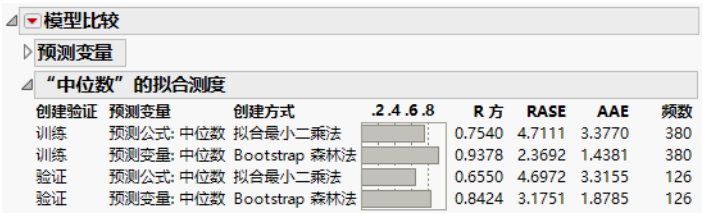
比较模型

1. 选择**分析 > 预测建模 > 模型比较**。
2. 选择这两个预测公式列并点击 **Y，预测变量**。
3. 选择**创建验证**并点击**分组**。

> **提示：**若未指定**组**列，JMP 会自动识别同一验证列已用于所有预测变量的情况，并提示您将其添加为分组变量。

4. 点击**确定**。

图 11.4 “模型比较”报表

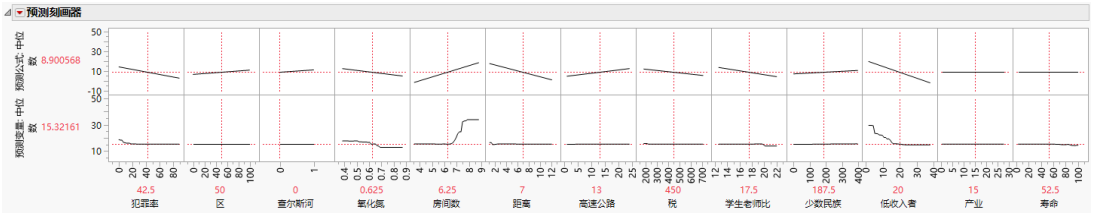


构建模型使用的是训练集中的行，所以“创建验证 = 训练”对应的 R 方统计量可能会有些虚高。在这种情况下，统计量不能代表模型的预测能力。这对于 Bootstrap 森林法模型尤其如此。

使用“创建验证 = 验证”的统计量比较模型。在这种情况下，Bootstrap 森林法模型的预测能力比回归模型好。

5. 点击 “模型比较” 旁边的红色小三角，然后选择 “刻画器”。

图 11.5 所有模型的 “预测刻画器”



预测刻画器支持您比较不同模型中每个因子的影响。当比较不同类型的模型（比如在此比较的回归模型和分割模型）时，该刻画器尤其有用。

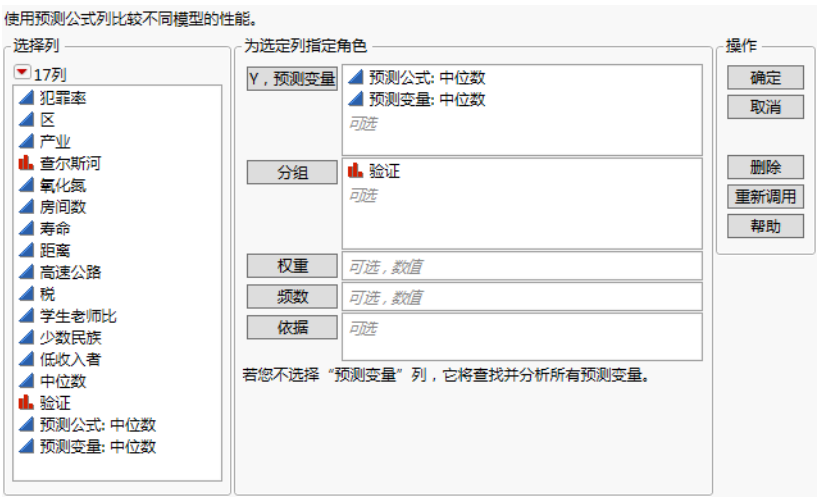
相关信息

- 《拟合线性模型》
- 第 47 页的 “分割模型” 一章

JMP PRO 启动 “模型比较” 平台

要启动 “模型比较” 平台，请选择**分析 > 预测建模 > 模型比较**。

图 11.6 “模型比较” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，预测变量 表示您要比较的模型的预测所在的列，这些列可以是公式列或者只是数据列。JMP 平台创建的预测公式列具有 “预测”或“响应概率”列属性。若您指定的列不包含上述属性之一，该平台会提示您指定哪一列要通过指定的 Y 列进行预测。

对于包含 *k* 个水平的分类响应，大多数模型拟合平台将 *k* 列保存至数据表，每一列预测一个水平的概率。需要将 *k* 个列全部指定为 **Y，预测变量**。对于未保存 *k* 个概率列的平台，可将包含预测响应水平的列指定为 **Y，预测变量**列。

若您不指定任何 **Y，预测变量**列，JMP 则使用数据表中具有 “预测”或“响应概率”列属性的预测公式列。

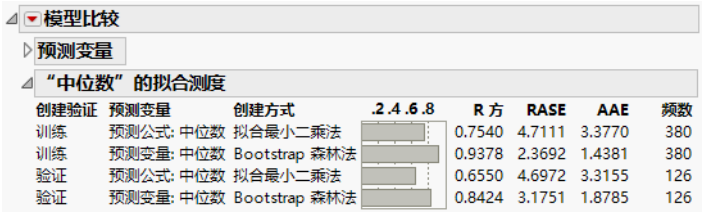
分组 用于将数据划分到不同的组以分别进行计算的列。若未指定**组**列，JMP 会自动识别同一验证列已用于所有预测变量的情况，并提示您将其添加为分组变量。

其他角色按钮是 JMP 平台中的常见按钮。请参见《使用 JMP》。

JMP PRO “模型比较”报表

图 11.7 显示了连续响应的初始 “模型比较”报表的示例。

图 11.7 最初的 “模型比较”报表


模型比较

预测变量

“中位数”的拟合测度

创建验证	预测变量	创建方式	.2 .4 .6 .8	R 方	RASE	AAE	频数
训练	预测公式: 中位数	拟合最小二乘法		0.7540	4.7111	3.3770	380
训练	预测变量: 中位数	Bootstrap 森林法		0.9378	2.3692	1.4381	380
验证	预测公式: 中位数	拟合最小二乘法		0.6550	4.6972	3.3155	126
验证	预测变量: 中位数	Bootstrap 森林法		0.8424	3.1751	1.8785	126

预测变量报表显示所有响应以及针对每个响应比较的所有模型。同时还列出创建预测变量列的拟合平台。

拟合测度报表显示每个模型的拟合测度。连续响应和分类响应的报表中会显示不同的列。

连续响应的“拟合测度”

R 方 *R* 方统计量。在不含缺失值的数据表中，“模型比较”报表与原始模型中的 *R* 方统计量保持一致。不过，若存在缺失值，*R* 方统计量会有所不同。

RASE 均方预测误差的平方根。它按以下方式计算：

- 对预测误差（实际响应和预测响应之间的差值）求平方和以获得 *SSE*。
- 用 *n* 表示观测数。
- RASE 是：

$$\text{RASE} = \sqrt{\frac{SSE}{n}}$$

AAE 平均绝对误差。

频数 包含每行频数统计的列。

分类响应的“拟合测度”

熵 R 方 1 减去拟合模型与常数概率模型的负对数似然比。它的范围介于 0 到 1 之间。

广义 R 方 可以应用到一般回归模型的测度。它基于似然函数 L，并且统一尺度后最大值为 1。值为 1 表示完美模型，值为 0 表示并不比常数模型好的模型。对于在标准最小二乘设置中的连续正态响应，“广义 R 方”测度简化为传统 R 方。“广义 R 方”亦称 Nagelkerke/Craig and Uhler R^2，它是 Cox and Snell 伪 R^2 的标准化版本。请参见 Nagelkerke (1991)。

-Log p 均值 -log(p) 的平均值，其中 p 是与发生的事件关联的拟合概率。

RASE 均方根误差，已针对自由度进行调整。对于分类响应，差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

绝对偏差的均值 响应的真实值与预测值的差值绝对值的平均值。对于分类响应，差值介于 1 和 p（实际发生的响应水平的拟合概率）之间。

误分类率 具有最高拟合概率的响应类别不是观测到的类别的比率。

数目 观测数。

相关信息

“神经网络”一章中第 41 页的“训练和验证拟合测度”提供了关于分类响应的拟合测度的详细信息。

JMP PRO “模型比较”平台选项

JMP PRO 连续和分类响应

模型平均 生成一个新列，包含预测值 （针对连续响应）或预测概率 （针对分类响应）的算术均值。

JMP PRO 连续响应

标绘 “预测值 - 实际值”图 显示 “预测值 - 实际值”散点图。不同模型的图叠加在一起。

标绘 “行号 - 残差”图 显示 “行号 - 残差”图。不同模型的图叠加在一起。

刻画器 根据数据中的预测公式列为每个响应显示一个刻画器。刻画器中每行分别显示一个用于比较的模型。

JMP PRO 分类响应

ROC 曲线 显示针对响应变量的每个水平的 ROC 曲线。不同模型的曲线叠加在一起。

AUC 比较 提供每个模型的 ROC 曲线下面积 (AUC) 的比较。曲线下面积是拟合优度的指标，其中 1 表示完美拟合。

该报表包含以下信息：

- 每个 AUC 的标准误差和置信区间
- 每对 AUC 之间的差值的标准误差、置信区间和假设检验
- 检验所有 AUC 是否相等的总体假设检验

提升曲线 显示针对响应变量的每个水平的提升曲线。不同模型的曲线叠加在一起。

累积增益曲线 显示针对响应变量的每个水平的累积增益曲线。累积增益曲线是显示模型标识的响应水平比例与所有响应比例的图。完美模型的累积增益曲线的响应水平总比例为 1.0。不同模型的曲线叠加在一起。

混淆矩阵 显示每个模型的混淆矩阵。混淆矩阵是实际响应和预测响应的双向分类。显示计数和比率混淆矩阵。该平台会为“分组”变量的每个水平分别生成混淆矩阵。

若响应具有“收益矩阵”列属性，则混淆矩阵右侧会显示“决策计数 - 实际值”矩阵和“决策率 - 实际值”矩阵。有关这些矩阵的详细信息，请参见“分割模型”一章中第 69 页的“分割的更多示例”。

刻画器 根据数据中的预测公式列为每个响应显示一个刻画器。刻画器中每行分别显示一个用于比较的模型。

决策阈值 （仅可用于二值分类响应。）显示或隐藏用于训练、验证和测试集（若指定）的“决策阈值”报表。每个报表包含每个模型的拟合概率分布图、每个模型的混淆矩阵以及用于比较模型拟合的分类图。请参见“模型筛选”一章中第 160 页的“‘决策阈值’报表”。

相关信息

- “分割模型”一章中第 65 页的“ROC 曲线”
- “分割模型”一章中第 66 页的“提升曲线”

JMP PRO 模型比较的更多示例

本例使用汽车数据来构建一个预测购买汽车尺寸的模型， 同时比较 Logistic 回归模型和决策树模型。

首先选择**帮助** > **样本数据库**并打开 Car Physical Data.jmp。

创建 Logistic 回归模型

1. 选择**分析** > **拟合模型**。
2. 选择**车型**并点击 **Y**。
3. 选择以下列并点击**添加**：**国家**、**车重**、**转弯半径**、**排气量**和**马力**。
4. 点击**运行**。

   “名义型 Logistic 拟合”报表随即显示。
5. 要将预测公式保存至列，请点击 “名义型 Logistic”红色小三角并选择**保存概率公式**。

创建决策树模型并将预测公式保存至列

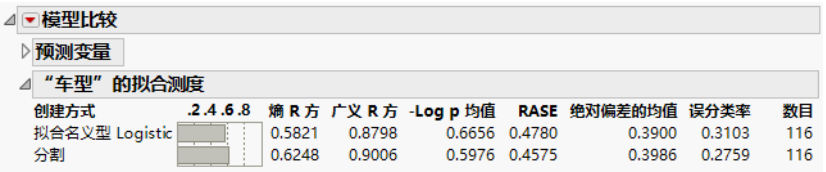
1. 选择**分析** > **预测建模** > **分割**。
2. 选择**车型**并点击 **Y，响应**。
3. 选择**国家**、**车重**、**转弯半径**、**排气量**和**马力**列，并点击 **X，因子**。
4. 确保在 “方法”列表中选定了**决策树**。
5. 点击**确定**。

   “分割”报表随即显示。
6. 点击 10 次**拆分**。
7. 要将预测公式保存至列，请点击 “分割”红色小三角并选择**保存列** > **保存预测公式**。

比较模型

1. 选择**分析** > **预测建模** > **模型比较**。
2. 选择以 Prob 开头的所有列，并点击 **Y，预测变量**。
3. 点击**确定**。

图 11.8 最初的 “模型比较”报表


模型比较

预测变量

“车型”的拟合测度

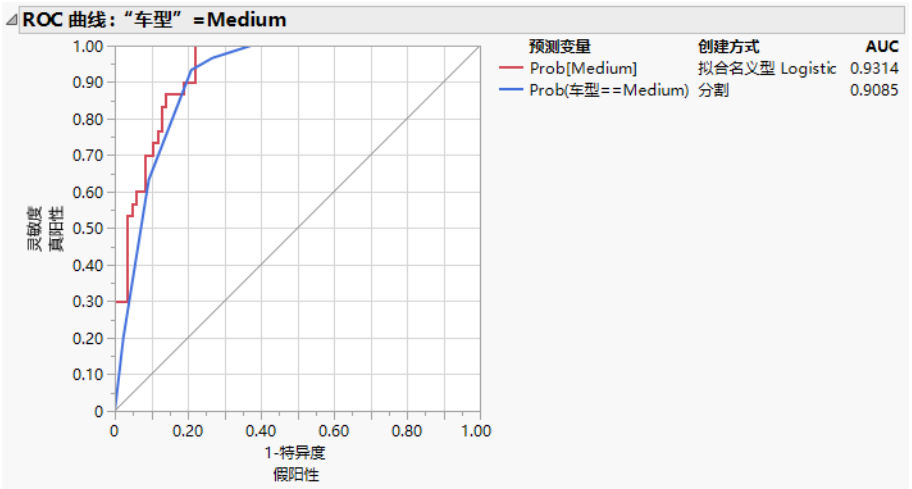
创建方式	.2 .4 .6 .8	熵 R 方	广义 R 方	-Log p 均值	RASE	绝对偏差的均值	误分类率	数目
拟合名义型 Logistic		0.5821	0.8798	0.6656	0.4780	0.3900	0.3103	116
分割		0.6248	0.9006	0.5976	0.4575	0.3986	0.2759	116

该报表显示“分割”模型的“熵 R 方”和“广义 R 方”的值略高，而“误分类率”的值略低。

4. 点击 “模型比较”红色小三角并选择 **ROC 曲线**。

   显示了针对每个**车型**的 ROC 曲线，图 11.9 显示了其中一条曲线。

图 11.9 车型为 “Medium” 的 ROC 曲线

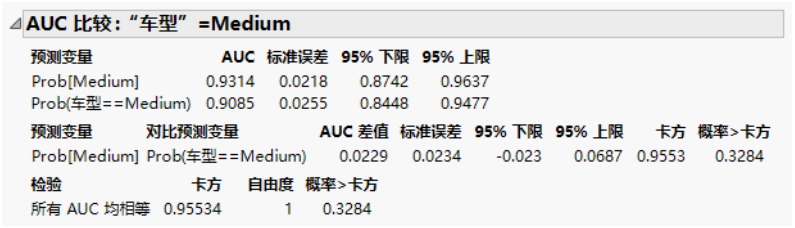


通过查看所有 ROC 曲线，您会发现这两个模型的预测能力差不多。

5. 点击 “模型比较” 红色小三角并选择 **AUC 比较**。

   显示了针对每个**车型**的 “AUC 比较” 报表，图 11.10 显示了其中一个报表。

图 11.10 车型为 “Medium” 的 AUC 比较



该报表显示了两个模型的 AUC 值（ROC 曲线下面积）之间的差值的假设检验结果。通过检查这些结果，您会发现**车型**的任何水平的 AUC 值之间都不存在统计差异。

您可以得出结论认为这两个模型的预测能力没有很大的差异，原因如下：

- R 方值和 ROC 曲线相似。
- AUC 值之间不存在统计上的显著性差异。

第 12 章

JMP PRO 生成验证列

将数据分为训练集、验证集和测试集

仅在 JMP Pro 中提供 “生成验证列”平台。

验证是使用数据集的一部分估计模型参数并使用另一部分评估模型预测能力的过程。您可以将数据划分为两个或三个集。“生成验证列”平台提供五种不同的方法来创建这些分区。

目录

“生成验证列”平台概述 177
“生成验证列”示例 177
启动“生成验证列”平台 179
“生成验证列”报表 181
指定比率或相对比率 181
设置割点 181
选项 182
执行 182
“生成验证列”平台选项 182
“生成验证列”平台的更多示例 182

JMP PRO “生成验证列”平台概述

验证是使用数据集的一部分估计模型参数并使用另一部分评估模型预测能力的过程。对于复杂数据，验证可以降低模型过度拟合的风险。

验证列将数据划分为两个或三个部分。

- 训练集用于估计模型参数。
- 验证集用于帮助选择具有良好预测能力的模型。
- 测试集可在选定模型后检查该模型的预测能力。

验证列可用作许多 JMP 平台中的验证方法。请参见“统计详细信息”附录中第 469 页的“JMP 建模中的验证”。

“生成验证列”平台支持您使用多种方法创建训练集、验证集和测试集。可以指定分层、分组或割点列，以确定用于创建验证列的方法。

JMP PRO “生成验证列”示例

Lipid Data.jmp 数据表包含来自加利福尼亚州医院 95 个研究对象的血液测量值、体格检查测量值和问卷数据。您关注于创建要在将来分析中使用的验证列。

1. 选择**帮助** > **样本数据库**，然后打开 Lipid Data.jmp。
2. 选择**分析** > **分布**。
3. 选择**性别**，然后点击 **Y，列**。
4. 点击**确定**。

图 12.1 Lipid Data.jmp 中性别的分布

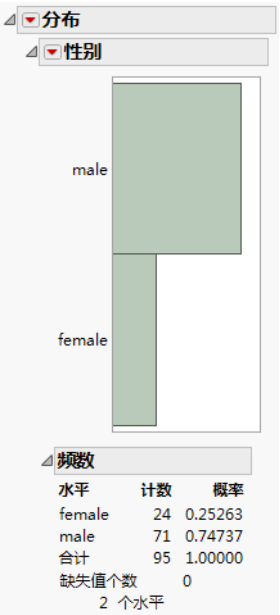


图 12.1 演示了数据集中**性别**的分布。请注意，表示出来的男性和女性的比例不相同。由于数据中的女性稀少，您想要确保验证集与训练集之间的性别比例均衡。

5. 选择**分析 > 预测建模 > 生成验证列**。
6. 选择**性别**并点击**分层列**。
7. 点击**确定**。

   将显示“生成验证列”报表，其中包含所选验证方法的说明。还可以通过选项来更改比率、列类型或设置种子。
8. （可选）在报表的“选项”部分中的**随机种子**旁边键入 1234。
9. 点击**执行**。

   验证列将添加至数据表。您可以通过创建马赛克图来探索验证集和训练集的分布。
10. 选择**分析 > 以 X 拟合 Y**。
11. 将**验证**分配给 **Y，响应**，将**性别**分配给 **X，因子**。
12. 点击**确定**。

图 12.2 性别在验证集和训练集之间的分布

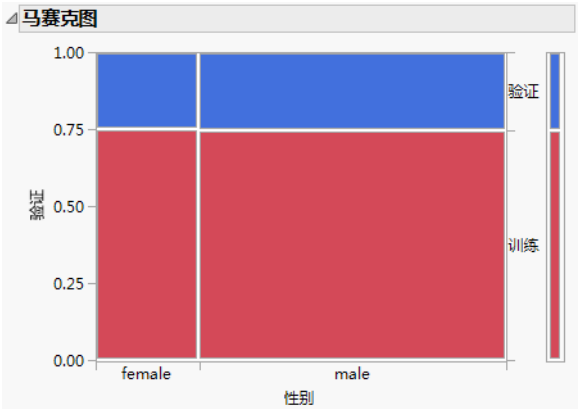
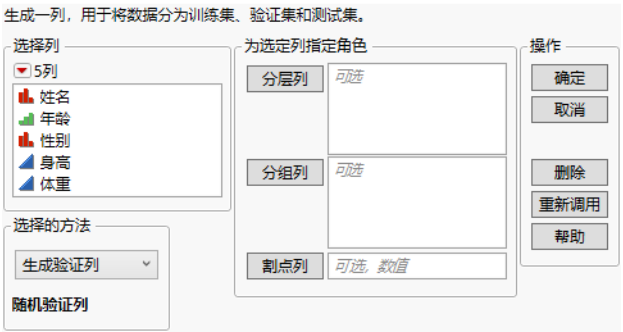


图 12.2 演示了**性别**在每个验证集和训练集之间的分布。请注意，男性和女性都有大约 75% 位于训练集中，并且男性和女性都有大约 25% 位于验证集中。

JMP PRO 启动“生成验证列”平台

通过选择**分析** > **预测建模** > **生成验证列**启动 “生成验证列”平台。

图 12.3 “生成验证列”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“生成验证列”启动窗口提供以下选项：

分层列 分配一个或多个分层列。

分组列 分配一个或多个分组列。

割点列 分配一个数值型割点列。

割点批次 ID 分配割点列时，还可以为割点批次 ID 分配一个列。这样您就能够确定“割点批次 ID”列的每个水平中的割点值。

选择的方法

提供两种验证方法。

生成验证列 基于指定的分层、分组和割点列创建验证列。由指定的分层、分组和割点列确定的验证列方法在该框下进行了说明。选择某种方法并点击 “确定”后，可以在 “生成验证列”报表中为每个集指定分配。请参见第 181 页的 “指定比率或相对比率”和第 181 页的 “设置割点”。有五种方法可用于构造保留集：

随机验证列 在启动窗口中未分配任何列时所用的默认方法。该方法基于在“生成验证列”报表输入的分配将数据划分到各个集中。

分层验证列 分配了一个或多个分层列时所选的方法。该方法基于指定的分层列的水平将数据划分为均衡的各个集。与在“随机验证列”方法中一样，基于在“生成验证列”报表中输入的分配将行随机分配给保留集。不过，分配时针对的是分层列的每个水平或水平组合。当需要均衡表示每个训练集、验证集和测试集中列的水平时，可使用该方法。

分组验证列 指定了一个或多个分组列时所选的方法。该方法通过以下方式将数据划分到各个集中：将指定列的全部水平或是两个或更多列的水平组合都放置在同一个集内。出于该原因，结果集的大小会与您指定的大小稍有不同。若在保留集之间拆分水平不可行，则使用该选项。

按组分层验证列 同时指定了分层和分组列时所选的方法。该方法划分数据以均衡分层列中的水平，同时要求指定的各组一同保留在同一个保留集中。正如在“分组验证列”中那样，可以将组创建为指定列的水平或是两个或更多列的水平组合。结果集的大小会与您指定的大小稍有不同。

割点验证列 指定了割点列时所用的方法。该方法基于时间序列割点将数据划分到各个集中。若您想要基于时间期间将数据分配到保留集，可使用该选项。训练集包含第一个割点与第二个割点之间的行。验证集包含第二个割点与第三个割点之间的行。测试集包含其余行。这些集是基于“设置割点”报表中的选项而选择的。

生成自动验证表 创建一个新的数据表，其中包含与原始数据表中的行拼接的原始数据表中的行的副本。可用于交叉验证的新数据表有四个附加列：

有效集 为原始数据指定值 0，为复制数据指定值 1。该列中的值指定训练集和验证集。在分析的启动窗口中的“验证”角色中使用该列。

有效 ID 分配原始观测的行号。这允许为每个原始观测匹配训练和验证集行。

有效权重 分配要在分析的启动窗口中的“频数”角色中使用的自动验证权重。对于**有效 ID** 的每个值，为训练观测和验证观测生成相同的均匀随机数。对于训练集，**有效权重**计算如下：

有效权重 = -log(1- 有效均匀)

对于验证集，**有效权重**计算如下：

有效权重 = -log(有效均匀)

构造**有效权重**列，以使训练数据权重与验证数据权重负相关。这样可以确保验证数据拟合的差异生成拟合方法的有效交叉验证。

空因子 为**有效** ID 的每个值指定相同的正态随机数。

提示：对小数据表使用“生成自动验证表”，其中使用一个子集作为训练数据可能会导致估计问题。

JMP PRO “生成验证列”报表

当您在启动窗口中点击“确定”时，“生成验证列”报表窗口随即显示。该报表包含对所选验证方法的说明以及与比率、割点、列类型和设置随机种子相关的选项。

JMP PRO 指定比率或相对比率

该报表支持您指定训练集、验证集和测试集的分配。在“训练集”、“验证集”和“测试集”旁边的框中，输入表示您想要加入以上每个集中的行比例或行数的值。默认值构造一个包含大约 75% 的行的训练集以及一个包含大约 25% 的行的验证集。

根据数据表中的行数和选定的验证方法，结果集的实际大小可能与指定的大小略有不同。“调整率”、“行计数”和“组计数”列提供集的实际大小。

JMP PRO 设置割点

仅当在启动窗口中指定割点列后才显示该报表。割点由以下四种方法之一确定：

比例 根据为每个集指定的行比例确定割点。在“训练集”、“验证集”和“测试集”旁边的框中，输入表示您想要加入以上每个集中的比例的值。根据行数和比例，结果集的实际大小可能与指定的大小略有不同。

行数 根据为每个集指定的行数确定割点。在“训练集”、“验证集”和“测试集”旁边的框中，输入表示您想要加入以上每个集中的行数的值。该选项支持您精确指定各个集。

固定时间或日期 根据指定割点列中的固定数据点确定割点。若选择该选项，将显示割点列的最小值和最大值以供参考。在“训练集”、“验证集”和“测试集”旁边的框中，输入表示您想要加入以上每个集中的最小值的值。

耗用时间 根据自割点列中的第一个时间戳以来经过的时间量确定割点。若选择该选项，则会显示自第一个值到最后一个值所经过的总时间以供参考。在“训练集”、“验证集”和“测试集”旁边的框中，输入表示您想要加入以上每个集中的耗用时间量的值。

JMP PRO 选项

您还可以指定以下附加选项：

新建列名 支持您指定验证列的名称。

验证列类型 （不可用于 “割点验证列”方法。）支持您将验证列类型指定为固定或公式。

- 固定列类型生成具有固定值的列。
- 公式列类型生成公式列作为验证列。

随机种子 （不可用于“割点验证列”方法，或是在指定公式列类型时不可用。）支持您设置一个随机种子，以便将来重现相同的验证列。

JMP PRO 执行

点击“生成验证列”报表中的**执行**可将验证列添加到数据表中。验证列具有可提供分层、分组或割点变量的“注释”属性。

JMP PRO “生成验证列”平台选项

“生成验证列”实用工具提供以下红色小三角菜单选项：

请参见《使用 JMP》获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

JMP PRO “生成验证列”平台的更多示例

本例使用一年内从美国的 16 个气象站收集的每周气象数据。运行数据表中的“气象站位置”脚本来查看位置图。并非每个气象站都提供一年中每周的周温度测量值。您关注的是根据数据收集的日期为该数据表创建验证列。对于每个气象站，您希望使用前 60% 的观测进行训练，使用接下来 25% 的观测进行验证，使用最后 15% 的观测进行测试。本例演示在此类情形中使用“批次 ID”列的重要性。

创建使用割点的验证列

1. 选择**帮助 > 样本数据库**，然后打开 Functional Data/Weekly Weather Data.jmp。
2. 选择**分析 > 预测建模 > 生成验证列**。
3. 选择**一年的某一周**并点击**割点列**。

4. 点击**确定**。
5. 在 “确定割点使用的是”旁边的列表中，选择**比例**。
6. 在 “训练集”、“验证集”和 “测试集”旁边的框中，分别输入 0.60、 0.25 和 0.15。
7. 在 “新建列名”旁边的框中，键入 “割点验证”。
8. 点击**执行**。

   名为 “割点验证”的验证列将添加至数据表。
9. 选择**分析 > 制表**。
10. 点击 ID 并将其拖至行的拖放区。
11. 点击**割点验证**并将其拖至**数目**的顶部。
12. 点击**行百分比**并将其拖至单元格顶部。

图 12.4 割点验证列比例

	割点验证		
	训练	验证	测试
ID	行百分比	行百分比	行百分比
Bristol, TN	62.79%	20.93%	16.28%
Corpus Christi, TX	70.27%	18.92%	10.81%
Denver, CO	58.14%	23.26%	18.60%
Goodwell, OK	64.71%	17.65%	17.65%
Grand Canyon National Park, AZ	53.66%	26.83%	19.51%
Greenville, ME	60.47%	20.93%	18.60%
Harrisburg, PA	60.53%	26.32%	13.16%
Helena, MT	60.00%	25.00%	15.00%
Lynchburg, VA	71.79%	17.95%	10.26%
Madera, CA	67.50%	22.50%	10.00%
Miami Beach, FL	63.16%	26.32%	10.53%
Minneapolis/St. Paul, MN	57.50%	27.50%	15.00%
N. Myrtle Beach, SC	69.05%	16.67%	14.29%
Peason Ridge Range, LA	57.89%	28.95%	13.16%
Philip, SD	59.52%	23.81%	16.67%
Terra Haute, IN	47.22%	33.33%	19.44%

图 12.4 显示并非所有的气象站都具有正确的训练集、验证集和测试集比例。使用“批次 ID”列获取正确的百分比。

创建使用割点和批次 ID 的验证列

1. 选择**分析 > 预测建模 > 生成验证列**。
2. 选择**一年的某一周**并点击**割点列**。
3. 选择 ID 并点击**割点批次 ID**。
4. 点击**确定**。
5. 在 “确定割点使用的是”旁边的列表中，选择**比例**。
6. 在 “训练集”、“验证集”和 “测试集”旁边的框中，分别输入 0.60、 0.25 和 0.15。
7. 在 “新建列名”旁边的框中，键入 “割点批次验证”。
8. 点击**执行**。

名为 “割点批次验证”的验证列将添加至数据表。

9. 选择**分析 > 制表**。
10. 点击 ID 并将其拖至行的拖放区。
11. 点击**割点批次验证**并将其拖至**数目**的顶部。
12. 点击**行百分比**并将其拖至单元格顶部。

图 12.5 使用批次 ID 比例的割点验证列

	割点批次验证		
	训练	验证	测试
ID	行百分比	行百分比	行百分比
Bristol, TN	60.47%	25.58%	13.95%
Corpus Christi, TX	59.46%	24.32%	16.22%
Denver, CO	60.47%	25.58%	13.95%
Goodwell, OK	58.82%	26.47%	14.71%
Grand Canyon National Park, AZ	60.98%	24.39%	14.63%
Greenville, ME	60.47%	25.58%	13.95%
Harrisburg, PA	60.53%	26.32%	13.16%
Helena, MT	60.00%	25.00%	15.00%
Lynchburg, VA	58.97%	25.64%	15.38%
Madera, CA	60.00%	25.00%	15.00%
Miami Beach, FL	60.53%	26.32%	13.16%
Minneapolis/St. Paul, MN	60.00%	25.00%	15.00%
N. Myrtle Beach, SC	59.52%	26.19%	14.29%
Peason Ridge Range, LA	60.53%	26.32%	13.16%
Philip, SD	59.52%	26.19%	14.29%
Terra Haute, IN	61.11%	25.00%	13.89%

图 12.5 显示使用 “割点批次 ID”列可确保每个气象站具有更接近指定值的训练集、验证集和测试集比例。

第 13 章

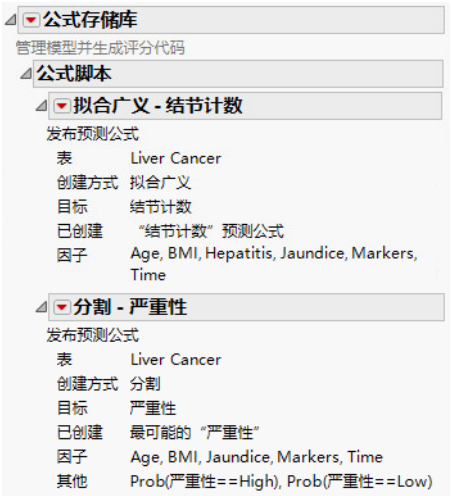
JMP PRO 公式存储库

管理模型和生成得分代码

仅在 JMP Pro 中提供 “公式存储库”平台。

“公式存储库”是一个用来组织、比较和刻画模型的储存库。可以在 JMP 内部或外部为发布到“公式存储库”的模型生成用于部署的得分代码。对于模型探索任务，您可以使用“公式存储库”在 JMP 数据表之外存储候选模型。可以从“公式存储库”访问模型刻画器和模型比较平台。选定要进一步使用的模型可保存至 JMP 表或保存至包含要评分的新数据的 JMP 表。要在 JMP 之外的环境中使用，您可以使用 C、Python、JavaScript、SAS 或 SQL 生成得分代码。

图 13.1 公式存储库示例



目录

“公式存储库”平台概述 187
“公式存储库”示例 187
启动“公式存储库”平台 188
将预测公式发布到“公式存储库”的平台 188
“公式存储库”平台选项 189
公式存储库模型选项 190
从“公式存储库”平台生成得分代码 190

JMP PRO “公式存储库”平台概述

“公式存储库”是一个用来对要部署的模型进行组织、比较、刻画和评分的储存库。模型即预测公式。预测公式作为列脚本保存到“公式存储库”中。您可以向 JMP 表添加预测公式以便对数据评分。您还可以使用“公式存储库”为预测公式生成得分代码，以便于在 JMP 之外的环境中部署模型。

“公式存储库”支持您执行以下任务：

- 将预测公式保存到数据表之外
- 将多个数据表中的预测公式保存到一个公共位置
- 保存预测公式中的中间清理公式
- 比较模型
- 刻画模型
- 向数据表添加预测公式以便对 JMP 中的新数据评分
- 生成得分代码（C、Python、JavaScript、SAS 或 SQL）以便在 JMP 之外部署模型
- “公式存储库”将中间公式包括在生成的得分代码中。基于列公式中的输入构造的模型将相应公式包括在得分代码中。例如，若您的模型包括一个重新编码的变量，则重新编码公式将包括在得分代码中。这样您就可以将预测公式应用到尚未处理的原始数据。

注意：中间公式不支持信息性缺失。

JMP PRO “公式存储库”示例

Liver Cancer.jmp 样本数据表包含在肝癌患者加入研究后的肝癌严重性数据。该文件还包含针对模型的一些表脚本。本例使用这些脚本生成用于演示 “公式存储库”的模型。

1. 选择**帮助 > 样本数据库**，然后打开 Liver Cancer.jmp。
2. 点击 “Lasso Poisson，验证列”脚本旁边的绿色小三角。
3. 点击 “使用 ‘验证列’验证的 ‘Poisson’自适应 ‘Lasso’”旁边的红色小三角，然后选择**保存列 > 发布预测公式**。

   该选项打开包含拟合广义模型的预测公式的 “公式存储库”。
4. 要向数据表添加预测公式，请从 “公式存储库”中的 “拟合广义 - 结节计数”红色小三角中选择**运行脚本**。

注意：可使用第 3 步中的**保存列 > 保存预测公式**获取第 3 步和第 4 步的结果。不过，此时预测公式就不再是 “公式存储库”的一部分了。

5. 要生成在 JMP 之外使用的得分代码，请从 “拟合广义 - 结点计数”红色小三角中选择一种代码类型。随即显示包含该代码的脚本窗口。请参见第 190 页的 “从 “公式存储库”平台生成得分代码”。
6. 要保存 “公式存储库”，请选择**文件 > 保存**。

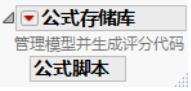
注意：“公式存储库”另存为 JMP 报表文件 (*.jrp)。

图 13.2 包含广义模型的 “公式存储库”

JMP PRO 启动 “公式存储库”平台

通过选择**分析 > 预测建模 > 公式存储库**启动 “公式存储库”。

图 13.3 启动后的空 “公式存储库”



或者，若没有打开的“公式存储库”，则会在您选择 “发布”命令后打开一个 “公式存储库”。

JMP PRO 将预测公式发布到 “公式存储库”的平台

发布预测公式和生成得分代码的平台包括：

- 判别
- 最小二乘回归
- Logistic 回归
- 分割
- 提升
- K 最近邻
- 朴素 Bayes

- 神经
- 潜在类分析
- 主成分
- 广义回归
- 偏最小二乘
- 高斯过程

在不向“公式存储库”发布预测公式的平台中，您可以将预测公式保存至数据表。可通过选择**从列添加公式**从该数据表将预测公式添加至“公式存储库”。不过，得分代码不一定能对此类模型发挥全部作用。有关得分代码的详细信息，请参见第 190 页的“从“公式存储库”平台生成得分代码”。

此外，您可以直接从公式编辑器发布公式。选择该公式，右击并选择**将表达式发布至公式存储库**。

JMP PRO “公式存储库”平台选项

从列添加公式 支持您将数据表列中的现有预测公式添加到当前公式存储库。若表中仅有一个公式，该公式将添加到存储库中。若有多个列公式，则会有一个对话框提示您选择相关公式。若已在数据表中选定公式列，这些列将在该对话框中处于预先选定状态。

显示多个脚本 打开新的“公式”窗口（或追加到打开的“公式”窗口），其中包含当前公式存储库中的所有公式的脚本。

复制多个脚本 将所有脚本从当前公式存储库复制到剪贴板。

将公式复制为函数 支持您从当前公式存储库中选择要作为函数复制的用于 JSL 的模型。选定的模型将复制到剪贴板中。这些函数针对标量变量而不是列操作。为每个选定模型都生成一个命名空间。每个命名空间都有一个函数（称为预测），该函数将模型输入用作参数并返回模型预测变量。

将公式复制为变换 支持您从当前公式存储库中选择要复制的模型。选定的模型在 `Transform Columns()` 语句中复制到剪贴板上。

运行脚本 支持您将模型从当前公式存储库保存至 JMP 数据表中的新列。带有中间列公式的公式将中间列添加到 JMP 数据表中。

> **注意：**若打开了不止一个数据表，则会显示一个“选择表”窗口，该窗口支持您指定要用于“运行脚本”的数据表。

生成 C 代码、生成 Python 代码、生成 JavaScript 代码、生成 SAS 代码、生成 SQL 代码

支持您从当前公式存储库中选择要生成代码的模型。随即显示一个新的脚本窗口，其中包含分别以 C、Python、JavaScript、SAS DS2 或 SQL 编写的选定模型的得分代码。您可以使用

该代码促进模型在您选择的环境或框架中的部署。请参见第 190 页的 “从 “公式存储库” 平台生成得分代码”。

模型比较 支持您使用模型比较实用工具从当前公式存储库中选择要比较的模型。若来自多个表的模型存储在存储库中，首先要选择相关表，然后再选择相关模型。选择 “模型比较” 选项时按住 Shift 键可在新窗口中启动 “模型比较”。请参见第 163 页的 “模型比较” 一章。

注意：若打开了不止一个数据表，则会显示一个 “选择表” 窗口，该窗口支持您指定要用于 “模型比较” 的数据表。

删除模型比较 从当前公式存储库中删除所有 “模型比较” 报表。

刻画器 支持您使用刻画器从当前公式存储库中选择要刻画的模型。若来自多个表的模型存储在存储库中，您首先要选择相关表，然后再选择相关模型。选择刻画器选项时按住 Shift 键可在新窗口中启动刻画器。请参见 《刻画器指南》。

注意：若打开了不止一个数据表，则会显示一个 “选择表” 窗口，该窗口支持您指定要用于 “刻画器” 的数据表。

删除刻画器 从当前公式存储库中删除所有刻画器。

请参见《使用 JMP》获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算” 选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

JMP PRO 公式存储库模型选项

保存在 “公式存储库” 中的每个预测公式都有一个单独的选项菜单。在与主菜单选项对应的 “显示”、“复制”、“运行” 和 “生成代码” 选项之外，单个菜单还包含以下选项：

重命名新列 支持您重命名模型。若通过运行脚本将预测公式添加至 JMP 数据表，这个新名称将作为列名应用。此外，该名称还包括在生成的代码中。

删除 从 “公式存储库” 中删除模型。该命令无法撤销。

JMP PRO 从 “公式存储库” 平台生成得分代码

生成得分代码是为了在生产环境或您选定的其他框架中加快使用在 JMP 中构建的模型。许多平台都向 “公式存储库” 发布预测公式；不过，并不是所有预测公式都生成完整代码。代码可以是完整的，可以是代码片段，代码还可包括要求额外编程才可以实现的不受支持的函数。错误消息经常指示函数调用不受支持，需要在目标语言中进行进一步的编程。实现较为复杂的函数

可能还需要特定于目标语言的逻辑库。下文中特定于语言的章节会对这些逻辑库进行说明。例如，您可以执行以下任务：

- 使用生成的 SAS 代码将模型部署到 SAS Model Manager 中。
- 使用生成的 SQL 代码通过 In-Database Scoring 扩充 ETL 过程。
- 借助用生成的 C 代码构建的应用程序创建用于数据变换管道的节点。
- 创建 Jupyter notebook，以使用生成的 Python 代码显示实时得分结果。
- 使用包括生成的 JavaScript 的 Web 应用程序支持用户对自己的数据评分。

对于 C、Python 和 JavaScript 语言，您必须在部署或编译生成的代码时包括 .h 文件之类的支持代码和实用程序库。这些文件在 JMP 安装文件夹的 Scoring 文件夹中提供。

提示：要在您的计算机上查找安装文件夹的位置，请使用 JSL 命令 `Get Path Variable ("$ALL_HOME/Scoring");`。

C 代码 生成的 C 得分代码必须编译为程序库，然后链接到应用程序中。您可以使用静态或动态链接方法。可以在您的 JMP 安装文件夹的 Scoring/C 文件夹中找到编译和链接所需的文件 jmp_lib.h、 jmp_parms.h 和 jmp_score.h。

Python 代码 可以在您的 JMP 安装文件夹的 Scoring/Python 文件夹中找到运行 Python 得分应用程序所需的文件 jmp_score.py。部署模型时，将 jmp_score.py 复制到可执行得分代码所在的位置。

调用生成的代码对数据评分时，输入数据务必要采用与 JMP 生成的代码兼容的数据结构传递。 Python 得分代码的输入和输出参数应该采用类似字典的类型。可通过键来存取和编写值，如 Spark 中的 DataFrame 行对象、 Dask 和 Pandas 框架，或是标准的 Python “dict”类。

使用线性代数运算符或 Vec Quadratic 和 Design Norm 之类的高级运算符需要 NumPy Python 逻辑库。

JavaScript 代码 可以在 Scoring/JavaScrip 文件夹中找到运行 JavaScript 得分应用程序所需的文件 jmp_score.js。

SAS 代码 生成的代码片段 （曾经包装在 PROC DS2 语句中）可用在包括 SAS In-Database Code Accelerator 在内的 SAS 应用程序中。该代码添加到 “SAS 窗口”并且在代码之前的注释代码块中包括变量名映射代码段。在 DATA 步的末尾会使用删除命令删除与中间列关联的所有临时变量名。

提示：对于具有 ifmax 调用的模型 （如分类响应的 Logistic 模型或神经网络模型），将临时变量声明移至 method run() 语句之前。

SQL 代码 曾经包装在 select 语句中的 SQL 代码片段可用在针对多数主要数据库服务器的 SQL 查询中。

注意：包含“placeholder”或“ERROR”的代码指示函数调用不受支持。

第 14 章

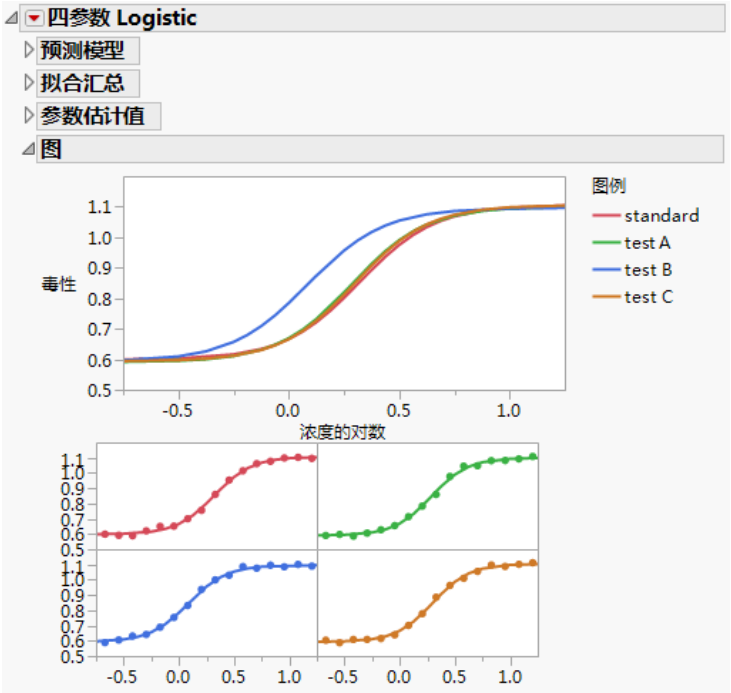
拟合曲线

拟合数据的内置非线性模型

在许多情况下，特别是在物理和生物科学中，可通过某些著名的非线性方程来描述变量之间的关系。例如药理生物测定实验，可以演示将某种药物的反应强度作为药物浓度的函数，看它是如何变化的。S 形曲线往往能够将反应强度作为药物浓度的函数进行准确地建模。另外一个示例是指数增长曲线，它能够对随时间变化的人口数量进行建模。

“拟合曲线”平台不要求您指定参数估计值的起始值或创建模型公式。要自行指定起始值并创建模型公式，可以使用具有更强定制功能的“非线性”平台，该特质还能拟合任何非线性模型。请参见第 215 页的“非线性回归”一章。

图 14.1 “拟合曲线”平台中的非线性拟合的示例



目录

“拟合曲线”平台概述 195
使用“拟合曲线”平台的示例 195
启动“拟合曲线”平台 198
“拟合曲线”报表 199
“拟合曲线”选项 199
“模型比较”报表 200
“模型拟合”报表 201
“模型拟合”选项 201
检验平行性 203
比较参数估计值 205
等价性检验 206
“拟合曲线”平台的统计详细信息 207
模型公式 207

“拟合曲线”平台概述

某些模型的参数是**线性**关系的（例如，二次式或其他多项式）；而某些模型的参数可变换为线性关系（例如，在您使用 x 的对数变换时）。“拟合模型”或“以 X 拟合 Y”平台在这些情形下更适用。《拟合线性模型》中的示例显示了吸氧量与跑步时间之间存在显著的线性关系。有关“拟合模型”的详细信息，请参见《拟合线性模型》。有关“以 X 拟合 Y”的详细信息，请参见《基本分析》。

“拟合曲线”平台支持您拟合参数具有**非线性**关系的模型。本章中的第一个示例将展示一个非线性关系的分析：药物毒性是浓度的函数， 但浓度对于毒性的影响随着剂量由低到高发生变化，所以这种关系是非线性的。

下面列出了线性和非线性函数的方程的示例。

线性函数：$Y = \beta_0 + \beta_1 e^{x}$

非线性函数：$Y = \beta_0 + \beta_1 e^{\beta_2 x}$

“拟合曲线”平台提供预定义的模型，如多项式、Logistic、Gompertz、指数、峰值和药代动力学模型。指定分组变量允许您为分组变量的每个水平分别估计模型参数， 因此您可以比较分组变量各水平下的拟合模型和估计参数。

“拟合曲线”还支持您构建模型以创建预测公式。之后您可以在“非线性”中设置参数上限和下限。请参见“非线性回归”一章中第 235 页的“设置参数限值的示例”。在 JMP Pro 中，还可以指定一组补充变量，并在“拟合曲线”平台中拟合广义回归模型，以确定这些变量如何影响响应。

使用“拟合曲线”平台的示例

本例演示如何构建一个关于药物的毒性与浓度之间函数关系的模型。您现有药物的标准配方，想要将其与三种新配方进行比较。

您关注在每种药物的特定浓度下，存活细胞与未存活细胞的毒性比。您从以往的研究中已经了解到每种药物配方在 16 种不同浓度下的毒性比。毒性比越高表示毒性越高，这不利于药物的研发。事先已经计算好浓度的对数，这样可以缩小浓度值的范围，方便检测曲线中的差异。

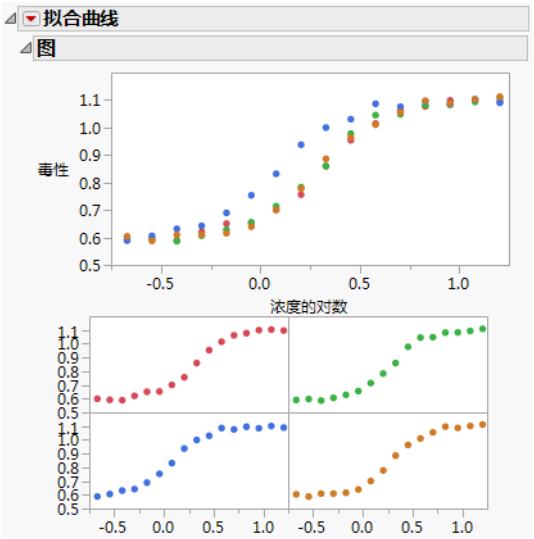
请按照以下步骤构建模型：

1. 选择**帮助** > **样本数据库**，然后打开 Nonlinear Examples/Bioassay.jmp。
2. 选择**分析** > **专业建模** > **拟合曲线**。
3. 将**毒性**分配给 **Y，响应**角色。
4. 将**浓度的对数**分配给 **X，回归变量**角色。
5. 将**配方**分配给**分组**角色。

6. 点击**确定**。

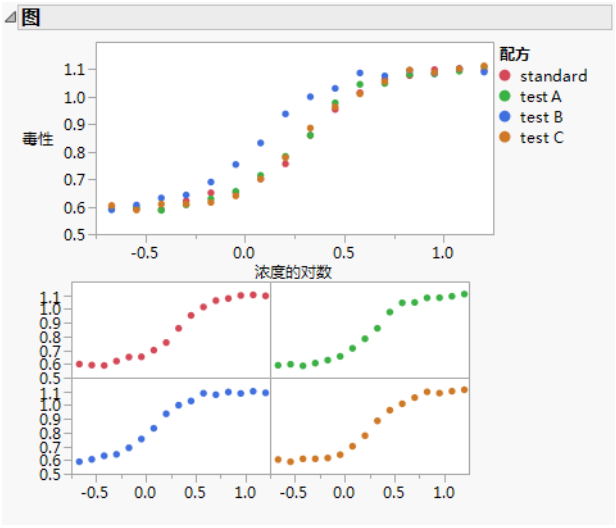
“拟合曲线”报表随即显示。“图”报表包含每种配方的拟合模型的叠加图。

图 14.2 初始 “拟合曲线”报表



7. 要查看标识每种药物配方的图例，请右击其中一个图并选择**行图例**。为该列选择**配方**并点击**确定**。

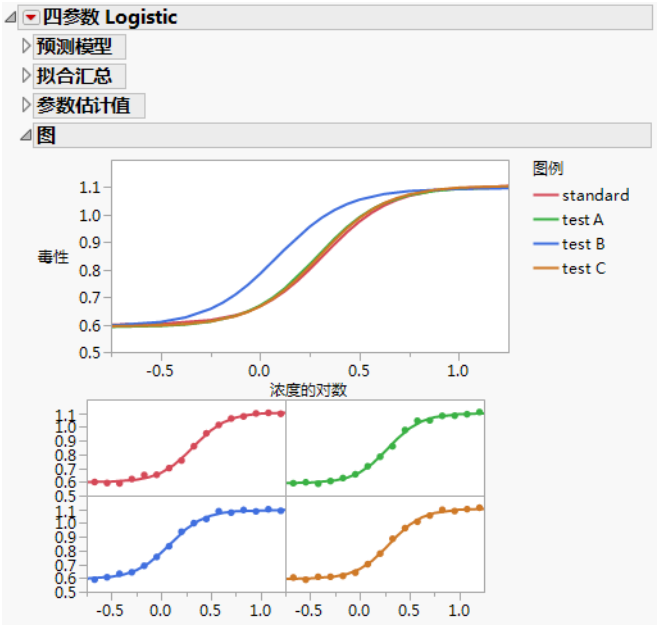
图 14.3 带有图例的 “拟合曲线”报表



这些曲线呈 S 形，所以 S 形曲线是合适的拟合。表 14.1 显示了 “拟合曲线”平台提供的不同类型模型的公式和图形描述。

8. 点击 “拟合曲线”红色小三角并选择 **S 形曲线 > Logistic 曲线 > 四参数 Logistic 拟合**。

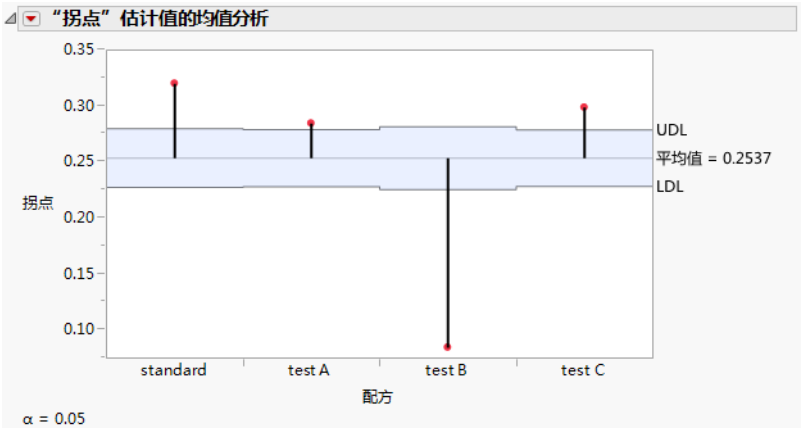
图 14.4 “四参数 Logistic”报表



“四参数 Logistic”报表随即显示。每种药物配方还单有一个图。拟合曲线图表明配方 B 可能不同，因为 “test B”曲线比其他曲线上升得都要早。拐点参数导致了曲线的上升。

9. 点击 “四参数 Logistic”红色小三角并选择**比较参数估计值**。

图 14.5 “参数比较”报表



注意到“test B”配方的“拐点”参数明显低于拐点平均值。该结果与图 14.4 中显示的图正好吻合。药物配方 B 的毒性比高于其他配方。

启动“拟合曲线”平台

要启动 “拟合曲线”平台，请选择**分析** > **专业建模** > **拟合曲线**。启动窗口如图 14.6 所示。

图 14.6 “拟合曲线”平台启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“拟合曲线”平台启动窗口具有以下功能：

Y，响应 指定 *Y* 变量。

X，回归变量 指定 *X* 变量。

分组 指定分组变量。分组变量的不同水平下的拟合模型包含不同的参数。这样您就可以比较分组变量各水平下的拟合模型和估计参数。

Z，补充 指定一个或多个补充变量。“拟合曲线”平台中的任何计算都不使用补充变量，而且把补充变量包括在内并不影响结果。补充变量是您可能希望在将来分析“拟合曲线”的结果时使用的变量。指定补充变量时，它们将包含在 “生成参数表”选项创建的表中。在 JMP Pro 中，若指定一个或多个补充变量和一个分组变量，则“曲线实验设计分析”选项可用于拟合模型。请参见第 202 页的 “曲线实验设计分析”。

权重 指定包含观测权重的变量。

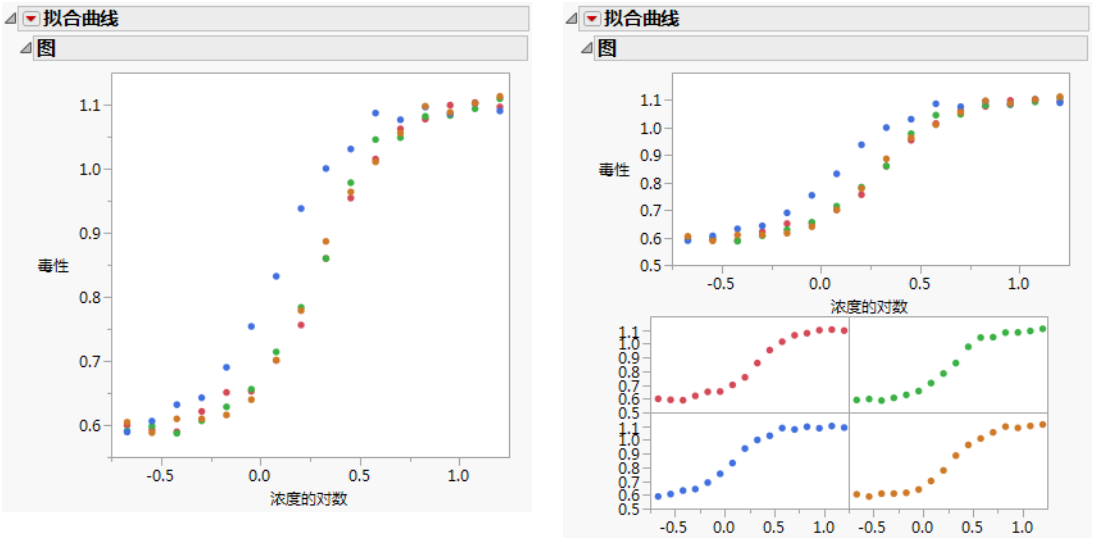
频数 指定包含观测频数的变量。

依据 指定一个变量，为该变量的每个水平分别进行分析。

“拟合曲线”报表

“拟合曲线”报表最初仅包含 X-Y 图。若您指定 “分组”变量，该报表则包含每一组拟合模型的叠加图和单个图。

图 14.7 “拟合曲线”报表：无分组变量 （左）和有分组变量 （右）



拟合模型之后，拟合的模型将显示在图中（在平台启动窗口中未指定分组变量的情况下）。“模型比较”报表显示在图上方，“模型报表”显示在图下方。每次拟合新模型时，它会显示在“模型比较”报表中，并且报表窗口中会显示新的模型报表。请参见第 200 页的 “模型比较”报表 和第 201 页的 “模型拟合”报表 。

注意：在数据表中排除的所有行也在“拟合曲线”图中隐藏。

“拟合曲线”选项

从“拟合曲线”红色小三角菜单中选择以下任意内置模型。请参见第 207 页的 “模型公式”。

多项式 拟合一阶到五阶多项式。

S 形曲线 拟合 Logistic、 Probit、 Gompertz 和 Weibull 模型。这些模型呈 S 形，具有上下渐近线。双参数、三参数和四参数 Logistic 模型以及双参数和四参数 Probit 模型都是对称的。五参数 Logistic 和双 Gompertz 模型不对称。仅当响应介于 0 与 1 之间时，双参数 Logistic 才可用。仅当响应值和回归变量值均不为负值时， Weibull Growth 才可用。举例来说，学习曲线和肿瘤生长建模曲线是 S 形曲线，这两种曲线都是起初增长而后增长逐渐减弱。

指数增长和衰减 拟合“指数”、“双指数”、“机理生长”和“细胞生长”模型。双参数指数和三参数指数相似，但是三参数模型有渐近线。双指数模型假定有两个不同的增长或衰减过程。“机理生长”和“三参数指数”模型一直保持增长（或降低），但增长率（或衰减率）会减慢，因此模型具有渐近线。举例来说，病毒传播和药物半衰期分别符合指数增长和指数衰减函数。

峰模型 拟合高斯峰和 Lorentzian 峰模型。这些模型增长到峰值然后衰减。高斯峰模型是统一尺度版本的高斯概率密度函数 (PDF)。Lorentzian 峰模型是统一尺度版本的 Cauchy 分布（连续概率分布）。这些模型可用于某些化学浓度试验和人工神经网络。

药物代谢动力学模型 拟合“单室口服剂量”模型、“双室快速静脉注射剂量”模型和“四参数双指数”模型。该选项用于对体内的药物浓度建模。

Michaelis Menten 拟合 拟合 Michaelis Menten 生化动力学模型，该模型将酶促反应率与底物浓度联系起来。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

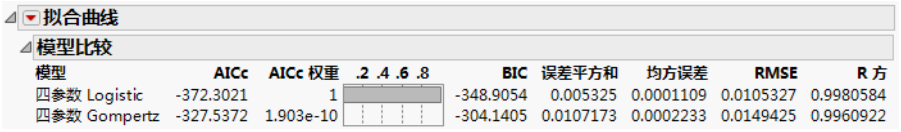
保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“模型比较”报表

要创建图 14.8 中所示的报表，请从“拟合曲线”红色小三角菜单中选择 **S 形曲线 > Logistic 曲线 > 四参数 Logistic 拟合**和 **S 形曲线 > 四参数 Gompertz 拟合**。

图 14.8 “模型比较”报表


拟合曲线

模型比较

模型	AICc	AICc 权重	.2 .4 .6 .8	BIC	误差平方和	均方误差	RMSE	R 方
四参数 Logistic	-372.3021	1		-348.9054	0.005325	0.0001109	0.0105327	0.9980584
四参数 Gompertz	-327.5372	1.903e-10		-304.1405	0.0107173	0.0002233	0.0149425	0.9960922

“模型比较”报表显示用于比较多个模型的拟合统计量。这些统计量包括 AICc、AICc 权重、BIC、误差平方和、均方误差、RMSE 和 R 方，它们的定义如下所述。

AICc 用于比较两个或多个统计模型的拟合优度测度。AICc 是针对小样本而调整的 AIC。仅当数据点数至少比参数个数多 2 个时，才可以计算 AICc。AICc 值最低的模型是最佳模型，在本例中是“四参数 Logistic”模型最佳。请参见《拟合线性模型》。

AICc 权重 提供加总为 1 的标准化 AICc 值。AICc 权重可解释为在其中的一个拟合模型成立的前提下，特定模型为真实模型的概率。因此，AICc 权重最接近 1 的模型是最佳拟合。在

本例中，很明显 “四参数 Logistic”模型是更好的拟合。仅使用非缺失 AICc 值计算 AICc 权重：

AICc 权重 = exp[-0.5(AICc-min(AICc))] / sum(exp[-0.5(AICc-min(AICc))])

其中， min(AICc) 是拟合模型中的最小 AICc 值。“AICc 权重”列是按照递减顺序排列的。

BIC 基于模型拟合的似然函数的测度，用于比较不同的拟合模型。模型的 BIC 值越低表明拟合的越好。请参见 《拟合线性模型》。

误差平方和 每个观测与其预测值之间差值的平方和。

均方误差 提供每个值的误差平方的平均值。

RMSE MSE （均方误差）的平方根，用于估计随机误差的标准差。

R 方 估计由模型而非随机误差解释的响应变量的变异的比例。R 方值越接近于 1，模型的拟合越好。

“模型比较”平台还提供其他选项，如标绘残差和实际值。请参见第 163 页的“模型比较”一章。

“模型拟合”报表

平台会为每个拟合模型创建一个报表。每个模型报表的红色小三角菜单包含以下选项。

预测模型 提供代数形式的预测公式和参数。

拟合汇总 提供与 “模型比较”报表相同的拟合统计量。

参数估计值 提供参数、标准误差、卡方统计量、 *p* 值和置信区间的估计值。同时还提供估计值的相关性和协方差。

提示：您可以通过点击您想要作为排序依据的列对 “参数估计值”报表表进行排序。

图 提供含拟合模型的数据图 （图 14.7） 。仅当您在平台启动窗口上选择分组变量时才显示这些图。

“模型拟合”选项

每个模型报表都有一个红色小三角菜单，其中包含以下部分或全部选项：

检验平行性 帮助确定曲线沿着 X 轴平移时曲线的形状是否相似。在特定情形下，在各组之间进一步比较之前确定平行性是很重要的。仅当在平台启动窗口中指定“分组”变量后，该选项才可用。该选项可用于 S 形模型 （Logistic 和 Gompertz） 以及线性回归模型，但高阶多项式除外。请参见第 203 页的 “检验平行性”。

曲线下面积 提供拟合曲线下的面积。该选项适用于以下模型：单室、双室、高斯峰和 Lorentzian 峰。该选项还适用于四参数双指数模型，但仅限于所有参数均为正的情况。积分范围取决于模型类型，它会在报表中指定。

若在平台启动窗口中指定了分组变量，则会执行 “均值分析”以便比较不同组之间的估计值。若某个组的结果超过了决策限，则该结果将被视为与整体的 AUC 均值存在差异。

到达峰值响应时间 显示回归变量 *X* 在拟合曲线峰值处的估计值。同时还显示估计值的标准误差。该选项可用于 “四参数细胞生长”和 “单室”模型。

峰值响应 显示响应 *Y* 在拟合曲线峰值处的估计值。同时还显示估计值的标准误差。该选项可用于 “四参数细胞生长”和 “单室”模型。

比较参数估计值 提供用于检验参数在分组变量各水平之间是否相等的分析。仅当在平台启动窗口中指定 “分组”变量后，该选项才可用。请参见第 205 页的 “比较参数估计值”。

等价性检验 提供用于检验模型在分组变量各水平之间是否等价的分析。仅当在平台启动窗口中指定 “分组”变量后，该选项才可用。请参见第 206 页的 “等价性检验”。

JMP PRO **曲线实验设计分析** （仅当在启动窗口中指定了分组变量和至少一个补充变量时才可用。）在 “拟合曲线”平台中启动 “广义回归”报表。将补充变量用作模型效应，对非线性模型的各个参数进行广义回归模型拟合。默认情况下，拟合二阶析因模型，“估计方法”为 “向前选择”。然后将这些模型组合起来，创建响应刻画器，作为回归变量和补充变量的函数。随后可以使用 “CDOE 刻画器”探索补充变量如何影响响应。

“曲线实验设计分析”报表包含以下红色小三角菜单选项：

模型参数的广义回归 显示或隐藏每个模型参数的“广义回归”报表。有关“广义回归”模型报表的详细信息，请参见《拟合线性模型》。

诊断图 显示或隐藏响应变量的“预测值-实际值”图和残差图。

CDOE 刻画器 显示或隐藏“CDOE 刻画器”，该刻画器支持您探索响应如何根据补充变量进行更改。有关“CDOE 刻画器”红色小三角菜单选项的详细信息，请参见《刻画器指南》。

保存预测公式 在数据表的新列中保存响应的预测公式。

生成参数表 将参数估计值、标准误差和 t 比保存到数据表中。仅当在平台启动窗口中指定“分组”变量后，该选项才可用。

标绘 “预测值 - 实际值”图 在垂直轴上标绘实际 Y 值，在水平轴上标绘预测的 Y 值。

标绘 “预测值 - 残差”图 在垂直轴上标绘残差，在水平轴上标绘预测的 Y 值。

刻画器 显示或隐藏拟合预测函数的刻画器。导数是预测函数对 *X* 变量求导得到的。有关刻画器的详细信息，请参见 《刻画器指南》。

保存公式 包含用于将多个公式列保存到数据表中的选项。

保存预测公式 保存预测方程。

保存预测值标准误差 保存预测值的标准误差。

保存带参数的预测公式 保存参数形式的预测方程。若您想要在定制“非线性”平台中使用拟合模型，该选项很有帮助。

保存残差公式 保存残差。

保存学生化残差公式 保存学生化残差公式，就是用标准残差除以其估计标准差。

保存一阶导数 保存预测函数对 *X* 变量求导得到的导数。

保存一阶导数的标准误差 保存一阶导数的标准误差的方程。

保存逆预测公式 保存用于从 *Y* 预测 *X* 的方程。

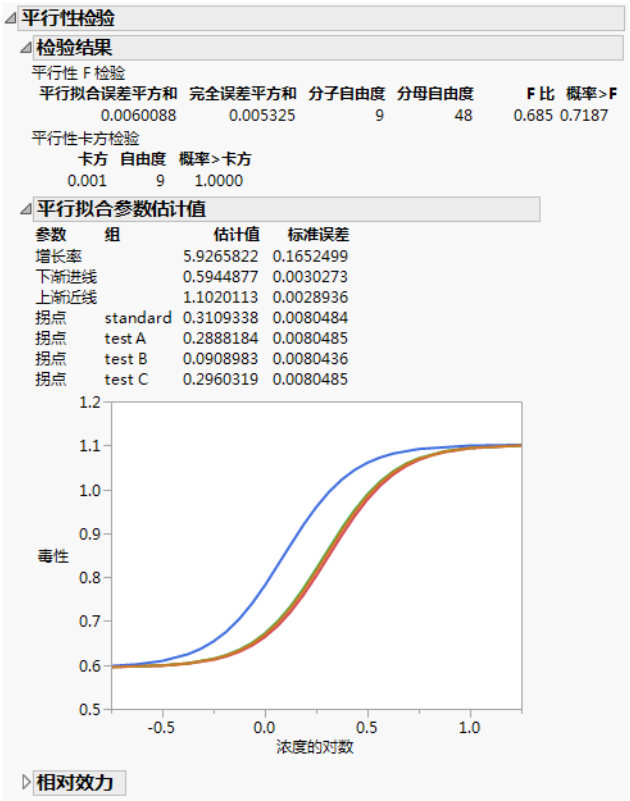
定制逆预测 预测特定 *Y* 值对应的 *X* 值。有关逆预测的详细信息，请参见 《拟合线性模型》。

删除拟合 删除模型报表、“模型比较”报表中的条目，并从图中删除拟合线。

检验平行性

“检验平行性”选项提供的分析用于检验不同的组之间拟合模型是否具有相同的形状，但拟合线会沿着 X 轴平移（图 14.9）。在 Bioassay 示例中，药物配方 B 的曲线平移到其他三条曲线的左侧。不过，您并不知道这些曲线是仍具有相同形状（平行），还是配方 B 具有不同的形状。“平行性检验”告诉我们不同的药物配方是否具有相似的形状，以及是否只是沿着水平轴的平移。从拟合模型的红色小三角菜单中选择**检验平行性**可以添加该报表。

图 14.9 平行性检验



该报表提供以下结果：

检验结果 提供针对平行性的 *F* 检验和卡方检验的结果。这些检验会比较完全模型的误差平方和 （**完全误差平方和**）和简化模型 （**拟合误差平方和**）的误差平方和。完全模型为每个组提供不同的参数。简化模型强制各组共享除拐点以外的所有参数。较小的 *p* 值指示组模型彼此之间显著不同。因此，带有共享参数的简化模型不充分，而应使用完全模型。在图 14.9 中，*p* 值大于 0.05，表示没有足够的证据可以推断出曲线之间存在差异。在这种情况下，带有共享参数的简化模型是合适的。

按以下方式计算拟合统计量：

F 比 [(拟合误差平方和 - 完全误差平方和) / NDF] / [完全误差平方和 / DDF]

卡方 拟合误差平方和 - 完全误差平方和

平行拟合参数估计值 给出简化模型下的参数估计值 （各组模型除拐点之外的参数都相同）。此外还提供了简化模型下的拟合曲线图。药物配方 B 的拐点比其他三种药物配方的拐点低得多。

相对效力 提供分组变量每个水平的效力和相对效力。效力为 10^(EC_{50})，其中 EC_{50} 是响应值位于基准值和最大值的中间位置时的浓度。对于双参数 Logistic、三参数 Logistic 和四参数

Logistic，效力为 10^（拐点参数）。相对效力是分组变量的一个水平的效力除以另一个水平的效力。例如，在图 14.10 中，test A 相对于 standard 的相对效力为效力 $_{\text{test A}}$/ 效力 $_{\text{standard}}$。

图 14.10 按组划分的相对效力

相对效力

相对效力与 standard

组	效力	相对效力	标准误差
standard	2.0461329	1	0
test A	1.9445466	1.0522416	0.0246535
test B	1.2328161	1.6597227	0.0388878
test C	1.977115	1.0349084	0.0242474

相对效力与 test A

组	效力	相对效力	标准误差
standard	2.0461329	0.950352	0.0222663
test A	1.9445466	1	0
test B	1.2328161	1.5773209	0.036957
test C	1.977115	0.9835273	0.0230435

相对效力与 test B

组	效力	相对效力	标准误差
standard	2.0461329	0.6025103	0.014117
test A	1.9445466	0.6339864	0.0148545
test B	1.2328161	1	0
test C	1.977115	0.6235429	0.0146098

相对效力与 test C

组	效力	相对效力	标准误差
standard	2.0461329	0.9662691	0.0226392
test A	1.9445466	1.0167486	0.0238219
test B	1.2328161	1.6037388	0.037576
test C	1.977115	1	0

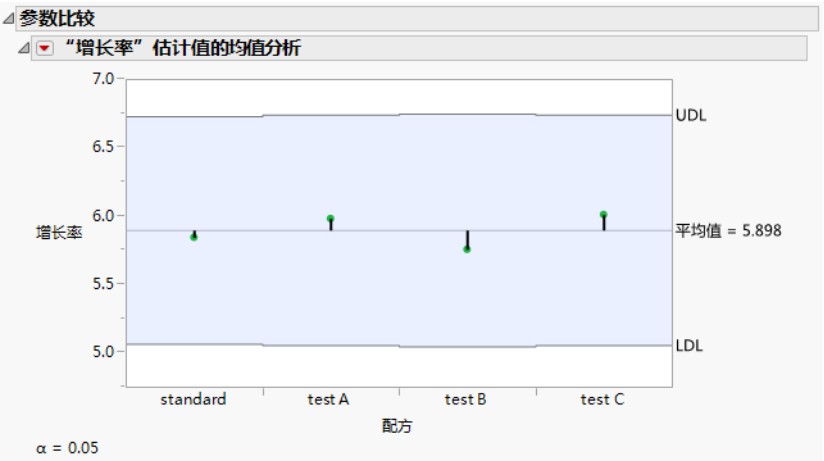
在图 14.10 中的**相对效力与 standard** 面板中，注意到药物配方 A 和 C 的相对效力都接近 1。这表明它们的效力类似于标准配方的效力。药物配方 B 的效力低于标准配方的效力。这意味着药物配方 B 在毒性 （作为浓度的函数）的增长速度上要比标准配方快。

在平行性检验中，所有曲线是平行的，这样您便可以计算相对效力。基于相对效力，您可以得出结论：配方 B 的相对效力比其他药物配方都要大。再加上之前的研究结果，药物配方 B 看起来毒性更大。

比较参数估计值

“比较参数估计值”报表提供检验参数在分组变量各水平之间是否相等的结果。每个参数都有一个“均值分析”(ANOM) 报表，用来检验参数是否等于整体均值。若某个参数的结果超过了决策限，则该参数与整体均值之间存在差异。图 14.11 显示了增长率估计值的 ANOM 报表。从拟合模型的红色小三角菜单中选择**比较参数估计值**可以添加该报表。

图 14.11 “增长率”估计值的参数比较



“均值分析”红色小三角菜单包含以下选项：

设置 Alpha 水平 设置检验的 alpha 水平。

显示汇总报表 显示或隐藏包含参数估计值、决策限以及参数是否超过限值信息的报表。

显示选项 包含用于显示或隐藏决策限、着色和中心线的选项。还包含用于更改点外观的选项。

有关“均值分析”报表的详细信息，请参见《基本分析》。

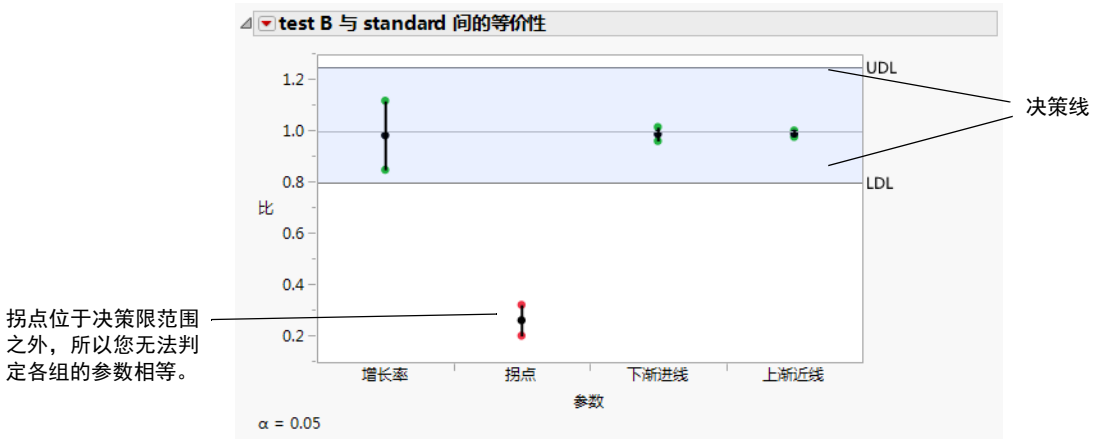
等价性检验

“等价性检验”报表提供检验模型在分组变量各水平之间是否等价的分析（图 14.12）。选择该选项后，您需要指定要针对其他各个水平进行检验的分组变量水平。之后会提供每个水平与所选水平对比的报表。从拟合模型的红色小三角菜单中选择**等价性检验**可以添加该报表。

参数是否相等通过分析参数的比值来检验。默认决策线放置在比值 0.8 和 1.25 的位置上，这表示 25% 的差异。

若所有置信区间都位于决策线的范围内，则两组模型实际上是相等的。若有一个区间落在决策线的范围之外（图 14.12），则无法判定两组模型相等。药物配方 B 比标准配方的拐点低，这与之前的研究结果一致。

图 14.12 等价性检验



“等价性”红色小三角菜单包含以下选项：

设置 Alpha 水平 设置检验的 alpha 水平。默认值为 0.05。

设置决策线 更改比值的决策限。默认值设置在比值 0.8 和 1.25 的位置上，这表示 25% 的差异。

显示汇总报表 显示或隐藏包含参数估计值、决策限以及参数是否超过限值信息的报表。

显示选项 包含用于显示或隐藏决策限、着色和中心线的选项。还包含用于更改点外观的选项。对于其他格式选项，请右击该图并选择**定制**。

“拟合曲线”平台的统计详细信息

模型公式

表 14.1 提供了 “拟合曲线”红色小三角菜单中的模型的公式。

表 14.1 “拟合曲线”模型公式

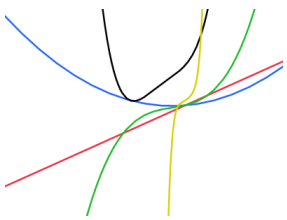
模型	公式
多项式 	$\beta_0 + \sum_{i=1}^{k} \beta_i x^i$ 其中，k 是多项式的阶数。您也可以使用“拟合模型”平台和“以 X 拟合 Y”平台拟合这些模型。

表 14.1 “拟合曲线”模型公式 (续)

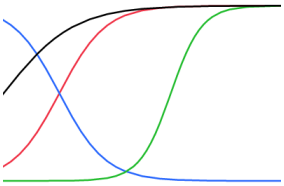
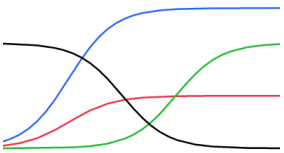
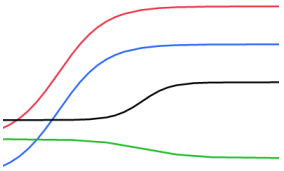
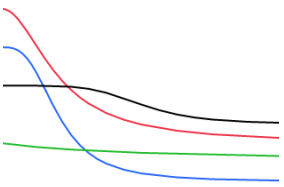
模型	公式
双参数 Logistic 	$\frac{1}{1+\text{Exp}(-a(x-b))}$ a = 增长率 b = 拐点 仅当所有响应值都介于 0 到 1 之间时才可用。
三参数 Logistic 	$\frac{c}{1+\text{Exp}(-a(x-b))}$ a = 增长率 b = 拐点 c = 渐近线
四参数 Logistic 	$c+\frac{d-c}{1+\text{Exp}(-a(x-b))}$ a = 增长率 b = 拐点 c = 下渐近线 d = 上渐近线
四参数 Logistic Rodbard 	$c+\frac{d-c}{1+(x/b)^a}$ a = 增长率 b = 拐点 c = 下渐近线 d = 上渐近线 仅当回归变量值为正时才可用。

表 14.1 “拟合曲线”模型公式 (续)

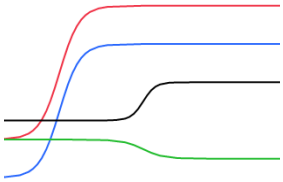
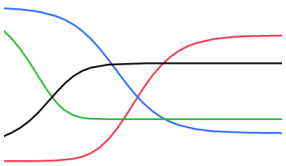
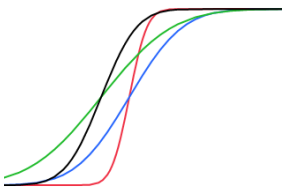
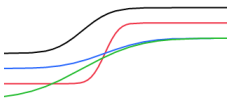
模型	公式
四参数 Logistic Hill 	$c + \frac{d-c}{1+10^{(-a(x-b))}}$ a = 增长率 b = 拐点 c = 下渐近线 d = 上渐近线
五参数 Logistic 	$c + \frac{d-c}{(1+\text{Exp}(-a(x-b)))^f}$ a = 增长率 b = 拐点 c = 渐近线 1 d = 渐近线 2 f = 幂次
双参数 Probit 	$\Phi\left(\frac{x-b}{a}\right)$ a = 增长率 b = 拐点 Φ = 正态分布 CDF 仅当所有响应值都介于 0 到 1 之间时才可用。
四参数 Probit 	$c + (d-c) \cdot \Phi\left(\frac{x-b}{a}\right)$ a = 增长率 b = 拐点 c = 渐近线 1 d = 渐近线 2 Φ = 正态分布 CDF

表 14.1 “拟合曲线”模型公式 (续)

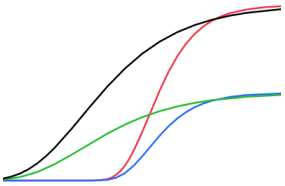
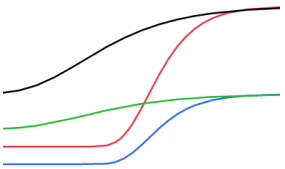
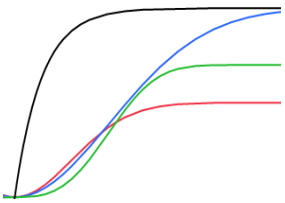
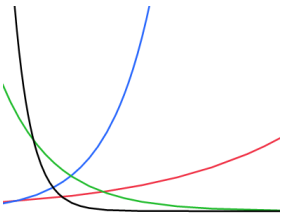
模型	公式
三参数 Gompertz 	$a\text{Exp}(-\text{Exp}(-b(x-c)))$ a = 渐近线 b = 增长率 c = 拐点
四参数 Gompertz 	$a+(b-a)\text{Exp}(-\text{Exp}(-c(x-d)))$ a = 下渐近线 b = 上渐近线 c = 增长率 d = 拐点
Weibull 生长 	$a(1-\text{Exp}(-(x/c)^{b}))$ a = 上渐近线 b = 增长率 c = 拐点 仅当响应值和回归变量值均为非负值时才可用。
双参数指数 	$a\text{Exp}(bx)$ a = 尺度 b = 增长率

表 14.1 “拟合曲线”模型公式 (续)

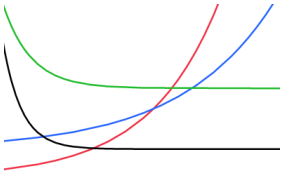
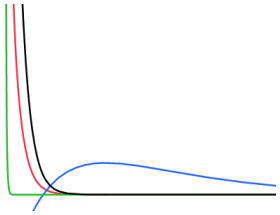
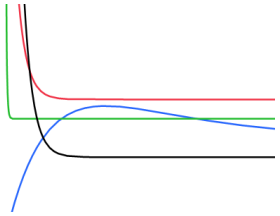
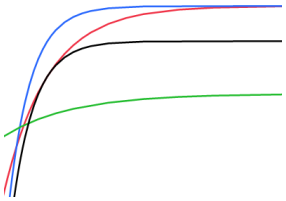
模型	公式
三参数指数 	$a + b\text{Exp}(cx)$ a = 渐近线 b = 尺度 c = 增长率
四参数双指数 	$a\text{Exp}(-bx) + c\text{Exp}(-dx)$ a = 尺度 1 b = 衰减率 1 c = 尺度 2 d = 衰减率 2 仅当响应值为正时才可用。
五参数双指数 	$a + b\text{Exp}(-cx) + d\text{Exp}(-fx)$ a = 渐近线 b = 尺度 1 c = 衰减率 1 d = 尺度 2 f = 衰减率 2
机理生长 	$a(1 - b\text{Exp}(-cx))$ a = 渐近线 b = 尺度 c = 增长率

表 14.1 “拟合曲线”模型公式 (续)

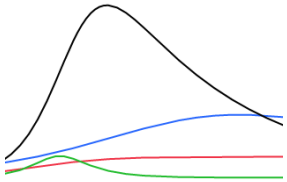
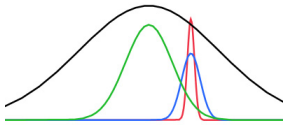
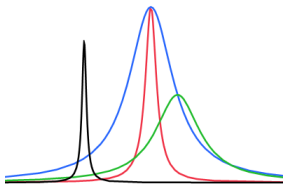
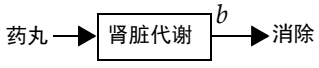
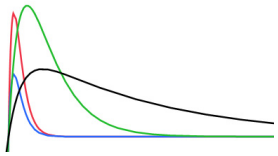
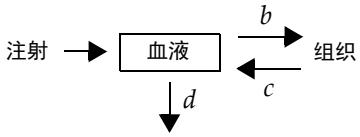
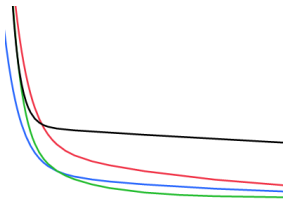
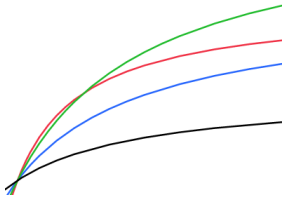
模型	公式
四参数细胞生长 	$\frac{ab}{(a-b)\text{Exp}(-cx)+b\text{Exp}(dx)}$ a = 峰值（若为死亡比率）d 为零 b = 时间 0 处的响应 c = 单元格分区比率 d = 单元格死亡比率 仅当响应值为正且回归变量值非负时才可用。
高斯峰 	$a\text{Exp}\left(-\frac{1}{2}\left(\frac{x-b}{c}\right)^2\right)$ a = 峰值 b = 临界点 c = 增长率
Lorentzian 峰 	$\frac{ab^2}{(x-c)^2+b^2}$ a = 峰值 b = 增长率 c = 临界点
单室口服剂量模型  	$\frac{abc}{c-b}(\text{Exp}(-bx)-\text{Exp}(-cx))$ a = 曲线下面积 b = 消除率 c = 吸收率 仅当响应值和回归变量值全为正值时才可用。

表 14.1 “拟合曲线”模型公式 (续)

模型	公式
双室快速静脉注射剂量模型  	$\frac{a}{\alpha-\beta}((\alpha-b)\mathrm{Exp}(-\alpha x)-(\beta-b)\mathrm{Exp}(-\beta x))$ $\alpha=\frac{1}{2}(b+c+d+\sqrt{(b+c+d)^2-4bd})$ $\beta=\frac{1}{2}(b+c+d-\sqrt{(b+c+d)^2-4bd})$ a = 初始浓度 b = 入向传输率 c = 出向传输率 d = 消除率 仅当响应值和回归变量值全为正值时才可用。
Michaelis Menten 	$\frac{ax}{b+x}$ a = 最大反应率 b = 逆亲和性 仅当响应值和回归变量值全为正值时才可用。

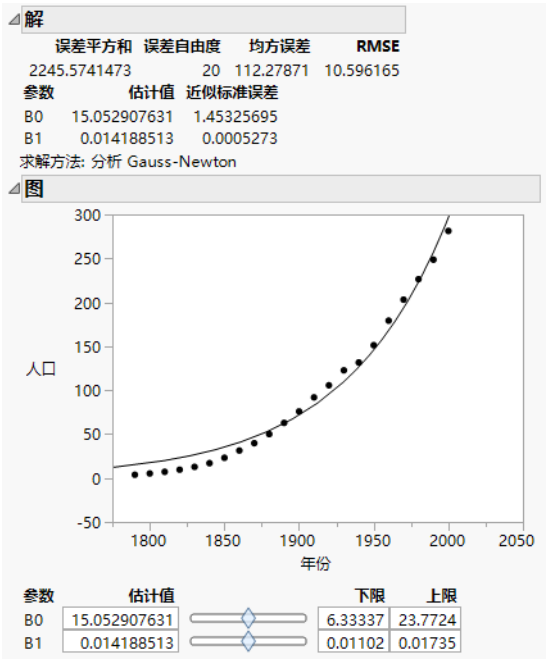
第 15 章

非线性回归

将数据拟合为定制的非线性模型

“非线性”平台适用于拟合参数具有**非线性**关系的模型。本章重点介绍定制的非线性模型，它们包括一个模型公式和几个要估计的参数。您可以使用默认的最小二乘损失函数或定制损失函数来拟合模型。该平台使不同观测的损失函数之和最小。

图 15.1 定制非线性拟合的示例



“非线性”平台还提供许多预定义的模型，如多项式、Logistic、Gompertz、指数、峰值和药代动力学模型。请参见第 193 页的“拟合曲线”一章。

注意：某些模型的参数是**线性**关系的（例如，二次式或其他多项式），或者可以变换为线性关系（例如，在您使用 x 的对数变换时）。“拟合模型”或“以 X 拟合 Y”平台在这些情形下更适用。有关这些平台的详细信息，请参见《拟合线性模型》和《基本分析》。

目录

拟合定制模型的示例 217
启动“非线性”平台 219
“非线性拟合”报表 220
“非线性”平台选项 224
使用模型库创建公式 226
更多示例 230
最大似然的示例：Logistic 回归 230
具有二项误差的 Probit 模型示例：数值导数 232
Poisson 损失函数的示例 233
设置参数限值的示例 235
“非线性”平台的统计详细信息 238
边侧似然置信限 239
定制损失函数的工作机制 239
关于导数的注释 240
关于有效的非线性建模的注释 242

拟合定制模型的示例

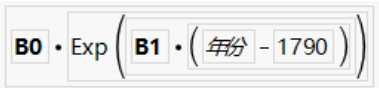
要拟合定制模型，您必须首先创建一个具有初始参数估计值的模型列。该方法需要的步骤比拟合内置模型所需的步骤多一点，但是它允许拟合任何非线性模型。此外，您可以自定义损失函数，并指定用于拟合过程的几个其他选项。

本节提供一个示例，介绍如何创建模型的公式列并在“非线性”平台中拟合模型。数据位于 US Population.jmp 数据表中。响应变量为美国人口 （以百万为单位），预测变量为年份。

1. 选择**帮助** > **样本数据库**，然后打开 Nonlinear Examples/US Population.jmp。
2. 创建名为**模型**的新列。
3. 右击**模型**列，然后选择**列属性** > **公式**。

   随即显示 “公式编辑器”。
4. 从列列表下方的列表中选择**参数**。
5. 选择**新建参数**。
6. 键入 B0 作为名称。
7. 键入 3.9 作为值。该值是初始参数估计值。
8. 点击**确定**。
9. 选择**新建参数**。
10. 键入 B1 作为名称，输入 0.022 作为值。
11. 点击**确定**。
12. 使用 “公式编辑器”函数、列**年份**和参数来输入模型公式。

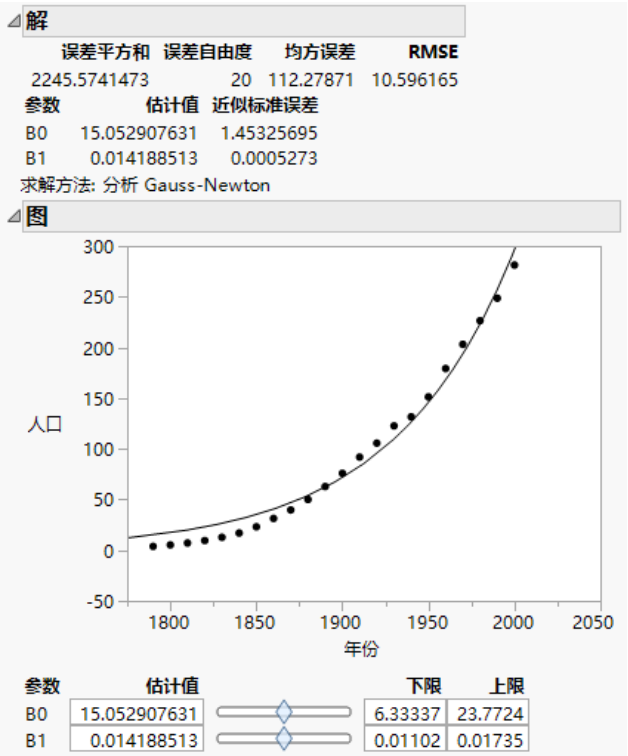
图 15.2 完成的模型公式



提示：点击 “超越”旁边的红色小三角以查找指数命令。

13. 点击**确定**。
14. 选择**分析** > **专业建模** > **非线性**。
15. 将**模型**分配给 **X，预测变量公式**角色。
16. 将**人口**分配给 **Y，响应**角色。
17. 点击**确定**。
18. 点击控制面板上的**执行**以拟合模型。

图 15.3 图和 “解” 报表



最终的参数估计值与其他拟合统计量一起显示在**解**报表中。拟合模型显示在图上。

包含分组变量的模型的参数

在公式编辑器中，当您添加参数时，可以看到**扩展到类别，选择列**复选框。该选项用于同时添加若干参数（例如，为分类变量的每个水平各添加一个参数）。当您选择该选项后，将显示一个对话框，您可以在其中选择一列。完成选择之后，“参数”列表中会显示一个新的参数，其名称为 D_列，其中 D 是您为参数指定的名称。在公式中使用该参数时，将插入一个 Match 表达式，它为分类变量的每个水平包含一个单独的参数。

启动“非线性”平台

要启动“非线性”平台，请选择**分析 > 专业建模 > 非线性**。

图 15.4 “非线性”平台启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“非线性”平台启动窗口具有以下功能：

Y，响应 选择 *Y* 变量。

X，预测变量公式 选择 *X* 变量或包含带有参数的模型公式列。

> **注意：**若您选择的列不包含带参数的模型公式而且您未指定定制公式，则改为启动“拟合曲线”平台。

> **注意：**若您选择的公式列包含未赋值的参数列表项，JMP 会将参数自动设置为 0 并在日志中显示一条消息。

分组 指定分组变量。分组变量的不同水平下的拟合模型包含不同的参数。这样您就可以比较分组变量各水平下的拟合模型和估计参数。

注意：若公式中每个组水平并未带有不同的参数，则 JMP 会忽略分组变量。

权重 指定包含观测权重的变量。

频数 指定表示观测频数的变量。

损失 指定提供损失函数的公式列。

依据 指定一个变量，为该变量的每个水平分别进行分析。

模型库 启动“模型库”工具，它帮助您选择初始值来创建公式列。请参见第 226 页的“使用模型库创建公式”。

拟合定制公式的选项 若在“X，预测变量公式”角色中指定的列不包含带参数的模型公式，则可以使用该选项创建定制公式。在“预测变量”字段中，指定参数、参数值和使用选定的 *X* 变量的定制公式。然后，点击**重置**按钮并启动该平台。

仅限于数值导数 仅使用数值导数。当您的模型太零乱而难以进行分析求导时，该选项很有用。该选项对于在困难情形下实现收敛也很有用。仅当在“X，预测变量公式”角色中提供公式列时，才使用该选项。

展开中间公式 若组成公式的成分列本身就含有公式，只要它引用了其他列，在保存公式时则会替换内部公式。为防止成分列展开，请设置该列的**其他**列属性，将属性命名为“展开公式”且值为 0。仅当在“X，预测变量公式”角色中提供公式列时，才使用该选项。

“非线性拟合”报表

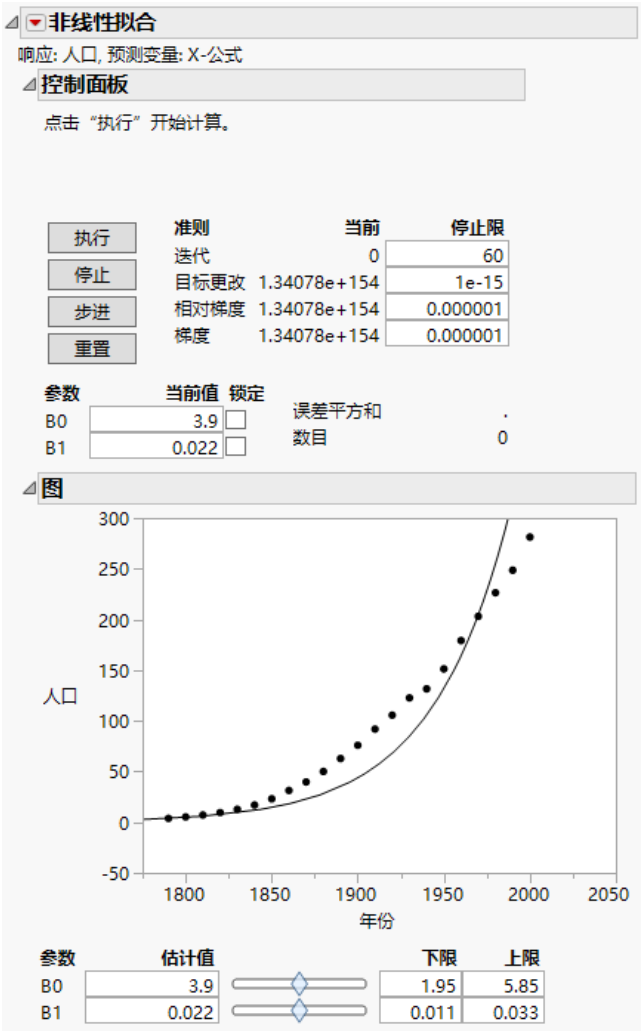
初始“非线性拟合”报表包含以下各项，如图 15.5 中所示。

控制面板 提供用于控制拟合过程的选项。

- **执行** 开始执行拟合。
- **停止** 停止拟合。
- **步进** 一步使拟合过程执行一次迭代。
- **重置** 重置公式中的可编辑值、重置迭代值并根据这些新值计算误差平方和。
- **准则** 显示拟合过程的迭代测度。
- **当前** 显示每个准则的当前值。
- **停止限** 设置“准则”下所列测度的限值。

图 为仅包含一个 *X* 变量的模型显示 *X* 和 *Y* 变量的图。图上显示当前参数估计值对应的模型。要更改参数的当前值，请使用图下方的滑块或编辑框。

图 15.5 初始 “非线性拟合”报表



点击**执行**以拟合模型之后，该报表还会包含以下项，如图 15.6 中所示。

保存估计值 将当前参数值保存到公式列中的参数。

置信限 计算所有参数的置信区间。区间是边侧似然置信区间，它们显示在 “解”报表中。针对每个参数的每次置信限的计算，会涉及一组新的迭代，这些迭代往往并不能成功找到置信限。**编辑 Alpha** 和**收敛准则**选项用于置信区间计算。有关**置信限的目标误差平方和**的详细信息，请参见第 239 页的 “边侧似然置信限”。

解 显示参数估计值和其他统计量。

误差平方和 显示误差平方的残差和。目标是要使误差平方和最小。若指定了定制损失函数，则该值是损失函数之和。

误差自由度 显示误差的自由度，它是使用的观测数减去拟合参数个数后的数值。

均方误差 显示均方误差。它是剩余误差的方差估计值，通过将误差平方和除以误差自由度得到。

RMSE 估计剩余误差的标准差，它是均方误差的平方根。

R 方 （仅当模型具有截距且残差之和接近零时才可用。）显示 R 方值。

参数 列出您在拟合公式中为参数指定的名称。

估计值 列出生成的参数估计值。请记住使用非线性回归时，即使一切正常该估计值也可能有问题。

近似标准误差 列出近似标准误差，它以近似于线性回归的方式计算得到。它通过均方根误差与导数叉积逆矩阵的对角线元素的平方根相乘得到。

注意：若观测数小于等于拟合模型中的参数数目，则不报告近似标准误差。

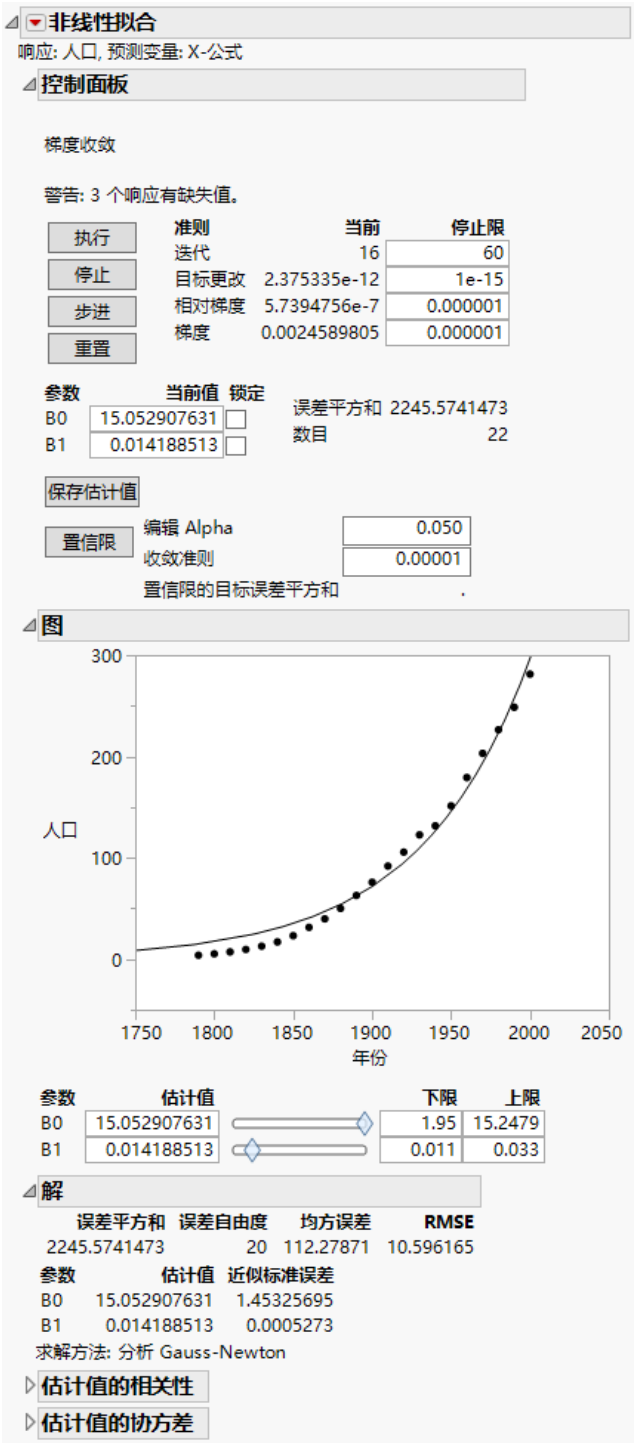
置信下限和置信上限 显示参数的置信限。您在控制面板上点击**置信限**之前它们是不显示的。有关置信区间的详细信息，请参见第 239 页的“边侧似然置信限”。

已排除数据 显示一个报表，其中列出排除的行的拟合统计量。使用未参与模型构建的观测进行模型验证时，该选项会很有用。您可以将该功能与“记住解”选项一起使用来排除不同的数据，然后得到关于新的排除数据的相关报表。

估计值的相关性 显示参数估计值之间的相关性。

估计值的协方差 显示参数估计值之间的协方差。

图 15.6 “拟合模型”报表


非线性拟合
响应: 人口, 预测变量: X-公式
控制面板
梯度收敛
警告: 3 个响应有缺失值。
执行
停止
步进
重置
准则 当前 停止限
迭代 16 60
目标更改 2.375335e-12 1e-15
相对梯度 5.7394756e-7 0.000001
梯度 0.0024589805 0.000001
参数 当前值 锁定
B0 15.052907631
B1 0.014188513
误差平方和 2245.5741473
数目 22
保存估计值
置信限
编辑 Alpha 0.050
收敛准则 0.00001
置信限的目标误差平方和 .
图
300
200
100
0
人口
1750 1800 1850 1900 1950 2000 2050
年份
参数 估计值 下限 上限
B0 15.052907631 1.95 15.2479
B1 0.014188513 0.011 0.033
解
误差平方和 误差自由度 均方误差 RMSE
2245.5741473 20 112.27871 10.596165
参数 估计值 近似标准误差
B0 15.052907631 1.45325695
B1 0.014188513 0.0005273
求解方法: 分析 Gauss-Newton
估计值的相关性
估计值的协方差

“非线性”平台选项

“非线性拟合”红色小三角菜单包含以下选项：

参数界限 设置参数的界限。选择该选项后，在控制面板中会显示可编辑的框。将该字段留空可表示无界限参数。

图 为仅包含一个 *X* 变量的模型显示或隐藏 *X* 和 *Y* 变量的图。图上显示的是当前参数估计值对应的模型。要更改参数的当前值，请使用图下方的滑块或编辑框。若您在启动时指定了“分组”变量，则为每个组显示一条曲线。

迭代选项 指定拟合算法的选项。

迭代日志 在新窗口中记录拟合过程的每个步骤。

仅限于数值导数 适用于您的模型太繁杂而难以进行分析求导的情况。该选项对于在困难情形下实现收敛也很有用。

展开中间公式 若组成公式的成分列本身就含有公式，只要它引用了其他列，告知 JMP 在保存公式时替换内部公式。为防止成分列展开，请设置该列的**其他**列属性，将属性命名为“展开公式”且值为 0。

牛顿算法 指定 Gauss-Newton（用于常规最小二乘法）或 Newton-Raphson（用于具有损失函数的模型）为优化方法。

SR1 拟牛顿算法 指定 QuasiNewton SR1 为优化方法。

BFGS 拟牛顿算法 指定 QuasiNewton BFGS 为优化方法。

接受当前估计值 告知 JMP 使用当前估计值生成解报表，即使估计值不收敛。

显示导数 在 JMP 日志中显示非线性公式的导数。有关导数的技术信息，请参见第 240 页的“关于导数的注释”。

非线程 在主计算线程中运行迭代。在多数情况下，JMP 在单独的计算线程中执行计算。这改进了 JMP 在执行非线性计算期间又执行其他操作时的响应能力。但是，在一些个别情形中（例如，某些模型有副作用，会调用显示例程）迭代应在主线程中运行，因此应启用该选项。

刻画器 提供多种刻画器来查看响应曲面。

刻画器 显示预测刻画器。“刻画器”使您可以依次查看曲面在每个 *x* 变量上的垂直切面，同时找到每个因子的最优值。

等高线刻画器 显示等高线刻画器。“等高线刻画器”使您可以查看二维等高线以及三维网格图。

曲面刻画器 创建三维曲面图。该选项仅适用于具有两个或更多变量的模型。

参数刻画器 显示预测刻画器，并将误差平方和或损失刻画为参数的函数。

参数等高线刻画器 显示等高线刻画器，并将误差平方和或损失刻画为参数的函数。

参数曲面刻画器 创建三维曲面图，并将误差平方和或损失刻画为参数的函数。该选项仅适用于具有两个或更多参数的模型。

误差平方和网格 创建一个由解估计值附近的值构成的网格，并计算每个值对应的误差平方和。解估计值应最具有最小的误差平方和。选择该选项后，会显示包含以下功能的**指定用于输出的网格**报表：

参数 列出模型中的参数。

最小值 显示网格计算中使用的最小参数值。默认情况下，**最小值**是解估计值减去 2.5 倍的**近似标准误差**。

最大值 显示网格计算中使用的最大参数值。默认情况下，**最大值**是解估计值加上 2.5 倍的**近似标准误差**。

点数 给出要为每个参数创建的点数。要计算新网格表中的总点数，请将所有“点数”值相乘。前两个参数的初始“点数”是 11，其余参数的初始“点数”是 3。若您要指定新值，请使用奇数值以确保网格表包括解估计值。对于任何参数将“点数”设置为 0 只会在网格表中记录解估计值。

当您点击**执行**时，JMP 会在新表中创建一个由点构成的网格。若解在表中，相应的解估计所在的行会突出显示并加以标记。

恢复原始参数 将平台重置为原始参数值 （公式列参数中给出的值）。

记住解 创建名为 “记住的模型”的报表，它包含当前参数估计值和汇总统计量。可以记住多个模型的结果并对它们进行比较。若您要比较基于不同的参数限制的模型或比较使用不同选项拟合的模型，该选项会很有用。点击特定模型对应的单选按钮，即可在“图”中显示该模型以及在 “控制面板”中的参数估计值。

定制估计 给出参数函数的估计值。您需要提供只涉及参数的表达式。 JMP 会使用当前参数估计值计算该表达式，还会使用一阶 Taylor 级数近似计算表达式的标准误差。

定制逆预测 在给定 *Y* 值的情况下估计 *X* 值。它还计算估计的 *X* 值的标准误差。JMP 必须能求出逆模型。标准误差基于使用反转表达式的一阶 Taylor 级数近似。置信区间是 Wald 区间，根据 *t* 分位数和标准误差计算。

显示预测表达式 在报表顶部显示预测模型或损失函数。

保存预测置信限 保存模型预测的渐近置信限。即针对给定的 *X* 值计算 *Y* 平均值的置信区间。

保存单值置信限 保存单值预测的渐近置信限。即针对给定的 *X* 值计算单个 *Y* 值的置信区间。

保存公式 给出用于将模型结果保存到数据表列的选项：

保存预测公式 保存具有当前参数估计值的预测公式。

保存预测值标准误差 保存模型预测的标准误差。这是针对给定的 *X* 值预测 *Y* 平均值的标准误差。公式采用以下形式：`Sqrt(VecQuadratic(matrix1,vector1))`。`matrix1` 是与参数估计值关联的协方差矩阵，`vector1` 是模型的对于每个参数的偏导数的组合。

保存单个预测值的标准误差 保存单个预测值的标准误差。这是针对给定的 *X* 值预测单个 *Y* 值的标准误差。公式采用以下形式：`Sqrt(VecQuadratic(matrix1,vector1)+mse)`。`matrix1` 是与参数估计值关联的协方差矩阵，`vector1` 是模型的对于每个参数的偏导数的组合，`mse` 是误差方差的估计值。

保存残差公式 保存计算残差的公式。

保存预测置信限公式 保存计算模型预测的置信区间的公式。即针对给定的 X 计算 Y 平均值的置信区间。

保存单值置信限公式 保存计算单个预测值的置信区间的公式。即针对给定的 X 计算单个 Y 值的置信区间。

保存逆预测公式 保存计算以下值的公式：模型的逆预测、逆预测的标准误差和单值逆预测的标准误差。

保存特定求解公式 等价于简单情况下的“保存逆预测公式”。但是，该命令允许公式为几个变量的函数并允许替换表达式。该功能仅限于只在公式中出现一次的算子和函数，比较容易求出可逆函数。

在选择该命令后，将显示一个对话框，允许您选择进行求解的变量。您还可以编辑生成的表中列的名称。您还可以在该对话框中替换名称的值。在这些情况下，会为那些替换的值求解公式。

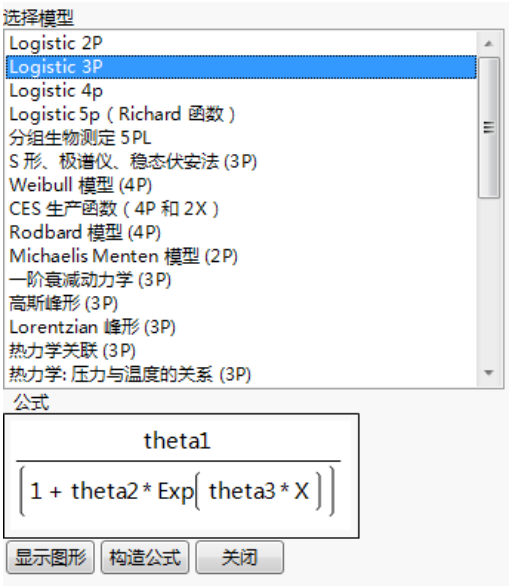
注意：仅当执行最小二乘估计或使用最大似然估计并且负对数似然值合理时，标准误差、置信区间和假设检验才正确。

将估计值保存至表 将参数估计值保存到新的数据表中。

使用模型库创建公式

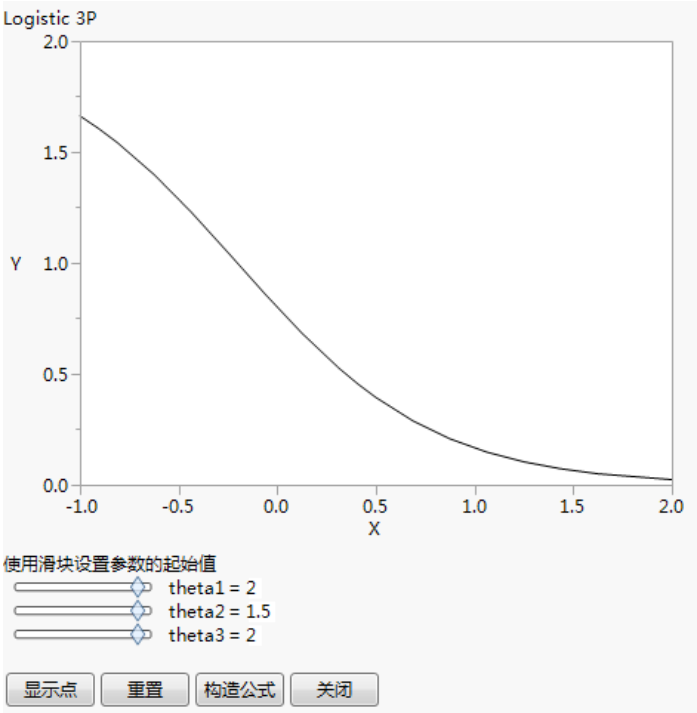
“模型库”可帮助您创建使用参数和初始值的公式列。点击“非线性”启动窗口上的**模型库**按钮可打开模型库。选择列表中的某个模型即可在**公式**框中查看其公式。

图 15.7 “非线性模型库”对话框



点击**显示图形**可为单参数模型显示二维理论曲线，为双参数模型显示三维曲面图。对于包含两个以上说明 (X) 变量的模型，将不提供图形。在图形窗口上，可以使用滑块更改参数的默认初始值，或是直接点击并输入值。

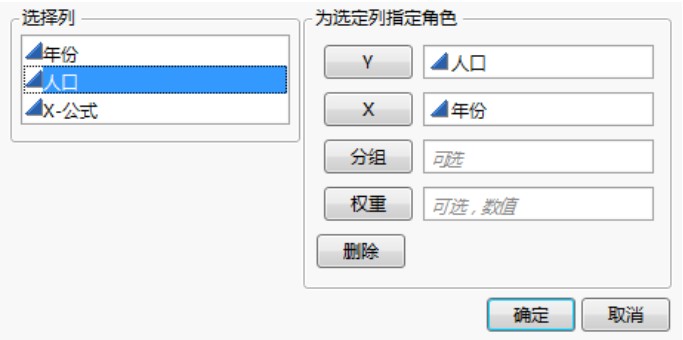
图 15.8 “模型库”中的图形示例



重置按钮用于将参数的初始值恢复为默认值。

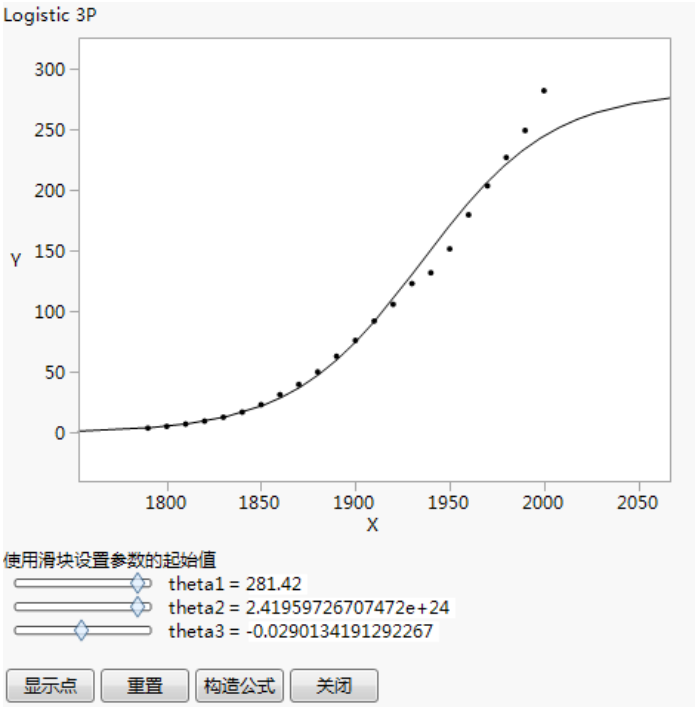
点击**显示点**可将实际数据点叠加到图上。图 15.9 中所示的对话框将打开，要求您给 X 和 Y 角色以及可选的“分组”角色分配列。“分组”角色允许您为分类变量的每个水平拟合模型。若您在此处指定了“分组”角色，则还要在平台启动窗口上指定“分组”列。

图 15.9 选择角色



对于大多数模型，起始值是常数。图中的散点有助于您调整参数值，以查看使用不同的参数值时模型拟合的好坏程度。对于 US population 数据表示例，图 15.10 中显示了这些点。

图 15.10 显示点



此时（在使用**显示点**后）点击**构造公式**将创建一个新数据表列，它会以您从“模型库”中选择的模型命名。该列包含一个公式，对应的参数值为最新设置的起始值。

注意：若您在使用**显示图形**或**显示点**按钮前点击**构造公式**，系统将要求您提供 *X* 和 *Y* 角色以及可选的“分组”角色（图 15.9）。在此之后，您将返回到图中，以便可以根据需要调整参数起始值。此时再次点击**构造公式**将创建新列。

在数据表中创建公式后，通过在“非线性”启动对话框中将新列分配为 **X，预测变量公式**即可继续进行分析。

定制非线性模型库

“模型库”是使用名为 NonlinLib.jsl 的内置脚本创建的，该脚本位于 JMP (Windows) 或应用程序包 (macOS) 所在文件夹中的 Resources/Builtins 文件夹中。您可以通过修改该脚本来定制非线性库脚本。

要添加模型，您必须在名为 `Listofmodellist#` 的列表中添加三行。这三行本身实际上是一个列表，它包含以下三部分：

- 模型名称，一个带引号的字符串
- 模型公式，一个表达式
- 模型尺度

例如，假定您要添加具有以下形式的名为 “Simple Exponential Growth”的模型：

$$y = b_1 e^{kx}$$

将以下行添加到 NonlinLib.jsl 脚本

```
{//Simple Exponential Growth
     "Simple Exponential Growth",
     Expr(Parameter({b1=2, k=0.5}, b1*exp(k * :X))),
     lowx = -1; highx = 2; lowy = 0;  highy = 2},
```

以下是要注意的一些事项：

- 第一行只有左括号（表示列表开头）和可选注释。第二行是在模型库窗口中显示的字符串。
- `lowx`、`highx`、`logy` 和 `highy` 的值指定理论图形的初始窗口。
- 上例中有一个逗号作为最后一个字符。若这是 `Listofmodellist#` 列表中的最后输入项，则可以省略逗号。
- 若模型使用两个以上的参数，请将最后一行（包含图形限值）替换为带引号的字符串“String Not Available”。

  要删除模型，请从 `Listofmodellist#` 列表中删除相应的三行列表。

更多示例

- “最大似然的示例：Logistic 回归”
- “具有二项误差的 Probit 模型示例：数值导数”
- “Poisson 损失函数的示例”
- “设置参数限值的示例”

最大似然的示例：Logistic 回归

该示例显示如何使用“非线性”平台将损失函数最小化。损失函数是负对数似然函数，因此生成最大似然估计值。

Nonlinear Examples 样本数据文件夹中的 Logistic w Loss.jmp 数据表包含一个使用损失函数拟合 Logistic 回归的示例。Y 列中 1 表示事件，0 表示事件没有发生。**模型** Y 列具有线性模型，**损失**列具有损失函数。在该示例中，损失函数是每个观测值的负对数似然，或观测到响应值的概率的负对数。

按以下步骤运行模型：

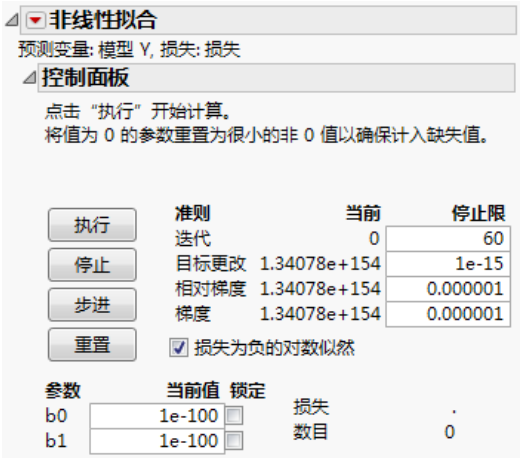
1. 选择**帮助** > **样本数据库**，然后打开 Nonlinear Examples/Logistic w Loss.jmp。
2. 选择**分析** > **专业建模** > **非线性**。
3. 将**模型** Y 分配给 **X，预测变量公式**角色。
4. 将**损失**分配给**损失**角色。

图 15.11 “非线性”启动窗口

5. 点击**确定**。

“非线性拟合”控制面板随即显示。

图 15.12 “非线性拟合”控制面板



6. 点击**执行**。

   参数估计值显示在 “解”报表中。

图 15.13 “解”报表

解

	损失	误差自由度	平均损失	均方根损失
	7.8070527905	18	0.4337252	0.6585781

参数	估计值	近似标准误差
b0	-5.511451052	2.43728165
b1	0.0347553478	0.01595111

求解方法: 分析 NR

“解”报表中的“损失”值是在参数估计值处求得的负对数似然。

具有二项误差的 Probit 模型示例：数值导数

Ingots2.jmp 样本数据表包含用不同加热时间和浸泡时间处理后测试其可用性的模具数。响应变量**可用数量**是二项变量，它取决于测试的模具数 （**总数**）以及加热和浸泡时间。使用以下方法获得具有二项误差的 probit 模型参数的最大似然估计值：

- 数值导数
- 负对数似然作为损失函数
- Newton-Raphson 方法。

可用的平均模具数是测试的模具数和在给定的加热和浸泡时间下模具可用的概率之积。使用 probit 模型， P 列包含以下模型公式：

Normal Distribution(**b0**+**b1*** 加热时间 +**b2*** 浸泡时间)

Normal Distribution 函数的参数是关于处理的线性模型。

要指定二项误差，损失函数**损失**具有以下公式：

-(可用数量 *Log(*p*) + (总数 - 可用数量)*Log(1 - *p*))

请按照以下步骤拟合模型：

1. 选择**帮助** > **样本数据库**，然后打开 Ingots2.jmp。
2. 选择**分析** > **专业建模** > **非线性**。
3. 将 P 分配给 **X，预测变量公式**角色。
4. 将**损失**分配给**损失**角色。
5. 选择**仅限于数值导数**选项。
6. 点击**确定**。
7. 点击**执行**。

平台使用了 “数值 SR1”方法来获取参数估计值，如图 15.14 中所示。

图 15.14 Ingots2 数据的解

解

	损失	误差自由度	平均损失	均方根损失
	47.479945327	16	2.9674966	1.7226423

参数	估计值	近似标准误差
b0	-2.8934153	0.51256572
b1	0.0399554554	0.01202329
b2	0.0362537934	0.15017139

求解方法: 数值 SR1

Poisson 损失函数的示例

Poisson 分布通常用于对计数数据建模。

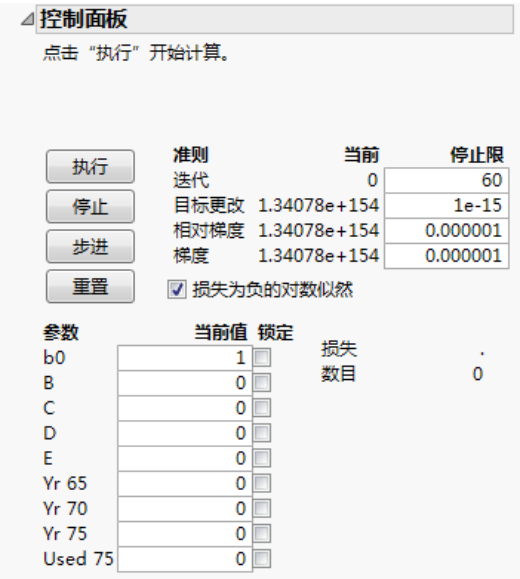
$$P(Y = n) = \frac{e^{-\mu}\mu^{n}}{n!}, n = 0, 1, 2, \ldots$$

其中 μ 可以为单个参数，也可以为具有多个参数的线性模型。很多文章和论文说明了如何变换该模型，并通过迭代方式进行加权最小二乘法来拟合该模型 (Nelder and Wedderburn 1972)。但是，在 JMP 中，直接拟合模型更简单。例如，McCullagh and Nelder (1989) 说明了如何分析报告的波浪导致的货船损坏事故数。

数据位于 Ship Damage.jmp 样本数据表中。模型公式保存在**模型**列中，损失函数（或负对数似然）在 Poisson 列中。要拟合模型，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Ship Damage.jmp。
2. 选择**分析** > **专业建模** > **非线性**。
3. 将**模型**分配给 **X，预测变量公式**角色。
4. 将 Poisson 分配给**损失**角色。
5. 点击**确定**。
6. 将 b0 的**当前值**（初始值）设置为 1，其他参数设置为 0。

图 15.15 输入新参数



7. 点击**执行**。
8. 点击**置信限**按钮。

“解”报表随即显示。 结果包括参数估计值和置信区间以及其他汇总统计量。

图 15.16 Poisson 损失示例的 “解”表

解

损失	误差自由度	平均损失	均方根损失	
68.281234087	25	2.7312494	1.6526492	
参数	**估计值**	**近似标准误差**	**置信下限**	**置信上限**
b0	-6.405914771	0.21744445	-6.8430512	-5.9896833
B	-0.543353982	0.17758996	-0.881379	-0.183552
C	-0.687418388	0.32904722	-1.3764541	-0.0745299
D	-0.075979835	0.2905789	-0.6715296	0.47523936
E	0.325581683	0.23587934	-0.143451	0.78520431
Yr 65	0.6971496668	0.1496412	0.40752821	0.99512758
Yr 70	0.8184263427	0.16977314	0.48728046	1.15369087
Yr 75	0.4534435316	0.23317069	-0.0123098	0.90388895
Used 75	0.3844880718	0.11827198	0.15341509	0.61742312

求解方法: 分析 NR

注意：仅当执行最小二乘估计或使用最大似然估计并且负对数似然值合理时，标准误差、置信区间和假设检验才正确。

设置参数限值的示例

“拟合曲线”特质允许您拟合模型，然后在整个“非线性”平台中使用该预测方程。该方法需要更多步骤和用户输入，但是允许拟合任何非线性模型。

完成“拟合曲线”一章中第 195 页的“使用“拟合曲线”平台的示例”以拟合模型。该示例说明如何从“拟合曲线”保存预测公式，然后在“非线性”中设置参数限值。

1. 点击“四参数 Logistic”红色小三角并选择**保存公式 > 保存带参数的预测公式**。

   名为**预测变量 : 毒性**的新列显示在数据表中。
2. 选择**分析 > 专业建模 > 非线性**。
3. 将**毒性**分配给 **Y，响应**角色。
4. 将**预测变量 : 毒性**分配给 **X，预测变量公式**角色。
5. 将**配方**分配给**分组**角色。
6. 点击**确定**。

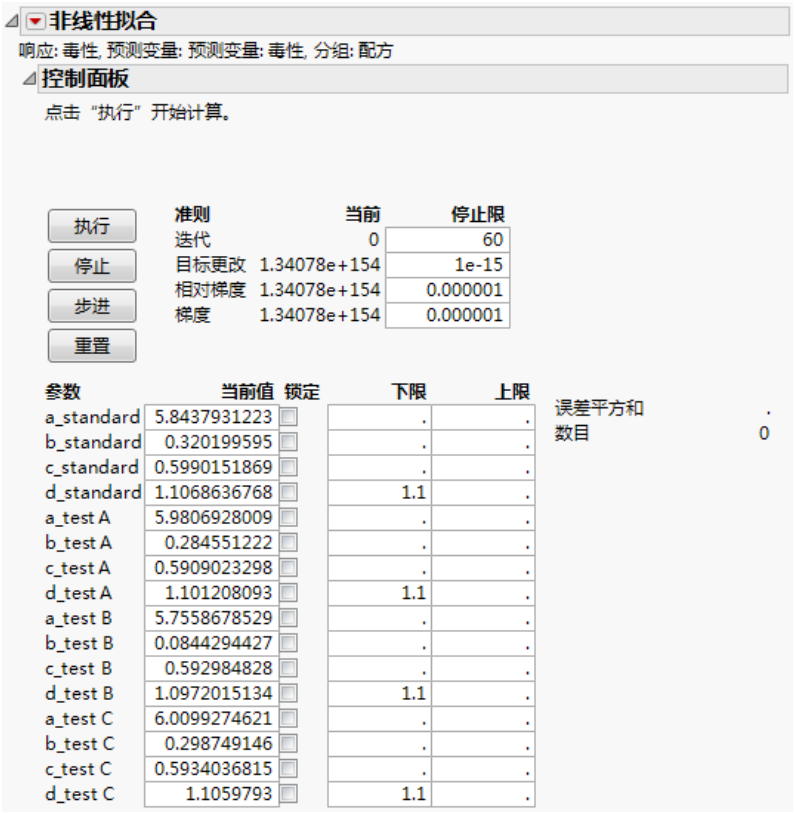
   随即显示“非线性拟合”窗口。在控制面板中，显示参数值和锁定选项。列在每个参数之前的字母对应于“拟合曲线”函数中预测模型的变量。

图 15.17 “非线性拟合”控制面板

提示：若您根据先验信息知道参数值，则可以锁定参数。

7. 点击“非线性拟合”红色小三角并选择**参数界限**。
   用于设置参数上限和下限的选项显示在参数旁边。
8. 设置参数的下限，如图 15.18 中所示。您从以前的经验知道药物的最大毒性至少为 1.1。

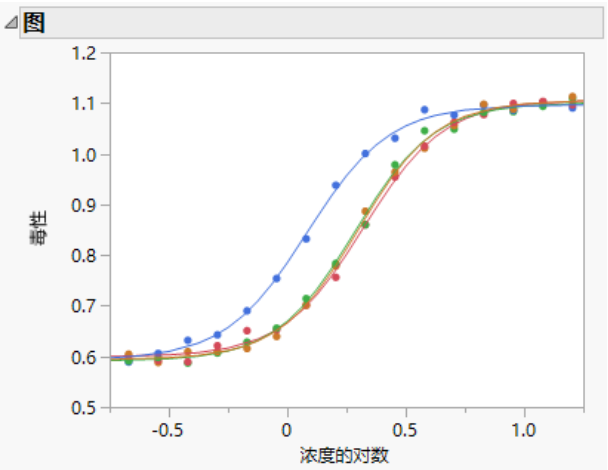
图 15.18 设置参数界限



9. 点击**执行**。

   最终的参数估计值与其他拟合统计量一起显示在**解**报表中。拟合模型显示在图上。

图 15.19 非线性拟合图和参数估计值



参数	估计值		下限	上限
a_standard	5.843793154		3.81464	7.87295
b_standard	0.3201995954		0.25624	0.38416
c_standard	0.5990151873		0.56407	0.63396
d_standard	1.1068636765		1.06734	1.14638
a_test A	5.9806928191		3.92762	8.03377
b_test A	0.2845512223		0.22253	0.34658
c_test A	0.5909023301		0.5552	0.6266
d_test A	1.1012080929		1.06407	1.13835
a_test B	5.6527018259		3.62286	7.68255
b_test B	0.0860027315		0.01687	0.15514
c_test B	0.5917244683		0.54246	0.64099
d_test B	1.1		1.06831	1.13169
a_test C	6.0099274821		3.95656	8.0633
b_test C	0.2987491449		0.23724	0.36026
c_test C	0.5934036814		0.55837	0.62843
d_test C	1.1059792995		1.06835	1.14361

图下方的选项允许调整参数限值和估计值。

“非线性”平台的统计详细信息

- “边侧似然置信限”
- “定制损失函数的工作机制”
- “关于导数的注释”
- “关于有效的非线性建模的注释”

边侧似然置信限

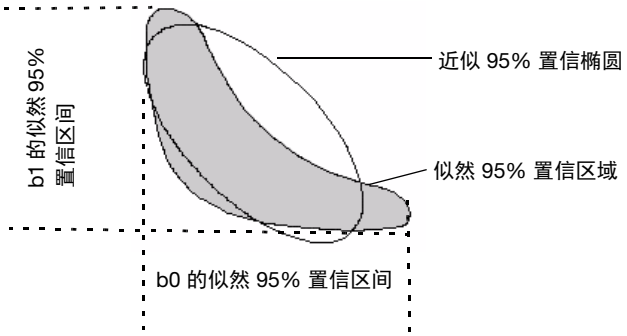
参数的置信上限和置信下限基于每个参数对于其他参数来说达到最小化之后，对该参数的值进行的一次搜索。该搜索查找符合以下条件的值：这些值所生成的误差平方和比解的最小误差平方和高出一定差值。这个差值的目标基于 *F* 分布。置信区间有时称为**似然置信区间**或**边侧似然置信区间**（Bates and Watts 1988；Ratkowsky 1990）。

边侧置信限都以**目标误差平方和**开始。这是 *F* 检验认为在给定的 alpha 水平下与解误差平方和存在显著差异的误差平方和（或损失函数之和）。若将损失函数指定为负对数似然，则使用卡方分位数替代 *F* 分位数。针对每个参数的置信上限，增加参数值直到误差平方和达到目标误差平方和。参数值向上移动时，根据刻画参数的变化，所有其他参数会调整到最小二乘估计值。从概念上看，这是一组复合的嵌套迭代。但内部可以使用 Johnton 和 DeLong 开发的一组迭代来执行此操作。请参见《SAS/ETS 用户指南》(SAS Institute Inc. 2020)。

图 15.20 显示了目标误差平方和或负似然的等高线，最小二乘 （或最小损失）解位于阴影区域内。

- 渐近标准误差生成的置信区间使用椭圆近似该区域并在极值处（在水平和垂直的切线处）取参数值。
- 边侧置信限在真实区域（而非近似的椭圆）的极值处查找参数值。

图 15.20 参数的置信限关系图



似然置信区间比根据近似标准误差计算得到的置信区间更可信。若找不到特定的限值，将对下一个限值开始计算。实现收敛很困难时，请尝试以下方法：

- 使用更大的 alpha，这会使置信区间更短，更有可能表现更好
- 放松置信限准则。

定制损失函数的工作机制

除了默认的残差平方和函数，非线性工具还可以最小化或最大化其他损失函数。本节将说明它的数学过程。

假定 $f(\beta)$ 为模型。然后 “非线性”平台尝试最小化以下形式的损失函数之和：

$$L = \sum_{i=1}^{n} \rho(f(\beta))$$

每行的损失函数 $\rho(\bullet)$ 可以是数据表中其他变量的函数。它必须具有非零的一阶和二阶导数。默认 $\rho(\bullet)$ 函数 （残差平方）为：

$$\rho(f(\beta)) = (y - f(\beta))^2$$

要指定使用定制损失函数的模型，请在数据表中构造一个变量并生成损失函数。在启动“非线性”平台后，选择包含损失函数的列作为损失变量。

按如下所示，通过对模型的 $\rho(\bullet)$ 求二阶导数并生成梯度和近似 Hessian，得到非线性最小化公式：

$$L = \sum_{i=1}^{n} \rho(f(\beta))$$

$$\frac{\partial L}{\partial \beta_j} = \sum_{i=1}^{n} \frac{\partial \rho(f(\beta))}{\partial f} \frac{\partial f}{\partial \beta_j}$$

$$\frac{\partial^2 L}{\partial \beta_j \partial \beta_k} = \sum_{i=1}^{n} \left[\frac{\partial^2 \rho(f(\beta))}{(\partial f)^2} \frac{\partial f}{\partial \beta_j} \frac{\partial f}{\partial \beta_k} + \frac{\partial \rho(f(\beta))}{\partial f} \frac{\partial^2 f}{\partial \beta_k \partial \beta_j} \right]$$

若 $f(\bullet)$ 的参数具有线性关系，则最后一个等式中的第二项为零。若不具有线性关系，您仍可以希望其和相对于第一项而言较小，并使用

$$\frac{\partial^2 L}{\partial \beta_j \partial \beta_k} \cong \sum_{i=1}^{n} \frac{\partial^2 \rho(f(\beta))}{(\partial f)^2} \frac{\partial f}{\partial \beta_j} \frac{\partial f}{\partial \beta_k}$$

若 ρ 是残差平方，则第二项可能很小，因为残差和很小。若有截距项，则第二项为零。对于最小二乘法，该项可以区分 Gauss-Newton 和 Newton-Raphson。

注意：仅当执行最小二乘估计或使用最大似然估计并且负对数似然值合理时，标准误差、置信区间和假设检验才正确。

关于导数的注释

“非线性”平台对于具有最常见运算的公式求符号导数。本节将说明生成的导数表达式类型。

若您打开 Negative Exponential.jmp 非线性样本数据示例，**非线性**列的实际公式类似于以下公式：

```
Parameter({b0=0.5, b1=0.5}, b0*(1-Exp(-b1*X)))
```

若您使用公式编辑器，则公式中的 Parameter 部分是隐藏的。这说明了公式在列中的储存方式以及在“非线性”启动对话框中的显示方式。公式中给出了名为 **b0** 和 **b1** 的两个参数的初始值，在公式中供拟合使用。

“非线性”平台生成公式的单独副本，并对它进行编辑以提取表达式中的参数。然后它将对这些参数的引用映射到估计参数的位置。“非线性”平台对预测公式的参数计算分析导数。若您使用**显示导数**命令，则可以获得日志中所列的公式，它类似于以下内容：

预测模型：

```
b0 * First(T#1=1-(T#2=Exp(-b1*X)), T#3=-(-1*T#2*X))
```

模型的对于参数的导数为：

```
{T#1, T#3*b0}
```

导数工具的工作机制如下所示：

- 为了避免重复计算子表达式，预测模型通过对线程赋值来储存导数计算所需的子表达式的值。这些赋值命名为 T#1、T#2 等。
- 预测模型需要计算更多子表达式时，它使用 First 函数来返回第一个参数表达式的值，同时也会计算其他参数表达式的值。在这种情况下，需要更多赋值来计算导数。
- 导数表本身是表达式的列表，每个要拟合的参数对应于一个表达式。例如，模型的对于 **b0** 的导数为 T#1，它在预测模型中的线程为 1-(Exp(-b1*X))。模型的对于 **b1** 的导数为 T#3*b0，若您对上述赋值中进行了替换，则它的线程为 -(-1*Exp(-b1*X)*X)*b0。尽管进行了很多优化，它有时并不能以最优方式组合运算。您从 T#3 的表达式中可以看出，该表达式并未消除双否定运算。

若您指定损失函数，则公式编辑器会对参数求导（若有的话）。若有一个参数，则求模型的一阶导数和二阶导数。

若导数机制不知道如何求函数的分析导数，则使用 NumDeriv 函数求数值导数。若求数值导数，平台会显示 delta 的值，它用于计算当参数表达式变化 delta 的时候，函数的相对变化量。您可能需要试用不同的 delta 设置以获得好的数值导数。

提示

始终有很多方法来表示给定的模型，一些方法的性能要比其他方法好很多。Ratkowsky (1990) 在文章中提到了替代方法。

若您有重复的子表达式，而它出现在公式的好几个地方，则最好将它赋值给一个临时变量。随后便可在公式中引用它。例如，上面的一个模型公式是这样的：

```
If(Y==0, Log(1/(1+Exp(model))), Log(1 - 1/(1 + Exp(model))));
```

该公式可以通过提取一个公共表达式并将它赋值给一个局部变量来进行简化：

```
temp=1/(1+Exp(model));
If(Y==0, Log(temp), Log(1-temp));
```

导数工具可以在不同赋值和条件下追踪导数。

关于有效的非线性建模的注释

我们强烈建议您**中心化多项式**。

当您具有一个完整的多项式项时，若它的参数具有线性关系，则最好中心化多项式。这样做可以改进需要优化的数值曲面的条件。例如，若表达式如下所示：

$$a_1 + b_1 x + c_1 x^2$$

您应将它变换为：

$$a_2 + b_2(x - \bar{x}) + c_2(x - \bar{x})^2$$

除变换了参数之外，这两个模型是等价的；但是若模型是非线性的，则第二个模型远比第一个模型更容易拟合。

变换参数之后很容易求解。

$$a_1 = a_2 - b_2\bar{x} + c_2\bar{x}^2$$
$$b_1 = b_2 - 2c_2\bar{x}$$
$$c_1 = c_2$$

若迭代次数仍达到最大值，请增加最大迭代次数或放松一个收敛准则。

真的没有一个综合性的优化方法可以解决所有问题。JMP 提供很多选项（如**牛顿算法**、**BFGS 拟牛顿算法**、**SR1 拟牛顿算法**和**仅限于数值导数**）来扩展“非线性”平台可以解决的问题范围。

若默认设置无法收敛到特殊问题的解，请使用这些设置的各种组合来增加获得收敛的机会。

一些模型对于参数的起始值很敏感。使用新的起始值通常能起到效果。您可以编辑起始值，然后点击**重置**来查看是否有效。图也可以给您带来帮助，使用滑块可以直观地修改曲线以达到更好的拟合效果。参数刻画器可以提供帮助，但是只适用于小数据集，对于其他数据集来说速度太慢。

第 16 章

JMP PRO 函数数据分析器

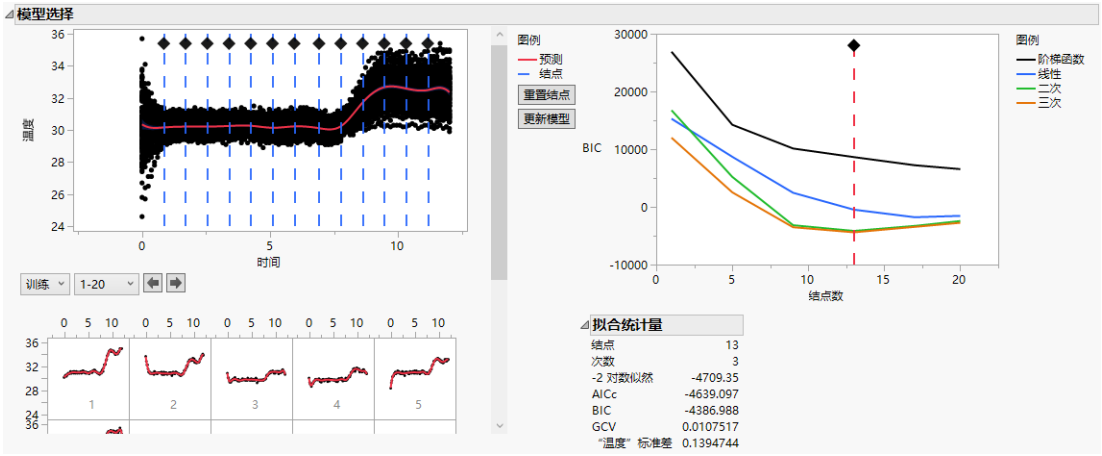
执行函数数据分析

仅在 JMP Pro 中提供 “函数数据分析器”平台。

“函数数据分析器 (FDE)”平台设计用于处理函数、信号或序列数据。它可以作为探索性数据分析工具或降维方法。降维时，FDE 平台将函数数据转换为可在另一 JMP 平台中分析的形式。

FDE 平台中提供数据预处理工具，包括输出数据的几种变换和输入数据的对齐。通过拟合数据的 B 样条、P 样条或傅里叶基函数模型来创建函数模型。拟合模型时，自动对函数模型执行函数主成分分析（函数 PCA）。将函数 PCA 的结果（例如函数主成分得分）保存到单独的数据表，以便特征提取或降维。

图 16.1 FDE 中的 B 样条模型选择



目录

“函数数据分析器”平台概述 245
函数数据分析器的示例 245
启动“函数数据分析器”平台 249
“函数数据分析器”报表 251
模型报表 252
“函数数据分析器”平台选项 255
“函数数据分析器组”选项 256
“数据处理”报表选项 257
模型报表选项 259
“函数数据分析器”平台的更多示例 260
多个函数过程的示例 260
函数实验设计示例 263
“函数数据分析器”平台的统计详细信息 266
函数模型拟合 266
函数汇总详细信息 267

JMP PRO “函数数据分析器”平台概述

函数数据可以定义为某个连续域上记录的数据，其中一个测量值集构成曲线或图像。通常该域是时间，测量值集由 ID 变量定义。将域上特定点 *t* 处的 ID 水平 *i* 的函数数据观测表示为 $f_i(t)$。通过“函数数据分析器”平台，您可以探索和分析函数数据。

函数数据的形式可以是密集或稀疏的。当 ID 变量的所有水平对应的观测值位于同一相等间距的点的网格上时，函数数据为密集的。当 ID 水平具有不同观测数（它们在域上散布，之间的间距不相等）时，函数数据为稀疏的。“函数数据分析器”平台可以处理这两种函数数据。

尽管函数数据可以用很多方式来表示，通常它可分为以下两种情况：

- 相关响应 *f(t)* 具有函数形式。
- 有一个或多个协变量 *f(t)*，它们具有函数形式。这些有时是指函数或信号处理。

“函数数据分析器”平台是一种有用的探索性工具，适用于任意类型的函数数据。但是，该平台的优势在于使用很多函数过程（可能与某个标量响应关联）并提取关键特征以在进一步建模中使用。可以首先使用 B 样条、P 样条或傅里叶基函数模型将数据拟合为函数模型。然后对该函数模型执行函数主成分分析（函数 PCA）。保存函数 PCA 的结果（例如函数主成分 (FPC) 得分）并在另一建模平台（例如“拟合模型”平台的“广义回归”特质）使用它进行特征提取和分析。或者，可以指定一组补充变量，并在 FDE 平台中拟合广义回归模型，以确定这些变量如何影响响应。

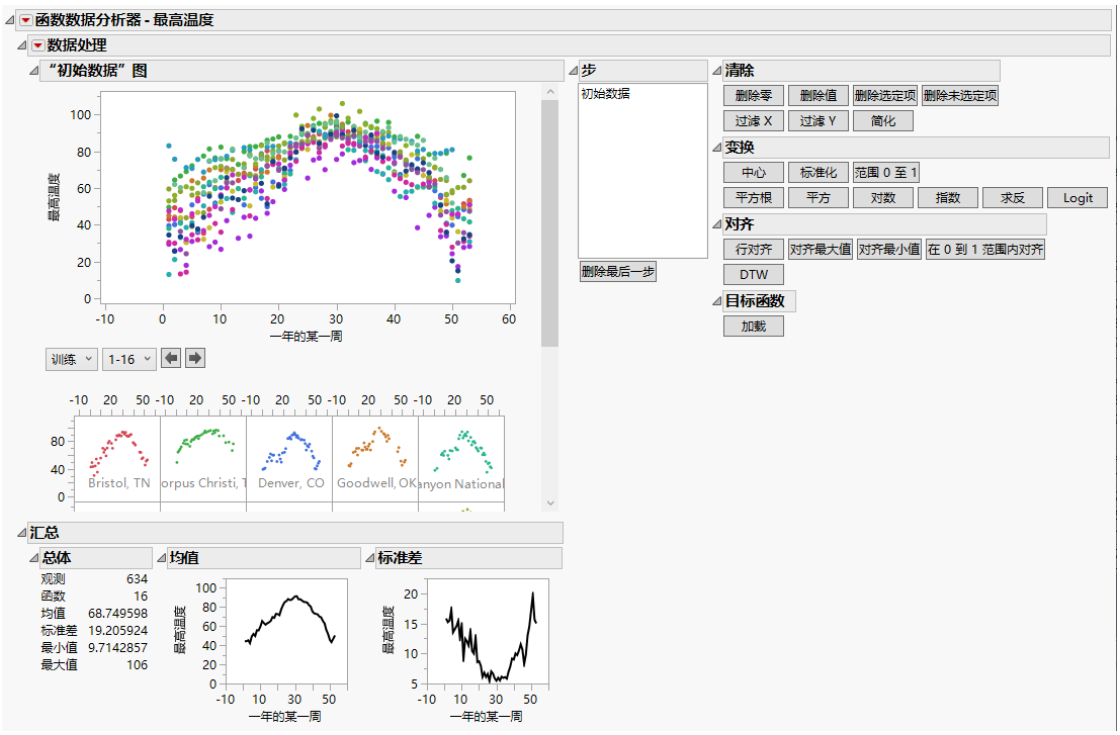
有关函数数据分析的详细信息，请参见 Ramsay and Silverman (2005)。

JMP PRO 函数数据分析器的示例

该示例分析每周从美国的 16 个气象站收集的气象数据。运行数据表中的“气象站位置”脚本来查看位置图。将每天温度汇总为每周平均值。并非每个气象站都提供一年中每周的周温度测量值。这是稀疏函数数据的一个示例。

1. 选择**帮助** > **样本数据库**，然后打开 Functional Data/Weekly Weather Data.jmp。
2. 选择**分析** > **专业建模** > **函数数据分析器**。
3. 选择**最高温度**并点击 **Y，输出**。
4. 选择**一年的某一周**并点击 **X，输入**。
5. 选择 ID 并点击 **ID，函数**。
6. 点击**确定**。

图 16.2 初始 “函数数据分析器” 报表



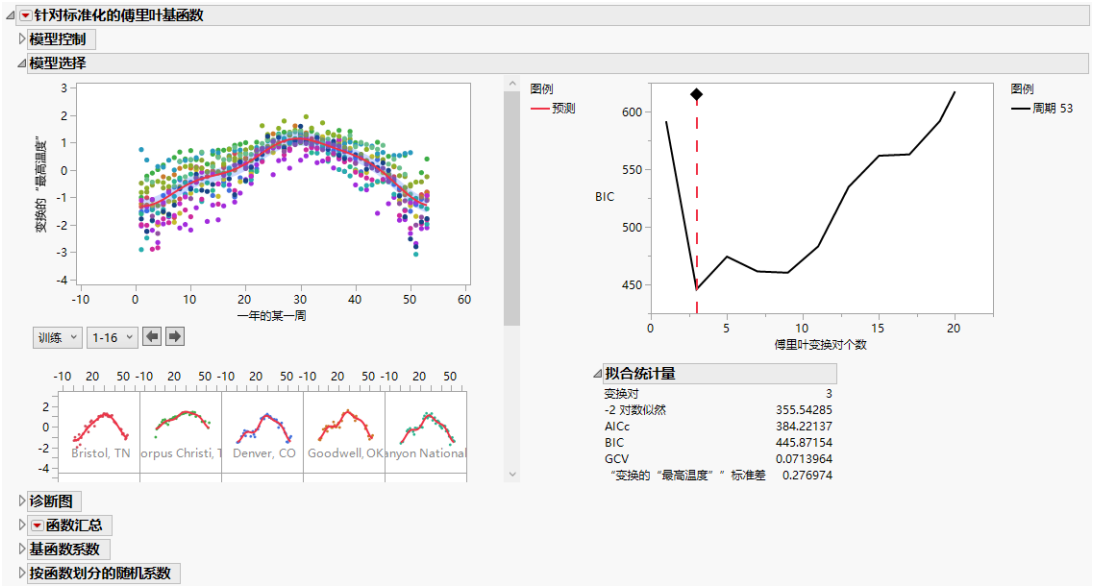
初始 “函数数据分析器” 报表包含原始数据图、汇总统计量和数据的函数均值和函数标准差的汇总图。还提供用于数据处理选项的按钮。数据处理选项也可以从 “数据处理” 红色小三角菜单访问。在建模前，通常要标准化输出数据。

7. 点击 “变换” 菜单下的**标准化**按钮。

   根据指定的变换更新数据图和汇总统计量。将 “标准化” 添加到 “步” 列表。

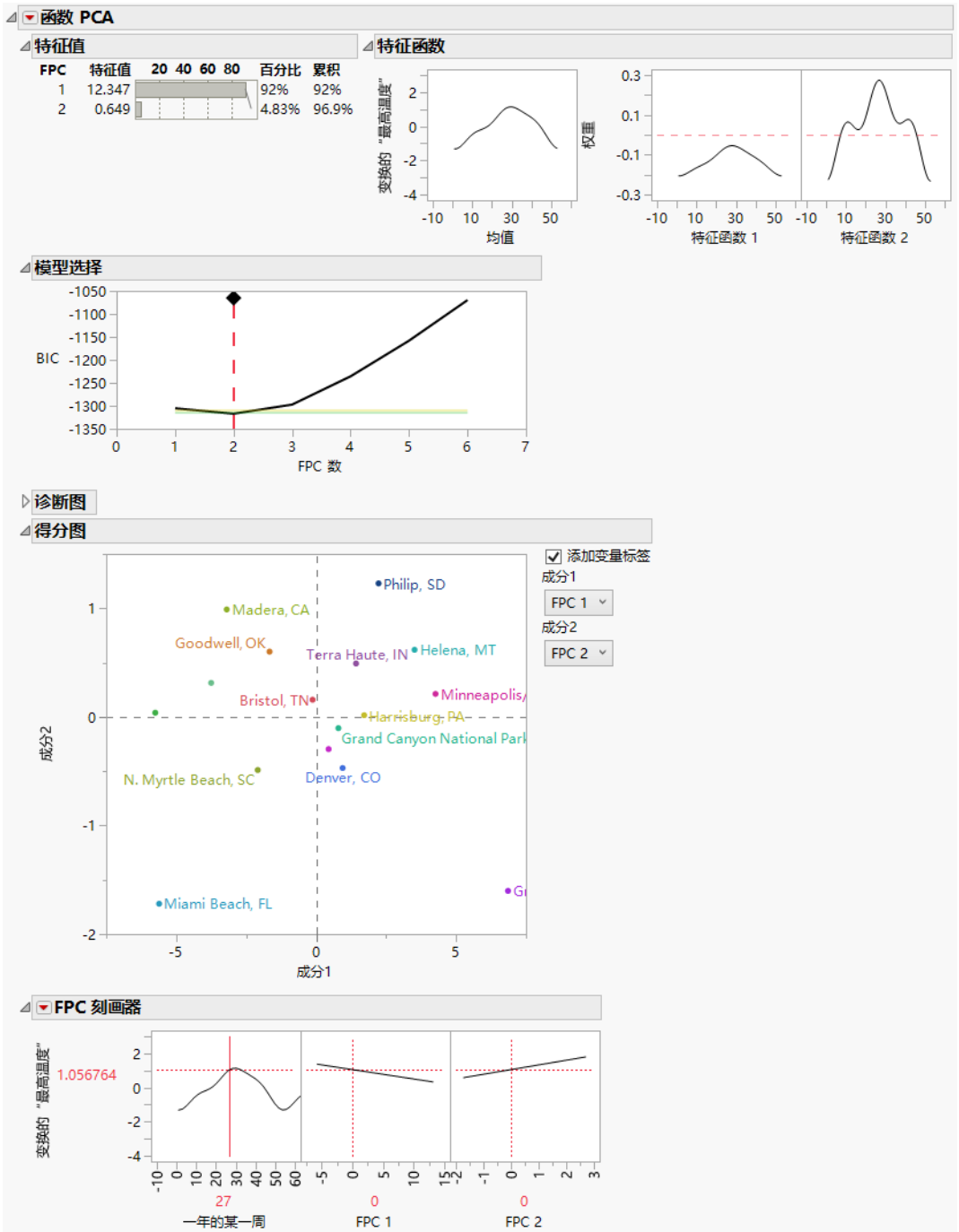
8. 点击 “函数数据分析器” 红色小三角并选择**模型 > 傅里叶基函数**。

图 16.3 傅里叶基函数模型报表



“傅里叶基函数”报表包含几个报表，这些报表包含有关所选模型的信息。在 “模型选择”报表中，显示的模型是根据 BIC 拟合准则最佳的拟合模型。对于气象数据，选择的 “傅里叶基函数”模型的期间为 53 并且具有三个基函数对。还提供该模型的拟合统计量和系数。向下滚动以查看 “函数 PCA”报表。

图 16.4 “函数 PCA”报表



“函数 PCA” 报表表明：前两个特征值解释了数据中近 97% 的变异。在 “模型选择” 图形中，点击并将红色虚线拖至 3 个 FPC 的位置，可以看到前三个特征值解释了数据中 99% 的变异。不过，第一个特征值自身就解释了 92% 的变异。您可以使用 “得分图” 检测与其他函数离群的各个函数。在 “得分图” 中，除了 “Miami Beach, FL” 和 “Greenville, ME” 这两个位置之外，大多数位置聚类在一起。向上滚动到单个函数图。“Miami Beach” 位置的函数更扁平，这指示它与其他位置相比温度变动更小。“Greenville” 位置的函数具有更小的最大值，这指示它普遍比其他位置的温度低。

提示：取消选择 “得分图” 报表中的 “添加变量标签” 选项以便更好地标识离群值。

JMP PRO 启动 “函数数据分析器” 平台

通过选择**分析 > 专业建模 > 函数数据分析器**来启动 “函数数据分析器” 平台。

图 16.5 “函数数据分析器” 启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“函数数据分析器” 启动窗口包括对应不同类型的数据格式的选项卡。基于数据格式选择相应的选项卡。

堆叠数据格式 选择其每行对应于单个观测的数据表。输出、输入和 ID 变量具有单独的列。

注意：堆叠数据格式是唯一允许您指定多个函数过程的数据格式。若您在 “堆叠数据格式” 选项卡中分配了多个列给 **Y，输出**，则单独分析每个 Y 变量。“拟合组” 报表包含 Y 变量的各个报表。

行作为函数 选择其每行对应于 ID 变量的一个水平的完整输出函数的数据表。每列是输入变量的一个水平。

警告：“行作为函数”格式假定观测在输入域中的间距相等，除非使用了“FDE X”列属性。“FDE X”列属性支持该数据格式使用列名中指定的输入变量。

列作为函数 选择其每列对应于 ID 变量的一个水平的完整输出函数的数据表。每行对应于输入变量的一个水平。

JMP PRO 启动窗口选项

Y，输出 分配函数过程 $f(t)$。对于 ID 变量的每个水平，必须有至少两个观测的输出值。

注意：观测的输出值少于两个的函数时将从分析中删除。

X，输入 （适用于“堆叠数据格式”和“列作为函数”。）分配输入变量 t。若没有为“X，输入”指定变量，则使用行号。使用行号假定观测值在输入域中是等间距的。

ID，函数 （适用于“堆叠数据格式”和“行作为函数”。）将 ID 变量分配给每个函数。对于“堆叠数据”，若没有分配 ID 变量，则假设所有观测来自一个函数。

Z，补充 （适用于“堆叠数据格式”和“行作为函数”。）分配一个或多个补充变量。“函数数据分析器”平台中的任何计算都不使用补充变量，而且包括补充变量不影响结果。补充变量是您可能希望在将来分析“函数数据分析器”的结果时使用的变量。指定补充变量时，它们将包含在由“保存数据”和“保存汇总”选项创建的表的**补充**列组中。这些列会保留原始数据表中指定的任何列属性。“函数实验设计分析”选项也可用于拟合模型。请参见第 259 页的“函数实验设计分析”。

频数 （仅适用于“堆叠数据格式”。）分配一个列，其数值表示分析中每行的频数。频数列的作用是扩展数据表，以便将具有整数频数 k 的任意行扩展为 k 个相同的行。

验证 分配用于定义验证集的可选数值列。该列应仅包含两个非重复值。较小的值定义训练集，较大的值定义验证集。若有两个以上的值，则最小的值定义训练集，所有其他值定义验证集。“FDE”平台使用验证列来训练和评估模型。有关验证的详细信息，请参见“统计详细信息”附录中第 469 页的“JMP 建模中的验证”。

注意：“验证”选项允许您提供完整函数而非每个函数的观测样本。因此，具有相同 ID 值的所有观测必须分类为测试或验证。在这两个集中不能有具有相同 ID 值的观测。有关此类验证列的详细信息，请参见“生成验证列”一章中第 180 页的“分组验证列”。

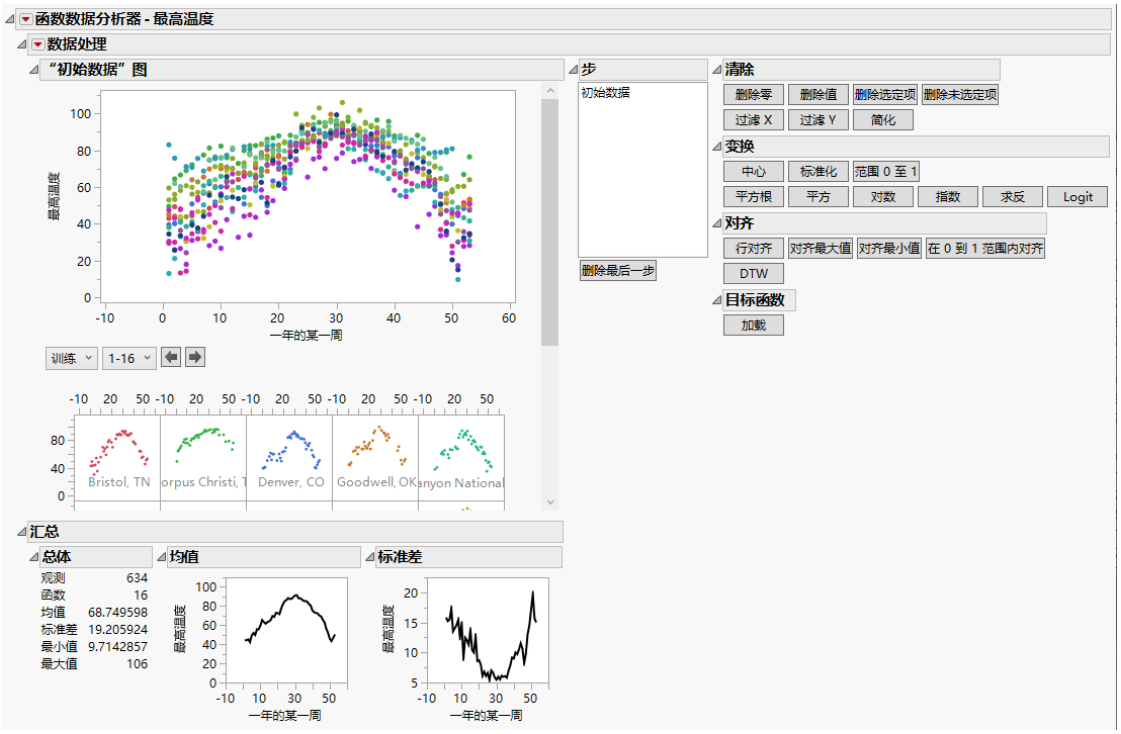
警告：若在“选择列”列表中没有选择任何列的情况下点击“验证”按钮，并且向数据表添加一个验证列，该列不会成为分组验证列；而是改用“生成验证列”平台，请参见“生成验证列”一章中第 180 页的“分组验证列”。

依据 分配用于创建报表的一个列，该报表包括变量的每个水平的单独分析。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独分析。

JMP PRO “函数数据分析器”报表

初始“函数数据分析器”报表包含一个“数据处理”报表和一个“汇总”报表。

图 16.6 初始“函数数据分析器”报表



“数据处理”报表包含所有观测的初始数据图，以及与 ID 变量各水平对应的单个数据图的网格。网格中显示的单个图的最大数目是 20。网格上方有下拉菜单和箭头，支持以交互方式选择要显示的单个图。若指定了验证集，则第一个下拉菜单支持您选择显示的单个图是来自训练集还是验证集。若有超过 20 个函数，则第二个下拉菜单和箭头支持您查看不同的图组。所有图形标绘对应输入变量的值的函数响应数据。“数据处理”报表还包含数据处理按钮。

“汇总”报表包含总体汇总统计量的表，包括观测数、函数数以及总体均值、标准差、最小值和最大值。还有函数均值和函数标准差的图。在输入变量的每个唯一值处计算图中显示的函数汇总统计量。

“数据处理”报表和“汇总”报表可用于在拟合模型或执行直接函数主成分分析之前对数据进行预处理。在“数据处理”报表中提供数据清除、变换和对齐按钮，分别对应于“数据处理”

红色小三角菜单中的选项。请参见第 257 页的“数据处理”报表选项”。每当执行预处理步骤时，自动更新这些图形和汇总结果。您可以按顺序执行多个预处理步骤。将每个步骤添加到“数据处理”报表的“步”列表中。要删除最近的步骤，请点击“步”分级显示项下面的**删除最后一步**按钮。

JMP PRO 模型报表

使用 “函数数据分析器”红色小三角菜单的 “模型”子菜单中的选项拟合模型数据。有关可用模型，请参见第 255 页的 “模型” 。每当将数据拟合为不同类型的模型时，显示模型报表。每个模型报表包含以下报表：

- 第 252 页的“模型控制”
- 第 253 页的“模型选择”
- 第 253 页的“诊断图”
- 第 253 页的“函数汇总”
- 第 254 页的“基函数系数”
- 第 254 页的“按函数划分的随机系数”
- 第 254 页的“函数 PCA”

JMP PRO 模型控制

“模型控制”报表允许您定义要在“模型选择”报表中比较的模型参数。“模型控制”报表的外观依赖于拟合的模型类型。

B 样条和 P 样条模型控制

拟合 B 样条和 P 样条模型时，您可以指定以下参数：

结点数 添加、删除或指定每个样条中的结点数范围。结点必须是非零整数。

> **注意：**B 样条模型允许的最大结点数是每个函数的最大观测数或唯一输入数。P 样条模型允许的最大结点数是唯一输入数减去 2。若指定大于最大值的数，将显示警告消息。

样条次数 添加或删除 “模型选择”报表中的样条次数拟合。

傅里叶基函数模型控制

拟合傅里叶基函数模型时，您可以指定以下参数：

傅里叶变换对个数 添加、删除或指定要比较的傅里叶变换对个数范围。

周期 更改函数的周期。

在指定模型控制后，点击**执行**以在“模型选择”报表中查看更新的模型。

提示：要在拟合模型前指定“模型控制”，请按 Shift 键，点击“函数数据分析器”红色小三角并选择所需的模型。请参见第 255 页的“模型”。

JMP PRO 模型选择

“模型选择”报表包含一个整体预测图、一个包含单个预测图的网格、一个解路径图和一个拟合统计量的表。单个预测图的网格与“数据处理”报表中单个图的网格具有相同的布局和控件。一次最多显示 20 个图。可使用下拉菜单和箭头来查看不同组的单个预测图。解路径图显示针对模型参数值标绘的模型选择准则。Bayesian 信息准则 (BIC) 是默认拟合准则。请参见第 259 页的“模型报表选项”。对于 B 样条和 P 样条模型，跨越定义的结点数标绘的每个样条次数都有单独的解路径。对于傅里叶基函数模型，针对定义的期间的傅里叶变换对个数标绘解路径。使用“模型选择”选项更改解路径图参数。

当前解用解路径图中的垂直虚线表示。将滑块置于对应具有最小模型选择准则值的节点数或傅里叶变换对的数目处。您可以拖动垂直虚线顶端的滑块来更改当前模型中的节点数或傅里叶变换对的数目。拖动滑块将自动更新“模型选择”报表中的预测图以及所有其他报表中的信息。

“拟合统计量”表包含当前解模型的说明。它还显示 -2 对数似然（AICc、BIC 和 GCV 模型拟合准则的值）和响应标准差（表示为 <Y，**输出**>标准差）的值。将响应标准差定义为拟合模型的残差 sigma。选择一个 P 样条模型时，还显示惩罚参数 λ (Lambda)。

这些预测图显示对应当前模型的原始数据和预测曲线。若存在验证集，则不会为验证集中的函数显示预测曲线。对于样条模型，选定的默认模型是具有最佳拟合的样条次数。点击解路径图或图例中的特定样条可更改当前选择的模型。总体预测图中的曲线是均值曲线的预测。单个预测图中的曲线是每个特定函数的预测曲线。对于 B 样条模型，总体预测图还显示节点的位置。您可以通过将蓝色滑块条拖动到不同位置来更改节点位置。要根据新节点位置更新模型报表，请点击**更新模型**按钮。要将节点重置为默认位置，请点击**重置结点**按钮。

JMP PRO 诊断图

“诊断图”报表包含“预测值-实际值”图和“预测值-残差”图。这些图帮助评估当前模型是否很好拟合了数据。默认情况下关闭“诊断图”报表。

JMP PRO 函数汇总

为 ID 变量的每个水平显示“函数 PCA”的汇总。默认情况下，显示与特征值（解释数据中 1% 以上的变异）关联的函数主成分。同时还显示均值、标准差、中位数、最小值、最大值、积分差值、积分平方根误差 (RISE) 和积分函数平方根 (RIFS)。积分差值和 RISE 汇总值用于确定 ID

特定函数与总体均值函数的差值。RIFS 汇总值用于最优曲线拟合。请参见第 267 页的“函数汇总详细信息”。“函数汇总”红色小三角菜单包含以下选项：

定制函数汇总 显示一个窗口，您可以在其中选择显示在“函数汇总”报表中的 FPC 数和汇总统计量。若指定了要显示的 FPC 数，还将更新“函数 PCA”报表。还有一个复选框“保存图形”，用于确定 “保存汇总”选项生成的数据表中是否包含每个函数的图形。

提示：若您有多个函数过程，可以通过点击 Ctrl 并选择 “定制函数汇总”来定制所有 “函数汇总”报表显示相同的汇总值。

保存汇总 将 “函数汇总”报表中指定的汇总统计量保存到新数据表。新数据表的名称描述了模型拟合。该数据表包含特征函数、均值函数、预测函数和条件预测函数的公式列。还有一列包含原始数据图形的图像和为每个函数拟合的指定模型。在数据表中，有一个刻画器脚本，用于启动预测公式和条件预测公式的预测刻画器。这些公式是输入变量、 ID 变量和特征函数的函数。

JMP PRO 基函数系数

显示估计的基函数系数及其标准差。它们对于 ID 变量的所有水平是相同的，是混合模型框架中的固定估计值。要查看这些系数的标准误差和置信区间，请在表中右击并选择**列**。

JMP PRO 按函数划分的随机系数

显示每个基函数和函数过程组合的估计随机系数。它们对于 ID 变量的每个水平是唯一的，是混合模型框架中的随机效应估计值。

JMP PRO 函数 PCA

对拟合的函数模型执行函数主成分分析（函数 PCA）。“函数 PCA”报表按从大到小的顺序列出对应每个函数主成分 (FPC) 的特征值。在条形图中列出并显示每个 FPC 解释的变异百分比和累积百分比。有一个均值函数图，每个成分还有一个与之对应的图。成分图显示特征函数值。

您可以在“函数 PCA”报表中执行模型选择，以细化选定数量的函数主成分。有一个解路径图，用来显示相对于 FPC 数量所标绘的 Bayes 信息准则 (BIC)。当前 FPC 数量用解路径图中的垂直虚线表示。包含不同数量 FPC 的模型可能具有相似的拟合。因此，解路径图提供了区域，即 BIC 统计量值的区间。有一个绿色区域和一个黄色区域。绿色区域包含最小 BIC 到最小 BIC 加 4 这个区间中的值，黄色区域包含最小 BIC 加 4 到最小 BIC 加 10 这个区间中的值。默认情况下，选择绿色区域内 FPC 数量最少的模型。您可以拖动垂直线顶端的滑块来更改 FPC 的数量。拖动滑块会自动更新“函数 PCA”报表中的其他信息。

注意：这些区域在您的图中可能看上去不大。放大 y 轴以更好地显示各个区域。

当执行“直接函数 PCA”时，还提供整体预测图和单个预测图的网格。单个预测图的网格与“数据处理”报表中单个图的网格具有相同的布局和控件。一次最多显示 20 个图，可使用下拉菜单和箭头来查看不同组的单个预测图。更新 FPC 的数量也会自动更新预测图。

这些预测图显示对应当前模型的原始数据和预测曲线。若存在验证集，则不会为验证集中的函数显示预测曲线。在给定指定数量的 FPC 的情况下，总体预测图中的曲线是均值曲线的预测。在给定指定数量的 FPC 的情况下，单个预测图中的曲线是每个特定函数的预测曲线。

注意：若仅对单个函数建模，则不显示“函数 PCA”报表。否则，若 JMP 无法执行“函数 PCA”，将在“函数 PCA”报表中显示错误消息。

“函数 PCA”红色小三角菜单提供以下选项：

诊断图 显示或隐藏“预测值 - 实际值”图和“预测值 - 残差”图。使用这些图有助于评估在给定选定数量的函数主成分的情况下，模型拟合数据的良好程度。

得分图 显示或隐藏 FPC 得分的得分图。使用“选择成分”下的列表来指定在得分图的每个轴上标绘哪个 FPC。若仅有一个 FPC，则 FPC 得分绘制在 $y = x$ 这条线上，并且不显示更改成分的列表。得分图对于检测离群值很有用。在 FPC 得分的示例中，得分图对于检测具有离群值函数的 ID 变量的水平很有用。若在得分图中选择某个点，“FPC 刻画器”将设置为该函数的得分。

提示：悬停在得分图中的某个点上方，可查看该 ID 变量水平的拟合曲线的预测图。

FPC 刻画器 显示或隐藏 FPC 得分的刻画器。 FPC 刻画器包括一个输入变量列，每个 FPC 得分对应一列。每个指定的目标函数都有两个附加的刻画器。一个用来测量与目标函数的差值，另一个用来测量与目标函数的积分误差。有关“FPC 刻画器”红色小三角菜单选项的详细信息，请参见《刻画器指南》。

定制 FPC 数 指定要在“函数 PCA”中显示的 FPC 得分数。在该选项中指定 FPC 得分数还会更新“函数汇总”报表。

JMP PRO “函数数据分析器”平台选项

“函数数据分析器”红色小三角菜单包含以下选项：

汇总 一个包含函数汇总统计量的以下选项的子菜单：

标绘均值 在“汇总”报表中显示或隐藏函数均值图。默认情况下处于打开状态。

标绘标准差 在“汇总”报表中显示或隐藏函数标准差图。默认情况下处于打开状态。

标绘中位数 在“汇总”报表中显示或隐藏函数中位数图。

模型 一个包含以下模型选项的子菜单：

B 样条 拟合数据的基样条（B 样条）模型。将 B 样条模型用于非周期数据。

P 样条 拟合数据的惩罚基样条（P 样条）模型。

傅里叶基函数 拟合数据的傅里叶基函数模型。将傅里叶基函数模型用于周期数据。周期模型假定函数在它开始的地方结束。请参见第 267 页的“傅里叶基函数模型”。

注意：若有不到三个唯一输入值，则既无法拟合数据的傅里叶基函数模型，也无法拟合数据的 P 样条模型，将显示警告消息。

模型控制 显示一个子菜单，该子菜单支持您在拟合模型之前打开“模型控制”面板。请参见第 252 页的“模型控制”。

直接函数 PCA （若只有一个函数，则不可用。）直接对数据执行函数主成分分析，无需先拟合基函数模型。这减少了计算时间，特别是对于大型数据集。“直接函数 PCA”的实现如下所示：

1. 对齐输入数据，使其介于 0 和 1 之间，并将观测内插到输入值的公共网格中。
2. 对数据执行函数主成分分析。
3. 使用 P 样条模型平滑处理第一个特征函数，每个网格点上一个结点。
4. 从数据中删除第一个平滑的特征函数，重复第 2 步到第 4 步，直到解释了数据中的大量变异。

一旦执行“直接函数 PCA”，即显示“函数 PCA”报表。请参见第 254 页的“函数 PCA”。

保存数据 将处理的数据保存到新数据表。处理的数据以堆叠数据格式保存。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO “函数数据分析器组”选项

仅当在启动窗口的“堆叠数据格式”选项卡中指定了多个函数过程时才提供“函数数据分析器组”选项。

汇总 选项的一个子菜单，用于显示或隐藏所有函数过程的函数汇总统计量图。请参见第 255 页的“汇总”。

数据处理 一个数据处理选项的子菜单，可应用于所有函数过程。请参见第 257 页的“ “数据处理”报表选项”。

模型 模型选项的一个子菜单，可对所有函数过程拟合。请参见第 255 页的“模型”。

保存数据 将所有函数过程的处理数据保存到新数据表。处理的数据以堆叠数据格式保存。

保存汇总 将在每个“函数汇总”报表中指定的汇总统计量保存到新数据表。新数据表的名称描述了模型拟合。该数据表包含每个 Y 变量的特征函数、均值函数、预测函数和条件预测函数的公式列。在数据表中，每个 Y 变量有一个刻画器脚本，用于启动预测公式和条件预测公式的预测刻画器。这些公式是输入变量、 ID 变量和特征函数的函数。

行内排列 指定在窗口中显示多少个报表。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO “数据处理”报表选项

“数据处理”红色小三角菜单包含以下选项：

清除 一个包含以下数据清除选项的子菜单：

删除零 删除为零值的观测。若数据中没有零，将显示警报，指示未找到零值。

删除值 显示一个规格窗口，允许您指定要从数据中删除的值。

删除选定项 删除一些观测，它们对应在数据表中选定的行。

删除未选定项 删除一些观测，它们对应在数据表中未选定的行。

警告：“删除选定项”和“删除未选定项”将删除行号。启用“自动重新计算”时，必须先添加或删除行，然后才能使用这些选项。

过滤 X 删除落在指定区间之外的 X 值。选择“过滤 X”选项时，必须指定“之下”和“之上”值。落在指定区间之外的 X 值不用在分析中。

过滤 Y 删除落在指定区间之外的 Y 值。选择“过滤 Y”选项时，必须指定“之下”和“之上”值。落在指定区间之外的 Y 值不用在分析中。

简化 使用以下方法之一简化 X 值上的数据：

- 使用“网格”选项卡将观测内插到值的公共网格中。您可以指定网格大小。默认情况下，网格大小是唯一输入值数量的一半，因此减少了总观测数。若您不想减少观测总数，而只是希望观测位于同一网格中，请将网格大小指定为唯一输入值的数目。
- 使用“箱”选项卡创建指定数量的箱，这些箱在唯一 X 值上均匀分布。对于每个函数（或“ID，函数”变量的水平），对箱中的观测进行平均，以生成对应箱水平的 Y 值。

– 使用“细”选项卡删除 X 值上的每个 N 观测，其中 N 由指定的细化率确定。针对每个函数（或“ID，函数”变量的水平）执行该操作。默认情况下，细化率为 2，这将删除每个函数中一半的观测值。

注意:“删除”选项将指定的观测从分析和建模报表中排除，但是观测在数据表中保持不变。

变换 一个包含以下选项的子菜单，用于变换输出数据：

中心 将输出中心化。

标准化 通过将数据中心化和统一尺度，使输出的均值为 0、方差为 1。

范围 0 至 1 将输出统一尺度到处于 0 到 1 的范围内。

平方根 通过计算输出的平方根来变换数据。输出值必须为非负数。

平方 通过计算输出的平方来变换数据。

对数 通过计算输出的自然对数来变换数据。

指数 通过计算输出的指数函数来变换数据。

求反 通过对输出求反来变换数据。

Logit 通过计算输出的 Logit 函数来变换数据。输出值必须介于 0 到 1 之间。

对齐 一个包含以下选项的子菜单，用于对齐输入数据：

行对齐 用行号替换输入值。

对齐最大值 使用观测的每个 ID 水平的最大输出值对齐函数。对于每个 ID 水平将与观测的最大输出值关联的输入值设置为零，基于观测的最大值和零之间的差值上移或下移其他输入值。

对齐最小值 使用观测的每个 ID 水平的最小输出值对齐函数。对于每个 ID 水平将与观测的最小输出值关联的输入值设置为零，基于观测的最小值和零之间的差值上移或下移其他输入值。

在 0 到 1 范围内对齐 对齐输出函数，使输入值的范围为 0 到 1。

提示：拟合 P 样条模型时，**在 0 到 1 范围内对齐**尤其有用。

动态时间规整 （仅当有多个函数时才可用。）使用动态时间规整 (DTW) 对齐输出函数。DTW 是一种函数对齐方法，它查找一个最佳规整来一起对齐两个或更多函数。选择 DTW 选项时，将显示“选择参考函数”窗口。使用它来选择参考函数。参考函数是其余函数要对齐到的函数。

选择一个参考函数并点击“确定”后，将显示一个规整函数图以及一个其余查询函数的列表。在规整函数图上，参考函数位于 y 轴，选定的查询函数位于 x 轴。与红色对角线 ($y = x$) 的偏差指示查询函数的输入已规整以便更好地对齐。

目标函数 （仅当有多个函数时才可用。）支持您加载目标函数的子菜单。

加载目标 显示一个窗口，该窗口支持您指定目标函数。目标函数用于曲线匹配，其中所有函数最好都看起来像目标函数。您还可以指定两个目标函数，将其余曲线与“最佳”和“更差”情况函数进行比较。

若指定一个或多个目标函数，则模型拟合中将不使用这些函数的数据。对于每个指定的目标函数，将向 FPC 刻画器添加两行。请参见第 255 页的“FPC 刻画器”。

注意：在执行任何其他预处理步骤之前，必须加载目标函数。

动态时间规整选项

标绘规整函数 显示或隐藏规整函数图。默认情况下处于打开状态。

保存距离矩阵 将距离矩阵保存到单独的数据表。距离矩阵对于函数聚类很有用。距离矩阵数据表包含一个层次聚类脚本。

保存规整函数 将规整函数保存到单独的数据表。数据表的每行包含进行了 DTW 调整的输入变量、原始输入变量和 ID 变量。

JMP PRO 模型报表选项

模型选择 显示作为模型选择准则的选项的子菜单。这些选项是 AICc、BIC 和 GCV。请参见《拟合线性模型》。

标绘基函数图 在一个图形上显示或隐藏所有基函数的图。

诊断图 显示或隐藏“诊断图”报表。请参见第 253 页的“诊断图”。

函数汇总 显示或隐藏“函数汇总”报表。请参见第 253 页的“函数汇总”。

基函数系数 显示或隐藏“基函数系数”报表。请参见第 254 页的“基函数系数”。

按函数划分的随机系数 显示或隐藏“按函数划分的随机系数”报表。请参见第 254 页的“按函数划分的随机系数”。

函数 PCA 显示或隐藏“函数 PCA”报表。请参见第 254 页的“函数 PCA”。

函数实验设计分析 （仅当在启动窗口中指定了至少一个补充变量时才可用。）在 FDE 平台中启动“广义回归”报表。将补充变量用作模型效应，对各个 FPC 得分函数进行广义回归模型拟合。默认情况下，拟合二阶析因模型，“估计方法”为“最佳子集”。请注意，分类二次效应未包括在内。若模型中的项数大于 21 或函数数目大于 1000，则“估计方法”会自动切换为“剪除向前选择”。或者，可以在原始数据表中指定定义所需模型拟合的模型脚本。使用补充变量对 FPC 得分进行建模可支持您使用模型拟合来确定响应如何基于补充变量进行更改。使用“FDOE 刻画器”探索补充变量如何影响响应。

“函数实验设计分析”报表包含以下红色小三角菜单选项：

“FPC 得分”的广义回归 显示或隐藏每个 FPC 得分的“广义回归”报表。有关“广义回归”模型报表的详细信息，请参见《拟合线性模型》。

诊断图 显示或隐藏响应变量的“预测值-实际值”图和残差图。

FDOE 刻画器 显示或隐藏“FDOE 刻画器”，该刻画器支持您探索响应如何根据补充变量进行更改。有关“FPC 刻画器”红色小三角菜单选项的详细信息，请参见《刻画器指南》。

保存列 显示一个列表，其中包含选项“保存预测公式”和“保存残差公式”。使用这些选项可将相应的公式保存至数据表中的新列。

保存数据 将建模的数据保存到新数据表。建模的数据以堆叠数据格式保存。

删除拟合 为指定的拟合删除模型报表。

JMP PRO “函数数据分析器”平台的更多示例

- “多个函数过程的示例”
- “函数实验设计示例”

JMP PRO 多个函数过程的示例

该示例使用 Fermentation Process.jmp 和 Fermentation Process Batch Yield Results.jmp 样本数据表来分析酶产量。“产量”是转基因的酵母生产的酶数量。每个批次取 100 次过程测量值，它们平均分布在 12 个小时的时间段上。

使用 “函数数据分析器”平台拟合数据模型并将函数主成分保存到新数据表，然后在 “拟合模型”平台的 “广义回归”特质中分析这些函数主成分。

拟合函数模型

1. 选择**帮助** > **样本数据库**，然后打开 Functional Data/Fermentation Process.jmp。
2. 选择**分析** > **专业建模** > **函数数据分析器**。
3. 在 “堆叠数据格式”选项卡中，选择**乙醇**到 pH 并点击 **Y，输出**。
4. 选择**时间**并点击 **X，输入**。
5. 选择**批次 ID** 并点击 **ID，函数**。
6. 点击**确定**。
7. 点击 “函数数据分析器组”红色小三角并选择**数据处理** > **对齐** > **在 0 到 1 范围内对齐**。这在每个 “函数数据分析器”报表中将输入变量对齐为介于 0 到 1 之间。
8. 点击 “函数数据分析器组”红色小三角并选择**模型** > **B 样条**。这对每个函数过程拟合 B 样条模型。

图 16.7 乙醇的 “函数数据分析器”报表

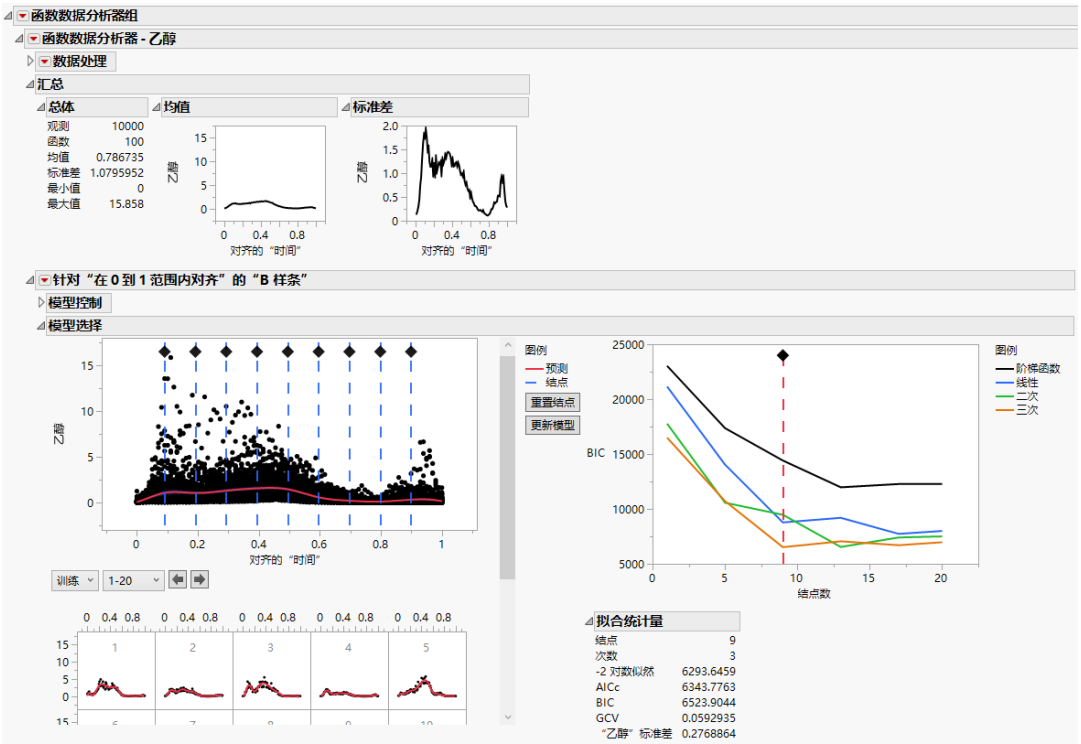


图 16.8 乙醇的 “模型汇总”报表

函数汇总

批次 ID	FPC 1	FPC 2	FPC 3	FPC 4	FPC 5	FPC 6	FPC 7	FPC 8	FPC 9	FPC 10	均值	标准差	中位数	最小值	最大值
1	0.6845185	0.4272826	0.3090187	-0.174881	-0.261418	0.2534972	0.1329424	0.0032906	-0.0227	0.0295575	1.2856117	1.2574705	0.7225838	0.0275282	3.4532315
2	0.2003363	-0.162552	0.1458985	-0.006107	0.0059254	0.0042297	0.0488981	0.090565	-0.044089	-0.007983	0.848664	0.71297	0.7468215	0.0333247	2.0280261
3	1.0570202	0.1670821	-0.025524	0.1554196	0.1599784	-0.213975	0.0257032	0.1420075	-0.077034	0.0378548	1.5221477	1.3465031	1.4533318	-0.018659	3.8349448
4	-0.225892	-0.342518	0.0456513	0.0469127	0.095793	-0.009734	0.0396208	0.0254217	-0.006387	0.0042403	0.5248517	0.4519468	0.4568174	0.0261908	1.6643826
5	0.7066471	0.8878144	-0.416058	0.1123721	0.0512395	0.1779	0.0420387	-0.099458	0.0015278	-0.002762	1.4912892	1.4352359	0.828356	0.068398	4.6865036
6	-0.164355	0.0402334	0.0698129	0.0188387	-0.004275	-0.031643	-0.01363	-0.001999	0.0493236	0.0057484	0.648615	0.4651278	0.6306233	0.0425897	1.5070566
7	0.4840975	0.0604706	-0.02545	0.138502	-0.044549	-0.079275	-0.003466	-0.047315	-0.096165	-0.073317	1.188724	0.8264775	1.3331523	0.0450854	2.337651
8	0.0598675	0.3232751	-0.194786	0.1468215	0.0999753	-0.021161	-0.007182	-0.06898	0.0521077	-0.059809	0.8913351	0.7786373	0.7196766	0.0029737	2.5753376
9	-0.35775	-0.244627	0.0950901	0.0453786	0.0302305	0.027442	0.0319439	0.0218378	0.024305	-0.022845	0.4330103	0.3329415	0.4001359	0.0494026	1.0654932
10	0.1030905	0.3791323	-0.32985	-0.309402	0.1014623	0.1254908	-0.011579	-0.009238	-0.031353	-0.041127	1.0564703	0.84232	0.8304595	0.068398	3.0158387

基函数系数

参数	均值估计值	标准差估计值
对齐的 “时间” [0]	0.068398	0
对齐的 “时间” [0.04]	0.3628335	0.6072559
对齐的 “时间” [0.1]	1.446524	2.3982214
对齐的 “时间” [0.19]	0.8752812	0.9861143
对齐的 “时间” [0.29]	1.3486495	1.4351681
对齐的 “时间” [0.39]	1.5859873	1.4953555
对齐的 “时间” [0.49]	1.69221	1.2326668
对齐的 “时间” [0.6]	0.396788	0.7335251
对齐的 “时间” [0.7]	0.2061623	0.2751322
对齐的 “时间” [0.8]	0.0235344	0
对齐的 “时间” [0.89]	0.3734323	0.6151888
对齐的 “时间” [0.95]	0.3903473	1.3136877
对齐的 “时间” [1]	0.1561339	0.1818355

图 16.7 和图 16.8 显示一个函数过程变量 “乙醇”的模型报表。滚动整个报表以查看每个过程变量的模型拟合情况。接着，在分析中使用 “函数汇总”报表中的 FPC。

保存 FPC 并链接产量结果

1. 按 Ctrl 的同时点击任意“函数汇总”红色小三角，然后选择**定制函数汇总**。
2. 在**输入要显示的 FPC 数**旁边的框中，键入 3。
3. 点击**取消选择所有汇总**框。
4. 点击**确定**。
5. 点击 “函数数据分析器组”红色小三角并选择**保存汇总**。
6. 选择**帮助 > 样本数据库**，然后打开 Functional Data/Fermentation Process Batch Yield Results.jmp。
7. 在**函数数据分析器模型汇总 .jmp** 数据表中，右击**批次 ID** 并取消选择**链接 ID**。
8. 在**函数数据分析器模型汇总 .jmp** 数据表中，右击**批次 ID** 并选择**链接引用 > Fermentation Process Batch Yield Results.jmp**。

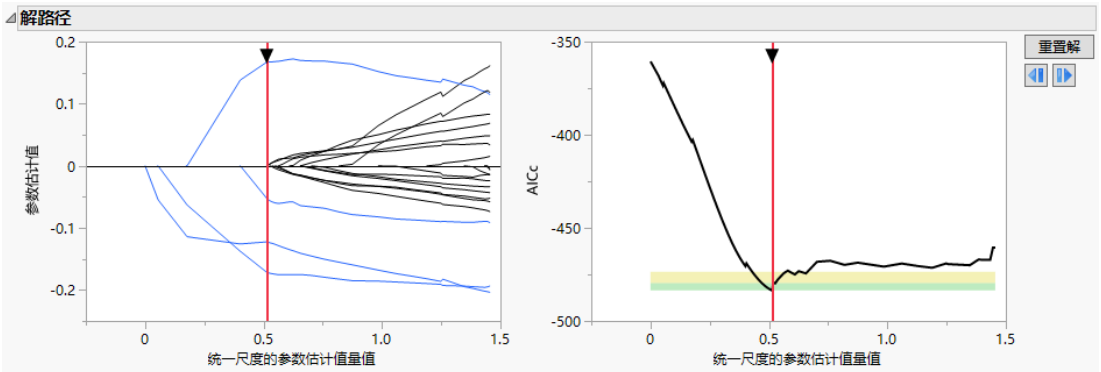
   这实际上连接了产量数据表和汇总数据表。

拟合广义回归模型

使用 “拟合模型”平台的 “广义回归”特质确定函数过程变量是如何影响**产量**的。

1. 在**函数数据分析器模型汇总 .jmp** 数据表中，选择**分析 > 拟合模型**。
2. 点击“由‘批次 ID’引用到‘Fermentation Process Batch Yield Results’”旁边的小三角。
3. 选择**产量 [批次 ID]** 并点击 **Y**。
4. 选择除**时间**和**批次 ID** 之外其余的列，然后点击**添加**。
5. 将 “特质”更改为 “广义回归”。
6. 点击**运行**。
7. 选择**自适应**框。
8. 点击**执行**。

图 16.9 批次产量的 “广义回归”报表



原始预测变量的参数估计值

项	估计值	标准误差	Wald 卡方	概率>卡方	95% 下限	95% 上限
截距	0.53436	0.0020214	69880.551	<.0001*	0.5303981	0.5383219
“乙醇” FPC 1	0	0	0	1.0000	0	0
“乙醇” FPC 2	-0.042765	0.0060948	49.23396	<.0001*	-0.054711	-0.03082
“乙醇” FPC 3	0	0	0	1.0000	0	0
“温度” FPC 1	0	0	0	1.0000	0	0
“温度” FPC 2	0	0	0	1.0000	0	0
“温度” FPC 3	0	0	0	1.0000	0	0
“糖蜜饲料” FPC 1	0	0	0	1.0000	0	0
“糖蜜饲料” FPC 2	8.4212e-5	1.4668e-5	32.960213	<.0001*	5.5463e-5	0.000113
“糖蜜饲料” FPC 3	0	0	0	1.0000	0	0
“NH3 饲料” FPC 1	0	0	0	1.0000	0	0
“NH3 饲料” FPC 2	-0.000598	8.3927e-5	50.851939	<.0001*	-0.000763	-0.000434
“NH3 饲料” FPC 3	0	0	0	1.0000	0	0
“空气” FPC 1	-9.712e-6	3.8557e-6	6.3454451	0.0118*	-1.727e-5	-2.156e-6
“空气” FPC 2	0	0	0	1.0000	0	0
“空气” FPC 3	0	0	0	1.0000	0	0
“罐内液面” FPC 1	0	0	0	1.0000	0	0
“罐内液面” FPC 2	0	0	0	1.0000	0	0
“罐内液面” FPC 3	0	0	0	1.0000	0	0
“pH” FPC 1	0	0	0	1.0000	0	0
“pH” FPC 2	0	0	0	1.0000	0	0
“pH” FPC 3	0	0	0	1.0000	0	0

“广义回归”报表显示**产量**显著受**乙醇**、**糖蜜饲料**、**NH3 饲料**和**空气**的某些成分的影响。模型的 R 方为 0.73225。通过首先使用 FDE 对函数过程进行降维，您大大减少了变量数，但是仍具有构建合理预测模型的能力。

函数实验设计示例

本例使用 Formulation For Homogeneity DOE.jmp 数据表探索不同成分组合如何影响不同温度下的化合物均匀性等级。构成化合物的三种成分是分别标记为**活性剂**、**水**和**溶剂**的主要成分。有 32 种成分组合；这些组合由**配方**列中的值表示。对于每个配方，在 20 到 70 摄氏度之间每隔 5 度测量**均匀性等级**。通过在不同温度下绘制这些测量值，可以为每个配方形成曲线。使用 “函数数据分析器”平台可确定曲线如何根据**活性剂**、**水**和**溶剂**的值而变化。

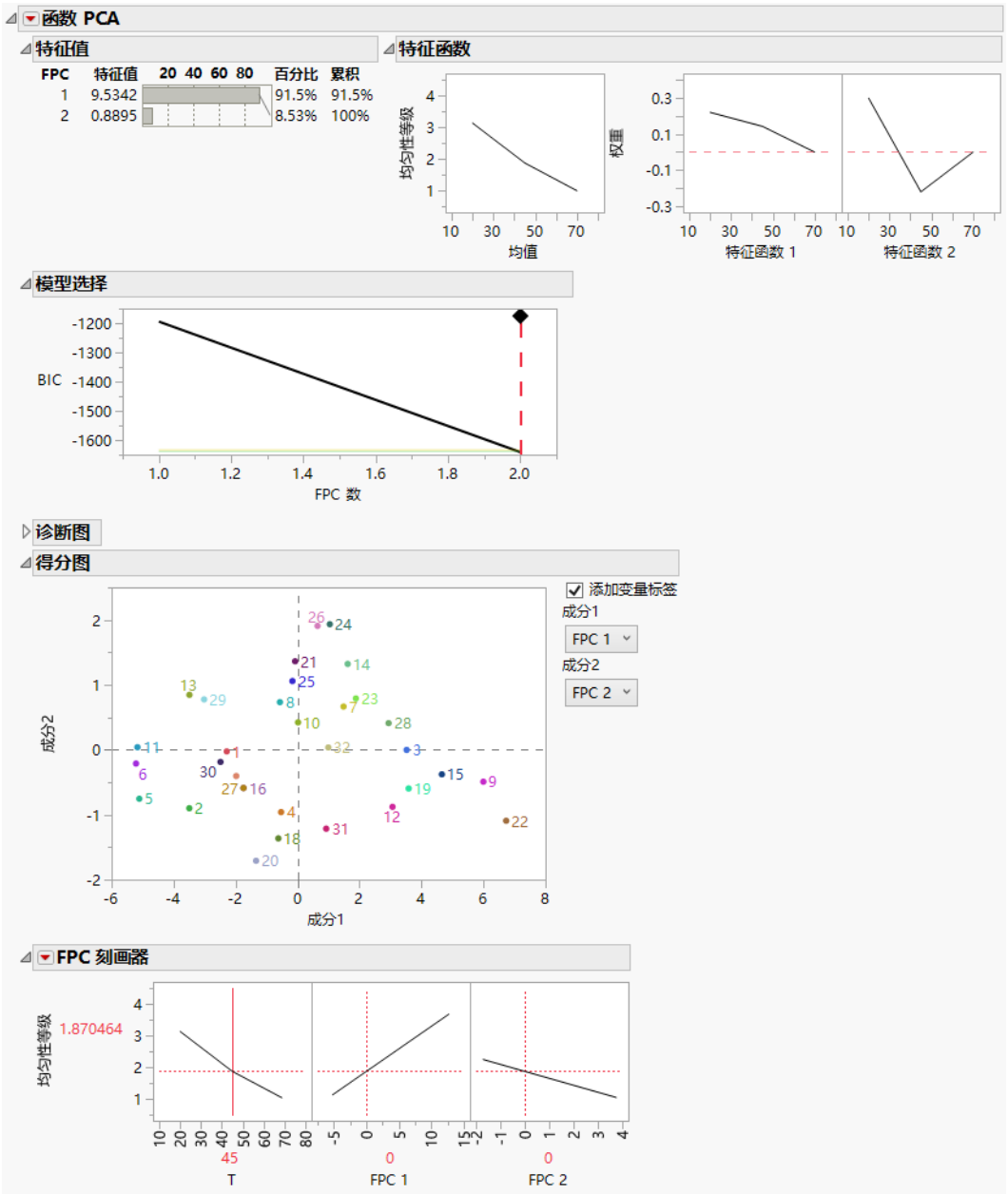
1. 选择**帮助** > **样本数据库**，然后打开 Functional Data/Formulation For Homogeneity DOE.jmp。
2. 选择**分析** > **专业建模** > **函数数据分析器**。
3. 在 “堆叠数据格式”选项卡中，选择**均匀性等级**并点击 **Y，输出**。

4. 选择 T 并点击 **X，输入**。
5. 选择**配方**并点击 **ID，函数**。
6. 选择**溶剂**、**活性剂**和**水**，然后点击 **Z，补充**。

   这些变量不用在最初的函数数据分析中。但是，通过在此将它们指定为补充变量，您可以在以后拟合模型时使用它们。
7. 点击**确定**。
8. 点击 “函数数据分析器”红色小三角并选择**模型** > **B 样条**。

   最佳拟合模型是带有一个结点的线性样条模型。

图 16.10 均匀性等级的函数 PCA



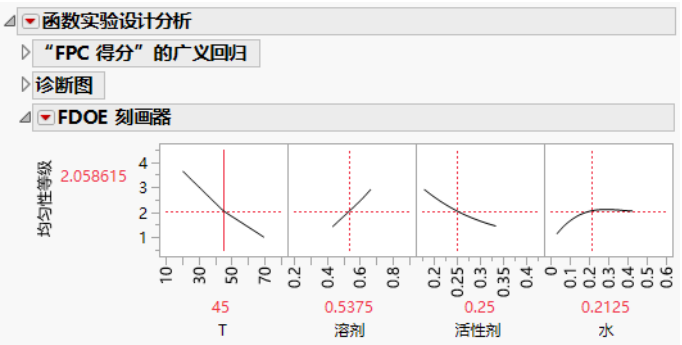
“函数 PCA”报表显示：需要两个函数主成分来解释数据中的变异。“FPC 刻画器”显示在不同温度下**均匀性等级**如何受 FPC 影响。不过，没有办法说明实际成分如何影响**均匀性等级**。

9. 点击“针对 ‘初始数据’ 的 ‘B 样条’”旁边的红色小三角，然后选择**函数实验设计分析**。

“函数实验设计分析”选项将三个补充变量用作预测变量，对每个主成分得分拟合广义回归模型。由于数据表包含 “模型”脚本，因此基于脚本中定义的模型拟合模型。

注意：若数据表不包含 “模型”脚本，则拟合二阶析因模型。

图 16.11 均匀性等级的 “FDOE 刻画器”



“FDOE 刻画器”支持您基于您实际关注的预测变量探索**均匀性等级**如何变化。通常，**均匀性等级**看起来随着**活性剂**数量的增加而减少。

注意：在本例中，**活性剂**、**溶剂**和**水**的值限定总和为 1。该约束体现在刻画器中。

JMP PRO “函数数据分析器”平台的统计详细信息

- “函数模型拟合”
- “函数汇总详细信息”

函数模型拟合

“函数数据分析器”平台中的所有模型拟合依赖于基函数扩展。基函数是一个独立函数集。任意函数 $f(t)$ 可用 K 个基函数的线性组合 （表示为 ϕ_k）来近似。通常，一个函数按以下方式近似：

$$\hat{f}(t) = \sum_{k=1}^{K} c_k \phi_k(t)$$

其中 c_k 参数是基系数。平滑程度由基函数的数目 K 决定。有关基函数扩展的详细信息，请参见 Ramsay and Silverman (2005)。

JMP PRO 傅里叶基函数模型

设有定义傅里叶基函数的 $k = 1,..., K$ 个函数，使 $\phi_0 = 1$, $\phi_{2k-1} = \sin(r\omega t)$ 且 $\phi_{2k} = \cos(r\omega t)$。则近似函数定义如下：

$$\hat{f}_i(t) = c_{i0} + c_{i1}\sin\omega t + c_{i2}\cos\omega t + c_{i3}\sin 2\omega t + c_{i4}\cos 2\omega t + \ldots$$

它使用周期 A （定义为 $A = 2\pi/\omega$）近似一个周期函数。这些系数是固定基函数系数和按函数划分的随机系数的组合。按以下方式定义每个 c：

$$c_{ik} = \beta_k + \alpha_{ik}$$

其中，β_k 是基函数 k 的固定系数，α_{ik} 是特定函数过程 i 的基函数 k 的随机系数。对于傅里叶基函数模型，β_k 和 α_{ik} 参数的估计值分别列在“基函数系数”表和“按函数划分的随机系数”表中。

傅里叶基函数模型具有一个截距项和相等数目的正弦和余弦项（称之为傅里叶变换对）。因此，K 始终为奇数。例如，若 $K = 7$，则有一个截距项和三对傅里叶变换。

函数汇总详细信息

“函数汇总”报表包括以下汇总统计信息：积分差值、积分平方根误差 (RISE) 和积分函数平方根 (RIFS)。这些汇总统计量在本节中定义。对于本节中的所有方程，假设输入值已在 0 和 1 之间对齐 。所有函数都使用以下符号：

$\hat{f}_i(x)$ 是曲线 i 在输入值 x 处的估计函数

$\hat{m}(x)$ 是输入值 x 处的估计均值函数

积分差值

积分差值是 ID 特定曲线上的点与总体均值曲线上的点的平均差值。它可用于确定 ID 特定曲线平均来说是落在均值曲线之上还是之下。曲线 i 的积分差值定义如下：

$$\mathrm{ID}_i = \int_0^1 (\hat{f}_i(x) - \hat{m}(x))dx$$

RISE

积分平方根误差 (RISE) 是 ID 特定曲线与总体均值曲线之间的平均距离。它可用于确定哪些 ID 特定曲线与总体均值曲线最相似或最不同。曲线 *i* 的 RISE 汇总值定义如下：

$$\text{RISE}_i = \sqrt{\int_0^1 (\hat{f}_i(x) - \hat{m}(x))^2 dx}$$

RIFS

积分函数平方根 (RIFS) 是用于分析最优设计曲线方案的汇总值。若预先减去最优曲线，则可以使用 RIFS 的最小值来确定哪些函数最接近最优曲线。曲线 *i* 的 RIFS 汇总值定义如下：

$$\text{RIFS}_i = \sqrt{\int_0^1 \hat{f}_i(x)^2 dx}$$

第 17 章

高斯过程

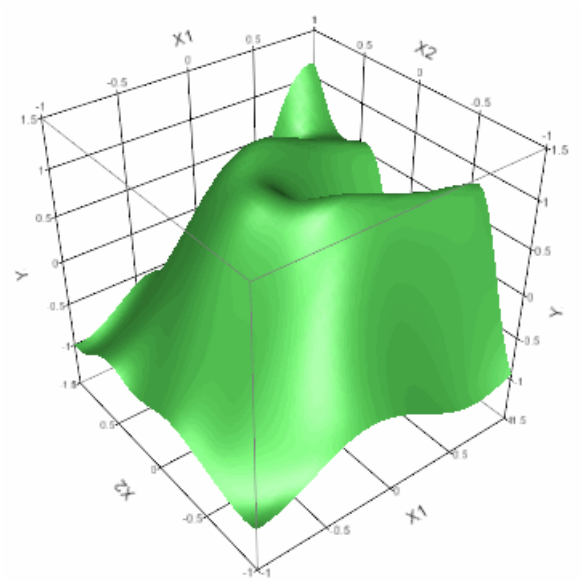
使用平滑模型拟合数据

使用“高斯过程”平台可以对某个连续响应与一个或多个预测变量之间的关系进行建模。这些类型的模型在计算机模拟实验（比如，有限元程序输出）中很常见，而且这些模型经常能够进行完美的数据插值。高斯过程可以处理这些无误差项模型，在这些模型中使用相同的输入值始终得到相同的输出值。

“高斯过程”平台拟合数据的空间相关性模型。随着自变量的值离得更远，两个观测之间的响应相关性会下降。

使用该平台的目的之一是获得可用于进一步分析和优化的预测公式。

图 17.1 高斯过程预测曲面示例



目录

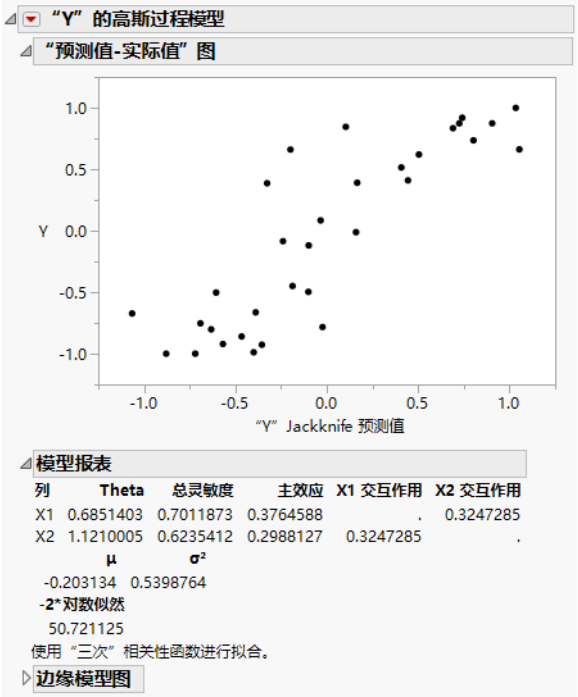
高斯过程示例 271
启动“高斯过程”平台 273
“高斯过程”报表 274
“预测值 - 实际值”图 274
模型报表 274
边缘模型图 274
“高斯过程”平台选项 275
“高斯过程”平台的更多示例 275
高斯过程模型的示例 276
具有分类预测变量的高斯过程模型的示例 277
“高斯过程”平台的统计详细信息 278
包含连续预测变量的模型 279
包含分类预测变量的模型 280
方差公式参数化 280
模型拟合详细信息 280

高斯过程示例

本例使用有关两个变量与 Y （响应）的确定性方程的空间填充设计的数据。您可以使用 “高斯过程”平台了解 X1 和 X2 针对 Y 的解释能力。您可以在列公式中查看 Y 的方程。

1. 选择**帮助** > **样本数据库**，然后打开 2D Gaussian Process Example.jmp。
2. 选择**分析** > **专业建模** > **高斯过程**。
3. 选择 X1 和 X2，然后点击 X。
4. 选择 Y 并点击 Y。
5. 选择**相关性类型** > **三次**。
6. JMP PRO 取消选择**快速 GASP**。
7. 点击**确定**。

图 17.2 “高斯过程”报表



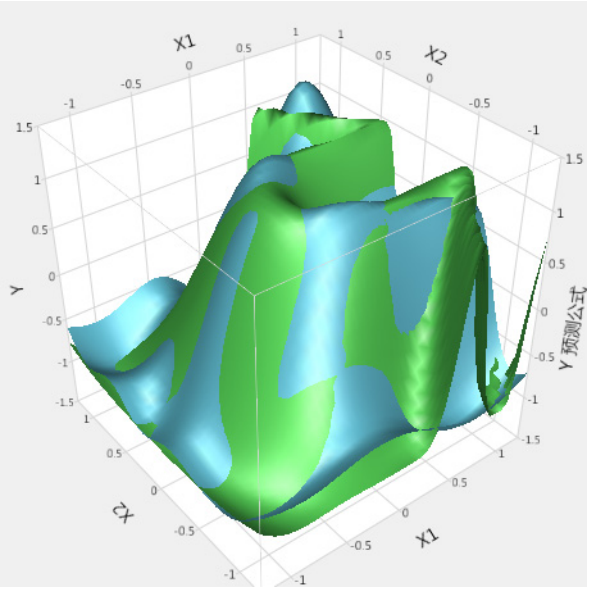
注意：考虑到最小化例程的起点不同、相关性类型的选择以及是否包含 Nugget 参数，估计参数可能有所不同。

现在，对拟合曲面与原始曲面进行直观地对比。

8. 点击 “‘Y’ 的高斯过程模型”旁边的红色小三角，然后选择**保存预测公式**。

9. 选择**图形 > 曲面图**。
10. 从 **X1** 一直选择到 **Y 预测公式**，然后点击**列**。
11. 点击**确定**。
12. 在 “曲面” 列中，为 “Y 预测公式” 选择**两侧**。

图 17.3 实际和预测 Y 的三维曲面图

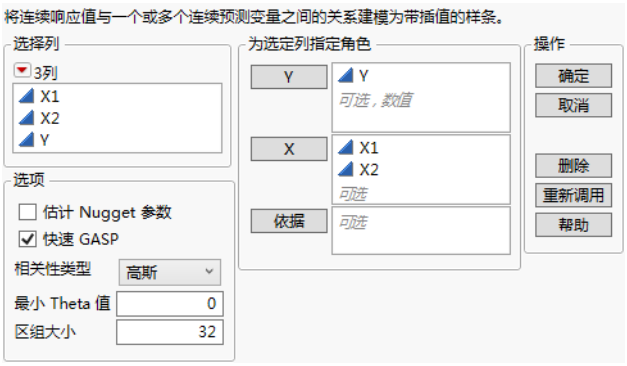


这两个曲面相似。可以直观看到 X1 和 X2 对于响应 Y 的影响。您可以旋转该图以从不同角度查看它。边缘图是另一种用于了解因子对响应的影响的工具。

启动“高斯过程”平台

通过选择**分析** > **专业建模** > **高斯过程**来启动 “高斯过程”平台。

图 17.4 “高斯过程”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y 分配要分析的连续列。

X 分配用作解释变量的列。若指定了 “快速 GASP”选项，则允许在 JMP Pro 中使用分类变量。

估计 Nugget 参数 将岭参数引入估计过程。若响应中有噪声或随机性，而您希望预测模型能平滑噪声而不是进行完美的插值，岭参数会很有用。

JMP PRO **快速 GASP** 用于使用 “快速 GASP”算法的选项。“快速 GASP”将高斯过程模型细分为小块 （称为区组）以加快计算。区组允许使用多个 CPU 和并行处理。

注意：若观测数超过 2,500，则需要使用 “快速 GASP”算法。

有关 “快速 GASP”的详细信息，请参见 Parker (2015)。

相关性类型 选择模型的相关性结构。该平台拟合数据的空间相关性模型，在这个模型中随着自变量的值离得更远，两个观测之间的响应相关性会下降。

高斯 将两点之间的相关性始终限制为非零，而不管点之间的距离有多远。

三次 允许相距足够远的点之间的相关性为零。该方法是广义的三次样条。

JMP PRO “快速 GASP”算法不支持三次相关性函数。

最小 Theta 值 设置要在拟合模型中使用的最小 theta 值。默认值为 0。 theta 值类似于常规回归模型中的斜率参数。较小的 theta 值指示变量对预测值几乎没有影响。

JMP PRO **区组大小** “快速 GASP”算法使用的每个计算区组中的观测数。每个区组必须至少有 25 个观测，而且数据集中的最大行数可设为 2,500。

“高斯过程”报表

- “‘预测值 - 实际值”图”
- “模型报表”
- “边缘模型图”

“预测值 - 实际值”图

“预测值-实际值”图在 Y 轴上显示实际 Y 值，在 X 轴上显示 jackknife 预测值。可以通过查看点在多大程度上靠近图的对角线 (Y = X) 来衡量拟合优度。

jackknife 值不是真实的 jackknife 值，这是因为模型并没有排除与每个 Y 关联的行进行重新拟合。实际上，针对预测模型，每个 Y 的关联的行是没有包含在内的，但相关性参数的计算是保留了这些行的贡献的。对于完美内插数据的高斯过程，该 jackknife 过程提供的预测不等于输入。

模型报表

“模型”报表为模型参数估计值显示函数方差分析表。具体而言，它是方差分析表，其中的变异是使用函数驱动的方法计算得到的。

Theta 高斯过程模型参数估计值。请参见第 278 页的 “‘高斯过程”平台的统计详细信息”。

总灵敏度 每个因子的主效应和所有交互作用项的总和。这个测度用于衡量某个因子及其所有双因子交互作用对于响应变量的影响程度。

“总变异”是整个实验空间上的综合变异。

主效应 每个因子的函数性主效应是仅由该因子导致的综合总变异。主效应是模型中的函数效应与每个因子的总变异之比。

交互作用 以与主效应类似的方式计算的函数交互作用效应。

JMP PRO **分类输入** 若模型包括分类因子，则提供每个分类因子的相关性矩阵。非对角线元素对应于高斯过程模型参数估计值。请参见第 280 页的 “包含分类预测变量的模型”。

Mu 和 Sigma^2 均值和方差模型参数。

Nugget 估计的 Nugget 值。若在 “高斯过程”启动窗口中选定**估计 Nugget 参数**，则报告 Nugget 值。若 JMP 已经添加 Nugget 参数以避免奇异协方差矩阵，则还会报告 Nugget 值。

-2 对数似然 负对数似然函数两倍的估计值。请参见 《拟合线性模型》。

边缘模型图

为模型中的每个因子都显示一个边缘图。对于每个图，所有因子（只有一个除外）都使用在数据中的其他因子的范围内均匀的分布被积分给积掉。其余关注因子的响应的边缘预测绘制为蓝

色。观测响应值的数据点被包括进来以显示边缘模型对数据的拟合程度。对于具有少量因子的模型，这些图可以用作诊断图。

“高斯过程”平台选项

“高斯过程”红色小三角菜单包含以下选项：

刻画器 打开标准刻画器。请参见 《刻画器指南》。

等高线刻画器 打开 “等高线刻画器”。请参见 《刻画器指南》。

曲面刻画器 打开 “曲面刻画器”。请参见 《刻画器指南》。

保存预测公式 在活动数据表中创建一个新的预测公式列。

保存方差公式 在活动数据表中创建一个新的方差公式列。

JMP PRO **发布预测公式** 创建预测公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见第 185 页的 “公式存储库”一章。

JMP PRO **发布方差公式** 创建方差公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见第 185 页的 “公式存储库”一章。

保存 Jackknife 预测值 将 jackknife 预测值保存至活动数据表。jackknife 预测值是“预测值 - 实际值”图的 X 轴值。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

“高斯过程”平台的更多示例

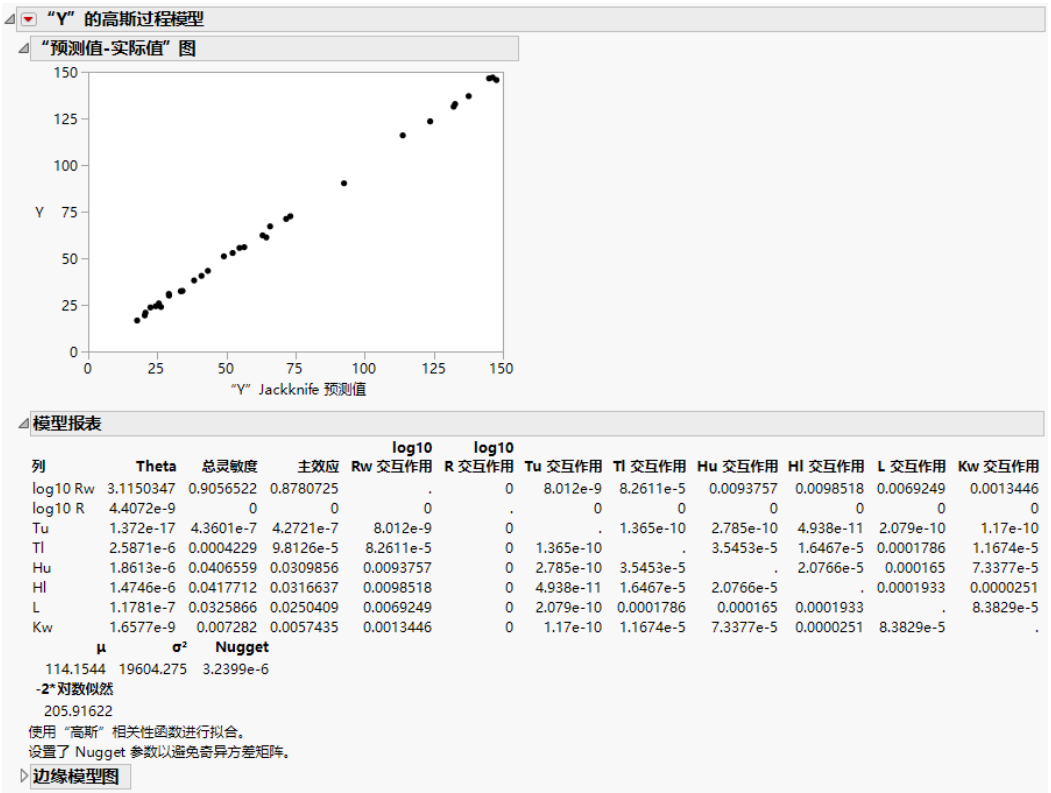
- “高斯过程模型的示例”
- “具有分类预测变量的高斯过程模型的示例”

高斯过程模型的示例

本例使用的数据演示了从地表穿过两个蓄水层的钻孔中通过的水流。在使用指定的工程模型的前提下，高斯过程能让我们了解包含在模型中的因子对响应 Y 的影响。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Borehole Latin Hypercube.jmp。
2. 选择**分析 > 专业建模 > 高斯过程**。
3. 从 log10 Rw 一直选择到 Kw，然后点击 X。
4. 选择 Y 并点击 Y。
5. JMP PRO 在 JMP Pro 中，要更快运行分析，可以保持选中 “快速 GASP”。
6. 点击**确定**。

图 17.5 “Borehole Latin Hypercube”报表



列	Theta	总灵敏度	主效应	log10 Rw 交互作用	log10 R 交互作用	Tu 交互作用	Tl 交互作用	Hu 交互作用	Hl 交互作用	L 交互作用	Kw 交互作用
log10 Rw	3.1150347	0.9056522	0.8780725	.	0	8.012e-9	8.2611e-5	0.0093757	0.0098518	0.0069249	0.0013446
log10 R	4.4072e-9	0	0	0	.	0	0	0	0	0	0
Tu	1.372e-17	4.3601e-7	4.2721e-7	8.012e-9	0	.	1.365e-10	2.785e-10	4.938e-11	2.079e-10	1.17e-10
Tl	2.5871e-6	0.0004229	9.8126e-5	8.2611e-5	0	1.365e-10	.	3.5453e-5	1.6467e-5	0.0001786	1.1674e-5
Hu	1.8613e-6	0.0406559	0.0309856	0.0093757	0	2.785e-10	3.5453e-5	.	2.0766e-5	0.000165	7.3377e-5
Hl	1.4746e-6	0.0417712	0.0316637	0.0098518	0	4.938e-11	1.6467e-5	2.0766e-5	.	0.0001933	0.0000251
L	1.1781e-7	0.0325866	0.0250409	0.0069249	0	2.079e-10	0.0001786	0.000165	0.0001933	.	8.3829e-5
Kw	1.6577e-9	0.007282	0.0057435	0.0013446	0	1.17e-10	1.1674e-5	7.3377e-5	0.0000251	8.3829e-5	.

μ	σ²	Nugget
114.1544	19604.275	3.2399e-6

-2*对数似然
205.91622

使用 “高斯” 相关性函数进行拟合。
设置了 Nugget 参数以避免奇异方差矩阵。

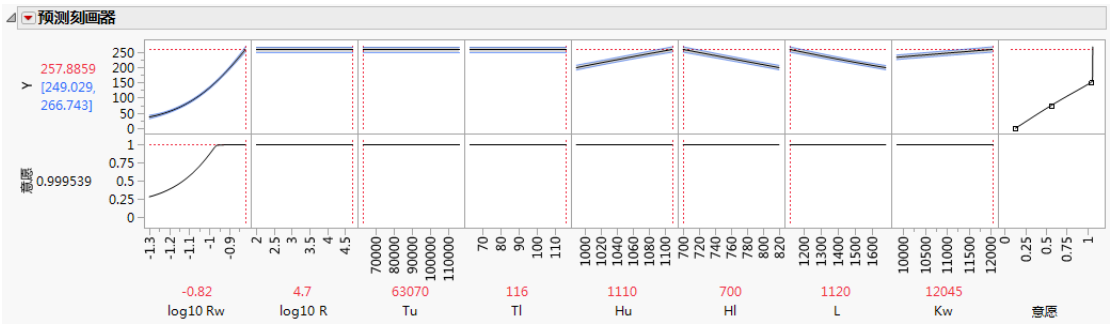
“预测值 - 实际值”图中的数据落在 Y = X 沿线，指示高斯过程预测模型是真实函数的良好近似。在“模型报表”中，您会看到第一个因子 log10 Rw 具有最高的总灵敏度。log10 Rw 的估计总灵敏度解释了响应中有超过 90% 的变异。theta 值较小的因子对于预测公式几乎没有 （或没有任何）影响。使用刻画器可直观演示灵敏度。

7. 点击 “‘Y’ 的高斯过程模型”红色小三角并选择**刻画器**。

由于响应 Y 具有 “响应限”列属性，因此会自动显示意愿函数。

8. 点击 “预测刻画器”红色小三角，然后选择**优化和意愿 > 最大化意愿**。

图 17.6 “高斯过程模型”刻画器



意愿函数设置为将响应 Y 最大化，因为 Y 的目标在 “响应限”列属性中设置为 “最大化”。最大化意愿函数确定了使响应最大化的因子的值。着色带代表 95% 置信区间。

注意：您的估计值可以不同于图 17.5 中的显示，该图中的显示使用“快速 GASP”算法得到。

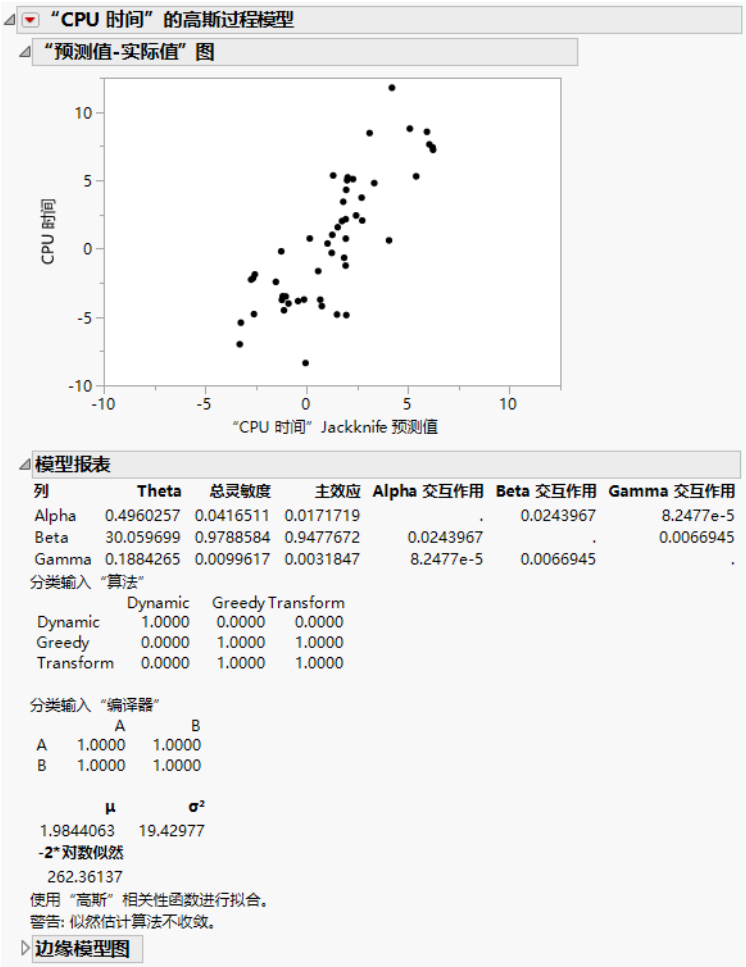
JMP PRO 具有分类预测变量的高斯过程模型的示例

本例使用 Algorithm Data.jmp 样本数据表。这些数据为对 CPU 时间的模拟，来自一个 50 次的空间填充设计实验。Algorithm Factors.jmp 样本数据表提供设计的因子和设置。该设计有三个连续因子和两个分类因子。目标是要使用同时包含连续和分类因子的高斯过程模型预测 CPU 时间。

1. 选择**帮助 > 样本数据库**，然后打开 Design Experiment/Algorithm Data.jmp。
2. 选择**分析 > 专业建模 > 高斯过程**。
3. 从 Alpha 一直选择到**编译器**，然后点击 X。
4. 选择 CPU 时间，然后点击 Y。
5. JMP PRO 要运行分析，请保持 “快速 GASP”处于选中状态。点击**确定**。

注意：“快速 GASP”选项必须用于包含分类因子的模型。请参见第 280 页的 “包含分类预测变量的模型”。

图 17.7 “Algorithm Data”报表



“预测值 - 实际值”图显示实际 CPU 时间与预测 CPU 时间之间存在强相关性。这表明高斯过程预测模型是真实函数的良好近似。在“模型报表”中， Beta 预测变量具有最高的总灵敏度。这表明在连续预测变量中，Beta 解释了响应中的大多数变异。每个分类预测变量（**算法**和**编译器**）都有一个单独的“分类输入”矩阵。这些矩阵是相关性矩阵，并且显示每个分类预测变量的水平之间的相关性。这些矩阵的非对角线元素是 τ 参数。

“高斯过程”平台的统计详细信息

- “包含连续预测变量的模型”
- “包含分类预测变量的模型”
- “方差公式参数化”

- “模型拟合详细信息”

包含连续预测变量的模型

若高斯过程模型仅包含连续预测变量，则“高斯过程”平台执行两个可能的相关性结构：“高斯”和“三次”相关。

高斯相关性结构使用幂为 2 的指数积相关性函数作为估计模型。该模型假设：Y 服从均值为 μ 且协方差矩阵为 $\sigma^2\mathbf{R}$ 的正态分布。$\mathbf{R}$ 矩阵的元素定义如下：

$$r_{ij} = \exp\left(-\sum_{k=1}^{K} \theta_k (x_{ik} - x_{jk})^2\right)$$

其中

K = 连续预测变量的数目

θ_k = 第 k 个预测变量的 theta 参数

x_{ik} = 对象 i 的第 k 个预测变量的值

x_{jk} = 对象 j 的第 k 个预测变量的值

“三次”相关性结构也假定 Y 服从均值为 μ 且协方差矩阵为 $\sigma^2\mathbf{R}$ 的正态分布。$\mathbf{R}$ 矩阵包含以下元素：

$$r_{ij} = \prod_k \rho(d;\theta_k)$$

其中

$$d = x_{ik} - x_{jk}$$

$$\rho(d;\theta) = \begin{cases} 1 - 6(d\theta)^2 + 6(|d|\theta)^3, & |d| \le \frac{1}{2\theta} \\ 2(1 - |d|\theta)^3, & \frac{1}{2\theta} < |d| \le \frac{1}{\theta} \\ 0, & \frac{1}{\theta} < |d| \end{cases}$$

请参见 Santer (2003)。“三次”相关性结构中使用的 theta 参数是文献中经常使用的参数的倒数。使用倒数是为了在 theta 对模型没有影响时，让 rho 具有零值而非无穷大。

JMP PRO 包含分类预测变量的模型

若高斯过程模型包括分类预测变量，则将高斯相关性结构用于相关性结构。 **R** 矩阵的元素定义如下：

$$r_{ij} = \left(\prod_{p=1}^{P} \tau_{p_{ij}} \right) \exp\left(- \sum_{k=1}^{K} \theta_k (x_{ik} - x_{jk})^2 \right)$$

其中

K = 连续预测变量的数目

P = 分类预测变量的数目

θ_k = 第 k 个连续预测变量的 theta 参数

x_{ik} = 对象 i 的第 k 个连续预测变量的值

x_{jk} = 对象 j 的第 k 个连续预测变量的值

$\tau_{p_{ij}}$ = 对象 i 的预测变量 p 的观测水平与对象 j 的预测变量 p 的观测水平之间的相关性

分类变量水平的每一组合都对应一个 τ 参数，其中，τ_{ij} 对应于对象 i 的观测水平与对象 j 的观测水平所构成的唯一组合。因此，协方差元素 r_{ij} 取决于从第 i 个和第 j 个观测获取的分类预测变量水平的组合。请参见 Qian et al.(2012)。

方差公式参数化

保存的方差公式使用以前定义的 **R** 的参数化，但在模型包括分类预测变量时例外。当高斯过程模型包含分类预测变量时，保存的方差公式将使用 **R** 的以下参数化：

$$r_{ij} = \exp\left(- \sum_{k=1}^{K} \theta_k (x_{ik} - x_{jk})^2 - \sum_{p=1}^{P} \phi_{p_{ij}} \right)$$

其中，$\phi_{p_{ij}} = -\ln(\tau_{p_{ij}})$ 和所有其他变量都和以前定义的一样。

模型拟合详细信息

通过最大似然拟合模型参数。拟合参数在平台报表中提供。下面列出了参数：

- μ 是高斯过程均值，
- σ^2 是高斯过程方差，
- Theta 对应于 **R** 定义中 θ_k 的值。
- 分类输入相关性矩阵的非对角线元素对应于 **R** 定义中 τ_{pij} 的值。

注意：若您的报表中包含说明**设置了 Nugget 参数以避免奇异方差矩阵**，则 JMP 会向方差矩阵添加一个岭参数以便该矩阵是可逆的。

第 18 章

时间序列分析

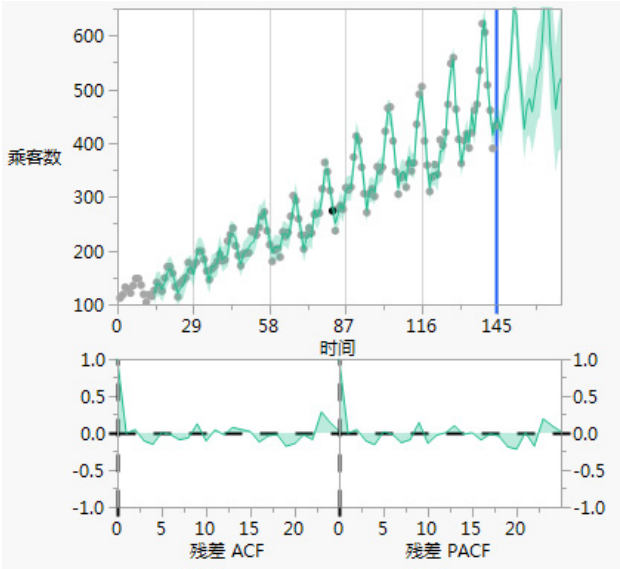
拟合时间序列模型和转换函数

使用“时间序列”平台，您可以探索、分析和预测一元时间序列。时间序列是在一系列等间隔的时间期间上得到的一组观测值。时间上靠得很近的观测通常相关。时间序列方法利用这个观测之间的相关性更好地预测序列在将来是什么样子。

时间序列数据所共有的特征包括季节性、趋势和自相关性。“时间序列”平台提供处理这些特征的选项。可使用一些图形（如变差图、自相关性图、偏自相关性图和谱密度图）来标识适合描述和预测时间序列的模型类型。在平台中还提供几个分解方法，使您可以去除数据中的季节性或一般趋势来简化分析。或者，平台可以拟合更复杂的 ARIMA 模型和“状态空间平滑”模型，它们可以在一个模型中包含季节性和长期趋势。还可以执行 Box-Cox 变换，并对变换后的序列进行分析和建模。

有若干方法来评估模型的预测性能。“针对保留进行预测”功能将时间序列划分到一个训练部分中以构建模型，还划分到一个保留部分中以评估预测性能。

图 18.1 预测图



目录

“时间序列”平台概述 285
“时间序列”平台示例 285
启动“时间序列”平台 287
时间序列分析报表 289
时间序列图 289
时间序列基本诊断图 289
“时间序列”平台选项 291
时间序列诊断 291
差分和分解 292
ARIMA 和季节性 ARIMA 模型 294
平滑模型 295
状态空间平滑模型 296
转换函数模型 297
平滑模型规格窗口 299
报表 301
差分报表 301
分解报表 301
“模型比较”报表 303
ARIMA 和季节性 ARIMA 模型报表 305
“状态空间平滑”报表 308
转换函数报表 311
“谱密度”报表 312
“时间序列”平台的更多示例 312
创建时间 ID 列的示例 313
使用保留集的示例 318
“时间序列”平台的统计详细信息 321
谱密度的统计详细信息 321
X-11 分解的统计详细信息 321
平滑模型的统计详细信息 322
ARIMA 模型的统计详细信息 326
转换函数的统计详细信息 327

“时间序列”平台概述

时间序列是在一系列等间隔的时间期间上得到的一组观测值 $y_1, y_2, \ldots, y_N$。时间序列数据的一些示例包括季度销售报表、每月平均温度和太阳黑子计数。“时间序列”平台支持您探索在这些数据类型中找到的模式和趋势，然后可以使用这些模式和趋势来预测未来发展。

时间序列数据所共有的特征包括季节性、趋势和自相关性。**季节性**是指在已知的一段时间上发生的模式。例如，历年收集的数据中，每月收集的数据在夏季月份上都很相似。**趋势**是指序列的长期变动情况，例如值随着时间逐渐增加或减小。**自相关性**是指序列中的每个点与序列中早先的值相关的程度。

“时间序列”平台提供很多不同的模型和预测方法。但是，不是所有方法都可以处理趋势或季节性。为了选择合适的模型，必须确定序列中有哪些特征。“时间序列”平台提供一些图形（如变差图、自相关性图、偏自相关性图和谱密度图），它们可用于标识适合描述和预测时间序列发展的模型类型。在平台中还提供几个差分和分解方法，使您可以去除数据中的季节性或一般趋势来探索和简化分析。您还可以查看数据并将 Box-Cox 变换应用到数据中。

或者，平台可以拟合可包含季节性和长期趋势的更复杂的模型。其中一个这样的模型是 Winter 加法，它是一个高级指数平滑模型。此外，平台可以拟合“自回归求和移动平均 (ARIMA)”模型和“状态空间平滑”模型。这两类模型在统计上最复杂，但也提供了最大的灵活性。虽然高级指数平滑、ARIMA 和“状态空间平滑”模型更难解释，但是它们都是极好的预测工具。

“时间序列”平台还可以在提供输入序列时拟合转换函数模型。

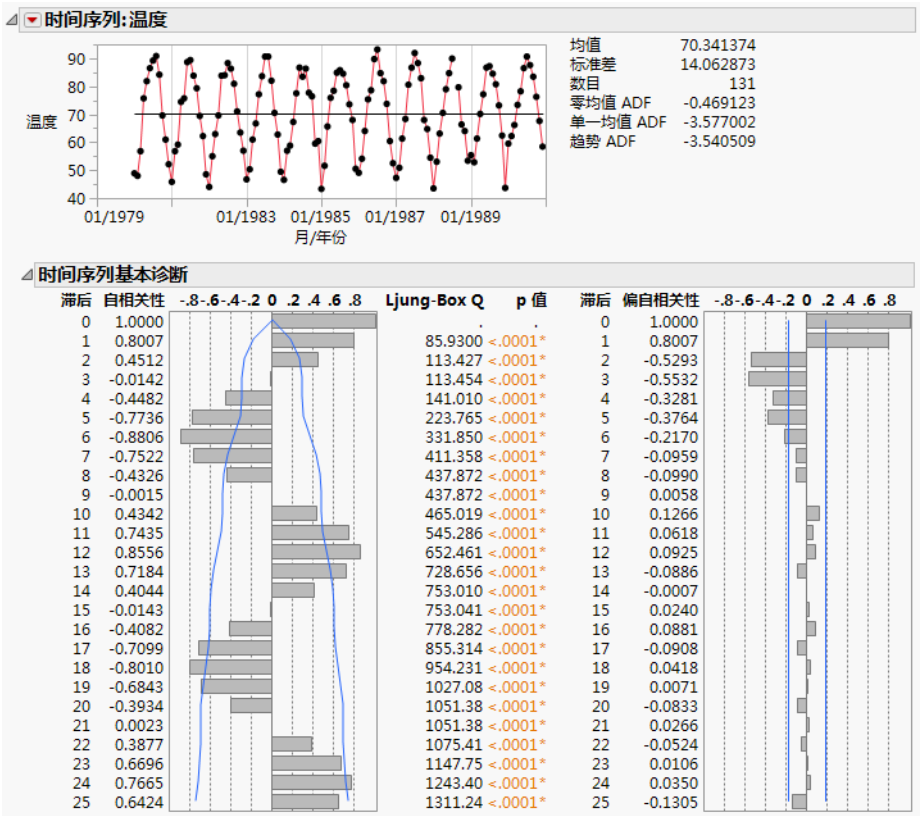
“时间序列”平台示例

该示例使用 Raleigh Temps.jmp 样本数据表，它包含从 1980 到 1990 年测量的每月最高温度（华氏度）。使用 “时间序列”平台来查看序列并预测接下来两年的每月最高温度。

1. 选择**帮助** > **样本数据库**，然后打开 Time Series/Raleigh Temps.jmp。
2. 选择**分析** > **专业建模** > **时间序列**。
3. 选择**温度**并点击 **Y，时间序列**。
4. 选择**月 / 年份**并点击 **X，时间 ID**。
5. 在**预测周期数**旁边的框中键入 24。

   这是拟合数据的模型所要预测的将来周期数。您要预测接下来的两年 （涵盖 24 个月）的每月温度。
6. 点击**确定**。

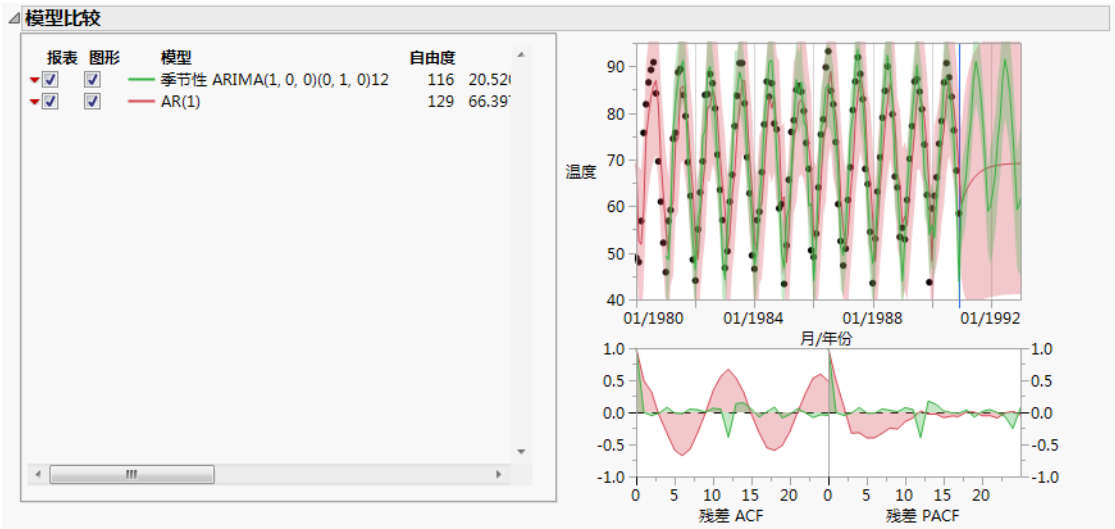
图 18.2 Raleigh Temps.jmp 的时间序列分析报表



“时间序列”图显示序列是周期性的。该周期性成分在自相关性图中也很明显。相距 1 个滞后期的点是正相关的，自相关性值为 0.8007。当点的距离变远时，变为负相关，然后再次正相关，以该模式重复下去。“时间序列”图和自相关性图提供了时间序列中存在季节性的证据。

7. 点击“时间序列”红色小三角菜单，然后选择 ARIMA。
8. 将自回归阶数 *p* 设置为 1，因为该序列显示了自相关性的证据。
9. 点击**估计**。
10. 点击“时间序列”红色小三角菜单，然后选择**季节性 ARIMA**。
11. 在“ARIMA”框中，将自回归阶数 *p* 设置为 1，因为该序列显示了自相关性的证据。
12. 在“季节性 ARIMA”框中，将季节差分阶数 *D* 设置为 1，因为该序列显示了季节性的证据。
13. 点击**估计**。
14. 在“模型比较”表中，为两个模型选中图形下的框。

图 18.3 Raleigh Temps.jmp 的 “模型比较”表

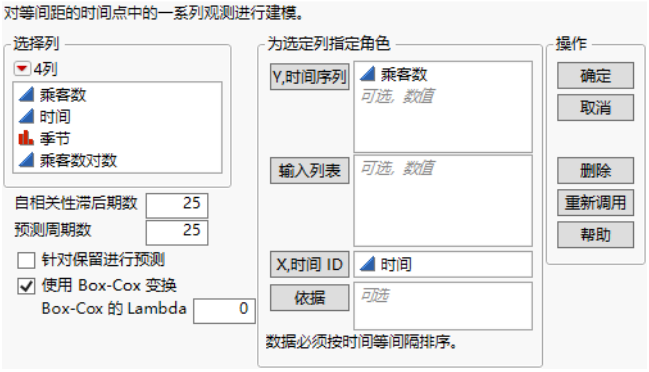


“模型比较”表按 AIC 统计量的降序排序。这意味着最佳拟合模型显示在报表顶部。季节性 ARIMA 模型的 AIC 值 (693.4) 远小于常规 ARIMA 模型的 AIC 值 (924.5)。图形显示尽管 ARIMA 模型预测观测点相对较好，但是残差大于季节性 ARIMA 模型的残差。此外，季节性 ARIMA 模型对于将来的观测的预测更符合实际，预测区间更狭窄。这些结果是合理的，因为序列显示了季节性成分的证据。

启动“时间序列”平台

通过选择**分析 > 专业建模 > 时间序列**来启动“时间序列”平台。Seriesg.jmp 样本数据表的“时间序列”启动窗口显示在图 18.4 中。

图 18.4 “时间序列”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“时间序列”平台启动窗口包含以下选项：

Y，时间序列 将一个或多个列分配为时间序列变量。显示在 Y 轴上。

输入列表 将一个或多个列分配为输入序列变量。显示在 “输入时间序列面板”中，在转换函数模型中使用。输入序列变量必须是数值，它作为时间序列或指示符。

X，时间 ID 分配用于标记时间轴 （X 轴）的一个变量。若没有为“时间 ID”指定变量，则使用行号。

注意：若您使用 “X，时间 ID”变量，可以通过使用“时间频率”列属性来指定时间频率。可以选择“每年”、“每季”、“每月”、“每周”、“每天”、“每小时”、“每分钟”和“每秒”。这可以帮助 JMP 在标绘预测值时确定数据的间距。若未指定频率，则将数据视为等间距的数值数据。

警告：假定分配给 **X，时间 ID** 的变量的观测是等间距的。不过，“时间序列”平台仅检查时间戳是否增加。该平台不检查观测是否等间距。

依据 分配用于创建报表的一个列，该报表包括变量的每个水平的单独分析。若指定了多个“依据”变量，将为 “依据”变量水平的每种可能组合生成单独的分析。

注意：若您使用 “依据”变量，可能得根据 “依据”变量的每个水平有多少个观测更改自相关性滞后期数。滞后期数必须大于 1 且小于每个水平的观测数。

自相关性滞后期数 指定在计算自相关性和偏相关性时要使用的滞后期数。这是在计算这些相关性时使用的点之间的最大周期数。它必须大于 1 且小于行数。默认滞后期数为 25。

提示：最大滞后期数的一个常用的规则是 $n/4$，其中 n 是观测数。

预测周期数 指定使用每个数据拟合模型做预测的观测数。默认预测数是 25。

- 若选择“针对保留进行预测”选项，则“预测周期数”值指定序列末尾不包含在模型拟合中的观测数。随后使用拟合模型预测这些值。
- 若未选择“针对保留进行预测”选项，“预测周期数”值指定在序列末尾后预测的将来观测数。

针对保留进行预测 确定预测是基于将来观测还是基于保留观测。若选定“针对保留进行预测”选项，则针对 “预测周期数”选项中指定的数字所确定的保留集进行预测。

使用 Box-Cox 变换 使用 “Box-Cox 的 Lambda”选项中指定的 Lambda 的 Box Cox 变换来变换原始数据。若选定该选项，将对变换的数据执行 “时间序列”报表中的所有分析。不过，所有的预测都会变换回来，并按原来的尺度报告。

Box-Cox 的 Lambda （仅当选定“使用 Box-Cox 变换”时才可用。）指定用于原始数据的 Box-Cox 变换的 lambda 参数。

注意：为了保持时间序列的顺序和间隔，“时间序列”平台将数据表中排除的行视为缺失值。

时间序列分析报表

初始“时间序列”报表显示时间序列图、时间序列变量的汇总统计量和检验，以及基本诊断报表。若为启动窗口中的“输入列表”指定了列，则显示“转换函数分析”报表和“输入时间序列面板”报表。“转换函数分析”报表和“输入时间序列面板”报表包含的初始信息都和“时间序列”报表一样。

- “时间序列图”
- “时间序列基本诊断图”

时间序列图

时间序列图按时间 ID 标绘每个时间序列。若未指定时间 ID，则使用行号。若指定了保留集，则图上有一条垂直参考线，用于将训练数据与保留数据分开。

平台还使用 Augmented Dickey-Fuller (ADF) 检验执行几个平稳性检验。以下检验和汇总统计量显示在时间序列图旁边：

Box-Cox 的 Lambda （仅当指定 Box-Cox 变换时才显示。）Box-Cox 变换中使用的 lambda 值。

均值 样本均值。

标准差 样本标准差。

数目 序列长度。

零均值 ADF 针对均值为零的随机漫步的检验，定义如下：

$$x_t = \phi x_{t-1} + e_t$$

单一均值 ADF 针对均值不为零的随机漫步的检验，定义如下：

$$x_t - \mu = \phi(x_{t-1} - \mu) + e_t$$

趋势 ADF 针对均值不为零且具有线性趋势的随机漫步的检验，定义如下：

$$x_t - \mu - \beta t = \phi[x_{t-1} - \mu - \beta(t-1)] + e_t$$

时间序列基本诊断图

“时间序列基本诊断”图中显示的信息取决于“时间序列报表”红色小三角菜单选项。显示或隐藏诊断图中信息的红色小三角菜单选项为“自相关性”、“偏自相关性”、“变差图”和“自回归系数”。默认情况下，显示“自相关性”和“偏自相关性”。

自相关性图

“自相关性”选项显示或隐藏时间序列基本诊断图中的以下列：

滞后 各点之间的期间数。

注意：滞后个数默认从 0 开始。要计算从滞后 1 期开始的相关性，请先修改 JMP 首选项，然后再生成图形。选择**文件 > 首选项 > 平台 > 时间序列**，然后选择 > **隐藏 ACF 和 PACF 中的滞后 0**。

自相关性 第 k 个滞后的自相关性计算如下所示：

$$r_k = \frac{c_k}{c_0} \text{ 其中，} c_k = \frac{1}{N}\sum_{t=k+1}^{N}(y_t - \bar{y})(y_{t-k} - \bar{y})$$

$\bar{y}$ 是时间序列中 N 个非缺失点的均值。按照定义，第一个自相关性（滞后 0）始终具有长度 1。

自相关图以条形图的形式来表示自相关系数的大小。蓝色曲线表示两倍的 Large-lag 标准误差 （±2 倍标准误差），其计算公式如下：

$$\text{SE}_k = \sqrt{\frac{1}{N}\left(1 + 2\sum_{i=1}^{k-1} r_i^2\right)}$$

Ljung-Box Q 用于检验一组自相关性是否与零存在显著差异，或检验模型残差是否可以与白噪声区分。Q 是检验统计量。

p 值 Ljung-Box 检验的 p 值。

偏自相关性图

“偏自相关性”选项显示或隐藏时间序列基本诊断图中的以下列：

滞后 各点之间的期间数。

偏自相关性 第 k 个滞后期的偏自相关性。

这些条以图形方式来表示偏自相关性的大小。蓝线表示近似 95% 预测限的 ±2 个标准误差，其中标准误差的计算方式如下：

$$\text{SE}_k = \frac{1}{\sqrt{n}} \quad \text{（对于所有 } k\text{）}$$

变差图

“变差图”选项显示或隐藏时间序列基本诊断图中的以下列：

滞后 各点之间的期间数。

变差图 变差图测量相距 k 个滞后期的点差分的方差，并将它与相距 1 个滞后期的点差分的方差进行比较。变差图是从自相关性计算得到的，公式如下：

$$V_k = \frac{1 - r_{k+1}}{1 - r_1}$$

其中 r_k 是滞后 k 的自相关性。

自回归系数图

“自回归系数”选项显示或隐藏时间序列基本诊断图中的以下列：

滞后 各点之间的期间数。

自回归系数 这些系数近似于您从拟合高阶纯自回归模型获得的那些系数。

“时间序列”平台选项

- “时间序列诊断”
- “差分和分解”
- “ARIMA 和季节性 ARIMA 模型”
- “平滑模型”
- “状态空间平滑模型”
- “转换函数模型”
- “平滑模型规格窗口”

时间序列诊断

图形 显示控制时间序列图外观的选项子菜单。

时间序列图 显示或隐藏时间序列图。

显示点 显示或隐藏时间序列图中的点。

连接线 显示或隐藏连接时间序列图中点的线。

均值线 在时间序列图中显示或隐藏描绘时间序列均值的水平线。

自相关性 在时间序列基本诊断图中显示或隐藏自相关性图。自相关性图描述时间序列中针对给定的时间间隔 （滞后）的所有点对之间的相关性。请参见第 290 页的 “自相关性图”。

提示：样本的自相关性图通常称为**样本自相关性函数**。

偏自相关性 在时间序列基本诊断图中显示或隐藏偏自相关性图。偏自相关性图描述时间序列中针对给定的时间间隔 （滞后）的所有点对之间的偏相关性。请参见第 290 页的 “偏自相关性图”。

提示：自相关性图和偏自相关性图可帮助您确定时间序列是否平稳（即它在一段时间内具有固定的均值和标准差）以及可能适合拟合时间序列的模型。

变差图 在时间序列基本诊断图中显示或隐藏变差图。请参见第 290 页的 “变差图”。

自回归系数 在时间序列基本诊断图中显示或隐藏自回归 (AR) 系数的最小二乘估计值图。请参见第 291 页的 “自回归系数图”。

谱密度 显示或隐藏谱密度图，谱密度图是周期和频率的函数。谱密度选项还显示白噪声检验报表，它给出对数据的两个检验结果。请参见第 312 页的 “‘谱密度’报表”和第 321 页的 “谱密度的统计详细信息”。

差分和分解

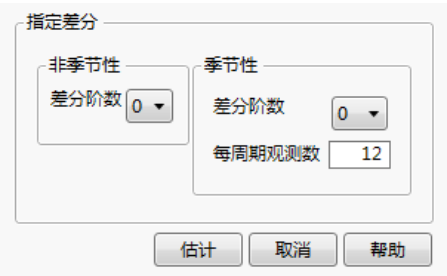
差分 显示“差分说明”窗口（图 18.5）。使用该窗口可以指定要应用于时间序列的差分运算。在时间序列中对值求差分可以将不平稳序列转换为平稳序列。按以下等式给出差分序列：

$$w_t = (1-\mathrm{B})^d(1-\mathrm{B}^s)^D y_t$$

其中 t 是时间索引， **B** 是由 $\mathbf{B}y_t = y_{t-1}$ 定义的后移算子。

注意：很多时间序列不具有固定均值，如具有趋势或季节性的时间序列。这样的不平稳序列不适合用假定平稳时间序列的时间序列模型 （如 ARMA 模型）来描绘。去除趋势和 / 或季节性将创建一个平稳的差分序列，使您可以使用假设序列是平稳的模型描绘序列。

图 18.5 “指定差分”窗口



使用 “指定差分”窗口可以指定 “非季节性差分阶数” d、“季节性差分阶数” D 和 “每周期观测数” s。差分阶数为 0 则相当于没有进行相应的差分运算。每次指定差分运算并点击**估计**时，将在报表窗口中显示新差分报表。请参见第 312 页的 “‘时间序列’平台的更多示例”。

分解 （在启动窗口中选定 “针对保留进行预测”或 “使用 Box-Cox 变换”时不可用。）显示分解方法的子菜单。时间序列数据的分解用于隔离和去除时间序列中的线性趋势和季节性周期。这有助于更好地估计模型。提供三个 “分解”选项。

线性趋势 使用线性回归模型估计时间序列的线性趋势并去除数据中的线性趋势。将去除趋势的序列的“时间序列”报表连同线性趋势信息添加到报表窗口。请参见第 289 页的“时间序列分析报表”和第 302 页的“线性趋势报表”。

删除周期 使用单个余弦波估计时间序列的周期性成分，然后将它从数据中去除。选择“删除周期”选项时，显示“定义周期”对话框。使用该对话框窗口，您可以指定每个周期的单位数并指示是否应从数据中减去一个常数。将去除周期的序列的“时间序列”报表连同周期信息添加到报表窗口。请参见第 289 页的“时间序列分析报表”和第 302 页的“周期报表”。

X11 使用美国人口普查局开发的 X-11 方法（Shiskin 等人 1967）去除趋势和季节效应。有关 X-11 方法的详细信息，请参见第 321 页的“X-11 分解的统计详细信息”。选中时，显示“选择分解类型”对话框。使用该对话框窗口可以指定乘法或加法 X-11 调整。点击**确定**后，将一个 X11 报表添加到报表窗口。请参见第 302 页的“X11 报表”。

X11 选项仅适用于每月或每季度数据。**X，时间 ID** 列必须为按月度或季度等间距分隔的数值列，不能有任何时间间隔或缺失值。若用来进行 X11 分析的时间列不满足这些要求，JMP 会返回错误。有关 X11 分析的适当时间列的示例，请参见第 313 页的“创建合适的“时间 ID”列”。

注意：选择**线性趋势**或**删除周期**选项时，JMP 将一列添加到数据表，其中包含去除趋势或去除周期的数据。若选择该选项时该列在数据表中已存在， JMP 将覆盖现有列。

提示：通常您按以下顺序开始分解：删除任何线性趋势，然后删除长周期 （如 12 个月的周期），接着删除短周期 （如 6 个月的周期）。

显示滞后图 显示或隐藏一个图，它在 Y 轴上具有时间 *t* 处的观测，在 X 轴上具有时间 *t* +/- *p* 处的观测。+/- *p* 即为滞后。该图可帮助确定时间 *t* 处的观测与时间 *t* +/- *p* 处的观测的关系。若该图不存在可识别的结构，则观测不相关。但是，若该图存在明显的结构，则指示一段时间上的观测之间存在一定关系。构建时间序列模型时，标识该结构很有帮助。

显示 Box-Cox 变换图 （若在启动窗口中指定了 “使用 Box-Cox 变换”则不可用。）显示或隐藏 Box-Cox 变换图。 Box-Cox 变换是基于 Lambda 参数 (λ) 的幂变换并使用以下公式：

$$Y_\lambda = \begin{cases} \dfrac{y^\lambda - 1}{\lambda} & \text{若 } \lambda \neq 0 \\ \ln(y) & \text{若 } \lambda = 0 \end{cases}$$

Box-Cox 变换图显示变换后的时间序列图和用于设置 Lambda 参数 (λ) 的选项。Lambda 框显示参数的当前值。您还可以使用 Lambda 滑块更改参数的值，更改后会自动更新该图。“下限”框和 “上限”框设置滑块的范围。

提示：使用 Box-Cox 变换图可查看不同的 Lambda 值的变换时间序列。一旦决定使用某个值，重新启动分析并选择**使用 Box-Cox 变换**，然后在 “Box-Cox 的 Lambda”旁边输入所需的 Lambda。这会针对变换的序列运行 “时间序列”平台。

交叉相关性 （仅在“转换函数分析”红色小三角菜单中可用。）在报表上显示或隐藏交叉相关性图。该图的长度是自相关图长度的两倍，或 2xACF 长度 + 1。该图同时以数值和图形的形式展示了输出序列相对于所有输入序列的相关性。蓝线指示两个标准误差。

注意：对于交叉相关图，原假设下标准误差计算为 1/ 平方根 (n - k)，其中， n 是非缺失数据值的数量， k 是自相关滞后期数。

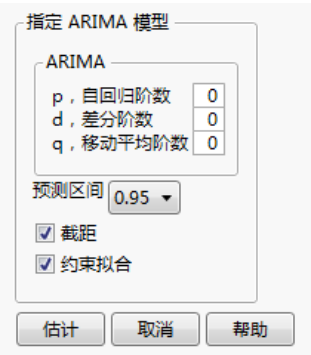
预白化 （仅在 “输入序列”红色小三角菜单中可用。）显示 “指定预白化”窗口，您可以在其中设置预白化阶数。预白化是用于帮助标识转换函数模型的方法。这种方法涉及拟合输入序列的 ARIMA 模型，以便残差等同于白噪声。同一模型随后用于估计输出序列。您可以使用过滤的输入和过滤的输出序列之间的交叉相关性来确定适用于转换函数模型的滞后。有关预白化的信息，请参见 Box 等人 (1994)。

ARIMA 和季节性 ARIMA 模型

ARIMA 显示 “指定 ARIMA”窗口，您可以在其中指定要拟合的 ARIMA 模型。 ARIMA 模型利用过去值和一系列误差（也称为**随机扰动项**或**新息**）的线性组合来预测时间序列的将来值。针对时间序列指定的 ARIMA 模型， ARIMA 模型执行最大似然拟合。请参见第 326 页的 “ARIMA 模型”。

注意：ARIMA 模型通常表示为 ARIMA(p,d,q)。若 p、 d 或 q 中的任何项为零，则通常会删除相应的字母。例如，若 p 和 d 为零，则模型只是移动平均模型，用 MA(q) 表示。

图 18.6 “指定 ARIMA”窗口



p，自回归阶数 多项式 φ(B) 算子的阶数 p。

d，差分阶数 差分算子的阶数 d。

q，**移动平均阶数** 差分算子 θ(B) 的阶数 *q*。

预测区间 支持您为预测区间设置介于 0 到 1 的预测水平。

截距 确定截距项 μ 是否是模型的一部分。

约束拟合 若选中，拟合过程将约束自回归参数以始终将其限定在稳定区域内，并将移动平均参数限定在可逆区域内。

提示：若拟合器难以找到真正的最优值或您要加快拟合，请取消选择“约束拟合”选项。可以使用“模型汇总”表来查看生成的拟合模型是否稳定和可逆。

一旦指定模型并点击**估计**，“模型报表”即添加至报表窗口。请参见第 301 页的“报表”。

季节性 ARIMA 显示“指定季节性 ARIMA”窗口，您可以在其中指定要拟合的季节性 ARIMA 模型。该窗口与“指定 ARIMA”窗口具有相同的元素，但是它还要指定季节性元素。新增的“每周期观测数”选项用于指定每个周期的观测数，用 *s* 表示。有关季节性 ARIMA 模型的详细信息，请参见第 326 页的“季节性 ARIMA 模型”。

注意：季节性 ARIMA 模型表示为季节性 ARIMA(*p*,*d*,*q*)(*P*,*D*,*Q*)*s*。

一旦指定模型并点击**估计**，“模型报表”即添加至报表窗口。请参见第 301 页的“报表”。

平滑模型

显示平滑模型的子菜单。您选择一个平滑模型后，将显示规格窗口。请参见第 300 页的“““平滑模型”窗口”。对于指定的每个模型，在报表窗口中显示“平滑模型报表”。请参见第 305 页的“ARIMA 和季节性 ARIMA 模型报表”。平滑模型用模型表示时间序列的变化过程：

$$y_t = \mu_t + \beta_t t + s(t) + a_t$$

其中

μ_t 是时变均值项

β_t 是时变斜率项

$s(t)$ 是 s 时变季节性项中的一项

a_t 是随机扰动项

有关一般平滑模型等式的详细信息，请参见第 322 页的“平滑模型的统计详细信息”。提供以下平滑模型：

简单移动平均 （在启动窗口中选定“针对保留进行预测”或“使用 Box-Cox 变换”时不可用。）通过使用几个相邻点的平均值（由平滑窗口定义）估计值的模型。使用“简单平滑平均规格”窗口可以对平滑窗口各个参数进行设置。一旦指定平滑窗口选项，就会显示“简单移动平均”报表。默认情况下，该报表生成等于时间窗口中连续观测平均值的标绘值。可以在同一图上添加和显示多个简单移动平均模型。请参见第 299 页的““简单平滑平均规格”窗口”。

简单指数平滑 具有水平成分的模型。请参见第 322 页的 “简单指数平滑”。

双指数平滑 具有水平成分和趋势成分的模型。这是线性指数平滑的特殊情况。请参见第 323 页的 “二次 (Brown) 指数平滑”。

线性指数平滑 具有水平成分和趋势成分的模型。请参见第 323 页的“线性 (Holt) 指数平滑”。

阻尼趋势线性指数平滑 具有水平成分和阻尼趋势成分的模型。该模型适合显示比线性趋势更复杂的趋势的序列。请参见第 324 页的 “阻尼趋势线性指数平滑”。

季节性指数平滑 具有水平成分和季节性成分的模型。请参见第 324 页的“季节性指数平滑”。

Winters 方法 具有水平成分、趋势成分和季节性成分的模型。请参见第 325 页的 “Winters 方法 （加法）”。

注意：每个平滑模型具有等价的 ARIMA 模型。您可能不能使用 ARIMA 选项指定等价的 ARIMA 模型，因为一些平滑模型从根本上对 ARIMA 模型参数进行了约束，而这些约束方式在 ARIMA 选项下是无法实现的。

状态空间平滑模型

显示“指定状态空间平滑模型”窗口，该窗口支持您拟合 Hyndman et al. (2008) 中定义的多种状态空间平滑模型。(2008)。基于状态空间平滑模型的误差、趋势成分和季节性成分定义状态空间平滑模型。

- 误差可以是加法 (A) 或乘法 (M)。
- 趋势成分可以为“无 (N)”、“加法 (A)”、“带阻尼的加法 (A_d)”、“乘法 (M)”或“带阻尼的乘法(M_d)”。
- 季节性成分可为“无 (N)”、“加法 (A)”或“乘法 (M)”。

特定模型可用其 ETS（误差、趋势、季节性）来表示。使用“指定状态空间平滑模型”窗口中的复选框为所需模型选择误差、趋势和季节性。点击**选择推荐项**选中与平台推荐的模型对应的复选框。该窗口打开时会显示选定的推荐模型。点击**全选**可选中所有复选框，点击**取消全选**可取消选中所有复选框。该窗口还包含以下选项：

周期 指定模型拟合过程中要考虑的季节性值。

约束参数 以这样一种方式约束参数，即：观测离当前越远，对模型当前状态的影响就越小。在“状态空间平滑”模型中，在给定所有以前的观测的情况下，时间 *t* 的预测与截至时间 *t* 的所有观测的加权和相同。这些权重是参数的函数。因此，约束参数可以确保过去观测的权重趋近零，并且观测距离现在越远，权重接近零的速度就越快。

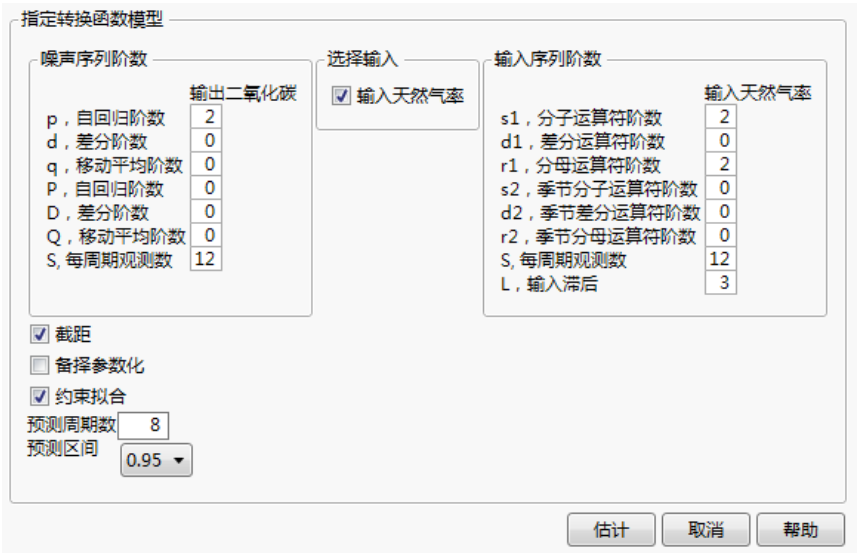
点击**确定**时，将拟合指定的一组模型。每个状态空间平滑模型的汇总值都添加到“模型比较”表中。单个拟合报表添加到报表窗口中。

转换函数模型

转换函数 （仅在“转换函数分析”红色小三角菜单中可用。）显示“指定转换函数模型”窗口。生成转换函数模型与生成 ARIMA 模型相似，都是一个探索、拟合和比较模型的迭代过程。在生成模型前和数据探索过程中，有时预白化数据很有用。请参见第 294 页的“预白化”。有关转换函数的详细信息，请参见第 327 页的“转换函数的统计详细信息”。

注意：目前，“转换函数”选项对缺失值提供有限支持。由于“时间序列”平台也将排除的行视为缺失值，因此无法对包含排除行的数据表拟合出转换函数模型。

图 18.7 “指定转换函数模型”窗口



“指定转换函数模型”窗口包含以下部分：

噪声序列阶数 包含噪声序列的规格。小写字母是非季节性多项式的系数，大写字母则是季节性多项式的系数。

选择输入 允许您选择模型的输入序列。

输入序列阶数 包含输入序列的规格。前三个阶数的设置与非季节性多项式有关。接下来的四个阶数的设置与季节性多项式有关。最后的选项用于输入滞后期数。

还有另外三个控制模型拟合的选项：

截距 指定模型是否具有截距。若未选中，则假定 μ 为 0。

备择参数化 指定一般回归系数是否要从分子多项式中提取出来。

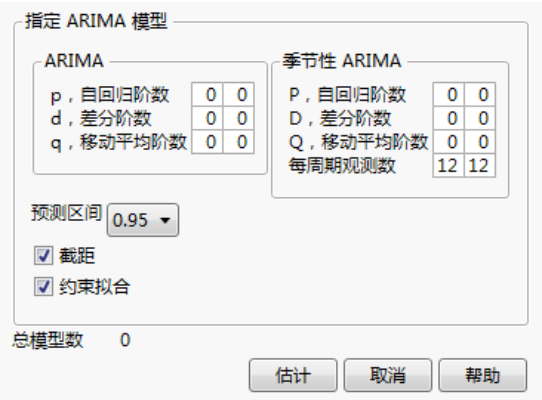
约束拟合 选择是否对 AR 和 MA 的系数进行约束。

预测周期数 指定用于预测的预测周期数。若有位于数据表末尾的行（它们包含 Y 变量的缺失值和输入变量的非缺失值），则在初始预测设置中使用这些行。这些输入变量的值视为输入变量的将来值。

预测区间 指定预测区间的置信水平。

ARIMA 模型组 显示“ARIMA 模型组”窗口，您可以在其中通过指定阶数范围来拟合各种 ARIMA 或季节性 ARIMA 模型。您将范围输入窗口时，相应更新“总模型数”。

图 18.8 ARIMA 模型组指定窗口



一旦指定模型并点击**估计**，每个指定模型的“模型报表”即添加至报表窗口。请参见第 301 页的“报表”。

合并并保存模型的预测 使用报表中所有模型拟合的合并结果创建一个新的数据表。

保存谱密度 创建一个包含谱密度和周期图的新数据表，其中第 *(i+1)* 行对应于频率 $f_i = i / N$（即 $1 / N$ 的第 i 个谐波）。新数据表具有以下列：

周期 第 i 个谐波的周期 $1 / f_i$。

频数 谐波的频率 f_i。

角频率 谐波的角频率 $2\pi f_i$。

正弦 傅里叶正弦系数 a_i。

余弦 傅里叶余弦系数 b_i。

周期图 周期图 $I(f_i)$。

谱密度 谱密度，即平滑后的周期图。

预测周期数 显示一个窗口，您可以在其中设置为拟合模型预测的将来周期数。在“时间序列”启动窗口中设置初始值。更改它后，所有现有和将来的预测结果都使用新周期数。

最多迭代次数 显示一个窗口，在将来拟合的 ARIMA 模型进行优化迭代运算时，您可以重置最大迭代次数。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

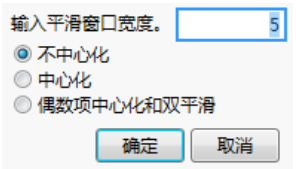
保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

平滑模型规格窗口

“简单平滑平均规格”窗口

当您选择“简单移动平均”作为平滑模型时，显示“简单平滑平均规格”窗口。设 w 为简单移动平均 (SMA) 模型中的平滑窗口宽度。则针对某一时间点 t，$f_t=(y_t+y_{t-1}+y_{t-2}+...y_{t-(w-2)}+y_{t-(w-1)})/w$ 为 w 个连续观测的平均值。

图 18.9 “简单平滑平均规格”窗口



输入平滑窗口宽度 平滑窗口宽度 w，它定义要平均的连续点数。窗口宽度越大，序列越平滑。

不中心化 平滑窗口由时间点 t（时间序列要估计的时间点）之前的点到时间点 t（包括 t）来构造。换句话说，f_t 是时间 t 的标绘值。

中心化 平滑窗口以时间序列要估计的时间点为中心。

- 对于奇数 w，f_t 是时间 t-(w-1)/2 的标绘值。
- 对于偶数 w，f_t 是时间 t-(w-1)/2 的标绘值。保存到数据表时，f_t 位于 t-(w-2)/2 处。

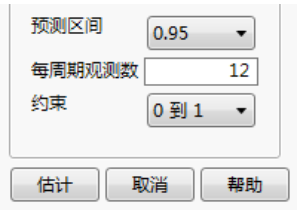
偶数项中心化和双平滑 对于偶数 w，平滑窗口无法以正要估计序列的时间点为中心。该选项创建两个几乎中心化的平滑窗口，取它们的平均值。按以下方式计算平滑估计值：

$$f_{t-\frac{w}{2}} = \frac{y_t + 2\sum_{i=1}^{w-1} y_{t-i} + y_{t-w}}{2w}$$

“平滑模型”窗口

当您选择不是“简单移动平均”的一个平滑模型选项时，显示“平滑模型”规格窗口。窗口标题和可用选项取决于您选择的平滑模型选项。

图 18.10 “平滑模型”规格窗口



预测区间 支持您设置预测区间的预测水平。

每周期观测数 （仅对季节性平滑模型可用。）支持您在季节性平滑模型中设置每周期观测数。

约束 允许您指定在拟合期间要对平滑权重施加的约束类型。以下约束选项可用：

0 到 1 约束平滑权重值处于范围 0 到 1 内。

无约束 允许任意范围的参数。

稳定可逆 约束参数以便等价的 ARIMA 模型是稳定可逆的。

定制 展开对话框以允许设置各个平滑权重的约束。根据权重名称旁边的弹出菜单设置，每个平滑权重可以为**界限**、**固定**或**无约束**。输入固定或界限权重的值时，这些值可以为正负实数。

图 18.11 定制平滑权重

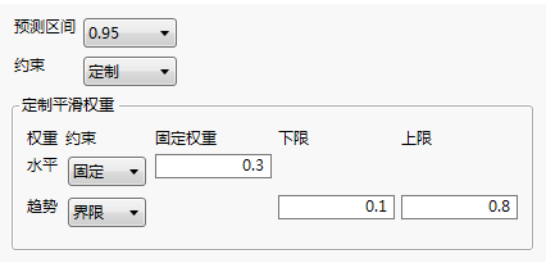


图 18.11 中显示的示例具有固定为 0.3 的水平权重 (α)，界限为 0.1 和 0.8 的趋势权重 (γ)。在该例中，趋势权重的值可以在范围 0.1-0.8 内变动，而水平权重则固定为 0.3。请注意，您可以使用这些定制约束预先指定所有平滑权重。这样就不用从数据估计权重，尽管仍计算预测和残差。

报表

- “差分报表”
- “分解报表”
- ““模型比较”报表”
- “ARIMA 和季节性 ARIMA 模型报表”
- ““状态空间平滑”报表”
- “转换函数报表”
- ““谱密度”报表”

差分报表

“差分报表”包含差分序列的自相关性和偏自相关性图。这些图可用于确定差分序列是否平稳。

在“时间序列”平台中提供的 ARIMA 和季节性 ARIMA 模型包含差分运算。在两步过程中，这些模型首先根据差分运算求时间序列差分，然后拟合差分序列。“差分”选项是有用的预处理工具，可以决定为 ARIMA 模型指定的差分阶数。

“差分”红色小三角菜单包含以下选项：

图形 显示控制差分序列图外观的选项子菜单。请参见第 291 页的 ““时间序列”平台选项”。

自相关性 显示或隐藏差分序列的自相关性。

偏自相关性 显示或隐藏差分序列的偏自相关性。

变差图 显示或隐藏差分序列的变差图。

保存 将包含差分序列中值的新列保存到原始数据表。前面的一些元素会在差分过程中丢失，因此它们在保存的“差分”列中表示为缺失值。

删除拟合 从报表窗口中删除“差值”报表。

分解报表

本节提供有关从三个分解选项获得的报表的详细信息：

- 第 302 页的“线性趋势报表”
- 第 302 页的“周期报表”
- 第 302 页的“X11 报表”

线性趋势报表

包含拟合数据的线性回归模型中的 β_0 和 β_1 值。

$$趋势_t = \beta_0 + \beta_1 time$$

去除趋势的序列等于 $D_t = O_t$ - 趋势$_t$，其中 O_t 是原始时间序列。

周期报表

包含拟合数据的周期性成分的值：

$$周期_t = C + A \cdot \cos\left(2 \cdot \pi \cdot \left(\left(\frac{1}{U}\right) \cdot t + P\right)\right)$$

参数值定义如下：

- C 是（可选的）常数
- A 是余弦波的振幅
- U 是每周期单位数
- P 是余弦波的相位
- t 是给定观测的行号减去 1

去除周期的序列等于 $D_t = O_t$ - 周期$_t$，其中 O_t 是原始时间序列。

X11 报表

根据您选择的分解类型， X11 选项会添加 “X11 - 乘法”报表或 “X11 - 加法”报表。这些报表包含相同的四个图：

原始和调整 在原始时间序列 O_t 上叠加 X11 调整后的时间序列。 X11 调整值对于乘法调整为 O_t/S_t，对于加法调整为 $O_t - S_t$。

D10 - 最终季节因子 标绘随时间变化的季节因子成分 S_t。

D12 - 最终趋势周期 标绘随时间变化的趋势周期成分 C_t。

D13 - 最终不规则序列 标绘随时间变化的不规则成分 I_t。

X11 报表选项

X11 报表具有以下红色小三角选项：

显示表 显示或隐藏 X11 汇总表，如 Shiskin 等人 (1967)。这些表分组为五个类别 （标记为 B 到 F），如表 18.1 中所述。

保存列 将四列保存到数据表：季节调整时间序列、季节性成分 (S_t)、趋势周期成分 (C_t) 和不规则序列成分 (I_t)。

保存全部列 对于由**显示表**选项在报表中生成的所有表，保存列到数据表。

删除拟合 从报表窗口中删除 X11 报表。

表 18.1 X11 输出表的类别说明

字母前缀	类别说明
B	季节性成分、趋势周期成分和不规则成分的初步估计值
C	季节性成分、趋势周期成分和不规则成分的中间估计值
D	季节性成分、趋势周期成分和不规则成分的最终估计值
E	分析表
F	汇总测度

有关 X11 输出表的内容的详细信息，请参见 Shisken et. al.(1967) 或 SAS/ETS 15.2 User's Guide（搜索“The output from PROC X11”）。

“模型比较”报表

拟合模型后，在报表窗口中显示“模型比较”报表。该报表包含“模型比较”表和模型图。每次拟合新模型时，将一个新行添加到“模型比较”表，并用唯一的颜色编码。“模型比较”表汇总每个模型的拟合统计量，用于比较针对同一时间序列拟合的几个模型。默认情况下，这些模型按 AIC 统计量的升序排序。要按不同统计量对模型排序，请右击“模型比较表”中的任意列并选择“按列排序”。或者，您还可以点击列标题进行排序。有关拟合统计量的定义，请参见第 306 页的“‘模型汇总’表”。对“模型比较表”唯一的一个拟合统计量是“权重”。该拟合统计量是标准化的 AIC 权重。按以下方式计算模型的 AIC 权重：

$$\text{AIC 权重} = \exp[-0.5(\text{AIC} - \text{BestAIC})] / \sum_{k=1}^{K} (\exp[-0.5(\text{AIC}_k - \text{BestAIC})])$$

K 是模型总数，AIC_k 是模型 k 的 AIC 值，BestAIC 是具有最小 AIC 值的模型的 AIC 值。

若指定保留集，则对训练数据计算的拟合统计量不会显示在“模型比较”表中。相反，会计算以下量度来确定各模型对保留集的预测性能：均方根误差 (RMSE)、均方误差 (MSE)、平均绝对百分比误差 (MAPE) 和平均绝对误差 (MAE)。默认情况下，这些模型按 RMSE 统计量的升序排序。要按不同统计量对模型排序，请右击“模型比较表”中的任意列并选择“按列排序”。或者，您还可以点击列标题进行排序。有关拟合统计量的定义，请参见第 306 页的“‘模型汇总’表”。

警告：若未指定保留集但要拟合状态空间平滑模型和 ARIMA 模型的组合，则“模型比较”表顶部会显示警告消息。在这种情况下，状态空间平滑模型与 ARIMA 模型不具有可比性，因为似然值和基于似然的信息准则（如 AIC 统计量）在两个不同类型的模型之间不具有可比性。不过，若指定保留集，则可以比较模型，因为使用了 RMSE 统计量。

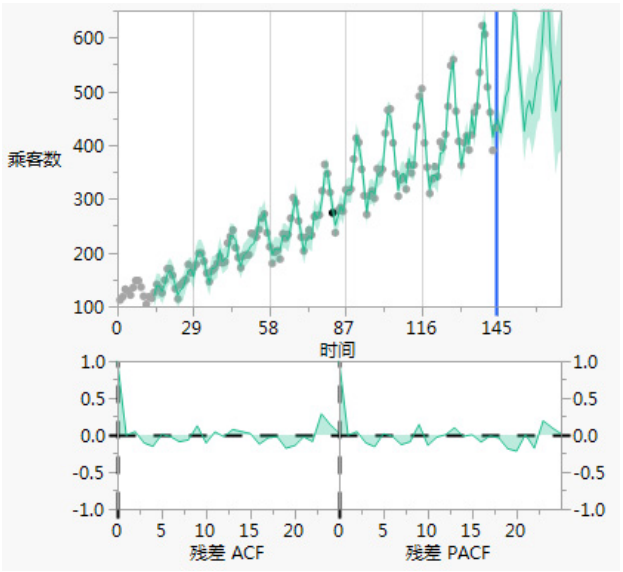
图 18.12 “模型比较”表

模型比较

报表	图形	模型	自由度	方差	AIC	SBC	R 方	-2LogLH	权重	.2 .4 .6 .8	均值绝对百分误差	均值绝对误差
☑	☐	季节性 ARIMA(0, 1, 1)(0, 1, 1)12	128	138.49122	1020.9047	1029.5302	0.990	1014.9047	1.000000		3.197010	8.990978
☑	☐	ARIMA(1, 1, 1)	140	978.9437	1394.1215	1403.0101	0.932	1388.1215	0.000000		8.687058	24.364166
☑	☐	ARMA(1, 1)	141	989.13835	1407.7483	1416.6577	0.919	1401.7483	0.000000		9.704486	25.348507
☑	☐	AR(1)	142	1134.3871	1426.1794	1432.1190	0.909	1422.1794	0.000000		9.873726	26.556219
☑	☐	MA(1)	142	4264.6121	1616.8626	1622.8022	0.693	1612.8626	0.000000		23.865796	53.716974

您可以使用**报表**复选框选择在报表窗口中显示的完整模型报表。两个模型图显示在“模型比较”表右侧。顶部图是数据、预测和预测限的时间序列图。它的下面是残差的自相关性和偏自相关性函数图。您可以使用**图形**复选框选择在模型图上显示的模型。

图 18.13 模型图



“模型比较”报表选项

“模型比较”报表中的每个模型具有以下红色小三角菜单选项：

拟合新模型 打开模型的一个规格窗口。您可以更改设置以拟合另一模型。

模拟一次 （若指定了保留集，则不可用。）向时间轴的上端提供一次模型模拟。该模拟显示在“模型比较”时间序列图上。

模拟多次 （若指定了保留集，则不可用。）向时间轴的上端提供指定次数的模型模拟。这些模拟显示在 “模型比较”时间序列图上。

删除模型模拟 （若指定了保留集，则不可用。）删除模型的模拟。

删除全部模拟 （若指定了保留集，则不可用。）删除所有模型的模拟。

生成模拟 （若指定了保留集，则不可用。）生成模型的模拟，并将结果储存在数据表中。您可以指定随机种子、模拟次数和预测周期数。

设置种子 （若指定了保留集，则不可用。）指定用于生成模拟轨迹的种子。

删除拟合 从报表窗口中删除相应模型。

删除所有模型 从报表窗口中删除 “模型比较”表中列出的所有模型。

删除未选定项 从报表窗口中删除所有未选中 “报表”复选框的模型。

删除选定项 从报表窗口中删除所有选中 “报表”复选框的模型。

隐藏所有报表 从报表窗口中隐藏 “模型比较”表中列出的所有模型。这相当于取消选中所有“报表”复选框。

隐藏所有图形 从模型图中隐藏“模型比较”表中列出的所有模型。这相当于取消选中所有“图形”复选框。

注意：“时间序列”平台将排除行视为缺失值。若数据表中的最后几个观测被排除或缺失，则“模型比较”报表中的模拟选项将无法正常工作。该模拟过程使用时间序列的最后几个观测。若缺少最后几个观测，则模拟会在将来生成所有缺失值。

ARIMA 和季节性 ARIMA 模型报表

时间序列建模选项用于拟合序列的理论模型并使用拟合模型预测序列的将来值。这些选项还生成统计量和残差，以便您确定选择要使用的模型的合理性。您可以多次选择建模选项。每次选择模型时，该模型都会添加到“模型比较”表中。选中“模型比较”表中某模型旁边的“报表”复选框时，会为该模型生成一个报表。该报表在标题中指明相应的模型。

默认显示以下报表：

- “模型汇总”表
- “参数估计值”表
- 预测图
- 残差
- 迭代历史记录

“模型汇总”表 包含模型的拟合统计量。在下面的公式中，n 是序列的长度，k 是模型中的拟合参数数目。

自由度 拟合中的自由度数，计算公式为：$n - k$。

创新平方和 创新平方和。

残差平方和 残差平方和。

方差估计值 创新平方和除以自由度数 $(n - k)$。这是第 326 页的“ARIMA 模型”一节中所述的随机扰动项 a_t 的方差的样本估计值。

标准差 方差估计值的平方根。这是随机扰动项 a_t 的标准差的样本估计值。

Akaike A 信息准则 AIC 值，计算为 –2对数似然 + 2k。AIC 值越小指示拟合效果越好。

Schwarz Bayes 准则 SBC 值，计算为 –2 对数似然 + k*ln(n)。SBC 值越小指示拟合效果越好。Schwarz Bayesian 准则等价于 Bayesian 信息准则 (BIC)。

R 方 R 方值，计算如下：

$$R^2 = 1 - \frac{\text{SSE}}{\text{SST}}$$

其中

$$\text{SST} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

$$\text{SSE} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$\hat{y}_i$ 是“单步”预测

$\bar{y}$ 是均值

若模型拟合序列效果不好，则模型误差平方和 SSE 可能大于总平方和 SST，导致 R^2 为负数。

调整 R 方 调整的 R^2 值，计算如下：

$$1 - \left[\frac{(n-1)}{(n-k)}(1 - R^2)\right]$$

均值绝对百分误差 绝对百分比误差均值，计算如下：

$$\frac{100}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

均值绝对误差 绝对误差均值，计算如下：

$$\frac{1}{n}\sum_{i=1}^{n}|y_i-\hat{y}_i|$$

–2 对数似然 在最佳拟合参数估计值处求值的负对数似然函数的两倍。值越小表示拟合效果越好。请参见《拟合线性模型》。

稳定 指示自回归算子是否稳定。即是否 $\phi(z)=0$ 的所有根位于单位圆之外。

可逆 指示移动平均算子是否可逆。即是否 $\theta(z)=0$ 的所有根位于单位圆之外。

注意：ϕ 和 θ 算子在第 326 页的 “ARIMA 模型”一节中定义。

“参数估计值”表 显示时间序列模型参数的估计值。每个模型类型具有自己的一组参数，它们在有关特定时间序列模型的节中介绍。每个 “参数估计值”表包含以下列：

项 参数的名称，它在每个模型类型的对应节中介绍。一些模型包含**截距**或均值项。在这些模型中，还显示相关的**常数估计值**。常数估计值的定义在 ARIMA 模型的描述下面给出。

因子 （仅对乘法季节性 ARIMA 模型显示。）包含参数的模型的因子。在乘法季节性模型中，因子 1 是非季节性的，因子 2 是季节性的。

滞后 （仅对 ARIMA 和季节性 ARIMA 模型显示。）滞后期数或者与参数相乘的后移算子的阶数。

估计值 时间序列模型的参数估计值。

标准误差 参数估计值的标准误差估计值。这些估计值用于计算检验和预测区间。

t 比 假设每个参数为零的检验统计量。参数的检验统计量是参数估计值与标准误差的比。若假设为真，则该统计量近似服从 Student *t* 分布。根据一般规则，通常查找绝对值大于 2 的 *t* 比来判断显著性，因为它近似于 0.05 显著性水平。

概率>|t| 为每个参数计算的观测 *p* 值。*p* 值是得到 *t* 比的绝对值大于计算值的概率（在假设成立的前提下）。

常数估计值 （仅针对包含截距项或均值项的模型显示。）常数估计值的定义在 ARIMA 模型下给出。

Mu （仅对 ARIMA 和季节性 ARIMA 模型显示。）ARIMA 或季节性 ARIMA 模型的截距值估计值。

预测 显示时间序列的观测值和预测值的图。该图被一条垂直线分为两个区域。

若指定了保留集，则会用一条垂直线将训练数据与保留数据分开。在垂直线的左侧，一步超前预测叠加在训练数据点上。在垂直线的右侧，模型预测的值叠加在保留数据点上。同时还显示预测的预测区间。您可以使用平台启动窗口中的 “预测周期数”选项来控制预测值的数量。

若未指定保留集，则垂直线将观测数据与未来预测分开。在垂直线的左侧，“单步”预测叠加在观测的数据点上。在垂直线的右侧，显示模型所预测的将来值和预测的预测区间。通过在平台启动窗口中使用 “预测周期数”选项或通过从 “时间序列”红色小三角菜单中选择“预测周期数”，您可以控制将来预测值的个数。

残差 显示基于拟合模型的残差值的图形。这些值是时间序列的观测值减去“单步”预测值得到的。还显示这些残差的自相关性和偏自相关性报表。可以使用这些报表来确定拟合模型是否充分描述了原有的数据。若拟合模型适合，则残差图中的点应在零附近呈正态分布，并且残差的自相关性和偏自相关性在任何大于零的滞后期内都不存在显著性。

维持序列的预测误差 （仅当指定保留集时才可用。）保留集中的观测的预测误差图。还显示这些误差的自相关性和偏自相关性报表。

迭代历史记录 包含每次迭代的目标函数的值。这对于诊断拟合过程的问题很有用。尝试拟合很不适合数据的模型可能导致无法收敛到似然的最优值。“迭代历史记录”表包含以下信息：

迭代 迭代编号。

迭代历史记录 每步的目标函数值。

步进 迭代步的类型。

目标值准则 目标函数的梯度的范数。

ARIMA 和季节性 ARIMA 模型报表选项

每个 ARIMA 和季节性 ARIMA 模型报表都有一个包含以下选项的红色小三角菜单：

显示点 显示或隐藏预测图中的数据点。

显示预测区间 显示或隐藏预测图中的预测区间。

保存列 创建一个新数据表，它包含表示模型结果的列。

保存预测公式 （若指定了保留集，则不可用。）将数据和预测公式保存到新数据表。

创建 SAS 作业 创建在 SAS 中复制模型分析的 SAS 代码。

提交至 SAS 将复制模型分析的 SAS 代码提交到 SAS。若您未连接到 SAS 服务器，该选项将引导您完成连接过程。

残差统计量 控制为模型显示哪些残差统计量。这些显示内容在第 291 页的 “‘时间序列’平台选项”一节中介绍。但是，它们应用到残差序列。

删除拟合 从报表中删除单个模型拟合。

“状态空间平滑”报表

每个“状态空间平滑”模型都添加到“模型比较”表中。若在“模型比较”表中选中某个状态空间平滑模型的“报表”复选框，则会显示“状态空间平滑”报表。每个“状态空间平滑”报表都有可标识该模型的唯一分级显示项名称。包括以下报表：

模型类型 根据误差、趋势和季节性描述序列的最佳拟合模型。

模型汇总 包含模型的拟合统计量。在下面的公式中， n 是序列的长度， k 是模型中的拟合参数数目。

–2 对数似然 在最佳拟合参数估计值处求值的负对数似然函数的两倍。值越小表示拟合效果越好。请参见《拟合线性模型》。

AIC Akaike 信息准则 (AIC) 值，计算为 –2对数似然 + 2k。AIC 值越小指示拟合效果越好。

BIC Schwarz Bayes 准则 (SBC) 值，计算为 –2对数似然 + k*ln(n)。SBC 值越小指示拟合效果越好。Schwarz Bayesian 准则等价于 Bayesian 信息准则 (BIC)。

参数数目 信息准则计算中的参数数目。该数目比似然函数中的自由参数数目多一个。

注意：参数数目是参数估计值表中的参数个数。

Sigma 一步超前预测标准差的估计值。

R 方 R 方值，计算如下：

$$R^2 = 1 - \frac{\text{SSE}}{\text{SST}}$$

其中

$$\text{SST} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

$$\text{SSE} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$\hat{y}_i$ 是 “单步” 预测

$\bar{y}$ 是均值

若模型拟合序列效果不好，则模型误差平方和 SSE 可能大于总平方和 SST，导致 R^2 为负数。

调整 R 方 调整的 R^2 值，计算如下：

$$1 - \left[\frac{(n-1)}{(n-k)}(1 - R^2)\right]$$

均值绝对百分误差 绝对百分比误差均值，计算如下：

$$\frac{100}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

其中

$\hat{y}_i$ 是 “单步” 预测

均值绝对误差 绝对误差均值，计算如下：

$$\frac{1}{n}\sum_{i=1}^{n}\left|y_i-\hat{y}_i\right|$$

其中

$\hat{y}_i$ 是 “单步”预测

参数估计值 显示模型中每个参数的估计值、标准误差和 95% 置信区间的表。

预测 显示时间序列的观测值和预测值的图。请参见第 307 页的 “预测”。

一步超前预测误差 随时间变化的一步超前预测误差图。对于每个时间点 t，误差计算为 t 处的时间序列的观测值减去 t 处的一步超前预测值。若指定保留集，则仅为训练数据计算误差。还显示这些误差的自相关性和偏自相关性报表。

维持序列的预测误差 （仅当指定保留集时才可用。）保留集中的观测的预测误差图。还显示这些误差的自相关性和偏自相关性报表。

成分状态 一个报表，其中包含模型的每个状态的图以及相应的模型公式。每个图都包括序列的状态估计值和状态预测。若指定了保留集，则状态估计值用于训练部分，而状态预测用于保留部分。若未指定保留集，则状态估计值针对时间序列中直到最后一个观测之前的部分，而状态预测针对时间序列最后一个观测之后的部分。一个模型最多可以有三种状态：水平、趋势和季节性。

“成分状态”红色小三角菜单包含一个“保存状态”选项。选择该选项可将每个时间点的成分状态保存至单独的数据表。

乘法误差模型的一步超前相对预测误差 （仅可用于乘法误差模型。）随时间变化的一步超前相对预测误差图。对于每个时间点 t，误差计算为 t 的一步超前预测误差除以 t 的一步超前预测值。

“状态空间平滑”报表选项

每个状态空间平滑报表都有一个包含以下选项的红色小三角菜单：

显示点 显示或隐藏预测图中的数据点。

显示预测区间 显示或隐藏预测图中的预测区间。

保存列 创建一个新数据表，它包含表示模型结果的列。

删除拟合 从报表中删除单个模型拟合。

转换函数报表

将每个转换函数模型添加到“模型比较”表。若选中“模型比较”表中某个转换函数模型的报表复选框，将生成一个“转换函数模型”报表。若指定了保留集，则“模型比较”报表中还提供模型图。

每个“转换函数模型”报表包含以下报表：

- 模型汇总
- 参数估计值
- 残差
- 维持序列的预测误差
- 交互式预测
- 迭代历史记录

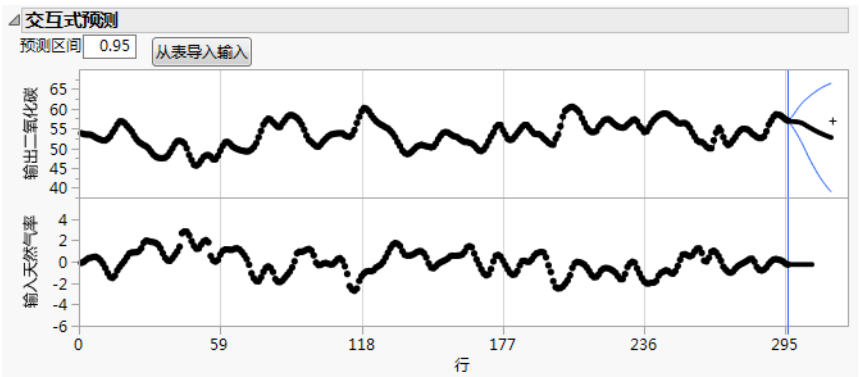
“模型汇总”、“参数估计值”、“残差”、“维持序列的预测误差”和“迭代历史记录”等报表中的信息与 ARIMA 和季节性 ARIMA 模型报表中的信息相同。有关这些报表的详细信息，请参见第 305 页的“ARIMA 和季节性 ARIMA 模型报表”。“参数估计值”表后面是模型公式，其中 **B** 是后移算子。

交互式预测

“交互式预测”报表根据“转换函数规格”窗口中指定的预测区间提供一个预测图。围绕预测的预测区间用蓝色显示。通过在图形上方的“预测区间”框中输入一个数，可以更改该预测区间的置信水平。

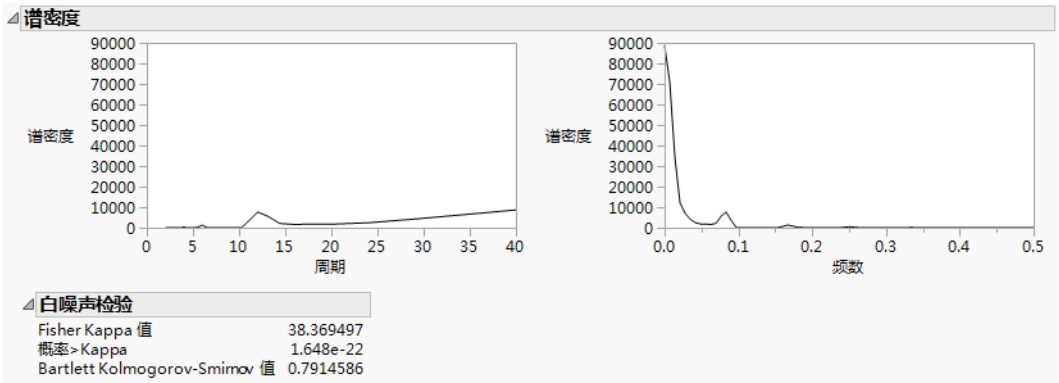
您可以拖动图形中的加号来更改图形中的预测周期数。在预测周期中，您可以使用“从表导入输入”按钮或通过拖动输入图形中的点到不同值来更改输入值。您的更改结果反映在输出图的预测周期中。

图 18.14 交互式预测图



“谱密度”报表

图 18.15 谱密度图和 “白噪声检验”报表



“白噪声检验”报表包含以下统计量：

Fisher Kappa 值 检验以下原假设：序列中的值取自方差为 1 的正态分布，而备择假设为序列存在某一周期性成分。 Kappa 是周期图 $I(f_i)$ 的最大值与它的平均值之比。

概率 > Kappa 若原假设为真，观测值大于 Kappa 的概率，它用以下等式表示：

$$Pr(k > \kappa) = 1 - \sum_{j=0}^{q} (-1)^j \binom{q}{j} \left[\max\left(1 - \frac{jk}{q}, 0\right)\right]^{q-1}$$

其中

$q = N / 2$ （若 N 为偶数）； $q = (N - 1) / 2$ （若 N 为奇数）

κ 是 Kappa 的观测值

若该概率小于显著性水平 α，则拒绝原假设。

Bartlett Kolmogorov-Smirnov 值 将标准化累积周期图与区间 (0, 1) 上均匀分布的累积分布函数进行比较。该检验统计量等于累积周期图和均匀分布累积分布函数的最大绝对差值。若它超过 $a/(\sqrt{q})$，则人们通常拒绝序列来自正态分布的假设。值 $a = 1.36$ 和 $a = 1.63$ 分别对应于显著性水平 5% 和 1%。

“时间序列”平台的更多示例

- “创建时间 ID 列的示例”
- “使用保留集的示例”

创建时间 ID 列的示例

该示例使用 SeriesP.jmp 样本数据表来显示如何执行时间序列分析。您首先创建一个适合“时间 ID”的新列。

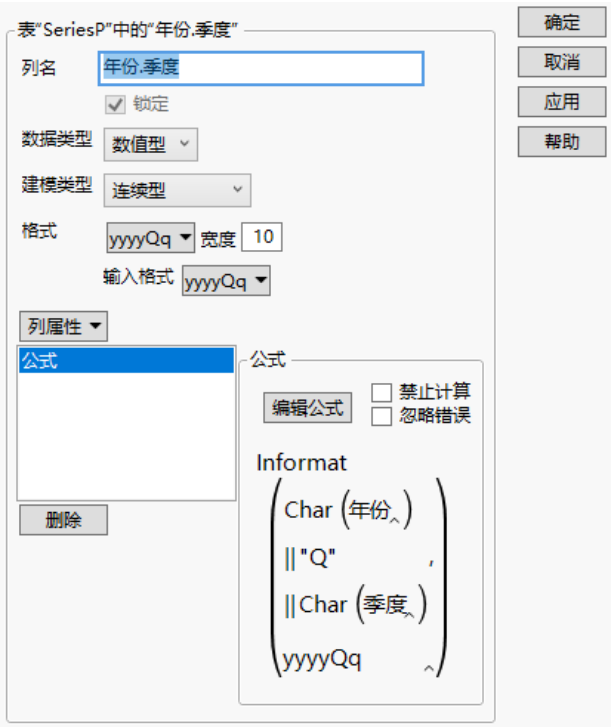
创建合适的“时间 ID”列

1. 选择**帮助** > **样本数据库**，然后打开 Time Series/SeriesP.jmp。

   SeriesP.jmp 数据表包含一个**年份**列和**季度**列来标识在观测响应期间的时间期间。但是，“时间序列”平台需要具有唯一、等间距的时间点的一列来标记 X 轴。若未指定 “时间 ID”，则使用行号标识时间期间。为避免该问题和使报表更易于解释，您从**年份**和**季度**构造一个“时间 ID”列。
2. 选择**列** > **新建列**。在 “列名”框中，键入**年份 . 季度**。
3. 选择 “格式”旁边的**最佳** > **日期** > yyyyQq。
4. 选择**列属性** > **公式**。
5. 点击 “日期时间”旁边的灰色小三角并点击 Informat。
6. 点击 “字符”旁边的灰色小三角。
7. 选择**年份**，点击 Char，然后再点击 Concat。
8. 在框中键入 “Q”（包含引号）并按 Enter 键。
9. 点击 Concat。
10. 选择**季度**，然后点击 Char。
11. 点击 <formatName> 框，键入 yyyyQq，然后按 Enter 键。
12. 点击**确定**。

    完成的 “新建列”窗口应如图 18.16 中所示。

图 18.16 新建列



13. 点击**确定**。

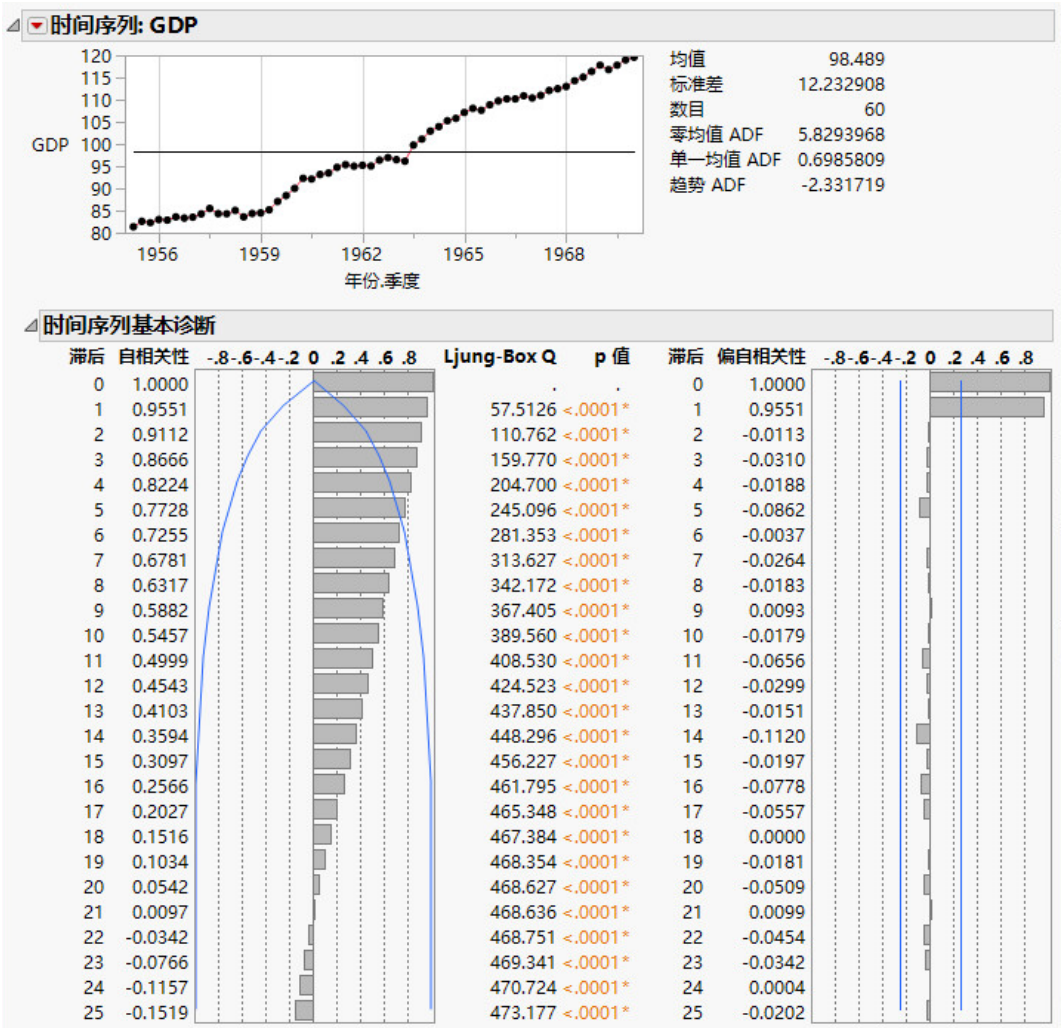
注意：该时间列同样适用于 X11 分析。

时间序列分析

数据表包含合适的“时间 ID”列后，继续执行分析。

1. 选择**分析 > 专业建模 > 时间序列**。
2. 选择 GDP 并点击 **Y，时间序列**。
3. 选择**年份 . 季度**并点击 **X，时间 ID**。
4. 点击**确定**。

图 18.17 SeriesP.jmp 的“时间序列”报表

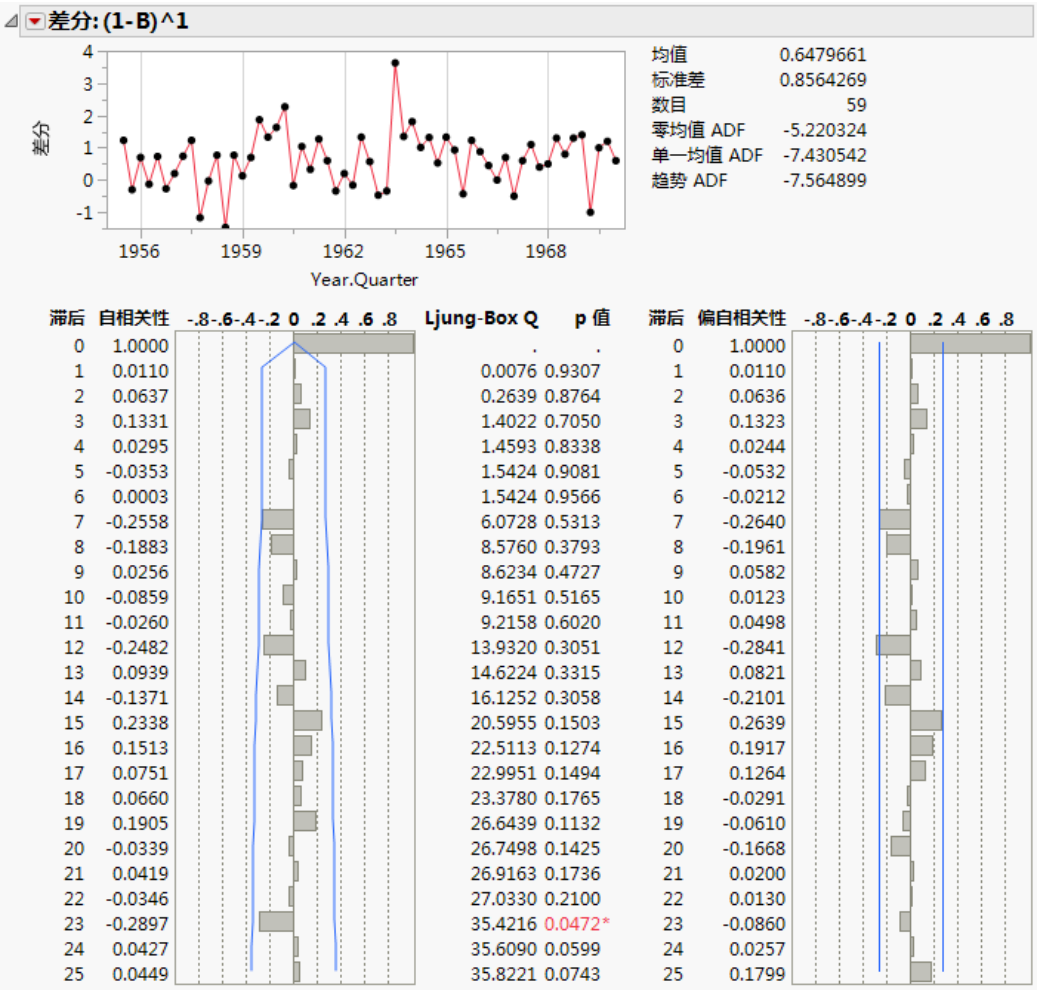


滞后	自相关性	Ljung-Box Q	p 值	滞后	偏自相关性
0	1.0000	.	.	0	1.0000
1	0.9551	57.5126	<.0001*	1	0.9551
2	0.9112	110.762	<.0001*	2	-0.0113
3	0.8666	159.770	<.0001*	3	-0.0310
4	0.8224	204.700	<.0001*	4	-0.0188
5	0.7728	245.096	<.0001*	5	-0.0862
6	0.7255	281.353	<.0001*	6	-0.0037
7	0.6781	313.627	<.0001*	7	-0.0264
8	0.6317	342.172	<.0001*	8	-0.0183
9	0.5882	367.405	<.0001*	9	0.0093
10	0.5457	389.560	<.0001*	10	-0.0179
11	0.4999	408.530	<.0001*	11	-0.0656
12	0.4543	424.523	<.0001*	12	-0.0299
13	0.4103	437.850	<.0001*	13	-0.0151
14	0.3594	448.296	<.0001*	14	-0.1120
15	0.3097	456.227	<.0001*	15	-0.0197
16	0.2566	461.795	<.0001*	16	-0.0778
17	0.2027	465.348	<.0001*	17	-0.0557
18	0.1516	467.384	<.0001*	18	0.0000
19	0.1034	468.354	<.0001*	19	-0.0181
20	0.0542	468.627	<.0001*	20	-0.0509
21	0.0097	468.636	<.0001*	21	0.0099
22	-0.0342	468.751	<.0001*	22	-0.0454
23	-0.0766	469.341	<.0001*	23	-0.0342
24	-0.1157	470.724	<.0001*	24	0.0004
25	-0.1519	473.177	<.0001*	25	-0.0202

该序列显示一个随时间增长的趋势，它简直是线性的。此外，自相关性图显示在靠得很近的两点之间存在强相关。滞后期数为 1、 2 和 3 的点的自相关性值分别为 0.9551、 0.9112、 0.8666。

5. 点击“时间序列 : GDP”红色小三角菜单，然后选择**差分**。
6. 为“非季节性差分阶数”选择 1，然后点击**估计**。

图 18.18 SeriesP.jmp 的 “差分”报表



“差分”报表帮助确定要针对原始时间序列拟合的合适模型。差分图显示差分序列不再具有在原始数据中观测到的趋势。这指示 “滞后 -1”差分是合适的选择。此外，即使去除了该趋势，序列也未显示季节性特征。因此，拟合原始序列的模型应可以处理线性趋势，但是没有必要处理季节性。线性指数平滑和 ARIMA 模型都是合适的。

7. 点击 “时间序列 : GDP”红色小三角菜单，然后选择**平滑模型 > 线性指数平滑**。
8. 点击**估计**。
9. 点击“时间序列 : GDP”红色小三角菜单，然后选择 **ARIMA 模型组**。这样您可以针对 (*p,d,q*) (*P,D,Q*) 的值范围拟合多个 ARIMA 模型。
10. 在 “ARIMA”框中，设置以下范围：

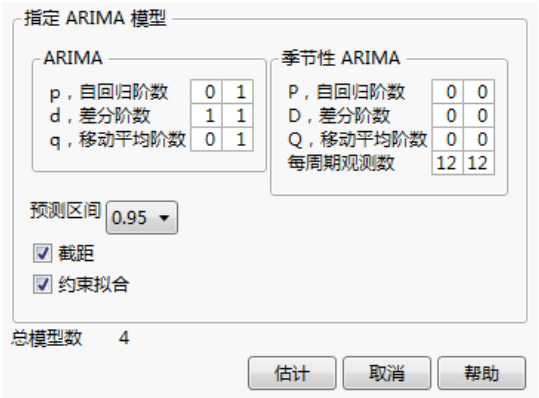
– 通过设置范围 0 到 1，将差分阶数 *d* 固定为 1，因为差分报表已显示“滞后-1”差分是合适的。
– 将自回归阶数 *p* 设置为 0 到 1 的范围，因为原始序列显示了自相关性的证据。
– 将移动平均阶数 *q* 设置为 0 到 1 的范围。

注意：在大多数情况下，保持 *p* 和 *q* 小就足够了。

– 让 *P*、*D* 和 *Q* 设置为 0，因为序列没有显示季节性的证据。

这些设置导致总共拟合 4 个模型。

图 18.19 指定 ARIMA 模型组



11. 点击**估计**。

图 18.20 “模型比较”表

模型比较

报表	图形	模型	自由度	方差	AIC	SBC	R 方	-2LogLH	权重	.2 .4 .6 .8	均值绝对百分误差	均值绝对误差
☑	☐	I(1)	58	0.746113	151.14636	153.22390	0.995	149.14636	0.477865		0.675816	0.649664
☑	☐	ARI(1, 1)	57	0.7591096	153.13924	157.29432	0.995	149.13924	0.176424		0.676216	0.649960
☑	☐	IMA(1, 1)	57	0.7591199	153.14002	157.29510	0.995	149.14002	0.176355		0.676179	0.649933
☑	☐	ARIMA(1, 1, 1)	56	0.748633	153.91882	160.15143	0.995	147.91882	0.119474		0.668189	0.641981
☑	☐	线性 (Holt) 指数平滑	56	0.7878954	155.66571	159.78660	0.994	151.66571	0.049882		0.716452	0.686737

根据 AIC 准则，“模型比较”表按以下方式排序：最佳拟合模型位于列表顶部。在本例中，ARIMA(0,1,0) 模型（在报表中表示为 I(1)）拟合原始时间序列的效果最好。还应注意，尽管 I(1) 模型是“最佳的”，所有模型的拟合统计量值都非常相近。它们都可以被视为合适的模型。

图 18.21 ARIMA(0,1,0) 的模型报表

I(1) 的模型报表显示预测图。蓝线指示预测区间。预测 GDP 将以线性速度继续增长。

使用保留集的示例

本例使用保留集评估对每月铅产量数据所做的预测。由于使用了保留集，因此可以对“状态空间平滑”模型和 ARIMA 模型进行拟合和比较，以找到数据的最佳拟合模型。

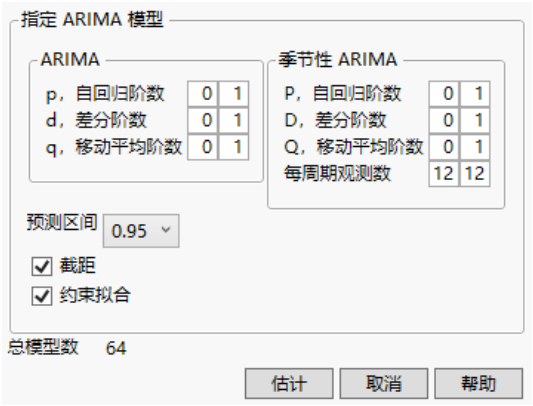
1. 选择**帮助 > 样本数据库**，然后打开 Time Series/Lead Production.jmp。
2. 选择**分析 > 专业建模 > 时间序列**。
3. 选择**铅产量**并点击 **Y，时间序列**。
4. 选择**日期**并点击 **X，时间 ID**。
5. 选择**针对保留进行预测**。

   选择该选项意味着预测基于保留集，而不是基于将来的观测。
6. 在 “预测周期数”旁边输入 12。

   这会将数据中的最后 12 个观测分配给保留集。由于这是月度数据，这相当于去年的数据。

7. 点击**确定**。
8. 点击 “时间序列：铅产量”红色小三角并选择**状态空间平滑**。
9. 在 “周期”旁边键入 12。
10. 选择 “约束参数”。
11. 点击**确定**。
12. 点击 “时间序列：铅产量”红色小三角并选择 **ARIMA 模型组**。
13. 将所有参数的范围设置为 0 到 1 之间。“指定 ARIMA 模型”窗口与图 18.22 中所示的窗口一样。

图 18.22 指定 ARIMA 模型组



14. 点击**估计**。

图 18.23 模型比较

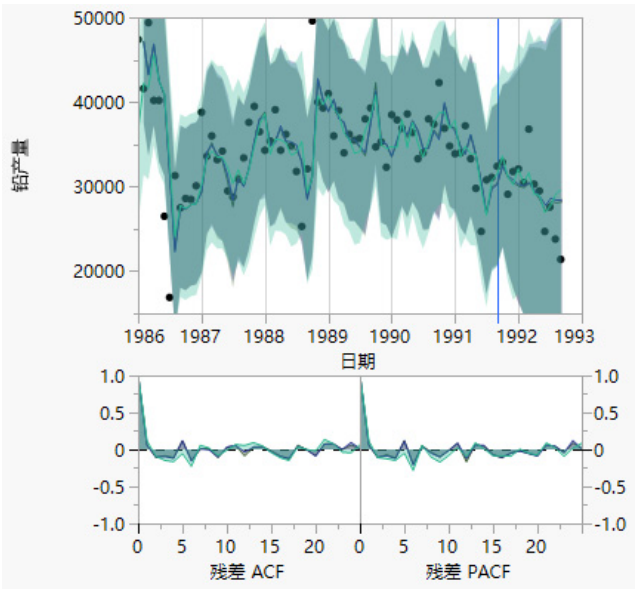
模型比较

报表	图形	模型类	模型	RMSE	MSE	保留计算 均值绝对百分误差	保留计算 均值绝对误差
☐	☐	ARIMA	季节性 ARIMA(0, 1, 1)(1, 0, 0)12	3219.4131	10364621	8.900164	2416.3319
☐	☐	ARIMA	季节性 ARIMA(0, 1, 1)(0, 0, 1)12	3316.8746	11001657	9.148390	2456.0873
☐	☐	状态空间平滑	AAA12 约束	3350.9303	11228734	8.623519	2245.5657
☐	☐	ARIMA	季节性 ARIMA(1, 1, 0)(1, 0, 0)12	3397.0855	11540190	9.082427	2392.5823
☐	☐	ARIMA	季节性 ARIMA(0, 1, 1)(1, 1, 0)12	3448.6372	11893099	9.829459	2649.8142
☐	☐	ARIMA	季节性 ARIMA(1, 1, 0)(1, 1, 0)12	3484.3764	12140879	9.708056	2604.6290
☐	☐	ARIMA	季节性 ARIMA(1, 1, 0)(0, 0, 1)12	3491.2648	12188930	9.302681	2435.7441
☐	☐	状态空间平滑	MAM12 约束	3512.1513	12335206	10.552196	2813.4655
☐	☐	ARIMA	季节性 ARIMA(0, 1, 0)(1, 0, 0)12	3525.0858	12426230	9.374006	2440.6892
☐	☐	ARIMA	季节性 ARIMA(0, 1, 0)(1, 1, 0)12	3551.3973	12612423	9.367947	2480.3210
☐	☐	ARIMA	季节性 ARIMA(0, 1, 1)(1, 0, 1)12	3579.0242	12809414	10.520274	2739.9652
☐	☐	ARIMA	季节性 ARIMA(0, 1, 0)(0, 1, 0)12	3597.3098	12940638	11.622868	3219.0476
☐	☐	ARIMA	季节性 ARIMA(0, 1, 0)(0, 0, 1)12	3631.8176	13190099	9.792052	2544.4269
☐	☐	ARIMA	季节性 ARIMA(1, 1, 0)(0, 1, 0)12	3661.8395	13409068	12.030672	3366.0935
☐	☐	ARIMA	季节性 ARIMA(0, 1, 1)(0, 1, 0)12	3673.6369	13495608	12.063523	3379.4644
☐	☐	ARIMA	季节性 ARIMA(1, 1, 0)(1, 0, 1)12	3700.6621	13694900	11.077791	2890.6426
☐	☐	ARIMA	季节性 ARIMA(0, 1, 0)(1, 0, 1)12	3723.7664	13866437	11.222827	2935.8075

由于“状态空间平滑”模型和 ARIMA 模型是针对保留集进行评估的，因此这两个模型的拟合统计数据具有可比性。

15. 选中 “模型比较”报表中前三个模型旁边的 “图形”复选框。

图 18.24 模型比较图形



评估统计量和预测图表明：前几个模型的预测性能没有显著差异。根据该报表， ARIMA 模型或 “状态空间平滑”模型都适用于该数据。

“时间序列”平台的统计详细信息

- “谱密度的统计详细信息”
- “X-11 分解的统计详细信息”
- “平滑模型的统计详细信息”
- “ARIMA 模型的统计详细信息”
- “转换函数的统计详细信息”

谱密度的统计详细信息

傅里叶序列的系数的最小二乘估计值按以下方式计算：

$$a_t = \frac{2}{N}\sum_{i=1}^{N} y_t \cos(2\pi f_i t)$$

$$b_t = \frac{2}{N}\sum_{i=1}^{N} y_t \sin(2\pi f_i t)$$

之后，与 $f_i = i/N$ 进行组合形成周期图 $I(f_i) = (N/2)(a_i^2 + b_i^2)$，表示频率 f_i 处的强度。

对周期图进行平滑处理，同时除以 (4π) 以统一尺度，得到谱密度。

X-11 分解的统计详细信息

该方法使用乘法或加法分解调整原始时间序列。模型使用迭代过程拟合来估计三个 X-11 因素：趋势周期、季节性和不规则因素。趋势周期成分包含长期趋势和长期周期性效应。不规则成分包含趋势和季节性成分无法解释的变异效应。有关 X-11 方法的发展史概述，请参见 SAS/ETS 15.2 User’s Guide（搜索“Historical Development of X-11”）。

乘法调整拟合以下模型：

$$O_t = C_t \cdot S_t \cdot I_t$$

其中

O_t 是原始时间序列

C_t 是趋势周期成分

S_t 是季节性成分

I_t 是不规则成分

经乘法调整后的趋势为 O_t/S_t。

加法调整拟合以下模型：

$$O_t = C_t + S_t + I_t$$

经加法调整后的趋势为 $O_t - S_t$。

平滑模型的统计详细信息

按以下方式定义平滑模型：

$$y_t = \mu_t + \beta_t t + s(t) + a_t$$

其中

μ_t 是时变均值项

β_t 是时变斜率项

$s(t)$ 是 s 时变季节性项中的一项

a_t 是随机扰动项

没有趋势的模型具有 $\beta_t = 0$，非季节性模型具有 $s(t) = 0$。按以下方式定义这些时变项的估计量：

L_t 是估计 μ_t 的平滑水平

T_t 是估计 β_t 的平滑趋势

S_{t-j}（$j = 0, 1, \ldots, s - 1$）是 $s(t)$ 的估计值

每个平滑模型定义一组递归平滑方程，用于描述这些估计量的变化过程。平滑方程是由被称为**平滑权重**的模型参数定义的：

α 是水平平滑权重

γ 是趋势平滑权重

φ 是趋势阻尼权重

δ 是季节性平滑权重

当这些参数以不同方式进入每个模型（或根本不进入模型）时，它们具有共同的性质，即较大的权重对于最近数据有更大的影响，而较小的权重对最近数据有更小的影响。

简单指数平滑

简单指数平滑的模型为 $y_t = \mu_t + \alpha_t$。

根据单个平滑权重 α 来定义平滑方程 $L_t = \alpha y_t + (1-\alpha)L_{t\text{-}1}$。该模型等价于满足以下条件的 ARIMA(0, 1, 1) 模型：

$(1-B)y_t = (1-\theta B)\alpha_t$ 其中，$\theta = 1-\alpha$

模型的移动平均形式定义如下：

$$y_t = a_t + \sum_{j-1}^{\infty} \alpha a_{t-j}$$

二次 (Brown) 指数平滑

双指数平滑模型为 $y_t = \mu_t + \beta_1 t + a_t$。

具有单个平滑权重 α 的平滑方程定义如下：

$$L_t = \alpha y_t + (1-\alpha)L_{t-1}$$

$$T_t = \alpha(L_t - L_{t-1}) + (1-\alpha)T_{t-1}$$

该模型等价于满足以下条件的 ARIMA(0, 1, 1)(0, 1, 1)$_1$ 模型：

$(1-B)^2 y_t = (1-\theta B)^2 a_t$ ，其中 $\theta_{1,1} = \theta_{2,1}$ 且 $\theta = 1-\alpha$

模型的移动平均形式定义如下：

$$y_t = a_t + \sum_{j=1}^{\infty} (2\alpha + (j-1)\alpha^2)a_{t-j}$$

线性 (Holt) 指数平滑

线性指数平滑模型为 $y_t = \mu_t + \beta_t t + a_t$。

具有平滑权重 α 和 γ 的平滑方程定义如下：

$$L_t = \alpha y_t + (1-\alpha)(L_{t-1} + T_{t-1})$$

$$T_t = \gamma(L_t - L_{t-1}) + (1-\gamma)T_{t-1}$$

该模型等价于满足以下条件的 ARIMA(0, 2, 2) 模型：

$(1-B)^2 y_t = (1-\theta B - \theta_2 B^2)a_t$ ，其中 $\theta = 2-\alpha-\alpha\gamma$ 且 $\theta_2 = \alpha - 1$

模型的移动平均形式定义如下：

$$y_t = a_t + \sum_{j=1}^{\infty} (\alpha + j\alpha\gamma)a_{t-j}$$

阻尼趋势线性指数平滑

阻尼趋势线性指数平滑模型为 $y_t = \mu_t + \beta_t t + a_t$。

具有平滑权重 α、γ 和 φ 的平滑方程定义如下：

$$L_t = \alpha y_t + (1-\alpha)(L_{t-1} + \varphi T_{t-1})$$

$$T_t = \gamma(L_t - L_{t-1}) + (1-\gamma)\varphi T_{t-1}$$

该模型等价于满足以下条件的 ARIMA(1, 1, 2) 模型：

$$(1-\varphi B)(1-B)y_t = (1-\theta_1 B - \theta_2 B^2)a_t$$

其中

$$\theta_1 = 1 + \varphi - \alpha - \alpha\gamma\varphi$$

$$\theta_2 = (\alpha - 1)\varphi$$

模型的移动平均形式定义如下：

$$y_t = \alpha_t + \sum_{j=1}^{\infty} \left(\frac{\alpha + \alpha\gamma\varphi(\varphi^j - 1)}{\varphi - 1} \right) \alpha_{t-j}$$

季节性指数平滑

季节性指数平滑模型为 $y_t = \mu_t + s(t) + a_t$。

具有平滑权重 α 和 δ 的平滑方程定义如下：

$$L_t = \alpha(y_t - S_{t-s}) + (1-\alpha)L_{t-1}$$

$$S_t = \delta(y_t - L_{t-s}) + (1-\delta)S_{t-s}$$

该模型等价于一个季节性 ARIMA$(0, 1, s+1)(0, 1, 0)_s$ 模型：

$$(1-B)(1-B^s)y_t = (1-\theta_1 B - \theta_2 B^2 - \theta_3 B^{s+1})a_t$$

其中

$$\theta_1 = 1-\alpha$$

$$\theta_2 = (1-\delta)(1-\alpha)$$

$$\theta_3 = (1-\alpha)(\delta-1)$$

模型的移动平均形式定义如下：

$$y_t = a_t + \sum_{j=1}^{\infty} \psi_j a_{t-j} \text{ 其中，} \psi = \begin{cases} \alpha \text{ ，对于 } j \bmod s \neq 0 \\ \alpha + \delta(1-\alpha) \text{ ，对于 } j \bmod s = 0 \end{cases}$$

Winters 方法（加法）

Winters 方法的加法版本的模型为 $y_t = \mu_t + \beta_t t + s(t) + a_t$。

具有权重 α、γ 和 δ 的平滑方程定义如下：

$$L_t = \alpha(y_t - S_{t-s}) + (1-\alpha)(L_{t-1} + T_{t-1})$$

$$T_t = \gamma(L_t - L_{t-1}) + (1-\gamma)T_{t-1}$$

$$S_t = \delta(y_t - L_t) + (1-\delta)S_{t-s}$$

该模型等价于按以下方式定义的一个季节性 ARIMA(0, 1, *s*+1)(0, 1, 0)s 模型：

$$(1-B)(1-B^2)y_t = \left(1 - \sum_{i=1}^{s+1} \theta_i B^i\right) a_t$$

模型的移动平均形式定义如下：

$$y_t = a_t + \sum_{j=1}^{\infty} \Psi_j a_{t-j}$$

其中

$$\psi = \begin{cases} \alpha + j\alpha\gamma, & j \bmod s \neq 0 \\ \alpha + j\alpha\gamma + \delta(1-\alpha), & j \bmod s = 0 \end{cases}$$

ARIMA 模型的统计详细信息

ARIMA 模型

对于响应序列 $\{y_i\}$， ARIMA 模型的一般形式定义如下：

$$\phi(B)(w_t - \mu) = \theta(B)a_t$$

其中

t 是时间索引

B 是定义为 $By_t = y_{t-1}$ 的后移算子

$w_t = (1 - B)^d y_t$ 是差分后的响应序列

μ 是截距或均值项

$\phi(B)$ 和 $\theta(B)$ 分别为自回归算子和移动平均算子，定义如下：

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p$$

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q$$

其中

a_t 是随机扰动项序列

假定 a_t 独立，服从均值为零且方差为常数的正态分布。

该模型可以改写为：

$$\phi(B)w_t = \delta + \theta(B)a_t$$

常数估计值 δ 由以下关系等式给出：

$$\delta = \phi(B)\mu = \mu - \phi_1\mu - \phi_2\mu - \ldots - \phi_p\mu$$

季节性 ARIMA 模型

在**季节性** ARIMA 建模过程中，差分、自回归和移动平均算子是季节性和非季节性多项式的乘积：

$$w_t = (1 - B)^d (1 - B^s)^D y_t$$

$$\varphi(B) = (1 - \varphi_{1,1}B - \varphi_{1,2}B^2 - \ldots - \varphi_{1,p}B^p)(1 - \varphi_{2,s}B^s - \varphi_{2,2s}B^{2s} - \ldots - \varphi_{2,Ps}B^{Ps})$$

$$\theta(B) = (1 - \theta_{1,1}B - \theta_{1,2}B^2 - \ldots - \theta_{1,q}B^q)(1 - \theta_{2,s}B^s - \theta_{2,2s}B^{2s} - \ldots - \theta_{2,Qs}B^{Qs})$$

其中 s 是每周期观测数。系数的第一个下标是因子编号（1 表示非季节性，2 表示季节性），第二个下标为滞后项。

转换函数的统计详细信息

具有 m 个输入的典型转换函数模型可以表示为：

$$Y_t - \mu = \frac{\omega_1(B)}{\delta_1(B)} X_{1,\, t-d1} + \ldots + \frac{\omega_m(B)}{\delta_m(B)} X_{m,\, m-dm} + \frac{\theta(B)}{\varphi(B)} e_t$$

其中

Y_t 表示输出序列

X_1 到 X_m 表示 m 个输入序列

e_t 表示噪声序列

$X_{1,t-d1}$ 表示滞后 $d1$ 期的 X_1 序列 （t 为时间索引）

μ 表示模型的均值水平

$\varphi(B)$ 和 $\theta(B)$ 表示 ARIMA 模型的自回归和移动平均多项式

$\omega_k(B)$ 和 $\delta_k(B)$ 表示单个转换函数的分子和分母因子（或多项式），k 表示从 1 到 m 的各个输入序列的索引。

以上模型中的每个多项式可以包含两个部分：非季节性、季节性或如季节性 ARIMA 中这两者之积。指定模型时，对于不想包含在模型中的任何部分保留默认值 0。

第 19 章

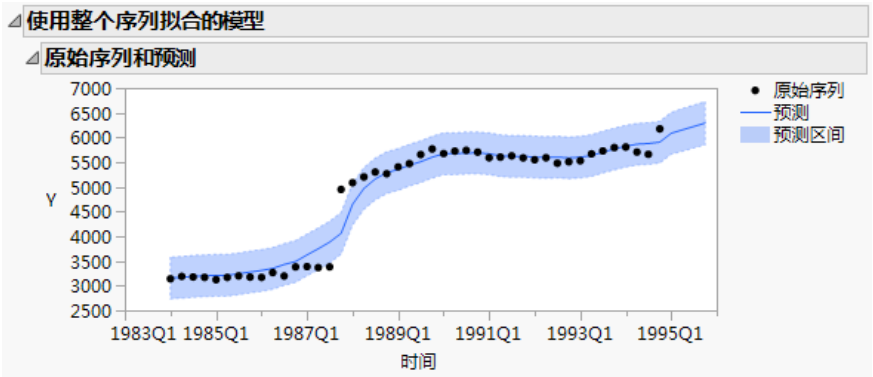
时间序列预测

预测多个时间序列

“时间序列预测”平台支持您预测多个时间序列，其中序列的数量可能非常大。该平台提供了多个时间序列的简要汇总，并提供高效拟合指数平滑系列中多达 30 个不同预测模型的选项。自动选定最佳拟合模型。

“时间序列预测”平台设计用于从一组指数平滑模型中自动选择最佳拟合模型。这与“时间序列”平台不同，后者需要手动选择最佳拟合模型并使用 ARIMA 模型。

图 19.1 原始序列和预测



目录

时间序列预测示例 331
启动“时间序列预测”平台 333
数据格式 333
“时间序列预测”报表 334
时间模式分析 335
时间的分组汇总 335
Y 的分组汇总 335
建模规格 336
“时间序列预测”平台选项 338
模型报表 339

时间序列预测示例

1. 选择**帮助** > **样本数据库**，然后打开 Time Series/M3C Quarterly.jmp。
2. 选择**分析** > **专业建模** > **时间序列预测**。
3. 选择 Y 并点击 Y。
4. 选择**序列**并点击**分组**。
5. 选择**时间**并点击**时间**。
6. 点击**确定**。

   初始 “时间序列预测”报表随即显示。其中包含 “时间”和 “Y”变量的汇总。还有一个 “建模规格”报表，它提供用于拟合预测模型的选项。

图 19.2 “时间序列预测”的初始报表

时间序列预测

时间模式分析

日期分辨率 每季度数据。
建议的季节性 4

时间的分组汇总

序列	行数	缺失值	唯一时间戳	开始	结束	缺失记录
N 646	44	否	是	1984Q1	1994Q4	否
N 647	44	否	是	1984Q1	1994Q4	否
N 648	46	否	是	1983Q3	1994Q4	否
N 649	44	否	是	1984Q1	1994Q4	否
N 650	44	否	是	1984Q1	1994Q4	否
N 651	44	否	是	1984Q1	1994Q4	否
N 652	43	否	是	1984Q1	1994Q3	否
N 653	43	否	是	1984Q1	1994Q3	否
N 654	45	否	是	1983Q4	1994Q4	否
N 655	44	否	是	1984Q1	1994Q4	否

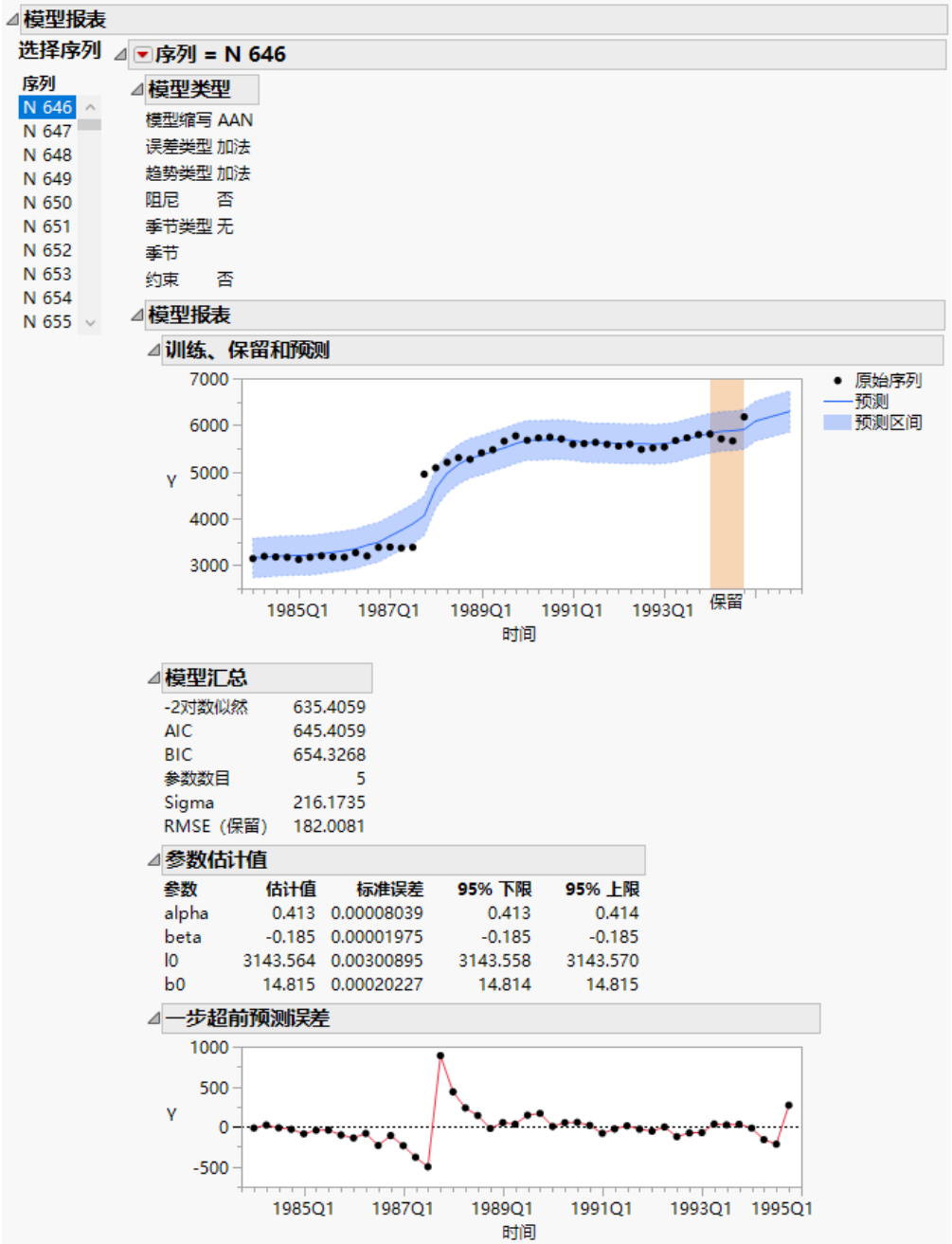
“Y”的分组汇总

序列	最小值	最大值	长度	N 个观测	缺失值个数	全部为正	注释
N 646	3124.38	6176.6	44	44	0	是	
N 647	1522	4320.8	44	44	0	是	
N 648	1549.72	5970	46	46	0	是	
N 649	2136.7	5023.2	44	44	0	是	
N 650	875.51	6908	44	44	0	是	
N 651	7512.4	8526.2	44	44	0	是	
N 652	4861.63	9913.57	43	43	0	是	
N 653	2797	7202	43	43	0	是	
N 654	2372.74	6735.1	45	45	0	是	
N 655	2449.06	8904.28	44	44	0	是	

7. 在 “模型规格”报表中，点击 “完整规格”选项卡。
8. 点击 “模型选型策略”下的菜单并选择 “预测性能”。
9. 点击**运行**。

这将对数据拟合一组推荐模型。为分组变量的每个水平选择最佳模型并在“模型报表”中报告。请参见第 332 页的 “模型报表”。

图 19.3 模型报表



显示了一个序列 “序列 N646”的模型报表。这与在 “选择序列”列表中选择的序列名称相对应。您可以使用向下箭头键扫描每个序列的模型报表。或者，可以点击列表中的其他名称以显示选定序列的模型报表。

启动“时间序列预测”平台

通过选择**分析** > **专业建模** > **时间序列预测**来启动 “时间序列预测”平台。

图 19.4 “时间序列预测”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“时间序列预测”平台启动窗口包含以下选项：

Y 将一个或多个列分配为时间序列变量。可以将多个时间序列组织为表中各列，也可以使用一个或多个分组变量将它们堆叠在单个列中。

分组 将一个或多个列指定为分组变量。当时间序列采用堆叠格式时，可使用分组变量标识各个时间序列。

时间 为时间变量分配一列。若各个时间序列组织为各列，则时间列中的时间戳不得包含任何重复项。

依据 分配用于创建报表的一个列，该报表包括变量的每个水平的单独分析。若分配了多个“依据”变量，则为 “依据”变量水平的每个可能组合生成单独分析。

数据格式

可以将多个时间序列组织为表中各列，也可以使用一个或多个分组列将它们堆叠在单个列中以标识各个序列。还可以组合多个 Y 列和多个分组列。

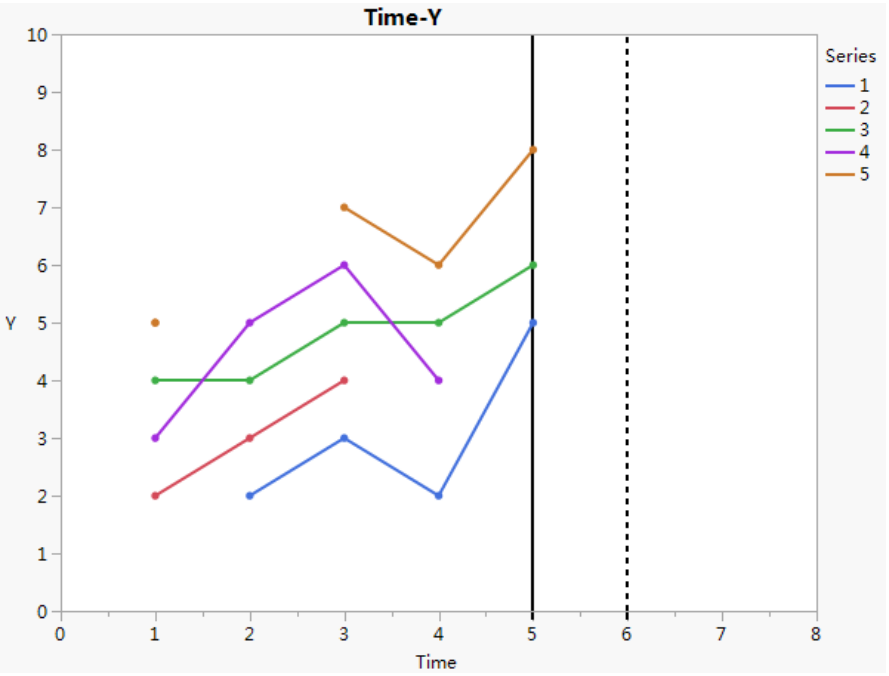
该平台做出以下关于数据的假设：

- 所有时间序列都以相同的时间分辨率记录。
- 所有时间序列都收集到同一最终观测时间点。这个最终观测时间点被定义为发生在任何单个时间序列中的最近时间点。到最终观测时间点没有数据的序列被认定包含缺失值。例

如，在图 19.5 中，最终观测时间点为 5，用黑色实线表示。可以认定：序列 2 在时间点 4 和 5 处缺失观测，序列 4 在时间点 5 处缺失观测。

- 所有预测都在相同的时间点开始。这个起始时间点是跟随最终观测时间点的下一个时间点。例如，在图 19.5 中，最终观测时间点为 5，用黑色实线表示。因此，所有预测都从时间点 6 开始，用黑色虚线表示。

图 19.5 假设演示



“时间序列预测”报表

“时间序列预测”报表包含基于启动窗口中指定的变量的汇总表。始终提供按组列出的时间序列的汇总表。若指定了“时间”列，则报表还包含时间模式的汇总和按组列出的时间汇总表。该报表还包含一个“建模规格”报表，该报表支持您拟合每个时间序列的模型。

- “时间模式分析”
- “时间的分组汇总”
- “Y 的分组汇总”
- “建模规格”

时间模式分析

若指定了时间列，则“时间序列预测”平台将分析该列的条目以确定模式。该平台旨在识别各种时间模式，包括每日、每个工作日、每月、每两周、每季度等。模式报告为“日期分辨率”。同时基于时间模式提供“建议的季节性”。

时间的分组汇总

若指定了时间列，则报表包含各个时间序列的汇总。汇总中包含关于每个时间序列的以下信息：

< 分组列名称 > （仅当指定了分组变量时才可用。）以堆叠格式提供数据时分组变量的水平用于定义单个序列。

行数 序列中的观测数。

缺失值 指示是否有任何时间值为缺失值。

唯一时间戳 指示时间戳是否唯一。

开始 开始时间戳。

结束 结束时间戳。

缺失记录 根据开始和结束时间戳以及数据分辨率 / 模式，指示数据表中是否缺失观测。

Y 的分组汇总

该报表包含每个 Y 变量的汇总表。汇总中包含关于每个时间序列的以下信息：

变量 （仅当有多个 Y 列且有至少一个分组列时才可用。）时间序列变量的列名。

时间序列 （仅当有多个序列时才可用。）各个时间序列的名称。当数据以堆叠格式提供时，这是分组变量的水平。当数据以宽格式提供并且有多个列时，这是各个序列的列名。

注意：当数据以堆叠格式提供时，表中的列标题是 “分组”列的名称。

最小值 各个序列的最小 Y 值。

最大值 各个序列的最大 Y 值。

长度 序列的持续时间。这是根据开始和结束时间戳以及 “时间模式分析”的结果确定的。

N 个观测 观测数。若无缺失记录，该数字与 “长度”相同。

缺失值个数 缺失 Y 值的数目。

全部为正 指示所有 Y 值是否均为正值。

注释 指示序列是否有问题，例如非唯一时间戳。

建模规格

“建模规格”报表支持您对各个时间序列拟合模型。考虑多种模型。Hyndman et al.(2008) 基于模型的误差、趋势成分和季节性成分定义状态空间平滑模型：

- 误差可以是加法 (A) 或乘法 (M)。
- 趋势成分可以为“无 (N)”、“加法 (A)”、“带阻尼的加法 (A_d)”、“乘法 (M)”或“带阻尼的乘法 (M_d)”。
- 季节性成分可为“无 (N)”、“加法 (A)”或“乘法 (M)”。

特定模型可用其 ETS（误差、趋势、季节性）来表示。以下是在“时间序列预测”中使用的模型。

“建模规格”报表中有两个选项卡。使用“推荐规格”选项卡可拟合一组平台挑选的状态空间平滑模型。使用“完整规格”选项卡可选择您想要拟合的特定状态空间平滑模型。对于每个单独的时间序列，给定集中的最佳拟合模型随后用于预测。

选择模型

仅可用于“完整规格”选项卡，如第 337 页的““完整规格”选项卡”中所示。“选择模型”报表支持您指定状态空间平滑模型以拟合到各个时间序列。使用复选框为模型选择误差、趋势和季节性。点击**选择推荐项**选中与平台推荐的模型对应的复选框。点击**全选**可选中所有复选框，点击**取消全选**可取消选中所有复选框。点击**约束参数**以这样一种方式约束参数，即：观测离当前越远，对模型当前状态的影响就越小。在“状态空间平滑”模型中，在给定所有以前的观测的情况下，时间 *t* 的预测与截至时间 *t* 的所有观测的加权和相同。这些权重是参数的函数。因此，约束参数可以确保过去观测的权重趋近零，并且观测距离现在越远，权重接近零的速度就越快。

图 19.6 “完整规格”选项卡

预测设置

支持您指定以下可选设置：

超前步数 指定要预测的超前步数。步数必须非负。

周期 指定模型拟合过程中要考虑的季节性值。默认情况下，周期设置为“时间模式分析”报表中建议的季节性值。若要考虑具有不同季节性的模型，请在此处指定其他周期，并用逗号分隔它们。该周期必须大于零。

模型选型策略

支持您指定如何选择最佳模型。

信息准则 基于指定的信息准则确定最佳模型。可用信息准则包括 AIC 和 BIC。

> **提示：**若时间序列较长，则使用 BIC。

预测性能 基于涉及保留集的性能选择算法确定最佳模型。首先，将时间序列划分为训练集和保留集。“保留数”的值指定保留集中的观测数。然后，该算法针对训练集中的数据拟合所有推荐的或用户指定的模型。使用单个拟合模型对保留集进行预测。将这些预测与实际保留观测进行比较，并使用指定的量度对模型进行评估。选择具有最佳量度的模型作为最终模型。最后，选择的最终模型对整个时间序列（训练集和保留集）进行拟合，并使用重新拟合后的模型对时间序列的最后一个观测之后进行预测。

量度 指定用于评估单个模型拟合所做预测的量度。可用的量度包括均方根误差 (RMSE)、均方误差 (MSE) 和绝对误差均值 (MAE)。

保留数 指定保留集中使用的观测数。

其他选项

支持您指定以下选项：

保留模型选择准则 保存拟合数据的一组模型用到的模型选择准则。模型拟合之后，该信息显示在“所有序列的所有模型的信息准则”报表中。

预测区间水平 预测的预测区间水平。这将更改预测图中着色区域的宽度。

对适用的模型进行插补 指定模型拟合过程中使用的插补方法。

注意：这些选项仅更改用于模型拟合的数据。原始数据不会改变。

无 不插补缺失值。

最后一个值 使用缺失观测序列之前可用的最后一个值来插补缺失值。

中间值 通过对缺失观测序列之前可用的最后一个值和缺失观测序列之后可用的第一个值取平均值来插补缺失值。

线性内插 通过在缺失观测序列之前可用的最后一个值和缺失观测序列之后可用的第一个值之间创建线性内插来插补缺失值。

点击**运行**时，将拟合指定的一组模型。将出现一个进度条，报告活动线程数、已完成任务数和任务总数以及已完成任务的百分比。请注意，任务定义为一个序列和响应变量组合的所有指定模型的拟合。拟合过程完成后，将显示“模型报表”报表。请参见第 339 页的“模型报表”。若对“模型规格”报表中的选项进行任何更改并再次点击**运行**，则“模型报表”报表将替换为新报表。

“时间序列预测”平台选项

“时间序列预测”红色小三角菜单包含以下选项：

报表选项 （仅在拟合模型后有效。）包含以下报表选项：

设置预测区间水平 支持您更改预测区间水平。自动更新各个模型报表。

设置超前步数 支持您更改要预测的超前步数。自动更新各个模型报表。

设置维持数 支持您更改维持集的持续时间。自动更新各个模型报表。

保存结果 （仅在拟合模型后有效。）将预测保存到名为“预测”的单独数据表。“预测”数据表具有与原始数据表相同的布局和格式。若维持集由“维持数”指定，则会生成一个附加数据表，用于保存维持数据的实际值和预测。可以使用该表计算不同类型的误差估计值。

“请求输出”窗口中显示了以下附加保存选项：

保存预测区间 将预测区间上下限保存为“预测”数据表中的列。

保存一步超前预测 堆叠一步超前预测和预测，并将其保存到单独的数据表。

保存原始序列 堆叠原始序列和预测，并将其保存到单独的数据表。

将预测结果保存至原始数据表 将结果保存至原始数据表中的新列。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

模型报表

“模型报表”报表包含一个“选择序列”列表和一个或多个单独的序列建模报表。显示的各个报表由“选择序列”列表确定。点击该列表中序列的名称以显示所选序列的报表。若使用 Ctrl 或 Shift 选择多个序列名称，则将为每个选定序列显示单个模型拟合报表。若从“选择序列”列表中取消选择一个或多个序列，则会删除相应序列的报表。

“选择序列”列表链接到数据表。当您在“选择序列”列表中选择某个序列后，也会在数据表中选定序列和响应变量组合对应的观测。“选择序列”列表还链接到“‘Y’的分组汇总”表。若指定了“时间”变量，则“选择序列”列表还会链接到“时间的分组汇总”表。当您在“选择序列”列表中选择某个序列后，也会在表中选定相应的条目。

每个单独的模型报表都包含以下信息：

模型类型 根据误差、趋势和季节性描述序列的最佳拟合模型。

模型报表 基于拟合序列的模型的一组报表。

序列和预测 （仅当将“信息准则”指定为模型选型策略时才可用。）原始序列和预测图。在原始序列的时间周期中，显示一步超前预测。在原始序列的时间周期外，显示预测。着色区域表示预测区间。

注意：所有预测都从紧跟跨所有单个序列的最后一个时间点的时间点开始。请参见第 333 页的“数据格式”。

训练、保留和预测 （仅当将“预测性能”指定为模型选型策略时才可用。）训练序列、保留序列和预测序列的图。着色垂直区域表示保留序列。由于这是对整个时间序列的模型重新拟合，因此在整个原始序列的时间段内显示一步超前预测。在最后一个观测之后的时间段内显示预测。着色区域表示预测区间。

模型汇总 提供 -2 对数似然、AIC、参数个数和 sigma 值。sigma 值是一步超前预测标准差的估计值。若将“预测性能”指定为模型选型策略，则还会显示保留集的均方根误差 (RMSE)。请参见《拟合线性模型》。

参数估计值 提供模型的参数估计值。这些参数对应于最佳拟合模型（在“模型类型”报表中说明）中的参数。

一步超前预测误差 随时间变化的一步超前预测误差图。对于每个时间点 t，误差都计算为 $e(t) = y(t) - \mu(t)$。

乘法误差模型的一步超前相对预测误差 （仅可用于乘法误差模型。）随时间变化的一步超前相对预测误差图。对于每个时间点 t，误差都计算为 $e(t) = [y(t) - \mu(t)]/\mu(t)$。

第 20 章

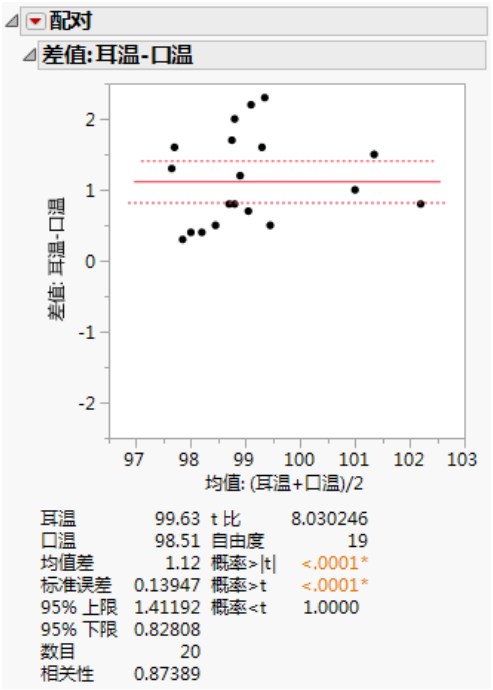
配对分析

比较关于同一对象的测量值

“配对”平台比较两个或更多相关变量之间的均值并评估差值。例如，您可以比较同一对象在治疗前和治疗后的血压测量值。称为**配对 t 检验**的统计方法会将相互关联的响应都考虑在内。

该平台生成“配对均值-配对差值”图，以及全部三个备择假设的配对 *t* 检验结果。该平台还为两个以上配对响应以及分组跨样本检验（重复测量分析的简易版本）提供了额外的功能。

图 20.1 配对分析示例



目录

“配对”平台概述 343
关于比较配对的示例 343
启动“配对”平台 344
多个 Y 列 345
“配对”报表 345
差值图和报表 346
组间 346
“配对”平台选项 347
比较组间配对的示例 348
“配对”平台的统计详细信息 349
配对图 349
响应的相关性 350

“配对”平台概述

“配对”平台使用配对 *t* 检验逐行比较两个响应列之间的差值。这两列经常代表同一对象在某种处理之前和之后的测量值。或者，测量值可以表示使用两个不同设备对同一对象提取的数据。

若您的配对数据排列在两个数据表列中，那么您随时可以使用 “配对”平台。不过，若您所有的测量值都包含在一列中，则执行以下任务之一：

- 使用**表**菜单中的**拆分**选项可将测量值列拆分为两列。然后您可以使用“配对”平台。
- 对于两个响应列，创建第三列，用于计算这两个响应之间的差值。然后使用“分布”平台检验差值列的均值是否为零。
- 对于存储在单列中的两个响应，您可以执行双因子方差分析。一个因子（ID 变量）标识这两个响应，另一个因子标识对象。将“以 X 拟合 Y”中的“单因子”平台与分区组变量（对象列）配合使用，或使用“拟合模型”平台执行双因子方差分析。针对 ID 因子的检验等同于配对 *t* 检验。

注意：若数据是配对的，则不要执行常规独立 *t* 检验。不要将数据堆叠到一列中，也不要不指定区组变量就对 ID 使用“以 X 拟合 Y 单因子方差分析”。这样做会忽略响应之间的相关性。若响应之间负相关，这会导致检验过度估计效应；若响应之间正相关，这会导致对效应估计不足。

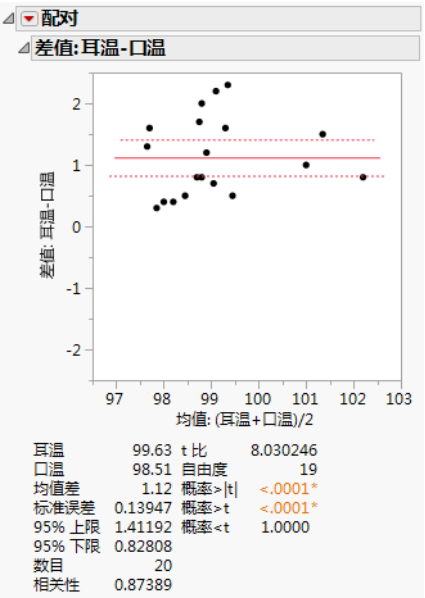
关于比较配对的示例

本例使用 Therm.jmp 样本数据表。数据中包含 20 个人的温度测量值。使用两种温度计测量体温：口温和耳温。您想要确定这两种温度计是否生成相等的温度读数。请注意，不同人员之间的温度差值不重要。配对分析检验的是温度计之间的差异。

1. 选择**帮助** > **样本数据库**，然后打开 Therm.jmp。
2. 选择**分析** > **专业建模** > **配对**。
3. 选择**口温**和**耳温**，然后点击 **Y，配对响应**。
4. 点击**确定**。

   随即显示报表窗口。

图 20.2 “配对” 报表窗口



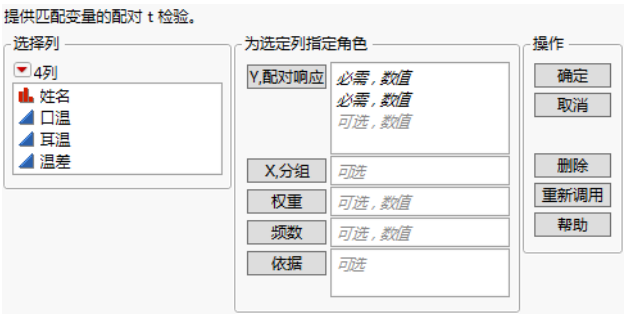
结果显示：耳温计测量的温度比口温计测量的温度平均高 1.12 度。较小的 p 值（概率 > |t|）指示该差异在统计上很显著，而且不是偶然出现的。

请注意，该配对分析并不表明哪种温度计正确（若其中一种是正确的），而是只表明温度计之间有差异。

启动 “配对” 平台

通过选择**分析** > **专业建模** > **配对**启动 “配对” 平台。

图 20.3 “配对” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，配对响应 提供两个响应列。有关分析两个以上响应的信息，请参见第 345 页的“多个 Y 列”。

X，分组 提供一个分组变量以比较不同组之间的差异。请参见第 346 页的 “组间”。

权重 标识一列，该列的数值为分析中的每一行都分配一个权重。

频数 标识一列，该列的数值为分析中的每一行都分配一个频数。

依据 为“依据”变量的每个水平执行单独的配对分析。

点击**确定**后即显示“配对”报表窗口。请参见第 345 页的“ “配对”报表”。

多个 Y 列

您可以有两个以上响应。若响应数为奇数，则分析所有可能的对。下表显示三个响应的示例。

Y2-Y1	Y3-Y1
	Y3-Y2

若响应数为偶数，“配对”平台会询问您是否想要分析所有可能对。若不想分析所有可能对，则将邻近响应作为一对进行分析。下表显示四个响应的分析安排。

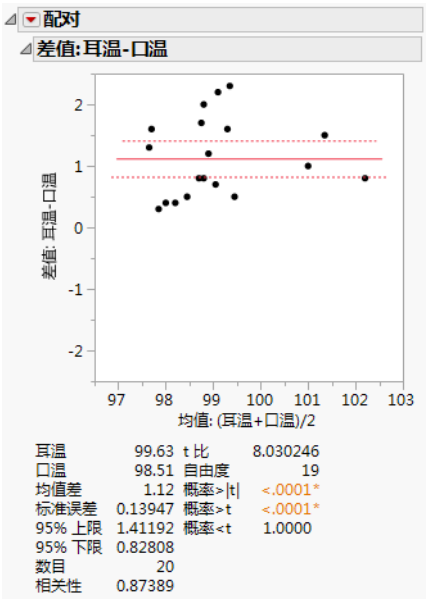
Y2-Y1	Y4-Y3

“配对”报表

请按第 343 页的“关于比较配对的示例”中的说明来生成图 20.4 中所示的报表窗口。

“配对”报表显示一个 Tukey 均值 - 差值图、汇总统计量和配对 *t* 检验的结果。请参见第 346 页的 “差值图和报表” 。若指定了 X，分组变量，该报表还会包含 “组间”报表。请参见第 346 页的 “组间”。

图 20.4 “配对”报表的示例



注意：红色小三角菜单提供可以向初始报表窗口添加报表的更多选项。请参见第 347 页的 “ “配对”平台选项”。

差值图和报表

“差值”图显示均值 - 差值图。在 “差值”图中，请注意以下事项：

- 均值差显示为一条水平线，95% 置信区间的上下限显示为虚线。若置信区域包括零，则均值在 0.05 水平下没有显著差异。在本例中，差异显著。
- 若添加了参考系，配对的均值通过垂线来表示。有关参考框架的详细信息，请参见第 347 页的 “ “配对”平台选项”。

“差值”报表显示每个响应的均值、均值差值以及差值的置信区间。“差值”报表还显示配对 *t* 检验的结果。

组间

注意：仅当指定了 X，分组变量的情况下才显示“组间”报表。

“组间”分析对应于简单的重复测量分析。您可以使用“拟合模型”平台的**多元方差分析**特质获得相同的检验结果。

均值差 显示配对的两列之间跨每组中的各行的差值的均值。换言之，这是对象间 - 对象内交互作用，或整区 - 裂区交互作用。

平均均值 显示配对的两列之间跨每组中的各行的均值的均值。换言之，这是对象间或整区效应。

组间检验 两个 *F* 检验确定组间值是否不同：

均值差 检验响应对之间的变化在不同组中是否不同。

平均均值 检验某个对象的平均响应在不同组中是否不同。

相关信息

- 第 348 页的“比较组间配对的示例”

“配对”平台选项

“配对”红色小三角菜单包含以下选项：

标绘 “均值 - 差值”图 显示或隐藏“配对均值 - 配对差值”图。有关该图的详细说明，请参见第 346 页的“差值图和报表”。

标绘 “行 - 差值”图 显示或隐藏“行号 - 配对差值”图。

参考系 在“标绘‘均值 - 差值’图”图上显示或隐藏参考系。有一个矩形显示“Y1-Y2”图在图上的位置，倾斜或者可能呈扁平状。显示一条表示平均均值的垂直红线。若差值范围大于数据范围的一半，则在初始就会显示参考系。

Wilcoxon 符号秩 显示或隐藏 Wilcoxon 符号秩检验。Wilcoxon 符号秩检验应用于配对差值。这是一个非参数检验，该检验将正差值的大小与负差值的大小作比较。该检验使用 Pratt 方法处理零差值。该检验还假设差值分布是对称的。请参见《基本分析》。另见 Lehman (2006)、Conover (1999, p. 350) 和 Cureton (1967)。

符号检验 显示或隐藏符号检验。这是非参数版本的配对 *t* 检验，仅使用差值的符号（正或负）进行检验。

设置 α 水平 更改分析中使用的 alpha 水平。影响报表中和图上的置信区间。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

比较组间配对的示例

本例使用 Dogs.jmp 样本数据表。本例显示如何生成 “组间配对”报表以及使用 “拟合模型”生成相应的 “多元方差分析”报表。

1. 选择**帮助 > 样本数据库**，然后打开 Dogs.jmp。
2. 选择**分析 > 专业建模 > 配对**。
3. 选择**组织胺 0 的对数**和**组织胺 1 的对数**，然后点击 **Y，配对响应**。
4. 选择**药物**并点击 **X，分组**。
5. 点击**确定**。

随即显示图 20.5 中左侧的报表。

现在使用相同的数据表生成 “拟合模型”报表。

1. 选择**分析 > 拟合模型**。
2. 选择**组织胺 0 的对数**和**组织胺 1 的对数**，然后点击 **Y**。
3. 选择**药物**，然后点击**添加**。
4. 选择**多元方差分析**特质。
5. 点击**运行模型**。
6. 在 “响应规格”报表中，从**选择响应**菜单中选择**重复测量**。
7. 点击**确定**。

图 20.5 组间配对和具有重复测量的拟合模型多元方差分析的示例

“组间”报表中的“均值差”的“F 比”对应于“对象内”报表下的“时间*药物”的“F 比”。“组间”报表中的“平均均值”的“F 比”对应于“多元方差分析拟合”报表中的“对象间”下的药物的“F 比”。

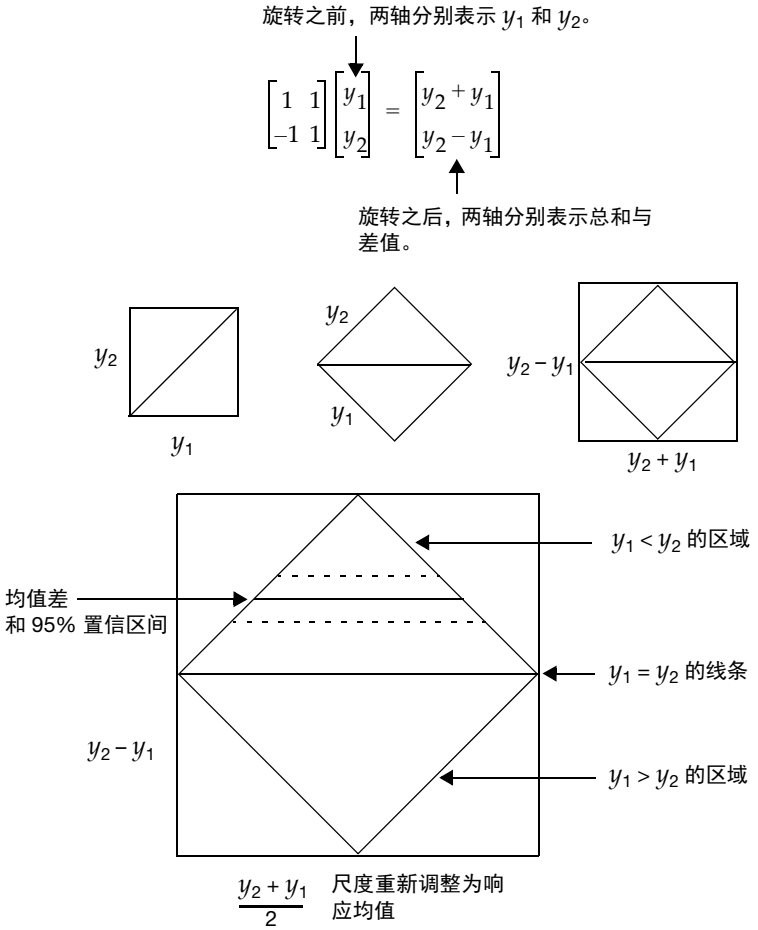
“配对”平台的统计详细信息

- “配对图”
- “响应的相关性”

配对图

平台中的主要图形是 Tukey 均值-差值图 (Cleveland 1994, p.130)。该图标绘的是针对水平轴上两个响应的均值的垂直轴上两个响应的差值。该图与两个原始变量的散点图相同，只不过旋转了 45 度。45 度旋转和尺度调整可以将原坐标变换为差值和均值。

图 20.6 旋转 45 度后变换为 “均值 - 差值”的示例

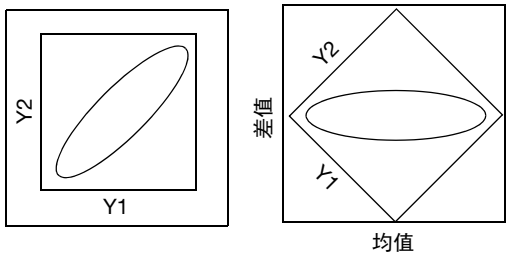


响应的相关性

若成对测量值取自同一个体在不同时间的值，则大多数情况下，它们之间都正相关。不过，若它们表示竞争的响应，则可能为负相关。

图 20.7 显示两个响应的正相关如何成为差值（垂直轴）上的小方差。若为负相关，则椭圆将在另一个方向上延伸，并且旋转图形的方差在垂直轴上较大。

图 20.7 旋转前后的正相关示例



第 21 章

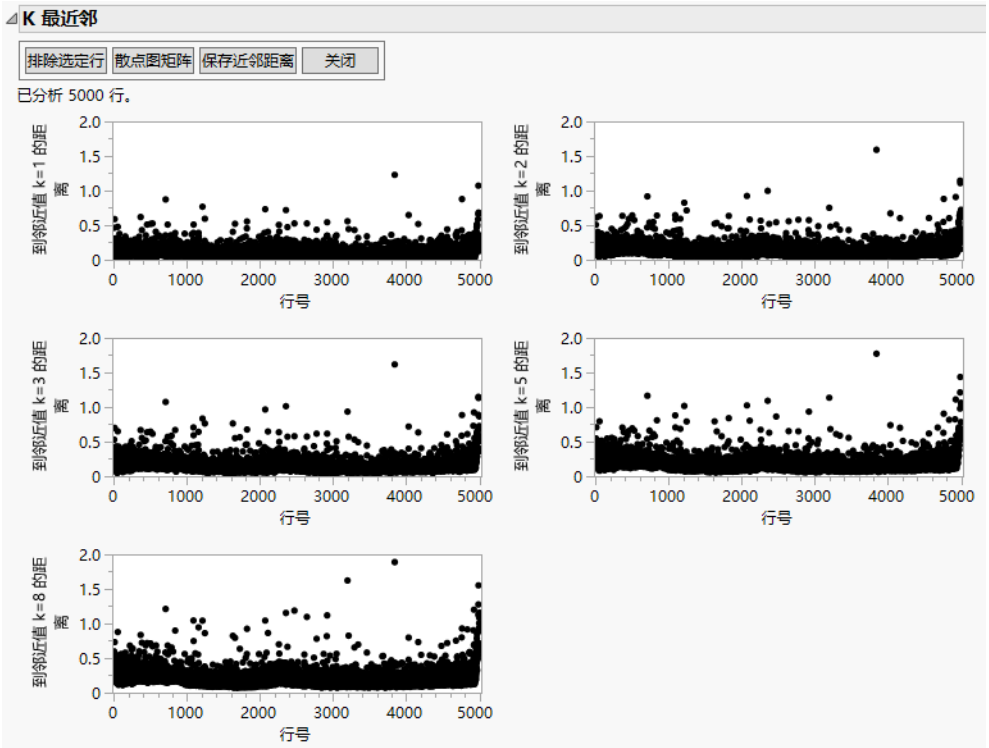
建模工具

探索离群值、缺失值和模式

建模工具是设计用于在数据分析的数据清理和预处理阶段中提供协助的一套实用工具。每个实用工具都提供探索性工具，支持您更深入彻底地了解数据。利用建模工具，您可以执行以下操作：

- 探索一元和多元情况下的离群值。
- 探索和插补数据中的缺失值。
- 探索并发现数据中的异常特性或模式。

图 21.1 多元 k 最近邻离群值示例



目录

“探索离群值”实用工具 355
“探索离群值”实用工具的示例 355
启动“探索离群值”实用工具 358
分位数范围离群值 359
稳健拟合离群值 362
稳健 PCA 离群值 364
K 最近邻离群值 366
“探索离群值”实用工具选项 368
“探索离群值”实用工具的更多示例 368
“探索缺失值”实用工具 370
“探索缺失值”实用工具的示例 370
缺失值报表 372
“探索缺失值”实用工具选项 377
“探索模式”实用工具 377
“探索模式”实用工具的示例 377
启动“探索模式”实用工具 379
“探索模式”报表 380
“探索模式”实用工具选项 381
“探索模式”实用工具的更多示例 383
“探索模式”实用工具的统计详细信息 385

“探索离群值”实用工具

“探索离群值”工具提供四个不同选项来标识、探索和管理离群值。探索和了解数据中的离群值是分析中的重要组成部分。数据中之所以出现离群值可能是因为：数据收集或报告出错、测量系统失效、在数据集中包含了错误或缺失值代码，或只是异常值。离群值的存在可能会使估计值失真，使结果向这些离群值偏倚。

离群值还会增大样本方差。不过有时保留数据中的离群值也是有必要的，删除它们会低估样本方差，让数据朝着反方向偏倚。

无论是删除还是保留离群值，最好都要找到这些离群值。可通过很多方式直观检查离群值。例如，箱线图、直方图和散点图就很容易显示这些极值。请参见《发现 JMP》。

“探索离群值”工具提供以下选项：

一元 可通过两个选项来探索一元数据中的离群值。

分位数范围离群值 使用每列的分位数分布将离群值标识为极值。该工具适用于发现数据中的缺失值或错误代码。开始探索数据中的离群值时推荐使用该方法。请参见第 359 页的“分位数范围离群值”。

稳健拟合离群值 查找每列的中心和散度的稳健估计值，将远离这些值的数据点标识为离群值。请参见第 362 页的“稳健拟合离群值”。

多元 可通过两个选项来探索多元数据中的离群值。

稳健 PCA 离群值 将数据分解为低秩矩阵和残差，并使用残差检测离群值。请参见第 364 页的“稳健 PCA 离群值”。

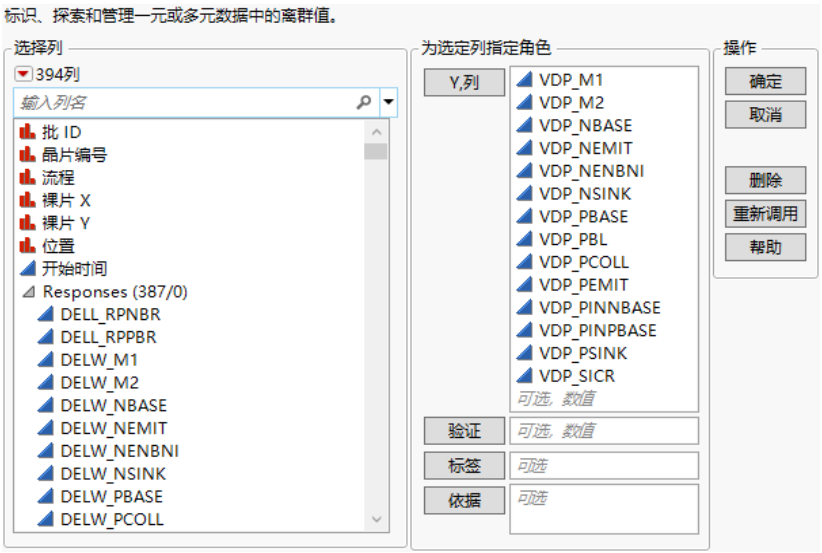
K 最近邻离群值 将离群值标识为远离其 k 最近邻的值。请参见第 366 页的“K 最近邻离群值”。

“探索离群值”实用工具的示例

使用“探索离群值”实用工具可标识离群值，随后可使用“分布”平台检查这些离群值。Probe.jmp 样本数据表包含对 5800 个半导体晶片测量的 387 个特征（**响应**列组）。**批 ID** 和**晶片编号**列唯一标识晶片。您关注的是：标识数据集中选定的一组列内的离群值。

1. 选择**帮助** > **样本数据库**并打开 Probe.jmp 样本数据表。
2. 选择**分析** > **筛选** > **探索离群值**。
3. 点击 Responses (387/0) 旁边的小三角以显示组中所有列。
4. 从列 VDP_M1 一直选择到 VDP_SICR，然后点击 **Y，列**。应已选定 14 列。

图 21.2 探索离群值启动窗口



5. 点击**确定**。
6. 点击**分位数范围离群值**。

   “分位数范围离群值”报表显示每列并列出找到的离群值的数量和标识。
7. 在“分位数范围离群值”报表中，选择**仅显示带离群值的列**。这会将列列表限定为仅包含离群值的那些列。

   请注意，有若干列包含离群值 9999。许多行业都将 9999 用作缺失值代码。
8. 在 “9999”报表中，选择每列。
9. 点击**将最高 9999 添加至缺失值代码**。

   会有一条 JMP 警示提醒您应该使用**另存为**命令保留原始数据。
10. 点击**确定**。
11. 在 “分位数范围离群值”报表中，点击**重新扫描**。
12. 选择**将搜索限制为整数**。

    在连续数据的情况下，整数值往往是错误代码或其他编码数据值。请注意，该组列中未包含其他错误代码。
13. 取消选择**将搜索限制为整数**。

检查数据

1. 选择“分位数范围离群值”报表中其余所有列。
2. 点击**选择行**。
3. 选择**分析 > 分布**。

4. 将选定列分配到 **Y，列**角色。由于您在“分位数范围离群值”报表中选定了这些列名，所以它们在“分布”启动窗口中已然处于选定状态。
5. 点击**确定**。

图 21.3 带有已选定离群值的列的分布

在列 VDP_M1 和 VDP_PEMIT 中，请注意有些选定的离群值与大多数数据有些接近。对于其余的列，选定的离群值看起来远离大多数数据。调查数据点并将其从分析中排除。

精简排除的离群值

1. 在“分位数范围离群值”报表中，按住 Ctrl 并取消选择列 VDP_M1 和 VDP_PEMIT。
2. 在选定报表中其余列的情况下，点击**排除行**。
3. 将 Q 改为 20。
4. 点击**重新扫描**。
5. 选择报表中的列 VDP_M1 和 VDP_PEMIT。
6. 点击**选择行**。

重新检查数据

1. 再次检查“分布”报表。请注意，选定的离群值现在足够远离大多数数据，可以选定它们并从分析中排除它们。

2. 在“分位数范围离群值”报表中，点击**排除行**。
3. 在“分布”报表中，点击“分布”红色小三角并选择**重新运行 > 重新运行分析**。

图 21.4 带有已排除离群值的列的分布

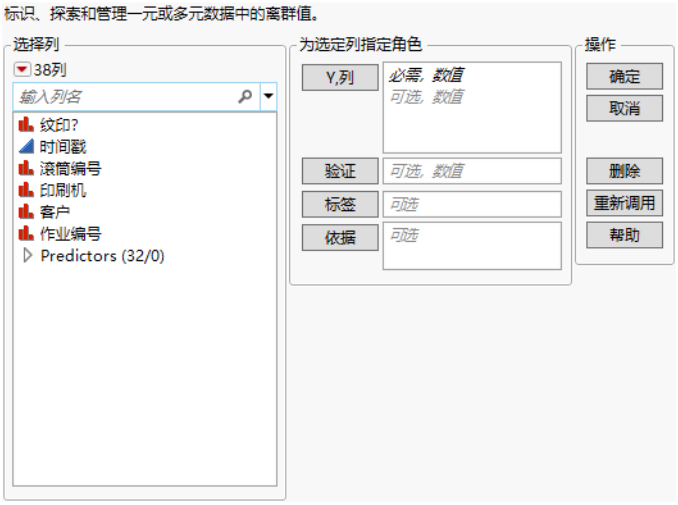
排除离群值后，所显示的数据分布现在更能够说明问题。

启动“探索离群值”实用工具

要启动“探索离群值”，请选择**分析 > 筛选 > 探索离群值**。

注意：“探索离群值”命令只分析建模类型为“连续”的列。可以在启动窗口中输入其他列，但这些列将被忽略。

图 21.5 “探索离群值”实用工具启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 指定要分析的列。

验证 指定用于“稳健 PCA 离群值”的验证列。

标签 指定用标签替换多元分析报表中的行号的列。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

点击**确定**后，将显示“探索离群值”报表。您可以看到以下四个离群值分析命令：

- 第 359 页的“分位数范围离群值”
- 第 362 页的“稳健拟合离群值”
- 第 364 页的“稳健 PCA 离群值”
- 第 366 页的“K 最近邻离群值”

提示：要对**依据**变量的所有水平运行离群值分析，请按 Ctrl 键并点击所需的离群值分析命令按钮。

分位数范围离群值

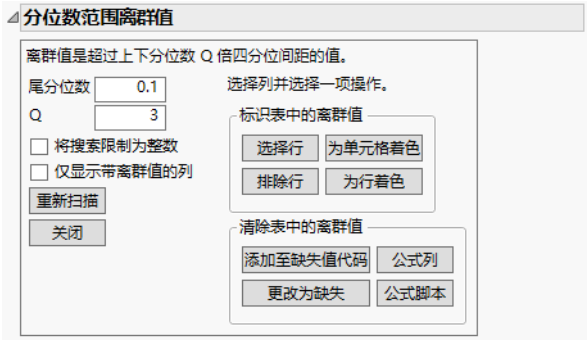
检测离群值的“分位数范围离群值”方法使用列中值的分位数分布来查找极值。分位数有助于检测离群值，因为分位数与分布假设无关。数据只是从小到大排序。例如，第 20 个分位数就是在这个值处，有 20% 的值较小。使用四分位间距（两个指定的分位数之间的距离）的倍数来查找极值。有关如何计算分位数的详细信息，请参见《基本分析》。

“分位数范围离群值”实用工具还有助于标识存储在数据中的缺失值代码。正如之前指出的那样，在某些行业中，缺失值输入为多个 9（如 999 或 9999）。该实用工具将大于上四分位数的所有 9 都标识为可疑的缺失值代码。随后该实用工具会支持您将那些缺失值代码添加为数据表中的列属性。

“分位数范围离群值”选项

“分位数范围离群值”面板支持您指定如何标识离群值以及您希望如何管理离群值。

图 21.6 “分位数范围离群值”窗口



离群值是超过上下分位数 *Q* 倍四分位间距的所有值。您可以调整 *Q* 值和四分位间距的大小。

尾分位数 用于计算四分位间距的下分位数的概率。上分位数的概率通过“1 - 尾分位数”计算得出。例如，尾分位数值 0.1 意味着四分位间距介于数据的 0.1 和 0.9 分位数之间。默认值为 0.1。

Q 用于确定离群值阈值的乘数。落在超过“尾分位数”值或“1 - 尾分位数”值的 *Q* 倍四分位间距之外的值将标识为离群值。比起较小值，较大的 *Q* 值能提供更为保守的一组离群值。默认值为 3。

将搜索限制为整数 将离群值仅限定为整数值。该设置限制搜索离群值，以便查找行业特定的缺失值代码和错误代码。

仅显示带离群值的列 将报表中的列列表限定为仅包含离群值的那些列。

使用您的指定显示报表后，可通过许多方法对这些极值进行操作。您可以通过选择“分位数范围离群值”报表中的指定列来选择列中的离群值。

提示：若在报表中未选择任何列，并且您点击以下按钮之一，则会显示一条“JMP 警示”，支持您选择所有列。

选择行 选择数据表中选定列内的各行离群值。

排除行 开启“分位数范围离群值”报表的选定列中的离群值的排除行状态。点击**重新扫描**更新“分位数范围离群值”报表。

为单元格着色 对数据表中选定的离群值的单元格着色。

为行着色 为包含数据表中选定列的离群值的行着色。

添加至缺失值代码 将选定的离群值添加至“缺失值代码”列属性。使用该选项标识数据中已知的缺失值或错误代码。缺失值和错误代码通常是整数，有时是一串 9。点击**重新扫描**更新“分位数范围离群值”报表。

注意：若在启动窗口中指定了“依据”变量，则“添加至缺失值代码”不可用于“分位数范围离群值”。

更改为缺失 将离群值改为数据表中的缺失值。将值改为缺失值时务必要小心。只有在明确数据无效或不准确时才将值改为缺失值。点击**重新扫描**更新“分位数范围离群值”报表。

注意：若选定的离群值已添加到缺失值代码中，则该离群值不会更改为缺失值。

公式列 为启动窗口中指定的每列创建新的公式列。若原始列的值位于离群值限值内，则每个新列都包含原始列的值，否则设置为缺失。新列以用户指定的名称作为前缀或后缀，以区别于原始列。默认情况下，后缀设置为“已剔除”。

公式脚本 创建一个添加到数据表的脚本。运行该脚本时，为启动窗口中指定的每列创建新的公式列。若原始列的值位于离群值限值内，则每个新列都包含原始列的值，否则设置为缺失。新列以用户指定的名称作为前缀或后缀，以区别于原始列。默认情况下，后缀设置为“已剔除”。

重新扫描 在执行离群值操作之后重新扫描数据。

注意：按 Ctrl 键并点击**重新扫描**将重新扫描所有命令组。

关闭 关闭“分位数范围离群值”面板。

注意：按 Ctrl 键并点击**关闭**可关闭所有命令窗口。

“分位数范围离群值”报表

“分位数范围离群值”报表列出使用指定选项发现的所有包含离群值的列。该报表显示上下分位数的值及其上下阈值。这些阈值之外的值被视为离群值。报表会指出每列中的离群值数。每个离群值的值列在报表的最后一列中。对于某列中出现多次的离群值，还在括号中列出其出现的次数。要从列表中删除不带离群值的列，请选择**仅显示带离群值的列**。

读取该报表时需要注意以下几点。

- 错误代码。对于某些连续数据，可疑的较高整数值很可能是错误代码。例如，若上下分位数值都小于 0.5，则 1049 或 -777 之类的离群值就很可能是错误代码。
- 零。有时零也指示缺失值。若大多数数据都相当大，而您注意到离群值为零，则零很可能是缺失数据。

“9999”报表

“分位数范围离群值”窗口中的“9”报表显示一个列列表，其中包含可能的缺失值代码。 这些缺失值代码是一串 9（通常为 9999），而且是所有一连串 9 的数字中的最大值并且高于上分位数。若计数较高，则这些离群值很可能实际上是缺失值代码。若计数很低，则应该进一步研究以确定该值是离群值还是缺失值代码。“9999”报表包括上分位数值。

仅当识别出可能的缺失值代码时才显示该报表。

将最高 9 添加至缺失值代码 将选定的离群值添加至 “缺失值代码”列属性。您必须点击**重新扫描**更新 “分位数范围离群值”报表。

注意：若在启动窗口中指定了 “依据”变量，则 “将最高 9999 添加至缺失值代码”不可用于 “分位数范围离群值”。

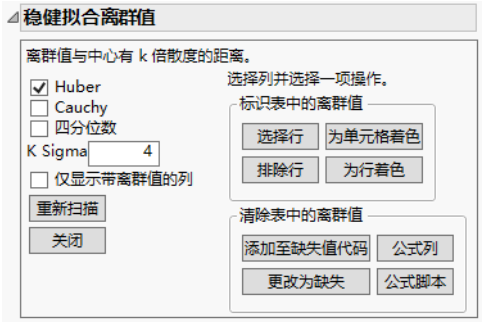
将最高 9 更改为缺失 用数据表中的缺失值替换选定的离群值。

注意：首次通过选择某个操作（比如：**更改为缺失**或**排除行**）来更改数据时，警示窗口会提醒您使用**另存为**命令将数据表另存为新文件，以保留原始数据的副本。显示该窗口时，请点击**确定**。若决定保存新数据文件，系统会自动提示您使用新名称保存该文件。

稳健拟合离群值

比起非稳健估计值，参数的稳健估计值对于离群值较不敏感。“稳健拟合离群值”提供您数据的中心和散度稳健估计值的若干类型，以确定用于标识离群值的阈值。

图 21.7 “稳健拟合离群值”窗口



“稳健拟合离群值”选项

在给定中心和散度稳健估计值的前提下，离群值定义为与中心有 *K* 倍稳健散度的距离的那些值。“稳健拟合离群值”窗口提供若干选项，用于计算稳健估计值和倍数 *K*，并提供用于管理发现的离群值的工具。

Huber 使用 Huber M 估计来估计中心和散度。该选项是默认选项。请参见 Huber and Ronchetti (2009)。

Cauchy 用 Cauchy 分布来计算中心和散度的估计值。 Cauchy 估计值提供高度细分的点，通常比 Huber 估计值更为稳健。不过，若数据划分为聚类，则 Cauchy 分布往往只考虑构成较近聚类的一半数据，而忽略其余数据。

四分位数 使用四分位间距 (IQR) 估计散度。中心的估计值为中位数。散度估计值是 IQR 除以 1.34898 后的结果。用该因子除以 IQR 可令散度对应于一个标准差 （若为正态分布数据）。

K 一个倍数，用于确定离群值与中心有 *K* 倍散度的距离。比起较小值，较大的 *K* 值能提供更为保守的一组离群值。默认值为 4。

仅显示带离群值的列 将报表中的列列表限定为仅包含离群值的那些列。

一旦使用您的指定显示报表，即可通过许多方法来探索这些极值。您可以通过选择 “稳健估计值和离群值”报表中的指定行来选择行中的离群值。

提示：若在报表中未选择任何列，并且您点击以下按钮之一，则会显示一条“JMP 警示”，支持您选择所有列。

选择行 选择包含数据表中选定列的离群值的行。

排除行 为数据表中选定列的离群值设置 “排除行”状态。点击**重新扫描**以更新 “稳健估计值和离群值”报表。

为单元格着色 对数据表中选定的离群值的单元格着色。

为行着色 为包含数据表中选定列的离群值的行着色。

添加至缺失值代码 针对选定列，将选定的离群值添加至 “缺失值代码”列属性。使用该选项标识数据中已知的缺失值或错误代码。点击**重新扫描**以更新 “稳健估计值和离群值”报表。

注意：若在启动窗口中指定了 “依据”变量，则 “添加至缺失值代码”不可用于 “稳健拟合离群值”。

更改为缺失 将离群值改为数据表中的缺失值。点击**重新扫描**以更新 “稳健估计值和离群值”报表。

公式列 为启动窗口中指定的每列创建新的公式列。若原始列的值位于离群值限值内，则每个新列都包含原始列的值，否则设置为缺失。新列以用户指定的名称作为前缀或后缀，以区别于原始列。默认情况下，后缀设置为 “已剔除”。

公式脚本 创建一个添加到数据表的脚本。运行该脚本时，为启动窗口中指定的每列创建新的公式列。若原始列的值位于离群值限值内，则每个新列都包含原始列的值，否则设置为缺失。新列以用户指定的名称作为前缀或后缀，以区别于原始列。默认情况下，后缀设置为“已剔除”。

重新扫描 在执行离群值操作之后重新扫描数据。

注意：按 Ctrl 键并点击**重新扫描**将重新扫描所有命令组。

关闭 关闭“稳健拟合离群值”面板。

注意：按 Ctrl 键并点击**关闭**可关闭所有命令窗口。

稳健 PCA 离群值

您可以使用“稳健 PCA 离群值”实用工具快速识别相关多元数据中的离群值单元格。这种方法很有用，因为许多其他多元方法只识别离群值行。在将该方法应用于数据之前，首先对列进行中心化（可选）和统一尺度。尺度因子定义如下：

max [Q(.75) - Q(.50), Q(.50) - Q(.25)] / [正态分位数 (0.75)]

其中

Q(p) 是第 p 分位数

注意：若 Q(75) 或 Q(25) 等于中位数，则会使用更极端的分位数，直到有一个非零的范围。

对数据进行中心化和统一尺度后，“稳健 PCA 离群值”实用工具执行一系列奇异值分解和阈值处理步骤，对数据矩阵进行分解。数据被分解成一个低秩矩阵和一个稀疏的残差矩阵。阈值的处理会使得残差要么非常大（对于离群值），要么非常接近零（对于非离群值）。该算法确定一个适于捕获系统化变异而没有离群值或小噪声的矩阵秩。对于不在低秩空间中的离群值的检测是基于对其残差的检测。请参见 Candes et al (2009) 和 Lin et al (2013)。若有缺失值，则在中心化和统一尺度步骤之后，它们最初将替换为零。然后，在每一次奇异值分解 (SVD) 迭代之后，用来自 SVD 的缺失值的预测值来更新这些缺失值。

“稳健 PCA 离群值”报表

从命令列表中选择“稳健 PCA 离群值”后，您必须指定 Lambda 的值并且选择是否应对数据进行中心化。若在按 Shift 键的同时点击“稳健 PCA 离群值”按钮，则同时提供以下选项：

Lambda 指定一个值，该值确定残差矩阵的稀疏性。对于较大的 Lambda 值，残差矩阵较为稀疏。对于具有 n 个训练行和 p 个列的数据表， Lambda 的默认值定义如下：

$$\lambda = \frac{2}{\sqrt{\max(n, p)}}$$

最大迭代次数 指定 SVD 的最大迭代次数。

收敛准则 确定何时停止该算法。

离群值阈值 指定离群值阈值，用于确定 “单元格大残差”表中显示哪些离群值。若统一尺度的残差大于以下值，则显示一个观测：

min[0.99 × max{abs(残差)}, 离群值阈值]

默认情况下，“离群值阈值”为 2。

中心 确定在执行 “稳健 PCA 离群值”算法之前是否对数据中心化。

注意：若行数小于等于 10，则不对数据进行中心化。

尺度 确定在执行 “稳健 PCA 离群值”算法之前是否对数据统一尺度。

注意：若行数小于等于 10，则不对数据统一尺度。

“稳健 PCA 离群值”报表包含一个表，其中包含关于该方法的信息。该表包括低秩矩阵的秩、SVD 迭代次数、收敛准则、 Lambda 值和插补缺失值的数量。该报表还包含以下表和报表：

单元格大残差 显示最大离群值单元格的表。显示的单元格数量取决于 “离群值阈值” 。该表包含单元格的列名和行号、残差值和统一尺度的残差值。

提示：要对数据表中的特定离群值单元格着色，请选择“单元格大残差”表中的行并点击**着色**。

均方根行 显示数据表中每行的均方根值的表。使用统一尺度的残差计算均方根。

提示：若在 “均方根行”表中选择了某行，在数据表中会选定相应的行。

均方根列 显示启动窗口中指定的每列的均方根值的表。使用统一尺度的残差计算均方根。

提示：若在“均方根列”表中选择某行，然后点击**选择列**，则会在数据表中选定相应的列。

快照 数据表中离群值单元格的图形表示。离群值单元格着红色。

残差 矩阵分解后的残差矩阵。若统一尺度残差的绝对值大于以下值，则会对单元格着色：

min[0.99 × max{abs(残差)}, 离群值阈值]

低秩近似 矩阵分解后的统一尺度残差矩阵。

奇异值 来自 SVD 的奇异值向量。

“稳健 PCA 离群值”选项

“稳健 PCA 离群值”报表底部有一些按钮，提供保存报表不同部分的选项。

关闭 关闭“稳健 PCA 离群值”报表。

保存大离群值 将“单元格大残差”表中的信息保存到新的数据表中。

保存已清除 打开一个窗口，其中提供了几种方法，可以根据阈值清除离群值并将新列保存到数据表中。

切尾 若相应的绝对统一尺度残差大于指定阈值，则去除离群值单元格。默认情况下，阈值为 10。选择“颜色” 为离群值单元格着红色。去除的单元格设置为未统一尺度的阈值。

插补 若相应绝对统一尺度残差大于指定阈值，则将离群值单元格设置为低秩近似的值。默认情况下，阈值为 100。选择“颜色” 为这些单元格着绿色。

生成缺失值 若相应的绝对统一尺度残差大于指定阈值，则将离群值单元格设置为缺失。默认情况下，阈值为 1000。选择“颜色” 为这些单元格着蓝色。

对缺失值插补着色 若选定，则为最初缺失值并被插补的单元格着色。

保存残差 将残差保存至原始数据表中的新列。

保存统一尺度的残差 将统一尺度的残差保存至原始数据表中的新列。

保存低秩近似 将低秩近似保存至原始数据表中的新列。

K 最近邻离群值

使用“K 最近邻离群值”根据到其最近邻的距离来识别离群值。对于每个 *K* 值，“K 最近邻离群值”实用工具都显示每个点到其第 *K* 个最近邻的欧氏距离图。您指定 *k* 的最大值，表示为 *K*。为 *k* = 1,2,3,5,...,*K* 提供了图，使用 Fibonacci 序列以避免显示过多图。

在计算最近邻之前，对列进行中心化和统一尺度。尺度因子定义如下：

max [Q(.75) - Q(.50), Q(.50) - Q(.25)] / [正态分位数 (0.75)]

其中

Q(*p*) 是第 *p* 分位数

注意：若 Q(75) 或 Q(25) 等于中位数，则会使用更极端的分位数，直到有一个非零的范围。

该方法对于指定的 *k* 值很敏感。较小的 *k* 可能会漏标离群值点，较大的 *k* 可能会将点误分类为离群值：

- 假定指定的 *k* 较小，以便您只研究少数近邻。若有超过 *k* 个点的聚类远离其他点，则该聚类内的点与其最近邻的距离较小。您可能无法检测离群值聚类。
- 假定指定的 *k* 较大，以便您研究大量近邻。若聚类内的数据点少于 *k* 个，则这些聚类内的点可能看起来就是离群值。此时您就可以忽略点构成聚类这一事实，而将单个聚类成员解释为离群值。

“K 最近邻离群值”报表

从命令列表中选择“K 最近邻离群值”时，您需要指定 *k* 的值，以用作要考虑的最远邻居的上限。还必须指定是否应插补缺失值。请注意，*K* 设置为 8，并且默认选定“插补缺失值”。

该报表显示选定的 *k* 值一直到 *K* 值的图。每个图的 *k* 值都显示在其垂直轴标签内。该标签的格式为“到邻近值 K = <a> 的距离”，其中，a 是一个表示第 *a* 个最近邻的整数。每个图都显示从第 *i* 行中的点到其第 *a* 个最近邻的距离。与其近邻的距离较远、跨多个 *K* 值的点很可能是离群值。

图上方的按钮用于执行以下操作：

排除选定行 将与选定点对应的行从进一步分析中排除。这些行在数据表中被分配排除行状态。您需要决定是要重新运行还是关闭“K 最近邻”报表。重新运行该分析将标识新的最近邻。图随之更新，排除的点不再显示。

散点图矩阵 打开一个单独的窗口，其中包含分析中的所有列的散点图矩阵。您可以探索可能的离群值，这需要在 “K 最近邻”图中选择这些离群值并在散点图矩阵中查看它们。

保存近邻距离 将每行到其第 *n* 个最近邻的距离另存为数据表中的新列。

关闭 关闭 “K 最近邻”报表。

最大离群值

该报表还包括“最大离群值”表。该表包含与其 *K* 个最近邻距离最大的 20 个观测。表中包含以下列：

行 观测的行号。

距离 指定行中的观测到第 *K* 个最近邻的距离。该表按该列降序排序。

最近邻 列出 *k* 个最近邻的行号。第一个行号是最近的最近邻。最后一个行号是第 *K* 个最近邻，该观测与指定行之间的距离可在 “距离”列中找到。

列 <n> 指定相应的 RSM 值的列名。

RSM<n> 计算每列的 *k* 最近邻之间的均方根差值。五个最大 RSM 值按顺序显示，其中 RSM1 是最大 RSM 值。第 *p* 个 RSM 值计算如下：

$$\mathrm{RSM}_p = \sqrt{\frac{\sum_{k=1}^{K} (D_{p,i} - D_{p,i_k})^2}{K}}$$

其中

D_p 是第 *p* 列

$D_{p,i}$ 是行 *i* 的第 *p* 列的值

$D_{p,ik}$ 是行 *i* 的第 *k* 个最近邻的第 *p* 列的值

注意：“最大离群值”表中显示的“列”和 RSM 列的数目是启动中指定的列数与数字 5 中的最小值。

“探索离群值”实用工具选项

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

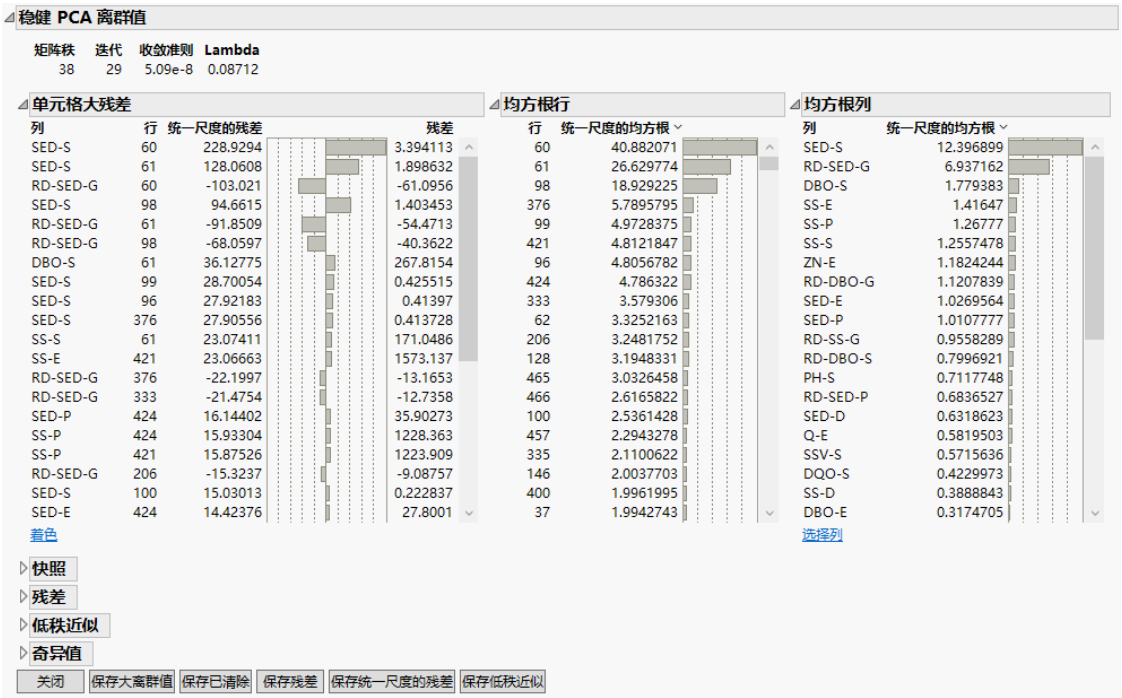
“探索离群值”实用工具的更多示例

“稳健 PCA 离群值”示例

Water Treatment.jmp 数据集包含来自一家城市污水处理厂的 38 个传感器的每日测量值。您关注的是探索这些数据中的可能离群值。可能的离群值可包括传感器故障、风暴和其他情况。

1. 选择**帮助** > **样本数据库**，然后打开 Water Treatment.jmp。
2. 选择**分析** > **筛选** > **探索离群值**。
3. 选择 Sensor Measurements 列组并点击 **Y，列**。
4. 点击**确定**。
5. Shift+Click **稳健 PCA 离群值**。
6. 在 “离群值阈值”旁边输入 10。
   使用其他选项的默认值。
7. 点击**确定**。

图 21.8 “稳健 PCA 离群值”报表



“单元格大残差”表显示统一尺度的残差大于 10 或小于 -10 的单元格。 SED-S 列中有若干单元格具有较大的统一尺度的残差。若您查看 “均方根列”表，它显示 SED-S 列实际上确实具有最大的统一尺度的“均方根”值。具有最大的统一尺度“均方根”值的行为第 60 行。

8. 点击报表底部的**保存已清除**。

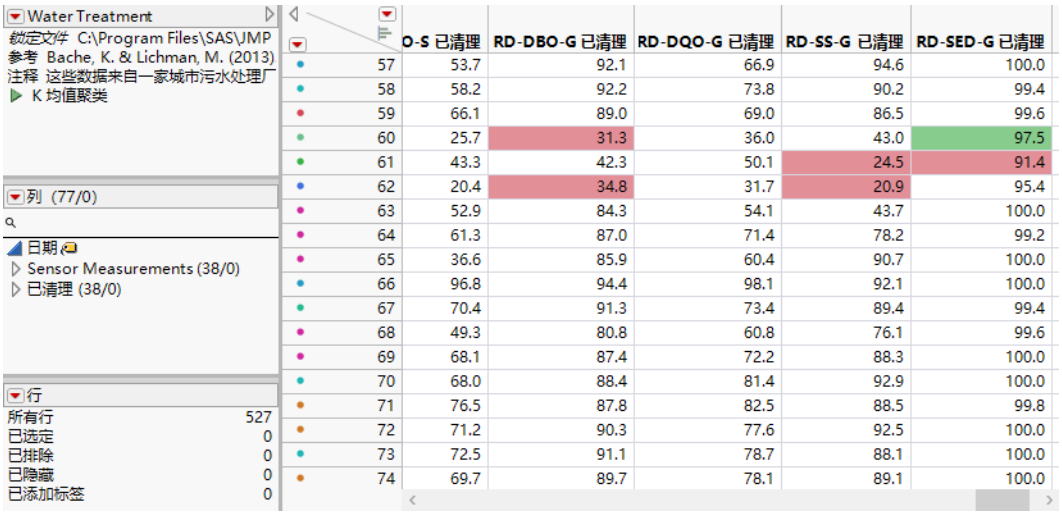
   该选项支持您通过去除值、插补新值或将单元格设置为缺失来调整具有较大离群值的单元格。“保存清理的列”报表提供以下选项。

9. 选择 “插补”。默认已选定 “切尾”。

10. 点击**确定**。

    有 38 个新的已清理列添加到 Water Treatment.jmp 数据表中。因为您知道第 60 行的统一尺度“均方根”值最大，所以请滚动到该行以查看已去除和插补的一些单元格。您现在可以在任意类型的分析中使用已清理的列。

图 21.9 “稳健 PCA 离群值”中的已清理列



	O-S 已清理	RD-DBO-G 已清理	RD-DQO-G 已清理	RD-SS-G 已清理	RD-SED-G 已清理
57	53.7	92.1	66.9	94.6	100.0
58	58.2	92.2	73.8	90.2	99.4
59	66.1	89.0	69.0	86.5	99.6
60	25.7	31.3	36.0	43.0	97.5
61	43.3	42.3	50.1	24.5	91.4
62	20.4	34.8	31.7	20.9	95.4
63	52.9	84.3	54.1	43.7	100.0
64	61.3	87.0	71.4	78.2	99.2
65	36.6	85.9	60.4	90.7	100.0
66	96.8	94.4	98.1	92.1	100.0
67	70.4	91.3	73.4	89.4	99.4
68	49.3	80.8	60.8	76.1	99.6
69	68.1	87.4	72.2	88.3	100.0
70	68.0	88.4	81.4	92.9	100.0
71	76.5	87.8	82.5	88.5	99.8
72	71.2	90.3	77.6	92.5	100.0
73	72.5	91.1	78.7	88.1	100.0
74	69.7	89.7	78.1	89.1	100.0

“探索缺失值”实用工具

数据集中有缺失值可能影响使用数据作出的结论。例如，若有若干健康的参与者退出了纵向研究，而他们的数据之后一直缺失，则研究结果可能会向留下的不健康的个人偏倚。缺失数据值不仅必须要标识出来，还必须在执行进一步分析之前了解这些缺失数据值。

“探索缺失值”实用工具提供若干方法来帮助您标识和了解数据中的缺失值。它还提供方法来执行缺失值的多元正态插补。这些插补方法假设数据**随机缺失**，这意味着缺失数据和非缺失数据之间的任何差异都不能用研究中其他变量的值来解释。若您怀疑缺失值**不是**随机缺失，则考虑使用“信息性缺失”过程，有若干平台都提供该过程。请参见《拟合线性模型》。

警告：插补缺失值后分析数据时一定要小心，因为结果可能有偏倚。

“探索缺失值”实用工具的示例

Arrhythmia.jmp 样本数据表包含来自 452 位患者的心电图 (ECG) 信息。最初收集数据是为了将不同模式的 ECG 归类为心率不齐。不过，该数据表中有缺失值。您主要关注的是探索这些缺失值并在必要时进行插补。由于您仅为建模类型为“连续”的列执行了缺失值插补，所以您需要在两个阶段中执行分析。

检查缺失值

1. 选择**帮助 > 样本数据库**，然后打开 Arrhythmia.jmp。
2. 选择**分析 > 筛选 > 探索缺失值**。
3. 选择所有列 （共计 280 个）并点击 **Y，列**。

4. 点击**确定**。选中**仅显示带缺失值的列**复选框。

图 21.10 “缺失值”报表

图 21.10 中所示的 “缺失列”报表指示只有五列有缺失数据。在总共 452 行中，列 J 有 376 个缺失值。因为它缺失大量，插补值可能不会生成有意义的分析。对于此类数据，您可以在支持 “信息性缺失”选项的平台中使用**列 J** 的 “信息性缺失”选项来探索模型。

请注意，两个 “插补”选项 — “多元插补”和 “多元 SVD 插补”并未显示。有一条消息指示插补已被禁用，因为分析中包括的某些列是分类列。该数据表包含的若干列是数值列，但建模类型是名义型。这些均不可用于插补。

插补缺失值

包含缺失值的五列是连续列。您继续使用针对数据表中的连续列的多元插补来插补除**列 J** 外的四列的值。通过这样操作，您默认假定：值缺失的概率仅依赖于连续变量的值而不依赖于排除的名义型变量的值。要执行这个新分析，您需要再次启动 “探索缺失值”实用工具。

1. 选择**分析 > 筛选 > 探索缺失值**。
2. 在启动窗口中，点击 **280 列**旁边的红色小三角。

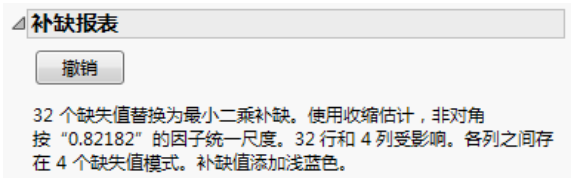
   使用列过滤器菜单仅查看 “选择列”列表中建模类型为 “连续”的列。
3. 选择**建模类型 > 全部取消选中**。

   这将从 “选择列”列表中删除所有列。
4. 选择**建模类型 > 连续型**。

   “选择列”列表现在仅包含 207 个 “连续”列。
5. 选择全部 207 列。然后在按住 Ctrl 键的同时点击 J 列 （将其取消选定），然后点击 **Y，列**。
6. 点击**确定**。

7. 点击**多元正态插补**。
   随即显示一个窗口，询问您是否要对协方差使用收缩估计量。
8. 点击**是，收缩**。
   随即显示一条 JMP 警示，提醒您应该使用**另存为**命令保留原始数据。
9. 点击**确定**。

图 21.11 插补报表



“插补报表”指示插补了多少缺失值以及具体的插补详细信息。曾经包含缺失值的四列中不再留有任何缺失数据。

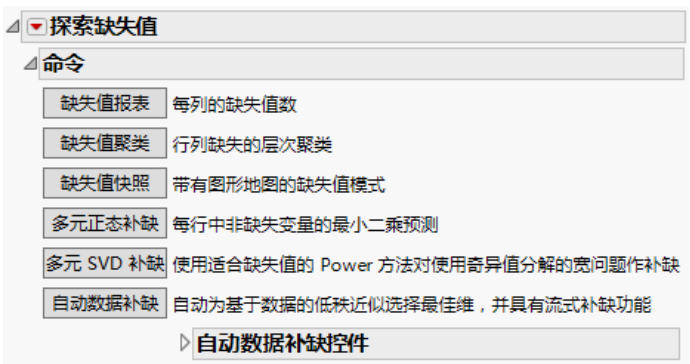
启动“探索缺失值”实用工具

通过选择**分析 > 筛选 > 探索缺失值**启动“探索缺失值”建模工具。将关注的列输入“Y，列”列表。您还可以指定**依据**变量。

注意：您只能在“探索缺失值”实用工具中输入建模类型为“数值”的列。

缺失值报表

图 21.12 Arrhythmia.jmp 中连续变量的缺失值报表



在启动窗口中点击“确定”后，该报表随即打开以显示“命令”分级显示项和一个“缺失列”报表。命令如下：

- 第 373 页的“‘缺失值’报表”

- 第 373 页的“缺失值聚类”
- 第 374 页的“缺失值快照”
- 第 374 页的“多元正态插补”（若在启动窗口中输入了建模类型为“名义型”或“有序型”的“数值”列则不可用。）
- 第 374 页的“多元 SVD 插补”（若在启动窗口中输入了建模类型为“名义型”或“有序型”的“数值”列则不可用。）
- JMP PRO 第 375 页的“自动数据插补”（若在启动窗口中输入了建模类型为“名义型”或“有序型”的“数值”列则不可用。）

提示：要对**依据**变量的所有水平运行缺失值命令，请按 Ctrl 键并点击所需的命令按钮。

“缺失值”报表

使用“缺失值报表”可打开“缺失列”报表，其中列出每列的名称和该列中缺失值的个数。

仅显示带缺失值的列 从列表中删除不含缺失值的列。

关闭 关闭“缺失列”报表。

选择行 选择数据表中的特定行，这些行包含您在“缺失列”报表中选择的一个或多个列的缺失值。

排除行 针对数据表中的特定行应用排除行状态，这些行包含您在“缺失列”报表中选择的列的缺失值。

为单元格着色 为数据表中的特定单元格着色，这些单元格包含您在“缺失列”报表中选择的列的缺失值。

为行着色 为数据表中的特定行着色，这些行包含您在“缺失列”报表中选择的一个或多个列的缺失值。

缺失值聚类

“缺失值聚类”提供对缺失数据的层次聚类分析。

- 图右侧的树状图显示缺失数据模式行的聚类。这些是您通过使用“表”>“缺失数据模式”得到的行。
- 图下方的树状图显示变量聚类。

使用该报表可确定特定列组是否倾向于具有相似的缺失值模式。

图的各行通过缺失数据模式来定义；每个模式对应一行。列对应变量。每个红色方格都指示图下方所列的列的一组缺失值。悬停在方格上方可查看所代表的值的列表。在图中点击可选择缺失数据模式行。显示垂直直条以指示选定的模式。

缺失值快照

“缺失值快照”显示缺失值的方格图。列表示变量。黑色方格指示缺失值。该图对于理解纵向数据的缺失尤其有用；缺失纵向数据时，对象可能在数据收集期结束之前退出研究。

多元正态插补

“多元正态插补”实用工具基于多元正态分布插补缺失值。该过程要求所有变量的建模类型都为“连续型”。该算法使用最小二乘插补。使用配对协方差构造协方差矩阵。使用每个变量的所有非缺失值计算对角线元素（方差）。使用任意两个变量均无缺失的所有观测计算这两个变量的非对角线元素。对于协方差矩阵奇异的情况，算法使用基于 Moore-Penrose 伪逆矩阵的最小范数最小二乘插补。

“多元正态插补”允许选择对协方差使用收缩估计量。使用收缩估计量是改善协方差矩阵的估计的一种方法。有关收缩估计量的详细信息，请参见 Schafer and Strimmer (2005)。

注意：若指定了验证列，则使用来自训练集的观测计算协方差矩阵。

“多元正态插补”报表

插补报表解释多元插补过程的结果。结果包括以下内容：

- 插补方法（最小二乘或最小范数最小二乘）
- 替换了多少值
- 是否使用收缩估计量
- 非对角线元素按哪个因子统一尺度
- 有多少行列受到影响
- 有多少不同的缺失值模式

一旦完成插补，与数据表中的插补值对应的方格就会变为浅蓝色。若“缺失列”报表处于打开状态，则该报表将更新以显示没有缺失值。

点击**撤销**可撤销插补，用缺失值替换插补的数据。

多元 SVD 插补

“多元 SVD 插补”实用工具使用奇异值分解 (SVD) 插补缺失值。该实用工具适用于有数以百计或数以千计变量的数据。由于 SVD 计算不要求计算协方差矩阵，所以推荐对包含大量变量的广泛问题使用 SVD 方法。该过程要求所有变量的建模类型都为“连续型”。

奇异值分解将观测矩阵 **X** 表示为 $\mathbf{X} = \mathbf{UDV}'$，其中，**U** 和 **V** 是正交矩阵，**D** 是对角矩阵。

“多元 SVD 插补”实用工具中默认使用的 SVD 算法是稀疏 Lanczos 方法，亦称**隐式重新启动的 Lanczos 双对角化方法** (IRLBA)。请参见 Baglama and Reichel (2005)。该算法执行以下计算：

1. 每个缺失值都用其列均值替代。

2. 针对观测矩阵 **X** 执行 SVD 分解。
3. 包含缺失值的每个单元格都替换为从 SVD 分解获得的 **UDV′** 矩阵的相应元素。
4. 重复执行步骤 2 和 3，直到 SVD 收敛到矩阵 **X**。

“插补方法”窗口

点击“多元 SVD 插补”后，“插补方法”窗口即显示推荐的设置。

奇异向量数 在插补中计算和使用的奇异向量数。

> **注意：**一定注意：不要指定过多的奇异向量，否则 SVD 和插补不会在各次迭代之间改变。

最多迭代次数 插补缺失值时使用的迭代次数。

显示迭代日志 打开一个“详细信息”报表，其中显示迭代次数并提供关于准则的详细信息。

对于较大的问题，进度条会显示 SVD 完成了多少个维。您可以停止插补并随时使用该维数。

“多元 SVD 插补”报表

插补报表解释多元插补过程的结果。

- 插补方法
- 替换了多少值
- 有多少行列受到影响

一旦完成插补，即自动显示“缺失列”报表，该报表指示插补的列中没有缺失值。插补值显示为浅蓝色。

点击**撤销**可撤销插补，用缺失值替换插补的数据。

JMP PRO 自动数据插补

“自动数据插补”(ADI) 实用工具使用低秩矩阵近似方法（亦称矩阵完成）插补缺失值。经过训练后，ADI 模型能够对得分公式中的流数据执行缺失数据插补。流数据是随时变得可用而添加的观测行，并不用于优化或验证插补的模型。该实用工具灵活稳健，可以为低秩近似自动选择最佳维。这些功能使得 ADI 能够很好地适用于许多不同类型的数据集。

矩阵的低秩近似采用 **X** = **UDV′** 的形式，可视为奇异值分解 (SVD) 的扩展。ADI 将 Soft-Impute 方法用作插补模型，而且经过专门设计，使得数据能够确定低秩近似的秩。

ADI 算法执行以下步骤：

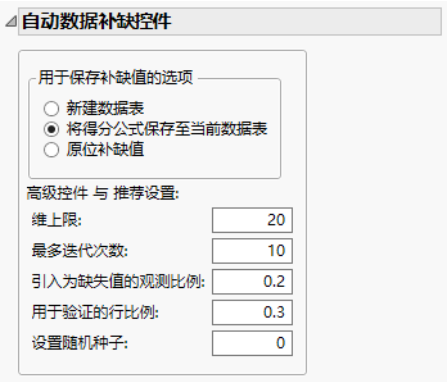
1. 数据划分到各个训练集和验证集中。
2. 每个集都使用训练集中的观测值进行中心化和统一尺度。
3. 对于每个划分的数据集，在每列中添加其他缺失值，这些值被称为引入的缺失 (IM) 值。
4. 针对训练数据集沿着调节参数的解路径拟合插补模型。 IM 值用于确定调节参数的最佳值。

5. 使用训练数据集通过消除第 4 步中所选插补模型的结果的偏倚来执行其他秩缩减。
6. 执行最终秩缩减以校准流数据模型，以防过度拟合。这需要将第 5 步中确定的秩用作上限，针对验证集拟合插补模型。

自动数据插补控件

ADI 实用工具包含用于保存插补值的选项和高级控件。

图 21.13 ADI 控件



用于保存插补值的选项 提供以下用于保存 ADI 方法的插补值的三个选项：

新建数据表 创建与原始数据表具有相同维的新数据表。在新数据表中，启动窗口中的选定列包含插补值。

将得分公式保存至当前数据表 将名为**插补_** 的列组保存至包含在启动窗口中指定的插补列的当前数据表。隐藏列 **ADI 插补列**也会添加至包含数据插补中使用的插补向量和得分公式的当前数据表。若向数据表添加任何其他行，列公式会自动更新，从而支持对流数据进行缺失数据插补。这是默认选项。

原位插补值 在当前数据表中插补缺失值。插补值显示为浅蓝色。

高级控件 包含以下高级控件，它们默认为基于数据的推荐设置：

维上限 确定低秩近似中允许的最大秩。由所选列构成的矩阵的维确定。

最多迭代次数 确定要迭代的值的个数，以确定插补模型的调节参数。默认值为 10。

引入为缺失值的观测比例 确定添加到训练集和验证集的 IM 值的比例。每个集的默认比例为 0.2。

用于验证的行比例 确定在训练集和验证集中使用的行的比例。验证集的默认比例为 0.3。

设置随机种子 确定 ADI 的随机种子。使用该选项获取可重现的结果。

“探索缺失值”实用工具选项

ADI 载荷矩阵 （仅适用于 ADI 实用工具。）显示或隐藏包含对应于每个成分的因子载荷的列的报表。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“探索模式”实用工具

数据可能很混乱，因此有时很难仅通过目测来确定数据的有效性。“探索模式”实用工具是用于检测数据中的异常或意外模式的工具。尽管有些模式是偶然出现的，或者是一些无害的机制，但它们可能表明数据被伪造或篡改。若值精确，以致于匹配和模式就不太可能碰巧出现，此时该实用程序最有用。“探索模式”实用工具还设计用于高效处理大型数据表。

“探索模式”实用工具查找重复的值和重复的值序列。这些类型的模式可以指示数据篡改策略，例如多次粘贴相同的值、跨一定范围内的单元格粘贴相同的值，或者复制一定范围内的数据并将其粘贴到其他位置。该实用程序还分析格式化值的属性，以确定数据是实际测量值还是可能由随机数生成器生成。还有一些选项可以检查分布属性并且跨行组检测列之间的线性关系。

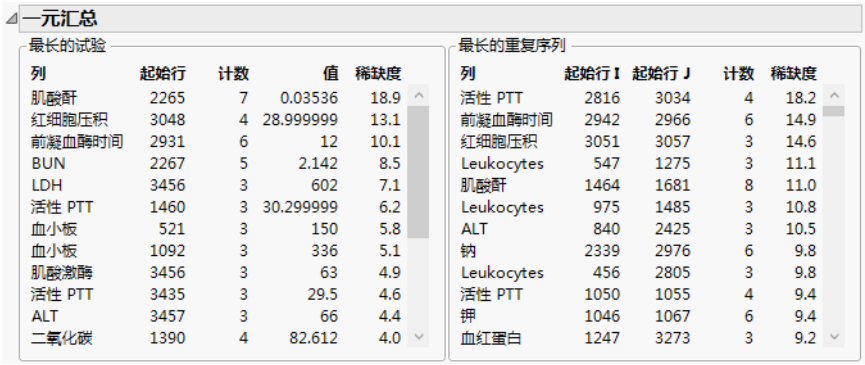
注意：“探索模式”实用工具忽略排除的行，但结果不一定等同于删除了该行。对于行的长度，例如“最长的长度”选项和“最长的重复序列”选项，排除的行将停止运行。

“探索模式”实用工具的示例

Nicardipine Lab Patterns.jmp 样本数据表改编自关于药物尼卡地平的一项研究。该数据表包含 27 列数值型实验室结果。使用 “探索模式”实用程序可确定实验室结果中是否存在任何异常模式。

1. 选择**帮助** > **样本数据库**并打开 Nicardipine Lab Patterns.jmp 样本数据表。
2. 选择**分析** > **筛选** > **探索模式**。
3. 选择 Laboratory Results 列组并点击 **Y，列**。
4. 点击**确定**。
5. 点击 “一元汇总”旁边的灰色展开图标以打开 “一元汇总”报表。

图 21.14 探索模式 “一元汇总”报表



一元汇总

最长的试验

列	起始行	计数	值	稀缺度
肌酸酐	2265	7	0.03536	18.9
红细胞压积	3048	4	28.999999	13.1
前凝血酶时间	2931	6	12	10.1
BUN	2267	5	2.142	8.5
LDH	3456	3	602	7.1
活性 PTT	1460	3	30.299999	6.2
血小板	521	3	150	5.8
血小板	1092	3	336	5.1
肌酸激酶	3456	3	63	4.9
活性 PTT	3435	3	29.5	4.6
ALT	3457	3	66	4.4
二氧化碳	1390	4	82.612	4.0

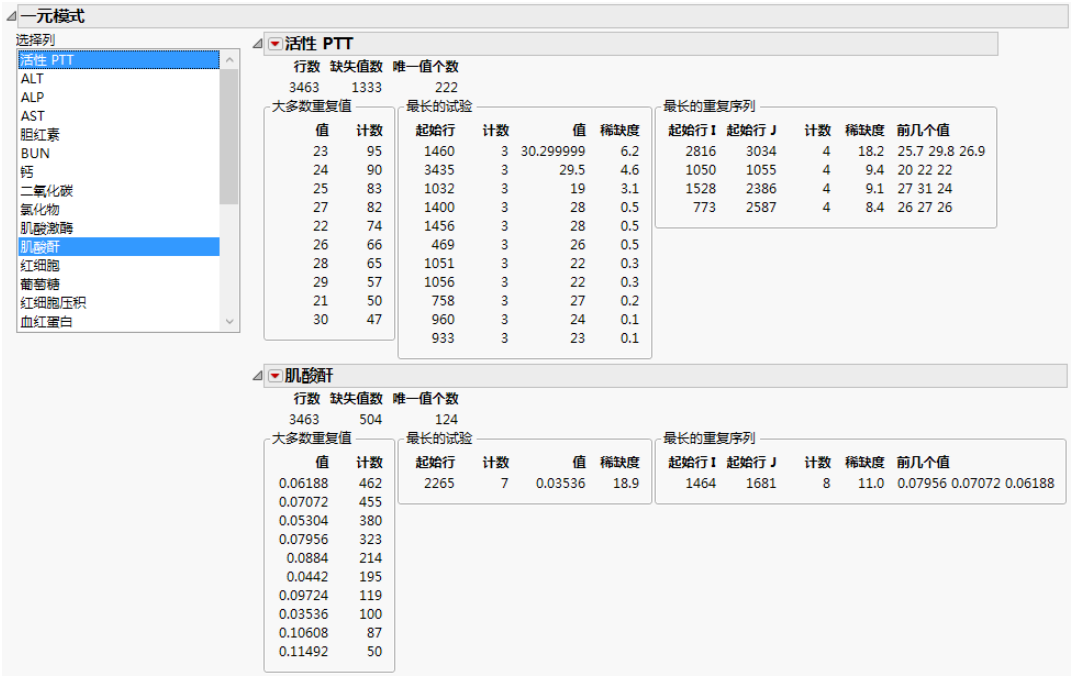
最长的重复序列

列	起始行 I	起始行 J	计数	稀缺度
活性 PTT	2816	3034	4	18.2
前凝血酶时间	2942	2966	6	14.9
红细胞压积	3051	3057	3	14.6
Leukocytes	547	1275	3	11.1
肌酸酐	1464	1681	8	11.0
Leukocytes	975	1485	3	10.8
ALT	840	2425	3	10.5
钠	2339	2976	6	9.8
Leukocytes	456	2805	3	9.8
活性 PTT	1050	1055	4	9.4
钾	1046	1067	6	9.4
血红蛋白	1247	3273	3	9.2

“一元汇总”报表汇总了在所有列中找到的最长的长度和最长的重复序列。每个表都是按“稀缺度”排序，“稀缺度”计算的是长度或序列偶然出现的可能性。针对 “最长的长度”稀缺度得分最高的列是**肌酸酐**，而针对 “最长的重复序列”稀缺度得分最高的列是**活性 PTT**。您将在 “一元模式”报表中对这些列进行进一步检查。

6. 在 “一元模式”报表中，最初显示第一列**活性 PTT** 的报表。按 Ctrl 键并从 “选择列”列表中选择**肌酸酐**。

图 21.15 探索模式 “一元模式”报表



一元模式

选择列

活性 PTT
ALT
ALP
AST
胆红素
BUN
钙
二氧化碳
氧化物
肌酸激酶
肌酸酐
红细胞
葡萄糖
红细胞压积
血红蛋白

活性 PTT

行数	缺失值数	唯一值个数
3463	1333	222

大多数重复值

值	计数
23	95
24	90
25	83
27	82
22	74
26	66
28	65
29	57
21	50
30	47

最长的试验

起始行	计数	值	稀缺度
1460	3	30.299999	6.2
3435	3	29.5	4.6
1032	3	19	3.1
1400	3	28	0.5
1456	3	28	0.5
469	3	26	0.5
1051	3	22	0.3
1056	3	22	0.3
758	3	27	0.2
960	3	24	0.1
933	3	23	0.1

最长的重复序列

起始行 I	起始行 J	计数	稀缺度	前几个值
2816	3034	4	18.2	25.7 29.8 26.9
1050	1055	4	9.4	20 22 22
1528	2386	4	9.1	27 31 24
773	2587	4	8.4	26 27 26

肌酸酐

行数	缺失值数	唯一值个数
3463	504	124

大多数重复值

值	计数
0.06188	462
0.07072	455
0.05304	380
0.07956	323
0.0884	214
0.0442	195
0.09724	119
0.03536	100
0.10608	87
0.11492	50

最长的试验

起始行	计数	值	稀缺度
2265	7	0.03536	18.9

最长的重复序列

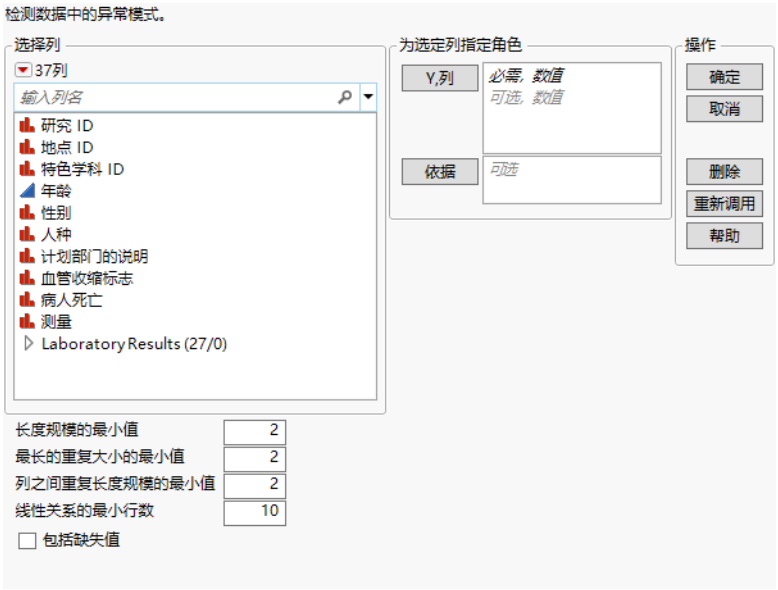
起始行 I	起始行 J	计数	稀缺度	前几个值
1464	1681	8	11.0	0.07956 0.07072 0.06188

从“选择列”列表中选择列名时，将显示相应的列报表。**活性 PTT** 报表显示该列包含若干个长度为 3 的长度和计数为 4 的序列。**肌酸酐**报表显示，虽然它是针对“最长的长度”稀缺度得分最高的一列，但该列中只有一段长度。该列还有一个长度为 8 的序列。

启动“探索模式”实用工具

通过选择**分析 > 筛选 > 探索模式**启动“探索模式”实用工具。

图 21.16 “探索模式”实用工具启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 分配要分析的列。变量必须是数值型和连续型。

依据 分配一个或多个列，这些列的水平用于定义单独的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

长度规模的最小值 为要报告为一段长度的序列指定一行中的最小值数。

最长的重复大小的最小值 指定序列的最小长度，以便将其报告为列中的重复序列。

列之间重复长度规模的最小值 指定序列的最小长度，以便将其报告为列间的重复序列。

线性关系的最小行数 指定用于检测跨行组的列之间的线性关系的最小行数。

注意：最小行数必须大于等于 3。

包括缺失值 指定在模式检测中包含缺失值。

“探索模式”报表

初始“探索模式”报表包含“控制面板”、“一元汇总”报表和“一元模式”报表。这些报表中的许多表都包含可以选定的行。若选定了一行，并且右击第一列，则会出现一个菜单，该菜单支持您选择数据表中的相关行和列或对其着色。

控制面板

控制面板包含支持您选择要在报表窗口中显示的报表的选项。“探索模式”红色小三角菜单中也提供相同的选项。请参见第 381 页的““探索模式”实用工具选项”。控制面板中也有一部分用于指定报表设置。这些设置与启动窗口中的设置相同。请参见第 379 页的“启动“探索模式”实用工具”。若更改控制面板中的设置，则报表将自动更新。

一元汇总

“一元汇总”报表组合来自跨列的若干模式报表的结果。默认情况下，“最长的长度”和“最长的重复序列”都有汇总表。表中显示了稀缺度为 ≥2 的“最长的长度”和“最长的重复序列”。

若选定“分数长度”选项，则同时显示“分数长度”汇总表。该表显示了每列中分数长度大于或等于 15 的值的比例。不包含任何分数长度大于或等于 15 的值的列不会显示在“分数长度”汇总表中。

若 Y 列中至少有一列具有“规格限”列属性，并且选定了“关于规格限的分布”选项，则还会显示“规格限分布”汇总表。该表显示在规格限之外具有统计上显著的观测数的列。

有关“一元汇总”报表内容的详细信息，请参见第 381 页的““探索模式”实用工具选项”。

一元模式

“一元模式”报表包含“选择列”列表以及一个或多个列特定模式报表。所显示的列特定报表由“选择列”列表确定。要检查每个列的报表，请选择第一列，然后依次使用向下箭头键前进到下一列。或者，可以点击该列表中的列名以显示选定列的模式报表。若使用 Ctrl 或 Shift 键选择多个列名，则将显示每个选定列的模式报表。若从“选择列”列表中取消选择一列或多列，则会删除相应列的报表。

每个单独的模式报表都显示一个汇总表，其中包含给定列的行数、缺失观测数和唯一值数。默认情况下，还显示针对“最多重复值”、“最长的长度”和“最长的重复序列”的表。若从控制面板或“探索模式”红色小三角菜单中选择了“格式化的宽度”、“分数长度”或“前导和尾部数位”选项，则会在每个单独的模式报表中添加相应的表。

若在“一元模式”报表中的任何表内的行上右击，将显示一个右击选项菜单。

选择行和列 选择原始数据表中与“一元模式”报表的特定表中的选定行相对应的列和行。例如，若在特定列的“前导数位”表的第二行上右击，则该列中前导数位为 2 的行将在原始数据表中被选定。

着色单元格 为原始数据表中与“一元模式”报表的特定表中的选定行相对应的单元格着色。同时选定已着色单元格的行。

针对列特定的模式报表提供红色小三角选项。请参见第 383 页的““一元模式”报表选项”。

“探索模式”实用工具选项

控制面板和“探索模式”红色小三角菜单包含用于汇总和模式表的选项。

最多重复值 显示或隐藏数据中最常出现的值的表。为每列显示多达十个数字。

最长的长度 显示或隐藏相同值的最长的长度表。该表包含以下信息：

起始行 与试验开始对应的行号。

计数 长度的幅度。

值 长度中条目的值。

稀缺度 用于测量该长度偶然发生的罕见度的测量值。较高的稀缺度表明该长度不可能是偶然发生的。请参见第 385 页的“最长的长度中的稀缺度”。

最长的重复序列 显示或隐藏在同一列中出现多次的值的最长序列表。同时还考虑重复值序列。若序列重复多次，则每次重复都显示在表中。该表包含以下信息：

起始行 I 与第一个序列的开始对应的行号。

起始行 J 与重复序列的开始对应的行号。

计数 序列的长度。

稀缺度 用于测量重复序列偶然发生的罕见度的测量值。较高的稀缺度表明重复序列不可能是偶然发生的。请参见第 385 页的“最长序列中的稀缺度”。

前几个值 重复序列的前几个值。为每个序列显示多达三个值。

格式化的宽度 显示或隐藏格式化值的总宽度和小数宽度。该表包含以下信息：

宽度 宽度大小。

总计数 列中总宽度等于指定宽度的观测数。

小数计数 列中小数宽度等于指定宽度的观测数。

分数长度 显示或隐藏值的连续的分数长度表。连续的分数是将一个数字表示为一系列连续相除项的形式。分数长度可以表示一个数字的有理度。长度为 15 或更多表示该数是无理数，例如从根函数、包含无理数的函数或从随机数生成器生成的数 (Benford, 1938)。

注意：在“探索模式”中，最大分数长度为 15。分数长度大于等于 15 的数字均报告为 15。

长度 连续的分数的长度。

计数 指定的连续的分数长度等于连续的分数长度的列中的观测数。

前导和尾部数位 显示或隐藏每个前导和尾部数位（1 到 9）的计数表。对于前导数位，还提供包含相应前导数位的期望比率以及最小值和最大值的列。期望比率基于本福特定律，该定律指出某个数的第一个数位等于 a 的频率为 $F_a = \log_{10}(a+1/a)$。这条定律适用于数字范围非常大（都有四个或更多数位）的数据。

规格限匹配 （当至少有一列具有“规格限”列属性时可用。）显示或隐藏恰好是规格下限、规格上限或目标值的值表。若无精确匹配，则不显示报表。

关于规格限的分布 （仅当至少有一列具有“规格限”列属性时才可用。）显示或隐藏超出规格限的期望值 - 观测值表。仅当观测到的观测数显著高于预期观测数时，才会出现“卡方”和“p 值”列。

各列之间的重复项 显示或隐藏“各列之间的重复项”报表。该报表包含一个值序列表，这些值出现在跨多个列的同一行中。报表顶部有一个注释，用于说明将一个序列可被视为重复序列时所要求的最小长度。该数字在启动窗口或控制面板中指定为“列之间重复长度规模大小的最小值”。报表顶部也有一个“着色”选项。请参见第 382 页的“着色各列之间的重复项”。

着色各列之间的重复项 （仅在选定“各列之间的重复项”之后才可用。）为数据表中的单元格着色，使其与“各列之间的重复项”报表中的重复匹配项相对应。同时选定包含重复项的行。

线性关系 显示或隐藏变量之间的线性关系报表。该报表包含一个表，该表具有在连续行指定的数目上各列之间的精确线性关系。精确线性关系定义为 R^2 值大于等于 0.999999 的线性关系。报表顶部有一个注释，用于说明所考虑的最小行数。该数字在启动窗口或控制面板中指定为“线性关系的最小行数”。报表顶部也有一个“着色”选项。请参见第 382 页的“着色线性关系”。

着色线性关系 （仅在选定“线性关系”之后才可用。）为数据表中的单元格上色，使其与“变量之间的线性关系”表中的线性关系相对应。同时选定包含线性关系的行。

保存列 包含以下保存选项：

- **保存试验表** 将所有列的最长的长度保存至新数据表。该数据表包含的列与“一元模式”报表中的“最长的长度”表中的列相同，并按“起始行”排序。
- **保存重复序列表** 将所有列的最长重复序列保存至新数据表。该数据表包含的列与“一元模式”报表中的“最长重复序列”表中的列相同，并按“起始行 I”排序。
- **保存各列之间的重复项** 将“各列之间的重复项”表保存至新数据表。
- **保存线性关系** 将值表之间的线性关系保存至新数据表。

排序列 包含用于排序的选项。更改顺序时，“一元模式”报表中的列列表将重新排序。将自动选择顶部列，并且仅显示该列的报表。

- **按试验稀缺度排序** 按最长的长度的稀缺度值对列列表排序。较高的稀缺度值排在顶部。
- **按序列稀缺度排序** 按最长的重复序列的稀缺度值对列列表排序。较高的稀缺度值排在顶部。
- **按列名排序** 按列名的字母顺序对列列表排序。
- **原始顺序** 按在启动窗口中指定列时的最初顺序对列列表排序。

清除单元格颜色 清除数据表中选定列的单元格颜色，若数据表中没有选定任何列，则清除所有列的单元格颜色。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“一元模式”报表选项

每个“一元模式”红色小三角菜单都包含以下选项：

选择最多重复值 选择数据表中与“最多重复值”表中的值相对应的观测。

选择最长的长度 选择数据表中与“最长的长度”表中的值相对应的观测。

选择最长的重复序列 选择数据表中与“最长的重复序列”表中的值相对应的观测。

按格式化的宽度选择 选择数据表中与“格式化的宽度”表中具有用户指定的总宽度的值相对应的观测。

按小数宽度选择 选择数据表中与“格式化的宽度”表中具有用户指定的小数宽度的值相对应的观测。

按分数长度选择 选择数据表中与“分数长度”表中具有用户指定的分数长度的值相对应的观测。

按前导数位选择 选择数据表中与“前导和尾部数位”表中具有用户指定的前导数位的值相对应的观测。

“探索模式”实用工具的更多示例

Water Treatment.jmp 数据表包含来自城市污水处理厂的数据。列中包含处理厂中不同传感器的每日测量值。使用“探索模式”实用程序可确定传感器测量值中是否存在任何异常模式。

1. 选择**帮助** > **样本数据库**并打开 Water Treatment.jmp 样本数据表。
2. 选择**分析** > **筛选** > **探索模式**。
3. 选择 Sensor Measurements 列组并点击 **Y，列**。
4. 点击**确定**。
5. 在“控制面板”中，“各列之间”下方，点击“各列之间的重复项”旁边的框。

图 21.17 “各列之间的重复项”报表

各列之间的重复项

试验至少为 2 着色

自行	至行	数值行	数值相等	重复集
219	230	12	12	PH-P, PH-D
207	217	11	11	PH-E, PH-P
214	217	4	8	PH-E, PH-P, PH-D
330	333	4	8	PH-E, PH-P, PH-D
198	204	7	7	PH-P, PH-D
91	96	6	6	PH-P, PH-D
94	96	3	6	PH-E, PH-P, PH-D
208	210	3	6	PH-E, PH-P, PH-D
216	217	2	6	PH-E, PH-P, PH-D, PH-S
228	229	2	6	PH-E, PH-P, PH-D, PH-S
247	252	6	6	PH-E, PH-P
290	295	6	6	PH-E, PH-P
293	295	3	6	PH-E, PH-P, PH-D
427	429	3	6	PH-E, PH-P, PH-D
472	477	6	6	PH-E, PH-P
26	30	5	5	PH-E, PH-P
63	67	5	5	PH-P, PH-D
141	145	5	5	PH-E, PH-D
299	303	5	5	PH-E, PH-P
352	356	5	5	PH-P, PH-D

“各列之间的重复项”报表显示重复列的长列表。不过，长度规模最小值为 2，所以列表中有许多小的长度。您想要关注规模至少为 10 的大型长度。

6. 在“控制面板”的“设置”下，在“列之间重复长度规模的最小值”旁边的框中输入 10。

提示：点击框外任意位置以自动更新该报表。

图 21.18 长度规模至少为 10 的“各列之间的重复项”报表

各列之间的重复项

试验至少为 10 着色

自行	至行	数值行	数值相等	重复集
219	230	12	12	PH-P, PH-D
207	217	11	11	PH-E, PH-P

更新的“各列之间的重复项”报表显示有两个重复集包含规模至少为 10 的长度。这两个集包括 PH-P 列。

7. 点击“各列之间的重复项”报表中的**着色**。

图 21.19 已着色的 Water Treatments.jmp 数据表

	PH-E	DBO-E	DQO-E	SS-E	SSV-E	SED-E	COND-E	PH-P	DBO-P	SS-P	SSV-P	SED-P	COND-P	PH-D
206	7.7	124.0	234.0	120.0	61.7	1.8	1550.0	7.9	123.0	220.0	59.0	4.0	1651.0	7.8
207	7.6	172.0	394.0	226.0	51.3	5.5	1324.0	7.6	197.0	302.0	47.0	4.0	1306.0	7.7
208	7.6	268.0	324.0	226.0	53.1	4.0	1390.0	7.6	265.0	236.0	53.4	4.5	1389.0	7.6
209	7.6	176.0	200.0	176.0	50.0	3.0	859.0	7.6	149.0	172.0	51.2	2.0	860.0	7.6
210	7.6	146.0	243.0	164.0	57.3	3.2	1095.0	7.6	193.0	208.0	55.8	3.0	1089.0	7.6
211	7.8	139.0	380.0	256.0	53.1	5.5	2040.0	7.8	201.0	340.0	47.6	8.5	2000.0	7.9
212	7.9	205.0	529.0	264.0	61.4	6.5	2750.0	7.9	220.0	324.0	60.5	6.0	2850.0	7.8
213	7.8	117.0	366.0	500.0	27.2	4.2	3230.0	7.8	135.0	620.0	32.3	7.0	3170.0	7.7
214	7.6	148.0	400.0	564.0	31.6	7.0	1554.0	7.6	235.0	798.0	28.6	11.0	1515.0	7.6
215	7.8	158.0	256.0	194.0	47.4	3.0	1831.0	7.8	218.0	240.0	46.7	4.0	1860.0	7.8
216	7.7	123.0	246.0	186.0	39.8	2.5	1288.0	7.7	241.0	300.0	25.3	4.5	1328.0	7.7
217	7.7	133.0	410.0	204.0	62.7	4.5	1490.0	7.7	161.0	256.0	57.8	4.5	1421.0	7.7
218	7.5	237.0	388.0	178.0	57.3	4.0	1620.0	7.6	300.0	240.0	53.3	4.5	1571.0	7.7
219	7.7	203.0	357.0	222.0	58.6	3.0	1828.0	7.6	215.0	208.0	73.1	5.0	1817.0	7.6
220	7.8	199.0	380.0	244.0	55.7	5.0	1453.0	7.8	110.0	196.0	60.2	3.5	1430.0	7.8
221	7.8	214.0	941.0	304.0	53.3	6.5	1454.0	7.8	396.0	320.0	57.5	7.0	1528.0	7.8
222	7.7	208.0	298.0	260.0	37.7	4.0	1236.0	7.8	245.0	284.0	46.5	5.0	1285.0	7.8
223	7.9	206.0	305.0	188.0	56.4	4.0	1200.0	7.9	220.9	272.0	50.0	6.0	1228.0	7.9
224	7.7	247.0	361.0	286.0	51.0	4.5	1180.0	7.7	146.0	212.0	56.4	2.5	1251.0	7.7
225	7.5	438.0	681.0	370.0	56.8	6.0	1396.0	7.6	212.0	348.0	64.4	6.0	1446.0	7.6
226	7.8	255.0	282.0	166.0	67.5	2.0	950.0	7.8	284.0	172.0	65.1	2.0	965.0	7.8
227	7.7	283.0	310.0	170.0	61.2	2.5	1065.0	7.8	235.0	270.0	51.1	5.5	1100.0	7.8
228	7.9	313.0	341.0	512.0	33.6	6.0	915.0	7.9	205.0	520.0	33.8	7.5	976.0	7.9
229	7.7	300.0	610.0	452.0	45.1	8.0	1313.0	7.7	375.0	556.0	43.9	10.0	1312.0	7.7
230	7.8	177.0	412.0	196.0	69.4	4.5	2190.0	7.8	330.0	476.0	52.1	7.5	2330.0	7.8

“着色”功能可对原始数据表中与报表中的重复集相对应的单元格进行着色。

“探索模式”实用工具的统计详细信息

最长的长度中的稀缺度

要计算最长的长度的稀缺度，请首先定义以下变量：

n = 列中的行数

k = 列中出现特定值的次数

$p = k/n$ = 在列中观测到特定值的概率

m = 长度的幅度

N = 唯一长度数

那么，最长的长度的稀缺度计算如下：

$$\text{稀缺度} = -\log_2(1 - (1 - p^{m-1})^N)$$

最长序列中的稀缺度

要计算最长序列的稀缺度，请首先定义以下变量：

p = 在列中观测到一次特定序列的概率

k = 序列的起始值在列中出现的次数

那么，最长序列的稀缺度计算如下：

$$\text{稀缺度} = -\log_2(1 - (1 - p)^k)$$

第 22 章

响应筛选

在大型数据中检验很多响应

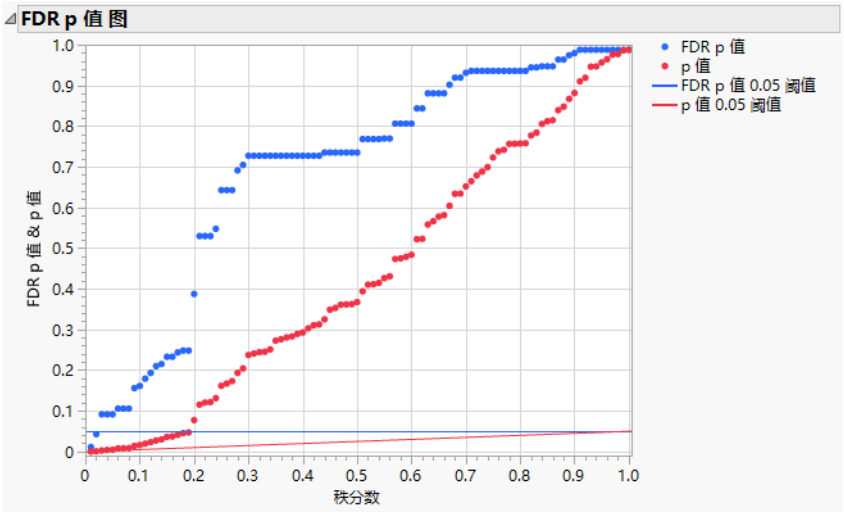
大型数据集的分析涉及对某一零件或某一生物体取数百或数千个测量值，因此分析这样的数据集需要创新性的方法。但是若没有合适的方法，要对多个响应变量进行因子效应检验且保证结果不会产生误导会比较困难。

“响应筛选”自动执行对大量的响应进行检验的过程。为了支持数据探索，您的检验结果和汇总统计量显示在数据表而不是报表中。假发现率方法可以防止对检验的显著性做出错误的判定。*p* 值图的尺度使用 LogWorth 进行了调整，这使得它们易于解释。

因为大型数据集通常很零乱，“响应筛选”提供了解决不规则分布数据和缺失数据的方法。稳健估计方法允许离群值留在数据中，但是降低了检验对这些离群值的敏感度。缺失数据选项允许在分析中包括缺失值。使用这些功能，您无需进行大量的数据质量检查工作就可以直接进行数据分析。

当您有很多观测值时，即使没有实际意义的差值也可能在统计上是显著的。“响应筛选”提供检测实际差值的检验，您在该检验中指定实际要检测的差值。另一方面，您可能想知道差值是否未超过给定的量值，即均值是否等价。为此，“响应筛选”提供了等价性检验。

图 22.1 响应筛选图的示例



目录

“响应筛选”平台概述 389
响应筛选的示例 390
启动“响应筛选”平台 392
“响应筛选”报表 394
结果表 394
FDR p 值图 395
FDR LogWorth 依据 : 效应大小 396
FDR LogWorth 依据 : R 方 397
“p 值”数据表 397
“p 值”数据表脚本 398
“响应筛选”平台选项 398
“均值”数据表 399
“比较均值”数据表 400
“拟合模型”中的“响应筛选”特质 402
在“拟合模型”中启动“响应筛选” 402
“拟合响应筛选”报表 403
“拟合响应筛选”选项 404
“PValues”数据表 404
“Y Fits”数据表 405
响应筛选的更多示例 406
实际显著性检验和等价性检验的示例 406
“LogWorth 最大值”选项的示例 408
稳健拟合的示例 410
“响应筛选”特质 412
“响应筛选”平台的统计详细信息 413
假发现率 413

“响应筛选”平台概述

“响应筛选”自动执行对大量的响应进行检验的过程。它针对您指定的每个因子检验所选的每个响应。响应筛选解决了有关大规模数据的两个主要问题：其一是需要执行很多检验，其二是需要有效地处理离群值和缺失值。

响应筛选有一个单独的平台，同时也存在于拟合模型平台的分析方法特质中。在这两种情况下，它执行与“以 X 拟合 Y”平台中类似的检验，如表 22.1 中所示。 将响应筛选作为特质使用时，它针对各个模型效应执行响应检验。

为了帮助和支持多个统计推断的分析，“响应筛选”提供以下功能：

数据表 结果显示在数据表和报表中，这样您可以探索、排序、搜索您的结果并进行绘图。同时提供便于解释图的统计量，如 *p* 值的 LogWorth。

假发现率 因为您执行的检验很多，需要控制将检验判定为显著的总比率。响应筛选可控制**假发现率**。假发现率 (FDR) 是显著检验被错误地声明为显著的期望比例 （Benjamini and Hochberg 1995； Westfall et al. 2011）。

实际显著性的检验 当您有很多观测值时，即使没有实际意义的小效应也可能导致统计显著性。要解决该问题，您可以定义一个视为具有**实际显著性**的效应大小。然后便可执行实际显著性的检验，因此只检测足够大到有实际影响的效应。

等价性检验 研究很多因子时，您通常关注那些对响应具有基本同等影响的因子。在这种情况下，您可以指定达到实际等价性时的效应大小，然后执行等价性检验。

为了解决处理零乱数据引起的问题，“响应筛选”提供了处理离群值和缺失数据的功能。使用这些功能，您可以直接分析数据，而不必先解决数据质量问题：

稳健估计 您数据中的离群值会增大标准误差的估计值，导致检验对真实效应不敏感。选择“稳健”选项来执行 Huber M 估计，这样离群值仍在数据中，但是可降低检验对这些离群值的敏感度。

缺失值选项 平台包含一个选项来以信息性方式处理分类预测变量的缺失值。

表 22.1 响应筛选执行的分析

响应	因子	“以 X 拟合 Y”分析	说明
连续	分类	单因子	方差分析
连续	连续	二元	简单线性回归
分类	分类	列联	卡方
分类	连续	Logistic	简单 Logistic 回归

“响应筛选”平台生成一个报表和一个数据表：“响应筛选”报表和“P 值”表。拟合模型中的“响应筛选”特质生成一个报表和两个数据表：“拟合响应筛选”报表、“P 值”表和“Y 拟合”表。

JSL 命令 `Summarize YbyX` 执行与“响应筛选”平台相同的功能，但是不创建平台窗口。请参见《JSL Syntax Reference》。

响应筛选的示例

Probe.jmp 样本数据表包含对 5800 个晶片测量的 387 个特征（**响应**列组）。**批 ID** 和**晶片编号**列唯一标识晶片。您关注在流程更改 （**流程**）时哪些特征会显示不同的值。

1. 选择**帮助 > 样本数据库**，然后打开 Probe.jmp。
2. 选择**分析 > 筛选 > 响应筛选**。
3. 选择 Responses 列组并点击 **Y，响应**。
4. 选择**流程**并点击 **X**。
5. 在 LogWorth **最大值**框中输入 100。

   logworth ($-\log_{10}(p\text{-value})$) 为 100 或更大值对应于极小的 p 值。设置 “LogWorth 最大值”的值有助于控制这些图的尺度，因为它可以限制报告的 Logworth 值。
6. 点击**确定**。

   该报表显示 p 值结果表和 FDR p 值图，还包含另外两个图报表。

“FDR p 值图”对于 387 个检验的每个检验显示两种 p 值：“FDR p 值”和“p 值”。这两种值是针对秩分数标绘的。“p 值”是**流程**-Y 检验的常规 p 值。“FDR p 值”是经过调整的 p 值，以确保伪发现率 (FDR) 不超过指定值（本例为 0.05）。FDR p 值用蓝色标绘，p 值用红色标绘。“秩分数”将 FDR p 值按从小到大（显著性降低）的顺序排名。

图上的水平蓝线和倾斜的红线是 FDR 显著性的阈值。针对假发现率进行调整后，FDR p 值位于蓝线下的检验在 0.05 水平下是显著的。位于红线下的普通 p 值对应的检验在 0.05 水平下是显著的。这样，在图中可以通过任意一组 p 值了解 FDR 显著性。

图 22.2 针对流程执行的 387 个检验的 “响应筛选”报表

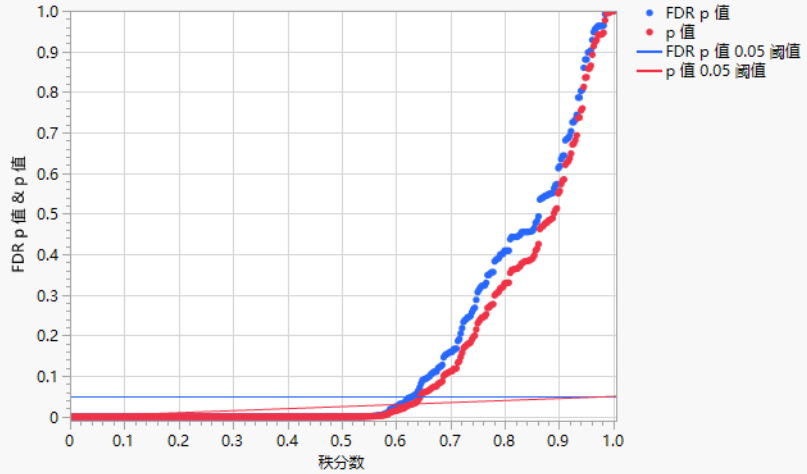
响应筛选

LogWorths 限制为 100 以便于图形统一尺度

结果表

Y	X	计数	p 值	LogWorth	FDR p 值	FDR LogWorth	效应大小	秩分数	R 方
DELW_M2	流程	5728	4e-322	100.000	0	100.000	0.47114	0.09115	0.2297
DELW_NENBNI	流程	5792	4e-322	100.000	0	100.000	0.48812	0.08594	0.2383
DELW_NSINK	流程	5782	4e-322	100.000	0	100.000	0.58572	0.07031	0.3431
DELW_PCOLL	流程	5785	4e-322	100.000	0	100.000	0.69477	0.00521	0.8866
NISO_TUB-TRENCH_VGATERTFF	流程	5800	2e-164	100.000	1.4e-163	100.000	0.34772	0.11458	0.1209
R_RPB_100X20	流程	5784	4e-128	100.000	3.3e-127	100.000	0.30889	0.1224	0.0954
R_TRENCH	流程	5800	4e-322	100.000	0	100.000	0.67328	0.0026	0.9748
VDP_NEMIT	流程	5788	2e-202	100.000	2.2e-201	100.000	0.36968	0.10417	0.1472
VDP_NENBNI	流程	5792	4e-322	100.000	0	100.000	0.8138	0.02344	0.6624
VDP_NSINK	流程	5787	4e-322	100.000	0	100.000	0.66372	0.02604	0.5487
VDP_PBASE	流程	5791	1e-251	100.000	1e-250	100.000	0.40048	0.09635	0.1800
VDP_PBL	流程	5793	4e-322	100.000	0	100.000	0.98523	0.07292	0.3365
VDP_PCOLL	流程	5800	4e-322	100.000	0	100.000	0.6806	0.01823	0.6950
30D1_7X100_VDON1U	流程	5794	8e-124	100.000	6.1e-123	100.000	0.68971	0.125	0.0922
30N1_12X100_BVCEO	流程	5800	4e-322	100.000	0	100.000	0.69387	0.03646	0.4815
30N1_12X100_HFE1UA	流程	5771	4e-322	100.000	0	100.000	0.59561	0.0599	0.3766
30N1_12X100_HFEPEAK*VA	流程	5795	4e-322	100.000	0	100.000	0.6609	0.08333	0.2698
30N1_12X100_ICHFEPEAK	流程	5764	4e-322	100.000	0	100.000	0.5898	0.03385	0.5048
30N1_12X100_ICHFEQ	流程	5711	3e-267	100.000	3.1e-266	100.000	0.50793	0.09375	0.1924
30N1_12X100_VA1.E-5	流程	5770	4e-246	100.000	4.3e-245	100.000	0.46474	0.09896	0.1769

FDR p 值图



“FDR p 值图”显示 60% 以上的检验是显著的。少量检验从未调节 *p* 值来看是显著的，但是从 FDR *p* 值来看它们不是显著的。这些检验对应于在红线上方和蓝线下方的红点。

为了识别在不同**流程**之间存在显著差异的特征，您可以在图中合适的点周围拖曳出一个矩形。这样会选中“结果”表中对应于这些点的行，表中的第一列给出了特征的名称。

“结果”表包含 387 行，每一行分别对应于**响应**组中的每个响应测度。响应名称在名为 Y 的第一列中给出。每个响应都针对 X 列（即**流程**）中的效应进行检验。

其余列给出有关 X-Y 检验的信息。在此处检验是单因子方差分析。除了其他信息外，该表还给出检验的 *p* 值、LogWorth、FDR（假发现率）*p* 值和 FDR LogWorth。使用该表可按各种统计量排序、选择行或针对感兴趣的变量进行作图。

请注意，对应于 *p* 值 1e-100 或更小的 LogWorth 和 FDR LogWorth 值报告为 100，因为您在启动窗口中将“LogWorth 最大值”设置为 100。此外，对应于大于 2 的 FDR LogWorth 值的单元格采用强度梯度着色。

有关报表和 PValues 表的详细信息，请参见第 394 页的“ “响应筛选”报表”。

启动“响应筛选”平台

通过选择**分析** > **筛选** > **响应筛选**来启动 “响应筛选”平台。

图 22.3 “响应筛选”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

启动窗口角色

Y，响应 标识包含要分析的测量值的响应列。

X 标识要针对其检验响应的列。

分组 对于指定列的每个水平，分别分析相应的行，但是在单个表和报表中显示结果。

权重 标识其值用于为每行分配权重的列。这些值在分析中作为权重使用。请参见 《拟合线性模型》。

频数 标识其值用于为每个行分配频数的列。这些值允许您将预先汇总数据考虑在内。请参见 《拟合线性模型》。

依据 对于指定列的每个水平，分析相应的 Y 和 X 并在单独的表和报表中显示结果。

启动窗口选项

启动时的 p 值表 创建包含 *p* 值和单值模型拟合统计量的数据表。

稳健 对于连续响应，使用 Huber M 估计值 (Huber and Ronchetti 2009) 拟合回归和方差分析模型。该稳健方法降低了离群值的权重，以降低检验对离群值的敏感性。若无离群值，这些估计值接近最小二乘估计值。请注意该选项会增加处理时间。有关 Huber M 估计的详细信息，请参见 《基本分析》。有关示例，请参见第 410 页的 “稳健拟合的示例”。

Cauchy 假定误差服从 Cauchy 分布。 Cauchy 分布具有比正态分布更肥大的尾部，从而弱化了对离群值的强调。若您的数据中有较大比例的离群值，该选项会很有用。不过，若您的数据接近正态，只含有少数离群值，该选项会导致不正确的推断。 Cauchy 选项使用最大似然和 Cauchy 连结函数估计参数。

Poisson Y 将每个 Y 响应拟合为具有 Poisson 分布的计数。仅为分类 X 执行该检验。当您的响应是计数时，适合使用该选项。

缺失值为类别 对于任何分类 X 变量，将 X 上的缺失值视为一个类别。

配对 X 和 Y 根据 X 列和 Y 列在 **Y，响应**和 **X** 列表中的顺序，仅对与 X 列配对的 Y 列执行检验。第一个 Y 与第一个 X 配对，第二个 Y 与第二个 X 配对等。

实际差值部分 规格范围的比例或估计的六个标准差范围的比例，它表示您认为有实际意义的差值。若 “规格限”未设置为列属性，则为响应估计六个标准差的范围。标准差估计值从四分位间距 (IQR) 计算得出，即 $\hat{\sigma} = (IQR)/(1.3489795)$。

若未指定 “实际差值比例”，则它的值默认为 0.10。实际显著性检验和等价性检验使用该差值确定实际差值。请参见第 400 页的 ““比较均值”数据表”。

LogWorth 最大值 用于控制涉及 LogWorth 值（*p* 值的 $-\log_{10}$）的图的尺度。将超过“LogWorth 最大值”的 LogWorth 值标绘为 “LogWorth 最大值”以防止 LogWorth 图中出现极端尺度。相关示例，请参见第 408 页的 ““LogWorth 最大值”选项的示例”。

高级选项 包含以下选项：

Kappa 将名为 Kappa 的新列添加到数据表。若 Y 和 X 都是分类变量且具有相同的水平，则提供 kappa。这是 Y 和 X 之间的一致性测度。

相关性 “相关性”选项根据值排序定义的索引计算 Pearson 积矩相关性。

若 X 和 Y 都是二值型，Pearson 积矩计算将给出 Spearman rho 和 Kendall Tau-b。否则，高量值的相关性值指示关联；低量值的相关性值不排除关联。

强制 X 为分类变量 忽略建模类型并将所有 X 列视为分类变量。

强制 X 为连续变量 忽略建模类型并将所有 X 列视为连续变量。

强制 Y 为分类变量 忽略建模类型并将所有 Y 列视为分类变量。

强制 Y 为连续变量 忽略建模类型并将所有 Y 列视为连续变量。

非线程 禁止对计算速度使用多线程处理。

“响应筛选”报表

“响应筛选”报表包含一个结果表和若干个图。这些图侧重于假发现率 (FDR) 统计量。请参见第 413 页的“假发现率”。

默认图为“FDR p 值图”、“FDR LogWorth 依据: 效应大小”图和“FDR LogWorth 依据: R 方”图。若您在启动窗口中选择了“稳健”选项，还显示这些报表的稳健版本。此外还会提供“稳健 LogWorth 依据: FDR LogWorth”图来帮助评估使用稳健拟合的影响。

结果表

“响应筛选结果”表对于每对 Y 和 X 变量都包含一行。该表的列包含特定于所选拟合以及 Y 和 X 建模类型的测度和模型拟合统计量。

分组 （仅当有分组变量时才显示。）分组列的水平。

Y 指定的响应列。

X 指定的因子列。

计数 用于检验的行数，或相应的“频数”或“权重”变量之和。

p 值 对应于每对 Y 和 X 变量的显著性检验的 *p* 值。有关“以 X 拟合 Y”统计量的详细信息，请参见《基本分析》。

LogWorth 数量 $-\log_{10}(p \text{ 值})$。该变换调整 *p* 值以提供适用于绘图的尺度。超过 2 的值在 0.01 水平下显著 ($-\log_{10}(0.01) = 2$)。

FDR p 值 使用 Benjamini-Hochberg 方法计算的假发现率 *p* 值。该方法调整 *p* 值以控制多重检验的假发现率。若没有“分组”变量，则多重检验集包含表中显示的所有检验。若有“分组”变量，则多重检验集包含为“分组”变量的每个水平执行的所有检验。有关 FDR 校正的详细信息，请参见 Benjamini and Hochberg (1995)。有关假发现率的详细信息，请参见第 413 页的“假发现率”。

FDR LogWorth 数量 -$\log_{10}$(FDR p 值)。这是用于绘制和评估显著性的统计量。请注意，小的 *p* 值将生成高的 FDR LogWorth 值。对应于 FDR LogWorth 值大于 2（*p* 值小于 0.01）的单元格用强度梯度着色。

效应大小 表示响应值在 X 的不同水平或值之间的差异程度。效应大小的尺度是不变的。

- 当 Y 为连续变量时，效应等于来自假设检验的平均离差平方和的平方根除以响应标准差的稳健估计值。若四分位间距 (IQR) 非零且 IQR > 极差/20，则标准差估计值为 IQR/1.3489795。否则使用样本标准差。
- 当 Y 是分类变量且 X 是连续变量时，效应大小是整体模型检验的平均卡方值的平方根。
- 当 Y 和 X 都是分类变量时，效应大小是平均 Pearson 卡方的平方根。

秩分数 表示为检验数的分数形式的 FDR LogWorth 的秩。若检验数为 *m*，则最大 FDR LogWorth 值具有秩分数 1/*m*，最小 FDR LogWorth 值则具有秩分数 1。“秩分数”用于以显著性下降的顺序对 p 值和 FDR p 值绘图。

R 方 （仅当 Y 为连续变量时才显示。）决定系数，它测量模型解释了多大比例的总变异。

Kappa （仅当 Y 和 X 都是分类变量并且具有相同的水平数时才显示。）这是 Y 和 X 之间的一致性测度。

相关性 （仅当 Y 和 X 均为分类变量时才显示。）根据值排序定义的索引计算的 Pearson 积矩相关性。

当您在启动窗口中选择“稳健”选项时，以下列会添加到“结果”表。仅当 Y 为连续变量时“稳健”选项才可用。因此，当 Y 为分类变量时“稳健”列单元格为空。

稳健 p 值 对应于使用稳健的每对 Y 和 X 变量的显著性检验的 *p* 值。

稳健 LogWorth 数量 -$\log_{10}$(稳健 p 值)。

稳健 FDR p 值 使用 Benjamini-Hochberg 方法为稳健 p 值计算的假发现率。若没有 “分组”变量，则多重检验调整应用于表中显示的所有检验。若有“分组”变量，则多重检验调整应用于为“分组”变量的每个水平执行的所有检验。

稳健 FDR LogWorth 数量 -$\log_{10}$(稳健 FDR p 值)。

稳健秩分数 表示为检验数的分数形式的稳健 FDR LogWorth 的秩。

稳健卡方 与稳健检验关联的卡方值。

稳健 Sigma 误差标准差的稳健估计值。

稳健离群值部分 与稳健均值的距离超过稳健 Sigma 三倍的那部分值。

FDR p 值图

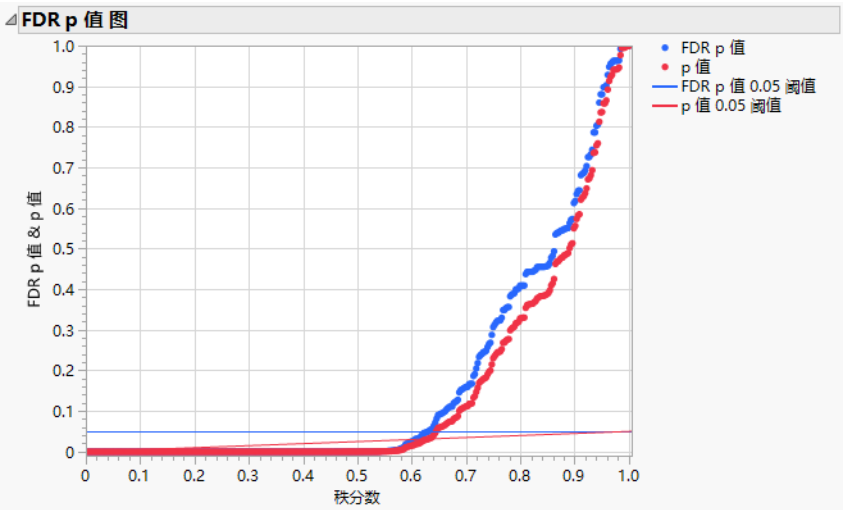
“FDR p 值图”报表显示针对秩分数的 FDR p 值和 p 值的图。秩分数按显著性降低的顺序对 p 值排名。FDR p 值用蓝色标绘，p 值用红色标绘。

蓝色水平线显示 0.05 显著性水平。请注意您可以通过以下方式更改该水平：双击垂直轴、删除当前参考线，然后添加新的参考线。

红色递增线提供未调整的 *p* 值的 FDR 阈值。当 FDR 调整的 *p* 值位于蓝线下方时，*p* 值恰好位于红线下方。这使您可以从调整或未调整的 *p* 值了解 FDR 的显著性。

图 22.4 显示了 Probe.jmp 样本数据表的“FDR p 值图”。请注意，根据常规 *p* 值，一些检验是显著的，但是根据 FDR *p* 值它们则不显著。

图 22.4 FDR p 值图



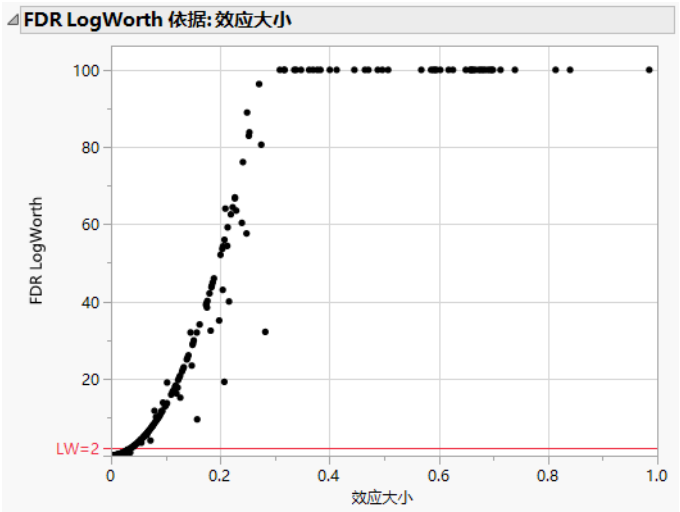
FDR LogWorth 依据 : 效应大小

当您有大的效应时，相关的 *p* 值通常很小。以图形方式可视化这些小的值会很困难。若转换为 LogWorth (-$\log_{10}$(*p* 值)) 尺度，高度显著的 *p* 值具有大的 LogWorth，而不显著的 *p* 值具有小的 LogWorth。LogWorth 为零对应于不显著的 *p* 值 1。任何 2 以上的 LogWorth 值对应于小于 0.01 的 *p* 值。

在“FDR LogWorth 依据: 效应大小”图中，垂直轴为“FDR LogWorth”，水平轴为“效应大小”。通常情况下，效应越大导致 *p* 值越显著，LogWorth 值越大。但是，该关系不一定总是成立，因为显著性也依赖于误差方差。实际上，由于误差方差的大小不同，大的 LogWorth 可能与小效应关联，小的 LogWorth 可能与大效应关联。使用“FDR LogWorth 依据: 效应大小”图可以探索该关系。

图 22.5 显示 Probe.jmp 样本数据表（“LogWorth 最大值”设置为 100）的“FDR LogWorth 依据: 效应大小”图。大多数 FDR LogWorth 值超过 2，这指示大多数效应在 0.01 水平上是显著的。FDR LogWorth 值 100 对应于极小的 *p* 值。

图 22.5 FDR LogWorth 依据 : 效应大小



FDR LogWorth 依据 : R 方

“FDR LogWorth 依据: R 方”图在垂直轴上显示“FDR LogWorth”，在水平轴上显示 R 方值。较大的 LogWorth 值趋向于与较大的 R 方值关联，但是该关系也依赖于观测数。

“p 值”数据表

若在启动窗口中选定“启动时的 p 值表”选项，则会在启动“响应筛选”平台时创建“p 值”数据表。该数据表包含与“响应筛选”报表中的“结果”表相同的信息。“p 值”数据表还包含其他模型特定列：

Y 均值 Y 的均值。

误差平方和 当 Y 为连续变量时显示。误差的平方和。

误差自由度 当 Y 为连续变量时显示。误差的自由度。

均方误差 当 Y 为连续变量时显示。均方误差。

F 比 当 Y 为连续变量时显示。方差分析或回归检验的 F 比。

截距 当 Y 和 X 均为连续变量时显示。将相应的 Y 和 X 变量对关联起来的回归模型的截距。

斜率 当 Y 和 X 均为连续变量时显示。将相应的 Y 和 X 变量对关联起来的回归模型的斜率。

自由度 当 Y 和 X 均为分类变量时显示。卡方检验的自由度。

LR 卡方 当 Y 和 X 均为分类变量时显示。似然比卡方统计量的值。

稳健耗用时间 （仅当在启动窗口中指定了 “稳健”时才显示。）创建 “稳健”报表所需的计算时间 （秒）。

“p 值”数据表脚本

相关的脚本会保存到“p 值”数据表。所有这些脚本（其中一个除外）都可以重现报表中提供的图。当您在“p 值”表中选择行时，“拟合选定项”脚本会生成合适的“以 X 拟合 Y”分析。

“响应筛选”平台选项

“响应筛选”红色小三角菜单包含一些选项，可以定制显示内容以及计算并保存计算得到的数据。

结果表 显示或隐藏结果表。

拟合选定项 对于选定的关系，将合适的 “以 X 拟合 Y”报表添加到 “响应筛选”报表。您可以通过选择图中的点、“结果”表中的行或“p 值”数据表（若已打开）中的行来选择关系。

选择列 选择原始数据表中的列，这些列对应于您在 “结果”表中选择的行、在 “响应筛选”报表窗口的图中选择的点、或 “p 值”表 （若已打开）中的行。首先选择行或点，然后再选择 “选择列”。数据表中相应的列即被选定。若您想选择对应于 “p 值”表中的其他行或图中的其他点的列，则可以先选择这些行或点，然后再选择 “选择列”。要选择对应于不同的行或点的列，则需要先在原始数据表中清除当前的列选择。

选择符合条件的行 打开 “选择符合条件的行”窗口。您可以通过使用 “比较”菜单和 “值”文本框在“结果”表中选择对应特定条件的特定响应。例如，您可以选择“效应大小 > 0.80”的所有响应。点击 “确定”后，即在 “结果”表中选定这些响应。

提示：还可以通过右击 “结果”表中的任何位置访问 “选择符合条件的行”窗口。

保存 p 值 创建 “p 值”数据表。请参见第 397 页的 “‘p 值’数据表”。

保存均值 对于连续变量 Y 和分类变量 X，创建一个数据表，其中包含针对分类变量每个水平的计数、均值和标准差。若选择了 “稳健”选项，还会包含稳健均值。

保存比较均值 对于连续变量 Y 和分类变量 X，检验不同分类变量水平之间的所有配对比较。对于每个比较，数据表给出常规 *t* 检验、实际显著性检验、等价性检验和使用颜色编码汇总结果的列。该数据表还包含一个脚本，用于绘制 “相对实际差值 - 实际 LogWorth”图。请参见第 400 页的 “‘比较均值’数据表”。有关示例，请参见第 406 页的 “实际显著性检验和等价性检验的示例”。

保存标准残差 将一组新列保存到原始数据表并将它们放置到名为 “残差组”的列组。该选项会为每个连续变量 Y 和分类变量 X 构造一列，该列包含残差除以估计的标准差所得的值。换句话说，该列包含标准化残差。该列由公式定义。

若选择了 “稳健”选项，则使用稳健拟合和稳健估计值构造标准化残差列。

保存离群值指示符 将一组新列保存到原始数据表并将它们放置到名为“离群值组”的列组。当您选择了“稳健”选项时，“保存离群值指示符”最为有效。

对于每个连续变量 Y 和分类变量 X，该选项会构造一列来指示离群值。离群值是一个点，该点与预测值的距离超过了 sigma 估计值的三倍。换句话说，离群值是标准化残差超过 3 的点。该列由公式定义。

若选择了“稳健”选项，则使用稳健拟合和稳健估计值。离群值是一个点，该点与预测值的距离超过了 sigma 稳健估计值的三倍。

“聚类离群值”脚本会添加到原始数据表。该脚本将离群值通过数据的层次聚类图显示。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“均值”数据表

“均值”数据表中的每一行对应于响应和每个 X 水平的组合。对于 Probe.jmp 样本数据表，有 387 个响应变量，每个变量针对“流程”的两个水平进行检验。“均值”表包含 387x2 = 774 行。

图 22.6 “均值”数据表

Save Means

列 (6/0)

Y
X
水平
计数
均值
标准差

行

所有行	774
已选定	0
已排除	0
已隐藏	0
已添加标签	0

	Y	X	水平	计数	均值	标准差
1	DELL_RPNBR	流程	New	3044	0.2035840106	0.1916031302
2	DELL_RPNBR	流程	Old	2750	0.2346329436	0.1278299414
3	DELL_RPPBR	流程	New	3044	-0.068072506	0.1784161107
4	DELL_RPPBR	流程	Old	2750	-0.043321135	1.9165048167
5	DELW_M1	流程	New	3039	-0.04387197	1.0774346893
6	DELW_M1	流程	Old	2750	-0.071400861	0.0389764703
7	DELW_M2	流程	New	3039	0.8014275806	0.0698264259
8	DELW_M2	流程	Old	2689	0.7274958056	0.0649383327
9	DELW_NBASE	流程	New	3028	0.1957655669	77.801881218
10	DELW_NBASE	流程	Old	2750	1.4765185729	0.1326787074
11	DELW_NEMIT	流程	New	3038	0.3411755212	0.0940762694
12	DELW_NEMIT	流程	Old	2750	0.3813085954	0.0976561441
13	DELW_NENBNI	流程	New	3043	3.7172025841	1.1419433505
14	DELW_NENBNI	流程	Old	2749	4.6556238745	0.1893626325
15	DELW_NSINK	流程	New	3032	9.2876152816	1.2840573691
16	DELW_NSINK	流程	Old	2750	7.8902297629	0.3781489659
17	DELW_PBASE	流程	New	3036	1.1651478779	0.1351193171
18	DELW_PBASE	流程	Old	2750	1.2087969753	0.1486923744
19	DELW_PCOLL	流程	New	3035	-0.572546047	0.2173568201

“均值”数据表包含以下列：

Y 连续响应变量。

X 分类变量。

水平 分类 X 变量的水平。

计数 相应水平上值的计数。

均值 针对指定水平的 Y 变量的均值。

标准差 针对指定水平的 Y 变量的标准差。

稳健均值 均值的稳健 M 估计值。当您在启动窗口中选择 “稳健”选项时显示。

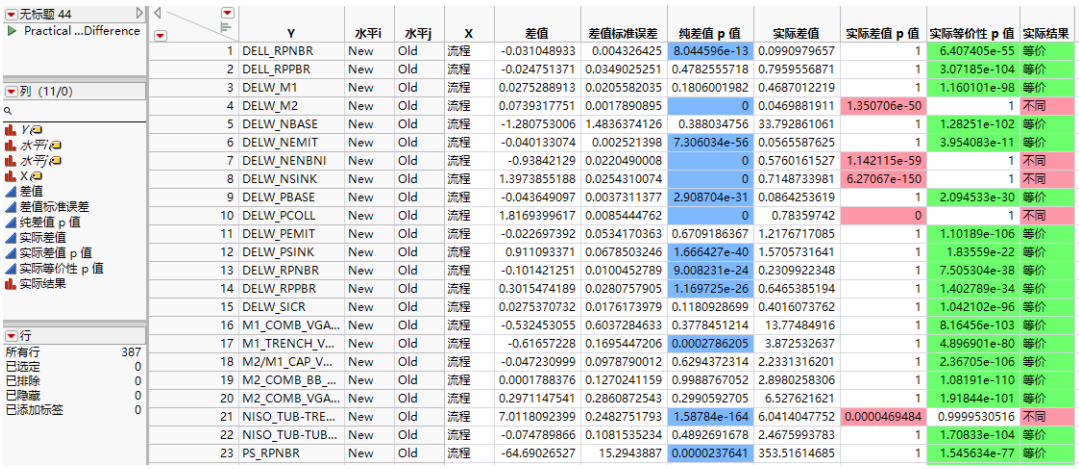
“比较均值”数据表

当您的数据表包含很多行（*n* 很大）时，检验中使用的标准误差会很小。因此，检验可能在统计上是显著的，但实际上观测到的差值太小，没有实际意义。实际显著性检验允许您指定您认为的值得检测的差值大小。该差值称为**实际差值**。这样您检验的是差值是否超过实际差值而不是检验差值为零。因此，检验更有意义，需要仔细检查的检验也会更少。

等价性检验允许您从实际的角度出发判定两个水平对响应的影响是否基本相同。换句话说，等价性检验用于检验差值是否小于实际差值。

“比较均值”数据表提供实际差值检验和实际等价性检验的结果。每一行针对分类因子的两个水平进行响应的比较。配对比较的结果会用颜色编码以便于解释。有关如何指定实际差值的说明，请参见第 401 页的“实际差值”。相关示例，请参见第 406 页的“实际显著性检验和等价性检验的示例”。

图 22.7 “比较均值”数据表


无标题 44
Practical ...Difference
列 (11/0)
Y
水平i
水平j
X
差值
差值标准误差
纯差值 p 值
实际差值
实际差值 p 值
实际等价性 p 值
实际结果
行
所有行 387
已选定 0
已排除 0
已隐藏 0
已添加标签 0

	Y	水平i	水平j	X	差值	差值标准误差	纯差值 p 值	实际差值	实际差值 p 值	实际等价性 p 值	实际结果
1	DELL_RPNBR	New	Old	流程	-0.031048933	0.004326425	8.044596e-13	0.0990979657	1	6.407405e-55	等价
2	DELL_RPPBR	New	Old	流程	-0.024751371	0.0349025251	0.4782555718	0.7959556871	1	3.07185e-104	等价
3	DELW_M1	New	Old	流程	0.0275288913	0.0205582035	0.1806001982	0.4687012219	1	1.160101e-98	等价
4	DELW_M2	New	Old	流程	0.0739317751	0.0017890895	0	0.0469881911	1.350706e-50	1	不同
5	DELW_NBASE	New	Old	流程	-1.280753006	1.4836374126	0.388034756	33.792861061	1	1.28251e-102	等价
6	DELW_NEMIT	New	Old	流程	-0.040133074	0.002521398	7.306034e-56	0.0565587625	1	3.954083e-11	等价
7	DELW_NENBNI	New	Old	流程	-0.93842129	0.0220490008	0	0.5760161527	1.142115e-59	1	不同
8	DELW_NSINK	New	Old	流程	1.3973855188	0.0254310074	0	0.7148733981	6.27067e-150	1	不同
9	DELW_PBASE	New	Old	流程	-0.043649097	0.0037311377	2.908704e-31	0.0864253619	1	2.094533e-30	等价
10	DELW_PCOLL	New	Old	流程	1.8169399617	0.0085444762	0	0.78359742	0	1	不同
11	DELW_PEMIT	New	Old	流程	-0.022697392	0.0534170363	0.6709186367	1.2176717085	1	1.10189e-106	等价
12	DELW_PSINK	New	Old	流程	0.911093371	0.0678503246	1.666427e-40	1.5705731641	1	1.83559e-22	等价
13	DELW_RPNBR	New	Old	流程	-0.101421251	0.0100452789	9.008231e-24	0.2309922348	1	7.505304e-38	等价
14	DELW_RPPBR	New	Old	流程	0.3015474189	0.0280757905	1.169725e-26	0.6465385194	1	1.402789e-34	等价
15	DELW_SICR	New	Old	流程	0.0275370732	0.0176173979	0.1180928699	0.4016073762	1	1.042102e-96	等价
16	M1_COMB_VGA...	New	Old	流程	-0.532453055	0.6037284633	0.3778451214	13.77484916	1	8.16456e-103	等价
17	M1_TRENCH_V...	New	Old	流程	-0.61657228	0.1695447206	0.0002786205	3.872532637	1	4.896901e-80	等价
18	M2/M1_CAP_V...	New	Old	流程	-0.047230999	0.0978790012	0.6294372314	2.2331316201	1	2.36705e-106	等价
19	M2_COMB_BB_...	New	Old	流程	0.0001788376	0.1270241159	0.9988767052	2.8980258306	1	1.08191e-110	等价
20	M2_COMB_VGA...	New	Old	流程	0.2971147541	0.2860872543	0.2990592705	6.527621621	1	1.91844e-101	等价
21	NISO_TUB-TRE...	New	Old	流程	7.0118092399	0.2482751793	1.58784e-164	6.0414047752	0.0000469484	0.9999530516	不同
22	NISO_TUB-TUB...	New	Old	流程	-0.074789866	0.1081535234	0.4892691678	2.4675993783	1	1.70833e-104	等价
23	PS_RPNBR	New	Old	流程	-64.69026527	15.2943887	0.0000237641	353.51614685	1	1.545634e-77	等价

“比较均值”数据表包含一个绘制“Practical LogWorth by Relative Practical Difference”图的脚本。相对实际差值定义为真实差值除以实际差值。

Y 连续响应变量。

X 分类变量。

水平 i 分类 X 变量的水平。

水平 j 要与水平 i 比较的分类 X 变量的水平。

差值 两个水平上均值的差值估计。若选择了“稳健”选项，则使用均值的稳健估计值。

差值标准误差 均值差值的标准误差。若选择了“稳健”选项，则这是稳健估计值。

纯差值 p 值 配对比较的常规 Student *t* 检验的 *p* 值。若选择了“稳健”选项，则这是 *t* 检验的稳健版本。在 0.05 水平下显著的检验会突出显示。

实际差值 视为有实际意义的均值差值。若您将“规格限”属性分配给 Y 变量，则实际差值计算为规格限差值乘以实际差值比例。若未指定实际差值比例，则实际差值为规格限差值乘以 0.10。

若您未将“规格限”属性分配给 Y 变量，则从四分位间距 (IQR) 计算其标准差的估计值。该估计值是 $\hat{\sigma} = (IQR)/(1.3489795)$。实际差值计算为 $6\hat{\sigma}$ 乘以实际差值比例。若未指定实际差值比例，则实际差值计算为 $6\hat{\sigma}$ 乘以 0.10。

实际差值 p 值 一个检验的 *p* 值，该检验用于检测水平 i 和水平 j 之间 Y 的差值均值的绝对值是否大于等于实际差值。小 *p* 值表示绝对差值超过了实际差值。这表示水平 i 和水平 j 的差值具有实际意义。

实际等价性 p 值 使用双单侧检验 (TOST) 方法来检验均值之间的实际差值 (Schuirmann, 1987)。实际差值指定一个差值阈值，比它小的差值被视为实际等价的。针对以下两个原假设构造单侧 *t* 检验：真实差值超过实际差值；真实差值小于实际差值的负数。若这两个检验都被拒绝，则表示均值的绝对差值位于实际差值范围内。因此，组被视为实际等价的。

实际等价性 p 值是在单侧 *t* 检验中获得的最大 *p* 值。小的实际等价性 p 值表示水平 i 的响应均值在实际意义上与水平 j 的响应均值等价。

实际结果 实际差值检验和等价性检验结果的说明。值会进行颜色编码来帮助识别显著结果。

- 不同（粉色）：表示绝对差值显著大于实际差值。
- 等价（绿色）：表示绝对差值显著位于实际差值范围内。
- 无结果（灰色）：表示实际差值检验和实际等价性检验都不显著。

“拟合模型”中的“响应筛选”特质

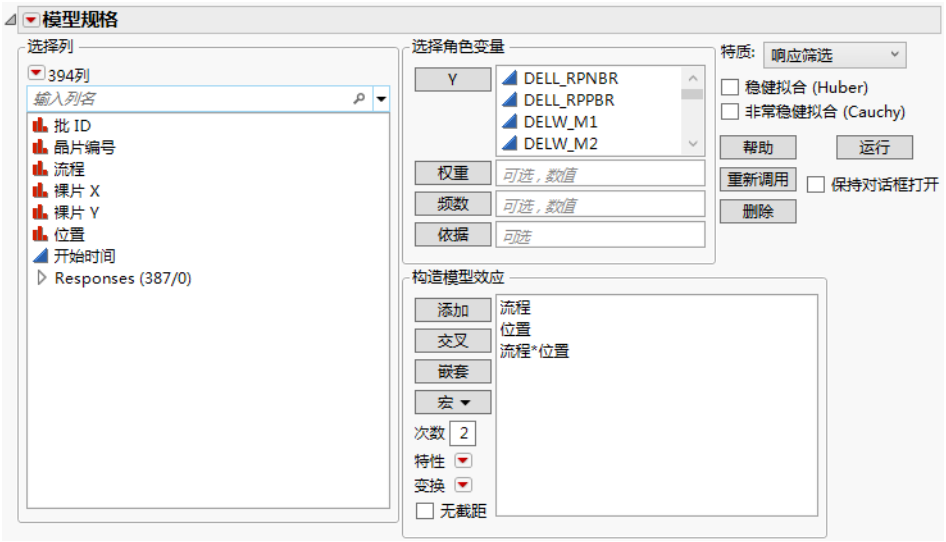
若您想要针对线性模型效应进行一元检验，可以在 “拟合模型”中拟合 “响应筛选”特质。生成的报表和表包含所有响应对所有模型效应的检验。

- “在“拟合模型”中启动“响应筛选””
- ““拟合响应筛选”报表”
- ““拟合响应筛选”选项”
- ““PValues”数据表”
- ““Y Fits”数据表”

在“拟合模型”中启动“响应筛选”

选择**分析** > **拟合模型**。输入您的 Y 变量和模型效应。从 “特质”列表中选择**响应筛选**。

图 22.8 “拟合模型”窗口中的 “响应筛选”



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

有两个选项可用于稳健估计：

稳健拟合 (Huber) 使用稳健 (Huber) 估计来减小连续响应离群值的权重。若无离群值，这些估计值接近最小二乘估计值。请注意该选项会增加处理时间。

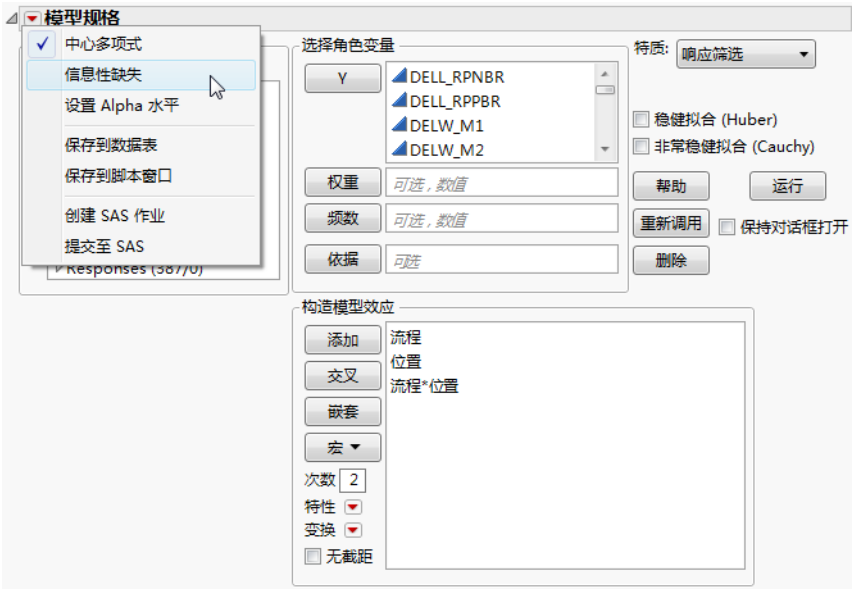
非常稳健拟合 (Cauchy) 指定非常稳健 (Cauchy) 估计来减小连续响应离群值的权重。假定误差服从 Cauchy 分布。 Cauchy 分布具有比正态分布更肥大的尾部，从而弱化了对离群值的强调。若您的数据中有较大比例的离群值，该选项会很有用。不过，若您的数据接近正态，

只含有少数离群值，该选项会导致不正确的推断。 Cauchy 选项使用最大似然和 Cauchy 连结函数估计参数。

提示：若同时选择了两个稳健选项，则平台仅使用 Cauchy 估计。

信息性缺失选项提供缺失值的编码系统。尽管存在缺失值，“信息性缺失”编码允许预测模型的估计。这在缺失数据可提供信息的情形中很有用。从“模型规格”红色小三角菜单中选择该选项。

图 22.9 “信息性缺失”选项



有关“拟合模型”窗口的详细信息，请参见《拟合线性模型》。

“拟合响应筛选”报表

“拟合响应筛选”报表显示两个图：

- FDR p 值图
- “秩分数-FDR LogWorth”图

FDR p 值图同筛选平台对该图的解释。请参见第 394 页的“ “响应筛选”报表”。

“秩分数 -FDR LogWorth”图显示针对 *p* 值的秩标绘的 FDR LogWorth 值。标绘的点随着秩分数的增加而减小或保持不变。该图指出检验的多大比例是显著的。第 412 页的 ““响应筛选”特质”中给出了使用 “响应筛选”特质的示例。

“拟合响应筛选”选项

“拟合响应筛选”红色小三角菜单包含用于保存计算数据的选项。

模型对话框 打开一个窗口，其中包含您刚刚运行以获得给定报表的模型对话框。

保存估计值 打开一个数据表，表中的每一行对应于一个响应，列对应于模型项。这些条目是通过拟合指定模型得到的参数估计值。该数据表还包含一个名为“原始数据”的表变量，它给出用于分析的数据表的名称。若您指定了 “依据”变量，则 JMP 会为 “依据”变量的每个水平创建估计值表，并且 “原始数据”变量给出 “依据”变量及其水平。

保存预测公式 向原始数据表添加包含所有响应的预测方程的列。

保存最小二乘均值 打开一个数据表，表中的每一行对应于一个响应以及效应设置的组合。该行包含这个设置组合的最小二乘均值和标准误差。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

“PValues”数据表

“PValues”数据表中的每一行对应于一对 Y 变量和相应的模型效应项。表中的列包含以下信息。若在启动窗口中选择 “稳健拟合”选项，则使用 Huber M 估计拟合模型。

Y 指定的响应列。

效应 指定的模型效应。

F 比 效应检验的检验统计量。该值可在 “最小二乘法拟合”的 “效应检验”报表中找到。

p 值 显著性检验对应于 F 比的 *p* 值。有关“效应检验”的详细信息，请参见《拟合线性模型》。

LogWorth 数量 $-\log_{10}(p \text{ 值})$。该变换调整 *p* 值以提供适用于绘图的尺度。超过 2 的值在 0.01 水平下显著 （因为 $-\log_{10}(0.01) = 2$）。

FDR p 值 使用 Benjamini-Hochberg 方法计算的假发现率 *p* 值。该方法调整 *p* 值以控制多重检验的假发现率。有关 FDR 校正的详细信息，请参见 Benjamini and Hochberg (1995)。有关假发现率的详细信息，请参见第 413 页的 “假发现率”或 Westfall et al.(2011)。

FDR LogWorth 数量 -$\log_{10}$(FDR p 值)。这是用于标绘和评估显著性的最佳统计量。请注意，小的 *p* 值将生成高的 FDR LogWorth 值。

秩分数 表示为检验数的分数形式的 FDR LogWorth 的秩。若检验数为 *m*，则最大 FDR LogWorth 值具有秩分数 1/*m*，最小 FDR LogWorth 值则具有秩分数 1。也可以说，秩分数以秩 / 检验总数的分数形式按升序对 *p* 值排序。“秩分数”用于以显著性下降的顺序对 p 值和 FDR p 值绘图。

检验自由度 效应检验的自由度。

“PValues”数据表还包含名为“原始数据”的表变量，该变量给出用于分析的数据表的名称。若您指定了“依据”变量，JMP 会为“依据”变量的每个水平创建一个“PValues”表，并且“原始数据”变量给出“依据”变量及其水平。

“Y Fits”数据表

“Y Fits”数据表的每一行对应于一个 Y 变量。对于每个 Y，表中的列汇总了有关模型拟合的信息。若在启动窗口中选择 “稳健拟合”选项，则使用 Huber M 估计拟合模型。

Y 指定的响应列。

R 方 多重相关系数。

RMSE 均方根误差。

计数 观测数 （或 “权重”变量之和）。

总体 F 比 “最小二乘法拟合”中 “方差分析”报表的模型拟合的检验统计量。

总体 p 值 模型显著性的总体检验的 *p* 值。

总体 LogWorth 模型显著性的总体检验的 *p* 值的 LogWorth。

总体 FDR p 值 针对假发现率调整的总体 *p* 值。（请参见第 394 页的 ““响应筛选”报表”。）

总体 FDR LogWorth 总体 FDR p 值的 LogWorth。

总体秩分数 总体 FDR LogWorth 的秩与检验数之比。若检验数为 *m*，则最大总体 FDR LogWorth 值具有秩分数 1/*m*，最小总体 FDR LogWorth 值则具有秩分数 1。

< 效应 > p 值 这些列包含每个模型效应检验的 *p* 值。这些列在 “列”面板中名为 “p 值”的组中排列。

< 效应 > LogWorth 这些列包含每个模型效应检验的 *p* 值的 LogWorth。这些列在 “列”面板中名为 “LogWorth”的组中排列。

< 效应 > FDR LogWorth 这些列包含每个模型效应检验的 FDR LogWorth。这些列在 “列”面板中名为 “FDR LogWorth”的组中排列。

“Y Fits”数据表还包含名为“原始数据”的表变量，该变量给出用于分析的数据表的名称。若您指定了 “依据”变量， JMP 会为 “依据”变量的每个水平创建一个 “Y Fits”表，并且 “原始数据”变量给出 “依据”变量及其水平。

响应筛选的更多示例

- “实际显著性检验和等价性检验的示例”
- ““LogWorth 最大值”选项的示例”
- “稳健拟合的示例”
- ““响应筛选”特质”

实际显著性检验和等价性检验的示例

该示例使用 Probe.jmp 样本数据表检验实际差值。

1. 选择**帮助** > **样本数据库**，然后打开 Probe.jmp。
2. 选择**分析** > **筛选** > **响应筛选**。

   将显示 “响应筛选”启动窗口。
3. 选择 Responses 列组并点击 **Y，响应**。
4. 选择**流程**并点击 **X**。
5. 在 “实际差值部分”框中键入 0.15。
6. 点击**确定**。
7. 点击 “响应筛选”红色小三角并选择**保存比较均值**。

   图 22.10 显示了数据表的一部分。对于 Y 中的每个响应，相应的行给出有关**流程**的 “New”和 “Old”水平的检验信息。

图 22.10 “比较均值”表，部分视图

	Y	水平i	水平j	X	差值	差值标准误差	纯差值 p 值	实际差值	实际差值 p 值	实际等价性 p 值	实际结果
1	DELL_RPNBR	New	Old	流程	-0.031048933	0.004326425	8.044596e-13	0.1486469486	1	1.57079e-153	等价
2	DELL_RPPBR	New	Old	流程	-0.024751371	0.0349025251	0.4782555718	1.1939335307	1	2.40983e-225	等价
3	DELW_M1	New	Old	流程	0.0275288913	0.0205582035	0.1806001982	0.7030518329	1	1.40119e-217	等价
4	DELW_M2	New	Old	流程	0.0739317751	0.0017890895	0	0.0704822866	0.0269479498	0.9730520502	不同
5	DELW_NBASE	New	Old	流程	-1.280753006	1.4836374126	0.388034756	50.689291592	1	6.62236e-223	等价
6	DELW_NEMIT	New	Old	流程	-0.040133074	0.002521398	7.306034e-56	0.0848381437	1	7.74847e-69	等价
7	DELW_NENBNI	New	Old	流程	-0.93842129	0.0220490008	0	0.864024229	0.0003726256	0.9996273744	不同
8	DELW_NSINK	New	Old	流程	1.3973855188	0.0254310074	0	1.0723100972	3.228438e-37	1	不同
9	DELW_PBASE	New	Old	流程	-0.043649097	0.0037311377	2.908704e-31	0.1296380428	1	8.21675e-113	等价
10	DELW_PCOLL	New	Old	流程	1.8169399617	0.0085444762	0	1.17539613	0	1	不同
11	DELW_PEMIT	New	Old	流程	-0.022697392	0.0534170363	0.6709186367	1.8265075627	1	1.2499e-228	等价
12	DELW_PSINK	New	Old	流程	0.911093371	0.0678503246	1.666427e-40	2.3558597461	1	3.147655e-97	等价
13	DELW_RPNBR	New	Old	流程	-0.101421251	0.0100452789	9.008231e-24	0.3464883523	1	1.6441e-125	等价
14	DELW_RPPBR	New	Old	流程	0.3015474189	0.0280757905	1.169725e-26	0.9698077792	1	7.48088e-120	等价
15	DELW_SICR	New	Old	流程	0.0275370732	0.0176173979	0.1180928699	0.6024110643	1	7.77189e-215	等价
16	M1_COMB_VGA...	New	Old	流程	-0.532453055	0.6037284633	0.3778451214	20.662273741	1	1.83138e-223	等价
17	M1_TRENCH_V...	New	Old	流程	-0.61657228	0.1695447206	0.0002786205	5.8087989555	1	2.39165e-191	等价
18	M2/M1_CAP_V...	New	Old	流程	-0.047230999	0.0978790012	0.6294372314	3.3496974301	1	2.54456e-228	等价
19	M2_COMB_BB_...	New	Old	流程	0.0001788376	0.1270241159	0.9988767052	4.3470387459	1	3.07472e-234	等价
20	M2_COMB_VGA...	New	Old	流程	0.2971147541	0.2860872543	0.2990592705	9.7914324315	1	1.42412e-221	等价
21	NISO_TUB-TRE...	New	Old	流程	7.0118092399	0.2482751793	1.58784e-164	9.0621071628	1	9.077234e-17	等价
22	NISO_TUB-TUB...	New	Old	流程	-0.074789866	0.1081535234	0.4892691678	3.7013990674	1	8.97914e-226	等价
23	PS_RPNBR	New	Old	流程	-64.69026527	15.2943887	0.0000237641	530.27422027	1	3.12381e-189	等价

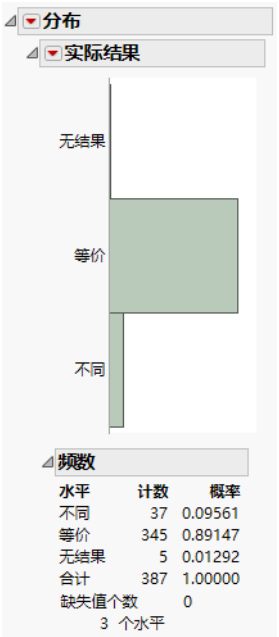
因为 Probe.jmp 中规格限没有保存为列属性，JMP 会计算每个响应的实际差值。将您指定的实际差值 0.15 与响应的 6σ 范围估计值相乘，该值用于检验实际差值和等价性。它显示在**实际差值**列中。

“纯差值 p 值”列根据 *p* 值对显著的响应进行了标识。“实际差值 p 值”和 “实际等价性 p 值”列给出实际差值检验和实际等价性检验的 *p* 值。请注意很多列在**统计上**显示存在显著差值，但是**实际上**不显示存在显著差值。

8. 显示 “比较均值”数据表并选择**分析 > 分布**。
9. 选择**实际结果**，然后点击 **Y，列**。
10. 点击**确定**。

图 22.11 显示了实际显著性的结果的分布情况。根据指定的实际差值检验，只有 37 个检验不同。有 5 个响应的检验是无结果的。您不能判断这些响应在**流程**前后是否产生了有实际意义的差异。

图 22.11 实际显著性结果的分布



可以通过点击图中相应的直条来选择 37 个响应进行进一步研究。

“LogWorth 最大值”选项的示例

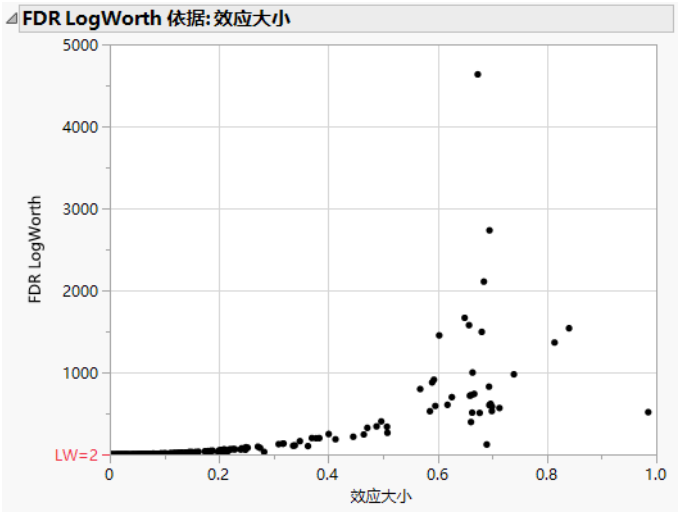
使用 “响应筛选 LogWorth 最大值”选项控制 LogWorth 尺度不被非常大的值扭曲。数据集有大量观测值时， *p* 值可能很小。这种情况下可以通过对 *p* 值的 LogWorth 值绘图进行研究。在某些数据集中， *p* 值非常小，导致 LogWorth 尺度被非常大的值扭曲。

1. 选择**帮助** > **样本数据库**，然后打开 Probe.jmp。
2. 选择**分析** > **筛选** > **响应筛选**。
3. 在 “响应筛选”启动窗口中，选择 Responses 列组，然后点击 **Y，响应**。
4. 选择**流程**并点击 **X**。
5. 选中**稳健**复选框。
6. 点击**确定**。

   分析涉及的数据很多，可能需要一些时间才能完成。
7. 在 “响应筛选”报表中，打开 “稳健 FDR LogWorth 依据 : 效应大小”报表。

   很难看清图中的详细信息，因为有一个巨大的稳健 FDR LogWorth 值，约为 58,000。要确保您的图形显示足够的详细信息，您可以设置 LogWorth 的最大值。

图 22.12 效应大小 - 稳健 FDR LogWorth，未设置 “LogWorth 最大值”

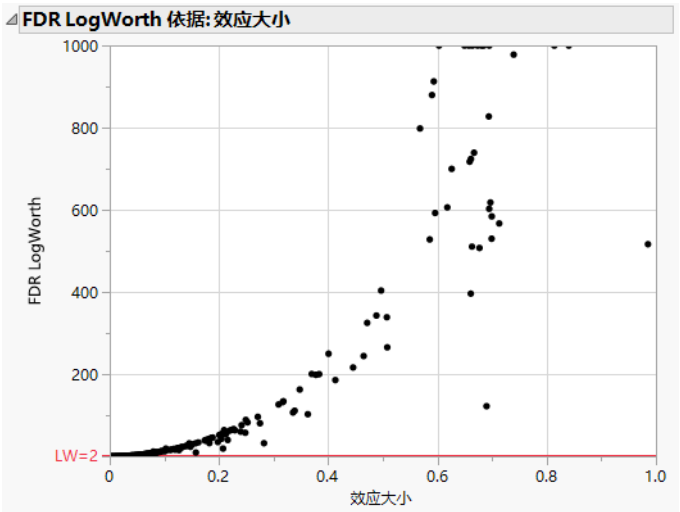


8. 重复第 1 步到第 5 步。
9. 在启动窗口底部的 “LogWorth 最大值”框中键入 1000。
10. 点击**确定**。

    该分析可能需要一些时间才能完成。
11. 在 “响应筛选”报表中，打开 “稳健 FDR LogWorth 依据 : 效应大小”报表。

    现在图中的详细信息很清晰了。

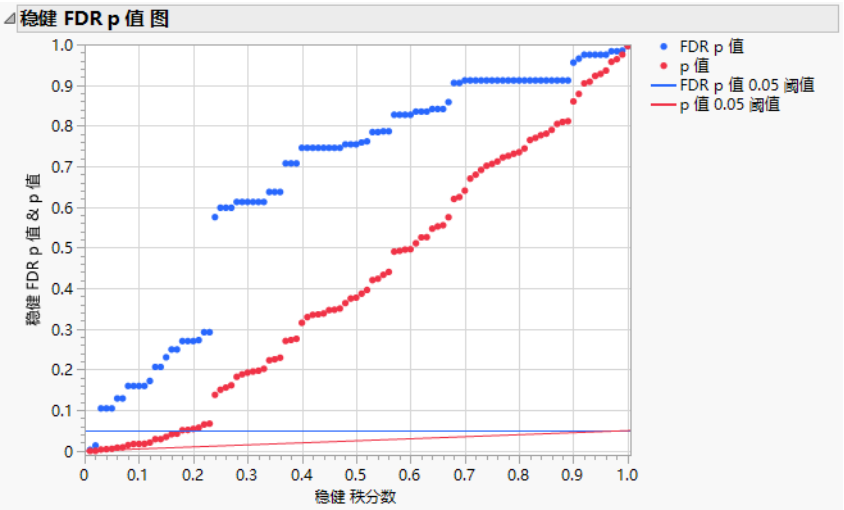
图 22.13 效应大小 - 稳健 FDR LogWorth， LogWorth 最大值 = 1000



稳健拟合的示例

1. 选择**帮助** > **样本数据库**，然后打开 Drosophila Aging.jmp。
2. 选择**分析** > **筛选** > **响应筛选**。
3. 选择所有连续列，然后点击 **Y，响应**。
4. 选择**品系**，然后点击 **X**。
5. 选中**稳健**。
6. 点击**确定**。

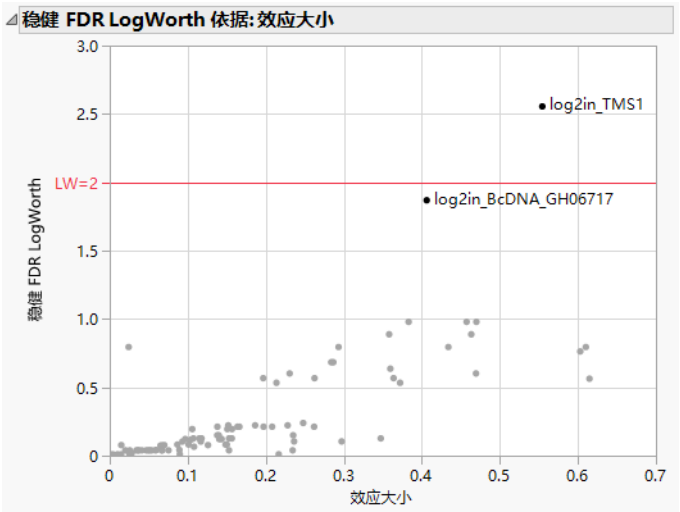
图 22.14 Drosophila 数据的 “稳健 FDR p 值图”



请注意，使用未调整的稳健 *p* 值很多检验是显著的，如小于 0.05 的红点所示。但是，根据稳健 FDR *p* 值，只有两个检验是显著的。这两个点在显示 FDR LogWorth 的图中更容易识别。

7. 点击 “稳健 FDR LogWorth 依据 : 效应大小” 展开图标。
8. 在 “稳健 FDR LogWorth” 值超过 1.5 的点周围拖曳出一个矩形。
9. 在该图形中，右击并选择**行标签**。

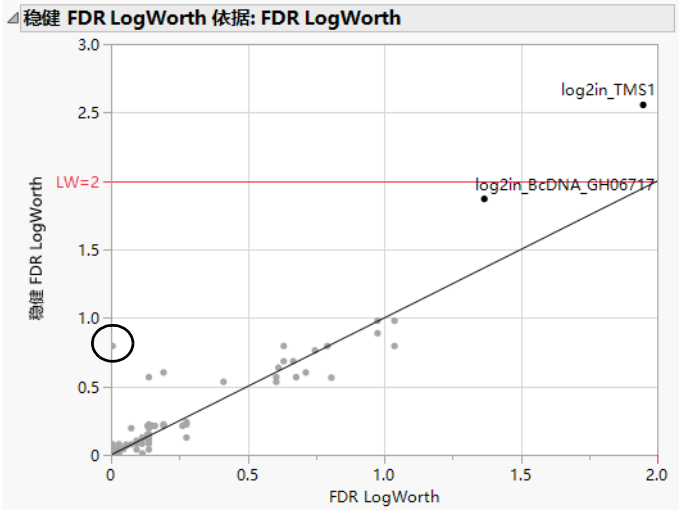
图 22.15 Drosophila 数据的 “稳健 FDR LogWorth 依据 : 效应大小”



在位于 2 值处的红线以上的点具有低于 0.01 的显著性水平 ($-\log_{10}(0.01) = 2$)。大约在 1.3 值处的水平线对应于 0.05 显著性水平。

10. 点击 “稳健 LogWorth 依据 : LogWorth” 展开图标。

图 22.16 Drosophila 数据的 “稳健 LogWorth 依据 : LogWorth”



若响应的稳健检验与常规检验相同，则它的对应点将位于图 22.16 中的对角线上。图中带圆圈的点不在对角线附近，因为它的稳健 LogWorth 值超过了 LogWorth 值。

11. 在图中该点的周围拖曳出一个矩形。

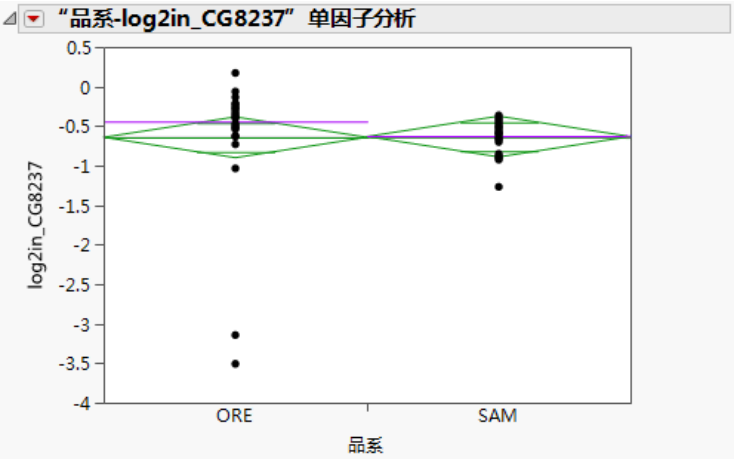
12. 在 “结果”表中找到对应该点的行。

    请注意响应 log2in_CG8237 的 p 值为 0.9568，稳健 p 值为 0.0176。

13. 点击 “响应筛选”红色小三角并选择**拟合选定项**。

    将显示 “拟合选定项”报表，其中包含响应 log2in_CG8237 的单因子分析。该图显示 ORE **品系**有两个离群值 （图 22.17）。这些离群值指出为什么稳健检验和常规检验给出了不同的结果。离群值增加了非稳健检验的误差方差，这使得更难发现显著的效应。相反，稳健拟合降低了这些离群值的权重，因此减小了它们对误差方差的贡献。

图 22.17 log2in_CG8237 的单因子分析



“响应筛选”特质

“拟合模型”中的 “响应筛选”特质支持您研究多个响应对线性模型效应的检验。

1. 选择**帮助 > 样本数据库**，然后打开 Drosophila Aging.jmp。
2. 选择**分析 > 拟合模型**。
3. 选择所有连续列，然后点击 **Y**。
4. 选择**渠道**，然后点击**添加**。
5. 选择**性别**、**品系**和**年龄**，然后选择**宏 > 完全析因**。
6. 从 “特质”列表中选择**响应筛选**。
7. 点击**运行**。

   将显示 “拟合响应筛选”报表。还显示两个数据表：“Y Fits”数据表汇总了总体模型检验，“PValues”数据表针对每个 Y 检验模型中的各个效应。

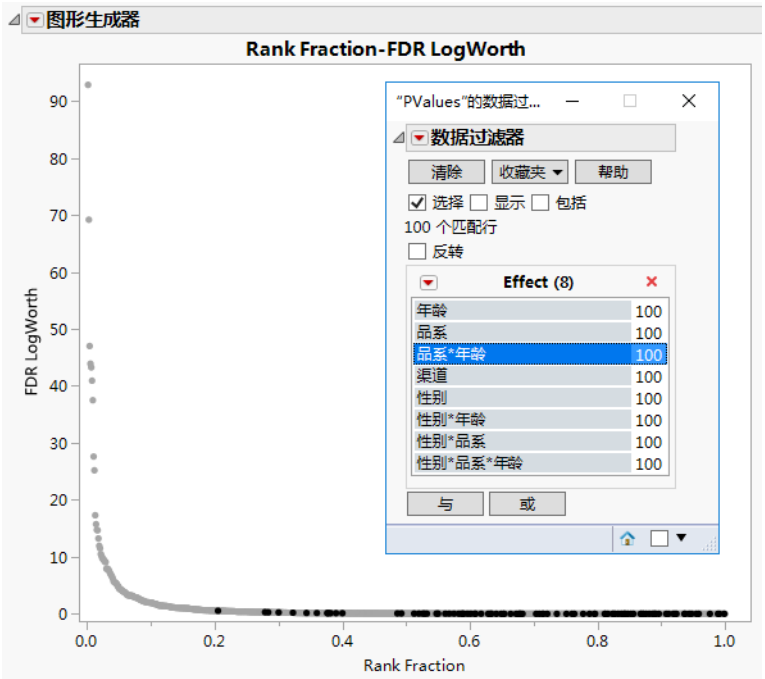
   要大概了解哪些效应是重要的，请执行以下步骤：

8. 运行 “PValues”数据表中的**秩分数 -FDR LogWorth** 脚本。
9. 选择**行 > 数据过滤器**。

10. 在 “数据过滤器”窗口中，选择**效应**，然后点击**添加**。
11. 在 “数据过滤器”中，在模型效应列表中点击各个效应，与此同时查看 “秩分数 -FDR LogWorth”图中的选定点。

    请记住，超过 2 的 LogWorth 值在 0.01 水平下是显著的。通过使用 “数据过滤器”，您发现除了**性别**和**渠道**外，模型效应在 0.01 水平下很少是显著的。图 22.18 显示了位于值 2 处的参考线。选择了**品系 * 年龄**交互作用效应检验所对应的点。这些点在 0.01 水平下不显著。

图 22.18 选择了品系 * 年龄检验的 “秩分数 -FDR LogWorth”图



“响应筛选”平台的统计详细信息

假发现率

所有“响应筛选”图涉及使用 Benjamini and Hochberg (1995) 中所述 FDR 方法执行的检验的 p 值。另见 Westfall et al. (2011)。该方法假定 p 值是独立的，且均匀分布。

JMP 使用以下步骤控制 α 水平上的假发现率：

1. 执行 m 个相关假设检验以获取 p 值 $p_1, p_2, \ldots, p_m$。
2. 按从小到大的顺序将 p 值排名。用 $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(m)}$ 表示这些值。
3. 找到 $p_{(i)} \leq (i/m)\alpha$ 最大的 p 值。假定这第一个 p 值是第 k 个最大 $p_{(k)}$。

4. 拒绝与 p 值小于等于 $p_{(k)}$) 关联的 k 个假设。

该过程确保假发现率不超过 α。

针对假发现率调整的 p 值 （表示为 $p_{(i),FDR}$），按以下方式计算：

$$p_{(i),FDR} = \begin{cases} p_{(m)} & \text{对于 } i = m \\ min\left[p_{(i+1),FDR}, \left(\frac{m}{i}\right)p_{(i)}\right] & \text{对于 } i = m-1, \ldots, 1 \end{cases}$$

若某个假设具有小于 α 的 FDR 调整的 p 值，则过程拒绝该假设。

第 23 章

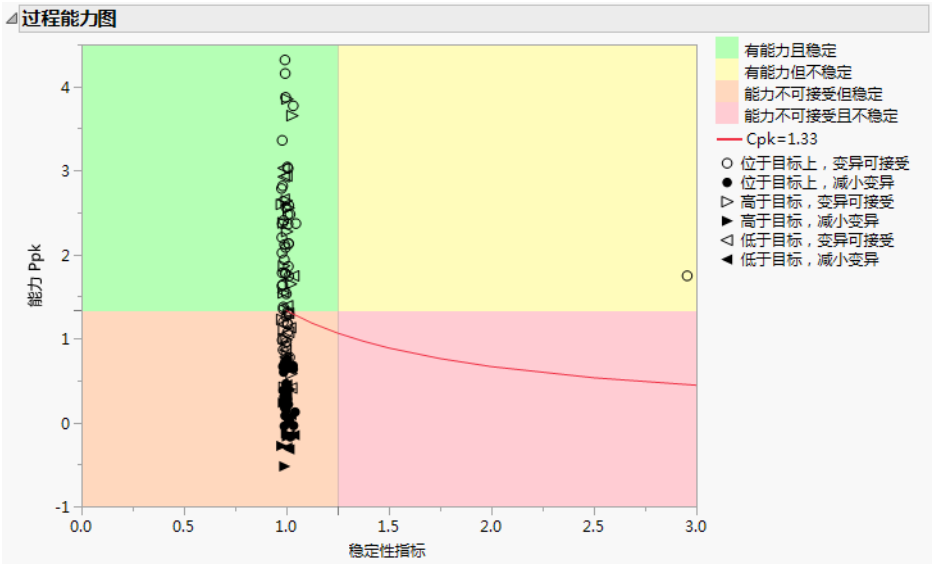
过程筛选

根据稳定性和能力筛选很多过程

使用“过程筛选”平台探索一段时间上的大量过程。该平台计算过程稳定性和过程能力指标。该平台创建控制图并检测大的过程偏移。该平台旨在通过快速聚焦不稳定、不满足规格限或发生均值偏移的过程让您加快对大量过程的评估。

根据您的初始结果，您可以选择以图形方式或更大的分析力度来探索特定过程。您可以轻松访问“控制图生成器”和“过程能力”平台。您可以为所有过程或特定过程保存详细结果。

图 23.1 过程能力图的示例



目录

“过程筛选”平台概述 417
过程筛选的示例 417
启动“过程筛选”平台 420
启动窗口角色 420
启动窗口选项 421
限值表 422
“过程筛选”报表 424
“过程筛选”平台选项 428
过程能力图 432
过程筛选的更多示例 433
“过程筛选”平台的统计详细信息 435
使用中位数估计 Sigma 的尺度因子 435

“过程筛选”平台概述

“过程筛选”平台方便执行从大量过程评估数据的稳定性和能力的任务。结果主要基于控制图计算来确定过程失控的时间。“过程筛选”平台支持您执行以下操作：

- 指定常数子组大小或将包含子组标识符的变量用于使用子组的控制图计算。
- 使用分组变量。针对组变量的每个值组合为每个过程变量提供分析。
- 使用中位数来使中心线和 sigma 计算对离群值稳健。
- 获取有关过程中均值出现大偏移的位置信息。

您可以定制“汇总”报表来显示特定的控制图检验，包括过程均值和离差变动的检验。指定规格限时，该报表还提供能力信息。“过程能力图”根据稳定性和能力给您一个过程能力的直观表示形式。“偏移图形”显示向上偏移和向下偏移的位置。

提示：有关添加规格限的信息，请参见《使用 JMP》。

“过程筛选”使您可以轻松选择特定过程来进行进一步分析。该平台提供这些过程的小的运行图 - 图的大小使您易于一次查看相当大的数。您还可以链接到“控制图生成器”和“过程能力”平台来分析选定过程。

您可以针对整个过程集或仅针对选定过程以各种形式保存包含结果的数据表。

过程筛选的示例

Semiconductor Capability.jmp 样本数据表包含 128 列过程测量值。每个列包含 1,455 个测量值。您想确定哪些过程不稳定。此外，每个列包含 “规格限”列属性。若过程稳定，则适合计算过程能力。接下来针对该数据表您要评估过程的稳定性和能力。

1. 选择**帮助 > 样本数据库**，然后打开 Semiconductor Capability.jmp。
2. 选择**分析 > 筛选 > 过程筛选**。
3. 选择 Processes 列组并点击**过程变量**。

   请注意将 “控制图类型”设置为 “单值和 MR”。
4. 点击**确定**。

图 23.2 初始报表的部分视图

过程筛选

单值和 MR

列	稳定性指标	变异性 组内 Sigma	变异性 总 Sigma	汇总 均值	汇总 计数	控制图报警 报警率	控制图报警 检验 "1"	控制图报警 最新报警	能力 Ppk	能力 Cpk	能力 超出规格计数	能力 超出规格比率	能力 最新超出规格限
PMS1	2.96	6.4e-15	1.9e-14	-50	1455	0.03643	53	3	1.746	5.159	0	0	.
IVP8	1.05	0.45226	0.47293	-21.69	1455	0.00550	8	49	2.369	2.477	0	0	.
B1	1.04	5.70202	5.94318	125.972	1455	0.00275	4	43	0.124	0.130	604	0.4151	1
IVP7	1.04	5.92093	6.14904	-79.865	1455	0.00275	4	114	-0.151	-0.157	1032	0.7093	1
SIT2	1.03	3.59488	3.71963	41.1722	1455	0.00550	8	21	1.745	1.806	0	0	.
CAP	1.03	0.92619	0.95805	38.1075	1455	0.00412	6	12	-0.037	-0.039	1149	0.7897	1
VPM7	1.03	3.20356	3.3107	85.321	1455	0.00550	8	107	0.634	0.655	54	0.0371	13
NPN10	1.03	1.76238	1.82043	73.5168	1455	0.00481	7	38	3.771	3.895	0	0	.
INV2	1.03	0.6307	0.65112	17.3705	1455	0.00481	7	11	0.676	0.697	33	0.0227	25
NPN9	1.03	2.40507	2.47045	47.8068	1455	0.00344	5	9	3.653	3.752	0	0	.
IVP9	1.03	1.61702	1.65762	-30.721	1455	0.00344	5	137	0.410	0.420	145	0.0997	1
VTN210	1.02	2.30557	2.35977	0.09116	1455	0.00344	5	9	0.582	0.595	55	0.0378	6
SIT1	1.02	15.3977	15.7506	149.659	1455	0.00481	7	16	0.088	0.090	581	0.3993	2
INM1	1.02	3.28224	3.34868	82.4373	1455	0.00412	6	3	1.649	1.682	0	0	.
IVP3	1.02	3.08312	3.14263	-49.493	1455	0.00206	3	3	0.395	0.403	168	0.1155	4
VTP210	1.02	3.31312	3.3769	1.29622	1455	0.00344	5	25	1.128	1.150	1	0.0007	1052
PBL1	1.02	0.26689	0.27188	2.71456	1455	0.00481	7	65	0.669	0.681	31	0.0213	53
NPN6	1.02	1.12479	1.14543	43.2968	1455	0.00206	3	69	-0.173	-0.176	1111	0.7636	2
E2A1	1.02	0.00132	0.00134	0.62966	1455	0.00344	5	21	2.478	2.522	0	0	.
VDP1	1.02	0.23752	0.24127	28.8111	1455	0.00137	2	17	0.775	0.787	15	0.0103	27
M1_M1	1.02	0.20527	0.20846	0.23703	1455	0.00412	6	26	-0.149	-0.151	1007	0.6921	1
A2N	1.01	8.53741	8.66111	56.0799	1455	0.00344	5	25	0.019	0.019	925	0.6357	1

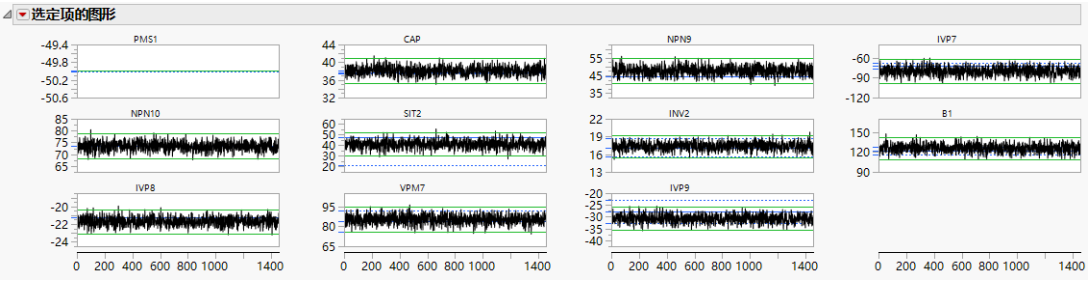
"过程筛选"窗口出现时，显示每个过程的结果表。该表按"稳定性指标"排序。这是过程的稳定性测度，其中稳定过程的稳定性指标接近 1。稳定性指标值越高表示过程越不稳定。（该排序由报表中"稳定性指标"旁边的插入符号表示。）您想更仔细查看稳定性指标值大于等于 1.03 的过程。

5. 在报表窗口中，选择过程 PMS1 到 IVP9。

   前 11 个过程在"稳定性指标"列中的值均为 1.03 或更大。

6. 右击选定的过程并选择**选定项的快速图形**。

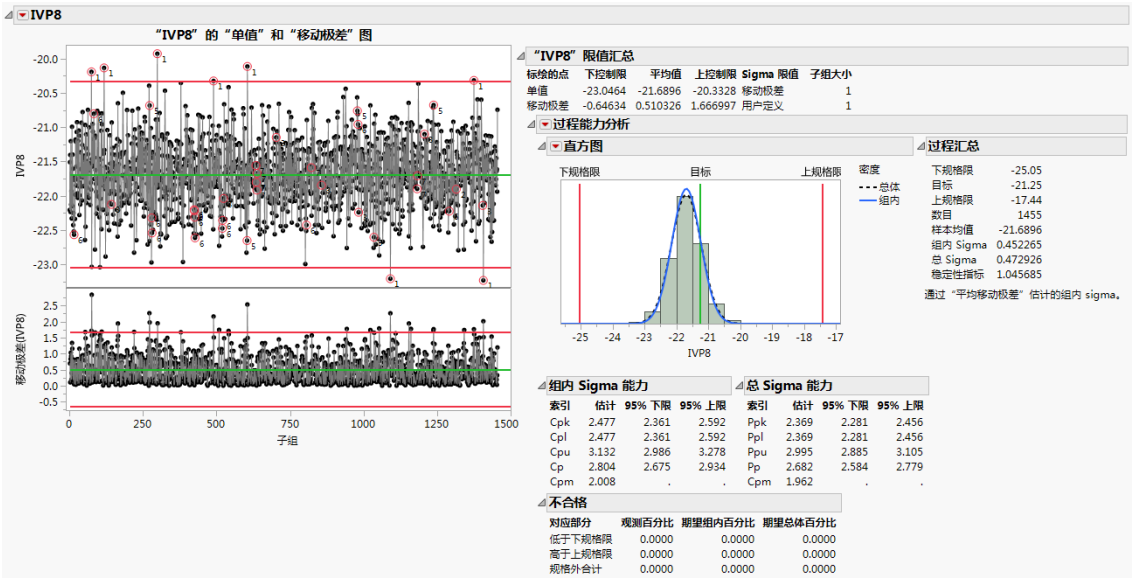
图 23.3 最高报警率过程的快速图形



您决定更仔细查看 IVP8（选定项的图形中的第 3 行，第 1 列）。

7. 在"汇总"表中选择第二个过程，它对应于 IVP8。
8. 右击选定的过程并选择**选定项的控制图**。

图 23.4 IVP8 的 “控制图生成器” 报表

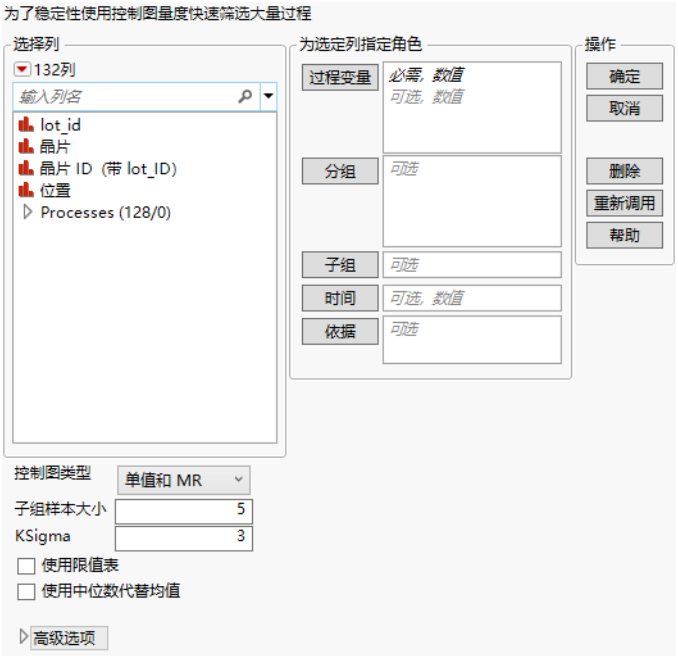


将显示 “控制图生成器” 报表。因为 IVP8 有 “规格限” 列属性，该报表还包含能力分析。

启动 “过程筛选” 平台

通过选择**分析** > **筛选** > **过程筛选**来启动 “过程筛选” 平台。

图 23.5 “过程筛选” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

启动窗口角色

过程变量 包含要分析的测量值的过程数据列。这些列的数据类型必须是 “数值” 。若一个过程变量具有 “控制限” 列属性，“过程筛选” 平台会使用这些限值来计算 “指定的 Sigma”。

注意：该平台不支持在散度 (R) 图的 “控制限” 列属性中指定的控制限。

分组 分配为分组变量的列。在分组变量的每个水平组合处分析每个过程变量。结果显示在单个报表中。

子组 其值用于为每个行分配子组标识符的列。在执行计算前，过程数据按 “子组” 变量排序。

时间 其值用于数据的时间顺序的数值列。将 “时间” 角色用于有时间戳的数据。该时间戳用于快速图形和偏移图形中的时间轴。在执行计算前，过程数据按 “时间” 变量排序。

依据 一列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在单独的表和报表中。若分配了多个 “依据”变量，则为 “依据”变量水平的每个可能组合生成单独分析。

启动窗口选项

控制图类型 选择五个控制图类型中的一个：“单值和 MR”（单个测量值和移动极差）、“均值和 R”、“均值和 S”、“均值、MR 和 R”或 “均值、MR 和 S”。“均值、MR 和 R”控制图和 “均值、MR 和 S”控制图也称为三因子图。有关统计详细信息的更多信息，请参见《质量和过程方法》。

注意：仅当子组大小大于 1 时三因子控制图才可用。若选择了三因子控制图且子组大小为 1，将显示错误消息。

注意：若某过程的子组大小为 1，则该图自动切换为 “单值和 MR”图。

子组样本大小 指定子组的常数样本大小。最小子组大小为 2。子组大小 5 为默认值。对于 “单值和 MR”图或指定子组变量时，忽略指定 “子组样本大小”。

KSigma 指定 sigma 乘数。KSigma 是在计算控制限时乘以 sigma 的值。默认情况下，KSigma 为 3。

使用限值表 支持您从数据表导入历史控制限和规格限。选择该选项并点击启动窗口中的 “确定”时，显示 “选择限值表”窗口。选择限值表并点击 “确定”后，将显示 “限值指定”窗口。将限值表中的列分配给合适的角色，然后点击 “确定”。请参见第 422 页的“限值表”。

注意：若您未选择该选项，将从过程变量的 “控制限”和 “规格限”列属性获取限值。若不选择该选项并且 “过程变量”不具有控制限列属性，则将根据数据计算控制限。

使用中位数代替均值 使用观测的中位数估计中心线。通过使用 Monte Carlo 模拟获取的尺度因子估计 Sigma。因子表在第 435 页的“‘过程筛选’平台的统计详细信息”中给出。该计算取决于选择的图类型：

- 对于 “均值和 R”图或 “单值和 MR”图，使用极差的统一尺度中位数估计 sigma。
- 对于 “均值和 S”图，使用标准差的统一尺度中位数估计 sigma。
- 对于不等子组大小，尺度因子对应于舍入到最近整数的平均子组大小。

当一个或多个离群值影响中心线位置时，很多子组可能显示失控。使用中位数可以缓解这一问题。

注意：选择 “使用中位数代替均值”时，从 “选定项的控制图”或 “选定项的过程能力”红色小三角菜单选项获取的结果与 “过程筛选”结果不匹配。

高级选项 包含以下选项：

偏移阈值 指定控制偏移图形的灵敏度的值。默认情况下将“偏移阈值”设置为 3。删除离群值后，偏移图形显示超过偏移阈值所指定的组内 Sigma 单位数的所有过程偏移的出现时间图。请参见第 429 页的“偏移图形”。

离群值阈值 指定一个值，它控制偏移图形中离群值删除的灵敏度以检测最近的大偏移。默认情况下将“离群值阈值”设置为 5。若某个观测到它的两个邻近观测的组内 sigma 单位数超过指定的离群值阈值，则使用距离其最近邻近观测一个单位组内 sigma 的值替换该观测。请参见第 427 页的“偏移量值和位置”。

偏移 Lambda 支持您更改在偏移图形中使用的指数加权移动平均 (EWMA) 权重。请参见第 427 页的“偏移量值和位置”。

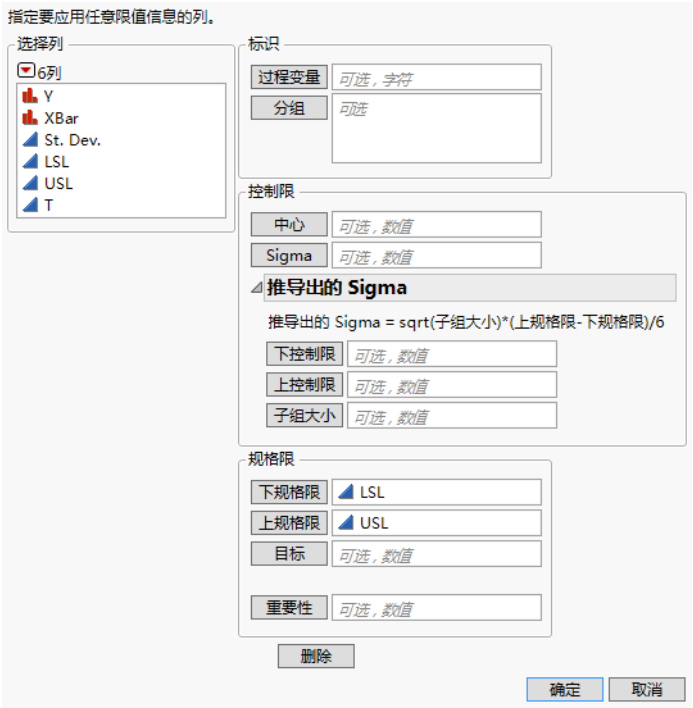
漂移 Beta 用于漂移检测的、指数加权移动平均 (EWMA) 中使用的权重。默认情况下，漂移 Beta 设置为 0.05。

最小过程长度 要包括在分析中所需的过程必须具有的最小数据值数。默认情况下，该值设置为 3。

限值表

“限值表”对于由过程数据表中“过程变量”和“分组”变量定义的每个过程包含相应的一行。使用限值表时，“限值指定”窗口支持您使用以下所列的角色指定变量。无需为所有这些角色指定变量。所有这些角色都是可选的。

图 23.6 “限值指定”窗口



若“限值表”中的列具有相应名称或匹配角色按钮的名称，则会自动填充这些列。例如，所有名为“过程”、“列”或“参数”的列自动填充到“过程变量”列表。

若您有控制限但是没有 “中心”或 “Sigma”的列，可以使用 “推导出的 Sigma”选项。

过程变量 包含对应于过程数据表中列名的值的列。

分组 一个或多个列，它们包含过程数据表的分组变量值。

中心 包含每个过程的中心线值的列。这通常是历史过程均值。

Sigma 包含每个过程的组内标准差值的列。这通常是历史标准差。

推导出的 Sigma 基于给定的控制限和子组大小计算 sigma。按以下方式计算推导出的 sigma：

$$\text{Sigma} = d^*(\text{上控制限} - \text{下控制限})/6$$

其中 d 是子组大小的平方根。

下控制限 包含每个过程的下控制限的列。

上控制限 包含每个过程的上控制限的列。

子组大小 包含每个过程的子组大小的列。

下规格限 包含每个过程的下规格限的列。

上规格限 包含每个过程的上规格限的列。

目标 包含每个过程的目标值的列。

重要性 包含每个过程的重要性值的列。“重要性”列提供按您喜欢的顺序对过程排序的机制。

“过程筛选”报表

“过程筛选”报表打开，其中有一个“汇总”表，它包含有关过程稳定性的结果。“汇总”表之上有一个标签，它显示结果所基于的控制图类型。除了“单值和 MR”之外，每个图表类型的汇总表上方还显示样本大小或子组变量。若您提供了规格限，“汇总”表还包含能力结果。这些过程和组最初按“稳定性指标”的降序排序。若用户指定了“重要性”值，则按“稳定性指标”和“重要性”的降序对过程和组排序。“稳定性指标”、“Ppk”、“Cpk”、“Cp”和“目标指标”的列分别着绿色、黄色和红色以指示足够、边缘和很差的稳定性或能力。该颜色编码方案与过程能力图的颜色编码方案匹配。

提示：要按某列排序报表，请点击列名。一个插入符号显示在列名的右侧。插入符号的方向指示排序是降序还是升序。要更改排序顺序，请再次点击列名。

“汇总”表中的控制图计算包括 Nelson 检验和“超出范围限值”检验。这些检验假定以下有关控制图限值的条件成立：

- 均值或 X 控制图的中心线由所有测量值的均值给出。若您选择“使用中位数代替均值”选项，中心线将由观测的中位数给出。
- 控制限位于距离中心线的 *K* 个 sigma 单位处。使用启动窗口中的 KSigma 选项指定 *K*。默认情况下，KSigma 为 3.
- 使用对应于您指定的控制图类型的约定估计 Sigma，若您使用中位数代替均值，则如第 421 页的“使用中位数代替均值”中所述进行估计。

提示：“过程筛选”平台中的八个 Nelson 检验遵循“控制图生成器”平台首选项中的检验设置。您可以在“文件”>“首选项”>“平台”>“控制图生成器”中定制检验。

“汇总”表可以包含以下信息：

列 作为“过程”输入的列。对于“过程”和“分组”列的每个不同组合有相应一行。若只有一个过程列，则隐藏该列。

提示：要访问对选定项进行操作的选项，请右击该**列**列。

分组列 对于您在数据表中作为“分组”输入的每列，有一个报表列。列出“分组”列的水平，以便对于每个不同的“过程”名称和“分组”列值的组合在报表表中有唯一的一行。

重要性 （仅当用户指定了“重要性”值时显示，如来自限值表的输入。）用户指定的用于过程的重要性值。

变异性 包含以下列：

稳定性指标 过程的稳定性测度。稳定过程的稳定性指标接近 1。该值越大稳定性越小。稳定性指标定义如下：

(总 Sigma/ 组内 Sigma)

若在启动窗口中选择了三因子控制图，则稳定性指标定义如下：

(总 Sigma/ 组间和组内 Sigma)

稳定性比 过程的稳定性测度。稳定过程的稳定性比接近 1。该值越大稳定性越小。稳定性比定义如下：

$(\text{总 Sigma}/\text{组内 Sigma})^2$

若在启动中选择了三因子控制图，则稳定性比定义如下：

$(\text{总 Sigma}/\text{组间和组内 Sigma})^2$

组内 Sigma 基于子组内部变异估计标准差。该估计值基于您指定的控制图类型，是变异的短期测度。有关统计详细信息，请参见《质量和过程方法》。若您选择“中位数代替均值”，则按第 421 页的“使用中位数代替均值”中所述计算组内 Sigma。

总 Sigma 基于所有观测的标准差的通常估计值。

组间 Sigma （仅当在启动窗口中选定三因子控制图时才显示。）基于子组间变异估计标准差。有关统计详细信息，请参见《质量和过程方法》。

组间和组内 Sigma （仅当在启动窗口中选定三因子控制图时才显示。）基于子组间变异和子组内变异估计标准差。组间和组内 Sigma 估计值定义如下：

$$\text{组间和组内 Sigma} = \sqrt{\text{组内 Sigma}^2 + \text{组间 Sigma}^2}$$

指定的 Sigma “限值指定”对话框中由 sigma 控制限指定的标准差或从控制限和子组大小推导出的标准差估计值。请参见第 423 页的“推导出的 Sigma”。可使用限值表或“控制限”列属性指定控制限和子组大小。

汇总 包含以下列：

中心线 （仅当执行以下操作之一时显示：在启动窗口中选择“使用中位数代替均值”，使用限值表导入中心值或导入控制限以推导 sigma。）“中心线”下列出的值在控制图计算中作为中心线使用。

– 若在启动窗口中选择“使用中位数代替均值”，则显示观测的总中位数。

– 若从限值表导入中心值，则显示该值。

– 若导入控制限，则按以下方式计算显示的值：(上控制限 + 下控制限)/2。

均值 所有观测的平均值。

计数 观测数。

子组 子组数。

控制图报警 包含有关导致各种检验（包括 8 个 Western Electric 规则）报警的子组的信息。标准差估计值为“组内 Sigma”值。默认情况下，仅在“汇总”表中显示“报警率”、“检验 1”和“最新报警”列。

报警率 对于在“选择检验”选项（“任何报警”）下选择的任何检验导致报警的子组数除以非缺失子组数（“子组”）。

任何报警 （仅当显示一个以上的“检验”列时显示。）对于在“选择检验”选项下选择的任何检验触发报警的子组数。这些检验包括八个 Nelson 检验和一个“超出范围限值”检验。

提示：“过程筛选”平台中的八个 Nelson 检验遵循“控制图生成器”平台首选项中的检验设置。您可以在“文件”>“首选项”>“平台”>“控制图生成器”中定制检验。

检验 1 一个点距离中心线超过三个标准差。与该点关联的子组触发报警。

检验 2 九个或更多连续点位于中心线的同一侧。与第九个点关联的子组触发报警。

检验 3 六个或更多连续点连续增加或减小。与第六个点关联的子组触发报警。

检验 4 十四个连续点的方向交替变化：先增加然后减小，或先减小然后增加。与第 14 个点关联的子组触发报警。

检验 5 位于中心线同侧的三个连续点中的两个点距离中心线超过两个标准差。与超过两个标准差的第二个点关联的子组触发报警。

检验 6 位于中心线同侧的五个连续点中的四个点距离中心线超过一个标准差。与超过一个标准差的第四个点关联的子组触发报警。

检验 7 位于中心线两侧的连续十五个点距离中心线均小于一个标准差。与第 15 个点关联的子组触发报警。

检验 8 位于中心线两侧的连续八个点距离中心线均超过一个标准差。与第八个点关联的子组触发报警。

超出范围限值 在 R、S 或 MR 图计算中超过上控制限的子组数。

超出移动极差限值 在三因子控制图计算中超出移动极差限值的子组数。

最新报警 针对任何 Nelson 或“超出范围限值”检验发出最新报警的子组位置（从最后一个子组起计数）。

能力 （仅当一些过程指定了“规格限”时显示。）包含以下选项：

Ppk 能力指标基于“总 Sigma”且假定服从正态分布。有关统计详细信息，请参见《质量和过程方法》。默认开启。

Cpk 基于组内 Sigma 或组间和组内 Sigma 的能力指标，并假定服从正态分布。有关统计详细信息，请参见《质量和过程方法》。默认开启。

Cp 解决了目标和漂移问题后的潜在能力。有关详细信息，请参见《质量和过程方法》。

目标指标 过程平均值不同于目标值的短期标准差数目。这用于测量命中目标值的过程的能力。目标指标计算为 3(Cp - Cpk)。大于 1 则认定目标指标很差，介于 0.5 到 1 之间为边缘，小于 0.5 则为足够。请参见 White et al. (2018)。

超出规格计数 位于规格限之外的观测数。默认情况下处于打开状态。

超出规格比率 位于规格限之外的观测比例。默认情况下处于打开状态。

超出规格比率期望值 落在规格限之外的观测的预期比例。“超出规格比率期望值”假设一个稳定的正态分布过程，并使用总 sigma。

最新超出规格限 从最后一个观测算起，到位于规格限之外的最新观测之间的观测数。默认情况下处于打开状态。

(均值-目标)/规格范围 规格中心化的均值。它与“过程能力”中的“规格标准化均值偏移”相同。有关统计详细信息，请参见《质量和过程方法》。

标准差/规格范围 规格统一尺度的标准差。它与“过程能力”中的“规格标准化标准差”相同。有关统计详细信息，请参见《质量和过程方法》。

下规格限 下规格限。

上规格限 上规格限。

目标 目标值。

偏移量值和位置 （仅当在“过程筛选”红色小三角菜单中选择了“偏移检测”选项时显示。）执行偏移检测以标识超过一个单位组内 sigma 的偏移。该算法对单个观测使用离群值校正和 EWMA 平滑法。下面列出了算法：

- 删除离群值以便单个离群值不指示偏移。在启动窗口中作为“离群值阈值”指定的值（默认为 5）控制离群值删除的灵敏度。若某个观测到它的两个邻近观测的组内 sigma 单位数超过指定的离群值阈值，则使用距离其最近邻近观测一个单位组内 sigma 的值替换该观测。
- 以正向时间顺序为子组均值构造 EWMA 拟合，并以逆向时间顺序为子组均值构造另一 EWMA 拟合。EWMA 拟合的 lambda 等于 0.3。
- 标识超过一个单位组内 sigma 的两个连续 EWMA 值的最大正差值和负差值。
- 这些差值的绝对值除以组内 sigma 估计值就是报告为“最大的向上偏移”和“最大的向下偏移”的值。
- 这些最大偏移中涉及的第一个子组位置定义“向上偏移位置”和“向下偏移位置”。

最大的向上偏移 超过一个单位组内 sigma 的最大向上偏移量值，它采用组内 sigma 单位来报告。

向上偏移位置或向上偏移“<时间变量>” 具有最大向上偏移的子组位置。若您指定一个时间变量，则该列在“汇总”表中命名为“向上偏移‘<时间变量>’”，并根据时间变量给出偏移位置。

最大的向下偏移 超过一个单位组内 sigma 的最大向下偏移量值，它采用组内 sigma 单位来报告。

向下偏移位置或向下偏移“<时间变量>” 具有最大向下偏移的子组位置。若您指定一个时间变量，则该列在“汇总”表中命名为“向下偏移‘<时间变量>’”，并根据时间变量给出偏移位置。

漂移量值和位置 （仅当在“过程筛选”红色小三角菜单中选择了“漂移汇总”选项时显示。）执行漂移检测是为了检测过程中更小、更渐进的变化。该算法与偏移检测中使用的算法相同，只不过偏移检测使用 Holt 双指数平滑线而不是 EWMA。下面列出了算法：

- 从偏移检测使用的相同过程中删除离群值。请参见第 427 页的“偏移量值和位置”。
- 偏移检测算法以正向时间顺序为子组均值拟合 Holt 双指数平滑模型，并以逆向时间顺序为子组均值拟合另一 Holt 双指数平滑模型。这两个平滑模型拟合每一个都有两个平滑常数：α 用于水平，β 用于斜率。β 平滑常数设置为 0.05，并且估计 α 平滑常数以将误差最小化。
- 正漂移值、负漂移值和绝对漂移值的均值分别报告为“均值向上漂移”、“均值向下漂移”和“均值绝对漂移”。

均值向上漂移 正漂移值之和除以计数。

均值向下漂移 负漂移值之和除以计数。

均值绝对漂移 所有漂移值的绝对值之和除以计数。

“过程筛选”平台选项

“过程筛选”红色小三角菜单包含定制显示内容和保存计算的统计量的选项。对选定项进行操作的这些选项也可以通过右击“汇总”表中的**列**列来访问。

汇总 显示或隐藏“汇总”表。请参见第 424 页的“‘过程筛选’报表”。

查找和选择 在启动窗口中输入您作为“过程变量”或“分组”输入的列的搜索字符串。为每个列显示一个面板。“汇总”表中相应的过程即被选定。

选择符合条件的行 打开“选择符合条件的行”窗口。您可以通过使用“比较”菜单和“值”文本框在“汇总”表中选择对应特定条件的特定过程。例如，您可以选择 Ppk < 1.33 的所有过程。点击“确定”后，在“汇总”表中选定这些过程。

提示：还可以通过右击“汇总”表中的任何位置访问“选择符合条件的行”窗口。

显示汇总列 支持您选择一些在“汇总”表中显示的汇总列。

选定项的快速图形 标绘您在“选定项的图形”报表的“汇总”表中选择的过程的小图形。该报表使我们可以立即查看和比较很多过程。根据其在启动窗口中的输入顺序排序这些图。每个快速图形还有用于规格限的线。蓝色实线表示目标值，蓝色虚线表示下规格线和上规格线值。要删除快速图形，请点击“选定项的图形”红色小三角并选择“删除”。

提示：要更改快速图形的布局，请点击“选定项的图形”红色小三角并选择“横向图数”。

选定项的控制图 为您在 “汇总”表中选择的过程打开一个 “控制图生成器”报表窗口。该控制图对应于您在 “过程筛选”启动窗口中选择的内容。

注意：只有在 “过程筛选”中选择的检验才会发送到 “控制图生成器”。不会发送禁用的检验。

选定项的过程能力 打开一个 “过程能力”报表窗口，其中显示您在 “汇总”表中选择的过程的 “单项详细报表” 。若您选择未指定规格限的过程，则显示 “规格限”窗口。在该窗口中，您可以通过选择一个数据表或直接输入值来指定规格限。

“过程能力”分析假定服从正态分布并使用对应于您在 “过程筛选”启动窗口中选择的 “控制图类型”的组内 sigma 值：

- “单值和 MR”的移动极差
- “均值和 R”的极差平均值
- “均值和 S”的无偏标准差平均值

注意：若使用 sigma 控制限或 “限值指定”窗口中的 “推导出的 Sigma”选项指定 sigma，则不在 “过程能力”中使用该值。这是因为 “过程筛选”和 “过程能力”平台以不同方式使用这些控制限。

为选定项着色 将您选择的颜色应用到 “汇总”表的选定行中的值。

删除选定项 删除在 “汇总”表中选定的行并在没有那些过程的情况下重新运行分析。

显示检验 显示或隐藏 Nelson 检验的结果，这些检验是在 “过程筛选”报表的 “汇总”表的 “选择检验”选项下选择的。

选择检验 支持您选择要在计算 “报警率”和 “任何报警”中包含的检验。

提示：要选择多个检验，请按住 Alt 键并点击 “过程筛选”红色小三角菜单以打开所有平台选项的菜单。

偏移检测 提供在删除离群值后检测偏移的选项。请参见第 427 页的 “偏移量值和位置”。

最大的向上偏移 将 “最大的向上偏移”和 “向上偏移位置”列添加到 “汇总”表。标识序列中超过一个单位组内 sigma 的最大向上偏移。请参见第 427 页的 “偏移量值和位置”。

最大的向下偏移 “最大的向下偏移”和 “向下偏移位置”列添加到了 “汇总”表中。标识序列中超过一个单位组内 sigma 的最大向下偏移。请参见第 427 页的 “偏移量值和位置”。

偏移图形 （仅当一些过程的偏移超过 “偏移阈值”时可用。）显示所有超过 “偏移阈值”（默认为 3）所指定的组内 sigma 单位数的过程偏移出现的时间图。您还可以使用偏移图形上的滑块条更改偏移阈值。悬停在菱形上方可以查看该值。绿色标记指示向上偏移，红色标记指示向下偏移。这些标记位于偏移的局部峰值处。

要标识对应于出现一次或多次偏移的过程，请选择这些点并点击“选择过程”。“汇总”表中相应的过程即被选定。不标绘没有偏移超过“偏移阈值”所指定的组内 sigma 单位数的过程。

注意：若偏移在量值上小于“偏移阈值”所指定的组内 sigma 单位数，则“偏移图形”不显示在“汇总”表中显示的“最大的向上偏移”和“最大的向下偏移”值的位置。请参见第 433 页的“过程筛选的更多示例”。

在快速图形中显示偏移 （仅当已将“快速图形”添加到报表窗口时可用。）使用绿色和红色垂直线在快速图形中显示偏移的位置。

漂移汇总 将“均值向上漂移”、“均值向下漂移”和“均值绝对漂移”的列添加到“汇总”表。请参见第 428 页的“漂移量值和位置”。

已选定漂移图形 为您在“汇总”表中选择的每个过程显示漂移图形。漂移图形支持您在选定的过程中检测更小的、更渐变的变动。标绘的值是 Holt 双指数平滑模型中的斜率估计值。请参见第 428 页的“漂移量值和位置”。每个漂移图显示 Holt 双指数平滑模型中的组内 sigma 值和两个平滑参数的估计值。

过程能力图 （仅当为至少一个过程变量定义规格限时才可用。）显示根据稳定性和能力评估过程能力的四象限图形。请参见第 432 页的“过程能力图”。

过程能力图边界 （仅当“过程能力图”位于报表窗口中时才可用。）打开一个窗口，您可以在其中设置过程能力图的稳定性指标值和 Ppk 能力边界。

提示：可以在“文件”>“首选项”>“平台”>“过程筛选”中为所需的稳定性指标和 Ppk 能力边界设置首选项。

目标图 （仅当为至少一个过程变量定义规格限时才可用。）对于每个变量，目标图在 X 轴上显示规格标准化均值偏移，在 Y 轴上显示规格标准化标准差。若为过程定义重要性值，则依据重要性确定目标图点的大小。该图可用于快速提供关于变量是否符合规格限的汇总视图。有关目标图及其特点的详细信息，请参见 《质量和过程方法》。

提示：悬停在目标图中的某个点上方可查看该过程的“快速图形”。点击“快速图形”将其添加到报表窗口。

“目标图”红色小三角菜单包含以下选项：

显示组内 Sigma 点 显示或隐藏使用组内 sigma 估计值计算的点。

显示组内或组间和组内 Sigma 点 （仅当在启动窗口中选定三因子控制图时才可用。）显示或隐藏使用组间和组内 sigma 估计值计算的点。

显示总 Sigma 点 显示或隐藏使用总 sigma 估计值计算的点。

着色水平 显示或隐藏 Ppk 水平着色。默认情况下启用它。着色区域取决于 p 和 Ppk 之间的关系，其中 p 表示 Ppk 之下框中显示的值。默认情况下，$p = 1.33$。

– 红色区域中的点：Ppk < p。

– 黄色区域中的点：$p < \text{Ppk} < 2p$。
– 绿色区域中的点：$2p < \text{Ppk}$。

添加组内 Sigma 点标签 显示或隐藏使用组内 sigma 估计值计算的点的标签。

标记组内或组间和组内 Sigma 点 （仅当在启动窗口中选定三因子控制图时才可用。）显示或隐藏使用组间和组内 sigma 估计值计算的点的标签。

添加总 Sigma 点标签 显示或隐藏使用总 sigma 估计值计算的点的标签。

缺陷率等高线 显示或隐藏表示指定缺陷率的等高线。

显示能力 （仅当为至少一个过程变量定义规格限时才可用。）显示一个子菜单，使用该子菜单可在 “汇总”表中显示或隐藏 “能力”。

注意：“规格中心化均值”和 “规格统一尺度标准差”在 “汇总”表中分别显示为 “(均值 - 目标)/ 规格范围”和 “标准差 / 规格范围”。“规格限”选项在 “汇总”表中显示 “下规格限”、“上规格限”和 “目标”列。

子菜单中还提供以下选项：

对超出规格限的值着色 对数据表中与超出规格限的值对应的单元格着色。若值高于上规格限，则单元格着蓝色；若值低于下规格限，则单元格着红色。

提示：要删除特定单元格中的颜色，请选择所有相关单元格。右击其中某一个单元格，然后选择 “清除颜色”。要删除所有单元格中的颜色，请取消选择 “对超出规格限的值着色”。

保存汇总表 将可以在 “汇总”表中显示的所有信息保存到 “过程汇总”数据表。若为至少一个过程指定了规格限，则 “过程汇总”数据表还包含规格限详细信息。正如在汇总表中，“稳定性指标”、“Ppk”、“Cpk”、“Cp”和 “目标指标”的列分别着绿色、黄色和红色以指示足够、边缘和很差的稳定性或能力。

保存包含图形的汇总表 创建具有标题为**图形**的附加列的“过程汇总”数据表。**图形**列包含“过程汇总”表中每个过程的快速图形。

保存详细信息表 将有关控制图计算的详细信息保存到 “过程详细信息”数据表。对于过程和分组变量的每个组合，该表为每个子组包含相应的一行，其中显示：

– 子组样本统计量的值。
– 控制限。
– 子组大小。
– 针对触发报警（若有）的指示符的列表。用检验编号指示 Nelson 检验的报警。用 R 指示“超出范围限值”检验的报警。
– 漂移

保存选定的详细信息 对于 “汇总”表中的选定行，保存一个 “过程详细信息”表，其中包含当您选择 “保存详细信息表”时保存的信息。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

过程能力图

“过程能力图”是一个能力-稳定性四象限图。为其提供规格限的每个过程用一个标记来表示。标记类型取决于过程是否违反了目标指标和 Cp 值的准则。指向左侧的三角形指示过程低于目标，圆圈表示过程位于目标上，指向右侧的三角形指示过程高于目标。空心标记指示过程的变异是合格的，实心标记指示过程的变异可以减少。标记组合的完整说明在表 23.1 中提供。此外，若为过程定义重要性值，则“过程能力图”上的标记将按重要性调整大小。

表 23.1 “过程能力图”标记

图形标记	目标指标	Cp
○	位于目标上	合格变异
●	位于目标上	减小变异
▷	高于目标	合格变异
▶	高于目标	减小变异
◁	低于目标	合格变异
◀	低于目标	减小变异

在图上，水平坐标表示过程的稳定性指标，垂直坐标表示过程的能力，以 Ppk 表示。根据以下默认边界将图形分为四个象限：

– 超过 1.25 的稳定性指标指示过程不稳定。

– 小于 1.33 的 Ppk 指示过程没有能力。

此外，在图形上 1.33 除以稳定性指标处有一条红线，指示 Cpk 值为 1.33 的位置。该线根据过程可能被修正的方式将不稳定且没有能力的象限中的过程分类。在红线之下的过程可通过

特殊原因修正，在红线之上的过程可通过减小变异性来修正。选择图形中的点会在 “汇总” 表中选择相应过程。

提示：悬停在 “过程能力图” 中的某个点上方可查看该过程的 “快速图形”。点击 “快速图形” 将其添加到报表窗口。

过程筛选的更多示例

本例说明如何使用 “分组” 列以及如何使用 Consumer Prices.jmp 构造 “偏移图形”。

消费者价格指数数据表包含 17 种产品的每月数据。时间期间根据产品不同而不同。排列数据，使所有 17 种产品在名为**系列**的单个列中列出。为了分离产品，您必须将**系列**视为 “分组” 列。

1. 选择**帮助** > **样本数据库**，然后打开 Consumer Prices.jmp。
2. 选择**分析** > **筛选** > **过程筛选**。
3. 选择**价格**并点击**过程变量**。
4. 选择**系列**并点击**分组**。

   这确保将**系列**的每个水平视为单独过程。
5. 选择**日期**并点击**时间**。
6. 将**控制图类型**设置为 “均值和 R”。
7. 将**子组样本大小**设置为 3。

   因为数据按月给定，子组大小 3 表示季度。
8. 点击**确定**。
9. 按 Alt 并点击 “过程筛选” 旁边的红色小三角菜单。

   这将打开一个窗口，其中显示可用的红色小三角菜单选项。您可以在该窗口一次选择多个选项。
10. 检查以下项：**最大的向上偏移**、**最大的向下偏移**和**偏移图形**。
11. 点击**确定**。

    “最大的向上偏移”、“向上偏移 ‘日期’”、“最大的向下偏移” 和 “向下偏移 ‘日期’” 添加到了 “汇总” 表中。这些偏移是超过**一个**单位组内 sigma 的最大偏移。根据时间变量**日期**给出每个偏移的位置。请参见第 427 页的 “偏移量值和位置”。

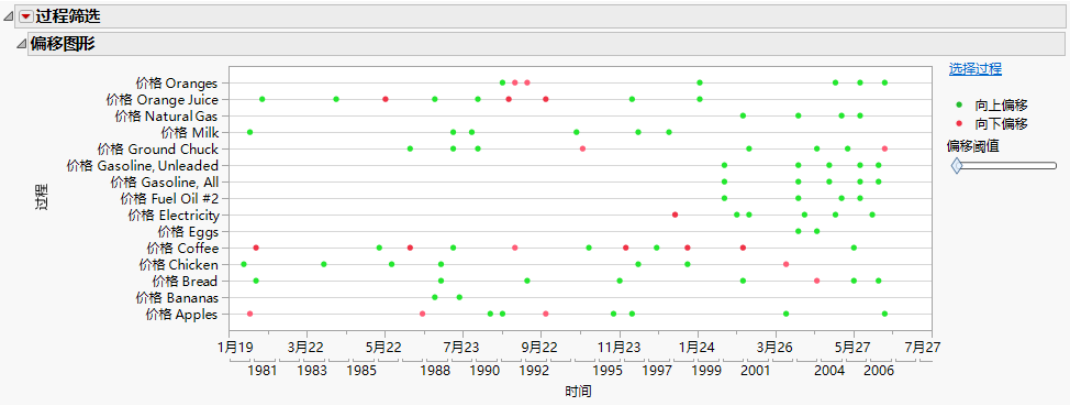
    还显示 “偏移图形”。“偏移图形” 显示超过**偏移阈值**（默认设置为 3）所指定的组内 sigma 单位数的所有偏移。请参见第 429 页的 “偏移图形”。绿色点对应于向上偏移，红色点对应于向下偏移。

    请注意，对于 “汇总” 表中的 “最大的向上偏移” 和 “最大的向下偏移”， Gasoline, All 都有值。“最大的向下偏移” 值 1.8296 小于 3。因为 “偏移图形” 仅显示大于等于 3 个单位组内 sigma 的偏移， Gasoline, All 的 “最大的向下偏移” 值未标绘在 “偏移图形” 上。

还请注意， Tomatoes 未包含在 “偏移图形”上。对于 Tomatoes，未找大于等于 3 个单位组内 sigma 的偏移。

12. 双击 “偏移图形”的水平轴以打开 “X 轴设置”窗口。
13. 在 “刻度 / 箱增量”面板中，设置**副刻度数**为 1。
14. 设置**标签行嵌套**为 2。
15. 点击**确定**。

图 23.7 偏移图形

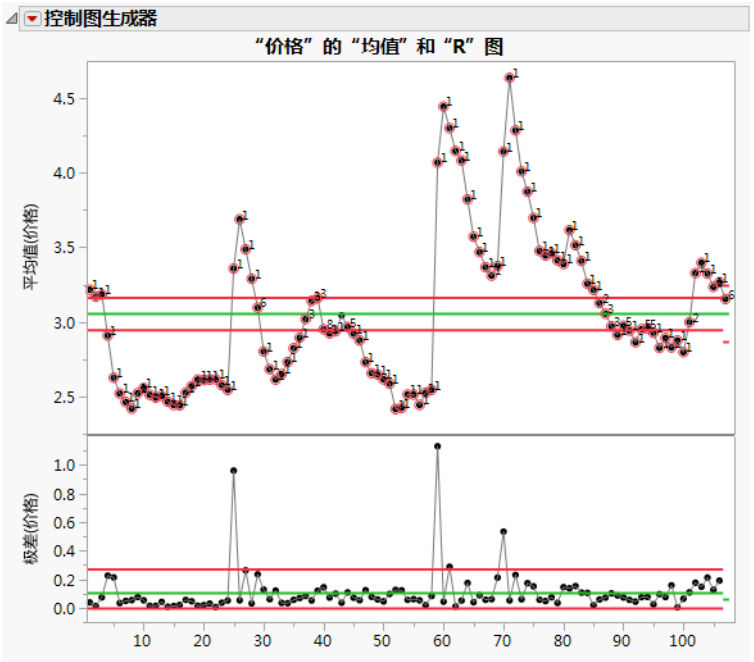


大多数序列主要显示向上偏移。但是“价格 Coffee”有几个交替的向下偏移和向上偏移。为了更好了解该序列，获取一个控制图。

16. 选择 “偏移图形”中 “价格 Coffee”右侧的任意点，然后点击**选择过程**。

    该操作选择 “汇总”表中对应 Coffee 的行。
17. 右击 “汇总”表中的选定过程并选择**选定项的控制图**。

图 23.8 Coffee 的控制图



该控制图显示了“偏移图形”中识别到的向上偏移和向下偏移。

“汇总”表指示对于包含 1994 年 9 月的子组发生最大的向上偏移（25.399 个单位组内 sigma）。在图 23.8 所示的控制图中，这是位置 59 处的子组。“汇总”表还指示对于包含 1981 年 3 月的子组发生最大的向下偏移（9.1674 个单位组内 sigma）。这是控制图中位置 5 处的子组。

因为偏移是使用 EWMA 平滑序列和离群值校正算法计算的，偏移位置可能不恰好对应于 Shewhart 控制图上似乎开始偏移的子组。

“过程筛选”平台的统计详细信息

使用中位数估计 Sigma 的尺度因子

选择“使用中位数代替均值”时，使用统一尺度的中位数极差或中位数标准差估计 sigma。下表给出一些尺度因子，它们是使用 Monte Carlo 模拟得到的。

对于从正态分布绘制的大小为 *n* 的子组，满足以下条件：

- 极差的理论中位数大约为 *d2_Median**σ，其中 *d2_Median* 是对应 *n* 的值。
- 标准差的理论中位数大约为 *c4_Median**σ，其中 *c4_Median* 是对应 *n* 的值。

表 23.2 中位数极差和中位数标准差的尺度常数

数目	d2_Median	c4_Median
2	0.953	0.675
3	1.588	0.833
4	1.978	0.888
5	2.257	0.917
6	2.471	0.933
7	2.646	0.944
8	2.792	0.952
9	2.915	0.959
10	3.024	0.963
11	3.118	0.967
12	3.208	0.969
13	3.286	0.972
14	3.357	0.975
15	3.422	0.976
16	3.483	0.978
17	3.539	0.979
18	3.590	0.980
19	3.640	0.981
20	3.685	0.982
21	3.731	0.983
22	3.770	0.984
23	3.811	0.984
24	3.846	0.985
25	3.883	0.986

第 24 章

预测变量筛选

针对显著效应筛选很多预测变量

大型数据集的分析涉及对某一零件、过程或样本取数百或数千个测量值，因此分析这样的数据集需要创新性的方法。“预测变量筛选”平台根据预测结果的能力提供筛选很多预测变量的方法。例如，预测变量筛选可用于帮助从患者（有某种症状和没有症状）检验的数千生物标记中识别生物标记来预测是否患有该症状。

预测变量筛选不同于响应筛选。响应筛选一次检验一个作为响应预测变量的因子。预测变量筛选使用 Bootstrap 森林法分割来评估预测变量对响应的贡献。针对多个预测变量构造分割模型。预测变量筛选可以标识这样的预测变量：它们单独存在时可能很弱但是与其他预测变量结合时变得很强。请参见第 387 页的“响应筛选”一章。

图 24.1 “预测筛选”报表的示例

目录

“预测变量筛选”平台概述 439
预测变量筛选的示例 439
启动“预测变量筛选”平台 441
“预测变量筛选”报表 441
“预测变量筛选”平台选项 442

“预测变量筛选”平台概述

预测变量筛选对于从很多候选项标识显著预测变量很有用。假定您有数百个 X，需要确定哪些 X 作为结果的预测变量最显著。

“预测变量筛选”平台使用 Bootstrap 森林法来标识响应的潜在预测变量。对于每个响应，构建使用 100 个决策树的 Bootstrap 森林法模型。将每个预测变量对 Bootstrap 森林法模型的列贡献排序。因为 Bootstrap 森林法涉及随机成分，重新运行报表时列贡献可能不同。有关决策树的详细信息，请参见第 47 页的“分割模型”一章。

预测变量筛选的示例

Bands Data.jmp 数据表包含轮转凹版印刷业务中机器的测量值。数据集包含 539 个记录和 38 个变量。响应 Y 是列**纹印？**，它的值是 “BAND”和 “NOBAND”。您想了解哪些属性最可能对响应有贡献。

1. 选择**帮助** > **样本数据库**，然后打开 Bands Data.jmp。
2. 选择**分析** > **筛选** > **预测变量筛选**。
3. 选择**纹印？**作为 **Y，响应**。
4. 选择分组列**粒度过滤**到**铬含量**，然后点击 **X**。
5. 点击**确定**。

图 24.2 排序的列贡献

预测变量筛选

纹印？ 复制选定

预测变量	贡献	对应部分	秩
油墨百分比	27.3767	0.1609	1
溶剂百分比	14.5981	0.0858	2
印刷机	14.0203	0.0824	3
上光油百分比	13.6577	0.0803	4
辊硬度	11.8470	0.0696	5
印刷速度	11.0130	0.0647	6
ESA 电压	7.1478	0.0420	7
粘度	6.9979	0.0411	8
印刷机类型	6.5136	0.0383	9
ESA 电流	6.4143	0.0377	10
粒度过滤	4.6011	0.0270	11
单元编号	4.3955	0.0258	12
湿度	4.1744	0.0245	13
刮刀压力	4.0893	0.0240	14
纸张产地	3.7328	0.0219	15
油墨类型	3.6759	0.0216	16
油墨温度	3.4103	0.0200	17
活字滚筒	2.8272	0.0166	18
裁样	2.8073	0.0165	19
硬化剂	2.7894	0.0164	20
阳极区比例	2.2329	0.0131	21
电流密度	1.8053	0.0106	22
纸张厚度	1.6249	0.0095	23
滚筒类型	1.3343	0.0078	24
刮刀制造商	1.2237	0.0072	25
粗糙度	1.1170	0.0066	26
涂布油墨打样	1.1017	0.0065	27
纸张类型	1.0461	0.0061	28
镀槽	0.9256	0.0054	29
蜡	0.8162	0.0048	30
溶剂类型	0.7797	0.0046	31
直接蒸汽	0.0556	0.0003	32
铬含量	0.0000	0.0000	33

注意：因为该分析基于有随机选择成分的 Bootstrap 森林法，您的结果可能与图 24.2 中的结果略有不同。请参见第 81 页的“Bootstrap 森林法”。

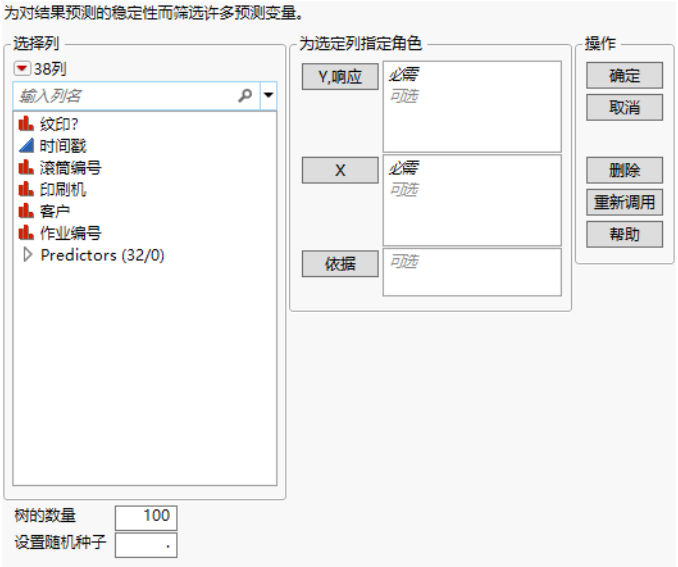
按对 Bootstrap 森林法模型的贡献对这些列排序和排名。具有最大贡献的预测变量是相关响应的强候选预测变量。

提示：点击列标题可更改列在表中的排序方式。

启动“预测变量筛选”平台

通过选择**分析** > **筛选** > **预测变量筛选**来启动 “预测变量筛选”平台。

图 24.3 “预测变量筛选”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，响应 响应列。

X 预测变量列。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据”变量，将为 “依据”变量水平的每种可能组合生成单独的报表。

树的数量 Bootstrap 森林法中的决策树的数目。

设置随机种子 设置拟合过程中使用的起始值的种子。若您要重现分析，“设置随机种子”很有用。若您设置一个随机种子并保存脚本，该种子将自动保存在脚本中。

“预测变量筛选”报表

报表（图 24.2）显示具有相应贡献和排名的预测变量列表。具有最大贡献的预测变量可能对预测 Y 很重要。

“贡献”列显示每个预测变量对 Bootstrap 森林法模型的贡献。报表中的“对应部分”列显示每个变量的百分比贡献。

您可以在“预测变量筛选”报表中选择重要的预测变量。选择重要的预测变量会在数据表中选择相应列，使您可以轻松将这些列输入建模平台的启动窗口。“复制选定”按钮支持您将选定列复制到剪贴板，以便您也能将其粘贴回模型。“预测变量筛选”以这种方式改进了建模过程。

“预测变量筛选”平台选项

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

第 25 章

JMP PRO 关联分析

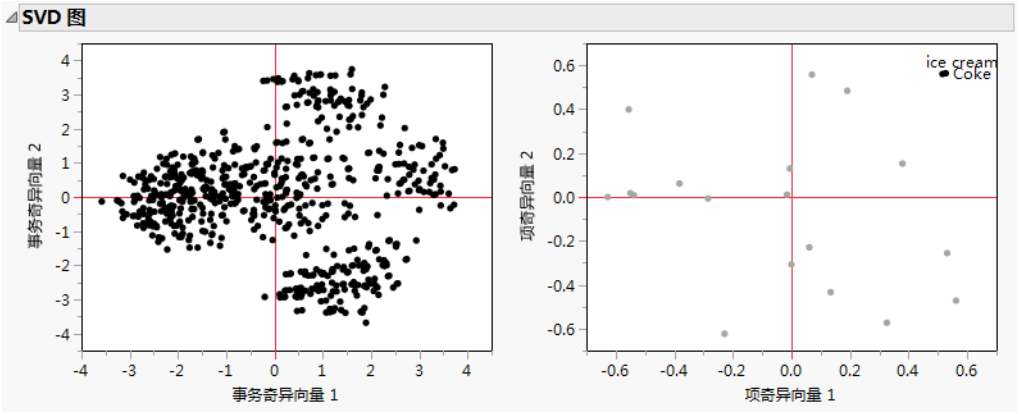
执行市场购物篮分析

JMP PRO 仅在 JMP Pro 中提供 “关联分析”平台。

关联分析支持您识别彼此有密切关联的项。该分析经常用于分析交易数据（亦称**市场购物篮**），用以识别在交易中经常一起出现的项。例如，杂货店和在线商家使用关联分析战略性地组织和推荐往往一起购买的产品。

关联分析还用于识别从属的或关联的事件。例如，您可以识别看起来基本上同时发生故障的汽车零件。在这种情况下，汽车检修可被视为市场购物篮，您需要分析各次检修中发现的各组故障零件之间的关联。

图 25.1 奇异值分解图的示例



目录

“关联分析”平台概述 445
“关联分析”平台示例 445
启动“关联分析”平台 448
数据格式 449
“关联分析”报表 449
频繁项集 449
规则 450
“关联分析”平台选项 450
奇异值分解 451
SVD 报表 451
SVD 报表选项 452
主题分析 453
“主题分析”报表 453
“主题分析”报表选项 454
更多示例：SVD 分析 454
“关联分析”平台的统计详细信息 456
生成常用项集 456
关联分析性能测度 456

JMP PRO “关联分析”平台概述

“关联分析”平台识别独立事件或**交易**中各组项之间的关系。在关联分析中，**项**是基本的相关对象。例如，项可以是产品、网页或服务。**项集**是一个或多个项的列表。

两个项集之间的关系由**关联规则**来定义。关联规则由**条件**项集和**结果**项集构成。**前项**是条件项集中的单个项。关联分析识别关联规则，这些规则预测在某单交易中已经包含条件项集的前提下，结果项集也会出现在该交易中。某些关联规则作用较强，因而比其他规则更有用。以下三个性能测度描述关联规则的强度：

- **支持度**是其中出现项集的交易的比例。较高的支持度值指示该项集经常出现。
- **置信度**是在交易包含条件项集的前提下，包含结果项集的交易的比例。置信度测量关联规则的隐含强度或预测能力。请注意，关联分析中的置信度与置信区间的概念无关。
- **提升度**是关联规则的置信度与其期望置信度的比率，假设前提是：条件项集与结果项集在交易中出现的概率是独立的。提升度测量结果项集在多大程度上依赖于条件项集的存在。提升度的最小值为 0。
  - 小于 1 的提升比指示条件和结果彼此排斥，因为它们一起出现的频率小于其单独偶然出现的频率。
  - 接近 1 的提升比指示结果出现在包含条件的交易中的频率与其单独偶然出现的频率相同。
  - 大于 1 的提升比指示结果项集与条件项集具有密切关联。结果项集与条件项集一起出现的频率比其单独偶然出现的频率要高。

有关这些性能测度的详细信息，请参见第 456 页的“关联分析性能测度”。

“关联分析”平台还支持您执行奇异值分解。奇异值分解 (SVD) 使用不同于关联分析中所用方法的矩阵简化方法对相似交易以及相似项进行分组。使用 SVD 方法获得的深入分析可以为从关联分析中了解到的信息提供有益补充。

有关关联分析的详细信息，请参见 Hastie et al. (2009) 和 Shmueli et al. (2010)。有关奇异值分解的详细信息，请参见 Jolliffe (2002)。

JMP PRO “关联分析”平台示例

本例使用 Grocery Purchases.jmp 样本数据表，其中包含一家杂货店报告的交易数据。该数据表列出 1001 个客户购买的商品项，为每一个客户都分配了唯一的客户 ID。您想要探索项之间的关联以辨识客户行为模式。

1. 选择**帮助** > **样本数据库**，然后打开 Grocery Purchases.jmp。
2. 选择**分析** > **筛选** > **关联分析**。
3. 选择**产品**并点击**项**。
4. 选择**客户 ID** 并点击 **ID**。

5. 点击**确定**。

   默认情况下，“规则”报表按**置信度**降序排序。不过，具有极高置信度的关联规则还往往在条件集中具有较高的项数。由于您想要查看具有较小条件集的关联规则，请按照**置信度**对报表排序，但按照升序排序。

6. 在 “规则”报表中右击并选择**按列排序**。

   随即显示 “选择列”窗口。

7. 选择**置信度**，然后选中 “升序”选项。

8. 点击**确定**。

图 25.2 “关联分析”报表

规则

条件	结果	置信度	提升度
apples	sardines	40%	1.357
Coke	turkey	40%	1.422
sardines	chicken, Heineken	40%	2.002
herring, olives	ham, turkey	40%	3.006
ice cream	turkey	40%	1.424
artichoke	baguette, Heineken	40%	1.547
ham	corned beef, herring	40%	1.648
ham	corned beef, olives	40%	1.703
peppers	sardines	41%	1.371
sardines	peppers	41%	1.371
Coke	Heineken, sardines	41%	2.23
sardines	Coke, Heineken	41%	2.332
avocado, Heineken	crackers, ham	41%	2.707
soda	baguette, herring	41%	1.631
corned beef	ham	41%	1.343
Heineken, soda	bourbon, crackers, olives	41%	3.152
artichoke, Heineken	avocado, baguette	41%	1.921
Heineken, herring	avocado, baguette	41%	1.924
apples	peppers	41%	1.4
Coke	chicken, ice cream	42%	2.971
sardines	Heineken, ice cream	42%	2.261
herring	corned beef, olives	42%	1.755
corned beef, herring	ham, turkey	42%	3.11
corned beef, herring	ham, olives, turkey	42%	3.444
Heineken, soda	bourbon, olives	42%	1.701
avocado, Heineken	artichoke, baguette	42%	3.097

9. 在报表中向下滚动到**置信度**值为 58% 的位置。

   “规则”报表表中有一个条目指示 58% 的客户在购买了鳄梨的同时也购买了洋蓟。提升值为 1.908，这指示存在可能的依赖性。您想要验证有很大一部分交易都会同时出现鳄梨和洋蓟。

10. 点击 “频繁项集”旁边的展开图标。

图 25.3 “频繁项集”报表

频繁项集

项集	支持度	N 个项
{Heineken}	60%	1
{crackers}	49%	1
{herring}	49%	1
{olives}	47%	1
{bourbon}	40%	1
{baguette}	39%	1
{corned beef}	39%	1
{crackers, Heineken}	37%	2
{avocado}	36%	1
{soda}	32%	1
{chicken}	31%	1
{apples}	31%	1
{ice cream}	31%	1
{artichoke}	30%	1
{ham}	30%	1
{Coke}	30%	1
{peppers}	30%	1
{sardines}	30%	1
{Heineken, herring}	29%	2
{turkey}	28%	1
{baguette, Heineken}	26%	2
{Heineken, soda}	26%	2
{herring, olives}	26%	2
{artichoke, Heineken}	25%	2

“频繁项集”报表显示 36% 的客户购买了鳄梨。图 25.2 中的 “规则”报表显示这些客户中有 58% 的客户也购买了洋蓟。由于较大一部分客户都存在该行为，杂货店管理人员可以使用该信息战略性地对鳄梨和洋蓟进行位置摆放。

您还决定查看具有最高提升值的关联规则。

11. 在 “规则”报表表中右击并选择**按列排序**。

    随即显示 “选择列”窗口。

12. 选择**提升度**并点击**确定**。

    “规则”表按照提升值降序排列。请注意，第二个关联规则的提升值为 5.642，置信度为 83%。您想要验证条件集 {chicken, ice cream} 和结果项集 {Coke, sardines} 都具有足够的支持度。

13. 在 “频繁项集”报表中右击并选择**按列排序**。

    随即显示 “选择列”窗口。

14. 选择**项集**，然后选中 “升序”选项。

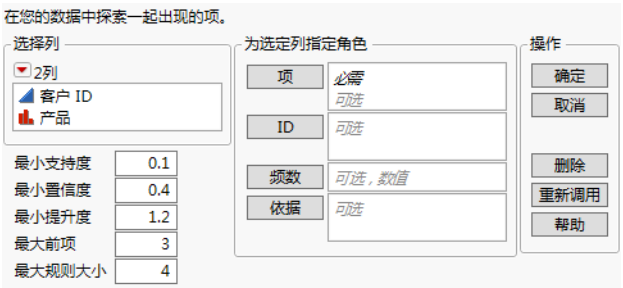
15. 点击**确定**。

    “频繁项集”表按项集的字母顺序排序。在列表中滚动浏览，直到看到条件项集 {chicken, ice cream} 的支持度为 14%。结果项集 {Coke, sardines} 的支持度为 15%。该关联规则有较高的提升值，但和您查看的第一个关联规则比，其表示的交易要少的多。

JMP PRO 启动“关联分析”平台

通过选择**分析** > **筛选** > **关联分析**启动“关联分析”平台。

图 25.4 “关联分析”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

项 一个或多个分类列，其中包含要分析的项数据。若列的建模类型为“多重响应”，则每行中的多重响应都被视为一项。若指定多列，则每行中的每列都被视为一个项。

ID 标识某项所属的交易的列。ID 角色是必需的，除非“项”角色填充了建模类型为“多重响应”的列或是“项”角色填充了多个列。

注意：若指定 ID 角色，则具有相同 ID 的所有行都组合为一个交易。

频数 标识“项”列中交易频数的列。例如，若某个交易的频数为 3，则将其视为包含“项”列中的项的三个相同的交易。

注意：若“项”角色填充的列的建模类型不是“多重响应”，则忽略“频数”角色。

依据 为“依据”变量的每个水平生成单独报表。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

最小支持度 指定某个项集出现的比例的最小值。该值必须介于 0 到 1 之间。只有支持度等于或超过该值的项集才会在分析中加以考虑。

最小置信度 指定在包含条件项集的交易中，结果项集出现的比例的最小值。该值必须介于 0 到 1 之间。只有置信度等于或超过该值的关联规则才会出现在报表中。

最小提升度 指定最小依赖性比率。提升度值必须是为 0 或 0 以上的值。只有提升度值等于或超过该值的关联规则才会出现在报表中。

最大前项 指定条件项集中的最大项数。条件集中的项数超过该项数的关联规则不会在分析中加以考虑。

最大规则大小 指定出现在条件和结果项集的并集中的最大项数。超过该合并项数的关联规则不会在分析中加以考虑。默认值为 4。

注意：您可以在启动窗口中使用最小支持度、最大前项和最大规则大小等选项来减少大型数据集的计算时间。有关这些测度的详细信息，请参见第 456 页的““关联分析”平台的统计详细信息”。

数据格式

“关联分析”平台接受三种数据格式：

- 您可以在“项”角色中的每行中指定一个项响应，并且使用 ID 角色标识包括在每个交易中的项。在这种格式中，“项”和 ID 角色是必需的，“频数”角色将被忽略。
- 您可以为“项”角色指定“多重响应”建模类型列。在这种格式中，“项”角色是必需的，ID 和“频数”角色是可选的。若指定 ID 角色，则具有相同 ID 的所有行都组合为一个交易。若指定“频数”角色，则将频数应用到交易而不是交易中的特定项。
- 可以在“项”角色中指定多个项响应列。在这种格式中，ID 和“频数”角色是可选的。若指定 ID 角色，则具有相同 ID 的所有行都组合为一个交易。若指定“频数”角色，则将频数应用到交易而不是交易中的特定项。

JMP PRO “关联分析”报表

默认情况下，“关联分析”报表包含以下报表：

- 第 449 页的“频繁项集”
- 第 450 页的“规则”

提示：要按照其中任意列对报表中的表内容排序，请在该表中右击并选择**按列排序**。

JMP PRO 频繁项集

“频繁项集”报表按照支持度降序列出各个项集。列出的项集满足您在启动窗口中指定的 “最小支持度”值。每个项集都被视为一个条件项集或一个结果项集以便构成关联规则。该表包含以下列：

项集 被视为关联规则的条件项集或结果项集的项集。

支持度 交易中项集中的所有项都出现的比例。

N 个项 项集中的项数。

JMP PRO 规则

“规则”报表显示以置信度的降序排序的关联规则表。只有满足您在启动窗口中指定的“最小支持度”、“最小置信度”、“最小提升度”、“最大前项”和“最大规则大小”等要求的关联规则才会显示在该报表中。

“规则”报表表包含以下列：

规则 通过合并条件项集和结果项集构成的关联规则。

条件 被认为可影响交易中结果项集的出现的项集。

结果 认为由于条件项集的出现而被影响其出现的项集。

置信度 在交易包含条件项集的前提下，包含结果项集的交易的比例。置信度测量关联规则的隐含强度或预测能力。

提升度 关联规则的置信度与其期望置信度的比率，假设前提是：条件项集与结果项集均出现在交易中，且彼此无关联。提升度测量结果项集在多大程度上依赖于条件项集的存在。提升度的最小值为 0。

- 小于 1 的提升比指示条件项集和结果项集彼此排斥，因为它们一起出现的频率小于偶然出现的频率。
- 接近 1 的提升比指示结果项集出现在包含条件项集的交易中的频率与偶然出现的频率相同。
- 大于 1 的提升比指示结果项集与条件项集具有密切关联。结果项集与条件项集一起出现的频率比其单独偶然出现的频率要高。

JMP PRO “关联分析”平台选项

“关联分析”红色小三角菜单包含以下选项：

交易列表 显示或隐藏一个表，其中列出每个交易 ID 值和包含在该交易中的项。该表按照“交易 ID”列排序。

频繁项集 显示或隐藏项集列表，这些项集的支持度超过在启动窗口中指定的 “最小支持度”值。请参见第 449 页的 “频繁项集”。

规则 显示或隐藏关联规则表，这些规则满足您在启动窗口中指定的 “最小支持度”、“最小置信度”、“最小提升度”、“最大前项”和 “最大规则大小”等要求。请参见第 450 页的 “规则”。

SVD 显示或隐藏项的关联矩阵的偏奇异值分解 (SVD) 报表。这种分解将关联矩阵精简为用户指定的维数，以便进行分析。SVD 报表还包含用于可视化奇异向量、对交易和项进行聚类、向外保存奇异向量以及执行旋转 SVD 分析的选项。请参见第 451 页的 “奇异值分解”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO 奇异值分解

奇异值分解 (SVD) 通过提供另一种方法来识别彼此具有密切关联的项，从而对关联分析进行了补充。交易项矩阵的奇异值分解将矩阵简化为数量可控的若干维，从而支持您对相似交易和相似项分组。SVD 分析等同于对相关性矩阵执行主成分分析 (PCA)。

交易项矩阵是这样一个矩阵：其中的每行对应一单交易，每列对应一项。矩阵条目由 0 和 1 构成。若某项出现在某个交易中，相应的行和列条目为 1。否则，行和列条目为 0。由于在交易项矩阵包含的值中，0 通常多于 1，所以称之为**稀疏矩阵**。

偏奇异值分解使用三个矩阵来近似列标准化的交易项矩阵：**U**、 **S** 和 **V**′。这三个矩阵的关系定义如下：

交易项矩阵 ≈ **U** * **S** * **V**′

nTran 定义为交易项矩阵中的交易（行）数，*nItems* 定义为交易项矩阵中的项（列）数，*nVec* 定义为指定的奇异向量数。请注意，*nVec* 必须小于或等于 min(*nTran, nItem*)。由此判定：**U** 是一个 *nTran* x *nVec* 矩阵，该矩阵包含交易项矩阵的左奇异向量。**S** 是维 *nVec* 的对角矩阵。**S** 中的对角线元素是交易项矩阵中的奇异值。**V**′ 是 *nVec* * *nItem* 矩阵。**V**′ 中的行（或 **V** 中的列）是右奇异向量。

右奇异向量可以获取具有相似功能或主题领域的不同项之间的关联。若三个项倾向于出现在相同的交易中，SVD 可能在 **V**′ 中生成一个奇异向量，其中这三个项对应的值都很大。**U** 奇异向量表示交易在新的项空间的投影。

SVD 还获取间接关联。若两个项从不一起出现在同一交易中，但是它们通常出现在具有第三个项的交易中，则 SVD 可以获取一些这样的关联。若两单交易没有相同的项但是其包含的项在降维空间中有关联，则它们在 SVD 图中映射的向量会很类似。

SVD 将交易数据变换为固定维的向量空间，使它适用于聚类、分类和回归技术。使用“保存”选项可以将这个向量空间导出到其他 JMP 平台去分析。

在执行奇异值分解之前，将对交易项矩阵进行中心化和统一尺度，并在除以 *nTran* 后减 1。该分析等同于交易项矩阵的相关性矩阵的 PCA。SVD 实现利用交易项矩阵的稀疏性。

JMP PRO SVD 报表

SVD 选项从奇异值分解生成两个 SVD 图和一个奇异值表。

JMP PRO SVD 图

第一个图为每个交易都包含一个点。对于给定的交易，标绘的点由前两个奇异向量（**U** 矩阵的前两列）中的交易值与对角线奇异值矩阵 (**S**) 的乘积来定义。该图等同于“主成分”平台中的得分图。该图中的每个点都表示一个交易。在交易 SVD 图中，明显分组在一起的点表明这些交易的构成非常相似。

第二个图为每个项都包含一个点。对于给定项，标绘的点由前两个奇异向量（**V′** 矩阵的前两行）中的项值与对角线奇异值矩阵 (**S**) 的乘积来定义。该图等同于“主成分”平台中的“载荷图”。该图中的每个点都表示一个项。在项 SVD 图中，明显分组在一起的项指示项具有相似的功能或主题领域。

JMP PRO 奇异值

在交易和项 SVD 图下方，显示一个奇异值表。这些值是交易项矩阵的奇异值分解中的 **S** 矩阵的对角线元素。“奇异值”表还包含等价的主成分分析的相应特征值列。与在“主成分”平台中类似，每个特征值（或奇异值）所解释的变异的百分比和累积百分比都会有相应的列。您可以使用“累积百分比”列来决定您想要保留的交易项矩阵的方差百分比，然后使用相应数量的奇异向量。

JMP PRO SVD 报表选项

“SVD”红色小三角菜单包含以下选项：

SVD 散点图矩阵 显示或隐藏项和交易奇异值分解向量的散点图矩阵。选择该选项时，您必须选择散点图矩阵的大小。该散点图矩阵支持您直观演示奇异值分解中的不止前两个维。

主题分析，旋转 SVD 执行项交易矩阵的可变最大旋转偏奇异值分解，以生成称为主题的项组。您可以多次选择该选项以查找不同的主题数。请参见第 453 页的 “主题分析”。

聚类项 显示或隐藏数据中项的层次聚类分析。系统树图右侧提供了以下选项：

设置聚类 指定在系统树图中着色并在“保存聚类”选项中使用的聚类数。“项 SVD”图中的点也按聚类着色。

保存聚类 保存包含项及其聚类分配的数据表。在点击“保存聚类”之前，必须设置聚类数。

有关聚类和系统树图的详细信息，请参见 《多元方法》。

聚类交易 显示或隐藏数据中交易的层次聚类分析。系统树图右侧提供了以下选项：

设置聚类 指定在系统树图中着色并在“保存聚类”选项中使用的聚类数。“交易 SVD”图中的点也按聚类着色。

保存聚类 保存包含交易及其聚类分配的数据表。在点击“保存聚类”之前，必须设置聚类数。

有关聚类和系统树图的详细信息，请参见 《多元方法》。

保存交易奇异向量 将来自交易奇异值分解的用户指定的奇异向量数作为列保存到一个新数据表，该表的每行对应一个交易。前两个保存的列表示在交易 SVD 图中绘制的点。请参见第 451 页的 “奇异值分解”。

保存项奇异向量 将来自项奇异值分解的用户指定的奇异向量数作为列保存到一个新数据表，该表的每行对应一个项。前两个保存的列表示在项 SVD 图中绘制的点。请参见第 451 页的 “奇异值分解”。

JMP PRO 主题分析

“主题分析，旋转 SVD”选项对交易项矩阵的偏奇异值分解 (SVD) 执行最大方差法旋转。请参见第 451 页的 “奇异值分解”。您必须指定旋转奇异向量的数目，该数目对应于您想要保留的交易项矩阵中的主题数。指定主题数后，“主题分析”报表随即显示。

主题分析等同于旋转主成分分析（旋转 PCA）。最大方差法旋转选取一组奇异向量并旋转它们，以使其更直接地指向坐标方向（朝向项）。这种旋转使得向量有助于解释交易的构成，因为每个旋转向量都朝向一组项。负值指示排斥力。与具有正值的项相比，具有负值的项在主题中出现的频率较低。

JMP PRO “主题分析”报表

“主题分析”报表显示旋转后的每个主题中具有最大载荷的项。还有其他显示旋转奇异值分解的成分的报表。

“按主题划分的前几位载荷”报表为每个主题都显示一个表。每个表中的项是对每个主题而言具有最大绝对值载荷的项。按载荷绝对值的降序对每个表排序。可以使用这些表来确定对应于每个主题的概念性主题。

“主题分析”报表还包含以下报表：

主题载荷 包含跨各项的主题的载荷矩阵。该矩阵等同于旋转 PCA 中的因子载荷矩阵。

按主题划分的词云 包含词云矩阵，每个主题一个词云。

主题得分 包含每个主题的交易得分的矩阵。主题中具有更高得分的交易更可能与该主题关联。

主题得分图 以一维散点图形式显示所有交易的主题得分。负值指示与主题具有负关联的交易。使用这些图可探索每个主题内的交易分布。

每个主题解释的方差 包含由每个主题解释的方差表。该表还包含每个主题解释的变异的百分比和累积百分比列。

旋转矩阵 包含最大方差法旋转的旋转矩阵。

JMP PRO “主题分析”报表选项

“主题分析”红色小三角菜单包含以下选项：

主题散点图矩阵 显示或隐藏旋转奇异值分解向量的散点图矩阵。

显示选项 包含用于显示或隐藏“主题分析”报表中显示的内容的选项。请参见第 453 页的“‘主题分析’报表”。

重命名主题 显示一个对话框，该对话框支持您为一个或多个主题添加说明性名称。

保存交易主题向量 将旋转奇异值分解中的用户指定的奇异向量数作为列保存到数据表。

保存项主题向量 将主题向量作为列保存到新的“项主题得分”数据表中。

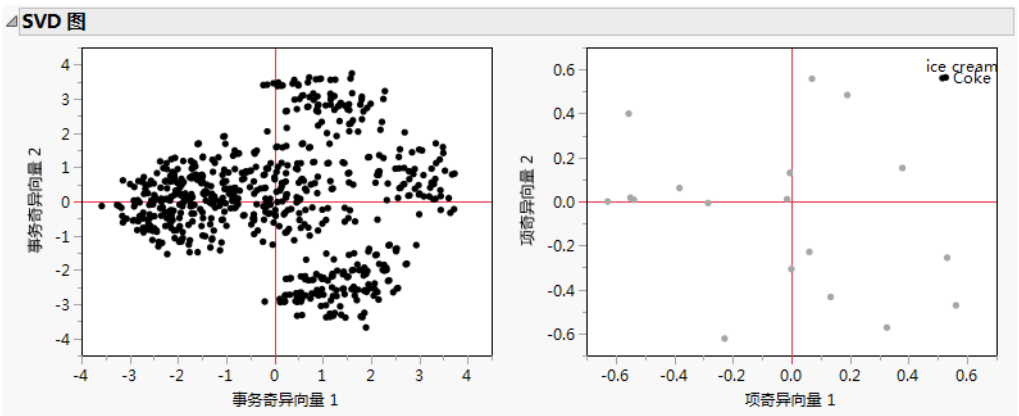
删除 从 SVD 报表中删除“主题分析”报表。

JMP PRO 更多示例：SVD 分析

在本例中，您使用交易项矩阵的奇异值分解来获取关于 Grocery Purchases.jmp 样本数据的进一步深入的信息。

1. 选择**帮助** > **样本数据库**，然后打开 Grocery Purchases.jmp。
2. 选择**分析** > **筛选** > **关联分析**。
3. 选择**产品**并点击**项**。
4. 选择**客户 ID** 并点击 **ID**。
5. 点击**确定**。
6. 点击“关联分析”红色小三角并选择 **SVD**。
7. 点击**确定**。

图 25.5 SVD 图



交易 SVD 图表明可能有两个或三个交易组。在项 SVD 图的右上角，请注意表示可乐和冰激淋的点重叠在一起。这两个项的邻近关系指示它们之间存在很强的密切关联。

8. 点击 “SVD”红色小三角并选择**主题分析，旋转 SVD**。
9. 在 “主题数 （旋转奇异向量）”旁边键入 3，然后点击**确定**。

“主题项”报表和 “主题得分”报表随即显示。

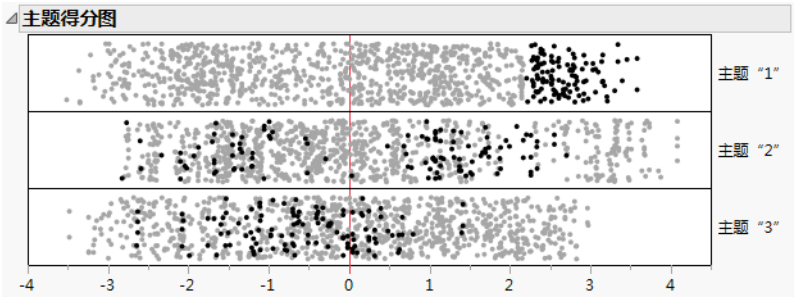
图 25.6 “主题项”报表

按主题划分的前几位载荷

主题 “1”		主题 “2”		主题 “3”	
项	载荷	项	载荷	项	载荷
avocado	-0.7191	Coke	0.7017	crackers	0.7050
olives	0.7142	ice cream	0.6945	apples	-0.6437
baguette	-0.6089	herring	-0.6550	soda	0.6411
turkey	0.5697	sardines	0.6122	Heineken	0.6409
artichoke	-0.4889	chicken	0.5158	sardines	-0.3618
bourbon	0.4860	corned beef	-0.4257	steak	-0.3468
Heineken	-0.4329	steak	-0.3395	corned beef	-0.3409
corned beef	0.3836	olives	-0.3035	bourbon	0.3340
Coke	0.3096	ham	-0.3022		

在 “主题项”报表中创建并显示了三个组 （或主题）。列在 “主题项”表中的项表示该组构成的主要项。例如，“主题 1”是这样一个组，该组主要由不含鳄梨 （但包含橄榄）的交易来标识。

图 25.7 主题得分



为 1001 个交易中的每一单交易分配的主题得分都标绘在 “主题得分”报表中。选择某一主题的各组点可查看这些交易与其他主题如何相关。例如，在 “主题 1”上有极高值的交易往往在 “主题 2”和 “主题 3”上的值很低。

10. 打开 “奇异值”报表。

图 25.8 “奇异值”表

奇异值

数目	奇异值	特征值	百分比	累积百分比
1	1.7399	3.0272	15.1362	15.1362
2	1.6494	2.7204	13.6021	28.7383
3	1.5093	2.2780	11.3901	40.1284
4	1.4077	1.9816	9.9079	50.0363
5	1.2565	1.5788	7.8938	57.9301
6	1.1331	1.2839	6.4196	64.3497
7	1.0333	1.0676	5.3381	69.6877
8	0.9927	0.9854	4.9270	74.6148
9	0.9265	0.8583	4.2916	78.9063
10	0.7888	0.6223	3.1114	82.0177
11	0.7247	0.5253	2.6263	84.6440
12	0.6647	0.4419	2.2094	86.8534
13	0.6465	0.4179	2.0897	88.9431
14	0.6379	0.4069	2.0346	90.9777
15	0.6266	0.3927	1.9634	92.9411
16	0.6140	0.3770	1.8848	94.8259
17	0.6077	0.3693	1.8465	96.6724
18	0.5817	0.3383	1.6917	98.3641
19	0.5590	0.3125	1.5623	99.9264
20	0.1213	0.0147	0.0736	100.000

如图 25.8 所示，前两个奇异值只解释杂货店数据中大约 30% 的变异性。可能需要更多的维来解释足够的变异性。

JMP PRO “关联分析”平台的统计详细信息

- “生成常用项集”
- “关联分析性能测度”

JMP PRO 生成常用项集

“关联分析”平台使用 Apriori 算法减少生成频繁项集时所用的计算时间。Apriori 算法利用了这样一个事实：某个项集的支持度绝不会大于其各个子集的支持度。该平台通过组合符合最小支持水平的较小项集来生成较大的项集。此外，该平台不会生成超过指定的最大前项数或最大规则大小的项集。在处理大型数据集时这些选项很有用，因为可能的总规则数会随着项数呈指数增长。有关 Apriori 算法的详细信息，请参见 Agrawal and Srikant (1994)。

JMP PRO 关联分析性能测度

本节定义“关联分析”平台中使用的性能测度。使用条件项集 X 和结果项集 Y 通过 $X \Rightarrow Y$ 的形式来表示关联规则。Hahsler (2015) 包含关联分析中使用的性能测度集合，包括支持度、置信度和提升度。

JMP PRO 支持度

支持度是其中出现项集的交易的比例。也可以将支持度视为交易包含项集的概率。

条件项集 X 的支持度 S 定义如下：

$$supp(X) = \frac{N_X}{N} = Pr(X)$$

其中：

N_X 是包含项集 X 的交易数。

N 是总交易数。

关联规则的支持度定义如下：

$$supp(X \Rightarrow Y) = supp(X \cup Y) = Pr(X \cap Y)$$

在这种情况下，支持度相当于交易同时包含项集 X 和 Y 的概率。

对于项集和关联规则这两种情况，支持度均介于 0% 到 100% 之间。

JMP PRO 置信度

置信度是在交易包含条件项集的前提下，包含结果项集的交易的比例。也可以将置信度视为在交易包含条件项集的前提下，该交易包含结果项集的条件概率。

$$conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)} = \frac{Pr(X \cap Y)}{Pr(X)} = Pr(Y|X)$$

置信度介于 0% 到 100% 之间。置信度为 0% 的关联规则的结果项集不会在任何交易中与条件项集一同出现。置信度为 100% 指示包含条件项集的每个交易都同时包含结果项集。

注意：关联分析中的置信度与置信区间的概念无关。

JMP PRO 提升度

提升度测量的是 X 与 Y 之间的依赖性。

$$lift(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X) \times supp(Y)} = \frac{Pr(X \cap Y)}{Pr(X)Pr(Y)}$$

提升度的分子是 X 与 Y 同时出现的交易的比例。分母是 X 与 Y 同时出现的期望概率的估计值（假定它们的出现无关联）。

提升度介于 0 到 ∞ 之间。提升度值为 1 指示 X 和 Y 同时在交易中出现的频率，该频率被认为是仅仅出于偶然。增大的提升值表明在存在 X 的前提下，Y 发生的概率比期望值高。

注意：关联规则 $X \Rightarrow Y$ 的提升度等于关联规则 $Y \Rightarrow X$ 的提升度。

第 26 章

过程历史分析器

标识复杂过程历史中的问题构成

注意：“过程历史分析器”平台是新增的试用平台。在将来的发行版本中会随时变化。

在复杂的生产情形中（如：在半导体工厂中），有缺陷的工具或路线组合可能导致产量不足或质量下降。不过，某个单元经历的整个生产过程中可能会涉及到成百上千的步骤，并使用许多不同的工具。所以很难确定导致产量不足的确切因素组合。

“过程历史分析器”平台将所有生产跟踪数据组织起来，并支持您标识看起来与产量不足有关的因素。您可以随后执行逐步回归。可以分析某个单元在过程各个步骤或工具之间等待的时间，以便查看等待时间如何与产量相关。您还可以分析从一个工具转换到另一个工具对产量的影响。

图 26.1 “过程历史分析器”报表

过程历史分析器

1250 个单元，使用 ID columns: 批, 晶片
125 个步骤，使用“columns”：层, 操作
156250 个操作，使用时间“columns”：开始时间, 停止时间
X columns 中的 195 个水平: 工具, 路线
1 response: 产量

响应“产量”

具有最低产量的水平

项	水平	计数	均值	标准误差	标准差
工具	PlanarTool008	1000	0.75299	0.00188	0.05952
工具	DepTool003	1375	0.75577	0.00168	0.06245
工具	EtchTool019	750	0.76192	0.0021	0.05749
工具	EtchTool012	500	0.76203	0.0026	0.05807
工具	EtchTool034	850	0.76279	0.00199	0.0581
工具	PlanarTool021	1050	0.7637	0.00186	0.06028
工具	PlanarTool030	1025	0.76539	0.00184	0.05894
工具	EtchTool016	900	0.76566	0.00191	0.05734
工具	EtchTool009	925	0.76598	0.00197	0.05994
工具	Dep2Tool018	1525	0.76766	0.00156	0.06088
工具	Dep2Tool014	1825	0.76813	0.00138	0.05893
工具	DepTool019	1650	0.76839	0.00143	0.05794
工具	EtchTool020	1000	0.76909	0.00177	0.05586
工具	Dep2Tool015	1475	0.76948	0.0015	0.05775
工具	PhotoTool010	3525	0.76996	0.00098	0.05822
工具	Dep2Tool013	1825	0.7705	0.00141	0.06025
工具	DepTool012	1750	0.77063	0.00145	0.06051
工具	Dep2Tool006	1400	0.77063	0.00158	0.05905
工具	PlanarTool028	900	0.77103	0.00202	0.06071
工具	Dep2Tool011	1600	0.77124	0.00145	0.05797

目录

过程历史分析器的示例 461
启动“过程历史分析器”平台 463
数据格式 464
“过程历史分析器”报表 464
“过程历史分析器”平台选项 464

过程历史分析器的示例

本例使用两个数据表。第一个数据表包含经历了复杂生产过程的 50 个批次中 25 个晶片的分步历史记录。第二个数据表包含每批中每个晶片的产量。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control > Lot Wafer History.jmp 和 Quality Control > Lot Wafer Yield.jmp。
2. 从 Lot Wafer History.jmp 数据表中，选择**分析 > 筛选 > 过程历史分析器**。
3. 选择**批**和**晶片**，然后点击 ID。
4. 选择**工具**和**路线**，然后点击 X，**过程**。
5. 选择**层**和**操作**，然后点击**步骤**。
6. 选择**开始时间**和**停止时间**，然后点击**时间戳**。
7. 点击**确定**。

   随即显示一个窗口，支持您选择相应的产量表。
8. 选择 Lot Wafer Yield，然后点击**确定**。

   随即显示一个窗口，支持您选择产量列。
9. 选择**产量**并点击**确定**。
10. 点击 “过程历史分析器”红色小三角并选择**具有最低产量的水平**。

图 26.2 “过程历史分析器”报表

过程历史分析器

1250 个单元，使用 ID columns: 批, 晶片
125 个步骤，使用 “columns” : 层, 操作
156250 个操作，使用时间 “columns” : 开始时间, 停止时间
X columns 中的 195 个水平: 工具, 路线
1 response: 产量

响应 “产量”

具有最低产量的水平

项	水平	计数	均值	标准误差	标准差
工具	PlanarTool008	1000	0.75299	0.00188	0.05952
工具	DepTool003	1375	0.75577	0.00168	0.06245
工具	EtchTool019	750	0.76192	0.0021	0.05749
工具	EtchTool012	500	0.76203	0.0026	0.05807
工具	EtchTool034	850	0.76279	0.00199	0.0581
工具	PlanarTool021	1050	0.7637	0.00186	0.06028
工具	PlanarTool030	1025	0.76539	0.00184	0.05894
工具	EtchTool016	900	0.76566	0.00191	0.05734
工具	EtchTool009	925	0.76598	0.00197	0.05994
工具	Dep2Tool018	1525	0.76766	0.00156	0.06088
工具	Dep2Tool014	1825	0.76813	0.00138	0.05893
工具	DepTool019	1650	0.76839	0.00143	0.05794
工具	EtchTool020	1000	0.76909	0.00177	0.05586
工具	Dep2Tool015	1475	0.76948	0.0015	0.05775
工具	PhotoTool010	3525	0.76996	0.00098	0.05822
工具	Dep2Tool013	1825	0.7705	0.00141	0.06025
工具	DepTool012	1750	0.77063	0.00145	0.06051
工具	Dep2Tool006	1400	0.77063	0.00158	0.05905
工具	PlanarTool028	900	0.77103	0.00202	0.06071
工具	Dep2Tool011	1600	0.77124	0.00145	0.05797

该报表包含有关过程的信息，其中包括 X 列中的单元、步骤、操作和水平的数目。“具有最低产量的水平”表包含关于 X 列中 195 个水平的计数和产量信息。请注意，PlanarTool008 工具具有最低产量，有 1,000 个单元使用了该工具。

启动“过程历史分析器”平台

通过选择**分析** > **筛选** > **过程历史分析器**启动“过程历史分析器”平台。

图 26.3 “过程历史分析器”启动

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

启动窗口角色

ID 指定标识过程中的单元的一个或多个列。

注意：ID 列必须在产量表中有对应列。

X，过程 指定一个或多个列，这些列标识构成过程各步的组成部分或特定详细信息。

步骤 可选列，用于指定某个单元经过的步骤或子步骤。例如，层或层内的操作类型。

时间戳 指定操作时间。该时间可指定为单个时间点，或指定为指示操作起止时间的一对时间点。

顺序 可选列，用于指定一批单元经过“X，过程”角色中指定的各个组件的顺序。

依据 为“依据”变量的每个水平生成单独报表。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

启动窗口选项

目标是最小化 Y 指定分析的目标是将响应最小化。默认情况下，分析的目的是要将响应最大化。

注意：响应是在第二个数据表中指定的，而不是在用于启动平台的数据表中指定的。

数据格式

必须在两个数据表中指定“过程历史分析器”的数据。第一个表用于启动平台并包含过程的步骤和详细信息。第二个表包含 ID 列的每个水平的响应（或产量）。

样本数据表 Quality Control > Lot Wafer History 和 Quality Control > Lot Wafer Yield 提供关于如何在两个数据表中指定数据的示例。

“过程历史分析器”报表

初始 “过程历史分析器”报表提供数据表中信息的汇总。

- 单元数，由 ID 列确定。若有多个 ID 列，这就是唯一组合的数目。
- 步骤数，由**步骤**列确定。若有多个**步骤**列，这就是唯一组合的数目。
- 操作数，由**时间戳**列确定。与数据表中的行数相同。
- 水平数，由 X，**过程**列确定。若有多个 X，**过程**列，这就是每列中水平数的总和。
- 响应数，由**产量**列确定。

选择“过程历史分析器”红色小三角菜单中的选项以继续您的分析。请参见第 464 页的 “过程历史分析器”平台选项”。

“过程历史分析器”平台选项

“过程历史分析器”红色小三角菜单包含以下选项：

具有最低产量的水平 显示每个 X，**过程**变量的每个水平的产量汇总统计量表。为每个水平提供的汇总统计量包括操作数，以及产量的均值、标准误差和标准差。默认情况下，该表按产量均值降序排序。

带有时间过滤器的产量最低的水平 显示前 50 个具有最低产量的水平的时间区间的汇总统计量表。该例程会搜索 “具有最低产量的水平”表中前 50 个水平中每一个水平的时间区间，以便将该区间内的产量均值最小化。该区间必须包含该水平中至少 25% 的操作。为每个水平提供的汇总统计量都包含标识的区间中的操作数、标识的区间中的产量均值、该水平中的第一个和最后一个操作的时间戳（**第一个时间**和**最后一个时间**），以及标识的区间中第一个和最后一个操作的时间戳 （**运行开始时间**和**运行结束时间**）。

逐步回归 显示 “逐步回归”窗口，该窗口支持您指定用于逐步回归分析的选项。您可以使用逐步回归查找过程中的问题，问题就是指导致产量下降的因素。在回归中的每一步，哪一项具有最大负产量，哪一项就进入模型。这样可以帮助您标识对产量影响最大的过程步骤。“逐步回归”提供一个表，其中显示回归的每一步中进入模型的项的顺序。

注意：若在启动窗口中选定 “目标是最小化 Y”选项，则 “逐步回归”会进行调整，以检测可能导致产量增加的项。

目标 指定在逐步回归模型中使用的准则。“最大个体差值”选项标识个体导致产量变化最大的项。“最大总差值”选项标识导致最大总差值（即：乘以受影响单元数后的差值）的项。“最佳预测”选项可生成传统逐步回归模型。

时间过滤 支持您使用逐步回归模型中的组件的开始时间。若选择“无”选项，则取整个时段的平均时间。若选择“开始时间”选项，则允许组件在特定的开始时间失效。开始时间是将负产量影响最大化的时间。若选择“开始时间”选项，“逐步回归”报表表将包含“开始时间”列。

X 变换 支持您使用逐步回归中 X，过程 变量的常规计数数据或 Log(Count + 1) 数据。“Log(Count + 1)”选项可变换回归中的变量，以使高频计数在回归中具有较少的线性影响。若选择“无”选项，高频计数在回归中具有线性影响。

回归步数 “步进回归”分析中的步数。默认值为 10。

图 26.4 “逐步回归”窗口

等待时间分析 （仅当指定了两个 “时间戳”变量时才可用。）每个唯一 ID 一次只能经历步骤中的一个水平。此外，每一步一次只能处理一个 ID。“等待时间分析”为每个 ID 提供等待时间汇总统计量表。在该表中，等待时间是在一步中完成一个 ID 处理之后到开始另一步之前的时间量。同时提供了其他 “等待时间分析”表。这些表包含 “步骤”变量的各个水平内每个 ID 的等待时间汇总统计量。在这些表中，等待时间是针对 “步骤”变量的特定水平完成 ID 处理之后到再次经历同一水平之前的时间量。这些表提供以下统计量：

等待总和 ID 与“步骤”变量水平的给定组合的等待时间总和。

等待计数 ID 与“步骤”变量水平的给定组合的等待时间计数。

等待均值 ID 与“步骤”变量水平的给定组合的等待时间均值。

最长等待时间 ID 与“步骤”变量水平的给定组合的最大等待时间。

转换分析 显示一个报表，该报表有助于您确定使得产量结果出现问题的从一个组件到另一个组件的转换。该报表可以帮助您标识过程中的校准和注册问题。例如，过程列的某些水平相对于过程列中的其他水平可能未校准。若存在转换问题，您会发现重复特定水平的项与经历过程列中不同水平的项相比产量更高。

若选择该选项，您必须将一个或多个“X，过程”列以及每列中的一个或多个水平选入分析。对于您为分析指定的每一列，都会显示一个包含四个表的报表。第一个表为从该列的特定水平到该列的另一个特定水平的转换显示产量和计数。第二个表为离开该列特定水平的转换显示产量和计数。第三个表为进入该列特定水平的转换显示产量和计数。第四个表为单元重复该列特定水平的转换显示产量和计数。

顺序交互作用 （仅当在启动窗口指定了“顺序”变量时才可用。）显示针对“顺序”列的单因子产量分析报表。该报表对每个“X，过程”列都包含一个表。在每个表中，过程列的水平都以相应方差分析模型的 LogWorth 降序列出。该表还包含过程列每个水平的 *p* 值和热图。

保存计数表 创建包含经历“X，过程”变量每个水平的单元计数的新数据表。该计数表对每个唯一单元（由 ID 值确定）都包含一行。

保存计数对数表 创建与“计数”表格式相同（Log(Count + 1) 条目除外）的新数据表。

保存等待时间 （仅当指定了两个“时间戳”变量时才可用。）将“等待时间分析”报表中的每个表保存到单独的数据表。

附录 A

统计详细信息

预测和专业建模

本附录讨论了每个平台中可用的拟合测度和交叉验证的类型。

目录

拟合测度 469
熵 R 方 469
JMP 建模中的验证 469
“验证列”角色 469
K 折和保留验证 471

拟合测度

熵 R 方

为训练集计算“熵 R 方”，若使用了验证，则为验证集和测试集计算“熵 R 方”。对于训练集，按以下方式计算 “熵 R 方”：

- 使用训练集拟合指定模型。
- 基于该模型，得到了各响应水平的预测概率。
- 使用这些预测概率，为训练集中的观测计算似然值。称之为**似然_完全**$_{训练}$。
- 使用训练集拟合简化模型（无预测变量）。
- 使用简化模型中各响应水平的预测概率计算训练集中观测的似然值。将该量称为**似然 _ 简化**$_{训练}$。
- 训练集的“熵 R 方”为：

$$熵\ R\ 方_{训练} = 1 - \frac{\log(似然_完全_{训练})}{\log(似然_简化_{训练})}$$

验证集和测试集的“熵 R 方”计算方式与训练集的“熵 R 方”计算方式相似。

JMP 建模中的验证

为了避免过度拟合以及选择较好的预测模型，可以在建模前将数据划分为多个集。该过程使用部分原始数据来估计参数，并使用其余的数据来调优和 / 或评估参数。在 JMP Pro 中，您可以通过以下方式将数据划分为两个或三个集：

训练和评估 将数据划分为两个集，称为 “训练”和 “验证”。训练集用于估计模型参数。验证集用于独立评估拟合模型的性能。

训练和调优 将数据划分为两个集，称为 “训练”和 “验证”。训练集用于估计模型参数。在模型拟合算法中使用验证集对模型参数进行调优，最终选择具有良好预测能力的模型。在这种情况下不执行独立的模型评估。

训练、调优和评估 将数据划分为三个集，称为 “训练”、“验证”和 “测试”。训练集用于估计模型参数。在模型拟合算法中使用验证集对模型参数进行调优，最终选择具有良好预测能力的模型。测试集随后用于独立评估拟合模型的性能。

JMP PRO “验证列”角色

仅在 JMP Pro 中提供“验证列”角色。对于 JMP，请参见第 472 页的“排除行作为验证保留”。

创建数据分割的一种方式就是使用“验证列”角色。“验证列”角色使用该列的值将数据划分为多个部分。使用平台启动窗口中的“验证”角色分配该列。有关如何创建验证列的信息，请参见第 175 页的“生成验证列”一章。

警告：验证列的用法是特定于平台的。不同平台对验证列水平的使用方式有所不同。请参见表 A.1 中的注释。

表 A.1 按平台划分的验证列

平台	训练和评估	训练和调优	训练、调优和评估	注意
拟合模型				
拟合最小二乘法	是	否	否	若有三个以上水平，则忽略验证列。
逐步回归	否	是	是	若有三个以上水平，则使用“K 折交叉验证”。
Logistic 回归	是	否	否	若有三个以上水平，则忽略验证列。
广义回归	否	是	是	若有三个以上水平，则使用“K 折交叉验证”。
偏最小二乘	否	是	是	若有三个以上水平，则使用“K 折交叉验证”。
预测模型				
神经	否	是	是	若有三个以上水平，则使用“K 折交叉验证”。
分割	否	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。
Bootstrap 森林法	否	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。
提升树	否	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。

表 A.1 按平台划分的验证列 (续)

平台	训练和评估	训练和调优	训练、调优和评估	注意
K 最近邻	否	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。
朴素 Bayes	否	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。
支持向量机	是	是	是	若有三个以上水平，则使用“K 折交叉验证”。
专业模型				
函数数据分析器	是	否	否	必须创建为“分组随机”验证列。 若有两个以上水平，则最小的值定义训练集，所有其他值定义验证集。
多元模型				
判别	是	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。
偏最小二乘	否	是	是	若有三个以上水平，则使用“K 折交叉验证”。
提升	否	是	是	若有三个以上水平，则平台仅使用具有三个最小值的行。

K 折和保留验证

“验证列”角色提供了一个将数据划分为交叉验证集的框架。此外，某些 JMP 平台还支持 K 折验证和多种类型的“保留”验证。

K 折交叉验证 将原始数据划分为 *K* 个子集。这 *K* 个子集依次对基于剩余 k-1 个子集的数据构建的模型进行验证，总共拟合 *K* 个模型。选择生成最佳验证统计量的模型作为最终模型，构建该模型时未使用的那一折提供测试集性能统计量。

注意：对于某些平台，您必须在模型控制面板中指定 “K 折交叉验证”。对于其他平台，您必须在平台启动窗口中指定 “K 折交叉验证”。还有另外一些平台，您必须通过包含三个以上水平的验证列指定 “K 折交叉验证”。

随机验证保留 （可作为特定平台的启动选项。）将原始数据随机划分为训练集和验证集。也可包括测试集。您可以指定要在每个集中使用的原始数据的比例。

留一法验证保留 （可用作特定平台的选项。）重复拟合模型，一次留一个观测。留一法验证也称为 Jackknife 过程。

排除行作为验证保留 将数据表中的排除行用作验证保留集。对于 JMP Pro，可通过在平台首选项中选择使用该选项。

注意：对于支持将排除行用作验证保留集的平台，仅当启动窗口中没有指定验证列或验证比例时才使用排除行。

表 A.2 按平台划分的 K 折和保留验证

平台	排除行作为验证保留	随机验证保留	留一法保留	K 折交叉验证
拟合模型				
拟合最小二乘法	否	否	否	否
逐步回归	否	否	否	是（仅用于连续响应模型）
Logistic 回归	否	否	否	否
JMP PRO 广义回归	否	是	是	是（通过模型控制）
JMP PRO 偏最小二乘	否	是	是	是（通过模型控制）
预测模型				
神经	是	是	否	是（通过模型启动或验证列）
分割	是	是	否	是（在平台首选项中选择该选项）
JMP PRO Bootstrap 森林法	是	是	否	否
JMP PRO 提升树	是	是	否	否
JMP PRO K 最近邻	是	是	否	否

平台	排除行作为验证保留	随机验证保留	留一法保留	K 折交叉验证
JMP PRO 朴素 Bayes	是	是	否	否
JMP PRO 支持向量机	否	是	否	是（通过模型启动）
专业模型				
JMP PRO 函数数据分析器	否	否	否	否
多元模型				
判别	可选	否	否	否
偏最小二乘	否	是	是	是（通过模型启动或验证列）
JMP PRO 提升	否	是		否

附录 B

参考书目

Agrawal, R., and Srikant, R. (1994).“Fast Algorithms for Mining Association Rules.”In *Proceedings of the 20th VLDB Conference*.Santiago, Chile: IBM Almaden Research Center.Accessed July 5, 2016. https://web.stanford.edu/class/cs345d-01/rl/ar-mining.pdf.

Baglama, J., and Reichel, L. (2005).“Augmented implicitly restarted Lanczos bidiagonalization methods.”*SIAM Journal on Scientific Computing*, 27:19–42.

Bates, D. M., and Watts, D. G. (1988).*Nonlinear Regression Analysis and Its Applications*.New York: John Wiley & Sons.

Benford, F. (1938).“The law of anomalous numbers.”*Proceedings of the American philosophical society*, 551–572.

Benjamini, Y., and Hochberg, Y. (1995).“Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.”*Journal of the Royal Statistical Society, Series B* 57:289–300.

Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. (1994).*Time Series Analysis: Forecasting and Control*.3rd ed. Englewood Cliffs, NJ: Prentice-Hall.

Candes, E. J., Li, X., Ma, Y., and Wright, J. (2009).“Robust Principal Component Analysis?”*Journal of the ACM*, 58:1–37.

Cleveland, W. S. (1994).*Visualizing Data*, Summit, NJ: Hobart Press.

Conover, W. J. (1999).*Practical Nonparametric Statistics*.3rd ed. New York: John Wiley & Sons.

Cureton, E. E. (1967).“The Normal Approximation to the Signed-Rank Sampling Distribution when Zero Differences are Present.”*Journal of the American Statistical Association* 62: 319, 1068–1069.

Hahsler, M. (2015).“A Probabilistic Comparison of Commonly Used Interest Measures for Association Rules.”Accessed September 17, 2018. http://michael.hahsler.net/research/association_rules/measures.html.

Hand, D. J., Mannila, H., and Smyth, P. (2001).Principles of Data Mining.Cambridge, MA: MIT Press.

Hastie, T. J., Tibshirani, R. J., and Friedman, J. H. (2009).*The Elements of Statistical Learning: Data Mining, Inference, and Prediction*.2nd ed. New York: Springer-Verlag.

Hawkins D. M., and Kass G. V. (1982).“Automatic Interaction Detection.”In *Topics in Applied Multivariate Analysis*, edited by D. M. Hawkins, 267–300.Cambridge: Cambridge University Press.

Huber, P. J., and Ronchetti, E. M. (2009).*Robust Statistics*.2nd ed. New York: John Wiley & Sons.

Hyndman, R. J., Koehler, A. B., Ord, J. K., and Snyder, R. D. (2008).*Forecasting with Exponential Smoothing: The State Space Approach*.Berlin: Springer-Verlag.

Jolliffe, I. T. (2002).*Principal Component Analysis*.New York: Springer-Verlag.

Kass, G. V. (1980)."An Exploratory Technique for Investigating Large Quantities of Categorical Data."*Journal of the Royal Statistical Society, Series C* 29:119–127

Lehman, E. L. (2006).*Nonparametrics: Statistical Methods Based on Ranks*.2nd ed. New York: Springer.

Lin, Z., Chen, M., and Ma, Y. (2013).The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. arXiv preprint arXiv:1009.5055.

Mason, R. L., and Young, J. C. (2002).*Multivariate Statistical Process Control with Industrial Applications*.Philadelphia: SIAM.

McCullagh, P., and Nelder, J. A. (1989).*Generalized Linear Models*.2nd ed. London: Chapman & Hall.

Nagelkerke, N. J. D. (1991)."A Note on a General Definition of the Coefficient of Determination."*Biometrika* 78:691–692.

Nelder, J. A., and Wedderburn, R. W. M. (1972)."Generalized Linear Models."*Journal of the Royal Statistical Society, Series A* 135:370–384.

Parker, R. J. (2015).*Efficient Computational Methods for Large Spatial Data*. Ph.D. diss., Department of Statistics, North Carolina State University.Accessed June 30, 2016. https://repository.lib.ncsu.edu/ir/bitstream/1840.16/10572/1/etd.pdf.

Platt, J. (1998).*Sequential minimal optimization: A fast algorithm for training support vector machines*.Technical Report MST-TR-98-14, Microsoft Research. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-98-14.pdf.

Qian, P. Z., Huaiquing, W., and Wu, C. F. (2012)."Gaussian process models for computer experiments with qualitative and quantitative factors."*Technometrics* 50:383–396.

Ramsay, J. O., and Silverman, B. W. (2005).*Functional Data Analysis*.2nd ed. New York: Springer.

Ratkowsky, D. A. (1990).*Handbook of Nonlinear Regression Models*.New York: Marcel Dekker.

Sall, J. (2002)."Monte Carlo Calibration of Distributions of Partition Statistics."SAS Institute Inc.,Cary, NC.Accessed July 29, 2015. https://www.jmp.com/content/dam/jmp/documents/en/white-papers/montecarlocal.pdf.

Santer, T., Williams, B., and Notz, W. (2003).*The Design and Analysis of Computer Experiments*.New York: Springer-Verlag.

SAS Institute Inc.(2020).*SAS/ETS 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/etsug/15.2/content/etsug.pdf.

Schäfer, J., and Strimmer, K. (2005)."A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics."*Statistical Applications in Genetics and Molecular Biology* 4, Article 32.

Schuirmann, D. J. (1987).“A Comparison of the Two One-sided Tests Procedure and the Power Approach for Assessing the Equivalence of Average Bioavailability.”*Journal of Pharmacokinetics and Biopharmaceutics* 15:657–680.

Shiskin, J., Young, A. H., and Musgrave, J. C. (1967).*The X-11 Variant of the Census Method II Seasonal Adjustment Program.*Technical Report 15, US Department of Commerce, Bureau of the Census.

Shmueli, G., Patel, N. R., and Bruce, P. C. (2010).*Data Mining For Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel with XLMiner.*2nd ed. Hoboken, NJ: John Wiley & Sons.

Shmueli, G., Bruce, P. C., Stephens M. L., and Patel, N. R. (2017).*Data Mining For Business Intelligence: Concepts, Techniques, and Applications with JMP Pro.*Hoboken, NJ: John Wiley & Sons.

Westfall, P. H., Tobias, R. D., and Wolfinger, R. D. (2011).*Multiple Comparisons and Multiple Tests Using SAS.*2nd ed. Cary, NC: SAS Institute Inc.

White, K., Szarka, J., and Jensen, W. (2018).“A Recommended Set of Indices for Evaluating Process Health.”Presentation at the 2018 Fall Technical Conference, sponsored by the American Statistical Association and the American Society for Quality.

附录 C

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

  - You must give any other recipients of the Work or Derivative Works a copy of this License; and
  - You must cause any modified files to carry prominent notices stating that You changed the files; and
  - You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
  - If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
  - You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

多元方法

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020–2021。JMP® 16 多元方法.Cary, NC: SAS Institute Inc.

JMP® 16 多元方法

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录
多元方法

1 了解 JMP 15
文档和其他资源
JMP 文档中的格式规范 17
JMP 帮助 18
JMP 文档库 18
学习 JMP 的其他资源 24
JMP 教程 24
样本数据表 24
学习统计和 JSL 术语 25
学习 JMP 的提示与技巧 25
JMP 工具提示 25
JMP 用户社区 25
免费在线统计思维课程 25
JMP 新用户欢迎套件 26
统计学知识门户 26
JMP 培训 26
用户编写的 JMP 手册 26
“JMP 起始页”窗口 26
JMP 技术支持 26

2 多元分析介绍 27
多元方法概述

3 相关性和多元方法 29
探索变量的多维行为
“多元”平台的示例 31
启动“多元”平台 33
“多元”报表 34

“多元”平台选项 34
非参数相关性 37
散点图矩阵 38
离群值分析 39
项目信度 41
补缺缺失数据 42
“项目信度”示例 42
计算和统计详细信息 43
估计方法 43
Pearson 乘积矩相关系数 44
关联的非参数测度 44
逆相关性矩阵 46
距离测度 46
Cronbach α 值 48

4 主成分 49
对数据降维
“主成分分析”平台概述 51
主成分分析的示例 51
启动“主成分”平台 52
缺失数据 54
“主成分”报表 54
“主成分”报表选项 55
离群值分析 63
“主成分分析”平台的统计详细信息 64
估计方法 64
DModX 计算 66
离群值分析中的计算 66

5 判别分析 69
基于连续变量预测分类
“判别分析”平台概述 71
判别分析的示例 71
启动“判别分析”平台 72
逐步选择变量 74

判别方法 77
收缩协方差 80
“判别分析”报表 80
主成分 81
典型图和典型结构 81
判别得分 84
得分汇总 85
“判别分析”选项 87
得分选项 89
典型选项 90
“典型三维图”的示例 93
指定先验值 94
考虑新水平 94
保存判别矩阵 95
散点图矩阵 95
JMP 和 JMP Pro 中的验证 96
“判别分析”平台的统计详细信息 97
宽线性算法的说明 97
保存的公式 97
多元检验 103
近似 F 检验 104
组间协方差矩阵 105

6 偏最小二乘模型 107
使用 Y 和 X 之间的相关性构建模型

“偏最小二乘”平台概述 109
“偏最小二乘”示例 109
启动“偏最小二乘”平台 113
中心化和统一尺度 116
标准化 X 116
“模型启动”控制面板 117
“偏最小二乘”选项 118
“偏最小二乘”报表 118
模型比较汇总 118

“交叉验证”报表 119
“模型拟合”报表 123
模型拟合选项 124
变量重要性图 126
系数 -VIP 图 126
保存列 127
“偏最小二乘”的其他示例 128
“偏最小二乘”平台的统计详细信息 130
偏最小二乘 130
van der Voet T^2 131
T^2 图 132
X 得分散点图矩阵的置信椭圆 132
预测的标准误差和置信限 132
标准化得分和载荷 133
PLS 判别分析 (PLS-DA) 134

7 多重对应分析 135
识别分类变量各水平之间的关联
多重对应分析的示例 137
启动“多重对应分析”平台 140
“多重对应分析”报表 141
“多重对应分析”平台选项 142
对应分析选项 143
显示图 144
显示详细信息 144
显示调整惯量 145
显示坐标 145
显示汇总统计量 145
显示对惯量的部分贡献 146
显示平方余弦 146
Cochran Q 检验 147
交叉表 147
“多重对应分析”平台的更多示例 147
使用补充变量的示例 148

使用补充 ID 的示例 149
Cochran Q 检验示例 149
“多重对应分析”平台的统计详细信息 151
“详细信息”报表 151
调整惯量 152
汇总统计量 152
对惯量的部分贡献 152
Cochran Q 统计量 153

8 结构化方程模型 155
拟合结构化方程模型
结构化方程模型概述 157
结构化方程模型示例 159
启动“结构化方程模型”平台 163
“结构化方程模型”报表 164
“模型规格”报表 164
“模型比较”报表 169
“卡方差异检验”报表 170
“结构化方程模型拟合”报表 170
“结构化方程模型”平台选项 171
模型选项 172
定制 SEM 路径图 175
路径图弹出式菜单选项 175
“定制关系图外观”选项 177
结构化方程模型的更多示例 177
“结构化方程模型”平台的统计详细信息 184
拟合汇总测度的统计详细信息 184

9 因子分析 189
标识数据中的潜在变量
“因子分析”平台概述 191
“因子分析”平台的示例 191
启动“因子分析”平台 194
“因子分析”报表 195
模型启动 196

旋转方法 197
“因子分析”平台选项 198
因子分析模型拟合选项 199

10 多维尺度化 203
直观表示一组对象间的邻近性
“多维尺度化”平台概述 205
“多维尺度化”示例 205
启动“多维尺度化”平台 208
“多维尺度化”报表 209
多维尺度化图 209
Shepard 图 209
拟合详细信息 209
“多维尺度化”平台选项 210
Waern 链接 210
“多维尺度化”平台的更多示例 211
“多维尺度化”平台的统计详细信息 213
Stress 213
变换 213
特性列表格式 214

11 项目分析 215
按项目和对象分析测验结果
项目分析的示例 217
启动“项目分析”平台 219
Logistic 3PL 模型详细信息 219
数据格式 220
“项目分析”报表 220
特征曲线 220
信息图 221
对偶图 221
参数估计值 222
“项目分析”平台选项 223
“项目分析”平台的统计详细信息 223
项目响应曲线 223

项目响应曲线模型 224
IRT 模型假设 226
拟合 IRT 模型 227
能力公式 228

12 层次聚类 229
使用聚类树将观测分组

“层次聚类”平台概述 231
对观测聚类的平台概述 231
聚类示例 232
启动“层次聚类”平台 235
聚类方法 236
计算距离的方法 236
数据结构 237
“Y, 列”变量的变换 238
“层次聚类”报表 239
“系统树图”报表 239
系统树图和距离图的演示 240
“聚类历史”报表 241
“层次聚类”选项 241
“层次聚类”平台的更多示例 244
距离矩阵的示例 244
使用“空间测度”进行晶片次品分类的示例 246
“层次聚类”平台的统计详细信息 248
空间测度 248
距离方法公式 250

13 K 均值聚类 253
使用距离对观测分组

“K 均值聚类”平台概述 255
对观测聚类的平台概述 255
“K 均值聚类”的示例 256
启动“K 均值聚类”平台 259
“迭代聚类”报表 260
“迭代聚类”选项 260

“迭代聚类”控制面板 261
K 均值报表 262
“聚类比较”报表 262
K 均值报表 262
“K 均值”报表选项 262
自组织图 264
“自组织图”控制面板 264
“自组织图”报表 265
SOM 算法的说明 265
“K 均值聚类”平台的更多示例 266
自组织图示例 266

14 正态混合 269
使用概率对观测分组

“正态混合聚类”平台概述 271
对观测聚类的平台概述 271
“正态混合聚类”的示例 272
启动“正态混合聚类”平台 274
“基于聚类的模型”报表 275
“基于聚类的模型”选项 275
“基于聚类的模型”控制面板 276
“正态混合”报表 277
“聚类比较”报表 277
“正态混合”报表 277
“正态混合”报表选项 277
“正态混合聚类”平台的统计详细信息 279

15 潜在类分析 281
将分类变量的观测进行分组

“潜在类分析”平台概述 283
“潜在类分析”的示例 283
启动“潜在类分析”平台 286
“潜在类分析”报表 287
“聚类比较”报表 287
潜在类模型报表 287

“潜在类分析”平台选项 289
“潜在类分析”选项 289
“潜在类模型”选项 289
“潜在类分析”平台的更多示例 290
绘制聚类成员关系的概率图 290
“潜在类分析”平台的统计详细信息 291
潜在类模型拟合 291
最大聚类数 292

16 聚类变量 295
将类似变量分组到代表组中
“聚类变量”平台概述 297
“聚类变量”平台的示例 297
启动“聚类变量”平台 299
“聚类变量”报表 299
相关性色图 300
聚类汇总 300
聚类成员 300
标准化成分 301
“聚类变量”平台选项 301
“聚类变量”平台的更多示例 301
相关性色图的示例 302
“聚类变量”平台有关降维的示例 303
“聚类变量”平台的统计详细信息 306
变量聚类算法 306

A 统计详细信息 309
多元方法
“宽线性”方法和奇异值分解 311
奇异值分解 311
SVD 和协方差矩阵 311
SVD 和逆协方差矩阵 312
计算 SVD 312

B 参考书目 313
C 技术许可声明 319

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 17
JMP 帮助 18
JMP 文档库 18
学习 JMP 的其他资源 24
JMP 教程 24
样本数据表 24
学习统计和 JSL 术语 25
学习 JMP 的提示与技巧 25
JMP 工具提示 25
JMP 用户社区 25
免费在线统计思维课程 25
JMP 新用户欢迎套件 26
统计学知识门户 26
JMP 培训 26
用户编写的 JMP 手册 26
“JMP 起始页”窗口 26
JMP 技术支持 26

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助 > JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助 > JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 18 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助 > JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

多元分析介绍

多元方法概述

《多元方法》说明以下用于同时分析若干变量的方法：

- “多元”平台检查多个变量以查看它们之间如何彼此相关。请参见第 3 章，“相关性和多元方法”。
- “主成分”平台从一组测量变量中得到少数几个相互独立的线性组合（主成分），使用它们来捕获原始变量中尽可能多的变异性。这是一种有用的探索性方法，可帮助您创建预测模型。请参见第 4 章，“主成分”。
- “判别”平台尝试找到一种根据已知的连续响应 (Y) 来预测分类 (X) 变量（名义型或有序型）的方法。它可以视为多元方差分析 (MANOVA) 的逆预测。请参见第 5 章，“判别分析”。
- “偏最小二乘”平台根据因子即解释变量 (X) 的线性组合来拟合线性模型。“偏最小二乘”利用 X 和 Y 之间的相关性来揭示底层的潜在结构。请参见第 6 章，“偏最小二乘模型”。
- “多重对应分析”(MCA) 平台适用于多个分类变量，并力求确定这些变量各水平之间的关联。MCA 常用在社会科学中，在法国和日本尤为普遍。可将其用在调查分析中，找出测试对象对不同问题的态度一致性。请参见第 7 章，“多重对应分析”。
- JMP PRO “结构化方程模型”支持您拟合各种模型，包括确认性因子分析、具有或不具有潜在变量的路径模型、测量值误差模型以及潜在变量增长曲线模型。请参见第 8 章，“结构化方程模型”。
- “因子分析”平台支持您从更大的一组观测变量中构造因子。这些因子可以表示为观测变量子集的线性组合。通过因子分析，您可以探索由一组测量的观测变量解释的因子数量，以及因子与变量之间关系的强度。请参见第 9 章，“因子分析”。
- “多维尺度化 (MDS)”平台支持您分析一组对象间的邻近性（相似性、相异性或距离），将它们之间的模式通过图形直观呈现出来。请参见第 10 章，“多维尺度化”。
- “项目分析”平台支持您拟合项目反应理论模型。项目反应理论 (IRT) 方法用于对测量手段（如：测试和问卷）进行分析和评分。IRT 使用模型体系将个体的特征与该个体正面或正确响应某个项目的概率相关联。IRT 可用于研究标准化测验、认知发展和消费者偏好。请参见第 11 章，“项目分析”。
- “层次聚类”平台将在几个变量上享有相似值的行分组在一起。这是一种有用的探索性方法，可帮助您理解数据的聚簇结构。请参见第 12 章，“层次聚类”。
- “K 均值聚类”平台将在几个变量上享有相似值的观测分组在一起。请参见第 13 章，“K 均值聚类”。
- “正态混合”平台可使您在数据来自重叠的正态分布时对观测聚类。请参见第 14 章，“正态混合”。

- “潜在类分析”平台可找到分类响应变量的观测聚类。该模型采取多项式混合模型的形式。请参见第 15 章，“潜在类分析”。
- “聚类变量”平台将相似变量分到典型组中。您可以将“聚类变量”用作降维方法。您不用在建模中使用大量变量，聚类中最典型的变量的聚类成分可用于解释数据中的大部分变异。请参见第 16 章，“聚类变量”。

第 3 章

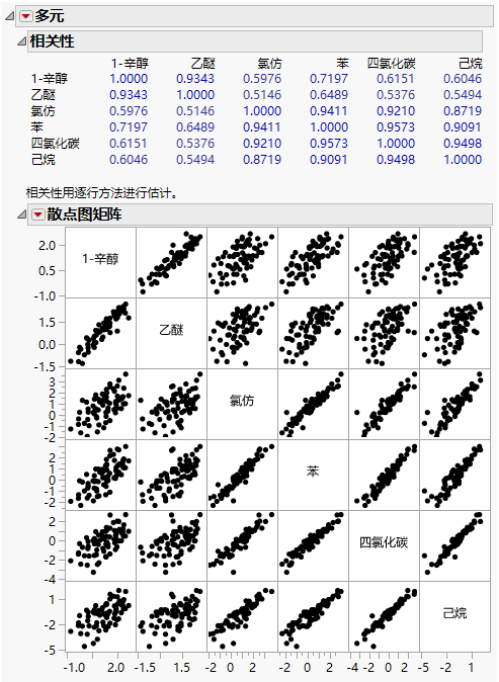
相关性和多元方法

探索变量的多维行为

多元数据涉及多个变量，而不是一个变量（一元）或两个变量（二元）。使用“多元”平台可探索多个变量如何彼此相关。“多元”平台提供许多方法来汇总和检验每对响应变量之间的线性关系的强度。该平台提供了参数和非参数相关性检验。还可以使用图形功能（例如散点图矩阵和色图）来标识变量之间的相依性、离群值和聚类。

还有其他的多元分析方法来进一步检验变量之间的关系，包括主成分分析、离群值分析和项目信度。这些方法可以通过“多元”报表获得。您还可以使用 JMP 中的“主成分分析”和“离群值分析”平台更深入地实施这些方法。

图 3.1 “多元”报表的示例



目录

“多元”平台的示例 31
启动“多元”平台 33
“多元”报表 34
“多元”平台选项 34
非参数相关性 37
散点图矩阵 38
离群值分析 39
项目信度 41
补缺缺失数据 42
“项目信度”示例 42
计算和统计详细信息 43
估计方法 43
Pearson 乘积矩相关系数 44
关联的非参数测度 44
逆相关性矩阵 46
距离测度 46
Cronbach α 值 48

“多元”平台的示例

本例使用 Body Measurements.jmp 样本数据表检查针对人体的不同测量值之间的关系。

1. 选择**帮助** > **样本数据库**，然后打开 Body Measurements.jmp。
2. 选择**分析** > **多元方法** > **多元**。
3. 选择除**体重**之外的所有列并点击 **Y，列**。
4. 点击**确定**。

   初始多元报表包含相关性矩阵和散点图矩阵。还有一条注释告知您使用的是逐行估计方法。所有变量都是正相关，但强度不同。
5. 点击 “多元”红色小三角并取消选择**散点图矩阵**。
6. 点击 “多元”红色小三角并选择**相关性概率**。
7. 点击 “多元”红色小三角并选择**色图** > **相关性色图**。

图 3.2 针对人体测量数据的多元报表

多元

相关性

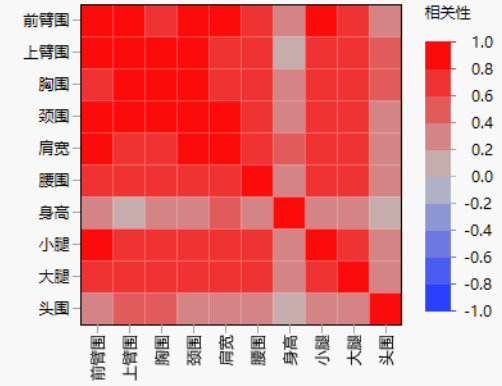
	前臂围	上臂围	胸围	颈围	肩宽	腰围	身高	小腿	大腿	头围
前臂围	1.0000	0.8134	0.7671	0.8134	0.8670	0.7577	0.3537	0.8130	0.7877	0.3029
上臂围	0.8134	1.0000	0.8552	0.8240	0.7202	0.6652	0.0657	0.6421	0.6974	0.4199
胸围	0.7671	0.8552	1.0000	0.8279	0.7999	0.7154	0.2017	0.6436	0.7802	0.4379
颈围	0.8134	0.8240	0.8279	1.0000	0.8312	0.7530	0.3912	0.6095	0.7055	0.3223
肩宽	0.8670	0.7202	0.7999	0.8312	1.0000	0.7462	0.4839	0.6847	0.6825	0.2258
腰围	0.7577	0.6652	0.7154	0.7530	0.7462	1.0000	0.3513	0.6893	0.7403	0.2997
身高	0.3537	0.0657	0.2017	0.3912	0.4839	0.3513	1.0000	0.3977	0.2860	0.1576
小腿	0.8130	0.6421	0.6436	0.6095	0.6847	0.6893	0.3977	1.0000	0.7497	0.2742
大腿	0.7877	0.6974	0.7802	0.7055	0.6825	0.7403	0.2860	0.7497	1.0000	0.2363
头围	0.3029	0.4199	0.4379	0.3223	0.2258	0.2997	0.1576	0.2742	0.2363	1.0000

相关性用逐行方法进行估计。

相关性概率

	前臂围	上臂围	胸围	颈围	肩宽	腰围	身高	小腿	大腿	头围
前臂围	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.1064	<.0001	<.0001	0.1706
上臂围	<.0001	<.0001	<.0001	<.0001	0.0002	0.0007	0.7715	0.0013	0.0003	0.0517
胸围	<.0001	<.0001	<.0001	<.0001	<.0001	0.0002	0.3681	0.0012	<.0001	0.0415
颈围	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0718	0.0026	0.0002	0.1436
肩宽	<.0001	0.0002	<.0001	<.0001	<.0001	<.0001	0.0225	0.0004	0.0005	0.3124
腰围	<.0001	0.0007	0.0002	<.0001	<.0001	<.0001	0.1089	0.0004	<.0001	0.1753
身高	0.1064	0.7715	0.3681	0.0718	0.0225	0.1089	<.0001	0.0668	0.1969	0.4836
小腿	<.0001	0.0013	0.0012	0.0026	0.0004	0.0004	0.0668	<.0001	<.0001	0.2168
大腿	<.0001	0.0003	<.0001	0.0002	0.0005	<.0001	0.1969	<.0001	<.0001	0.2898
头围	0.1706	0.0517	0.0415	0.1436	0.3124	0.1753	0.4836	0.2168	0.2898	<.0001

相关性色图

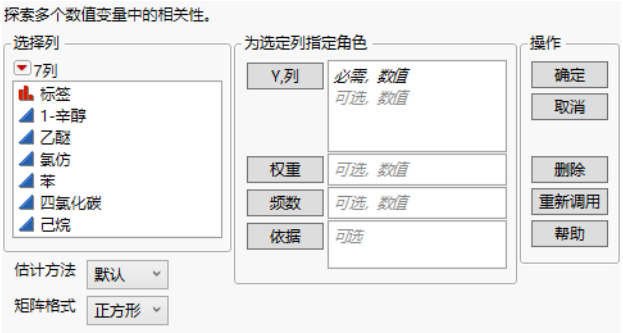


“相关性色图”报表提供“相关性矩阵”中信息的更简明版本。图中大部分是深红色，表明大多数变量高度相关。两个浅色的行和列表示身高和头围测量值与其他变量的相关性不高。“相关性概率”表中的“身高”和“头围”的大部分非显著 p 值都进一步支持了这一点。

启动“多元”平台

通过选择**分析** > **多元方法** > **多元**来启动“多元”平台。

图 3.3 “多元”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 标识一个或多个响应列。响应列必须具有数值数据类型，但建模类型可以是连续型或有序型。

注意：若您指定有序型响应变量，则在“启动”窗口中点击“确定”后会出现 JMP 警报。该警报指示哪些变量是有序型变量，并确认您打算在分析中包含有序型变量。

权重 标识一列，该列的数值为分析中的每一行都分配一个权重。

频数 标识一列，该列的数值为分析中的每一行都分配一个频数。

依据 为“依据”变量的每个水平生成单独的报表。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独报表。

估计方法 指定计算相关性的方法。“REML”和“配对”是最常用的方法。其中有些方法解决了缺失数据的处理。您也可以使用“逐行”以外的方法估计缺失值，然后选择“补缺缺失数据”命令。请参见第 42 页的“补缺缺失数据”。

默认 默认选项使用“逐行”、“配对”或“REML”方法。

- 逐行估计用于不含缺失值的数据表。
- 配对估计用于包含缺失值并且多于 10 列、多于 5,000 行或列数多于行数的数据表。
- 在其他情况下使用 REML 估计。

注意：若“默认”选项在其他情况下导致 REML 估计，但拟合不能正确收敛，则平台转换为“配对”方法。当数据表中有缺失值且以下情形之一成立时就会出现这种情况：您的数据表少于 10 列、少于 5,000 行，或列数少于行数。若显示的估计方法为“配对”，这意味着 REML 拟合未收敛。

REML 限制最大似然 (REML) 估计使用所有数据，即使存在缺失值。由于存在偏倚修正因子，若您的数据集很大且包含许多缺失值，该方法会很慢。因此，REML 最适用于较小的数据集。若数据中不含缺失单元格，则 REML 和 ML 估计值等价于样本协方差矩阵。若存在缺失单元格，与 ML 估计相比，REML 的方差和协方差估计值的偏倚更小。有关统计详细信息，请参见第 43 页的“REML”。

ML 最大似然 (ML) 估计使用所有数据，即使存在缺失值。由于 ML 的估计值生成速度更快，该方法最适用于包含缺失数据的大数据表。

稳健 稳健估计使用所有数据，即使存在缺失值。该方法降低了极值的权重，因此最适用于可能具有离群值的数据表。有关统计详细信息，请参见第 44 页的“稳健”。

逐行 逐行估计为每对列计算 Pearson 相关性系数。有关统计详细信息，请参见第 44 页的“Pearson 乘积矩相关系数”。逐行估计不使用包含缺失值的行。该方法适用于排除具有缺失数据的观测。

配对 配对估计使用所有数据，即使存在缺失值。该估计方法使用这两列中不含缺失值的所有观测为每对列计算 Pearson 相关性系数。有关统计详细信息，请参见第 44 页的“Pearson 乘积矩相关系数”。配对估计最适用于包含缺失值，或者列数多于行数、多于 10 列，或者多于 5,000 行的数据表。

注意：若您选择“REML”、“ML”或“稳健”，而您的数据表中的列数多于行数并且具有缺失值，JMP 会将“估计方法”切换为“配对”。

矩阵格式 选择用于“散点图矩阵”的格式选项。“正方形”选项为所有排序列组合显示图。“下三角”在对角线下方显示图，前 n - 1 个列显示在水平轴上。“上三角”在对角线上方显示图，前 n - 1 个列显示在垂直轴上。

“多元”报表

默认的多元报表显示标准相关性矩阵、散点图矩阵，以及一条指示估计相关性所用的方法的注释。在某些情况下，还会显示解释为何使用给定方法的信息。平台菜单还列有其他的相关性选项以及分析方法用于研究多个变量之间的关系。请参见第 34 页的““多元”平台选项”。

“多元”平台选项

注意：“多元”报表中的相关性值和 p 值使用“首选项”中的条件格式设置来设置格式。请参见《使用 JMP》。

“多元”红色小三角菜单包含以下选项：

多元相关性 显示或隐藏“相关性”表。该表是一个相关系数矩阵，该矩阵汇总了每对响应 (Y) 变量之间的线性关系的强度。默认启用该选项。请参见第 44 页的“Pearson 乘积矩相关系数”。

注意：该相关性矩阵使用您在启动窗口中选择的方法来计算。

相关性概率 显示或隐藏“相关性概率”表。该表是一个 *p* 值矩阵。每个 *p* 值都对应变量之间的真实相关性为 0 这一原假设下的检验。这是针对两个响应变量之间不存在线性关系的检验。

相关性置信区间 显示或隐藏相关性的双侧置信区间。

提示：默认置信系数为 95%。使用设置 α 水平选项可更改置信系数。

逆相关性 显示或隐藏“逆相关性”报表，该报表是逆相关性矩阵。矩阵的对角线元素是一个函数，它表示为一个变量在多大程度上接近由其他变量构成的线性函数。在逆相关性表中，对角线为 $1/(1 - R^2)$ 的值。R^2 从对其他所有变量回归该变量的简单线性模型计算得出。若多重相关性为 0，则对角线逆元素为 1。若多重相关性为 1，则逆元素变为无穷大，所以报告为缺失。有关逆相关性的统计详细信息，请参见第 46 页的 “逆相关性矩阵”。

偏相关性 显示或隐藏“偏相关性”报表，该报表是偏相关性矩阵。偏相关性矩阵显示了考虑其他所有变量的效应进行调整之后，两个变量之间关系的测度。该表是按照单位对角线统一尺度的负的逆相关性矩阵。这意味着矩阵经过统一尺度，以使对角线元素等于 1。

偏相关性概率 显示或隐藏“偏相关性概率”报表，该报表是 *p* 值的矩阵。每个 *p* 值都对应变量之间的真正偏相关性为 0 这一原假设下的检验。这是在对其他变量的效应进行调整之后，针对两个响应变量之间不存在线性关系的检验。

注意：在没有足够的自由度时，“偏相关性概率”选项不可用。当变量数多于观测数时就可能出现这种情况。

协方差矩阵 显示或隐藏协方差矩阵，该矩阵测量的是一对变量同时改变的程度。

配对相关性 显示或隐藏“配对相关性”表，该表列出每对 Y 变量的 Pearson 乘积矩相关性。相关性通过配对删除方法计算。若任何变量对中的任一个变量含有缺失值，计数值将有所不同。“配对相关性”报表还显示显著性概率并在条形图中比较相关性。所有结果都基于配对方法。

注意：一旦所考虑的两个变量中任意一个变量存在缺失值，该选项会排除相应的行。

Hotelling T^2 检验 支持您针对作为 Y 输入的变量执行多元分布的单样本均值检验。在选择“Hotelling T^2 检验”选项后，会出现一个窗口，该窗口支持您指定原假设下的均值向量。输入每个变量的假设均值。该检验假定 Y 变量服从多元正态分布。“Hotelling T^2 检验”报表提供以下信息：

变量 作为 Y 输入的变量。

均值 每个变量的样本均值。

假设均值 您指定的原假设均值。

检验统计量 Hotelling T^2 统计量的值。

F 比 检验统计量的值。若您有 *n* 行和 *k* 个变量，则 *F* 比的计算公式定义如下：

$$\frac{n-k}{k(n-1)}T^2$$

概率 > F 检验的 p 值。在原假设下，F 比服从自由度为 k 和 n - k 的 F 分布。

注意：要删除报表，请点击 T^2 检验旁边的红色小三角并选择**删除检验**。

简单统计量 该菜单包含两个选项，每个选项都可以显示或隐藏每列的简单统计量（均值、标准差、总和、最小值和最大值）。若含有缺失值或使用了“稳健”方法，一元和多元简单统计量会有所不同。

一元简单统计量 显示对每列计算的统计量，而不考虑其他列中的值或缺失值。这些值与“分布”平台生成的那些值相同。

多元简单统计量 显示与在启动窗口中选定的估计方法对应的统计量，以及是否存在缺失数据。若没有缺失观测，该选项仅可用于稳健方法。若有缺失观测，该选项可用于配对之外的所有估计方法。“REML”、“ML”或“稳健”方法，选定的方法估计均值向量和协方差矩阵。对于“逐行”方法，则从均值和方差的计算中排除至少包含一个缺失值的所有行。

非参数相关性 该菜单包含三个非参数测度：Spearman Rho、Kendall Tau 和 Hoeffding D。每个选项显示或隐藏相应测度的非参数报表。请参见第 37 页的“非参数相关性”。

设置 α 水平 您可以为相关性置信区间指定任意 alpha 值。共列出四个 alpha 值：0.01、0.05、0.10 和 0.50。选择**其他**可输入任何其他值。

散点图矩阵 显示或隐藏每对响应变量的散点图矩阵。默认启用该选项。请参见第 38 页的“散点图矩阵”。

色图 “色图”菜单包含三种色图。该菜单中的每个选项显示或隐藏相应类型的色图。提供以下类型的色图：

相关性色图 生成一个方格图，该图采用由红 (+1) 到蓝 (-1) 的尺度显示变量之间的相关性。

p 值色图 生成一个方格图，该图采用由 $p = 0$（红）到 $p = 1$（蓝）的尺度显示相关性的显著性。

对相关性进行聚类 生成一个方格图，该图将相似变量聚类在一起。该相关性与“相关性色图”的相关性相同，但变量的位置可能有所不同。

配对相关性色图 生成一个方格图，该图采用由红 (+1) 到蓝 (-1) 的尺度显示变量之间的配对相关性。

Spearman ρ 色图 生成一个方格图，该图采用由红 (+1) 到蓝 (-1) 的尺度显示变量之间的 Spearman ρ 非参数相关性。

Kendall τ 色图 生成一个方格图，该图采用由红 (+1) 到蓝 (-1) 的尺度显示变量之间的 Kendall τ 非参数相关性。

Hoeffding D 色图 生成一个方格图，该图采用由红 (+1) 到蓝 (-1) 的尺度显示变量之间的 Hoeffding D 非参数相关性。

平行坐标图 显示或隐藏变量的平行坐标图。

三维椭圆图 显示或隐藏带 95% 置信椭圆的三维散点图。选择该选项时，系统会提示您指定三个变量及其相应轴。

主成分 该菜单包含用于显示或隐藏主成分报表的选项。可以基于相关性、协方差或未统一尺度的值选择主成分报表。在已经显示某个主成分报表时选择其中一个选项可将该报表改为新选项。选择**无**可删除该报表。

主成分是对原始变量计算线性组合的一种方法。第一主成分具有最大变异，第二主成分具有第二大的变异 （第二主成分必须与第一主成分正交），依此类推。请参见第 49 页的 “主成分”一章。

离群值分析 该菜单包含的每个选项均用于显示或隐藏特定图，这些图使用以下方法之一测量多元意义上的距离：Mahalanobis 距离、Jackknife 距离和 T^2 统计量。请参见第 39 页的“离群值分析”。

项目信度 该菜单包含的每个选项都显示或隐藏一个项目信度报表。这些报表使用 Cronbach α 值或标准化 α 指示一组工具在测量总响应上的一致性程度。请参见第 41 页的“项目信度”。

补缺缺失数据 （仅当数据表包含缺失值时才可用。）生成一个新数据表，它复制您的数据表并用估计值替换所有缺失值。请参见第 42 页的 “补缺缺失数据”。

保存插补公式 （仅当数据表包含缺失值时才可用。）对于包含缺失值的列，将包含用于估计缺失值的公式的新列保存到数据表中。新列称为**补缺 _< 列名 >**。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

非参数相关性

“非参数相关性”菜单提供三个用于配对相关性的非参数测度。每个 “非参数相关性”报表提供所选关联测度的显著性概率并且在条形图上显示关联值。共有三个非参数相关性测度。

Spearman Rho 一个基于数据值的秩而不是数据值本身计算的相关系数。

Kendall Tau 基于观测的一致对和不一致对的数目。若具有较大 X 值的观测也具有较大的 Y 值，则为**一致**对。若具有较大 X 值的观测具有较小的 Y 值，则为**不一致**对。同分对 （即具有相等的 X 值或相等的 Y 值的观测对）会进行校正。

Hoeffding D 范围从 –0.5 到 1 的统计尺度。较大的正值说明较强的相关性。该统计量近似于 2 x 2 分类表的卡方统计量的观测加权和。通过将每个数据值设置为阈值生成 2 x 2 表。该统计量可以检测到更一般的独立性偏离。

注意：使用“配对”方法计算非参数相关性，即便在启动窗口中选定了其他估计方法。

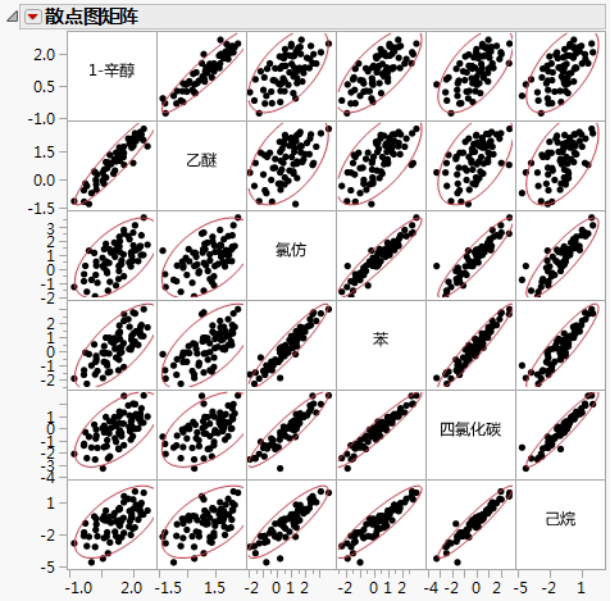
注意：当指定了“权重”变量时，缺失值和零值权重会从非参数相关性计算中排除。所有其他权重值均视为 1。

有关这三种方法的统计详细信息，请参见第 44 页的“关联的非参数测度”。

散点图矩阵

散点图矩阵帮助您直观演示每对响应变量之间的相关性。

图 3.4 散点图矩阵



启用“密度椭圆”选项可在每个散点图中都显示一个 95% 二元正态密度椭圆。假定每对变量都服从二元正态分布，该椭圆将覆盖大约 95% 的点。椭圆的窄度反映变量的相关度。若椭圆较圆，不沿着对角方向延伸，则变量之间不相关。若该椭圆较窄且向对角方向延伸，则变量之间相关。

提示：

- 调整任意单元的大小将调整所有单元的大小。
- 将标签单元拖至另一个标签单元可对矩阵重新排序。
- 查找散点图矩阵中的模式时，您可以看到变量根据其相关性聚类到组中。图 3.4 显示了两个相关性聚类：前两个变量（顶部，左侧）和后四个变量（底部，右侧）。

散点图矩阵选项

“散点图矩阵”的红色小三角菜单支持您使用颜色和密度椭圆以及设置 α 水平来定制矩阵。

显示点 在散点图中显示或隐藏点。

拟合线 显示或隐藏回归线和拟合回归线的 95% 水平置信曲线。

密度椭圆 在散点图中显示或隐藏 95% 密度椭圆。使用椭圆 α 菜单可更改 α 水平。

着色椭圆 为每个椭圆着色。使用椭圆透明度和椭圆颜色菜单可更改透明度和颜色。

显示相关性 在每个散点图的左上角显示或隐藏每对变量的相关性。

矩阵选项 （仅当在启动窗口中选定“正方形”矩阵格式时才可用。）显示用于更改散点图矩阵右上小三角形外观的选项子菜单。一次只能选择以下选项之一。

显著性圆圈 在散点图矩阵的右上三角中显示或隐藏相关性圆圈。每个圆圈的颜色采用由红 (+1) 到蓝 (-1) 的尺度来表示每对变量之间的相关性。每个圆圈的大小表示变量之间的显著性检验。较大的圆圈指示关系更为显著。

热图 在散点图矩阵的右上三角中显示或隐藏相关性热图。热图中每个方格的颜色采用由红 (+1) 到蓝 (-1) 的尺度来表示每对变量之间的相关性。

显示直方图 在标签单元中显示或隐藏水平或垂直直方图。一旦添加直方图，选择**显示计数**可在直方图的每个直条上标记其计数。选择**水平**或**垂直**可更改直方图的方向或删除直方图。

椭圆 α 设置用于椭圆的 α 水平。在菜单中选择一个标准 α 水平，或选择**其他**输入不同值。

椭圆透明度 若椭圆已着色，则设置其透明度。选择一个默认水平，或选择**其他**输入不同值。默认值为 0.2。

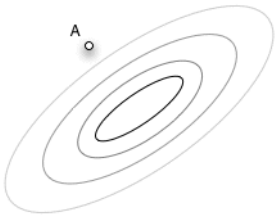
椭圆颜色 若椭圆已着色，则设置其颜色。在调色板中选择一种颜色，或选择**其他**使用另一种颜色。默认值为红色。

非参数密度 基于描述数据点密度的平滑非参数二元曲面显示或隐藏着色密度等高线。显示非参数曲面的 10% 和 50% 分位数的等高线。

离群值分析

“离群值分析”菜单包含的每个选项都显示或隐藏各图，这些图用于测量相对于相关性结构的多元意义上的距离。例如，在图 3.5 中，“点 A”是一个离群值，这是因为它位于相关性结构之外，即便它在任意坐标方向上都不是一个离群值。

图 3.5 离群值示例



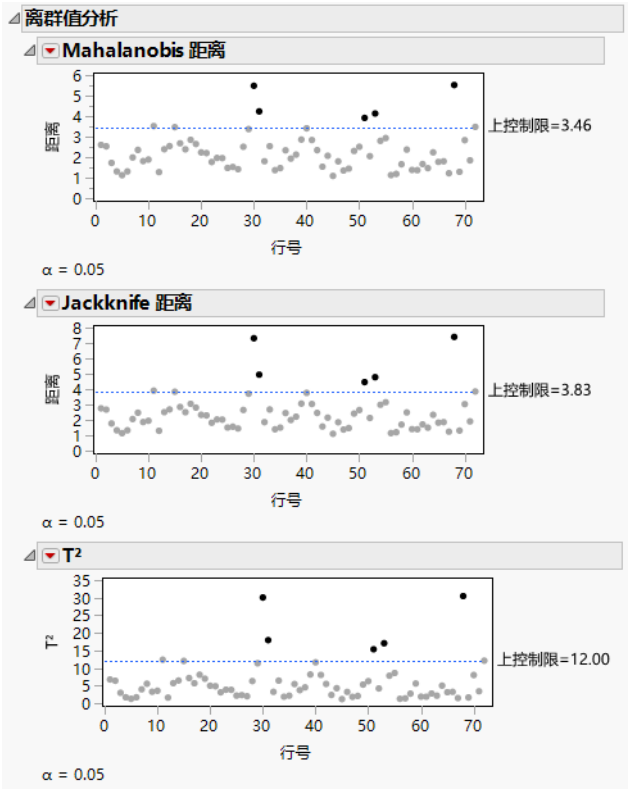
“离群值分析”菜单中的每个选项都显示或隐藏相应距离的图。检验是在图底部显示的 alpha 水平下执行的。以下选项可用：

Mahalanobis 距离 显示或隐藏每个点与多元均值（重心）的 Mahalanobis 距离。标准 Mahalanobis 距离取决于数据的均值、标准差和相关性的估计值。为每个观测编号都标绘距离。通过突出显示具有最大距离值的点来标识极端多元离群值。请参见第 46 页的“Mahalanobis 距离测度”。

Jackknife 距离 显示或隐藏使用 Jackknife 方法计算的距离。使用均值、标准差和相关性矩阵（不含观测本身）的估计值计算每个观测的距离。当存在离群值时 Jackknife 距离会很有用。在这种情况下，Mahalanobis 距离会失真，往往会隐藏离群值或令其他点看起来比实际上离群。请参见第 47 页的“Jackknife 距离测度”。

T^2 显示或隐藏 Mahalanobis 距离平方后的距离。该图更适用于多元控制图，它包含计算的 T^2 统计量的值及其上控制限。落在该限值之外的值可能为离群值。请参见第 47 页的“T^2 距离测度”。

图 3.6 离群值分析图



保存距离和值

您可以通过从该图的红色小三角菜单中选择**保存**选项，将所有距离保存至数据表。

注意：没有任何公式与 Jackknife 距离列一同保存。这意味着您修改数据表后**不**会重新计算该距离。若在数据表中添加/删除列或更改值，请再次选择**分析 > 多元方法 > 多元**以计算新的 Jackknife 距离。

除了保存每行的距离值，还会创建一个列属性，用于对指定的“离群值分析”类型提供上控制限 (UCL) 值。

项目信度

项目信度指示一组工具测量总响应的一致性。 Cronbach α 值 (Cronbach 1951) 是一种信度测度。 Cronbach α 值的两个主要应用分别为工具信度和问卷分析。

Cronbach α 值基于在测量值尺度内项目的相关性平均值。它相当于计算数据表中所有折半相关性的平均值。若项目的方差变化很大，则可以请求计算“标准化 α”。

注意：Cronbach α 值与显著性水平 α 没有关系。此外，项目信度与生存时间信度分析无关。

要查看单个项目的影响，JMP 将从计算中排除该项目，并显示 Cronbach α 值的影响。若 α 值在您排除某个变量（项目）后增大，则该变量不与其他变量高度相关。若 α 值减小，您可以得出结论：该变量在该尺度上与其他项目相关。

有关计算的详细信息，请参见第 48 页的“Cronbach α 值”。

补缺缺失数据

要补缺缺失数据，请点击“多元”红色小三角并选择**补缺缺失数据**。随即创建新的数据表。该数据表与您的数据表完全相同，只不过缺失值由估计值代替。

补缺值是以每行的非缺失值为前提条件的期望值。通过启动窗口中选择的方法来估计的均值和协方差矩阵用于补缺计算。所有多元检验和选项随后可用于补缺数据集。

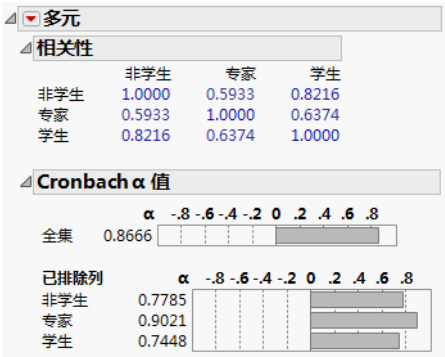
“项目信度”示例

本例使用 sample data 文件夹中的 Danger.jmp 数据。该表列出具有一定程度的固有危险的 30 个项目。有三组人群 （学生、非学生、专家）根据自己所认为的危险程度对这些项目进行了排名。请注意，学生和非学生都将 “Nuclear power”评级为非常危险 (1)，但专家给出的排名却靠后 (20)。另一方面，全部三个判断组都将 “motorcycles”排在第五位或第六位。

您可以使用 Cronbach α 值按照各组所认为的项目排名方式来评估一致性。请注意，在这类示例中，值对每组而言都是相同的一组排名，将数据标准化没有任何影响。

1. 选择**帮助** > **样本数据库**，然后打开 Danger.jmp。
2. 选择**分析** > **多元方法** > **多元**。
3. 选择除**活动**之外的所有列并点击 **Y, 列**。
4. 点击**确定**。
5. 点击 “多元”红色小三角并选择**项目信度** > Cronbach α 值。
6. （可选）点击 “多元”红色小三角并选择**散点图矩阵**以隐藏该图。

图 3.7 “Cronbach α 值”报表



“Cronbach α 值”结果显示总 α 值为 0.8666，该值指示三个组中的排名值具有较高的相关性。不仅如此，当您从分析中排除专家后，**非学生**和**学生**对危险的排名几乎相同，二者的 Cronbach α 值得分分别为 0.7785 和 0.7448。

计算和统计详细信息

- “估计方法”
- “Pearson 乘积矩相关系数”
- “关联的非参数测度”
- “逆相关性矩阵”
- “距离测度”
- “Cronbach α 值”

估计方法

REML

当数据包含缺失值时，与 ML（最大似然）估计方法相比，REML（限制最大似然）估计值的偏倚更小。REML 方法基于误差对比将边缘似然最大化。REML 方法经常用于估计方差和协方差。“主成分”平台中的 REML 方法与针对重复测量数据（具有非结构化协方差矩阵）使用的混合模型的 REML 估计相同。有关混合模型的 REML 估计的信息，请参见 SAS Institute Inc.(2020e) 中的“MIXED 过程”一章。

稳健

该方法实际上通过极大地降低所有离群值的权重来忽略它们。使用以下权重执行一系列迭代再加权数据拟合：

若 $Q < K$，则 $w_i = 1.0$；其他情况下 $w_i = K/Q$

在此， K 是一个等于卡方分布的 0.75 分位数的常数，该分布的自由度等于数据表中的列数。 Q 定义如下：

$$Q = (y_i - \mu)^T (S^2)^{-1} (y_i - \mu)$$

在该等式中，y_i = 第 i 个观测的响应，μ = 均值向量的当前估计值，S^2 = 协方差矩阵的当前估计值，T = 转置矩阵运算。最后一步是减小方差矩阵的偏倚。

这是一个折衷方法：当数据中的离群值不多时，您可以得到较高的方差估计值；但当数据确实包含离群值时，您可以得到准确得多的方差估计值。

Pearson 乘积矩相关系数

Pearson 乘积矩相关系数测量两个变量之间线性关系的强度。对于响应变量 X 和 Y，该系数表示为 r，计算公式如下所示：

$$r = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{\sum (x - \bar{x})^2} \sqrt{\sum (y - \bar{y})^2}}$$

若两个变量之间存在精确线性关系，则相关系数为 1 或 –1，具体取决于变量之间是正相关还是负相关。若不存在线性关系，则相关系数趋向于 0。

关联的非参数测度

对于 Spearman、Kendall 或 Hoeffding 相关性，数据首先要进行秩排序。随后根据数据值的秩执行计算。若存在结值，则使用平均秩。

注意：当指定了“权重”变量时，缺失值和零值权重会从非参数相关性计算中排除。所有其他权重值均视为 1。

Spearman ρ (rho) 系数

使用上述 Pearson 相关性公式基于数据的秩计算 Spearman ρ 相关系数。

Kendall τ_b 系数

Kendall τ_b 系数基于一致对和不一致对的数目。若两个变量所对应的一对行对于哪一个变量较大这一点上保持一致，则认为这一对行**一致**。否则这一对行不一致或为同分对。

公式

$$\tau_b = \frac{\sum_{i<j} \text{sgn}(x_i - x_j)\text{sgn}(y_i - y_j)}{\sqrt{(T_0 - T_1)(T_0 - T_2)}}$$

计算 Kendall τ_b，其中：

$$T_0 = (n(n-1))/2$$

$$T_1 = \sum((t_i)(t_i-1))/2$$

$$T_2 = \sum((u_i)(u_i-1))/2$$

请注意以下几点：

- 若 z>0，则 sgn(z) 等于 1；若 z=0，则 sgn(z) 等于 0；若 z<0，则 sgn(z) 等于 -1。
- t_i (u_i) 是第 i 组 x (y) 结值中的 x (y) 结值的数目。
- n 是观测数。
- Kendall τ_b 的取值范围为 –1 到 1。若指定了权重变量，则忽略该统计量。

按照以下方式执行计算：

- 根据第一个变量的值对观测进行秩排序。
- 然后根据第二个变量的值对观测重新进行秩排序。
- 第一个变量的交换次数用于计算 Kendall τ_b。

Hoeffding D 统计量

Hoeffding D (1948) 的公式如下

$$D = 30\left(\frac{(n-2)(n-3)D_1 + D_2 - 2(n-2)D_3}{n(n-1)(n-2)(n-3)(n-4)}\right)$$

其中：

$$D_1 = \Sigma_i(Q_i-1)(Q_i-2)$$

$$D_2 = \Sigma_i(R_i-1)(R_i-2)(S_i-1)(S_i-2)$$

$$D_3 = \Sigma_i(R_i-2)(S_i-2)(Q_i-1)$$

请注意以下几点：

- R_i 和 S_i 分别是 x 值和 y 值的秩。
- Q_i（有时称为二元秩）是 1 加上 x 值和 y 值均小于第 i 个点的点数。
- 在其 x 值或 y 值上有结值的点（而不是这两个值上同时具有结值的点）为 Q_i 贡献了 1/2（若其他值小于第 i 个点的相应值）。同时在 x 和 y 上有结值的点为 Q_i 贡献了 1/4。

若观测中没有结值，D 统计量的值介于 –0.5 和 1 之间，其中 1 指示完全依赖。若指定了权重变量，则忽略该统计量。

逆相关性矩阵

逆相关性矩阵提供了有用的多元信息。逆相关性矩阵的对角线元素有时称为方差膨胀因子 (VIF)，它是一个函数，表示为一个变量在多大程度上接近由其他变量构成的线性函数。具体而言，若相关性矩阵表示为 **R** 并且逆相关性矩阵表示为 $\mathbf{R}^{-1}$，则对角线元素表示为 r^{ii}，它的计算公式如下：

$$r^{ii} = \mathrm{VIF}_i = \frac{1}{1 - \mathrm{R}_i^2}$$

其中，R_i^2 是用第 i 个解释变量对其他解释变量作回归的模型的变异系数。因此，较大的 r^{ii} 指示第 i 个变量与任意数量的其他变量高度相关。

距离测度

“离群值分析”图显示数据表中每个点的指定距离测度。

Mahalanobis 距离测度

Mahalanobis 距离将数据的相关性结构和不同的尺度考虑在内。对于每个值， Mahalanobis 距离都表示为 M_i，它的计算公式如下：

$$M_i = \sqrt{(\mathrm{Y}_i - \bar{\mathrm{Y}})'\mathrm{S}^{-1}(\mathrm{Y}_i - \bar{\mathrm{Y}})}$$

其中：

Y_i 是第 i 行的数据

$\bar{Y}$ 是均值行

S 是数据的估计协方差矩阵

“Mahalanobis 距离”图上绘制的上控制限参考线 (Mason and Young, 2002) 计算如下：

$$UCL_{Mahalanobis} = \sqrt{\frac{(n-1)^2}{n}\beta_{\left[1-\alpha;\frac{p}{2};\frac{n-p-1}{2}\right]}}$$

其中：

n = 观测数

p = 变量 （列）数

$\beta_{\left[1-\alpha;\frac{p}{2};\frac{n-p-1}{2}\right]}$ = Beta $\left(\frac{p}{2}, \frac{n-p-1}{2}\right)$ 分布的第 (1–α) 分位数

若某个变量是其他变量的精确线性组合，则相关性矩阵是奇异矩阵，并且该变量的行和列将被清零。生成的广义逆仍可用于距离计算。

Jackknife 距离测度

使用均值、标准差和相关性矩阵 （不含观测本身）的估计值计算 Jackknife 距离。对于每个值，Jackknife 距离计算如下：

$$J_i = \sqrt{\frac{(n-2)n^2}{(n-1)^3} \times \frac{M_i^2}{1-\frac{nM_i^2}{(n-1)^2}}}$$

其中：

n = 观测数

p = 变量 （列）数

M_i = 第 i 个观测的 Mahalanobis 距离

“Jackknife 距离”图上绘制的上控制限参考线 (Penny, 1996) 计算如下：

$$UCL_{Jackknife} = \sqrt{\frac{(n-2)n^2}{(n-1)^3} \times \frac{UCL_{Mahalanobis}^2}{1-\frac{n \bullet UCL_{Mahalanobis}^2}{(n-1)^2}}}$$

T^2 距离测度

T^2 距离是 Mahalanobis 距离的平方，所以 $T_i^2 = M_i^2$。

T^2 距离的上控制限为：

$$UCL_{T^2} = \frac{(n-1)^2}{n}\beta_{\left[1-\alpha;\frac{p}{2};\frac{n-p-1}{2}\right]} = (UCL_{Mahalanobis})^2$$

其中

n = 观测数

p = 变量 （列）数

$\beta_{\left[1-\alpha;\frac{p}{2};\frac{n-p-1}{2}\right]}$ = Beta $\left(\frac{p}{2}, \frac{n-p-1}{2}\right)$ 分布的第 (1–α) 分位数

多元距离可用于发现多维数据中的离群值。不过，若变量在多元意义上高度相关，那么某个点在多元空间中会被视为离群值，但沿着任意维子集的方向却无异常之处。换言之，若各值相关，沿着一两个轴查看某个点时，该点可能并不值得注意，但因为违反了相关性仍可能为离群值。

Cronbach α 值

Cronbach α 值定义如下：

$$\alpha = \frac{kc}{v + (k-1)c}$$

其中

k = 尺度范围中的项目数

c = 项目之间的平均协方差

v = 项目之间的平均方差

若项目经过标准化具有常数方差，公式将变为

$$\alpha = \frac{k(r)}{1 + (k-1)r}$$

其中

r = 项目之间的相关性平均值

总体 α 系数越大，您越能够自信地认为您的项目对尺度或检验的信度有贡献。若有许多高度相关的项目，该系数可能接近 1.0。

第 4 章

主成分

对数据降维

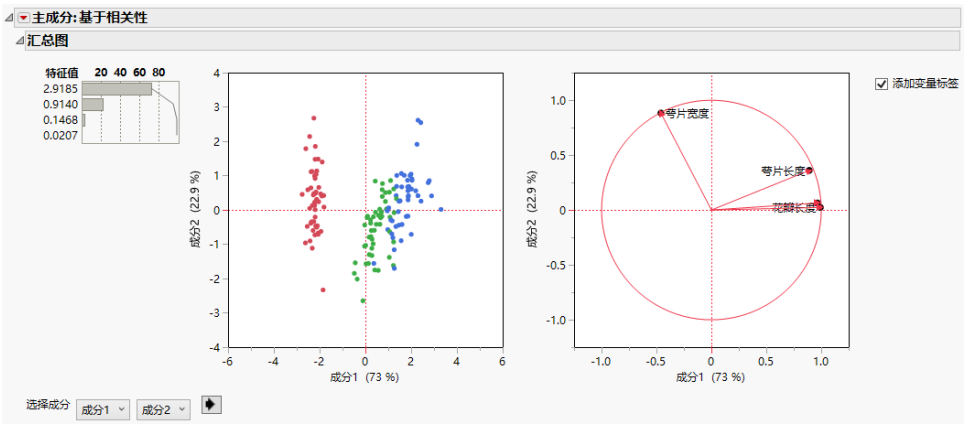
主成分分析的目的是要从一组测量变量中得到少数几个相互独立的线性组合（**主成分**），使用它们来捕获原始变量中尽可能多的变异性。主成分分析是一种降维方法，也是一种探索性数据分析工具。主成分分析也可用于构造预测模型，如**主成分分析回归**（亦称 PCA 回归或 PCR）所述。

对于包含大量变量的数据，“主成分”平台提供称为“宽”方法的估计方法。“宽”方法使您能够在较短的计算时间内计算主成分。随后可将这些主成分用在 PCA 回归中。

对于包含大量零的数据，也称为**稀疏数据**，“主成分”平台提供“稀疏”估计方法。与“宽”方法类似，“稀疏”方法能够以较短的计算时间计算主成分。与“宽”方法的不同之处在于，“稀疏”方法计算用户定义的固定数量的主成分，而不是完整集合。

“主成分”平台还支持因子分析。JMP 提供若干类型的正交和斜交因子分析样式的旋转，用来帮助解释提取的成分。有关因子分析的信息，请参见第 189 页的“因子分析”一章。

图 4.1 主成分示例



目录

“主成分分析”平台概述 51
主成分分析的示例 51
启动“主成分”平台 52
缺失数据 54
“主成分”报表 54
“主成分”报表选项 55
离群值分析 63
“主成分分析”平台的统计详细信息 64
估计方法 64
DModX 计算 66
离群值分析中的计算 66

“主成分分析”平台概述

主成分分析采用一组变量的少数几个相互独立的线性组合（**主成分**）的形式，对这组变量中的变异建模。

若您想要查看各点在许多相关变量中的排列，则可以使用主成分分析显示高维数据的最显著方向。使用主成分分析可减少一组数据的维度。主成分是使用尽量少的变量尽可能完全标绘数据结构的一种方式。

对于 p 个变量，这是 p 个主成分形成的方式：

- 第一主成分是进行标准化之后的原始变量的线性组合，具有最大可能方差。
- 随后的每个主成分都是具有最大可能方差并且与之前定义的所有成分都无关的变量的线性组合。

通过将相关性矩阵（协方差矩阵或平方和与叉积矩阵）的特征向量与变量进行线性组合来计算每个主成分。特征值表示每个成分的方差。

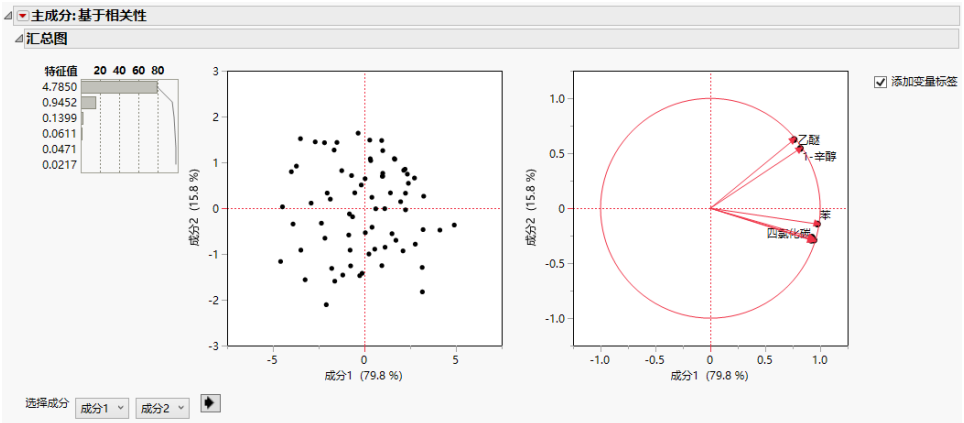
“主成分”平台使您能够针对相关性矩阵、协方差矩阵或未统一尺度的数据执行分析。您也可以在“主成分”平台内执行“因子分析”。请参见第 189 页的“因子分析”一章。

主成分分析的示例

要查看双因子数据表的 “主成分分析”报表示例，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Solubility.jmp。
2. 选择**分析** > **多元方法** > **主成分**。
3. 选择所有连续列并点击 **Y, 列**。
4. 保留默认估计方法，然后点击**确定**。

图 4.2 “主成分 : 基于相关性”报表



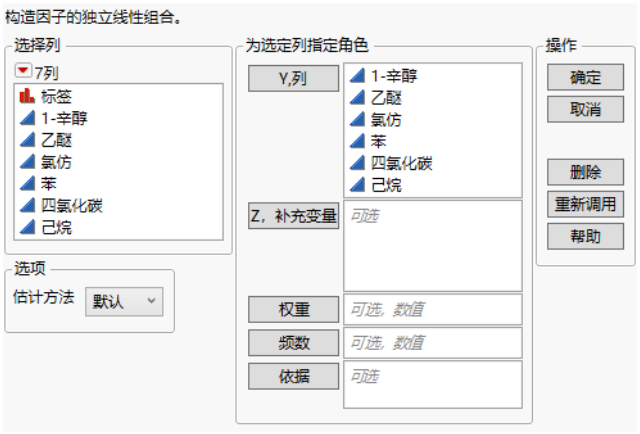
该报表提供特征值和一个条形图，图中显示了每个主成分所解释的变异百分比。在本例中，第一个主成分解释数据中大约 80% 的变异。前两个主成分一起，共同解释了数据中几乎全部的变异 (95.5%)。同时还有一个“得分图”和一个“载荷图”。请参见第 54 页的““主成分”报表”。

启动“主成分”平台

通过选择**分析 > 多元方法 > 主成分**来启动“主成分”平台。“多元”平台和“三维散点图”平台也提供主成分分析。

第 51 页的“主成分分析的示例”中所述的示例使用 Solubility.jmp 样本数据表中的所有连续变量。

图 4.3 “主成分”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 要进行成分分析的变量。

Z，补充变量 要显示的补充变量。补充变量不包含在主成分计算中，包括它们不会影响结果。连续型补充变量可以投影到载荷图中，用于增强解释。

权重 标识一列，该列的数值为分析中的每一行都分配一个权重。

注意：“宽”估计方法和“稀疏”估计方法忽略“权重”角色。

频数 标识一列，该列的数值为分析中的每一行都分配一个频数。

注意：“宽”估计方法和“稀疏”估计方法忽略“频数”角色。

依据 为“依据”列指定的每个值创建“主成分”报表，以便您可以为每个组执行单独的分析。

估计方法 指定计算相关性的方法。其中有些方法解决了缺失数据的处理。

默认 **默认**选项使用“逐行”、“配对”或“REML”方法。“JMP 警示”还建议在适当的时候切换至“宽”方法。

– **逐行**估计用于不含缺失值的数据表。

– **配对**估计用于包含缺失值并且多于 10 列、多于 5,000 行或列数多于行数的数据表。

– 在其他情况下使用 REML 估计。

– 对于数据表超过 500 列并且列数多于行数的情况，“JMP 警示”窗口会建议使用**宽**估计。这是因为在列数过多时使用其他方法，计算时间会相当长。点击**宽**切换至“宽”方法，或点击**继续**使用原本选择的方法。

REML 限制最大似然 (REML) 估计使用所有数据，即使存在缺失值。由于存在偏倚修正因子，若您的数据集很大且包含许多缺失值，该方法会很慢。因此，REML 最适用于较小的数据集。若数据中不含缺失单元格，则 REML 和 ML 估计值等价于样本协方差矩阵。若存在缺失单元格，与 ML 估计相比，REML 的方差和协方差估计值的偏倚更小。有关统计详细信息，请参见第 64 页的“REML”。

ML 最大似然 (ML) 估计使用所有数据，即使存在缺失值。由于 ML 的估计值生成速度更快，该方法最适用于包含缺失数据的大数据表。

稳健 稳健估计使用所有数据，即使存在缺失值。该方法降低了极值的权重，因此最适用于可能具有离群值的数据表。有关统计详细信息，请参见“相关性和多元方法”一章中第 44 页的“稳健”。

逐行 逐行估计为每对列计算 Pearson 相关性系数。有关统计详细信息，请参见“相关性和多元方法”一章中第 44 页的“Pearson 乘积矩相关系数”。逐行估计不使用包含缺失值的观测。该方法可用于排除包含缺失数据的所有观测。

配对 配对估计使用所有数据，即使存在缺失值。该估计方法使用这两列中不含缺失值的所有观测为每对列计算 Pearson 相关性系数。有关统计详细信息，请参见“相关性和多元方法”一章中第 44 页的“Pearson 乘积矩相关系数”。配对估计最适用于包含缺失值并且列数多于行数、多于 10 列或多于 5,000 行的数据表。

宽 宽估计不使用含缺失值的观测，因此在应用该方法之前会删除包含缺失单元格的行。该估计方法使用基于完整奇异值分解的算法。该算法避免计算协方差矩阵，因此计算效率较高，适用于数据中列非常多的情况。有关统计详细信息，请参见第 65 页的“宽”。

JMP PRO **稀疏** 稀疏估计使用所有数据，即使存在缺失值。该估计方法使用基于部分奇异值分解的算法，它只计算第一个指定数量的奇异值和奇异值向量。该算法避免计算协方差矩阵以及不必要的主成分，因此计算效率较高，当数据稀疏（即包含很多零时）或当数据中具有大量列时，该算法非常有用。有关统计详细信息，请参见第 65 页的“稀疏”。

注意：若您选择“REML”、“ML”或“稳健”，而您的数据表中的列数多于行数并且具有缺失值，JMP 会将“估计方法”切换为“配对”。

成分数 （仅当将“稀疏”指定为估计方法时才可用。）指定要估计的成分数。通常，成分数远小于数据的维。

缺失数据

不同的估计方法配备了多种方式处理缺失数据。您还可以采用以下方式估计缺失值：

- 使用**多元方法** > **多元**下面的“补缺缺失数据”选项。请参见“相关性和多元方法”一章中第 42 页的“补缺缺失数据”。
- 使用**分析** > **筛选** > **探索缺失值**下面的“多元正态补缺”或“多元 SVD 补缺”实用工具。请参见《预测和专业建模》。

“主成分”报表

若选定了“宽”和“稀疏”之外的任何估计方法，则最初显示的是“主成分: 基于相关性”报表。（若从“主成分”红色小三角菜单中为“主成分”选项选择“基于协方差”或“基于未统一尺度”，该报表的标题将随之更改。）

若您选择“宽”方法，则出现“宽主成分”报表。若您选择“稀疏”方法，则出现“稀疏主成分”报表。

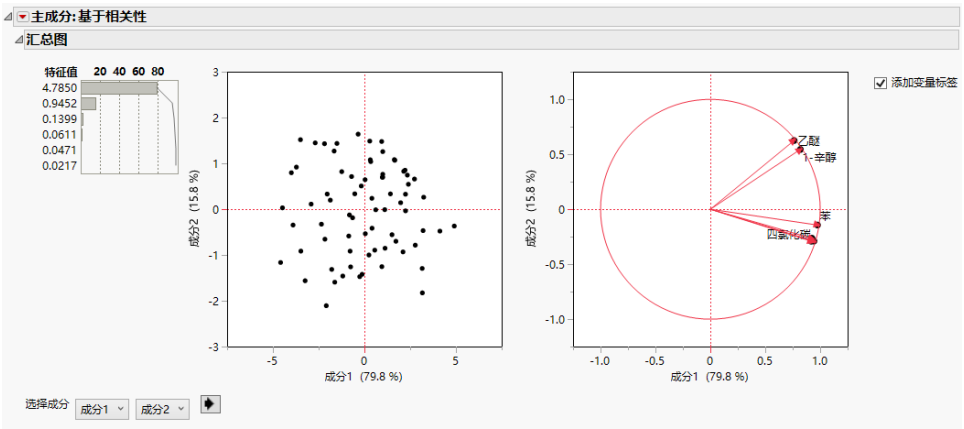
初始“主成分”报表是基于相关性的分析。该报表使用主成分汇总了指定 *Y* 变量的变异（图 4.4）。通过从红色小三角菜单中选择“主成分”选项，您可以切换到基于协方差矩阵或未统一尺度数据的分析。

根据您的选择，主成分可以由以下任一矩阵的特征值分解推导得出：

- 相关性矩阵
- 协方差矩阵
- 未统一尺度和未中心化数据的平方和与叉积矩阵

报表中的详细信息显示主成分如何吸收数据变异。主成分点从变量的特征向量线性组合推导得出。

图 4.4 “主成分 : 基于相关性”报表



该报表提供特征值和一个条形图，图中显示了每个主成分所解释的变异百分比。同时还有一个“得分图”和一个“载荷图”。特征值指示基于每个成分贡献的方差量提取的总成分数。

“得分图”绘制每个成分相对于其他成分的计算值，按照均值和标准差调整了每个值。

“载荷图”绘制变量与成分之间的未旋转载荷矩阵。值越接近 1，成分对变量的影响越大。

默认情况下，报表为前两个主成分显示“得分图”和“载荷图”。使用“选择成分”旁边的列表指定在“得分图”和“载荷图”上绘制的主成分。

“主成分”报表选项

“主成分”红色小三角菜单包含以下选项：

注意：某些选项对于“宽”估计方法或“稀疏”估计方法不可用。

主成分 （对“宽”估计方法或“稀疏”估计方法不可用。）支持您基于**相关性**、**协方差**或**未统一尺度**创建主成分。

相关性 （对“宽”估计方法或“稀疏”估计方法不可用。）变量之间的相关性矩阵。

注意：对角线上的值为 1.0。

协方差矩阵 （对“宽”估计方法或“稀疏”估计方法不可用。）显示或隐藏变量的协方差。

特征值 按顺序从大到小列出与各个主成分对应的特征值。特征值表示多元样本中总变异的一部分。

特征值的尺度取决于您选择用于提取主成分的矩阵：

- 对于“基于相关性”选项，特征值已统一尺度为总和等于变量数。
- 对于“基于协方差”选项，特征值不统一尺度。
- 对于“基于未统一尺度”选项，特征值需要除以总观测数。

若从红色小三角菜单中选择 **Bartlett 检验**选项，将为每个特征值提供假设检验 （图 4.6）(Jackson, 2003)。

图 4.5 特征值

特征值

数目	特征值	百分比	20 40 60 80	累积百分比
1	4.7850	79.750		79.750
2	0.9452	15.754		95.504
3	0.1399	2.331		97.835
4	0.0611	1.018		98.853
5	0.0471	0.785		99.638
6	0.0217	0.362		100.000

特征向量 按顺序从左到右显示或隐藏每个主成分的特征向量的表。使用这些系数形成原始变量的线性组合，可生成主成分变量。根据标准规范，特征向量的模为 1。

注意：显示的特征向量数等于相关性矩阵的秩或在启动窗口中所指定的成分数（若选择“稀疏”方法）。

Bartlett 检验 （对“宽”估计方法或“稀疏”估计方法不可用。）显示或隐藏齐性检验的结果（追加至“特征值”表）。该检验通过计算检验的卡方、自由度 (DF) 和 *p* 值（概率 > 卡方）来确定特征值是否具有相同的方差。请参见 Bartlett (1937, 1954)。

图 4.6 Bartlett 检验

特征值

数目	特征值	百分比	20 40 60 80	累积百分比	卡方	自由度	概率>卡方
1	4.7850	79.750		79.750	701.245	11.243	<.0001*
2	0.9452	15.754		95.504	317.186	13.125	<.0001*
3	0.1399	2.331		97.835	58.444	9.270	<.0001*
4	0.0611	1.018		98.853	17.589	5.280	0.0044*
5	0.0471	0.785		99.638	9.723	1.899	0.0069*
6	0.0217	0.362		100.000	0.000	.	.

载荷矩阵 显示或隐藏每个成分的载荷表。这些值绘制到载荷图中。表值的透明度指示绝对载荷值与 0 之间的距离。越靠近 0 的绝对载荷值比远离 0 的绝对载荷值更加透明。

若指定补充变量，则会为每个补充连续变量和补充分类变量的每个水平显示一个附加坐标表。这些值绘制到连续补充变量的载荷图中。

载荷和坐标的尺度取决于您选择用于提取主成分的矩阵：

- 对于“基于相关性”选项，载荷的第 *i* 个列是第 *i* 个特征向量乘以第 *i* 个特征值的平方根。第 *i,j* 个载荷是第 *i* 个变量与第 *j* 个主成分之间的相关性。
- 对于“基于协方差”选项，载荷的第 *i* 个列中的第 *j* 个条目是第 *i* 个特征向量乘以第 *i* 个特征值的平方根再除以第 *j* 个变量的标准差。第 *i,j* 个载荷是第 *i* 个变量与第 *j* 个主成分之间的相关性。
- 对于“基于未统一尺度”选项，载荷的第 *i* 个列中的第 *j* 个条目是第 *i* 个特征向量乘以第 *i* 个特征值的平方根再除以第 *j* 个变量的标准误差。第 *j* 个变量的标准误差是平方和与叉积矩阵的第 *j* 个对角线元素除以行数 ($\mathbf{X}'\mathbf{X}/n$)。

注意：分析未统一尺度数据时，第 *i,j* 个载荷**不**是第 *i* 个变量与第 *j* 个主成分之间的相关性。

格式化的载荷矩阵 显示或隐藏每个成分的载荷表。该表按第一个主成分的载荷降序排序。因此，变量按照第一个成分的载荷降序列出。

图 4.7 格式化的载荷矩阵

格式化的载荷矩阵

	主成分1	主成分2	主成分3	主成分4	主成分5	主成分6
苯	0.974761	-0.143460	-0.081914	-0.023369	-0.108186	-0.101488
四氯化碳	0.942849	-0.289099	0.069134	-0.059654	-0.099757	0.095746
己烷	0.923483	-0.263641	0.256572	0.026770	0.099672	-0.034523
氯仿	0.917422	-0.290351	-0.242516	0.075629	0.093454	0.027695
1-辛醇	0.819019	0.544318	-0.041397	-0.162739	0.068711	0.002762
乙醚	0.761978	0.625283	0.044774	0.155130	-0.045337	0.016883

隐藏小于该值的绝对载荷值: 0.3
文本变暗 0.4

隐藏小于该值的绝对载荷值 该值确定哪些载荷在“格式化的载荷矩阵”报表中不可用。您可以使用文本框或滑块将绝对值落在选定值之下的载荷灰显。

文本变暗 “格式化的载荷矩阵”报表中的灰显值的透明度。您可以使用文本框或滑块设置灰显载荷的透明度。透明度的范围从 0 到 1，值越小越透明。例如，将透明度设置为 0 时完全将不可用载荷从矩阵中移除，设置为 1 时载荷仍可看到。

变量的平方余弦 显示或隐藏包含变量的平方余弦的表。若指定补充变量，则会显示补充变量平方余弦的附加表。对于每个变量，各主成分的平方余弦值的总和等于 1。平方余弦使您能够查看主成分表示变量的好坏程度。您还可以确定表示特定变量需要多少个主成分。该选项还显示前三个主成分的平方余弦图。

注意：若使用 “稀疏”估计方法并且选择的成分数小于 3，则在图上只显示指定的成分数。

变量的部分贡献 显示或隐藏包含变量的部分贡献的表。部分贡献使您能够查看每个变量对每个主成分的贡献百分比。该选项还显示前三个主成分的部分贡献图。

注意：若使用 “稀疏”估计方法并且选择的成分数小于 3，则在图上只显示指定的成分数。

汇总图 显示或隐藏在默认报表中生成的汇总信息。该汇总信息包括特征值图、得分图和载荷图。默认情况下，报表为前两个主成分显示得分图和载荷图。报表中提供选项，可用于指定要绘制的主成分。请参见第 54 页的 “‘主成分’报表”。

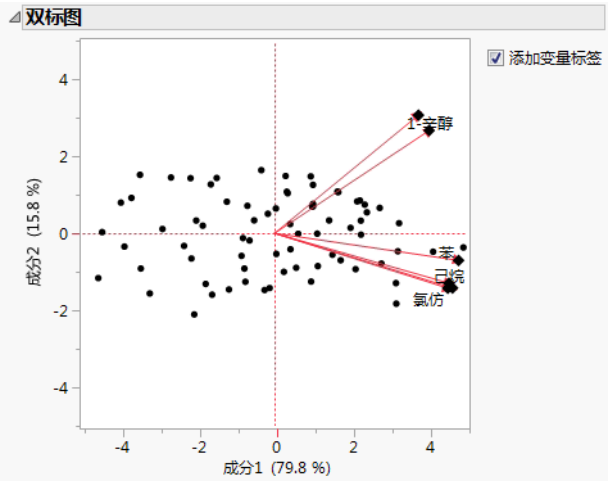
注意：若您的数据包含缺失值，则插补得分将绘制在得分图上。

提示：选择载荷图中的箭头尖端可选择数据表中对应的列。按 Ctrl 键并点击某个箭头尖端可取消选择相应的列。

双标图 显示或隐藏一个图，用来叠加指定成分数的得分图和载荷图。

注意：若您的数据包含缺失值，则插补得分将绘制在 “双标图”上。

图 4.8 双标图

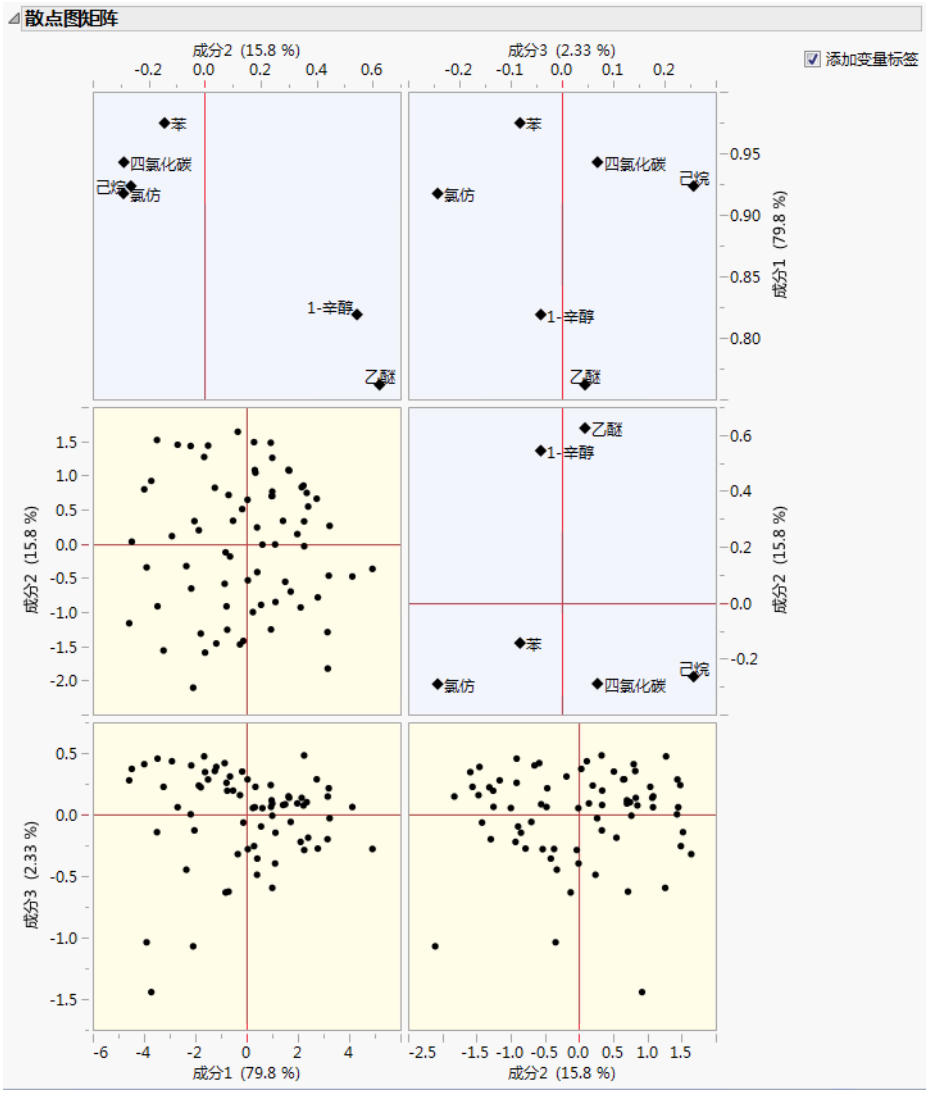


注意：得分图标记为圆点，载荷图标记为菱形。

散点图矩阵 显示或隐藏指定主成分数的得分图和载荷图的矩阵。散点图矩阵在一个空间内排列得分图和载荷图。得分图具有黄色阴影背景。载荷图具有蓝色阴影背景。

注意：若您的数据包含缺失值，则插补得分将绘制在 “散点图矩阵”中。

图 4.9 散点图矩阵



注意：显示在 “散点图矩阵”中的载荷图矩阵是您选择 “载荷图”选项时获得的载荷图矩阵的转置矩阵。

陡坡图 显示或隐藏每个成分的特征值图。该陡坡图帮助直观演示数据空间的维度。

得分图 显示或隐藏指定成分数的成对主成分的得分散点图矩阵。该图显示在图 4.4 中 （最左侧图）。

载荷图 显示或隐藏指定成分数的因子载荷的二维表示矩阵。若变量数未超过 30 个，则载荷图会对变量添加标签。若变量数超过 30 个，则默认不显示标签。该信息显示在图 4.4 中（最右侧图）。

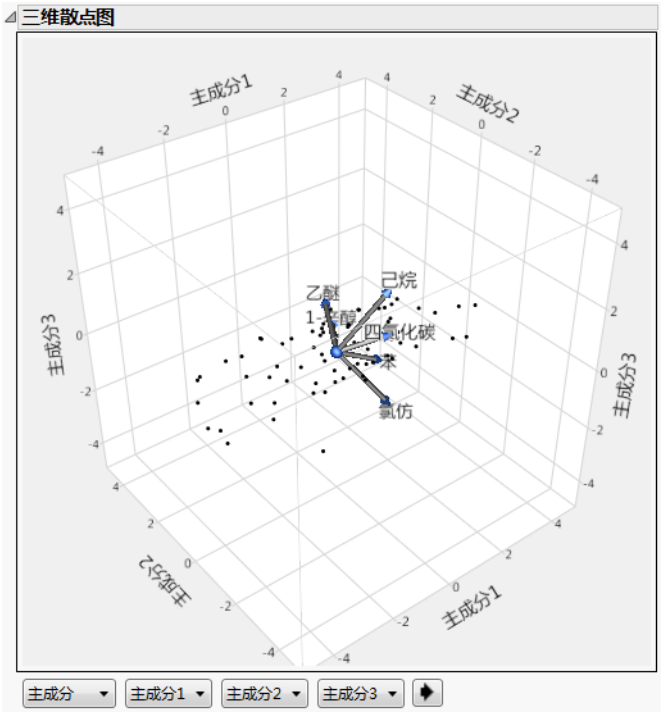
提示: 选择载荷图中的箭头尖端可选择数据表中对应的列。按 Ctrl 键并点击某个箭头尖端可取消选择相应的列。

完成补缺的得分图 （对“宽”估计方法或“稀疏”估计方法不可用。）补缺所有缺失值并创建得分图。仅当包含缺失值时，该选项才可用。

三维得分图 （对“宽”估计方法或“稀疏”估计方法不可用。）显示或隐藏任何三个主成分得分的三维散点图。首次调用该命令时，将显示前三个主成分。

注意：若您的数据包含缺失值，则插补得分将绘制在“三维得分图”上。

图 4.10 三维散点图



图来源 图中数据点的来源。可用选项为“主成分”、“旋转主成分”和“数据列”。

轴控件 每个轴的内容。若选择“主成分”选项或“旋转成分”选项，则“轴控件”的选项为“主成分”。若选择“数据列”选项，则选项为分析中的变量。

循环按钮 在所有轴内容可能性中循环。

变量在图中显示为射线。这些射线，称为**双标图射线**，将变量近似计算为轴上主成分的函数。若只有两三个变量，射线就正好代表变量。射线对应于主成分载荷。

得分椭圆 在每对主成分的汇总得分图上显示或隐藏椭圆。椭圆既可以构造为基于 alpha 水平的置信椭圆，也可以构造为基于观测距中心的距离的控制限椭圆。默认情况下，这些椭圆为 95% 置信椭圆。

得分椭圆覆盖率 显示一个支持您更改得分椭圆的构造方式的子菜单。按照置信水平或按照用 *k* sigma 表示的距中心的距离来指定得分椭圆。置信水平 *p* 与 *k* sigma 之间的关系为 $p = 1 - \exp(-k^2/2)$。

显示选项

箭头线 允许您在可显示箭头的所有图上显示或隐藏箭头。若变量数不超过 1000 个，则显示箭头。若变量数超过 1000 个，则默认不显示箭头。

显示补充变量 （仅当您指定补充变量时可用。）在双标图、得分图和载荷图中，显示或隐藏连续补充变量的箭头线或分类补充变量的标签标记。

离群值分析 显示或隐藏“离群值分析”报表，该报表支持您通过 T^2 和贡献统计量检测数据中的离群值。请参见第 63 页的 “离群值分析”。

因子分析 （对“宽”估计方法或“稀疏”估计方法不可用。）用主成分法进行因子旋转得到的因子分析，或者其他方法得到的因子分析。请参见第 189 页的 “因子分析”一章。

JMP PRO **聚类变量** （对“宽”估计方法或“稀疏”估计方法不可用。）通过将变量划分为不重叠的聚类，对这些变量执行聚类分析。变量聚类提供将类似变量分组到代表组中的方法。每个聚类随后可由单个成分或变量表示。成分是聚类中所有变量的线性组合。或者，聚类也可由标识为聚类中最典型成员的变量来表示。请参见第 295 页的 “聚类变量”一章。

注意：“聚类变量”对所有计算都使用相关性矩阵，即便您选择“基于协方差”或“基于未统一尺度”选项也是如此。

保存主成分 将您指定的主成分数与用于计算每个成分的公式一同保存到数据表中。该公式无法计算包含缺失值的行。

主成分的计算取决于您选择用于提取主成分的矩阵：

- 对于“基于相关性”选项，第 *i* 个主成分是将第 *i* 个特征向量的条目用作系数的中心化和统一尺度观测的线性组合。
- 对于“基于协方差”选项，第 *i* 个主成分是将第 *i* 个特征向量的条目用作系数的中心化观测的线性组合。
- 对于“基于未统一尺度”选项，第 *i* 个主成分是将第 *i* 个特征向量的条目用作系数的原始观测的线性组合。

注意：若指定的成分数超过相关性矩阵的秩，则保存的成分数设置为相关性矩阵的秩。

保存预测值 将预测变量 （给定了指定数量的主成分）保存到数据表中的新列。

保存 DModX 将到主成分模型的观测距离 (DModX)（给定了指定数量的主成分）保存到数据表的新列中。较大的 DModX 值表示数据中存在轻中度的离群值。请参见第 66 页的“DModX 计算”。

保存单值平方余弦 将单值平方余弦 （给定了指定数量的主成分）保存到数据表中的新列。

保存单值部分贡献 将单值部分贡献 （给定了指定数量的主成分）保存到数据表中的新列。

保存旋转成分 （对 “宽”估计方法或 “稀疏”估计方法不可用。 ）将旋转成分与用于计算成分的公式一同保存到数据表中。该选项仅在使用“因子分析”选项后可用。该公式无法计算包含缺失值的行。

保存完成补缺的主成分 （对 “宽”估计方法或 “稀疏”估计方法不可用。 ）补缺缺失值，并将主成分保存到数据表中。该列包含用于执行补缺并计算主成分的公式。仅当包含缺失值时，该选项才可用。

保存完成补缺的旋转成分 （对 “宽”估计方法或 “稀疏”估计方法不可用。）补缺缺失值，并将旋转成分保存到数据表中。该列包含用于执行补缺并计算旋转成分的公式。使用“因子分析”选项后，若存在缺失值，该选项才可用。

保存插补公式 （仅当数据表包含缺失值时才可用。）对于包含缺失值的列，将包含用于估计缺失值的公式的新列保存到数据表中。新列称为**补缺 _< 列名 >**。

JMP PRO **发布成分公式** 创建指定数量的主成分公式并在 “公式存储库”平台中将它们保存为公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

JMP PRO **发布 DModX 公式** 将基于指定数量的主成分的 DModX 公式另存为“公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

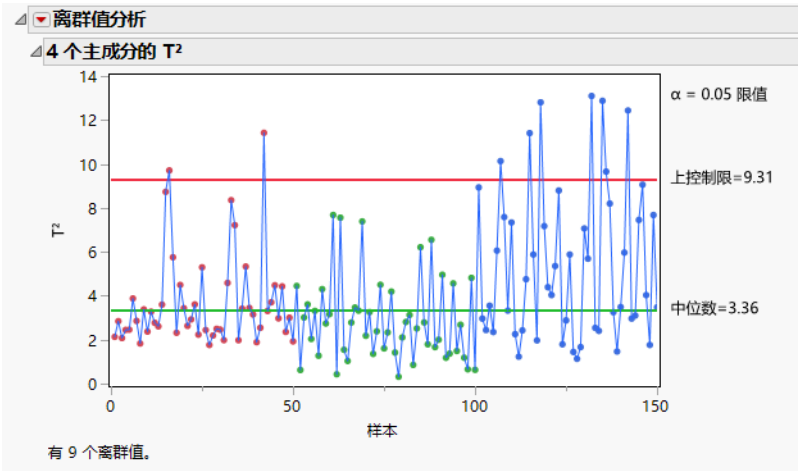
重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

离群值分析

图 4.11 “离群值分析”报表



默认情况下，“离群值分析”报表显示 <*A*> 主成分图的 T^2。该图显示每个观测的 T^2 值以及位于中位数和上控制限 (UCL) 处的水平线。有关如何计算 T^2 值、中位数和上控制限的详细信息，请参见第 66 页的“离群值分析中的计算”。

用于计算上控制限的 α 水平显示在图旁边。检测到的离群值数目显示在图下方。该数字是 T^2 值大于上控制限的观测数。

提示：悬停在 T^2 图中的某个点上方以查看该观测的 T^2 贡献比例图。点击 T^2 贡献比例图将其添加到报表窗口。

“离群值分析”报表选项

T^2 图 显示或隐藏 T^2 图。默认开启。

贡献热图 为所有观测显示或隐藏 T^2 贡献值的热图。

贡献比例热图 显示或隐藏所有观测 （表示为各行的 T^2 的比例）的 T^2 贡献值的热图。通过计算贡献的平方再除以对单个行的所有贡献的平方和，可以得到该单个行的比例。

选定样本的贡献图 （仅当在 T^2 图中选定一个或多个点时才可用。）显示一个报表，其中为每个选定的样本都显示一个 T^2 贡献图。T^2 贡献图显示每个变量对样本的 T^2 统计量的贡献。有关如何计算贡献的详细信息，请参见第 66 页的“离群值分析中的计算”。使用贡献图可调查离群值。具有最大正 / 负贡献的变量是那些对样本贡献最大（具有较大 T^2 值）的变量。请参见第 64 页的“‘贡献图’报表选项”了解有关红色小三角菜单选项的信息。

选定样本的贡献比例图 （仅当在 T^2 图中选定一个或多个点时才可用。）显示一个报表，其中为每个选定的样本都显示一个 T^2 贡献比例图。T^2 贡献比例图显示选定观测的贡献值，这些

值表示为各行的 T^2 的比例。这是 “贡献比例热图”中所示信息的一个不同的演示。请参见第 64 页的 ““贡献图”报表选项”了解有关红色小三角菜单选项的信息。

标准化 DModX 图 （仅当成分数少于变量数时才可用。）显示或隐藏标准化 DModX 值的图。DModX 值可用于检测数据中的中度离群值。

成分数 支持您指定在 T^2 和 T^2 贡献统计量中使用的主成分数。更改成分数时， T^2 图、热图和标准化 DModX 图自动更新。

设置 α 水平 支持您指定 α 水平。

保存 T^2 将 T^2 值保存到数据表中的新列。

保存贡献 将 T^2 贡献保存到数据表中的新列。每个 Y 变量都有一列。

保存标准化 DModX （仅当成分数少于变量数时才可用。）在数据表的新列中保存标准化 DModX 值。

“贡献图”报表选项

选定样本的 T^2 贡献图、选定样本的 T^2 贡献比例图和选定样本的 T^2 均值贡献比例图包含以下红色小三角菜单选项。

直条标签 显示用于为贡献图中的直条添加标签的选项子菜单。标签选项包括“无标签”、“将值用作标签”和“将列用作标签”。

删除图 从报表中删除贡献图。

选定项的控制图 （仅当在贡献图中选择了一个或多个条形段时才可用。）打开 “控制图生成器”窗口，其中包含每个选定过程和组的控制图结果。

提示：也可以通过悬停在贡献图中的条形段上方来查看控制图。点击控制图打开“控制图生成器”窗口。

“主成分分析”平台的统计详细信息

- “估计方法”
- “DModX 计算”
- “离群值分析中的计算”

估计方法

REML

当数据包含缺失值时，与 ML（最大似然）估计方法相比，REML（限制最大似然）估计值的偏倚更小。REML 方法基于误差对比将边缘似然最大化。REML 方法经常用于估计方差和协方差。

“主成分”平台中的 REML 方法与针对重复测量数据（具有非结构化协方差矩阵）使用的混合模型的 REML 估计相同。请参见 SAS PROC MIXED 文档，了解有关混合模型的 REML 估计。

宽

“宽”方法使用可避免计算协方差矩阵的高效计算算法。该算法基于奇异值分解。考虑以下符号：

- n = 行数
- p = 变量数
- $\mathbf{X}$ = 数据值的 $n \times p$ 矩阵

非零特征值数以及得到的主成分数均等于 **X** 的相关性矩阵的秩。非零特征值数不得超过 n 和 p 中较小的那个值。

若选择 “宽”方法，数据将被标准化。要将某个值标准化，需减去其均值，再除以其标准差。用 $\mathbf{X}_s$ 来表示标准化数据值的 $n \times p$ 矩阵。之后，标准化数据的协方差矩阵成为 **X** 的相关性矩阵，该矩阵定义如下：

$$Cov = \mathbf{X}_s'\mathbf{X}_s/(n-1)$$

使用奇异值分解，$\mathbf{X}_s$ 可表示为 $\mathbf{U}Diag(\Lambda)\mathbf{V}'$。这种表示法用于获取 $\mathbf{X}_s'\mathbf{X}_s$ 的特征向量和特征值。主成分或得分通过 $\mathbf{X}_s\mathbf{V}$ 计算得出。更多背景信息，请参见“统计详细信息”附录中第 311 页的“‘宽线性’方法和奇异值分解”。

JMP PRO 稀疏

与“宽”方法类似，“稀疏”方法基于奇异值分解。因此，“稀疏”方法的算法避免计算协方差矩阵，计算效率较高。

考虑对第 65 页的“宽”中所述的 **X** 使用相同的符号和标准化。**X** 的相关性矩阵由 $\mathbf{X}_s$ 的协方差矩阵表示：

$$Cov(\mathbf{X}_s) = \mathbf{X}_s'\mathbf{X}_s/(n-1)$$

“稀疏”方法与“宽”方法的不同之处在于奇异值分解的计算方式。“宽”方法执行完整奇异值分解，而“稀疏”方法使用的算法在奇异值分解时仅计算第一个指定数量的奇异值和奇异值向量。因此，只返回第一个指定数量的特征值和主成分。有关算法的详细信息，请参见 Baglama and Reichel (2005)。

DModX 计算

DModX 是到主成分模型的观测距离，其定义如下：

$$\text{DModX} = \sqrt{\frac{\sum_k e_{ik}^2}{K - A}}$$

其中

e_{ik} = 模型的残差

K = 变量数

A = 主成分数

较大的 DModX 值表示数据中存在轻中度的离群值。

离群值分析中的计算

“离群值分析”报表中的计算使用以下符号。

n = 观测数

A = 主成分数

X_{ci} = 第 i 个观测的标准化数据

T^2 统计量

按以下方式计算第 i 个观测的 T^2 统计量：

$$T_i^2 = X_{ci} P_A L^{-1} P_A^{\mathrm{T}} X_{ci}^{\mathrm{T}}$$

其中，P_A 是包含第一个 A 特征向量的矩阵，L 是包含第一个 A 特征值的对角线矩阵。

T^2 图的中位数和上控制限计算如下：

$$CL_{T^2, q} = \frac{(n-1)^2}{n} \beta\left[q, \frac{A}{2}, \frac{n-A-1}{2}\right]$$

其中

$$\beta\left[q, \frac{A}{2}, \frac{n-A-1}{2}\right] = \text{Beta}\left(\frac{A}{2}, \frac{n-A-1}{2}\right) \text{ 分布的第 } q \text{ 分位数}$$

要计算中位数，请使用 $q = 0.5$。要计算上控制限，请使用 $q = (1 - \alpha)$。

贡献统计量

第 i 个观测的 T^2 贡献统计量的向量计算如下：

$$con_i = X_{ci} P_A L^{-1/2} P_A^{\mathrm{T}}$$

注意：单值的平方贡献的总和等于该单值的 T_i^2。

第 5 章

判别分析

基于连续变量预测分类

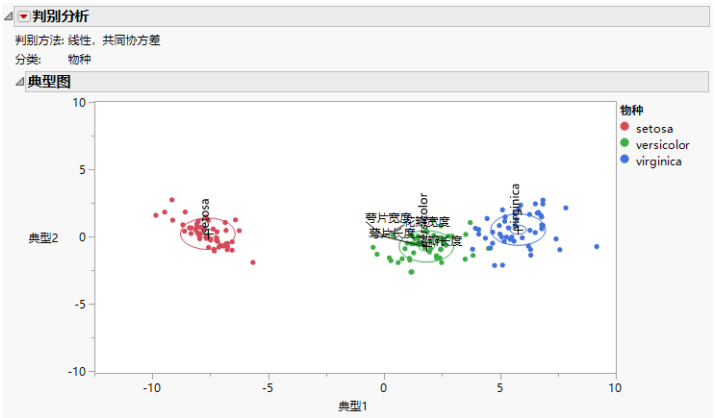
判别分析基于几个连续变量的观测值预测一个组或一个类别的成员关系。具体来说，判别分析基于已知的连续响应 (Y) 来预测一个分类 (X) 变量（分类）。判别分析的数据包含具有已知组成员关系的观测样本以及它们对应连续变量的值。

例如，您可能尝试基于期望概率将贷款申请人分为三个信贷类别 (X)：低利率贷款、长期贷款或无贷款。您可能使用一些连续变量（如当前工资、当前工作年数、年龄和债务负担）(Y) 来预测个人的最大利润贷款类别。您可以使用判别分析生成一个预测模型，从而将个人归为某个贷款类别。

“判别”平台的功能包括：

- 一个逐步选择选项来帮助选择能很好进行判别的变量。
- 拟合方法选项：“线性”、“二次”、“正则”和“宽线性”。
- 一个典型图和误分类汇总。
- 判别得分以及到各组的平方距离。
- 用于将预测距离和概率保存到数据表的选项。

图 5.1 典型图



目录

“判别分析”平台概述 71
判别分析的示例 71
启动“判别分析”平台 72
逐步选择变量 74
判别方法 77
收缩协方差 80
“判别分析”报表 80
主成分 81
典型图和典型结构 81
判别得分 84
得分汇总 85
“判别分析”选项 87
得分选项 89
典型选项 90
“典型三维图”的示例 93
指定先验值 94
考虑新水平 94
保存判别矩阵 95
散点图矩阵 95
JMP 和 JMP Pro 中的验证 96
“判别分析”平台的统计详细信息 97
宽线性算法的说明 97
保存的公式 97
多元检验 103
近似 F 检验 104
组间协方差矩阵 105

“判别分析”平台概述

判别分析尝试将由连续变量值描述的一系列观测值进行分组。由分类变量 *X* 定义的组成员关系由连续变量来预测。这些变量称为**协变量**并用 *Y* 表示。

判别分析不同于 Logistic 回归。在 Logistic 回归中，分类变量是随机的并由连续变量来预测。在判别分析中，分类是固定的而协变量 (*Y*) 是通过随机变量实现的。但是，在这两种方法中，分类值均由连续变量来预测。

“判别”平台提供拟合模型的四种方法。所有方法均使用 Mahalanobis 距离估计每个观测到每个组的多元均值（**重心**）的距离。您可以指定组成员关系的先验概率，在距离计算中要采用它们。观测会被归类到距离最近的组中。

拟合方法包括以下几种：

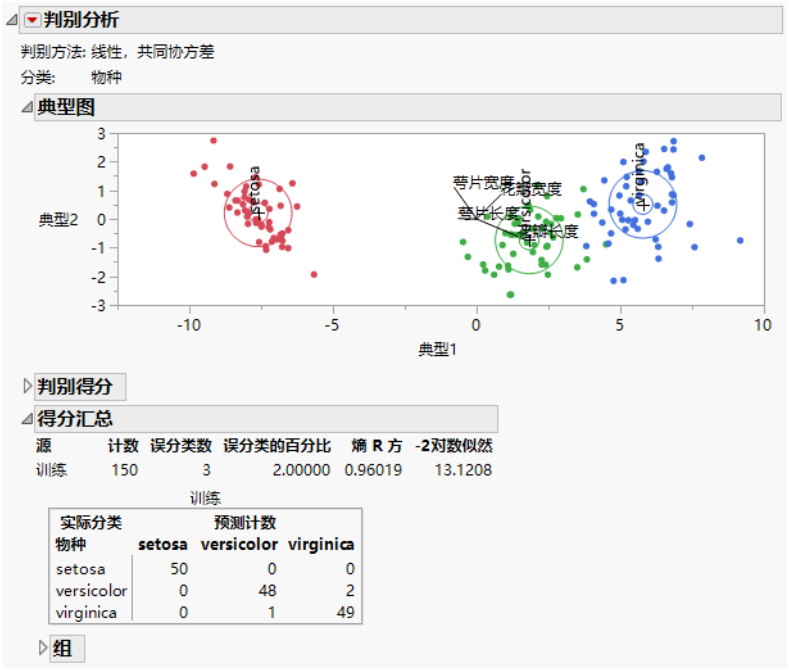
- 线性 — 假定组内协方差矩阵是相等的。假定由 *X* 定义的组的协变量均值不同。
- 二次 — 假定组内协方差矩阵不同。这要求估计比线性方法更多的参数。若组样本大小很小，则可能得到不稳定的估计值。
- 正则 — 在组内协方差矩阵不同时提供两种方法来增加估计值的稳定性。若组样本大小很小，这是很有用的选项。
- 宽线性 — 在拟合有很多协变量的模型时很有用，此时其他方法可能计算困难。它假定所有协方差矩阵是相等的。

判别分析的示例

在 Fisher 的鸢尾花数据集中，对包含三个不同物种的鸢尾花的样本取了四个测量值。目标是使用四个测量值准确识别物种。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**分析** > **多元方法** > **判别**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，协变量**。
4. 选择**物种**并点击 **X，类别**。
5. 点击**确定**。

图 5.2 “判别分析”报表窗口

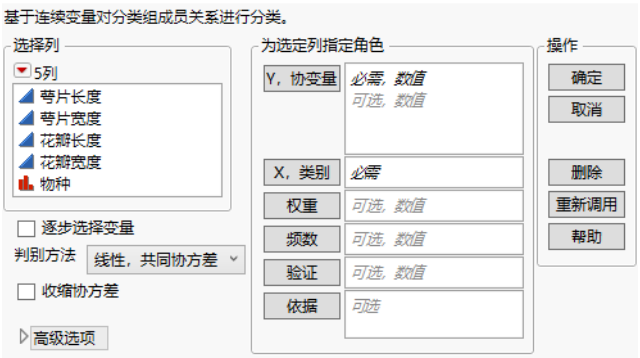


因为**物种**有三个分类，因此有两个典型变量。在“典型图”中，针对两个典型坐标绘制每个观测。该图显示这两个坐标分隔了三个物种。因为没有验证集，“得分汇总”报表只显示“训练集”的面板。没有验证集时，将整个数据集视为训练集。在 150 个观测中，只有三个被误分类。

启动“判别分析”平台

通过选择**分析** > **多元方法** > **判别**来启动“判别”平台。

图 5.3 Iris.jmp 的“判别”启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

注意： “验证” 按钮仅显示在 JMP Pro 中。在 JMP 中，您可以使用排除行来定义验证集。请参见第 96 页的 “JMP 和 JMP Pro 中的验证”。

Y，协变量 包含用于将观测分为各个类别的连续变量的列。

X，类别 包含将观测归入其中的类别或组的列。

权重 用于将一个权重值分配给要分析的每个行的列。

频数 用于将一个频数值分配给要分析的每个行的列。一般来说，频数列的作用是扩展数据表，以便具有整数频数 *k* 的任何行扩展为 *k* 行。您可以指定小数频数。

JMP PRO **验证** 用于定义验证集的数值列。该列应包含至多三个非重复值。

- 若有两个值，较小的值定义训练集，较大的值定义验证集。
- 若有三个值，这些值按大小递增的顺序定义训练集、验证集和测试集。
- 若验证列有三个以上的水平，则包含最小三个值的行将定义验证集。其他所有行都从分析中排除。

“判别” 平台使用验证列来训练和评估模型，除非使用 “逐步选择变量”。若在启动时选定 “逐步选择变量” 选项，则 “判别” 平台使用验证列来训练和微调模型，或者训练、微调和评估模型。有关验证的详细信息，请参见 《预测和专业建模》。

提示： 若未使用 “逐步选择变量”，则验证列应只包含两个非重复值。

若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。有关 “生成验证列” 实用工具的详细信息，请参见 《预测和专业建模》。

依据 为指定列的每个水平执行单独的分析。

逐步选择变量 使用协方差分析和 *p* 值逐步选择变量。请参见第 74 页的 “逐步选择变量”。

若指定了验证集，则还会显示验证集的统计量。验证集统计量用于确定使用 “执行” 按钮时要执行的步骤数。

注意： 不为 “宽线性” 判别方法提供该选项。

判别方法 提供四种执行判别分析的方法。请参见第 77 页的 “判别方法”。

收缩协方差 收缩合并的组内协方差矩阵和组内协方差矩阵的非对角线元素。这样可以提高预测的稳定性并且减小预测的方差。请参见第 80 页的 “收缩协方差”。

高级选项 包含以下选项：

未中心化典型 禁止典型得分中心化以便与较旧版本的 JMP 兼容。

使用伪逆元素 当协方差矩阵是奇异矩阵时，在分析中使用 Moore-Penrose 伪逆元素。所得的分数涉及所有协变量。若未选中它，则分析会删除以下协变量：这些协变量是在 Y，**协变量**列表中位于它们之前的协变量的线性组合。

按排除行交叉验证 指定排除行将形成计算其拟合统计量的验证集。

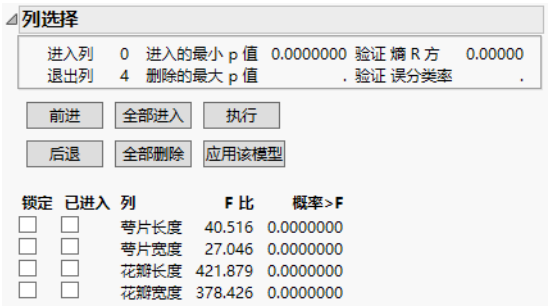
逐步选择变量

注意：“逐步选择变量”不适用于“宽线性”方法。

若您在启动窗口中选择了“逐步选择变量”选项，打开“判别分析”报表时会显示“列选择”面板。您可以执行逐步分析，使用按钮选择变量或使用“锁定”和“已进入”复选框手动选择它们。根据您的选择会更新 *F* 比和 *p* 值。有关如何更新这些值的详细信息，请参见第 74 页的“更新“F 比”和“概率>F””。

若指定任何类型的验证集，则会显示“执行”按钮。当您点击“执行”时，JMP 使用验证集统计量来确定要执行的步骤数。

图 5.4 Iris.jmp 具有验证集的 “列选择”面板



更新“F 比”和“概率 >F”

当您在模型中输入变量或删除变量时，会根据对具有以下结构的协方差模型的分析更新“F 比”和 “概率 >F”值：

- 考虑之中的协变量为响应变量。
- 已进入模型的协变量为预测变量。
- 组变量为预测变量。

“逐步”报表中给出的“F 比”和“概率>F”值是组变量的协方差分析检验的 *F* 比和 *p* 值。组变量的协方差分析检验是它相对于考虑之中的协变量的判别能力的指示符。

统计量

进入列 当前已选择进入判别模型的列数。

退出列 当前可以进入判别模型的列数。

进入的最小 p 值 所有可以进入模型的协变量的 *p* 值中最小的那个 *p* 值。

删除的最大 p 值 在当前已选择进入模型的所有协变量的 *p* 值中最大的那个 *p* 值。

验证熵 R 方 验证集的熵 R 方。值越大表示拟合效果越好。“熵 R 方”的值为 1 表示分类预测完美。由于判别模型的预测概率的不确定性很典型，因此 “熵 R 方”值往往很小。

请参见第 86 页的 “熵 R 方”。仅当使用验证集时才可用。

注意：“验证熵 R 方”可能为负数。

验证误分类率 验证集的误分类率。值越小表示分类效果越好。仅当使用验证集时才可用。

按钮

前进 使尚未进入的协变量中最显著的协变量进入。若使用了验证集，则 “概率 >F”值基于训练集。

后退 从已进入但是未锁定的协变量中删除最不显著的协变量。若使用了验证集，则“概率 >F”值基于训练集。

全部进入 通过选中在 “已进入”列中未锁定的所有协变量来使所有协变量进入。

全部删除 通过在 “已进入”列中取消选择未锁定的所有协变量来删除它们。

应用该模型 基于在 “已进入”列中选中的协变量生成判别分析报表。关闭 “选择列”分级显示项，并更新 “判别分析”窗口来基于选定的判别方法显示分析结果。

提示：在点击**应用该模型**后，您选择的列将显示在 “得分汇总”报表的顶部。

执行 在前进步中使协变量进入，直到 “验证熵 R 方”开始减小。当执行两个前进步而未改进 “验证熵 R 方”时终止进入。仅当在 JMP 中具有排除行或在 JMP Pro 中具有验证列时可用。

列

锁定 强制协变量保持当前状态而不管使用按钮执行的任何步进操作。

请注意以下几点：

- 若您使一个协变量进入然后为它选择**锁定**，则它仍留在模型中而不管使用控件按钮所做的选择。已锁定的协变量的**已进入**框显示变暗的选中标记，以指示它在模型中。
- 若您为未进入的协变量选择了**锁定**，则它不进入模型而不管使用控件按钮所做的选择。

已进入 指示哪些列当前在模型中。您可以手动选择进入或退出模型的列。变暗的选中标记指示已进入模型的锁定的协变量。

列 关注的协变量。

F 比 使用协方差分析模型获得的组变量检验的 *F* 比。请参见第 74 页的 “更新 “F 比” 和 “概率 >F” ”。

概率 > F 使用协方差分析模型获得的组变量检验的 *p* 值。请参见第 74 页的 “更新 “F 比” 和 “概率 >F” ”。

逐步示例

为了说明如何使用 “逐步”，我们使用了 Iris.jmp 样本数据表。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**分析** > **多元方法** > **判别**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，协变量**。
4. 选择**物种**并点击 **X，类别**。
5. 选择**逐步选择变量**。
6. 点击**确定**。
7. 点击**前进**三次。

   三个协变量进入模型。“进入的最小 p 值” 显示在顶部面板。它的值为 0.0103288，指示剩余协变量**萼片长度**在**物种**的判别分析模型中也可能是很有用的。

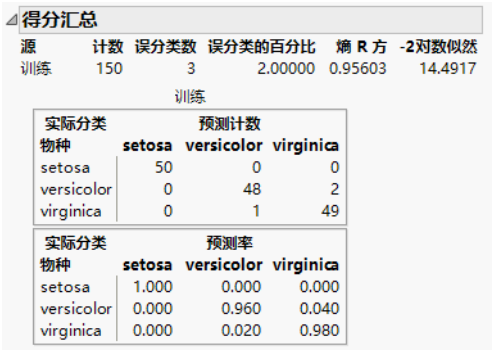
图 5.5 Iris.jmp 的逐步模型

8. 点击**应用该模型**。

   “列选择” 分级显示项关闭。根据进入的协变量和您选择的判别方法，窗口更新以显示拟合的报表。

   请注意，您为模型选择的协变量在 “得分汇总” 报表顶部列出。

图 5.6 显示所选协变量的“得分汇总”报表



得分汇总

源	计数	误分类数	误分类的百分比	熵 R 方	-2对数似然
训练	150	3	2.00000	0.95603	14.4917

训练

实际分类 物种	预测计数 setosa	versicolor	virginica
setosa	50	0	0
versicolor	0	48	2
virginica	0	1	49

实际分类 物种	预测率 setosa	versicolor	virginica
setosa	1.000	0.000	0.000
versicolor	0.000	0.960	0.040
virginica	0.000	0.020	0.980

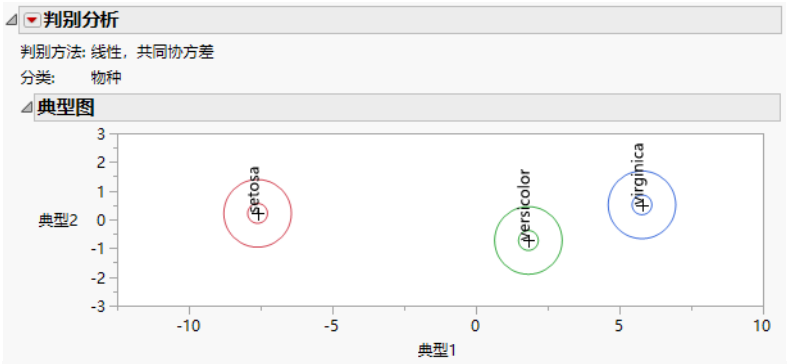
判别方法

JMP 提供以下方法来执行“判别分析”：“线性”、“二次”、“正则”和“宽线性”。前三个方法的基础模型不同。当协变量数很大时，“宽线性”方法是拟合线性模型的有效方式。

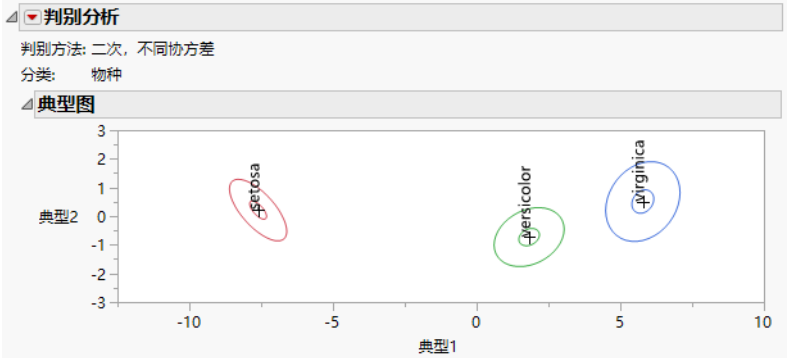
注意：您使 500 个以上的协变量进入时，“JMP 警示”建议您切换到“宽线性”方法。这是因为在列数过多时使用其他方法，计算时间会相当长。点击**宽线性，许多列**可切换到“宽线性”方法。点击**继续**可使用您最初选定的方法。

图 5.7 “线性”、“二次”和“正则”判别分析

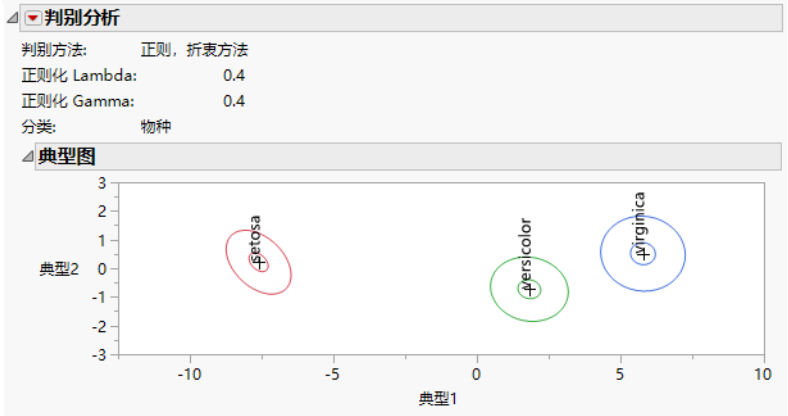
线性



二次



正则 (λ=0.4, γ=0.4)



在图 5.7 中举例说明了“线性”、“二次”和“正则”方法。在此处简要介绍一下这些方法。请参见第 97 页的“保存的公式”。

线性，共同协方差 执行线性判别分析。该方法假定组内协方差矩阵是相等的。请参见第 98 页的“线性判别方法”。

二次，不同协方差 执行二次判别分析。该方法假定组内协方差矩阵不同。该方法需要估计比线性方法更多的参数。若组样本大小很小，则可能得到不稳定的估计值。请参见第 99 页的“二次判别方法”。

若协变量在 X 变量的各水平之间保持不变，则它在组内协方差矩阵中的相关元素的协方差为零。为了使矩阵可以求逆，零协方差被替换为相应的合并组内协方差。完成后，一条注释会显示在报表窗口中，标识有问题的协变量和 X 的水平。

提示：二次方法的缺点在小数据集中显露出来。它很难构造可逆且稳定的协方差矩阵。正则法在允许组间差异的基础上改善了以上问题。

正则，折衷方法 当组内协方差矩阵不同时提供使估计值稳定的两种方法。当组样本大小很小时该选项很有用。请参见第 79 页的“正则，折衷方法”和第 100 页的“正则判别方法”。

宽线性，许多列 基于很多协变量拟合模型时很有用，此时使用其他方法计算会很困难。该方法假定所有组内协方差矩阵是相等的。该方法使用奇异值分解方法来计算合并的组内协方差矩阵的逆矩阵。请参见第 97 页的“宽线性算法的说明”。

注意：使用“宽线性”选项时，通常为其他判别方法显示的几个功能不可用。这是因为该算法不显式计算很大的合并组内协方差矩阵。

正则，折衷方法

正则判别分析由两个非负参数确定。

- 第一个参数（Lambda，收缩到共同协方差）指定如何混合单个协方差矩阵和组协方差矩阵。对于该参数，1 对应于线性判别分析，0 对应于二次判别分析。
- 第二个参数（Gamma，收缩到对角线）是一个乘数，指定对非对角元素（各变量上的协方差）应用多少缩小量。若您选择 1，则强制协方差矩阵为对角矩阵。

为这两个参数都赋值 0 与请求执行二次判别分析的效用相同。类似地，为 Lambda 赋值 1 并为 Gamma 赋值 0 表示请求线性判别分析。使用表 5.1 帮助您决定正则化。有关线性、二次和正则判别分析的示例，请参见图 5.7。

表 5.1 正则判别分析

使用较小的 Lambda	使用较大的 Lambda	使用较小的 Gamma	使用较大的 Gamma
协方差矩阵不同	协方差矩阵相同	变量相关	变量不相关
很多行	很少行		
很少变量	很多变量		

收缩协方差

在“判别”启动窗口中，您可以选择“收缩协方差”选项。当一些组具有少量的观测时建议使用该选项。判别分析需要对协方差矩阵求逆。收缩非对角元素可以提高稳定性并减小预测方差。“收缩协方差”选项通过一个因子对非对角线元素进行收缩，该因子使用 Schafer and Strimmer (2005) 中所述的方法来确定。

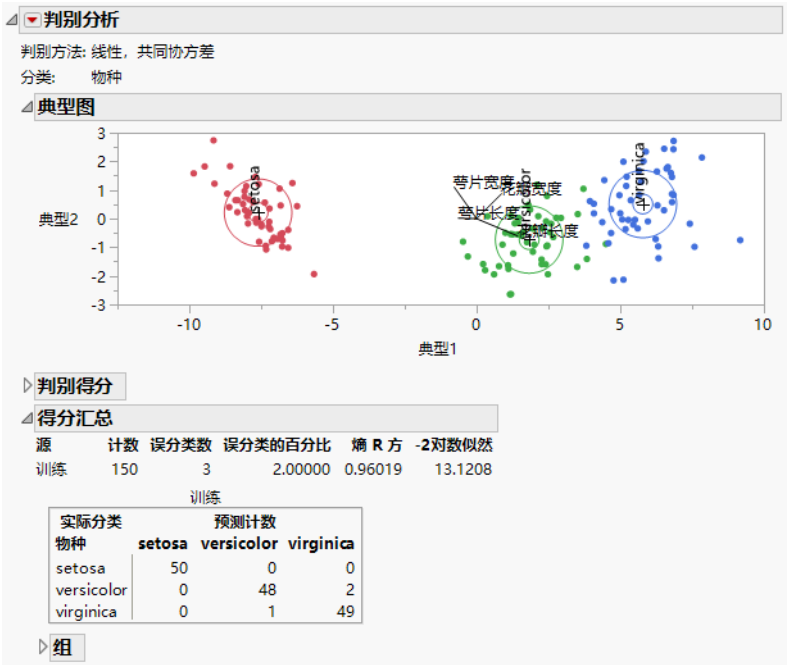
若您在启动窗口中对“线性”判别方法使用“收缩协方差”选项，这样会提供协方差矩阵收缩，它等效于具有适当 Lambda 和 Gamma 值的“正则”判别方法提供的收缩。当您选择“收缩协方差”选项并运行分析时，“收缩”报表将给出“总体收缩”值和“总 Lambda”值。要使用“正则”方法获得相同的分析结果，请在“正则化参数”窗口中，输入 1 作为 Lambda 并使用“收缩”报表中的“总 Lambda”作为 Gamma。

“判别分析”报表

“判别分析”报表基于您选择的判别方法提供判别结果。“判别方法”和“分类”变量显示在报表顶部。若您选择了“正则”方法，还显示相关参数。

您可以通过从“判别分析”红色小三角菜单中选择选项来更改“判别方法”。将更新报表中的结果以反映选择的方法。

图 5.8 “判别分析”报表的示例



默认“判别分析”报表包含以下部分：

- 当您选择“宽线性”判别方法时，将显示“主成分”报表。请参见第 81 页的“主成分”。
- “典型图”显示分组效果最佳的两个维中的点和多元均值。请参见第 81 页的“典型图和典型结构”。
- “判别得分”报表提供有关如何对每个观测进行分类的详细信息。请参见第 84 页的“判别得分”。
- “得分汇总”报表提供观测分类效果的概况。请参见第 84 页的“判别得分”。

主成分

该报表仅对“宽线性”方法显示。考虑以下符号：

- 用 **Y** 表示 n x p 的协变量矩阵，其中 n 是观测数，p 是协变量数。
- 对于 **Y** 中的每个观测，减去协变量均值，并将差值除以协变量的合并标准差。用 $\mathbf{Y_s}$ 表示所得的矩阵。

该报表给出以下值：

数目 提取的特征值数。提取特征值，直到“累积百分比”至少为 99.99%，这表示解释了 99.99% 的变异。

特征值 按降序排列的 $\mathbf{Y_s}$ 的协方差矩阵的特征值，即 $(\mathbf{Y_s}'\mathbf{Y_s})/(n - p)$。

累积百分比 特征值累积和占所有特征值之和的百分比。这些特征值的和等于 $\mathbf{Y_s}'\mathbf{Y_s}$ 的秩。

奇异值 按降序排列的 $\mathbf{Y_s}$ 的奇异值。

典型图和典型结构

“典型图”是一个双标图，它描述变量的典型相关性结构。

典型结构

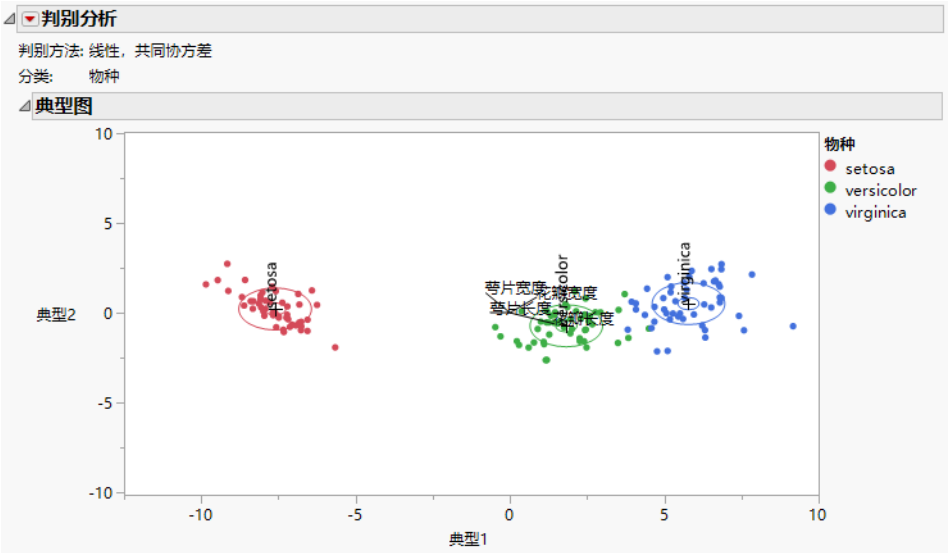
“X，类别”列的每个水平定义一个指示符变量。典型相关性在表示类别的指示符变量组和协变量之间执行。会推导得出协变量的线性组合（称为**典型变量**）。这些典型变量尝试汇总类别间的变异。

第一个典型变量是协变量的线性组合，它最大化类别指示符变量和协变量之间的多重相关性。第二个典型变量是与第一个典型变量不相关的协变量的线性组合，它最大化协变量与类别的多重相关性。若“X，类别”列具有 k 个水平，则获取 $k - 1$ 个典型变量。

典型图

图 5.9 显示了数据表 Iris.jmp 的线性判别分析的典型图。这些点已按**物种**着色。

图 5.9 Iris.jmp 的典型图



双标图轴是前两个典型变量。这些变量定义两个维度来提供组之间的最大分隔。每个典型变量是协变量的线性组合。（请参见第 81 页的“典型结构”。）双标图显示如何用典型变量表示每个观测以及每个协变量对典型变量的贡献大小。

- 观测值和每个组的多元均值表示为双标图上的点。它们用前两个典型变量表示。
  - 对应于每个多元均值的点用加号 (“+”) 标记表示。
  - 为每个均值标绘 95% 置信水平椭圆。若两个组显著不同，则置信椭圆倾向于不相交。
  - 为每个组绘制表示 50% 等高线的椭圆。这在前两个典型变量的空间中绘制一个区域来包含大约 50% 的观测（假定正态性）。
- 图中显示的一组射线表示协变量。
  - 对于每个典型变量，线性组合中协变量的系数可以解释为**权重**。
  - 为了帮助在权重之间进行比较，协变量进行了标准化，使得每个协变量的均值为 0 且标准差为 1。标准化协变量的系数称为**典型权重**。协变量的典型权重越大，它与典型变量的关联越强。
  - 双标图中每条射线的长度和方向指示相应的协变量与前两个典型变量的关联度。射线长度是典型权重的倍数。
  - 这些射线从点 (0,0) 发出，该点表示用典型变量表示的数据的总均值。
  - 通过从“判别分析”红色小三角菜单中选择**典型选项 > 显示典型详细信息**可获取权重系数值。在“典型详细信息”报表底部，点击“标准化得分系数”。请参见第 92 页的“标准化得分系数”。

修改典型图

有更多选项供您修改双标图：

- 通过从“判别分析”红色小三角菜单中选择**典型选项 > 显示均值置信限椭圆**来显示或隐藏 95% 置信椭圆。
- 通过从“判别分析”红色小三角菜单中选择**典型选项 > 显示双标图射线**来显示或隐藏射线。
- 将双标图射线中心拖到图中其他位置。通过从“判别分析”红色小三角菜单中选择**典型选项 > 双标图射线位置**来指定其位置和尺度。除非需要调整以使射线可见，“典型图”中显示的默认“射线尺度”为 1.5。
- 通过从“判别分析”红色小三角菜单中选择**典型选项 > 显示正态 50% 等高线**来显示或隐藏 50% 等高线。
- 通过从“判别分析”红色小三角菜单中选择**典型选项 > 点着色**来对点进行颜色编码以匹配椭圆。

分为三个或更多类别

对于 Iris.jmp 数据，有三个**物种**，因此只有两个典型变量。图 5.9 中的图显示了使用这两个典型变量很好地分隔开三个组。

图中的射线指示以下信息：

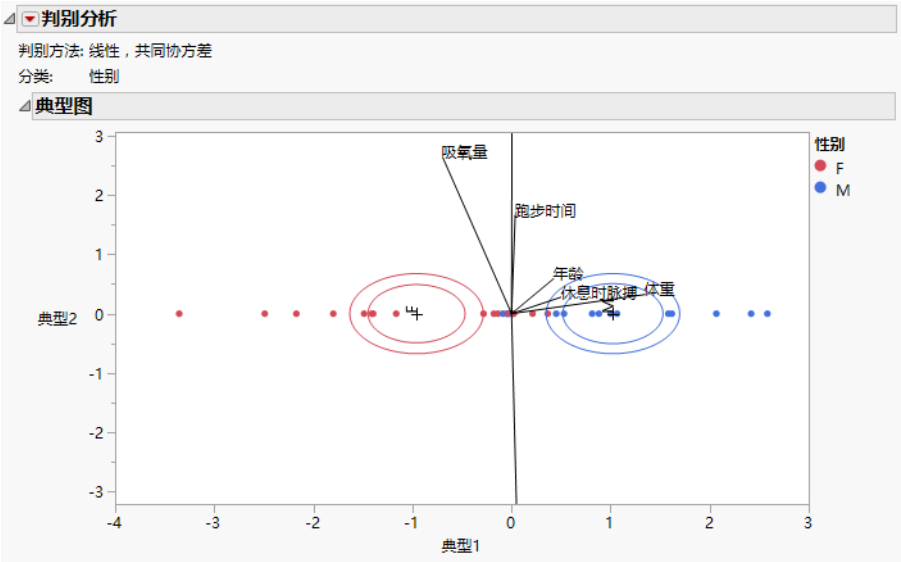
- **花瓣长度**与“典型 1”正相关，与“典型 2”负相关。与“典型 2”相比，定义“典型 1”时它具有更大的权重。
- **花瓣宽度**与“典型 1”和“典型 2”均正相关。在定义两个典型变量时，它具有大概相同的权重。
- **萼片宽度**与“典型 1”负相关，与“典型 2”正相关。与“典型 1”相比，定义“典型 2”时它具有更大的权重。
- **萼片长度**在定义“典型1”时负加权，并且与定义“典型2”具有很弱的关联。

分为两个类别

分类变量只有两个水平时，针对单个典型变量（在图中用“典型1”表示）绘制点。每个协变量的典型权重仅与“典型1”有关。射线只显示垂直成分来分隔它们。将这些射线投影到“典型1”轴以比较它们与单个典型变量的相对关联度。

图 5.10 显示了样本数据表 Fitness.jmp 的“典型图”。使用七个连续变量将个体分类为 M（男性）或 F（女性）。因为分类变量只有两个类别，因此只有一个典型变量。

图 5.10 Fitness.jmp 的 “典型图”



“典型图”中的点已经按**性别**着色。请注意，这两个组用“典型1”的值很好地分隔开。

尽管对应于七个协变量的射线有垂直成分，在这种情况下您必须仅根据它们在 “典型 1”轴上的投影来解释射线。您注意到：

- **最大脉搏**、**跑步时间**和**跑步时脉搏**与“典型1”的关联度很低。
- **体重**、**休息时脉搏**和**年龄**与“典型 1”正相关。**体重**的关联度最高。协变量**休息时脉搏**和**年龄**具有类似但是更小的关联度。
- **吸氧量**与“典型1”负相关。

判别得分

“判别得分”报表提供每个观测的预测分类和支持信息。

行 数据表中的观测行。

实际分类 数据表中给出的观测分类。

SqDist(实际分类) 数据表中给出的观测分类的已保存公式 SqDist[< 水平 >] 的值。请参见第 89 页的 “得分选项”。

> **注意：**由于公式中有偏移项， SqDist(实际分类) 可能为负。

Prob(实际分类) 观测的实际分类的估计概率。

-Log(Prob) Prob(实际分类) 的对数的负数。这个负对数似然值很大则表示使用实际类别的成员关系预测观测的效果不好。

-Log(Prob) 的图显示在 -Log(Prob) 值右侧。大的直条指示预测效果不好。星号 (*) 指示误分类的观测。

若您使用验证集或测试集，则验证集中的观测使用“v”标记，测试集中的观测使用“t”标记。

预测分类 观测的预测分类。预测分类是具有成员关系的最高预测概率的类别。

Prob(预测分类) 观测的预测分类的估计概率。

其他 列出其他预测概率超过 0.1 的类别 （若存在）。

图 5.11 显示了使用“线性”判别方法得到的 Iris.jmp 样本数据表的“判别得分”报表。选择了**得分选项 > 仅显示需要的行**选项，因而只显示误分类的行或预测概率介于 0.05 到 0.95 之间的行。

图 5.11 仅显示需要的行

判别得分

行	实际分类	SqDist(实际分类)	Prob(实际分类)	-Log(Prob)		预测值	Prob(预测分类)	其他
71	versicolor	8.66970	0.2532	1.373	*	virginica	0.7468	
73	versicolor	4.87619	0.8155	0.204		versicolor	0.8155	virginica 0.18
78	versicolor	4.66698	0.6892	0.372		versicolor	0.6892	virginica 0.31
84	versicolor	8.43926	0.1434	1.942	*	virginica	0.8566	
120	virginica	8.19641	0.7792	0.249		virginica	0.7792	versicolor 0.22
124	virginica	3.57858	0.9029	0.102		virginica	0.9029	
127	virginica	3.90184	0.8116	0.209		virginica	0.8116	versicolor 0.19
128	virginica	3.31470	0.8658	0.144		virginica	0.8658	versicolor 0.13
130	virginica	9.08495	0.8963	0.109		virginica	0.8963	versicolor 0.10
134	virginica	7.23593	0.2706	1.307	*	versicolor	0.7294	
135	virginica	15.83301	0.9340	0.068		virginica	0.9340	
139	virginica	4.09385	0.8075	0.214		virginica	0.8075	versicolor 0.19

“*” 表示误分类

得分汇总

“得分汇总”报表提供判别得分的概况。图 5.12 中的表显示实际分类和预测分类。若所有观测均正确分类，则非对角计数为零。

图 5.12 Iris.jmp 的 “得分汇总”

得分汇总

源	计数	误分类数	误分类的百分比	熵 R 方	-2对数似然
训练	65	1	1.53846	0.94487	7.81888
验证	49	3	6.12245	0.86644	
测试	36	0	0.00000	0.98410	

训练

实际分类	预测计数		
物种	setosa	versicolor	virginica
setosa	18	0	0
versicolor	0	22	1
virginica	0	0	24

实际分类	预测率		
物种	setosa	versicolor	virginica
setosa	1.000	0.000	0.000
versicolor	0.000	0.957	0.043
virginica	0.000	0.000	1.000

验证

实际分类	预测计数		
物种	setosa	versicolor	virginica
setosa	16	0	0
versicolor	0	15	2
virginica	0	1	15

实际分类	预测率		
物种	setosa	versicolor	virginica
setosa	1.000	0.000	0.000
versicolor	0.000	0.882	0.118
virginica	0.000	0.063	0.938

测试

实际分类	预测计数		
物种	setosa	versicolor	virginica
setosa	16	0	0
versicolor	0	10	0
virginica	0	0	10

实际分类	预测率		
物种	setosa	versicolor	virginica
setosa	1.000	0.000	0.000
versicolor	0.000	1.000	0.000
virginica	0.000	0.000	1.000

“得分汇总”报表提供以下信息：

列 若您使用了 “逐步选择变量”构造模型，则列出进入模型的列 （图 5.6）。

源 若未使用验证，则所有观测构成训练集。若使用了验证，则分别显示一行表示训练集和验证集，或分别显示一行表示训练集、验证集和测试集。

误分类数 提供指定的集中被错误分类的观测数。

误分类的百分比 提供指定的集中被错误分类的观测百分比。

熵 R 方 一个拟合测度。值越大表示拟合效果越好。“熵 R 方”的值为 1 表示分类预测完美。由于判别模型的预测概率的不确定性很典型，因此 “熵 R 方”值往往很小。

请参见第 86 页的 “熵 R 方”。

注意：“熵 R 方”可能为负值。

-2 对数似然 训练集中观测的负对数似然的两倍 （基于模型）。值越小表示拟合效果越好。仅对训练集提供。请参见 《拟合线性模型》。

混杂矩阵 显示分类变量 X 的每个水平的 “预测计数 - 实际分类”矩阵。若使用的是带验证功能的 JMP Pro，将为每组观测提供一个矩阵。若您在 JMP 中使用了排除行，则将排除行视为验证集且给出单独的验证矩阵。请参见第 96 页的 “JMP 和 JMP Pro 中的验证”。

熵 R 方

“熵 R 方”是一个拟合测度。为训练集计算该值，若使用了验证，则为验证集和测试集计算该值。

训练集的“熵 R 方”

对于训练集，按以下方式计算 “熵 R 方”：

- 使用训练集拟合判别模型。
- 获取基于模型的预测概率。
- 使用这些预测概率，为训练集中的观测计算似然值。称之为似然_完全$_{训练}$。
- 使用训练集拟合简化模型（无预测变量）。
- 使用简化模型中 X 变量的各水平的预测概率计算训练集中观测的似然值。将该量称为似然_简化$_{训练}$。
- 训练集的“熵 R 方”为：

$$熵\ R\ 方_{训练} = 1 - \frac{\log(似然_完全_{训练})}{\log(似然_简化_{训练})}$$

验证集和测试集的“熵 R 方”

对于验证集，按以下方式计算 “熵 R 方”：

- 仅使用训练集拟合判别模型。
- 为所有观测获取基于训练集模型的预测概率。
- 使用这些预测概率，为验证集中的观测计算似然值。称之为似然_完全$_{验证}$。
- 仅使用训练集拟合简化模型（无预测变量）。
- 使用简化模型中 X 变量的各水平的预测概率计算验证集中观测的似然值。将该量称为似然_简化$_{验证}$。
- “验证熵 R 方”为：

$$验证熵\ R\ 方 = 1 - \frac{\log(似然_完全_{训练})}{\log(似然_简化_{训练})}$$

测试集的“熵 R 方”计算方式与验证集的“熵 R 方”计算方式相似。

“判别分析”选项

“判别分析”红色小三角菜单包含以下选项：

逐步选择变量 （对于 “宽线性”方法不可用。）显示或隐藏 “列选择”控制面板。该控制面板包含的选项支持您使用协方差分析和 *p* 值逐步选择变量。请参见第 74 页的“逐步选择变量”。

判别方法 指定判别方法。选择 “线性”、“二次”、“正则”和 “宽线性”。请参见第 77 页的“判别方法”。

判别得分 显示或隐藏每行的判别得分表。

得分选项 提供用于对观测评分的选项。特别是您可以保存得分公式。请参见第 89 页的“得分选项”。

典型图 显示或隐藏典型图。请参见第 81 页的“典型图和典型结构”。

典型选项 提供影响“典型图”的选项。请参见第 90 页的“典型选项”。

典型三维图 显示三维典型图。仅当分类变量 X 有四个或更多水平时该选项才可用。请参见第 93 页的“‘典型三维图’的示例”。

指定先验值 允许您指定 X 变量每个水平的先验概率。请参见第 94 页的“指定先验值”。

考虑新水平 指定某些点可能不适合任何已知组，并且应考虑来自未计算分数的新组。请参见第 94 页的“考虑新水平”。

显示组内协方差 显示或隐藏以下报表：

- 给出合并的组内协方差和相关性矩阵的“协方差矩阵”报表。
- 对于“二次”和“正则”方法，显示组内相关性矩阵的“每组的相关性”报表。

  对于每个组，还显示组内协方差矩阵的行列式的对数。
- 对于“二次”判别方法，将“组协方差”分级显示项添加到显示组内协方差矩阵的“协方差矩阵”报表。

“显示组内协方差”对于“宽线性”判别方法不可用。

显示组均值 显示或隐藏提供每个协变量均值的“组均值”报表。显示 X 变量的每个水平的均值和总均值。

保存判别矩阵 将名为**判别结果**的脚本保存到数据表。该脚本是在 JSL 中使用的以下对象的列表：

- 协变量 (Y) 的列表
- 分类变量 X
- X 的水平的列表
- X 不同水平下的协变量均值矩阵
- 合并的组内协方差矩阵

“保存判别矩阵”对于“宽线性”判别方法不可用。请参见第 95 页的“保存判别矩阵”。

散点图矩阵 打开一个“散点图矩阵”报表，其中显示包含每对协变量的散点图的矩阵。该选项会调用“散点图矩阵”平台，其中包含每个组的着色密度椭圆。散点图包含数据表中的所有观测，即使使用了验证。请参见第 95 页的“散点图矩阵”。

对于“宽线性”判别方法不可用。

刻画器 显示或隐藏交互式刻画器报表，其中的分类概率合并为单个刻画器行。因子值的更改体现在估计的分类概率中。有关红色小三角菜单中选项的详细信息，请参见《刻画器指南》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

得分选项

“得分选项”提供以下处理得分的选项：

仅显示需要的行 在“判别得分”报表中，只显示误分类的行和预测概率介于 0.05 到 0.95 之间的行。

显示分类计数 显示或隐藏“得分汇总”报表中的混淆矩阵和混淆率矩阵。混淆矩阵是实际响应和预测响应的双向分类。混淆率矩阵等同于混淆矩阵，只不过其中的数字要除以行合计。默认情况下，“得分汇总”报表显示分类变量 X 的每个水平的混淆矩阵和混淆率矩阵。若使用的是带验证功能的 JMP Pro，将为每组观测提供一个矩阵。若您在 JMP 中使用了排除行，则将排除行视为验证集且给出单独的验证矩阵。请参见第 96 页的 “JMP 和 JMP Pro 中的验证”。

显示各组距离 显示或隐藏一个报表，其中包含每个观测到每个组均值的 Mahalanobis 平方距离。

显示各组概率 显示或隐藏一个报表，其中包含某个观测属于分类变量 X 所定义的每个组的概率。

ROC 曲线 在“得分汇总”报表中显示或隐藏“受试者操作特征 (ROC)”曲线。有关 ROC 曲线的详细信息，请参见 《预测和专业建模》。

选择误分类的行 在数据表以及按行显示列表的报表窗口中选择误分类的行。

选择不确定的行 在数据表以及按行显示列表的报表窗口中选择具有不确定分类的行。不确定的行是指该行属于**任意**组的成员的概率既不接近于 0 也不接近于 1。

选择该选项时，将打开一个窗口，您可以在此指定反映不确定性的预测概率的范围。默认情况下，概率与 0 或 1 存在超过 0.1 的差值的任何行定义为不确定的行。因此，默认选择其概率介于 0.1 到 0.9 之间的行。

保存公式 将距离、概率和预测的成员关系公式保存到数据表。请参见第 97 页的 “保存的公式”。

- 距离公式为 SqDist0 和 SqDist[<水平>]，其中 <水平> 表示 X 的水平。距离公式生成与 Mahalanobis 距离计算有关的中间值。
- 概率公式为 Prob[<水平>]，其中 <水平> 表示 X 的水平。每个概率列给出在 X 的该水平上观测所属成员关系的后验概率。将“响应概率”列属性保存到每个概率列。有关“响应概率”列属性的详细信息，请参见《使用 JMP》。
- 预测的成员关系公式为 Pred <X>，它包含“最可能的水平”分类规则。
- “宽线性”方法还保存**判别数据矩阵**列，该列包含协变量的向量和**判别主成分**公式。请参见第 101 页的 “宽线性判别方法”。

注意：对于“宽线性”之外的其他方法，当您保存公式时，“RowEdit Prob”脚本会保存到数据表。该脚本选择数据表中不确定的行。该脚本将其概率与 0 或 1 存在超过 0.1 的差值的行定义为不确定的行。它还打开一个“行编辑器”窗口，您可以在其中查看不确定的行。若您拟合新模型 （“宽线性”除外）并选择 “保存公式”，则现有 “RowEdit Prob”脚本会被替换为适用于新拟合的脚本。

生成评分脚本 （仅在 JMP 标准版中可用。）创建一个脚本，它构造使用 “保存公式”选项保存的公式列。您可以保存并使用该脚本 （可能涉及其他数据表），以创建计算成员关系概率以及预测组成员关系的公式列。

JMP PRO **发布概率公式** （仅限于 JMP Pro。）创建概率公式并在“公式存储库”平台中将它们保存为公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

典型选项

下面列出的第一个选项与“典型图”或“典型三维图”的外观有关。其余选项提供与该图有关的计算的详细信息。

注意：仅当有三个或更多协变量且分组变量有四个或更多类别时“典型三维图”才可用。

与图外观有关的选项

显示点 在 “典型图”和 “典型三维图”中显示或隐藏点。

显示均值置信限椭圆 显示或隐藏 “典型图”和 “典型三维图”中的每个组的均值的 95% 置信椭圆 （假定正态性）。

显示正态 50% 等高线 显示或隐藏每个组的 50% 预测椭圆或椭球体。在 “典型图”中，每个椭圆在前两个典型变量的空间中绘制一个估计每组 50% 的观测会落入的区域 （假定正态性）。在 “典型三维图”中，每个椭球在前三个典型变量的空间中绘制一个估计 50% 的观测会落在的区域 （假定多元正态性）。

显示双标图射线 在 “典型图”和 “典型三维图”中显示或隐藏双标图射线。带标签的射线显示协变量在典型空间中的方向。该方向表示每个协变量与每个典型变量的关联度。

双标图射线位置 允许您在“典型图”和“典型三维图”中指定双标图射线的位置和射线尺度。

- 默认情况下，这些射线从点 (0,0) 发出，该点表示用典型变量表示的数据的总均值。在“典型图”中，您可以拖动射线或使用该选项指定坐标。
- “典型图”中的默认“射线尺度”为 1.5，除非需要调整以使射线可见。“射线尺度”的指定与“标准化得分系数”有关。

点着色 基于 *X* 变量的水平为“典型图”和 “典型三维图”中的点着色。将颜色标记添加到数据表中的行。该选项等效于选择**行 > 按列设定颜色或标记**并选择 *X* 变量。它也等效于右击图形并选择**行图例**，然后按分类列着色。

与计算有关的选项

显示典型详细信息 显示或隐藏“典型详细信息”报表。请参见第 91 页的“显示典型详细信息”。

显示典型结构 显示或隐藏“典型结构”报表。请参见第 92 页的“显示典型结构”。对于“宽线性”判别方法不可用。

保存典型得分 在数据表中创建包含每个观测的典型得分公式的列。第 *k* 个典型得分的列命名为 Canon[<k>]。

提示：在脚本中，将脚本命令 Save to New Data Table 发送到“判别”对象会将以下内容保存到新数据表：典型变量的组均值、标准化得分系数的射线尺度为 1.5 的双标图射线以及典型得分。对于“宽线性”判别方法不可用。

显示典型详细信息

“典型详细信息”报表显示说明协变量和分组变量 *X* 之间的关系的检验。相关矩阵显示在报表底部。

图 5.13 Iris.jmp 的“典型详细信息”

典型详细信息

从总合并组内协方差矩阵计算得到的典型详细信息。

特征值	百分比	累积百分比	典型相关性	似然比	近似的 F 值	分子自由度	分母自由度	概率>F
32.1919292	99.1213	99.1213	0.98482089	0.02343863	199.1453	8	288	<.0001*
0.28539104	0.8787	100.0000	0.47119702	0.77797337	13.7939	3	145	<.0001*

检验	值	近似的 F 值	分子自由度	分母自由度	概率>F
Wilks Lambda	0.0234386	199.1453	8	288	<.0001*
Pillai 迹	1.1918988	53.4665	8	290	<.0001*
Hotelling-Lawley	32.47732	582.1970	8	203.4	<.0001*
Roy 最大根	32.191929	1166.9574	4	145	<.0001*

▷组内矩阵
▷组间矩阵
▷得分系数
▷标准化得分系数

注意：计算报表中的结果所用的矩阵是合并的组内协方差矩阵（称之为组内矩阵）。该矩阵是所有判别方法的“典型详细信息”报表的基础。“典型详细信息”报表中的统计量和检验对于所有判别方法都是相同的。

统计量和检验

“典型详细信息”报表列出特征值并给出零特征值的似然比检验。对于典型相关性为零这个原假设提供四个检验。

特征值 组间矩阵和组内矩阵的逆矩阵之积的特征值。它们按从大到小的顺序列出。特征值的大小反映相关的判别函数所解释的变异量。

百分比 给定的特征值在特征值总和中所占的比例。

累积百分比 累积的比例和。

典型相关性 协变量和分类变量 X 所定义的组之间的典型相关性。假定您定义数值指标变量来表示 X 所定义的组。然后使用协变量作为一组变量并使用表示 X 中的各组的指标变量作为另一组变量来执行典型相关性分析。“典型相关性”值是从该分析得到的典型相关性值。

似然比 一个检验的似然比统计量，该检验确定相应典型相关性和所有更小的相关性的总体值是否为零。对于给定典型相关性和所有更小的典型相关性，该比值等于 (1 - 典型相关性 ^{2}) 值的乘积 。

检验 列出针对在各组上协变量的均值相等这个原假设的四个标准检验：Wilk Lambda、Pillai 迹、Hotelling-Lawley 和 Roy 最大根。请参见第 103 页的“多元检验”和“判别分析”附录中第 104 页的“近似 F 检验”。

近似的 F 值 与相应检验相关的 F 值。对于某些检验，F 值是近似值或是一个上限。请参见“判别分析”附录中第 104 页的“近似 F 检验”。

分子自由度 相应检验的分子自由度。

分母自由度 相应检验的分母自由度。

概率 >F 相应检验的 p 值。

矩阵

与典型结构有关的四个矩阵显示在报表的底部。要查看矩阵，点击其名称旁边的展开图标。要隐藏矩阵，点击该矩阵的名称。

组内矩阵 合并的组内协方差矩阵。

组间矩阵 组间协方差矩阵 $\mathbf{S}_B$。请参见第 105 页的“组间协方差矩阵”。

得分系数 用于根据原始数据计算典型得分的系数。这些是用于选项**典型选项 > 保存典型得分**的系数。有关如何计算这些系数的详细信息，请参见 SAS Institute Inc.(2020b) 中的“CANDISC 过程”一章。

标准化得分系数 用于根据标准化数据计算典型得分的系数。经常称为**典型权重**。有关如何计算这些系数的详细信息，请参见 SAS Institute Inc.(2020b) 中的“CANDISC 过程”一章。

显示典型结构

“典型结构”报表给出三个矩阵，它们提供典型变量和协变量之间的相关性。另一个矩阵显示组变量的各个水平上的均值。要查看矩阵，点击其名称旁边的展开图标。要隐藏矩阵，点击该矩阵的名称。

图 5.14 Iris.jmp 的 “典型结构”（显示组间典型结构）

合计典型结构 典型变量和协变量之间的相关性。通常称为**载荷**。

组间典型结构 典型变量的组均值和协变量的组均值之间的相关性。

合并的组内典型结构 典型变量和协变量之间的偏相关性，已针对组变量调整。

典型变量的组均值 提供每个典型变量在组变量的各个水平上的均值。

“典型三维图”的示例

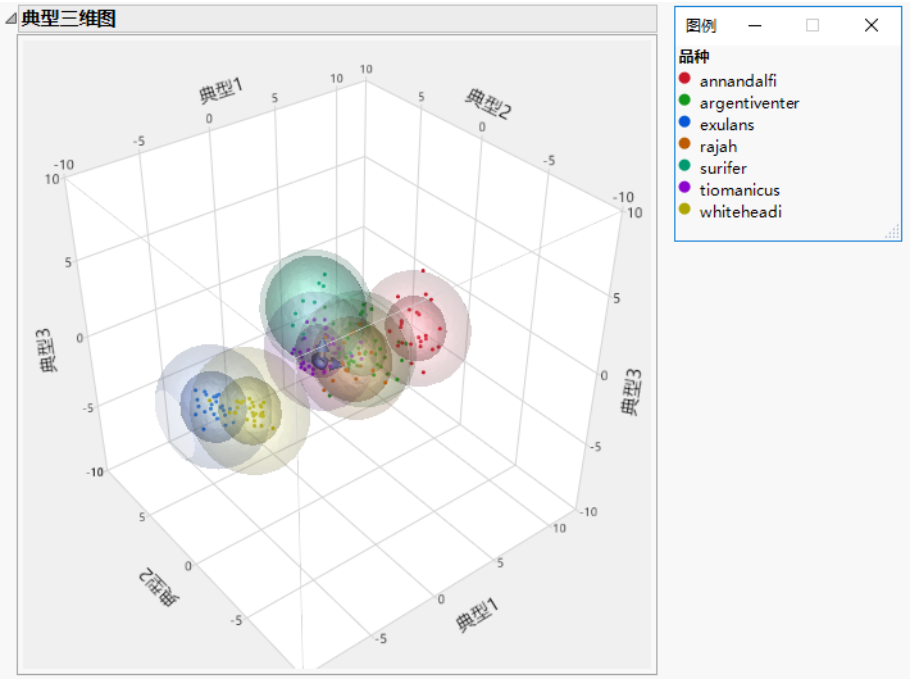
1. 选择**帮助** > **样本数据库**，然后打开 Owl Diet.jmp。
2. 选择第 180-294 行。

   这些是**品种**缺失的行。隐藏和排除这些行。
3. 选择**行** > **隐藏和排除**。
4. 选择**行** > **按列设定颜色或标记**。
5. 选择**品种**。
6. 从 “颜色”菜单中，选择 **JMP 深色**。
7. 选中**生成具有图例的窗口**。
8. 点击**确定**。

   随即显示一个小 “图例”窗口。数据表中的行按**品种**分配了颜色。
9. **选择分析** > **多元方法** > **判别**。
10. 指定**头骨长度**、**牙齿排数**、**上颚孔**和**下颚长度**作为 **Y，协变量**。
11. 指定**品种**作为 **X，类别**。
12. 点击**确定**。
13. 点击 “判别分析”红色小三角并点击**典型三维图**。

提示：点击 “图例”中的类别可在 “典型三维图”中突出显示相应的点。在三维图中点击并拖动可以旋转它。

图 5.15 具有 “图例”窗口的 “典型三维图”



指定先验值

以下选项可用于指定先验值：

相等概率 将相等先验概率分配给所有组。这是默认值。

发生比例 将先验概率分配给与观测数据中的频数成比例的组。

其他 允许您指定定制先验概率。

考虑新水平

若您怀疑一些观测对于分类变量的指定水平来说是离群值，可以使用“考虑新水平”选项。选择该选项时，将显示一个菜单，要求您指定新水平的先验概率。

对于用新组来拟合效果会更好的观测，会分配给称为“其他”的新水平。“其他”组中成员关系的概率假定这些观测服从整个观测集的分布，该分布假定**无组结构**。这导致得到与协方差结构相关的相应宽的正态等高线。距离计算按指定的先验概率进行了调整。

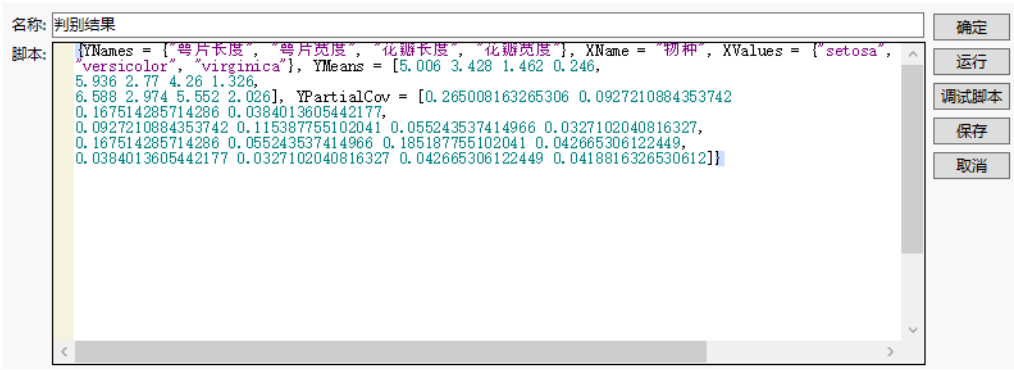
保存判别矩阵

保存判别矩阵会创建一个全局列表 (**判别结果**) 以供在 JMP 脚本语言中使用。该列表包含以下为训练集计算的信息：

- Y 名称，协变量 (Y) 的列表
- X 名称，分类变量
- X 值，X 的水平列表
- Y 均值，协变量均值 X 的水平的矩阵
- Y 偏相关性，组内协方差矩阵

考虑使用 Iris.jmp 样本数据表中的**判别**脚本获得的分析。若您从“判别分析”红色小三角菜单中选择**保存判别矩阵**，则会将脚本**判别结果**保存到数据表。

图 5.16 Iris.jmp 的 “判别结果”表脚本



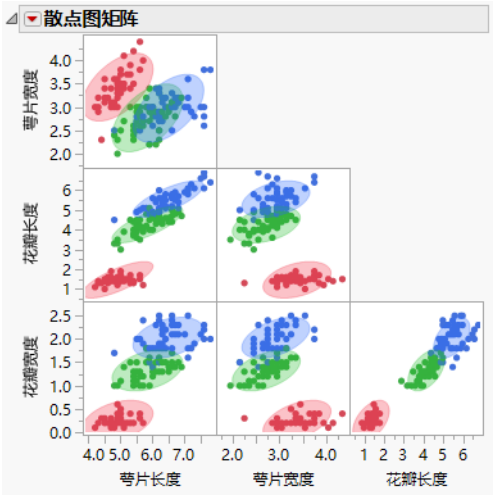
注意：在脚本中，您可以将脚本命令“Get Discrim Matrices”发送到“判别”平台对象。这将获得与“保存判别矩阵”相同的值，但是不在数据表中储存它们。

散点图矩阵

“散点图矩阵”命令在单独窗口中调用“散点图矩阵”平台，结果包含协变量的下三角散点图矩阵。将为数据表中的所有观测绘制点。

对于分类变量 X 的每个水平显示具有 90% 覆盖率的椭圆。对于“线性”判别方法，这些椭圆基于合并的组内协方差矩阵。图 5.17 显示了 Iris.jmp 样本数据表的“散点图矩阵”窗口。

图 5.17 Iris.jmp 的 “散点图矩阵”



《基本绘图》对“散点图矩阵”红色小三角菜单中的选项进行了说明。

JMP 和 JMP Pro 中的验证

在 JMP 中，您可以通过排除构成验证集的行来指定验证集。选择要用作验证集的行，然后选择**行 > 排除/撤销排除**。没有被排除的行会被视为训练集。

注意：在 JMP Pro 中，您可以在“判别”启动窗口中指定验证列。验证列必须具有数值数据类型，应包含至少两个非重复值。

请注意以下事项：

- 若列包含两个值，则较小的值定义训练集，较大的值定义验证集。
- 若列包含三个值，则这些值按大小递增的顺序定义训练集、验证集和测试集。
- 若列包含四个或更多非重复值，则只使用最小的三个值和相关观测按该顺序定义训练集、验证集和测试集。

指定验证集时，“判别”平台执行以下操作：

- 使用训练数据拟合模型。
- “逐步选择变量”选项给出模型的“验证熵 R 方”和“验证误分类率”统计量。请参见第 74 页的“统计量”和第 87 页的“验证集和测试集的“熵 R 方””。
- “判别得分”报表显示一个指示符，它用于标识验证集和测试集中的行。
- “得分汇总”报表显示训练集、验证集和测试集的“预测分类-实际分类”。

“判别分析”平台的统计详细信息

- “宽线性算法的说明”
- “保存的公式”
- “多元检验”
- “近似 F 检验”
- “组间协方差矩阵”

宽线性算法的说明

按以下方式执行 “宽线性”判别分析：

- 通过减去组均值后除以合并的标准差对数据进行标准化。
- 使用奇异值分解从一组奇异向量获得主成分变换矩阵。
- 保留的成分数表示奇异值平方和至少为 0.9999。
- 对变换后的数据执行线性判别分析，其中数据未进行组均值变换。这是快速计算，因为合并的组内协方差矩阵是对角矩阵。

保存的公式

本节给出通过**得分选项** > **保存公式**保存的推导公式。这些公式依赖于判别方法。

对于由分类变量 X 定义的每个组，假定协变量的观测服从 p 维多元正态分布，其中 p 是协变量数。表 5.2 中给出了公式中使用的符号。

表 5.2 “保存公式”选项给出的公式符号

p	协变量数
T	组（X 的水平）总数
$t = 1,..., T$	用于区分由 X 定义的各组的下标
n_t	第 t 组中的观测数
$n = n_1 + n_2 + ... + n_T$	总观测数
$\mathbf{y}$	某个观测的协变量的 1 x p 向量
$\mathbf{y}_{it} = (y_{i1t}, y_{i2t}, \cdots, y_{ipt})$	组 t 中第 i 个观测，它包含 p 个协变量的向量

表 5.2 “保存公式”选项给出的公式符号 (续)

$\bar{\mathbf{y}}_t$	针对组 *t* 中的观测，协变量 **y** 的均值的 *1* x *p* 向量
$\mathbf{y}_{bar}$	所有观测的协变量均值的 *1* x *p* 向量
$\mathbf{S}_t = \frac{1}{n_t - 1}\sum_{i=1}^{n_t}(\mathbf{y}_{it} - \bar{\mathbf{y}}_t)(\mathbf{y}_{it} - \bar{\mathbf{y}}_t)'$	估计的第 *t* 组的组内协方差矩阵 (*p* x *p*)
$\mathbf{S}_p = \frac{1}{n - T}\sum_{t=1}^{T}(n_t - 1)\mathbf{S}_t$	估计的 (*p* x *p*) 合并的组内协方差矩阵
q_t	组 *t* 的成员关系的先验概率
$p(t\|\mathbf{y})$	**y** 属于组 *t* 的后验概率
$\|\mathbf{A}\|$	矩阵 A 的行列式

线性判别方法

在线性判别分析中，假定所有组内协方差矩阵是相等的。共同协方差矩阵通过 $\mathbf{S}_p$ 来估计。请参见表 5.2 了解相关符号。

观测 **y** 到组 *t* 的 Mahalanobis 距离按以下方式定义：

$$d_t^2 = (\mathbf{y} - \bar{\mathbf{y}}_t)'\mathbf{S}_p^{-1}(\mathbf{y} - \bar{\mathbf{y}}_t)$$

观测 **y** 属于第 *t* 组的似然函数按以下方式估计：

$$\begin{aligned} l_t(\mathbf{y}) &= (2\pi)^{-T/2}\left|\mathbf{S}_p\right|^{-1/2}\exp(-(\mathbf{y} - \bar{\mathbf{y}}_t)'\mathbf{S}_p^{-1}(\mathbf{y} - \bar{\mathbf{y}}_t)/2) \\ &= (2\pi)^{-T/2}\left|\mathbf{S}_p\right|^{-1/2}\exp(-d_t^2/2) \end{aligned}$$

请注意，必须为合并的协方差矩阵估计的参数数目是 *p(p+1)/2*，必须为均值估计的参数数目是 *Tp*。必须估计的参数总数是 *p(p+1)/2 + Tp*。

按以下方式定义组 *t* 中的成员关系的后验概率：

$$p(t|\mathbf{y}) = \frac{q_t l_t(\mathbf{y})}{\sum_{u=1}^{T} q_u l_u(\mathbf{y})} = \frac{1}{1 + \sum_{u \neq t}\exp(-[(d_u^2 - 2\log(q_u)) - (d_t^2 - 2\log(q_t))]/2)}$$

观测 **y** 被分配给具有最大后验概率的组。

按以下方式定义 “线性”判别方法所保存的公式：

SqDist[0]	$\mathbf{y}'\mathbf{S}_p^{-1}\mathbf{y}$
SqDist[<组 *t*>]	$d_t^2 - 2\log(q_t)$
Prob[<组 *t*>]	$p(t\|\mathbf{y})$
Pred <*X*>	t，对于它 $p(t\|\mathbf{y})$ 为最大值， $t = 1, ..., T$

二次判别方法

在二次判别分析中，不假定组内协方差矩阵是相等的。组 t 的组内协方差矩阵由 $\mathbf{S}_t$ 估计。这意味着必须为组内协方差矩阵估计的参数数目是 $Tp(p+1)/2$，必须为均值估计的参数数目是 Tp。必须估计的参数总数是 $Tp(p+3)/2$。

组样本大小相对于 p 很小时，组内协方差矩阵的估计值倾向于很不稳定。判别得分受组内协方差矩阵的逆矩阵的最小特征值影响很大。请参见 Friedman (1989)。因此，若您的组样本大小相对于 p 来说很小，您可能要考虑第 100 页的“正则判别方法”中所述的“正则”方法。

请参见表 5.2 了解相关符号。观测 **y** 到组 t 的 Mahalanobis 距离按以下方式定义：

$$d_t^2 = (\mathbf{y} - \bar{\mathbf{y}}_t)'\mathbf{S}_t^{-1}(\mathbf{y} - \bar{\mathbf{y}}_t)$$

观测 **y** 属于第 t 组的似然函数按以下方式估计：

$$\begin{aligned} l_t(\mathbf{y}) &= (2\pi)^{-T/2}\left|\mathbf{S}_t\right|^{-1/2}\exp(-(\mathbf{y} - \bar{\mathbf{y}}_t)'\mathbf{S}_t^{-1}(\mathbf{y} - \bar{\mathbf{y}}_t)/2) \\ &= (2\pi)^{-T/2}\left|\mathbf{S}_t\right|^{-1/2}\exp(-d_t^2/2) \end{aligned}$$

组 t 的成员关系的后验概率为：

$$\begin{aligned} p(t|\mathbf{y}) &= (q_t l_t(\mathbf{y}))/\left(\sum_{u=1}^{T} q_u l_u(\mathbf{x})\right) \\ &= \frac{1}{1 + \sum_{u \neq t}\exp(-[(d_u^2 + \log\left|\mathbf{S}_u\right| - 2\log(q_u)) - (d_t^2 + \log\left|\mathbf{S}_t\right| - 2\log(q_t))]/2)} \end{aligned}$$

观测 **y** 被分配给具有最大后验概率的组。

按以下方式定义“二次”判别方法所保存的公式：

SqDist[<组 *t*>]	$d_t^2 + \log\left\|\mathbf{S}_t\right\| - 2\log(q_t)$
Prob[<组 *t*>]	$p(t\|\mathbf{y})$
Pred <*X*>	t，对于它 $p(t\|\mathbf{y})$ 为最大值， $t = 1, ..., T$

注意：SqDist[<组 *t*>] 可为负。

正则判别方法

“正则”判别分析允许两个参数：λ 和 γ。

- 参数 λ 权衡分配给合并的协方差矩阵和组内协方差矩阵（不假定相等）的权重。
- 参数 γ 确定向对角矩阵的收缩量。

该方法允许您利用正则化的两个特征，提高了二次判别分析估计值的稳定性。请参见 Friedman (1989)。请参见表 5.2 了解相关符号。

对于正则方法，组 t 的协方差矩阵为：

$$\Sigma_t = (1-\gamma)(\lambda\mathbf{S}_p + (1-\lambda)\mathbf{S}_t) + \gamma Diag((\lambda\mathbf{S}_p + (1-\lambda)\mathbf{S}_t))$$

观测 $\mathbf{y}$ 到组 t 的 Mahalanobis 距离按以下方式定义：

$$d_t^2 = (\mathbf{y} - \bar{\mathbf{y}}_t)'\Sigma_t^{-1}(\mathbf{y} - \bar{\mathbf{y}}_t)$$

观测 $\mathbf{y}$ 属于第 t 组的似然函数按以下方式估计：

$$\begin{aligned} l_t(\mathbf{y}) &= (2\pi)^{-T/2}\left|\Sigma_t\right|^{-1/2}\exp(-(\mathbf{y} - \bar{\mathbf{y}}_t)'\Sigma_t^{-1}(\mathbf{y} - \bar{\mathbf{y}}_t)/2) \\ &= (2\pi)^{-T/2}\left|\Sigma_t\right|^{-1/2}\exp(-d_t^2/2) \end{aligned}$$

按以下方式计算组 t 的成员关系的后验概率：

$$p(t|\mathbf{y}) = (q_t l_t(\mathbf{y}))/\left(\sum_{u=1}^{T} q_u l_u(\mathbf{x})\right)$$

$$= \frac{1}{1 + \sum_{u \neq t} \exp(-[(d_u^2 + \log|\Sigma_u| - 2\log(q_u)) - (d_t^2 + \log|\Sigma_t| - 2\log(q_t))]/2)}$$

观测 **y** 被分配给具有最大后验概率的组。

“正则”判别方法所保存的公式定义如下：

SqDist[<组 *t*>]	$d_t^2 + \log\|\Sigma_t\| - 2\log(q_t)$
Prob[<组 *t*>]	$p(t\|\mathbf{y})$
Pred <*X*>	t，对于它 $p(t\|\mathbf{y})$ 为最大值， $t = 1, ..., T$

注意：SqDist[<组 *t*>] 可为负。

宽线性判别方法

当您有很多协变量特别是协变量数超过观测数 ($p > n$) 时，“宽线性”方法很有用。该方法的核心是高效计算合并的组内协方差矩阵 $\mathbf{S}_p$ 的逆矩阵或它的转置矩阵（若 $p > n$）。它使用奇异值分解方法来避免为大的协方差矩阵反转和分配空间。

“宽线性”方法假定组内协方差矩阵相等，若观测数等于或超过协变量数，则该方法等效于“线性”方法。

宽线性计算

请参见表 5.2 了解相关符号。在 “宽线性”计算中使用以下步骤：

1. 计算组内样本均值的 T x p 矩阵 **M**。**M** 的第 (t,j) 个元素 m_{tj} 是第 j 个协变量上的组 t 成员的样本均值。
2. 对于每个协变量 j，计算各组的合并标准差。称之为 s_{jj}。
3. 用 $\mathbf{S}_{diag}$ 表示具有对角元素 s_{jj} 的对角矩阵。
4. 对每个协变量的值进行中心化和统一尺度：
   - 减去观测所属组的均值。
   - 将差值除以合并标准差。

使用符号，对于组 t 中的观测 i，第 j 个协变量的组中心化和统一尺度值为：

$$y^*_{ij} = \frac{y_{ij} - m_{t(i)j}}{s_{jj}}$$

符号 $t(i)$ 表示观测 i 所属的组 t。

5. 用 $\mathbf{Y}_s$ 表示 y^*_{ij} 值的矩阵。
6. 用 $\mathbf{R}$ 表示组中心化和统一尺度的协变量的合并组内协方差矩阵。按以下方式计算矩阵 $\mathbf{R}$：

$$\mathbf{R} = (\mathbf{Y}_s'\mathbf{Y}_s)/(n-T)$$

7. 将奇异值分解应用到 $\mathbf{Y_s}$：

$$\mathbf{Y}_s = \mathbf{UDV}'$$

其中 $\mathbf{U}$ 和 $\mathbf{V}$ 是正交的，$\mathbf{D}$ 是对角线上具有正元素（奇异值）的对角矩阵。请参见“统计详细信息”附录中第 311 页的“奇异值分解”。

则 $\mathbf{R}$ 可以表示为：

$$\mathbf{R} = (\mathbf{Y}_s'\mathbf{Y}_s)/(n-T) = (\mathbf{VD}^2\mathbf{V}')/(n-T)$$

8. 若 $\mathbf{R}$ 是满秩的，按以下方式得到 $\mathbf{R}^{-1/2}$：

$$\mathbf{R}^{-1/2} = (\mathbf{VD}^{-1}\mathbf{V}')/\sqrt{n-T}$$

其中 $\mathbf{D}^{-1}$ 是对角矩阵，其对角元素为 $\mathbf{D}$ 的对角元素的逆。

若 $\mathbf{R}$ 不是满秩的，则按以下方式定义 $\mathbf{R}$ 的伪逆矩阵：

$$\mathbf{R}^- = (\mathbf{VD}^{-2}\mathbf{V}')/(n-T)$$

然后按以下方式定义 $\mathbf{R}$ 的平方根倒数：

$$(\mathbf{R}^-)^{1/2} = (\mathbf{VD}^{-1}\mathbf{V}')/\sqrt{n-T}$$

9. 若 $\mathbf{R}$ 是满秩的，结果就是 $\mathbf{R}^- = \mathbf{R}^{-1}$。因此，为了保持完整性，我们使用伪逆矩阵继续讨论。

按以下方式定义 $p \times p$ 的矩阵 $\mathbf{T_s}$：

$$\mathbf{T}_s = (\mathbf{S}^{-1}_{diag}\mathbf{VD}^-)/(\sqrt{n-T})$$

则：

$$(\mathbf{T}_s\mathbf{T}_s') = \left(\mathbf{S}^{-1}_{diag}\mathbf{V}(\mathbf{D}^-)^2\mathbf{V}'\mathbf{S}^{-1}_{diag}\right)/(n-T) = \mathbf{S}^{-1}_{diag}\mathbf{R}^-\mathbf{S}^{-1}_{diag} = \mathbf{S}^-_p$$

其中，$\mathbf{S}^-_p$ 是使用 SVD 计算的原始数据的合并组内协方差矩阵的广义逆矩阵。

Mahalanobis 距离

Mahalanobis 距离、似然函数和后验概率的公式与第 98 页的“线性判别方法”中所述的那些公式相同。但是，$\mathbf{S}_p$ 的逆矩阵由使用奇异值分解计算的广义逆矩阵替代。

当您保存公式时， Mahalanobis 距离以分解的形式给出。对于观测 $\mathbf{y}$，到组 t 的平方距离如下所示，其中最后一个等式中的 *SqDist*[0] 和判别主成分。在第 103 页的 “保存的公式”中定义：

$$\begin{aligned} d_t^2 &= (\mathbf{y}-\bar{\mathbf{y}}_t)'\bar{\mathbf{S}}_p(\mathbf{y}-\bar{\mathbf{y}}_t) \\ &= (\mathbf{y}-\bar{\mathbf{y}}_t)'\mathbf{T}_s\mathbf{T}_s'(\mathbf{y}-\bar{\mathbf{y}}_t) \\ &= ((\mathbf{y}-\bar{\mathbf{y}})-(\bar{\mathbf{y}}_t-\bar{\mathbf{y}}))'\mathbf{T}_s\mathbf{T}_s'((\mathbf{y}-\bar{\mathbf{y}})-(\bar{\mathbf{y}}_t-\bar{\mathbf{y}})) \\ &= (\mathbf{T}_s'(\mathbf{y}-\bar{\mathbf{y}}))'(\mathbf{T}_s'(\mathbf{y}-\bar{\mathbf{y}}))-2(\mathbf{T}_s'(\bar{\mathbf{y}}_t-\bar{\mathbf{y}}))'(\mathbf{T}_s'(\mathbf{y}-\bar{\mathbf{y}}))+(\mathbf{T}_s'(\bar{\mathbf{y}}_t-\bar{\mathbf{y}}))'(\mathbf{T}_s'(\bar{\mathbf{y}}_t-\bar{\mathbf{y}})) \\ &= SqDist[0]-2(\mathbf{T}_s'(\bar{\mathbf{y}}_t-\bar{\mathbf{y}}))'\text{判别主成分}+(\mathbf{T}_s'(\bar{\mathbf{y}}_t-\bar{\mathbf{y}}))'(\mathbf{T}_s'(\bar{\mathbf{y}}_t-\bar{\mathbf{y}})) \end{aligned}$$

保存的公式

这些是 “宽线性”判别方法所保存的公式：

判别数据矩阵	观测值在协变量上的向量
判别主成分	使用主成分得分矩阵变换的数据，它呈现组内不相关的数据。通过 $\mathbf{T}_s'(\mathbf{y}-\bar{\mathbf{y}})$ 给出，其中 $\bar{\mathbf{y}}$ 是包含总均值的 *1* x *p* 向量。
SqDist[0]	$(\mathbf{y}-\bar{\mathbf{y}})'\mathbf{T}'_s\mathbf{T}_s'(\mathbf{y}-\bar{\mathbf{y}})$
SqDist[<组 *t*>]	观测值到组重心的 Mahalanobis 距离。请参见第 103 页的“Mahalanobis 距离”。
Prob[<组 *t*>]	$p(t\|\mathbf{y})$，在第 98 页的“线性判别方法”中给出
Pred <*X*>	t，对于它 $p(t\|\mathbf{y})$ 为最大值，$t = 1, ..., T$

多元检验

在下文中，**E** 是残差叉积矩阵，**H** 是模型叉积矩阵。**E** 的对角线元素是每个变量的残差平方和。**H** 的对角线元素是每个变量的模型平方和。在判别分析文献中，**E** 通常称为 **W**，其中 **W** 表示**组内**。

多元结果表中的检验统计量是 $\mathbf{E}^{-1}\mathbf{H}$ 的特征值 λ 的函数。以下列表说明了每个检验统计量的计算。

注意：在指定响应设计后，将初始 **E** 和 **H** 矩阵用 **M**′ 自左乘，然后用 **M** 自右乘。

- Wilks Lambda

$$\Lambda = \frac{\det(\mathbf{E})}{\det(\mathbf{H}+\mathbf{E})} = \prod_{i=1}^{n}\left(\frac{1}{1+\lambda_i}\right)$$

- Pillai 迹

$$V = \text{迹}[\mathbf{H}(\mathbf{H}+\mathbf{E})^{-1}] = \sum_{i=1}^{n}\frac{\lambda_i}{1+\lambda_i}$$

- Hotelling-Lawley 迹

$$U = \text{迹}(\mathbf{E}^{-1}\mathbf{H}) = \sum_{i=1}^{n}\lambda_i$$

- Roy 最大根

$\Theta = \lambda_1$，$\mathbf{E}^{-1}\mathbf{H}$ 的最大特征值。

E 和 **H** 定义如下：

$$\mathbf{E} = \mathbf{Y}'\mathbf{Y} - \mathbf{b}'(\mathbf{X}'\mathbf{X})\mathbf{b}$$

$$\mathbf{H} = (\mathbf{L}\mathbf{b})'(\mathbf{L}(\mathbf{X}'\mathbf{X})^{-}\mathbf{L}')^{-1}(\mathbf{L}\mathbf{b})$$

其中 **b** 是模型系数的估计向量，$\mathbf{A}^{-}$ 表示矩阵 **A** 的广义逆。

整体模型 **L** 是与以下单位矩阵拼接而成的零列（用于截距）：该矩阵的行数和列数与模型中的参数数目相等。效应的 **L** 矩阵是整体模型 **L** 矩阵的行的子集。

近似 F 检验

为了计算 F 值和自由度，用 p 表示 $\mathbf{H}+\mathbf{E}$ 的秩。用 q 表示 $\mathbf{L}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{L}'$ 的秩，其中 **L** 矩阵标识与要检验的效应关联的 $\mathbf{X}'\mathbf{X}$ 的元素。用 v 表示误差自由度，用 s 表示 p 和 q 中的最小项。还假定 $m = 0.5(|p-q|-1)$ 和 $n = 0.5(v-p-1)$。

第 104 页上的表 5.3 列出了从相应的检验统计量如何计算每个近似 F 值。

表 5.3 近似 F 统计量

检验	近似 F	分子自由度	分母自由度
Wilks Lambda	$F = \left(\frac{1-\Lambda^{1/t}}{\Lambda^{1/t}}\right)\left(\frac{rt-2u}{pq}\right)$	pq	$rt-2u$

表 5.3 近似 *F* 统计量 (续)

检验	近似 F	分子自由度	分母自由度
Pillai 迹	$F = \left(\frac{V}{s-V}\right)\left(\frac{2n+s+1}{2m+s+1}\right)$	$s(2m+s+1)$	$s(2n+s+1)$
Hotelling-Lawley 迹	$F = \frac{2(sn+1)U}{s^2(2m+s+1)}$	$s(2m+s+1)$	$2(sn+1)$
Roy 最大根	$F = \frac{\Theta(v-\max(p,q)+q)}{\max(p,q)}$	$\max(p,q)$	$v-\max(p,q)+q$

组间协方差矩阵

使用表 5.2 中的符号，按以下方式定义该矩阵：

$$\mathbf{S}_B = \frac{1}{T-1}\sum_{t=1}^{T} T\left(\frac{n_t}{n}\right)(\bar{\mathbf{y}}_t - \mathbf{y}_{bar})(\bar{\mathbf{y}}_t - \mathbf{y}_{bar})'$$

第 6 章

偏最小二乘模型

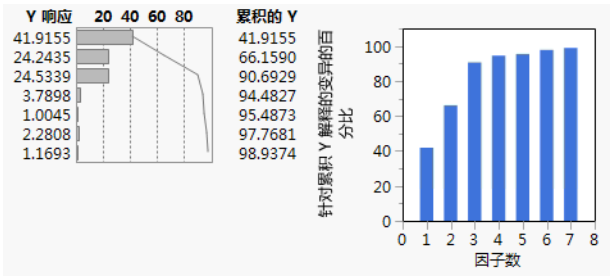
使用 Y 和 X 之间的相关性构建模型

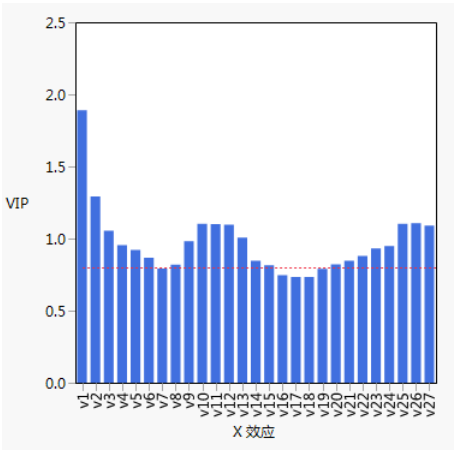
“偏最小二乘”(PLS) 平台基于因子（即解释变量 (X) 的线性组合）来拟合线性模型。这些因子是通过将 *X* 与一个或多个响应 (*Y*) 之间的协方差最大化得到的。“偏最小二乘”利用 *X* 和 *Y* 之间的相关性揭示底层的潜在结构。

JMP PRO JMP Pro 提供更多功能，从而支持您构造“PLS 判别分析 (PLS-DA)”，包括各种模型效应、利用几种验证方法、补缺缺失数据以及获取各种统计量分布的 Bootstrap 估计值。

偏最小二乘适用于以下情况：X 变量数比观测数多；X 变量之间高度相关；X 变量非常多；只有几个 Y 变量但有很多 X 变量，此时使用普通最小二乘法将无法得到满意的结果。

图 6.1 “偏最小二乘”报表的一部分





目录

“偏最小二乘”平台概述 109
“偏最小二乘”示例 109
启动“偏最小二乘”平台 113
中心化和统一尺度 116
标准化 X 116
“模型启动”控制面板 117
“偏最小二乘”选项 118
“偏最小二乘”报表 118
模型比较汇总 118
“交叉验证”报表 119
“模型拟合”报表 123
模型拟合选项 124
变量重要性图 126
系数 -VIP 图 126
保存列 127
“偏最小二乘”的其他示例 128
“偏最小二乘”平台的统计详细信息 130
偏最小二乘 130
van der Voet T^2 131
T^2 图 132
X 得分散点图矩阵的置信椭圆 132
预测的标准误差和置信限 132
标准化得分和载荷 133
PLS 判别分析 (PLS-DA) 134

“偏最小二乘”平台概述

与普通最小二乘法相比，当预测变量数比观测数多时，可以使用 PLS。PLS 广泛用于对高维数据建模的领域中，如光谱学、化学计量学、基因组学、心理学、教育、经济学、政治学和环境学。

当解释变量数比观测数多或解释变量高度相关时，使用 PLS 方法拟合模型很有用。您可以使用 PLS 来同时对几个响应变量拟合一个模型。请参见 Garthwaite (1994), Wold (1994), Wold et al. (2001)、Eriksson et al. (2006) 和 Cox and Gaudard (2013)。

该平台提供两种模型拟合算法：非线性迭代偏最小二乘 (NIPALS) 和“PLS 的统计启发修改” (SIMPLS)。有关 NIPALS 的详细信息，请参见 Wold (1980)。有关 SIMPLS 的详细信息，请参见 De Jong (1993)。有关两种方法的说明，请参见 Boulesteix and Strimmer (2007)。SIMPLS 算法开发的目的在于解决特定的最优问题。对于单个响应，这两种方法会给出相同的模型。对于多重响应，它们略有不同。

在 JMP 中，只能通过“分析”>“多元方法”>“偏最小二乘”来访问 PLS 平台。在 JMP Pro 中，您还可以通过“分析”>“拟合模型”来访问“偏最小二乘”特质。

JMP PRO 在 JMP Pro 中，您可以：

- 使用“拟合模型”中的“偏最小二乘”特质，通过拟合具有名义型建模类型的响应来执行 PLS-DA（PLS 判别分析）。
- 使用“拟合模型”中的“偏最小二乘”特质拟合多项式、交互作用和分类效应。
- 在几种验证和交叉验证方法中进行选择。
- 补缺缺失数据。
- 获取各种统计量分布的 Bootstrap 估计值。右击关注的报表。有关 Bootstrap 估计值的详细信息，请参见《基本分析》。

“偏最小二乘”使用 van der Voet T^2 检验和交叉验证来帮助您选择要提取的最佳因子数。

- 在 JMP 中，该平台使用“留一法”交叉验证。您也可以选择不使用验证。
- JMP PRO 在 JMP Pro 中，您可以选择“K 重”、“留一法”或“随机保留”交叉验证，也可以指定一个验证列。您也可以选择不使用验证。

“偏最小二乘”示例

本例来自光谱仪校准，这是偏最小二乘很有效的领域。假定您在研究波罗的海的污染问题。您想使用海水样本的光谱确定这些样本中存在的三种化合物的含量。

关注的三种化合物为：

- 木质素磺酸盐 (ls)，它是纸浆工业产生的污染
- 黑腐酸 (ha)，它是天然森林的产物
- 洗涤剂 (dt) 产生的增白剂

这三种化合物在每个样本中的含量用响应表示。预测变量表示为在某个波长范围内 (v1–v27) 测量的光谱发射强度。

为了校准模型，使用了已知成分的样本。校准数据包含已知木质素磺酸盐、黑腐酸和洗涤剂浓度的 16 个样本。记录了 27 个等距波长下的发射强度。使用 “偏最小二乘”平台生成模型来根据光谱发射强度预测化合物含量。

1. 选择**帮助 > 样本数据库**，然后打开 Baltic.jmp。

   注意: Baltic.jmp 数据表中的数据在 Umetrics (1995) 中报告。原始来源是 Lindberg, Persson and Wold (1983)。

2. 选择**分析 > 多元方法 > 偏最小二乘**。
3. 将 ls、 ha 和 dt 分配给 **Y，响应**角色。
4. 将 Intensities。（它包含 27 个强度变量 v1-v27）分配给 **X，因子**角色。
5. 点击**确定**。

   随即显示 “偏最小二乘模型启动”控制面板。
6. 选择**留一法**作为验证方法。
7. 点击**执行**。

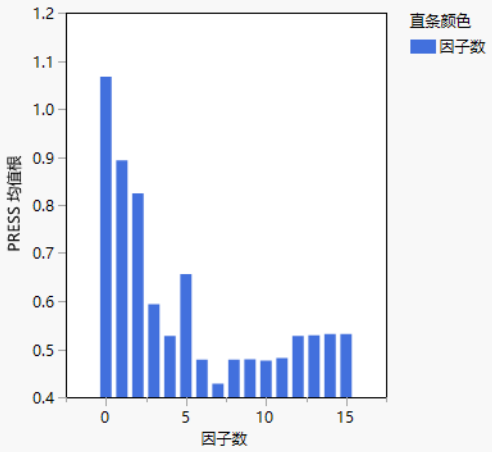
   因为 van der Voet 检验是随机化检验，您的 “概率 > van der Voet T^2”值可能略有不同。

图 6.2 “偏最小二乘”报表

留一法交叉验证（方法为“NIPALS”，使用快速 SVD）

因子数	PRESS 均值根	van der Voet T^2	概率 > van der Voet T^2	Q^2	累积 Q^2	R^2X	累积 R^2X	R^2Y	累积 R^2Y
0	1.066667	10.070473	0.0020*	-0.137778	-0.137778	0.000000	0.000000	0.000000	0.000000
1	0.892851	11.141942	<.0001*	-0.098077	-0.098077	0.974480	0.974480	0.421421	0.421421
2	0.824551	8.605908	0.0020*	0.773385	0.751159	0.022034	0.996514	0.247724	0.669146
3	0.593922	8.034522	0.0060*	0.774570	0.943904	0.001772	0.998286	0.241683	0.910829
4	0.527742	7.879423	0.0040*	0.656229	0.980716	0.001153	0.999438	0.037043	0.947872
5	0.656013	6.883024	0.0120*	0.356257	0.987586	0.000386	0.999825	0.012502	0.960374
6	0.478479	3.446269	0.3370	0.640049	0.995532	0.000113	0.999938	0.019350	0.979724
7	0.428141	0.000000	1.0000	0.809320	0.999148	0.000017	0.999955	0.010643	0.990367
8	0.478562	7.031041	0.0110*	0.818516	0.999845	0.000011	0.999967	0.004666	0.995032
9	0.478739	3.066627	0.4080	0.810850	0.999971	0.000012	0.999979	0.001452	0.996485
10	0.476024	3.847943	0.2790	0.845631	0.999995	0.000007	0.999986	0.001417	0.997902
11	0.481440	4.822874	0.1080	0.820847	0.999999	0.000004	0.999990	0.001248	0.999150
12	0.527436	3.968884	0.1800	0.810490	1.000000	0.000004	0.999994	0.000637	0.999787
13	0.529249	4.028960	0.2100	0.812180	1.000000	0.000003	0.999997	0.000158	0.999945
14	0.531553	4.408278	0.1440	0.816103	1.000000	0.000003	1.000000	0.000055	1.000000
15	0.531553	4.408278	0.1480	0.816103	1.000000	0.000000	1.000000	0.000000	1.000000

PRESS 均值根图

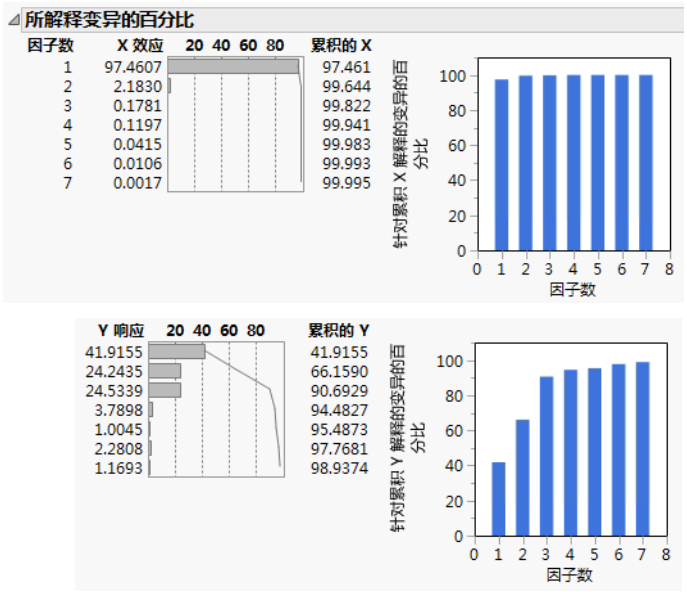


注意: 最小 PRESS 均值根是 0.42814，且最低因子数是 7。

“PRESS (预测残差平方和) 均值根图”显示当因子数为 7 时 PRESS 均值根的值最小。这在“PRESS 均值根图”下的注释中有说明。生成名为**带 7 个因子的 “NIPALS”拟合，使用快速 SV** 的报表。该报表的一部分显示在图 6.3 中。

van der Voet T^2 统计量检验具有不同因子数的模型是否与具有最小 PRESS 值的模型显著不同。常见做法是提取 van der Voet 显著性水平超过 0.10 的最小因子数 (SAS Institute Inc, 2020f; Tobias 1995)。若您要在此处应用该方法，可以通过在**模型启动**面板中输入 6 作为**因子数**来拟合新模型。

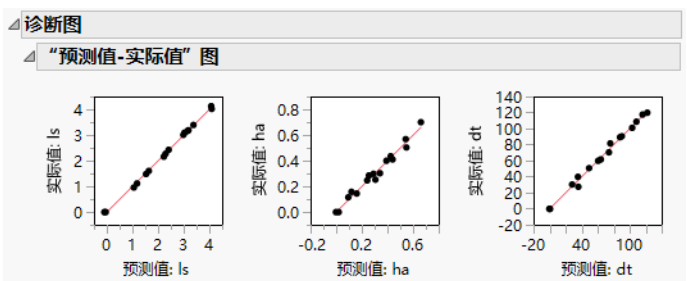
图 6.3 提取的七个因子



8. 点击 “带 7 个因子的 “NIPALS”拟合，使用快速 SV”红色小三角并选择**诊断图**。

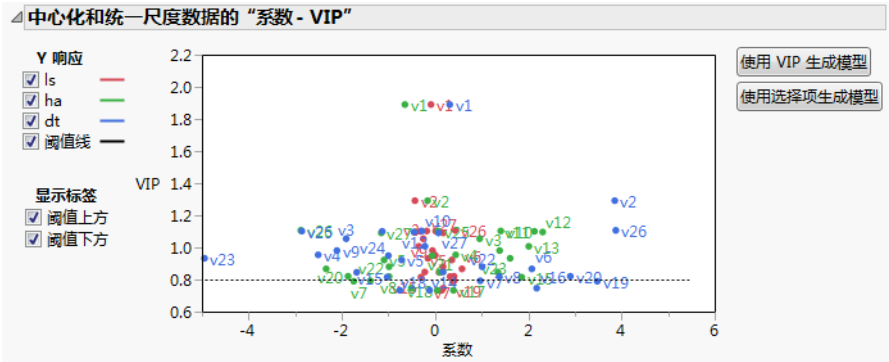
   这得到一个报表用于显示 “预测值 - 实际值”图以及三个报表用于显示各种残差图。“‘预测值 - 实际值’图”显示预测的化合物含量与实际含量相一致的程度。

图 6.4 诊断图



9. 选择 “带 7 个因子的 ‘NIPALS’拟合，使用快速 SV”红色小三角并选择**系数 -VIP 图**。

图 6.5 系数 -VIP 图



“系数 -VIP 图” 帮助识别对拟合多个响应有影响的变量。例如， v23、 v2 和 v26 都有超过 0.8 的 VIP 值和相对较大的系数。

启动 “偏最小二乘” 平台

有两种启动 “偏最小二乘” 平台的方法：

- 选择**分析 > 多元方法 > 偏最小二乘**。
- JMP PRO 选择**分析 > 拟合模型**，然后从 “特质” 菜单中选择**偏最小二乘**。该方法允许您执行以下操作：
  - 输入分类变量作为 *Y* 或 *X*。通过输入分类 *Y* 来执行 PLS-DA。
  - 将交互作用项和多项式项添加到您的模型。
  - 使用 “标准化 X” 选项以使用中心化和统一尺度的列构造高阶项。
  - 保存您的模型规格脚本。

“拟合模型” 启动窗口上的一些功能不适用于 “偏最小二乘” 特质：

- “权重”、“嵌套”、“特性”、“变换” 和 “无截距”。

  提示：您可以通过在 “选择列” 框中右击某个变量并选择一个 “变换” 选项来变换该变量。

- 下面的宏：混料响应曲面和 Scheffé 三次项。

图 6.6 JMP Pro “偏最小二乘”启动窗口 （选择了 EM 作为补缺方法）

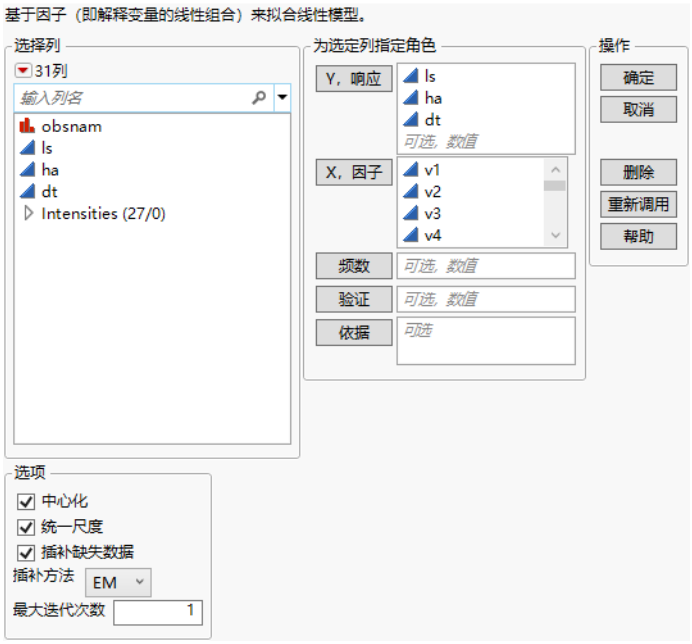
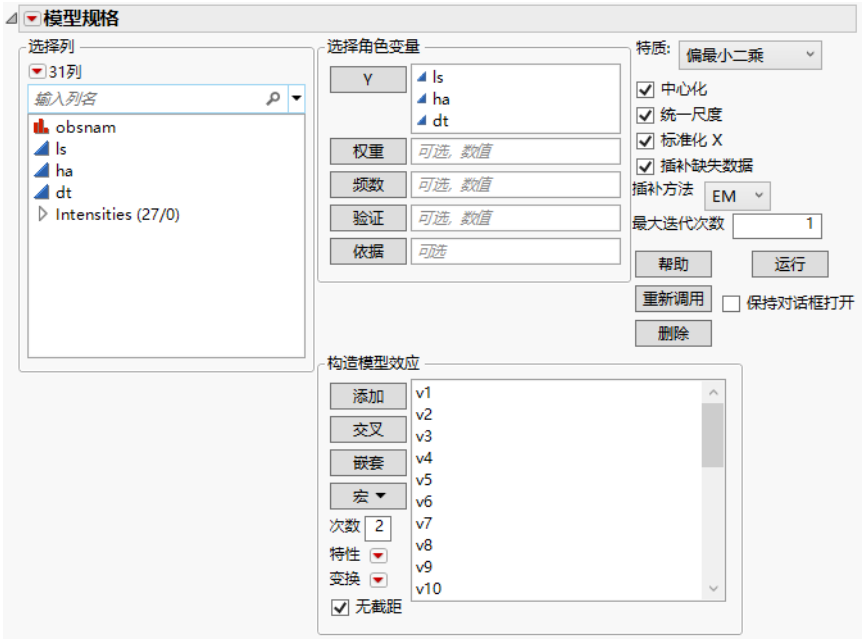


图 6.7 JMP Pro 拟合模型 “偏最小二乘”启动窗口



有关“拟合模型”启动选项的详细信息，请参见《拟合线性模型》。

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“偏最小二乘”启动窗口包含以下选项：

Y，响应 输入数值响应列。若您输入了多个列，则它们联合建模。

JMP PRO 在 JMP Pro 中，您可以在“拟合模型”启动窗口中输入名义型响应列来执行 PLS-DA。请参见第 134 页的 “PLS 判别分析 (PLS-DA)”。

X，因子 输入预测变量列。“偏最小二乘”启动窗口只允许数值型预测变量。

JMP PRO 在 JMP Pro 中，您可以在 “拟合模型”启动窗口中输入名义型和有序型模型效应。有序型效应被视为名义型处理。

频数 若您的数据进行了汇总，则输入其值包含各行计数的列。

JMP PRO **验证** 用于定义验证集的数值列。验证列只能包含连续的整数值。

- 若验证列有两个水平，则较小的值定义训练集，较大的值定义验证集。
- 若验证列有三个水平，则这些值按大小递增的顺序定义训练集、验证集和测试集。
- 若验证列有三个以上的水平，则使用“K 重交叉验证”。有关其他验证选项的信息，请参见第 117 页的“验证方法”。

PLS 平台使用验证列来训练和微调模型，或者训练、微调和评估模型。有关验证的详细信息，请参见 《预测和专业建模》。

注意：若在 “选择列”列表中没有选择任何列的情况下点击 “验证”按钮，您可以向数据表添加一个验证列。有关“生成验证列”实用工具的详细信息，请参见《预测和专业建模》。

依据 输入一列，用于创建为变量的每个水平包含单独分析的报表。若分配了多个 “依据”变量，则为 “依据”变量水平的每个可能组合生成单独分析。

中心化 通过从每个列减去均值将所有 Y 变量和模型效应中心化。请参见第 116 页的 “中心化和统一尺度”。

统一尺度 通过将每个列除以其标准差对所有 Y 变量和模型效应统一尺度。请参见第 116 页的 “中心化和统一尺度”。

JMP PRO **标准化 X** （仅可用于“拟合模型”启动窗口。）将构造模型效应时使用的所有列中心化和统一尺度。若未选择该选项，则使用原始数据表列构造高阶效应。然后基于所选的 “中心化”和 “统一尺度”选项将每个高阶效应中心化或统一尺度。请注意，“标准化 X”选项不将 Y 变量中心化或统一尺度。请参见第 116 页的 “标准化 X”。

JMP PRO **补缺缺失数据** 使用非缺失值替代 Y 或 X 中的缺失数据值。从**补缺方法**列表中选择合适的方法。

若未选择**补缺缺失数据**，则从分析中排除在任何 X 变量上具有缺失观测的行，而且不为这些行计算预测值。此外还会从分析中排除在 X 变量上不具有缺失观测但是在 Y 变量上具有缺失观测的行，但是计算预测值。

JMP PRO **补缺方法** （仅在**补缺缺失数据**时显示。）从以下补缺方法中选择：

均值 对于每个模型效应或响应列，使用非缺失值的均值替代缺失值。

EM 使用迭代期望值最大化 (EM) 方法来补缺缺失值。在第一次迭代时，使用均值替代效应或响应的缺失值，对数据拟合指定的模型。使用 Y 模型的预测值和 X 模型的预测值来补缺缺失值。对于后续迭代，在使用当前估计值给出条件分布时，使用其预测值替代缺失值。

为了进行补缺，多项式中的项被视为单独的预测变量处理。当指定了多项式中的项时，将根据原始数据计算该项；若选中了“标准化 X”，则根据标准化的列值计算该项。若某一行在定义多项式中的项所涉及的列中存在缺失值，则该条目对于多项式中的项是缺失的。使用这样定义的多项式中的项执行补缺。

有关 EM 方法的详细信息，请参见 Nelson, Taylor and MacGregor (1996)。

JMP PRO **最大迭代次数** （仅在将 EM 选作补缺方法时才显示。）支持您设置算法使用的最大迭代次数。若缺失值的当前估计值和前一估计值之间的最大差值不超过 10^{-8}，算法将终止。

在完成启动窗口并点击**确定**后，将显示“模型启动”控制面板。请参见第 117 页的““模型启动”控制面板”。

中心化和统一尺度

默认情况下选定“中心化”和“统一尺度”选项。这意味着预测变量和响应会中心化和统一尺度以具有均值 0 和标准差 1。将预测变量和响应中心化可以使它们相对于变异是对等的。若没有中心化，则在构建后续因子中会涉及变量的均值以及围绕该均值的变异。为了进行说明，假定**时间**和**温度**是两个预测变量。对它们统一尺度表示**时间**的一个标准差变化大约等效于**温度**的一个标准差变化。

标准化 X

JMP PRO 在“拟合模型”窗口中选择“偏最小二乘”特质时，默认选定“标准化 X”选项。这确保作为模型效应输入的所有列以及交互作用项或多项式项涉及的所有列被标准化。

假定您有两个列 X1 和 X2，并且在 “拟合模型”窗口中输入交互作用项 X1*X2 作为模型效应。当选定 “标准化 X”选项时，在形成交互作用项前已将 X1 和 X2 中心化和统一尺度。形成的交互作用项按以下方式计算：

$$\left(\frac{X1 - mean(X1)}{std(X1)}\right) \times \left(\frac{X2 - mean(X2)}{std(X2)}\right)$$

在进入模型前，根据您对**中心化**和**统一尺度**选项的选择，将所有模型效应中心化或统一尺度。

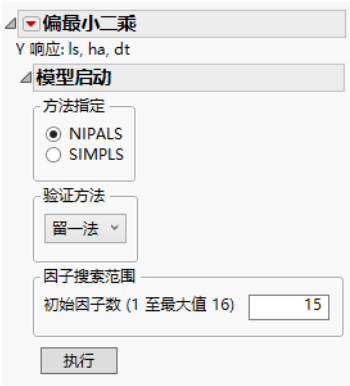
若未选择 “标准化 X”选项但是选择了**中心化**和**统一尺度**，则按以下方式计算进入模型的项：

$$\frac{X1 \times X2 - mean(X1 \times X2)}{std(X1 \times X2)}$$

“模型启动”控制面板

在平台启动窗口中点击**确定**（或在“拟合模型”窗口中点击**运行**）后，将显示“模型启动”控制面板。

图 6.8 “偏最小二乘模型启动”控制面板



注意：在 JMP Pro 中，显示的“模型启动”控制面板的“验证方法”部分有所不同。

“模型启动”控制面板包含以下选项：

方法指定 选择模型拟合算法的类型。有两个算法选项：NIPALS 和 SIMPLS。当只有一个响应变量时，这两种方法生成相同的系数估计值。有关两种算法之间的差异的详细信息，请参见第 130 页的 ““偏最小二乘”平台的统计详细信息”。

验证方法 选择验证方法。验证用于确定要提取的最佳因子数。对于 JMP Pro，若在平台启动窗口中指定了验证列，则不显示这些选项。

JMP PRO **保留** 随机选择指定比例的数据作为验证集，并使用除验证集外的数据拟合模型。随机选择基于对模型因子的分层抽样，意图创建比基于简单随机抽样的训练集和验证集更平衡的训练集和验证集。

JMP PRO **K 重** 将数据分割为 *K* 个子集或**重**。依次使用每个重验证拟合其余数据的模型（拟合总共 *K* 个模型）。该方法最合适小数据集，因为它有效使用了有限的数据量。

留一法 执行留一法交叉验证。

无 不使用验证来选择要提取的因子数。在“因子搜索范围”中指定因子数。

因子搜索范围 指定在未使用验证时要提取多少个潜在因子。若使用验证，则这是在选择最佳因子数前平台尝试拟合的最大因子数。最大因子数是非缺失行数和因子数中的最小值。于是，初始因子数是 15 和最大因子数中的最小值。

因子规格 在点击**执行**以拟合初始模型后显示。指定要在拟合新模型中使用的因子数。

执行 使用给定的规格启动模型拟合。

“偏最小二乘”选项

“偏最小二乘”红色小三角菜单包含一些在点击“模型启动”控制面板中的“执行”之前有用的选项。

JMP PRO **设置随机种子** 设置用于“K 重”和“保留”验证的随机化过程的种子。若您要重新生成分析，该选项会很有用。将种子设置为正值然后保存脚本，该种子将自动保存在脚本中。运行该脚本可以始终生成相同的交叉验证分析。当“验证方法”设置为“无”或使用验证列时，不显示该选项。

JMP PRO **SVD** 显示一个子菜单，该子菜单支持您在模型拟合算法中选择 SVD 算法的快速实现或经典实现。默认值为“快速 SVD”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“偏最小二乘”报表

在“模型启动”控制面板（图 6.8）中首次点击**执行**时，将从“模型启动”窗口中删除“验证方法”面板。若您指定了验证列或在“验证方法”面板中选择了“保留”，则报表中的所有模型拟合均基于训练数据。否则，所有模型拟合均基于整个数据集。

若您使用了验证，则显示三个报表：

- 模型比较汇总
- “交叉验证”报表
- 带 <N> 个因子的 NIPALS（或 SIMPLS）拟合

若您选择了**无**作为交叉验证方法，则显示两个报表：

- 模型比较汇总
- 带 <N> 个因子的 NIPALS（或 SIMPLS）拟合

要拟合更多模型，请在“模型启动”面板中指定所需的因子数。

模型比较汇总

“模型比较汇总”显示每个拟合模型的汇总结果。

图 6.9 模型比较汇总

模型比较汇总

方法	SVD	行数	因子数	针对累积 X 解释的变异的百分比	针对累积 Y 解释的变异的百分比	VIP 数目 > 0.8
NIPALS	快速	16	7	99.995152	98.937438	22
NIPALS	快速	16	6	99.993471	97.768092	22

报表包含以下汇总信息：

方法 显示您在 “模型启动”控制面板中指定的分析方法。

SVD 显示指定的 SVD 算法实现。

行数 显示训练集中使用的观测数。

因子数 显示提取的因子数。

针对累积 X 解释的变异的百分比 显示模型解释的 X 的变异百分比。

针对累积 Y 解释的变异的百分比 显示模型解释的 Y 的变异百分比。

VIP>0.8 的数目 显示 VIP（变量投影重要性）值大于 0.8 的模型效应数。VIP 得分是衡量变量相对 X 和 Y 建模的重要性的指标 (Wold 1994; Eriksson et al. 2006)。

“交叉验证”报表

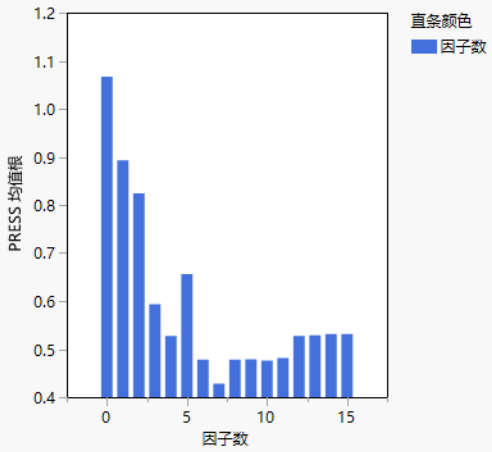
仅当在“模型启动”控制面板中将某种交叉验证方式选为“验证方法”时，才显示该报表。报表标题被动态地命名为 <交叉验证方法> 且方法 = <方法指定>，具体取决于在控制面板中选定的交叉验证和方法选项。它显示模型拟合的汇总统计量，拟合所用的因子数为 0 到提取的最大因子数（在“模型启动”控制面板中指定）。该报表还提供“PRESS 均值根值图”。请参见第 122 页的“PRESS 均值根图”。使用最小 PRESS 均值根统计量标识最佳因子数。

图 6.10 “交叉验证”报表

留一法交叉验证 (方法为“NIPALS”，使用 快速 SVD)

因子数	PRESS 均值根	van der Voet T²	概率 > van der Voet T²	Q²	累积 Q²	R²X	累积 R²X	R²Y	累积 R²Y
0	1.066667	10.070473	0.0020*	-0.137778	-0.137778	0.000000	0.000000	0.000000	0.000000
1	0.892851	11.141942	<.0001*	-0.098077	-0.098077	0.974480	0.974480	0.421421	0.421421
2	0.824551	8.605908	0.0020*	0.773385	0.751159	0.022034	0.996514	0.247724	0.669146
3	0.593922	8.034522	0.0060*	0.774570	0.943904	0.001772	0.998286	0.241683	0.910829
4	0.527742	7.879423	0.0040*	0.656229	0.980716	0.001153	0.999438	0.037043	0.947872
5	0.656013	6.883024	0.0120*	0.356257	0.987586	0.000386	0.999825	0.012502	0.960374
6	0.478479	3.446269	0.3370	0.640049	0.995532	0.000113	0.999938	0.019350	0.979724
7	0.428141	0.000000	1.0000	0.809320	0.999148	0.000017	0.999955	0.010643	0.990367
8	0.478562	7.031041	0.0110*	0.818516	0.999845	0.000011	0.999967	0.004666	0.995032
9	0.478739	3.066627	0.4080	0.810850	0.999971	0.000012	0.999979	0.001452	0.996485
10	0.476024	3.847943	0.2790	0.845631	0.999995	0.000007	0.999986	0.001417	0.997902
11	0.481440	4.822874	0.1080	0.820847	0.999999	0.000004	0.999990	0.001248	0.999150
12	0.527436	3.968884	0.1800	0.810490	1.000000	0.000004	0.999994	0.000637	0.999787
13	0.529249	4.028960	0.2100	0.812180	1.000000	0.000003	0.999997	0.000158	0.999945
14	0.531553	4.408278	0.1440	0.816103	1.000000	0.000003	1.000000	0.000055	1.000000
15	0.531553	4.408278	0.1480	0.816103	1.000000	0.000000	1.000000	0.000000	1.000000

PRESS 均值根图



注意: 最小 PRESS 均值根是 0.42814，且最低因子数是 7。

JMP PRO 当选定**标准化 X** 选项时，标准化会一次性应用到整个数据表。它不会重新应用到各个训练集。但是，当选定**中心化**或**统一尺度**选项的任意组合时，该选择组合会应用到每个交叉验证训练集。使用这些训练集执行交叉验证，若选定这些选项则会对它们单独进行中心化和统一尺度。

报表中显示以下统计量。若使用了任何形式的验证或交叉验证，则报告的结果是训练集统计量的汇总。

因子数 拟合模型时使用的因子数。

PRESS 均值根 所有响应的 PRESS 值的平均值的平方根。请参见第 122 页的“PRESS 均值根”。

van der Voet T^2 van der Voet 检验的统计量，它检验具有不同提取因子数的模型是否与最佳模型存在显著差异。每个 van der Voet T^2 检验的原假设假定基于相应因子数的模型与最佳模型没有差异。备择假设是该模型与最佳模型有差异。请参见第 131 页的“van der Voet T^2”。

概率 > van der Voet T^2 van der Voet T^2 检验的 p 值。请参见第 131 页的“van der Voet T^2”。

Q^2 预测能力的无量纲测度，它定义为：从 1 中减去 PRESS 值除以 Y 的总平方和所得的比值，即：

$$1 - \text{PRESS/SSY}$$

请参见第 123 页的 “Q^2 的计算”。

累积 Q^2 具有给定因子数或更少因子数的模型的预测能力指标。对于给定的因子数 *f*，按以下方式定义累积 Q^2：

$$1 - \prod_{i=1}^{f} (\text{Press}_i/\text{SYY}_i)$$

此处 $PRESS_i$ 和 SSY_i 对应于它们针对 *i* 个因子的值。

R^2X 由指定因子解释的 X 变异的百分比。具有较大 R^2X 的成分解释 X 变量中的大部分变异。请参见第 123 页的 “使用验证时 R^2X 和 R^2Y 的计算”。

累积 R^2X 具有给定因子数的模型所解释的 X 变异的百分比。它是 R^2X 值之和 （$i = 1$ 到给定的因子数）。

R^2Y 由指定因子解释的 Y 变异的百分比。具有较大 R^2Y 的成分解释 Y 变量中的大部分变异。请参见第 123 页的 “使用验证时 R^2X 和 R^2Y 的计算”。

累积 R^2Y 具有给定因子数的模型所解释的 Y 变异的百分比。它是 R^2Y 值之和 （$i = 1$ 到给定的因子数）。

Q^2 和累积 R^2Y 的解释

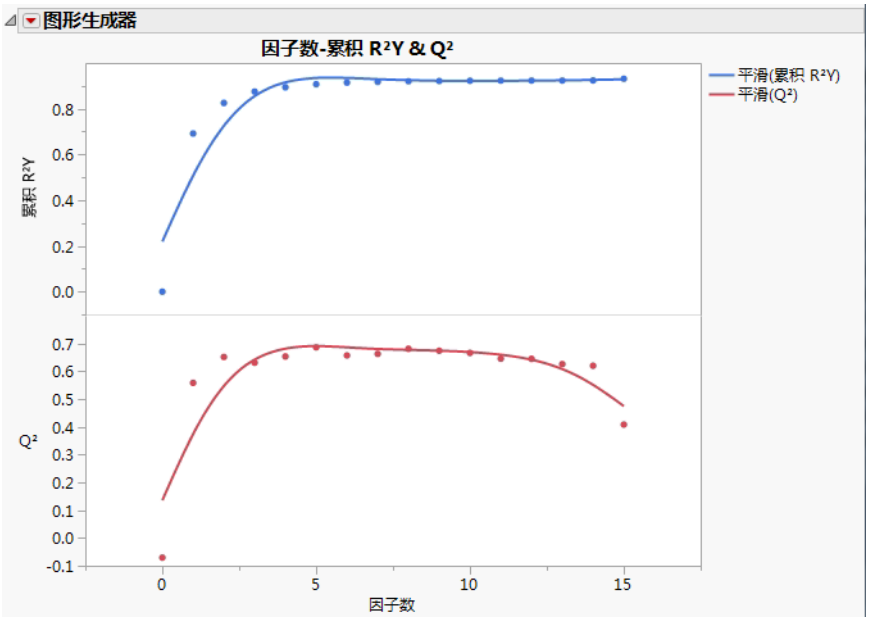
统计量 Q^2 和累积 R^2Y 均可衡量模型的预测能力，但方法不同。

- 累积 R^2Y 随因子数的增加而增大。这是因为随着因子添加至模型，更多的变异得到解释。
- Q^2 的趋势是随因子数的增加，先增大后减小（或至少停止增大）。这是因为随着更多的因子添加至模型，模型开始调整训练集，对于新数据却得不到较好的结果，导致 PRESS 统计量减小。

Q^2 和累积 R^2Y 分析是 van der Voet 检验的一个替代方法，用于确定要在模型中包括多少因子。选择一个因子数，使得 Q^2 较大且没有开始减小。您还希望累积 R^2Y 较大。

图 6.11 显示 Penta.jmp 数据表针对因子数标绘的累积 R^2Y 和 Q^2，验证方法为“留一法”。累积 R^2Y 增加，并且在大约四个因子时开始趋于平稳。统计量 Q^2 在两个因子时最大，然后开始趋于平稳。该图表明具有两个因子的模型将能够解释 Y 中的大部分变异，且不会过拟合数据。

图 6.11 Penta.jmp 的累积 R^2Y 和 Q^2



PRESS 均值根图

该条形图在水平轴上显示因子数，在垂直轴上显示“PRESS 均值根”值。它等效于在“交叉验证”报表（图 6.10）中显示在“PRESS 均值根”列右侧的水平条形图。

PRESS 均值根

对于指定的因子数 *a*，使用以下步骤计算 PRESS 均值根：

1. 对每个训练集拟合具有 *a* 个因子的模型。
2. 将得到的预测公式应用到验证集中的观测。
3. 对于每个 Y：
   - 对于每个验证集，计算每个验证集的观测值与其预测值的差值平方（预测误差平方）。
   - 对于每个验证集，求这些差值平方的平均值并将结果除以响应的方差估计值。对于“K 重”和“留一法”验证方法，除以整个响应列的方差。对于“保留”验证方法，除以训练集中响应值的方差。
   - 将这些均值相加，若有多个验证集时，则将它们的总和除以验证集数减 1。这是给定 Y 的 PRESS 统计量。
4. *a* 个因子的 “PRESS 均值根”是所有响应的 PRESS 值的平均值的平方根。
5. 多个 Y 的 PRESS 统计量通过计算在第 3 步中获得的全部响应的 PRESS 统计量的平均值得到。

Q^2 的计算

统计量 Q^2 定义为 1 - *PRESS* / *SSY*。*PRESS* 统计量是模型的所有响应的预测误差平方和平均值，该模型基于训练集构建，但是基于验证集计算。*SSY* 的值是所有响应的 Y 的平方和平均值，这些响应基于验证集中的观测值。

根据选择的验证方法，按以下方式计算 “交叉验证”报表中的统计量 Q^2：

留一法 Q^2 是针对验证集计算的 1 - *PRESS* / *SSY* 值的平均值，指标计算基于通过一次留一个观测构造的模型。

K 重 Q^2 是针对验证集计算的 1 - *PRESS* / *SSY* 值的平均值，指标计算基于通过每次留 *K* 个子集中的一个构造的 *K* 个模型。

保留或验证集 Q^2 是针对验证集计算的 1 - *PRESS* / *SSY* 值的平均值，指标计算基于使用单个训练数据集构造的模型。

使用验证时 R^2X 和 R^2Y 的计算

根据选择的验证方法，按以下方式计算“交叉验证”报表中的统计量 R^2X 和 R^2Y：

注意：对于所有这些计算，以类似方式计算 R^2Y。

留一法 通过每次留一个观测来构造模型，针对这些模型的 X 效应解释的变异百分比的平均值即为 R^2X。

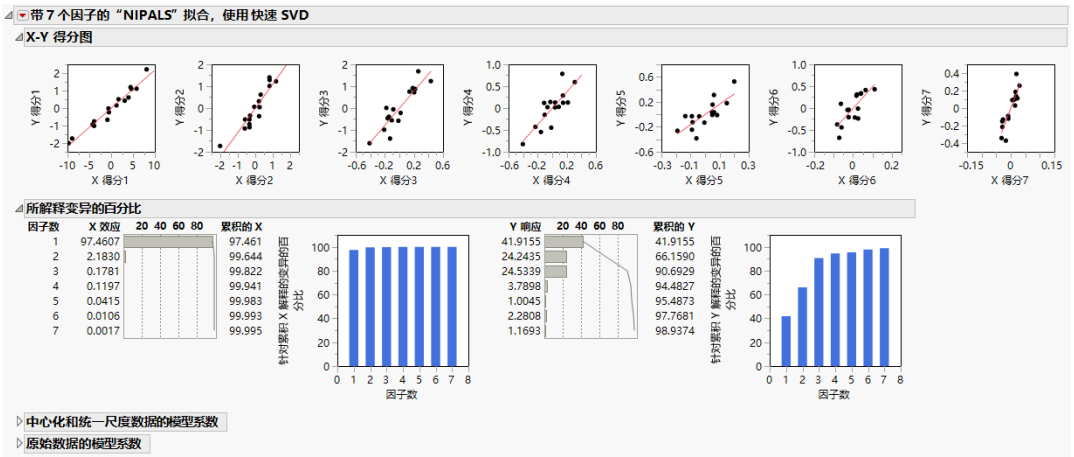
K 重 通过每次留一个子集来构造 K 个模型，针对这些模型的 X 效应解释的变异百分比的平均值即为 R^2X。

保留或验证集 使用训练数据来构造模型，针对该模型的 X 效应解释的变异百分比即为 R^2X。

“模型拟合”报表

“模型拟合”报表显示每个拟合模型的详细结果。该拟合使用基于交叉验证的最佳因子数，若未指定交叉验证方法，则使用指定的因子数。报表标题会指出使用的是 NIPALS 还是 SIMPLS，使用的是“快速 SVD”还是“经典 SVD”，并给出提取的因子数。

图 6.12 “模型拟合”报表



“模型拟合”报表包含以下汇总信息：

X-Y 得分图 提取的每个因子的 X 和 Y 得分的散点图。

所解释变异的百分比 显示针对 X 和 Y 解释的变异的百分比和累积百分比。针对每个提取的因子给出结果。

中心化和统一尺度数据的模型系数 对于每个 Y，显示基于中心化和统一尺度数据的模型的 X 的系数。

模型拟合选项

“模型拟合”红色小三角菜单包含以下选项：

变异百分比图 添加名为“针对 X 效应解释的变异的百分比”和“针对 Y 效应解释的变异的百分比”的两个图。它们显示堆叠条形图，分别表示针对 *X* 和 *Y* 的每个提取的因子所解释的变异百分比。

变量重要性图 标绘每个 X 变量的 VIP 值。VIP 得分显示在“变量重要性表”中。请参见第 126 页的 “变量重要性图”。

系数 -VIP 图 针对模型系数标绘 VIP 统计量。您可以只显示与选定的 *Y* 对应的那些点。会额外提供标签选项。它们是根据中心化和统一尺度数据以及原始数据绘制的图。请参见第 126 页的 “系数 -VIP 图”。

设置 VIP 阈值 设置 “变量重要性图”、“方差重要性表”和 “系数 -VIP 图”的阈值水平。

系数图 针对 X 变量上的每个响应标绘模型系数。您可以只显示与选定的 *Y* 对应的那些点。它们是根据中心化和统一尺度数据以及原始数据绘制的图。

载荷图 对每个提取的因子标绘 X 和 Y 载荷。它们是针对 *X* 和 *Y* 单独绘制的图。

载荷散点图矩阵 显示 X 载荷和 Y 载荷的散点图矩阵。

载荷相关图 显示叠加在同一个图上的 X 和 Y 载荷的单个散点图或散点图矩阵。选择该选项时，您需要指定要标绘多少个因子。

- 若您指定两个因子，则显示单个载荷相关散点图。在图下面选择用于定义轴的两个因子。点击右箭头按钮可接连在图上显示因子的每种组合。
- 若您指定两个以上的因子，则显示一个散点图矩阵，每个矩阵单元内显示一对因子的散点图，显示的因子数由您选择的数目决定。

在这两种情况下，使用复选框来控制标签。

X-Y 得分图 包含以下选项：

拟合线 在“X-Y 得分图”上显示或隐藏穿过数据点的拟合线。

显示置信带 在“X-Y 得分图”上显示或隐藏拟合线的 95% 置信带。

得分散点图矩阵 显示 X 得分和 Y 得分的散点图矩阵。每个 X 得分散点图显示 95% 置信椭圆，该椭圆可用于离群值检测。有关置信椭圆的统计详细信息，请参见第 132 页的“X 得分散点图矩阵的置信椭圆”。

距离图 显示以下距离的图：

- 每个观测到 X 模型的距离
- 每个观测到 Y 模型的距离
- 到 X 模型的距离与到 Y 模型的距离的散点图

在拟合效果好的模型中，X 和 Y 距离均很小，因此点接近原点 (0,0)。使用这些图可以查找相对 X 或 Y 的离群值。若一组点聚类在一起，则它们可能有共同的特性，可以单独分析。在使用验证集或验证集和测试集时，将为它们和训练集分别提供不同的报表。

T 方图 显示每个观测的 T^2 统计量以及控制限的图。基于观测在提取因子上的得分计算观测的 T^2 统计量。有关 T^2 和控制限计算的详细信息，请参见第 132 页的“T^2 图”。

诊断图 显示诊断图以评估模型拟合效果。提供四种图：“预测值 - 实际值”图、“预测值 - 残差”图、“行号 - 残差”图和“残差正态分位数图”。这些图提供给每个响应。在使用验证集或验证集和测试集时，将为它们和训练集分别提供不同的报表。

刻画器 为每个 Y 变量显示一个刻画器。

谱刻画器 显示一个刻画器，其中所有响应变量都显示在图中的第一个单元中。该刻画器对于同时可视化 X 变量的变化对 Y 变量的影响很有用。

保存列 包含用于保存各种公式和结果的选项。请参见第 127 页的“保存列”。

删除拟合 从主平台报表中删除模型报表。

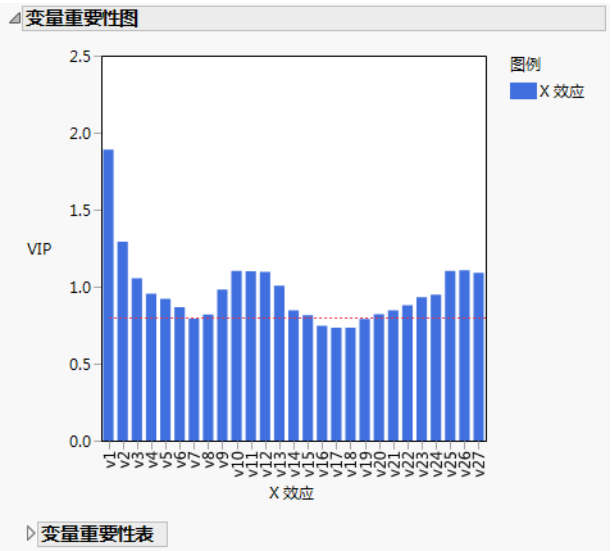
使用 VIP 生成模型 打开并填充一个启动窗口，其中相应的响应输入为 *Y*，其 VIP 超过指定阈值的变量输入为 *X*。执行与“中心化和统一尺度数据的‘系数 - VIP’”报表中的按钮相同的功能。请参见第 126 页的“系数 -VIP 图”。

已保存 X 得分的模型驱动的多元控制图 保存每个 X 得分的公式，并启动“模型驱动多元控制图”(MDMCC) 启动窗口。在 MDMCC 启动窗口中，得分公式作为过程列分配。在点击“确定”之前，可以添加或删除过程、添加时间 ID 或设置历史数据的结束位置。请参见《质量和过程方法》。

变量重要性图

“变量重要性图”对每个 X 变量的 VIP 值绘图。“变量重要性表”显示 VIP 得分。VIP 得分用于衡量变量在 X 和 Y 建模中的重要性。若变量有小的系数和小的 VIP，则它是要从模型中删除的候选项 (Wold 1994)。值为 0.8 通常视为小的 VIP (Eriksson et al. 2006)，在图上的 0.8 处绘制了一条红色虚线。

图 6.13 变量重要性图



系数 -VIP 图

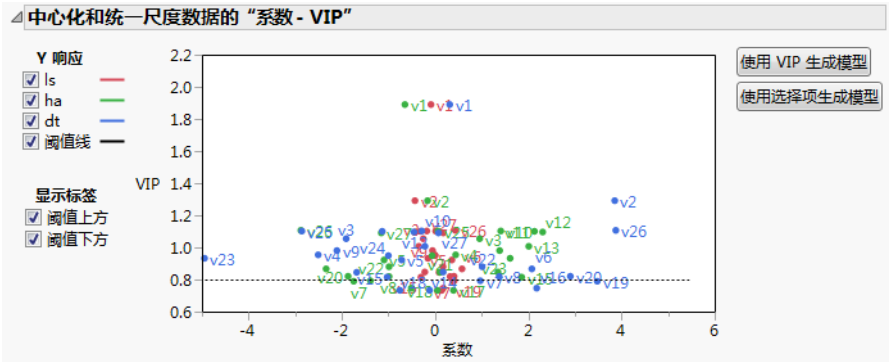
图右侧的两个选项帮助减少变量和生成模型：

使用 VIP 生成模型 打开并填充一个启动窗口，其中相应的响应输入为 *Y*，其 VIP 超过指定阈值的变量输入为 *X*。

使用选择项生成模型 允许您在图中直接选择 *X* 并将 *Y* 和选择的 *X* 输入启动窗口。

要基于当前的列选择使用另一平台，请打开所需的平台。请注意，在启动窗口中保留了选择内容。点击角色按钮，将填充所选的列。

图 6.14 中心化和统一尺度数据的 “系数 -VIP” 图



保存列

保存预测公式 对于每个 Y 变量，将名为**预测公式 : < 响应 >** 的列保存到数据表，该列包含预测公式。

将预测保存为 X 得分公式 对于每个 Y 变量，将名为**预测公式 : < 响应 >** 的列保存到数据表，该列包含根据 X 得分计算的预测公式。

保存预测公式的标准误差 对于每个 Y 变量，将名为**预测值标准误差 : < 响应 >** 的列保存到数据表，该列包含预测均值的标准误差。请参见第 132 页的 “预测的标准误差和置信限”。

保存均值置信限公式 对于每个 Y 变量，将名为 **95% 均值下限 : < 响应 >** 和 **95% 均值上限 : < 响应 >** 的两列保存到数据表。这些列包含响应均值的 95% 置信限。请参见第 132 页的 “预测的标准误差和置信限”。

保存单值置信限公式 对于每个 Y 变量，将名为 **95% 单值下限 : < 响应 >** 和 **95% 单值上限 : < 响应 >** 的两列保存到数据表。这些列包含单值的 95% 预测限。请参见第 132 页的 “预测的标准误差和置信限”。

保存得分公式 将两组列保存至数据表：

- 名为 **X 得分 <N> 公式**的列，它包含每个 X 得分的公式。
- 名为 **Y 得分 <N> 公式**的列，它包含每个 Y 得分的公式。

请参见第 130 页的 “偏最小二乘”。

保存 Y 预测值 将 Y 变量的预测值保存到数据表中的列。

保存 Y 残差 将 Y 变量的残差值保存到数据表中的列。

保存 X 预测值 将 X 变量的预测值保存到数据表中的列。

保存 X 残差 将 X 变量的残差值保存到数据表中的列。

保存针对 X 效应解释的变异的百分比 将所有提取因子上针对每个 X 变量所解释的变异百分比保存到新数据表。

保存针对 Y 响应解释的变异的百分比 将所有提取因子上针对每个 Y 变量所解释的变异百分比保存到新数据表。

保存得分 将每个提取因子的 X 和 Y 得分保存到数据表。

保存载荷 将 X 和 Y 载荷保存到新数据表。

保存标准化得分 将在构造“载荷相关图”中使用的 X 和 Y 标准化得分保存到数据表。有关公式的信息，请参见第 133 页的 “标准化得分和载荷”。

保存标准化载荷 将在构造“载荷相关图”中使用的 X 和 Y 标准化载荷保存到新数据表。有关公式的信息，请参见第 133 页的 “标准化得分和载荷”。

保存 T 方 将包含 T^2 公式的列保存到数据表中。该列中的值还用在 T 方图中。

保存距离 将“到 X 模型的距离”(DModX) 和“到 Y 模型的距离”(DModY) 值保存到数据表。这些是在距离图中使用的值。

保存 X 权重 将所有提取因子上针对每个 X 变量的权重保存到新数据表。

JMP PRO **保存验证** 将描述如何在验证中使用每个观测的新列保存到数据表。对于 “保留”验证，该列标识行是用于训练还是验证。对于 “K 重”验证，该列标识给行分配的子组编号。

JMP PRO **保存补缺** 若选择 “补缺缺失数据”，将打开一个新数据表，其中包含指定为 X 和 Y 的数据表列，缺失值用补缺值替代。不显示用于多项式项的列。若指定一个验证列，则还包含该验证列。

JMP PRO **发布预测公式** 创建预测公式并将它们保存为 “公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见《预测和专业建模》。

JMP PRO **发布得分公式** 创建 X 和 Y 得分公式并在 “公式存储库”平台中将它们保存为公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

JMP PRO “偏最小二乘”的其他示例

在本例中，您将使用“拟合模型”中的“偏最小二乘”特质对分类响应和大量变量进行分析。通过“拟合模型”使用“偏最小二乘”还支持您使用“K 重交叉验证”。示例数据集包括从 165 名男性采集的血清样本。84 位男性患有前列腺癌，81 位男性没有。分析的目标是根据血清样本确定哪些男性患有前列腺癌。

1. 选择**帮助** > **样本数据库**，然后打开 Prostate Cancer.jmp。
2. 选择**分析** > **拟合模型**。
3. 选择**状态**并点击 Y。
4. 选择**蛋白质**列组并点击**添加**。
5. 从 “特质”列表中，选择**偏最小二乘**。
6. 点击**运行**。

7. （可选）点击 “偏最小二乘”红色小三角并选择**设置随机种子**。
8. （可选）在 “指定随机种子”旁边，输入 1234。

   通过指定随机种子，您可以重现本例中显示的结果。
9. （可选。）点击**确定**。
10. 在 “模型启动”中，点击**执行**。

图 6.15 “偏最小二乘”报表

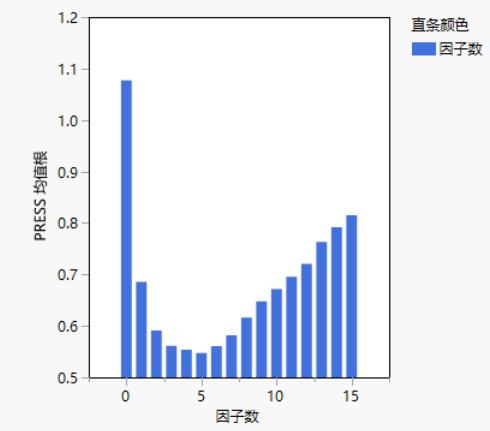
模型比较汇总

方法	SVD	行数	因子数	针对累积 X 解释的变异的百分比	针对累积 Y 解释的变异的百分比	VIP 数目 > 0.8
NIPALS	快速	165	5	52.886468	85.671251	626

K 重交叉验证，其中 K=7，方法=NIPALS，使用快速 SVD

因子数	PRESS 均值根	van der Voet T²	概率 > van der Voet T²	Q²	累积 Q²	R²X	累积 R²X	R²Y	累积 R²Y
0	1.076926	130.46053	<.0001*	-0.000625	-0.000625	0.000000	0.000000	0.000000	0.000000
1	0.685794	16.150335	0.0010*	0.593265	0.593265	0.342651	0.342651	0.608907	0.608907
2	0.591463	2.927629	0.0790	0.696945	0.876737	0.077029	0.419680	0.132190	0.741097
3	0.561730	0.480087	0.4740	0.726683	0.966310	0.039013	0.458693	0.066929	0.808026
4	0.553949	0.278455	0.5910	0.734125	0.991043	0.043112	0.501805	0.034490	0.842515
5	0.547744	0.000000	1.0000	0.740188	0.997673	0.055223	0.557027	0.017535	0.860051
6	0.560870	1.357190	0.2350	0.728045	0.999367	0.048092	0.605120	0.015012	0.875063
7	0.581726	3.490817	0.0600	0.707162	0.999815	0.034371	0.639491	0.012864	0.887927
8	0.616081	7.673681	0.0010*	0.671501	0.999939	0.023573	0.663064	0.013162	0.901089
9	0.647871	12.869719	<.0001*	0.636867	0.999978	0.020427	0.683491	0.011323	0.912412
10	0.671625	14.823639	<.0001*	0.610463	0.999991	0.015562	0.699053	0.010699	0.923112
11	0.695867	17.385858	<.0001*	0.581816	0.999996	0.019040	0.718092	0.007945	0.931057
12	0.720865	19.399541	<.0001*	0.551351	0.999998	0.021476	0.739568	0.007050	0.938107
13	0.763351	22.554517	<.0001*	0.496903	0.999999	0.011706	0.751274	0.008857	0.946963
14	0.791689	26.405173	<.0001*	0.458986	1.000000	0.011510	0.762784	0.006236	0.953199
15	0.815424	28.291824	<.0001*	0.426187	1.000000	0.011048	0.773832	0.005424	0.958623

PRESS 均值根图



注意: 最小 PRESS 均值根是 0.54774，且最低因子数是 5。

“PRESS 均值根图”显示当因子数为 5 时， PRESS 均值根的值最小。这在 “PRESS 均值根图”下的注释中有说明。生成名为**带 5 个因子的 “NIPALS”拟合，使用快速 SVD** 的报表。

“偏最小二乘”平台的统计详细信息

本节提供有关“偏最小二乘”平台中使用的一些方法的详细信息。请参见 Hoskuldsson (1988)、Garthwaite (1994) 或 Cox and Gaudard (2013)。

- “偏最小二乘”
- “van der Voet $T^{2''}$”
- “T^2 图”
- “X 得分散点图矩阵的置信椭圆”
- “预测的标准误差和置信限”
- “标准化得分和载荷”
- “PLS 判别分析 (PLS-DA)”

偏最小二乘

偏最小二乘基于解释变量 (*X*) 的线性组合（称为因子）来拟合线性模型。这些因子是通过将 *X* 与一个或多个响应 (*Y*) 之间的协方差最大化得到的。这样，PLS 可利用 *X* 和 *Y* 之间的相关性来揭示底层的潜在结构。这些因子实现了解释响应变异和预测变量变异的组合目标。当您的 X 变量数比观测数多或 X 变量高度相关时，偏最小二乘特别有用。

NIPALS

NIPALS 方法一次提取一个因子。用 $\mathbf{X} = \mathbf{X}_0$ 表示预测变量的中心化和统一尺度的矩阵，用 $\mathbf{Y} = \mathbf{Y}_0$ 表示响应值的中心化和统一尺度的矩阵。PLS 方法从预测变量的一个线性组合 $\mathbf{t} = \mathbf{X}_0\mathbf{w}$ 开始，其中 **t** 称为**得分向量**， **w** 是相关的**权重向量**。 PLS 方法通过 **t** 的回归来预测 $\mathbf{X}_0$ 和 $\mathbf{Y}_0$：

$\hat{\mathbf{X}}_0 = \mathbf{t}\mathbf{p}'$，其中 $\mathbf{p}' = (\mathbf{t}'\mathbf{t})^{-1}\mathbf{t}'\mathbf{X}_0$

$\hat{\mathbf{Y}}_0 = \mathbf{t}\mathbf{c}$，其中 $\mathbf{c}' = (\mathbf{t}'\mathbf{t})^{-1}\mathbf{t}'\mathbf{Y}_0$

向量 **p** 和 **c** 分别称为 *X* 和 *Y* **载荷**。

特定线性组合 $\mathbf{t} = \mathbf{X}_0\mathbf{w}$ 是在指定某些响应线性组合 $\mathbf{u} = \mathbf{Y}_0\mathbf{q}$ 的情况下具有最大协方差 $\mathbf{t}'\mathbf{u}$ 的组合。另一特性是 *X*- 和 *Y*-权重 **w** 和 **q** 与协方差矩阵 $\mathbf{X}_0'\mathbf{Y}_0$ 的第一个左奇异向量和右奇异向量成比例。或者分别等效于 $\mathbf{X}_0'\mathbf{Y}_0\mathbf{Y}_0'\mathbf{X}_0$ 和 $\mathbf{Y}_0'\mathbf{X}_0\mathbf{X}_0'\mathbf{Y}_0$ 的第一个特征向量。

这说明了如何提取第一个 PLS 因子。通过将 $\mathbf{X}_0$ 和 $\mathbf{Y}_0$ 替换为第一个因子中的 *X* 和 *Y* 残差，以相同方式提取第二个因子：

$$\mathbf{X}_1 = \mathbf{X}_0 - \hat{\mathbf{X}}_0$$

$$\mathbf{Y}_1 = \mathbf{Y}_0 - \hat{\mathbf{Y}}_0$$

这些残差还被称为**缩小的** *X* 和 *Y* 区组。根据要提取的因子数，重复提取得分向量和缩小数据矩阵的过程。

SIMPLS

SIMPLS 算法开发的目的在于优化统计准则：它寻找在 *X* 得分是正交的前提下使 *X* 和 *Y* 的线性组合之间的协方差最大化的得分向量。不同于 NIPALS，该算法缩小了矩阵 $\mathbf{X}_0$ 和 $\mathbf{Y}_0$，SIMPLS 缩小的是叉积矩阵 $\mathbf{X}_0'\mathbf{Y}_0$。

在单个 *Y* 变量的情况下，这两个算法是等效的。但是，对于多元 *Y*，模型有所不同。SIMPLS 是 De Jong (1993) 提出的。

van der Voet T^2

van der Voet T^2 检验帮助确定具有指定提取因子数的模型是否与建议的最优模型显著不同。该检验是基于以下原假设的随机化检验：两个模型的残差平方具有相同的分布。直观上，可以将该原假设表述为：两个模型具有相同的预测能力。

要获得“交叉验证”报表中给出的 van der Voet T^2 统计量，对每个验证集执行下面的计算。在单个验证集的情况下，结果是报告的值。若使用“留一法”和“K 重”验证，则对每个验证集的结果求平均值。

用 $R_{i,jk}$ 表示模型的响应 *k* 的第 *j* 个预测残差，该模型具有 *i* 个提取因子。用 $R_{opt,jk}$ 表示模型的相应量，该模型基于建议的最佳因子数 *opt*。检验统计量基于以下差值：

$$D_{i,jk} = R^2_{i,jk} - R^2_{opt,jk}$$

假定有 *K* 个响应。考虑以下符号：

$$\mathbf{d}_{i,j} = (D_{i,j1}, D_{i,j2}, \ldots, D_{i,jK})'$$

$$\mathbf{d}_{i,.} = \sum_j \mathbf{d}_{i,j}$$

$$\mathbf{S_i} = \sum_j \mathbf{d}_{i,j}\mathbf{d}_{i,j}'$$

按以下方式定义 *i* 个提取因子的 van der Voet 统计量：

$$C_i = \mathbf{d}_{i,.}'\mathbf{S}_\mathbf{i}^{-1}\mathbf{d}_{i,.}$$

通过比较 C_i 与将 $R^2_{i,jk}$ 和 $R^2_{opt,jk}$ 随机交换所得值的分布，来获得显著性水平。模拟了这类值的 Monte Carlo 样本，并将显著性水平近似为模拟临界值大于等于 C_i 的比例。

T^2 图

按以下方式计算第 i 个观测的 T^2 值：

$$T_i^2 = (n-1)\sum_{j=1}^{p}\left(t_{ij}^2 / \sum_{k=1}^{n} t_{kj}^2\right)$$

其中 t_{ij} = 第 i 行和第 j 个提取因子的 X 得分，p = 提取的因子数，n = 用于训练模型的观测数。若未使用验证，则 n = 总观测数。

按以下方式计算 T^2 图的控制限：

$((n\text{-}1)^2/n)$*BetaQuantile(0.95, p/2, $(n\text{-}p\text{-}1)/2$)

其中 p = 提取的因子数，n = 用于训练模型的观测数。若未使用验证，则 n = 总观测数。请参见 Tracy et al. (1992)。

X 得分散点图矩阵的置信椭圆

“得分散点图矩阵”选项将 95% 置信椭圆添加到 X 得分散点图。X 得分是不相关的，因为 NIPALS 和 SIMPLE 算法都生成正交得分向量。该椭圆假定每对 X 得分服从具有零相关性的二元正态分布。

考虑垂直轴上得分 i 和水平轴上得分 j 的散点图。椭圆的上、下、左、右端点的坐标定义如下：

- 上、下端点为 +/-sqrt(var(score i)*z)
- 左、右端点为 +/-sqrt(var(score j)*z)

其中 z = $((n\text{-}1)*(n\text{-}1)/n)$*BetaQuantile(0.95, 1, $(n\text{-}3)/2$)。有关 z 值的背景信息，请参见 Tracy et al. (1992)。

预测的标准误差和置信限

用 **X** 表示预测变量的矩阵，用 **Y** 表示响应值的矩阵，该矩阵可能基于您在启动窗口中的选择进行了中心化和统一尺度。假定 **Y** 的成分是独立的，且服从具有公共方差 σ^2 的正态分布。

Hoskuldsson (1988) 注意到基于得分构建的 **Y** 的 PLS 模型在形式上类似于多元线性回归模型。他使用这个相似性推导出预测值方差的近似公式。另见 Umetrics (1995)。但是，Denham (1997) 指出 PLS 预测的任何值是 Y 的非线性函数。他提出采用 Bootstrap 和交叉验证方法来获取预测区间。PLS 平台使用 Umetrics (1995) 中所述的基于正态性的方法。

用 **T** 表示其列为得分的矩阵，考虑 **X** 的新观测 $\mathbf{x}_0$。通过 **T** 对 **Y** 做回归来得到 **Y** 的预测模型。用 $\mathbf{t}_0$ 表示与 $\mathbf{x}_0$ 关联的得分向量。

用 a 表示因子数。将 s^2 定义为残差平方和除以自由度，若数据中心化，则自由度 $df = n - a - 1$，若数据未中心化，则自由度 $df = n - a$。s^2 的值是 σ^2 的估计值。

预测公式的标准误差

按以下方式估计 $\mathbf{x}_0$ 处的预测均值的标准误差：

$$SE(\bar{Y}_{x_0}) = s\sqrt{\left(\frac{1}{n} + \mathbf{t}_0(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_0'\right)}$$

均值置信限公式

用 $t_{0.975,\,df}$ 表示 t 分布的 0.975 分位数，若数据已中心化，则该 t 分布的自由度 $df = n - a - 1$，若数据未中心化，则自由度 $df = n - a$。

按以下方式计算均值的 95% 置信区间：

$$\bar{Y}_{x_0} \pm t_{0.975,\,df} SE(\bar{Y}_{x_0})$$

单值置信限公式

按以下方式估计 $\mathbf{x}_0$ 处的单个响应预测值的标准误差：

$$SE(\hat{Y}_{x_0}) = s\sqrt{\left(\frac{1}{n} + 1 + \mathbf{t}_0(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_0'\right)}$$

用 $t_{0.975,\,df}$ 表示 t 分布的 0.975 分位数，若数据已中心化，则该 t 分布的自由度 $df = n - a - 1$，若数据未中心化，则自由度 $df = n - a$。

按以下方式计算单个响应的 95% 预测区间：

$$\bar{Y}_{x_0} \pm t_{0.975,\,df} SE(\hat{Y}_{x_0})$$

标准化得分和载荷

考虑以下符号：

- n_{tr} 是训练集中的观测数
- m 是 X 中的效应数
- k 是 Y 中的响应数
- $VarX_i$ 是第 i 个因子解释的 X 的变异百分比
- $VarY_i$ 是第 i 个因子解释的 Y 的变异百分比
- $\mathbf{XScore}_i$ 是第 i 个因子的 X 得分的向量
- $\mathbf{YScore}_i$ 是第 i 个因子的 Y 得分的向量
- $\mathbf{XLoad}_i$ 是第 i 个因子的 X 载荷的向量
- $\mathbf{YLoad}_i$ 是第 i 个因子的 Y 载荷的向量

标准化得分

按以下方式定义第 i 个标准化 X 得分的向量：

$$\frac{\mathbf{XScore}_i}{(n_{tr}-1)\sqrt{mVarX_i/n_{tr}}}$$

按以下方式定义第 i 个标准化 Y 得分的向量：

$$\frac{\mathbf{YScore}_i}{(n_{tr}-1)\sqrt{kVarY_i/n_{tr}}}$$

标准化载荷

按以下方式定义第 i 个标准化 X 载荷的向量：

$$\mathbf{XLoad}_i\sqrt{mVarX_i}$$

按以下方式定义第 i 个标准化 Y 载荷的向量：

$$\mathbf{YLoad}_i\sqrt{kVarY_i}$$

PLS 判别分析 (PLS-DA)

在启动窗口中将某个分类变量输入为 Y 时，使用指示符编码对其编码。若有 k 个水平，则用一个指示符变量来表示每个水平，对于属于该水平的行用值 1 表示，不属于该水平的行用 0 表示。得到的 k 个指示符变量被视为连续变量，PLS 分析按处理连续 Y 的方式处理这些指示符变量。

第 7 章

多重对应分析

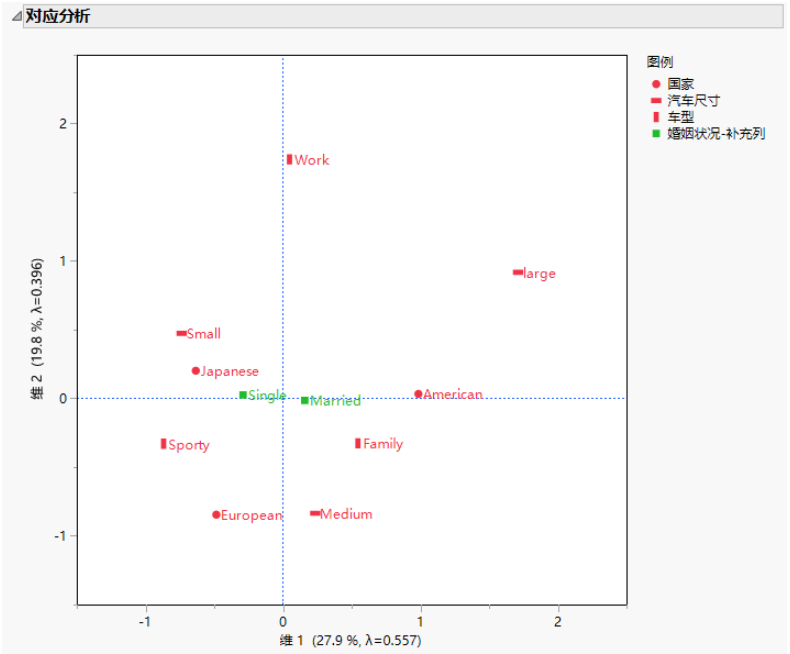
识别分类变量各水平之间的关联

“多重对应分析”(MCA) 针对多个分类变量，试图查找这些变量各水平之间的关联。MCA 将对应分析从两个变量扩展到多个变量的情况。您可以认为它就如同针对定量变量的主成分分析。与其他多元方法类似，多重对应分析它是一种降维方法，将原始的多维数据表示为二维或三维空间中的点。

社会科学领域经常应用多重对应分析。可将其用在调查分析中，找出测试对象对不同问题的态度一致性。还可以在消费者研究中使用该方法来确定产品的潜在市场。

有关多重对应分析的详细信息，请参见 LeRoux and Rouanet (2010)。

图 7.1 多重对应分析



目录

多重对应分析的示例 137
启动“多重对应分析”平台 140
“多重对应分析”报表 141
“多重对应分析”平台选项 142
对应分析选项 143
显示图 144
显示详细信息 144
显示调整惯量 145
显示坐标 145
显示汇总统计量 145
显示对惯量的部分贡献 146
显示平方余弦 146
Cochran Q 检验 147
交叉表 147
“多重对应分析”平台的更多示例 147
使用补充变量的示例 148
使用补充 ID 的示例 149
Cochran Q 检验示例 149
“多重对应分析”平台的统计详细信息 151
“详细信息”报表 151
调整惯量 152
汇总统计量 152
对惯量的部分贡献 152
Cochran Q 统计量 153

多重对应分析的示例

本例使用从 55 名 JMP 员工收集的关于他们在以下领域的偏好（或品味）的数据：

- 偏好的电视节目（8 类）：新闻、喜剧、警察、自然、体育、电影、戏剧或肥皂剧。
- 偏好的影片（8 类）：动作片、喜剧、古装剧、纪录片、恐怖片、音乐剧、言情片或科幻片。
- 偏好的艺术类型（7 类）：表演、风景、文艺复兴、静物、肖像、现代或印象。
- 偏好的外出就餐场所（6 类）：炸鱼薯条、酒吧、印度餐厅、意大利餐厅、法国餐厅或牛排馆。

这些数据还包括调查的个人的人口统计数据。您想探究针对这四个问题的 55 个回答之间的关联。这有助于了解不同的员工群体。

注意：偏好的测度基于 LeRoux and Rouanet (2010) 中的问题。

1. 选择**帮助 > 样本数据库**，然后打开 Employee Taste.jmp。
2. 选择**分析 > 多元方法 > 多重对应分析**。
3. 选择**电视节目**、**电影**、**艺术和餐厅**，然后点击 **Y，响应**。

   在 MCA 中，与主成分分析一样，因子通常被认为是响应，而不是一些是响应，另一些是解释变量。
4. 点击**确定**。

图 7.2 已完成的 “多重对应分析” 启动窗口

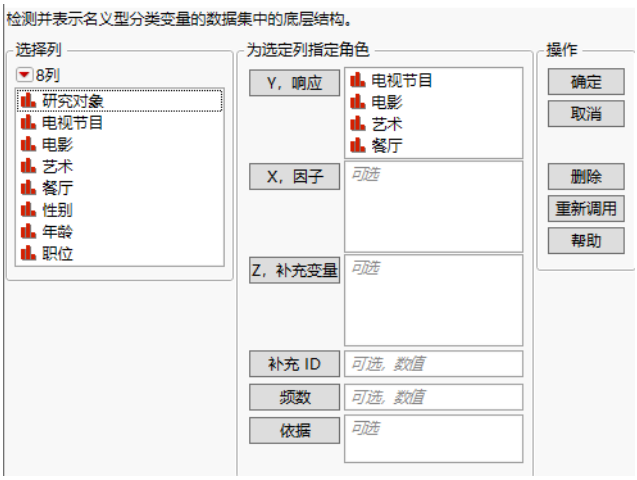
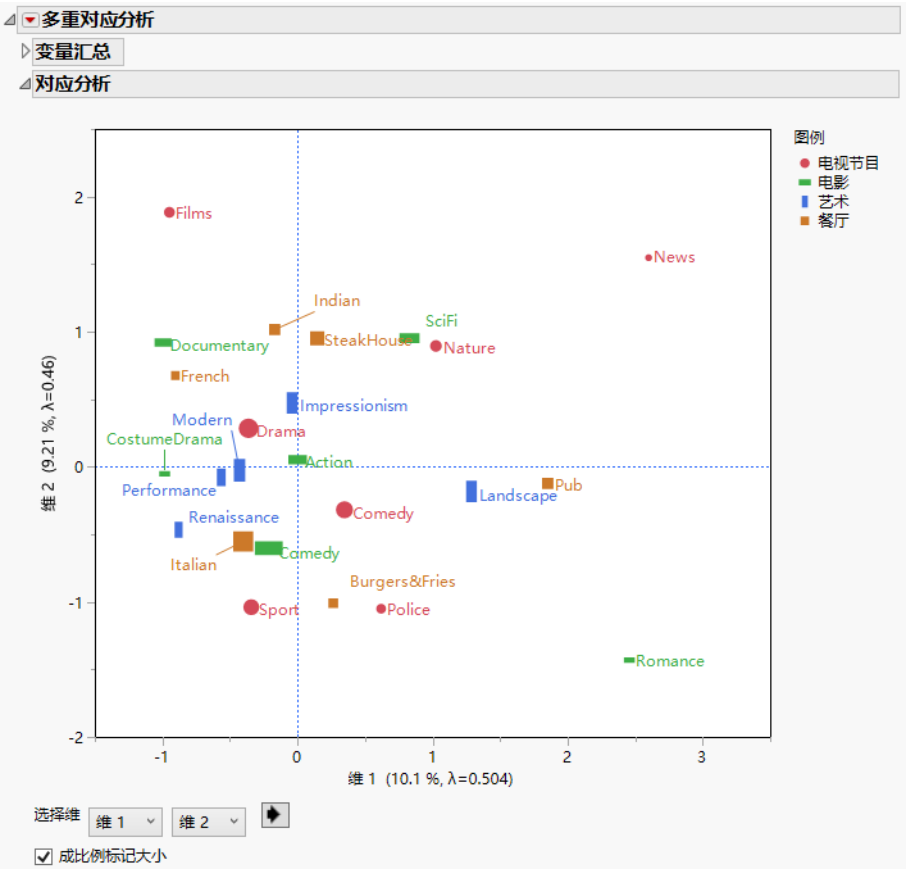


图 7.3 最初的 “多重对应分析”报表的一部分



提示：点击并拖动重叠标签将其重新排列。

5. 在该图下方，选择**成比例标记大小**。

   标记大小指示该类别中响应的相对比例。

   “对应分析”报表显示了四个变量的类别在前两个主轴或维上的投影。使用控件更改标绘的维。点与点之间的距离代表了员工响应模式之间的差异。

   从该图中，您可以用您对主题的了解来解释以下发现：

   – 聚集在“Burgers and Fries”餐厅偏好附近，显示有“Sport”和“Police”电视节目偏好的聚类。该聚类可称为“大众文化”群体。

   – 从“维 2”来看，个人品味从大众文化群体转向可以归类为更“复杂”的品味 — 那些喜欢纪录片、偏好电视剧和特色餐厅（如法国菜和印度菜）的人群。

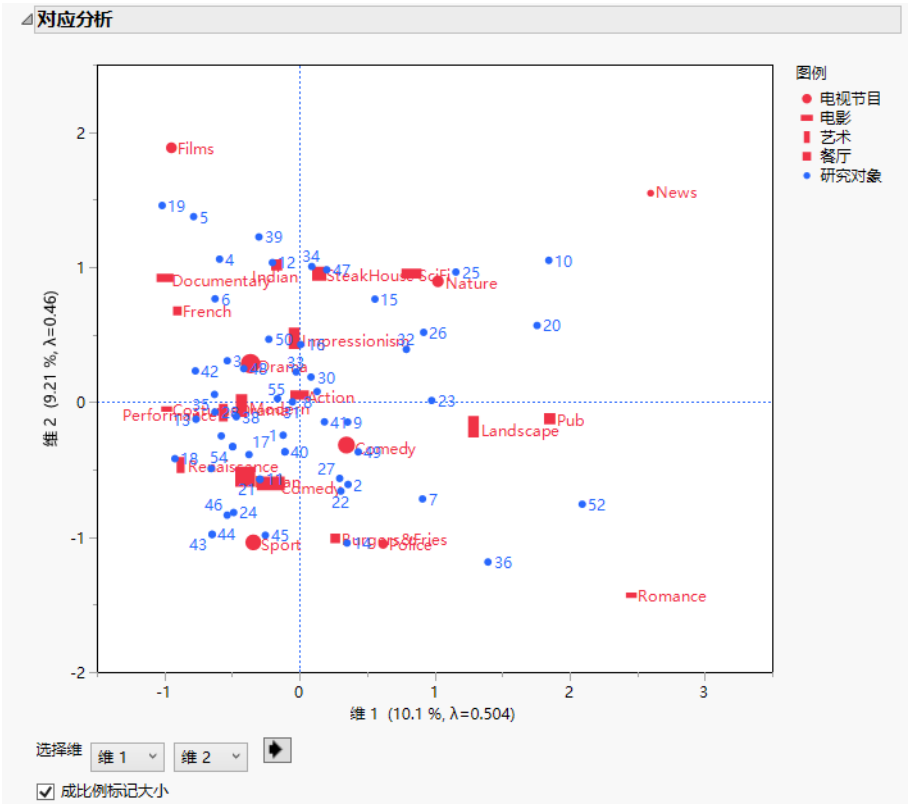
   要获得个人得分，请将对象添加为 **X，因子**：

6. 打开 “变量汇总”分级显示项。

“变量汇总”面板允许您修改分析而不必重新启动该平台。它还提供已完成分析的简明视图。

7. 选择**研究对象**并点击**添加 X**。分析自动更新。

图 7.4 包含研究对象的 “多重对应分析”报表

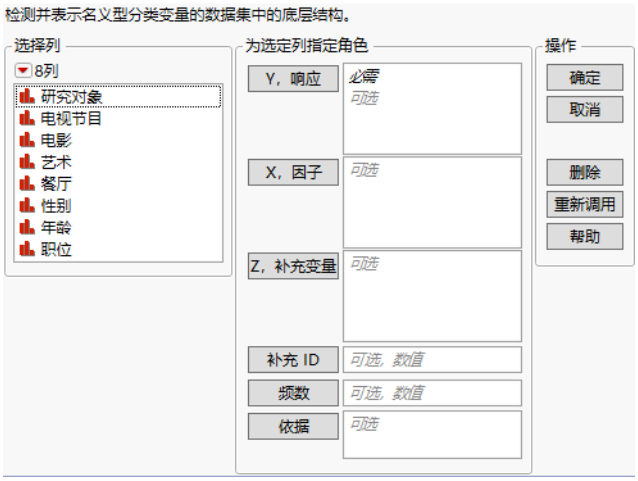


包括研究对象后，则可在图中左下象限突出显示有着相似品味的员工的聚类。这些员工属于我们认定的大众文化领域。

启动“多重对应分析”平台

通过选择**分析** > **多元方法** > **多重对应分析**启动“多重对应分析”平台。

图 7.5 “多重对应分析”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，响应 指定要分析的分类列。在 MCA 中，您通常关注变量之间的关联，但是没有明确的“解释”和“响应”变量。

X，因子 分配一个或多个要用作因子或解释变量的分类列。

注意：将某个对象 ID 列用作单个 X 来获取个人得分。

Z，补充变量 指定要用作补充变量的列。您对这些变量和响应的关系感兴趣，但进行计算时并不把它们包括在内。补充变量用于改进数据解释。

补充 ID 指定列用于标识不包含在分析计算中的行。“补充 ID”列通常具有值 1 或 0。与 ID 0 关联的行被视为补充行。若补充行中的 X 或 Y 变量水平在非补充行中并未出现，则忽略“补充 ID”列。

注意：一次只能指定“补充 ID”或“Z，补充变量”角色中的一种。

频数 指定一个频数变量。该选项适用于汇总数据。

依据 为“依据”变量的每个水平生成单独报表。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

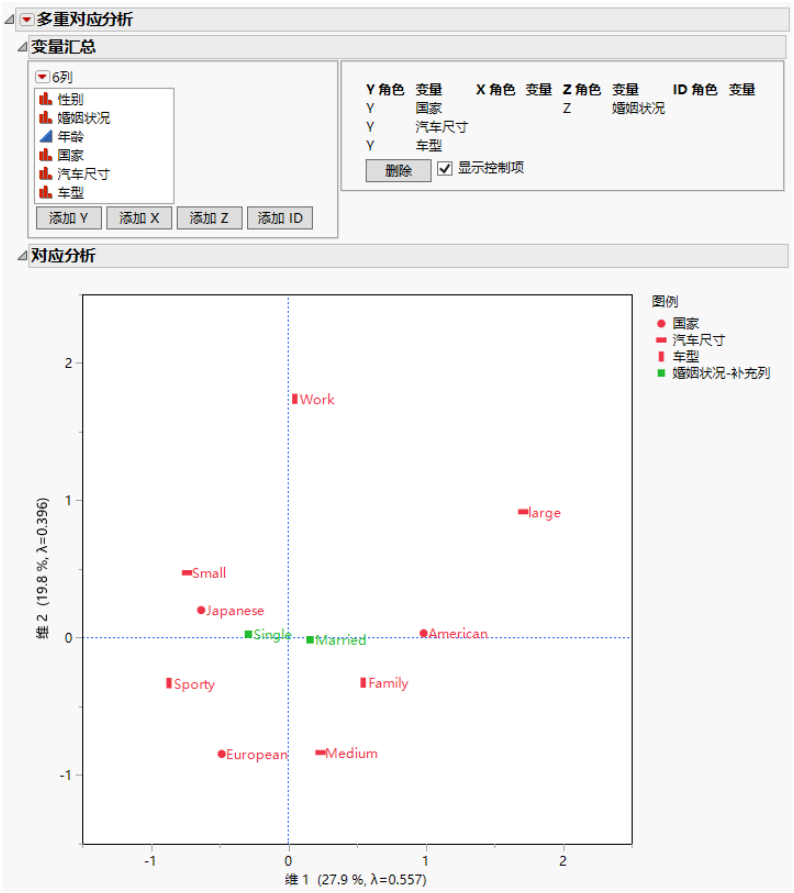
注意：“多重对应分析”平台处理缺失值的方式不同于其他许多 JMP 平台。该分析使用行中的所有非缺失单元格对。它不会从计算中删除整个行。

“多重对应分析”报表

初始“多重对应分析”报表显示变量汇总、对应分析图和按重要性排列的数据维度的详细信息。从类别或个体的对应图中，您可以发现数据中存在的关联。这些细节提供了图中两个维是否足够解释数据内关系的信息。

交互式面板中的“变量汇总”允许您修改分析而不必重新启动该平台。该面板显示分析中使用的列和角色。若您选中“显示控制项”复选框，包含数据表中各列的列表将显示在左侧。您可以通过选择一列并点击“添加 Y”、“添加 X”、“添加 Z”或“添加 ID”来更改分析中的列。或者，您可以将该列拖到变量汇总表中的标题处。如此您不必返回到启动窗口就可以修改分析设置。

图 7.6 选中了 “显示控制项”的 “多重对应分析”报表



“多重对应分析”平台选项

使用 “多重对应分析”红色小三角菜单选项，您可以根据自己的需要定制报表。根据您指定的分析类型，有以下报表可供您选择。

交叉表 为选定的变量角色适当显示或隐藏 Burt 表或列联表。请参见第 147 页的 “交叉表”。

对应分析 显示或隐藏对应分析报表。这些报表给出图、详细信息、坐标和汇总统计量。请参见第 147 页的 ““多重对应分析”平台的更多示例”。

补充行的交叉表 （在指定一个或多个补充 Z 和 X 变量或在指定 ID 时可用。）依据以下规则显示或隐藏 Burt 或列联表：

- 当指定一个或多个 X 和 Z 变量时，显示或隐藏“响应变量 - X 变量”列联表。
- 若指定了 ID 但未指定 X 变量，则显示或隐藏补充观测的 Burt 表。补充观测是 ID 变量等于 0 的那些观测。

– 若指定了 ID 并且指定了一个或多个 X 变量，则显示或隐藏补充观测的列联表。补充观测是 ID 变量等于 0 的那些观测。

补充列的交叉表 （当指定一个或多个补充 Z 变量时可用。）依据以下规则显示或隐藏列联表：

– 在未指定任何 X 变量时，显示或隐藏“补充变量 - 响应变量”列联表。

– 当指定一个或多个 X 变量时，显示或隐藏“补充变量 - X 变量”列联表。

马赛克图 （仅当有一个 X 变量和一个 Y 变量时才可用。）显示或隐藏 X - Y 马赛克图。马赛克图是堆叠的条形图，其中每一段与该组的频数计数成比例。

独立性检验 （仅当有一个 X 变量和一个 Y 变量时才可用。）显示或隐藏 X 和 Y 变量之间的独立性检验。该检验有两种方法：Pearson 形式和似然比形式，两者都计算卡方统计量。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

对应分析选项

“对应分析”下提供的报表取决于您执行的多重对应分析的类型。其中有个别报表是默认输出。

显示图 将类别数据投影到提取的前两个主轴上的二维对应分析表。默认显示该图。该图使用等距尺度。

显示详细信息 提供分析的详细信息，包括奇异值、惯量、卡方统计量、百分比和累积百分比。默认显示该报表。请参见第 144 页的 “显示详细信息”。

显示调整惯量 提供 Benzecri 和 Greenacre 调整惯量的报表。请参见 Benzécri (1979) 和 Greenacre (1984)。该选项在有一个或多个 X 变量时不可用。请参见第 145 页的 “显示调整惯量”。

显示坐标 提供一个报表，针对每个类别最多包含对应分析中前三个维度的坐标值，具体视情况而定。请参见第 145 页的 “显示坐标”。

显示汇总统计量 提供一个报表，它显示分析中每个类别对应的汇总统计量、质量、量和惯量。请参见第 145 页的 “显示汇总统计量”。

显示对惯量的部分贡献 提供一个报表，它显示每个类别对最多前三个维惯量的贡献。请参见第 146 页的 “显示对惯量的部分贡献”。

显示平方余弦 提供一个报表，针对每个类别，显示最多前三个维的平方余弦。该报表包括一个条形图，其中为每个 Y 变量的每个水平显示最多前三个维中每个维的平方余弦值。请参见第 146 页的 “显示平方余弦”。

Cochran Q 检验 （仅在所有 Y 变量都仅包含相同的一组两水平并且 X 变量每行都具有唯一值的情况下，该选项才可用。）提供 Cochran *Q* 统计量，该统计量检验特定响应的边缘概率在各个 Y 变量中是否不变。Cochran *Q* 统计量是针对两个以上响应变量的广义的 McNemar 统计量。请参见 Agresti (2013)。请参见第 147 页的 “Cochran Q 检验”。

三维对应分析 提供将空间中 Y、X 和 Z 变量的类别数据投影到前三个主轴上的三维对应分析图。若少于三个维度，则该选项不可用。

保存坐标 将主坐标保存到一个或多个 JMP 数据表。列坐标、行坐标、补充列坐标和补充行坐标保存到单独的 JMP 数据表。您可以选择要保存的列数。

保存坐标公式 将多个维度的主坐标的公式列保存到数据表中。每个观测的值是以每个维的奇异值统一尺度的 Y 变量坐标的平均值。您可以选择要保存的列数。

显示图

该图显示类别或个体在前两个主轴所构成的平面上的投影中的一个投影。该图使用等距尺度。您可以使用图下方的“选择维”控件来切换图中显示的维。第一个控件定义图的水平轴，第二个控件定义图的垂直轴。点击箭头按钮可以在图中显示的维之间循环切换。使用**成比例标记大小**复选框指定图中各点的大小是否应与对应每个点的观测计数成比例。

注意：选择对应分析图中的某个点也会选择报表窗口内其他表中的对应行，但并不选择该数据表中的行。要选择图中与特定变量关联的所有点，请在图例中选择变量的名称。

显示详细信息

显示奇异值表。

奇异值 显示列联表或 Burt 表的奇异值分解中的奇异值。有关该公式，请参见第 151 页的“‘详细信息”报表”。

惯量 列出奇异值的平方，反映了典型维中解释的相对变异。

卡方 列出针对 Burt 表或列联表计算的总卡方值拆分到当前维的部分。

百分比 各维惯量占总惯量的比例。

累积百分比 显示惯量的累积比例。若前两个奇异值捕获绝大部分惯量，则二维对应分析图足以显示表中的关系。

显示调整惯量

MCA 中 Burt 表的主惯量为特征值。这些惯量存在的问题是对拟合好坏的判定过于悲观。Benzécri 提出了一种惯量调整方法，而 Greenacre 认为 Benzécri 调整过高估计了拟合质量，因此他提出了新的调整方法来替代。我们计算了两种调整值供您参考。请参见第 152 页的 “调整惯量”。

惯量 列出奇异值的平方，反映了典型维中解释的相对变异。

调整惯量 列出了根据 Benzécri 或 Greenacre 调整的调整惯量。

百分比 调整的惯量占总惯量的比例。

累积百分比 显示调整惯量的累积比例。若前两个奇异值捕获绝大部分惯量，则二维对应分析图足以显示表中的关系。

显示坐标

显示 “列坐标”表，或 “行坐标”表和 “列坐标”表。

X 列出指定为 “X，因子”变量的列。

Y 列出指定为 “Y，响应”变量的列。

Z 列出指定为 “Z，补充变量”的列。

类别 列出 X、 Y 或 Z 变量的水平。

维 1、维 2、维 3 对于每个水平或每个响应，在相应的主轴上列出其坐标。默认情况下，表中显示最多前三个维的坐标。隐藏其他维的坐标列。要显示这些可选列，请右击某个表并从**列**子菜单中选择维列。

注意：若有指定为 “X，因子”变量的列，则 “坐标”报表会显示 X 表和 Y 表在同一个标题下。若指定了补充变量 Z，则会在 X 和 Y 坐标下列出相应的坐标（若适用）。

显示汇总统计量

显示 “列点的汇总统计量”表，或 “行点和列点的汇总统计量”表。 Y 表提供每个响应的每个水平的质量、量和惯量，这些统称为**列点**。X 表为 “X，因子”变量的每个水平提供质量、量和惯量。请参见第 152 页的 “汇总统计量”。

X 列出指定为 “X，因子”变量的列。

Y 列出指定为 “Y，响应”变量的列。

类别 列出 X 和 Y 变量的水平。

质量 （维 =2） 列出用解表示水平的质量。

量 列出响应水平的行频数除以总频数的结果。在 Burt 表中，等同于每行的 “合计百分比”。

惯量 列出响应水平所占总惯量的比例。各个水平及其响应的惯量值之和为 1。

注意：若有指定为“X，因子”变量的列，则“汇总统计量”报表会将 X 变量和 Y 变量相关的表显示在同一报表标题下。

显示对惯量的部分贡献

显示“关于列点对惯量的部分贡献”表，或“关于行和列点对惯量的部分贡献”表。还显示“关于列点对惯量的部分贡献图”。这是一个条形图，其中针对每个 Y 变量的每个水平，显示其对表中显示的每个维的部分贡献。

X 列出指定为“X，因子”变量的列。

Y 列出指定为“Y，响应”变量的列。

类别 列出 X 和 Y 变量的水平。

维 1、维 2、维 3 列出响应或因子水平对所指维的惯量的贡献。默认情况下，表中显示最多前三个维的列。其他列会隐藏起来。要显示这些可选列，请右击某个表并从**列**子菜单中选择维列。

每个响应的每个水平都对每个维的惯量做贡献。每个维中的部分贡献加总为 1。

注意：若有指定为“X，因子”变量的列，则“对惯量的部分贡献”报表会显示 X 表和 Y 表在同一个标题下。请参见第 152 页的“对惯量的部分贡献”。

显示平方余弦

显示“列点的平方余弦”表，或“行和列点的平方余弦”。还显示“列点的平方余弦图”。这是一个条形图，其中针对每个 Y 变量的每个水平，显示最多前三个所示维中每个维的平方余弦值。

X 列出指定为“X，因子”变量的列。

Y 列出指定为“Y，响应”变量的列。

类别 列出 X 和 Y 变量的水平。

维 1、维 2、维 3 列出用所指维表示水平的**质量**。默认情况下，表中显示最多前三个维的结果。其他列会隐藏起来。要显示这些可选列，请右击某个表并从**列**子菜单中选择维列。

这些值表示每个列点在相应维上的质量。平方余弦可以解释为点与维的相关性的平方。前两个维的平方余弦之和等于“汇总统计量”报表中的“质量（维 =2）”。请参见第 152 页的“汇总统计量”。

注意：若有指定为“X，因子”变量的列，则“平方余弦”报表会将 X 变量和 Y 变量相关的表显示在同一报表标题下。

Cochran Q 检验

Cochran *Q* 检验是三个或更多二值结果的匹配样本的非参数齐性检验。您可以用它来检验配对中的比例差异。Cochran *Q* 检验是用于两种结果的 McNemar 检验的扩展。

交叉表

该交叉表选项显示或隐藏 Burt 表或列联表。当您选择多个“Y，响应”列且没有选择任何“X，因子”列时，将创建 Burt 表。若您选择任何“X，因子”列，则创建传统的列联表而非 Burt 表。分级显示项节点标题反映了交叉表的结构。

Burt 表是多重对应分析的基础。它是描述所有分类变量两两交叉列联的分区对称表。对角分区是对角矩阵（变量与其自身的交叉表）。非对角分区是普通的列联表。

Burt 表或列联表的红色小三角菜单包含要在表中显示的统计量的以下选项。

计数 单元格频数、边缘总频数和总计 （总样本大小）。默认显示该选项。

合计百分比 单元格计数和边缘合计占总计的百分比。默认显示该选项。

单元格卡方值 为每个单元格计算的卡方值，公式为 $(O - E)^2 / E$。

列百分比 每个单元格计数占列合计的百分比。

行百分比 每个单元格计数占行合计的百分比。

期望值 在独立性假设下每个单元格的期望频数 (E)。它由相应行合计与列合计之积除以总计得到。

偏差 观测的单元格频数 (O) 减去期望的单元格频数 (E) 所得的值。

列累积 累积列合计。

列累积百分比 累积列百分比。

行累积 累积行合计。

行累积百分比 累积行百分比。

制成数据表 为表中显示的每个统计量创建一个数据表。

“多重对应分析”平台的更多示例

- “使用补充变量的示例”
- “使用补充 ID 的示例”
- “Cochran Q 检验示例”

使用补充变量的示例

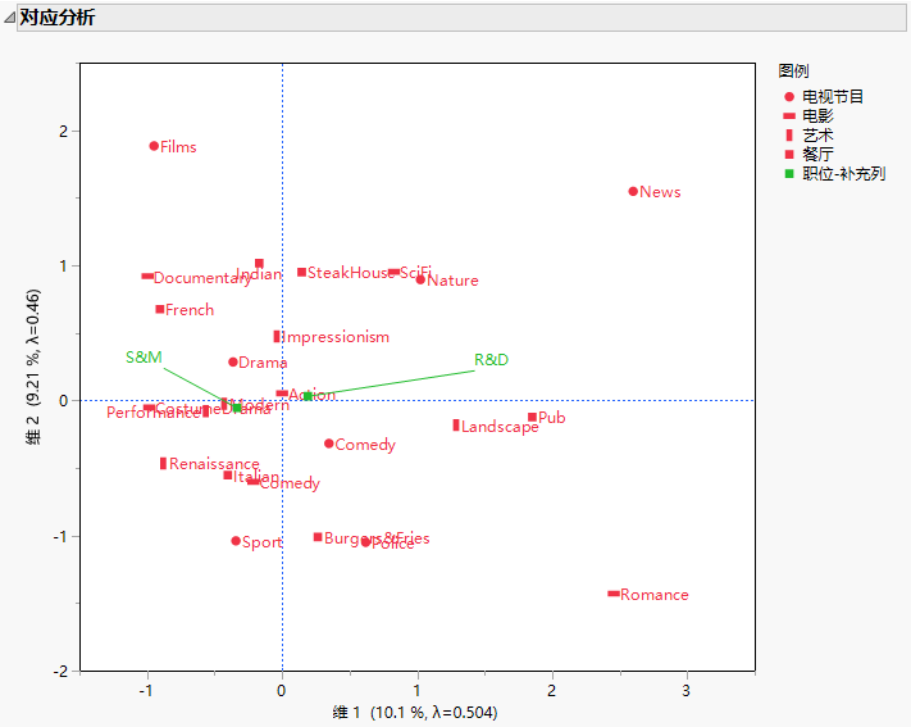
本例使用第 137 页的 “多重对应分析的示例”中所述的 JMP 员工首选项的样本数据。

您想要探索员工工作与他们的偏好之间的关系。

1. 选择**帮助 > 样本数据库**，然后打开 Employee Taste.jmp。
2. 选择**分析 > 多元方法 > 多重对应分析**。
3. 选择**电视节目**、**电影**、**艺术和餐厅**，然后点击 **Y，响应**。
4. 选择**职位**，然后点击 **Z，补充变量**。
5. 点击**确定**。

注意：由于**职位**是补充变量，所以在计算中不使用。计算完成后会标绘工作类型。

图 7.7 具有补充变量的 MCA 报表



从图中可以看出，R&D（研究和开发）和 S&M（销售和营销）员工所对应的数据点非常接近。我们认为这意味着没有由于员工工作职能的原因，其在偏好方面有很大的差异。换句话说，员工的工作角色并不是影响其偏好的显著因子。

使用补充 ID 的示例

美国人口普查可用于查看上个世纪的人口增长率。 US Regional Population.jmp 样本数据表包含 1920 到 2010 年间每个人口普查年份的美国 50 个州的人口数，50 个州按地区进行了分组。阿拉斯加州和夏威夷州被视为补充地区，因为在整个普查期间它们还没建州，而且不与美国大陆接壤。您想关注这两个州的人口增长率是否不同于美国其他州的人口增长率。

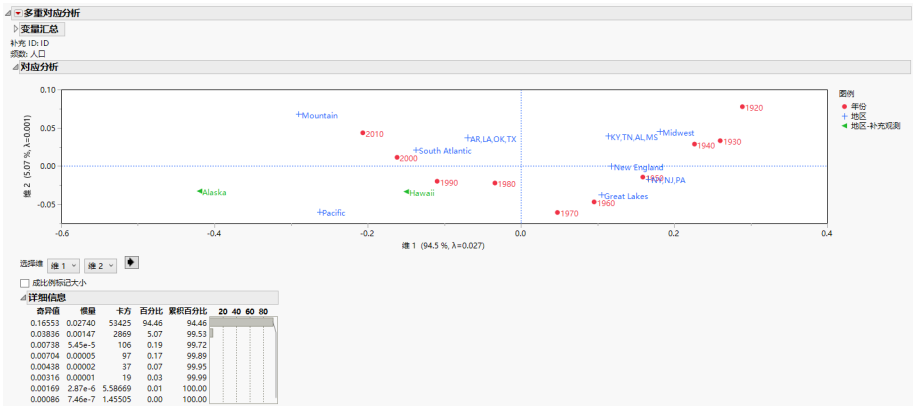
1. 选择**帮助** > **样本数据库**，然后打开 US Regional Population.jmp。
2. 选择**分析** > **多元方法** > **多重对应分析**。
3. 选择**年份**，然后点击 **Y，响应**。
4. 选择**地区**，然后点击 **X，因子**。
5. 选择 ID，然后点击**补充 ID**。
6. 选择**人口**，然后点击**频数**。
7. 点击**确定**。

“详细信息”报表显示年份和地区之间的关联几乎可以完全使用第一个维来解释。从该图可以看出第一维度上的年份顺序符合自然规律，该排序在整个对应分析过程中自然地进行，事先没有提供排序信息。该图突出显示用于标绘数据的等距尺度。

从地区的排序可以看出人口的迁移规律是从中西部到东北部、再到南部、最后到山地和西部。

上述对应分析的计算没有包含阿拉斯加州和夏威夷州的数据，但我们可以根据结果来标绘它们。它们的增长模式类似于太平洋地区各州的模式，阿拉斯加州的增长率比太平洋地区更为极端。

图 7.8 具有补充 ID 的 MCA 报表

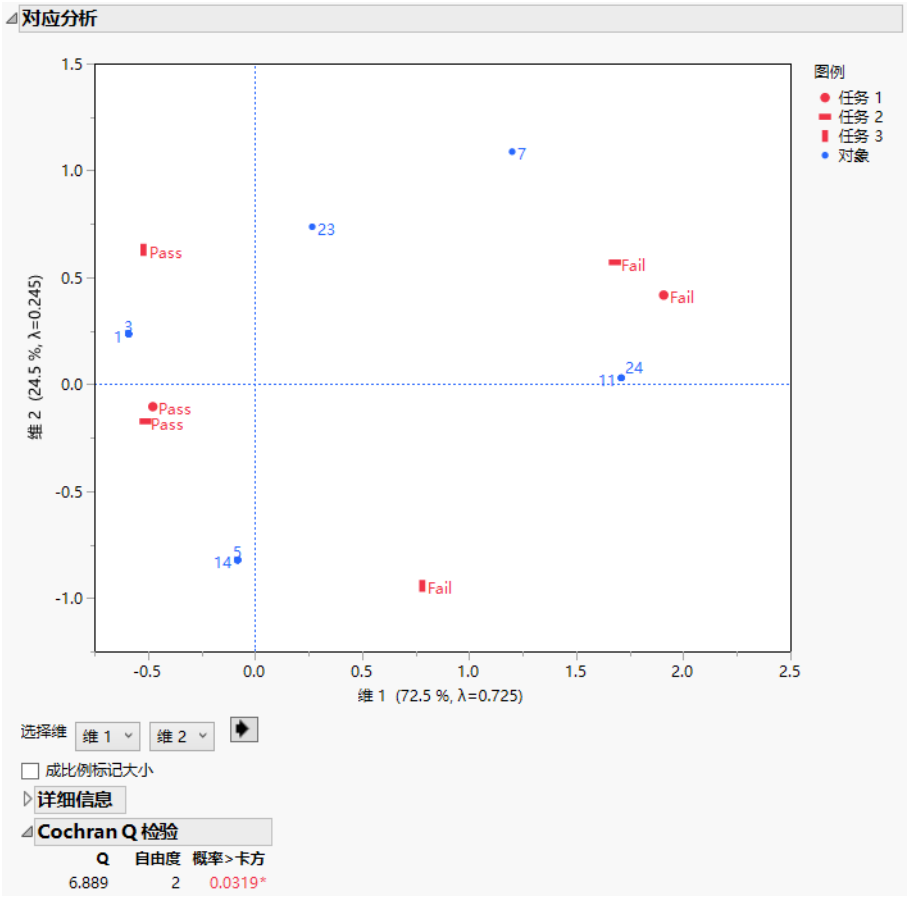


Cochran Q 检验示例

该示例使用一个模拟数据表来评估 30 个对象完成三项任务的容易程度。模拟任务 1 具有 80% 的完成率。模拟任务 2 和 3 具有与任务 1 相同的结果，但分别针对 95% 和 80% 的对象。

1. 选择**帮助** > **样本数据库**，然后打开 Cohcrans Q.jmp。
2. 选择**分析** > **多元方法** > **多重对应分析**。
3. 选择**任务 1**、**任务 2** 和**任务 3**，然后点击 **Y，响应**。
4. 选择**对象**并点击 **X，因子**。
5. 点击**确定**。
6. 点击“多重对应分析”红色小三角菜单并选择**对应分析** > **Cochran Q 检验**。

图 7.9 Cochran *Q* 检验



该对应图显示了在将“Pass”和“Fail”结果聚在一起显示时任务 1 与任务 2 之间的相似性。不过，任务 3 更加远离任务 1 和 2。Cochran *Q* 检验统计量 6.889 关联的 *p* 值为 0.0319。该 *p* 值支持拒绝在 0.05 显著性水平下所有任务的通过率相同的假设。

“多重对应分析”平台的统计详细信息

- “‘详细信息”报表”
- “调整惯量”
- “汇总统计量”
- “对惯量的部分贡献”
- “Cochran Q 统计量”

“详细信息”报表

执行简单的 “对应分析”时，报表列出的奇异值来自以下奇异值分解：

$$\mathbf{D}_r^{-0.5}(\mathbf{P} - rc')\mathbf{D}_c^{-0.5} = \mathbf{U}Diag(\Lambda)\mathbf{V}'$$

其中：

- **P** 是计数除以总频数所得的值构成的矩阵
- *r* 和 *c* 是 **P** 的行总和与列总和
- **D** 矩阵是 *r* 和 *c* 的值构成的对角矩阵
- Λ 是奇异值的列向量

执行 “多重对应分析”时，奇异值分解扩展为以下等式：

$$\mathbf{D}^{-0.5}(\mathbf{C} - dd')\mathbf{D}^{-0.5} = \mathbf{U}Diag(\Lambda)\mathbf{V}'$$

其中：

- **C** 是 Burt 表
- *d* 是 **C** 的列总和的列向量（*d* 还是行总和，因为 **C** 是对称的）
- **D** 是 *d* 的值的对角矩阵

在“详细信息”报表中，惯量是列向量 Λ。奇异值是惯量向量的平方根。列坐标计算如下：

$$\mathbf{D}^{-0.5}\mathbf{V}Diag\left(\Lambda^{0.5}\right)$$

调整惯量

在 MCA 中从 m 个分类变量构造的 Burt 表的常见主惯量为来自 Λ^2 的特征值 λ_k。这些惯量对于拟合好坏的判定过于悲观。Benzécri (1979) 提出了以下惯量调整方法； Greenacre (1984, p. 145) 也对该调整方法进行过描述。

$$\left(\frac{m}{m-1}\right)^2 \times \left(\lambda_k - \frac{1}{m}\right)^2 \text{ 对于 } \lambda_k > \frac{1}{m}$$

针对所有大于 $1/m$ 的惯量，该调整计算调整惯量占调整惯量之和的百分比。

Greenacre (1984, p. 156) 认为 Benzécri 调整过高估计了拟合的质量。Greenacre 建议计算调整惯量与以下值的百分比：

$$\frac{m}{m-1}\left(\text{tr}\left(Diag\left(\Lambda^4\right)\right) - \frac{n_c - m}{m^2}\right)$$

对于大于 $1/m$ 的所有惯量，其中 $\text{tr}(Diag(\Lambda^4))$ 是惯量平方和，n_c 是 m 个变量上的总类别数。

汇总统计量

质量是某点距离指定维数所定义空间中的原点的平方距离与具有最大维数的空间中的原点的距离之比。对于卡方这个量度，给定维中某点的质量可通过其向量与定义该维的向量所构成的余弦得到。质量还等于指定维中的惯量总和与所有维中的惯量总和的比值。质量表示低维空间中在多大程度上表示了该点的信息。

量是行或列的总频数与总频数之比。

惯量类似于主成分分析中的方差。整体惯量是二维频数表的总 Pearson 卡方除以表中所有观测值总和得到的值。

相对惯量是点对总惯量的贡献比例。在汇总统计量表中，相对惯量列在标记为“惯量”的列中。

对惯量的部分贡献

行或列对维的惯量的贡献计算如下：

贡献 = (量)(坐标)2/(维惯量)

Cochran Q 统计量

Cochran *Q* 检验统计量计算如下：

$$Q = k(k-1)\frac{\sum_{j=1}^{k}\left(\sum_{i=1}^{b} X_{ij} - \frac{N}{k}\right)^2}{\sum_{i=1}^{b}\left(\sum_{j=1}^{k} X_{ij}\left(k - \sum_{j=1}^{k} X_{ij}\right)\right)}$$

其中

K 是处理数

b 是区组数

X_{ij} 是第 *k* 个处理的第 *i* 个对象的响应 （0 或 1）

N 是所有对象和处理的正值响应的总计

第 8 章

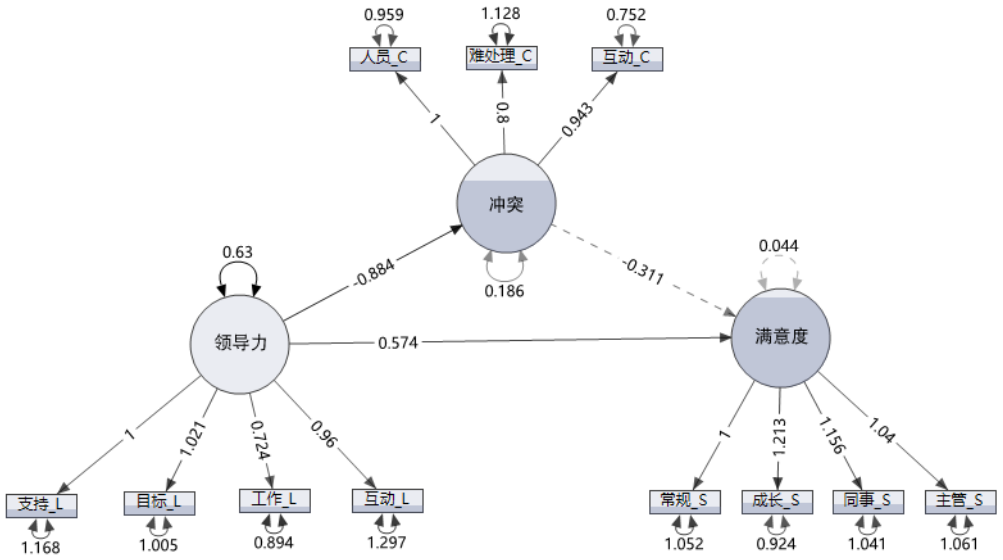
JMP PRO 结构化方程模型

拟合结构化方程模型

“结构化方程模型”平台仅在 JMP-Pro 中提供。

“结构化方程模型”平台支持您拟合多种模型。这些模型包括确认性因子分析、具有或不具有潜在变量的路径模型、测量值误差模型以及潜在变量增长曲线模型。您可以通过选择**自**和**至**变量来指定模型并指明它们如何关联。在示意图中，这种关联通过单向箭头或双向箭头表示。这些选择填充了模型的两个备选视图：动态路径图和模型边缘列表。该平台还会在您指定模型时检查模型身份。

图 8.1 结构化方程模型路径图



目录

结构化方程模型概述 157
结构化方程模型示例 159
启动“结构化方程模型”平台 163
“结构化方程模型”报表 164
“模型规格”报表 164
“模型比较”报表 169
“卡方差异检验”报表 170
“结构化方程模型拟合”报表 170
“结构化方程模型”平台选项 171
模型选项 172
定制 SEM 路径图 175
路径图弹出式菜单选项 175
“定制关系图外观”选项 177
结构化方程模型的更多示例 177
“结构化方程模型”平台的统计详细信息 184
拟合汇总测度的统计详细信息 184

JMP PRO 结构化方程模型概述

“结构化方程模型”(SEM) 平台支持您拟合可用于检验变量间关系理论的众多模型。模型中的变量可以是观测到的（**显**变量），也可以是未观测到的（**潜在**变量）。结构化方程建模在社会科学和行为科学中很普遍。

默认情况下，该平台为所有变量指定一个具有均值和方差的模型。然后，该平台提供了一个模型构建界面，使您能够在构建模型时看到模型的多个视图。它还提供模型构造过程中的一些模型细节，这些细节在运行模型之前会向您提醒哪些模型不受支持。

拟合一个或多个模型后，可以在“模型比较”报表中比较拟合的模型和两个基线模型。基线模型是不受限模型和独立模型。不受限模型是完全饱和模型，该模型拟合指定模型变量的所有均值、方差和协方差，而不对数据强加任何结构。独立模型拟合指定的模型变量的所有均值和方差。指定模型变量之间的所有协方差都被固定为零，这会导致高度受限的模型。

SEM 平台使用全信息最大似然 (Finkbeiner 1979) 方法。这使您能够充分利用数据中的所有可用信息，即使在具有随机缺失值的观测比例较高时也是如此。

有关结构化方程建模的详细信息，请参见 SAS Institute Inc.(2020a) 中的“CALIS 过程”一章、Bollen (1989) 和 Kline (2016)。

注意：“结构化方程模型”平台中的所有模型都用一个平均结构来估计，这意味着包含一个“常数”项。若不想让结构基于观测变量的均值，则应按照默认模型规格中的方式自由估计均值。

模型类型

本节介绍了可以在 “结构化方程模型”平台中拟合的一些不同类型的模型：

- **路径分析**支持您检验观测到的变量之间关联的备选解释模型。当研究中每个关注的结构只有一个变量可用时，通常使用此方法。也许最简单的“路径分析”模型是一个标准回归模型，在该模型中 X 预测 Y。SEM 平台支持您拟合该简单回归模型，但您也可以指定更有趣的模型。例如，您可能有变量 Z，根据理论或之前的研究，设定该变量为 $X \Rightarrow Y$ 关系的中介。换言之，X 预测 Z，然后 Z 预测 Y。因此，最初的 $X \Rightarrow Y$ 关系可能只因在原始模型中排除了 Z 才存在。可通过执行以下步骤在 SEM 平台中执行路径分析：
  1. 在启动窗口中选择所有观测到的变量，点击“模型变量”，然后点击“确定”。
  2. 在 “模型规格”报表中，在 “自列表”中选择预测变量，在 “至列表”中选择对应结果，然后点击单向箭头按钮。

  注意：所有外生变量（那些没有指向它们的任何单向箭头的变量）必须在模型中自由相关，除非检验的是零相关的假设。使用双向箭头按钮指定这些协方差。

- **确认性因子分析** (CFA) 支持您检验备选测量模型。CFA 通常用于调查开发，并用作拟合结构化回归模型之前的初始步骤。SEM 平台通过执行以下步骤支持拟合确认性因子分析模型：
  1. 在启动窗口中选择所有观测到的变量，点击“模型变量”，然后点击“确定”。

2. 使用 “模型规格”下的 “至列表”，选择假定要加载到潜在变量上的变量。
3. 在 “至列表”下方的框中输入潜在变量的名称，然后点击添加潜在变量 ⊞ 按钮。
4. 重复该过程，直到指定了模型的所有潜在变量。

请注意，SEM 平台总是包含一个平均结构，因此所有观测到的变量都列在“均值 / 截距”列表中，作为 “常数”项的结果。此外，若在启动窗口中选择了 “标准化潜在变量”选项，则通过将其第一个指标的载荷设置为 1（默认值）或将其方差设置为 1，可自动识别所有潜在变量。最后，传统的 CFA 模型允许所有潜在变量共变。通过选择 “自列表”和 “至列表”中的所有潜在变量，然后点击双向箭头按钮，可以指定这些协方差。

- **结构化回归** (SR) 模型亦称**带潜在变量的路径分析**。这些模型通常在通过确认性因子分析 (CFA) 确定为合适的测量模型后使用。SR 模型支持您检验潜在变量之间特定的关系模式。换言之，尽管 CFA 不会针对潜在变量之间的效应强加任何方向（所有潜在变量都允许自由共变），但 SR 模型会强加。在假设管理层“领导力”在工作场所会导致较少的团队“冲突”和更高的员工“满意度”的例子中，“领导力”潜在变量可以预测“冲突”和“满意度”潜在变量。通过执行以下步骤，您可以在执行 CFA 后指定这些方向效应（回归）：
  1. 在启动窗口中选择所有观测到的变量，点击“模型变量”，然后点击“确定”。
  2. 使用 “模型规格”下的 “至列表”，选择假定要加载到潜在变量上的变量。
  3. 在 “至列表”下方的框中输入潜在变量的名称，然后点击添加潜在变量 ⊞ 按钮。
  4. 重复该过程，直到指定了模型的所有潜在变量。
  5. 在 “模型规格”报表中，在 “自列表”中选择预测变量，在 “至列表”中选择对应结果，然后点击单向箭头按钮。
- **潜在变量增长曲线** (LGC) 模型支持您拟合和检验重复测量数据的替代潜在轨迹。这些模型与混合模型框架中的随机效应模型非常相似。通常，您希望将无增长模型与线性模型进行比较。在无增长模型中，个体的起点可能不同，但轨迹平坦。在线性模型中，个体的起点和线性斜率都会随时间而变化。若有足够的数据可用，您还可以将这些模型与二次模型进行比较，在二次模型中，个体的起点以及线性和二次变化率均随时间而变化。您可以使用 SEM 平台模型规格来拟合 LGC 模型，但是该平台通过执行以下步骤来集成 LGC 模型的拟合：
  1. 在启动窗口中选择所有观测到的变量（重复测量），点击“模型变量”，然后点击 “确定”。

     注意：为了使“潜在变量增长”模型快捷方式正确地指定模型，观测到的变量必须按时间顺序升序列出，并且必须具有相等的时间间隔。

  2. 使用 “模型快捷方式”选项，选择 “纵向分析” > “仅截距增长曲线”模型，然后点击 “运行”。
  3. 使用 “模型快捷方式”选项，选择 “纵向分析” > “线性潜在变量增长曲线”模型，然后点击 “运行”。
  4. 使用 “模型快捷方式”选项，选择 “纵向分析” > “二次潜在变量增长曲线”模型，然后点击 “运行”。

  “模型比较”表显示了备选的拟合指数，并且可以选择最佳模型。

- 按照上述步骤确定理想的增长轨迹后，可以使用**条件潜在变量增长曲线**模型。此时，可以将截距和变化因子的预测变量添加到模型中。这些预测变量可能被证明是决定生长过程初始得分和随后变化的重要因子。要拟合条件 LGC，请在启动窗口中选择所有观测到的变量（重复测量），包括潜在变量的假设预测变量。确保预测变量是“模型变量”列表中最后的变量，以便于执行以下步骤：
  1. 使用“模型快捷方式”选项选择适当的增长轨迹。该选项指定 LGC 模型中的所有变量，包括预测变量。因此，您需要从增长过程中排除预测变量，并正确地将它们指定为预测变量。
  2. 在“载荷”列表中找到预测变量，选择涉及它们的所有效应，然后点击“删除”。
  3. 在“自列表”中选择预测变量，在“至列表”中选择“截距”或“斜率”。
  4. 点击单向箭头以指定条件 LGC。

注意：若有多个预测变量，则必须通过在“自列表”和“至列表”中选择预测变量并点击双向箭头按钮来指定它们的协方差。

JMP PRO 结构化方程模型示例

在本例中，人力资源部门的员工希望提高工作满意度。本例使用“结构化方程模型”平台分析了关于 200 个人对其工作满意度调查的回复。该调查包含对与工作满意度相关的 11 个问题的回复。您想要构造一个结构化回归模型，将调查问题的答复与领导力特征、角色冲突和整体工作满意度等潜在变量联系起来。

1. 选择**帮助** > **样本数据库**，然后打开 Job Satisfaction.jmp。
2. 选择**分析** > **多元方法** > **结构化方程模型**。
3. 从**支持 _L** 一直选择到**主管 _S**，然后点击**模型变量**。
4. 点击**确定**。

   “结构化方程模型”报表“模型规格”分级显示项随即显示。

创建潜在变量

5. 在“至列表”中从**支持 _L** 一直选择到**互动 _L**，在“至列表”下方的框中键入“领导力”，然后点击添加潜在变量 ⊞ 按钮。
6. 在“至列表”中从**人员 _C** 一直选择到**互动 _C**，在“至列表”下方的框中键入“冲突”，然后点击添加潜在变量 ⊞ 按钮。
7. 在“至列表”中从**常规 _S** 一直选择到**主管 _S**，在“至列表”下方的框中键入“满意度”，然后点击添加潜在变量 ⊞ 按钮。

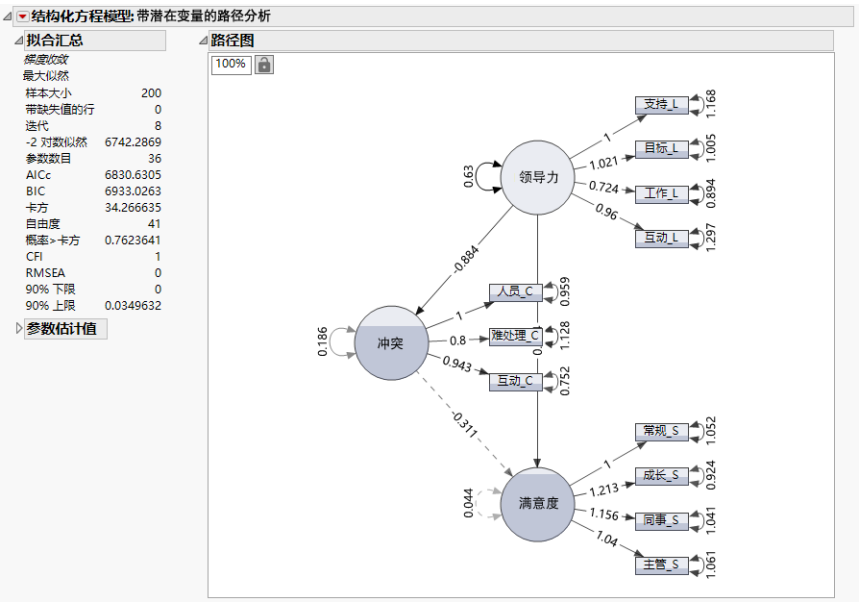
添加载荷和回归变量

8. 在“自列表”中选择**领导力**，在“至列表”中选择**满意度**，然后点击单向箭头 → 按钮。

9. 在 “自列表”中选择**领导力**，在 “至列表”中选择**冲突**，然后点击单向箭头 → 按钮。
10. 在 “自列表”中选择**冲突**，在 “至列表”中选择**满意度**，然后点击单向箭头 → 按钮。
11. 在 “模型名称”（位于 “模型规格”报表左上角）下方的文本框中，键入 “带潜在变量的路径分析”。
12. 点击**运行**。
13. （可选。）点击 “结构化方程模型：带潜在变量的路径分析”旁边的红色小三角，然后选择**路径图设置 > 布局 > 从上到下**。
14. （可选。）点击 “参数估计值”旁边的灰色展开图标。

    关闭 “参数估计值”报表将支持您查看整个路径图。

图 8.2 “结构化方程模型”报表



“拟合汇总”报表中列出的该模型的卡方统计量为 34.27，自由度为 41。请注意，相应的 p 值为 0.7624，该值不显著。这表明，没有证据可以拒绝模型拟合良好这一原假设。因此，您可以得出结论：该模型对数据拟合良好。

卡方值取决于样本大小，因此，一些拟合良好的模型仍然可以生成显著的卡方值。比较拟合指数 (CFI) 和近似的均方根误差 (RMSEA) 为确定模型拟合提供了额外的指导。这些指数介于 0 和 1 之间。 CFI 值最好大于 0.90， RMSEA 值最好小于 0.10（Browne and Cudeck 1993 ；Hu and Bentler 1999）。在此， CFI 为 1 且 RMSEA 为 0，指示拟合极佳。

15. 点击 “结构化方程模型：带潜在变量的路径分析”旁边的红色小三角，然后选择**标准化参数估计值**。

图 8.3 “标准化参数估计值”报表

标准化参数估计值

均值/截距	标准化估计值	标准误差	Wald Z	概率>|Z|
Constant → 支持_L	4.0682065	0.2153502	18.891119	<.0001*
Constant → 目标_L	5.1119076	0.2651961	19.275951	<.0001*
Constant → 工作_L	7.9634812	0.404404	19.691897	<.0001*
Constant → 互动_L	2.1272415	0.1277219	16.655258	<.0001*
Constant → 人员_C	1.629217	0.1078697	15.103567	<.0001*
Constant → 难处理_C	2.2080708	0.1311066	16.841798	<.0001*
Constant → 互动_C	1.339463	0.097393	13.753172	<.0001*
Constant → 常规_S	6.1033674	0.3132534	19.483803	<.0001*
Constant → 成长_S	4.7045478	0.2456255	19.153333	<.0001*
Constant → 同事_S	4.7723896	0.248876	19.175776	<.0001*
Constant → 主管_S	2.2784553	0.1340835	16.992805	<.0001*
载荷	**标准化估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
领导力 → 支持_L	0.5920492	0.0567471	10.433128	<.0001*
领导力 → 目标_L	0.6288603	0.0535719	11.738623	<.0001*
领导力 → 工作_L	0.5192588	0.0618179	8.3998073	<.0001*
领导力 → 互动_L	0.5560815	0.0582571	9.545293	<.0001*
冲突 → 人员_C	0.6439066	0.0563634	11.424196	<.0001*
冲突 → 难处理_C	0.5275504	0.0617549	8.5426493	<.0001*
冲突 → 互动_C	0.6674685	0.0530474	12.582499	<.0001*
满意度 → 常规_S	0.573911	0.0565405	10.150439	<.0001*
满意度 → 成长_S	0.671917	0.0479837	14.003039	<.0001*
满意度 → 同事_S	0.6316909	0.0512488	12.325959	<.0001*
满意度 → 主管_S	0.5875892	0.0545478	10.771997	<.0001*
回归	**标准化估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
领导力 → 冲突	-0.852045	0.0682055	-12.49232	<.0001*
领导力 → 满意度	0.6339309	0.2475569	2.5607479	0.0104*
冲突 → 满意度	-0.356977	0.2528216	-1.411974	0.1580
方差	**标准化估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
支持_L ↔ 支持_L	0.6494777	0.0671941	9.6656955	<.0001*
目标_L ↔ 目标_L	0.6045347	0.0673785	8.9722227	<.0001*
工作_L ↔ 工作_L	0.7303703	0.064199	11.376656	<.0001*
互动_L ↔ 互动_L	0.6907734	0.0647914	10.661493	<.0001*
人员_C ↔ 人员_C	0.5853843	0.0725855	8.0647503	<.0001*
难处理_C ↔ 难处理_C	0.7216906	0.0651576	11.076069	<.0001*
互动_C ↔ 互动_C	0.5544858	0.0708149	7.830072	<.0001*
常规_S ↔ 常规_S	0.6706261	0.0648984	10.333469	<.0001*
成长_S ↔ 成长_S	0.5485275	0.0644821	8.506667	<.0001*
同事_S ↔ 同事_S	0.6009666	0.0647468	9.2817923	<.0001*
主管_S ↔ 主管_S	0.6547389	0.0641035	10.213786	<.0001*
领导力 ↔ 领导力	1	3.42e-17	2.924e+16	<.0001*
冲突 ↔ 冲突	0.2740191	0.1162284	2.357592	0.0184*
满意度 ↔ 满意度	0.0850648	0.0780564	1.0897857	0.2758

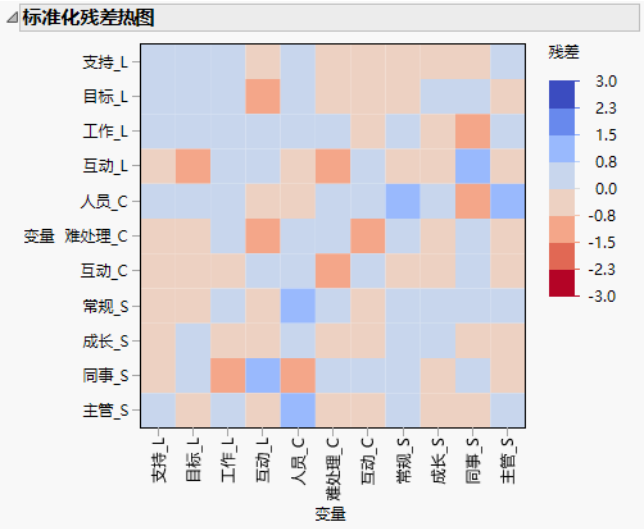
“标准化参数估计值”报表中的“载荷”的估计值有助于解释定义潜在变量的测量模型。标准化载荷指示观测到的变量与未观测到的潜在变量的相关性。报表中所有潜在变量的载荷都介于 0.52 到 0.67 之间，这表明“领导力”、“冲突”和“满意度”构造良好。

在图 8.3 中，“回归”下的参数估计值表明，“领导力”对“冲突”以及“冲突”对“满意度”的影响是负面的，而“领导力”对“满意度”的影响是正面的。因此，“领导力”得分越高，“冲突”得分就越低，“满意度”得分就越高；“冲突”得分越高，“满意度”得分就越低。参数估计值相应的 p 值显示在“回归”下方。“领导力 $\Rightarrow$ 冲突”和“领导力 $\Rightarrow$ 满意度”回归参数在 $\alpha = 0.05$ 水平下均显著。因此，您可以得出结论，潜在变量之间存在非零关系。

注意：您还可以将“标准化参数估计值”报表中的“回归”参数估计值用作效应大小。这些效应大小被解释为预测变量中一个标准差变化的结果中标准差单位的变化。

16. 点击“结构化方程模型：带潜在变量的路径分析”旁边的红色小三角，然后选择**标准化残差热图**。

图 8.4 标准化残差热图



“标准化残差热图”在正方向或负方向上没有超过 2.0 个单位的值；这进一步证明了该模型拟合数据良好。残差也可以在更细粒度上诊断拟合不良的模型。该模型中的标准化残差没有显示出局部失拟的证据。

JMP PRO 启动 “结构化方程模型” 平台

通过选择**分析 > 多元方法 > 结构化方程模型**来启动 “结构化方程模型” 平台。

图 8.5 “结构化方程模型” 启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

启动窗口包括两种不同数据格式的选项卡。

宽数据格式 选择这些数据表：其表中每行对应于单个观测，而列包含了要在模型中使用的变量。仅包含缺失值的行会从分析中排除。

汇总数据格式 选择这些数据表：其数据汇总为相关性或协方差矩阵。均值与标准差也可以指定为列。若未指定它们，则均值假定为 0，标准差假定为输入矩阵对角线的平方根。

JMP PRO 启动窗口选项

模型变量 要包含在模型中的列。必须指定至少一列。所有列都必须具有数值型数据类型和连续型建模类型。

对于 “汇总数据格式”，“模型变量” 列是包含汇总数据的相关性或协方差矩阵的列。

依据 （仅可用于 “宽数据格式”。）一列，用于创建为变量的每个水平包含单独分析的报表。若分配了多个 “依据” 变量，则为 “依据” 变量水平的每个可能组合生成单独分析。

警告：使用 “依据” 变量不执行多组分析。

均值 （仅可用于“汇总数据格式”。）与相关性或协方差矩阵的每行中的变量相对应的均值列。若未指定，则均值假定为 0。

标准差 （仅可用于“汇总数据格式”。）与相关性或协方差矩阵的每行中的变量相对应的标准差列。若未指定，则标准差假定为输入矩阵对角线的平方根。

标签 （仅可用于“汇总数据格式”。）与相关性或协方差矩阵的每行中的变量相对应的标签列。若未指定，则“模型规格”中的变量将使用包含输入矩阵的列的名称。

标准化潜在变量 若选定，该选项会通过将潜在变量的方差固定为 1 并允许对所有载荷进行自由估计来设置潜在变量的尺度。

拟合不受限模型 若选定，则在拟合第一个模型时，将自动拟合不受限模型。随后可以在“模型比较”报表中将拟合模型与不受限模型进行比较。不受限模型是完全饱和模型，该模型拟合指定模型变量的所有均值、方差和协方差，而不对数据强加任何结构。

拟合独立模型 若选定，则在拟合第一个模型时，将自动拟合独立模型。随后可以在“模型比较”报表中将拟合模型与独立模型进行比较。独立模型拟合指定的模型变量的所有均值和方差。指定模型变量之间的所有协方差都被固定为零，这会导致高度受限的模型。

样本大小 （仅可用于“汇总数据格式”。）指定汇总数据表示的观测数。

JMP PRO “结构化方程模型”报表

- “‘模型规格’报表”
- “‘模型比较’报表”
- “‘卡方差异检验’报表”
- “‘结构化方程模型拟合’报表”

JMP PRO “模型规格”报表

“模型规格”报表包含用于指定模型的控件。在启动窗口中点击“确定”后，“模型规格”报表的“关系图”选项卡和“列表”选项卡中将显示默认的独立模型。独立模型包含指定的模型变量的所有均值和方差。

JMP PRO “规格”面板

“规格”面板包含用于构建模型的控件。

模型名称 支持您指定模型名称。在“操作”面板中点击“运行”后，将创建使用“模型名称”中的文本的模型报表。

自列表 列出在启动窗口中指定的模型变量以及“Constant”项。“Constant”项通过在“Constant”上回归每个变量来估计任何观测到的或潜在变量的均值。在点击箭头按钮之一

将项添加到模型之前，先在该列表中选择一个或多个变量并且在“至列表”中选择一个或多个变量。若创建潜在变量，则潜在变量会添加到“自列表”和“至列表”。可以使用列表顶部的搜索过滤器控件过滤该列表中的项目。请参见第 168 页的 “搜索过滤器选项”。

至列表 列出在启动窗口中指定的模型变量以及“Constant”项。“Constant”项通过在“Constant”上回归每个变量来估计任何观测到的或潜在变量的均值。在点击箭头按钮之一将项添加到模型之前，先在“自列表”中选择一个或多个变量并且在该列表中选择一个或多个变量。若创建潜在变量，则潜在变量会添加到“自列表”和“至列表”。必须选择“至列表”中的变量才能将潜在变量添加到模型中。可以使用列表顶部的搜索过滤器控件过滤该列表中的项目。请参见第 168 页的 “搜索过滤器选项”。

单向箭头 → 指定“自列表”和“至列表”中选定的变量之间的关系类型。单向箭头等效于回归效应。

双向箭头 ↔ 指定“自列表”和“至列表”中选定的变量之间的关系类型。双向箭头等效于协方差效应。

添加潜在变量 + 向“自列表”和“至列表”添加潜在变量。变量的命名基于 + 按钮左侧框中的文本。

注意：在点击 + 按钮之前，必须在“至列表”中选择潜在变量的指标。

删除潜在变量 − 删除“自列表”或“至列表”中当前选定的任何潜在变量。从“自列表”和“至列表”中删除某个潜在变量也会从模型中删除该潜在变量。

模型快捷方式 包含三类用于快速构建常见模型的选项。请注意，选择其中一个选项时，将清除当前所有模型规格。

规格帮助 包含的选项可快捷地处理为外生变量或内生变量添加协方差的过程。

截面经典 包含用于确认性因子分析和中介分析的选项。

纵向分析 包含用于常见潜在变量增长曲线模型的选项：仅截距、线性、二次和潜在基函数增长曲线。此外，通过“拟合并比较增长模型”选项，您可以在一次点击中拟合无增长、线性增长和二次增长模型。

注意：若在选择某一潜在变量增长曲线选项时选择了“至列表”中的变量，则模型仅应用于选定的变量。否则，该模型将应用于“至列表”中的所有变量。

JMP PRO “操作”面板

“操作”面板中的按钮针对“列表”选项卡中的指定模型变量列表进行操作。可使用以下操作：

固定为 支持您将当前选定效应的参数值固定为常数值。为变量固定参数值时，固定值将显示在项名称后面的括号中。

注意：若点击“确定”时保持默认值为 0，则将从模型中删除选定效应。

设置等于 支持您将两个或更多选定效应的参数值限定为相等。若将两个或更多效应设置为具有相同的参数值，则在项名称后的括号中会显示任意字母数字标签 (“c1”)。若多组效应的参数值都设置为彼此相等，则使用顺序编号 （“c1”、“c2”等）。

注意：仅允许在同一类型的参数中使用等式约束。

自由 支持您删除针对模型中选定效应的约束。约束包括已设置为固定值的效应或已设置为与其他效应相等的效应。

删除 从模型中删除选定的效应。

撤销 撤销对模型的最后一次修改。

恢复 恢复对模型上次撤销的更改。

运行 为当前指定模型拟合和创建报表。

重置 将模型规格重置为独立模型，这是默认值。

JMP PRO “关系图”面板

“关系图”面板包含支持您调整模型关系图布局的按钮。第一个按钮支持您旋转关系图中的显变量（用矩形表示）。第二个按钮支持您循环查看关系图的两种不同排列。第三个按钮支持您定制关系图。请参见第 175 页的“定制 SEM 路径图”。

JMP PRO “详细信息”面板

“详细信息”面板包含有关“模型规格”报表中当前指定的模型的信息。有关模型的信息包括显变量的数量、潜在变量的数量、自由估计参数的数量和自由度的数量。您还可以调整最大迭代次数。

JMP PRO “关系图”选项卡

“关系图”选项卡包含一个支持您对当前指定模型进行可视化的模型关系图。潜在变量用圆圈表示，显变量用矩形表示。单向箭头表示载荷和回归。双向箭头表示方差和协方差。方差由从一个变量到自身的双向箭头指定。协方差由两个变量之间的双向箭头指定。

路径图是交互式的，因此可以拖动各项来排列它们。您还可以使用箭头键移动路径图中的选定项。可以使用“关系图”面板中的按钮重新排列路径图中的项。可以使用路径图弹出式菜单中的选项来添加或删除路径图中的项。右击路径图本身时，会出现路径图弹出式菜单。弹出式菜单的内容根据您是点击空白绘制区的某个部分还是点击路径图的某个元素而有所不同。也可以通过调整路径图左上角的缩放设置来放大或缩小路径图。路径图左上角还包含一个“锁定”按钮，支持您锁定路径图中各项的位置。当路径图被锁定时，“锁定”按钮为蓝色。在路径图中进行选择后，可以通过在弹出式菜单中选择**选择 > 隐藏选择**选项来隐藏选择。当路径图中的项被隐藏时，路径图顶部会显示一个“全部取消隐藏”按钮，该按钮支持您显示所有隐藏的项。

有关路径图定制选项的详细信息，请参见第 175 页的“定制 SEM 路径图”。请参见第 166 页的“‘关系图’面板”、第 166 页的“‘详细信息’面板”和第 171 页的“路径图设置”。

提示：要将路径图复制为图像，请右击该图并选择**复制关系图**。要保持尽可能高的质量，请将剪贴板图像粘贴为向量图形。

JMP PRO “列表”选项卡

“列表”选项卡包含模型中每种类型变量的列表。这些列表按照在模型图中表示它们所用的箭头来分类。单向箭头用于指定均值或截距、载荷和回归。双向箭头用于指定方差和协方差。模型的均值和截距仅显示在“列表”选项卡中，而不出现在“关系图”选项卡中。

可以使用每个列表顶部的搜索过滤器控件过滤每个列表中的项目。请参见第 168 页的“搜索过滤器选项”。

JMP PRO “状态”选项卡

“状态”选项卡包含用于模型可识别性的检查。必须标识结构化方程模型以获取可靠估计值。SEM 上下文中的识别意味着：根据输入数据的总体协方差矩阵可对模型中的每个参数求出唯一解。因为可以在 SEM 框架中指定如此多的模型，所以有多种识别规则。“状态”选项卡包含三个信息面板：“识别规则”、“模型详细信息”和“数据详细信息”。

- “识别规则”面板包含一个列表，其中包含最多八个适用于指定模型的规则。一些规则是必要的，而另一些规则是保证模型可识别的充分条件。若必要规则失败，则必须在拟合模型之前采取步骤更正这些规则。若充分规则失败，您不一定需要解决任何问题。有时，充分规则可能会失败，而这并不能证明模型有任何错误。请注意，所有规则都假设协方差矩阵是正定的。若协方差矩阵不是正定的，则会在“模型详细信息”面板下显示警告。

  提示：有关特定识别规则的详细信息，请点击该规则所在表中的行，该规则的说明将显示在该表的右侧。

- “模型详细信息”面板包含当前指定模型的描述性值列表。这些值可用于检测模型的潜在问题。
- “数据详细信息”面板包含用于输入数据的描述性值列表。这些值可用于检测数据的潜在问题。

若指定的数据列中存在奇异性，则报表包含“奇异性详细信息”表。

“状态”选项卡本身包含一个动态图标，该图标显示指定模型的当前状态。该图标指定以下状态：

- 所有适用识别规则均通过。
- 至少有一个必要识别规则未通过，您必须在拟合模型之前采取步骤更正问题。

至少有一个非必要的识别规则未通过，需要进一步调查以确定模型是否指定无误。通常，SEM 的高级应用会导致这种情况；这并不意味着模型有问题。相反，它表明识别规则不能保证模型可识别。

图 8.6 “状态”选项卡示例

JMP PRO 搜索过滤器选项

“模型规格”报表的“列表”选项卡中显示的“自列表”、“至列表”和列表框包含搜索过滤器，这些过滤器支持您过滤特定列表中的项目。

点击搜索框旁边的下箭头按钮以优化搜索。

包含词条 返回包含一部分搜索条件的项。搜索“ease oom”返回如“Release Zoom”这样的消息。

包含短语 返回包含完全搜索条件的项。搜索 “text box”返回包含 “text”后面直接跟着 “box”的条目 （例如，“Context Box”和 “Text Box”）。

以短语开头 返回以搜索条件开始的项。

以短语结尾 返回以搜索条件结束的项。

整个短语 返回包含整个字符串的项。搜索 “text box”返回仅包含 “text box”的条目。

正则表达式 允许您在搜索框中使用通配符 (*) 和句点 (.)。搜索 “get.*name”查找包含 “get”后面跟着一个或多个单词的项。它返回 “Get Color Theme Names”、“Get Name Info”、“Get Effect Names”等。

反转结果 返回不匹配搜索条件的项。

匹配全部词条 返回同时包含两个字符串的项。搜索 “t test”返回包含任一搜索字符串或两者的元素：“Pat Test”、“Shortest Edit Script”和 “Paired t test”。

忽略大小写 忽略搜索条件中的大小写。

全字匹配 基于“匹配全部词条”设置返回包含字符串中每个单词的项。若您搜索“data filter”，并且选择了 “匹配全部词条”，则返回同时包含 “data”和 “filter”的条目。

JMP PRO “模型比较”报表

“模型比较”报表包含已经拟合的所有模型的表。使用“显示”列中的复选框来控制“模型比较”报表下方显示的模型报表。“模型名称”右侧的其余列支持您根据各种条件比较模型。

模型的“AICc 权重”值可解释为在其中的一个拟合模型成立的前提下，特定模型为真实模型的概率。因此，AICc 权重最接近 1 的模型是最好的拟合。仅使用非缺失 AICc 值计算 AICc 权重，该权重定义如下：

AICc 权重 = exp[-0.5(AICc-min(AICc))] / sum(exp[-0.5(AICc-min(AICc))])

其中， min(AICc) 是拟合模型中的最小 AICc 值。

有关“模型比较”报表的其他准则的信息，请参见第 170 页的 “ “结构化方程模型拟合”报表”。

注意：若某个模型不收敛，则“模型比较”报表中模型名称的开头会显示一个星号。

为了提供拟合模型的性能的上下文，“模型比较”报表中默认显示以下两个模型：

不受限 拟合指定模型变量的所有均值、方差和协方差，而不对数据强加任何结构。

独立 拟合指定模型变量的所有均值和方差，并将所有协方差固定为零。

“模型比较”报表选项

“模型比较”表下显示两个选项。在 “模型比较”表中通过点击行使其突出显示后，这些选项可用。

比较选定模型 （仅当表的两行或更多行突出显示时才可用。）计算选定行的所有嵌套模型组合之间的嵌套卡方差异检验。

提示：若您选择的任何模型没有嵌套，则会出现警告，并且“卡方差异检验”报表中不会出现非嵌套模型组合。

清除选择 （仅当表的一个或多个行突出显示时才可用。）从表行中清除所有选择。

JMP PRO “卡方差异检验”报表

“卡方差异检验”报表包含嵌套卡方检验的表。该表包含定义模型的两列。第一列包含两个模型中约束较强的模型，第二列包含两个模型中约束较弱的模型。较小模型嵌套在较大模型中。其余列显示了卡方值、自由度、CFI 和 RMSEA 中的差异，以及嵌套卡方检验的 p 值。列名中的 Δ 符号指示差异。显著的 Δ 卡方值指示嵌套模型中的附加约束在统计上显著增加了失配，应保留约束较少的模型。由于卡方检验受样本大小的影响，以致于随着样本量的增加，卡方检验更可能显著，所以还应该考虑 ΔCFI 和ΔRMSEA；理想情况下，ΔCFI 不应超过 –0.01，ΔRMSEA 不应超过 0.015 (Chen 2007)。

警告：该报表中的差异检验仅对嵌套模型有意义。

您可以通过点击红色的 X 按钮删除报表的任何行。要完全删除报表，必须删除表中的所有行。若在主报表中删除“结构化方程模型”节点，则将从表中删除涉及该模型的所有差异检验。

JMP PRO “结构化方程模型拟合”报表

每次点击 “运行”时，都会显示指定模型的 “结构化方程模型”报表。默认情况下，该报表包含 “拟合汇总”报表、“参数估计值”报表和 “路径图”。

拟合汇总 有关模型拟合 （包括收敛状态和估计方法）的信息表。该表中报告以下统计量：

样本大小 用于拟合模型的观测（行）数。

带缺失值的行 包含至少一个缺失值的观测（行）数。所有缺失值都使用全信息最大似然 (Finkbeiner 1979) 来处理。

迭代 用于拟合模型的迭代数。

-2 对数似然 拟合模型的对数似然乘以 -2。该值可用于比较嵌套模型；两个模型的 -2 对数似然值之间的差值服从卡方分布，其自由度等于两个模型之间的自由度差值。请参见《拟合线性模型》。

参数数目 模型中的自由估计参数数目。

AICc 校正的 Akaike 信息准则。该值可用于比较模型，其中较小的数字表示模型拟合更好。请参见第 184 页的“AICc 和 BIC”。

BIC Bayes 信息准则。该值可用于比较模型，其中较小的数字表示模型拟合更好。请参见第 184 页的“AICc 和 BIC”。

卡方 模型的卡方统计量。

自由度 模型拟合的卡方检验的自由度。

概率>卡方 模型的卡方统计量的 *p* 值。

CFI Bentler 比较拟合指数 (CFI) 为确定模型拟合提供了额外的指导。CFI 介于 0 和 1 之间。值最好大于 0.90（Browne and Cudeck 1993； Hu and Bentler 1999）。请参见第 184 页的“CFI”。

RMSEA 近似的均方根误差 (RMSEA) 为确定模型拟合提供了额外的指导。RMSEA 介于 0 和 1 之间。值最好小于 0.10（Browne and Cudeck 1993； Hu and Bentler 1999）。请参见第 186 页的“RMSEA”。

90% 下限 RMSEA 的 90% 置信下限。请参见第 186 页的“RMSEA”。

90% 上限 RMSEA 的 90% 置信上限。请参见第 186 页的“RMSEA”。

参数估计值 模型参数的估计值表。该表按均值 / 截距、载荷、回归和方差等部分来组织。对于每个估计值，给出标准误差 (Std Error)、Wald 检验统计量 (Wald Z) 和相应的 *p* 值（概率 >|Z|）。

提示：“参数估计值”表包含隐藏列，这些列标识每个模型参数的“自”成分和“至”成分。要显示这些列，请右击该表并选择**列 > 自**和**列 > 至**。

路径图 显示拟合模型的路径图表示。请参见第 166 页的“‘关系图’选项卡”。

JMP PRO “结构化方程模型”平台选项

“结构化方程模型”红色小三角菜单包含以下选项：

路径图设置 包含以下用于修改路径图显示的选项：

定制关系图 支持您定制路径图的许多方面。请参见第 177 页的“‘定制关系图外观’选项”。

布局 包含更改路径图总体形状的两个选项。您可以选择“从左到右”的布局或“从上到下”的布局。

提示：还可以拖动路径图中的项以更改特定项的排列方式。

复制关系图 将路径图的图像保存到剪贴板。要保持尽可能高的质量，请将剪贴板图像粘贴为向量图形。

复制关系图属性 将当前路径图属性复制到剪贴板。随后可将属性粘贴到另一个 SEM 路径图中。

粘贴关系图属性 将剪贴板中的路径图属性粘贴到当前 SEM 路径图中。

描述性统计量 包含以下用于生成描述性统计量的选项：

一元简单统计量 显示或隐藏一元简单统计量报表。该报表中的统计量是独立于可能具有缺失数据的其他列对每个列进行估计的。

全信息多元统计量 显示或隐藏多元简单统计量报表。该报表中的统计量使用全信息最大似然来估计，以解释缺失数据。

启动探索离群值 启动“探索离群值”平台。请参见《预测和专业建模》。

启动探索缺失值 （有缺失值时不可用。）启动“探索缺失值”平台。请参见《预测和专业建模》。

添加显变量 支持您向当前模型添加新的显变量。选择要添加的新的显变量后，将启动一个新的“结构化方程模型”报表。新报表对新添加的显变量使用现有的模型规格。

删除显变量 支持您在删除指定显变量的情况下启动新的“结构化方程模型”报表。选择要删除的显变量后，将启动一个新报表。新报表在已删除显变量的情况下使用现有模型规格。

复制模型规格 将当前结构化方程模型规格复制到剪贴板。随后可以将模型规格粘贴到另一个 SEM 平台报表中。

粘贴模型规格 将剪贴板中的模型规格粘贴到当前的模型规格中。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO 模型选项

点击“运行”后，就会显示指定模型的“结构化方程模型”报表。该报表有一个包含以下选项的红色小三角菜单：

显示路径图 在模型报表中显示或隐藏路径图。

路径图设置 包含以下用于修改模型路径图的选项：

定制关系图 支持您定制路径图的许多方面。请参见第 177 页的““定制关系图外观”选项”。

布局 包含更改路径图总体形状的两个选项。您可以选择“从左到右”的布局或“从上到下”的布局。

提示：还可以拖动路径图中的项以更改特定项的排列方式。

复制关系图 将路径图的图像保存到剪贴板。要保持尽可能高的质量，请将剪贴板图像粘贴为向量图形。

复制关系图属性 将当前路径图属性复制到剪贴板。随后可将属性粘贴到另一个 SEM 路径图中。

粘贴关系图属性 将剪贴板中的路径图属性粘贴到当前 SEM 路径图中。

拟合指数 显示或隐藏包含各种指数值的报表，这些指数值支持您评估拟合的模型。除 “拟合汇总”报表中显示的值之外，“拟合指数”报表还包含以下指数值：

TLI Tucker-Lewis 指数 (TLI) 为确定模型拟合提供更多指导。该指数也称为非规范拟合指数 (NNFI)。TLI 介于 0 和 1 之间。值最好大于 0.95 (West et al. 2012)。请参见第 185 页的“TLI”。

NFI 规范拟合指数 (NFI) 为确定模型拟合提供了额外的指导。NFI 介于 0 和 1 之间。值最好大于 0.95 (West et al. 2012)。请参见第 185 页的“NFI”。

修正的 GFI 修正的拟合优度指数为确定模型拟合提供了额外的指导。修正的 GFI 介于 0 和 1 之间。值最好大于 0.95 (West et al. 2012)。请参见第 185 页的“修正的 GFI 和修正的 AGFI”。

修正的 AGFI 修正的已调整拟合优度指数为确定模型拟合提供了额外的指导。修正的 AGFI 介于 0 和 1 之间 (West et al. 2012)。请参见第 185 页的“修正的 GFI 和修正的 AGFI”。

RMR 均方根残差 (RMR) 为确定模型拟合提供了额外的指导。RMR 的残差来自观测的协方差与隐含模型的协方差之间的差值。RMR 为正且最好为较小值 (West et al. 2012)。请参见第 187 页的“RMR 和 SRMR”。

SRMR 标准化均方根残差 (RMR) 为确定模型拟合提供了额外的指导。SRMR 的残差来自观测的协方差与隐含模型的协方差之间的标准化差值。SRMR 为正且最好为较小值 (West et al. 2012)。请参见第 187 页的“RMR 和 SRMR”。

注意：有关 “拟合指数”报表中其他值的说明，请参见第 170 页的 “ “结构化方程模型拟合”报表”。

拟合汇总 显示或隐藏包含模型拟合详细信息的报表。

参数估计值 显示或隐藏包含模型的非标准化参数估计值的报表。

标准化参数估计值 显示或隐藏包含模型的标准化参数估计值的报表。

置信区间 显示或隐藏 “参数估计值”报表和 “标准化参数估计值”报表中的置信区间。

预测刻画器 支持您查看一组预测变量对一组结果变量的条件预期值的影响。选择该选项时，将出现一个窗口，您必须在其中选择一个或多个预测变量和一个或多个结果。预测和 95% 置信区间基于隐含模型的协方差矩阵。有关 “预测刻画器”的详细信息，请参见 《刻画器指南》。

注意：设置窗口中的初始变量列表仅限于与模型一致的变量。例如，“选择预测变量”列表仅包含预测模型中某项的变量，“选择结果”列表仅包含由模型中的其他某个变量预测的变量。选中**显示全部变量**框可查看两个列表中的所有模型变量。

隐含模型的协方差 显示或隐藏包含模型所隐含协方差矩阵的报表。

隐含模型的均值 显示或隐藏包含模型所隐含每个变量均值的报表。

残差 显示或隐藏包含模型的残差矩阵的报表。该矩阵是隐含模型的协方差矩阵与样本协方差矩阵之间的差值。

标准化残差 显示或隐藏包含模型的标准化残差矩阵的报表。

标准化残差热图 显示或隐藏包含模型的标准化残差热图的报表。

RAM 矩阵 显示或隐藏包含网状动作模型 (RAM) 表示法所使用模型矩阵的报表。

估计值的协方差 显示或隐藏包含模型参数估计值的协方差矩阵的报表。

内生变量的 R^2 （仅当模型为递归模型且包含内生变量时才可用。）显示或隐藏包含模型中每个内生变量的 R^2 值的报表。用 1 减去每个内生变量的残差方差与隐含模型的方差之比，即可计算出该值。R^2 值表示模型在内生变量中所解释的方差。内生变量是指在路径图中有一条指向它的路径的变量。

修改指标 支持您显示模型修改指标的全部或部分估计值。这些值可用于确定可以向模型添加哪些参数以改进模型拟合。每个表都按 “卡方”列降序排序。

所有修改指标 显示或隐藏一个表，其中包含所有模型修改指标的估计值。该表包含指示每个估计值的参数类型的一列。

均值的修改指标 显示或隐藏一个表，其中包含均值和截距的模型修改指标的估计值。

载荷的修改指标 显示或隐藏一个表，其中包含载荷参数的模型修改指标的估计值。

回归的修改指标 显示或隐藏一个表，其中包含回归参数的模型修改指标的估计值。

协方差的修改指标 显示或隐藏一个表，其中包含协方差参数的模型修改指标的估计值。

评估测量模型 （仅可用于唯一因子之间没有协方差的确认性因子模型。）显示或隐藏用于量化关于检验和测度的可靠性和有效性的各种统计量和图形，包括指标可靠性、系数 omega 和 H 以及构造有效性矩阵。

构造有效性矩阵有助于确定潜在变量是否在测量您认为它们在测量的内容：

- 下三角元素包含潜在变量相关性。这些元素支持您检查潜在变量之间的相关性有多强，并将其与假设的相关性强度进行比较。
- 上三角元素是潜在变量相关性的平方。这些元素让您能够关注潜在变量之间方差的重叠。在与矩阵中的对角线元素比较时，这些统计量尤其重要。

– 对角线元素包含每个潜在变量提取的平均方差量，相当于每个潜在变量的指标可靠性的平均值。一个好的潜在变量应该在对角线上有很高的值，因为它的指标有足够的系统方差来正确定义它。理想情况下，每个潜在变量的对角线元素应高于其上方和右侧的元素。

构造有效性矩阵的可视化支持您比较对角线元素和上三角元素。

保存列 支持您将基于拟合结构化方程模型的列保存到数据表中。

保存因子得分 （仅当模型中有潜在变量时才可用。）将包含使用回归方法为每个潜在变量计算的因子得分的列保存到数据表中。因子得分在隐藏列中计算，该列也会添加到数据表中。该隐藏列使用 `Estimate Factor Score()` JSL 函数。有关该函数的详细信息，请参见“帮助”>“脚本索引”。

保存 Bartlett 因子得分 （仅当模型中有潜在变量时才可用。）将包含使用 Bartlett 方法为每个潜在变量计算的因子得分的列保存到数据表中。因子得分在隐藏列中计算，该列也会添加到数据表中。该隐藏列使用 `Estimate Bartlett Factor Score()` JSL 函数。有关该函数的详细信息，请参见“帮助”>“脚本索引”。

保存预测公式 （仅当模型中至少有一个内生或因变量时可用。）将包含每个变量观测结果预测值公式的列保存到数据表中。当模型中存在潜在变量时，使用 Bartlett 方法计算的因子得分也会保存到数据表中。

保存观测残差 （仅当模型中至少有一个内生或因变量时可用。）将包含每个变量观测结果残差值的列保存到数据表中。当模型中存在潜在变量时，使用 Bartlett 方法计算的因子得分也会保存到数据表中。

复制模型规格 将当前结构化方程模型规格复制到剪贴板。随后可以将模型规格粘贴到另一个 SEM 平台报表中。

在模型规格中重新调用 将“模型规格”报表中的模型设置为指定的模型。

删除拟合 从报表窗口删除指定的模型报表。

JMP PRO 定制 SEM 路径图

您可以采用多种方式定制路径图。大多数定制选项位于以下一个或两个位置：路径图中的弹出式菜单和“定制关系图外观”窗口。

- “路径图弹出式菜单选项”
- ““定制关系图外观”选项”

JMP PRO 路径图弹出式菜单选项

右击路径图本身时，会出现路径图弹出式菜单。菜单中可用的一组特定选项取决于在关系图中右击的位置，以及右击时是否选择了路径图的任何元素。

选择 包含在 “从列表”和 “至列表”中进行选择、向关系图中添加单向和双向箭头、创建新的潜在变量、隐藏选定项以及显示所有隐藏项等选项。在路径图中选择项之前，该子菜单中的许多选项都未启用。

提示：当路径图中的项被隐藏时，路径图顶部会显示一个 “全部取消隐藏”按钮，该按钮支持您显示所有隐藏的项。

显示 包含显示或隐藏路径图的各种元素的选项。选择**显示默认值**可返回最初显示的一组元素。

注意：“显示默认值”选项不影响 “显示均值 / 截距”选项的设置。

对于拟合模型报表中的路径图，该子菜单还包含用于更改路径图中箭头上显示的估计值的选项。您可以在显示非标准化参数估计值、标准化参数估计值或不显示估计值之间进行选择。

选择该类型的全部项 （仅当右击路径图中的某项时才可用。）选择路径图中与右击的项类型相同的其他所有项。

选择潜在变量组 （仅当右击路径图中的潜在变量时才可用。）选择与所选潜在变量关联的整个项组。若未选择任何潜在变量，则该选项仅选择与右击的潜在变量关联的项组。

重命名变量 （仅当右击路径图中的潜在变量时才可用。）支持您更改路径图中潜在变量的名称。

注意：在路径图中重命名潜在变量时，将在 “从列表”和 “至列表”中更新该名称。

添加回归 （仅当右击路径图中的某个变量时才可用。）支持您添加单向箭头，该箭头表示右击的变量与路径图中的另一个变量之间的回归。选择该选项后，点击您希望箭头指向的变量。

添加协方差 （仅当右击路径图中的某个变量时才可用。）支持您添加双向箭头，该箭头表示右击的变量与路径图中的另一个变量之间的协方差。选择该选项后，点击您希望箭头指向的变量。

定制关系图 启动 “定制关系图外观”窗口。请参见第 177 页的 ““定制关系图外观”选项”。

旋转潜在变量组 以 90 度角为增量顺时针旋转路径图中选定的潜在变量的潜在变量指标。若未选定任何潜在变量，该选项将旋转路径图中的所有潜在变量指标。

布局 支持您为路径图中的项选择两个排列选项之一。

复制关系图 将路径图的图像保存到剪贴板。要保持尽可能高的质量，请将剪贴板图像粘贴为向量图形。

复制关系图属性 将当前路径图属性复制到剪贴板。随后可将属性粘贴到另一个 SEM 路径图中。

粘贴关系图属性 将剪贴板中的路径图属性粘贴到当前 SEM 路径图中。

撤销 撤销对路径图上次进行的更改。

恢复 恢复对路径图上次撤销的更改。

编辑 包含用于 JMP 中图形的标准选项。有关这些选项的详细信息，请参见 《使用 JMP》。

重置布局 将路径图重置为原始设置。

JMP PRO “定制关系图外观”选项

点击“关系图”面板中的“定制”按钮，从红色小三角菜单中选择“定制关系图”，或从路径图弹出式菜单中选择“定制关系图”时，将显示“定制关系图外观”窗口。“定制关系图外观”窗口包含四个面板和一个下拉式菜单，该菜单支持您从两个预设颜色主题中进行选择。

主题预设 包含用于预设颜色主题的选项。您可以在“黑色和白色”主题或“蓝色”主题之间进行选择。选择其中一个可快速更新设置以遵循任一主题。

变量外观 包含用于设置路径图中的填充颜色、边框颜色、文本颜色和项的大小的选项。您可以设置显变量、潜在变量和常量变量的外观，以及这其中每个项的字体。

路径设置 包含用于箭头外观的选项。可以将粗细和透明度设置为固定值，或使其基于每个路径箭头的标准化估计值来设置粗细和透明度。也可以用虚线表示不显著的 p 值，将 α 水平设置为不显著，并更改与箭头关联的颜色和字体。默认情况下，箭头的粗细是固定的，箭头的透明度与标准化参数估计值成正比。

其他设置 包含以下选项：

启用网格 显示或隐藏路径图中的网格，该网格可用于帮助对齐路径图中的各项。

锁定关系图 锁定或解除锁定路径图中各项的位置。当路径图被锁定时，您不能四处拖动项，并且路径图顶部的锁定指示符将突出显示。

显示均值/截距 显示或隐藏路径图中的均值和截距。默认情况下，均值结构不会显示在路径图中。

显示潜在指示符 显示或隐藏路径图中的潜在变量指标。

显示回归 显示或隐藏路径图中的回归箭头。

显示方差 显示或隐藏路径图中的方差箭头。

显示协方差 显示或隐藏路径图中的协方差箭头。

显示等式约束 显示或隐藏路径图中的等式约束。

显示 R 方值 显示或隐藏路径图中每个节点的 R^2 值。

用 R 方填充节点 显示或隐藏路径图中每个节点对应于每个节点的 R^2 值的填充颜色。

预览 包含基于窗口中当前设置的路径图外观预览。

提示：使用“定制关系图外观”窗口底部的**保存至首选项**按钮将窗口中的当前设置保存到平台首选项，以便将来的路径图使用相同的设置。

JMP PRO 结构化方程模型的更多示例

在本例中，您要为 Bollen (1989) 中所述的工业化和政治民主构建一个结构化方程模型，文中使用来自 75 个发展中国家 / 地区的数据。数据表中的变量包括 1960 年和 1965 年的四个民主测度和 1960 年的三个工业化测度。这些变量在数据表每一列中的“注释”列属性中说明。要查看

“注释”列属性，请右击列名，选择 “列信息”，然后在 “列属性”下选择 “注释”。您要构建的结构化方程模型的类型是结构化回归模型。

模型指定过程包含四个主要步骤：创建潜在变量、添加载荷和回归变量、添加协方差项以及对载荷变量设置约束。

1. 选择**帮助 > 样本数据库**，然后打开 Political Democracy.jmp。
2. 选择**分析 > 多元方法 > 结构化方程模型**。
3. 从 Prod60 一直选择到 Legis65 并点击**模型变量**。
4. 点击**确定**。

   “结构化方程模型”报表 “模型规格”分级显示项随即显示。
5. 点击 “视图”面板框中**列表**选项卡。

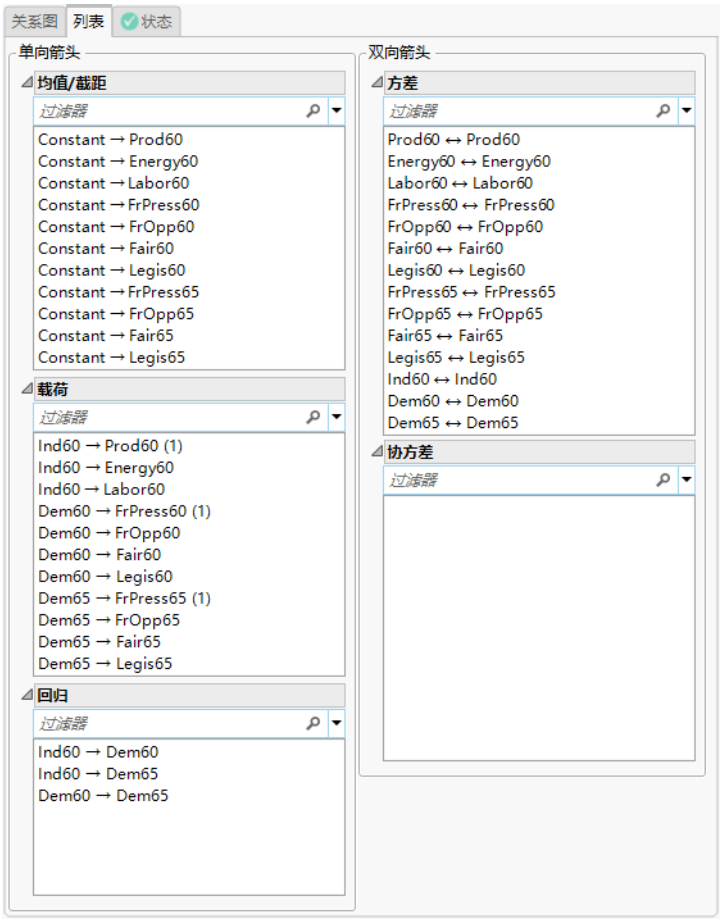
创建潜在变量

6. 在“至列表”中从 Prod60 一直选择到 Labor60，在 “至列表”下方的框中键入 “Ind60”，然后点击添加潜在变量 ⊞ 按钮。
7. 在“至列表”中从 FrPress60 一直选择到 Legis60，在“至列表”下方的框中键入“Dem60”，然后点击添加潜在变量 ⊞ 按钮。
8. 在“至列表”中从 FrPress65 一直选择到 Legis65，在“至列表”下方的框中键入“Dem65”，然后点击添加潜在变量 ⊞ 按钮。

添加载荷和回归变量

9. 在 “自列表”中选择 Ind60，在 “至列表”中选择 Dem60，然后点击单向箭头 → 按钮。
10. 在 “自列表”中选择 Ind60，在 “至列表”中选择 Dem65，然后点击单向箭头 → 按钮。
11. 在 “自列表”中选择 Dem60，在 “至列表”中选择 Dem65，然后点击单向箭头 → 按钮。

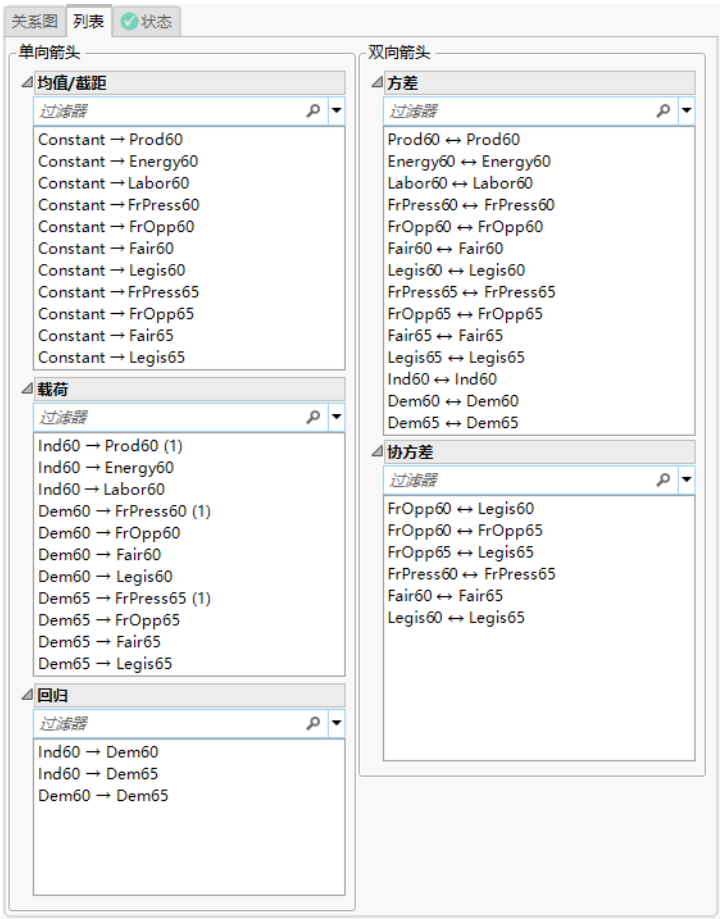
图 8.7 载荷和回归



添加协方差

12. 在“自列表”中选择 FrOpp60，在“至列表”中选择 Legis60 和 FrOpp65，然后点击双向箭头 ↔ 按钮。
13. 在“自列表”中选择 FrOpp65，在“至列表”中选择 Legis65，然后点击双向箭头 ↔ 按钮。
14. 在“自列表”中选择 FrPress60，在“至列表”中选择 FrPress65，然后点击双向箭头 ↔ 按钮。
15. 在“自列表”中选择 Fair60，在“至列表”中选择 Fair65，然后点击双向箭头 ↔ 按钮。
16. 在“自列表”中选择 Legis60，在“至列表”中选择 Legis65，然后点击双向箭头 ↔ 按钮。

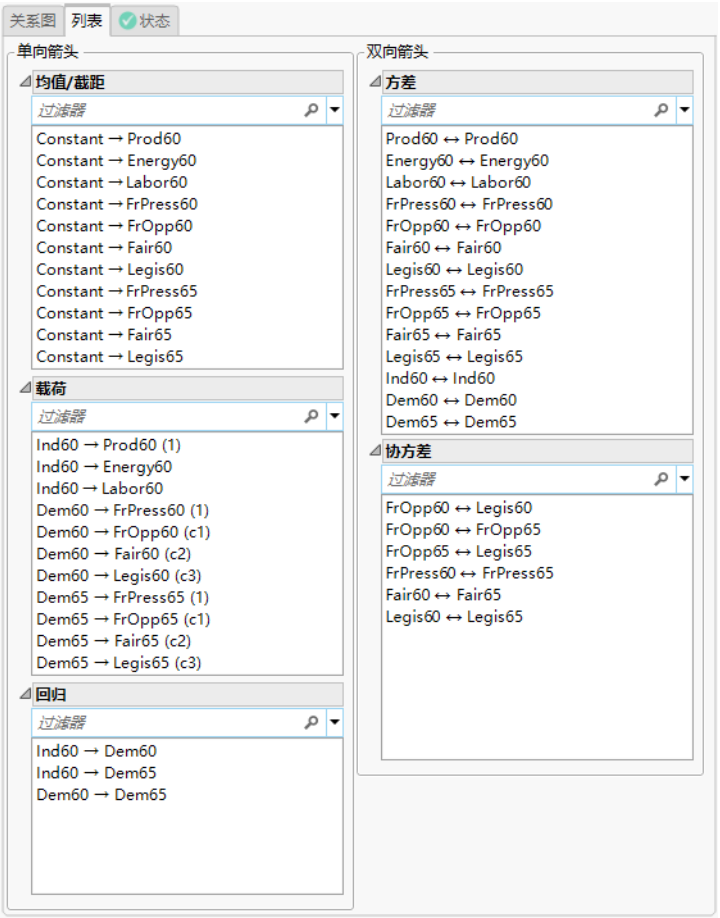
图 8.8 协方差



添加针对载荷的约束

17. 在 “载荷”列表中选择 Dem60->FrOpp60 和 Dem65->FrOpp65，然后点击**设置等于**。
18. 在 “载荷”列表中选择 Dem60->Fair60 和 Dem65->Fair65，然后点击**设置等于**。
19. 在 “载荷”列表中选择 Dem60->Legis60 和 Dem65->Legis65，然后点击**设置等于**。

图 8.9 完成的模型规格



使用字母数字标签指定针对载荷的约束。例如，您可以看到 Dem60->FrOpp60 和 Dem65->FrOpp65 被设置为相等，因为它们都被标记为“c1”。

20. 在“模型名称”下方的文本框中，键入“工业化和政治民主”。
21. 点击**运行**。

图 8.10 结构化方程模型 “拟合汇总”报表

拟合汇总	
梯度收敛	
最大似然	
样本大小	75
带缺失值的行	0
迭代	7
-2 对数似然	3097.6362
参数数目	39
AICc	3264.779
BIC	3266.0182
卡方	40.179489
自由度	38
概率>卡方	0.3738824
CFI	0.9967743
RMSEA	0.0276538
90% 下限	0
90% 上限	0.0870678

“拟合汇总”报表中列出的该模型的卡方统计量为 40.18，自由度为 38。请注意，相应的 p 值为 0.3739，该值不显著。这表明，没有证据可以拒绝模型拟合良好这一原假设。因此，您可以得出结论：该模型对数据拟合良好。

卡方值取决于样本大小，因此，一些拟合良好的模型仍然可以生成显著的卡方值。比较拟合指数 (CFI) 和近似的均方根误差 (RMSEA) 为确定模型拟合提供了额外的指导。这些指数介于 0 和 1 之间。 CFI 值最好大于 0.90， RMSEA 值最好小于 0.10（Browne and Cudeck 1993 ；Hu and Bentler 1999）。在此， CFI 为 0.9968， RMSEA 为 0.0277，这指示拟合极佳。

图 8.11 结构化方程模型 “参数估计值”报表

参数估计值

均值/截距	估计值	标准误差	Wald Z	概率>|Z|
Constant → Prod60	5.0543838	0.0840624	60.126559	<.0001*
Constant → Energy60	4.7921946	0.1732697	27.657432	<.0001*
Constant → Labor60	3.5576898	0.1612318	22.065682	<.0001*
Constant → FrPress60	5.4646667	0.2989114	18.281892	<.0001*
Constant → FrOpp60	4.2564429	0.4389904	9.6959807	<.0001*
Constant → Fair60	6.5631103	0.3939809	16.658446	<.0001*
Constant → Legis60	4.452533	0.3796307	11.728591	<.0001*
Constant → FrPress65	5.1362519	0.3044457	16.870832	<.0001*
Constant → FrOpp65	2.9780741	0.3923504	7.5903426	<.0001*
Constant → Fair65	6.1962639	0.3643999	17.004021	<.0001*
Constant → Legis65	4.0433897	0.3753526	10.772243	<.0001*
载荷	**估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
Ind60 → Prod60	1	.	.	.
Ind60 → Energy60	2.1796569	0.1389146	15.690626	<.0001*
Ind60 → Labor60	1.8182095	0.152128	11.951843	<.0001*
Dem60 → FrPress60	1	.	.	.
Dem60 → FrOpp60	1.1907835	0.1416451	8.4068115	<.0001*
Dem60 → Fair60	1.1745402	0.1197987	9.8042797	<.0001*
Dem60 → Legis60	1.2509806	0.1229262	10.176679	<.0001*
Dem65 → FrPress65	1	.	.	.
Dem65 → FrOpp65	1.1907835	0.1416451	8.4068115	<.0001*
Dem65 → Fair65	1.1745402	0.1197987	9.8042797	<.0001*
Dem65 → Legis65	1.2509806	0.1229262	10.176679	<.0001*
回归	**估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
Ind60 → Dem60	1.471331	0.3914721	3.7584565	0.0002*
Ind60 → Dem65	0.6004747	0.2382837	2.5199993	0.0117*
Dem60 → Dem65	0.8650414	0.0756835	11.429724	<.0001*
方差	**估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
Prod60 ↔ Prod60	0.0813877	0.0196996	4.13144	<.0001*
Energy60 ↔ Energy60	0.1204277	0.0699024	1.7227975	0.0849
Labor60 ↔ Labor60	0.4666599	0.0891233	5.2361183	<.0001*
FrPress60 ↔ FrPress60	1.8546493	0.4569855	4.0584422	<.0001*
FrOpp60 ↔ FrOpp60	7.5813412	1.3449672	5.6368223	<.0001*
Fair60 ↔ Fair60	4.9556729	0.96122	5.1556073	<.0001*
Legis60 ↔ Legis60	3.2244878	0.7417156	4.3473373	<.0001*
FrPress65 ↔ FrPress65	2.3130434	0.4834138	4.7848104	<.0001*
FrOpp65 ↔ FrOpp65	4.9681894	0.8945151	5.554059	<.0001*
Fair65 ↔ Fair65	3.5600372	0.7379206	4.8244178	<.0001*
Legis65 ↔ Legis65	3.3076951	0.7128724	4.6399537	<.0001*
Ind60 ↔ Ind60	0.448599	0.0867473	5.1713335	<.0001*
Dem60 ↔ Dem60	3.8753207	0.8886194	4.3610581	<.0001*
Dem65 ↔ Dem65	0.1644634	0.2333453	0.7048069	0.4809
协方差	**估计值**	**标准误差**	**Wald Z**	**概率>|Z|**
FrOpp60 ↔ Legis60	1.4401257	0.6909886	2.0841528	0.0371*
FrOpp60 ↔ FrOpp65	2.1830097	0.7311078	2.9858931	0.0028*
FrOpp65 ↔ Legis65	1.3717876	0.5781788	2.3726008	0.0177*
FrPress60 ↔ FrPress65	0.5825448	0.3644286	1.5985156	0.1099
Fair60 ↔ Fair65	0.7115828	0.6194143	1.1487994	0.2506
Legis60 ↔ Legis65	0.362804	0.4607898	0.7873525	0.4311

接下来，“回归”下的参数估计值表明 Ind60 对 Dem60 和 Dem65 具有正效应， Dem60 对 Dem65 也具有正效应。因此，Ind60 的较高得分与 Dem60 和 Dem65 较高相关，Dem60 的较高得分与 Dem65 的较高得分相关。参数估计值相应的 p 值显示在 “回归”下方。 3 个回归参数在 $\alpha = 0.05$ 水平下均显著。因此，您可以得出结论，潜在变量之间存在非零关系。

JMP PRO “结构化方程模型”平台的统计详细信息

JMP PRO 拟合汇总测度的统计详细信息

JMP PRO AICc 和 BIC

AICc 和 BIC 的公式定义如下：

$$\text{AICc} = -2\text{logL} + 2k + \frac{2k(k+1)}{n-(k+1)}$$

$$\text{BIC} = -2\text{logL} + k\ln(n)$$

其中：

- -2logL 是负对数似然的两倍。
- n 是样本大小。
- k 是参数个数。

有关“模型比较”报表中基于似然的测度的详细信息，请参见《拟合线性模型》。

JMP PRO CFI

比较拟合指数 (CFI) 计算如下：

$$\text{CFI} = \frac{max(\chi_0^2 - df_0, 0) - max(\chi_{min}^2 - df_{min}, 0)}{max(\chi_0^2 - df_0, 0)}$$

其中：

- χ_0^2 是独立模型的卡方统计量。
- df_0 是独立模型的自由度。
- χ_{min}^2 是拟合模型的卡方统计量。
- df_{min} 是拟合模型的自由度。

有关 CFI 的详细信息，请参见 Bentler (1990)。

JMP PRO TLI

Tucker-Lewis 指数 (TLI) 定义如下：

$$\text{TLI} = \frac{\frac{\chi_0^2}{df_0} - \frac{\chi_{min}^2}{df_{min}}}{\frac{\chi_0^2}{df_0} - 1}$$

其中：

- χ_0^2 是独立模型的卡方统计量。
- df_0 是独立模型的自由度。
- χ_{min}^2 是拟合模型的卡方统计量。
- df_{min} 是拟合模型的自由度。

详细信息，请参见 West et al.(2012)。

JMP PRO NFI

Bentler-Bonett 规范拟合指数 (NFI) 定义如下：

$$\text{NFI} = \frac{\chi_0^2 - \chi_{min}^2}{\chi_0^2}$$

其中：

- χ_0^2 是独立模型的卡方统计量。
- χ_{min}^2 是拟合模型的卡方统计量。

详细信息，请参见 West et al.(2012)。

JMP PRO 修正的 GFI 和修正的 AGFI

修正的拟合优度指数 （修正的 GFI）定义如下：

$$\text{修正的 GFI} = \frac{p}{p + 2\left(\frac{\chi_{min}^2 - df_{min}}{n-1}\right)}$$

其中：

- χ^2_{min} 是拟合模型的卡方统计量。
- $df_{\min}$ 是拟合模型的自由度。
- p 是拟合模型的观测变量数。
- n 是样本大小。

修正的拟合优度指数 （修正的 AGFI）定义如下：

$$\text{修正的 AGFI} = 1 - \frac{p^*}{df_{min}}(1 - \text{修正的}GFI)$$

其中：

- p^* 是协方差矩阵中唯一条目数和观测变量的均值向量。
- $df_{\min}$ 是拟合模型的自由度。

详细信息，请参见 West et al.(2012)。

JMP PRO RMSEA

近似的均方根误差 (RMSEA) 计算如下：

$$\text{RMSEA} = \sqrt{\frac{\max(\chi^2_{min} - df_{min}, 0)}{n \times df_{min}}}$$

其中：

- n 是样本大小。
- $df_{\min}$ 是拟合模型的自由度。
- χ^2_{min} 是拟合模型的卡方统计量。

利用非中心卡方分布的累积分布函数 $\Phi(x|\lambda, d)$ 计算 RMSEA 的置信限。90% 置信限计算如下：

$$\text{下限} = \sqrt{\frac{\lambda_L}{n \times df_{min}}}$$

$$\text{上限} = \sqrt{\frac{\lambda_U}{n \times df_{min}}}$$

其中：

- λ_L 满足 $\Phi(\chi^2_{min}|\lambda_L, df_{\min}) = 0.95$。
- λ_U 满足 $\Phi(\chi^2_{min}|\lambda_U, df_{\min}) = 0.05$。

详细信息，请参见 Maydeu-Olivares et al.(2017) 中的表 J.1a、J.1b、J.6a 和 J.6b。

JMP PRO RMR 和 SRMR

RMR 和 SRMR 的公式定义如下：

$$\text{RMR} = \sqrt{\frac{1}{b}\left[\sum_{i}^{p}\sum_{j}^{i}(s_{ij}-\hat{\sigma}_{ij})^2+\sum_{i}^{p}(\bar{x}_i-\hat{\mu}_i)^2\right]}$$

$$\text{SRMR} = \sqrt{\frac{1}{b}\left[\sum_{i}^{p}\sum_{j}^{i}\frac{(s_{ij}-\hat{\sigma}_{ij})^2}{s_{ii}s_{jj}}+\sum_{i}^{p}\frac{(\bar{x}_i-\hat{\mu}_i)^2}{s_{ii}}\right]}$$

其中：

- p 是显变量数。
- b 是观测变量的协方差矩阵和均值向量中的唯一条目数：

$$b = \frac{p(p+1)}{2}+p$$

- s_{ij} 是输入协方差矩阵的第 (i, j) 个元素。
- $\hat{\sigma}_{ij}$ 是预测协方差矩阵的第 (i, j) 个元素。
- $\bar{x}_i$ 是样本均值向量的第 i 个元素。
- $\hat{\mu}_i$ 是向量预测均值的第 i 个元素。

有关详细信息，请参见 SAS Institute Inc.(2020b) 中的“CANDISC 过程”一章。

第 9 章

因子分析

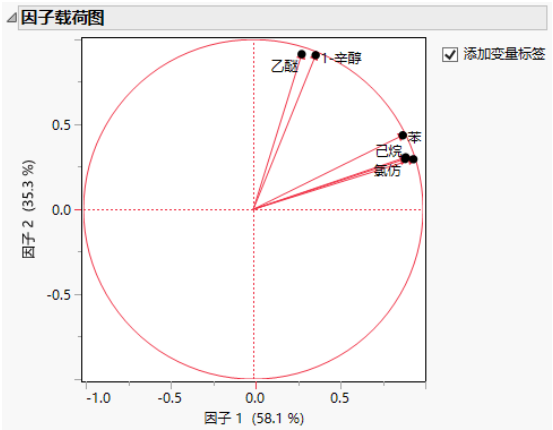
标识数据中的潜在变量

因子分析旨在用少量（不可观测）的潜在变量或因子描述可观测变量。因子分析也称为公因子分析和探索性因子分析。这些因子可以定义为观测变量的线性组合（加上误差）。旨在解释观测变量中的**共有**变异。因子分析的目的是通过未观测到的因子来发现针对观测变量的有意义的解释，除此之外我们还可以减少变量数。

因子分析在许多领域中都有广泛的应用，其发源于心理学、社会学和教育学。在这些领域中，因子分析有助于理解如何通过潜在模式和结构来解释表象行为。例如，用来衡量参与户外活动、爱好、锻炼和旅游的测度可能全都与可描述为“个性活跃/不活跃”的因子相关。

在您需要探索或解释数据中的潜在模式和结构时可使用因子分析。还可考虑通过因子分析借助少量的潜在变量来汇总变量中的信息。

图 9.1 旋转的因子载荷



目录

“因子分析”平台概述 191
“因子分析”平台的示例 191
启动“因子分析”平台 194
“因子分析”报表 195
模型启动 196
旋转方法 197
“因子分析”平台选项 198
因子分析模型拟合选项 199

“因子分析”平台概述

因子分析根据较小数量的无法观测的因子对一组可观测的变量建模。构造这些因子可解释观测变量之间的相关性或协方差。因子旋转用于更改因子的参考轴，使其更加容易解释。

考虑有十个观测变量的情形：$X_1, X_2, \ldots, X_{10}$。假定您想要根据两个潜在因子 F_1 和 F_2 对这十个变量建模。为方便起见，假定这些因子之间不相关，而且每个因子的均值为 0，方差为 1。您想要得到的模型形式如下：

$$X_i = \beta_{i0} + \beta_{i1}F_1 + \beta_{i2}F_2 + \varepsilon_i$$

由此判定：$\text{Var}(X_i) = \beta_{i1}^2 + \beta_{i2}^2 + \text{Var}(\varepsilon_i)$。其中可归因于因子的 X_i 的方差部分为 $\beta_{i1}^2 + \beta_{i2}^2$，我们称之为公共方差或**公因子方差**。而剩余方差 $\text{Var}(\varepsilon_i)$ 是特殊方差，被视为 X_i 所特有的特定和误差方差的组合。

该平台为相关性或协方差矩阵的特征值提供了一张陡坡图。您可以根据陡坡图来确定要提取的因子数。该平台的默认因子数是超过 1 的特征值数。

该平台提供两种因子分解方法来估计该模型的参数：主轴和最大似然。通过两个“先验公因子方差”选项可估计每个变量的公因子对公差贡献的比例。这些选项针对相关性（或协方差）矩阵的对角线有不同的假设前提。“主成分”选项从相关性矩阵（对角线元素为 1）或协方差矩阵（对角线元素为变量的方差）出发进行后续分析，“公因子分析”选项将对角线元素设置为多重相关性的平方。这些值反映与其他变量分享的变异比例。

可利用因子旋转来支持对提取因子的解释。“因子分析”平台提供了多种旋转方法，其中包含正交旋转和斜交旋转。

与考虑公共方差的因子分析相比，主成分分析解释观测变量的总方差。请参见第 49 页的“主成分”一章。

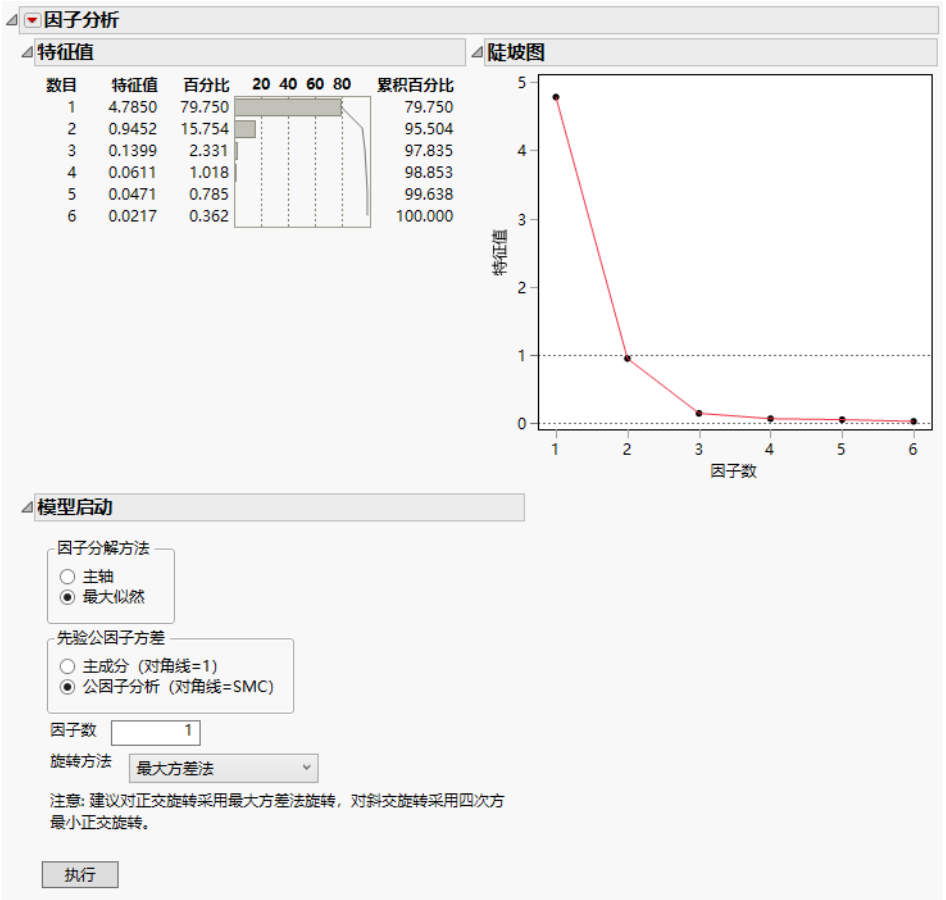
有关因子分析的详细信息，请参见 Jöreskog (1977) 或 Cudeck and MacCallum (2007)。

“因子分析”平台的示例

本例使用 Solubility.jmp 样本数据表提取由六种溶剂解释的两个因子。

1. 选择**帮助** > **样本数据库**，然后打开 Solubility.jmp。
2. 选择**分析** > **多元方法** > **因子分析**。
3. 从 **1- 辛醇**一直选到**己烷**，然后点击 **Y，列**。
4. 点击**确定**。

图 9.2 初始 “因子分析”报表



陡坡图的弯肘部指示包含两个因子的模型很合适。

5. 在 “模型启动”分级显示项中，做出以下选择：
   - 因子分解方法= **最大似然**
   - 先验公因子方差= **公因子分析**
   - 因子数 = 2
   - 旋转方法= **最大方差法**
6. 点击**执行**。

图 9.3 “因子分析”报表

因子分析

特征值 陡坡图

模型启动

基于相关性的因子分析 带 2 个因子: 最大似然/ 最大方差法

最终公因子方差估计值

每个因子解释的方差

显著性检验

检验	自由度	卡方	概率>卡方
H0: 无公因子。 HA: 至少一个公因子。	15	691.106	<.0001*

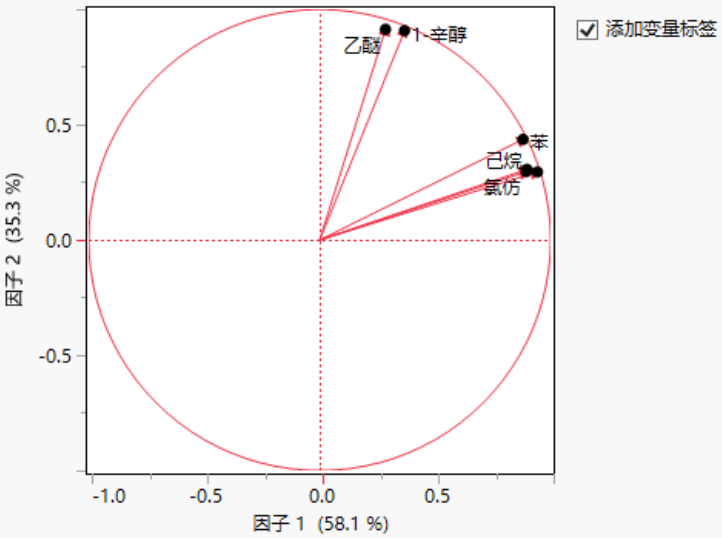
检验	自由度	Criterion	卡方	概率>卡方
H0: 2 个因子即足够。 HA: 需要更多因子。	4	0.389	26.031	<.0001*

拟合测度

拟合测度	拟合指数
不带 Bartlett 校正的卡方	27.654
AIC	19.654
BIC	10.547
Tucker 和 Lewis 指数	0.878
近似的均方根误差	0.289

旋转的因子载荷

因子载荷图



该报表列出公因子方差估计值、方差估计值、显著性检验、拟合测度、旋转的因子载荷和因子载荷图。“旋转的因子载荷”和“因子载荷图”表明：“因子 1”与**四氯化碳-氯仿-苯-己烷**这

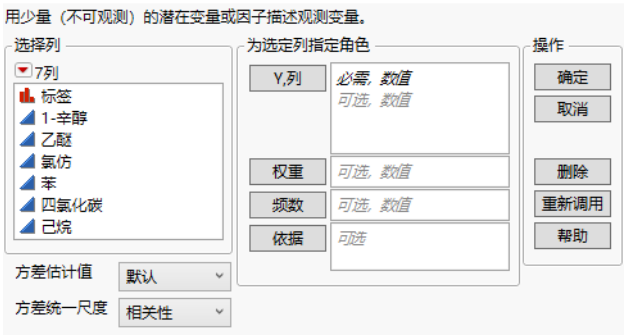
一组变量相关，“因子 2”与**乙醚–1-辛醇**这一组变量相关。请参见第 199 页的“因子分析模型拟合选项”，了解有关报表中所显示信息的详细信息。

提示：点击“因子载荷图”中的点可以选择和移动标签。点击右下角可增大图的大小，以便更容易查看标签。

启动“因子分析”平台

通过选择**分析 > 多元方法 > 因子分析**启动 “因子分析”平台。

图 9.4 “因子分析”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 要分析的列。这些列的数据类型必须为 “数值”。

权重 包含数据表中每个观测的权重值的列。仅当行值大于零时才在分析中包含该行。

频数 为分析中的每行分配一个频数。该选项适用于汇总数据。

依据 为 “依据”变量的每个水平生成单独报表。若指定了多个 “依据”变量，将为 “依据”变量水平的每种可能组合生成单独的报表。

方差估计 列出用于估计分析的方差 - 协方差矩阵的方法。有关这些方法的详细信息，请参见第 34 页的 ““多元”报表”。

方差统一尺度 列出用于执行因子分析的统一尺度方法。

相关性 支持相关性分析的默认方法。

协方差 支持针对加权相关性矩阵的分析，其中的权重是变量方差。

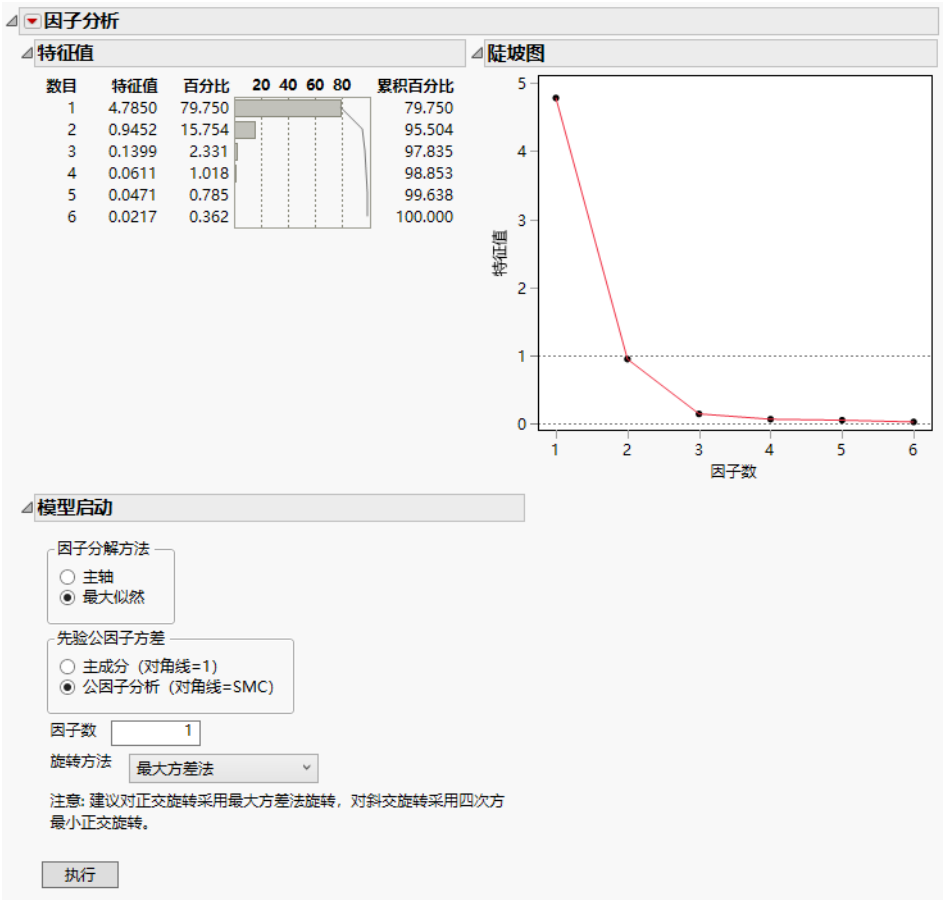
未统一尺度 支持针对已经中心化或统一尺度的变量执行分析。

“因子分析”报表

初始“因子分析”报表显示了“特征值”和“陡坡图”。“特征值”从主成分分析获得。“陡坡图”是对这些特征值进行绘制的图形。“模型启动”中的初始因子数等于超过 1.0 的特征值的数目。您可以更改要提取的因子数。

可借助陡坡图的指导来选择因子数。陡坡图平坦之前的特征值数提供因子数的上限。对于本例，陡坡图建议两个成分（因子）。

图 9.5 “因子分析”报表

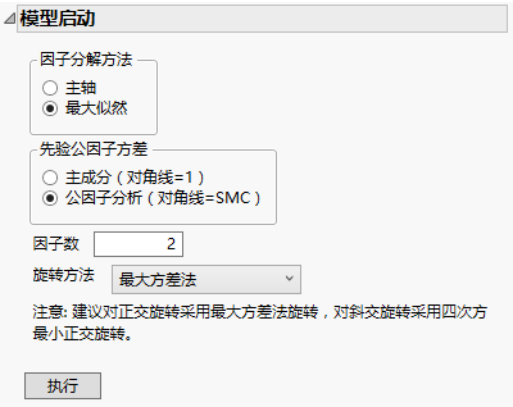


该“特征值”表显示第一个特征值解释了变异的 79.75%，第二个特征值解释了 15.75%。因此，前两个特征值共解释了总变异的 95.50%。第三个特征值仅解释了 2.33% 的变异，而其余特征值的贡献可忽略不计。尽管**因子数**框最初设置为 1，该分析建议提取 2 个因子才是合理的。

模型启动

从“模型启动”控制面板配置“因子分析”。点击**执行**获取因子分析的结果。

图 9.6 模型启动



因子分解方法 定义提取因子的方法：

主轴 针对简化的相关性或协方差矩阵（其中矩阵的对角线被变量的公因子方差估计值取代）执行特征值分解。该方法计算效率高，但它不提供假设检验。

最大似然 支持您检验关于公因子数目的假设，并且获取模型拟合统计量。

注意：最大似然方法要求正定相关性矩阵。若相关性矩阵不是正定矩阵，请选择“主轴”方法。

先验公因子方差 定义公因子对每个变量的方差贡献比例的估计方法。

主成分（对角线 = 1） 将所有公因子方差设置为 1，指示每个变量的方差 100% 都是由所有因子解释的。

提示：在“因子分解方法”设置为“主轴”时使用该选项将生成主成分分析。

公因子分析（对角线 = SMC） 将公因子方差设置为多重相关的平方 (SMC) 系数。对于给定的变量，SMC 即为该变量对其他所有变量作回归的 R 方。

因子数 指定从分析中提取的因子数。默认值是大于或等于 1.0 的特征值数。您可以设置因子数至少为 1 且不超过变量数。

旋转方法 定义旋转方法。默认方法为“最大方差法”。请参见第 197 页的“旋转方法”，了解可用旋转方法的说明。

旋转方法

使用旋转可更改因子的参考轴，从而使因子更容易解释。旋转应用于从数据中提取的因子。旋转方法基于各种复杂性或简便性函数。有关旋转的详细信息，请参见 SAS Institute Inc.(2020c) 中的“FACTOR 过程”一章、Browne (2001) 或 Frank and Todeschini (1994)。

初始提取之后的因子彼此之间是不相关的。若因子是通过正交变换旋转的，则旋转之后的因子也不相关。若因子是通过斜交变换旋转的，则旋转之后的因子就会彼此相关。斜交旋转往往生成比正交旋转更容易解释的因子。不过因子相关后，在考察对变量的解释时，很难衡量各因子的重要性。

正交旋转方法

最大方差法 将某个因子在所有变量上的载荷平方的方差之和最大化。这种公共方法导致每个变量在每个因子上都有或小或大的载荷。（$\gamma = 1$ 的直交旋转法。）

双四次幂极大法 最大方差法和四次方最大正交旋转法的等权解。（$\gamma = 0.5$ 的直交旋转法。）

相等最大值法 最大方差法旋转和四次方最大正交旋转之间的加权解。（$\gamma = N/2$ 的直交旋转法，其中 N = 因子数。）

Parsimax 因子法 旨在将因子复杂性最小化的一种解决方案。该方法可能导致交叉载荷，因为算法中未考虑变量复杂性。（$\gamma = N$ 的直交旋转法，其中 N = 因子数。）

直交旋转法 一种常规加权旋转方法，其中的权重用 γ 表示。许多特定正交旋转方法都是带有特定 γ 的直交旋转法。

Parsimax 平衡变量与因子复杂性。（$\gamma = (I(N-1))/(I+N-2)$ 的直交旋转法，其中 I = 项数， N = 因子数。）

四次方最大正交旋转 将解释每个变量所需的因子数最小化。（$\gamma = 1$ 的直交旋转法。）

斜交旋转方法

双四次幂极小法 将协方差比最小化的旋转方法 （$\tau = 0.5$ 的斜交转轴法。）

协方差极小法 “斜交最大方差法”旋转。（$\tau = 1$ 的斜交转轴法。）

斜交双四次幂极大法 “斜交双四次幂极大法”旋转。

斜交相等最大值法 “斜交相等最大值法”旋转。

斜交 Parsimax 因子法 “斜交 Parsimax 因子法”旋转。

斜交转轴法 一种常规加权斜交旋转方法，其中的权重用 τ 表示。许多特定斜交旋转方法都是带有特定 τ 的 “斜交转轴法”旋转。

斜交 Parsimax 法 “斜交 Parsimax 法”旋转

四次最大正交旋转法 “斜交四次最大正交旋转法”旋转，等效于“四次方最小正交旋转”方法。

方差最大旋转法 “斜交最大方差法”旋转。

四次方最小正交旋转 “斜交四次方最小正交旋转”旋转，等效于“斜交四次方最大正交旋转法”（$\tau = 0$ 的斜交转轴法。）

斜交旋转法 两步旋转，首先执行最大方差法旋转，然后使用 Procrustes 旋转获得简单结构。这是一种计算高效的方法，可作为斜交转轴法的备选方法。

“因子分析”平台选项

“因子分析”红色小三角菜单包含以下选项：

特征值 显示或隐藏原始相关性矩阵、协方差矩阵或未统一尺度的矩阵的特征值表。该表包含每个特征值所表示的总方差百分比、演示贡献百分比的条形图，以及每个后续特征值所贡献的累积百分比。大于等于 1.0 的特征值的数目可指导您选择合适的因子数进行分析。

陡坡图 显示或隐藏成分（因子）数 - 特征值图。该图也可用作附加指导，用来确定提供了最大方差贡献率的因子数。标绘线变得平坦处的临界点可用作合适的因子数以进行分析。

Kaiser-Meyer-Olkin 检验 显示或隐藏 Kaiser-Meyer-Olkin (KMO) 检验的结果。该检验是一个方差比例指标，可能是公共方差，而该方差可能是由于内在因子导致的。该检验统计量针对每个单独变量和所有变量的集合计算，在 JMP 中称为抽样适当性测度 (MSA)。以下值指示变量是否适用于因子分析：

- 0.00 到 0.49 不可接受
- 0.50 到 0.59 较差
- 0.60 到 0.69 普通
- 0.70 到 0.79 中等
- 0.80 到 0.89 卓越
- 0.90 到 1.00 极好

注意：若相关性矩阵是奇异的，则 KMO 检验不可用。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

因子分析模型拟合选项

“因子分析模型拟合”红色小三角菜单包含以下选项：

先验公因子方差 （仅适用于“公因子分析”。）显示或隐藏每个变量的公因子方差的初始估计值。对于给定的变量，该估计值是多重相关平方 (SMC) 系数，即该变量对其他所有变量构建回归模型得到的 R 方。

特征值 （仅适用于“公因子分析”。）显示或隐藏简化相关性矩阵的特征值以及这些特征值解释的公共方差百分比。简化相关性矩阵是对角线元素被公因子方差估计值取代的相关性矩阵，特征值表明因子所解释的公共方差。“累积百分比”可超过 100%，因为简化相关性矩阵不一定是正定矩阵，可以包含负特征值。

请注意，该表指出了进行后续分析保留的因子数。

未排序和未旋转的因子载荷 显示或隐藏在排序和旋转之前的因子载荷矩阵。

未旋转的因子载荷 显示或隐藏旋转之前的因子载荷矩阵。因子载荷测量公因子对变量的影响。由于未旋转的因子是正交的，因此该因子载荷矩阵是变量与因子之间的相关性矩阵。载荷的绝对值越接近 1，因子对变量的影响就越大。

使用滑块或输入相应值来**隐藏小于该值的绝对载荷值**（该值即表中的指定值）。隐藏的值会根据**文本变暗**所指定的设置相应地变暗。

使用**文本变暗**滑块或输入相应值，控制绝对值比**隐藏小于该值的绝对载荷值**的指定值小的因子载荷值的字体透明度梯度。

图 9.7 带有文本变暗控件的未旋转的因子载荷

注意：“未旋转的因子载荷”矩阵经过排序，因此与同一因子关联的变量会显示在彼此上下。

旋转矩阵 显示或隐藏用于对因子载荷图和因子载荷矩阵进行旋转的值。

因子间相关性 （仅适用于斜交旋转。）显示或隐藏因子间的相关性矩阵。

目标矩阵 （仅适用于“斜交旋转法”旋转。）显示或隐藏最大方差法因子模式要旋转到的矩阵。

因子结构 （仅适用于斜交旋转。）显示或隐藏变量与公因子之间的相关性的矩阵。

最终公因子方差估计值 显示或隐藏拟合因子模型后得到的公因子方差估计值。若因子是正交的，变量的最终公因子方差估计值等于该变量的因子载荷平方和。

标准得分系数 显示或隐藏将旋转后的因子保存至源数据表时用于估计因子得分的乘数表。

每个因子解释的方差 （仅适用于正交旋转。）显示或隐藏每个旋转因子解释的方差、公共方差的百分比和累积百分比。

每个因子解释的方差，忽略其他因子 （仅适用于斜交旋转。）显示或隐藏每个旋转因子解释的方差和公共方差百分比（忽略其他因子）。

显著性检验 （仅适用于“最大似然”因子分解方法。）提供两个卡方检验的结果。

第一个检验的 H_0：无公因子。该原假设指出没有任何公因子可以解释变量之间的交互相关。该检验为 Bartlett 球形检验，其原假设为：因子的相关性矩阵是一个单位矩阵 (Bartlett, 1954)。

第二个检验的 H_0：N 个因子足够多，其中的 N 是指定的因子数。拒绝该原假设即表明需要更多因子来解释变量之间的交互相关 (Bartlett, 1954)。准则即对数似然目标函数值。

拟合测度 （仅适用于“最大似然”因子分解方法。）显示或隐藏拟合测度：未经 Bartlett 修正的卡方、 AIC、 BIC、 Tucker-Lewis 指数以及近似的均方根误差。

因子得分测度 显示或隐藏因子得分确定性测度：多重 R、多重 R 平方和最小相关性。这些测度用于评估因子得分对于二次分析是否有用。

未排序和旋转的因子载荷 显示或隐藏旋转之后的未排序因子载荷矩阵。

旋转的因子载荷 显示或隐藏旋转之后的因子载荷矩阵。若为正交旋转，这些值为变量与旋转因子之间的相关性。

使用滑块或输入相应值来**隐藏小于该值的绝对载荷值**（该值即表中的指定值）。隐藏的值会根据**文本变暗**所指定的设置相应地变暗。

使用**文本变暗**滑块和值控制该表的字体透明度梯度。值越小，绝对值比指定的**隐藏小于该值的绝对载荷值**小的因子值的字体就越透明。

图 9.8 带有文本变暗控件的旋转的因子载荷

注意：“旋转的因子载荷”矩阵经过排序，因此与同一因子关联的变量会显示在彼此上下。

因子载荷图 显示或隐藏旋转的因子载荷图。对两个以上因子建模时，载荷图是多个图的矩阵。

得分图 显示或隐藏估计的因子得分的散点图。对两个以上因子建模时，得分图是多个图的矩阵。

完成补缺的得分图 （仅适用于存在缺失值的情况。）显示或隐藏带有缺失值的补缺值的估计因子得分的散点图。

显示选项 支持您显示或隐藏载荷图上的箭头。

保存旋转因子 将旋转因子得分和公式保存到数据表中。

注意：该公式无法计算包含缺失值的行。

删除拟合 从“因子分析”报表中删除拟合模型结果。该选项支持您更改“模型启动”配置以生成新的报表。

第 10 章

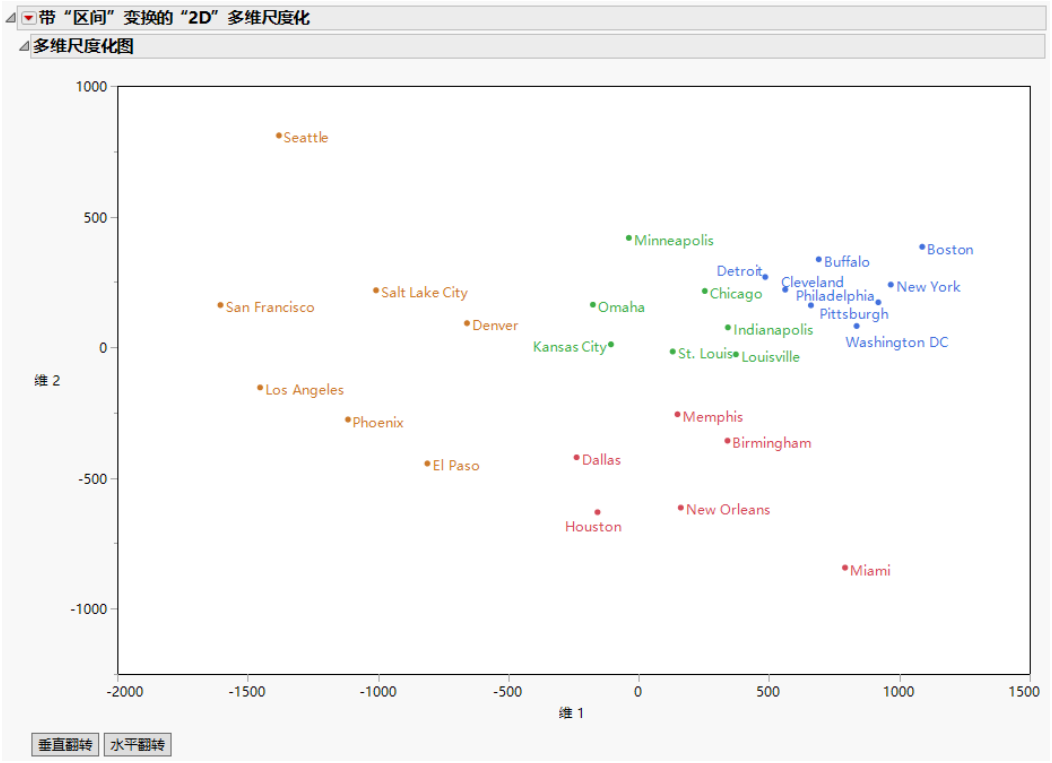
多维尺度化

直观表示一组对象间的邻近性

“多维尺度化 (MDS)”是一种方法，用于分析一组对象间的邻近性（相似性、相异性或距离），将它们之间的模式通过图形直观呈现出来。例如，考虑到城市之间的距离矩阵，MDS 可用于生成二维城市地图。

“多维尺度化”常用在消费者研究中，研究人员收集了关于品牌、口味或其他产品特性的感知测度数据。在其他很多领域，当人们想要根据一组特性（或邻近性）直观展示对象之间的邻近性时，MDS 也同样适用。

图 10.1 “多维尺度化”示例



目录

“多维尺度化”平台概述 205
“多维尺度化”示例 205
启动“多维尺度化”平台 208
“多维尺度化”报表 209
多维尺度化图 209
Shepard 图 209
拟合详细信息 209
“多维尺度化”平台选项 210
Waern 链接 210
“多维尺度化”平台的更多示例 211
“多维尺度化”平台的统计详细信息 213
Stress 213
变换 213
特性列表格式 214

“多维尺度化”平台概述

“多维尺度化”平台生成一组对象之间的邻近性图。该图可用于直观探索数据集内的结构。MDS 是一种多元方法，用于分析一组对象间的邻近性（相似性或距离），以较少的维将它们之间的关系进行可视化展现。MDS 适用于距离矩阵。MDS 图的坐标通过最小化 Stress 函数（实际邻近性和预测邻近性的差异）来获取。

距离一词可指一种物理距离（比如城市之间的距离）测度。更多情况下，距离只是一种客观评估而不是精确测量。邻近性可测量不同品牌产品之间的感知相似性、犯罪率相关性或样本国家/地区的经济相似性。距离还可称为邻近性或相似性（相异性）。若数据作为特性列表提供，则首先从特性列表入手构造距离矩阵。

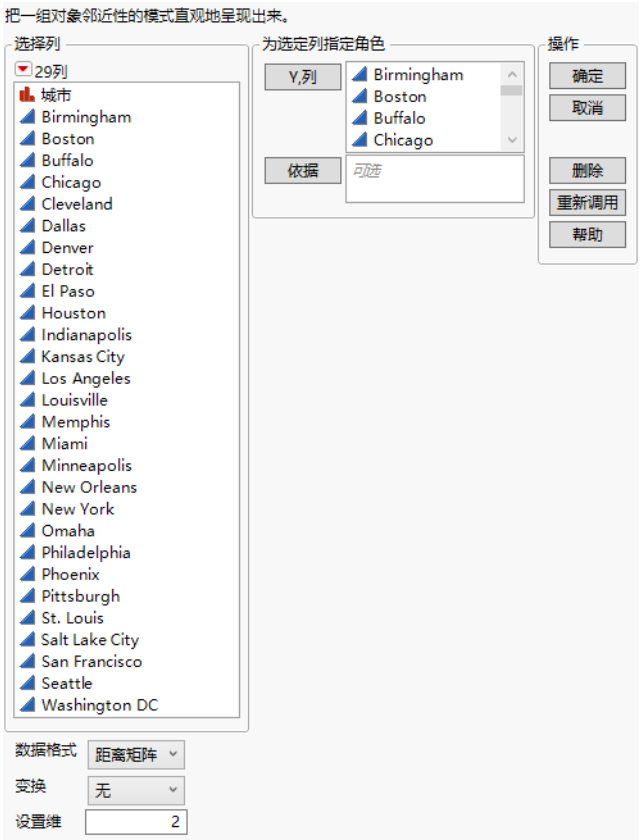
有关多维尺度化的详细信息，请参见 Borg and Groenen (2005) 或 Jackson (2003)。

“多维尺度化”示例

本例使用 Flight Distances.jmp 样本数据表，该表是美国 28 个城市之间的飞行距离的距离矩阵。您可以基于数据表中两两城市间的距离使用 MDS 构造二维城市地图。

1. 选择**帮助 > 样本数据库**，然后打开 Flight Distances.jmp。
2. 选择**分析 > 多元方法 > 多维尺度化**。
3. 从 Birmingham 一直选到 Washington DC，然后点击 **Y，列**。

图 10.2 完成的 “多维尺度化”启动窗口



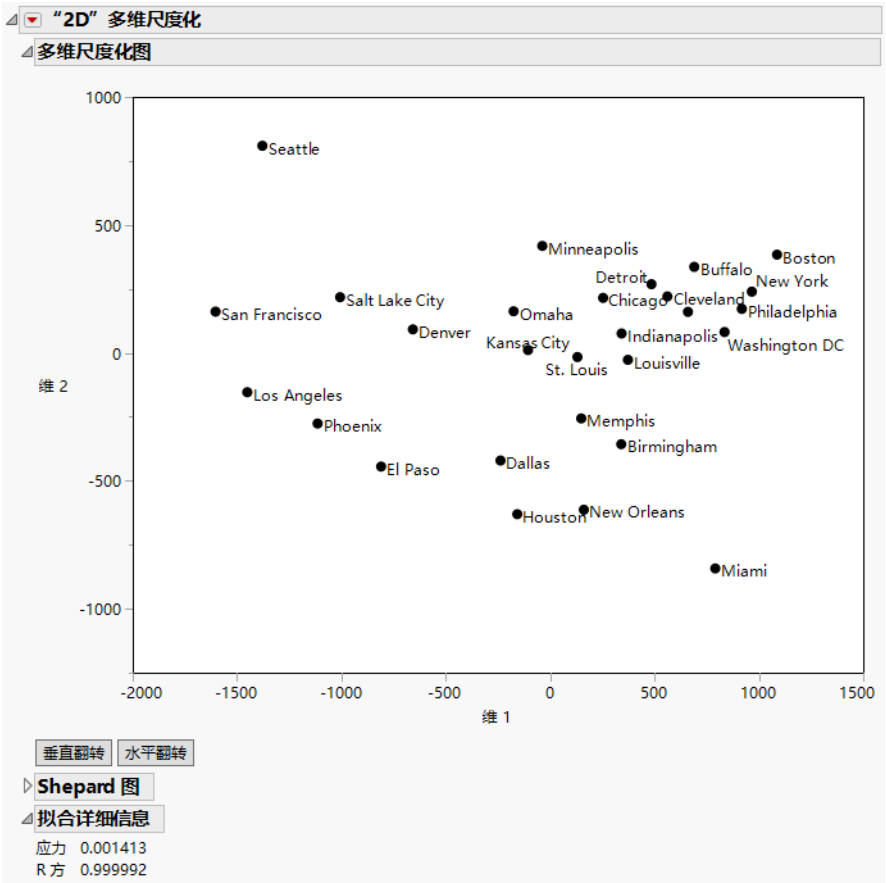
4. 点击**确定**。

   在 “多维尺度化图”中，悬停在数据点上方以查看行号或行标签。在接下来的 7 步中要对 MDS 图执行添加标签和旋转操作。

5. 选择 Flight Distances 数据表。
6. 右击列**城市**，然后选择**添加标签 / 撤销标签**。
7. 选择**行 > 行选择 > 选择所有行**。
8. 选择**行 > 添加标签 / 撤销标签**。
9. 选择 “多维尺度化图”。
10. 点击**垂直翻转**按钮。
11. 点击**水平翻转**按钮。

    “垂直翻转”和“水平翻转”按钮支持您更改 MDS 图的方向。MDS 结果不受方向影响。若方向在结果中是已知的，如物理位置，则您可能需要旋转或翻转您的图形。

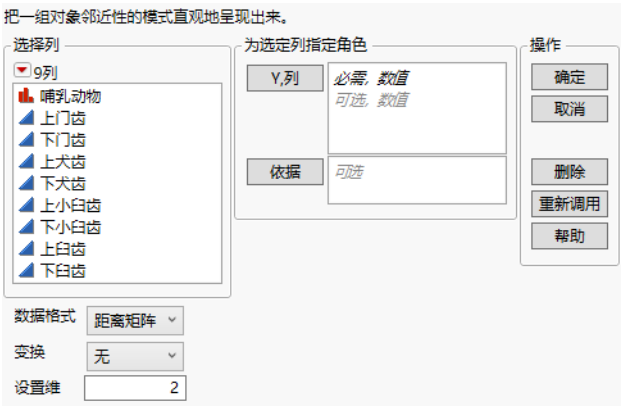
图 10.3 多维尺度化图


“2D” 多维尺度化
多维尺度化图
1000
500
0
-500
-1000
维 2
-2000
-1500
-1000
-500
0
500
1000
1500
维 1
Seattle
Minneapolis
Boston
Buffalo
Detroit
New York
Salt Lake City
Chicago
Cleveland
San Francisco
Omaha
Philadelphia
Denver
Indianapolis
Kansas City
Washington DC
St. Louis
Louisville
Los Angeles
Phoenix
Memphis
Birmingham
El Paso
Dallas
Houston
New Orleans
Miami
垂直翻转
水平翻转
Shepard 图
拟合详细信息
应力 0.001413
R 方 0.999992

启动“多维尺度化”平台

通过选择**分析** > **多元方法** > **多维尺度化**启动 “多维尺度化”平台。

图 10.4 “多维尺度化”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 要分析的列。这些列的数据类型必须为 “数值”。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若指定了多个 “依据”变量，将为 “依据”变量水平的每种可能组合生成单独的报表。

注意：使用距离矩阵时，“依据”变量要求为 “依据”变量的每个水平生成完整矩阵。

数据格式 MDS 支持两种数据格式：

距离矩阵 完全对称、下或上三角矩阵，其中的行数等于列数。对角线元素可以为零或缺失。

特性列表 包含质量测度或对象特征的一组列。对象通常在列中命名。对象列不用在分析中，但用作 MDS 图上数据点的标签。

变换 支持的变换包括 “比”、“区间”和 “有序型”。

无 不使用变换。

比 数据从小到大排序，值间差异有意义，并且尺度具有真实的零。用来对 MDS 图进行尺度变换。

区间 数据从小到大排序，并且值间差异有意义。用来对 MDS 图进行尺度变换和偏移。

有序型 数据从小到大排序。用于有序型数据。

设置维 用于直观表示对象间的邻近性的维数。通常使用二维或三维。三维以上的直观表示会变得较为复杂。

注意：选定的维可介于 1 到 n - 1 之间，其中 n = 对象数，否则维设置为 2。

“多维尺度化”报表

初始“多维尺度化”报表显示以下报表：“多维尺度化图”、“Shepard 图”和“拟合详细信息”。若在启动窗口中为拟合指定了三维或多维，“多维尺度化图”则提供相应的控件选项供您选择查看相应的维。

MDS 图上距离相近的对象具有类似的特征。向图中添加标签和颜色有助于发现类似组。Shepard 图和拟合汇总统计量提供一些测度，用于衡量 MDS 图在多大程度上反映了对象间的邻近性。

多维尺度化图

MDS 图将多维尺度化结果在二维空间进行展示。该图下方有两个按钮，分别用于在垂直和水平方向上翻转轴。可以反射、旋转或转换 MDS 解，而不会改变点间的邻近性。在处理具有已知地图方向的地理位置时，对轴进行旋转或反射是最常用的。

若在分析中使用了两个以上的维，则可以使用图下方的**选择维**控件切换该图中显示的维。第一个控件定义图的水平轴，第二个控件定义图的垂直轴。

Shepard 图

Shepard 图是实际或变换的邻近性与预测邻近性的对比图。该图反映了多维尺度化图在多大程度上反应了实际的邻近性。Shepard 图类似于“预测值-实际值”图。理想情况下，这些点落在 Y = X 线上（这条线显示为红色）。

拟合详细信息

“拟合详细信息”提供了汇总统计量以衡量 MDS 邻近性与实际邻近性之间的吻合程度，同时还在使用变换时提供关于变换的详细信息。

Stress 在拟合过程中最小化的 Stress 函数 (Stress1) 的值。 Stress 可介于 0 和 1 之间，较低的值表示较好的拟合。

R 方 “预测的邻近性 - 实际或变换的邻近性”的线性拟合的 R^2 值。

斜率 若使用了比或区间变换，则提供变换的斜率。这是实际邻近性针对变换的邻近性的线性回归的斜率。

截距 若使用了区间变换，则提供变换的截距。这是实际邻近性针对变换的邻近性的线性回归的截距。

“多维尺度化”平台选项

使用 “多维尺度化”红色小三角菜单选项，您可以根据自己的需要定制报表。可用选项取决于您用于分析的数据类型和维数。

MDS 图 显示或隐藏 MDS 图。

诊断 提供 MDS 的诊断。

Shepard 图 显示实际邻近性（若使用变换，则为变换的邻近性）与预测的邻近性的对比图。默认显示该报表。请参见“Shepard 图”。

Waern 链接 在 MDS 图上显示 Waern 链接。选定该选项时，将可以使用针对此部分（最小或最大）的控件。请参见“Waern 链接”。

显示坐标 提供解坐标报表。这些是“多维尺度化图”上的点的坐标。该报表显示至多三个维的坐标。右击该报表并选择**列**可将更多维添加到报表中。最大维数是在启动窗口中设置的维数。

显示邻近性 显示邻近性报表。在每对对象之间都提供了原始和派生的邻近性（距离）。对象对在“从对象”和“至对象”列中标识。若使用了变换，变换的邻近性也包含在该表中。

保存邻近性 （仅当数据格式是“特性列表”时可用。）将距离矩阵保存到数据表中。

3D 图 （仅当在启动窗口中为“设置维”指定了三个或更多维时才可用。）显示前三个维的 3D 图。

保存坐标 将解坐标保存到数据表中的单独列内。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

Waern 链接

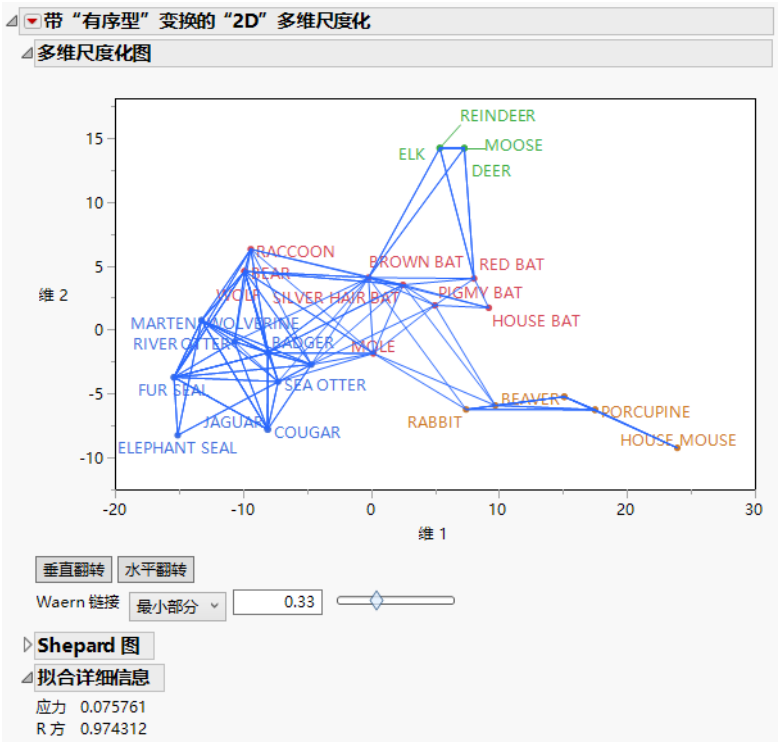
通过将实际邻近性与预测邻近性加以比较，Waern 链接提供对 MDS 结果的可视检查。这些链接基于“多维尺度化图”上各点的实际邻近性连接各点。具有最小（最大）邻近性的对象将连接起来。可以考虑的典型方案是对象之间最小 33% 的邻近性。若 MDS 图能很好地表示邻近性，则针对最小实际邻近性的链接应连接图中最靠近的对象。若针对较小邻近性的链接在图中延伸，以至于连接了相距较远的对象，则需要质疑 MDS 拟合。

Waern 链接控件

您可以从一个列表中进行选择，以显示图中链接的“最小部分”或“最大部分”。通过在框中输入值或使用滑块可以控制显示的链接比例。图 10.5 显示针对 Teeth.jmp 数据表中最小的占比 33% 的 Waern 链接。

有关 Waern 链接的详细信息，请参见 Waern (1972)。

图 10.5 带 Waern 链接的 MDS 图



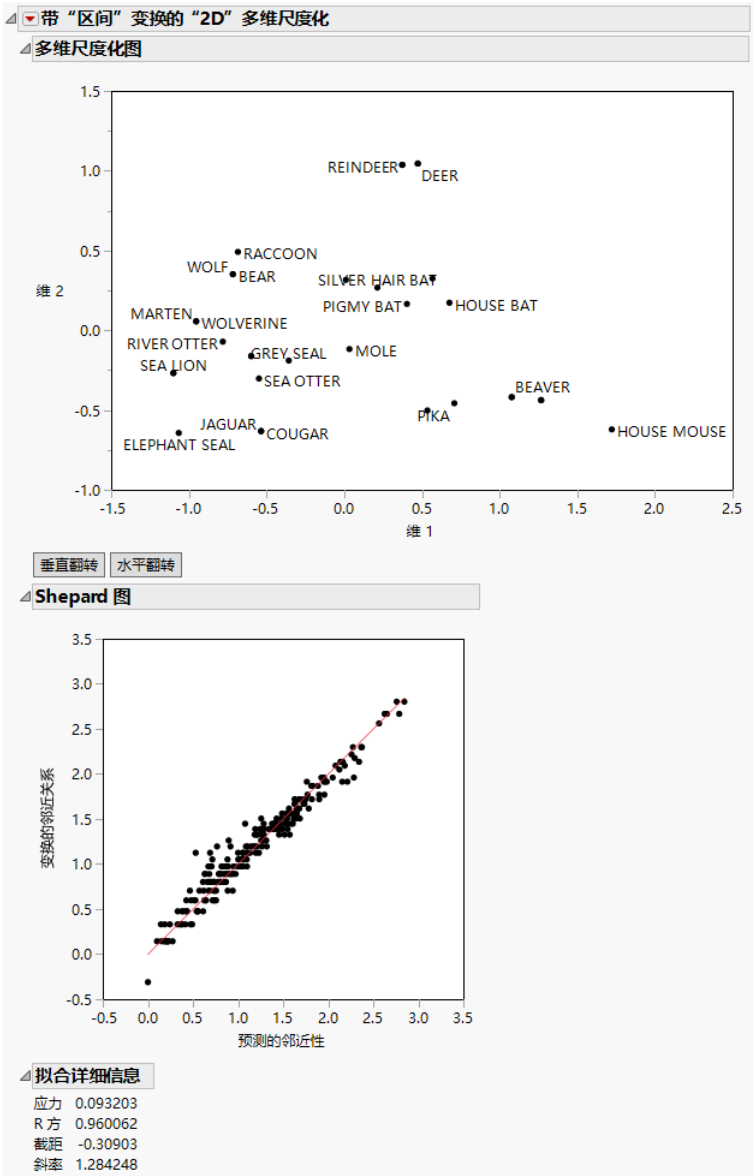
“多维尺度化”平台的更多示例

本例使用 Teeth.jmp 样本数据表，该数据表是关于 32 种哺乳动物的八种牙齿的计数的特性列表。您可以基于哺乳动物的牙齿特性数据使用 MDS 探索其相似性。使用区间变换来演示该变换的输出。数据确实具有有意义的排序 （四颗牙齿的数量是两颗牙齿的两倍之多） 。您可以探索其他变换，如有序型变换。

1. 选择**帮助** > **样本数据库**，然后打开 Teeth.jmp。
2. 右击列**哺乳动物**，然后选择**添加标签 / 撤销标签**。
3. 选择**行** > **行选择** > **选择所有行**。
4. 选择**行** > **添加标签 / 撤销标签**。

5. 选择**分析** > **多元方法** > **多维尺度化**。
6. 从**上门齿**一直选择到**下臼齿**，然后点击 **Y，列**。
7. 选择**数据格式** > **特性列表**。
8. 选择**变换** > **区间**。
9. 点击**确定**。

图 10.6 “多维尺度化”报表



“Shepard 图”和“拟合详细信息”表明：由于动物牙齿的相似性， MDS 图可以很好地表示动物的相似性。“Stress”统计量 0.093 较低，“预测的邻近性 - 变换的邻近性”的 R^2 拟合高达 0.96。此外，“拟合详细信息”提供实际邻近性的变换的截距和斜率。

“多维尺度化”平台的统计详细信息

JMP 使用拟牛顿优化法通过最小化 Stress 函数来确定 MDS 坐标。这种最小化会根据拟合前确定的维数生成一组坐标值，以最小化多维空间里对象间的派生的邻近性测度值。若数据为有序型，则使用单调回归。否则使用标准最小二乘回归。

- “Stress”
- “变换”
- “特性列表格式”

Stress

使用以下符号定义 Stress：

- i, j - 对象数的下标
- d_{ij} - 对象 i 与 j 之间的派生距离

  δ_{ij} - 对象 i 与 j 之间的观测相对距离
- $f(\delta_{rs})$ - 距离的变换函数

Stress 函数定义如下：

$$\text{Stress} = \left[\frac{\sum_{i<j} [f(\delta_{ij}) - d_{ij}]^2}{\sum_{i<j} d_{ij}^2} \right]^{\frac{1}{2}}$$

这一 Stress 测度亦称 Kruskal Stress I 型，或简称 Stress1。

变换

本节使用第 213 页的“Stress”中所述的符号。使用变换来对实际的邻近性进行尺度变换。考虑到数据中的特定结构，变换被认为能够改进 MDS 对实际邻近性的呈现。变换函数中的参数在最小化算法中变为额外参数。

比变换

对于比数据：

$$f(\delta_{rs}) = b\delta_{rs}$$

区间变换

对于区间数据：

$$f(\delta_{rs}) = a + b\delta_{rs}$$

有序型变换

对于有序型数据，数据不进行变换，算法使用单调回归而不是最小二乘回归。

特性列表格式

若数据为特性列表，它将转换为距离矩阵，然后应用 MDS。使用欧氏距离计算距离矩阵。对于每对项，通过以下公式定义项之间的距离：

$$\delta_{ij} = \sqrt{\sum_{k} \frac{(x_{ki} - x_{kj})^2}{k}}$$

其中 k 是特性数目。

注意：有关 MDS 平台的高级示例，请参见 San Francisco Crime Distances.jmp 样本数据表和该表的源脚本。该脚本使用配对相关性创建距离矩阵。生成的距离矩阵随后用于探索犯罪类别之间的关系。

第 11 章

项目分析

按项目和对象分析测验结果

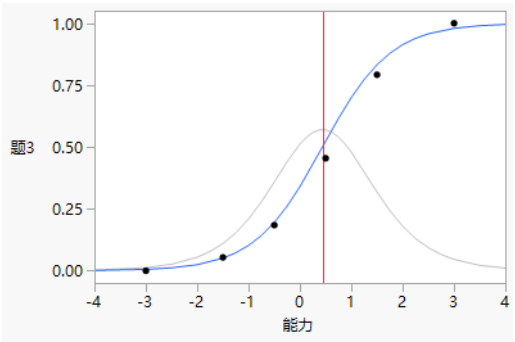
“项目分析”平台支持您拟合**项目反应理论**模型。项目反应理论 (IRT) 方法用于对测量手段（如：测试和问卷）进行分析和评分。项目反应理论通过一系列模型将个人的某项特征或能力与其正面或正确响应某个项目的概率相关联。通常该特征或相关能力不能直接测量，因此被称为**潜在**特征。IRT 可用于研究标准化测验、认知发展和消费者偏好。IRT 是经典测验理论 (CTT) 的替代方法，CTT 侧重于观测得分合计而不是项目得分。

“项目分析”平台执行 IRT 方法，该方法生成以下结果：

- 在项目级别上对测量手段评分，可深入了解每个项目在潜在响应中的贡献。
- 在同一尺度上生成响应者和项目的得分。
- 响应者和项目得分显示在单个图中。
- 显示项目特征曲线。这些曲线可用于探索项目与响应者的潜在特征或能力之间的关系。

有关项目反应理论的详细信息，请参见 de Ayala (2009)。

图 11.1 项目分析特征图



目录

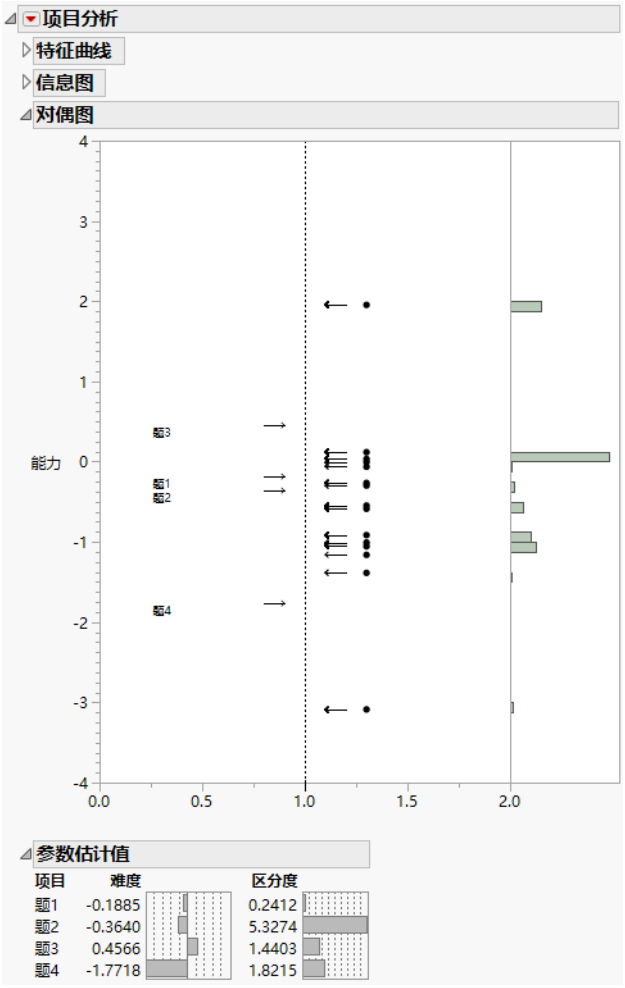
项目分析的示例 217
启动“项目分析”平台 219
Logistic 3PL 模型详细信息 219
数据格式 220
“项目分析”报表 220
特征曲线 220
信息图 221
对偶图 221
参数估计值 222
“项目分析”平台选项 223
“项目分析”平台的统计详细信息 223
项目响应曲线 223
项目响应曲线模型 224
IRT 模型假设 226
拟合 IRT 模型 227
能力公式 228

项目分析的示例

本例使用 MathScienceTest.jmp 样本数据表，该数据表是来自 1996 年开展的第三届国际数学和科学研究项目 (TIMMS) 的数据的子集。该数据表包含关于 1263 个学生在 14 个问题上的得分（1 = 正确，0 = 不正确）。您可以通过检查前四个问题来了解答题者的数学能力和这四个问题之间的关系。测验中的问题即用于测量潜在数学能力的项目。拟合该数据的 2PL 模型。

1. 选择**帮助 > 样本数据库**，然后打开 MathScienceTest.jmp。
2. 选择**分析 > 多元方法 > 项目分析**。
3. 从**题 1** 一直选择到**题 4**，点击 **Y，测验项目**，然后点击**确定**。

图 11.2 项目响应报表

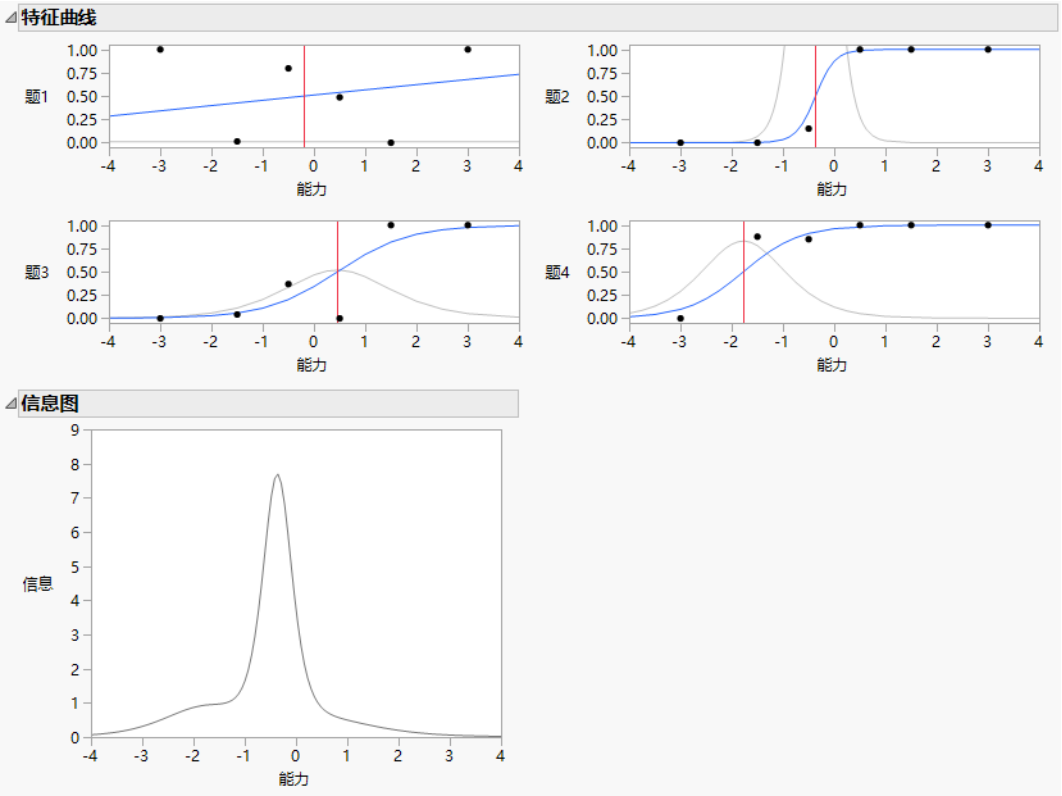


从对偶图中，您会注意到：题 4 是四个问题中最容易回答的问题，这是因为它具有最低的难度得分 -1.78。题 3 是最难的问题，其难度得分为 0.46。大多数响应者都落在能力尺度的中间到低端，如图中心部分的数据点所示。在该直方图中，您可以看到大约 40% 的响应者落在稍微高于能力尺度上的 0 的位置。

注意：不为答案全部正确或全部不正确的个人计算能力得分。请参见第 227 页的“拟合 IRT 模型”。

4. 点击灰色的“特征曲线”报表展开图标将其打开。
5. 点击“项目分析”红色小三角并选择**横向图数**。
6. 输入 2 并点击**确定**。
7. 点击灰色的“信息图”报表展开图标将其打开。

图 11.3 项目响应示例



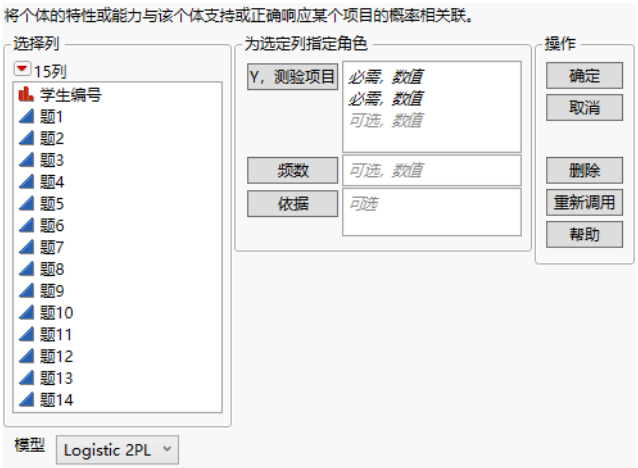
题1 具有平直的特征曲线和平直的信息曲线。这表明题1 没有为区分响应者的数学能力提供太多信息。题2 的特征曲线比较陡峭，这表明题2 对于区分响应者能力很有用。每个图中的垂线都位于特征曲线的拐点。这条垂线即响应者有 50% 的概率正确回答指定问题的能力水平。

信息图指示所分析的四个问题共同提供了关于 -1（大约）到 0 之间的能力水平的多数信息。在模型中包括更多难度更大的问题将会加宽信息曲线。

启动 “项目分析” 平台

通过选择**分析 > 多元方法 > 项目分析**启动 “项目分析” 平台。

图 11.4 “项目分析” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，测验项目 分配两个或更多要分析的列。这些列必须是数值和连续型，并且仅包含 0 和 1。

提示：若需要将数据重新编码为 0 和 1，请使用**列 > 重新编码**。请参见 《使用 JMP》。

频数 指定一个频数变量。该选项适用于汇总数据。

依据 为 “依据” 变量的每个水平生成单独报表。若分配了多个 “依据” 变量，则为 “依据” 变量水平的每个可能组合生成单独报表。

模型 通过以下选项指定所需模型：

Logistic 2PL 2 参数 Logistic 模型。

Logistic 3PL 3 参数 Logistic 模型。

Logistic 1PL 带 Rasch 参数化的单参数 Logistic 模型。

Logistic 3PL 模型详细信息

若为模型选择 Logistic 3PL，则系统会提示您在点击 “确定” 后为猜测参数输入一个罚值。对于 3PL 模型，罚值的默认值为 0。不过，您可以为 *c* 参数 （凭猜测答对某题的概率）输入非零

罚值。该罚值类似于您在岭回归中使用的罚值参数的类型。该罚值与估计猜测参数的方差有关。使用罚值有以下优点：

- 稳定模型参数的估计。
- 加快计算。
- 减少各项之间的猜测参数的变异性，但会带来一些偏倚。

较大的罚值会强制猜测参数为 0，而较小值则会帮助减少各项之间的猜测参数的变异性。值为 0 可用于表示无罚值。

数据格式

“项目分析”平台要求数据表中每个单值对应一行，每个项目对应一列。项列必须是数字列，并且只包含 0 和 1，分别指示不正确或正确的响应。MathScienceTest.jmp 样本数据表演示了针对 14 个测试问题的 1,263 个人的项目响应分析所要求的数据格式。

“项目分析”报表

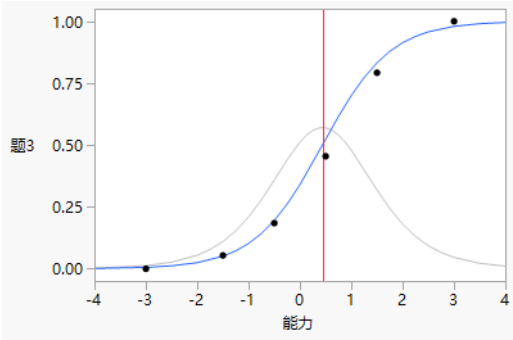
- “特征曲线”
- “信息图”
- “对偶图”
- “参数估计值”

特征曲线

“特征曲线”包含您在启动窗口中指定的每个项的项目特征曲线 (ICC)。“特征曲线”最初处于关闭状态。

项目特征曲线绘制根据能力正确回答某个项目的概率。能力的测量使用了标准化的尺度，所以能力为 0 的响应者即具有平均能力的响应者。固定能力水平的正确回答的观测概率数据点将会绘制出来。将拟合特征曲线与数据点作比较为每个单独的项提供了直观的模型拟合优度测度。此外，特征图还包含一条背景信息曲线以及位于特征曲线拐点处的垂线。背景信息曲线是项目特征曲线的斜率图，在拐点处最大化。

图 11.5 项目特征曲线

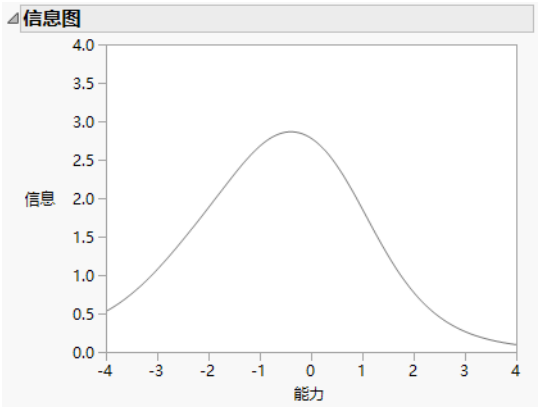


提示：使用“项目分析”红色小三角菜单中的“横向图数”选项，您可以调整报表每行中显示的特征曲线数。

信息图

“信息图”报表包含整体信息曲线图，通过加总各个项的信息曲线可构造整体信息曲线图。可以通过信息图洞察该测验能够测量的正确能力水平。图 11.6 描述了这样一个测验，测验项目更适用于评估具有平均到低能力水平的个人，而不是具有高能力水平的个人。该图最初处于关闭状态。

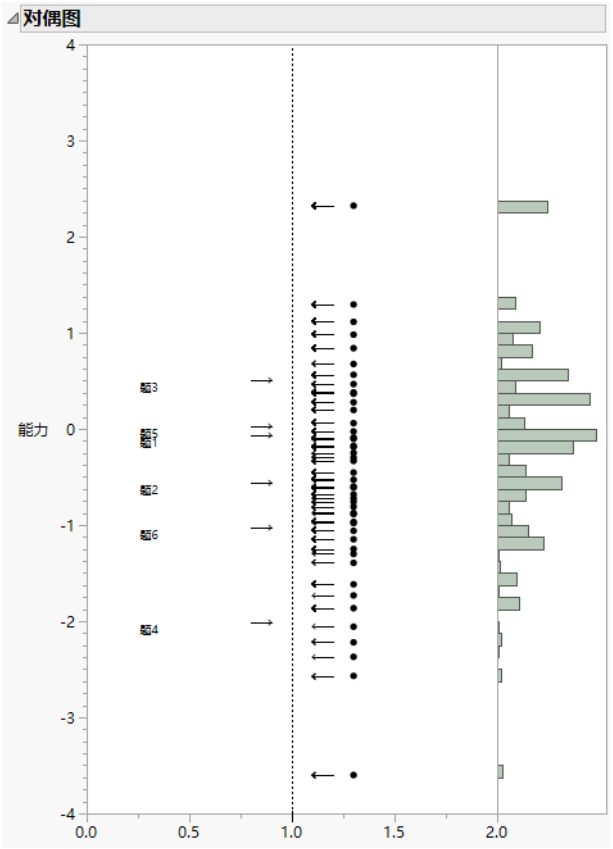
图 11.6 信息图



对偶图

“对偶图”报表包含一个图，这一个图中既显示项目难度又显示对象能力。难度和能力以相同的标准化度量尺度显示在 y 轴上。项目按照其难度标绘在图的左侧。对象绘制在右侧，随同显示的还有数据点和一个直方图。对偶图支持您将每个项目的难度关联到每个响应者的能力。

图 11.7 对偶图



参数估计值

“参数估计值”报表包含每个项目的估计参数表。提供的参数取决于分析中使用的模型（1PL、2PL 或 3PL）。

项目 测验项目。

难度 *b* 参数或项目难度的测度。难度参数的直方图显示在难度估计值旁边。

分辨力 （仅适用于 2PL 和 3PL 模型。）*a* 参数或项目分辨力的测度。分辨力参数的直方图显示在分辨力估计值旁边。

下渐近线 （仅适用于 3PL 模型。） *c* 参数或猜测测度。

“项目分析”平台选项

横向图数 支持您指定在“特征曲线”报表中的每行图中显示多少个 ICC 图。默认设置为每行显示一个 ICC 图。

保存能力公式 在数据表的新列中保存能力公式。该选项使用 `IRT Ability()` JSL 函数。有关该函数的详细信息，请参见“帮助”>“脚本索引”。

请参见《使用 JMP》获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“项目分析”平台的统计详细信息

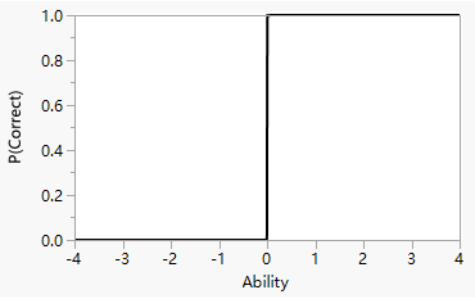
项目反应理论 (IRT) 使用一系列等式将项目关联到未观测到的（潜在）特征或能力。项目（或问题）是无法直接观测到的底层潜在构造的指示符。收集数据时，对象能力和项目特征均未知。

- “项目响应曲线”
- “项目响应曲线模型”
- “IRT 模型假设”
- “拟合 IRT 模型”
- “能力公式”

项目响应曲线

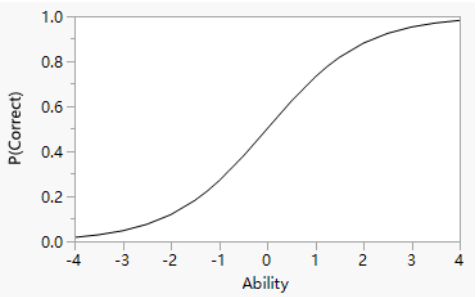
项目响应曲线（项目特征曲线）用于描述能力（在**能力**尺度上定义）与每个项目之间的关系。项目响应曲线绘制针对不同能力水平正确回答某个项目的概率。对于完美分辨力的项目，能力低于阈值的响应者答对的概率为 0%，而能力高于阈值的响应者答对的概率为 100%。

图 11.8 有极佳分辨力的项目的特征曲线



正确回答某个项目的概率与能力之间的典型关系是一个具有下渐近线和上渐近线的 S 形函数。随着响应者能力的增加，其正确回答该项目的概率将增至 100%。特定项目的曲线形状与该项目的难度和分辨力属性相关。

图 11.9 典型的项目响应曲线



项目响应曲线模型

1 参数、2 参数和 3 参数 Logistic 模型可用于对项目响应曲线建模。3 参数 Logistic (3PL) 模型定义如下。

$$P(\theta) = c + \frac{1-c}{1+e^{-(a)(\theta-b)}}$$

- $P(\theta)$ 是针对能力水平 θ 正确回答该项目的概率。有关拟合项目反应理论模型的详细信息，请参见第 227 页的“拟合 IRT 模型”。
- a 参数定义曲线拐点处的陡度。它提供该项目分辨力的估计值。
- b 参数定义拐点在能力轴上的位置。它提供该项目难度的估计值。
- c 参数是下渐近线。它提供通过猜测正确回答某项目的概率估计值。
- 对于 2PL 模型，c 参数设置为 0。

$$P(\theta) = \frac{1}{1 + e^{-(a)(\theta - b)}}$$

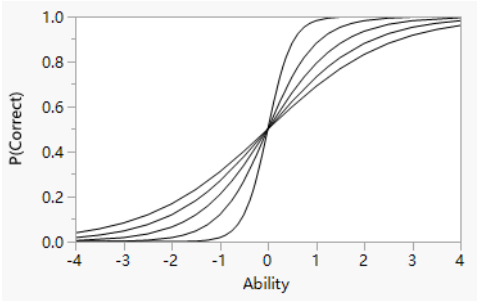
- 对于 1PL 模型，*c* 参数设置为 0，*a* 参数设置为 1。这种参数化亦称 Rasch 模型 (Rasch, 1980)。

$$P(\theta) = \frac{1}{1 + e^{-(\theta - b)}}$$

a 参数：项目分辨力

在 2PL 和 3PL 模型中，*a* 参数（或曲线拐点处的陡度）提供项目分辨力的测度。项目分辨力是指项目可以区分低能力水平响应者与高能力水平响应者的能力。陡峭的项目响应曲线指示该项目具有很强的分辨力。低能力水平响应者正确响应项目的概率较低；高能力水平响应者正确响应项目的概率较高。曲线相对平直的项目具有较低的分辨力。分辨力较低的项目可作为候选项从测量手段中删除。

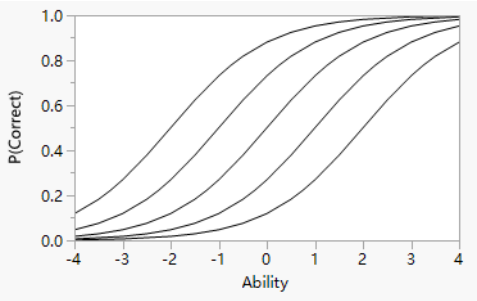
图 11.10 不同 *a* 值的 Logistic 模型



b 参数：项目难度

b 参数（或相对于能力的拐点位置）提供了项目难度测度。拐点在能力尺度上靠右的项目响应曲线指示项目比起拐点靠左的项目更难回答。在 1PL 和 2PL 模型中，*b* 参数是有 50% 的概率正确回答项目所需的能力水平的估计值。

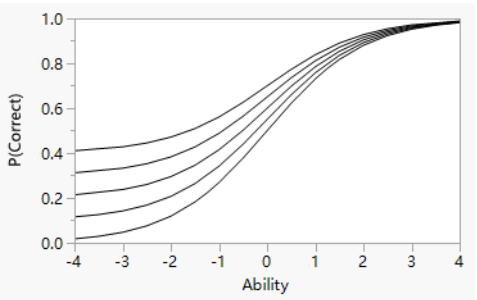
图 11.11 不同 *b* 值的 Logistic 曲线



c 参数：猜测

在 3PL 模型中，*c* 参数（或项目响应曲线的下渐近线）提供猜测参数的测度。非零下渐近线表示能力水平极低的个人正确回答项目的概率不为零。

图 11.12 不同 *c* 值的 Logistic 模型



IRT 模型假设

2PL 模型是“项目分析”平台中的默认模型。当您可以假设所有项目都具有相同的分辨力时，1PL 模型适用。若该假设不正确，则应使用 2PL 或 3PL 模型。2PL 模型比 3PL 模型的数字稳定性强，特别是对于小型数据集。此外，在 2PL 模型中，*b* 可解释为响应者 50% 的概率正确回答项目所需的能力水平。

IRT 模型假定底层特征是一维的。也就是说，只有一个底层的潜在构造。若存在多个特征，而要测量的每个特征相互之间存在复杂的交互作用，这种情况下一维模型就不适合了。IRT 模型适用于连续潜在变量。对于分类潜在变量，您应考虑潜在类模型。请参见第 281 页的“潜在类分析”一章。IRT 模型假定项目不变。项目不变性意味着 $P(\theta)$ 解释为一组具有能力水平 θ 的个体的答对概率。 若一大群具有相同能力水平的个体回答了该项目，则 $P(\theta)$ 预测的是正确回答该项目的人的比例。这意味着 IRT 模型具备项目参数的不变性，不管是什么样的测验群体，您都会得到同样的参数估计。此外，IRT 模型假定具备局部独立性，这意味着一旦解释了潜在构造，项目就彼此独立。

拟合 IRT 模型

使用“边缘最大似然估计”(MMLE) 拟合 IRT 模型。MMLE 是“联合最大似然估计”(JLE) 的备选方法。 MMLE 将对象视为随机效应。项目和能力作为条件概率彼此相关。公式定义如下：

$$p(\mathbf{x}|\theta, \vartheta) = \prod_{j=1}^{L} p_j(\theta)^{x_j}(1-p_j(\theta))^{1-x_j}$$

其中，$p(\mathbf{x}|\theta, \vartheta)$ 是在给定对象能力 θ 和项目参数的向量 ϑ 时，响应向量**x** 的概率。项目参数的数目取决于使用的模型（1PL、2PL 或 3PL）。

MMLE 使用高斯求积分方法对对象效应求积分，以此获取项目参数估计值。响应向量 **x** 的概率计算如下：

$$p(\mathbf{x}) = \int_{-\infty}^{\infty} p(\mathbf{x}|\theta, \vartheta)g(\theta|\upsilon)d\theta$$

其中，$g(\theta|\nu)$ 是对象分布，ν 是总体位置和尺度参数的向量。在 JMP 中，均值为 0、标准差为 1 的正态分布用于 $g(\theta|\nu)$。

注意：针对测验问题的缺失值被视为错误响应。不为答案全部正确或全部不正确的个人计算能力得分。这些测试对象的响应模式包括在模型估计中。

用于拟合 IRT 模型的 MMLE 过程可与在两个阶段中拟合随机效应模型作比较。能力参数被视为方差为 1 的随机效应。在第一步中，使用高斯求积分对这些随机效应求积分。项目参数被视为使用边缘似然（已对能力参数求积分）的 ML 估计的固定效应。能力参数实际上是使用完全未积分（联合）似然估计的最佳线性无偏预测，且估计时项目参数已知，并且保持在第一阶段中获取的值。

对于 L 个项目，有 2^L 个响应模式。每个模式的能力水平可以通过计算在该响应模式下出现最高概率的能力水平，可通过应用以下公式直到 θ 进行计算。

$$\theta_i^{t+1} = \theta_i^{t} - \frac{X_i - \sum_{j=1}^{L} p_{ij}(t)}{-\sum_{j=1}^{L} p_{ij}(t)(1-p_{ij}(t))}$$

其中：

θ 将获取响应模式的似然最大化

t 是迭代次数

L 是项目数

X_i 是观测的得分

p_{ij} 是在特定的项目参数下，第 i 个人答对第 i 个项目的概率。

能力公式

“项目分析”红色小三角菜单中的“保存能力公式”选项将能力公式保存到数据表的新列中。该公式可用于为添加到数据表中的其他测试对象评分，也可以将其复制到新表中为新的一组测试对象评分。

保存到数据表的函数称为 `IRT Ability` 函数。项目参数估计值以矩阵的形式存储在该函数中。有关该函数的详细信息，请参见“帮助”>“脚本索引”。

第 12 章

层次聚类

使用聚类树将观测分组

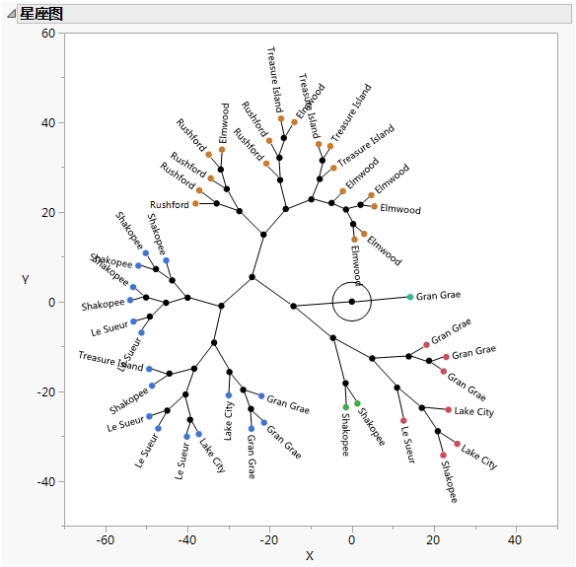
聚类是将在几个变量上享有相似值的观测分组在一起的一种多元方法。它可用于您理解数据的聚簇结构。

“层次聚类”依次组合聚类。该方法首先将每个观测视为一个聚类。然后逐步将距离上最相近的两个聚类合并成一个聚类。结果被描绘成一棵树，称为**系统树图**。

可以对不多于数万行的小数据表使用“层次聚类”。该算法占用较长时间，因此大数据表运行会很缓慢。对于较大的数据表，则使用“K 均值聚类”或“正态混合”。

注意：“层次聚类”支持字符列；“K 均值聚类”或“正态混合”要求使用数值列。

图 12.1 星座图的示例



目录

“层次聚类”平台概述 231
对观测聚类的平台概述 231
聚类示例 232
启动“层次聚类”平台 235
聚类方法 236
计算距离的方法 236
数据结构 237
“Y, 列”变量的变换 238
“层次聚类”报表 239
“系统树图”报表 239
系统树图和距离图的演示 240
“聚类历史”报表 241
“层次聚类”选项 241
“层次聚类”平台的更多示例 244
距离矩阵的示例 244
使用“空间测度”进行晶片次品分类的示例 246
“层次聚类”平台的统计详细信息 248
空间测度 248
距离方法公式 250

“层次聚类”平台概述

“层次聚类”是 JMP 提供的对观测进行聚类的四个平台之一。有关四种方法的比较，请参见第 231 页的“对观测聚类的平台概述”。

层次聚类方法首先将每个观测作为一个聚类。在每一步中，聚类过程会计算各对聚类之间的距离，并将两个相距最近的聚类组合起来。该组合过程会一直进行到所有点都位于一个聚类中。层次聚类亦称为**自下而上聚类**，因为它使用的是一个组合方法。

自下而上的过程被描绘为一棵树，称为系统树图。为了帮助您确定聚类数，JMP 会提供距离图。您可以通过确定聚类之间的距离何时不再具有实际意义的方式来选择聚类数。

层次聚类也支持字符列。可通过两种方式定义距离。

- 若列为有序型，则用于聚类的值就是有序类别的索引，将像处理连续数据那样处理有序类别。这些值将像处理连续数据那样进行标准化。
- 若列为名义型，则类别匹配的两个观测之间的距离为 0。若类别不同，则距离为 1。

“层次聚类”提供给您五个规则用于定义聚类之间的距离：类平均法、重心法、Ward 法、最短距离法和最长距离法。每个规则会生成不同序列的聚类。

提示：层次聚类过程针对 n 个观测会从 $n(n+1)/2$ 个距离开始计算，但使用“快速 Ward”方法时除外。因此，当 n 较大时，该方法会运行较长时间。对于大量数值观测的情况，考虑使用“K 均值聚类”或“正态混合”。

对观测聚类的平台概述

聚类是将在几个变量上享有相似值的观测分组在一起的一种多元方法。通常情况下，观测在 p 维空间内散布不均，其中 p 是变量数。这些观测反而会形成聚簇或聚类。标识出这些聚类使您可以更深层次地了解您的数据。

注意：JMP 还提供可以对变量聚类的平台。请参见第 295 页的“聚类变量”一章。

JMP 提供四个平台供您对观测聚类：

- “层次聚类”可用于最多包含数万行的小型表，并且允许字符数据。“层次聚类”将行按描绘为一棵树的层次序列形式进行组合。您可以在生成树后选择最适合您数据的聚类数。
- “K 均值聚类”适用于多达数百万行的大型表，并且只允许数值数据。您需要提前指定聚类数 k。该算法可以对聚类种子点做出推测。随后开始在将数据点分配到相应类别和重新计算聚类中心之间交替进行迭代过程。
- “正态混合”适用于数据来自重叠的多元正态分布的混合分布这种情况，并且只允许数值数据。对于具有多元离群值的情形，您可以使用假设具有均匀分布的离群值聚类。

  您需要提前指定聚类数。最大似然用于同时估计混合比例以及均值、标准差和相关性。为每个点指定属于每个组的概率。使用 EM 算法获取估计值。

- “潜在类分析”适用于大多数变量是分类变量这种情况。您需要提前指定聚类数。该算法拟合假定具有多项式混合分布的模型。为每个观测计算聚类成员关系的最大似然估计值。观测会被归类到其成员关系概率最大的聚类中。

表 12.1 聚类方法汇总

方法	数据类型或建模类型	数据表大小	指定聚类数
层次聚类	任意	使用快速 Ward 法，最多 200,000 行 使用其他方法，最多 5,000 行	否
K 均值聚类	数值	多达数百万行	是
正态混合	数值	任意大小	是
潜在类分析	名义型或有序型	任意大小	是

有些聚类平台提供用于处理数据中的离群值的选项。但是，若数据中有离群值，则最好在分析之前探索这些离群值。可以使用“探索离群值”实用工具完成该操作。详细信息，请参见《预测和专业建模》“建模工具”一章。

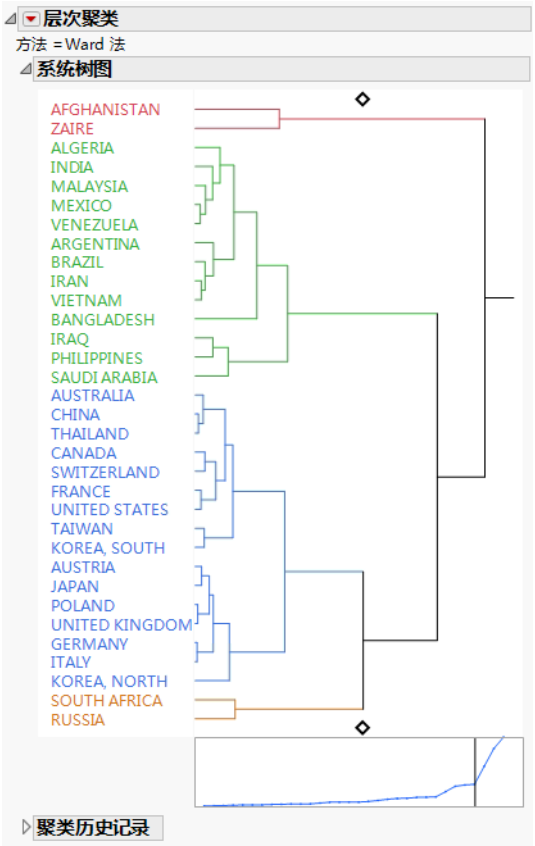
聚类示例

在本例中，我们将按照各国 2009 年每 1,000 人的粗出生率和死亡率将各国分组。

1. 选择**帮助 > 样本数据库**，然后打开 Birth Death Subset.jmp。
2. 选择**分析 > 聚类 > 层次聚类**。
3. 选择**出生率**和**死亡率**并点击 **Y, 列**。
4. 选择**国家**并点击**标签**。

   该选择可确保**国家**列 （而不是行号）用于对点击 “确定”后出现的系统树图添加标签。
5. 点击**确定**。
6. 点击 “层次聚类”红色小三角并选择**聚类着色**。

图 12.2 “层次聚类”报表



该系统树图显示聚类是如何执行的。可以从左到右读取系统树图来查看聚类过程。每步包括将两个**最近**聚类组成一个聚类。

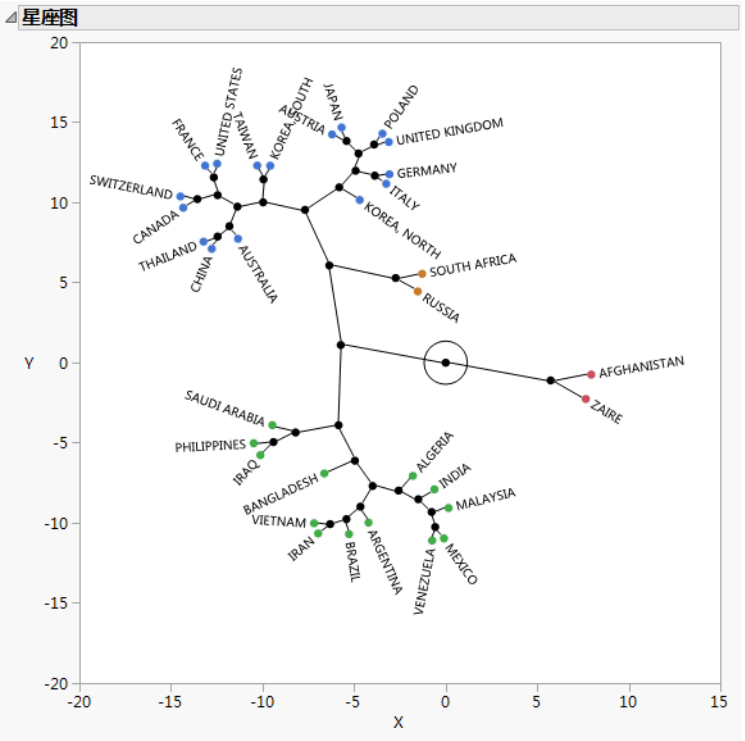
在系统树图中，聚类之间的相对距离由连接聚类的垂线之间的水平距离给出。例如，Afghanistan 和 Zaire 之间的距离大于 Malaysia 与 Mexico 和 Venezuela 组成的聚类之间的距离。

系统树图下方显示的图，两个聚类连成一个聚类的每一步对应一个点。水平坐标表示聚类数，它们从左至右递减。点的垂直坐标是连接起来形成指定聚类数的两个聚类之间的距离。您可以点击系统树图中的菱形并拖动线以选择最能代表数据的聚类数。还可以使用“层次聚类”红色小三角菜单中的“聚类数”选项以选择聚类数。

有四个聚类时，距离图斜率有明显变化。斜率变化指出在剩下四个聚类之前所连接的聚类之间的差异相对较小。这表明 4 是聚类数不错的选择。注意到这是默认显示的聚类数。

7. 点击“层次聚类”红色小三角并选择**星座图**。

图 12.3 星座图

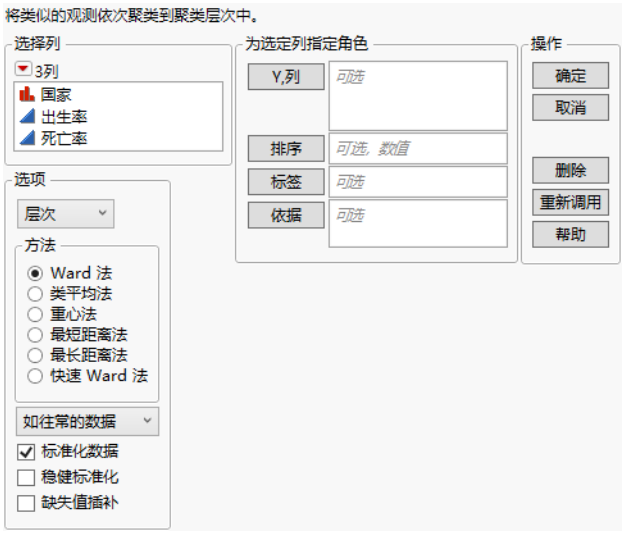


该星座图将国家显示为端点，将每个聚类连接显示为新点。各条线表示聚类中的成员关系。聚类连接之间的线长度近似等于连接的聚类之间的距离。星座图指出包含 Afghanistan 和 Zaire 的聚类和其余国家组成的聚类，相比上半部分图中其余国家和下半部分图中其余国家构成的两个聚类，它们之间的距离是差不多的。

启动 “层次聚类” 平台

通过选择**分析** > **聚类** > **层次聚类**来启动 “层次聚类” 平台。

图 12.4 “层次聚类” 启动对话框



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 用于对观测聚类的变量。

排序 基于指定列按聚类的均值对其排序。

> **提示：**将通过执行主成分分析获取的第一主成分用作 “排序” 列。聚类按这些值排序。

特性 ID （仅当选择**数据被堆叠**作为数据结构时才可用。）指定堆叠的变量。

对象 ID （仅当选择**被汇总的数据**或**数据被堆叠**作为数据结构时才可用。 ）为堆叠了其测量值的每个单元提供唯一标识符的一列或多列。

标签 其值用于对报表中的系统树图添加标签的列。

> **注意：**若选定的数据结构为**数据为距离矩阵**，“标签” 列的数据类型必须为字符型。

依据 一列，其水平定义不同的分析。对于指定列的每个水平，都分析相应行。结果显示在不同的报表中。若分配了多个 “依据” 变量，则为 “依据” 变量水平的每个可能组合生成单独分析。

启动窗口具有下列菜单和选项：

- “聚类方法”

- “计算距离的方法”
- “数据结构”
- ““Y, 列”变量的变换”

聚类方法

“层次”是默认聚类方法，但您可以在对话框中切换为“K 均值”或“正态混合”。若您选择“K 均值”或“正态混合”，点击“确定”时会出现“控制面板”，您可以在其中选择下列选项之一作为“方法”：

K- 均值聚类 请参见第 253 页的“K 均值聚类”一章。

正态混合 请参见第 269 页的“正态混合”一章。

稳健正态混合 请参见“正态混合”一章中第 277 页的““正态混合”报表”。

自组织图 请参见“K 均值聚类”一章中第 264 页的“自组织图”。

计算距离的方法

选择用于计算距离的方法。有关距离公式的信息，请参见第 250 页的“距离方法公式”。

Ward 法 在 Ward 最小方差法中，两个聚类之间的距离是两个聚类之间的方差分析平方和（所有变量的方差分析和相加）。每一次生成聚类时，在通过合并上一次生成的两个聚类可能获得的所有划分中，类内平方和进行了最小化。当平方和除以总平方和时会得到方差比例（平方半偏相关），这样可以更容易地解释该平方和。

在假定使用多元正态混合、球形协方差矩阵和等抽样概率的前提下，Ward 法通过将层次每个水平下的似然最大化来生成聚类。

Ward 法倾向于生成有少量观测的聚类，而且特别偏向于生成具有大致相同观测数的聚类。它还对离群值非常敏感。请参见 Milligan (1980)。

类平均法 两个聚类之间的距离是观测对之间的平均距离。类平均法倾向于连接具有较小方差的聚类，并且略微偏向于生成具有相同方差的聚类。请参见 Sokal and Michener (1958)。

重心法 两个聚类之间的距离定义为其均值之间的欧氏距离平方。与大多数其他层次方法相比，重心法对离群值更为稳健，但其他方面的性能却不如 Ward 法或类平均法。请参见 Milligan (1980)。

最短距离法 两个聚类之间的距离是一个聚类中的某个观测与另一聚类中的某个观测之间的最小距离。最短距离法提供许多用户所需的理论属性，但在 Monte Carlo 研究中表现不好。请参见 Jardine and Sibson (1971)、Fisher and Van Ness (1971)、Hartigan (1981) 和 Milligan

(1980)。最短距离法源自 Florek et al. (1951a, 1951b)，后来又被 McQuitty (1957) 和 Sneath (1957) 改写。

由于不对聚类形状施加约束，最短距离法能够检测细长和不规则的聚类，却牺牲了恢复紧密聚类的性能。最短距离法往往会在区分主聚类之前，先切割掉分布的尾部。请参见 Hartigan (1981)。

最长距离法 两个聚类之间的距离是一个聚类中的某个观测与另一聚类中的某个观测之间的最大距离。最长距离法强烈偏向于生成具有大致相等直径的聚类，并且可能被中度离群值严重扭曲变形。请参见 Milligan (1980)。

快速 Ward 法 应用对大量行计算 Ward 法更快的算法。由于该算法不需要计算距离矩阵，因此缩短了计算时间。只要超过 2,000 行就自动使用该方法。

数据结构

以下选项介绍在计算多元距离时使用的数据格式。

如往常的数据 矩形数据，每个观测对应一行，每个变量对应一列。

被汇总的数据 按一个或多个标识列的水平汇总的数据。当您选择该选项时，启动窗口中会出现 “对象 ID” 文本框。指定标识列作为 “对象 ID”。**被汇总的数据**选项计算水平均值并将这些均值视为输入数据。

数据为距离矩阵 由观测之间的距离组成的数据。对于 n 个观测，距离表应有 n 行和 $n+1$ 列。有一列 （通常为第一列）必须包含 n 个观测的唯一标识符。其余列包含该观测和 n 个观测之间的距离。请注意以下几点：

- 表的对角线元素应为 0 或缺失，因为点与其本身之间的距离为 0。不为 0 或缺失的值会视为 0，并且报表中会显示一条注释。
- 距离列可以是对称方矩阵，也可以是上三角或下三角矩阵并且缺失条目出现在下部或上部。若距离按方矩阵给出，则表不对称时报表中会出现一条警告。
- 您可以先开始使用不同的数据结构，然后保存距离矩阵。请参见第 243 页的 “保存距离矩阵”。

  当您选择**数据为距离矩阵**选项时，输入距离列作为 “Y, 列”，标识符列作为 “标签”。“标签” 列必须具有 “字符型” 数据类型。有关示例，请参见第 244 页的 “距离矩阵的示例”。

数据被堆叠 具有单个关注响应且每个对象对应多行的数据。

当您选择**数据被堆叠**选项时，启动窗口中会显示 “特性 ID” 和 “对象 ID” 文本框。

- 输入**单个**列作为 “Y, 列”。
- 输入描述 “Y, 列” 变量分组的列作为 “特性 ID”。若仅输入两列且选择 “添加空间测度”，则可以在聚类分析中添加要使用的空间成分。请参见第 239 页的 “添加空间测度”。
- 输入对象的标识列作为 “对象 ID”。

执行的分析等价于按“特性 ID”列拆分“Y, 列”变量然后在不标准化响应列的情况下执行层次聚类。

提示：将该选项与“添加空间测度”选项一起使用可执行二维空间聚类。例如，晶片数据经常使用每个裸片对应一行的方式进行记录。关注点集中在晶片聚类。请参见第 246 页的“使用 “空间测度”进行晶片次品分类的示例”。

警告：因为有单个测量值列，“标准化数据”选项不适用于堆叠数据。

没有足够的非缺失数据警示

当您使用**被汇总的数据**或**数据被堆叠**数据结构时， JMP 警示**没有足够的非缺失数据**很难理解。下列情形会出现该警示：

- 对于**如往常的数据**，当所有行或除一行之外的所有行有至少一个“Y, 列”变量的值缺失时。
- 对于**被汇总的数据**，当针对“对象 ID”列汇总您的数据，所有行或除一行之外的所有行有至少一个汇总“Y, 列”变量的值缺失时。要查看“聚类”平台正在分析的数据结构，选择“表 > 汇总”，输入“对象 ID”列作为“分组”，“Y, 列”变量作为“统计量 > 均值”。
- 对于**数据被堆叠**，当针对“特性 ID”列拆分您的数据，所有行或除一行之外的所有行有至少一个拆分“Y, 列”变量的值缺失时。要查看“聚类”平台正在分析的数据结构，选择**表 > 拆分**，输入“特性 ID”列作为“拆分依据”，“Y, 列”变量作为“拆分列”，“对象 ID”列作为“分组”。

“Y, 列”变量的变换

下列选项指定在聚类分析中使用的 “Y, 列”变量的形式：

标准化数据 解决连续型和有序型列具有不同测量值尺度的问题。除**数据被堆叠**选项选中时，每列的值会通过减去列均值并除以列标准差的方式进行标准化。若不希望基于标准化值计算聚类距离，则取消选中 “标准化数据”复选框。

稳健标准化 减小离群值对连续型和有序型列的均值和标准差估计值的影响。该选项使用均值和标准差的 Huber M 估计值（Huber 1964； Huber 1973； Huber and Ronchetti 2009）。对于包含离群值的列，该选项在确定多元距离时可以更好地用标准化值来表示不同的测度。

注意：若同时选中“标准化数据”和“稳健标准化”，则通过减去列的稳健均值再除以其稳健标准差对每列进行标准化。若列表示不同的测量值尺度**或**观测仅在特定维上趋向于是离群值，这样该选项很有用。

注意：若取消选中“标准化数据”而选中“稳健标准化”，则所有列中的值的稳健均值和稳健标准差将组合起来共同用于标准化每一列。若列全部表示同一测量值尺度**并且**观测在所有维上都趋向于是离群值，这样该选项很有用。

缺失值补缺 补缺缺失值。若变量数小于等于 50 或少于行数的一半，则使用多元正态补缺。否则使用多元 SVD 补缺。

多元正态补缺计算配对协方差以构造响应列的协方差矩阵。然后，通过等价于回归预测的方法，使用给定变量不带缺失值的所有预测变量补缺每个缺失值。若构造的协方差矩阵不是正定矩阵，则使用其列均值对缺失值补缺。

多元 SVD 补缺通过使用奇异值分解避免构造协方差矩阵。请参见 《预测和专业建模》。

警告：缺失值补缺假定不存在任何聚类、数据来自单个多元正态分布，并且值的缺失是完全随机的。因为这些假设通常在实践中是不合理的，需谨慎使用该功能。不过与放弃大多数的数据相比，该功能可生成更具说明性的结果。

添加空间测度 （仅当选择**数据被堆叠**作为数据结构时才可用。）当您的数据进行了堆叠并且包含两个对应于空间坐标（例如水平和垂直坐标）的特性列时，选择**添加空间测度**选项。该选项将打开一个窗口，您可在其中选择和权衡空间成分以帮助聚类缺陷模式。这是专业方法，仅在非常特定的设置下适用。请参见第 248 页的 “空间测度”和第 246 页的 “使用 “空间测度”进行晶片次品分类的示例”。

“层次聚类”报表

“层次聚类”报表显示使用的方法、系统树图和 “聚类历史记录”表。若您在启动窗口中分配了一列作为 “标签”，则该列的值会在系统树图中标识每个观测。

- “ “系统树图”报表”
- “系统树图和距离图的演示”
- “ “聚类历史”报表”

“系统树图”报表

系统树图是一个树形图，用于表示观测如何划分到聚类中。系统树图还提供有关聚类相异性程度的信息。

可以从左到右读取系统树图来查看聚类过程。每步包括将两个**最近**聚类组成一个聚类。

- 由垂直线相连的水平线表示聚类的生成。
- 垂直线的水平位置表示，为了形成指定的聚类数，最近生成的两个类别之间的距离。

注意：当观测数小于 256 时，距离与 “距离图”中显示的距离成比例。否则，使用 “几何间距”。请参见第 242 页的 “系统树图尺度”。

您可以执行下列任务：

- 点击并拖动系统树图顶部或底部的菱形控点来标识给定数目的聚类。

- 点击任意聚类茎来选择系统树图和数据表中该聚类的所有成员。

距离图

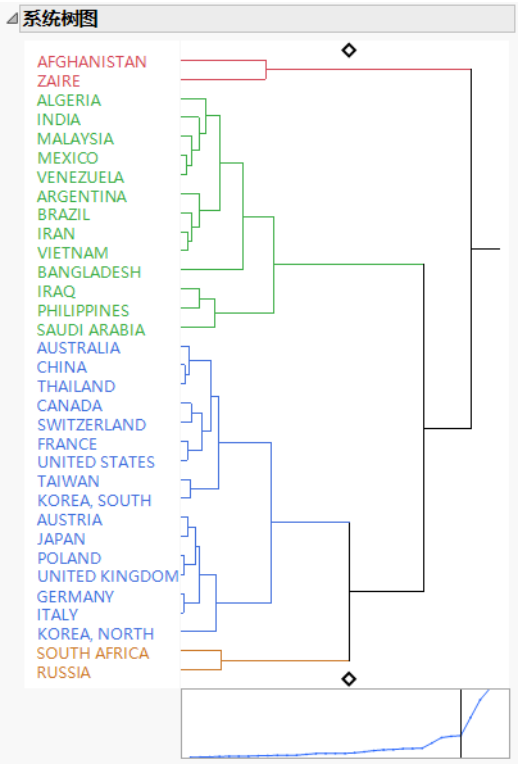
“距离图”是系统树图下方显示的图。对于两个聚类连接成一个聚类的每一步，该图形都对应一个点。水平坐标表示聚类数，它们从左至右递减。点的垂直坐标是在给定步连接的两个聚类之间的距离。

您可以点击并拖动系统树图中的菱形控点来控制选择的聚类数。当您点击并拖动菱形时，图中会显示一条垂直线，您可以将它移动到对应的聚类数。经常有一个点对应的距离图斜率会变得平坦。此点表明趋势的自然转折，可帮助您确定聚类数。

系统树图和距离图的演示

考虑第 232 页的“聚类示例”中有关 Birth Death Subset.jmp 的系统树图报表。

图 12.5 Birth Death Subset.jmp 的系统树图报表



菱形设置在四个聚类处。最近连接起来形成四聚类模型的两个聚类是由 Algeria 到 Bangladesh 组成的聚类以及由 Iraq 到 Saudi Arabia 组成的聚类。这两个聚类之间的距离是菱形设置为 4

时距离图上由垂直线指示的点。该距离在“聚类数”等于 4 旁边的“聚类历史”报表中给出。此处显示该距离为 1.618708760 并且聚类从 Algeria 和 Iraq 开始组合形成四个聚类。

组合起来形成五个聚类的两个聚类是由 Australia 到 Korea, South 组成的聚类以及由 Austria 到 Korea, North 组成的聚类。这些聚类在系统树图中对应的垂直线与连接起来形成四聚类模型的聚类（Algeria 到 Bangladesh 和 Iraq 到 Saudi Arabia）对应的垂直线从左开始算有差不多相同的水平距离。由此判定：连接起来形成四聚类模型的聚类与连接起来形成五聚类模型的聚类之间在距离上没有太大的差别。

距离图从四聚类模型开始变得平坦这一事实表明计算到该点的聚类分组不会导致聚类之间产生太大距离。但是，四聚类模型显示聚类之间可以很好地分隔开。

“聚类历史”报表

“聚类历史”表包含聚类历史。

聚类数 列出执行“前导对象”和“连接对象”所指示的连接之后得到的聚类数。聚类数从第一次连接开始计算，此时有 *n* - *1* 个聚类，其中 *n* 是对象数。该报表以降序方式列出聚类数，直到最后所有对象都包含在一个聚类中。这样，“聚类历史”遵循系统树图从左至右的顺序。

距离 聚类之间的距离，它根据您在启动窗口上选择的距离方法计算。请参见第 236 页的 “计算距离的方法”。

前导对象 代表系统树图中要连接的第一个聚类。“前导对象”列中显示的聚类顺序和代表对象是数据排序产生的结果，没有内在意义。

连接对象 代表系统树图中要连接的第二个聚类。“连接对象”列中显示的聚类顺序和代表是根据数据如何排序产生的结果，没有内在意义。

“层次聚类”选项

“层次聚类”红色小三角菜单包含以下选项：

聚类着色 根据聚类成员关系对系统树图的标签及其关联的连接条进行着色。此外将相应的颜色分配给数据表的行。若您更改聚类数，颜色也会更新。若您取消选择该选项，颜色将不再根据变量数进行更新。

标记聚类 将标记分配给数据表中与该行所属的聚类对应的那些行。若您更改聚类数，标记也会更新。若您取消选择该选项，标记将不再根据变量数进行更新。

聚类数 提示您输入聚类数并将系统树图滑块放置在该数目对应的位置上。

聚类准则 提供针对聚类数整个范围的三次聚类准则 (CCC)。CCC 用于估计聚类数。它可以与任何基于距离的聚类算法一起使用。 CCC 值越大则表明在特定聚类数下拟合的效果越好。请参见 SAS Institute Inc. (1983)。（选定**数据为距离矩阵**时不可用。）

显示系统树图 显示或隐藏“系统树图”报表。

系统树图尺度 包含用于统一系统树图尺度的选项：

距离尺度 基于在启动窗口中指定的距离方法，将任意两个连接点之间的水平距离显示为在该点连接的两个聚类之间的距离。距离尺度与“距离图”中使用的是同一个尺度，它是系统树图的默认尺度。

等间距 显示任意两个连接点之间的水平距离相等。

几何间距 随聚类数的增加，连接点之间的水平距离也随之增加。当有许多对象并且您想较小的聚类比较大的聚类更加明显时可以使用该选项。

距离图 显示或隐藏系统树图下方的距离图。

显示聚类数控点 在系统树图上显示或隐藏用于手动更改聚类数的控点。

缩放至选定行 在系统树图中选择聚类之后，选择并放大特定聚类。或者，您可以双击聚类将其放大。使用 “解除缩放”可恢复到原始视图。

解除缩放 在放大之后，使系统树图恢复到原始视图。

以选定聚类为轴心转动 反转当前选定聚类的两个子聚类的顺序。

色图 提供选项来添加色图或热图，用于显示按值着色的各个 “Y, 列”变量。子菜单中提供了若干颜色主题选择。

双向聚类 按在 “Y, 列”中指定的变量以及行来进行聚类。色图和系统树图一起添加，并且在其底部添加了“Y, 列”变量。通常情况下，对于双向聚类，变量以相同的尺度进行测量，您不需要选择 “标准化数据”。（选定**数据被堆叠**时不可用。）

定位 提供用于更改标签和系统树图的其他部分的位置的选项。

图例 显示或隐藏色图中使用的颜色的图例。该选项仅在启用色图时才可用。

更多色图列 为指定列添加色图。（选定**被汇总的数据**、**数据为距离矩阵**或**数据被堆叠**时不可用。）

星座图 显示或隐藏另一种在层次聚类系统树图中显示信息的方式。每个观测 （行）由一个端点表示，每个聚类连接由一个新点表示。绘制的线条表示聚类成员关系。线条长度表示聚类之间的距离。较长的线表示聚类之间的距离较长。

您可以将鼠标悬停在星座图中的线条上来查看其长度。但是，长度值只有相对意义。轴尺度、点方向和线条角度是任意的。这样安排可以使节点端间隔开，图不会显得很混乱，这对于大数据集来说十分重要。

要关闭端点处的标签，请在 “星座图”内右击并选择**显示标签**。

保存星座坐标 将星座图坐标保存至数据表。（选定**被汇总的数据**、**数据为距离矩阵**或**数据被堆叠**时不可用。）

保存聚类 创建包含聚类数的数据表列。若在启动窗口中选择了 “添加空间测度”，则聚类数也会保存到 Hough 数据表。

保存最近聚类公式 创建包含最近聚类的公式的数据表列。该选项计算每个聚类重心之间的欧氏距离平方，并选择最靠近的聚类。请注意，该公式不一定总能重现“层次聚类”提供的聚

类分配，因为聚类的确定方式有所不同。不过，聚类分配非常相似。（选定**被汇总的数据**、**数据为距离矩阵**或**数据被堆叠**时不可用。）

保存显示顺序 创建包含行在系统树图中的显示顺序的数据表列。

保存聚类层次结构 创建一个数据表，其中包含编写自定义系统树图的脚本所需的信息。对于每个聚类连接，该选项都输出三行：第一行表示连接对象、第二行表示前导对象、第三行表示结果，用于给出聚类中心、大小和其他信息。

保存聚类树 创建一个新的数据表，其中包含在 JMP 和 SAS 之间比较聚类树所需的信息。对于每个聚类连接，该选项都为每个新聚类输出一行，包含该聚类的大小和其他信息。

保存距离矩阵 创建一个新的数据表，其中包含观测之间的距离。

保存聚类均值 创建一个新的数据表，其中包含每个聚类中的行数和每列的均值。

聚类汇总 （选定**数据为距离矩阵**时不可用。）显示以下信息：

聚类均值 给出每个聚类的观测数（若数据被堆叠则为“对象 ID”）和每个变量的均值的表。

聚类标准差 给出每个聚类的观测数（若数据被堆叠则为“对象 ID”）和每个变量的标准差的表。

聚类均值图 聚类均值的平行图或二维热图。

该图为平行图，但选中**数据被堆叠**并且有两个“特性 ID”变量时除外。对于平行图，每个变量的轴都统一了尺度。

– 若选中“标准化数据”，轴的范围在均值的上下两个标准差，其中标准差和均值基于原始数据进行计算。若聚类均值超出该范围，则轴会扩展以包括该均值。

– 若未选中“标准化数据”，则会使用显示了尺度的公共垂直轴。（该尺度等价于“图形生成器”中的“统一尺度”选项）。

当选中**数据被堆叠**并且有两个“特性 ID”变量时，在每个位置为每个聚类显示 Y 变量均值的二维图。这些图使用“由蓝经灰到红”颜色梯度着色。

列汇总 对于每个变量，给出表示聚类所解释的变异比例的 R 方值。该数值是聚类中变量回归的 R 方值。选项还给出 R 方值的条形图。

散点图矩阵 使用所有变量创建散点图矩阵。（选定**被汇总的数据**、**数据为距离矩阵**或**数据被堆叠**时不可用。）

平行坐标图 为每个聚类创建平行坐标图。（选定**被汇总的数据**、**数据为距离矩阵**或**数据被堆叠**时不可用。）轴的尺度参照“聚类均值图”的说明。请参见第 243 页的“聚类均值图”。

聚类处理比较 （仅当您按住 Shift 并点击“层次聚类”红色小三角时才可用。）选择响应列和二水平处理列。创建“层次聚类差值”报表。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“层次聚类”平台的更多示例

- “距离矩阵的示例”
- “使用“空间测度”进行晶片次品分类的示例”

距离矩阵的示例

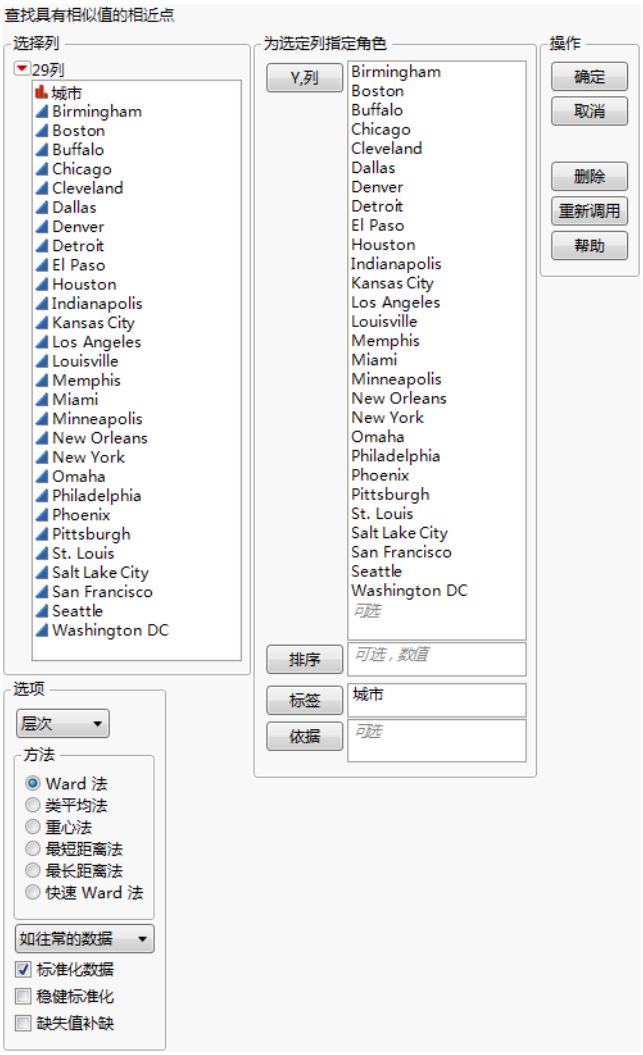
距离矩阵的正确数据表结构包含以下项：

- 具有“字符型”数据类型的标识符列（通常为第一列）。
- 一组 *n* 个列，其中 *n* 也是行数。这 *n* 个列定义对角线上值为 0 或缺失值的对称矩阵。

注意到 Flight Distances.jmp 中的距离矩阵遵循前面的格式。

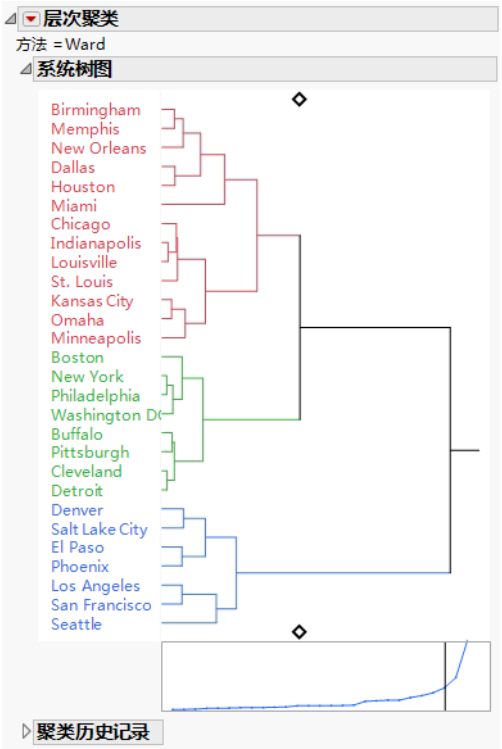
1. 选择**帮助** > **样本数据库**，然后打开 Flight Distances.jmp。
2. 选择**分析** > **聚类** > **层次聚类**。
3. 在启动窗口左下角的列表中，将**如往常的数据**更改为**数据为距离矩阵**。
4. 选择**城市**并点击**标签**。
5. 选择所有其余列并点击 **Y, 列**。

图 12.6 完成的 “距离矩阵”启动窗口



6. 点击**确定**。
7. 点击 “层次聚类”红色小三角并选择**聚类着色**。

图 12.7 “Flight Distances”的“系统树图”报表



在显示飞行距离的“系统树图”报表中，菱形的位置指示模型将城市分为三个聚类。聚类在系统树图中用不同颜色显示。有关如何解释报表的详细信息，请参见第 239 页的““系统树图”报表”。

使用“空间测度”进行晶片次品分类的示例

“层次聚类”平台中提供了称为“空间测度”的专业聚类选项。在本例中，您使用该选项对晶片聚类。有关该选项的信息，请参见第 248 页的“空间测度”。

1. 选择**帮助** > **样本数据库**，然后打开 Wafer Stacked.jmp。
2. 选择**分析** > **聚类** > **层次聚类**。
3. 在左下角的列表中，将**如往常的数据**更改为**数据被堆叠**。
   针对堆叠数据的更多选项显示在启动窗口中。
4. 选择**缺陷数**并点击 **Y, 列**。
5. 选择 **X_ 裸片**和 **Y_ 裸片**并点击**特性 ID**。
6. 选择**批**和**晶片**并点击**对象 ID**。
7. 从左下角的选项列表中选择**添加空间测度**。

图 12.8 完成的 “聚类”启动窗口

8. 点击**确定**。

图 12.9 “空间成分”窗口

缺陷数在 1423 个位置进行的测量，因此有 1423 个特性变量。

9. 点击**确定**接受 “空间”窗口中的选择。

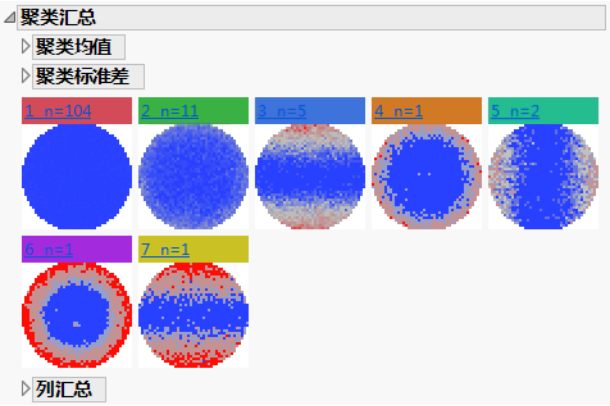
   两个窗口打开：“层次聚类”报表和 “Wafer Stacked 缺陷数空间”数据表。

10. 在 “系统树图”图中，点击并拖动顶部的菱形控点来探索各种数量的聚类。

当您拖动控点时，系统树图下方的距离图中的垂直线会移动到相应的聚类数。垂直坐标是在给定步生成的两个聚类之间的距离。图形看起来在聚类数为 7 时趋于水平。

11. 点击 “层次聚类”红色小三角并选择**聚类数**。
12. 输入 7 并点击**确定**。
13. 点击 “层次聚类”红色小三角并选择**聚类汇总**。

图 12.10 “聚类汇总”报表



晶片图指出每个聚类的次品数空间性质。“聚类 1”包含 104 个晶片，它们的次品数较少，次品均匀分布在晶片中。“聚类 3”有 5 个晶片，次品集中在上半圆和下半圆的两端处。您可以在聚类分析生成的数据表中查看各晶片及其 Hough 空间映射的图。请参见第 248 页的“空间测度”。

“层次聚类”平台的统计详细信息

- “空间测度”
- “距离方法公式”

空间测度

要使用“添加空间测度”选项，您的数据必须被堆叠并且包含两个对应于空间坐标的特性列。某些空间测度使用 Hough 变换构造。请参见 White et al. (2008) 和 Ballard (1981)。请参见第 246 页的“使用“空间测度”进行晶片次品分类的示例”。

“选择空间成分”窗口

若您在启动窗口中执行以下操作，则会显示 “选择空间成分”窗口：

- 选择“数据被堆叠”数据结构

- 指定对应于空间坐标的两列作为“特性 ID”
- 指定“对象 ID”
- 选择“添加空间测度”

在“选择空间成分”窗口中，您选择并权衡用于聚类分析的空间成分。这些成分用于构造聚类分析中使用的变量。随即打开一个新表，其中每个对象对应一行。该表包含每个对象的计算空间成分。

变量 聚类分析中构造并使用的变量类型。使用空间成分和响应 Y 构造变量。

特性 对于每个对象，针对每个位置（由两个“特性 ID”变量定义）计算的 Y 变量的值。

角度，饼图 反映楔形或半球形的变量。

半径，圆圈 反映圆形的变量。

条纹角度 反映具有相同角度的条纹的变量。

条纹位置 反映具有相同空间位置的条纹的变量。

曝光区中的位置 基于曝光区中裸片位置的变量。“曝光区中的位置”变量表示为 ShotPos[vShotSize, hShotSize]，其中 vShotSize 和 hShotSize 分别为定义的垂直和水平曝光区大小。

曝光区 标识对象所在矩形的变量，您可以指定对象在矩形中的水平位置和垂直位置数。术语**曝光区**用于半导体晶片数据中，以标识晶片上哪些裸片在一起成像。

输入“曝光区水平大小”和“曝光区垂直大小”的值。将水平曝光区大小指定为 4，垂直曝光区大小指定为 5，这样指示曝光区中最多有 20 个裸片。创建的总标识符数计算如下：

floor[(hSize+hShotSize-1)/hShotSize]* floor[(vSize+vShotSize-1)/vShotSize]

其中 hSize 和 vSize 分别是水平和垂直最大位置数，hShotSize = 曝光区水平大小，vShotSize = 曝光区垂直大小。

注意: 曝光区变量表示为 Shot[vert, horiz]，其中 vert 和 horiz 分别表示垂直和水平裸片位置。

数目 构造的给定类型的总变量数。

权重 用于确定聚类的给定类型的变量的重要性测度。

空间测度报表

当您在“选择空间成分”窗口中点击“确定”时，会出现两个窗口。

“层次聚类”报表

当您使用堆叠数据和两个特性 ID 执行分析时，“聚类汇总”报表显示 Y 变量的空间映射。每个图都是一个二维图，显示由“特性 ID”变量定义的每个位置的聚类均值。该图使用具有分位数尺度的“由蓝经灰到红”颜色梯度。使用分位数尺度可减轻离群值的影响。

空间数据表

空间测度的数据表为每个唯一对象 ID 提供一行。使用“由蓝经灰到红”默认颜色梯度显示列以表示 Y 变量。该表包含以下列：

对象 显示每个空间位置 （由两个 “特性 ID”变量定义）处的 Y 变量的热图的表达式列。

Hough 显示每个对象的 Hough 空间的热图的表达式列。请参见 White et al. (2008)。

空间测度 显示每个对象的计算值的每个空间测度对应的列。单元格按值着色。

距离方法公式

本节提供基于您在启动窗口中选择的方法计算距离的公式。有关方法说明，请参见第 236 页的“计算距离的方法”。

公式使用以下符号，小写符号通常与观测有关，大写符号通常与聚类有关：

n 是观测数

v 是变量数

x_i 是第 i 个观测

C_K 是第 K 个聚类，是 {1, 2,..., n} 的子集

N_K 是 C_K 中的观测数

$\bar{x}$ 是样本均值向量

$\bar{x}_K$ 是聚类 C_K 的均值向量

$\|x\|$ 是 **x** 各元素的平方和的平方根 （向量 **x** 的欧氏长度）

$d(x_i, x_j)$ 为 $\|x_i - x_j\|^2$

类平均法 类平均连接聚类法的距离计算如下：

$$D_{KL} = \sum_{i \in C_K} \sum_{j \in C_L} \frac{d(x_i, x_j)}{N_K N_L}$$

重心法 聚类重心法的距离计算如下：

$$D_{KL} = \left\| \bar{x}_K - \bar{x}_L \right\|^2$$

Ward 法 Ward 法距离计算如下：

$$D_{KL} = \frac{\left\| \bar{x}_K - \bar{x}_L \right\|^2}{\frac{1}{N_K} + \frac{1}{N_L}}$$

最短距离法 最短距离聚类法的距离计算如下：

$$D_{KL} = \min_{i \in C_K} \min_{j \in C_L} d(x_i, x_j)$$

最长距离法 最长距离聚类法的距离计算如下：

$$D_{KL} = \max_{i \in C_K} \max_{j \in C_L} d(x_i, x_j)$$

第 13 章

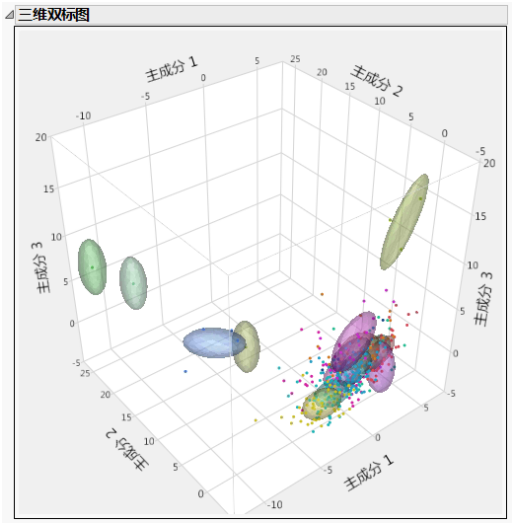
K 均值聚类

使用距离对观测分组

使用“K 均值聚类”平台可将在几个变量上享有相似值的观测分组在一起。*K*-均值法适用于包含大约 200 到 100,000 个观测的大型数据表。

“K 均值聚类”平台使用迭代算法对观测进行划分来构造指定数量的聚类。该方法称为 **K-均值**，它将观测划分聚类以便最小化到聚类重心的距离。您必须提前指定聚类数 *k*。不过，您可以比较不同 *k* 值的结果以选择适合您的数据的最优聚类数。

图 13.1 三维双标图



目录

“K 均值聚类”平台概述 255
对观测聚类的平台概述 255
“K 均值聚类”的示例 256
启动“K 均值聚类”平台 259
“迭代聚类”报表 260
“迭代聚类”选项 260
“迭代聚类”控制面板 261
K 均值报表 262
“聚类比较”报表 262
K 均值报表 262
“K 均值”报表选项 262
自组织图 264
“自组织图”控制面板 264
“自组织图”报表 265
SOM 算法的说明 265
“K 均值聚类”平台的更多示例 266
自组织图示例 266

“K 均值聚类”平台概述

“K 均值聚类”是 JMP 提供的对观测进行聚类的四个平台之一。有关四种方法的比较，请参见第 255 页的“对观测聚类的平台概述”。

“K 均值聚类”平台使用迭代拟合过程形成指定数量的聚类。*K*- 均值算法首先选择一组称为聚类种子的 k 个点作为对聚类均值的最初推测。每个观测被分配到最近的聚类种子，形成一组临时聚类。然后这些种子被聚类均值替代，各点会重新分配，最后该过程一直持续到聚类中不再有进一步的变化。

K-均值算法是 EM 算法的一种特殊情况，其中 *E* 代表期望，*M* 代表最大化。在 *K*-均值算法中，计算临时聚类均值代表期望步，将点分配给最近的聚类代表最大化步。

K-均值聚类仅支持数值列。*K*-均值聚类忽略建模类型（名义型和有序型），而将所有数值列视为连续型。

您必须提前指定聚类数 *k* 或 *k* 的值范围。不过，您可以比较不同 *k* 值的结果以选择适合您的数据的最优聚类数。

有关 *K* 均值聚类的背景信息，请参见 SAS Institute Inc.(2020d) 中的“FASTCLUS 过程”一章以及 Hastie et al.(2009)。

对观测聚类的平台概述

聚类是将在几个变量上享有相似值的观测分组在一起的一种多元方法。通常情况下，观测在 *p* 维空间内散布不均，其中 *p* 是变量数。这些观测反而会形成聚簇或聚类。标识出这些聚类使您可以更深层次地了解您的数据。

注意：JMP 还提供可以对变量聚类的平台。请参见第 295 页的“聚类变量”一章。

JMP 提供四个平台供您对观测聚类：

- “层次聚类”可用于最多包含数万行的小型表，并且允许字符数据。“层次聚类”将行按描绘为一棵树的层次序列形式进行组合。您可以在生成树后选择最适合您数据的聚类数。
- “K 均值聚类”适用于多达数百万行的大型表，并且只允许数值数据。您需要提前指定聚类数 *k*。该算法可以对聚类种子点做出推测。随后开始在将数据点分配到相应类别和重新计算聚类中心之间交替进行迭代过程。
- “正态混合”适用于数据来自重叠的多元正态分布的混合分布这种情况，并且只允许数值数据。对于具有多元离群值的情形，您可以使用假设具有均匀分布的离群值聚类。

  您需要提前指定聚类数。最大似然用于同时估计混合比例以及均值、标准差和相关性。为每个点指定属于每个组的概率。使用 EM 算法获取估计值。
- “潜在类分析”适用于大多数变量是分类变量这种情况。您需要提前指定聚类数。该算法拟合假定具有多项式混合分布的模型。为每个观测计算聚类成员关系的最大似然估计值。观测会被归类到其成员关系概率最大的聚类中。

表 13.1 聚类方法汇总

方法	数据类型或建模类型	数据表大小	指定聚类数
层次聚类	任意	使用快速 Ward 法，最多 200,000 行 使用其他方法，最多 5,000 行	否
K 均值聚类	数值	多达数百万行	是
正态混合	数值	任意大小	是
潜在类分析	名义型或有序型	任意大小	是

有些聚类平台提供用于处理数据中的离群值的选项。但是，若数据中有离群值，则最好在分析之前探索这些离群值。可以使用“探索离群值”实用工具完成该操作。详细信息，请参见《预测和专业建模》“建模工具”一章。

“K 均值聚类”的示例

在本例中，您使用 Cytometry.jmp 样本数据表通过 “K 均值聚类”对观测进行聚类。血细胞计数用于检测细胞表面的标记，这些标记的读数有助于诊断某些疾病。在本例中，基于血细胞计数分析中四种标记的读数来对观测进行分组。

1. 选择**帮助 > 样本数据库**，然后打开 Cytometry.jmp。
2. 选择**分析 > 聚类 > K 均值聚类**。
3. 选择 CD3、 CD8、 CD4 和 MCB，然后点击 **Y, 列**。
4. 点击**确定**。
5. 在**聚类数**旁边输入 3。
6. 在**聚类范围**旁边输入 15 （可选）。

   由于“聚类范围”设置为 15，平台提供 3 到 15 个聚类的拟合。您可以随后确定偏好的聚类数。
7. 点击**执行**。

图 13.2 “聚类比较”报表

聚类比较

方法	聚类数	CCC	最佳
K-均值聚类	3	23.1784	
K-均值聚类	4	8.80709	
K-均值聚类	5	29.5123	
K-均值聚类	6	52.5517	
K-均值聚类	7	49.5876	
K-均值聚类	8	56.5308	
K-均值聚类	9	54.053	
K-均值聚类	10	69.8707	
K-均值聚类	11	70.5239	最优 CCC
K-均值聚类	12	61.5326	
K-均值聚类	13	68.1277	
K-均值聚类	14	66.4044	
K-均值聚类	15	69.9928	

“聚类比较”报表显示在报表窗口顶部。最佳拟合由最高 CCC 值确定。在本例中，最佳拟合出现在拟合 11 个聚类时。

8. 滚动到 “K 均值聚类数 =11”报表。

图 13.3 “K 均值聚类数 =11”报表

K 均值聚类数=11

逐列统一尺度

聚类汇总

聚类	计数	步进	准则
1	816	24	0
2	482		
3	157		
4	577		
5	856		
6	498		
7	447		
8	549		
9	377		
10	123		
11	118		

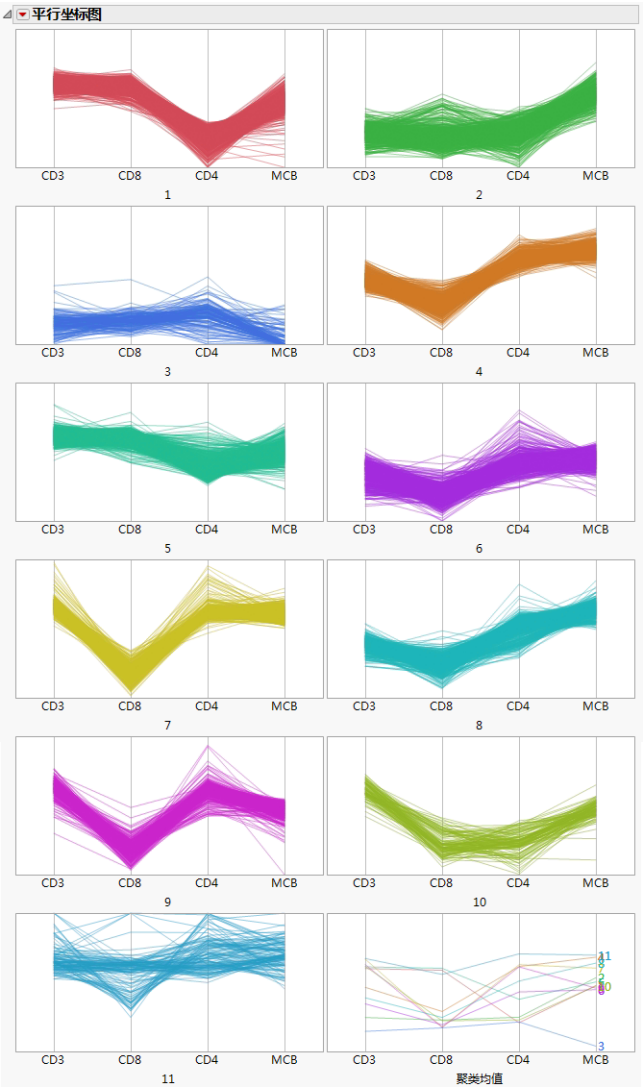
聚类均值

聚类	CD3	CD8	CD4	MCB
1	314.073529	300.270833	106.615196	183.4375
2	140.091286	116.365145	127.024896	205.014523
3	90.7898089	87.5477707	109.356688	15.0191083
4	247.426343	148.064125	314.074523	261.831889
5	320.287383	307.372664	193.117991	193.174065
6	188.686747	99.0863454	220.062249	171.682731
7	347.496644	89.9574944	320.782998	231.557047
8	210.258652	126.125683	260.327869	245.588342
9	328.206897	89.7824934	312.909814	175.965517
10	322.105691	113.715447	116.666667	181.731707
11	349.864407	284.813559	360.932203	267.474576

“聚类汇总”报表显示 11 个聚类中每一个聚类的观测数。“聚类均值”报表显示每个聚类的四个标记读数的均值。

9. 点击 “K 均值聚类数 =11”红色小三角，然后选择**平行坐标图**。

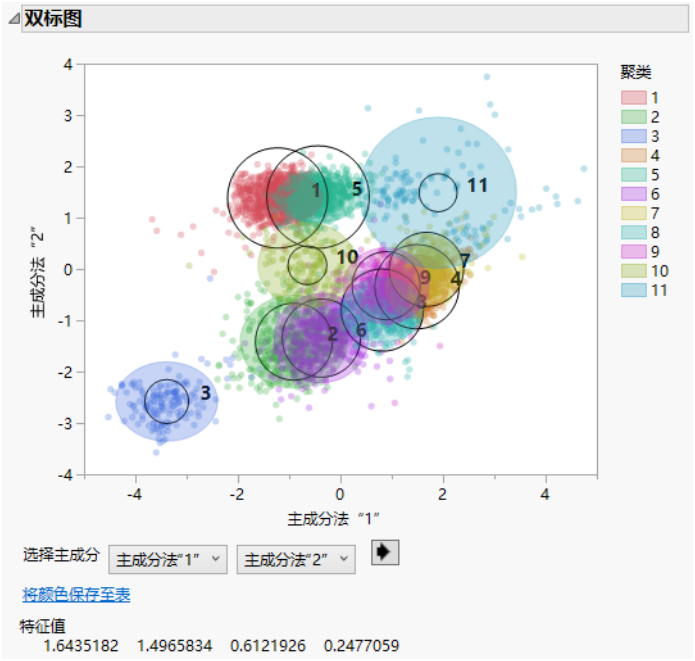
图 13.4 血细胞计数数据的 “平行坐标图”



“平行坐标图”显示每个聚类中观测的结构。使用这些图可查看聚类有哪些不同。聚类 4、6、7、8 和 9 往往有相对低的 CD8 值和高的 CD4 值。另一方面，聚类 1 有更高的 CD8 值和更低的 CD4 值。

10. 点击 “K 均值聚类数 =11” 红色小三角，然后选择**双标图**。

图 13.5 血细胞计数数据的 “双标图”

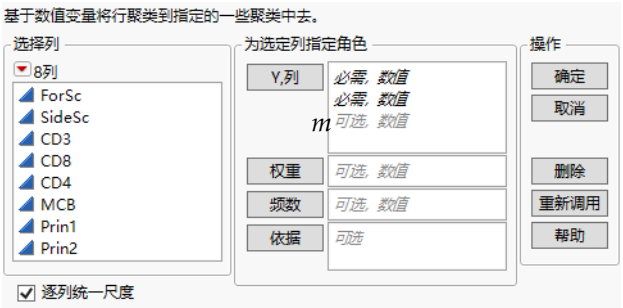


标识聚类颜色的图例显示在图的右侧。基于聚类前两个主成分，聚类 3、 10 和 11 显示为与其他聚类分的最开。图 13.4 中这几个聚类的平行坐标图支持该结论，它们与其他聚类的图有所不同。使用该图下面的列表可查看其他主成分组合的双标图。

启动 “K 均值聚类” 平台

通过选择**分析** > **聚类** > **K 均值聚类**来启动 “K 均值聚类” 平台。

图 13.6 “K 均值聚类” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 用于对观测聚类的变量。

注意：*K*- 均值聚类仅支持数值列。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，列中的数值为分析中的每行分配一个频数。

依据 一列，其水平定义不同的分析。对于指定列的每个水平，都分析相应行。结果显示在不同的报表中。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独分析。

启动窗口选项

逐列统一尺度 独立于其他列调整每列的尺度。在变量不共享相同的测量值尺度，而您不希望一个变量支配聚类过程的情况下使用。例如，一个变量具有介于 0 和 1000 之间的值，而另一个变量具有介于 0 和 10 之间的值。在这种情况下，您可以使用该选项，以便聚类过程不被第一个变量所支配。

当您点击“确定”时，“控制面板”随即显示。请参见第 261 页的““迭代聚类”控制面板”。

“迭代聚类”报表

当您在启动窗口中点击“确定”时，会出现“迭代聚类”报表窗口，该窗口显示用于拟合模型的“控制面板”。请参见第 261 页的““迭代聚类”控制面板”。在拟合模型时，其他报表会添加至窗口。请参见第 262 页的“K 均值报表”。

“迭代聚类”选项

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

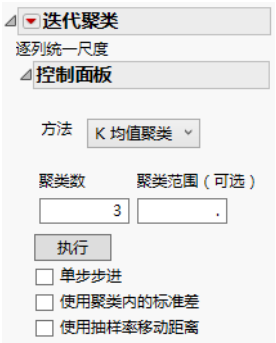
保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“迭代聚类”控制面板

Cytometry.jmp 数据表的 “控制面板”显示在图 13.7 中。您可以反复拟合不同数量的聚类，也可以使用 “聚类范围”选项指定一个范围。

图 13.7 “迭代聚类”控制面板



“控制面板”包含以下选项：

方法 下列聚类方法可用：

K 均值聚类 在本章中进行了说明。

自组织图 在第 264 页的““自组织图”控制面板”中进行了说明。

聚类数 指定要形成的聚类数。

聚类范围 （可选） 提供要形成的聚类数的上限。若在此输入了某个数字，平台将为“聚类数”与 “聚类范围 （可选）”中输入的值之间的每个整数创建单独的分析。

执行 除非选择了 “单步步进”，否则自动拟合聚类。

单步步进 支持您每次执行一个迭代来逐步完成聚类过程。当您选择 “单步步进”并点击 “确定”时，会显示 “K 均值聚类”报表，没有聚类结果，而是含有 “执行”和 “步进”按钮。

- 点击“步进”按钮每次执行一个迭代来逐步完成聚类过程。
- 点击“执行”按钮自动拟合聚类。

使用聚类内的标准差 针对每个聚类内的观测，使用每个变量在聚类内的估计标准差对距离统一尺度。若不选择该选项，距离按照每个变量的标准差的总估计值统一尺度。

使用抽样率移动距离 基于聚类大小调整距离。若您具有大小不等的聚类，则观测应具有更高的概率被分配给较大的聚类，因为观测来自较大聚类的先验概率更高。

K 均值报表

当您在 “控制面板” 中点击 “执行” 时，会显示以下报表：

- “聚类比较” 报表。请参见第 262 页的 “ “聚类比较” 报表” 。
- 一个或多个 K 均值报表。报表被动态地命名为 “K 均值聚类数=<k>” ，具体取决于 *k*，即拟合的聚类数。每次执行拟合时，都会出现 “K 均值聚类数=<k>” 报表。

Cytometry.jmp 数据表（其中变量 CD3 到 MCB 作为 “Y, 列” ）的 “聚类比较” 报表和 “K 均值聚类数=11” 报表显示在图 13.2 和第 257 页的 “ “K 均值聚类数=11” 报表” 中。

“聚类比较” 报表

“聚类比较” 报表提供拟合统计量以比较各种模型。拟合统计量为 “三次聚类准则” (CCC)。CCC 值越大表示拟合效果越好。最佳拟合在名为 “最佳” 的列中用最优 CCC 指示。请参见 SAS Institute Inc. (1983)。CCC 计算中不包括常数列。

K 均值报表

每个 “K 均值” 报表为每个聚类提供以下汇总统计量：

- “聚类汇总” 报表提供聚类数和每个聚类中的观测数，以及所需的迭代次数。
- “聚类均值” 报表为每个变量提供每个聚类中观测的均值。
- “聚类标准差” 报表为每个变量提供每个聚类中观测的标准差。

“K 均值” 报表选项

每个 “K 均值” 报表包含下列选项：

双标图 以数据的前两个主成分显示点和聚类的图，以及用于标识聚类颜色的图例。围绕聚类中心绘制圆圈，而且圆圈的大小与聚类内的计数成比例。着色区域是围绕均值的密度等高线。默认情况下，该区域指示该聚类中 90% 的观测所在的位置 (Mardia et al. 1980)。使用该图下方的列表可将图轴改为其他主成分。或者，使用箭头按钮在所有可能的轴组合之间循环切换。该图之下还有一个用于将聚类颜色保存到数据表的选项。请参见第 263 页的 “将颜色保存至表” 。特征值以降序方式显示。

注意：若在启动窗口中选中 “逐列统一尺度” ，则双标图使用相关性矩阵。若未选中 “逐列统一尺寸” ，则双标图使用协方差矩阵。

双标图选项 包含用于控制 “双标图” 外观的下列选项：

显示双标图射线 显示双标图射线。带标签的射线显示协变量在由主成分定义的子空间中的方向。它们表示每个变量与每个主成分的关联程度。

双标图射线位置 可让您指定双标图射线的位置和射线尺度。默认情况下，这些射线从点 (0,0) 发出。在该图中，您可以拖动射线或使用该选项指定坐标。您还可以使用“射线尺度”选项调整射线的尺度，以便更加清晰地显示。

双标图等高线密度 支持您指定密度等高线的置信水平。默认置信水平为 90%。

标记聚类 将标识聚类的标记分配给数据表的行。

三维双标图 显示数据的三维双标图。仅当有三个或更多变量时可用。

平行坐标图 为每个聚类创建平行坐标图。图报表提供用于显示和隐藏数据和均值的选项。请参见 《基本绘图》。

散点图矩阵 使用所有 Y 变量创建散点图矩阵。

将颜色保存至表 将标识聚类的颜色分配给数据表的行。若报表窗口中有双标图，保存到数据表的颜色将与双标图中的聚类颜色相匹配。若双标图中的颜色改变，而且再次选定“将颜色保存至表”选项，那么表中的颜色将更新以便与双标图中的那些颜色相匹配。

注意：选定任何保存选项时，每个保存的列都包含一个“注释”列属性，该属性指定这一特定列数据的聚类数。这使您能够保存来自多个聚类拟合的列，并使用列属性来标识保存的列来自哪个聚类拟合。

保存聚类 将以下两列保存至数据表：

- **聚类**列包含分配了给定行的聚类的编号。
- （不适用于“自组织图”。）**距离**列包含给定观测与其聚类均值之间的欧氏距离平方。对于每个变量，将观测值与该变量的聚类均值之间的差值除以该变量的总标准差。然后对所有变量的这些统一尺度的差值进行平方与求和计算。

保存聚类距离 （不适用于“自组织图”。）将**距离**列保存到数据表中。该列与从**保存聚类**选项获得的**距离**列相同。

保存聚类公式 将名为“聚类公式”的公式列保存至数据表。这是标识每个聚类的聚类成员关系的公式。

保存距离公式 （不适用于“自组织图”。）将名为“距离公式”的公式列保存至数据表。这是用于计算到所分配聚类的距离的公式。

保存 K 聚类距离 （不适用于“自组织图”。）保存包含到每个聚类中心的欧氏距离平方的 *k* 列。

保存 K 距离公式 （不适用于“自组织图”。）保存包含到每个聚类中心的欧氏距离平方的公式的 *k* 列。

发布聚类公式 向“公式存储库”发布在“保存聚类公式”选项中使用的相同的得分代码。

模拟聚类 使用聚类均值和标准差，创建包含 Y 变量的模拟聚类观测的新数据表。

删除 删除聚类报表。

自组织图

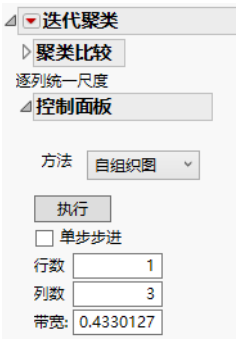
自组织图 (SOM) 方法最初由 Teuvo Kohonen (1989, 1990) 开发，之后被另外一些神经网络爱好者和统计学家进一步推广。最初的 SOM 被视为一个学习过程，就像最初的神经网络算法一样，但在此使用的版本是 *K*- 均值聚类的变异。在 SOM 文献中该变异称为使用**局部加权线性平滑统计量**的**批处理算法**。

SOM 的目标不仅仅是要在聚类网格上以特定的布点形成聚类，那些在 SOM 网格中彼此邻近的聚类中的点在多元空间中也是彼此邻近的。在经典的 *K*-均值聚类中，聚类结构是任意的，但在 SOM 中，聚类具有网格结构。该网格结构可帮助在二维中解释聚类：相距较近的聚类比起相距较远的聚类要更类似。请参见第 265 页的“SOM 算法的说明”。

“自组织图”控制面板

从“迭代聚类控制面板”中的“方法”列表中选择“自组织图”选项。

图 13.8 “自组织图”控制面板



第 261 页的“‘迭代聚类’控制面板”中说明了面板上的部分选项。其余选项在下文中进行说明。

行数 聚类网格中的行数。

列数 聚类网格中的列数。

带宽 指定邻近聚类对预测重心的影响。较小的带宽会对更近的聚类指定更大的权重。

“自组织图”报表

图 13.9 “自组织图”报表

迭代聚类

聚类比较

方法	聚类数	行数	CCC	最佳
自组织图	6	2	2.87879	最优 CCC

逐列统一尺度

控制面板

2 X 3 的自组织图网格

逐列统一尺度

带宽: 0.4330127

聚类汇总

聚类	计数	步进	准则
1	28	12	0
2	20		
3	31		
4	21		
5	28		
6	22		

聚类均值

聚类	萼片长度	萼片宽度	花瓣长度	花瓣宽度
1	6.98187076	3.15339949	5.85011658	2.16765916
2	6.14554483	2.62528195	5.11988902	1.79042135
3	6.20930867	3.05499872	4.51264582	1.46300228
4	5.46721399	2.56978376	3.79572186	1.15921467
5	5.22582725	3.67826605	1.47738883	0.27226749
6	4.7258207	3.10958329	1.44133246	0.21218254

聚类标准差

“聚类比较报表”显示聚类总数以及请求的行数。SOM 报表根据请求的网格大小进行命名。“带宽”在 “SOM 网格”报表的顶部给出。该报表本身类似于 “K 均值聚类数”报表。请参见第 262 页的 “K 均值报表”。

有关“自组织图”红色小三角选项的详细信息，请参见第 262 页的“ “K 均值”报表选项”。

SOM 算法的说明

本节包含在 JMP 中实现 SOM 的步骤。

- 初始聚类种子采用可为多维空间提供良好覆盖率的方式进行选择。JMP 使用主成分来确定两个方向，用来捕获数据中的大多数变异。
- 随后，JMP 在这个主成分空间中展开一个网格，网格边缘与每个方向的中心相距 2.5 个标准差。通过将该网格转换回变量初始空间来确定聚类种子。
- 按照 *K*-均值方法那样继续聚类分配，将每个点分配给最靠近它的聚类。
- 像使用 *k* 均值方法那样为每个聚类估计均值。JMP 随后使用这些均值设置加权回归，它将每个变量设置为回归中的响应，并将 SOM 网格坐标设置为回归变量。权重函数使用核心函数将较大权重分配给要估计其中心的聚类，将较小权重分配给 SOM 网格中距离该聚类较远的聚类。新聚类均值是从该回归得到的预测值。

- 这些迭代继续执行，直到该过程收敛。

“K 均值聚类”平台的更多示例

自组织图示例

本例使用 iris.jmp 样本数据表，其中包括三种鸢尾花的萼片长度、萼片宽度、花瓣长度和花瓣宽度的测量值。

1. 选择**帮助** > **样本数据库**，然后打开 Iris.jmp。
2. 选择**分析** > **聚类** > **K 均值聚类**。
3. 选择**萼片长度**、**萼片宽度**、**花瓣长度**和**花瓣宽度**，然后点击 **Y，列**。
4. 点击**确定**。
5. 从“控制面板”上的“方法”菜单中选择**自组织图**。
6. 设置**行数**等于 1，**列数**等于 2。
7. 点击**执行**。
8. 打开“控制面板”报表。
9. 设置**行数**等于 1，**列数**等于 3。
10. 点击**执行**。
11. 打开“控制面板”报表。
12. 设置**行数**等于 2，**列数**等于 2。
13. 点击**执行**。

图 13.10 SOM 聚类比较

聚类比较

方法	聚类数	行数	CCC	最佳
自组织图	2	1	3.35952	
自组织图	3	1	4.95837	最优 CCC
自组织图	4	2	3.64938	

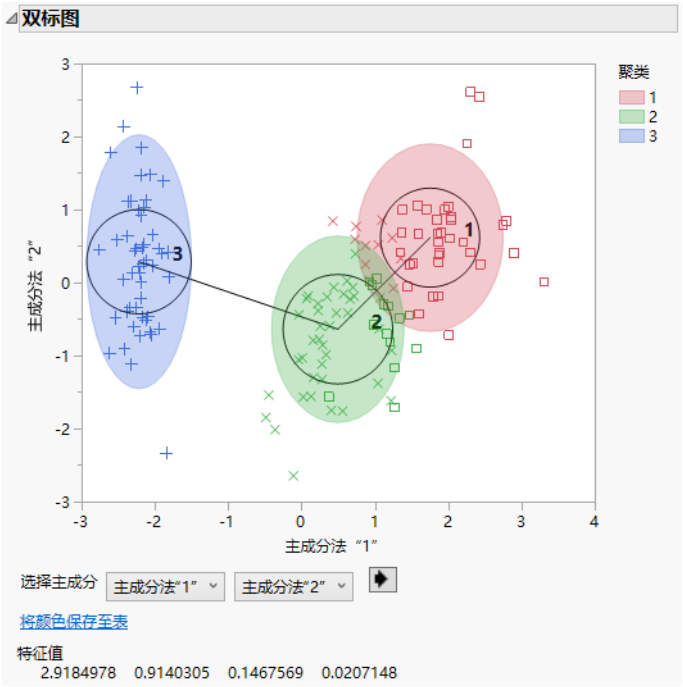
“聚类比较”报表显示在报表窗口顶部。最佳拟合由最高 CCC 值确定。请注意：给出最大 CCC 的聚类编号为 3，即物种数。

14. 滚动到“1 x 3 的自组织图网格”报表。我们可以看到该分类不完美；每个聚类应表示每个物种，每个聚类有 50 行。

图 13.11 Iris.jmp 的 “自组织图”报表

15. 在该数据表中，选择**物种**列并选择**行 > 按列设定颜色或标记**。
16. 选择 “标记”下面的**经典**选项。
17. 点击**确定**。
18. 点击 “1 x 3 的自组织图网格”旁边的红色小三角菜单，然后选择**双标图**。

图 13.12 SOM 双标图



我们可以看到 “聚类 3”中的所有行都正确标识为 setosa 物种。另外两个物种，virginica 和 versicolor 略有重叠，可能彼此混淆。

第 14 章

正态混合

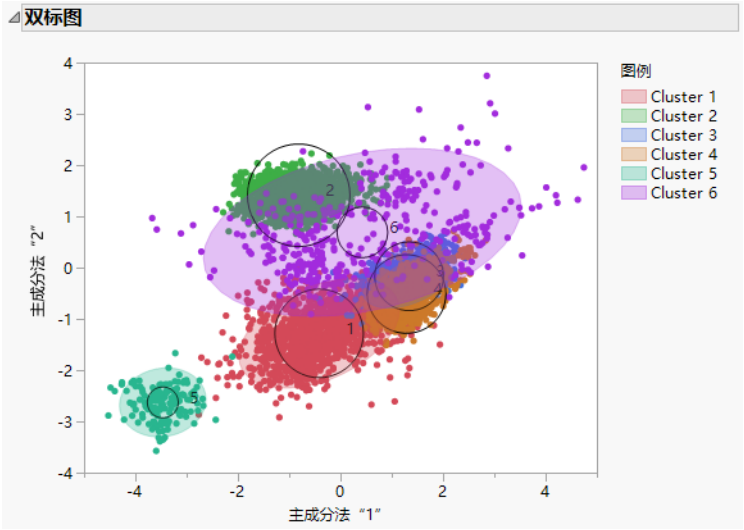
使用概率对观测分组

当数据来自重叠正态分布时，使用“正态混合”进行聚类。您需要提前指定聚类数。

“正态混合”是基于假设使用多元正态分布的混合分布近似计算观测的联合概率分布的迭代方法。这些混合代表不同的聚类。各聚类都具有多元正态分布。

若聚类分隔地很好，层次聚类和 *K*-均值聚类都适用。但若聚类重叠，则正态混合是更好的替代方法，因为它基于聚类成员概率，而不是基于边界的任意聚类分配。

图 14.1 正态混合双标图



目录

“正态混合聚类”平台概述 271
对观测聚类的平台概述 271
“正态混合聚类”的示例 272
启动“正态混合聚类”平台 274
“基于聚类的模型”报表 275
“基于聚类的模型”选项 275
“基于聚类的模型”控制面板 276
“正态混合”报表 277
“聚类比较”报表 277
“正态混合”报表 277
“正态混合”报表选项 277
“正态混合聚类”平台的统计详细信息 279

“正态混合聚类”平台概述

“正态混合”是 JMP 提供的对观测进行聚类的四个平台之一。有关四种方法的比较，请参见第 271 页的“对观测聚类的平台概述”。

正态混合是针对数值变量的迭代聚类方法。不过，它也可预测每个聚类内期望的响应比例。正态混合假设测量值列的联合概率分布可以由多元正态的混合分布近似计算，混合分布代表了不同的聚类。为每个聚类估计均值向量和协方差矩阵。请参见 McLachlan and Krishnan (1997) 和Hand et al. (2001)中的第 9.6 节。

注意：“正态混合”算法涉及的迭代首先要对聚类中心进行随机猜测。因此，每次运行分析的结果可能略有不同。

若您怀疑有多元离群值，可使用两个选项。您可以使用离群值聚类或“探索离群值”实用工具。离群值聚类选项假定服从均匀分布，与标准的“正态混合”选项相比对离群值更不敏感。“探索离群值”实用工具支持您在分析之前探索和处理离群值。请参见第 276 页的“离群值聚类”和《预测和专业建模》。

对观测聚类的平台概述

聚类是将在几个变量上享有相似值的观测分组在一起的一种多元方法。通常情况下，观测在 p 维空间内散布不均，其中 p 是变量数。这些观测反而会形成聚簇或聚类。标识出这些聚类使您可以更深层次地了解您的数据。

注意：JMP 还提供可以对变量聚类的平台。请参见第 295 页的“聚类变量”一章。

JMP 提供四个平台供您对观测聚类：

- “层次聚类”可用于最多包含数万行的小型表，并且允许字符数据。“层次聚类”将行按描绘为一棵树的层次序列形式进行组合。您可以在生成树后选择最适合您数据的聚类数。
- “K 均值聚类”适用于多达数百万行的大型表，并且只允许数值数据。您需要提前指定聚类数 k。该算法可以对聚类种子点做出推测。随后开始在将数据点分配到相应类别和重新计算聚类中心之间交替进行迭代过程。
- “正态混合”适用于数据来自重叠的多元正态分布的混合分布这种情况，并且只允许数值数据。对于具有多元离群值的情形，您可以使用假设具有均匀分布的离群值聚类。

  您需要提前指定聚类数。最大似然用于同时估计混合比例以及均值、标准差和相关性。为每个点指定属于每个组的概率。使用 EM 算法获取估计值。
- “潜在类分析”适用于大多数变量是分类变量这种情况。您需要提前指定聚类数。该算法拟合假定具有多项式混合分布的模型。为每个观测计算聚类成员关系的最大似然估计值。观测会被归类到其成员关系概率最大的聚类中。

表 14.1 聚类方法汇总

方法	数据类型或建模类型	数据表大小	指定聚类数
层次聚类	任意	使用快速 Ward 法，最多 200,000 行 使用其他方法，最多 5,000 行	否
K 均值聚类	数值	多达数百万行	是
正态混合	数值	任意大小	是
潜在类分析	名义型或有序型	任意大小	是

有些聚类平台提供用于处理数据中的离群值的选项。但是，若数据中有离群值，则最好在分析之前探索这些离群值。可以使用“探索离群值”实用工具完成该操作。详细信息，请参见《预测和专业建模》“建模工具”一章。

“正态混合聚类”的示例

血细胞计数用于测量细胞的各种特征。细胞标记测量值有助于诊断特定疾病。在本例中，您基于血细胞计数分析中四种标记的读数来对观测进行聚类。

1. 选择**帮助** > **样本数据库**，然后打开 Cytometry.jmp。
2. 选择**分析** > **聚类** > **正态混合**。
3. 选择 CD3、 CD8、 CD4 和 MCB，然后点击 **Y, 列**。
4. 点击**确定**。
5. 在**聚类数**旁边输入 6。
6. 点击**执行**。

注意：您的结果可能有所不同，因为算法具有随机起始值。

图 14.2 “正态混合聚类数 =6”报表

基于聚类的模型

聚类比较

方法	聚类数	BIC	AICc	最佳
正态混合	6	208033	207456	最小 BIC 最小 AICc

控制面板

正态混合聚类数=6

聚类汇总

聚类	计数	比例
1	393	0.08550
2	944	0.18467
3	1194	0.24341
4	147	0.02932
5	720	0.14049
6	1602	0.31661

p

聚类均值

聚类	CD3	CD8	CD4	MCB
1	336.456533	193.385204	238.384846	206.503738
2	233.766979	140.184238	298.679748	256.78221
3	173.951704	109.760064	187.942955	195.539918
4	87.8647731	86.2326743	107.791654	10.1473172
5	338.61353	86.6669975	315.560339	208.345296
6	317.251206	306.00611	150.866465	189.617978

聚类标准差

-对数似然	BIC	AICc
103637.52	208033.06	207456.3

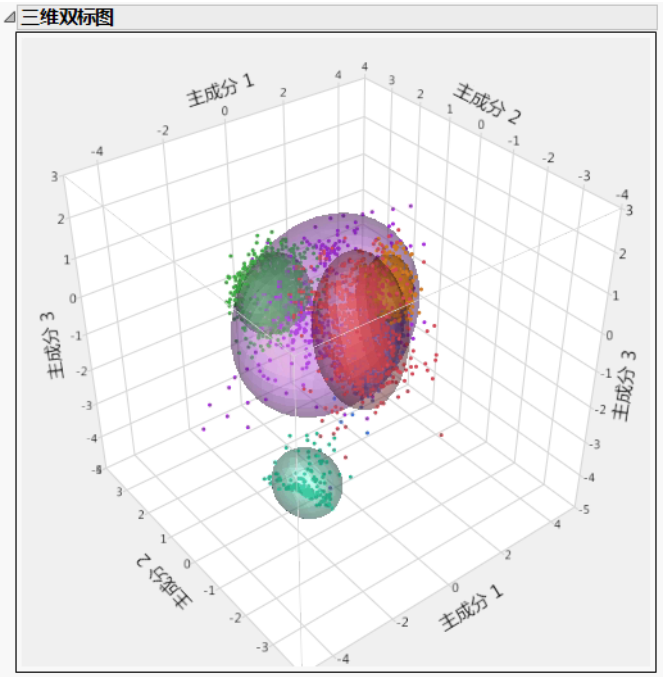
正态混合的相关性

“聚类汇总”报表显示 6 个聚类中每一个聚类的观测数。“聚类均值”报表显示每个聚类的四个标记读数的均值。

7. 点击“正态混合聚类数 =6”旁边的红色小三角，然后选择**三维双标图**。

注意：您的三维双标图可能有所不同，因为算法具有随机起始值。

图 14.3 血细胞计数数据的 “三维双标图”

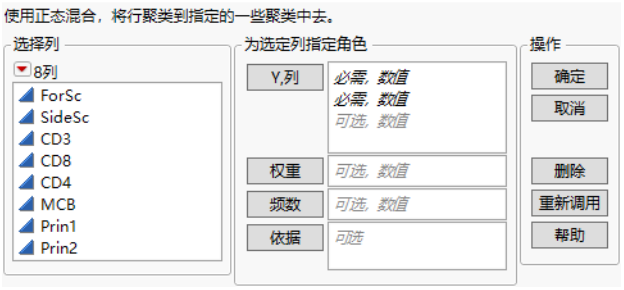


该图显示聚类的拟合正态密度等高线。注意到依据前 3 个主成分，一个聚类明显与其他聚类分开。

启动 “正态混合聚类” 平台

通过选择 “分析 > 聚类 > 正态混合”，启动 “正态混合聚类” 平台。

图 14.4 “正态混合” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 用于对观测聚类的变量。

注意：正态混合聚类仅支持数值列。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，列中的数值为分析中的每行分配一个频数。

依据 一列，其水平定义不同的分析。对于指定列的每个水平，都分析相应行。结果显示在不同的报表中。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独分析。

当您点击“确定”时，“控制面板”随即显示。请参见第 276 页的“‘基于聚类的模型’控制面板”。

“基于聚类的模型”报表

当您在启动窗口中点击“确定”时，“基于聚类的模型”报表窗口随即打开，该窗口显示用于拟合模型的“控制面板”。请参见第 276 页的“‘基于聚类的模型’控制面板”。在拟合模型时，其他报表会添加至窗口。请参见第 277 页的“‘正态混合’报表”。

“基于聚类的模型”选项

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

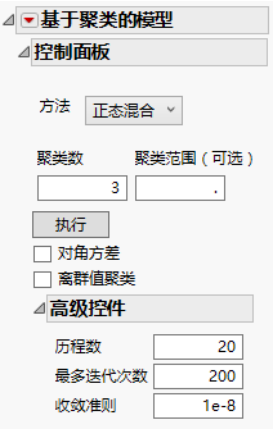
重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

“基于聚类的模型”控制面板

Cytometry.jmp 数据表的“控制面板”（变量 CD3 到 MCB 作为 Y, 列）显示在图 14.5 中。您可以通过“控制面板”反复拟合各种数量的聚类，也可以使用“聚类范围”选项指定一个范围。

图 14.5 “正态混合”方法的“控制面板”



“基于聚类的模型”控制面板有以下选项：

聚类数 指定要形成的聚类数。

聚类范围（可选） 提供要形成的聚类数的上限。若在此输入了某个数字，平台将为**聚类数**和**聚类范围（可选）**中输入的值之间的每个整数创建单独的分析。

执行 拟合聚类。

对角方差 将协方差矩阵的非对角线元素限定为 0。该平台拟合变量之间无相关性的多元正态分布。

注意：有时候需要使用“对角方差”选项，这样在观测数少于变量数时可避免生成奇异的协方差矩阵。该选项也可用于避免对大量变量估计非常大的协方差矩阵。

离群值聚类 拟合聚类，将未能落入任何正常聚类中的离群值收拢其中。若创建了该聚类，则将其指定为“聚类 0”，并且观测计数显示在“聚类汇总”报表中。假定落入离群值聚类中的观测在包含这些观测的超立方中服从均匀分布。

高级控件 以下高级控件可用：

历程数 估计过程的独立重新开始数。每次重新开始具有不同的起始值。独立开始有助于防止求得局部解。

最多迭代次数 EM 算法收敛阶段的最多迭代次数。

收敛准则 EM 迭代停止时的似然差值。

“正态混合”报表

当您在 “控制面板”中点击 “执行”时，会显示以下报表：

- “聚类比较”报表。请参见第 277 页的“ “聚类比较”报表”
- 一个或多个正态混合报表。报表被动态地命名为“正态混合聚类数=<k>”，具体取决于 *k*，即拟合的聚类数。每次执行拟合时，都会出现“正态混合聚类数=<k>”报表。

Cytometry.jmp 数据表（其中变量 CD3 到 MCB 作为 Y,列）的“聚类比较”报表和“正态混合聚类数=6”报表显示在图 14.2 和第 273 页的“ “正态混合聚类数=6”报表”中。

“聚类比较”报表

“聚类比较”报表提供拟合统计量以比较各种模型。拟合统计量为 BIC 和 AICc。每个统计量的值越小表示拟合效果越好。最佳拟合在名为最佳的列中指示。

“正态混合”报表

“正态混合”报表提供每个聚类的汇总统计量：

- “聚类汇总”报表提供每个聚类的观测数和比例。
- “聚类均值”报表为每个变量提供每个聚类中观测的均值。
- “聚类标准差”报表为每个变量提供每个聚类中观测的标准差。
- “负对数似然”表提供负对数似然、BIC 和 AICc。请参见《拟合线性模型》。
- “正态混合的相关性”报表提供每个聚类的估计相关性矩阵。

“正态混合”报表选项

双标图 以数据的前两个主成分显示点和聚类的图，以及用于标识聚类颜色的图例。围绕聚类中心绘制圆圈，而且圆圈的大小与聚类内的计数成比例。着色区域是围绕均值的密度等高线。默认情况下，该区域指示该聚类中 90% 的观测所在的位置 (Mardia et al. 1980)。使用该图下方的列表可将图轴改为其他主成分。或者，使用箭头按钮在所有可能的轴组合之间循环切换。该图之下还有一个用于将聚类颜色保存到数据表的选项。请参见第 278 页的“将颜色保存至表”。特征值以降序方式显示。

注意：双标图始终使用相关性矩阵计算主成分。

双标图选项 包含用于控制双标图外观的选项。

显示双标图射线 显示双标图射线。带标签的射线显示协变量在由主成分定义的子空间中的方向。它们表示每个变量与每个主成分的关联程度。

双标图射线位置 可让您指定双标图射线的位置和射线尺度。默认情况下，这些射线从点 (0,0) 发出。在该图中，您可以拖动射线或使用该选项指定坐标。您还可以使用“射线尺度”选项调整射线的尺度，以便更加清晰地显示。

双标图等高线密度 支持您指定密度等高线的置信水平。默认置信水平为 90%。

标记聚类 将标识聚类的标记分配给数据表的行。

三维双标图 显示数据的三维双标图。仅当有三个或更多变量时可用。

平行坐标图 为每个聚类创建平行坐标图。图报表提供用于显示和隐藏数据和均值的选项。请参见 《基本绘图》。

散点图矩阵 使用所有 Y 变量创建散点图矩阵。

将颜色保存至表 将标识聚类的颜色分配给数据表的行。若报表窗口中有双标图，保存到数据表的颜色将与双标图中的聚类颜色相匹配。若双标图中的颜色改变，而再次选定“将颜色保存至表”选项，那么表中的颜色将更新以便与双标图中的那些颜色相匹配。

注意：选定任何保存选项时，每个保存的列都包含一个“注释”列属性，该属性指定这一特定列数据的聚类数。这使您能够保存来自多个聚类拟合的列，并使用列属性来标识保存的列来自哪个聚类拟合。

保存聚类 将名为**聚类**的列添加到数据表，该列包含分配了给定行的聚类的编号。对于正态混合，这是最有可能的聚类。

保存聚类公式 将名为**聚类公式**的公式列添加至数据表。该公式标识行属于哪个聚类。

发布聚类公式 向“公式存储库”发布在“保存聚类公式”选项中使用的相同的得分代码。若已选定“发布聚类公式”并且从“公式存储库”中的模型选择“运行脚本”后，保存到数据表的列应与选定“保存聚类公式”时保存的那些列相匹配。

保存混合概率 为每个聚类添加名为**概率聚类 <k>** 的列，相应列包含观测属于该聚类的概率。

保存混合公式 将列添加至数据表，这些列包含用于计算混合概率的公式。使用这些公式列对已排除数据或添加到表中的数据的概率进行评分。

距离公式 <k> 在观测处求得的“聚类 <k>”的多元正态密度函数的估计。

总距离 距离公式列的总和。该列中的公式等价于由“保存密度公式”选项创建的**混合密度**列中的公式。

概率公式 <k> 观测属于“聚类 <k>”的概率。这些列包含为由“保存混合概率”选项创建的**概率聚类 <k>** 列提供值的公式。计算混合概率的列公式为：

$$\text{概率公式 <k>} = \frac{\text{距离公式 <k>}}{\text{总距离}}$$

保存密度公式 将名为**混合密度**的列添加至数据表，该列包含正态混合的估计密度函数。

模拟聚类 使用混合密度模拟预测变量值。将这些值和值所属的聚类保存至新数据表中。

删除 删除聚类报表。

“正态混合聚类”平台的统计详细信息

“正态混合”使用 EM 算法执行拟合，因为该算法比 Newton-Raphson 算法更稳定。此外，JMP 使用 EM 算法的 Bayes 正则版本，该版本允许我们顺利处理协方差矩阵奇异的情况。由于估计值极大地依赖于初始推测数，该平台将迭代若干历程，每一历程都随机选择一些点作为初始中心。

执行多个历程会令估计过程花费较长时间，所以处理较大问题时需要耐心等待。控件支持您指定历程和迭代限制。

第 15 章

潜在类分析

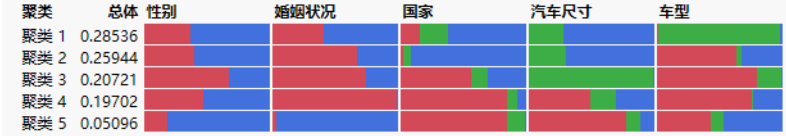
将分类变量的观测进行分组

潜在类分析可使您针对分类响应变量进行观测值的聚类。潜在变量是无法观测的分组变量。潜在变量的每个水平称为潜在类。“潜在类分析”平台拟合潜在类模型并确定每个观测的最可能的聚类或潜在类。在大多数情况下，主题专家使用潜在类分析的结果基于每个类的特征创建相应潜在类的定义。

图 15.1 “潜在类分析”的示例

参数估计值

		性别		婚姻状况		国家			汽车尺寸			车型		
聚类	总体	Female	Male	Married	Single	American	European	Japanese	Large	Medium	Small	Family	Sporty	Work
聚类 1	0.28536	0.3763	0.6237	0.4160	0.5840	0.1556	0.2307	0.6138	0.0009	0.2840	0.7151	0.0188	0.9639	0.0174
聚类 2	0.25944	0.4066	0.5934	0.6738	0.3262	0.0269	0.0562	0.9169	0.0029	0.2941	0.7030	0.6418	0.0369	0.3212
聚类 3	0.20721	0.6792	0.3208	0.7527	0.2473	0.5619	0.1376	0.3006	0.0025	0.9913	0.0062	0.8013	0.1950	0.0037
聚类 4	0.19702	0.4706	0.5294	0.9921	0.0079	0.8531	0.0827	0.0642	0.4963	0.2045	0.2991	0.7650	0.0020	0.2330
聚类 5	0.05096	0.1839	0.8161	0.0324	0.9676	0.8477	0.1470	0.0053	0.7764	0.1180	0.1057	0.4363	0.0958	0.4679



目录

“潜在类分析”平台概述 283
“潜在类分析”的示例 283
启动“潜在类分析”平台 286
“潜在类分析”报表 287
“聚类比较”报表 287
潜在类模型报表 287
“潜在类分析”平台选项 289
“潜在类分析”选项 289
“潜在类模型”选项 289
“潜在类分析”平台的更多示例 290
绘制聚类成员关系的概率图 290
“潜在类分析”平台的统计详细信息 291
潜在类模型拟合 291
最大聚类数 292

“潜在类分析”平台概述

“潜在类分析”平台对分类响应变量拟合潜在类模型并确定每个观测的最可能的聚类或潜在类。**潜在变量**是无法观测的分组变量。潜在变量的每个水平称为**潜在类**。例如，潜在类可以是按其风险偏好分组的调查响应者的聚类。

该模型采取多项式混合模型的形式。模型中有两组参数：γ 参数和 ρ 参数。γ 参数表示聚类成员关系的总概率。ρ 参数表示在聚类成员关系给定的条件下观测到给定响应的概率。这些条件概率的模式构成了潜在类的特征。

为使分析结果具有意义，主题专家必须对平台生成的聚类进行解释。该主题专家检查潜在类的特征并根据这些特征构建每个类的定义。

注意：在任何响应列中具有缺失值的行都会从分析中排除。

有关潜在类模型的详细信息，请参见 Collins and Lanza (2010) 和 Goodman (1974)。

“潜在类分析”的示例

该示例使用“潜在类分析”平台分析 2005 年美国高中生调查的响应数据。该调查针对学生提出了各种有关健康风险行为的多项选择题。

在本例中，您根据学生对 12 个问题的响应来拟合潜在类模型，识别学生聚类。从多项选择调查问题中将响应分为两类 （是 / 否），可以获得您分析的列。

1. 选择**帮助** > **样本数据库**，然后打开 Health Risk Survey.jmp。
2. 在 “Health Risk Survey”数据表中，点击**启动潜在类分析平台**脚本旁边的绿色小三角。

   该脚本选择 12 个所关注的列，打开“潜在类分析”启动窗口，并且输入这 12 个关注的列作为 “Y”。

   注意：要自行启动 LCA 平台，请选择 “分析” > “聚类” > “潜在类分析”。
3. 在**多达**旁边的框中键入 5。

   该选项针对 3 到 5 个 （最多 5 个）聚类拟合潜在类模型。
4. 点击**确定**。

图 15.2 “聚类比较”报表

聚类比较

聚类数	-对数似然	BIC	AIC	最佳
3	38713	77776.2	77501.9	
4	38207.1	76884.4	76516.3	
5	37964.8	76519.6	76057.6	最小 BIC 最小 AIC

“潜在类分析”分级显示项包括一个“聚类比较”报表和 3 个单独的“潜在类模型”报表。“潜在类模型”报表显示 3 个、4 个和 5 个聚类的模型。在“聚类比较”报表中，具有 5 个聚类的模型具有最小的 BIC 和 AIC，这表明该模型是这 3 个模型中的最佳拟合模型。您分析的是该模型。

5. 在“‘5’个聚类的潜在类模型”报表中，检查“参数估计值”下方的条形图。请注意以下事项：
   - “聚类 1”对所有风险行为的大部分回答多为“No”。
   - “聚类 2”对 13 岁前的 4 种风险行为的回答多为“Yes”。
   - “聚类 3”对酒驾和过去 30 天内喝过至少 5 杯酒的很多回答多为“Yes”。
   - “聚类 4”对除 13 岁之前的风险行为之外的其他大部分风险行为的回答多为“Yes”。
   - “聚类 5”对大部分风险行为的回答为“Yes”最多。

   使用该信息为聚类提供有意义的名称。
6. 点击“‘5’个聚类的潜在类模型”旁边的红色小三角，选择**重命名聚类**：
   - 为“聚类 1”输入“低风险”。
   - 为“聚类 2”输入“早期风险承担者”。
   - 为“聚类 3”输入“嗜酒者”。
   - 为“聚类 4”输入“后期高风险”。
   - 为“聚类 5”输入“高风险”。
7. 点击**确定**。
8. 在出现的“JMP 警示”中点击**确定**。

注意：新聚类名称不会保存到脚本中。

图 15.3 部分“参数估计值”报表

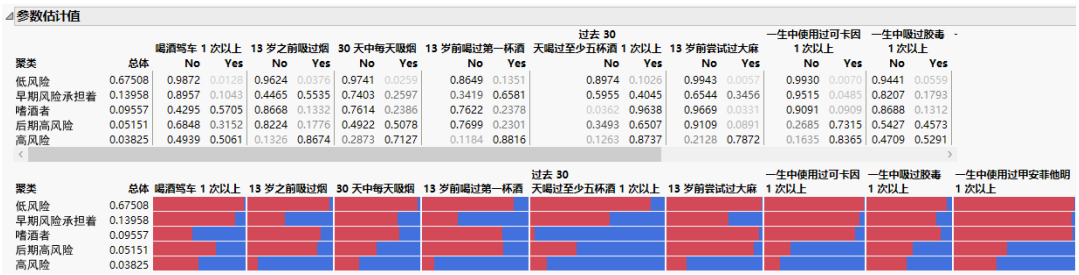


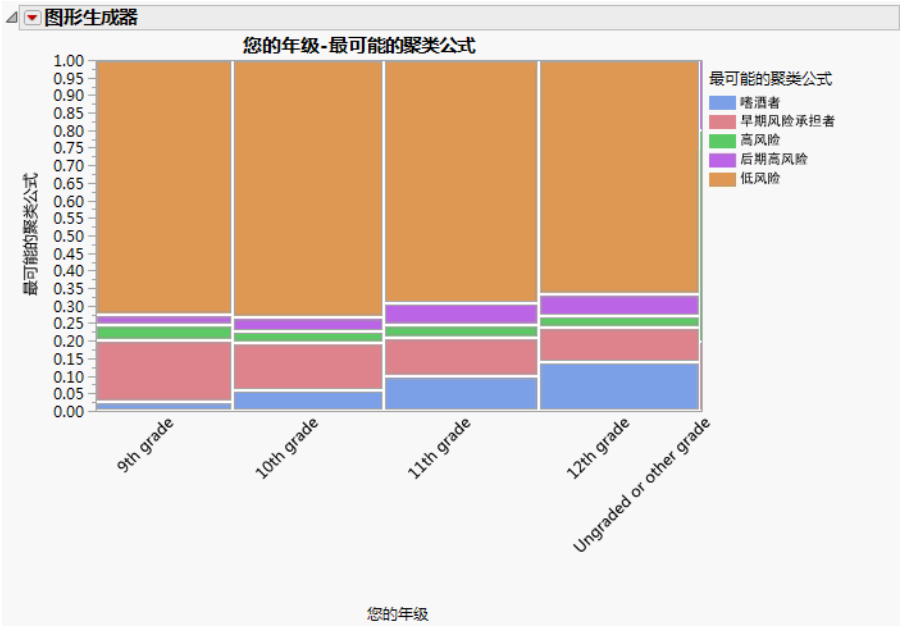
图 15.3 显示分析中前 8 个变量的参数估计值。新聚类名称出现在报表窗口中。

接下来，将聚类成员关系与人口统计学问题“您的年级”进行比较。

9. 点击“‘5’个聚类的潜在类模型”旁边的红色小三角，选择**保存混合和聚类公式**。
10. 选择**图形 > 图形生成器**。

11. 输入**您的年级**作为 X。
12. 输入**最可能的聚类公式**作为 Y。
13. 选择 “马赛克图” 元素。
14. 点击**完成**。

图 15.4 “年级 - 聚类成员关系” 的马赛克图

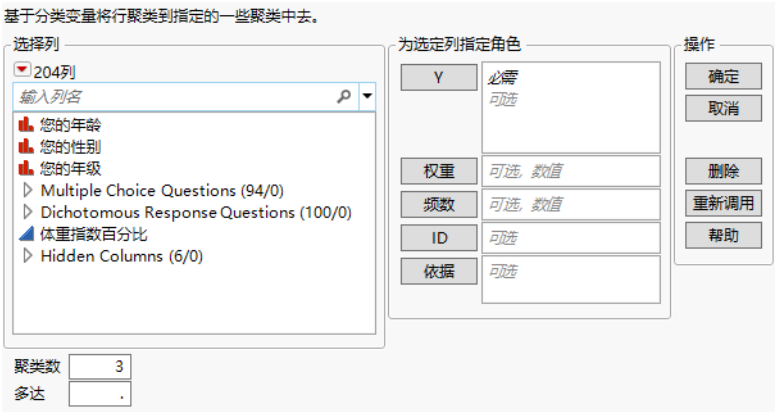


观测到大多数响应者落入 “低风险” 聚类。标有 “嗜酒者” 的类随着年级增长，响应者人数增加。

启动 “潜在类分析” 平台

通过选择**分析** > **聚类** > **潜在类分析**，启动 “潜在类分析” 平台。

图 15.5 “潜在类分析” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“潜在类分析” 平台启动窗口包括以下选项：

Y 您要分析的一个或多个列。您可以分析建模类型为名义型、有序型或多重响应的列。要分析名义型或有序型响应，需要两列或更多列。若包含多重响应并且建模类型为多重响应，则只需一列。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，列中的数值为分析中的每行分配一个频数。

ID 用于标识单独响应者的列。该标识在某些输出表中使用。

依据 一列，用于创建为变量的每个水平包含单独分析的报表。若分配了多个 “依据” 变量，则为 “依据” 变量水平的每个可能组合生成单独分析。

聚类数 要在分析中计算的聚类数。

多达 指定最大聚类数。若该数值超过为 “聚类数” 指定的值，则生成模型报表中的聚类数为介于 “聚类数” 和 “多达” 范围内的每个整数值。这些报表显示为 “潜在类分析” 报表分级显示项的一部分。

警告：LCA 模型可充分拟合的聚类数存在最大值。若您请求的聚类数超过最大值，报表窗口中会显示一条警告消息。LCA 平台拟合的聚类数上限是各列所支持的最大聚类数。有关确定最大聚类数的详细信息，请参见第 292 页的 “最大聚类数”。

“潜在类分析”报表

初始“潜在类分析”报表包含每个指定聚类数的“聚类比较”报表和“潜在类模型”报表。

“聚类比较”报表

“聚类比较”报表提供拟合统计量以比较各种模型。拟合统计量包括负对数似然（-对数似然）、BIC 和 AIC。每个统计量的值越小表示拟合效果越好。最佳拟合在名为**最佳**的列中指示。请参见《拟合线性模型》。

潜在类模型报表

每个“潜在类模型报表”被动态地命名为“<k> 个聚类的潜在类模型”，具体取决于 *k*，即拟合的聚类数。报表包含以下结果和分级显示项：

- 第 287 页的“模型汇总”
- 第 287 页的“参数估计值”
- 第 288 页的“转置参数估计值”
- 第 288 页的“效应大小”
- 第 288 页的“MDS 图”
- 第 289 页的“混合概率”

模型汇总

默认情况下，指定聚类数的模型汇总显示在每个“潜在类模型”报表的顶部。模型汇总包含 -对数似然、参数个数、BIC 和 AIC。这些汇总值可用于确定模型对数据拟合的好坏程度。“-对数似然”、“BIC”和“AIC”的值越小表示拟合效果越好。请参见《拟合线性模型》。“参数数目”值提供潜在类模型中的唯一参数个数。请参见第 291 页的““潜在类分析”平台的统计详细信息”。

参数估计值

“参数估计值”报表包含表格式和图形化的参数估计值汇总，并且在默认情况下显示出来。每个汇总包含的行对应于模型聚类。

“总体”列显示观测属于每个聚类的概率。（这些是 γ 参数。请参见第 291 页的““潜在类分析”平台的统计详细信息”。）

显示中的其余列依据在 “潜在类分析”启动窗口中指定的 Y 列使用竖分隔线分组。

- 每一组分类响应列都有一列对应于相应响应中的每个水平。在每组中，给定行列中的值是假定观测属于行所标识聚类的前提下，列所指示的响应的条件概率。（这些是 ρ 参数。）

- 每一组多重响应列都有一列对应于多重响应中的每个类别。在每组中，给定行列中的值是假定观测属于行所标识聚类的前提下，所指示的类别的较低级别下的响应的条件概率。（这些是 ρ 参数。）

图形化显示将条件概率值显示为**份额图**。对于每个聚类和每个 Y，用水平堆叠条形图绘制了给定聚类成员关系的条件概率。对于二值或名义型响应列，这些图中每个响应的百分比都加总为 1。对于多重响应列，百分比是每个类别的较低级别的百分比，加总不为 1。直条的堆叠按照值表中变量的顺序展示。您还可以悬停在直条上方，以查看变量的水平或类别。

提示：您可以在“参数估计值”报表的任一表中选择一行或多行，以选择分配至相应聚类的观测。

转置参数估计值

“转置参数估计值”报表包含“参数估计值”报表表的转置表。该报表中聚类显示为列。针对分析中每个 Y 列的每个响应类别显示了其在每个聚类出现的条件概率。

注意：“总体”列的估计值不包括在转置表中。

效应大小

“效应大小”表在各聚类之间比较 Y 列，且默认情况下显示出来。该表每行中的统计量从由 Y 列的水平或类别与聚类成员关系的期望计数构成的列联表分析中获得。期望计数是将每个聚类中的观测数乘以 Y 列每个水平或类别的条件概率得到。

对于每个响应，为由聚类与水平的期望计数构成的列联表计算 Pearson 卡方统计量 χ^2。用 n 表示观测数。“效应大小”列中的值定义如下：

$$\text{效应大小} = \sqrt{\frac{\chi^2}{n}}$$

“LR Logworth”列中的每个值显示 -log10(p_{LR})，其中 p_{LR} 是由期望计数构成的列联表的对数似然比检验 p 值。超过 2 的 Logworth 值对应于在 0.01 显著性水平下的显著性。

提示：您可以在“效应大小”表中选择一行或多行，以选择数据表中的相应列。

MDS 图

MDS 图对每个聚类都包含一个点，并且在默认情况下显示出来。它是聚类邻近关系的二维表示。越邻近的聚类越相似。该图基于 ρ 参数的相异度矩阵创建。有关 MDS 图的详细信息，请参见第 203 页的“多维尺度化”一章。

混合概率

“混合概率”表显示每行的聚类成员关系的概率。“最可能的聚类”列指示每行具有最高成员关系概率的聚类。

注意：一或多个 Y 列包含缺失值的行会从分析中排除，不会出现在“混合概率”表中。

“潜在类分析”平台选项

- “‘潜在类分析’选项”
- “‘潜在类模型’选项”

“潜在类分析”选项

“潜在类分析”红色小三角菜单包含以下选项：

新聚类数 使您可以使用其他聚类数运行另一个分析。新分析报表会追加至当前报表。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“潜在类模型”选项

“‘<k>’个聚类的潜在类模型”红色小三角菜单包含以下选项：

模型报表 支持您显示或隐藏可用模型报表。有关模型报表的详细信息，请参见第 287 页的“‘潜在类分析’报表”。

按聚类设定颜色 根据最可能的聚类对数据表中的每一行设定颜色。有关示例，请参见第 290 页的“‘潜在类分析’平台的更多示例”。

保存混合和聚类公式 将每个聚类的公式列以及最可能的聚类的公式列保存至数据表。

仅保存聚类公式 将确定最可能的聚类的公式列保存至数据表。

JMP PRO **发布概率公式** 创建概率公式并在“公式存储库”平台中将它们保存为公式列脚本。若未打开“公式存储库”报表，该选项将创建“公式存储库”报表。请参见《预测和专业建模》。

保存混合概率 将“混合概率”表中的值保存至数据表中的相应行。

仅保存聚类 将包含每行最可能的聚类的新列保存至数据表。该列不包含公式。

重命名聚类 可让您为报表中的聚类提供有意义的名称。

注意：除非为报表指定了随机种子，否则新聚类名称不会保存到脚本中。仅当通过脚本启动报表时才可设置随机种子。

删除拟合 从报表窗口中删除指定的聚类报表。

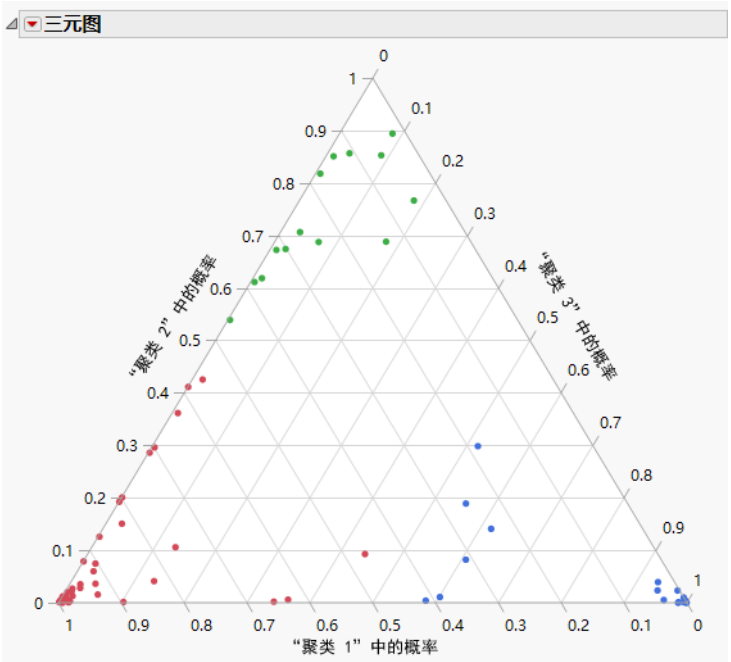
“潜在类分析”平台的更多示例

绘制聚类成员关系的概率图

该示例使用 Car Poll.jmp 样本数据表，其中包含车主和汽车制造商的调查数据。您打算将车主分为三个聚类，并且生成一个图以直观展示聚类成员关系的概率。当您有三个聚类时，三元图可提供良好的直观展现。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **聚类** > **潜在类分析**。
3. 选择除**年龄**外的所有列并点击 **Y**。
4. 点击**确定**。
5. 点击 “‘3’个聚类的潜在类模型”旁边的红色小三角，选择**按聚类设定颜色：**
6. 点击 “‘3’个聚类的潜在类模型”旁边的红色小三角，选择**保存混合概率：**
7. 在 “Car Poll”数据表窗口中，从列的列表中选择 **LCA 聚类概率**列组。
8. 选择**图形** > **三元图**。
9. 点击 **X, 绘图**。
10. 点击**确定**。

图 15.6 聚类成员关系概率的三元图



在每个观测的聚类概率的三元图中，大多数聚类成员关系概率落到顶点附近。这表明一个聚类具有较高的值，另外两个聚类具有较低的值。不过，图中部的某些点表明这些观测对任何聚类的聚类成员关系都不具有高概率。这些观测可能需要更仔细的检查，或者表明需要更多的聚类以便更好地表示数据。

注意：因为没有指定随机种子，您的结果可能有所不同。

“潜在类分析”平台的统计详细信息

- “潜在类模型拟合”
- “最大聚类数”

潜在类模型拟合

本节说明在“潜在类分析”平台中拟合的潜在类模型。有关潜在类模型的详细信息，请参见 Collins and Lanza (2010) 和 Agresti (2013)。

注意：在“文本分析器”平台中使用的 LCA 算法利用文档词条矩阵的稀疏性。因为这个原因，“文本分析器”平台中的 LCA 结果与“潜在类分析”平台中的结果不完全一致。

用 $j = 1,..., J$ 表示响应的观测列。它们是“潜在类分析”平台启动窗口中的 Y 列。列 j 的水平数表示为 R_j。

J 个变量的多维列联表包含 $W = R_1*...*R_J$ 个单元格。其中的每一个单元格依据其针对 J 个变量的响应模式来定义。因此，每个响应模式是形式为 $\mathbf{y} = y_1,..., y_j$ 的 J- 长度向量。将 $\mathbf{Y}$ 定义为所有响应模式视为行向量的 J x W 数组。 $\mathbf{Y}$ 中的每个元素 $\mathbf{y}_w$, 都具有概率 $\Pr(\mathbf{y}_w)$。这些概率之和为 1：

$$\sum_{w=1}^{W} Pr(y_w) = 1$$

考虑以下符号：

- C 是潜在类模型中的聚类数。
- γ_c 是聚类 c 中成员关系的概率。（γ_c 是潜在类流行度。）这些参数之和为 1。
- $r_{j,k}$ 是第 j 个响应的第 k 个水平。
- $\rho_{j,k|c}$ 是在属于类 c 的条件下，在列 j 中观测到响应 $r_{j,k}$ 的概率。（$\rho_{j,k|c}$ 是项目响应概率。）对于给定的聚类和响应变量 j，$\rho_{j,k|c}$ 之和为 1。
- $I(y_j = r_{j,k})$ 是指标函数，当 y_j 响应为第 j 个响应的第 k 个水平时该函数等于 1，其他情况下该函数等于 0。

观测到响应 $\mathbf{y}_w = y_1,..., y_j$ 的特定向量的概率是在 C 个潜在类下观测到该响应向量的条件概率之和：

$$\Pr(y) = \sum_{c=1}^{C} \gamma_c \prod_{j=1}^{J} \prod_{k=1}^{R_j} \rho_{j,k|c}^{I(y_j = r_{j,k})}$$

该方程是您从“潜在类分析”红色小三角菜单中选择“保存混合和聚类公式”选项时保存至数据表的 Prob Formula Cluster 公式的分母。Prob Formula Cluster 列中的公式给出 $\Pr(\text{聚类} = c \mid \mathbf{y}_w)$，其等于 $\Pr(\mathbf{y}_w, \text{聚类} = c) / \Pr(\mathbf{y}_w)$。

潜在类模型的 γ 和 ρ 参数使用迭代期望值最大化 (EM) 算法估计得到。潜在类模型中的唯一参数个数定义如下：

$$(C-1) + C\sum_{j=1}^{J}(R_j - 1)$$

最大聚类数

可以在 LCA 模型中拟合的最大聚类数取决于模型自由度。 LCA 模型中的自由度基于列创建的列联表的大小。列联表的大小是表中包含至少一个观测的单元数，表示为 K。若所有单元都包含至少一个观测，则 K 是响应列的水平数的乘积。自由度公式定义如下：

自由度 = K - {nCluster - 1 + nCluster(nTotalLevels - nCols)} - 1

其中

nCluster = 聚类数

nTotalLevels = 响应列的水平总和

nCols = 响应列数

为使 LCA 模型充分拟合，自由度必须为正。因此，要确保自由度 > 0，聚类的最大数目定义如下：

max(nCluster) < floor[*K*/(1 + nTotal Levels – nCols)]

第 16 章

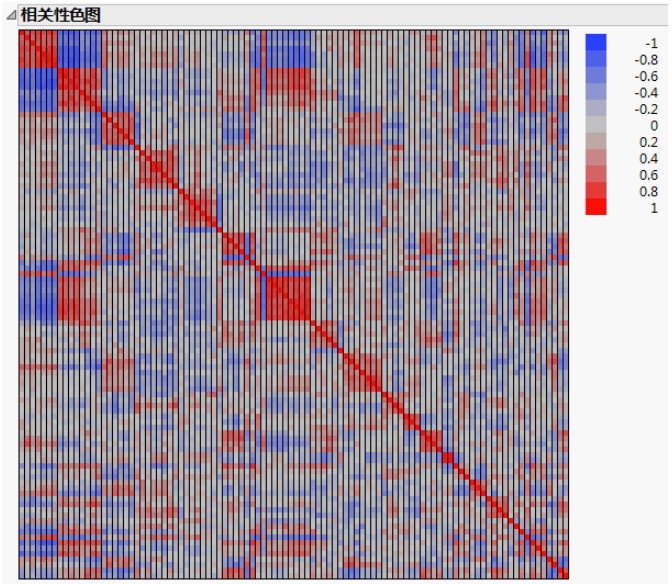
聚类变量

将类似变量分组到代表组中

变量聚类提供将类似变量分组到代表组中的方法。每个聚类可由单个成分或变量表示。成分是聚类中所有变量的线性组合。或者，聚类也可由标识为聚类中最典型成员的变量来表示。

您可以将“聚类变量”用作降维方法。您不用在建模中使用大量变量，聚类成分或聚类中最典型的变量便可解释数据中的大部分变异。此外，使用“聚类变量”的降维方法经常比使用主成分的降维方法更容易解释。

图 16.1 变量相关性色图的示例



目录

“聚类变量”平台概述 297
“聚类变量”平台的示例 297
启动“聚类变量”平台 299
“聚类变量”报表 299
相关性色图 300
聚类汇总 300
聚类成员 300
标准化成分 301
“聚类变量”平台选项 301
“聚类变量”平台的更多示例 301
相关性色图的示例 302
“聚类变量”平台有关降维的示例 303
“聚类变量”平台的统计详细信息 306
变量聚类算法 306

“聚类变量”平台概述

主成分分析构造的成分是分析中的所有变量的线性组合。相比之下，“聚类变量”选项构造的成分是同一类别的相似变量的线性组合。整个变量集划分为各个聚类。对于每个聚类，使用该聚类中的变量的第一主成分构造**聚类成分**。该聚类成分是一个线性组合，用于解释该聚类的变量中的尽可能多的变异。

您可以将“聚类变量”选项用作降维方法。大型变量集中的绝大部分变异往往可以通过聚类成分或聚类中最典型的变量来表示。随后，这些新变量可用在预测或其他建模方法中。基于聚类的新变量通常比基于所有变量的主成分更容易解释。

由一组相同变量构造的主成分相互之间正交。不过，聚类成分不正交，因为它们是由不同的变量集构造的。

当具有大量变量时，“聚类变量”平台使用基于奇异值分解的算法来缩短计算时间。有关更多背景信息，请参见“统计详细信息”附录中第 311 页的“‘宽线性’方法和奇异值分解”。

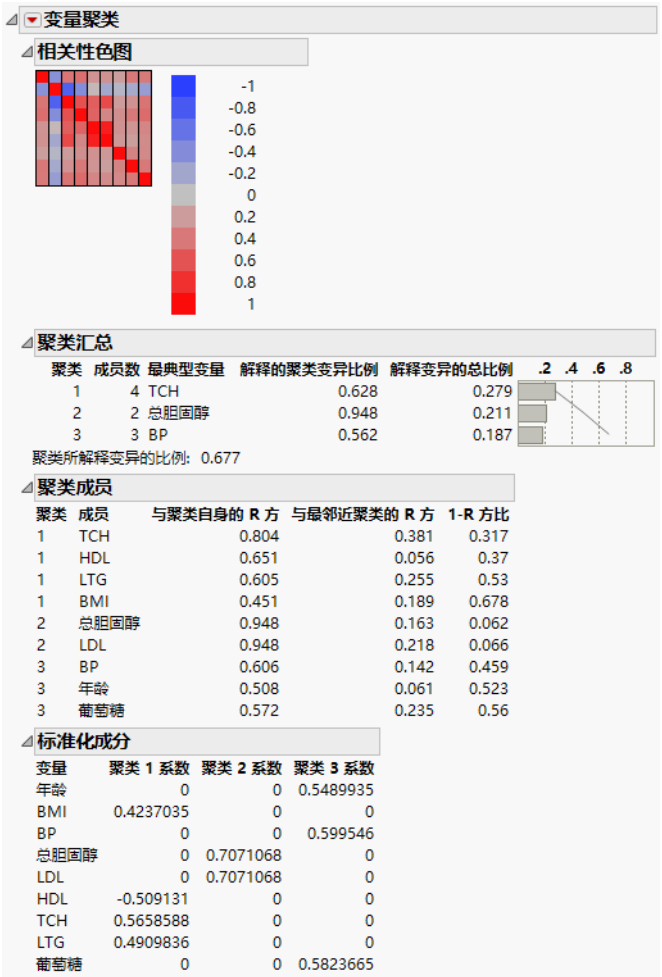
“聚类变量”平台的示例

Diabetes.jmp 样本数据表包含十个基线变量，用于对疾病发展进行建模。在本例中，您对连续型基线变量聚类。

1. 选择**帮助** > **样本数据库**，然后打开 Diabetes.jmp。
2. 选择**分析** > **聚类** > **聚类变量**。
3. 从**年龄**列一直选到**葡萄糖**列，但不包括**性别**（**年龄**、 BMI、 BP、**总胆固醇**、 LDL、 HDL、 TCH、 LTG 和**葡萄糖**），然后点击 **Y，列**。

   不能包括**性别**列，因为“聚类变量”要求提供数值型连续变量。
4. 点击**确定**。

图 16.2 糖尿病数据的聚类变量报表



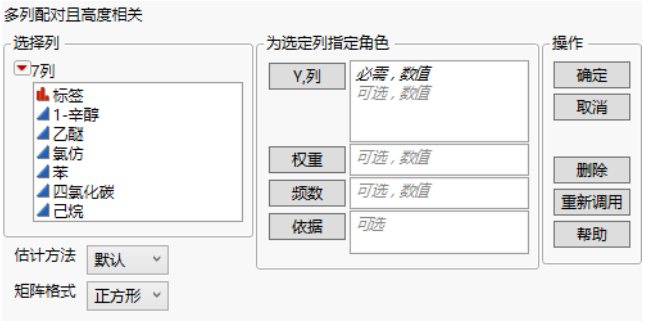
“聚类汇总”报表显示变量被分组到三个聚类中：

- “聚类 1”包括 TCH、HDL、LTG 和 BMI，如“聚类成员”报表中所示。“聚类汇总”报表显示 TCH 是对于“聚类 1”最典型的变量，而且对于“聚类 1”中的变量，62.8% 的变异是由第一个主成分解释的。
- “聚类 2”包括**总胆固醇**和 LDL。“聚类汇总”报表显示**总胆固醇**是对于“聚类 2”最典型的变量，而且对于“聚类 2”中的变量，94.8% 的变异是由第一个主成分解释的。
- 聚类 3 包括 BP、**年龄**和**葡萄糖**。“聚类汇总”报表显示最典型变量是 BP，对于“聚类 3”中的变量，56.2% 的变异是由第一个主成分解释的。

启动 “聚类变量” 平台

通过选择**分析** > **聚类** > **聚类变量**来启动 “聚类变量” 平台。

图 16.3 “聚类变量” 启动对话框



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，列 要聚类的变量。变量必须是数值型和连续型。

权重 一列，该列的数值为分析中的每一行都分配一个权重。

频数 一列，列中的数值为分析中的每行分配一个频数。

依据 一列，其水平定义不同的分析。对于指定列的每个水平，都分析相应行。结果显示在不同的报表中。若分配了多个 “依据” 变量，则为 “依据” 变量水平的每个可能组合生成单独分析。

“聚类变量” 报表

默认情况下，“聚类变量” 报表显示下列内容：

- 第 300 页的 “相关性色图”
- 第 300 页的 “聚类汇总”
- 第 300 页的 “聚类成员”
- 第 301 页的 “标准化成分”

提示：在所有 “聚类变量” 报表中，选择行以便选定数据表中的相应列。按 Ctrl 键并点击行可取消选择数据表中的对应列。

相关性色图

“相关性色图”报表显示变量之间的相关性的色图。变量按它们在“聚类成员”报表中列出的顺序排列。这种排列方式可确保同一聚类的成员在相关性图中是相邻的。请参见第 302 页的“相关性色图的示例”。

提示：悬停在色图的一个方块上时，可看到该方块中涉及的变量及其相关性。

同一聚类中的变量倾向于比不同聚类中的变量具有更高的绝对相关性（深红色或深蓝色）。因此，在相关性色图中，与给定的主成分中的变量相对应的方格形成的方块经常会沿对角线突出显示。

相关性使用逐行方法计算。该方法会从相关性计算中排除变量具有缺失数据的观测。有关逐行估计方法的详细信息，请参见“相关性和多元方法”一章中第 43 页的“估计方法”。

聚类汇总

“聚类汇总”报表包含以下信息：

聚类 聚类标识符。

成员数 聚类中的变量数。

最典型变量 与其聚类成分的平方相关性最大的那个聚类变量。

解释的聚类方差比例 聚类的变量中第一主成分解释的聚类方差比例。若聚类中仅有一个变量，则该值是 1。该统计量仅基于聚类中的变量而不是全部变量。

解释变异的总比例 聚类成分解释的方差总比例。这等价于仅使用每个聚类中的变量计算第一主成分。

表下方的注释给出所有聚类成分解释的变异的总比例。

聚类成员

“聚类成员”报表提供以下信息：

聚类 聚类标识符。

成员 聚类中包括的变量。

与聚类自身的 R 方 变量与其聚类成分的平方相关性。

与最邻近聚类的 R 方 变量与其最邻近聚类的聚类成分的平方相关性。最邻近聚类是变量与聚类成分的平方相关性次高的聚类。

1-R 方比 衡量某个变量所属的聚类与其最邻近聚类之间的相对接近性的测度。其定义如下：

(1 - 与聚类自身的 R 方)/(1 - 与最邻近聚类的 R 方)

标准化成分

“标准化成分”报表提供用于定义聚类成分的系数。这些系数是每个聚类中的第一主成分的特征向量。

“聚类变量”平台选项

“变量聚类”红色小三角菜单包含以下选项：

相关性色图 显示或隐藏“相关性色图”。请参见第 300 页的“相关性色图”。

聚类汇总 显示或隐藏“聚类汇总”报表。请参见第 300 页的“聚类汇总”。

聚类成员 显示或隐藏“聚类成员”报表。请参见第 300 页的“聚类成员”。

聚类成分 显示或隐藏“标准化成分”报表。请参见第 301 页的“标准化成分”。

保存聚类成分 将列作为组（称为**聚类成分**）保存到数据表中。每列名为**聚类 <i> 成分**并且包含公式，它用未中心化和未统一尺度的变量来表示聚类成分。

启动拟合模型 打开“模型规格”窗口，已经在该窗口中的“构造模型效应”列表中为每个聚类输入“最典型变量”。使用该选项可基于“最典型变量”构造模型。

提示：要使用成分拟合模型，首先选择**保存聚类成分**选项。然后用所需的“聚类成分”列替换“构造模型效应”列表中每个聚类的“最典型变量”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“聚类变量”平台的更多示例

- “相关性色图的示例”
- ““聚类变量”平台有关降维的示例”

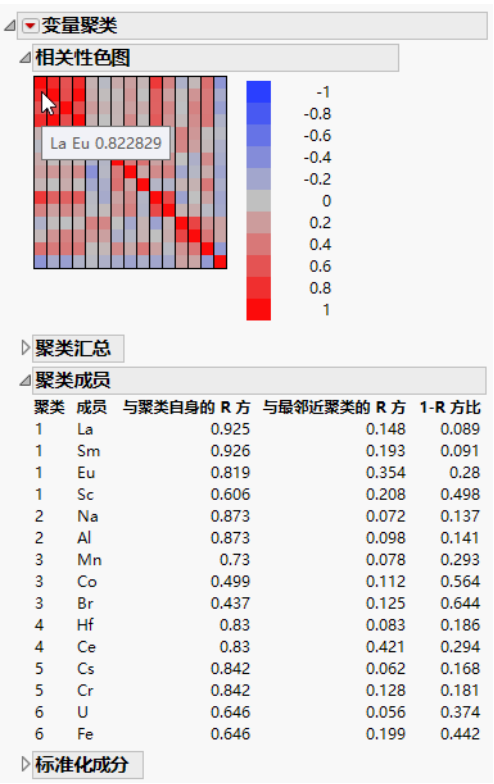
相关性色图的示例

在本例中，您构造并检查 “相关性色图”。

1. 选择**帮助** > **样本数据库**，然后打开 Cherts.jmp。
2. 选择**分析** > **聚类** > **聚类变量**。
3. 选择所有连续变量并点击 **Y, 列**。
4. 点击**确定**。
5. 关闭 “聚类汇总”和 “标准化成分”报表。
6. 悬停在色图的第二行第一列的单元格上方。

   随即出现工具提示，显示对应于该单元格的变量是 La 和 Eu，并且它们的相关性为 0.822829。

图 16.4 Cherts.jmp 的 “相关性色图”



聚类	成员	与聚类自身的 R 方	与最邻近聚类的 R 方	1-R 方比
1	La	0.925	0.148	0.089
1	Sm	0.926	0.193	0.091
1	Eu	0.819	0.354	0.28
1	Sc	0.606	0.208	0.498
2	Na	0.873	0.072	0.137
2	Al	0.873	0.098	0.141
3	Mn	0.73	0.078	0.293
3	Co	0.499	0.112	0.564
3	Br	0.437	0.125	0.644
4	Hf	0.83	0.083	0.186
4	Ce	0.83	0.421	0.294
5	Cs	0.842	0.062	0.168
5	Cr	0.842	0.128	0.181
6	U	0.646	0.056	0.374
6	Fe	0.646	0.199	0.442

“聚类成员”报表显示“聚类 1”中有四个变量。在“相关性色图”中，左上角中对应于这四个变量的由四乘四个方块构成的方格显示出明显的正相关性模式。色图还显示聚类 2、4 和 5 中变量的正相关性模式。色图右下角中对应于两个“聚类 6”变量的由二乘二个方块构成的方格显示它们是负相关的。请参见第 300 页的“相关性色图”。

“聚类变量”平台有关降维的示例

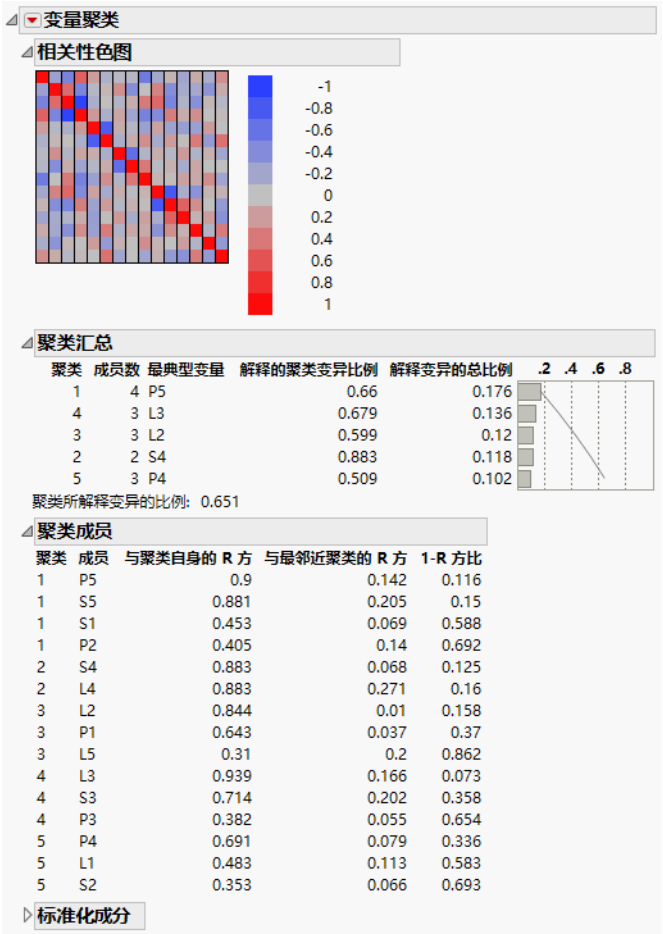
在本例中，您将“聚类变量”平台用作进行建模的降维工具。Penta.jmp 样本数据表包含 15 个变量用于预测响应变量**对数 RAI**。使用“聚类变量”可降低该数值。

聚类变量

1. 选择**帮助 > 样本数据库**，然后打开 Penta.jmp。
2. 选择**分析 > 聚类 > 聚类变量**。
3. 选择所有连续变量 （但不包括**对数 RAI**）并点击 **Y, 列**。
4. 点击**确定**。
5. 点击 “变量聚类”红色小三角并选择**保存聚类成分**。

   五个分组公式列将添加至数据表。

图 16.5 Penta.jmp 的 “聚类变量”报表



“聚类汇总”和“聚类成员”报表显示变量分为五个组，因此有五个聚类成分变量。

拟合模型

接下来，拟合并比较两个模型以预测**对数 RAI**：

- 使用所有连续变量作为预测变量的模型。
- 使用聚类成分作为预测变量的模型。

1. 点击“变量聚类”红色小三角并选择**启动拟合模型**。
2. 选择**对数 RAI** 并点击 **Y**。

   注意到五个聚类的 “最典型变量”已输入到 “构造模型效应”列表中。但是，您想输入所有预测变量。
3. 从 S1 一直选到 P5 ，选择所有连续变量并点击**添加**。

一定不要包括**对象名称**。

4. 选择**保持对话框打开**旁边的框。
5. 点击**运行**。

图 16.6 包含所有连续预测变量的模型的 “拟合最小二乘法”报表

响应“对数 RAI”

效应汇总

拟合汇总

R 方	0.929316
调整 R 方	0.853582
均方根误差	0.331225
响应均值	0.734333
观测数（或权重和）	30

方差分析

源	自由度	平方和	均方	F 比
模型	15	20.193596	1.34624	12.2709
误差	14	1.535941	0.10971	概率>F
校正总和	29	21.729537		<.0001*

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	-0.802632	0.924946	-0.87	0.4002
P5	2.0563803	1.651272	1.25	0.2335
S4	-0.062354	0.134935	-0.46	0.6511
L2	0.0860287	0.061206	1.41	0.1817
L3	0.3185383	0.080091	3.98	0.0014*
P4	0.4136598	0.394449	1.05	0.3121
S1	-0.09783	0.038948	-2.51	0.0249*
L1	0.032362	0.049732	0.65	0.5258
P1	-0.107951	0.085209	-1.27	0.2259
S2	0.086703	0.044276	1.96	0.0704
P2	0.0847235	0.086297	0.98	0.3429
S3	-0.037728	0.055602	-0.68	0.5085
P3	-0.027313	0.233655	-0.12	0.9086
L4	-0.029756	0.152012	-0.20	0.8476
S5	2.7123146	2.222039	1.22	0.2424
L5	-0.209128	0.270401	-0.77	0.4521

效应检验

效应详细信息

6. 在 “拟合模型”窗口中，选择 “构造模型效应”窗口中的所有变量，然后点击**删除**。
7. 选择**聚类成分**组并点击**添加**。
8. 点击**运行**。

图 16.7 聚类成分作为预测变量的模型的 “拟合最小二乘法”报表

响应 “对数 RAI”

效应汇总

拟合汇总

R 方	0.8214
调整 R 方	0.784191
均方根误差	0.402125
响应均值	0.734333
观测数（或权重和）	30

方差分析

源	自由度	平方和	均方	F 比
模型	5	17.848635	3.56973	22.0757
误差	24	3.880902	0.16170	概率>F
校正总和	29	21.729537		<.0001*

参数估计值

项	估计值	标准误差	t 比	概率>|t|
截距	0.6651552	0.074349	8.95	<.0001*
聚类 1 成分	-0.018483	0.056013	-0.33	0.7443
聚类 2 成分	0.0035891	0.069032	0.05	0.9590
聚类 3 成分	-0.204307	0.066714	-3.06	0.0053*
聚类 4 成分	-0.575455	0.065423	-8.80	<.0001*
聚类 5 成分	-0.046594	0.069786	-0.67	0.5107

效应检验

效应详细信息

仅包括五个聚类成分作为预测变量的模型解释响应中的绝大部分变异，其调整 R 方为 0.784。使用全部十五个预测变量的模型仅有略高的调整 R 方，它的值为 0.853（图 16.6）。

“聚类变量”平台的统计详细信息

变量聚类算法

聚类算法以迭代方式拆分变量的原始类别并将变量重新分配到新的类别，直到不可能再进一步拆分。初始聚类包含所有变量。该算法由 SAS 开发并且在 PROC VARCLUS 中实现 (SAS Institute Inc. 2020g)。

注意：该算法仅使用“Y,列”列表中的变量没有缺失值的观测。

以下是算法中的迭代步骤：

1. 对于所有聚类，请执行以下操作：
   a. 计算每个聚类中的变量的主成分。
   b. 若所有聚类的第二特征值均小于 1，则终止算法。
2. 使用以下步骤把第二特征值最大 （且大于 1）的聚类分成两个聚类：
   a. 使用斜交旋转来旋转当前聚类中的变量的主成分。
   b. 定义一个聚类，使其包含当前聚类中的变量满足：该变量与第一旋转主成分的平方相关性高于该变量与第二主成分的平方相关性。

c. 定义另一个聚类，使其包含原始聚类中的其余变量。这些变量与第二主成分具有更高的相关性。

d. 计算两个新聚类的主成分。

3. 通过检验来判定数据集中的任何变量是否应分配给不同的聚类。对于每个变量，请执行以下操作：

a. 计算变量与每个聚类的第一主成分的平方相关性。

b. 将变量放置在与其的平方相关性最高的聚类中。

注意：斜交旋转亦称原始四次方最大正交旋转。请参见 Harris and Kaiser (1964)。

附录 A

统计详细信息

多元方法

本附录将讨论“宽线性”方法和奇异值分解的使用。

目录

“宽线性”方法和奇异值分解 311
奇异值分解 311
SVD 和协方差矩阵 311
SVD 和逆协方差矩阵 312
计算 SVD 312

“宽线性”方法和奇异值分解

使用“聚类”、“主成分”和“判别”平台中的“宽线性”方法，您可以分析包含数千（甚至数百万）个变量的数据集。大多数多元方法要求计算协方差矩阵或对协方差矩阵求逆。当您的多元分析涉及很多变量时，协方差矩阵会非常大以至于对它进行计算或求逆很困难或计算时间很长。

假定您的数据包含 n 行和 p 列。协方差矩阵的秩最多是 n 和 p 中较小的那个。在宽数据集中，p 通常比 n 大很多。在这种情况下，协方差矩阵的逆矩阵最多有 n 个非零特征值。“宽线性”方法使用这一事实情况以及奇异值分解来提供高效的计算。请参见第 312 页的“计算 SVD”。

奇异值分解

奇异值分解 (SVD) 允许您先后通过一次旋转、一次缩放以及另一次旋转的方式来表示任意线性变换。 SVD 规定任何 $n \times p$ 的矩阵 **X** 可表示为：

$$\mathbf{X} = \mathbf{U}Diag(\Lambda)\mathbf{V}'$$

用 r 表示 **X** 的秩。用 $\mathbf{I}_r$ 表示 $r \times r$ 单位矩阵。

矩阵 **U**, $Diag(\Lambda)$ 和 **V** 具有以下属性：

U 是 $n \times r$ 半正交矩阵且 $\mathbf{U}'\mathbf{U} = \mathbf{I}_r$

V 是 $p \times r$ 半正交矩阵且 $\mathbf{V}'\mathbf{V} = \mathbf{I}_r$

$Diag(\Lambda)$ 是 $r \times r$ 对角矩阵，其正对角线元素由列向量 $\Lambda = (\lambda_1, \lambda_2, ..., \lambda_r)'$ 给出，其中 $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_r > 0$。

λ_i 是 **X** 的非零**奇异值**。

下面说明了 SVD 与正方形矩阵的谱分解的关系：

- λ_i 的平方是 **X′X** 的非零特征值。
- **V** 的 r 列是 **X′X** 的特征向量。

注意：在文献中有关于矩阵 **U**、**V** 以及包含奇异值的矩阵的维度的各种约定。但是，这些约定差异在 **X** 秩的范围内对分解没有实际影响。

有关奇异值分解的详细信息，请参见 Press et al.(1998, Section 2.6)。

SVD 和协方差矩阵

本节说明如何使用 SVD 获取协方差矩阵的特征向量和特征值。关注的矩阵具有至少一个大的维度时，计算 SVD 比计算它的协方差矩阵和特征值分解高效得多。

用 n 表示观测数，用 p 表示关注的多元分析中涉及的变量数。用 **X** 表示数据值的 $n \times p$ 矩阵。

通常对标准化数据应用 SVD。要将某个值标准化，需减去其均值，再除以其标准差。用 $\mathbf{X}_s$ 来表示标准化数据值的 $n \times p$ 矩阵。之后，标准化数据的协方差矩阵成为 **X** 的相关性矩阵，该矩阵定义如下：

$$Cov = \mathbf{X}_s'\mathbf{X}_s/(n-1)$$

可以对 $\mathbf{X}_s$ 应用 SVD 来获取 $\mathbf{X}_s'\mathbf{X}_s$ 的特征向量和特征值。当矩阵 **X** 很宽（很多列）或很高（很多行）时，这样做可以高效计算特征向量和特征值。此方法是宽 PCA 的基础。请参见“主成分”一章中第 54 页的““主成分”报表”。

SVD 和逆协方差矩阵

一些多元方法需要计算协方差矩阵的逆矩阵。本节说明如何使用 SVD 来计算协方差矩阵的逆矩阵。

用 $\mathbf{X}_s$ 表示标准化数据矩阵并定义 $\mathbf{S} = \mathbf{X}_s'\mathbf{X}_s$。奇异值分解允许您将 **S** 表示为：

$$\mathbf{S} = (\mathbf{U}Diag(\Lambda)\mathbf{V}')'(\mathbf{U}Diag(\Lambda)\mathbf{V}') = \mathbf{V}Diag(\Lambda)^2\mathbf{V}'$$

若 **S** 是满秩的，则 **V** 是 $p \times p$ 正交矩阵，您可以将 $\mathbf{S}^{-1}$ 表示为：

$$\mathbf{S}^{-1} = (\mathbf{V}Diag(\Lambda)^2\mathbf{V}')^{-1} = \mathbf{V}Diag(\Lambda)^{-2}\mathbf{V}'$$

若 **S** 不是满秩的，则 $\text{Diag}(\Lambda)^{-1}$ 可以使用广义逆矩阵 $\text{Diag}(\Lambda)^{+}$ 代替，其中 $Diag(\Lambda)$ 的对角线元素用其倒数替代。这便按以下方式定义 **S** 的广义逆矩阵：

$$\mathbf{S}^{-} = \mathbf{V}(Diag(\Lambda)^{+})^2\mathbf{V}'$$

该广义逆矩阵只使用 SVD 就可以计算得到。

有关将 SVD 应用于宽线性判别分析的详细信息，请参见“判别分析”一章中第 101 页的“宽线性判别方法”。

计算 SVD

在“多元方法”平台中，JMP 采用 Golub and Kahan (1965) 提出的方法计算矩阵的 SVD。Golub and Kahan 的方法是一个包含两个步骤的过程。第一步是将矩阵 **M** 简化为二对角矩阵 **J**。第二步是计算 **J** 的奇异值，它们与原始矩阵 **M** 的奇异值相同。通常将矩阵 **M** 的列标准化以便平衡变量对计算的影响。Golub and Kahan 方法的计算效率很高。

附录 B

参考书目

Agresti, A. (2013).*Categorical Data Analysis*.3rd ed. Hoboken, NJ: John Wiley & Sons.

Baglama, J., and Reichel, L. (2005)."Augmented implicitly restarted Lanczos bidiagonalization methods."*SIAM Journal on Scientific Computing* 27:19–42.

Ballard, D. H. (1981)."Generalizing the Hough Transform to Detect Arbitrary Shapes."*Pattern Recognition* 13:111–122.

Bartlett, M. S. (1937)."Properties of sufficiency and statistical tests."*Proceedings of the Royal Society of London, Series A* 160:268–282.

Bartlett, M. S. (1954)."A Note on the Multiplying Factors for Various Chi Square Approximations."*Journal of the Royal Statistical Society, Series B* 16:296–298.

Benzécri, J. P. (1979)."Sur le calcul des taux d'inertie dans l'analyse d'un questionnaire, addendum et erratum à [BIN.MULT.]."*Cahiers de l'Analyse des Données* 4:377–378.

Bentler, P. M. (1990)."Comparative Fit Indexes in Structural Models."*Psychological Bulletin* 107:238.

Bollen, K. A. (1989).*Structural Equations with Latent Variables*.New York: John Wiley & Sons.

Borg, I., and Groenen, P. J. F. (2005).*Modern Multidimensional Scaling: Theory and Applications*.2nd ed. New York: Springer.

Boulesteix, A.-L., and Strimmer, K. (2007)."Partial Least Squares: A Versatile Tool for the Analysis of High-Dimensional Genomic Data."*Briefings in Bioinformatics* 8:32–44.

Browne, M. (2001)."An Overview of Analytic Rotation in Exploratory Factor Analysis."*Multivariate Behavioral Research* 36:111 -150.

Browne, M. W., and Cudeck, R. (1993)."Alternative Ways of Assessing Model Fit."In *Testing Structural Equation Models*, edited by K. A. Bollen, and J. S. Long, 136–162.Newbury Park, CA: Sage Publications.

Chen, F. F. (2007)."Sensitivity of Goodness of Fit Indexes to Lack of Measurement Invariance."*Structural Equation Modeling: A Multidisciplinary Journal* 14:464–504.

Collins, L., and Lanza, S. (2010).*Latent Class and Latent Transition Analysis*.Hoboken NJ: John Wiley & Sons.

Cox, I., and Gaudard, M. (2013).*Discovering Partial Least Squares with JMP*.Cary, NC: SAS Institute Inc.

Cronbach, L. J. (1951)."Coefficient Alpha and the Internal Structure of Tests."*Psychometrika* 16:297–334.

Cudeck, R., and MacCallum, R. C., eds.(2007).Factor Analysis at 100, Historical Developments and Future Directions.Mahwah, NJ: Lawrence Erlbaum Associates.

de Ayala, R. J. (2009).*The Theory and Practice of Item Response Theory*.New York: Guilford Press.

De Jong, S. (1993)."SIMPLS: An Alternative Approach to Partial Least Squares Regression."*Chemometrics and Intelligent Laboratory Systems* 18:251–263.

Denham, M. C. (1997)."Prediction Intervals in Partial Least Squares."*Journal of Chemometrics* 11:39–52.

Eriksson, L., Johansson, E., Kettaneh-Wold, N., Trygg, J., Wikstrom, C., and Wold, S. (2006).*Multi- and Megavariate Data Analysis Basic Principles and Applications (Part I)*.Chapter 4.Umetrics.

Finkbeiner, C. (1979)."Estimation for the Multiple Factor Model when Data are Missing."*Psychometrika* 44:409–420.

Fisher, L., and Van Ness, J. W. (1971)."Admissible Clustering Procedures."*Biometrika* 58:91–104.

Florek, K., Lukaszewicz, J., Perkal, J., and Zubrzycki, S. (1951a)."Sur la liaison et la division des points d'un ensemble fini."*Colloquium Mathematicae* 2:282–285.

Florek, K., Lukaszewicz, J., Perkal, J., and Zubrzycki, S. (1951b)."Taksonomia Wroclawska."*Przeglad Antropologiczny* 17:193–211.

Frank, I. E., and Todeschini, T. (1994).*The Data Analysis Handbook*.New York: Elsevier.

Friedman, J. H. (1989)."Regularized Discriminant Analysis."*Journal of the American Statistical Association* 84:165–175.

Garthwaite, P. (1994)."An Interpretation of Partial Least Squares."*Journal of the American Statistical Association* 89:122–127.

Golub, G. H., and Kahan, W. (1965)."Calculating the singular values and pseudo-inverse of a matrix."*Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis* 2:205–224.

Goodman, L. A. (1974)."Exploratory Latent Structure Analysis Using Both Identifiable and Unidentifiable Models."*Biometrika* 61:215–231.

Greenacre, M. J. (1984).*Theory and Applications of Correspondence Analysis*.London: Academic Press.

Hand, D., Mannila, H., and Smyth, P. (2001).*Principles of Data Mining*.Cambridge, MA: MIT Press.

Harris, C. W., and Kaiser, H. F. (1964)."Oblique Factor Analytic Solutions by Orthogonal Transformation."*Psychometrika* 32:363–379.

Hartigan, J. A. (1981)."Consistency of Single Linkage for High–Density Clusters."*Journal of the American Statistical Association* 76:388–394.

Hastie, T., Tibshirani, R., and Friedman, J. H.(2009).*The Elements of Statistical Learning: Data Mining, Inference, and Prediction*.2nd ed. New York: Springer Verlag.

Hoskuldsson, A. (1988)."PLS Regression Methods." *Journal of Chemometrics* 2:211–228.

Hoeffding, W. (1948)."A Non-Parametric Test of Independence."*Annals of Mathematical Statistics* 19:546–557.

Hu, L.-T., and Bentler, P. M. (1999).“Cutoff Criteria for Fit Indices in Covariance Structure Analysis: Conventional Criteria versus New Alternatives.”*Structural Equation Modeling* 6:1–55.

Huber, P. J. (1964).“Robust Estimation of a Location Parameter.”*Annals of Mathematical Statistics* 35:73–101.

Huber, P. J. (1973).“Robust Regression: Asymptotics, Conjecture, and Monte Carlo.”*Annals of Statistics* 1:799–821.

Huber, P. J., and Ronchetti, E. M. (2009).*Robust Statistics*.2nd ed. Hoboken, NJ: John Wiley & Sons.

Jackson, J. E. (2003).*A User's Guide to Principal Components*.Hoboken, NJ: John Wiley & Sons.

Jardine, N., and Sibson, R. (1971).*Mathematical Taxonomy*.New York: John Wiley & Sons.

Jöreskog, K. G. (1977).“Factor Analysis by Least-Squares and Maximum Likelihood Methods.”In *Statistical Methods for Digital Computers*, edited by K. Enslein, A. Ralston, and H. Wilf, 125 - 165.New York: John Wiley & Sons.

Kline, R. B. (2016).*Principles and Practice of Structural Equation Methodology*.4th ed. New York: The Guilford Press.

Kohonen, T. (1989).*Self-Organization and Associative Memory*.3rd ed. Vol. 8 of Springer Series in Information.Berlin: Springer-Verlag.

Kohonen, T. (1990).“The Self-Organizing Map.”*Proceedings of the IEEE* 78:1464–1480.

Lindberg, W., Persson, J.-A., and Wold, S. (1983).“Partial Least-Squares Method for Spectrofluorimetric Analysis of Mixtures of Humic Acid and Ligninsulfonate.”*Analytical Chemistry* 55:643–648.

LeRoux, B., and Rouanet, H. (2010).Multiple Correspondence Analysis.Vol.07–163 of Sage University Paper Series on Quantitative Applications in the Social Sciences.Thousand Oaks, CA: Sage Publications.

Mardia, K., Kent, J., and Bibby, J. (1980).*Multivariate Analysis*.New York: Academic Press.

Mason, R. L., and Young, J. C. (2002).*Multivariate Statistical Process Control with Industrial Applications*.Philadelphia: SIAM.

Maydeu-Olivares, A., Shi, D., and Rosseel, Y. (2017).“Assessing Fit in Structural Equation Models: A Monte-Carlo Evaluation of RMSEA Versus SRMR Confidence Intervals and Tests of Close Fit.”*Structural Equation Model: A Multidisciplinary Journal* 25:389–402.

McLachlan, G. J., and Krishnan, T. (1997).*The EM Algorithm and Extensions*.New York: John Wiley & Sons.

McQuitty, L. L. (1957).“Elementary Linkage Analysis for Isolating Orthogonal and Oblique Types and Typal Relevancies.”*Educational and Psychological Measurement* 17:207–229.

Milligan, G. W. (1980).“An Examination of the Effect of Six Types of Error Perturbation on Fifteen Clustering Algorithms.”*Psychometrika* 45:325–342.

Nelson, P. R. C., Taylor, P. A., and MacGregor, J. F. (1996).“Missing Data Methods in PCA and PLS: Score calculations with incomplete observations.”*Chemometrics and Intelligent Laboratory Systems* 35:45–65.

Nunnally, J. C. (1978).*Psychometric theory*.2nd ed. New York: McGraw-Hill.

Penny, K. I. (1996).“Appropriate Critical Values When Testing for a Single Multivariate Outlier by Using the Mahalanobis Distance.”*Journal of the Royal Statistical Society, Series C* 45:73–81.

Press, W. H, Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1998).*Numerical Recipes in C: The Art of Scientific Computing*.2nd ed. Cambridge, England: Cambridge University Press.

Rasch, G. (1980).*Probabilistic Models for Some Intelligence and Attainment Tests*.Chicago: University of Chicago Press.

SAS Institute Inc.(1983).*SAS Technical Report A-108: Cubic Clustering Criterion*.Cary, NC: SAS Institute Inc. Accessed December 16, 2015. https://support.sas.com/kb/22/addl/fusion_22540_1_a108_5903.pdf.

SAS Institute Inc.(2020a).“The CALIS Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/calis.pdf.

SAS Institute Inc.(2020b).“The CANDISC Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/candisc.pdf.

SAS Institute Inc.(2020c).“The FACTOR Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/factor.pdf.

SAS Institute Inc.(2020d).“The FASTCLUS Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/fastclus.pdf.

SAS Institute Inc.(2020e).“The MIXED Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/mixed.pdf.

SAS Institute Inc.(2020f).“The PLS Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/pls.pdf.

SAS Institute Inc.(2020g).“The VARCLUS Procedure.”*SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/varclus.pdf.

Schafer, J., and Strimmer, K. (2005).“A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics.”*Statistical Applications in Genetics and Molecular Biology* 4 Article 32.

Sneath, P. H. A. (1957).“The Application of Computers to Taxonomy.”*Journal of General Microbiology* 17:201–226.

Sokal, R. R., and Michener, C. D. (1958).“A Statistical Method for Evaluating Systematic Relationships.”*University of Kansas Science Bulletin* 38:1409–1438.

Steiger, J. H. (1989).*EzPATH: A Supplementary Module for SYSTAT and SYGRAPH*.Evanston, IL: Systat.

Steiger, J. H. (1990).“Structural Model Evaluation and Modification: An Interval Estimation Approach.”*Multivariate Behavioral Research* 25:173–180.

Tobias, R. D. (1995).“An Introduction to Partial Least Squares Regression.”In *Proceedings of the Twentieth Annual SAS Users Group International Conference*, 1250–1257.Cary, NC: SAS Institute Inc. http://www.sascommunity.org/sugi/SUGI95/Sugi-95-210%20Tobias.pdf.

Tracy, N. D., Young, J. C., and Mason, R. R. (1992).“Multivariate Control Charts for Individual Observations.”*Journal of Quality Technology* 24:88–95.

Umetrics.(1995).*Multivariate Analysis (3-day course)*.Winchester, MA.

Waern, Y. (1972).“Structure in Similarity Matrices: A Graphic Approach.”*Scandinavian Journal of Psychology* 13:5–16.

West, S. G., Taylor, A. B., and Wu, W. (2012).“Model Fit and Model Selection in Structural Equation Modeling.”In *Handbook of Structural Equation Modeling*, edited by R. H. Hoyle, 209–231.New York: The Guilford Press.

White, K. P., Jr., Kundu, B., and Mastrangelo, C. M. (2008).“Classification of Defect Clusters on Semiconductor Wafers Via the Hough Transform.”*IEEE Transactions on Semiconductor Manufacturing* 21:272–278.

Wold, S. (1994).“PLS for Multivariate Linear Modeling.”In *QSAR: Chemometric Methods in Molecular Design.Methods and Principles in Medicinal Chemistry*, edited by H. van de Waterbeemd, pp. 195–218.Weinheim, Germany: Verlag-Chemie.

Wold, S., Sjostrom, M., and Eriksson, L. (2001).“PLS-Regression: A Basic Tool of Chemometrics.”*Chemometrics and Intelligent Laboratory Systems* 58:109–130.

附录 C

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

  - You must give any other recipients of the Work or Derivative Works a copy of this License; and
  - You must cause any modified files to carry prominent notices stating that You changed the files; and
  - You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
  - If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
  - You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

质量和过程方法

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020-2021。JMP® 16 质量和过程方法. Cary, NC: SAS Institute Inc.

JMP® 16 质量和过程方法

Copyright © 2020-2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录
质量和过程方法

1 了解 JMP 15
文档和其他资源
JMP 文档中的格式规范 17
JMP 帮助 18
JMP 文档库 18
学习 JMP 的其他资源 24
JMP 教程 24
样本数据表 24
学习统计和 JSL 术语 25
学习 JMP 的提示与技巧 25
JMP 工具提示 25
JMP 用户社区 25
免费在线统计思维课程 25
JMP 新用户欢迎套件 26
统计学知识门户 26
JMP 培训 26
用户编写的 JMP 手册 26
“JMP 起始页”窗口 26
JMP 技术支持 26

2 质量和过程方法介绍 27
过程和产品改进工具

3 控制图生成器 29
以交互方式创建控制图
“控制图生成器”概述 32
“控制图生成器”示例 32
控制图类型 34

Shewhart 计量控制图 34
Shewhart 计数控制图 35
稀有事件控制图 36
控制图类型 37
启动“控制图生成器” 39
“控制图生成器”交互式工作区 40
特定控制图的启动窗口 41
“控制图生成器”窗口 43
“控制图生成器”选项 44
“控制图生成器”红色小三角菜单选项 44
选项面板和右击图表选项 47
“控制图生成器”右击轴选项 55
使用控制限 55
控制限示例 56
“控制图生成器”中的排除和隐藏的样本 64
“控制图生成器”的更多示例 64
单个测量值和移动极差图示例 65
均值图和 R 图阶段示例 66
具有子组大小变化的均值图和 S 图的示例 69
运行图示例 71
P 图示例 72
NP 图示例 73
C 图示例 75
U 图示例 76
G 图示例 77
T 图示例 78
三因子控制图示例 80
“控制图生成器”的统计详细信息 82
均值图和 R 图的控制限 82
均值图和 S 图的控制限 83
单个测量值图和移动极差图的控制限 84
P 图和 NP 图的控制限 85
U 图和 C 图的控制限 86
Levey-Jennings 图 86

G 图控制限 86
T 图控制限 87
三因子控制图的 Sigma 计算 88

4 测量系统分析 91
使用 EMP 方法评估连续测量过程
“测量系统分析”概述 93
“测量系统分析”的示例 93
启动“测量系统分析”平台 96
数据格式 97
“测量系统分析”平台选项 97
平均图 99
极差图或标准差图 99
EMP 结果 100
有效分辨率 101
偏移检测刻画器 101
偏倚比较 105
复测误差比较 106
“测量系统分析”的更多示例 106
“测量系统分析”的统计详细信息 112
组间相关和概差的计算 112

5 变异性量具图 115
使用量具 R&R 评估连续测量过程
变异性图概述 117
变异性图的示例 118
启动“变异性/计数量具图”平台 119
数据格式 120
变异性量具图 120
“变异性量具”平台选项 121
方差非齐性检验 123
方差分量 123
关于“量具 R&R 方法” 125
“量具 R&R”选项 126
分辨力比 128

误分类概率 128
偏倚报表 129
线性研究 129
变异性图的更多示例 130
方差非齐性检验的示例 130
“偏倚报表”选项的示例 132
变异性图的统计详细信息 134
方差分量的统计详细信息 135
分辨力比的统计详细信息 136
误分类概率的统计详细信息 136

6 计数量具图 139
使用一致性测量评估分类测量过程
计数量具图概述 141
计数量具图示例 141
启动“变异性/计数量具图”平台 143
计数量具图和报表 144
一致性报表 145
有效性报表 145
“计数量具”平台选项 147
计数量具图的统计详细信息 148
一致性报表的统计详细信息 149

7 过程能力 153
测量随时间推移过程的变异性
“过程能力”平台概述 155
示例：包含正态变量的“过程能力”平台 156
包含非正态变量的“过程能力”平台的示例 158
启动“过程能力”平台 164
过程选择 165
过程子组 165
移动极差选项 166
历史信息 166
分布选项 167
其他指定 168

输入规格限 168
“规格限”窗口 168
限值数据表 169
规格限列属性 170
过程能力报表 171
目标图 172
能力箱线图 174
能力指标图 176
“过程能力”平台选项 178
单项详细报表 180
标准化箱线图 187
过程能力图 188
汇总报表 189
制作目标图汇总表 191
“过程能力”平台的更多示例 191
稳定过程的过程能力 191
不稳定过程的过程能力 195
模拟非正态过程 Ppk 的置信限 198
“过程能力”平台的统计详细信息 203
变异统计量 204
用于目标图和能力箱线图的符号 207
目标图 207
缺失目标的过程的能力箱线图 209
正态分布的能力指标 209
非正态分布的能力指标：百分位数和 Z 得分方法 214
分布的参数化 215

8 CUSUM 控制图 219
创建具有决策限的表格式 CUSUM 控制图
“CUSUM 控制图”平台概述 221
CUSUM 控制图示例 221
启动“CUSUM 控制图”平台 223
“CUSUM 控制图”平台报表 224
CUSUM 控制图的控制面板 224

CUSUM 图 225
“CUSUM 控制图”平台选项 225
“平均运行链长 (ARL)”报表 227
CUSUM 控制图的更多示例 228
CUSUM 控制图中的“数据单位”选项的示例 228
带子组的 CUSUM 图的示例 229
“CUSUM 控制图”平台的统计详细信息 230
CUSUM 控制图构造的统计详细信息 230
偏移检测的统计详细信息 231
平均运行链长的统计详细信息 231

9 EWMA 控制图 233
创建指数加权控制图

“EWMA 控制图”平台概述 235
“EWMA 控制图”平台示例 235
启动“EWMA 控制图”平台 237
“EWMA 控制图”平台报表 238
EWMA 控制图的控制面板 238
“EWMA 图”报表 239
“EWMA 控制图”平台选项 240
EWMA 控制图的“平均运行链长 (ARL)”报表 242
“EWMA 控制图”平台的统计详细信息 242

10 多元控制图 245
同时监控多个过程特征

多元控制图概述 247
多元控制图的示例 247
启动“多元控制图”平台 250
多元控制图 250
“多元控制图”平台选项 252
T 方已分割 252
变点检测 253
主成分 254
多元控制图的更多示例 254

监控使用子组数据的过程的示例 254
“T 方已分割”的示例 257
“变点检测”的示例 259
多元控制图的统计详细信息 260
单个观测的统计详细信息 260
合理子组中观测的统计详细信息 262
变点检测的统计详细信息 264

11 模型驱动的多元控制图 269
复杂过程的监控和诊断

模型驱动的多元控制图概述 271
模型驱动的多元控制图的示例 271
启动“模型驱动的多元控制图”平台 273
数据格式 274
“模型驱动的多元控制图”报表 275
“模型驱动的多元控制图”平台选项 275
图选项 276
得分图 277
“模型驱动的多元控制图”平台的更多示例 277
包含历史数据的 MDMVCC 的示例 278
包含 PLS 模型的 MDMVCC 的示例 279
“模型驱动的多元控制图”平台的统计详细信息 280
监控统计量 280
限值 282
贡献 284
得分图分组比较 286
得分图载荷 286

12 传统控制图 287
创建计量控制图和计数控制图

传统控制图示例 289
传统控制图类型 290
计量控制图 290
计数控制图 292
启动“传统控制图”平台 293

过程信息 294
限值指定 296
指定的统计量 297
传统控制图报表 297
V 形模板 CUSUM 图报表 299
解释双侧 V 形模板 CUSUM 图 299
解释单侧 CUSUM 图 301
“传统控制图”平台选项 301
传统控制图的窗口选项 301
传统控制图的图形选项 304
V 形模板 CUSUM 控制图的图表选项 306
保存和检索限值 306
排除、隐藏和删除的样本 309
“控制图”平台的更多示例 310
预先汇总图示例 310
V 形模板 CUSUM 图示例 312
单侧 CUSUM 图示例 314
UWMA 图示例 315
“控制图”平台的统计详细信息 315
中位数移动极差图的控制限 316
能力分析的统计详细信息 316
V 形模板 CUSUM 控制图的统计详细信息 321
加权移动平均图的统计详细信息 324

13 Pareto 图 327
着重关注重要少数的改善

“Pareto 图”平台概述 329
“Pareto 图”平台示例 329
启动“Pareto 图”平台 332
Pareto 图报表 333
“Pareto 图”平台选项 334
原因选项 335
“控制图”平台的更多示例 335
合并原因阈值的示例 336

使用组间常数大小的示例 337
使用组间非常数样本大小的示例 338
单因子比较 Pareto 图示例 340
双因子比较 Pareto 图示例 342
“Pareto 图”平台的统计详细信息 343
似然比卡方检验 343

14 因果图 345
确定根本原因
因果图概述 347
因果图示例 347
准备数据 347
启动“关系图”平台 348
因果图 349
右击菜单 349
“因果图”菜单选项 351
保存关系图 352
将关系图另存为数据表 352
将关系图另存为记录 352
将关系图另存为脚本 353

15 质量实用工具 355
管理规格限并创建 OC 曲线
“管理规格限”实用工具 357
“管理规格限”实用工具的示例 357
“管理规格限”选项 360
“操作特征曲线”实用工具 361
启动“OC 曲线”实用工具 361
控制图的 OC 曲线选项 361
用于验收抽样的 OC 曲线选项 362

A 参考书目 363
B 技术许可声明 367

第 1 章

了解 JMP

文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 17
JMP 帮助 18
JMP 文档库 18
学习 JMP 的其他资源 24
JMP 教程 24
样本数据表 24
学习统计和 JSL 术语 25
学习 JMP 的提示与技巧 25
JMP 工具提示 25
JMP 用户社区 25
免费在线统计思维课程 25
JMP 新用户欢迎套件 26
统计学知识门户 26
JMP 培训 26
用户编写的 JMP 手册 26
“JMP 起始页”窗口 26
JMP 技术支持 26

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助** > **JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助** > **JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 18 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助** > **JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

质量和过程方法介绍

过程和产品改进工具

《质量和过程方法》介绍 JMP 中提供的一些方法和工具，用来帮助您评价和改进质量和过程能力：

- 控制图对关键变量提供反馈，并显示过程何时受统计控制或何时不受统计控制。第 3 章，“控制图生成器”介绍使用称为“控制图生成器”的交互式控制图平台创建控制图的 JMP 方法。
- “测量系统分析”平台评估系统的精确性、一致性和偏倚。对某个过程开展研究之前，您需要确保能够精准地测量该过程。若变异源自测量本身，您将无法可靠地了解该过程。使用该分析可查明系统当前的性能。请参见第 4 章，“测量系统分析”。
- “变异性 / 计数量具图”平台创建变异性或计数量具图。变异性图分析连续测量值并展示系统当前的性能。计数图分析分类测量值并为您显示各响应之间的一致性测量。您还可以执行量具研究，查看数据中的变异测量。请参见第 5 章，“变异性量具图”和第 6 章，“计数量具图”。
- “过程能力”平台测量过程是否能够满足规格限要求。您可以将按过程中心化和变异性汇总的过程能力与规格限相比较。平台基于长期和短期变异计算能力指标。该分析可帮助标识相对于规格的变异，使合格值持续增加。请参见第 7 章，“过程能力”。
- CUSUM 图支持您基于累积和作出决策。这些图可检测过程中的较小偏移。请参见第 8 章，“CUSUM 控制图”。
- 指数加权移动平均 (EWMA) 图还可用于检测过程中的较小偏移。请参见第 9 章，“EWMA 控制图”。
- 若您需要同时监控多个过程特征，请参见第 10 章，“多元控制图”。
- “模型驱动的多元控制图”(MDMVCC) 平台支持您基于主成分或偏最小二乘模型生成控制图。请参见第 11 章，“模型驱动的多元控制图”。
- 第 12 章，“传统控制图”介绍了 JMP 中较早的控制图平台。我们鼓励您使用“控制图生成器”平台以及新的 CUSUM 和 EWMA 控制图平台，而不是使用较早的控制图平台。
- “Pareto 图”平台显示在与质量相关的过程或操作中问题出现的频率。Pareto 图可通过突出显示问题的频率和严重性，帮助您决定要首先解决的问题。请参见第 13 章，“Pareto 图”。
- “关系图”平台构造因果图，可通过对问题根源加以整理执行头脑风暴，或作为初步分析以标识变量，为进一步实验做准备。一旦完成，即可执行进一步分析以找出问题的根本原因。请参见第 14 章，“因果图”。
- “管理规格限”实用工具支持您同时快速添加或编辑若干列的许多规格限。请参见“质量实用工具”一章中第 357 页的“‘管理规格限’实用工具”。
- 使用“操作特征 (OC) 曲线”实用工具，您可以为控制图和计数验收抽样计划构造 OC 曲线。请参见“质量实用工具”一章中第 361 页的“‘操作特征曲线’实用工具”。

第 3 章

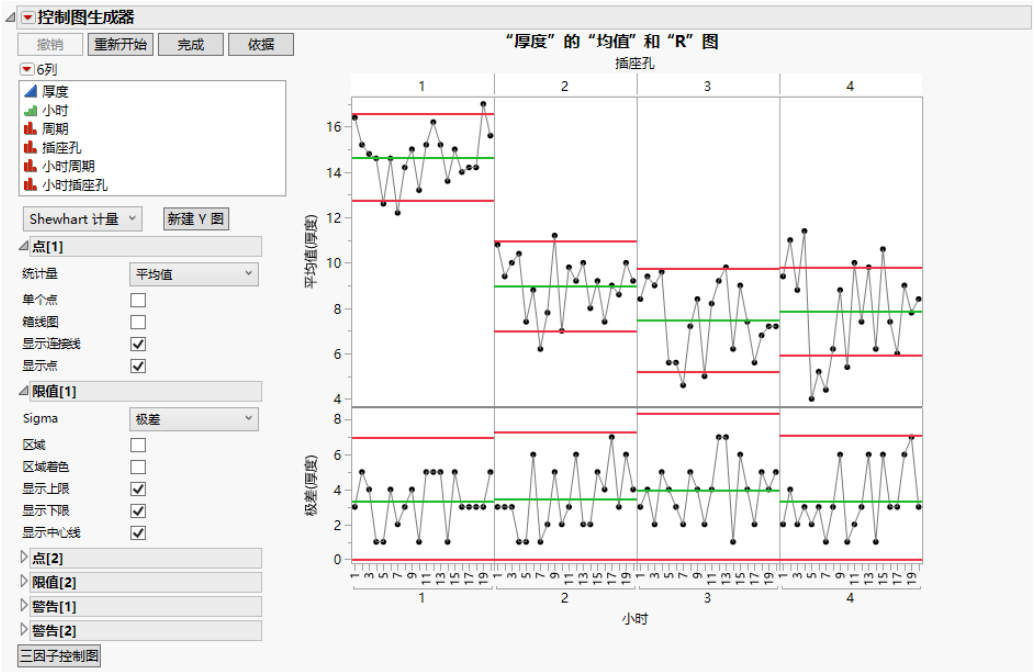
控制图生成器

以交互方式创建控制图

控制图是一种用于监控过程变异的图形化分析工具。过程中的自然变异可使用一组控制限来量化。控制限可帮助区分一般原因变异和特殊原因变异。通常会采取措施来确定和消除特殊原因变异。对过程中的一般原因变异进行量化也很重要，因为这决定了过程能力。

使用“控制图生成器”可创建过程数据的控制图。“控制图生成器”可以作为交互式工作区启动，也可以从特定的控制图菜单选项启动。在交互式工作区中，选择您要绘图的变量并将其拖入工作区。JMP 会基于数据自动选择适当的图表类型。您可以快速创建另一种图，也可以更改现有图的当前设置。

图 3.1 “控制图生成器”示例



目录

“控制图生成器”概述 32
“控制图生成器”示例 32
控制图类型 34
Shewhart 计量控制图 34
Shewhart 计数控制图 35
稀有事件控制图 36
控制图类型 37
启动“控制图生成器” 39
“控制图生成器”交互式工作区 40
特定控制图的启动窗口 41
“控制图生成器”窗口 43
“控制图生成器”选项 44
“控制图生成器”红色小三角菜单选项 44
选项面板和右击图表选项 47
“控制图生成器”右击轴选项 55
使用控制限 55
控制限示例 56
“控制图生成器”中的排除和隐藏的样本 64
“控制图生成器”的更多示例 64
单个测量值和移动极差图示例 65
均值图和 R 图阶段示例 66
具有子组大小变化的均值图和 S 图的示例 69
运行图示例 71
P 图示例 72
NP 图示例 73
C 图示例 75
U 图示例 76
G 图示例 77
T 图示例 78
三因子控制图示例 80
“控制图生成器”的统计详细信息 82
均值图和 R 图的控制限 82
均值图和 S 图的控制限 83
单个测量值图和移动极差图的控制限 84
P 图和 NP 图的控制限 85
U 图和 C 图的控制限 86
Levey-Jennings 图 86

G 图控制限 86
T 图控制限 87
三因子控制图的 Sigma 计算 88

“控制图生成器”概述

控制图是一种用于监控过程变异并识别过程中的特殊原因变异的图形化分析工具。建立控制限可过滤掉常规变异，这有助于确定某个过程是否稳定且可预测。若某个过程中的变异超过了预期，则可以调整该过程以创造更高质量的产出，同时还可能节省成本。

随着对过程不断地监控，所有过程都表现出测量的变异。过程测量中有两种变异：

- **常规**或**一般原因**变异。来自稳定过程的测量值仍显现随机变异。若过程测量值仅显现出一般原因变异，这些测量值会停留在期望的限值范围内。
- **异常**或**特殊原因**变异。特殊原因变异由控制图上观测到的模式表示。示例包括过程均值的偏移、控制限上方或下方的点，或呈现上升或下降趋势的测量值。过程测量值中的这些偏移可能是由工具或机器损坏、设备退化或是原材料变更等因素引起的。过程中的变化或缺陷往往通过过程测量值中的异常变异表现出来。

“控制图生成器”支持您创建若干类型的控制图，包括 Shewhart 和稀有事件控制图。Shewhart 控制图可大致划分为计量控制图和计数控制图两大类。稀有事件图专为较少出现的事件而设计。 JMP 提供了一种灵活的、用户定义的控制图生成方法。可以通过以下方式构造控制图：

- 使用交互式“控制图生成器”工作区。将某个数据列拖至工作区后，“控制图生成器”将基于数据类型和样本大小创建合适的图表。
- 使用控制图菜单选项可以通过启动窗口生成特定的控制图。

一旦通过上述任一方法创建了初始图表，就可以使用菜单和其他选项更改图表类型、更改图表上的统计量、重新设置图表格式或添加其他图表。

“控制图生成器”示例

Socket Thickness.jmp 样本数据表包含插座厚度的测量值。生产过程中的缺陷数呈上升趋势，所以您想要调查原因。本例说明如何在 “控制图生成器”中使用交互式方法或启动窗口方法执行该调查。若您了解要生成哪种类型的控制图，则第二种方法很方便。

“控制图生成器”交互式方法

使用交互式“控制图生成器”工作区可调查过程数据中的变异。

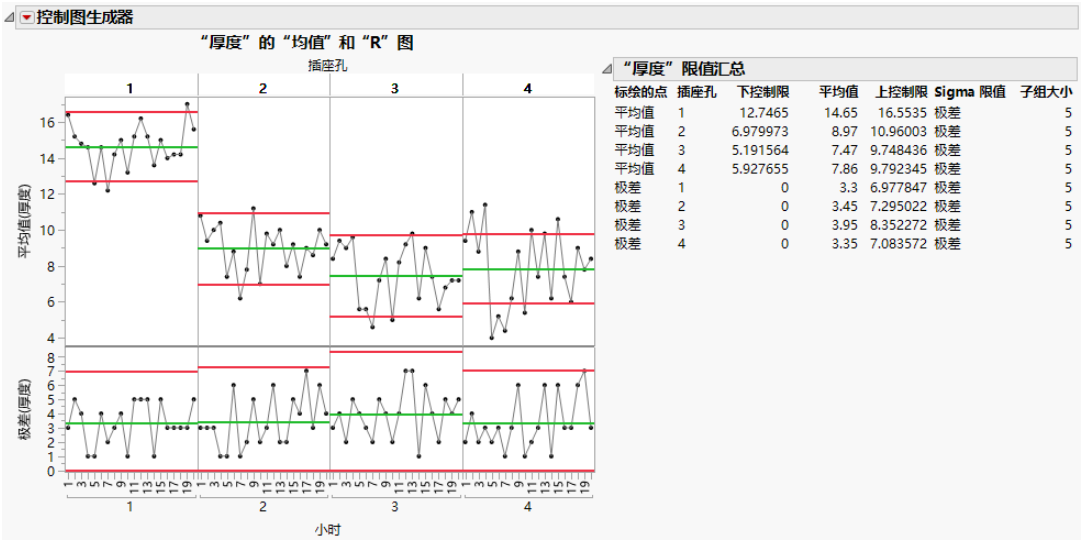
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Socket Thickness.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**厚度**拖至 Y 区域。
4. 将**小时**拖至**子组**区域 （位于底部）。

图 3.2 插座厚度控制图

通过查看平均图，您可以发现有若干点位于下控制限 7.788772 下方。您想要查看是否有其他变量导致了该问题。

5. 将**插座孔**拖入**阶段**区域。
6. 点击**完成**。

图 3.3 每个插座孔的控制图



从平均图中，您可以得到以下结论：

- 插座孔之间存在差异，表明每个插座孔需要单独的控制限。
- 插座孔 1 生产的插座的平均厚度高于其他插座孔生产的插座。这表明有必要进一步调查插座孔之间的差值。
- 所有插座孔都有位于控制限之外的点。因此，您应该为每个插座孔调查过程中为何缺乏控制。

每个插座孔的极差图显示子组内测量值处在受控状态并且在插座孔之间是相似的。

“控制图生成器”启动窗口方法

使用 “均值控制图”启动窗口获取与图 3.3 相同的图表。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Socket Thickness.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **均值控制图**。
3. 选择**厚度**并点击 Y。
4. 选择**小时**并点击**子组**。
5. 选择**插座孔**并点击**阶段**。
6. 点击**确定**。

   您应能看到图 3.3 中显示的控制图。

控制图类型

“控制图生成器”支持您创建若干类型的控制图，包括 Shewhart 计量图、Shewhart 计数图和稀有事件图。

- “Shewhart 计量控制图”
- “Shewhart 计数控制图”
- “稀有事件控制图”
- “控制图类型”

Shewhart 计量控制图

计量控制图根据在图上标绘的子组汇总统计量来分类。

- 均值图是一种显示子组均值（平均值）的位置图。
- R 图是一种显示子组范围（最大值 - 最小值）的散度图。
- S 图是一种显示子组标准差的散度图。
- 预先汇总图显示子组均值和标准差。

- 单个测量值图是一种显示单个测量值的位置图。
- 移动极差图是一种显示两个相继测量值的移动极差的散度图。

注意：若您删除散度图或禁用“文件”>“首选项”>“平台”>“控制图生成器”中的首选项“显示两个 Shewhart 图”，则只会看到位置图。所有关联的脚本都将包含 JSL 选项“显示两个 Shewhart 图”，且该选项设置为“关”(0)。

均值图、R 图和 S 图

对于在连续尺度上测量的质量特征，一个典型的分析会显示出过程均值及其变异性，并且会有一个均值图在其对应的 R 图或 S 图上方对齐。

单个测量值图

单个测量值图显示单个测量值。单个测量值图适用于每个抽样时间点仅有一个测量值的情况。若绘制的是单个测量值，则单个测量值图显示在其对应的移动极差图上方。移动极差图显示两个相继测量值的移动极差。

预先汇总图

若您的数据由同一过程单位的重复测量值构成，则可以将这些测量值合并为该单位的一个测量值。除非数据中的每个过程或测量单位都包含重复的测量值，否则不推荐使用预先汇总。

通过预先汇总，可基于所选的样本大小或样本标签将过程列汇总到样本均值和/或标准差中。然后基于在该窗口中所选的选项绘制汇总数据。

Levey-Jennings 图

Levey-Jennings 图显示过程均值，以及基于长期 sigma 的控制限。控制限位于距离中心线 3 *s* 远的位置。Levey-Jennings 图的标准差 *s* 计算方法与“分布”平台中的标准差计算方法相同。

Shewhart 计数控制图

计数图适用于计数数据。计数图基于二项模型和 Poisson 模型。由于是按子组来测量计数，所以在比较多个图时，确定各图之间的子组中项数是否相似这点很重要。计数图和计量图一样，按照在图上标绘的子组样本统计量来分类。

表 3.1 计数图确定

用于计算 Sigma 的分布	统计量类型：比例	统计量类型：计数
二项	P 图	NP 图
Poisson	U 图	C 图

“控制图生成器”根据所选变量为您做出一些决策。一旦创建基本图表，即可使用菜单和其他选项来更改该图表的类型、统计量和格式。

- P 图显示子组样本（大小可能会发生变化）中不合格（缺陷）项的比例。由于 P 图的每个子组都包含 N_i 项，而且任一项若非合格即为不合格，所以子组中的不合格项的最大数目为 N_i。
- NP 图显示子组样本中不合格（缺陷）项的数目。由于 NP 图的每个子组都包含 N_i 项，而且任一项若非合格即为不合格，所以子组 i 中的不合格项的最大数目为 N_i。
- C 图显示子组样本（通常但不一定总是包含一个检查单位）中不合格项（缺陷项）的数目。
- U 图显示在可能包含不同数目检查单位的子组样本中，每个单位的不合格项（缺陷项）的数目。

稀有事件控制图

稀有事件图是为数据来自稀有事件的过程提供相关信息的控制图。用传统控制图来跟踪记录不常发生的过程往往是无效的。稀有事件图专门设计用于响应稀有事件情形中的控制图限制。“控制图生成器”提供两种稀有事件图（G 图和 T 图）。G 图与 T 图之间的差值是用来测量稀有事件之间距离的数量。G 图测量事件之间的事件计数，而 T 图测量两次事件之间的时间间隔。

表 3.2 稀有事件图确定 基于 Sigma

用于计算 Sigma 的分布	图表类型
负二项	G 图
Weibull	T 图

G 图

G 图测量较少发生的错误之间的事件数或不合格事件，并创建随时间推移的过程图。图上每个点都表示相对稀少的事件各次出现之间的单位数。例如，在某个每天都生产的物品的生产环境中，可能会遇到意外生产线故障。您可以使用 G 图查看在各次生产线故障之间生产的单位数。

读取 G 图时，高于控制上限的点表示错误之间的事件数有所增加。若很少发生的错误之间的事件数或不合格事件增加了，这样很好。因此，当使用 G 图时，限值上方标记为失控的点通常被视为令人满意的效应。

T 图

T 图测量自上次事件以来经过的时间间隔。该图上的每个点都表示自上次出现的稀有事件之后经过的若干时间间隔。T 图可用于数值型非负数据、日期/时间数据和间隔时间数据。由于这些数据的传统图可能在 0 处包含许多点，在 1 处偶尔有个点，所以使用 T 图可避免将许多点标记为失控。“控制图生成器”中的 T 图的数据点限定为整数值。

读取 T 图时，上控制限上方的点指示事件之间的时间延长了。这意味着不良事件的发生率下降了。因此，当使用 T 图时，限值上方标记为失控的点通常被视为可取效应。

控制图类型

最常见的控制图在“控制图生成器”以及“分析”>“质量和过程”>“控制图”菜单中某些平台提供。首选“控制图生成器”，使用它可轻松快捷地生成图表。JMP 会基于数据自动选择适当的图表类型。表 3.3 - 表 3.7 汇总了各种控制图类型。

表 3.3 不带分组 (X) 变量或非汇总数据的计量图

图表类型	“控制图生成器”选项	
	点 > 统计量	限值 > Sigma
单值	单值	移动极差
基于单值的移动极差	移动极差	移动极差
单值（基于移动极差中位数计算限值）	单值	移动极差中位数
基于单值的移动极差中位数	移动极差	移动极差中位数
Levey Jennings	单值	Levey Jennings

表 3.4 带分组 (X) 变量或汇总数据的计量图

图表类型	“控制图生成器”选项	
	点 > 统计量	限值 > Sigma
均值（基于极差计算限值）	平均值	极差
均值（基于标准差计算限值）	平均值	标准差
R	极差	极差
S	标准差	标准差
Levey Jennings	平均值	Levey Jennings 或总标准差

表 3.5 预先汇总图

图表类型	“控制图生成器”选项	
	点 > 统计量	限值 > Sigma
基于组均值的单值	平均值	移动极差
基于组均值的单值（基于移动极差中位数计算限值）	平均值	移动极差中位数
基于组标准差的单值	标准差	移动极差
基于组标准差的单值（基于移动极差中位数计算限值）	标准差	移动极差中位数
基于组均值的移动极差	基于均值的移动极差	移动极差
基于组均值的移动极差中位数	基于均值的移动极差	移动极差中位数
基于组标准差的移动极差	基于标准差的移动极差	移动极差
基于组标准差的移动极差中位数	基于标准差的移动极差	移动极差中位数

表 3.6 计数图

图表类型	“控制图生成器”选项	
	点 > 统计量	限值 > Sigma
P 图	比例	二项
NP 图	计数	二项
C 图	计数	Poisson
U 图	比例	Poisson

表 3.7 稀有事件图

图表类型	“控制图生成器”选项	
	点 > 统计量	限值 > Sigma
G 图	计数	负二项
T 图	计数	Weibull

启动“控制图生成器”

您可以通过以下两种方式启动 “控制图生成器”：

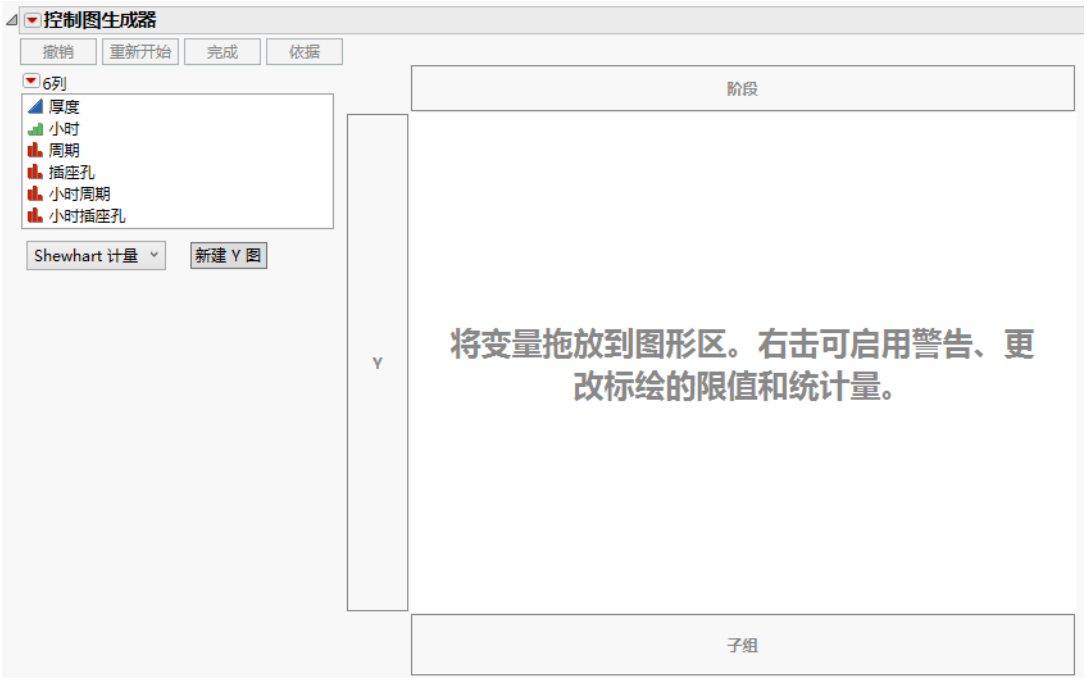
- 若不确定哪种类型的控制图适合您的数据，请选择**分析 > 质量和过程 > 控制图生成器**。该方法支持您将某个数据列拖至工作区，“控制图生成器”将基于数据类型和样本大小创建合适的图表。请参见第 40 页的““控制图生成器”交互式工作区”。
- 若您了解哪种类型的控制图适合您的数据，请从**分析 > 质量和过程 > 控制图生成器**子菜单中选择合适的图。这将显示控制图启动窗口。请参见第 41 页的“特定控制图的启动窗口”。提供以下控制图的控制图生成器启动窗口：
  - I/MR 控制图
  - 均值控制图
  - 运行图
  - P 控制图
  - NP 控制图
  - C 控制图
  - U 控制图
  - Levey Jennings 控制图
  - 基于均值的 I/MR 控制图
  - 三因子控制图

注意：CUSUM 控制图、EWMA 和多元控制图在各自的平台中启动，而不是在“控制图生成器”中启动，并单独记录。请参见第 219 页的“CUSUM 控制图”一章、第 245 页的“多元控制图”一章和第 233 页的“EWMA 控制图”一章。

一旦在启动窗口中点击“确定”，即显示“控制图生成器”窗口，默认情况下隐藏“控制面板”。其他所有选项和功能均相同。

“控制图生成器”交互式工作区

图 3.4 交互式 “控制图生成器”窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

要开始创建控制图，请将变量从**选择列**框拖入工作区。若将变量放置在中心位置， JMP 会根据是连续变量还是分类变量来猜测其放置位置。“控制图生成器”工作区包含以下区域：

Y 指定过程变量。

子组 指定子组变量。要将子组水平定义为多列的组合，请将多个变量添加至**子组**区域。指定某个子组变量后，控制图中的每个点都与该子组中所有点的某个汇总统计量相对应。

阶段 指定阶段变量。指定某个阶段变量后，将为每个阶段计算单独的控制限。另见第 68 页的“依据另一个变量过滤控制图”。

“控制图生成器”的初始窗口包含以下按钮：

撤销 撤销上次对该窗口所做的更改。

重新开始 将窗口恢复为默认状态，这样会删除所有数据并清空所有区域。

完成 隐藏按钮和选择列框，并删除所有拖放区外框。在这种适于演示的格式中，您可以将图形复制到其他程序。要将窗口还原为交互模式，请点击“控制图生成器”红色小三角并选择**显示控制面板**。

依据 标识变量，并为列中显示的每个值生成单独的分析。

Shewhart 计量 /Shewhart 计数 / 稀有事件 支持您选择 “Shewhart 计量” 、“Shewhart 计数”或“稀有事件”控制图类型。若选择“计数”图表类型，图上会显示 n 次试验框和区域。

n 次试验 （可用于计数图。）为计数控制图分配批大小。

新建 Y 图 为在选择列框中选定的每一列生成当前图的副本。新图使用 Y 角色中的选定列。

将变量拖至图中后，其他按钮和选项会显示在左侧，可支持您显示、隐藏或切换图上的项 （图 3.7）。其中许多功能 （点、限值、警告等）都与右击该图所提供的功能相同。请参见第 47 页的“选项面板和右击图表选项”。有关警告和规则的信息，请参见第 50 页的“检验”和第 53 页的“Westgard 规则”。

三因子控制图 支持您为计量图类型生成三因子控制图。子组大小必须大于 1。标绘的统计量基于子组平均值、子组内变异或子组间变异。默认的一组三个图包括一个使用移动极差限值的平均值预先汇总图、一个移动极差图和一个极差图。

事件选择器 允许图表实时响应选择更改。提供几组公认的且预评分的标准响应 （例如，及格 / 不及格、是 / 否、李克特量表、合格 / 不合格，以及缺陷 / 无缺陷）。若分析的是某项调查的结果，并且希望只专注于某方面的结果以解决一个或多个问题，则可以在屏幕上做出选择。做出选择时，图表会立即重新评分并重新标绘。在 “事件选择器”中选定的水平计为事件，其他所有水平都计为非事件。

事件选择器可用于响应列的建模类型为名义型或有序型的计数图。若想要“事件选择器”针对整数数值名义型或有序型响应列操作，您必须从“控制图生成器”红色小三角菜单中选择“使用事件选择器”选项。不会为建模类型为连续型的响应列显示 “事件选择器”。

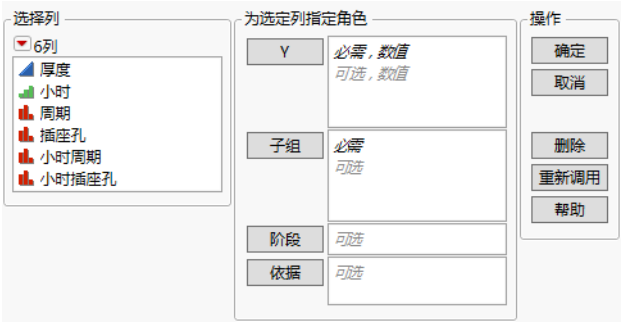
特定控制图的启动窗口

您在启动窗口中看到的选项有所不同，具体取决于您是启动计量控制图还是计数控制图。

计量控制图的启动窗口

本节包含有关 I/MR、均值、运行、 Levey Jennings、基于均值的 I/MR 和三因子等控制图的启动窗口的信息。

图 3.5 计量控制图的启动窗口



有关 “选择列”红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。

Y 分配过程变量。

子组 （仅可用于均值、 Levey Jennings、基于均值的 I/MR 和三因子等控制图。）分配子组变量。分配某个子组变量后，控制图中的每个点都与该子组中所有点的某个汇总统计量相对应。

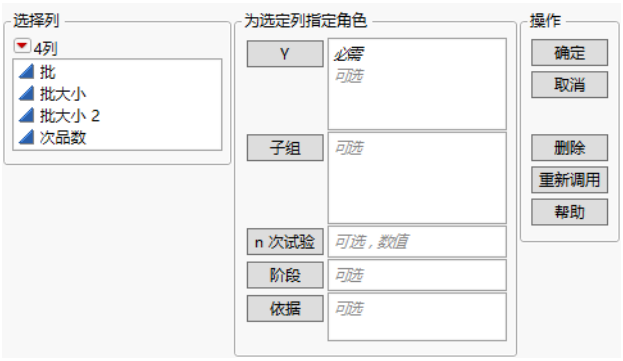
阶段 （不可用于运行图。）分配阶段变量。指定某个阶段变量后，将为每个阶段计算单独的控制限。

依据 标识一个变量，用于为列中出现的每个值生成单独分析。

计数控制图的启动窗口

本节包含有关 NP、 P、 C 和 U 控制图的启动窗口的信息。

图 3.6 计数控制图的启动窗口



有关 “选择列”红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。

Y 分配过程变量。

子组 分配子组变量。分配某个子组变量后，控制图中的每个点都与该子组中所有点的某个汇总统计量相对应。

n 次试验 分配子组样本大小。

阶段 分配阶段变量。指定某个阶段变量后，将为每个阶段计算单独的控制限。

依据 标识一个变量，用于为列中出现的每个值生成单独分析。

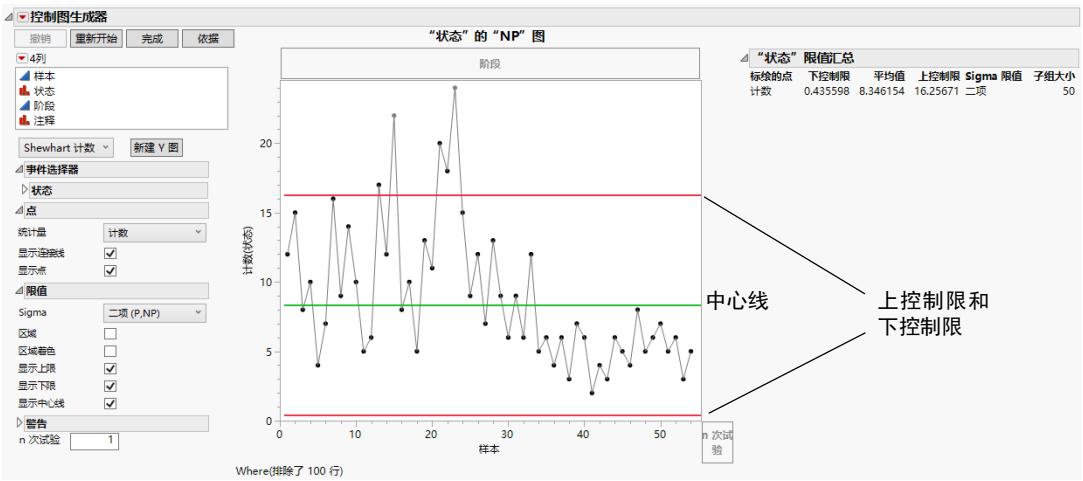
“控制图生成器”窗口

使用 “控制图生成器”可构造过程数据的控制图。该分析会生成一个图表，可用于评估某个过程是否处于统计控制状态中。报表随您选择的图表类型而变化。控制图随着在数据表中添加或更改数据而动态更新。图 3.7 显示了 Bottle Tops.jmp 样本数据表的 “控制图生成器”窗口。

要创建图表，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Bottle Tops.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**状态**拖至 Y 区域。
4. 将**样本**拖至**子组**区域 （位于底部）。

图 3.7 “控制图生成器”窗口



您可以将其他变量拖入不同区域以扩充分析，并使用 ““控制图生成器”选项”进一步检查数据。某些右击图表选项（例如，显示或隐藏点、限值、警告和区域；选择统计量和 sigma 选项）也会显示在图表左侧以便于访问。

控制图具有以下特征：

- 图上标绘的每个点都代表单个过程测量值或汇总统计量。应该**合理**选择子组；也就是说，在选择子组时，应该让在子组**之间**观测到真正的过程信号的概率最大化。通常，这需要了解确定最有效分组策略的过程。请参见 Wheeler (2004)；Woodall and Adams (1998)。
- 控制图的垂直轴尺度按照与汇总统计量相同的单位来调整。
- 控制图的水平轴标识子组样本并按照时间来排序。随时间推移观测过程对于评估过程是否在变化非常重要。

绿线是中心线，即表示数据的平均值。当过程处于统计控制状态时，中心线指示汇总统计量的平均值 （期望值）。测量值应均匀分布在中心线两侧。如若不然，则很可能表明过程平均值发生了变化。

- 两条红线分别为上控制限和下控制限，相应地标记为“上控制限”和“下控制限”。当过程处于统计控制状态时，这些限值可给出汇总统计量期望的变异范围。若过程仅显示出常规变异，则所有点都会随机散布在该范围内。

  注意：要隐藏散度图、计数图和稀有事件图的下控制限，请取消选择选项面板中的“显示下限”选项。要将默认设置更改为始终隐藏控制下限，请取消选择“文件”>“首选项”>“平台”>“控制图生成器”中的 “显示下控制限”首选项。

- 控制限之外的点指示存在特殊原因的变异。

使用“控制图生成器”窗口中的选项可以创建控制图，使这些控制图随着接收和记录样本或是随着向数据表添加样本而动态更新。当控制图指示异常变异时，应采取措施使过程恢复统计控制状态。若异常变异指示过程有所改善，则应对变异原因进行研究和实践。

双击轴时会显示相应的“轴规格”窗口，供您指定格式、坐标轴值、刻度数、网格线、参考线和其他要显示的选项。

“控制图生成器”选项

“控制图生成器”选项显示在红色小三角菜单中，也可通过右击某个图或轴来显示。某些右击图表选项也会显示在图表左下方以便于访问。您还可以在“文件”>“首选项”>“平台”>“控制图生成器”中设置 “控制图生成器”中许多选项的首选项。

- “‘控制图生成器’红色小三角菜单选项”
- “选项面板和右击图表选项”
- “‘控制图生成器’右击轴选项”

“控制图生成器”红色小三角菜单选项

“控制图生成器”红色小三角菜单包含以下选项：

显示控制面板 显示或隐藏以下元素：

- 按钮
- 选择列框
- 拖放区边框
- 复选框和下拉菜单

显示限值汇总 显示或隐藏 “限值汇总”报表。该报表显示图的控制限 （下控制限和上控制限）、中心线 （平均值）、标绘的点和限值，以及样本大小。稀有事件图不显示样本大小。

显示能力 （仅可用于包含规格限的 Shewhart 计量图。）显示或隐藏 “过程能力分析” 报表。由于该报表是 “限值汇总” 报表的一部分，因此仅当选定 “显示限值汇总” 选项时，才会显示 “过程能力” 报表。详细信息，请参见第 171 页的 “过程能力报表”。您可以在 “文件” > “首选项” > “平台” > “过程能力” 中，为 “控制图生成器” 中的 “过程能力” 报表内的许多选项设置首选项。

注意：若响应变量无变异，“显示能力” 不可用。

显示报警报表 显示或隐藏包含失控样本信息的报表。该报表反映每个图中当前所启用检验的失败情况，并且在启用和禁用不同的检验以及数据和行状态更改时自动更新。第二个表列出了每个图表当前启用的检验。第一个表包含以下列：

位置 指示图表的数值位置，从报表顶部开始。

失控总样本数 统计未通过至少一个选定检验的样本数。

报警率 失控样本总数除以非缺失样本总数。亦称 “失控比例”。

注意：仅当同时选定 “检验排除的子组” 和 “显示排除区域” 选项时，对报警率计算有贡献的计数将包括排除的样本。

显示限值标签 显示或隐藏每个图中限值的标签。限值显示在图表的右框内。

显示 Sigma 报表 （仅可用于 Shewhart 计量图。）显示或隐藏过程 Sigma 报表，这是一个 sigma 值表。过程 Sigma 报表包含总体样本量、子组数、样本均值、总 Sigma、组内 Sigma 和稳定性指标。对于三向控制图，还显示了组间 sigma 以及组间和组内 sigma 值。若指定了阶段变量，则为每个阶段提供一组值。

注意：只有打开 “限值汇总” 报表时才会显示 “过程 Sigma” 报表。

获取限值 检索存储在打开或保存的数据表中的控制限。

显示排除区域 显示 （打开）或移除 （关闭）图中已排除样本的区域。当完全排除的子组显示在位置图上时，它们将显示为暗显点，以指示它们已被排除。

警告：“显示排除区域” 选项会影响图表。排除的样本将从控制限的计算中移除，无论该选项是启用还是禁用。仅当同时选定 “检验排除的子组” 和 “显示排除区域” 选项时，排除的样本才会包含在报警率计算中。

设置子组大小 （若指定了子组变量，则不可用。）设置子组大小。计算限值和 sigma 时会考虑缺失值。

注意：若使用 “设置子组大小” 选项，则会自动启用 “显示排除区域” 选项。

保存限值 通过以下方式之一保存控制限：

在列中 将控制限另存为现有数据表中响应变量的列属性。若限值为常数，则保存报表中每种图表类型的下控制限值、平均值和上控制限值。该选项不可用于阶段图。此外，若样本大小对于每个图表并不一致，则该选项无效。

在新表中 将每个图表的标准差和均值保存到新的数据表中。若限值为常数，则同时保存每个图表的下控制限值、平均值、上控制限值和样本大小。若存在阶段，则为每个阶段保存一组新值。每个统计量对应一行，每个 Y 变量对应一列。

在高型新表中 （不适用于稀有事件图、计数图或阶段图。）将每个图表的标准差、均值和 Sigma 保存到新的数据表中。若限值为常数，则同时保存每个图表的下控制限值、平均值、上控制限值和样本大小。每个 Y 变量对应一行，每个统计量对应一列。此外还保存一个可用于“过程筛选”平台的 Sigma 列。

保存汇总 创建新数据表，其中包含样本标签、样本大小、所标绘的统计量、中心线、控制限以及任何检验、警告和失败之类的信息。包含在表中的特定统计量取决于图表类型。

图形间距 设置图形之间的空白量。

包括缺失类别 支持图形将含有缺失值的行收集到一个分类列中，并将这些缺失值在图形上显示为单独类别。若禁用该选项，则不仅在图形中隐藏含有缺失 X 值的所有行，还会将这些行从计算中删除。

该选项对于连续 X 变量或分类 Y 变量不可用，因为没有好的方式在相关轴上显示收集的缺失值。默认情况下启用该选项。

注意：若启用了“包括缺失类别”，则当分类 X 变量有缺失值时，“控制图生成器”中的能力分析结果将不匹配“过程能力”平台中的能力分析结果。

使用事件选择器 （仅可用于具有数值非连续 Y 变量的计数图。）将有序型数值数据分类并提供各个数值型水平建模选择。

警告脚本 支持您编写和运行指示数据未通过特殊原因检验的脚本。请参见第 50 页的“检验”。结果可以写入日志或大声说出，并且提供一个选项来包含对检验失败原因的解释。您还可以使用自定义脚本选项将结果发送到电子邮件。

在调用警告脚本时，以下变量在发出的脚本和随后的 JSL 脚本中都可用：

`qc_col` 是列名

`qc_test` 是失败的检验

`qc_sample` 是样本号

`qc_phase` 是出现失效阶段的标签

请参见《Scripting Guide》，了解关于编写定制警告脚本的详细信息。

注意：警告脚本在使用本地数据过滤器的报表中不可用。

按行顺序排序 按水平在数据表中的显示顺序排列所有子组和阶段变量。这适用于嵌套子组和阶段变量的所有组合。

检验排除的子组 （仅当选定“显示排除区域”选项时才可用。）在检验的计算中包括（启用）或排除 （禁用）完全排除的子组。当排除的子组被显示而且未选定 “检验排除的子组”选项时，排除的子组将被视为缺失值。

注意：对于任何依赖于依次相连点的检验 （游程检验），完全排除的子组将被视为缺失，并重新开始依次相连点的计数。

控制图对话框 （仅当通过“控制图”启动窗口启动控制图时可用。）使用用于创建控制图的原始设置打开 “控制图”启动窗口。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

- 列切换器仅对具有两个或更少相关图表的单个 Y 变量可用。根据所选的图表类型，只有适合 Y 角色的列才包含在列切换器的列列表中。
- 在“控制图生成器”中，默认情况下启用“自动重新计算”选项，不能禁用它。
- 使用本地数据过滤器时，可以取消选择“显示排除区域”选项以进行更有侧重点的探索。

选项面板和右击图表选项

以下选项显示在 “控制图生成器”报表左侧以便于访问，右击某个图表时也会显示这些选项。

点 提供以下选项：

统计量 更改在图上标绘的统计量。请参见第 49 页的“统计量”。

单个点 显示或隐藏子组中的单个观测。仅适用于子组变量或“设置样本大小”。该选项不适用于“计数”图表类型或稀有事件图。

箱线图 （仅可用于 Shewhart 计量图。）在图表上显示或隐藏箱线图。

显示连接线 显示各点之间的连接线。

显示点 显示或隐藏图上的点。

限值 提供以下选项：

Sigma 指定计算 sigma 的方法。请参见第 49 页的“Sigma”。

区域 （仅可用于“计量”和“计数”图表类型。）显示或隐藏图上的区域。均值两侧有三个等宽区。不在下控制限下方或上控制限上方绘制区域。若“计量”图的限值不以均值为中心，则将以 min(平均值-下控制限,上控制限-平均值)/3 所得的值用作每个区域的宽度。“计数”图的区域使用 (上控制限-平均值)/3 的宽度。

区域着色 按范围显示或隐藏着色区域。区域 C 着绿色，区域 A 和 B 着黄色，而区域 A 之外的区域着红色。

规格限 （仅当数据表具有“规格限”列属性或是您使用“添加规格限”选项指定“规格限”时才可用。）显示或隐藏图上的规格限。默认情况下，若“规格限”列属性具有选定的“显示为图形参考线”选项，则会显示规格限。有关添加“规格限”列属性的信息，请参见《使用 JMP》。

添加规格限 支持您输入规格限。

设置控制限 支持您输入检验的控制限。在“设置控制限”窗口中点击确定后，在各组间均匀设置指定的控制限。再次选择该选项可删除指定的控制限。

添加限值 指定要在图上标绘的其他控制限。这些限值不用在检验中。

显示上限 显示或隐藏图上的上控制限。若在图上隐藏上控制限，则“检验超出界限”和“检验 1”选项不会标记与隐藏的上控制限关联的点。

显示下限 显示或隐藏图上的下控制限。若在图上隐藏下控制限，则“检验超出界限”和“检验 1”选项不会标记与隐藏的下控制限关联的点。

显示中心线 显示或隐藏图上的中心线。

添加散度图 向图区域添加散度图。使用点选项更改图表类型。散度图可通过标绘众多散度形式之一（包括极差、标准差或移动极差）来说明数据中的变异。仅适用于“计量”图表类型。

注意：您可以使用“文件”>“首选项”>“平台”>“控制图生成器”中的“散度图”和“汇总散度图”首选项定制默认散度图类型。

设置子组大小 设置子组大小。计算限值和 sigma 时会考虑缺失值。

警告 提供以下选项：

定制检验 支持您设计定制检验，并同时选择或取消选择多个检验。选定该选项后，随即显示“定制检验”窗口，供您设计检验。选择检验说明，然后输入所需的编号 (n) 和标签。您可以将设置保存到首选项中，也可以恢复默认设置。仅适用于“计量”和“计数”图表类型。

检验 支持您选择要启用的统计控制检验。有关检验的详细信息，请参见第 50 页的“检验”。仅适用于“计量”和“计数”图表类型。

注意：悬停在图上某个标记点上方可查看失败的检验的说明。

Westgard 规则 指定已启用的 Westgard 统计控制检验集。由于 Westgard 规则基于 sigma 而不是区域，所以可以在不考虑常数样本大小的情况下计算这些规则。有关检验的详细信息，请参见第 53 页的“Westgard 规则”。仅适用于“计量”和“计数”图表类型。

检验超出界限 （在 JMP 中称为检验 15）支持对超出控制限的所有点进行检验。这些点标识在图上。该检验适用于具有限值的所有图，不论样本大小是否相同。

注意：若隐藏上控制限或下控制限，则“检验超出界限”选项不会标记超出控制图上未显示的界限的点。

删除图形 删除控制图。

删除位置图 （仅当右击位置图时才可用。）从报表中删除位置图。

删除散度图 （仅当右击散度图时才可用。）从报表中删除散度图。

注意：有关“行”、“图形”、“定制”和“编辑”菜单的说明，请参见《使用 JMP》。

统计量

您可以更改图上各点所表示的统计量。可用选项取决于选定的图表类型。

对于“计量”图表类型，您可以使用以下选项更改图上各点所表示的统计量：

单值 创建一个图，其中每个点都表示数据表中的单个值。

平均值 创建一个图，其中每个点都表示子组中的值的平均值。

极差 创建一个图，其中每个点都表示子组中的值的极差。

标准差 创建一个图，其中每个点都表示子组中的值的标准差。

基于均值的移动极差 计算两个相邻子组均值之间的极差差值。

基于标准差的移动极差 计算两个相邻子组标准差之间的极差差值。

移动极差 创建一个图，其中每个点都是两个相邻观测之间的差值。

注意：仅当指定了样本大小大于 1 的子组变量或设置了样本大小时，“平均值”、“极差”、“标准差”、“基于均值的移动极差”和“基于标准差的移动极差”方法才会显示。

对于“计数”图表类型，您可以使用以下选项更改图上各点所表示的统计量：

比例 创建一个图，其中每个点都表示子组样本中的项比例。

计数 创建一个图，其中每个点都表示子组样本中的项数。

对于“稀有事件”图表类型，图上各点所表示的统计量使用“计数”选项。

Sigma

您可以更改计算图表 sigma 的方法。可用选项取决于选定的图表类型。

对于“计量”图表类型，您可以使用以下选项：

极差 使用子组中数据的极差估计 sigma。

标准差 使用子组中数据的标准差估计 sigma。

移动极差 使用移动极差估计 sigma。移动极差是两个相邻点之间的差值。

移动极差中位数 使用移动极差中位数而不是平均移动极差来估计 sigma。

Levey-Jennings 使用所有观测的标准差估计 sigma。若图中有阶段，则为各个阶段分别计算 sigma。

对于“计数”图表类型，您可以使用以下选项：

二项 使用二项分布模型估计 sigma。该模型指示一系列实验中的成功次数，其中每次实验结果都有一定的成功概率。选择 “二项”可生成 P 图或 NP 图。

Poisson 使用 Poisson 分布模型估计 sigma。该模型指示事件数以及这些事件在给定时间间隔内出现的时间。选择 Poisson 可生成 C 图或 U 图。

对于“稀有事件”图表类型，您可以使用以下选项：

负二项 使用负二项分布模型估计 sigma。该模型指示在达到指定的失败次数之前，一系列试验中的成功次数。选择 “负二项”可生成 G 图。

Weibull 使用 Weibull 分布模型估计 sigma。该模型指示平均失败间隔时间。选择 Weibull 可生成 T 图。

检验

弹出菜单或窗口左侧中的“警告”选项显示“检验”选项的子菜单。您可以从菜单中选择一个或多个特殊原因的检验（Western Electric 规则）。Nelson (1984) 制定了一种用于在控制图上标识特殊检验的编号表示法。这些检验同时适用于大小相等和不等的样本。

若选定的检验对于特定样本显示为阳性，则该点将使用检验编号来标注。若选择要显示多个检验，而有不止一个检验在某个特定点上发出信号，则该点旁边将显示指定的编号最低的检验的标签。您可以悬停在图上某个标记点上方以查看失败的检验的说明。

提示：要一次添加或删除几个检验，请在**警告 > 检验**下的“控制面板”中选择或取消选择这些检验。

第 51 页上的表 3.8 列出并解释了八个检验，图 3.9 演示了这些检验。以下规则适用于各检验：

- 上限与下限之间的范围划分为六个区，每个区的宽度均为一个标准差。
- 这些区分别标注为 A、B、C、C、B、A，C 区距离中心线最近。
- 若某个点位于 C 区与 B 区分隔线之外，即距离中心线超过一个标准差，则该点位于 B 区之中或之外。
- 位于两个区分隔线上的任何点均被视为属于靠内区。因此，若某个点位于 A 区和 B 区之间的线上，该点被视为处于 B 区中。
- 当指定阶段变量时，每个检验的计数在每个阶段开始时重置。

注意：

- 检验 1 到 8 适用于所有 Shewhart“计量”图表类型。

- 检验 1、2、5 和 6 分别适用于图的上半部和下半部。
- 检验 3、4、7 和 8 适用于整个图。
- 一旦触发试验检验（基于相邻观测的检验），转移到下一个样本之前，计数不会重置为 0。
- 游程检验基于“显示排除区域”和“检验排除的子组”选项的设置来处理排除的行。
  - 默认情况下，两个选项都处于选定状态，排除的行包括在游程检验计算中。
  - 若选定“显示排除区域”选项而未选定“检验排除的子组”选项，则排除的行将被视为缺失，并且在移动到下一个样本时，游程检验的计数将重置为 0。
  - 若未选定“显示排除区域”选项，则排除的行会被认为好像被删除了一样。
- 检验 5 到 8 不可用于计数图。

请参见 Nelson (1984, 1985)，了解关于如何使用这些检验的更多建议。

图 3.8 Western Electric 区域规则

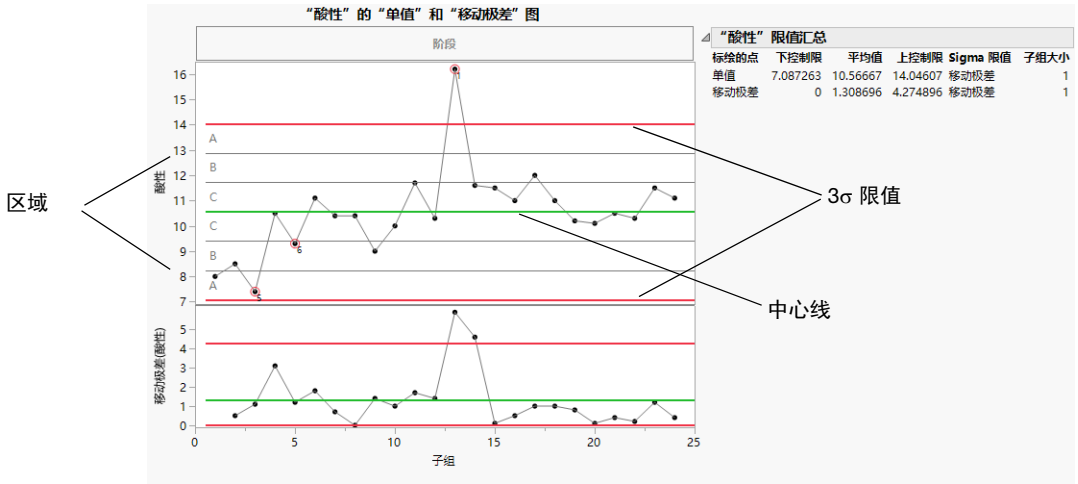


表 3.8 特殊原因检验的说明与解释[a]

检验 1	A 区外的一点（上或下）	检测均值中的偏移、标准差增大或过程中的单个偏差。为了解释检验 1，可以使用任何散度图（R、S 或 MR）排查变异增大情况。 请注意，若隐藏上控制限或下控制限，则“检验 1”选项不会标记与控制图上未显示的界限关联的点。
检验 2	C 区单侧（上或下）或 C 区外的连续九个点	检测过程均值中的偏移。

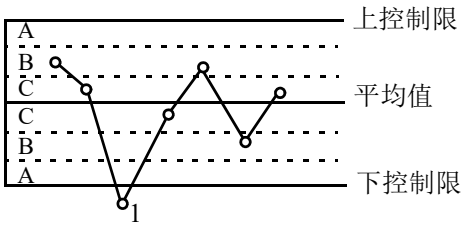
表 3.8 特殊原因检验的说明与解释[a] (续)

检验 3	持续增大或减小的连续六个点（图上任意位置）	检测过程均值中的趋势或漂移。
检验 4	交替升降的连续十四个点（图上任意位置）	检测系统效应，如两个轮换使用的机器、供应商或操作员。
检验 5	连续三个点中的两个点落在 A 区之中或之外，而且点本身落在 A 区之中或之外，这两个点必须位于同侧（上或下）	检测过程平均值中的偏移或标准差增大。三个点中任意两个点都给出阳性检验结果。
检验 6	连续五个点中的四个点落在 B 区之中或之外，而且点本身落在 B 区之中或之外，这四个点必须位于同侧（上或下）	检测过程均值中的偏移。五个点中任意四个点都给出阳性检验结果。
检验 7	C 区之中、中心线上下连续十五个点	当单个子组中的观测来自具有不同均值的多个源时，检测分层的子组。还检测变异减小。
检验 8	在中心线两侧连续八个点，没有一点落在 C 区	当一个子组中的观测来自单个源，但不同子组来自具有不同均值的多个源时，检测分层的子组。

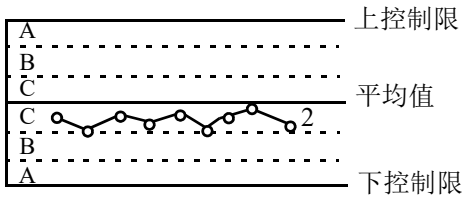
a. Nelson (1984, 1985)

图 3.9 特殊原因检验的演示[1]

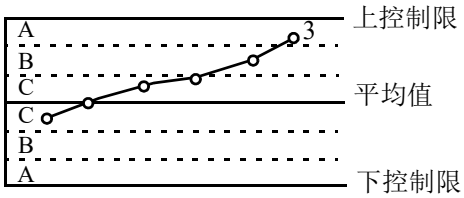
检验 1：A 区外的一点



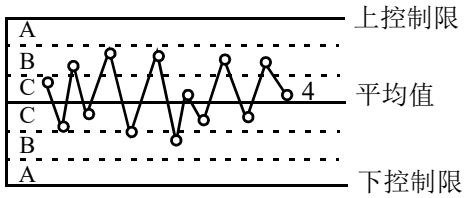
检验 2：C 区单侧（上或下）或 C 区外的连续九个点



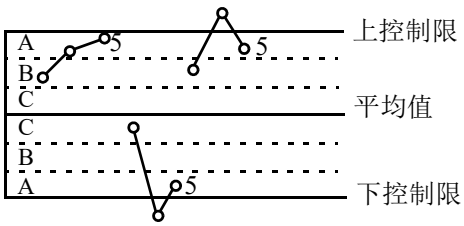
检验 3：持续增大或减小的连续六个点



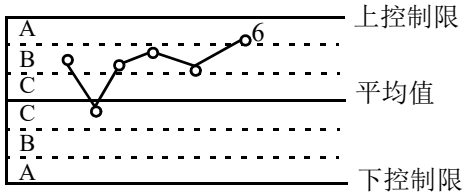
检验 4：交替升降的连续十四个点



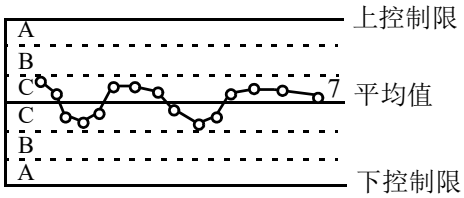
检验 5：连续三个点中的两个点落在 A 区之中或之外



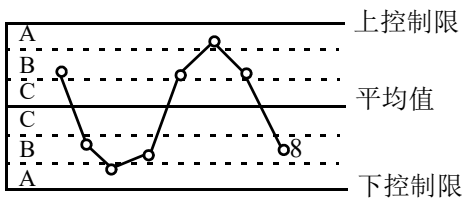
检验 6：连续五个点中的四个点落在 B 区之中或之外



检验 7：C 区之中（中心线上下）连续十五个点



检验 8：在中心线两侧连续八个点，没有一点落在 C 区之中



Westgard 规则

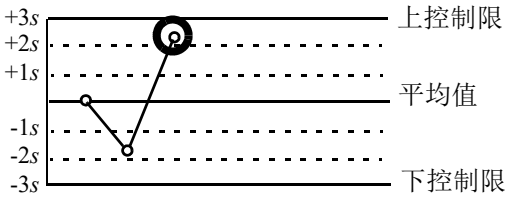
当您右击图或窗口左侧后，可以通过“警告”选项的“Westgard 规则”子菜单来实施 Westgard 规则。不同的检验缩写为针对特定检验的决策规则。例如，1 2s 指的是一个点距离均值两个标准差的检验。

1. Nelson (1984, 1985)

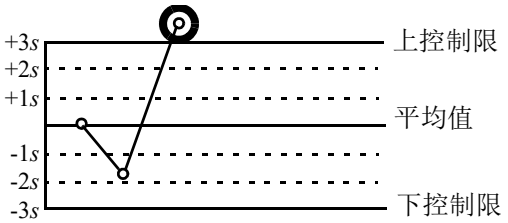
注意：

- 一旦触发试验检验（基于相邻观测的检验），转移到下一个样本之前，计数不会重置为 0。
- 游程检验基于“显示排除区域”和“检验排除的子组”选项的设置来处理排除的行。
  - 默认情况下，两个选项都处于选定状态，排除的行包括在游程检验计算中。
  - 若选定“显示排除区域”选项而未选定“检验排除的子组”选项，则排除的行将被视为缺失，并且在移动到下一个样本时，游程检验的计数将重置为 0。
  - 若未选定“显示排除区域”选项，则排除的行会被认为好像被删除了一样。

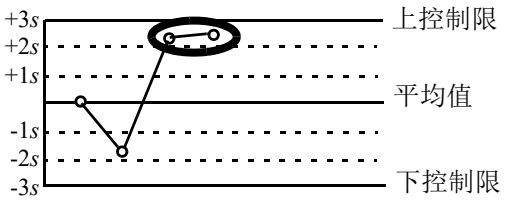
规则 1 2S（在 JMP 中称为检验 9）通常用于 Levey-Jennings 图，其中的控制限设置为距离均值两个标准差。有任一点超出这些限值就会触发该规则。



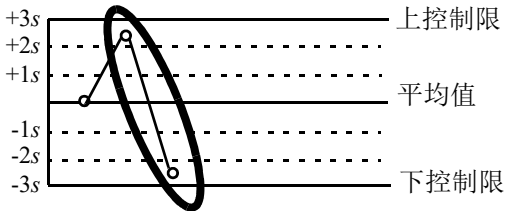
规则 1 3S（在 JMP 中称为检验 10）指的是常用于 Levey-Jennings 图的规则，其中的控制限设置为距离均值三个标准差。有任一点超出这些限值就会触发该规则。



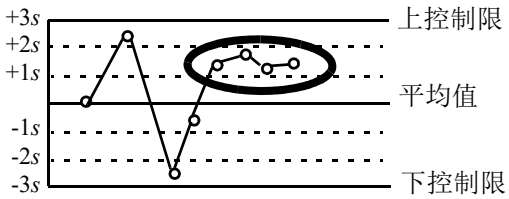
规则 2 2S（在 JMP 中称为检验 11）在两个连贯控制测量值位于距离均值两个标准差以外时触发。



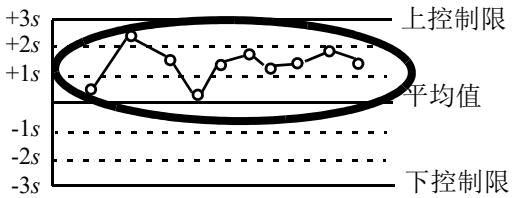
规则 R 4S（在 JMP 中称为检验 12）在以下条件下触发：一个测量值位于距离均值两个标准差以外，而且前一个测量值位于相反方向上距离均值两个标准差之外的位置，以至于差值大于四个标准差。



规则 4 1S（在 JMP 中称为检验 13）在四个连贯测量值位于距离均值一个标准差以外时触发。



规则 10 X（在 JMP 中称为检验 14）在十个连贯点位于均值同侧时触发。



“控制图生成器”右击轴选项

删除图形 删除整个图形。

删除 删除某个变量。

注意: 若图中有多个图表类型，则会显示列出不同图表的子菜单。您可以选择要删除的图表。

有关“轴设置”、“恢复轴”、“添加/删除轴标签”、“保存至列属性”和“编辑”选项的详细信息，请参见《使用 JMP》。

使用控制限

控制限基于您过程的能力，并且可告知您关于过程中变异性的信息。基于创建控制限时的数据计算上控制限 (UCL)、中心线和下控制限 (LCL)。您可以使用这些计算出的控制限来指示过程何时发生了更改。

务必要注意的是：控制限不同于规格限，后者常常用在能力分析中。

表 3.9 控制限与规格限的比较

控制限	规格限
基于数据计算	由客户或设计定义
基于变异性	基于系统要求
过程指标	客户指标

控制限示例

在本例中，考虑某公司的印刷过程。变异可能导致线条失真，包括偏度、厚度和长度等问题。在本例中，我们要考虑线条的长度。印刷长度为 16 cm +/- 0.2 cm 的线条被视为合格线条。线条再长些时，句子可能超出页面。线条短一些，就会浪费大量页面空间。对于每批刊印，都抽取第一本书和最后一本书进行测量。在每本书中间的指定页面上测量线长。

您想要了解：该过程是否受控（稳定）？印刷质量是否稳定一致？当我们改进印刷工艺时会发生什么？质量会改进吗？要回答这些问题，我们需要创建控制图并使用控制限。

本例分为三个部分。多数情况下，您最初需要“创建基线控制图”，以便让 JMP 为您计算控制限。然后，要将这些控制限应用到新数据，您需要“指定控制限”或“指定多组控制限”（对于阶段数据）。

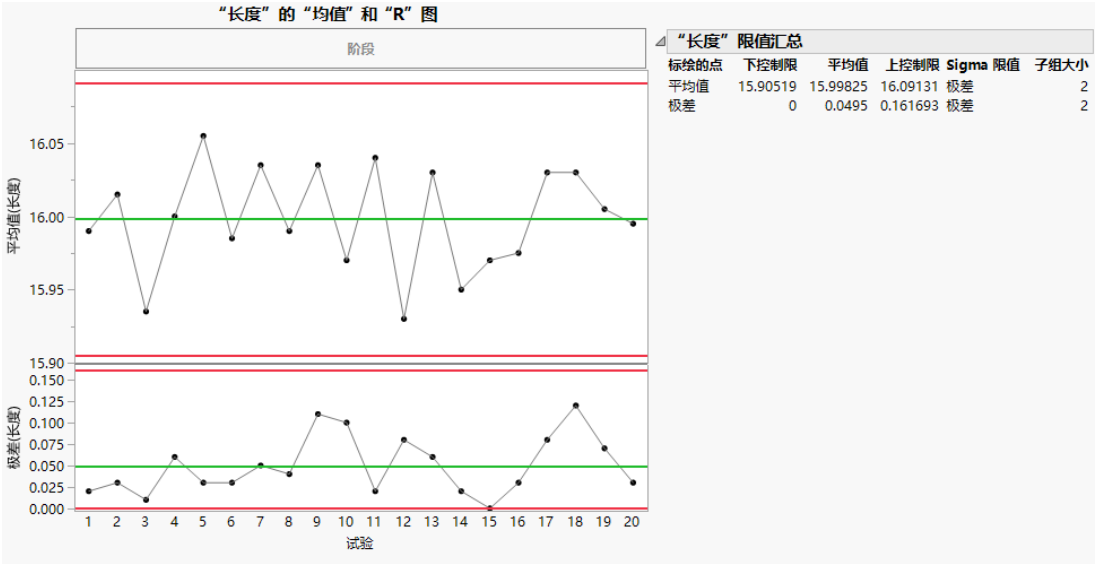
创建基线控制图

首先，检查现有过程是否受控。若受控，我们就可以将 JMP 创建的控制限用作基线或历史限值。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Line Length.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**长度**拖至 Y 区域。

   随即显示“长度”的“单值”和“移动极差”图。若您的数据没有自然子组，那么该图很合适。但是，本例存在自然子组，即每次印刷试验。
4. 将**试验**拖至**子组**区域（位于底部）。

图 3.10 按印刷试验划分的 “长度” 的均值图和 R 图



有三条线水平横穿均值图和 R 图。它们分别是计算的 LCL （下控制限）、 Avg （平均值）和 UCL （上控制限）。

理想状态下，我们希望所有点都落在控制限以内，而且我们希望这些点随机落在这些限值内。查看图形后我们会看到：没有任何点落在控制限以外，而且这些点看起来没有任何模式。为了进一步调查，执行 Western Electric 检验，以检查可能导致这些检验失败的模式和趋势。（Western Electric 检验亦称 Nelson 检验。）

5. 在均值图中，右击并选择**警告 > 检验 > 所有检验**。

   请注意，没有任何点被圈定或标记出来。这意味着我们的过程受控且稳定。

   若我们已经确定我们的过程不受控，我们会调查失控点，或者努力改变我们的过程以使其受控。对于本例，由于过程已然受控或稳定，所以我们可以跳过该步。现在，您可以将这些控制限用于新数据。前进到第 57 页的 “指定控制限” 或第 62 页的 “指定多组控制限” （查看带有阶段数据的示例）。

指定控制限

既然我们已经确定过程受控，我们就可以将这些历史限值用于新数据，来查看新数据如何与现有过程进行对比。要使用历史限值，我们需要指定控制限而不是让 JMP 计算控制限。

在 JMP 中指定控制限有若干方法：

- 第 58 页的 “ “设置控制限” 选项”
- 第 59 页的 “添加列属性”
- 第 60 页的 “使用 “获取限值” 选项”

- 第 61 页的“排除行”

“设置控制限”选项

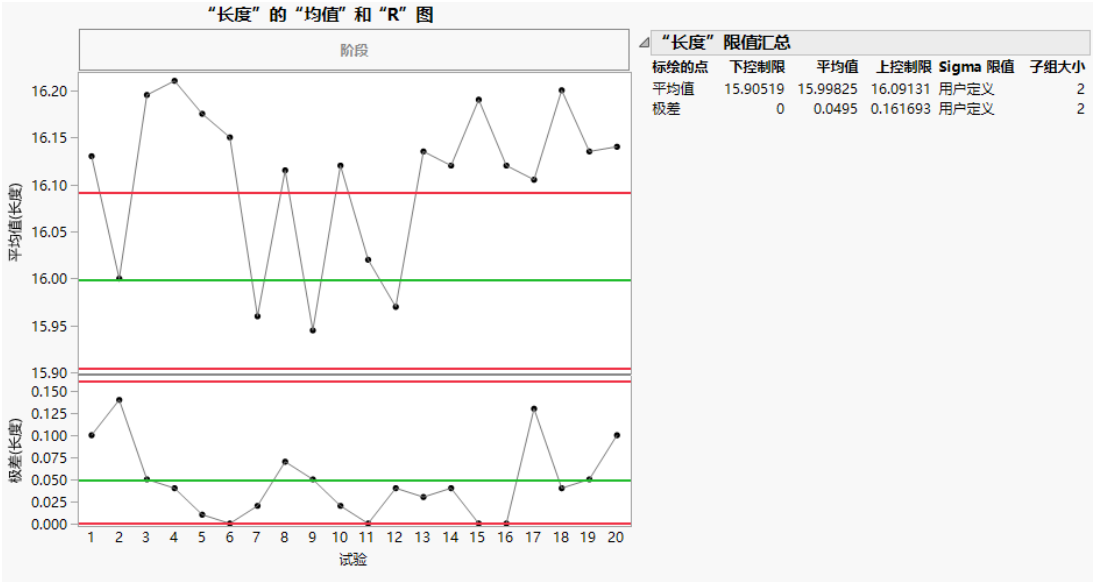
一种简便的指定控制限的方法就是使用 “控制图生成器”中的 “设置控制限”选项。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/New Length Data.jmp。
   这是包含新数据的表。
2. 选择**分析 > 质量和过程 > 控制图生成器**。
3. 将**长度**拖至 Y 区域。
4. 将**试验**拖至**子组**区域 （位于底部）。
5. 在平均值 （均值）图中右击，然后选择**限值 > 设置控制限**。
6. 输入以下限值：
   - 下控制限 - 15.90519
   - 平均值 - 15.99825
   - 上控制限 - 16.09131

   这些是来自图 3.10 中的平均值 （均值）图的历史限值。
7. 点击**确定**。
8. 在极差 (R) 图中右击，然后选择**限值 > 设置控制限**。
9. 输入以下限值：
   - 下控制限 - 0
   - 平均值 - 0.0495
   - 上控制限 - 0.161693

   这些是来自图 3.10 中的极差 (R) 图的历史限值。
10. 点击**确定**。

图 3.11 带有历史限值的 “长度” 的均值图和 R 图



JMP 使用您定义的历史控制限值而不是从数据计算限值。在 “‘长度’限值汇总” 表中，请注意 “Sigma 限值” 现在显示 “用户定义”。许多点现在落在限值以外。此外，平均值高于基线过程中的平均值。该过程看起来不同于我们用于计算基准控制限的原始过程。

添加列属性

另一种指定控制限的方法是向新数据表中的列添加 “控制限” 列属性。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/New Length Data.jmp。
   这是包含新数据的表。
2. 选择**长度**列，然后点击**列 > 列信息**。
3. 点击**列属性 > 控制限**。
4. 由于已选定均值，所以为平均值 （均值）图输入以下固定限值：
   - 平均值 - 15.99825
   - 下控制限 - 15.90519
   - 上控制限 - 16.09131

   这些是来自图 3.10 中的平均值 （均值）图的历史限值。
   让 “子组大小” 值保持缺失。该值不用在 “控制图生成器” 平台中。
5. 点击**均值 > R**。为极差 (R) 图输入以下固定限值：
   - 平均值 - 0.0495
   - 下控制限 - 0

– 上控制限 - 0.161693

这些是来自图 3.10 中的极差 (R) 图的历史限值。

让 “子组大小”值保持缺失。该值不用在 “控制图生成器”平台中。

6. 点击**确定**。

您已经为 “长度”列的均值图和 R 图输入了控制限。现在可以创建控制图。

7. 选择**分析 > 质量和过程 > 控制图生成器**。
8. 将**长度**拖至 Y 区域。
9. 将**试验**拖至**子组**区域 （位于底部）。

控制图与图 3.11 完全相同。

使用 “获取限值”选项

用于指定控制限的 “获取限值”方法是最为灵活的方法。您应该在以下情况下使用该方法：

- 若对许多不同的过程都有控制限
- 若对每个阶段有不同的控制限（请参见第 62 页的“指定多组控制限”）

要使用 “获取限值”方法，您需要一个定义历史限值的数据表。有关如何创建限值表的详细信息，请参见 “传统控制图”一章中第 306 页的“保存和检索限值”。

注意：若未指定子组变量，则 “获取限值”选项会使用限值表中的子组大小 (_Sample Size)。此外，当文件中缺少限值时，JMP 还会查找 sigma (_Std Dev)。若在限值文件中未指定下控制限和上控制限（若同时找到平均值和 sigma，并且子组大小为常数），则该选项将基于平均值、子组大小和 sigma 设置限值。

在本例中，已经创建了一个限值数据表。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/New Length Data.jmp。

这是包含新数据的表。

2. 选择**分析 > 质量和过程 > 控制图生成器**。
3. 将**长度**拖至 Y 区域。
4. 将**试验**拖至**子组**区域 （位于底部）。
5. 点击 “控制图生成器”红色小三角，然后选择**获取限值**。
6. 选择**其他**并点击**确定**。
7. 导航并打开本例的限值数据表 （名为 Length Limits.jmp）。默认情况下，该文件位于以下位置：
   – 在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data\Quality Control
   – 在 macOS 上：\Library\Application Support\JMP\16\Samples\Data\Quality Control

控制图与图 3.11 完全相同。

排除行

另一种指定控制限的方式是排除数据表中的行。该方法的一个优点是：您可以在同一图形中同时看到历史数据和新数据。这有助于直观演示和调查数据收集期间出现的差异。

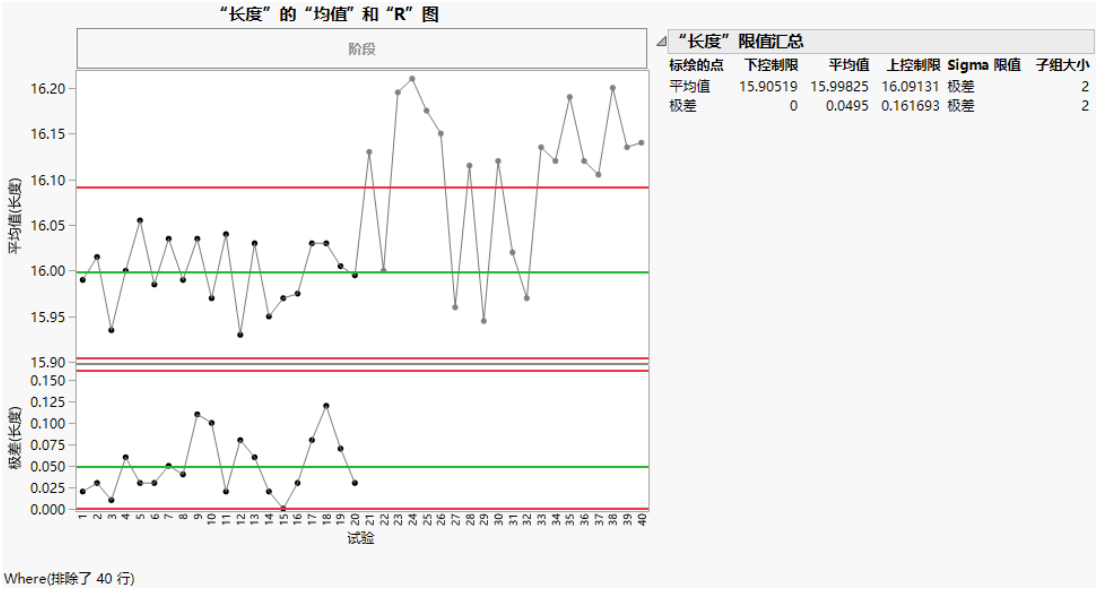
要使用该方法，您必须满足以下条件：

- 新数据和旧数据必须位于同一数据表中。
- 历史数据和新数据必须都具有相同的子组大小。
- 必须在数据表中排除所有新数据（使用“行”>“排除/撤销排除”）。

在本例中，新数据已被排除。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Combined.jmp。
   该表包含旧数据和新数据，并排除与新数据对应的行。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**长度**拖至 Y 区域。
4. 将**试验**拖至**子组**区域 （位于底部）。

图 3.12 带有已排除数据的 “长度”的均值图和 R 图



JMP 仅使用未排除行（历史数据）创建控制限。新数据（已排除数据）标绘在图形上（变暗），但这些数据不用在任何计算中。

指定多组控制限

在本例中，您想要为过程的不同阶段设置不同的控制限。列属性“设置控制限”和“排除行状态”方法在这种情况下无效，因为这些方法仅限于整个图只有一组控制限。对于带有阶段的控制图，您需要使用“获取限值”方法。

在这家印刷公司，目标是要在三个不同位置减少破坏纸张与书脊之间的连结所需的力量的变异性。每个地点都有不同的机器和不同的操作员，而且都位于不同国家 / 地区；因此，每个地点都有一组唯一的历史限值。对于全部三个位置，该公司都执行以下操作：

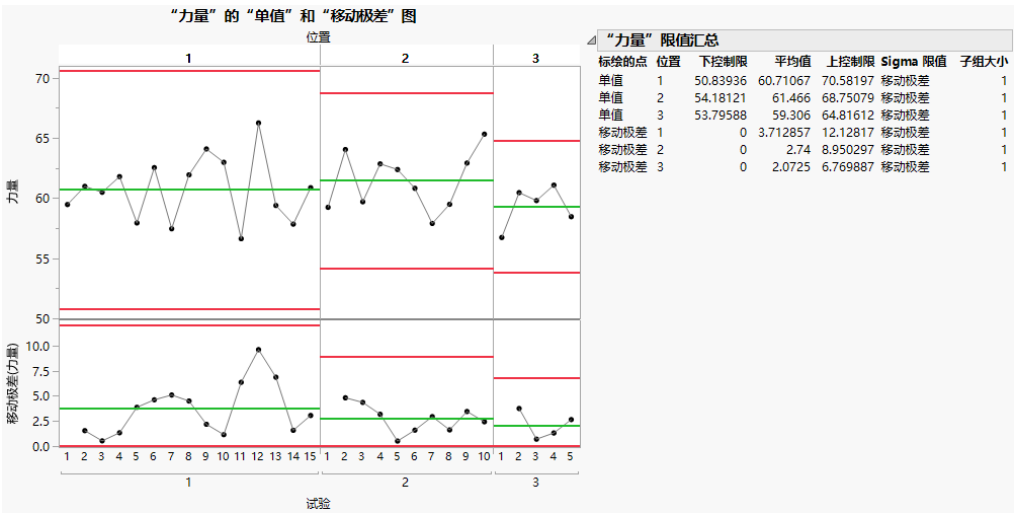
1. 基于现有过程数据创建基线控制图。
2. 基于设计的实验更改过程。
3. 从新过程收集数据。
4. 基于新的过程数据创建新的控制图。

目标是要使用旧过程中的历史限值在控制图上绘制新数据。这样一来，该印刷公司即可将新过程与旧过程的限值进行比较。

基于现有过程创建控制图

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/Phase Historical Data.jmp。
   该表包含全部三个位置的现有过程数据。
2. 选择**分析 > 质量和过程 > 控制图生成器**。
3. 将**力量**拖至 Y 区域。
4. 将**试验**拖至**子组**区域 （位于底部）。
5. 将**位置**拖至**阶段**区域。

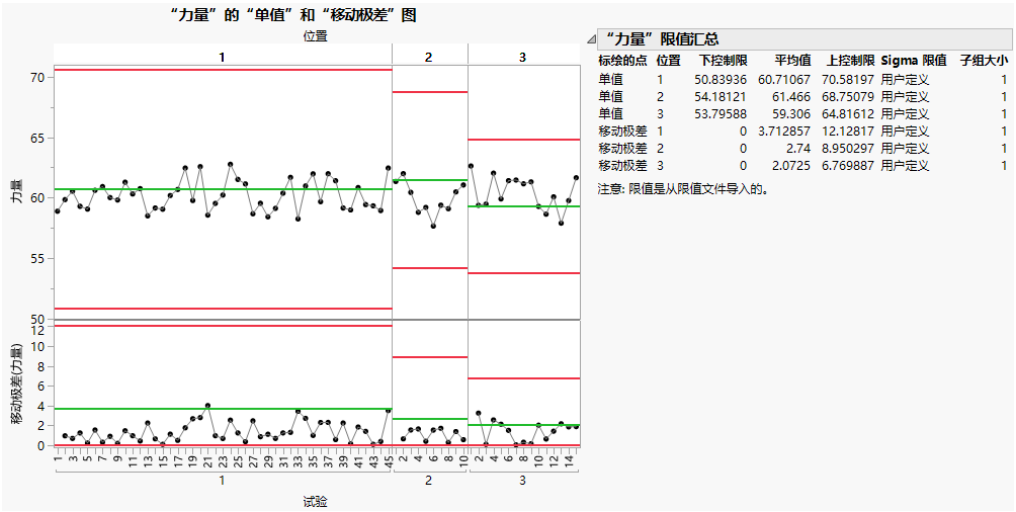
图 3.13 现有数据的基线控制图



基于已更新过程创建控制图

1. 从图 3.13 中的报表，点击“控制图生成器”红色小三角，然后选择**保存限值 > 在新表中**。这将创建限值表。
2. 将这个新的限值表保存到任意位置，以便您以后访问。
3. 选择**帮助 > 样本数据库**，然后打开 Quality Control/Phase New Data.jmp。该数据是在对过程进行更改后从全部三个位置收集的数据。
4. 选择**分析 > 质量和过程 > 控制图生成器**。
5. 将**力量**拖至 Y 区域。
6. 将**试验**拖至**子组**区域 （位于底部）。
7. 将**位置**拖至**阶段**区域。
8. 点击“控制图生成器”红色小三角，然后选择**获取限值**。打开您在第 2 步中保存的限值表。这会将历史限值应用到 “控制图生成器”报表中的新数据。

图 3.14 基于历史限值的新数据的控制图



现在您可以看到新数据（过程变化后）如何与历史过程限值（过程变化前）作比较。没有任何点落在位置或散度图的控制限之外。我们的目标是要减少变异性。通过查看移动极差图，您可以看到多数点都落在平均线以下。对于地点 1 和 2，很明显破坏纸张与书脊之间的连结所需的力量的变异性已经降低。地点 3 处的下降没有地点 1 和 2 处严重。印刷过程的改进看起来已经成功地降低了变异性。

“控制图生成器”中的排除和隐藏的样本

以下各项总结了控制图生成器中的排除和隐藏的样本的用法：

- 排除的子组不用于控制限的计算，默认情况下在图表上显示为暗显点。若未选定“显示排除区域”选项，则排除的子组的点不会出现在图中，在“特殊原因的检验”中被视为缺失，并且不包括在“特殊原因的检验”的点计数中。
- 隐藏的观测用在控制限计算中，但不显示在图中。
- 选定“检验排除的子组”选项时，隐藏和排除的行将包含在“特殊原因的检验”的点计数中。排除的行可使用特殊原因旗标来标记。无法标记隐藏点。若“特殊原因的检验”的旗标位于隐藏点上，则该旗标不会显示在图中。
- 对于部分排除的子组，若子组中有一个或多个观测被排除但该子组中至少有一个观测包含在内，则排除的观测不包含在点统计量或限值的计算中。
- 针对全部数据（甚至包括排除值）检查负数和非整数数据。
- 仅当选定“检验排除的子组”选项时，检验才应用于所有排除的子组。

“控制图生成器”的更多示例

注意：在本节中，某些示例显示“控制面板”而另一些未显示。要显示或隐藏“控制面板”，请从“控制图生成器”红色小三角菜单中选择“显示控制面板”。

- “单个测量值和移动极差图示例”
- “均值图和 R 图阶段示例”
- “具有子组大小变化的均值图和 S 图的示例”
- “运行图示例”
- “P 图示例”
- “NP 图示例”
- “C 图示例”
- “U 图示例”
- “G 图示例”
- “T 图示例”
- “三因子控制图示例”

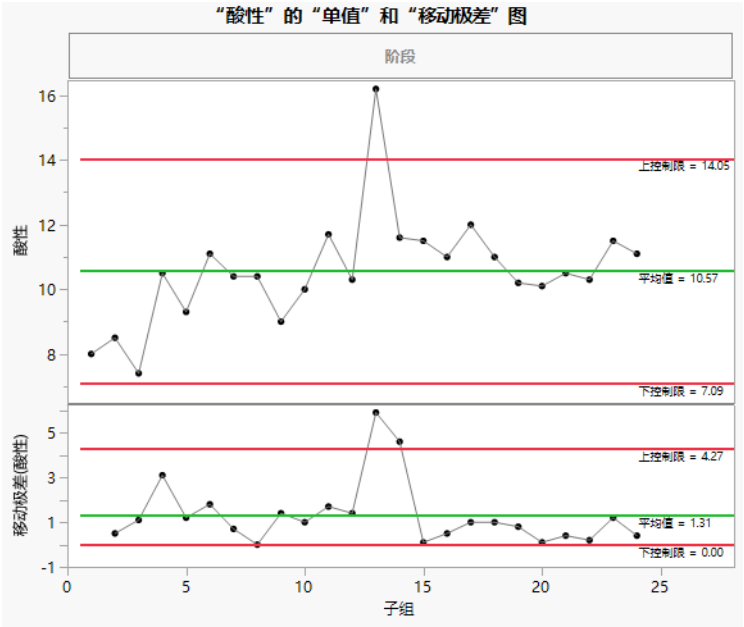
单个测量值和移动极差图示例

Quality Control 样本数据文件夹中的 Pickles.jmp 数据包含各缸泡菜的酸含量。由于泡菜对酸度很敏感并且使用大缸来生产，所以较高的酸度会破坏整缸泡菜。每天下午 1 点、2 点和 3 点都会测量四个缸中的酸度。数据表中记录了日期、时间和酸度的测量值。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Pickles.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**酸性**拖至 Y 角色。
4. 点击**控制图生成器**红色小三角并选择**显示限值标签**。

   该选项为两个图中的控制限和平均值添加标签。

图 3.15 针对 “酸性”的单个测量值和移动极差图

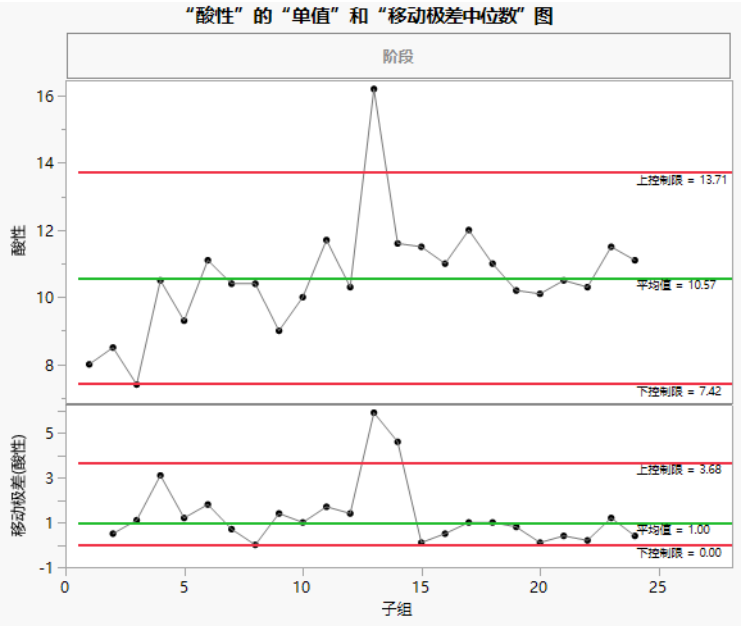


单个测量值和移动极差图监控生产的每一缸泡菜的酸度（子组大小为 1）。缸 13 的酸度高于上控制限 14.05。

您还可以查看“移动极差中位数”图。继续执行以下步骤，更改图表以使用移动极差中位数。

5. 在 “限值 [1]”分级显示项中，将 Sigma 设置改为**移动极差中位数**。
6. 在 “限值 [2]”分级显示项中，将 Sigma 设置改为**移动极差中位数**。

图 3.16 针对 “酸性”的单个测量值和移动极差中位数图



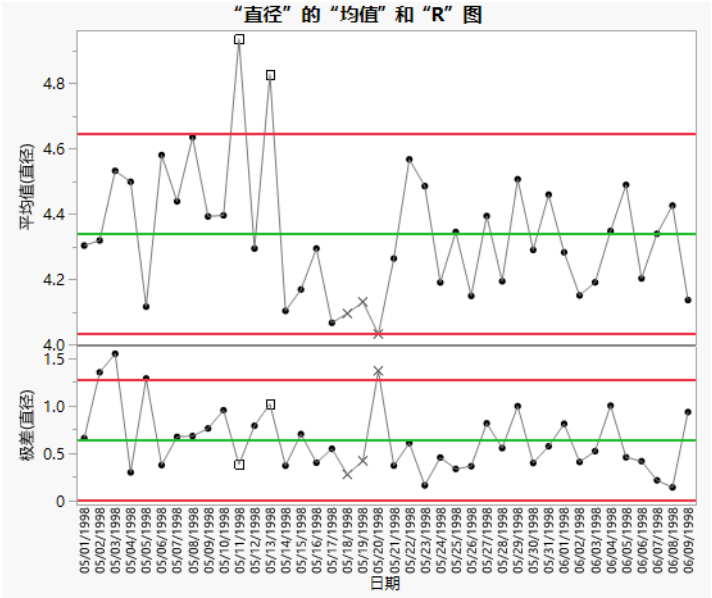
单个测量值和移动极差中位数图中的限值将移动极差中位数而不是平均移动极差用作 sigma。这导致**酸性**的控制限略窄。

均值图和 R 图阶段示例

某一医用导管生产商收集了用于新样机的导管直径数据。收集的这些数据来自过去 40 天的生产数据。在最初 20 天（阶段 1）后，公司对生产设备进行了一些调整。现在需要分析这些数据以确定过去 20 天（阶段 2）的生产是否处在受控状态中。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Diameter.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**直径**拖至 Y 角色。
4. 将**日期**拖至**子组**角色（位于底部）。

图 3.17 “直径”的控制图

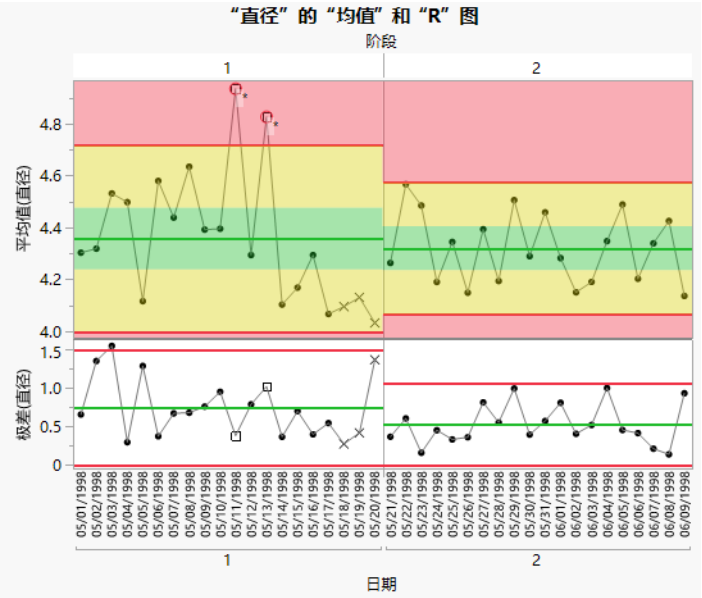


可以看出，前 20 天的变异性较高，在平均值中，有三个观测超出了控制限。为此，生产商对生产设备进行了调整，并且引入了新的控制限。

要为每个阶段计算单独的控制限，请执行以下步骤：

5. 将**阶段**拖至**阶段**角色。
6. 在平均图中，右击并选择**警告 > 检验超出界限**。
7. 在 “限值 [1]”分级显示项中，选择**区域着色**。

图 3.18 每个 “阶段” 的控制图



包括阶段变量意味着阶段 2 的控制限仅基于阶段 2 的数据。阶段 2 的所有观测都位于控制限内。这通过在图表上包含区域着色来突出显示。因此，您可以推断出在作出调整后过程处于受控状态。

依据另一个变量过滤控制图

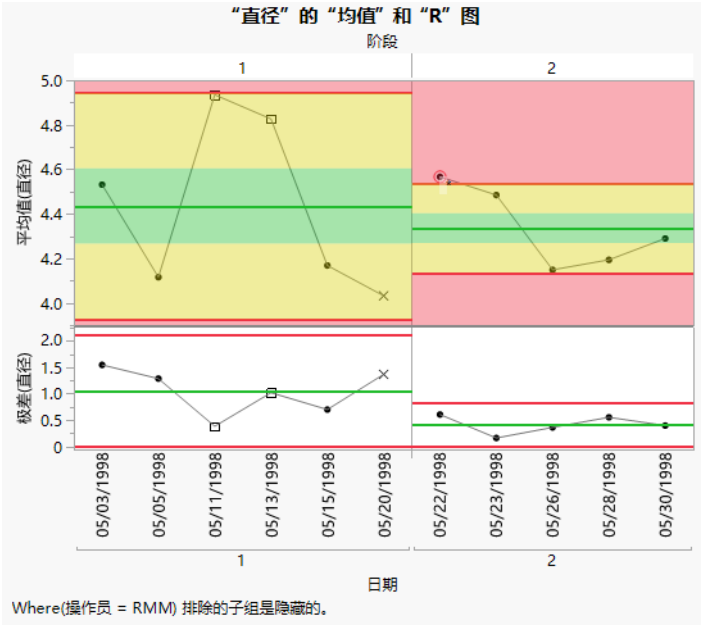
Diameter.jmp 这一数据表为每个样本的机器操作员都包含一列。您可以将 “本地数据过滤器” 与 “控制图生成器” 配合使用，以显示一部分操作员的数据。

8. 点击**控制图生成器**红色小三角并取消选择**显示排除区域**。

   禁用 “显示排除区域” 选项表示在 “本地数据过滤器” 中进行选择时，由 “本地数据过滤器” 中的设置排除的子组不再显示在控制图的水平轴上。因此，您只能看到关注的部分数据。

9. 点击**控制图生成器**红色小三角并选择**本地数据过滤器**。
10. 在 “本地数据过滤器” 中，先后点击**操作员**和 + 按钮。
11. 在 “本地数据过滤器” 中，选择标签为 RMM 的直条。

图 3.19 操作员 RMM 的 “直径”的均值图和 R 图



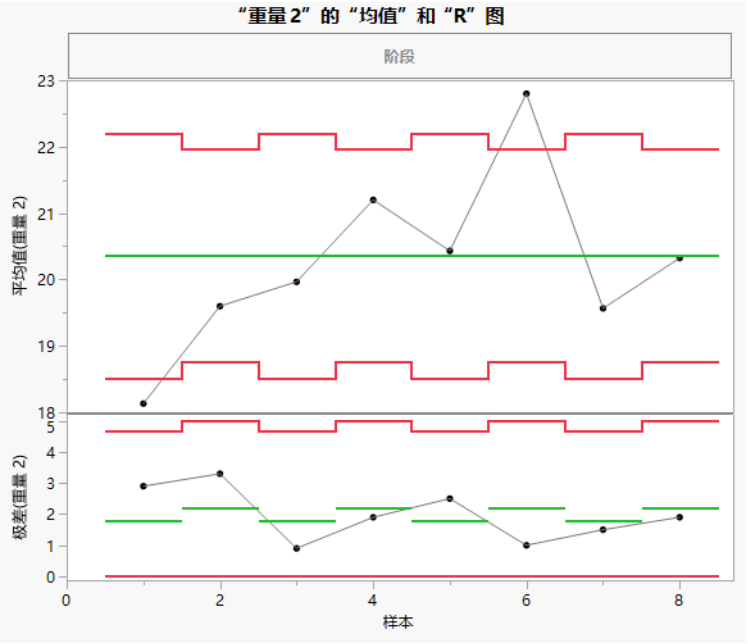
均值图和 R 图现在仅显示 RMM 操作员的点，如图表下方的 Where() 语句所示。对两个阶段的限值进行了调整，以反映其他三个操作员的观测已被排除。

具有子组大小变化的均值图和 S 图的示例

本例使用 Coating.jmp 数据表。这次关注的质量特征是**重量 2** 列。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Coating.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**重量 2** 拖至 Y 角色。
4. 将**样本**拖至**子组**角色 （位于底部）。

图 3.20 子组大小变化的均值图和 S 图

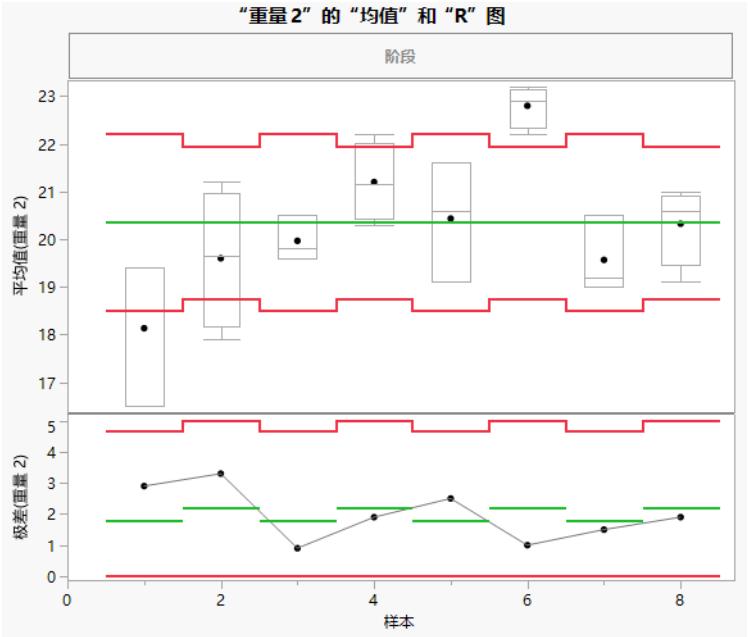


重量 2 的数据中有若干缺失值，所以图中有不规则的限值。尽管每个样本的观测数相同，但样本 1、 3、 5、 7 均包含缺失值。

您可以切换到查看每个样本处的箱线图，而不是查看连接每个样本平均值的线条。

5. 在**点 [1]** 分级显示项中，取消选择**显示已连接线条**选项。
6. 在**点 [1]** 分级显示项中，选择**箱线图**选项。

图 3.21 包含箱线图的均值和 S 图

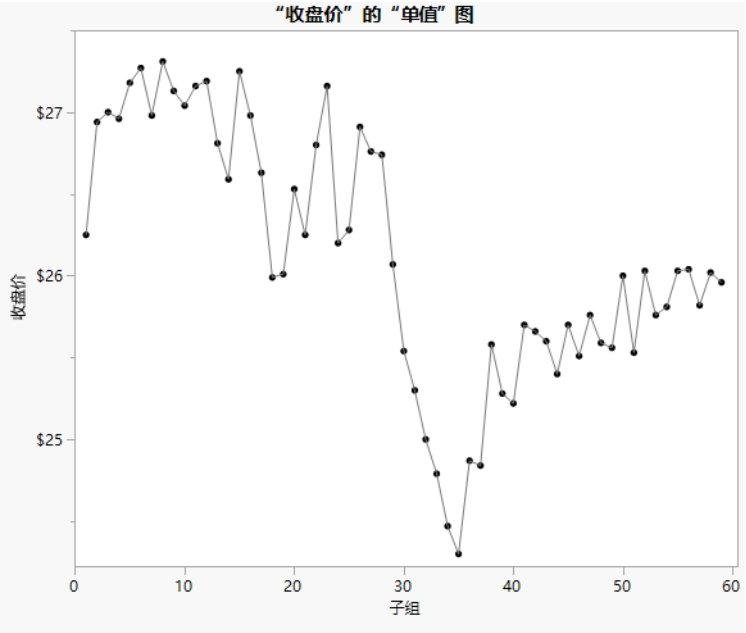


运行图示例

运行图将一列数据显示为一系列连接起来的点。本例显示 Quality Control 样本数据文件夹的 Stock Averages.jmp 中的**收盘价**变量的运行图。

1. 选择**帮助** > **样本数据库**，然后打开 Stock Averages.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **运行图**。
3. 选择**收盘价**并点击 Y。
4. 点击**确定**。

图 3.22 Stock Averages 数据表的收盘价运行图



P 图示例

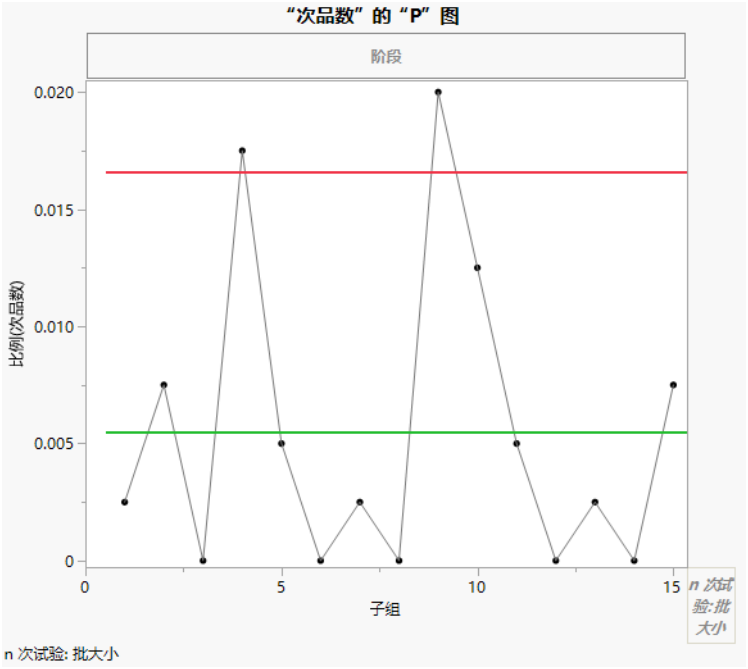
Washers.jmp 样本数据包含两个不同批大小的缺陷数据，这些数据取自 ASTM Manual on Presentation of Data and Control Chart Analysis，美国检验与材料协会 (1976)。要查看不变和可变样本大小之间的差异，您可以对比**批大小**和**批大小 2** 这两个图。

Quality Control 样本数据文件夹中的 Washers.jmp 数据包含每批次 400 个镀锌垫圈共计 15 个批次的缺陷计数。生产出的垫圈要经过检查，查看是否存在粗糙电镀和钢体裸露之类的镀层面缺陷。若垫圈存在镀层面缺陷，则认为其不合格或有缺陷。因此，缺陷计数表示每批次 400 个中有多少垫圈存在缺陷。使用 Washers.jmp 数据表，指定一个样本大小变量，该变量允许具有不同的样本大小。该数据表包含所有常数样本大小。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Washers.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **P 控制图**。
3. 选择**次品数**并点击 Y。
4. 选择**批大小**并点击 **n 次试验**。
5. 点击**确定**。
6. 在 “限值”分级显示项中，取消选择**显示下限**选项。

   这隐藏了在这种情况下不需关注的下限。

图 3.23 含样本大小的 “次品数” 的 P 图



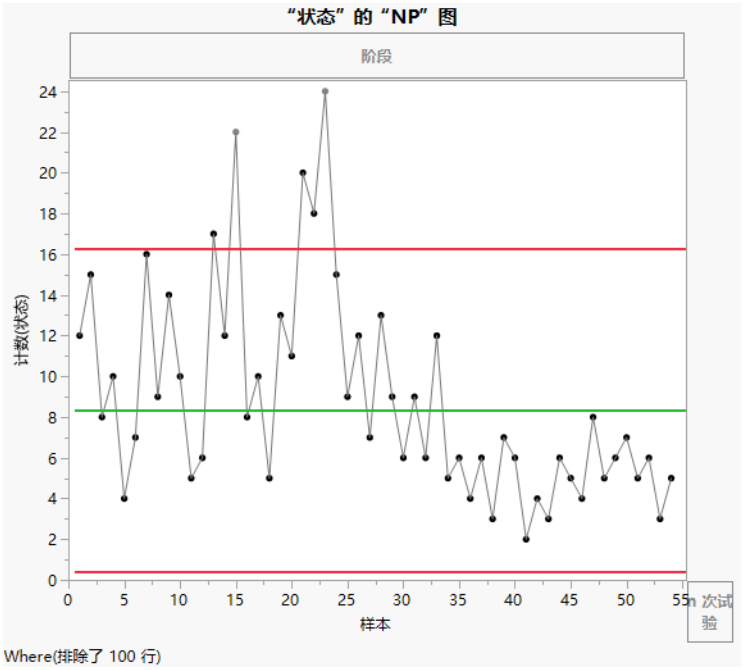
要查看不变和可变样本大小之间的差异，只需通过将变量拖至 “n 次试验” 区域，对**批大小**和**批大小 2** 这两个图进行比较即可。

NP 图示例

Bottle Tops.jmp 样本数据包含某一瓶盖生产过程的模拟数据。**样本**是每个瓶子的样本 ID 号。**状态**指示瓶盖是否符合设计标准。在**阶段**列中，第一阶段代表过程调整前的时间。第二阶段代表过程调整后的时间。过程中有关更改的注释也包括在内。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Bottle Tops.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**样本**拖至**子组**角色。
4. 将**状态**拖至 Y 角色。

图 3.24 “状态”（不合格）的 NP 图

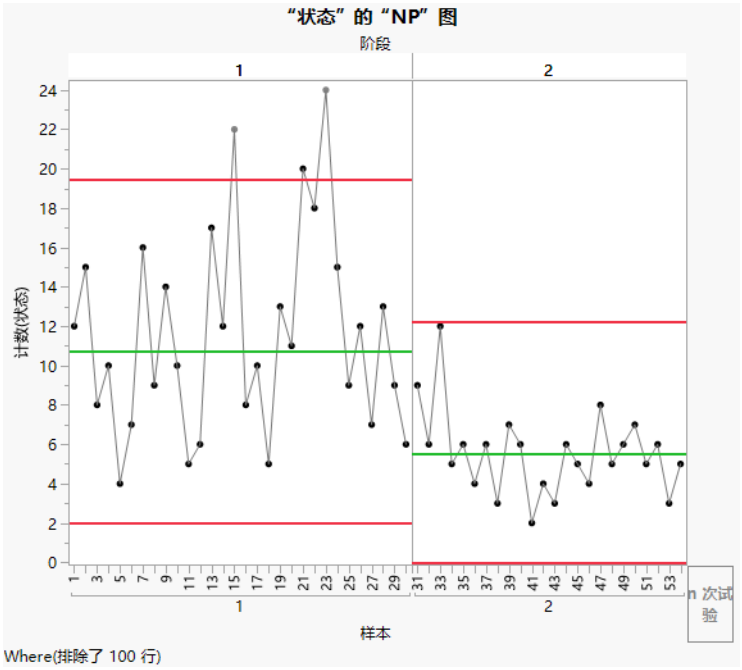


可以看出，原始观测有较高的变异性，共有五个观测（样本 13、15、21、22 和 23）位于上控制限之外。样本 15 和 23 的注释中指出过程中分别引入了新材料和新操作员。在该阶段末尾对生产设备进行了调整。因此，不应使用针对整个系列的控制限来评估阶段 2 中的控制。

要为每个阶段计算单独的控制限，请执行以下步骤：

5. 将**阶段**拖至**阶段**区域。

图 3.25 按 “阶段” 的 NP 图



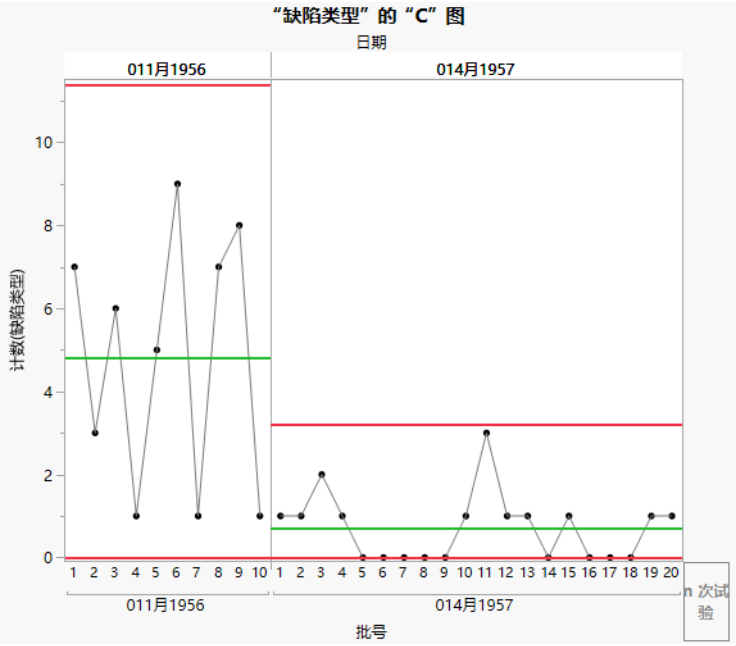
包括 “阶段” 变量意味着阶段 2 的控制限仅基于阶段 2 的数据。阶段 2 的所有观测都位于控制限内。因此，您可以推断出在作出调整后过程处于受控状态。

C 图示例

Cabinet Defects.jmp 样本数据表包含在经过两个时间期间的橱柜生产后所发现的各种缺陷的相关数据。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Cabinet Defects.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **C 控制图**。
3. 选择**缺陷类型**并点击 **Y**。
4. 选择**批号**并点击**子组**。
5. 选择**日期**并点击 **阶段**。
6. 点击**确定**。

图 3.26 带有阶段的 “缺陷类型” 的 C 图



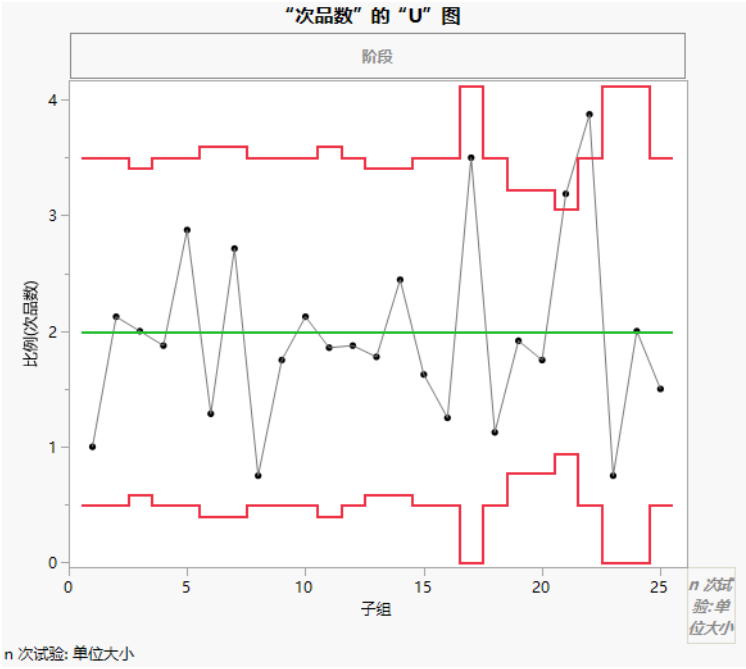
您现在可以查看两个不同日期的结果。这两个日期看来都未超出限值。要检查其他缺陷类型行为，请在**事件选择器 > 缺陷类型**下选择另一种缺陷类型，并在更新限值后查看结果。

U 图示例

Quality Control 样本数据文件夹中的 Braces.jmp 数据记录了盒装车用支架的缺陷计数。每盒支架都是一个检查单位。每天检查的盒数是子组样本大小，该大小是可变的。图 3.27 中的 U 图监控的是每个子组样本大小的支架缺陷数。上下边界根据检查的单位数而变化。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/Braces.jmp。
2. 选择**分析 > 质量和过程 > 控制图 > U 控制图**。
3. 选择**次品数**并点击 Y。
4. 选择**单位大小**并点击 n 次试验。
5. 点击**确定**。

图 3.27 “次品数”的 U 图



由于各子组间的样本大小不等，因此限值不规则。最后五个样品中有两个不在控制限内。

G 图示例

若要了解稀有事件出现的频率是否比期望要高，因此值得干预， G 图是一种有效的方式。请参见第 36 页的 “稀有事件控制图”。

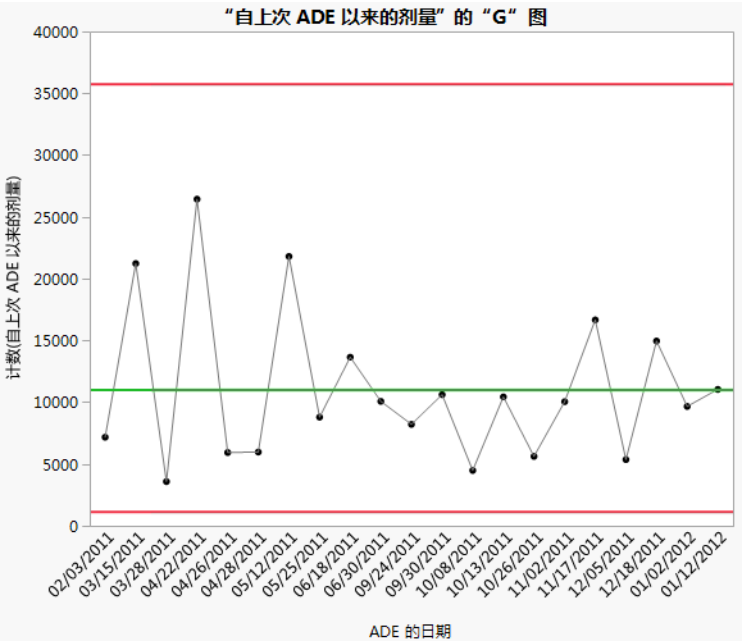
Adverse Reactions.jmp 样本数据表包含有关一组医院患者报告的药物不良事件 (ADE) 的模拟数据。ADE 是指患者服用药物后出现的任何类型的伤害或反应。反应日期和自上次反应以来的天数将被记录。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Adverse Reactions.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**自上次 ADE 以来的剂量**拖至 Y 角色。
4. 将 **ADE 的日期**拖至**子组**角色。

   “自上次 ADE 以来的剂量”的 “单值”和 “移动极差”图随即显示。
5. 在下拉列表中，选择**稀有事件**而不是 Shewhart 计量。

   随即显示 “自上次 ADE 以来的剂量”的 G 图，其中显示自上次事件以来服用药剂的次数。

图 3.28 “自上次 ADE 以来的剂量”的 G 图



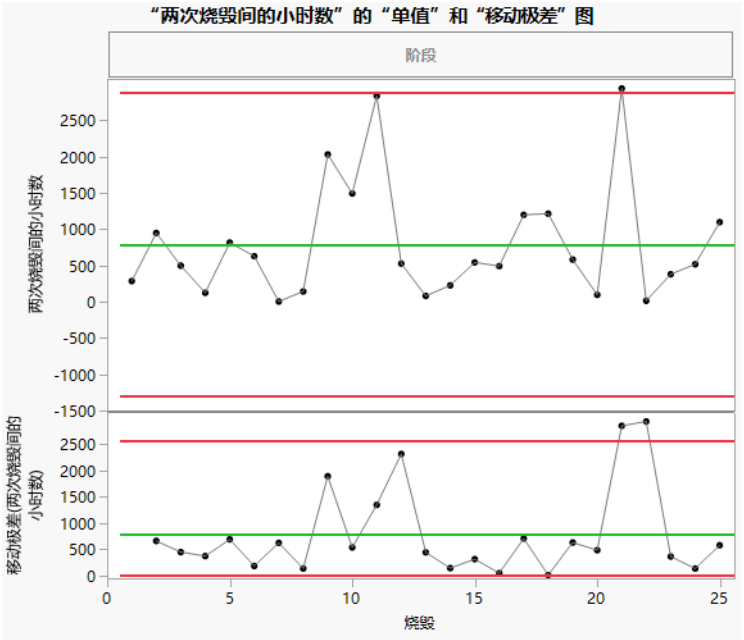
T 图示例

T 图用于测量自上次事件以来经过的时间。请参见第 36 页的 “稀有事件控制图”。

Fan Burnout.jmp 样本数据包含某一风扇生产过程的模拟数据。第一列表示烧毁的每个风扇。第二列表示两次烧毁间的小时数。

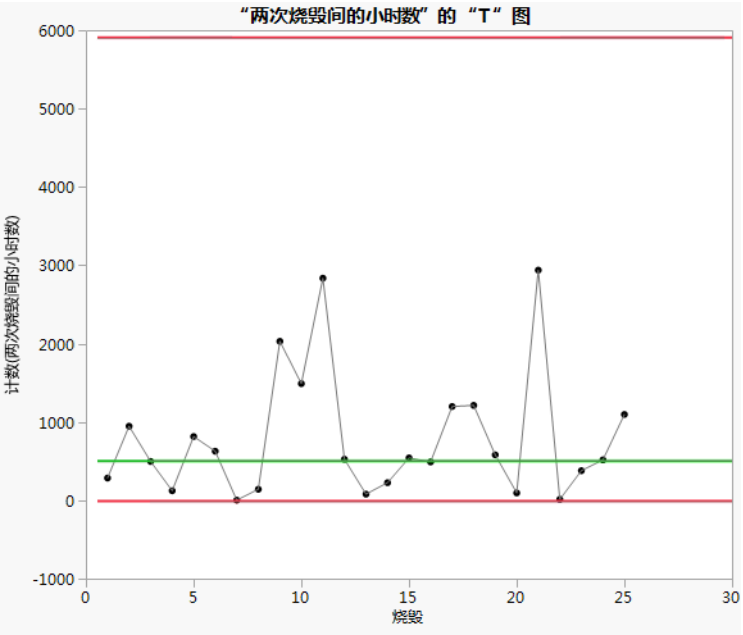
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Fan Burnout.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**两次烧毁间的小时数**拖至 Y 角色。
4. 将**烧毁**拖至**子组**角色。

图 3.29 “两次烧毁间的小时数”的“单值”和“移动极差”图



5. 在下拉列表中，选择**稀有事件**而不是 Shewhart 计量。
   “两次烧毁间的小时数”的 T 图随即显示。
6. 在 “限值”下，将 Sigma 从**负二项**改为 Weibull。

图 3.30 “两次烧毁间的小时数”的 T 图



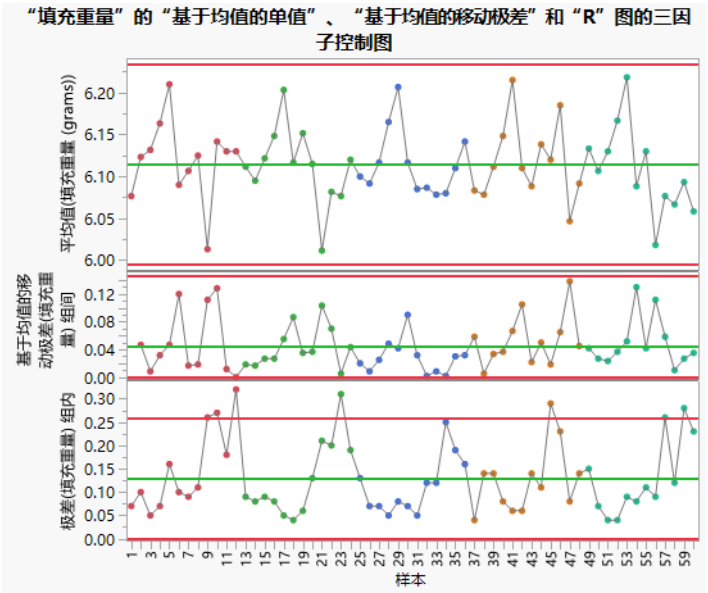
在 T 图中，所有点都显示在控制限内。很明显，“单值”和“移动极差”图对于分析不合适，因为限值太狭窄了。

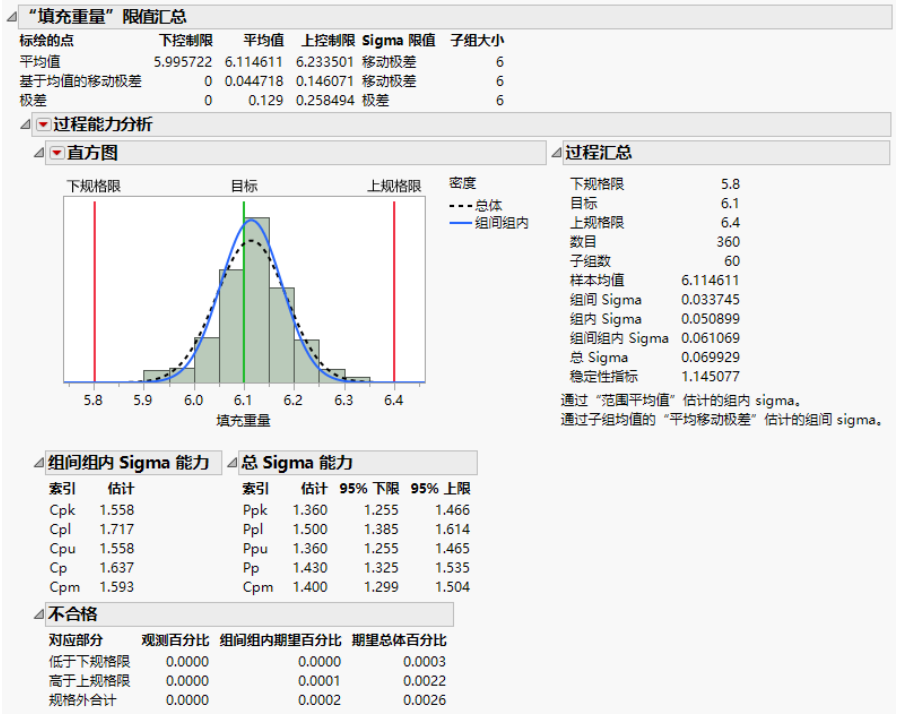
三因子控制图示例

当批次之间有变异且批次内也有变异时，三因子控制图很有用。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Vial Fill Weights.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **三因子控制图**。
3. 选择**填充重量**并点击 **Y**。
4. 选择**样本**并点击**子组**。
5. 点击**确定**。

图 3.31 “填充重量”的三因子控制图





在极差图与平均图之间出现了一个移动极差图。现在使用每个样本之间的移动极差计算平均值（均值）图中的限值。

“控制图生成器”的统计详细信息

注意：

- 控制限使用 sigma 乘数 K 计算，默认情况下，该乘数设置为 3。要更改 K 的默认值，请使用“文件”>“首选项”>“平台”>“控制图生成器”中的“KSigma”首选项。
- 对于最高 $n = 50$ 的样本大小，JMP 使用 Harter (1960) 的表 2 中定义的控制图常数 $d_2(n)$ 和 $d_3(n)$。对于样本大小大于 50 的样本，JMP 在计算 sigma 和控制限时使用样本大小 50 的控制图常数值。

- “均值图和 R 图的控制限”
- “均值图和 S 图的控制限”
- “单个测量值图和移动极差图的控制限”
- “P 图和 NP 图的控制限”
- “U 图和 C 图的控制限”
- “Levey-Jennings 图”
- “G 图控制限”
- “T 图控制限”
- “三因子控制图的 Sigma 计算”

均值图和 R 图的控制限

JMP 使用以下公式为均值图和 R 图生成控制限：

均值图的下控制限 = $\bar{X}_w - \frac{K\hat{\sigma}}{\sqrt{n_i}}$

均值图的上控制限 = $\bar{X}_w + \frac{K\hat{\sigma}}{\sqrt{n_i}}$

R 图的下控制限 = $\max\left(d_2(n_i)\hat{\sigma} - Kd_3(n_i)\hat{\sigma}, 0\right)$

R 图的上控制限 = $d_2(n_i)\hat{\sigma} + Kd_3(n_i)\hat{\sigma}$

R 图的中心线：默认情况下，第 i 个子组的中心线（其中 K 是 sigma 乘数），指示 R_i 的期望值的估计值。该值计算如下：$d_2(n_i)\hat{\sigma}$，其中 $\hat{\sigma}$ 是 σ 的估计值。

使用以下公式估计均值图和 R 图的标准差：

$$\hat{\sigma} = \frac{\frac{R_1}{d_2(n_1)} + \ldots + \frac{R_N}{d_2(n_N)}}{N}$$

其中：

$\bar{X}_w$ = 子组均值的加权平均值

K = sigma 乘数，且默认设置为 3

σ = 过程标准差

n_i = 第 i 个子组的样本大小

$d_2(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的期望值

$d_3(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的标准差

R_i 是第 i 个子组的极差

N 是 $n_i \geq 2$ 的子组数。

均值图和 S 图的控制限

JMP 使用以下公式为均值图和 S 图生成控制限：

均值图的下控制限 = $\bar{X}_w - \frac{K\hat{\sigma}}{\sqrt{n_i}}$

均值图的上控制限 = $\bar{X}_w + \frac{K\hat{\sigma}}{\sqrt{n_i}}$

S 图的下控制限 = $\max\left(c_4(n_i)\hat{\sigma} - Kc_5(n_i)\hat{\sigma}, 0\right)$

S 图的上控制限 = $c_4(n_i)\hat{\sigma} + Kc_5(n_i)\hat{\sigma}$

S 图的中心线：默认情况下，第 i 个子组的中心线 （其中 K 是 sigma 乘数），指示 s_i 的期望值的估计值。该值计算如下：$c_4(n_i)\hat{\sigma}$，其中 $\hat{\sigma}$ 是 σ 的估计值。

均值 /S 图中的标准差估计值为：

$$\hat{\sigma} = \frac{\frac{s_1}{c_4(n_1)} + \ldots + \frac{s_N}{c_4(n_N)}}{N}$$

其中：

$\overline{X}_w$ = 子组均值的加权平均值

K = sigma 乘数，且默认设置为 3

σ = 过程标准差

n_i = 第 i 个子组的样本大小

$c_4(n)$ 是服从单位标准差正态分布的 n 个自变量的标准差期望值

$c_5(n)$ 是 n 个服从标准差为 1 的独立、正态分布的自变量的标准差的标准误差

N 是 $n_i \geq 2$ 的子组数

s_i 是第 i 个子组的样本标准差

单个测量值图和移动极差图的控制限

单个测量值图的控制限计算如下：

单个测量值图的下控制限 $= \overline{X} - K\hat{\sigma}$

单个测量值图的上控制限 $= \overline{X} + K\hat{\sigma}$

单个测量值图（使用通过移动极差中位数估计得出的 sigma）的控制限计算如下：

单个测量值图的下控制限 $= \overline{X} - K\hat{\sigma}_{MMR}$

单个测量值图的上控制限 $= \overline{X} + K\hat{\sigma}_{MMR}$

移动极差图的控制限计算如下：

移动极差图的下控制限 $= \max\{d_2(n)\hat{\sigma} - Kd_3(n)\hat{\sigma}, 0\}$

移动极差图的上控制限 $= d_2(n)\hat{\sigma} + Kd_3(n)\hat{\sigma}$

中位数移动极差图的控制限计算如下：

下控制限 $_{MMR} = \max(0, MMR - kd_3(n)\hat{\sigma}_{MMR})$

上控制限 $_{MMR} = MMR + kd_3(n)\hat{\sigma}_{MMR}$

单个测量值图和移动极差图的标准差估计如下：

$$\hat{\sigma} = \frac{\overline{MR}}{d_2(n)}$$

使用中位数时的单个测量值图和移动极差图的标准差估计如下：

$\hat{\sigma}_{MMR}$ = MMR/0.954

其中：

$\overline{X}$ = 单个测量值的均值

K = sigma 乘数，且默认设置为 3

$\overline{MR}$ = 非缺失移动极差的均值，计算方式为 $(MR_2+MR_3+...+MR_N)/(N-1)$，其中 $MR_i = |x_i - x_{i-1}|$。

MMR = 非缺失移动极差的中位数

σ = 过程标准差

$d_2(n)$ = 服从单位标准差正态分布的 *n* 个自变量的极差的期望值

$d_3(n)$ = *n* 个服从标准差为 1 的独立、正态分布的自变量的极差的标准差

注意：“控制图生成器”中的移动极差图使用 *n*=2 的极差跨度。

P 图和 NP 图的控制限

分别使用以下公式计算下控制限 (LCL) 和上控制限 (UCL)：

P 图下控制限 = $\max(\bar{p} - K\sqrt{\bar{p}(1-\bar{p})/n_i}, 0)$

P 图上控制限 = $\min(\bar{p} + K\sqrt{\bar{p}(1-\bar{p})/n_i}, 1)$

NP 图下控制限 = $\max(n_i\bar{p} - K\sqrt{n_i\bar{p}(1-\bar{p})}, 0)$

NP 图上控制限 = $\min(n_i\bar{p} + K\sqrt{n_i\bar{p}(1-\bar{p})}, n_i)$

其中：

$\bar{p}$ 是从各个子组中抽取的不合格项的平均比例

$$\bar{p} = \frac{n_1 p_1 + \ldots + n_N p_N}{n_1 + \ldots + n_n} = \frac{X_1 + \ldots + X_N}{n_1 + \ldots + n_N}$$

n_i 是第 *i* 个子组中的项数

K 是 sigma 乘数，且默认设置为 3

U 图和 C 图的控制限

使用以下公式计算下控制限 (LCL) 和上控制限 (UCL)：

U 图下控制限 = $\max(\bar{u} - K\sqrt{\bar{u}/n_i}, 0)$

U 图上控制限 = $\bar{u} + K\sqrt{\bar{u}/n_i}$

C 图下控制限 = $\max(n_i\bar{u} - K\sqrt{n_i\bar{u}}, 0)$

C 图上控制限 = $n_i\bar{u} + K\sqrt{n_i\bar{u}}$

限值随 n_i 变化。

u_i 是第 i 个子组中的每单位不合格项数。通常， $u_i = c_i/n_i$。

K 是 sigma 乘数，且默认设置为 3

c_i 是第 i 个子组中的不合格项总数

n_i 是第 i 个子组中的检查单位数

$\bar{u}$ 是从各个子组中抽取的每单位不合格项的平均数。数量 $\bar{u}$ 计算为加权平均值

$$\bar{u} = \frac{n_1u_1 + \ldots + n_Nu_N}{n_1 + \ldots + n_N} = \frac{c_1 + \ldots + c_N}{n_1 + \ldots + n_N}$$

N 是子组数

Levey-Jennings 图

Levey-Jennings 图显示过程均值，以及基于长期 sigma 的控制限。控制限位于距离中心线 $K*s$ 远的位置，默认情况下，其中的 $K = 3$。

Levey-Jennings 图的标准差 s 计算方法与 “分布”平台中的标准差计算方法相同。

$$s = \sqrt{\sum_{i=1}^{N} \frac{(y_i - \bar{y})^2}{N-1}}$$

请参见 Levey and Jennings (1950)；Westgard (2002)。

G 图控制限

负二项分布是扩展的几何 (Poisson) 分布，允许相对于 Poisson 的过度离散。负二项分布可用于构建计数数据的精确和近似控制限。可基于负二项分布的卡方近似来获得近似控制限。所有数据都用作单个观测，而与子组大小无关。

假设 X 服从参数为 (μ, *k*) 的负二项分布，则：

$$P(X \le r) \sim P\left(\chi^2_v \le \frac{2r+1}{1+\mu k}\right)$$

其中：

χ^2_v 是使用 $v = 2\mu/(1+\mu k)$ 自由度的卡方变量。

基于以上近似计算，可确定近似的上控制限和下控制限。对于标定 α 水平的一个方向上的第一类错误概率，近似上控制限是上控制限的限值，以使以下等式为真：

$$P(X > \text{上控制限}) = 1 - P\left(\chi^2_v \le \frac{2\,\text{上控制限}+1}{1+\mu k}\right) = \alpha$$

同样，近似的下控制限 (LCL) 是一个使得以下等式为真的限值：

$$P(X < \text{下控制限}) = 1 - P\left(\chi^2_v \ge \frac{2\,\text{下控制限}+1}{1+\mu k}\right) = \alpha$$

因此，分别使用以下公式计算近似水平的下控制限 (LCL) 和上控制限 (UCL)：

$$\text{上控制限} = \frac{\chi^2_{v,\,1-\alpha}(1+\mu k)-1}{2}$$

$$\text{下控制限} = \max\left\{0, \frac{\chi^2_{v,\,\alpha}(1+\mu k)-1}{2}\right\}$$

其中：

$\chi^2_{v,\,1-\alpha}(\chi^2_{v,\,\alpha})$ 是使用 $v = 2\mu/(1+\mu k)$ 自由度的卡方分布的上 （下）百分位数。负下控制限可设置为零。

有关负二项控制限的详细信息，请参见 Hoffman (2003)。

T 图控制限

从数据中计算形状和尺度参数的估计值，并使用估计值来获取 Weibull 分布的百分位数。

注意：在估计 Weibull 分布参数时，响应值为 0 的子组被赋予权重 0。

定义以下数量：

对于正态分布 (0,1)， p1 = normalDist(-*K*)

对于正态分布 (0,1)， p2 = normalDist(0)

对于正态分布 (0,1)， p3 = normalDist(*K*)

然后使用以下公式计算限值：

下控制限 = Weibull 分位数 (p1, β, α)

控制限 = Weibull 分位数 (p2, β, α)

上控制限 = Weibull 分位数 (p3, β, α)

其中：

β 是 Weibull 分位数函数的形状参数， α 是其尺度参数。

K 是 sigma 乘数，且默认设置为 3

有关 Weibull 分位数函数的详细信息，请参见**帮助 > 脚本索引**。

三因子控制图的 Sigma 计算

基于极差平均值的组内 sigma

使用极差的平均值估计的三因子控制图的组内 sigma 估计值可用于 “基于均值的单值”、“基于均值的移动极差”和 R 图。

$$\hat{\sigma}_{\text{组内}} = \frac{\frac{R_1}{d_2(n_1)} + \ldots + \frac{R_N}{d_2(n_N)}}{N}$$

该公式使用以下符号：

R_i = 第 i 个子组的极差

n_i = 第 i 个子组的样本大小

$d_2(n_i)$ = 服从单位标准差正态分布的 n_i 个自变量的极差期望值

N = $n_i \geq 2$ 的子组数

基于无偏标准差的平均值的组内 sigma

使用无偏标准差估计的三因子控制图的组内 sigma 估计值可用于 “基于均值的单值”、“基于均值的移动极差”和 S 图。

$$\hat{\sigma}_{\text{组内}} = \frac{\frac{s_1}{c_4(n_1)} + \ldots + \frac{s_N}{c_4(n_N)}}{N}$$

该公式使用以下符号：

s_i = 第 i 个子组的样本标准差

n_i = 第 i 个子组的样本大小

$c_4(n_i)$ = 服从单位标准差正态分布的 n_i 个自变量的标准差期望值

$N = n_i \geq 2$ 的子组数

组间 Sigma

使用子组均值的移动极差估计的三因子控制图的组间 sigma 估计值。

$$\hat{\sigma}_{组间} = \sqrt{\left(\frac{\overline{MR}}{d_2(2)}\right)^2 - \frac{\hat{\sigma}^2_{组内}}{H}}$$

该公式使用以下符号：

$\overline{MR}$ = 非缺失移动极差的均值，计算方式为 $(MR_2+MR_3+...+MR_N)/(N-1)$，其中 $MR_i = |y_i - y_{i-1}|$。

$d_2(2)$ = 服从单位标准差正态分布的两个自变量的极差期望值。

$H = \dfrac{N}{\frac{1}{n_1} + \frac{1}{n_2} + \ldots + \frac{1}{n_N}}$，子组样本大小的调和均值。

注意：若组间 Sigma 估计为负值，则它设置为 0。

组间和组内 Sigma

使用组内 sigma 与组间 sigma 估计值的组合来估计三因子控制图的组间和组内 sigma 估计值。

$$\hat{\sigma}_{组间和组内} = \sqrt{\hat{\sigma}^2_{组内} + \hat{\sigma}^2_{组间}}$$

第 4 章

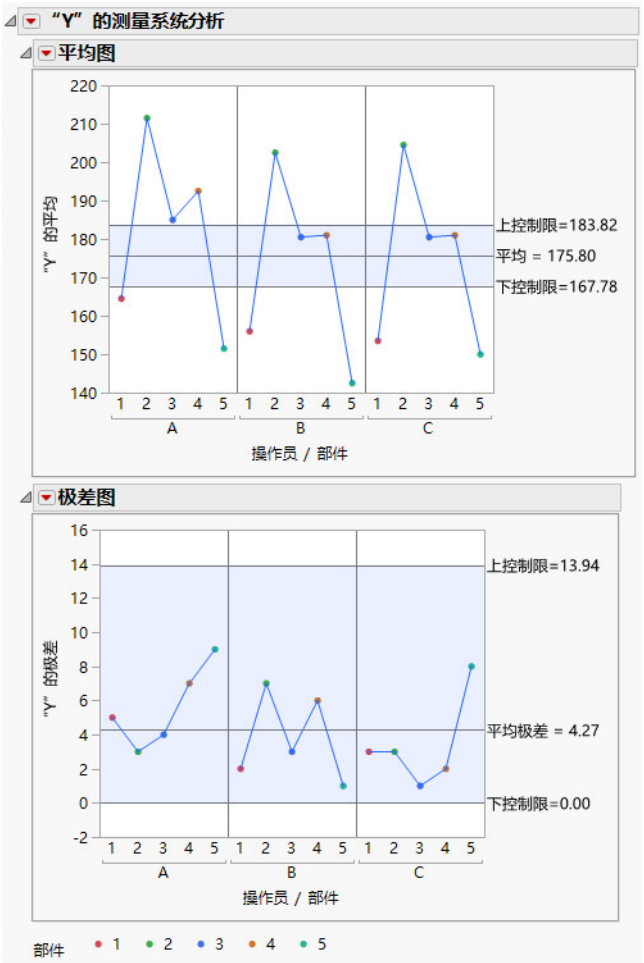
测量系统分析

使用 EMP 方法评估连续测量过程

“测量系统分析”(MSA) 平台评估测量系统的精确性、一致性和偏倚。在研究过程本身之前，您需要确保能够精准地测量过程。若您看到的大多数变异来自测量过程本身，您将无法可靠地了解过程。使用 MSA 可查明测量系统当前的性能。

本章介绍 EMP 方法。“量具 R&R”方法在第 115 页的“变异性量具图”一章中说明。

图 4.1 “测量系统分析”的示例



目录

“测量系统分析”概述 93
“测量系统分析”的示例 93
启动“测量系统分析”平台 96
数据格式 97
“测量系统分析”平台选项 97
平均图 99
极差图或标准差图 99
EMP 结果 100
有效分辨率 101
偏移检测刻画器 101
偏倚比较 105
复测误差比较 106
“测量系统分析”的更多示例 106
“测量系统分析”的统计详细信息 112
组间相关和概差的计算 112

“测量系统分析”概述

“测量系统分析”平台中的 EMP（评估测量过程）方法主要基于 Donald J. Wheeler 所著 EMP III Using Imperfect Data (2006) 一书中所述的方法。EMP 方法提供了易于解释的直观信息和结果，可帮助使测量系统发挥最大作用。

“量具 R&R”方法分析多少变异性来自操作员变异（再现性）和测量变异（重复性）。“量具 R&R”适用于许多交叉模型和嵌套模型的组合，不论模型是否平衡。请参见第 115 页的“变异性量具图”一章。

在 6 Sigma DMAIC 方法中，MSA（测量系统分析）解决测量阶段问题，过程行为图（或控制图）解决控制阶段问题。MSA 帮助您预测将来结果并刻画其特征。您可以使用从 MSA 收集的信息帮助解释和配置过程行为图。

有关控制图的详细信息，请参见第 29 页的“控制图生成器”。

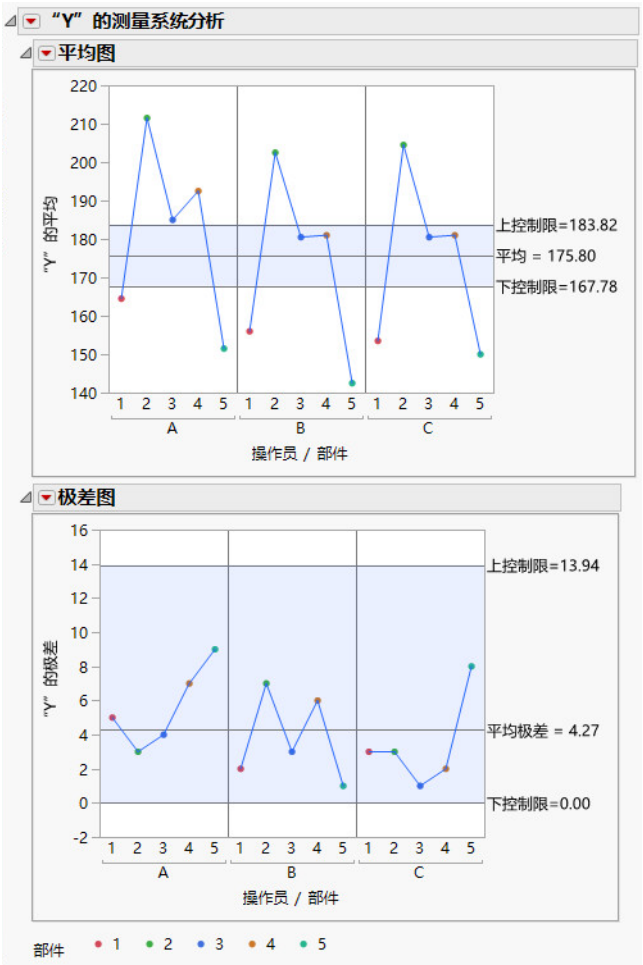
“测量系统分析”的示例

在该示例中，三名操作员测量了相同的五个部件。根据在测量中发现了多少变异，了解测量系统的运行状况。

1. 选择**帮助** > **样本数据库**，然后打开 Variability Data/Gasket.jmp。
2. 选择**分析** > **质量和过程** > **测量系统分析**。
3. 将 Y 分配给 **Y，响应**角色。
4. 将**部件**分配给**部件，样本 ID** 角色。
5. 将**操作员**分配给 **X，分组**角色。

   请注意， **MSA 方法**设置为 EMP，**散度图类型**设置为**极差**，**模型类型**设置为**交叉**。
6. 点击**确定**。

图 4.2 MSA 初始报表



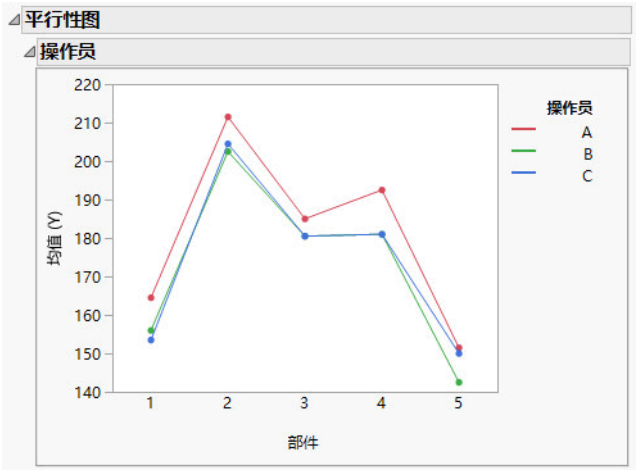
平均图显示每个操作员和部件组合的平均测量值。在该示例中，部件测量值的均值通常超过了控制限。这是想要的结果，因为它指示您可以检测部件间变异。

极差图显示每个操作员和部件组合的变异性。在该示例中，极差位于控制限内。这是想到的结果，因为它指示操作员采用相同的方式测量部件且具有相似的变异。

每个部件的颜色编码显示在图下方的图例中。

7. 点击 “Y 的测量系统分析”旁边的红色小三角并选择**平行性图**。

图 4.3 “操作员”和“部件”的平行性图



平行性图按操作员显示每个部件的平均测量值。因为线条通常是平行的，没有主要的交叉点，您可以推断出操作员和部件之间没有交互作用。

提示：交互作用指示存在需要进一步调查的严重问题。

8. 点击“Y 的测量系统分析”旁边的红色小三角并选择 **EMP 结果**。

图 4.4 “EMP 结果”报表

EMP 结果

EMP 检验	结果	说明
复测误差	3.7812	组内误差
自由度	13.39	用于估计组内误差的信息量
可能误差	2.5504	单次测量误差的中位数
组间相关（不含偏倚）	0.9745	归因于部件变异的变异比例（不带偏倚因子）
组间相关（带偏倚）	0.9437	归因于部件变异的变异比例（带偏倚因子）
偏倚影响	0.0308	偏移因子对组间相关性的降低量

系统	分类
当前（带偏倚）	一级
潜在（不含偏倚）	一级

监控等级图例

分类	组间相关	过程信号衰减	警告的概率，仅检验 1*	警告的概率，检验 1-4*
一级	0.80 - 1.00	小于 11%	0.99 - 1.00	1.00
二级	0.50 - 0.80	11% - 29%	0.88 - 0.99	1.00
三级	0.20 - 0.50	29% - 55%	0.40 - 0.88	0.92 - 1.00
四级	0.00 - 0.20	大于 55%	0.03 - 0.40	0.08 - 0.92

* 使用 Wheeler 检验在 10 个子组内的 3 标准误差偏移的警告概率，对应于 Nelson 检验 1、2、5 和 6。

“EMP 结果”报表计算若干统计量，可帮助您对测量系统进行评估和分类。“组间相关”指示可以归因于部件的总变异的比例。

从“EMP 结果”报表，您可以得出以下结论：

- “组间相关”值接近 1，这说明大多数变异来自部件而非测量系统。
- 分类为“一级”，这表示过程信号强度削弱了不到 11%。
- 仅使用检验 1 时至少有 99% 的机会检测到警告。
- 使用检验 1-4 时有 100% 的机会检测到警告。

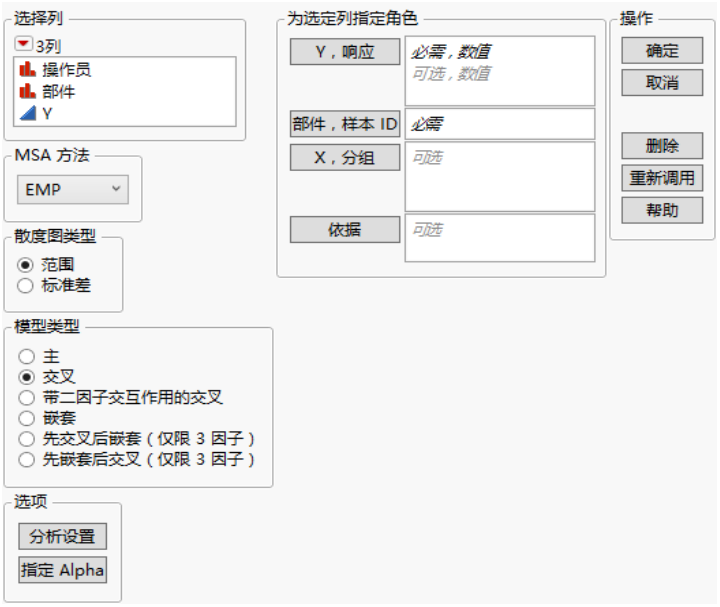
注意：有关检验和检测过程偏移的详细信息，请参见第 101 页的“偏移检测刻画器”。

操作员和部件之间没有交互作用，您的测量值有很小的变异（分类为“一级”）。因此，您可以推断出测量系统工作得很好。

启动“测量系统分析”平台

通过选择**分析 > 质量和过程 > 测量系统分析**启动“测量系统分析”平台。

图 4.5 “测量系统分析”窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“测量系统分析”窗口包含以下功能：

选择列 列出当前数据表中的所有变量。将所选列移到一个角色。

MSA 方法 选择要使用的方法：EMP（评估测量过程）或**量具 R&R**。本章介绍 EMP 方法。有关**量具 R&R** 方法的详细信息，请参见第 115 页的“变异性量具图”一章。

散度图类型 指定显示变异的图表类型。选择**极差**选项或**标准差**选项。

> **注意：**对于 EMP 方法，散度图类型确定如何计算 “EMP 结果”报表中的统计量。若选择**极差**选项，且您有一个因子或两个因子的平衡交叉模型，则该报表中的统计量基于极差。否则，该报表中的统计量基于标准差。

模型类型 指定模型类型：

主 将具有名义型或有序型建模类型的变量视为主效应。

交叉 当每个因子的每个水平与所有其他因子的每个水平同时出现时模型交叉。

带二因子交互作用的交叉 当二因子的每个水平与其他因子的每个水平同时出现时模型交叉。

嵌套 当一个因子的所有水平仅在任何其他因子的单个水平内出现时模型嵌套。

先交叉后嵌套（仅限 3 因子） 因子先交叉然后嵌套（仅限 3 因子）。

先嵌套后交叉（仅限 3 因子） 因子先嵌套然后交叉（仅限 3 因子）。

选项 包含以下选项：

分析设置 设置 REML 最多迭代次数和收敛。

指定 Alpha 指定 1-alpha 置信水平。

Y，响应 测量值的列。

部件，样本 ID 指定部件或单元的列。

X，分组 表示分组变量的列。

依据 标识一列，该列创建的报表包含变量每个水平的单独分析。

数据格式

要使用“测量系统分析”平台，所有响应测量必须位于单个响应列中。有时，响应记录在多个列中，其中每行都是某设计因子的一个水平，每列都是另一个设计因子的一个水平。必须在运行“测量系统分析”平台之前堆叠该格式的数据。请参见《使用 JMP》。

“测量系统分析”平台选项

平台选项显示在 “测量系统分析”旁边的红色小三角菜单中。选择一个选项可在 MSA 报表窗口中创建相应的图形或报表。取消选择一个选项将删除该图形或报表。从以下选项中选择：

平均图 部件和 X 变量的每个组合的平均测量值图。平均图帮助您检测产品变异 （不考虑测量变异）。在平均图中，失控数据是想要的，因为它检测部件间变异。请参见第 99 页的 “平均图”。

极差图 部件和 X 变量的每个组合的变异性统计量图。仅当您在启动窗口中选择了**极差**作为散度图类型时才显示它。极差图帮助您检查子组内的一致性。在极差图中，限值内的数据是想要的，这指示误差的齐性。请参见第 99 页的 “极差图或标准差图”。

标准差图 部件和 X 变量的每个组合的标准差统计量图。仅当您在启动窗口中选择了**标准差**作为散度图类型时才显示它。标准差图帮助您检查子组内的一致性。在标准差图中，限值内的数据是想要的，这指示误差的齐性。请参见第 99 页的 “极差图或标准差图”。

平行性图 反映每个部件的平均测量值的叠加图。若线条相对不平行或交叉，则部件和 X 变量之间可能有交互作用。

提示：交互作用指示存在需要进一步调查的严重问题。例如，部件和操作员之间的交互作用表示操作员平均来说在以不同方式测量不同的部件。因此，测量变异性无法预测。这个问题需要进一步调查，以了解操作员对于各部件不具有相同模式或刻画的原因。

EMP 结果 计算若干统计量以帮助您对测量系统进行评估和分类的报表。请参见第 100 页的 “EMP 结果”。

有效分辨率 包含测量系统分辨率结果的报表。请参见第 101 页的 “有效分辨率”。

偏倚比较 一个均值分析图，用于检验 X 变量是否有不同的平均值。请参见第 105 页的 “偏倚比较”。

复测误差比较 一个方差均值分析图或极差均值分析图，用于检验任何组是否具有不同的复测误差水平。请参见第 106 页的 “复测误差比较”。

偏移检测刻画器 一组交互式图，您可以调整它们来查看在过程行为图中得到警告的概率。请参见第 101 页的 “偏移检测刻画器”。

方差分量 一个包含指定模型的方差分量估计值的报表。该报表中的计算基于方差而不是极差。平衡数据使用 EMS 方法。非平衡数据使用 REML 方法。

注意：该报表类似于 “变异性图”平台中的 “方差分量”报表，除了它不计算 Bayesian 方差分量估计值之外。请参见 “变异性量具图”一章中第 123 页的 “方差分量”。

EMP 量具 R&R 结果 一个将测量值中的变异性分割为部件变异和测量系统变异的报表。该报表中的计算基于方差而不是极差。因为负方差分量设置为零，所以零值可能表示结果中有离群值。

注意：该报表类似于 “变异性图”平台中的 “量具 R&R”报表。但是，默认情况下，“再现性”的计算不包括交互作用。要指定交互作用包含在 “再现性”计算中，请选择 “在再现性中包含交互作用”平台首选项。该首选项位于 “文件”>“首选项”>“平台”>“EMP 测量系统分析”中。有关量具 R&R 研究的详细信息，请参见 “变异性量具图”一章中第 125 页的 “关于 “量具 R&R 方法””。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

平均图

“平均图”旁边的红色小三角菜单包含以下选项：

显示总均值 在图上绘制 *Y* 变量的总均值。

显示连接均值 绘制连接所有平均测量值的线条。

显示控制限 绘制表示上控制限 (UCL) 和下控制限 (LCL) 的线条并为这些值添加标签。“平均图”的控制限使用与均值控制图相同的计算。请参见 “控制图生成器”一章中第 82 页的“均值图和 R 图的控制限”。

显示控制限着色 在上控制限和下控制限之间添加着色。

显示分隔线 绘制在 *X* 变量之间描绘的垂直线。

显示数据 向图中添加数据点。

注意：可以使用以下两种方法之一替换平均图中的变量：通过将变量从一个轴拖放到另一个轴来交换现有变量，或点击相关数据表的“列”面板中的某个变量并将其拖到一个轴上。

极差图或标准差图

“极差图”或“标准差图”旁边的红色小三角菜单包含以下选项：

显示平均散度 在图上绘制平均极差或标准差。

显示连接的点 绘制连接所有极差或标准差的线条。

显示控制限 绘制表示上控制限 (UCL) 和下控制限 (LCL) 的线条并为这些值添加标签。有关“极差图”中使用的限值计算的详细信息，请参见 “控制图生成器”一章中第 82 页的“均值图和 R 图的控制限”。有关“标准差图”中使用的限值计算的详细信息，请参见 “控制图生成器”一章中第 83 页的“均值图和 S 图的控制限”。

显示控制限着色 在上控制限和下控制限之间添加着色。

显示分隔线 绘制在 *X* 变量之间描绘的垂直线。

注意：可以使用以下两种方法之一替换极差图或标准差图中的变量：通过将变量从一个轴拖放到另一个轴来交换现有变量，或点击相关数据表的“列”面板中的某个变量并将其拖到一个轴上。

EMP 结果

注意：在此情况下，该报表中的统计量基于极差：若您选择了 EMP 作为 MSA 方法以及**极差**作为**散度图类型**，并且具有一个因子或两个因子的平衡交叉模型。否则，该报表中的统计量基于方差。

“EMP 结果”报表计算若干统计量，可帮助您对测量系统进行评估和分类。使用该报表，您可以确定：

- 如何影响过程图。
- 设置哪些检验。
- 过程信号有多大程度的减弱。
- 偏倚因子在多大程度上影响您的系统和减小可能的组间相关系数。

“EMP 结果”报表包含以下计算：

复测误差 指示测量变异或重复性 （也称为组内误差或纯误差）。

自由度 指示用于估计组内误差的信息量。

概差 单个测量值的误差中位数。指示您的测量值的分辨率质量，帮助您确定在记录测量值时使用多少位。请参见第 101 页的 “有效分辨率”。

组间相关 指定您可以归因于部件的总变异的比例。若测量变异很小，则该数字接近 1。

组间相关（不含偏倚） 在计算结果时不考虑偏倚或交互作用因子。

组间相关（带偏倚） 在计算结果时考虑偏倚因子（如操作员、仪器等）。

组间相关（带偏倚和交互作用） 在计算结果时考虑偏倚和交互作用因子。仅当模型交叉并使用标准差而不是极差的情况下，才显示该计算。

偏倚影响 偏倚因子导致的组间相关减小量。

偏倚和交互作用影响 偏倚和交互作用因子导致的组间相关减小量。仅当模型交叉并使用标准差而不是极差的情况下，才显示该计算。

过程监控等级

要了解系统和等级参数，您必须首先了解 “监控等级图例”。

图 4.6 监控等级图例

监控等级图例

分类	组间相关	过程信号衰减	警告的概率，仅检验 1*	警告的概率，检验 1-4*
一级	0.80 - 1.00	小于 11%	0.99 - 1.00	1.00
二级	0.50 - 0.80	11% - 29%	0.88 - 0.99	1.00
三级	0.20 - 0.50	29% - 55%	0.40 - 0.88	0.92 - 1.00
四级	0.00 - 0.20	大于 55%	0.03 - 0.40	0.08 - 0.92

* 使用 Wheeler 检验在 10 个子组内的 3 标准误差偏移的警告概率，对应于 Nelson 检验 1、2、5 和 6。

该图例描述以下类别：一级、二级、三级和四级。每个等级指示以下信息：

- 相应的组间相关值
- 过程信号衰减（减小）量
- 使用 Wheeler 的检验 1 或所有四个检验在 10 个子组内检测到 3 标准误差偏移的机会

Wheeler (2006) 确认采用了四种检测检验，称为 Western Electric 区域检验。在“偏移检测刻画器”内，有八个检验可供您选择。对应于 Wheeler 检验的检验为第一个、第二个、第五个和第六个检验。

提示：要阻止出现图例，请在“EMP 测量系统分析”平台首选项中取消选择**显示监控等级图例**。

有效分辨率

“有效分辨率”报表帮助您确定测量值增量是否合适。记录测量值时，您可能需要增加或减小位数，也可能当前增量是合适的。请注意以下事项：

- “概差”计算测量值的误差中位数。
- “当前测量值增量”反映当前将数据近似到多少位，以 10 的最近次幂形式取自数据。将该数字与“最小有效增量”、“增量下限”和“最大有效增量”比较。根据比较结果给出建议。
- 大的测量值增量在最后一位的不确定性较小，但是误差中位数大。小的测量值增量的误差中位数小，但是在最后一位的不确定性更大。

偏移检测刻画器

使用“偏移检测刻画器”评估用于监控过程的控制图的灵敏度。“偏移检测刻画器”估计检测到产品均值或产品标准差中的偏移的概率。控制图限值提供了测量值误差变异的来源。根据这

些限值，“偏移检测刻画器”估计警告概率。这是监控过程均值的控制图对下 *k* 个子组发出警告信号的概率。

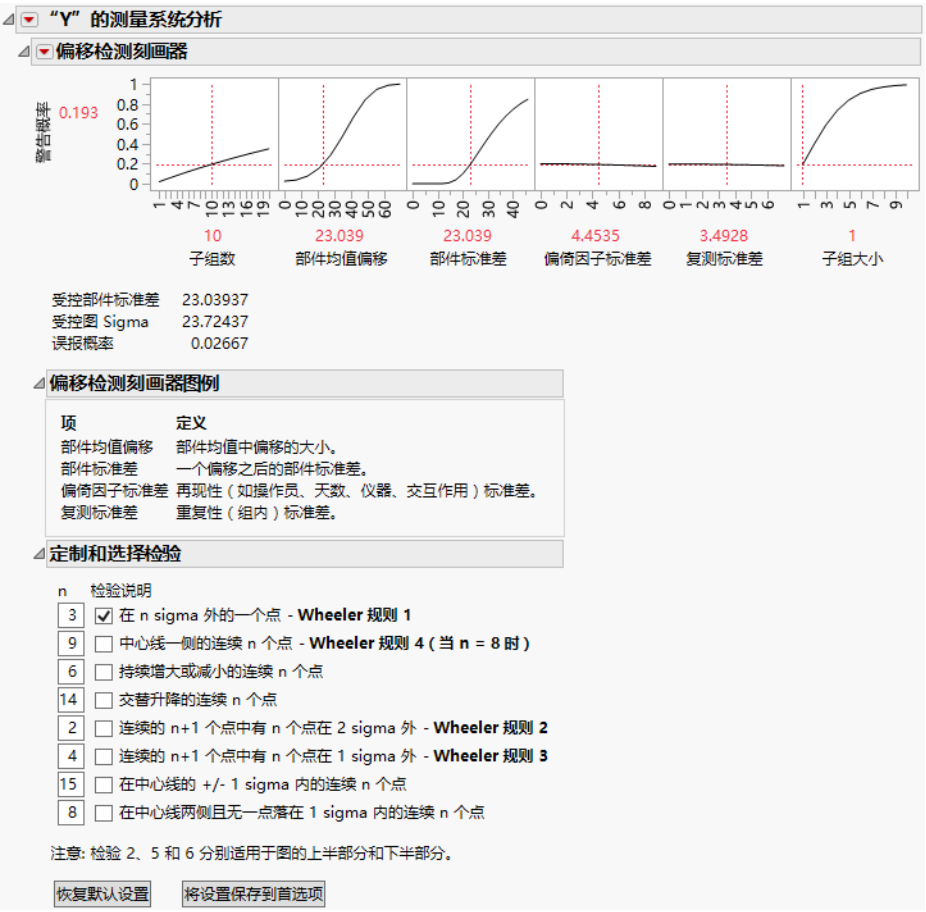
您可以设置要用于控制图的子组大小。请注意以下事项：

- 若“子组大小”等于 1，则控制图是单个测量值图。
- 若“子组大小”超过 1，则控制图为均值图。

您可以探索“子组大小”对控制图灵敏度的影响。还可以探索减小偏倚和复测误差的好处。

图 4.7 显示 Variability Data 文件夹中的 Gasket.jmp 样本数据表的“偏移检测刻画器”报表。

图 4.7 Gasket.jmp 对应的 “偏移检测刻画器”



警告概率

“警告概率”是在过程中检测到变化的概率。变化由“偏移检测刻画器”中的“部件均值偏移”和“部件标准差”设置定义。概率计算假定在“定制和选择检验”分级显示框中选择的检验会应用到刻画器中指定的“子组数”。

单个测量值图（子组大小 = 1）和均值图（子组大小 > 1）的控制限基于“受控图 Sigma”。“受控图 Sigma”考虑偏倚因子（再现性）变异和复测（重复性）变异。这些值初始设置为从 MSA 研究获取的值。“受控图 Sigma”还包含“受控部件标准差”。这两个值与基于“受控图 Sigma”的误报概率一起显示在刻画器下面。

受控部件标准差 稳定过程的真实部件值的标准差（排除了测量值误差）。“受控部件标准差”的默认值是 MSA 分析估计的部件分量的标准差，它显示在 “方差分量”报表中。

通常，进行 MSA 研究的部件选择为具有特定属性，并且不一定反映在生产中看到的部件间变异。因此，您可以通过从“偏移检测刻画器”红色小三角菜单选择**更改受控部件标准差**，指定受控部件标准差。

受控图 Sigma 用于计算控制限的 sigma 值。使用在 “偏移检测刻画器”中指定的 “受控部件标准差”、“偏倚因子标准差”和 “复测标准差”以及 “子组大小”来计算该值。假定再现性因子在子组中为常数。

对于大小为 n 的子组，控制限设置为以下值：

$\pm 3(\text{受控图 Sigma})/(\sqrt{n})$

这里的 “受控图 Sigma”是以下项平方和的平方根：

- 受控部件标准差
- “偏移检测刻画器”中指定的“偏倚因子标准差”乘以 $\sqrt{n}$
- “偏移检测刻画器”中指定的“复测标准差”

用 “偏倚因子标准差”乘以 $\sqrt{n}$ 来解释再现性因子在子组内是常数的假定。

当您更改 “受控部件标准差”、“偏倚因子标准差”、“复测标准差”或 “子组大小”时，JMP 更新 “受控图 Sigma”。

误报概率 当部件均值或标准差没有变化时控制图检验发出警告信号的概率。当您更改 “子组数”或 “定制和选择检验”中的检验时， JMP 会更新 “误报概率”。

有关“方差分量”报表的详细信息，请参见“变异性量具图”一章中第 123 页的“方差分量”。

“偏移检测刻画器”设置

子组数 计算警告概率所依据的子组数。若子组数设置为 k，刻画器基于这 k 个子组给出控制图至少发出一次警告信号的概率。默认情况下，“子组数”设置为 10。拖动图中的垂直线可更改 “子组数”。

部件均值偏移 部件均值的偏移。默认情况下，刻画器设置为检测 1 sigma 偏移。初始值是 MSA 分析估计的部件分量的标准差，可在 “方差分量”报表中找到。拖动图中的垂直线或点击图下方的值可更改 “部件均值偏移”。

部件标准差 真实部件值的标准差（排除了测量值误差）。“部件标准差”的初始值为 MSA 分析估计的部件分量的标准差，可在“方差分量”报表中找到。拖动图中的垂直线或点击图下方的值可更改 “部件标准差”。

偏倚因子标准差 与再现性有关的因子的标准差。偏倚因子包括操作员和仪器因素。偏倚因子变异不包括部件和重复性（组内）变异。初始值由 MSA 分析估计的再现性和交互作用方差分量推导得出，可在“方差分量”报表中找到。拖动图中的垂直线或点击图下方的值可更改“偏倚因子标准差”。

复测标准差 模型中复测（或重复性）变异的标准差。初始值是 MSA 分析估计的“组内”分量的标准差，可在“方差分量”报表中找到。拖动图中的垂直线或点击图下方的值可更改“复测标准差”。

子组大小 用于每个子组的样本大小。默认情况下它设置为 1。您可以增加样本大小来改进控制图的性能。从 1 开始增加样本大小可演示从单个测量值图到均值图的变化中发生了什么。拖动图中的垂直线可更改“子组大小”。

“偏移检测刻画器”选项

“偏移检测刻画器”的红色小三角菜单提供了几个选项。这里只介绍一个选项。

更改受控部件标准差 指定稳定过程的部件标准差的值。受控部件标准差应反映真实部件值的变异（排除了测量值误差）。输入一个新值，然后点击“确定”。

“受控部件标准差”最初设置为 MSA 分析估计的部件分量的标准差，可在“方差分量”报表中找到。

若选择用于 EMP 研究的部件不是来自过程的随机样本，该选项很有用。

重置因子网格 为每个因子显示一个窗口，允许您为因子的当前设置输入特定值，锁定该设置，以及控制网格的各个方面。请参见《刻画器指南》。

因子设置 包含以下选项的子菜单：

记住设置 将一个分级显示项节点添加到报表，每当调用“记住设置”命令时，该报表累计当前设置的值。每个记住的设置前面有一个单选按钮，用于重置这些设置。“记住设置”红色三角形菜单中提供用于删除选定设置或所有设置的选项。

复制设置脚本 将当前刻画器的设置复制到剪贴板。

粘贴设置脚本 将刻画器的设置从剪贴板粘贴到另一报表中的刻画器。

设置脚本 设置一个脚本，每当某个因子变化时调用该脚本。设置脚本接收表单的变量列表：

```
{factor1 = n1, factor2 = n2, ...}
```

例如，要将该列表写入日志，请首先定义一个函数：

```
ProfileCallbackLog = Function({arg},show(arg));
```

然后在“设置脚本”对话框中输入 ProfileCallbackLog。

类似函数将因子值转换为全局值：

```
ProfileCallbackAssign = Function({arg},evalList(arg));
```

或一次访问一个值：

```
 ProfileCallbackAccess =
Function({arg},f1=arg["factor1"];f2=arg["factor2"]);
```

偏移检测刻画器图例

该面板简要介绍了四个“偏移检测刻画器”设置。请参见第 103 页的““偏移检测刻画器”设置”。

提示：要阻止显示图例，请在“EMP 测量系统分析”平台首选项中取消选择**显示偏移检测刻画器图例**。

定制和选择检验

在“定制和选择检验”面板中，选择和定制要应用到控制图中 *k* 个子组的检验。这八个检验基于 Nelson (1984)。有关检验的详细信息，请参见“控制图生成器”一章中第 50 页的“检验”。

“偏移检测刻画器”计算将这些检验考虑在内。添加更多检验时，“警告概率”和“误报概率”值会增大。因为这些计算基于拟随机模拟，当更新刻画器时，可能有一点延迟。

“定制和选择检验”面板包含以下选项：

恢复默认设置 若没有设置保存到首选项，该选项仅将所选检验重置为第一个检验。*n* 的值也重置为“控制图生成器”一章中第 50 页的“检验”中所述的值。若设置已保存到首选项，该选项将所选检验和 *n* 的值重置为首选项中指定的值。

注意：您可以通过选择**文件 > 首选项 > 平台 > 控制图生成器**访问控制图检验的首选项。定制检验 1-8 对应于 “定制和选择检验”中所示的 8 个检验。

将设置保存到首选项 保存所选检验和 *n* 的值以在将来的分析中使用。这些首选项会添加到“控制图生成器”平台首选项。

偏倚比较

偏倚比较选项创建均值分析图。该图显示分组变量的每个水平的均值并将它们与总均值进行比较。您可以使用该图查看操作员平均来说测量部件得到的值是否太高或太低。

“均值分析”旁边的红色小三角菜单包含以下选项：

设置 Alpha 水平 从最常用的 alpha 水平中选择一个选项，或使用**其他**选项指定任意水平。更改 alpha 水平会修改上决策限和下决策限。

显示汇总报表 显示一个包含组均值和决策限的报表，并说明组均值是否大于上决策限或小于下决策限。

显示选项 包括以下选项：

显示决策限 绘制表示上决策限 (UDL) 和下决策限 (LDL) 的线条并定义这些值。

显示决策限着色 在 UDL 和 LDL 之间添加着色。

显示中心线 绘制表示平均值的中心线统计量。

点选项 将图显示方式更改为针、连接的点或点。

复测误差比较

复测误差比较选项创建方差均值分析图或极差均值分析图。该图显示操作员之间的复测误差是否存在差异。例如，您可以使用该图查看每个操作员的测量方法是否存在不一致。当极差用于方差分量时显示极差均值分析图。

- 有关“操作员方差检验”旁边的红色小三角菜单中的选项信息，请参见第 105 页的“偏倚比较”。
- 有关方差均值分析图的详细信息，请参见“变异性量具图”一章中第 123 页的“方差分量”。

“测量系统分析”的更多示例

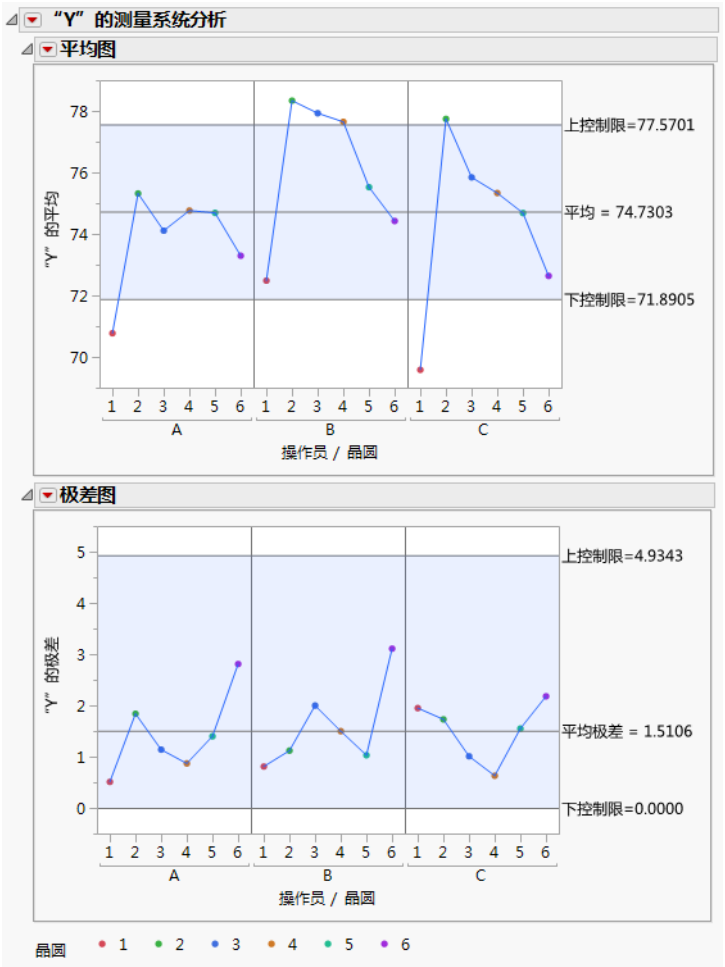
在该示例中，三名操作员对六个晶圆的单个特征测量了两次。执行详细的分析以了解测量系统的性能。

执行初始分析

1. 选择**帮助** > **样本数据库**，然后打开 Variability Data/Wafer.jmp。
2. 选择**分析** > **质量和过程** > **测量系统分析**。
3. 将 Y 分配给 **Y，响应**角色。
4. 将**晶圆**分配给**部件，样本 ID** 角色。
5. 将**操作员**分配给 **X，分组**角色。

   请注意， **MSA 方法**设置为 EMP，**散度图类型**设置为**极差**，**模型类型**设置为**交叉**。
6. 点击**确定**。

图 4.8 平均图和极差图



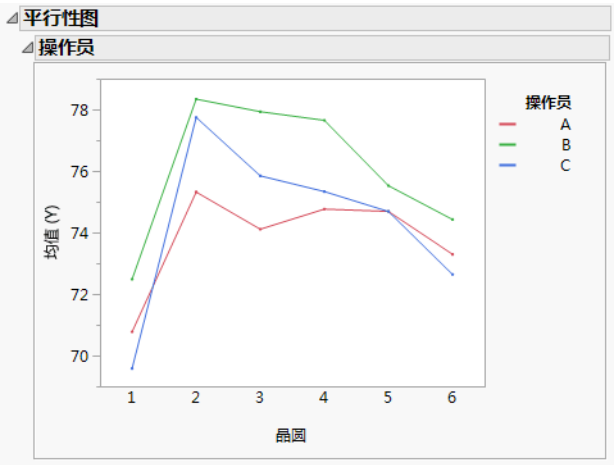
平均图显示某些部件的平均测量值超过控制限。这是想要的结果，指示可测量的部件间变异。

极差图显示没有点超过控制限。这是想要的结果，指示操作员测量值在部件内是一致的。

查看交互作用

我们来进一步查看操作员和部件之间的交互作用。点击“Y 的测量系统分析”旁边的红色小三角并选择**平行性图**。

图 4.9 平行性图

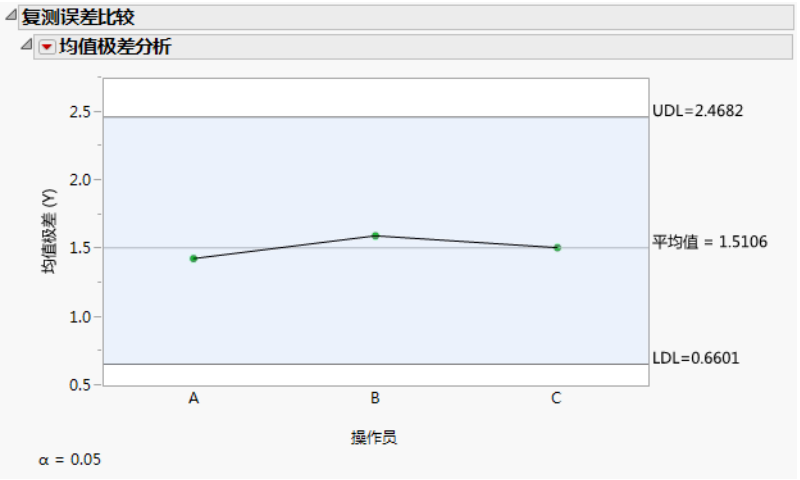


通过查看按操作员的平行性图，您可以看到线条相对平行，只有一些轻微的交叉。

查看操作员一致性

我们来进一步查看操作员之间的变异。点击“Y 的测量系统分析”旁边的红色小三角并选择**复测误差比较**。

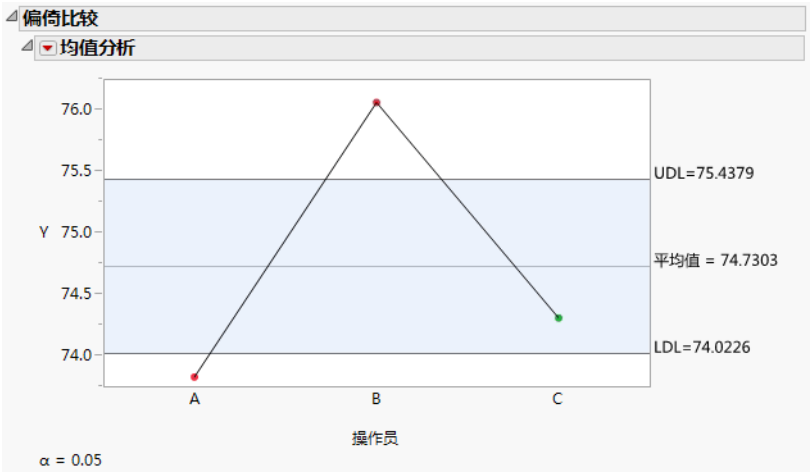
图 4.10 复测误差比较



通过查看“复测误差比较”，您可以看到没有操作员的复测误差与总复测误差有显著差异。这说明操作员的测量是一致的。

为了确定这一点，您决定查看偏倚比较图，该图指示操作员测量部件所得的值是否太高或太低。点击“Y 的测量系统分析”旁边的红色小三角并选择**偏倚比较**。

图 4.11 偏倚比较



通过查看偏倚比较图，您可以发现以下几点：

- 操作员 A 和操作员 B 具有可检测的测量偏倚，因为它们与总平均值具有显著差异。
- 操作员 A 的测量值明显偏低。
- 操作员 B 的测量值明显偏高。
- 操作员 C 的测量值与总平均值没有显著差异。

对测量系统分类

查看“EMP 结果”报表以对测量系统分类，寻找改进的机会。点击“Y 的测量系统分析”旁边的红色小三角并选择 **EMP 结果**。

图 4.12 EMP 结果

EMP 结果

EMP 检验	结果	说明
复测误差	1.3387	组内误差
自由度	16.017	用于估计组内误差的信息量
可能误差	0.9029	单次测量误差的中位数
组间相关（不含偏倚）	0.7385	归因于部件变异的变异比例（不带偏倚因子）
组间相关（带偏倚）	0.6272	归因于部件变异的变异比例（带偏倚因子）
偏倚影响	0.1113	偏移因子对组间相关性的降低量

系统	分类
当前（带偏倚）	二级
潜在（不含偏倚）	二级

监控等级图例

分类	组间相关	过程信号衰减	警告的概率，仅检验 1*	警告的概率，检验 1-4*
一级	0.80 - 1.00	小于 11%	0.99 - 1.00	1.00
二级	0.50 - 0.80	11% - 29%	0.88 - 0.99	1.00
三级	0.20 - 0.50	29% - 55%	0.40 - 0.88	0.92 - 1.00
四级	0.00 - 0.20	大于 55%	0.03 - 0.40	0.08 - 0.92

*** 使用 Wheeler 检验在 10 个子组内的 3 标准误差偏移的警告概率，对应于 Nelson 检验 1、2、5 和 6。**

该分类为“二级”，它指示仅使用检验 1 有 88% 以上的机会在 10 个子组内检测到 3 标准差偏移。您发现偏倚因子对组间相关有 11% 的影响。换句话说，若您可以消除偏倚因子，您的组间相关系数将提高 11%。

探索控制图检测过程变化的能力

使用“偏移检测刻画器”探索控制图可在过程中检测到变化的概率。点击“Y 的测量系统分析”旁边的红色小三角并选择**偏移检测刻画器**。

图 4.13 偏移检测刻画器

默认情况下，选择的唯一检验针对超过 3 个 sigma 限值的点。还请注意，默认子组大小为 1，这指示您在使用单个测量值图。

我们来探索一下在均值偏移之后的 10 个子组中检测到大小为两个部件标准差偏移的能力。点击**部件均值偏移**值 2.1701，将它更改为 4.34（2.17 乘以 2）。检测到两个部件标准差的偏移的概率为 56.9%。

接着，查看消除偏倚如何影响检测到两个部件标准差的偏移的能力。将**偏倚因子标准差**值从 1.1256 更改为 0。检测到偏移的概率增加到 67.8%。

最后，增加更多检验来查看检测到两个部件标准差偏移的概率如何变化。除了第一个检验外，选择第二个、第五个和第六个检验（Wheeler 规则 4、2 和 3）。使用这四个检验在没有偏倚变异的情况下，检测到偏移的概率为 99.9%。

您还可以探索使用较大的子组大小对控制图的影响。对于 2 或更大的子组大小，控制图为均值图。将**偏倚因子标准差**值改回 1.1256，只选择第一个检验。在刻画器中将**子组大小**设置为 4。此时检测到两个部件标准差偏移的概率为 98.5%。

查看测量值增量

最后，查看您的测量值增量是否合适。点击“Y 的测量系统分析”旁边的红色小三角并选择**有效分辨率**。

图 4.14 有效分辨率

有效分辨率

来源		值	说明
概差	(PE)	0.9029	单个测量值的误差中位数
当前测量值增量	(MI)	0.01	根据数据估计的测量值增量（以十分之一为单位）
增量下限	(0.1*PE)	0.0903	测量值增量不应低于该值
最小有效增量	(0.22*PE)	0.1986	测量值增量高于该值更有效
最大有效增量	(2.2*PE)	1.9865	测量值增量低于该值更有效

操作: 删除一位
原因: “0.01” 的测量值增量低于最低测量值增量边界，应调整为记录更少位数。

“当前测量值增量”为 0.01，低于“增量下限” 0.09，指示应调整以后的测量值使记录位数少 1。

“测量系统分析”的统计详细信息

有关“极差图”中使用的限值计算的详细信息，请参见“控制图生成器”一章中第 82 页的“均值图和 R 图的控制限”。有关“标准差图”中使用的限值计算的详细信息，请参见“控制图生成器”一章中第 83 页的“均值图和 S 图的控制限”。

组间相关和概差的计算

按以下方式计算不带偏倚的组间相关：

$$r_{pe} = \frac{\hat{\sigma}_p^2}{\hat{\sigma}_p^2 + \hat{\sigma}_{pe}^2}$$

按以下方式计算带偏倚的组间相关：

$$r_b = \frac{\hat{\sigma}_p^2}{\hat{\sigma}_p^2 + \hat{\sigma}_b^2 + \hat{\sigma}_{pe}^2}$$

按以下方式计算带偏倚和交互作用因子的组间相关：

$$r_{int} = \frac{\hat{\sigma}_p^2}{\hat{\sigma}_p^2 + \hat{\sigma}_b^2 + \hat{\sigma}_{int}^2 + \hat{\sigma}_{pe}^2}$$

按以下方式计算概差：

$$Z_{0.75} \times \hat{\sigma}_{pe}$$

请注意以下事项：

$\hat{\sigma}_{pe}^2$ = 纯误差的方差估计值

$\hat{\sigma}_{p}^2$ = 产品的方差估计值

$\hat{\sigma}_{b}^2$ = 偏倚因子的方差估计值

$\hat{\sigma}_{int}^2$ = 交互作用因子的方差估计值

$Z_{0.75}$ = 标准正态分布的 75% 分位数

第 5 章

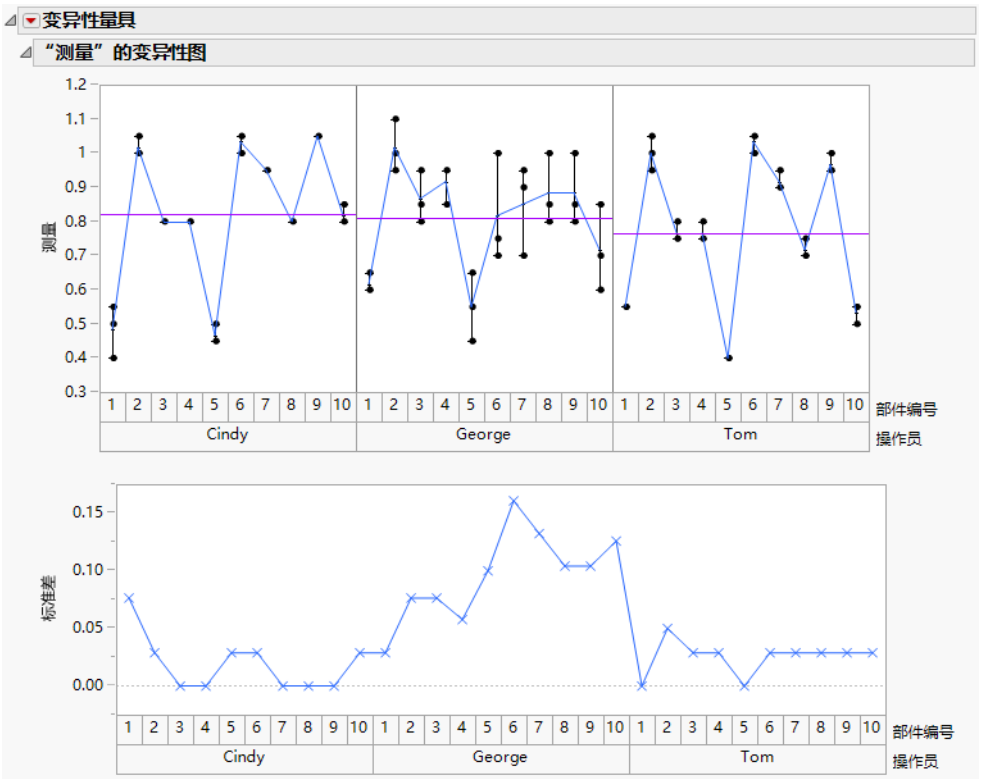
变异性量具图

使用量具 R&R 评估连续测量过程

变异性量具图用于分析连续测量值，可揭示您的测量系统当前的性能。您还可以执行量具研究，查看数据中的变异测量。

提示：本章只介绍变异性图。有关计数图的详细信息，请参见第 139 页的“计数量具图”一章。

图 5.1 变异性图的示例



目录

变异性图概述 117
变异性图的示例 118
启动“变异性 / 计数量具图”平台 119
    数据格式 120
变异性量具图 120
“变异性量具”平台选项 121
    方差非齐性检验 123
    方差分量 123
    关于“量具 R&R 方法” 125
    “量具 R&R”选项 126
    分辨力比 128
    误分类概率 128
    偏倚报表 129
    线性研究 129
变异性图的更多示例 130
    方差非齐性检验的示例 130
    “偏倚报表”选项的示例 132
变异性图的统计详细信息 134
    方差分量的统计详细信息 135
    分辨力比的统计详细信息 136
    误分类概率的统计详细信息 136

变异性图概述

提示：变异性图的传统名称为**多变异**图，但是因为该名称不广为人知，改用更一般的变异性图。

就像控制图显示过程中随时间的变异一样，变异性图显示各个类别（如部件、操作员、重复次数和仪器）的同类变异。变异性图针对分组因子的每个水平绘制数据和均值，将所有图并排放在一起。连同数据一起，您可以查看每个类别中数据的均值、极差和标准差，了解它们如何随类别变化。报表选项基于主要考察均值和方差如何随类别变化的假设。

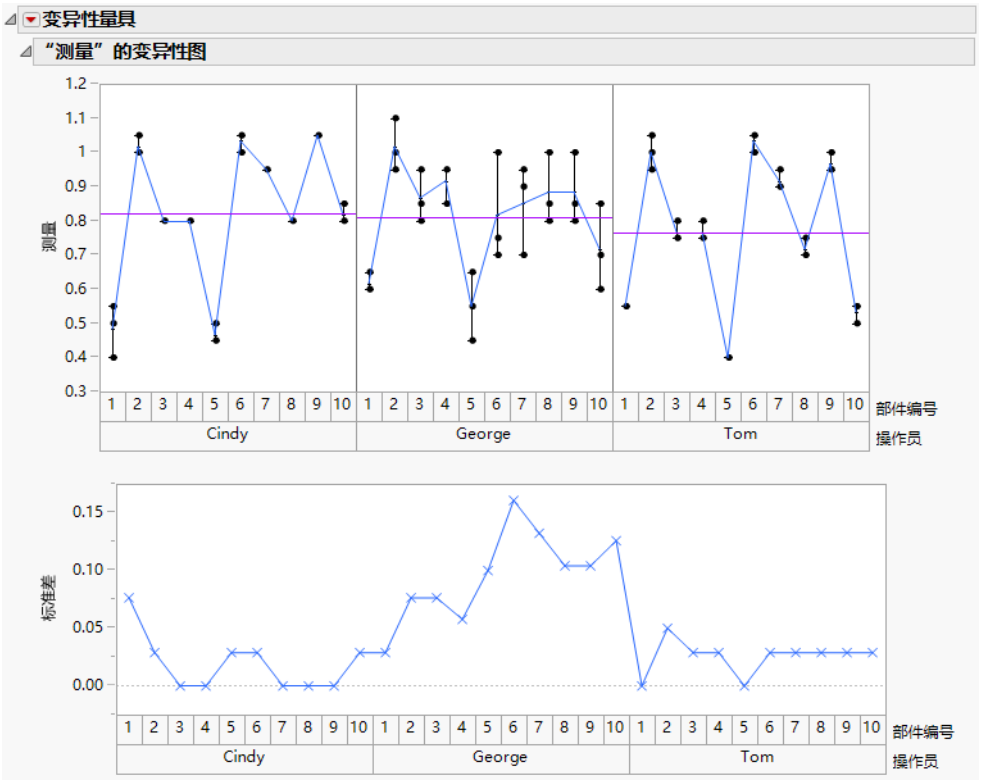
变异性图通常用于量具 R&R 这类测量系统分析。该分析查看多少变异性来自操作员变异（再现性）和测量变异（重复性）。“量具 R&R”适用于许多交叉模型和嵌套模型的组合，不论模型是否平衡。

变异性图的示例

假定您有包含部件测量值的数据。三名操作员 Cindy、 George 和 Tom 分别测量了 10 个部件。他们将每个部件都测量了三次，总共得到 90 个观测。您想确定操作员间的变异。

1. 选择**帮助** > **样本数据库**，然后打开 Variability Data/2 Factors Crossed.jmp。
2. 选择**分析** > **质量和过程** > **变异性 / 计数量具图**。
3. 对于**图表类型**，选择**变异性**。
4. 选择**测量**并点击 **Y，响应**。
5. 选择**操作员**并点击 **X，分组**。
6. 选择**部件编号**并点击**部件，样本 ID**。
7. 点击**确定**。
8. 点击 “变异性量具”红色小三角并选择**显示组均值**和**连接单元格均值**。

图 5.2 变异性图的示例

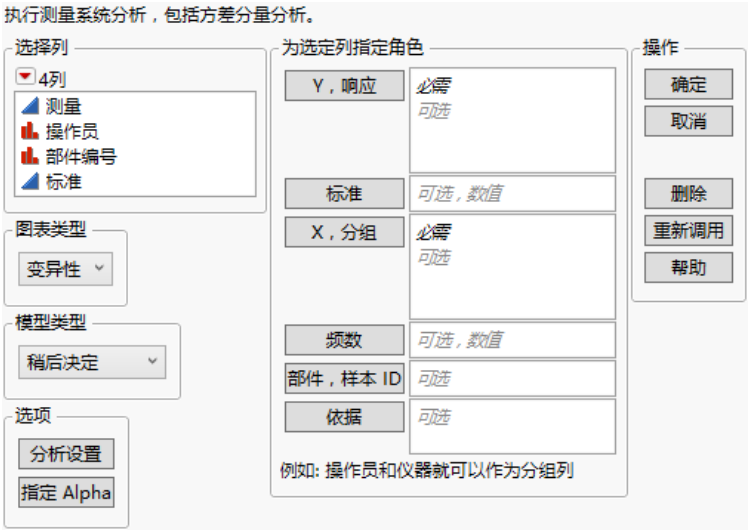


通过查看标准差图，您可以看到 Cindy 和 George 的测量值比 Tom 的测量值有更大的变异，Tom 测量部件是最一致的。George 的测量值具有最大的变异，因此他测量部件是最不一致的。

启动“变异性 / 计数量具图”平台

通过选择**分析 > 质量和过程 > 变异性 / 计数量具图**启动 “变异性 / 计数量具图”平台。将**图表类型**设置为**变异性**。

图 5.3 “变异性 / 计数量具图”启动窗口



有关 “选择列”红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。

图表类型 在变异性量具分析 （对于连续响应）和计数量具分析 （对于分类响应，通常为 “通过”或 “失败”）之间作出选择。

> **注意：**本章的内容仅涉及**变异性**图表类型。有关**计量**图类型的详细信息，请参见第 139 页的 “计数量具图”一章。

模型类型 选择模型类型 （**主效应**、**交叉**、**嵌套**等）。请参见第 135 页的 “方差分量的统计详细信息”。

分析设置 指定计算方差分量的方法。请参见第 124 页的 “分析设置”。

指定 Alpha 指定该平台使用的 alpha 水平。

Y，响应 指定测量值列。指定多个 *Y* 列可为每个响应生成单独的变异性图。

标准 指定一个标准或参考列，它包含测量部件的 “真实”或已知值。包括该列后可启用**偏倚**和**线性研究**选项。这些选项对观测到的测量值与参考或标准值之间的差异执行分析。请参见第 129 页的 “偏倚报表”和第 129 页的 “线性研究”。

X，分组 指定用于对测量值分组的分类列。若因子构成嵌套层次，请先指定高层次的项。若您在进行量具研究，请首先指定操作员，然后指定部件。

频数 标识数据表列，该列的值为每行分配频数。这在有汇总数据时很有用。

部件，样本 ID 标识所测量的部件或样本。

依据 标识一列，该列创建的报表包含变量每个水平的单独分析。

数据格式

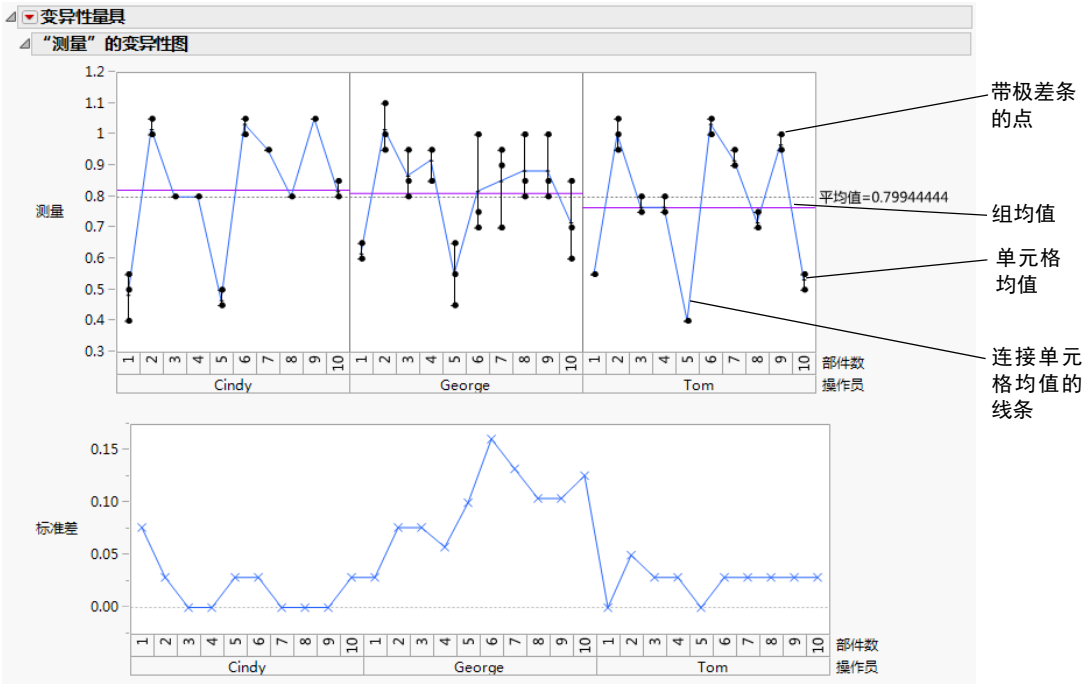
要使用“变异性图”平台，所有响应测量必须位于单个响应列中。有时，响应记录在多个列中，其中每行都是某设计因子的一个水平，每列都是另一个设计因子的一个水平。必须在运行“变异性图”平台之前堆叠该格式的数据。请参见《使用 JMP》。

变异性量具图

变异性图和标准差图显示变异的模式。您可以使用这些图来标识可能的变异组（子组内、子组间、随时间）。若您发现任何这些来源的变异很大，可以想办法减小该来源的变异。

按第 118 页的“变异性图的示例”中的说明操作以生成图 5.4 中所示的结果。

图 5.4 变异性量具图



这些图显示 y 轴上的响应和多水平分类的 x 轴。

在图 5.4 中，测量值图显示按部件划分的每名操作员的测量值极差。每个测量值显示在图上。最大值条和最小值条指示每个单元格值的极差，单元格均值条指示每个值组合的中位数值。标准差图绘制操作员测量每个部件所得测量值的标准差。

您可以向图中添加一些功能，如图 5.4 中所示。请参见第 121 页的“‘变异性量具”平台选项”。

要替换图中的变量，请执行以下操作之一：

- 通过将变量从一个轴标签拖到另一轴标签来交换现有变量。当您将变量拖到图上或点击某个轴标签时，轴标签会突出显示。这样可指示要放置变量的位置。
- 点击相关数据表的“列”面板中的某个变量，将它拖到某个轴标签上。

在其他平台上，相关数据表中排除的行仍显示在图上。但是，在变异性图中，排除的行不显示在图上。

“变异性量具”平台选项

使用红色小三角选项可修改图的外观、执行量具 R&R 分析和计算方差分量。

注意：图 5.4 显示了其中的一些选项。

提示：要设置这些选项的默认行为，请选择**文件 > 首选项 > 平台 > 变异性图**。

垂直图 将布局更改为水平或垂直。

变异性图 显示或隐藏变异性图。

显示点 显示或隐藏各个行对应的点。

显示极差条 显示或隐藏指示每个单元格的最小值和最大值的条。

显示单元格均值 显示或隐藏每个单元格的均值标记。

连接单元格均值 连接或不连接一组单元格内的单元格均值。

显示分隔线 显示或隐藏 **X，分组**变量的两个水平间的分隔线。

显示组均值 （仅在您有两个或多个“X，分组”变量或一个“X，分组”变量和一个“部件，样本 ID”变量时可用。）显示或隐藏由水平实线表示的单元格组的均值。将显示一个窗口，提示您选择一个分组变量。

显示总均值 显示或隐藏总均值，它用横跨整个图形的灰色虚线表示。

显示总中位数 显示或隐藏总中位数，它用横跨整个图形的蓝色虚线表示。

显示箱线图 显示或隐藏箱线图。

均值菱形 显示或隐藏均值菱形。置信区间使用每个单元格的组内标准差。

均值控制限 在变异性图上显示或隐藏上控制限和下控制限处的线条。有关这些限值计算的详细信息，请参见“控制图生成器”一章中第 82 页的 ““控制图生成器”的统计详细信息”。

随机散布点 向绘制的点添加一些随机噪声，以便同时发生的点在图上不重合。

显示标准均值 （仅当您指定了**标准**变量时可用。）显示或隐藏标准列的均值。

变异性汇总报表 显示或隐藏一个报表，该报表给出均值、标准差、变异系数 (CV)、均值标准误差、置信区间上限和下限、最小值、最大值以及观测数。

标准差图 显示或隐藏一个单独的图形，其中显示各类别单元格的单元格标准差。

标准差均值 在标准差图上显示或隐藏标准差均值处的线条。

标准差控制限 在标准差图上显示或隐藏指示上控制限和下控制限的线条。有关这些限值计算的详细信息，请参见 “控制图生成器”一章中第 82 页的 ““控制图生成器”的统计详细信息”。

标准差组均值 在标准差图上显示或隐藏均值线。

方差非齐性检验 执行一个检验来比较各组的方差。请参见第 123 页的 “方差非齐性检验”。

方差分量 估计特定模型的方差分量。计算以下模型的方差分量：主效应、交叉、嵌套、先交叉后嵌套（仅限 3 因子）和先嵌套后交叉（仅限 3 因子）。请参见第 123 页的“方差分量”。

量具研究 包含以下选项：

量具 R&R 将第一个因子解释为分组列，将最后一个因子解释为部件，并使用估计的方差分量创建量具 R&R 报表。（请注意启动窗口中还有一个“部件”字段）。请参见第 126 页的“量具 R&R”选项”。

分辨力比 描述特定产品的指定测量值的相对效用。它比较测量值的总方差与测量值误差的方差。请参见第 128 页的“分辨力比”。

误分类概率 显示拒绝良好部件和接受不良部件的概率。请参见第 128 页的“误分类概率”。

偏倚报表 显示观测值与标准之间的平均差异。将显示一个平均偏倚图和汇总表。仅当您在启动窗口中指定“标准”变量时，该选项才可用。请参见第 129 页的“偏倚报表”。

线性研究 使用标准值作为 X 变量且偏倚作为 Y 变量执行回归。该分析检查偏倚和部件大小之间的关系。理想情况下，您希望斜率为 0。非零斜率指示您的量具在测量不同大小的部件时表现不同。仅当您在启动窗口中指定“标准”变量时，该选项才可用。请参见第 129 页的“线性研究”。

量具 R&R 图 显示或隐藏均值图（按模型中每个主效应划分的响应均值）和标准差图。若模型是纯嵌套的，将使用嵌套结构显示图形。若模型是纯交叉的，则显示交互作用图形。否则，独立对每个效应绘制图形。对于标准差图，用红线连接每个效应的 $\sqrt{\text{加权方差的均值}}$。

AIAG 标签 允许您指定应根据 AIAG 标准（在汽车行业分析中使用很广泛）标记质量统计量。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

方差非齐性检验

注意：请参见第 130 页的“方差非齐性检验的示例”。

方差非齐性检验选项执行一个检验来比较各组的方差。该检验是基于方差均值分析 (ANOMV) 的方法。该方法指示任何组标准差是否与平均组方差的平方根有差异。

为了对非正态数据保持稳健，该方法使用置换模拟来计算决策限。有关该方法的详细信息，请参见 Wludyka and Sa (2004)。由于该方法使用模拟，所以决策限每次可能稍有不同。要每次获得相同的结果，请按 Ctrl+Shift，选择该选项，然后指定相同的随机种子。

检验报表的红色小三角菜单包含以下选项：

设置 Alpha 水平 设置检验的 alpha 水平。

显示汇总报表 显示或隐藏检验的汇总报表。报表中的值与图中显示的值相同。

显示选项 显示或隐藏决策限、阴影、中心线和针。

方差分量

方差分量选项用于对测量值间的变异建模。假定响应为常数均值加上与分类的各个水平关联的随机效应。

注意：您选择**方差分量**选项后，若之前未在启动窗口中选择**模型类型**（若选择了**稍后决定**），系统将提示您选择模型类型。有关模型类型的详细信息，请参见第 119 页的“启动“变异性/计数量具图”平台”。

图 5.5 “分差分量”报表的示例

方差分析

源	自由度	平方和	均方	F 比	概率>F
操作员	2	0.054889	0.02744	1.3150	0.2931
部件数	9	2.633583	0.29262	14.0209	<.0001*
操作员*部件数	18	0.375667	0.02087	5.0425	<.0001*
组内	60	0.248333	0.00414		
合计	89	3.312472	0.03722		

方差分量

成分	方差分量	占合计的百分比	20 40 60 80	平方根（方差分量）
操作员	0.00021914	0.5461		0.01480
部件数	0.03019444	75.2		0.17377
操作员*部件数	0.00557716	13.9		0.07468
组内	0.00413889	10.3		0.06433
合计	0.04012963	100.0		0.20032

仅当使用 EMS 方差分量估计方法时，才显示“方差分析”报表。该报表显示模型中每个效应的显著性。

“方差分量”报表显示估计值本身。请参见第 135 页的“方差分量的统计详细信息”。

分析设置

从启动窗口，点击**分析设置**可选择计算方差分量的方法。

图 5.6 “分析设置”窗口

选择最佳分析（EMS、REML 或 Bayesian） 使用以下逻辑从 EMS、REML 或 Bayesian 中选择最佳分析：

- 若数据平衡，且没有方差分量为负数，则使用 EMS（期望均方）来估计方差分量。
- 若数据不平衡，则使用 REML（限制最大似然）方法，除非估计方差分量为负数，此时使用 Bayesian 方法。
- 若使用 EMS 方法估计任何方差分量为负数，则使用 Bayesian 方法。
- 若方差分量中存在混杂，则使用有界 REML 方法，并且所有负方差分量估计值都设为零。

选择最佳分析（EMS 或 REML） 使用与**选择最佳分析（EMS、REML 或 Bayesian）**选项相同的逻辑从 EMS 或 REML 中选择最佳分析。但是，该选项从不使用 Bayesian 方法，即使方差分量为负数。使用有界 REML 方法并将任何负方差分量强制设置为 0。

使用 REML 分析 使用有界 REML 方法，即使数据平衡。有界 REML 方法可以处理不平衡数据并将任何负方差分量强制设置为 0。

使用 Bayesian 分析 使用 Bayesian 方法。Bayesian 方法可处理不平衡数据并将所有方差分量强制设置为正数且非零。若方差分量中存在混杂，则使用有界 REML 方法，并且所有负方差分量估计值都设为零。在 JMP 中实施的方法使用 Jeffreys 先验的修改版本计算后验均值。请参见 Portnoy (1971) 和 Sahai (1974)。

最多迭代次数 （仅适用于 REML 方法。）对于复杂的问题，您可能要增加迭代次数。增加该值意味着 JMP 在优化阶段将尝试更多次来求解。

收敛极限 （仅适用于 REML 方法。）对于要求更高精度的问题，您可能要将收敛极限变得小一点。减小该值意味着 JMP 在优化阶段将使用更高的准确度来求解。不过，这会增加求解的时间。提供较大的收敛值会更快返回结果，但精度也会降低。

迭代横坐标数 （仅适用于 Bayesian 方法。）对于要求更高准确度的问题，您可能要增加迭代横坐标数。不过，这会增加求解的时间。提供较小的数字会更快返回结果，但精度也会降低。

函数最大计算次数 （仅适用于 Bayesian 方法。）对于要求更高准确度的问题，您可能要增加函数最大计算次数。不过，这会增加求解的时间。提供较小的数字会更快返回结果，但精度也会降低。

关于“量具 R&R 方法”

“量具 R&R”方法分析测量系统中的多少变异性来自操作员变异（再现性）和测量变异（重复性）。“量具 R&R”研究适用于许多交叉模型和嵌套模型的组合，不论模型是否平衡。

提示：或者，您可以使用 EMP 方法来评估测量系统。请参见第 91 页的“测量系统分析”一章。

在执行量具 R&R 研究前，从您的过程中所有不同大小的部件中随机抽取作为部件样本。随机选择几名操作员来测量每个部件几次。变异归因于以下来源：

- **过程变异**，来自部件间。若测量值是可靠的，这是您要研究的最终变异。
- 执行多次测量固有的变异性，即**重复性**。在第 125 页上的表 5.1 中，这称为**组内变异**。
- 由于不同操作员测量部件导致的变异性，即**再现性**。

“量具 R&R”分析便根据重复性和再现性报告变异。

表 5.1 “量具 R&R”分析中的术语和求和定义

方差和	术语和缩写	备择术语
V(组内)	重复性 (EV)	设备变异
V(操作员)+V(操作员*部件)	再现性 (AV)	评估者变异
V(操作员*部件)	交互作用 (IV)	交互作用变异

表 5.1 “量具 R&R”分析中的术语和求和定义 (续)

方差和	术语和缩写	备择术语
V(组内)+V(操作员)+V(操作员*部件)	量具 R&R (RR)	测量变异
V(部件)	部件变异 (PV)	部件变异
V(组内)+V(操作员)+ V(操作员*部件)+V(部件)	总变异 (TV)	总变异

Shewhart 控制图可标识随时间失控的过程。变异性图还可以帮助标识在均值或方差中存在系统差异的操作员、仪器或部件来源。

“量具 R&R”选项

量具 R&R 选项显示操作员和部件的量具研究所解释的变异测量。

一旦选择**量具 R&R** 选项，若尚未选择模型类型，系统将提示您选择模型类型。然后修改“量具 R&R”规格。

注意：“变异性”的平台首选项包含“量具 R&R 规格”对话框选项。默认情况下选择该首选项。取消选择该首选项将使用在数据表中定义的规格限。

输入 / 验证量具 R&R 规格

“输入 / 验证量具 R&R 规格”窗口包含三个选项：

选择容差输入方法 选择如何输入容差：

选择**容差区间**可直接输入容差，其中容差 = 上规格限 - 下规格限。

选择**上规格限和 （或）下规格限**可输入规格限，然后使 JMP 计算容差。

K， Sigma 乘数 K 是您选择要与 sigma 相乘的常数值。例如，您可能键入 6 以便查看 6*sigma 或 6 sigma 过程。

提示：通过选择**文件 > 首选项 > 平台 > 变异性图**可修改 K 的默认值。

容差区间，上规格限 - 下规格限 输入过程的容差，该值为上规格限与下规格限的差值。

规格限 输入上规格限和下规格限。请参见 《使用 JMP》。

历史均值 计算单侧规格限的容差范围，即 “上规格限 - 历史均值”或 “历史均值 - 下规格限”。若未输入历史均值，则使用总均值。

历史 Sigma 输入一个描述变异的值 （您可以从历史记录或过去的经验得到该值）。

“量具 R&R”报表

图 5.7 “量具 R&R”报表的示例

量具 R&R

源测量值		变异 (6*标准差)		6*以下项的平方根
重复性	(EV)	0.3860052	设备变异	V(组内)
再现性	(AV)	0.4568005	评估者变异	V(操作员) + V(操作员*部件编号)
操作员		0.0888194		V(操作员)
操作员*部件编号		0.4480823		V(操作员*部件编号)
量具 R&R	(RR)	0.5980524	测量变异	V(组内) + V(操作员) + V(操作员*部件编号)
部件变异	(PV)	1.0425929	部件变异	V(部件编号)
总变异	(TV)	1.2019429	总变异	V(组内) + V(操作员) + V(操作员*部件编号) + V(部件编号)

6 k
49.7571 % 量具 R&R = 100*(RR/TV)
0.57362 （精度/部件变异）比率 = RR/PV
2 不同种类的分类数 = 1.41*(PV/RR)
对于部件使用最后一列“部件编号”。

完整的量具 R&R 报表

量具 R&R

源测量值		变异 (6*标准差)		6*以下项的平方根
重复性	(EV)	0.3860052	设备变异	V(组内)
再现性	(AV)	0.4568005	评估者变异	V(操作员) + V(操作员*部件编号)
操作员		0.0888194		V(操作员)
操作员*部件编号		0.4480823		V(操作员*部件编号)
量具 R&R	(RR)	0.5980524	测量变异	V(组内) + V(操作员) + V(操作员*部件编号)
部件变异	(PV)	1.0425929	部件变异	V(部件编号)
总变异	(TV)	1.2019429	总变异	V(组内) + V(操作员) + V(操作员*部件编号) + V(部件编号)

6 k
49.7571 % 量具 R&R = 100*(RR/TV)
0.57362 （精度/部件变异）比率 = RR/PV
2 不同种类的分类数 = 1.41*(PV/RR)

简化量具 R&R 报表

注意：要生成简化的量具 R&R 报表，请选择**文件 > 首选项 > 平台 > 变异性图 > 简化的量具 R&R 报表**。

在该示例中，“变异”列中的值是按 *k* 的值（该示例中为 6）统一尺度的方差分量之和的平方根。

表 5.2 显示 Barrentine (1991) 所建议的测量变异准则。

表 5.2 可接受的测量变异百分比

< 10%	非常好
11% 到 20%	合适
21% 到 30%	勉强可接受
> 30%	不可接受

请注意以下几点：

- 若您在“输入/验证量具 R&R 规格”窗口中提供了**容差区间**，则在“量具 R&R”报表中会显示**容差百分比**列。该列的计算公式为：100*(变异/容差)。此外在报表底部会显示（**精度/容差**）比率。该比率表示由于量具变异性丧失的容差或能力区间所占的比例。
- 若您在“输入/验证量具 R&R 规格”窗口中提供了**历史 Sigma**，则在“量具 R&R”报表中会显示**过程百分比**列。该列定义为 100*(变异/(K*历史 Sigma))。
- 区别分类数 (NDC) 定义为：(1.41*(PV/RR))，向下舍入到最接近的整数。

分辨力比

分辨力比描述特定产品的指定测量值的相对效用。通常，当分辨力比小于 2 时，测量无法检测到产品变异，这意味着测量过程需要改进。大于 4 的分辨力比足够检测到不可接受的产品变异，这意味着生产过程需要改进。

请参见第 136 页的“分辨力比的统计详细信息”。

误分类概率

由于测量变异，可能拒绝良好部件和接受不良部件。这称为误分类。误分类率随测量变异性降低而降低。您选择**误分类概率**选项后，若尚未选择模型类型和输入规格限，系统将提示您这样做。

图 5.8 “误分类概率”报表的示例

误分类概率

说明	概率
P（错误拒绝了良好部件）	0.0747
P（错误接受了不良部件）	0.2472
P（部件良好但被拒绝）	0.0654
P（部件不良但被接受）	0.0307
P（部件良好）	0.8757

误分类概率基于部件的测量值 *Y* 和部件的真实值 *X* 的联合概率函数。使用的联合概率密度函数服从二元正态分布。为理解这些说明，需定义以下概率：

$\delta = P[(LSL \le X \le USL) \text{ and } (Y < LSL \text{ or } Y > USL)]$

$\beta = P[(X < LSL \text{ or } X > USL) \text{ and } (LSL \le Y \le USL)]$

$\pi = P(LSL \le X \le USL)$

说明

P（错误拒绝了良好部件） 部件虽良好但被拒绝的条件概率，或 δ/π。

P（错误接受了不良部件） 部件虽不良但被接受的条件概率，或 $\beta/(1-\pi)$。

P（部件良好但被拒绝） 部件良好但被拒绝的联合概率，或 δ。

P（部件不良但被接受） 部件不良但被接受的联合概率，或 β。

P（部件良好） 部件良好的概率，或 π。

有关详细信息，请参见第 136 页的“误分类概率的统计详细信息”和 Burdick et al.(2005) 一书。

偏倚报表

偏倚报表显示总测量值偏倚图和汇总表，以及相对于标准值的测量值偏倚图和汇总表。显示 X 变量的每个水平的平均偏倚，即观测值与标准值之间的差值。还给出偏倚的 *t* 检验。

仅当在启动窗口中提供**标准**变量时，**偏倚报表**选项才可用。

“测量值偏倚报表”红色小三角菜单包含以下选项：

置信区间 计算每个部件的平均偏倚的置信区间，并在相对于标准值的测量值偏倚报表图上设置标记。

测量值误差图 生成测量值误差对所有分组列的图。还提供单独的每个分组列的测量值误差图。

线性研究

线性研究使用标准变量作为 X 变量且偏倚作为 Y 变量执行回归分析。该分析检查偏倚和部件大小之间的关系。理想上，您希望斜率为零。若斜率与零存在显著差异，您可以推断部件大小或作为标准测量的变量与测量能力之间有显著关系。

仅当在启动窗口中提供**标准**变量时，**线性研究**选项才可用。

该报表显示以下信息：

- 每个标准的偏倚汇总统计量。
- ANOVA 表，检验线条的斜率是否等于零。
- 线条参数，包括斜率（线性）和截距（偏倚）的检验。仅当斜率的检验无法拒绝斜率 = 0 的原假设时，截距的检验才有用。
- 线条的方程显示在图形正下方。

“线性研究”红色小三角菜单包含以下选项：

设置 Alpha 水平 更改在偏倚置信区间中使用的 alpha 水平。

分组线性图 为您在启动窗口中指定的 **X，分组**变量的每个水平生成单独的线性图。

变异性图的更多示例

- “方差非齐性检验的示例”
- ““偏倚报表”选项的示例”

方差非齐性检验的示例

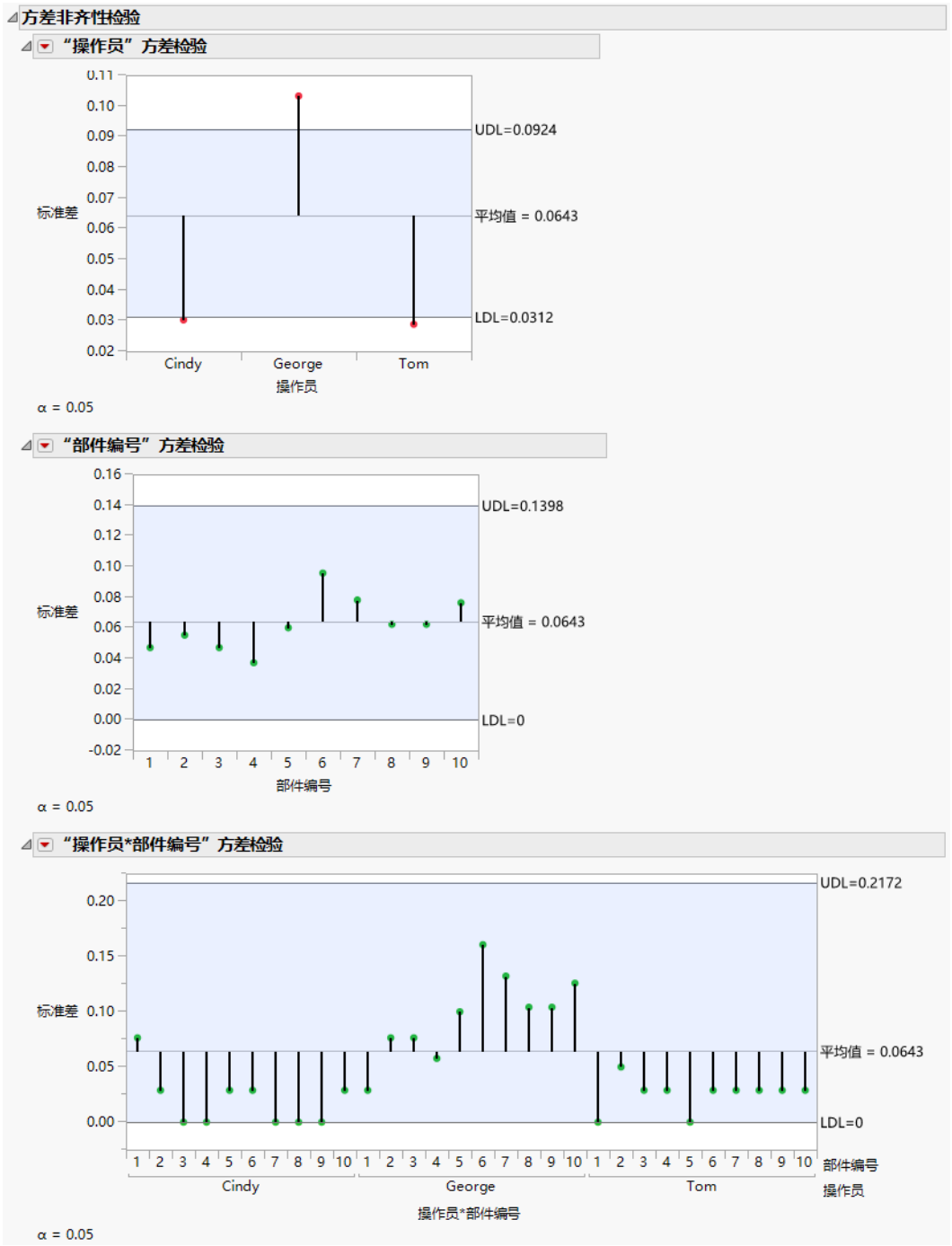
假定您有包含部件测量值的数据。三名操作员 （Cindy、 George 和 Tom）分别测量了 10 个部件。他们将每个部件都测量了三次，总共得到 90 个观测。您想查看以下事项：

- 每名操作员的测量值方差是否相同
- 每个部件的方差是否相同
- 每个操作员*部件组合的方差是否相同

理想上，您希望每个组的所有方差在统计上视为相同。

1. 选择**帮助 > 样本数据库**，然后打开 Variability Data/2 Factors Crossed.jmp。
2. 选择**分析 > 质量和过程 > 变异性 / 计数量具图**。
3. 选择**测量**并点击 **Y，响应**。
4. 选择**操作员**并点击 **X，分组**。
5. 选择**部件编号**并点击**部件，样本 ID**。
6. 点击**确定**。
7. 点击 “变异性量具”红色小三角并选择**方差非齐性检验**。
8. 选择**交叉**。
9. 点击**确定**。

图 5.9 “方差非齐性检验”报表



注意：由于该方法使用模拟，所以决策限每次可能稍有不同。

在“操作员”方差检验中，所有三个水平超过了上决策限和下决策限。据此您推断每名操作员与平均组方差的平方根有不同的变异性。您可能想要了解为什么每名操作员之间的变异不同。

在“部件编号”方差检验和交互作用（操作员*部件编号）方差检验中，没有超过决策限的水平。据此您推断这些方差与平均组方差的平方根在统计上没有差异。每个部件与其他部件具有相似的方差，每个操作员*部件编号组合与其他操作员*部件编号组合具有相似的方差。

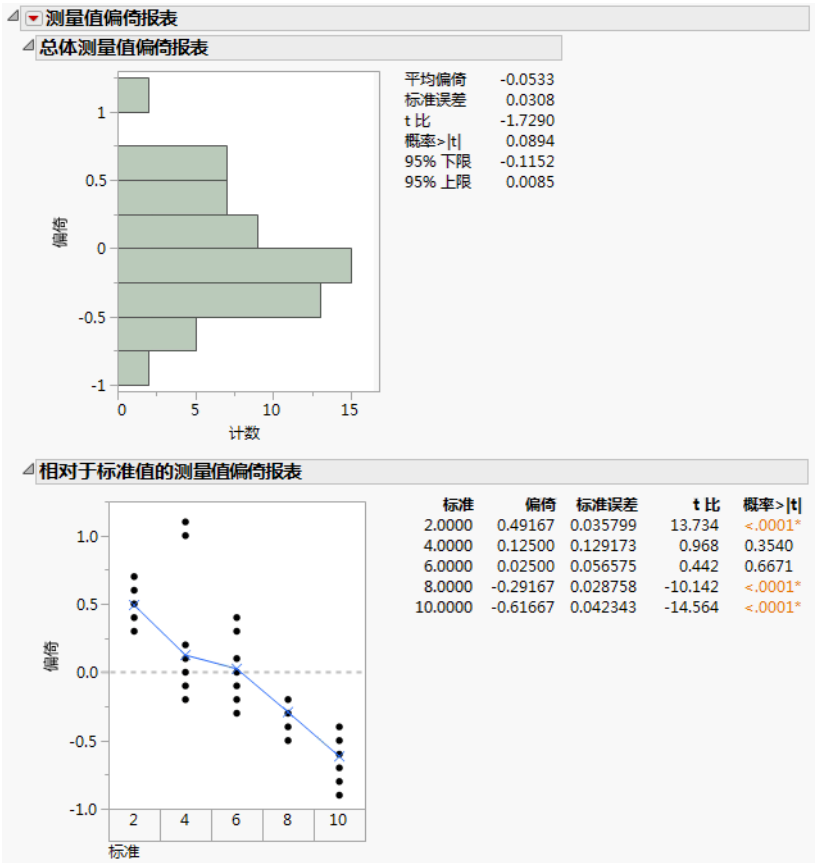
“偏倚报表”选项的示例

注意：这些数据来自汽车工业行动集团 (2002)。

假定您是工厂管理者，要在过程中引进新的测量系统。作为生产部件批准过程 (PPAP) 的一部分，需要评估测量系统的偏倚和线性。根据记录的过程变异，在测量系统的工作范围中，选择了 5 个部件。对每个部件进行全尺寸检验，以确定其参考值。随后由操作员组长对每个部件测量 12 次。这些部件是当天随机选择的。在该示例中，您要检查总偏倚和单个测量值偏倚（相对于标准值）。

1. 选择**帮助** > **样本数据库**，然后打开 Variability Data/MSALinearity.jmp。
2. 选择**分析** > **质量和过程** > **变异性 / 计数量具图**。
3. 选择**响应**并点击 **Y，响应**。
4. 选择**标准**并点击**标准**。
5. 选择**部件**并点击 **X，分组**。
6. 点击**确定**。
7. 点击 “变异性量具”红色小三角并选择**量具研究** > **偏移报表**。

图 5.10 测量值偏倚报表



为每个测量值计算偏倚（**响应 - 标准**）。“总测量值偏倚报表”显示偏倚的直方图和 *t* 检验结果，以查看平均偏倚是否为 0。您可以看到平均偏倚不为 0，它等于 -0.0533。但是，0 包含在置信区间 (-0.1152,0.0085) 内，这表示平均偏倚与 0 没有显著差异。在显著性水平为 0.05 时，您可以看到 *p* 值大于 0.05，这也显示平均偏倚与 0 没有显著差异。

“相对于标准值的测量值偏倚报表”显示每个部件的平均偏倚值。在图上绘制每个部件的偏倚平均值和实际偏倚值，以便您可以看到它的散布情况。在该示例中，部件编号 1（标准值为 2）具有高偏倚，部件 4 和 5（标准值为 8 和 10）具有低偏倚。

提示：要查看偏倚的置信区间，请在表中右击，然后选择**列 > 95% 下限**和 **95% 上限**。

“线性研究”的示例

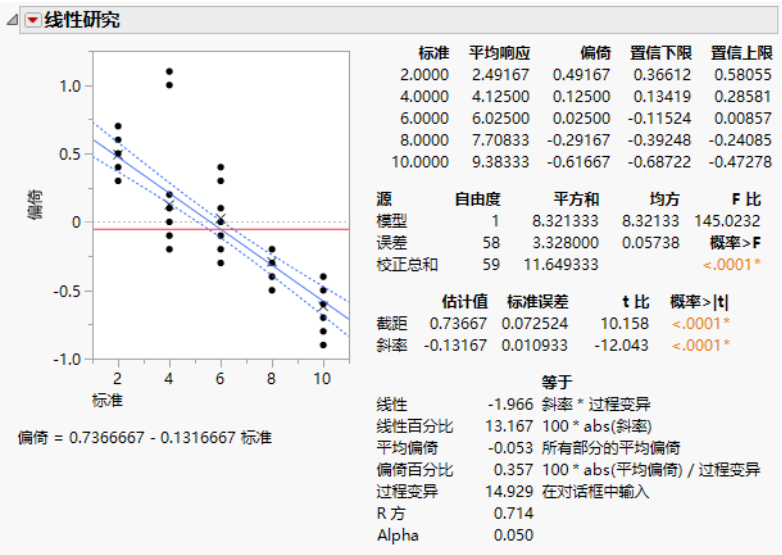
使用与**偏倚报表**选项相同的数据和情形，您现在可以检查线性以确定部件大小与操作员测量它们的能力之间是否存在显著关系。

1. 选择**帮助 > 样本数据库**，然后打开 Variability Data/MSALinearity.jmp。

2. 选择**分析 > 质量和过程 > 变异性 / 计数量具图**。
3. 选择**响应**并点击 **Y，响应**。
4. 选择**标准**并点击**标准**。
5. 选择**部件**并点击 **X，分组**。
6. 点击**确定**。
7. 点击“变异性量具”红色小三角并选择**量具研究 > 线性研究**。
8. 在提示您指定过程变异的窗口中，键入 14.9286。

   值 14.9286 是响应标准差 2.488105 的 6 倍。

图 5.11 线性研究



标准	平均响应	偏倚	置信下限	置信上限
2.0000	2.49167	0.49167	0.36612	0.58055
4.0000	4.12500	0.12500	0.13419	0.28581
6.0000	6.02500	0.02500	-0.11524	0.00857
8.0000	7.70833	-0.29167	-0.39248	-0.24085
10.0000	9.38333	-0.61667	-0.68722	-0.47278

源	自由度	平方和	均方	F 比
模型	1	8.321333	8.32133	145.0232
误差	58	3.328000	0.05738	概率>F
校正总和	59	11.649333		<.0001*

	估计值	标准误差	t 比	概率>|t|
截距	0.73667	0.072524	10.158	<.0001*
斜率	-0.13167	0.010933	-12.043	<.0001*

等于		
线性	-1.966	斜率 * 过程变异
线性百分比	13.167	100 * abs(斜率)
平均偏倚	-0.053	所有部分的平均偏倚
偏倚百分比	0.357	100 * abs(平均偏倚) / 过程变异
过程变异	14.929	在对话框中输入
R 方	0.714	
Alpha	0.050	

请注意以下几点：

- 斜率为 -0.131667。该值作为等式的一部分显示在图形下方，还显示在第三个表中。
- 与斜率检验相关的 *p* 值很小，<.0001。执行斜率的 *t* 检验，查看偏倚是否随标准值变化。

因为 *p* 值很小，您可以推断部件大小与操作员测量它们的能力之间存在显著的线性关系。您还可以在图中看到这一点。若部件或标准值很小，则偏倚很高，反之亦然。

变异性图的统计详细信息

- “方差分量的统计详细信息”
- “分辨力比的统计详细信息”

- “误分类概率的统计详细信息”

方差分量的统计详细信息

您选择的确切模型类型取决于如何收集数据。例如，操作员是测量相同部件 （在这种情况下您有交叉设计）还是测量不同部件 （在这种情况下您有嵌套设计）？为了说明这一点，假定在一个模型中， *B* 嵌套在 *A* 中，多个测量值同时嵌套在 *B* 和 *A* 中，有 *na*•*nb*•*nw* 个测量值，以下表述成立：

- *na* 个随机效应归因于 *A*
- *na*•*nb* 个随机效应归因于 *A* 内的每 *nb* 个 *B* 水平
- *na*•*nb*•*nw* 个随机效应归因于 *A* 内 *B* 内的每 *nw* 个水平

$$y_{ijk} = u + Za_i + Zb_{ij} + Zw_{ijk}$$

Z 是分类的每个水平的随机效应。假定每个 *Z* 的均值为零，独立于所有其他随机项。响应 *y* 的方差是归因于每个 *z* 分量的方差之和：

$$\text{方差}(y_{ijk}) = \text{方差}(Za_i) + \text{方差}(Zb_{ij}) + \text{方差}(Zw_{ijk})$$

表 5.3 显示支持的模型和模型中有什么效应。

表 5.3 “变异性图”平台支持的模型

模型	因子	模型中的效应
主效应	1	A
	2	A，B
	无限制	针对更多因子，依此类推
交叉	1	A
	2	A，B，A*B
	3	A，B，A*B，C，A*C，B*C，A*B*C
	4	A，B，A*B，C，A*C，B*C，A*B*C，D，A*D，B*D，A*B*D，C*D，A*C*D，B*C*D，A*B*C*D，
	无限制	针对更多因子，依此类推

表 5.3 “变异性图”平台支持的模型 (续)

模型	因子	模型中的效应
嵌套	1	A
	2	A，B(A)
	3	A，B(A)，C(A,B)
	4	A，B(A)，C(A,B)，D(A,B,C)
	无限制	针对更多因子，依此类推
先交叉后嵌套	3	A，B，A*B，C(A,B)
先嵌套后交叉	3	A，B(A)，C，A*C，C*B(A)

分辨力比的统计详细信息

分辨力比比较测量值的总方差 M 与测量值误差的方差 E。计算所有主效应 （包括嵌套的主效应）的分辨力比。按以下方式计算分辨力比 D：

$$D = \sqrt{2\left(\frac{P}{T-P}\right)+1}$$

其中：

P = 因子的估计方差

T = 估计的总方差

误分类概率的统计详细信息

本节说明 “误分类概率”报表中概率的计算。误分类概率基于部件的测量值 Y 和部件的真实值 X 的联合概率函数。联合概率分布函数 $F_{YX}(y, x)$ 使用具有均值向量 [μ, μ] 和以下协方差矩阵的二元正态分布：

$$\begin{bmatrix} \gamma_P+\gamma_M & \gamma_P \\ \gamma_P & \gamma_P \end{bmatrix}$$

其中，γ_P 是部件间变异，γ_M 是测量变异，μ 是总均值。这些数量可以在报表窗口中找到或从报表窗口中的数量派生。具体而言，$\gamma_P+\gamma_M$ 等于总变异 (TV) 除以 6 之后的平方：$(TV/6)^2$，γ_P 等于部件变异 (PV) 除以 6 之后的平方：$(PV/6)^2$。Y 与 X 之间的相关性 ρ_{YX} 定义为 $\gamma_P/(\gamma_P+\gamma_M)$ 的平方根。

下一步，定义以下概率：

$\delta = P[(LSL \le X \le USL) \text{ and } (Y < LSL \text{ or } Y > USL)]$

$\beta = P[(X < LSL \text{ or } X > USL) \text{ and } (LSL \le Y \le USL)]$

$\pi = P(LSL \le X \le USL)$

可以使用联合概率分布函数 $F_{YX}(y, x)$ 以及 Y 和 X 的边缘概率分布函数 $F_Y(y)$ 和 $F_X(x)$ 来表示这些概率：

$\delta = F_{YX}(LSL, USL) - F_{YX}(LSL, LSL) - F_{YX}(USL, USL) + F_{YX}(USL, LSL) + F_X(USL) - F_X(LSL)$

$\beta = F_{YX}(USL, LSL) - F_{YX}(LSL, LSL) - F_{YX}(USL, USL) + F_{YX}(LSL, USL) + F_Y(USL) - F_Y(LSL)$

$\pi = F_X(USL) - F_X(LSL)$

P（错误拒绝了良好部件）= δ/π

P（错误接受了不良部件）= $\beta/(1-\pi)$

P（部件良好但被拒绝）= δ

P（部件不良但被接受）= β

P（部件良好）= π

第 6 章

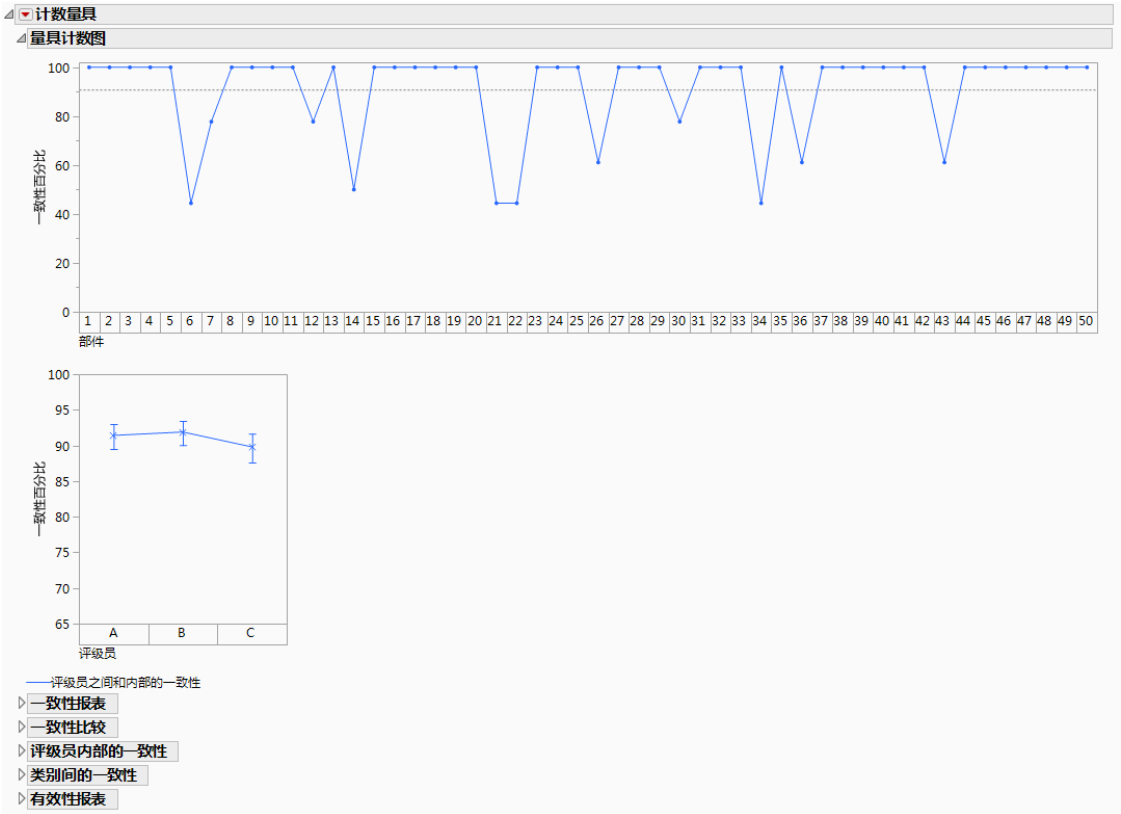
计数量具图

使用一致性测量评估分类测量过程

计数图分析分类测量值，并且有助于为您显示各响应（如评级员）之间的一致性测量。在**计数数据**中，关注的变量具有有限数量的类别。通常，数据仅有两种可能的结果，例如通过或失败。您可以检查如下方面：评级员在对某个部件分类时的有效程度、评级员相互之间的一致程度，以及评级员在若干次评级过程中保持自我一致的程度。

提示：本章只介绍计数图。有关变异性图的详细信息，请参见第 115 页的“变异性量具图”一章。

图 6.1 计数图示例



目录

计数量具图概述 141
计数量具图示例 141
启动“变异性 / 计数量具图”平台 143
计数量具图和报表 144
    一致性报表 145
    有效性报表 145
“计数量具”平台选项 147
计数量具图的统计详细信息 148
    一致性报表的统计详细信息 149

计数量具图概述

创建计数量具图之前，应使用以下准则设置数据的格式：

- 为了比较不同评级员之间的一致性，数据表中的每个评级员必须位于单独的列中。随后在启动窗口中将这些列分配至 **Y，响应**角色。在图 6.2 中，每个评级员（A、B 和 C）都位于单独的列中。
- 不同列中的响应可以是字符（通过或失败），也可以是数值（0 或 1）。在图 6.2 中，评级员响应为数值（0 代表通过，1 代表失败）。所有的响应列必须具有相同的数据类型。
- 您可能要用作 **X，分组**变量的任何其他关注变量都应各自堆叠在一列中显示（请参见图 6.2 中的“部件”列）。您还可以定义一个“标准”列，用于生成包含评级员与标准的比较结果的报表。“标准”列与响应列的数据类型必须相同。

图 6.2 计数量具数据

Attribute Gau...
锁定文件 C:\Program
计数图

列 (7/0)
部件
标准
代码
A
B
C
参考值

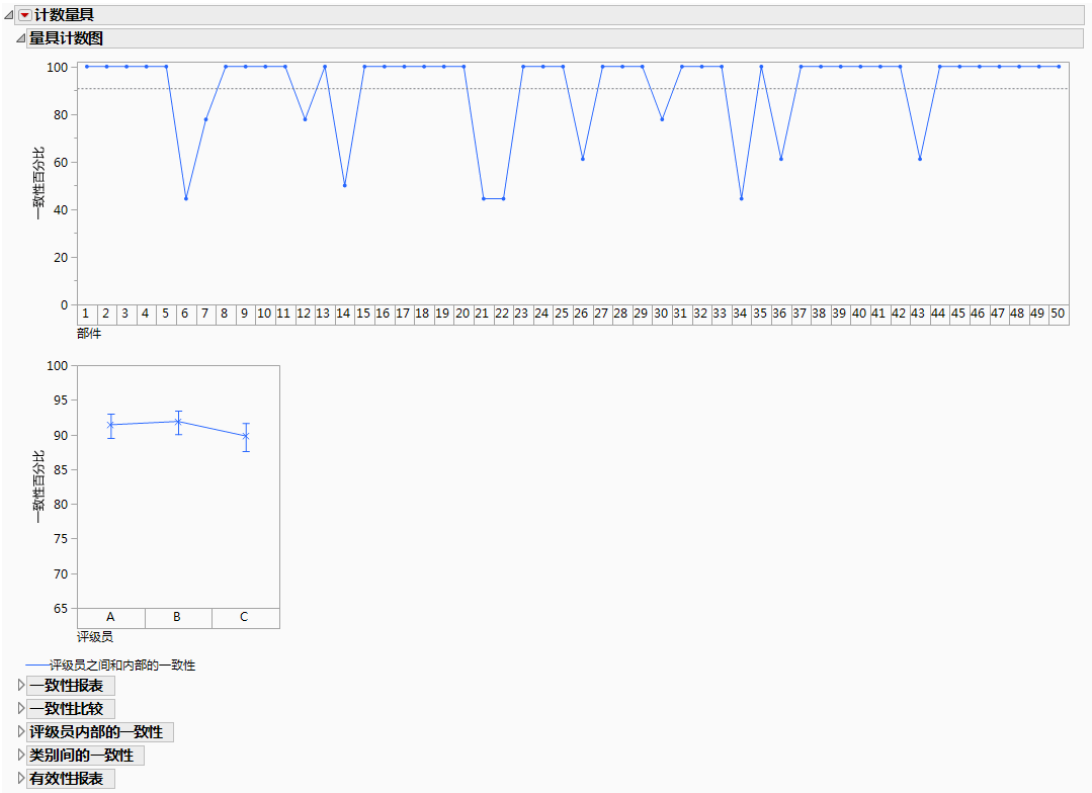
	部件	标准	代码	A	B	C	参考值
1	1	1	+	1	1	1	0.476901
2	1	1	+	1	1	1	0.476901
3	1	1	+	1	1	1	0.476901
4	2	1	+	1	1	1	0.509015
5	2	1	+	1	1	1	0.509015
6	2	1	+	1	1	1	0.509015
7	3	0	-	0	0	0	0.576459
8	3	0	-	0	0	0	0.576459
9	3	0	-	0	0	0	0.576459
10	4	0	-	0	0	0	0.566152
11	4	0	-	0	0	0	0.566152

计数量具图示例

假定您的数据中包含部件通过或失败的评级信息。三位评级员 （分别标识为 A、 B 和 C）各自记录了 50 个部件的评级信息：0 （通过）或 1 （失败），且每个部件记录三次。您想要检查以下方面：评级员对部件正确分类的有效程度、评级员相互之间的一致程度，以及评级员在各次评级过程中保持自我一致的程度。

1. 选择**帮助** > **样本数据库**，然后打开 Attribute Gauge.jmp。
2. 选择**分析** > **质量和过程** > **变异性 / 计数量具图**。
3. 对于**图表类型**，选择**计数**。
4. 选择 A、 B 和 C，然后点击 **Y，响应**。
5. 选择**标准**并点击**标准**。
6. 选择**部件**并点击 **X，分组**。
7. 点击**确定**。

图 6.3 计数图示例



第一个图（部件）显示评级员针对每一个部件相互之间保持一致的程度。例如，从该图上您可以看到部件 6、12、14、21、22 等的一致性百分比有所下降。这些部件可能较难分类。

第二个图（评级员）显示就给定的部件，每个评级员本人各次评价以及与其他评级员之间的一致性（评级已针对所有部件进行了汇总）。在本例中，看起来评级员的表现比较相似。评级员 C 的一致性最低，但差异不显著（约为 89%，而不是 91%）。

8. 打开 “有效性报表” 并向下滚动到 “合格性报表”。

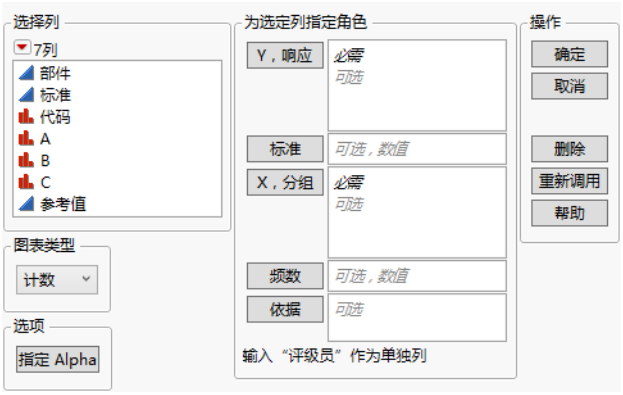
   您可以看到 0 = 不合格 （失败），1 = 合格 （通过）。不过在该数据中，情况正好相反：0 表示通过，1 表示失败。反转该设置。

9. 点击 “合格性报表” 红色小三角并选择**更改合格类别**。

启动“变异性 / 计数量具图”平台

通过选择**分析 > 质量和过程 > 变异性 / 计数量具图**启动 “变异性 / 计数量具图”平台。将**图表类型**设置为**计数**。

图 6.4 “变异性 / 计数量具图”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

图表类型 在变异性量具分析（对于连续响应）和计数量具分析（对于分类响应，通常为“通过”或“失败”）之间作出选择。

注意：本章的内容仅涉及**计数**图表类型。有关**变异性**图类型的详细信息，请参见第 115 页的“变异性量具图”一章。

指定 Alpha 指定该平台使用的 alpha 水平。

Y，响应 指定每个评级员给出的评级所在的列。您必须指定多个评级列。

标准 指定一个标准或参考列，其中包含部件的“真实”或已知值。在报表窗口中，将显示“有效性报表”，并且“一致性比较”报表中也会显示一个附加部分，用来显示评级员与标准的比较结果。

X，分组 指定用于对测量值分组的分类列。若因子构成嵌套层次，请先指定高层次的项。

频数 标识数据表列，该列的值为每行分配频数。这在有汇总数据时很有用。

依据 标识用于创建报表的一个列，该报表包括变量的每个水平的单独分析。

计数量具图和报表

计数量具图标绘“一致性百分比”，该百分比用来测量研究中每个部件对应的评级员一致性程度。每个部件的一致性是通过将每对评级员对该部件的评级与该部件的全部评级加以比较而计算得出的。请参见第 148 页的“计数量具图的统计详细信息”。

按第 141 页的“计数量具图示例”中的说明操作以生成图 6.5 中所示的结果。

图 6.5 计数量具图

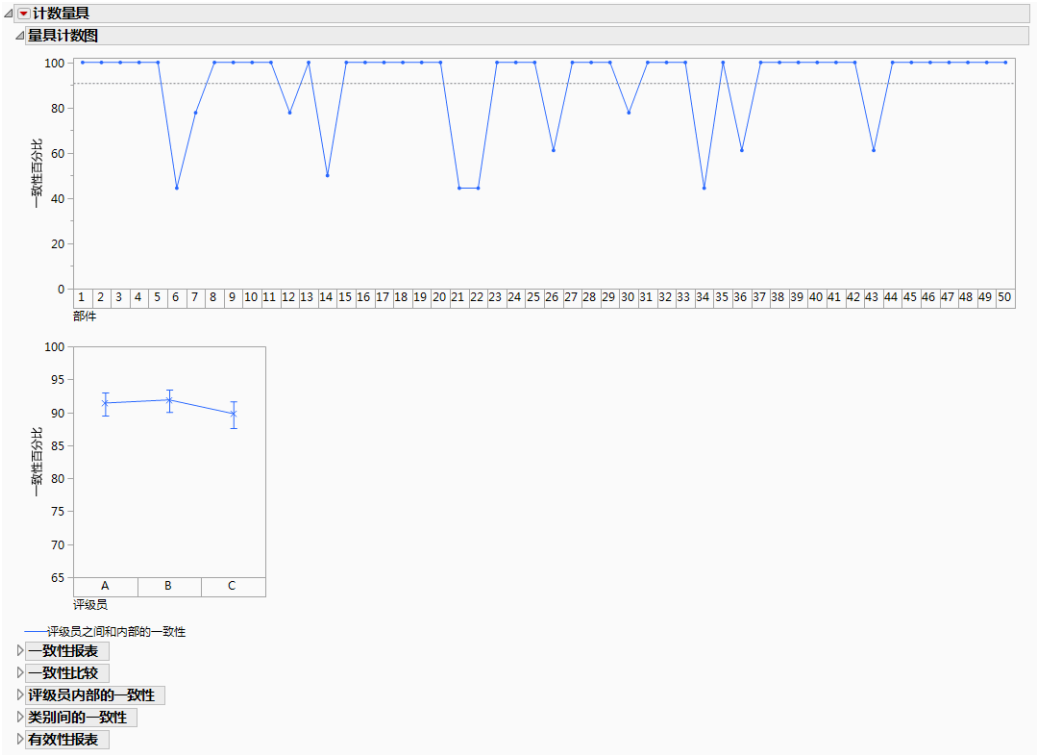


图 6.5 中的第一个图在 *x* 轴上使用所有 “X 分组”变量 （在该情况下为 “部件”）。第二个图在 *x* 轴上使用所有 Y 变量 （在该情况下为 “评级员”，通常也是如此）。

- 在第一个图中，您可以查找“一致性百分比”值较低的部件，调查以确定评级员为何对该特定部件给出的测量值不一致。
- 在第二个图中，您可以查找“一致性百分比”值较低的评级员，调查以确定这些评级员为何与其他评级员或是自我评级不一致。

有关其他选项的信息，请参见第 147 页的“ “计数量具”平台选项”。

一致性报表

注意：Kappa 值是一个用来表示一致性的统计量。Kappa 值越接近 1，一致性就越高。Kappa 值越接近 0 则指示一致性越低。

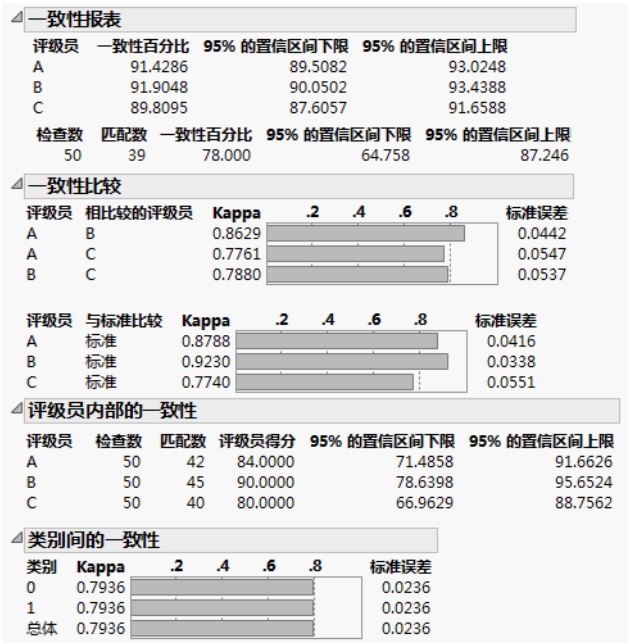
“一致性报表”显示针对每个评级员汇总的一致性以及总体一致性。该报表是在“计数量具图”报表的第二个图中表示的数据的数值形式（图 6.5）。

“一致性比较”报表使用 Kappa 统计量显示每个评级员与其他所有评级员的比较结果。仅当您在启动窗口中指定了“标准”变量后，才会将评级员与该标准加以比较。

“评级员内部的一致性”报表显示所检查的项数。置信区间为得分置信区间，正如 Agresti and Coull (1998) 所建议的那样。“匹配数”是检查的项数总和，其中每次检查单个项，评级员本人都给出了相同的评级。“评级员得分”是“匹配数”除以“检查数”后得到的结果。

“类别间的一致性”报表显示随机情况下期望的分类的一致性。它评估的是固定数目的评级员在对项分类时相互之间的一致程度。

图 6.6 一致性报表



有效性报表

仅当您在启动窗口中指定了“标准”变量后，才会显示“有效性报表”。有关“标准”变量的说明，请参见第 143 页的“启动“变异性/计数量具图”平台”。该报表将每个评级员与标准进行比较。

图 6.7 有效性报表

有效性报表

一致性计数

评级员	正确 (0)	正确 (1)	正确值合计	不正确 (0)	不正确 (1)	总计
A	45	97	142	3	5	150
B	45	100	145	3	2	150
C	42	93	135	6	9	150

有效性

评级员	有效性	95% 的置信区间下限	95% 的置信区间上限	误差率
A	94.6667	89.8296	97.2730	0.0533
B	96.6667	92.4348	98.5680	0.0333
C	90.0000	84.1565	93.8459	0.1000
总体	93.7778	91.1542	95.6603	0.0622

误分类

标准水平	0	1
0	.	16
1	12	.
其他	0	0

合格性报表

评级员	P (误报)	P (漏报)	假定
A	0.0490	0.0625	不合格 = 0
B	0.0196	0.0625	合格 = 1
C	0.0882	0.1250	

“一致性计数”表显示关于标准的每个水平的正确数和不正确数的单元格计数。在图 6.7 中，标准变量有两个水平：0 和 1。评级员 A 对于水平 0 有 45 个正确响应和 3 个不正确响应，对于水平 1 有 97 个正确响应和 5 个不正确响应。

有效性定义为正确决策数与决策机会总数之比。例如，假定评级员 A 对抽样的每个部件都评级三次。对于第六个部件，有一个决策不一致（例如，通过、通过、失败）。另外两个决策仍计为正确决策。这一有效性定义不同于 MSA 第 3 版中的定义。根据 MSA 的定义，评级员 A 对部件六的全部三个机会都将计为不正确。单独包含所有检查可为您提供关于整个检查过程的更多信息。

在“有效性”表中，关于有效性给出了 95% 置信区间。这些为得分置信区间。事实证明，得分置信区间可提供更高的覆盖概率，特别是在观测接近边界的情况下。请参见 Agresti and Coull (1998)。

“误分类”表显示不正确标签。行表示标准的水平或接受的参考值。列包含评级员给出的水平。

合格性报表

“合格性报表”显示误报概率和漏报概率。仅当评级有两个水平时（如“通过”或“失败”，0 或 1），“合格性报表”才会显示。

请谨记以下说明：

误报 部件被判为不合格，但实际上却合格。

漏报 部件被判为合格，但实际上却不合格。

P（误报） 误认为不合格的部件数除以认定为合格的部件总数。

P（漏报） 误认为合格的部件数除以实际上不合格的部件总数。

“合格性报表”红色小三角菜单包含以下选项：

更改合格类别 反转被视为合格的响应类别。

计算逃逸率 计算逃逸率，即生产出不合格部件但未检测出来的概率。逃逸率的计算方式：过程生产出不合格部件的概率乘以漏报概率。您需要指定过程生产出不合格部件的概率，亦称“不合格概率”。

注意：在该平台中缺失值被视为单独的类别。为避免这一单独的类别，请从数据表中排除包含缺失值的行。

“计数量具”平台选项

“计数量具”红色小三角菜单包含以下选项：

计数量具图 显示或隐藏量具计数图和效率图。

显示一致性点 显示或隐藏图上的一致性点。

连接一致性点 连接图中的一致性点。

评级员一致性的置信区间 显示或隐藏效率图上评级员一致性的置信区间。

显示一致性组均值 显示或隐藏量具计数图上的一致性组均值。在您指定了多个“X，分组”变量后，该选项才可用。

显示一致性总均值 显示或隐藏量具计数图上的一致性总均值。

显示有效性点 显示或隐藏图上的有效性点。

连接有效性点 在图中的有效性点之间绘制连接线。

评级员有效性的置信区间 在“计数量具图”报表的第二个图中显示或隐藏置信区间（图 6.5）。

有效性报表 显示或隐藏有效性报表。该报表使用 Kappa 统计量将每个评级员与标准进行比较。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

计数量具图的统计详细信息

对于图 6.5 中在 *x* 轴上标绘所有 **X，分组**变量的第一个图而言，“一致性百分比”计算如下：

$$\text{部件 } i \text{ 的一致性百分比} = \frac{\sum_{l=1}^{k} \binom{l \text{ 水平的响应数}}{2}}{\binom{N_i}{2}}$$

对于在图 6.5 中在 x 轴上标绘所有 **Y，响应**变量的第二个图而言，“一致性百分比”计算如下：

$$\text{评级员 } k \text{ 的一致性百分比} = \frac{\sum_{i=1}^{n}\left(\sum_{j=1}^{r_i} \begin{matrix}\text{该评论员 } k \text{ 在部件 } i \text{ 内针对重复次数 } j \text{ 的} \\ \text{未计匹配水平的数目}\end{matrix}\right)}{\sum_{i=1}^{n}\left(\sum_{j=1}^{r_i} N_i - j\right)}$$

请注意以下几点：

- n = 部件（分组变量）的数目
- r_i = 部件 i $(i = 1,...,n)$ 的重复次数
- m = 评级员数
- k = 水平数
- $N_i = m \times r_i$。针对部件 i $(i = 1,...,n)$ 的评级数。其中包含所有评级员的响应，以及针对某部件的重复评级次数。例如，若 3 个评级员每人给部件 i 测量了 3 次，则 N_i 为 3 x 3 = 9。

例如，请考虑下面这个包含三个评级员的数据表，每个评级员都有关于一个部件的三次重复。

表 6.1 评级员 A、B 和 C 的三次重复

	A	B	C
1	1	1	1
2	1	1	0
3	0	0	0

使用该表，您可以执行以下计算：

$$\text{一致性百分比} = \frac{\binom{4}{2} + \binom{5}{2}}{\binom{9}{2}} = \frac{16}{36} = 0.444$$

$$\text{一致性百分比 [评级员 A]} = \text{一致性百分比 [评级员 B]} = \frac{4+3+3}{8+7+6} = \frac{10}{21} = 0.476$$

$$\text{一致性百分比 [评级员 C]} = \frac{4+3+2}{8+7+6} = \frac{9}{21} = 0.4286$$

一致性报表的统计详细信息

简单 Kappa 系数是一个用于测量评级员之间一致性的测度。

$$\hat{\kappa} = \frac{P_0 - P_e}{1 - P_e}$$

其中：

$$P_0 = \sum_i p_{ii}$$

并且：

$$P_e = \sum_i p_{i.} p_{.i}$$

若将两个响应变量视为 n 个部件的两个独立评级，则当评级员之间完全一致时，Kappa 系数等于 +1。当观测到的一致性超过机会一致性时，Kappa 系数为正，其量值反映了一致性的强度。当观测到的一致性小于机会一致性时，Kappa 系数为负，尽管这在实际中并不常见。Kappa 的最小值介于 -1 到 0 之间，具体取决于边缘比例。

使用以下等式估计简单 Kappa 系数的渐近方差：

$$\text{方差} = \frac{A + B - C}{(1 - P_e)^2 n}$$

其中：

$$A = \sum_i p_{ii} \left[1 - (p_{i.} + p_{.i})(1 - \hat{\kappa}) \right]$$

$$B = (1 - \hat{\kappa})^2 \sum_{i \neq j} \sum p_{ij} (p_{.i} + p_{j.})^2$$

并且：

$$C = \left[\hat{\kappa} - P_e(1-\hat{\kappa})\right]^2$$

将会标绘 Kappa 系数，同时给出标准误差。

注意：即便变量水平不平衡，也会在“计数图”平台中显示 Kappa 统计量。

“类别间的一致性”报表中提供分类 Kappa 统计量 (Fleiss 1981)。

假设如下：

- n = 部件（分组变量）的数目
- m = 评级员数
- k = 水平数
- r_i = 部件 i (i = 1,...,n) 的重复次数
- $N_i = m \times r_i$。针对部件 i (i = 1, 2,...,n) 的评级数。其中包含所有评级员的响应，以及针对某部件的重复评级次数。例如，若 2 个评级员每人给部件 i 测量了 3 次，则 N_i 为 3 x 2 = 6。
- x_{ij} = 针对部件 i (i = 1, 2,...,n) 的水平 j (j=1, 2,...,k) 的评级数

各类别的 Kappa 定义如下：

$$\hat{\kappa}_j = 1 - \frac{\sum_{i=1}^{n} x_{ij}(N_i - x_{ij})}{(\bar{p}_j\bar{q}_j)\sum_{i=1}^{n} N_i(N_i-1)} \quad \text{其中} \quad \bar{p}_j = \frac{\sum_{i=1}^{n} x_{ij}}{\sum_{i=1}^{n} N_i} \quad \bar{q}_j = 1 - \bar{p}_j$$

总 Kappa 定义如下：

$$\hat{\bar{\kappa}} = \frac{\sum_{j=1}^{k} \bar{q}_j\bar{p}_j\hat{\kappa}_j}{\sum_{j=1}^{k} \bar{p}_j\bar{q}_j}$$

$\hat{\kappa}_j$ 和 $\hat{\bar{\kappa}}$ 的方差计算如下：

$$\text{方差}(\hat{\kappa}_j) = \frac{2}{nN(N-1)}$$

$$\text{方差}(\hat{\bar{\kappa}}) = \frac{2}{\left(\sum_{j=1}^{k} \bar{p}_j \bar{q}_j\right)^2 nN(N-1)} \times \left[\left(\sum_{j=1}^{k} \bar{p}_j \bar{q}_j\right)^2 - \sum_{j=1}^{k} \bar{p}_j \bar{q}_j (\bar{q}_j - \bar{p}_j)\right]$$

仅当每个部件的评级数相等时才会显示 $\hat{\kappa}_j$ 和 $\hat{\bar{\kappa}}$ 的标准误差（例如，对于所有 i=1,…,n，N_i=N）。

第 7 章

过程能力

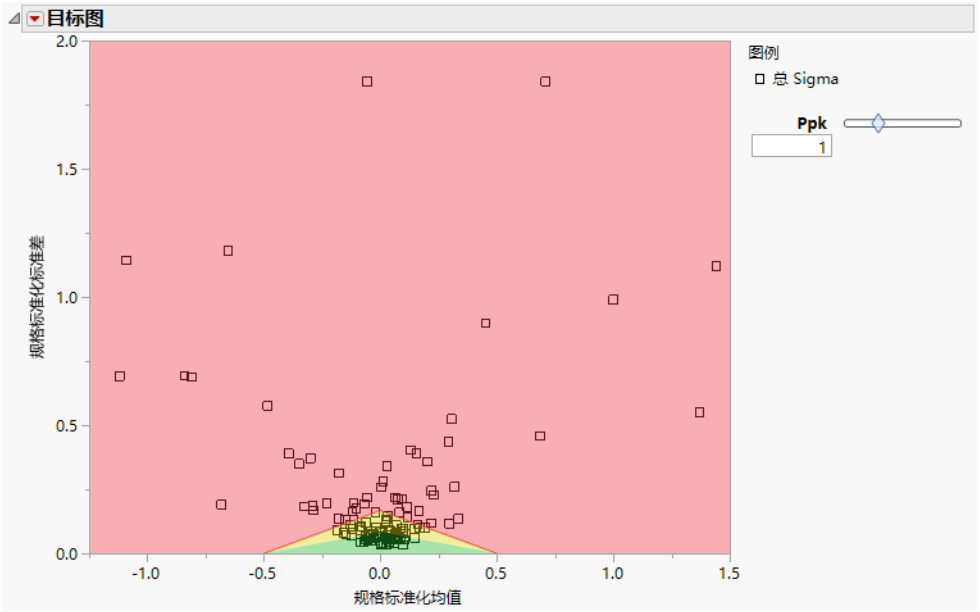
测量随时间推移过程的变异性

过程能力分析用在过程控制中，用来测量一个过程相对于给定规格限的表现情况。好的过程不仅稳定，并且能持续生产符合规格限的产品。能力指标测量相对于规格限的过程能力，它由过程中心和变异性来汇总。

目标图和箱线图之类的图形化工具为您提供了直观的方式，使您能够快速识别哪些过程或产品特征在规格限之内。单项详细报表为分析中的每个变量显示一个能力报表。该分析使您能够标识相对于规格或需求的变异，使合格值持续增加。

您可以指定子组以将过程的总变异与子组内变异相比较。可以为生成服从各种分布的测量值的过程计算能力指标。对于不服从任何指定分布的数据，您可以计算非参数能力指标。

图 7.1 “过程能力”平台示例



目录

“过程能力”平台概述 155
示例：包含正态变量的“过程能力”平台 156
包含非正态变量的“过程能力”平台的示例 158
启动“过程能力”平台 164
过程选择 165
过程子组 165
移动极差选项 166
历史信息 166
分布选项 167
其他指定 168
输入规格限 168
“规格限”窗口 168
限值数据表 169
规格限列属性 170
过程能力报表 171
目标图 172
能力箱线图 174
能力指标图 176
“过程能力”平台选项 178
单项详细报表 180
标准化箱线图 187
过程能力图 188
汇总报表 189
制作目标图汇总表 191
“过程能力”平台的更多示例 191
稳定过程的过程能力 191
不稳定过程的过程能力 195
模拟非正态过程 Ppk 的置信限 198
“过程能力”平台的统计详细信息 203
变异统计量 204
用于目标图和能力箱线图的符号 207
目标图 207
缺失目标的过程的能力箱线图 209
正态分布的能力指标 209
非正态分布的能力指标：百分位数和 Z 得分方法 214
分布的参数化 215

“过程能力”平台概述

“过程能力”平台提供测量过程是否符合给定的规格所需的工具。默认情况下，JMP 为您使用正态分布拟合的变量显示目标图、能力箱线图和能力指标图。在能力指标图上标绘非正态变量的能力指标。您可以为分析中的变量添加标准化箱线图、汇总报表和单项详细报表。

可以通过以下几种方式提供规格限：

- 在数据表中，使用列属性
- 在启动窗口中请求“规格限对话框”
- 从规格限数据表中加载限值
- 使用“管理规格限”实用工具（分析 > 质量和过程 > 管理规格限）

您可以指定双侧、单侧或非对称规格限。

注意：“过程能力”平台对“能力”分析（可通过“分析”>“分布”以及通过“分析”>“质量和过程”>“控制图”访问）进行了重大扩展。

能力指标

能力指标是一个比值，它将过程能力与生产满足规格限的产品相关联。该指标将质量特征的均值和标准差的估计值与规格限相关联。组内能力估计值基于从子组内变异构造的标准差。总能力估计值使用根据所有过程数据构造的标准差的估计值。请参见第 209 页的“正态分布的能力指标”和第 204 页的“变异统计量”。

仅当与中心化或离散相关的过程**稳定**时，均值或标准差的估计值才会准确定义。因此，组内能力指标的解释要求过程离散很稳定。总能力指标的解释要求过程中心化和离散均稳定。

根据小样本构造的能力指标可能会有很大变异。“过程能力”平台为多数能力指标提供置信区间。使用这些置信区间可为质量特征的实际能力确定潜在值的范围。

注意： JMP PRO 若不提供置信区间（例如，对于非正态分布），您可以使用“模拟”功能构造置信区间。有关示例，请参见第 198 页的“模拟非正态过程 Ppk 的置信限”。

有关能力指标值的准则，请参见 Montgomery (2013)。最小推荐值为 1.33。六西格玛方法以获取更高的能力水平为目标，相对应就是极低的每百万部件缺陷率。

非正态过程的能力指标

“过程能力”平台为具有以下分布的过程测量值构造能力指标：正态、Beta、指数、Gamma、Johnson、对数正态、两正态混合、三正态混合、SHASH 和 Weibull 分布。“最佳拟合”选项确定这些分布中的最佳拟合，并为该拟合提供能力指标。该平台还提供“非参数拟合”选项，该选项可以提供能力的非参数估计值。

对于非正态方法，使用两种方法构造估计值：ISO/分位数方法（百分位数）和 Bothe/Z 得分方法（Z 得分）。有关这些方法的详细信息，请参见第 214 页的“非正态分布的能力指标：百分位数和 Z 得分方法”。

注意：单个响应的“过程能力”分析可通过“分析”>“质量和过程”>“控制图生成器”访问。不过，只有“过程能力”平台才提供非正态分布。

总 Sigma 估计值和组内 Sigma 估计值

“过程能力”平台中的多数能力指标可基于**总**（长期）变异和**子组内**（短期）变异的估计值进行计算。若过程稳定，这两个变异测度应产生类似的结果，因为总变异和子组内变异应相似。标准化箱线图和汇总表可使用总变异或子组内变异进行计算。有关为稳定和不稳定过程计算的能力指标的示例，请参见第 191 页的“‘过程能力’平台的更多示例”。

您可以在启动窗口内指定子组以便估计子组内变异。可以指定定义子组的列，或选择一个常数子组大小。对于所有这些方法，您可以选择使用无偏标准差的平均值或使用极差的平均值来估计过程变异。若不指定子组，“过程能力”平台将使用大小为 2 的子组的移动极差来构建过程变异的子组内估计值。最后，您可以指定一个历史 sigma 用作过程标准差的估计值。

能力指标符号

“过程能力”平台提供了两组能力指标。有关能力指标计算的详细信息，请参见第 209 页的“正态分布的能力指标”。

- Cpk、Cpl、Cpu、Cp 和 Cpm。这些指标基于过程标准差的子组内（短期）估计值。
- Ppk、Ppl、Ppu、Pp 和 Cpm。这些指标基于过程标准差的总（长期）估计值。注意，若过程不稳定，则过程标准差不存在。请参见 Montgomery (2013)。

“过程能力”平台使用恰当的 AIAG 符号来表示能力指标：Ppk 标签表示总变异估计值构造的指标，Cpk 表示基于子组内变异估计值构造的指标。

注意：默认选定“添加 AIAG (Ppk) 标签”平台首选项。您可以通过取消选定“过程能力”下的这一首选项，将报告方式改为仅使用 Cp 符号。

有关过程能力分析的详细信息，请参见 Montgomery (2013) 和 Wheeler (2004)。

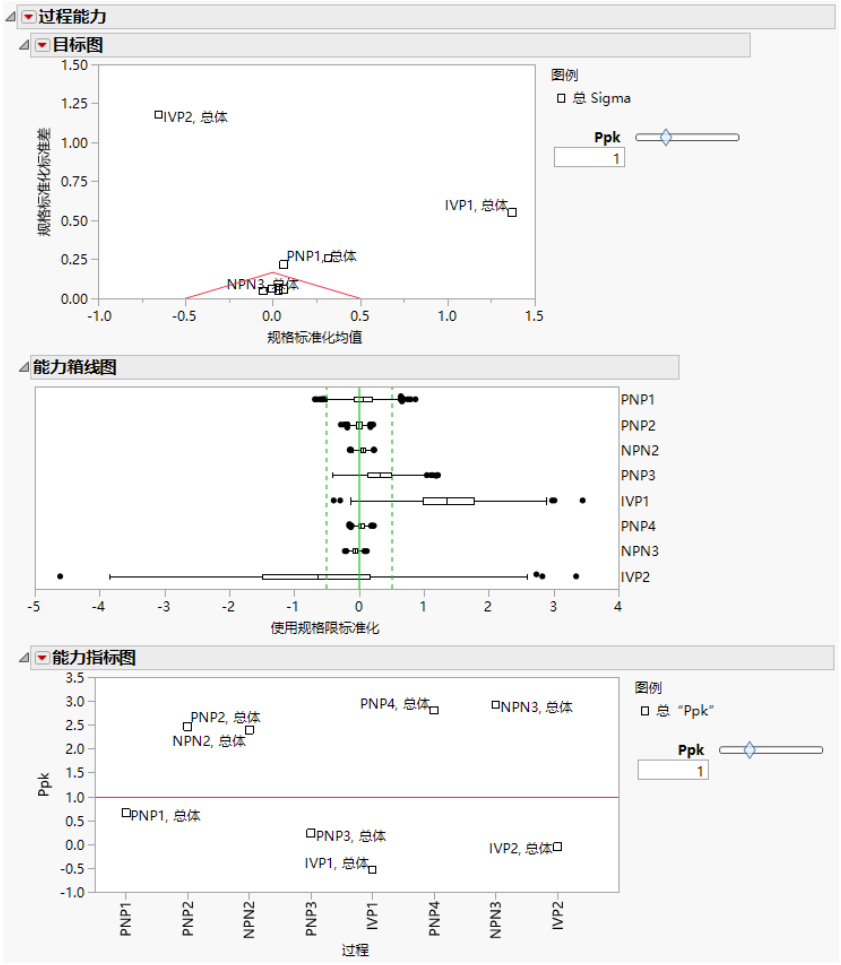
示例：包含正态变量的“过程能力”平台

本例使用 Semiconductor Capability.jmp 样本数据表。变量表示半导体生产商在加工晶片过程中可能采用的标准测量值。通过“列属性”>“规格限”属性在数据表中输入了变量的规格限。

1. 选择**帮助** > **样本数据库**，然后打开 Semiconductor Capability.jmp。
2. 选择**分析** > **质量和过程** > **过程能力**。
3. 点击**过程**旁边的白色小三角，查看所有连续变量。

4. 选择 PNP1、PNP2、NPN2、PNP3、IVP1、PNP4、NPN3 和 IVP2，然后点击 Y，过程。
5. 点击确定。
6. 点击 “目标图” 红色小三角并选择添加总 Sigma 点标签。
7. 点击 “能力指标图” 红色小三角并选择添加总 Sigma 点标签。

图 7.2 Semiconductor Capability.jmp 的示例结果



目标图为每个变量在 x 轴上显示规格标准化均值偏移，在 y 轴上显示规格标准化标准差。图下方中心处红线定义的三角区域是目标三角形。它定义了能力指标值的区域。您可以使用图右侧的 Ppk 滑块来调整目标三角形。当滑块设置为 1 时，请注意 PNP1、PNP3、IVP1 和 IVP2 均位于目标三角形之外，所以很可能超出规格范围。

“能力箱线图”报表为分析中的每个变量显示一个箱线图。每列的值以其目标值为中心，并以规格范围统一尺度。在本例中，所有过程变量都同时具有上规格限和下规格限，并且规格限关于目标值对称。在这里绿实线显示目标应该所处的位置，而虚线表示规格限。

看起来 IVP1 的多数点都位于其上规格限 (USL) 上方，而 IVP2 的多数点都小于其目标。PNP2 看起来位于目标上，并且所有数据值都位于规格限以内。

“能力指标图”绘制了每个变量的 Ppk 值。四个变量来自能力水平很高的过程，Ppk 值为 2 或更高。四个变量的 Ppk 值低于 1。

包含非正态变量的“过程能力”平台的示例

Process Measurements.jmp 数据表包含针对用于构造产品的七个不同过程得到的测量值。对于每个过程，规格限都另存为列属性。您最先要做的是检查您的过程数据的分布。可以看到分布不服从正态分布。然后您使用“过程能力”平台的非正态能力功能来计算能力指标。

查看分布

1. 选择**帮助** > **样本数据库**，然后打开 Process Measurements.jmp。
2. 选择**分析** > **分布**。
3. 从**选择列**列表中选择全部七列，然后点击 **Y，列**。
4. 选中**仅显示直方图**旁边的框。
5. 点击**确定**。

   对于多数过程，直方图都有证据表明测量值的理论分布有偏斜，不服从正态分布。因此，对于每个过程，您都会在所有可用参数分布中找到最佳拟合分布。

执行能力分析

1. 选择**分析** > **质量和过程** > **过程能力**。
2. 从**列**列表中选择全部七列，然后点击 **Y，过程**。
3. 选择 **Y，过程**列表中的全部七列。
4. 打开**分布选项**面板，然后从**分布**列表中选择**最佳拟合**。
5. 点击**设置过程分布**。

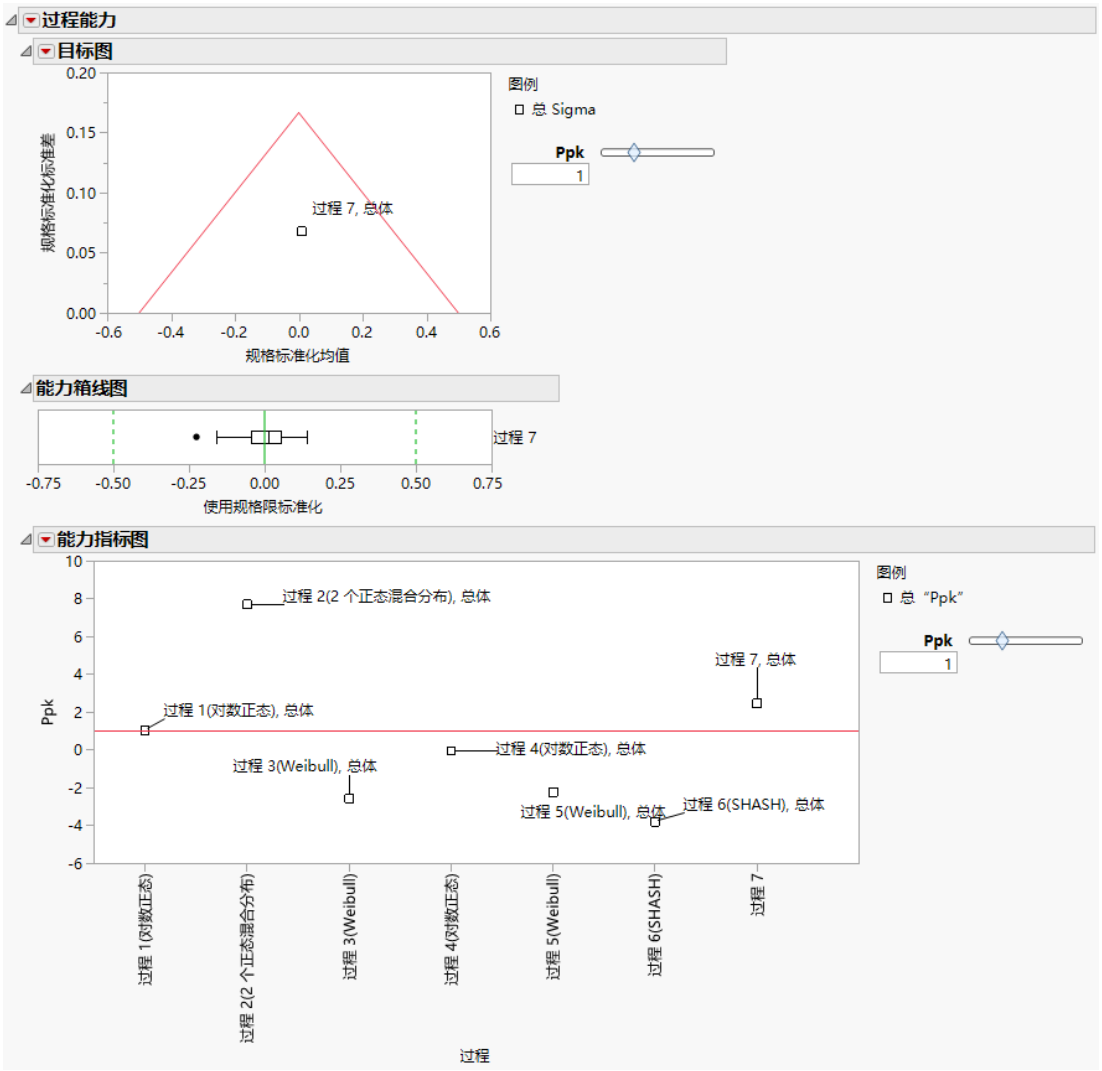
   后缀 **& 分布 (最佳拟合)** 添加至 “Y，过程”列表中的每个变量名。“最佳拟合”选项指定应该为每个变量拟合最佳拟合参数分布。可用的参数分布包括：正态、Beta、指数、Gamma、Johnson、对数正态、两正态混合、三正态混合、 SHASH 和 Weibull 分布 （图 7.3）。
6. 打开**非正态分布选项**分级显示项。请注意，“非正态能力指标方法”设置为**百分位数**，“Johnson 分布拟合方法”设置为**分位数匹配**，“分布比较准则”设置为 AICc。

图 7.3 完成的启动窗口

“分位数匹配”方法是用于拟合 Johnson 分布的默认方法，因为它有着优越于 “最大似然”的稳定性和速度。请注意，“最大似然”用在 “分布”平台中。

7. 点击**确定**。
8. 点击 “目标图”红色小三角并选择**添加总 Sigma 点标签**。
9. 点击 “能力指标图”红色小三角并选择**添加总 Sigma 点标签**。

图 7.4 已添加变量标签的初始报表



注意：点击图中某个标签并拖动它以便让该图更容易解释。点击 “能力指标图”的右侧框架，将其拖至右侧，以便让标签更易区分。

“目标图”仅显示一个点，并且该点对应**过程 7**。“能力箱线图”报表为**过程 7** 显示了一个箱线图。这是因为**过程 7** 的最佳拟合是正态分布。

10. 在 “能力指标图”右侧，将 Ppk 值设置为 2。

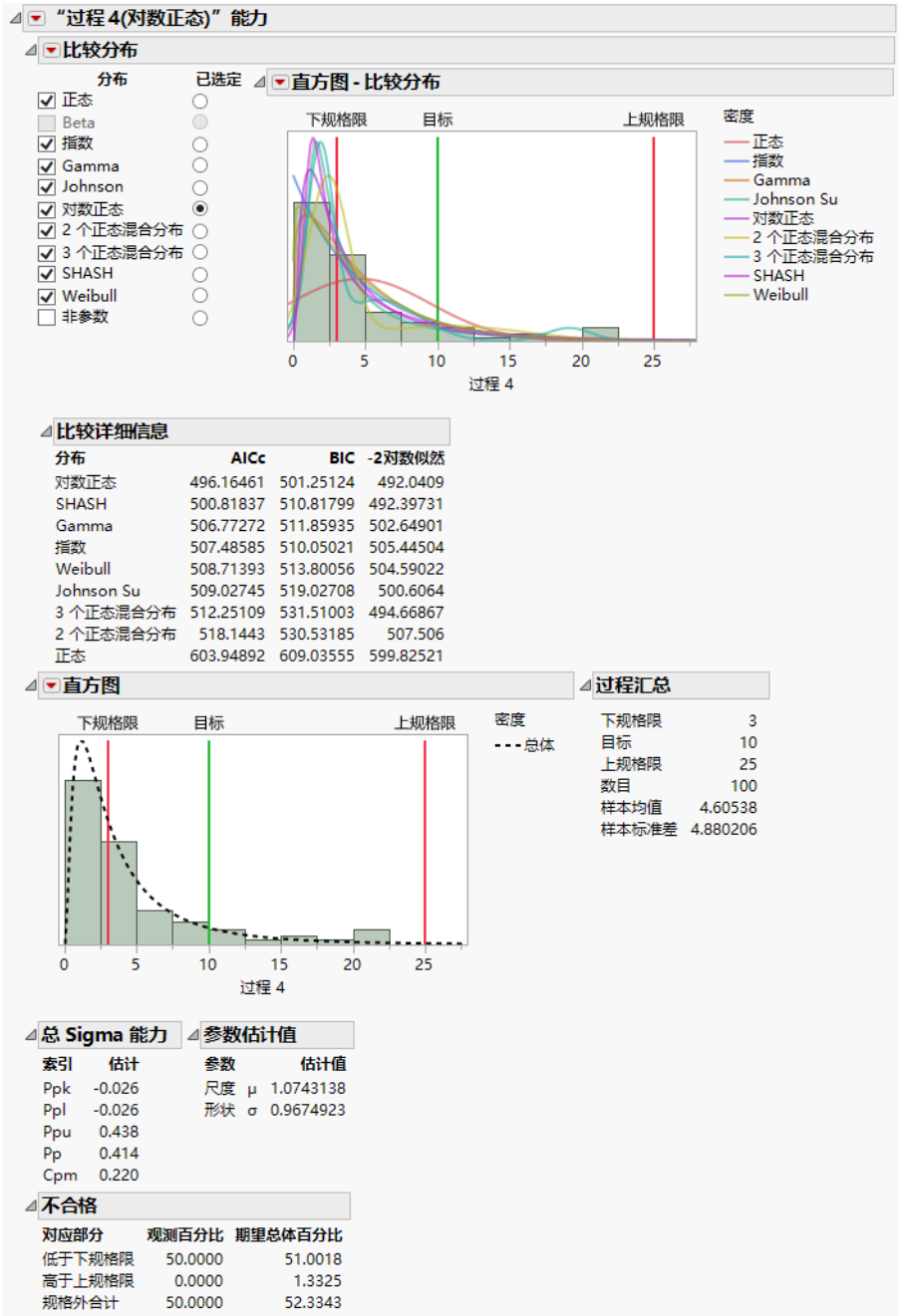
    “能力指标图”显示全部七个过程的 Ppk 值。只有两个过程 — **过程 2** 和**过程 7** — 的能力值超过 2。请注意，最佳拟合非正态分布显示在 “能力指标图”中变量名称右侧的括号中。**过程 7** 的最佳拟合分布由于是正态分布而未加以显示。

11. 点击 “过程能力”红色小三角并选择**单项详细报表**。

    由于您在启动窗口中请求了“最佳拟合”，所以从每个分布的红色小三角菜单中选定了“比较分布”选项。

12. 滚动至名为 **“过程 4(对数正态)”能力**的报表。

图 7.5 “过程 4”的单项详细报表



过程 4 的报表标题表明能力计算基于对数正态拟合。“比较分布”报表中除 “非参数”和 “Beta”复选框之外的所有复选框均已选中，这表示拟合了这九个分布。（这是因为您在启

动窗口中请求了 “最佳拟合”。）在 “已选定”列中选定的按钮指示 “对数正态”分布是用在 “过程 4(对数正态) 能力”报表中其余部分的分布，用来估计能力和不合格。

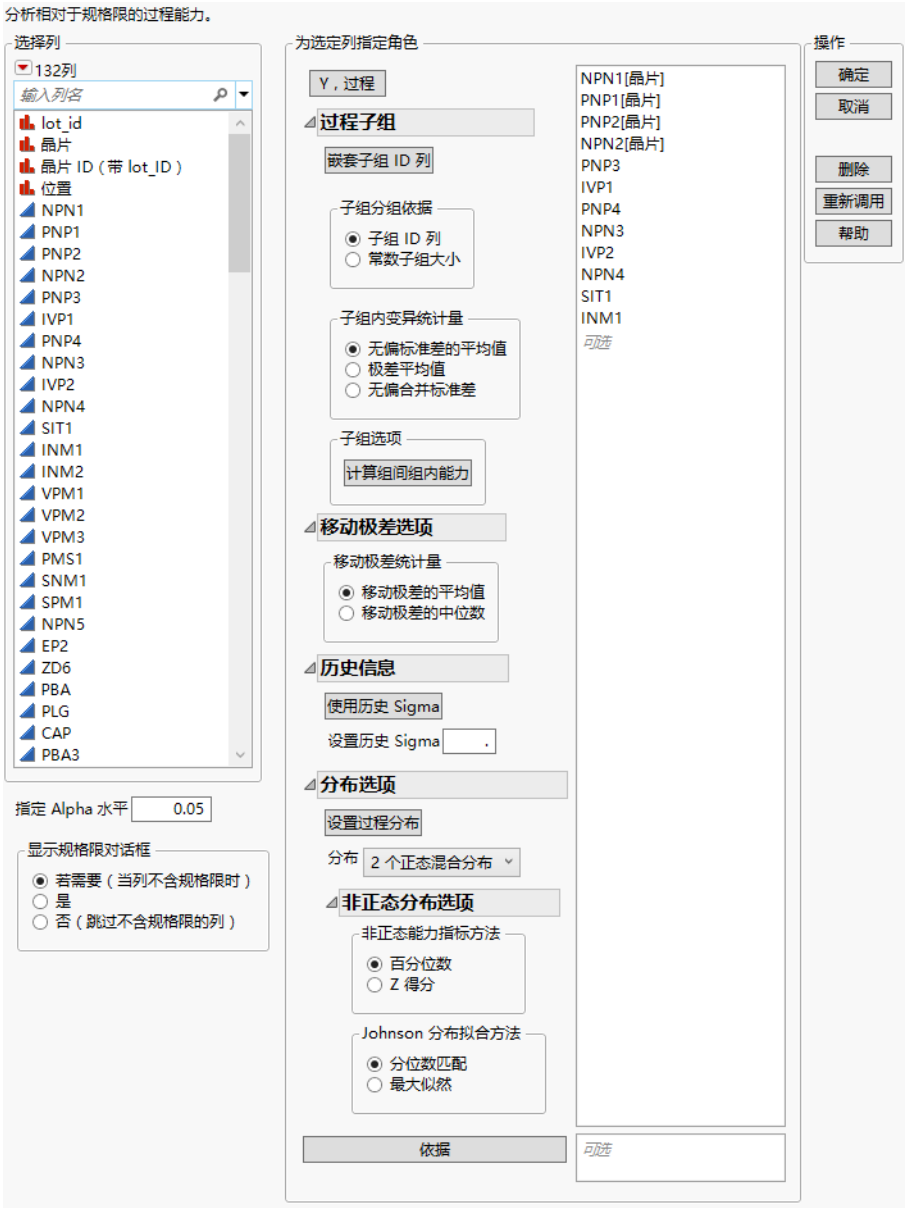
“比较分布”报表支持您比较九个分布拟合。“直方图 - 比较分布”报表给出拟合的直观估计，“比较详细信息”报表显示选定分布的拟合统计量。无论是图还是拟合统计量都指示对数正态分布在所有选定的分布中可提供最佳拟合。

默认显示的 “单项详细报表”信息包含一个显示估计的最佳拟合分布的直方图、过程信息的汇总、基于总 Sigma 估计值的能力指标、拟合对数正态分布的参数估计值，以及观测到的和预期的不合格水平。

启动“过程能力”平台

通过选择 “分析” > “质量和过程” > “过程能力”启动 “过程能力”平台。在图 7.6 中 （该图使用 Semiconductor Capability.jmp 数据表），所有分级显示项和面板均已打开。

图 7.6 “过程能力”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“过程能力”启动窗口包含以下分级显示项和选项：

- 第 165 页的“过程选择”
- 第 165 页的“过程子组”
- 第 166 页的“移动极差选项”
- 第 166 页的“历史信息”
- 第 167 页的“分布选项”
- 第 168 页的“其他指定”

在启动窗口中点击 “确定”后，除非出现以下情况之一，否则将显示 “规格限”窗口：

- 所有列都包含规格限。
- 您在启动窗口中选定了**否（跳过不含规格限的列）**。

若您在启动窗口中选择“是”，还会出现“规格限”窗口。否则会显示“过程能力”报表窗口。

过程选择

选择要包括在能力分析中的过程变量。

Y，过程 指定要分析的变量。

注意：

- “变换”菜单不可用于“过程能力”启动窗口中的“选择列”列表。右击数据表中的列标题，然后选择**新建公式列**创建要用在过程能力中的变换列。请参见《使用 JMP》，了解有关创建新公式列的详细信息。
- 虚拟连接表的引用列不可用于“过程能力”平台。

过程子组

这组选项支持您为“Y，过程”列表中的每个变量分配一个子组 ID 列或常数子组大小。

使用 ID 列创建子组

1. 选择“Y，过程”列表中的一个或多个变量。
2. 从**子组分组依据**选项中选择**子组 ID 列**。
3. 在 “选择列”列表中选择一个子组 ID 列。
4. 点击**嵌套子组 ID 列**。

子组 ID 列显示在“Y，过程”列表中变量名右侧的括号内。

使用常数子组大小创建子组

1. 选择“Y，过程”列表中的一个或多个变量。
2. 从**子组分组依据**选项中选择**常数子组大小**。
3. 在**设置常数子组大小**旁边输入子组大小。
4. 点击**按大小划分的子组**。

子组大小显示在 “Y，过程”列表中变量名右侧的括号内。

嵌套子组 ID 列 （在您选择 “子组 ID 列”时可用。）分配您从 “选择列”列表中选择的列，以便为选定的 “Y，过程”列定义子组。

按大小划分的子组 （在您选择“常数子组大小”时可用。）分配您在“设置常数子组大小”框中指定的子组大小，以便为选定的 “Y，过程”列定义子组。

设置常数子组大小 （在您选择 “常数子组大小”时可用。）为选定的 “Y，过程”列指定常数子组大小。您需要使用 “按大小划分的子组”分配该值。

子组内变异统计量 （在使用“过程子组”时可用。）指定是否使用标准差或极差计算标准差的子组内估计值。

计算组间和组内能力 （在使用“过程子组”时可用。）指定应该在能力分析中使用标准差的组间和组内子组估计值。

移动极差选项

未使用子组时，使用该分级显示项可指定在组内 sigma 估计中使用哪个移动极差统计量。

注意：指定子组并点击**计算组间和组内能力**时，请使用“移动极差选项”分级显示项来指定在组内 sigma 估计中使用的移动极差统计量。

移动极差的平均值 使用移动极差均值估计 sigma。移动极差是两个相邻点之间的差值。

移动极差的中位数 使用移动极差的中位数估计 sigma。

历史信息

使用该分级显示项可以向 “Y，过程”列表中的变量分配以往接受的标准差的值。

1. 选择“Y，过程”列表中的一个或多个变量。
2. 在 “设置历史 Sigma”旁边输入一个值。
3. 选择 “使用历史 Sigma”将该值分配给选定变量。

指定的值显示在“Y，过程”列表中变量名右侧的表达式“&Sigma()”中的括号内。

注意：若设置历史 sigma，则针对选定过程变量的子组分配将不再相关，所以将被删除。

分布选项

除非另外指定，分析所有“Y，过程”变量的假设前提是：所有变量都服从正态分布。使用“分布选项”分级显示项向 “Y，过程”列表中的变量分配其他分布或计算方法，并且指定与非正态计算相关的选项。

- 可用分布包括：正态、Beta、指数、Gamma、Johnson、对数正态、两正态混合、三正态混合、SHASH 和 Weibull 分布。除 Johnson 分布之外，使用最大似然估计来拟合分布。请参见第 167 页的“Johnson 分布拟合方法”。
- “最佳拟合”选项确定可用分布中的最佳拟合并且应用该拟合。
- “非参数”选项使用核密度估计拟合分布。

有关与非正态拟合相关的更多选项，请参见第 167 页的“非正态分布选项”。

指定分布

1. 选择“Y，过程”列表中的一个或多个变量。
2. 从 “分布”列表中选择一种分布。
3. 选择 “设置过程分布”将该分布分配给选定的变量。

指定的分布显示在“Y，过程”列表中变量名右侧的表达式“&分布()”中的括号内。

注意：若选择“正态”之外的分布，则您无法分配“子组 ID”列或“历史 Sigma”。用于计算非正态能力指标的方法不支持以上选择。请参见第 214 页的“非正态分布的能力指标：百分位数和 Z 得分方法”。

非正态分布选项

非正态能力指标方法 指定用于计算非正态分布的能力指标的方法。请参见第 214 页的 “非正态分布的能力指标：百分位数和 Z 得分方法”。

Johnson 分布拟合方法 指定用于查找最佳拟合 Johnson 分布的方法。估计参数之前，需要从 Johnson Su、Sb 和 Sl 系列中确定最佳拟合分布系列。Slifker and Shapiro (1980) 中所述的过程用于查找最佳拟合系列。

分位数匹配 默认方法。比最大似然更稳定更快速。分位数匹配参数估计值（假定采用最佳拟合系列）通过使用分位数匹配方法来获取。请参见 Slifker and Shapiro (1980)。

最大似然 最佳拟合系列的参数使用最大似然确定。

分布比较准则 （在选定“最佳拟合分布”时可用。）指定要用于确定最佳拟合的准则。该准则还确定 “比较详细信息”报表中分布的顺序。请参见第 187 页的 “排序依据比较准则”。

其他指定

依据 为“依据”变量的每个水平生成单独报表。若分配了多个依据变量，则为依据变量水平的每种可能组合都生成一个单独的报表。

指定 Alpha 水平 指定置信限的显著性水平。

显示规格限对话框 指定如何处理没有规格限的列。

注意：最好确保所有过程变量的规格限都指定为“规格限”列属性，或从“限值数据”表中加载规格限（请参见第 169 页的“限值数据表”）。如若不然，您可以在点击启动窗口中的“确定”后出现的“规格限”窗口中以交互方式指定限值（除非您在启动窗口中选择**否（跳过不含规格限的列）**）。

输入规格限

下规格限 (LSL)、上规格限 (USL) 和目标定义质量过程的下限、上限和目标值。

可通过多种方式输入规格限：

- 在启动窗口中选定列后，在“规格限”窗口中输入限值。请参见第 168 页的“‘规格限’窗口”。
- 从 JMP 数据表（称为“限值表”）导入限值。请参见第 169 页的“限值数据表”。
- 输入限值，将其作为数据表中的“规格限”列属性。请参见第 170 页的“规格限列属性”。
- 若通过运行 JSL 脚本创建“过程能力”报表，则在该脚本中输入限值。请参见第 171 页的“过程能力报表”。

每个选定列只需一个规格限。若仅指定上规格限，则箱线图和目标图点为蓝色。若仅指定下规格限，则箱线图和目标图点为红色。

“规格限”窗口

在启动窗口中点击确定后，若任何一列不包含限值，而您也未在启动窗口中选择**否（跳过不含规格限的列）**，此时便会显示“规格限”窗口。若您在启动窗口中选择“是”，还会出现“规格限”窗口。图 7.7 显示在启动窗口中将**臭氧**、**一氧化碳**、**二氧化硫**和**一氧化氮**选作过程变量后，Cities.jmp 样本数据表的“规格限”窗口。输入已知的规格限，然后点击确定查看“过程能力”报表。

您可以为每列指定过程重要性值。过程重要性值提供按您喜欢的顺序对过程排序的机制。过程重要性值用于调整“过程能力”报表中的许多图形内的标记大小。

若为过程选择“显示限值”选项，然后将规格限保存到列属性，则将在保存的“规格限”列属性中选定“显示为图形参考线”选项。若为过程选择“显示限值”选项，并将规格限保存到新

表中，则新表中的“显示限值”列将为该过程包含 1。使用**选择全部显示限值**按钮将为所有过程选择“显示限值”选项。

图 7.7 Cities.jmp 的 “规格限”窗口

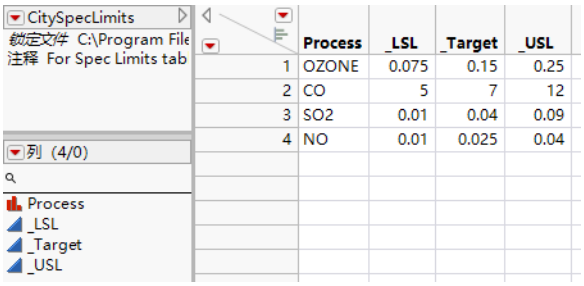
限值数据表

您还可以使用“规格限”窗口中的**从数据表加载规格限**选项指定限值数据表。点击**选择数据表**按钮，然后选择包含适用于分析的规格限的适当数据表。选择适当的限值表后，这些值将填充到该窗口中。点击**确定**查看“过程能力”报表。

限值数据表可采用两种不同的格式：**高型**或**宽型**。高型限值数据表将各个响应显示在一列中，其他各列分别显示限值关键字。宽型限值数据表让每个响应对应一列，再用一列添加限值关键字的标签。这两种格式均可使用**从数据表加载规格限**选项来读取。

- 高型表包含四或五列，每个过程对应一行。第一列的数据类型为字符型，其中包含在“过程能力”平台中分析的列的名称。接下来三列需分别命名为**下规格限**、**目标**和**上规格限**。这些列名之前还可以附加下划线字符。名为**显示限值**的最后一列（可选）指定规格限在某些分析图中是否显示为参考线。

图 7.8 示例：高型规格限表



	Process	_LSL	_Target	_USL
1	OZONE	0.075	0.15	0.25
2	CO	5	7	12
3	SO2	0.01	0.04	0.09
4	NO	0.01	0.025	0.04

- 宽型表对于“过程能力”平台中分析的每列都包含三行和一列，另有一个 _LimitsKey 列。在 _LimitsKey 列中，有三行需要包含标识符 _LSL、_Target 和 _USL。

图 7.9 示例：宽型规格限表

	_LimitsKey	OZONE	CO	SO2	NO
1	_LSL	0	5	0	0
2	_Target	0.05	10	0.03	0.025
3	_USL	0.1	20	0.08	0.6

最简单的限值数据表创建方法是保存“过程能力”平台计算的结果。“过程能力”红色小三角菜单中的“保存”选项支持您保存样本值中的限值。输入或加载规格限后，您可以执行以下操作：

- 选择**保存 > 将规格限保存为列属性**可将限值作为“规格限”列属性保存至数据表中的列。
- 选择**保存 > 将分布另存为列属性**可将计算能力时使用的分布另存为数据表中的列的“过程能力分布”列属性。
- 选择**保存 > 在新表中保存规格限**可将限值保存至新的高型规格限数据表。若您已选定至少一个非正态分布，包含指定分布的名为**分布**的列也会添加至限值数据表。

请参见第 178 页的“‘过程能力’平台选项”。

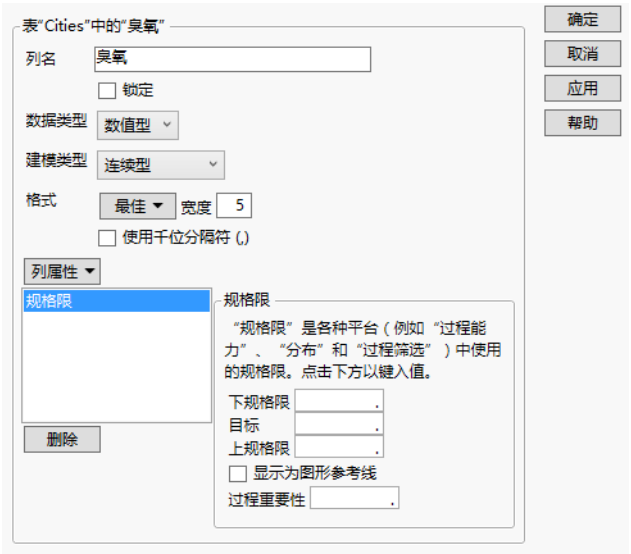
规格限列属性

执行能力分析时，您可以使用“列属性”>“规格限”将规格限另存为列属性。“规格限”属性仅应用于数值列。

某些过程具有单侧规格。有些过程则无目标。您可以输入以下适用的任意项：下规格限、上规格限、目标值或过程重要性值。

图 7.10 针对样本数据表 Cities.jmp 中的**臭氧**显示了“列属性”窗口的“规格限”部分。

图 7.10 “列属性”窗口的“规格限”部分



提示：将规格限另存为列属性可在重复执行分析时确保一致性。

过程能力报表

默认情况下，“过程能力”平台提供以下报表：

- 第 172 页的“目标图”（仅在至少有一个变量使用正态分布拟合时提供，并且仅显示使用正态分布拟合的变量对应的点）
- 第 174 页的“能力箱线图”（仅在至少有一个变量使用正态分布拟合时提供，并且仅显示使用正态分布拟合的变量对应的箱线图）
- 第 176 页的“能力指标图”

第 157 页上的图 7.2 显示了默认“过程能力”报表的示例。

使用过程能力红色小三角菜单，您可以添加单项详细报表、标准化箱线图，以及汇总报表。红色小三角菜单还包含一些选项，可用于标识数据表中的超出规格限的值、创建汇总数据表、更改分析列的显示顺序，以及保存规格外限值。第 178 页的“‘过程能力’平台选项”中对这些选项进行了说明。

您可以通过“文件”>“首选项”>“平台”>“过程能力”更改默认报表。对于通过“文件”>“首选项”>“平台”选择的相关“过程能力”主题选项所生成的报表，您还可以更改这些报表的外观。

目标图

目标图为每个变量在 *x* 轴上显示规格标准化均值偏移，在 *y* 轴上显示规格标准化标准差。该图可用于快速提供关于变量是否符合规格限的汇总视图。默认情况下，“目标图”仅显示每列使用总 sigma 计算的那些点。悬停在每个点上方可查看变量名以及用于计算点的 sigma 方法。有关“目标图”坐标计算的详细信息，请参见第 207 页的“目标图”。

注意：非“正态”分布的过程变量不会标绘在目标图上。

目标图点

目标图上各点对应的是列而不是行。在目标图中选择某个点也会选择数据表中的相应列。若指定了过程重要性值，则会根据重要性调整目标图点的大小。

悬停在“目标图”中的某个点上方以查看该过程的控制图。点击控制图以启动带有相应控制图和能力报表的“控制图生成器”。

注意：若将无偏合并标准差选作过程的组内变异统计量，则控制图不可用于该过程。

目标图上的点还链接至“目标图汇总表”中的行，其中每行都对应一列。您可以选择目标图中的某个点，右击并应用行状态。这些行状态应用于“目标图汇总表”中的行。您在“目标图汇总表”中应用的行状态反映在目标图中。要查看该表，请从“过程能力”红色小三角菜单中选择“制作目标图汇总表”。请参见第 191 页的“制作目标图汇总表”。

提示：若隐藏目标图中某个点，则可以通过更改“目标图汇总表”中的相应行状态再次显示该点。

目标图三角形

目标图三角形显示在目标图底部中央。图右侧的滑块支持您调整图中目标三角形的大小。

默认情况下，Ppk 滑块和其下的值设置为 Ppk = 1。这近似不合格率为 0.0027（若为正态分布）。目标三角形表示框中显示的 Ppk。要更改 Ppk 值，请移动滑块或在框中输入数字。

默认情况下，JMP 根据 Ppk 值提供“目标图”。您可以通过“文件”>“首选项”>“平台”>“过程能力”更改该首选项。若未选中“添加 AIAG (Ppk) 标签”首选项，则所有 Ppk 标签都将改为 Cpk 标签，包括目标图右侧滑块的标签。

目标图选项

“目标图”红色小三角菜单包含以下选项：

显示组内 Sigma 点　显示或隐藏使用组内 sigma 估计值计算的点。

显示组内或组间和组内 Sigma 点 （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏使用组内 sigma 估计值或组间和组内 sigma 估计值（若指定）计算的点。

显示总 Sigma 点 显示或隐藏使用总 sigma 估计值计算的点。

着色水平 显示或隐藏 Ppk 水平着色（图 7.11）。选择“着色水平”时，着色区域显示在图中。着色区域取决于 p 和 Ppk 之间的关系，其中的 p 表示 Ppk 之下的框中显示的值：

- 红色区域中的点：$\text{Ppk} < p$。
- 黄色区域中的点：$p < \text{Ppk} < 2p$。
- 绿色区域中的点：$2p < \text{Ppk}$。

添加组内 Sigma 点标签 显示或隐藏使用组内 sigma 估计值计算的点的标签。

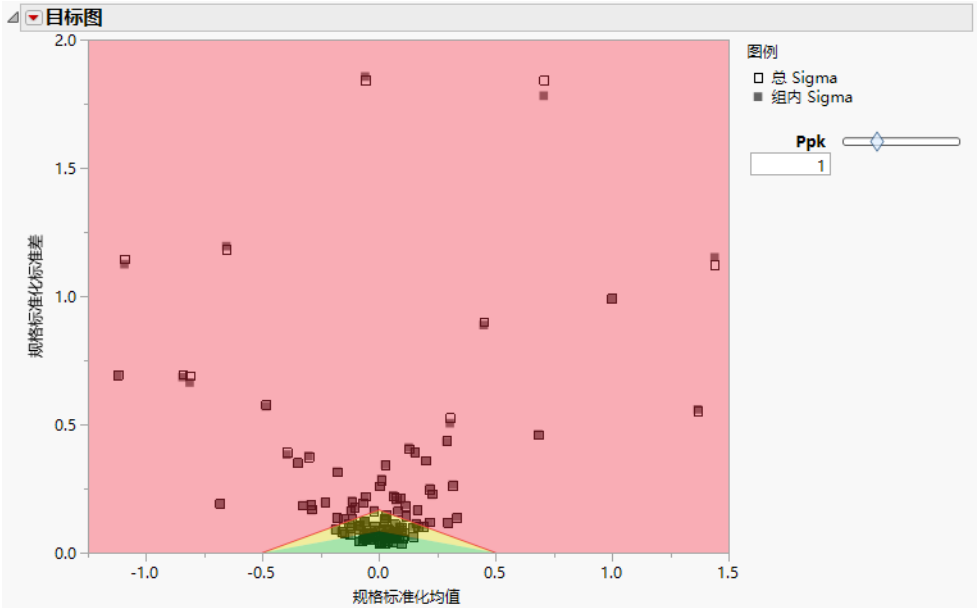
标记组内或组间和组内 Sigma 点 （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏使用组内 sigma 估计值或组间和组内 sigma 估计值（若指定）计算的点的标签。

添加总 Sigma 点标签 显示或隐藏使用总 sigma 估计值计算的点的标签。

缺陷率等高线 显示或隐藏表示指定缺陷率的等高线。

图 7.11 显示在从“目标图”红色小三角菜单中选择着色水平和显示组内 Sigma 点后，Semiconductor Capability.jmp 样本数据表的整个数据集的目标图。

图 7.11 目标图



单侧或缺失规格限

若某列只有一个规格限，则标记和颜色的使用方式如下：

- 若仅指定了上规格限 (USL)，目标图上的点通过指向右侧的三角形来表示且标记为蓝色。
- 若仅指定了下规格限 (LSL)，目标图上的点通过指向左侧的三角形来表示且标记为红色。
- 若至少有一个过程仅有上规格限，则目标三角形的右半部分为蓝色。
- 若至少有一个过程仅有下规格限，则目标三角形的左半部分为红色。

仅有上规格限的过程用蓝色表示，所以应该与目标三角形的蓝色（右）侧相比较。仅有下规格限的过程用红色表示，所以应该与目标三角形的红色（左）侧相比较。有关如何计算点坐标的详细信息，请参见第 207 页的“目标图”。

能力箱线图

能力箱线图为分析中选定的每个变量显示一个箱线图。每列的值以其目标值为中心，并以规格限之间的差值统一尺度。若目标不在规格限的中心，则值将按目标与规格限之间最小差值的两倍统一尺度。对于每个过程列 Y_j（请参见第 207 页的“用于目标图和能力箱线图的符号”，了解符号说明）：

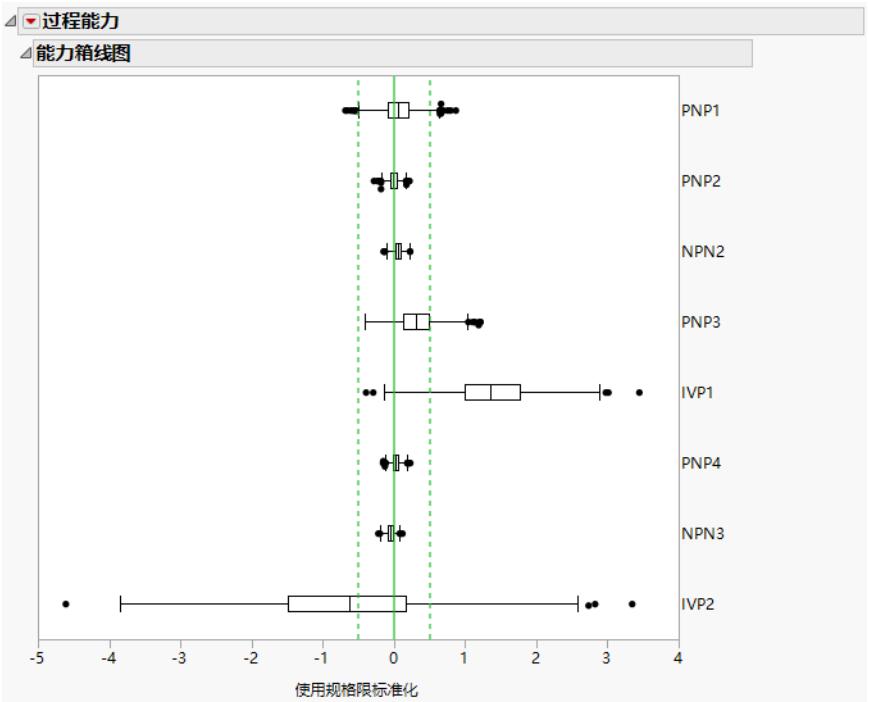
$$Z_{ij} = \frac{Y_{ij} - T_j}{2 \times \min(T_j - \mathrm{LSL}_j, \mathrm{USL}_j - T_j)}$$

对于仅有单侧规格的过程，请参见第 174 页的“单侧或缺失规格限”。有关未指定目标的情形，请参见第 209 页的“缺失目标的过程的能力箱线图”。

注意：非“正态”分布的过程变量不会标绘在能力箱线图上。

图 7.11 显示了 Semiconductor Capability.jmp 样本数据表中的八个变量的“能力箱线图”报表。

图 7.12 能力箱线图



该图显示在 ±0.5 处各绘制了一条绿色虚线。

- 对于目标在其规格限之间居中显示的过程，绿色虚线表示标准化规格限。
- 对于目标未在其规格限之间居中显示的过程，其中一条绿色虚线表示更接近目标的限值的标准化规格限。另一条绿色虚线表示反方向上相同的距离。

该图可用于将变量与其规格限加以比较。例如，在图 7.12 中，IVP1 的大多数点都高于其上规格限，而 IVP2 的大多数点都小于其目标。PNP2 看起来位于目标上，并且所有数据点都位于规格限以内。

单侧或缺失规格限

若某列只有一个规格限，则颜色的使用方式如下：

- 若仅指定了上规格限 (USL)，箱线图将显示为蓝色。
- 若仅指定了下规格限 (LSL)，箱线图将显示为红色。
- 若至少有一个过程仅有上规格限，则 0.5 处的虚线显示为蓝色。
- 若至少有一个过程仅有下规格限，则 -0.5 处的虚线显示为红色。

假定仅指定了下规格限和过程目标。能力箱线图基于变换观测的以下值。请参见第 207 页的“用于目标图和能力箱线图的符号”，了解符号说明：

$$Z_{ij} = \frac{Y_{ij} - T_j}{2(T_j - \mathrm{LSL}_j)}$$

假定仅指定了上规格限和过程目标。能力箱线图基于变换观测的以下值：

$$Z_{ij} = \frac{Y_{ij} - T_j}{2(\mathrm{USL}_j - T_j)}$$

有关如何在仅有单侧规格限时处理缺失目标的详细信息，请参见第 209 页的“单个规格限且无目标”。

能力指标图

能力指标图显示您输入为“Y，过程”的所有变量的 Ppk 值。每个变量名都显示在水平轴上并且 Ppk 值显示在垂直轴上。若拟合非正态分布，拟合分布名称显示在图中，作为变量名的后缀显示在括号内。若指定了过程重要性值，则会根据重要性调整能力指标图上点的大小。水平线位于图右侧的滑块指定的 Ppk 值处。

悬停在“能力指标图”中的某个点上方以查看该过程的控制图。点击控制图以启动带有相应控制图和能力报表的“控制图生成器”。

注意：若将无偏合并标准差选作过程的组内变异统计量，则控制图不可用于该过程。

图 7.13 显示 Process Measurements.jmp 样本数据表的“能力指标图”报表。使用非正态分布拟合了七个变量。使用正态分布拟合了**过程 7**。使用“能力指标图”红色小三角菜单中提供的“添加总 Sigma 点标签”选项为点添加了标签。

图 7.13 具有非正态分布的能力指标图

能力指标图选项

“能力指标图”红色小三角菜单包含以下选项：

显示组内 Sigma 点 显示或隐藏使用组内 sigma 估计值计算的点。

显示组内或组间和组内 Sigma 点 （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏使用组内 sigma 估计值或组间和组内 sigma 估计值（若指定）计算的点。

显示总 Sigma 点 显示或隐藏使用总 sigma 估计值计算的点。

着色水平 显示或隐藏 Ppk 水平着色。选择“着色水平”时，着色区域显示在图中。着色区域取决于 p 和 Ppk 之间的关系，其中的 p 表示 Ppk 之下的框中显示的值：

- 红色区域中的点：$\text{Ppk} < p$。
- 黄色区域中的点：$p < \text{Ppk} < 2p$。
- 绿色区域中的点：$2p < \text{Ppk}$。

添加组内 Sigma 点标签 显示或隐藏使用组内 sigma 估计值计算的点的标签。

标记组内或组间和组内 Sigma 点 （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏使用组内 sigma 估计值或组间和组内 sigma 估计值（若指定）计算的点的标签。

添加总 Sigma 点标签 显示或隐藏使用总 sigma 估计值计算的点的标签。

“过程能力”平台选项

“过程能力”红色小三角菜单包含以下选项：

单项详细报表 为分析中的每个变量显示或隐藏单项详细报表。请参见第 180 页的 “单项详细报表”。

目标图 显示或隐藏数据的目标图。目标图为每个变量在 *x* 轴上显示规格标准化均值偏移，在 *y* 轴上显示规格标准化标准差。请参见第 172 页的 “目标图”。（只有您指定了正态分布的变量才显示在图中。）

能力箱线图 为分析中的每个变量显示或隐藏能力箱线图。每列的值以其目标值为中心，并以目标值和规格限之间最小差值的两倍统一尺度。请参见第 174 页的 “能力箱线图”。（仅为您指定了正态分布的变量显示箱线图。）

标准化箱线图 针对为每个过程变量显示标准化箱线图的图提供两个选项。每一列都通过减去其均值并除以该列标准差的估计值进行标准化。该箱线图使用标准化值的分位数来构造。请参见第 187 页的 “标准化箱线图”。（仅为您指定了正态分布的变量显示标准化箱线图。）

- **组内 Sigma 标准化箱线图** 显示或隐藏称为“组内 Sigma 标准化箱线图”的图。该箱线图使用标准差的子组内估计值来构造。
- **组内或组间和组内 Sigma 标准化箱线图** （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏名为“组内或组间和组内标准化箱线图”的图。该箱线图使用标准差的组内估计值或组间和组内估计值（若指定）来构造。
- **总 Sigma 标准化箱线图** 显示或隐藏称为“总 Sigma 标准化箱线图”的图。该箱线图使用标准差的总估计值来构造。

能力指标图 显示您输入为 “Y，过程”的所有变量的总 Ppk 值。请参见第 176 页的 “能力指标图”。

过程能力图 显示或隐藏能力 - 稳定性四象限图。具有至少一个规格限的每个过程都用一个点来表示。请参见第 188 页的 “过程能力图”。

汇总报表 提供两个用于能力指标汇总报表的选项。请参见第 189 页的 “汇总报表”。

- **组内 Sigma 汇总报表** 显示或隐藏使用标准差的子组内估计值计算的能力指标的汇总报表。（仅为具有指定正态分布的变量提供结果）。
- **组内或组间和组内 Sigma 汇总报表** （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏使用标准差的组内估计值或者组间和组内估计值（若指定）计算的能力指标的汇总报表。
- **总 Sigma 汇总报表** 显示或隐藏使用标准差的总估计值计算的能力指标的汇总报表。

操作选项

以下红色小三角菜单选项执行下列操作：

超出规格限的值 提供的选项用于数据表中包含超出规格限的值的单元格。

选择超出规格限的值 选择数据表中包含至少一个不在规格限内的值的所有行列。

对超出规格限的值着色 对数据表中与超出规格限的值对应的单元格着色。若值高于上规格限，则单元格着蓝色；若值低于下规格限，则单元格着红色。

提示：要删除特定单元格中的颜色，请选择所有相关单元格。右击其中某一个单元格，然后选择 “清除颜色”。要删除所有单元格中的颜色，请取消选择 “对超出规格限的值着色”。

制作目标图汇总表 为 “目标图”中绘制的点创建汇总表。该表中包括变量名、其规格标准化均值偏移以及其规格标准化标准差。该表中每个变量都有两行：每个 sigma 类型有一行（组内和总）。请参见第 191 页的 “制作目标图汇总表”。

排序依据 重新排序箱线图、汇总报表和单项详细报表。您可以按照以下方式重新排序：初始顺序、反转初始顺序、组内 Sigma Cpk 升序、组内或组间和组内 Sigma Cpk 升序、组内 Sigma Cpk 降序、组内或组间和组内 Sigma Cpk 降序、总 Sigma Ppk 升序或总 Sigma Ppk 降序。依据 “组内 Sigma”排序的选项只会为具有指定正态分布的变量重新排序图元素。

注意：仅当为启动窗口中的至少一个过程选定 “计算组间和组内能力”时，按 “组内或组间和组内 Sigma”排序的选项才可用。

保存 提供用于保存规格限和分布的选项。

将规格限保存为列属性 将规格限保存至分析中每个变量的列属性。若“规格限”列属性不存在，则创建列属性。若“规格限”列属性存在，则覆盖列属性中的值。请参见第 170 页的“规格限列属性”。

将分布另存为列属性 将计算能力时使用的分布另存为“过程能力分布”列属性。请参见《使用 JMP》。

若某列包含的“分布”属性指定的是非正态分布并且不包含“过程能力分布”属性，则“过程能力”平台应用非正态拟合。“过程能力”平台使用“分布”列属性中指定的分布；若该分布在过程能力中不受支持，则使用 Johnson 拟合。若某列包含“过程能力分布”属性，则“过程能力”平台会使用“过程能力分布”列属性中指定的分布。

注意：若想要在“过程能力”平台中使用特定分布，请将其另存为“过程能力分布”列属性。

在新表中保存规格限 将每个过程的规格限和“显示限值”设置保存至高型的限值数据表。请参见第 169 页的“限值数据表”。

重新启动对话框 打开平台启动窗口并重新调用用于创建报表的设置。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

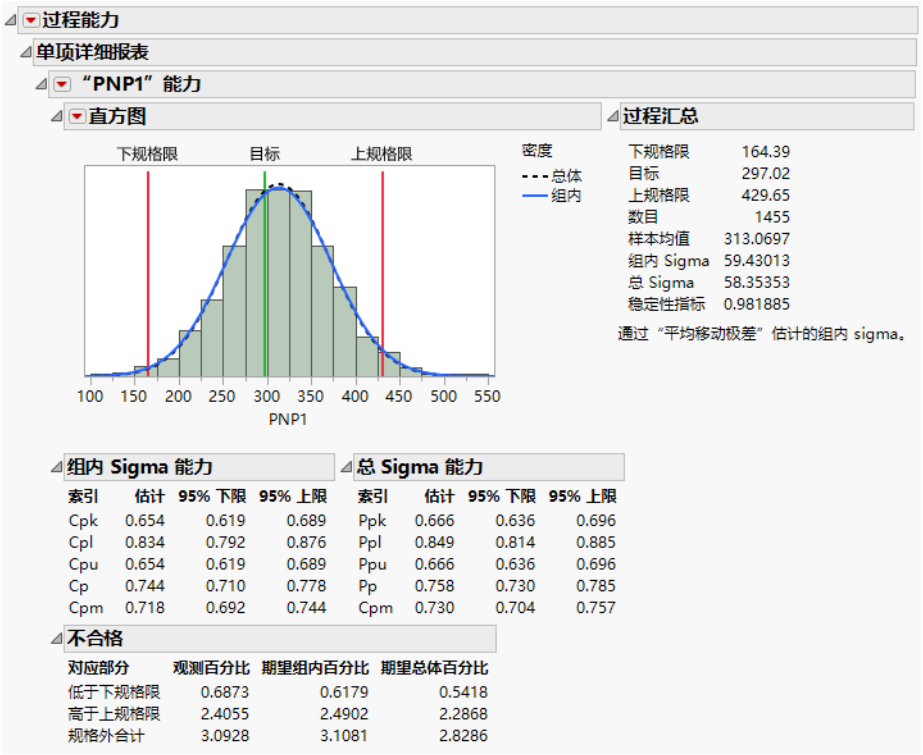
单项详细报表

“单项详细报表”选项为分析中的每个变量都显示一个能力报表。

正态分布

图 7.14 显示了 Semiconductor Capability.jmp 样本数据表中的 PNP1 的“单项详细报表”，如第 156 页的“示例：包含正态变量的“过程能力”平台”中所述。

图 7.14 单项详细报表



具有正态分布的某变量的“单项详细报表”显示直方图、过程汇总详细信息，以及能力和不合格统计量。该直方图显示值的分布、上规格限、下规格限和过程目标（若已指定），以及显示

假设分布的一条或两条曲线。图 7.14 中的直方图显示两条正态曲线 — 一条基于标准差的总估计值，另一条基于子组内估计值。

使用正态分布拟合过程时，“过程汇总”包括**稳定性指标**，这是一个用于衡量过程稳定性的测度。稳定性指标定义如下：

（总 Sigma/ 组内 Sigma）

若在启动窗口中为某个过程指定了 “计算组间和组内能力”，该过程的稳定性指标定义如下：

（总 Sigma/ 组间和组内 Sigma）

稳定过程的稳定性指标接近 1。较高的值指示稳定性较差。

注意：您可以在“文件”>“首选项”>“平台”>“过程能力”中更改稳定性评估类型的首选项。这将更改在整个“过程能力”平台中使用的稳定性评估类型。

非正态分布

注意：基于子组内变异的能力指标和稳定性指标不可用于您指定了非正态分布的过程。

图 7.15 显示了 Process Measurements.jmp 样本数据表中的**过程 1** 的“单项详细报表”，如第 158 页的“包含非正态变量的“过程能力”平台的示例”中所述。

图 7.15 “过程 1”的“单项详细报表”

该报表打开时会显示一条注释，总结您在启动窗口中选定的非正态分布选项。

具有非正态分布的某变量的“单项详细报表”显示直方图、过程汇总详细信息，以及能力和不合格统计量。该直方图显示值的分布、上规格限、下规格限和过程目标（若已指定）。显示拟合分布的曲线叠加在该直方图之上。若选定了非参数分布，直方图中显示的曲线将为非参数密度。

若选定非正态参数分布，报表中还会显示一个“参数估计值”报表；若选定非参数拟合，则报表中还会显示一个“非参数密度”报表。请参见第 184 页的“参数估计值”和第 184 页的“非参数密度”。

单项详细报表选项

“单项详细报表”部分中每个变量的分级显示项标题都采用 <变量名> 能力的格式。不过，若请求了非正态能力，相关分布名称将在分级显示项标题中显示在括号内。

每个能力报表都包含一个提供以下选项的红色小三角菜单：

比较分布 显示或隐藏用于比较过程分布的控制面板。请参见第 184 页的“比较分布”。

过程汇总 显示或隐藏变量的汇总统计量，包括总 sigma 估计值，以及组内 sigma 估计值和稳定性指标（若指定了正态分布）。若在启动窗口中为至少一个过程指定了“计算组间和组内能力”，则还会包括组间 Sigma 与组间和组内 sigma 估计值。

直方图 显示或隐藏变量值的直方图。直方图报表包括一个红色小三角菜单，用于控制直方图的以下功能：

显示规格限 在直方图上显示或隐藏过程规格限处的红色垂直线。

显示目标 在直方图上显示或隐藏过程目标处的绿色垂直线。

显示组内 Sigma 密度 在直方图上显示或隐藏近似正态密度函数，其中的均值来自样本均值，标准差来自组内 sigma 估计值。

显示组间和组内 Sigma 密度 （仅当为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）在直方图上显示或隐藏近似正态密度函数，其中的均值来自样本均值，标准差来自组间和组内 sigma 估计值。

显示总 Sigma 密度 在直方图上显示或隐藏近似正态密度函数，其中的均值来自样本均值，标准差来自总 sigma 估计值。

显示计数轴 在直方图右侧显示或隐藏用于显示观测计数的附加轴。

显示密度轴 在直方图右侧显示或隐藏用于显示密度的附加轴。

能力指标 控制以下能力指标报表的显示：

组内 Sigma 能力 （当分布为“正态”时可用。）显示或隐藏基于组内（短期）sigma 的能力指标（以及置信区间）。

组间和组内 Sigma 能力 （仅当分布为“正态”并且为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏基于组间和组内 sigma 的能力指标。

组内 Sigma Z 基准 （当分布为“正态”时可用。）显示或隐藏基于组内（短期）sigma 的 Z 基准指标。

组间和组内 Sigma Z 基准 （仅当分布为“正态”并且为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏基于组间和组内 sigma 的 Z 基准指标。

组内 Sigma 目标指标 （当分布为“正态”时可用。）显示或隐藏基于组内（短期）sigma 的目标指标的估计值。

组间和组内 Sigma 目标指标 （仅当分布为“正态”并且为启动窗口中的至少一个过程选定“计算组间和组内能力”时才可用。）显示或隐藏基于组间和组内 sigma 的目标指标的估计值。

总 Sigma 能力 显示或隐藏基于总（长期）sigma 的能力指标（以及置信区间）。

总 Sigma Z 基准 （当分布为“正态”时可用。）显示或隐藏基于总（长期）sigma 的 Z 基准指标。

注意：默认情况下，基于 $\alpha = 0.05$ 构造能力指标的置信区间。要更改默认置信水平，请选择“文件” > “首选项” > “平台” > “过程能力”。

不合格 显示或隐藏观测低于下规格限、高于上规格限、以及位于规格限之外的观测百分比和期望百分比。不合格表包含观测和期望 PPM 以及计数的隐藏列。

交互式能力图 显示或隐藏“交互式能力图”。交互式能力图支持您更改一个或多个汇总统计量的值，并查看更改如何影响能力分析。通过“原始”报表和“新”报表可显示原始和新

的汇总统计量、能力指标和预期 PPM。使用滑块控件或文本框更改规格限、均值和总 sigma 的原始值。也可以使用“均值偏移”框将均值偏移原始 sigma 的倍数。“交互式能力图”报表具有以下红色小三角菜单选项：

能力 在“原始”报表和“新”报表中显示或隐藏能力指标。

期望 PPM 在“原始”报表和“新”报表中显示或隐藏期望 PPM 值。

恢复为原始值 将“新”报表中的交互式能力图和汇总值还原为原始值。

将新的规格限保存为列属性 将新的规格限保存为原始数据表中的列的“规格限”列属性。

注意：除非启用“自动重新计算”选项，否则不会使用新的规格限重新运行分析。

参数估计值 （当分布非 “正态”或选定 “非参数”时可用。）显示或隐藏 “参数估计值”报表，该报表提供选定分布的参数的估计值。

除 Johnson 系列分布之外所有分布的估计值都使用最大似然得到。有关 Johnson 系列拟合的详细信息，请参见第 167 页的 “Johnson 分布拟合方法”。

第 214 页的 “非正态分布的能力指标：百分位数和 Z 得分方法”中对用于正态、 beta、指数、gamma、Johnson、对数正态和 Weibull 分布的参数和概率密度函数进行了说明。以上这些也是“分布”平台中使用的参数化，只不过“过程能力”不支持阈值参数。请参见《基本分析》。

固定参数 （当分布非 “正态”或选定 “非参数”时可用。）显示一个窗口，该窗口支持您修复非正态分布中的一个或多个参数值。在“用户定义的值”列中为要修复的参数输入值。点击 “确定”后，将根据固定的参数值重新估计省略的参数值。重新估计的参数值将显示在“参数估计值”报表中，同时显示一列，指示哪些参数是固定的。

非参数密度 （在将 “非参数”选作分布时可用。）显示或隐藏 “非参数密度”报表，其中提供拟合非参数分布时使用的**核带宽**。核带宽通过以下公式得出，其中，n 是观测数，S 是未校正样本标准差：

$$\text{带宽} = \frac{0.9S}{n^{1/5}}$$

比较分布

“比较分布”报表支持您比较和应用各种分布拟合。请注意以下事项：

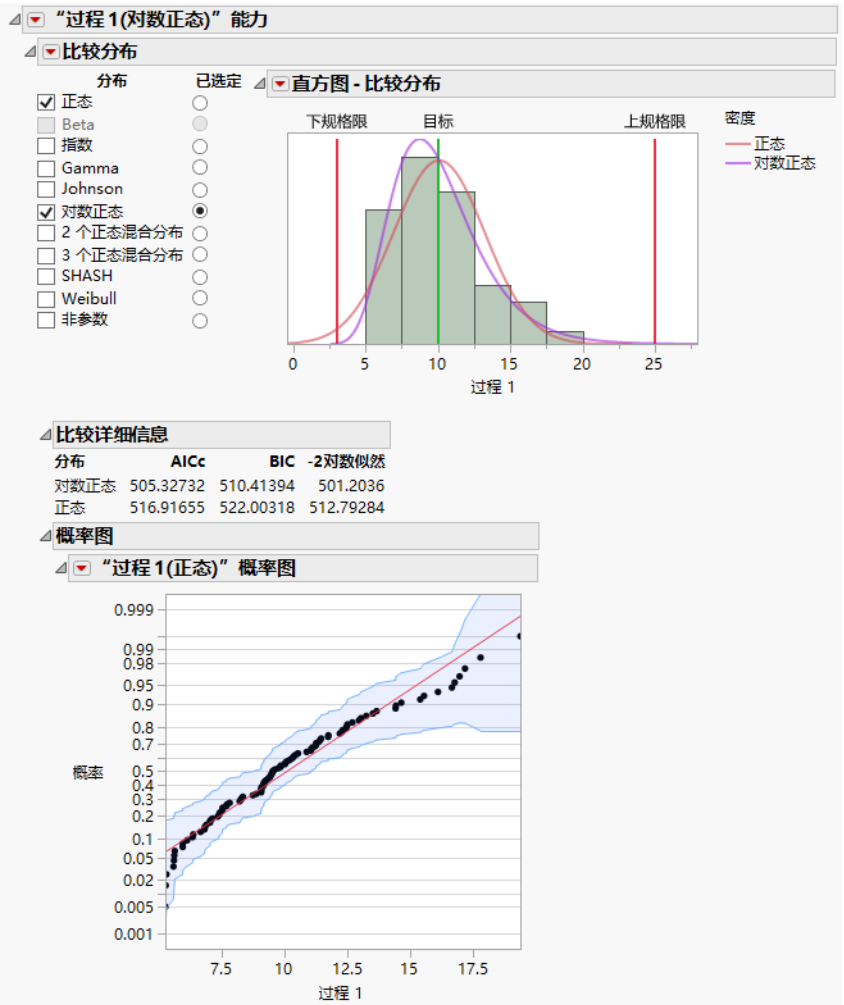
- 您选定的分布将在“已选定”列中指示出来。
- 该报表最初显示“已选定”分布以及“比较详细信息”报表中的其他拟合分布的拟合统计量。若选定了“最佳拟合”，“比较详细信息”报表最初显示所有参数拟合的统计量。
- 在“分布”列表中选中您想要比较的分布。
  - 您选择的每个系列中的最佳拟合分布的概率密度函数将叠加在“直方图 - 比较分布”报表中的直方图之上。

- 若为参数分布，则“比较详细信息”报表中会添加针对该系列的一行，其中包含拟合结果。
- 若在“分布”列表中选中了“非参数”，“比较分布”报表中会添加“非参数密度”报表，其中显示自动选定的核带宽。请参见第 184 页的“非参数密度”。
- 您可以更改选定的分布，只需选定“已选定”下对应的单选按钮。能力报表将随之更新以显示选定分布的结果。

图 7.16 显示 Process Measurements.jmp 样本数据表中**过程 1** 的“比较分布”报表。“已选定”分布（即“对数正态”）正与“正态”分布作比较。“比较详细信息”报表显示这两种分布的拟合统计量。

要获取概率图，请点击“比较分布”红色小三角并选择“概率图”。图 7.16 中正态分布的概率图中的各点并未紧贴线条分布。这表示拟合不佳。

图 7.16 比较带有正态概率图的分布



比较分布选项

“比较分布”红色小三角菜单包含以下选项：

比较详细信息 对于每个分布，提供 AICc、BIC 和 -2 对数似然值。请参见《拟合线性模型》。（不可用于非参数拟合。）

比较直方图 显示或隐藏直方图报表。

概率图 显示或隐藏为您拟合的每个参数分布显示概率图的报表（图 7.16）。观测的水平坐标是其观测的数据值。观测的垂直坐标是观测秩的拟合分布的分位数值。对于正态分布，总 sigma 估计值用于确定拟合分布。

与每个“概率图”关联的红色小三角菜单都包含以下选项。

联合经验置信限 在数据来自选定参数系列的前提下，显示或隐藏有联合 95% 置信水平包含真概率函数的置信限。这些限值在所有点处都包含相同的估计精度。用这些值来确定选定的参数分布是否能够很好地拟合数据。请参见 Nair (1984) 和 Meeker and Escobar (1998)。

联合经验置信限着色 显示或隐藏联合经验置信限之间的区域的着色。

参数拟合线 显示或隐藏基于拟合分布显示观测的预测概率的线条。

参数拟合置信限着色 显示或隐藏参数拟合置信区间之间区域的着色。参数拟合置信限具有置信水平 (1 - Alpha)，其中，Alpha 是您在启动窗口中指定的值。（仅当参数拟合置信限有意义并且能够计算它们的情况下才可用。）

若可能，通过将参数分布 F 表示为位置尺度系列来计算区间，以便 $F(y) = G(z)$，其中，$z = (y - \mu)/\sigma$。使用 delta 方法来计算拟合位置尺度分量在某点处的近似标准误差。使用标准误差估计值，将为每个点计算 z 的 Wald 置信区间。累积分布函数 F 的置信区间通过使用 G 变换 Wald 区间得到。请注意，在某些情况下，需要特别调整以便在过程测量值区间端点附近提供适当区间。

排序依据比较准则 根据您选择的准则对“比较详细信息”报表中的分布排序。默认依照 AICc 排序，除非您在启动窗口的 “分布比较准则”面板中选定了其他准则。

标准化箱线图

“标准化箱线图”选项显示或隐藏已使用标题中的指定 sigma 标准化的箱线图。绘制标准化箱线图时，JMP 会通过减去均值并除以标准差对各列进行标准化，之后使用这些标准化值为每列生成箱线图。

图 7.17 组内 Sigma 标准化箱线图

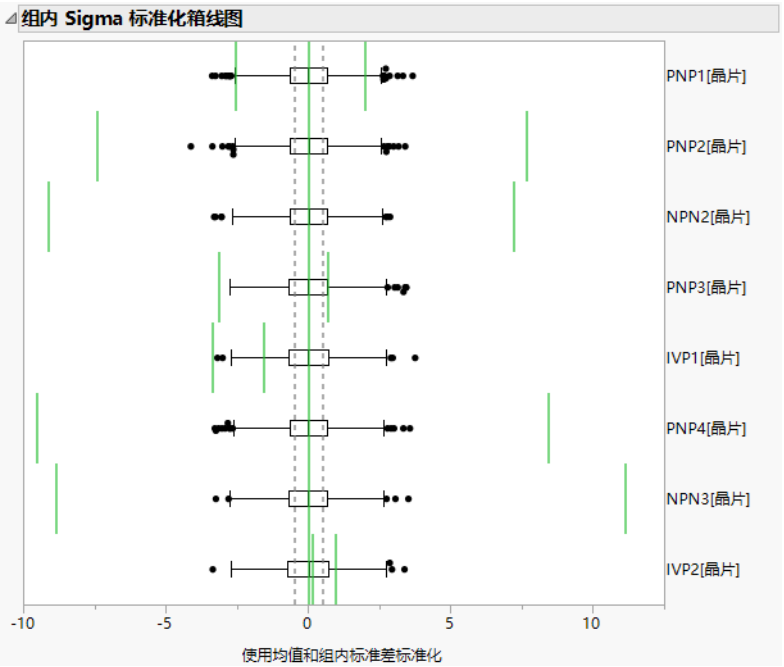


图 7.17 显示了在 Semiconductor Capability.jmp 样本数据表中使用**晶片**作为子组变量选择过程变量的“组内 Sigma 标准化箱线图”。

绿色垂直线表示通过每个变量的均值和标准差进行标准化的每个变量的规格限。在 ±0.5 的位置上绘制灰色垂直虚线，因为数据是按照标准差 1 进行的标准化。

过程能力图

“过程能力图”选项显示或隐藏能力 - 稳定性的四象限图。具有规格限的每个过程都由一个点来表示。若指定了过程重要性值，则会根据重要性调整点的大小。每个点的水平坐标等于过程的稳定性指标，每个点的垂直坐标等于过程的总 Ppk 能力。该图根据以下默认边界划分为四个着色象限：

- 超过 1.25 的稳定性指标指示过程不稳定。
- 小于 1.0 的 Ppk 指示过程能力不足。

此外，图上有一条红线指示 Cpk 值为 1 的位置。可以使用图右侧的“Ppk”和“稳定性指标”滑块控件调节定义四个象限的边界。您还可以在“文件”>“首选项”>“平台”>“过程性能图”和“文件”>“首选项”>“平台”>“过程能力”中设置所需的“能力和稳定性”边界和稳定性评估类型的首选项。

图例包含着色区域的说明。若有任何过程缺失下规格限或上规格限，图例还可以显示用于这些过程的标记。若图例中未显示标记，则图中所有过程都同时包含下规格限和上规格限。请参见第 174 页的“单侧或缺失规格限”。

悬停在“过程性能图”中的某个点上方以查看该过程的控制图。点击控制图以启动带有相应控制图和能力报表的“控制图生成器”。

注意：若将无偏合并标准差选作过程的组内变异统计量，则控制图不可用于该过程。

“过程能力图”红色小三角菜单包含以下选项：

添加点标签 显示或隐藏过程能力图中每个点的标签。

显示组内 Cpk 曲线 在“过程性能图”中显示或隐藏组内 Cpk 曲线。

图 7.18 过程能力图

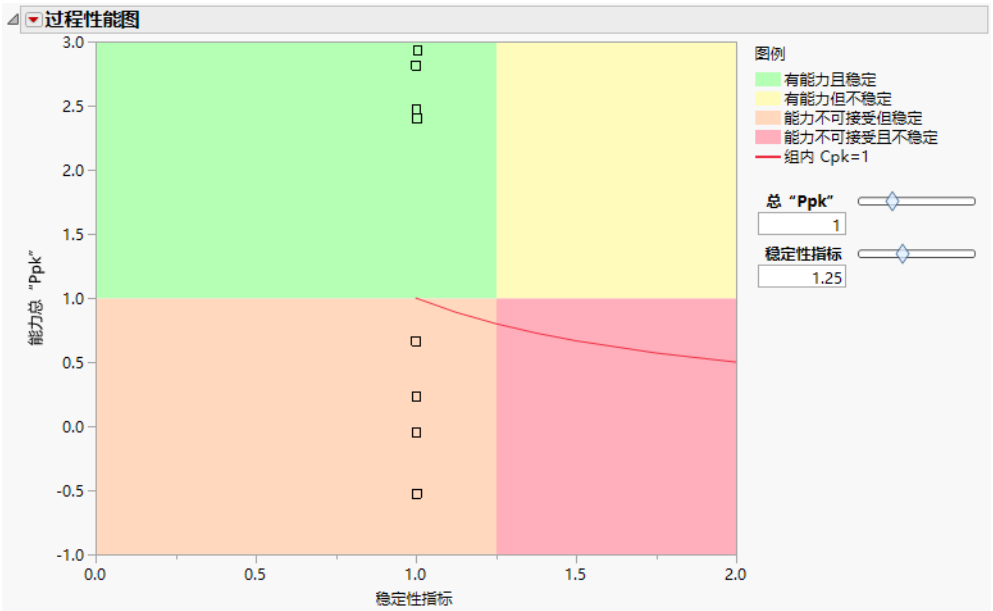


图 7.18 显示了在 Semiconductor Capability.jmp 样本数据表中使用**晶片**作为子组变量选择过程变量的“过程能力图”。

汇总报表

“汇总报表”选项显示或隐藏包含以下每个变量的统计量的表：下规格限、目标、上规格限、样本均值、各种 Sigma 估计值、稳定性指标、Cpk、Cpl、Cpu、Cp、Cpm 和不合格统计量。若至少有一个非缺失过程重要性值，则在汇总报表中还包含一个“重要性”列。使用在报表标题中指定的 sigma 估计值计算这些统计量。“稳定性指标”、“Cpk”、“Cpl”、“Cpu”

和“cp”等列着绿色、黄色和红色以指示足够、边缘和很差的稳定性或能力。该颜色编码方案与您在过程性能图中看到的方案匹配。

注意：您可以在“文件”>“首选项”>“平台”>“过程能力”中更改稳定性评估类型的首选项。这将更改在整个“过程能力”平台中使用的稳定性评估类型。

图 7.19 显示了两个汇总报表的部分列，如第 156 页的“示例：包含正态变量的“过程能力”平台”中所述。该报表提供以下可选列：

- Cpk、Cpl、Cpu、Cp 和 Cpm 的置信区间
- 期望和观测 PPM 统计量（规格外、低于下规格限、高于上规格限）

  注意：期望的 PPM 统计量是基于所选的分布您所期望看到的百分比。默认情况下，该分布为正态分布。观测的 PPM 统计量是基于实际数据的百分比。

- 样本标准差
- 样本大小（数目）、最小值和最大值。
- 目标指标

  注意：“目标指标”仅可用于 “组内 Sigma 能力汇总”报表。

要显示这些可选列，请右击报表并从“列”子菜单中选择列名。

注意，报表（基于总 sigma）显示总能力指标 Ppk、Ppl、Ppu 和 Pp，而不是组内能力指标 Cpk、Cpl、Cpu 和 Cp。总能力指标的标签取决于 AIAG (Ppk) 标签首选项的设置。

图 7.19 组内 Sigma 和总 Sigma 能力汇总报表

过程能力

组内 Sigma 能力汇总报表

过程	下规格限	目标	上规格限	样本均值	组内 Sigma	稳定性指标	Cpk	Cpl	Cpu	Cp	Cpm
PNP1	164.39	297.02	429.65	313.0697	59.43013	0.981885	0.654	0.834	0.654	0.744	0.718
PNP2	-136.12	465.44	1067.01	456.6157	79.27036	1.007008	2.492	2.492	2.567	2.530	2.514
NPN2	96.59	113.75	130.9	115.7421	2.131652	0.98552	2.370	2.995	2.370	2.683	1.959
PNP3	118.68	130.29	141.9	137.6146	6.160912	0.983744	0.232	1.024	0.232	0.628	0.404
IVP1	59.62	63.41	67.2	73.78072	4.238298	0.990097	-0.518	1.114	-0.518	0.298	0.113
PNP4	-54.43	238.74	531.91	256.3756	33.22573	0.981389	2.764	3.118	2.764	2.941	2.598
NPN3	97.32	120.8	144.29	118.1352	2.362847	1.000809	2.936	2.936	3.690	3.313	2.198
IVP2	139.2	142.31	145.41	138.2432	7.406516	0.989286	-0.043	-0.043	0.323	0.140	0.122

总 Sigma 能力汇总报表

过程	下规格限	目标	上规格限	样本均值	总 Sigma	稳定性指标	Ppk	Ppl	Ppu	Pp	Cpm
PNP1	164.39	297.02	429.65	313.0697	58.35353	0.981885	0.666	0.849	0.666	0.758	0.730
PNP2	-136.12	465.44	1067.01	456.6157	79.82589	1.007008	2.475	2.475	2.549	2.512	2.497
NPN2	96.59	113.75	130.9	115.7421	2.100786	0.98552	2.405	3.039	2.405	2.722	1.975
PNP3	118.68	130.29	141.9	137.6146	6.060762	0.983744	0.236	1.041	0.236	0.639	0.407
IVP1	59.62	63.41	67.2	73.78072	4.196326	0.990097	-0.523	1.125	-0.523	0.301	0.113
PNP4	-54.43	238.74	531.91	256.3756	32.60738	0.981389	2.817	3.177	2.817	2.997	2.636
NPN3	97.32	120.8	144.29	118.1352	2.364757	1.000809	2.934	2.934	3.687	3.310	2.197
IVP2	139.2	142.31	145.41	138.2432	7.327164	0.989286	-0.044	-0.044	0.326	0.141	0.123

制作目标图汇总表

“制作目标图汇总表”选项可生成一个汇总数据表，其中包含每个变量的名称、其规格标准化均值偏移（规格标准化均值），以及其规格标准化标准差（规格标准化标准差）。对于每个变量，每种 sigma 类型都对应一行。

注意：若某个变量是使用正态分布之外的分布拟合的，那么拟合分布的名称将括在变量名后面的括号中。不为非正态变量提供“规格标准化均值”和“规格标准化标准差”值。

目标图中的点链接至“目标图汇总表”中的行。若对目标图中的点应用行状态，您可以在“目标图汇总表”中更改相应的行状态。相反，若在“目标图汇总表”中应用行状态，则行状态将反映在目标图中。

图 7.20 显示了第 156 页的“示例：包含正态变量的“过程能力”平台”中所述的 Semiconductor Capability.jmp 样本数据表的“目标图汇总表”。

图 7.20 汇总表

	过程	Sigma 类型	规格标准化均值	规格标准化标准差
1	PNP1	组内	0.0605056038	0.2240448185
2	PNP2	组内	-0.00733452	0.0658873257
3	NPN2	组内	0.0580798256	0.0621472935
4	PNP3	组内	0.3154424678	0.2653278331
5	IVP1	组内	1.3681693325	0.5591421926
6	PNP4	组内	0.0300774244	0.0566663241
7	NPN3	组内	-0.05674621	0.0503161575
8	IVP2	组内	-0.65593547	1.1945993146
9	PNP1	总体	0.0605056038	0.2199861461
10	PNP2	总体	-0.00733452	0.0663490674
11	NPN2	总体	0.0580798256	0.0612473972
12	PNP3	总体	0.3154424678	0.2610147096
13	IVP1	总体	1.3681693325	0.5536050447
14	PNP4	总体	0.0300774244	0.0556117297

“过程能力”平台的更多示例

- “稳定过程的过程能力”
- “不稳定过程的过程能力”
- “模拟非正态过程 Ppk 的置信限”

稳定过程的过程能力

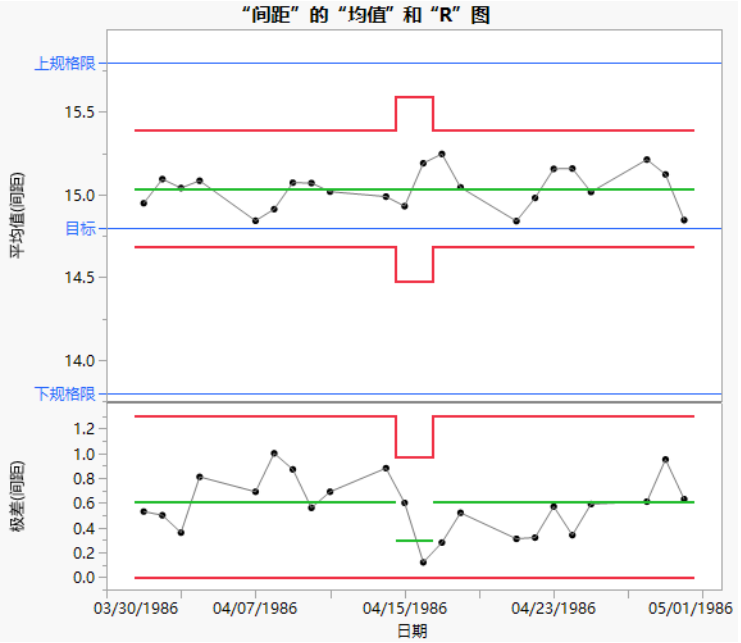
在本例中，您验证一些假设，基于这些假设您可以用能力分析来估计 PPM 缺陷率。可直接通过“控制图生成器”访问“过程能力”。数据中大小为 5 的子组有 22 个。缺失 6 个读数，两个连续子组中每个都缺失 3 个读数。

通过“控制图生成器”得到的过程能力

您可以使用“控制图生成器”查看过程稳定性以及过程特征的正态分布假定。还可以直接在“控制图生成器”中获得 “过程能力”信息。

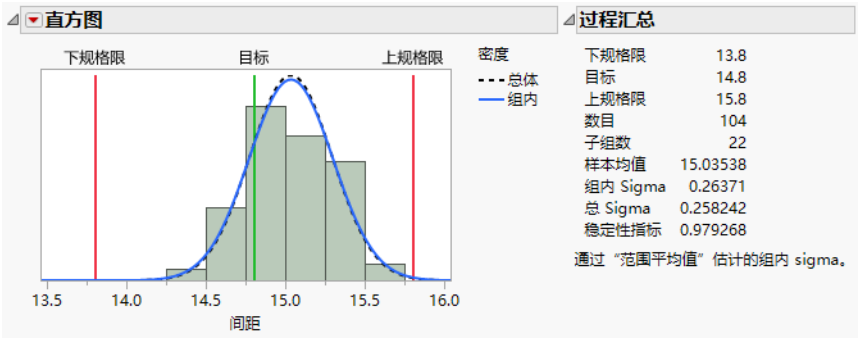
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Clips2.jmp。
2. 选择**分析** > **质量和过程** > **控制图生成器**。
3. 将**日期**拖至**子组**区域。
4. 将**间距**拖至 Y 区域。

图 7.21 “间距”的均值和 R 控制图



控制图指示**间距**随时间推移是稳定的。由于**间距**具有 “规格限”列属性，“过程能力分析”报表将出现在控制图的右侧。

图 7.22 “间距”的过程能力分析中的直方图



直方图和拟合正态蓝色曲线表明**间距**的分布近似服从正态分布。尽管过程是稳定的，**间距**的分布还是会向规格范围的右侧偏移。

“过程汇总”报表显示了保存至“规格限”列属性的规格限。还显示了根据子组内变异计算的 sigma 估计值（组内 Sigma）与样本标准差给出的总估计值（总 Sigma）没有很大差异。因此，稳定性指标接近 1 (0.979268)。由于过程是稳定的，因此这是期望的结果。

5. 右击“不合格”报表的主体，然后从“列”子菜单中选择**期望组内 PPM**。

图 7.23 能力指标和不合格报表

组内 Sigma 能力

索引	估计	95% 下限	95% 上限
Cpk	0.966	0.805	1.128
Cpl	1.562	1.314	1.808
Cpu	0.966	0.805	1.127
Cp	1.264	1.071	1.457
Cpm	0.943	0.828	1.058

总 Sigma 能力

索引	估计	95% 下限	95% 上限
Ppk	0.987	0.838	1.136
Ppl	1.595	1.367	1.821
Ppu	0.987	0.837	1.135
Pp	1.291	1.115	1.467
Cpm	0.954	0.841	1.072

不合格

对应部分	观测百分比	期望组内百分比	期望总体百分比	期望组内 PPM
低于下规格限	0.0000	0.0001	0.0001	1.402263
高于上规格限	0.0000	0.1869	0.1534	1869.0329
规格外合计	0.0000	0.1870	0.1535	1870.4352

使用子组变异计算的 Cpk 值为 0.966，指示过程能力水平不高。Cpl 值表明性能良好，但这是因为过程有偏离下规格限。缺陷部件通常是由于**间距**值较大产生的。

注意， Cpk 的置信区间很宽；范围介于 0.805 到 1.128 之间。即使有 104 个观测也是如此。由于能力指标是比率值，因此变化异常大。基于能力指标的点估计值很容易得出错误结论。

不合格报表中给出的超出规格范围的产品估计值提供了直接衡量过程能力的测度。不合格报表中的 PPM 值指示，**间距**很少落在下规格限 （每百万 1.4 个部件）之下。不过，**间距**落在上规格限之上的部件数为每百万 1869.0 个部件。

对于未中心化的过程，若过程调整为是中心化的，则 Cp 值指示潜在能力。若该过程调整为以目标值 14.8 为中心，则其能力为 1.264，且置信区间从 1.071 到 1.457。

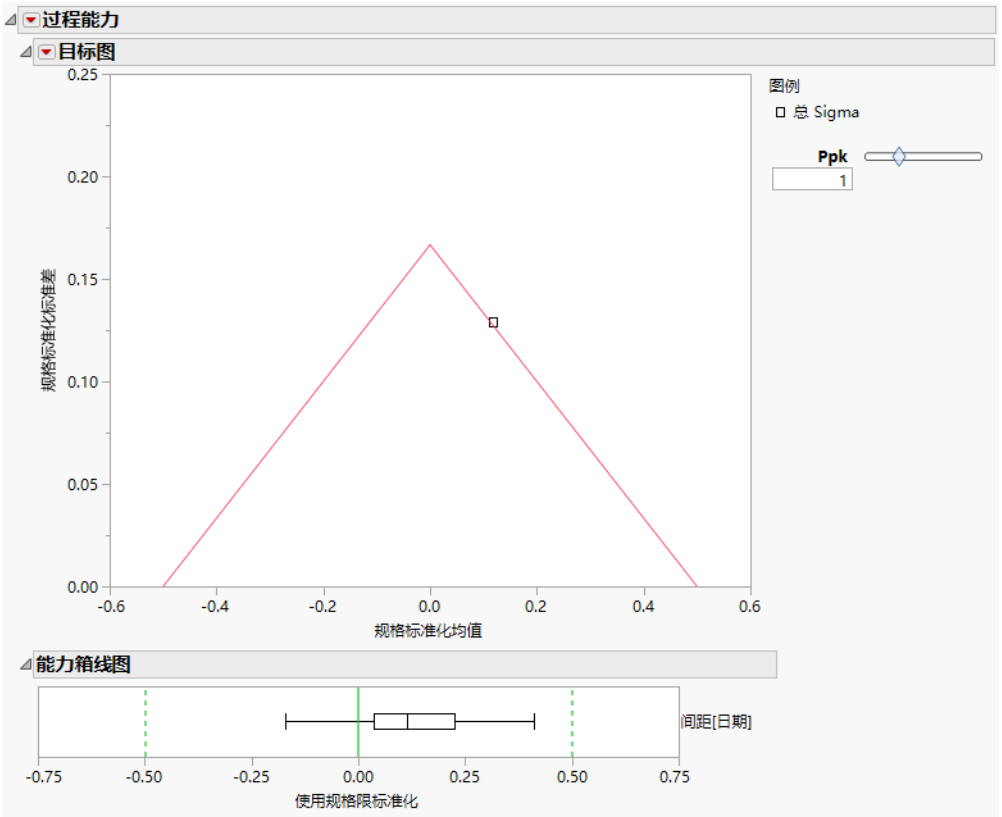
“过程能力”平台

您已验证了 “间距”的稳定性和正态性，现在您可以在 “过程能力”平台中获得其他信息。

1. 选择**分析 > 质量和过程 > 过程能力**。
2. 选择**间距**并点击 **Y，过程**。
3. 打开**过程子组**分级显示项。
4. 在 “选择列”列表中选择**日期**，并在 “角色”列表中选择**间距**。
5. 点击**嵌套子组 ID 列**。

   默认情况下，“子组内变异统计量”选项设置为 “无偏标准差的平均值”。在 “控制图生成器”示例中 （第 192 页的 “通过 “控制图生成器”得到的过程能力”），使用了子组极差。
6. 点击**确定**。

图 7.24 “间距”的目标图和箱线图



目标图显示 “间距”的 Ppk 指标基本上等于 1。箱线图显示大部分值落入规格范围内，但数据值的大多数向规格范围内的右侧偏移。

7. 点击 “过程能力”红色小三角并选择单项详细报表。

报表使用 “控制图生成器”获得，只是 “组内 Sigma”基于平均标准差而不是平均极差。请参见第 193 页的 “‘间距’的过程能力分析中的直方图”和第 193 页的 “能力指标和不合格报表”。

不稳定过程的过程能力

在本例显示的情况中，由于过程不稳定，总变异与组内变异存在差别。它使用 Quality Control 样本数据文件夹中的 Coating.jmp 数据表（数据取自 ASTM Manual on Presentation of Data and Control Chart Analysis）。关注的过程变量为**重量**列，按照**样本**列划分为子组。

“过程能力”平台

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Coating.jmp。
2. 选择**分析** > **质量和过程** > **过程能力**。
3. 选择**重量**并点击 **Y，过程**。
4. 打开**过程子组**分级显示项。
5. 在左侧的**选择列**列表中选择**样本**。
6. 在右侧的**为选定列指定角色**列表中选择**重量**。
7. 点击**嵌套子组 ID 列**。
8. 点击**确定**。
9. 在**规格限**窗口中，为**下规格限**输入 16，**目标**输入 20，**上规格限**输入 24。
10. 点击**确定**。
11. 点击 “目标图”红色小三角并选择**显示组内 Sigma 点**。
12. 点击 “过程能力”红色小三角并选择**单项详细报表**。

图 7.25 Coating.jmp 数据的 “过程能力”报表

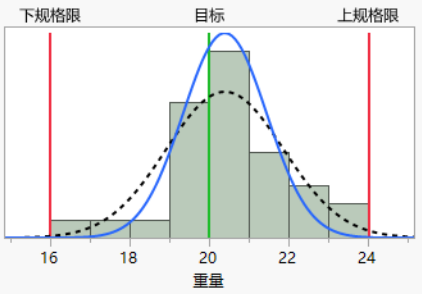
过程能力

单项详细报表

“重量[样本]”能力

按子组 ID 分组: 样本

直方图



过程汇总

下规格限	16
目标	20
上规格限	24
数目	32
子组数	8
样本均值	20.39688
组内 Sigma	1.051753
总 Sigma	1.475058
稳定性指标	1.402476

通过“无偏标准差的平均值”估计的组内 sigma。

组内 Sigma 能力

索引	估计	95% 下限	95% 上限
Cpk	1.142	0.789	1.495
Cpl	1.394	0.971	1.813
Cpu	1.142	0.789	1.491
Cp	1.268	0.899	1.635
Cpm	1.186	0.899	1.473

总 Sigma 能力

索引	估计	95% 下限	95% 上限
Ppk	0.814	0.581	1.048
Ppl	0.994	0.720	1.264
Ppu	0.814	0.580	1.045
Pp	0.904	0.680	1.127
Cpm	0.873	0.670	1.101

不合格

对应部分	观测百分比	期望组内百分比	期望总体百分比
低于下规格限	0.0000	0.0015	0.1437
高于上规格限	0.0000	0.0306	0.7289
规格外合计	0.0000	0.0321	0.8726

目标图

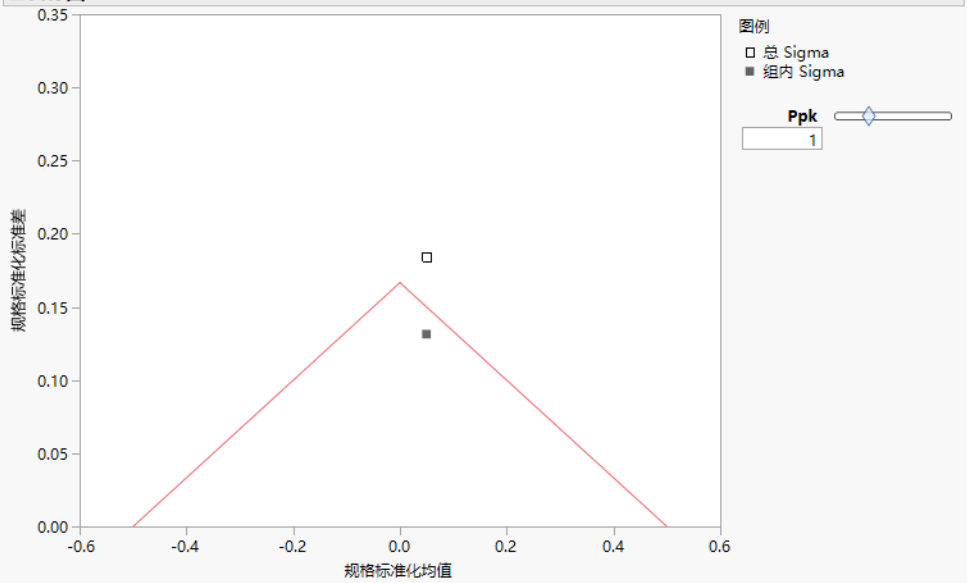


图 7.25 显示了所得到的“过程能力”报表。“目标图”显示的两个点分别代表根据规格限标准化的均值偏移和标准差。“目标图”旁边有一个图例用于标识这两个点。“总 Sigma”点使用总样本标准差计算。“组内 Sigma”点使用子组内标准差的估计值计算。

使用“总 Sigma”计算的点位于对应于 Ppk 为 1 的目标三角形之外。这表示变量**重量**将产生不合格产品。

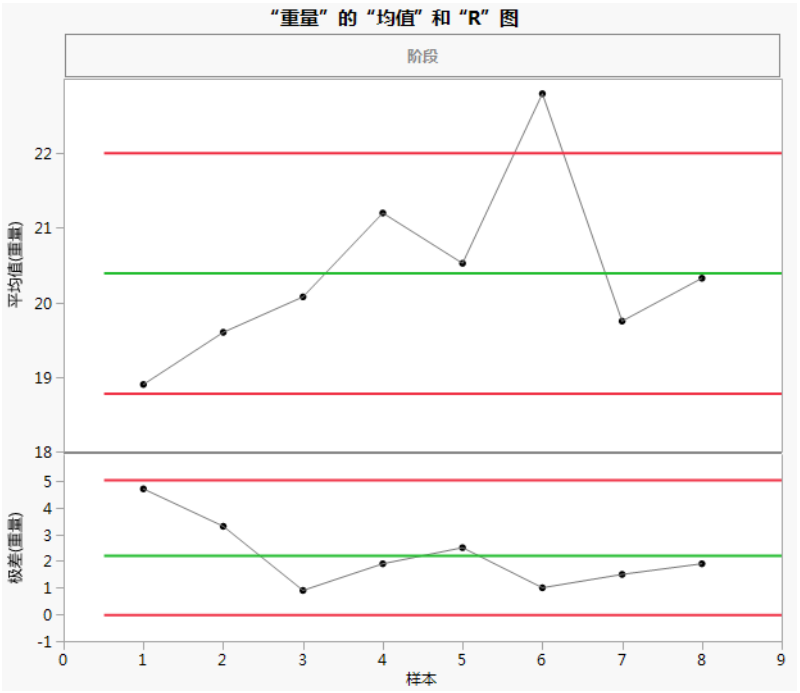
不过，使用“组内 Sigma”计算的点位于目标三角形内。这表示若过程稳定，**重量**值将很可能落入规格限范围内。

评估稳定性的控制图

使用 “控制图生成器”可确定**重量**测量值是否稳定。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Coating.jmp。
1. 选择**分析** > **质量和过程** > **控制图生成器**。
2. 将**样本**拖至**子组**区域。
3. 将**重量**拖至 Y 区域。

图 7.26 “重量”的均值和 R 图



该控制图指示**重量**测量值不稳定。过程受特殊原因影响且不可预测。这使得对能力指标和不合格估计值的解释非常不可靠。由于过程不可预测，甚至基于“总 Sigma”的估计值也变得不可靠。

图 7.25 中的直方图显示了**重量**值的分布，以及叠加在直方图上使用了两种 sigma 估计值的正态密度曲线。使用“总 Sigma”估计值的正态曲线比使用“组内 Sigma”估计值的正态曲线更加扁平。该正态曲线更为分散，这是因为“总 Sigma”的估计值受到使过程不稳定的特殊原因影响而膨胀。若过程稳定，更窄的正态曲线将反映过程行为。

您还可以将 Cpk 估计值 (1.142) 与 Ppk 估计值 (0.814) 进行比较。Ppk 远小于 Cpk 这一事实也可证实这是一个不可预测的过程。Cpk 估计值是在过程获得稳定状态下得到的对能力的预测。

注意：“单项详细报表截止”首选项可确定默认情况下是否显示“单项报表”。若启用该首选项，则当过程变量数小于等于首选项中指定的数值时，默认情况下显示“单项报表”。您可以在“首选项”>“平台”>“过程能力”中更改该首选项。

模拟非正态过程 Ppk 的置信限

在本例中，您首先为 Tablet Measurements.jmp 中的三个非正态变量执行能力分析。随后使用“模拟”为变量**纯度**查找不合格百分比置信限。

非正态能力分析

若不想遵循以下步骤，您可以通过运行 Tablet Measurements.jmp 中的**过程能力**表脚本来获取这一部分中的结果。

1. 选择**帮助** > **样本数据库**，然后打开 Tablet Measurements.jmp。
2. 选择**分析** > **质量和过程** > **过程能力**。
3. 选择**重量**、**厚度**和**纯度**，然后点击 **Y，过程**。
4. 在右侧的**为选定列指定角色**列表中选择**重量**、**厚度**和**纯度**。
5. 打开**分布选项**分级显示项。
6. 从**分布**列表中，选择**最佳拟合**。
7. 点击**设置过程分布**。

   后缀 **& 分布 (最佳拟合)** 添加至右侧列表中的每个列名。
8. 点击**确定**。

   随即显示一个能力指标图，其中显示 Ppk 值。请注意，只有**厚度**变量显示在表示 Ppk = 1 的那条线的上方。**纯度**几乎就位于这条线上。尽管测量值数目 250 看似较大，估计的 Ppk 值仍然变异较大。出于此原因，您需要为真正的**纯度** Ppk 值构造置信区间。

   注意：由于未显示目标图，您可以推断出正态分布拟合并不是这三个变量中任一个变量的最佳拟合。
9. 点击“过程能力”红色小三角并选择**单项详细报表**。

   每个过程的最佳拟合有所不同。

   – **重量**：对数正态

- 厚度：SHASH
- 纯度：Weibull

构造模拟列

要使用 “模拟”实用程序估计 Ppk 置信限，您需要构造反映拟合 Weibull 分布的模拟公式。若不想遵循以下步骤，您可以通过运行**添加模拟列**表脚本来获取这一部分中的结果。

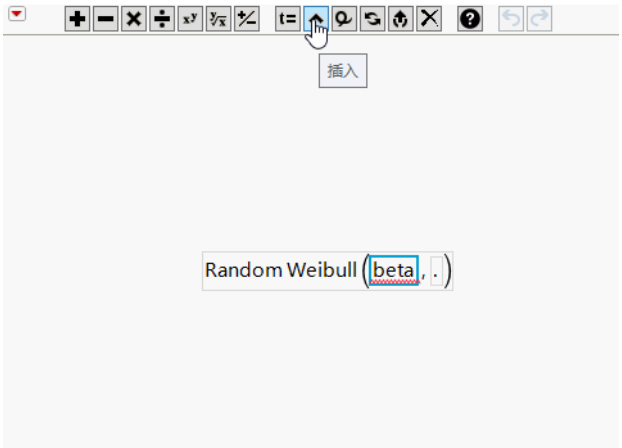
1. 滚至“纯度 (Weibull) 能力”报表并查找“参数估计值”报表。

图 7.27 “纯度”的 Weibull 参数估计值

这些是最佳拟合分布 （即 Weibull）的参数估计值。

2. 在 Tablet Measurements.jmp 样本数据表中，选择**列 > 新建列**。
3. 在**列名**旁边，输入**模拟纯度**。
4. 从**列属性**列表中，选择**公式**。
5. 在公式编辑器中，选择**随机 > 随机 Weibull**。
6. 随即选定 beta 的占位框。点击插入元素 (^)。

图 7.28 模拟的 “纯度”列的公式编辑器

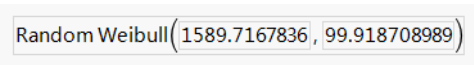


这将会添加参数 alpha 的占位框。

7. 在 “过程能力”报表中的 “纯度 (Weibull) 能力”下，右击 “参数估计值”报表表并选择**制成数据表**。
8. 复制**估计值**列第 2 行中的条目 (1589.7167836)。

9. 在公式编辑器窗口中，选择**随机 Weibull** 公式中 beta 的占位框并将 1589.7167836 粘贴到 beta 的占位框中。
10. 在您从“参数估计值”报表创建的数据表中，复制**估计值**列的第一行中的条目 (99.918708989)。
11. 在公式编辑器窗口中，选择**随机 Weibull** 公式中 alpha 的占位框并将 99.918708989 粘贴到 alpha 的占位框中。

图 7.29 完成的公式窗口



12. 在公式编辑器窗口中点击**确定**。
13. 在 Tablet Measurements.jmp 数据表中，右击 “模拟纯度”列并选择**列属性 > 规格限**。
14. 在**下规格限**旁边，键入 99.5。
15. 在 “新建列”窗口中点击**确定**。

    模拟纯度列包含模拟来自最佳拟合分布的值的公式。

模拟纯度 Ppk 和预期不合格百分比的置信区间

使用“模拟”时，将按照您指定的次数运行整个分析。为缩短计算时间，您可以通过仅运行所需的分析尽量减少计算负担。在本例中，由于您只对具有拟合的 Weibull 分布的**纯度**感兴趣，所以运行“模拟”之前，您只执行该分析。

注意：若您不在乎计算时间，您可以使用在上一节中创建的同一报表，并开始执行第 7 步。

1. 在“过程能力”报表中，点击“过程能力”红色小三角并选择**重新启动对话框**。
2. （可选）关闭 “过程能力”报表。
3. 在启动窗口中，从**为选定列指定角色**列表中，选择**重量 & 分布（对数正态）**和**厚度 & 分布 (SHASH)**。
4. 点击**删除**。
5. 点击**确定**。
6. 点击 “过程能力”红色小三角并选择**单项详细报表**。

   同时提供了 Ppk 值和 Ppl 值，但是这些值相同，因为**纯度**仅有下规格限。
7. 在 “总 sigma 能力”报表中，右击**估计值**列并选择**模拟**。

   在**换出列**列表中，确保选定**纯度**。在**换入列**列表中，确保选定**模拟纯度**。
8. 在**样本数**旁边，键入 500。

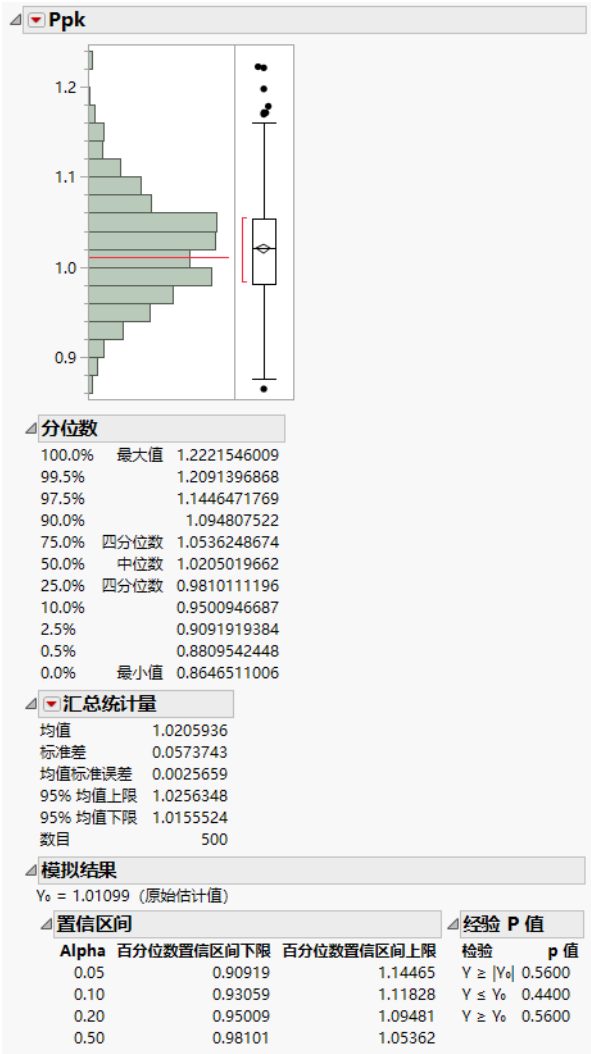
   注意：下一步非必需步骤。但该步可以确保您得到的模拟值与本例所显示的完全一致。
9. （可选）在**随机种子**旁边，键入 12345。

10. 点击**确定**。

    计算过程需要几秒钟的时间。随即显示名为“过程能力模拟结果（估计值）”的数据表。该表中的 Ppk 和 Ppl 列各自包含基于**模拟纯度**公式计算的 500 个值。第一行 （已排除）包含最初分析中得到的**纯度**的值。由于**纯度**仅有下规格限，所以 Ppk 值与 Ppl 值相同。

11. 在 “过程能力模拟结果 （估计值）”数据表中，点击**分布**脚本旁边的绿色小三角。

图 7.30 “纯度”的模拟 Ppk 值的分布



注意：根据 “模拟纯度”列公式中参数的小数精度，您的值可能与所示值略有不同。

显示了两个分布报表，一个针对 Ppk，另一个针对 Ppl。但是**纯度**仅有下规格限，所以 Ppk 与 Ppl 的值相同。出于此原因，两个分布报表完全相同。

“模拟结果”报表显示 Ppk 的 95% 置信区间介于 0.909 到 1.145 之间。修正偏倚后的 Ppk 95% 置信区间介于 0.899 到 1.124 之间。这两个置信区间都表明：真正的 Ppk 值可能高于 1.0，那么**纯度**就会位于您在第 198 页的“非正态能力分析”中构造的“能力指标图”的 Ppk = 1 线条上方。

12. 在“过程能力”报表中，右击“不合格”报表中的**期望总体百分比**列，然后选择**模拟**。

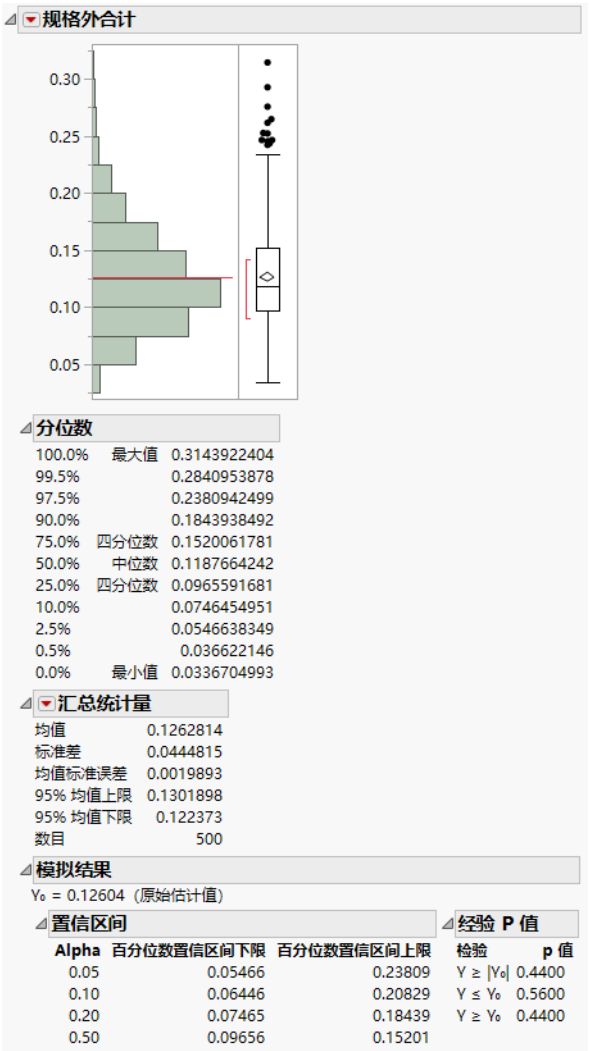
    在**换出列**列表中，确保选定**纯度**。在**换入列**列表中，确保选定**模拟纯度**。

13. 在**样本数**旁边，输入 500。
14. （可选）在**随机种子**旁边，输入 12345。
15. 点击**确定**。

    计算过程需要几秒钟的时间。随即显示名为“过程能力模拟结果（期望总体百分比）”的数据表。由于**纯度**仅有下规格限，所以**低于下规格限**值与**规格外合计**值相同。

16. 在“过程能力模拟结果（期望总体百分比）”数据表中，点击**分布**脚本旁边的绿色小三角。

图 7.31 “纯度”的模拟规格外合计值的分布



注意：根据 “模拟纯度”列公式中参数的小数精度，您的值可能与所示值略有不同。

同样，显示了两个相同的分布报表。“模拟结果”报表显示 “期望不合格总体百分比”的 95% 置信区间介于 0.055 到 0.238 之间。修正偏倚的置信区间介于 0.062 到 0.253 之间。

“过程能力”平台的统计详细信息

- “变异统计量”

- “用于目标图和能力箱线图的符号”
- “目标图”
- “缺失目标的过程的能力箱线图”
- “正态分布的能力指标”
- “非正态分布的能力指标：百分位数和 Z 得分方法”
- “分布的参数化”

变异统计量

使用 σ 表示过程的标准差。“过程能力”平台提供了两种类型的能力指标。Ppk 指标基于 σ 的估计值，它以不依赖于子组的方式使用所有数据。这一总估计值可能会反映特殊原因以及普通原因变异。Cpk 指标基于试图仅捕获普通原因变异的估计值。Cpk 指标使用 σ 的子组内或组间和组内子组估计值来构造。这样，它们试图反映真实的过程标准差。若过程不稳定，σ 的不同估计值会有明显差异。

总 Sigma

总 sigma 不依赖于子组。 σ 的总体估计值计算如下：

$$\hat{\sigma} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

该公式使用以下符号：

N = 整个数据集中的非缺失值个数

y_i = 第 i 个观测的值

$\bar{y}$ = 整个数据集中的非缺失值的均值

警告：若过程稳定，“总 Sigma”估计过程标准差。若过程不稳定，σ 的总估计值是不可靠值，因为过程标准差未知。

基于子组内变异的 Sigma 的估计值

基于子组内变异的 σ 的估计值可通过以下方式之一构造：

- 通过极差平均值估计的组内 sigma
- 通过无偏标准差的平均值估计的组内 sigma
- 通过移动极差估计的组内 sigma
- 无偏合并标准差估计的组内 sigma

若您在启动窗口中指定了子组 ID 列或常数子组大小，则可以指定首选子组内变异统计量。请参见第 164 页的“启动“过程能力”平台”。若不指定子组 ID 列、常数子组大小或历史 sigma，JMP 将使用第三种方法（大小为 2 的子组的移动极差）估计组内 sigma。

基于极差平均值的组内 sigma

通过**极差平均值**估计的组内 sigma 与**均值**和 *R* 图的标准差估计值相同：

$$\hat{\sigma} = \frac{\frac{R_1}{d_2(n_1)} + \ldots + \frac{R_N}{d_2(n_N)}}{N}$$

该公式使用以下符号：

R_i = 第 i 个子组的极差

n_i = 第 i 个子组的样本大小

$d_2(n_i)$ = 服从单位标准差正态分布的 n_i 个自变量的极差期望值

$N = n_{\text{i}} \geq 2$ 的子组数

基于无偏标准差的平均值的组内 sigma

通过**无偏标准差的平均值**估计的组内 sigma 与**均值**和 *S* 图的标准差估计值相同：

$$\hat{\sigma} = \frac{\frac{s_1}{c_4(n_1)} + \ldots + \frac{s_N}{c_4(n_N)}}{N}$$

该公式使用以下符号：

n_i = 第 i 个子组的样本大小

$c_4(n_i)$ = 服从单位标准差正态分布的 n_i 个自变量的标准差期望值

$N = n_{\text{i}} \geq 2$ 的子组数

s_i = 第 i 个子组的样本标准差

基于平均移动极差的组内 Sigma

通过**平均移动极差**估计的组内 sigma 与 “单个测量值”和 “移动极差”图的标准差的估计值相同：

$$\hat{\sigma} = \frac{\overline{MR}}{d_2(2)}$$

该公式使用以下符号：

$\overline{MR}$ = 非缺失移动极差的均值，计算方式为 $(MR_2+MR_3+...+MR_N)/(N-1)$，其中 $MR_i = |y_i - y_{i-1}|$。

$d_2(2)$ = 服从单位标准差正态分布的两个自变量的极差期望值。

基于移动极差中位数的组内 Sigma

通过移动极差中位数估计的组内 sigma：

$$\hat{\sigma} = \frac{MMR}{0.954}$$

该公式使用以下符号：

MMR = 非缺失移动极差的中位数，计算方式为 Median($MR_2, MR_3,..., MR_N$)，其中， $MR_i = |y_i - y_{i-1}|$。

基于无偏合并标准差的组内 Sigma

通过**无偏合并标准差**估计的组内 sigma：

$$\hat{\sigma} = \frac{\sqrt{(n_1-1)s_1^2 + \ldots + (n_N-1)s_N^2}}{c_4(n)\sqrt{n_1 + \ldots + n_2 - N}}$$

该公式使用以下符号：

n_i = 第 i 个子组的样本大小

$n = n_1 + \ldots + n_N$，合计样本大小

$c_4(n)$ = 服从单位标准差正态分布的 n 个自变量的标准差的期望值

$N = n_i \geq 2$ 的子组数

s_i = 第 i 个子组的样本标准差

基于组间变异的 Sigma 估计值

基于移动极差的组间 Sigma

基于子组间变异的 σ 的估计值通过子组均值的移动极差来估计：

$$\hat{\sigma} = \sqrt{\left(\frac{\overline{MR}}{d_2(2)}\right)^2 - \frac{\hat{\sigma}^2_{\text{组内}}}{H}}$$

该公式使用以下符号：

$\overline{MR}$ = 非缺失移动极差的均值，计算方式为 $(MR_2+MR_3+...+MR_N)/(N-1)$，其中 $MR_i = |y_i - y_{i-1}|$。

$d_2(2)$ = 服从单位标准差正态分布的两个自变量的极差期望值。

$\sigma^2_{\text{组内}}$ = 指定的组内 sigma 估计值。

$H = \dfrac{N}{\frac{1}{n_1} + \frac{1}{n_2} + \ldots + \frac{1}{n_N}}$，子组样本大小的调和均值。

基于组间和组内变异的 sigma 估计值

组间和组内 Sigma

基于组合的组间和组内变异的 Sigma 估计值定义如下：

$$\hat{\sigma} = \sqrt{\hat{\sigma}^2_{\text{组内}} + \hat{\sigma}^2_{\text{组间}}}$$

用于目标图和能力箱线图的符号

用于目标图和能力箱线图的公式使用以下符号：

Y_{ij} = 过程 j 的第 i 个观测

$\bar{Y}_j$ = 过程 j 的观测均值

$SD(Y_j)$ = 过程 j 的观测标准差

T_j = 过程 j 的目标值

LSL_j = 过程 j 的下规格限

USL_j = 过程 j 的上规格限

目标图

本节提供有关根据“目标图”中绘制的规格量标准化的均值偏移和标准差的计算的详细信息。本节使用第 207 页的“用于目标图和能力箱线图的符号”中定义的符号。

根据第 j 列的规格限标准化的均值偏移和标准差定义如下：

$$\text{规格标准化均值} = \frac{\bar{Y}_j - T_j}{2 \times \min(T_j - LSL_j, USL_j - T_j)}$$

$$\text{规格标准化标准差} = \frac{SD(Y_j)}{2 \times \min(T_j - \text{LSL}_j, \text{USL}_j - T_j)}$$

注意：若 LSL_j 或 USL_j 缺失，则“目标图”坐标的分母中将使用目标与非缺失规格限之间距离的两倍。

缺失目标的过程的目标图各点

假定过程同时具有上规格限和下规格限，但没有目标。则使用第 207 页的“目标图”中提供的公式，并且用两个规格限的中点替代 T_j。

假定过程仅有一个规格限且没有目标。要获取目标图上某个点的 (x,y) 坐标，将使用过程的能力指标。（请参见第 209 页的“正态分布的能力指标”，了解理论均值和标准差的定义。）对于样本观测，存在以下关系：

$$C_{pu} = \frac{\text{USL}_j - \overline{Y}_j}{3SD(Y_j)}$$

$$C_{pl} = \frac{\overline{Y}_j - \text{LSL}_j}{3SD(Y_j)}$$

若某过程具有两个规格限并且目标位于限值中点，则该点在目标图上的 (x,y) 坐标满足以下关系：

$$C_{pu} = (0.5 - x)/3y$$

$$C_{pl} = (0.5 + x)/3y$$

要在仅有一个规格限且无目标的情况下获取坐标，将使用以下关系。要确定唯一点，则需要关于从原点出发的线条（各点落在其上）的斜率的假设。对于仅有上规格限的情况，假定斜率为 0.5；对于仅有下规格限的情况，假定斜率为 -0.5。当能力值等于 1 且目标图三角形的 Ppk 滑块设置为 1 时，这些斜率会使各点落在目标图三角线上。

考虑仅有上规格限且无目标的情况。使用以下假设：(x,y) 坐标落在从原点出发斜率为 0.5 的线上，对 x 和 y 求解将得到以下坐标：

$$x = 1/(3C_{pu} + 2)$$
$$y = 1/(6C_{pu} + 4)$$

考虑仅有下规格限且无目标的情况。使用以下假设：(x,y) 坐标落在从原点出发斜率为 -0.5 的线上，对 x 和 y 求解将得到以下坐标：

$$x = -1/(3C_{pl} + 2)$$
$$y = 1/(6C_{pl} + 4)$$

注意：若 C_{pu} 或 C_{pl} 小于 -0.6，则在上面的公式中将设置为 -0.6。在值 -2/3 处，x 的分母假设值为 0。将能力值绑定在 -0.6 可防止分母假设值为 0 或切换符号。

缺失目标的过程的能力箱线图

无目标的列可同时具有上下规格限，或仅有一个规格限。本节使用第 207 页的“用于目标图和能力箱线图的符号”中定义的符号。

两个规格限且无目标

若没有为第 j 列指定目标，能力箱线图将基于变换观测的以下值：

$$Z_{ij} = \frac{Y_{ij} - (\mathrm{LSL}_j + \mathrm{USL}_j)/2}{\mathrm{USL}_j - \mathrm{LSL}_j}$$

单个规格限且无目标

假定仅指定了下规格限。（与仅指定上规格限的情况相似。）

若没有为第 j 列指定目标，能力箱线图将基于变换观测的以下值：

$$Z_{ij} = \frac{Y_{ij} - \bar{Y}_j}{2(\bar{Y}_j - \mathrm{LSL}_j)}$$

注意：若某列仅有一个规格限且没有目标值，而且样本均值落在规格区间之外，则不标绘该列的能力箱线图。

正态分布的能力指标

本节提供有关正态数据的能力指标计算的详细信息。

对于具有均值 μ 和标准差 σ 的过程特征，基于总体的能力指标定义如下：对于样本观测，参数将被其估计值取代：

$$\mathrm{Cp} = \frac{USL - LSL}{6\sigma}$$

$$\mathrm{Cpl} = \frac{\mu - LSL}{3\sigma}$$

$$\mathrm{Cpu} = \frac{USL - \mu}{3\sigma}$$

$$\mathrm{Cpk} = \min(\mathrm{Cpl}, \mathrm{Cpu})$$

$$\text{Cpm} = \frac{\min(T - LSL, USL - T)}{3\sigma\sqrt{1 + \left(\frac{T - \mu}{\sigma}\right)^2}}$$

目标指标 = 3(Cp – Cpk)

这些公式使用以下符号：

LSL = 下规格限

USL = 上规格限

T = 目标值

对于“组内 Sigma”能力的估计值，σ 使用您指定的子组方法估计。对于“总 Sigma”能力的估计值，σ 使用样本标准差估计。若任一规格限缺失，包含缺失规格限的能力指标将报告为缺失。

注意：使用默认的 AIAG (Ppk) 标签时，基于“总 Sigma”的指标表示为 Pp、Ppl、Ppu 和 Ppk。若使用“总 Sigma”，指标 Cpm 的标签不变。本节中的公式使用 Cp 标签定义。

能力指标的置信区间

能力指标的置信区间仅可用于服从正态分布的过程。同时针对组内和总 Sigma 能力计算置信区间，并且显示在“单项详细报表”中。

Cp

Cp 的 100(1 - α)% 置信区间计算如下：

$$\left(\hat{\text{Cp}}\sqrt{\frac{\chi^2_{\alpha/2,\,df}}{df}}, \hat{\text{Cp}}\sqrt{\frac{\chi^2_{1-\alpha/2,\,df}}{df}}\right)$$

其中

$\hat{\text{Cp}}$ 是 Cp 的估计值

$\chi^2_{\alpha/2,\,df}$ 是具有 *df* 个自由度的卡方分布的第 (α/2) 分位数。

df 是自由度

N 是观测数

m 是子组数

对于“总 sigma”能力，自由度等于 *N* - 1。

对于组内 Sigma 能力，自由度取决于子组和组内 sigma 估计方法。

- 对于具有不平衡子组的组内 Sigma 能力，无论采用哪种组内 sigma 估计方法，自由度的计算都相同。自由度等于 *N* - *m*。

- 对于具有大小为 $n = N/m$ 的平衡子组的组内 Sigma 能力，自由度的计算取决于组内 sigma 估计方法。
  - 通过无偏标准差的平均值估计组内 Sigma 时，自由度等于 $f * (N - m)$。尺度因子 f（在 0.875 和 1 之间变动）定义如下：

$$f = \frac{1}{2(n-1)\left(\frac{(n-1)}{2}\left(\frac{\Gamma\left(\frac{n-1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)}\right)^2 - 1\right)}$$

    其中，$\Gamma(n)$ 是在 n 处计算出的 gamma 函数。

    详细信息，请参见 Bissell (1990)。
  - 通过极差的平均值估计组内 Sigma 时，自由度计算为 $df = 1/A - (3/16) * A + (3/64) * A^2 + 0.25$。$A$ 定义如下：

$$A = \frac{2d_3(n)^2}{m \cdot d_2(n)^2}$$

    $d_2(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的期望值

    $d_3(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的标准差

    详细信息，请参见 David (1951)。
  - 通过无偏合并标准差估计组内 Sigma 时，自由度等于 $N - m$。
- 对于无子组的组内 Sigma 能力，自由度的计算取决于组内 sigma 估计方法。
  - 通过平均移动极差估计组内 Sigma 时，自由度计算为 0.62 * (N - 1)。
  - 通过移动极差的中位数估计组内 Sigma 时，自由度计算为 0.32 * (N - 1)。

  详细信息，请参见 Wheeler (2004, p. 82)。

Cpk

Cpk 的 100(1 - α)% 置信区间计算如下：

$$\left(\hat{C}_{pk}\left[1 - \Phi^{-1}{}_{1-\alpha/2}\sqrt{\frac{1}{9N\hat{C}_{pk}^2} + \frac{1}{2df}}\right], \hat{C}_{pk}\left[1 + \Phi^{-1}{}_{1-\alpha/2}\sqrt{\frac{1}{9N\hat{C}_{pk}^2} + \frac{1}{2df}}\right]\right)$$

其中

$\hat{C}_{pk}$ 是 Cpk 的估计值

$\Phi^{-1}{}_{1-\alpha/2}$ 是标准正态分布的第 (1 - α/2) 个分位数

df 是自由度

N 是观测数

m 是子组数

对于“总 sigma”能力，自由度等于 $N - 1$。

对于组内 Sigma 能力，自由度取决于子组和组内 sigma 估计方法。

- 对于具有不平衡子组的组内 Sigma 能力，无论采用哪种组内 sigma 估计方法，自由度的计算都相同。自由度等于 $N - m$。
- 对于具有大小为 $n = N/m$ 的平衡子组的组内 Sigma 能力，自由度的计算取决于组内 sigma 估计方法。
  - 通过无偏标准差的平均值估计组内 Sigma 时，自由度等于 $f^*(N - m)$。尺度因子 f（在 0.875 和 1 之间变动）定义如下：

$$f = \frac{1}{2(n-1)\left(\frac{(n-1)}{2}\left(\frac{\Gamma\left(\frac{n-1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)}\right)^2 - 1\right)}$$

    其中，$\Gamma(n)$ 是在 n 处计算出的 gamma 函数。

    详细信息，请参见 Bissell (1990)。
  - 通过极差的平均值估计组内 Sigma 时，自由度计算为 $df = 1/A - (3/16) * A + (3/64) * A^2 + 0.25$。$A$ 定义如下：

$$A = \frac{2d_3(n)^2}{m \cdot d_2(n)^2}$$

    $d_2(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的期望值

    $d_3(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的标准差

    详细信息，请参见 David (1951)。
  - 通过无偏合并标准差估计组内 Sigma 时，自由度等于 $N - m$。
- 对于无子组的组内 Sigma 能力，自由度的计算取决于组内 sigma 估计方法。
  - 通过平均移动极差估计组内 Sigma 时，自由度计算为 $0.62 * (N - 1)$。
  - 通过移动极差的中位数估计组内 Sigma 时，自由度计算为 $0.32 * (N - 1)$。

  详细信息，请参见 Wheeler (2004, p. 82)。

Cpm

注意：仅当目标值位于下规格限和上规格限的中心时才计算 Cpm 的置信区间。

Cpm 的 100(1 - α)% 置信区间计算如下：

$$\left(\hat{Cpm}\sqrt{\frac{\chi^2_{\alpha/2,\gamma}}{\gamma}}, \hat{Cpm}\sqrt{\frac{\chi^2_{1-\alpha/2,\gamma}}{\gamma}} \right)$$

其中

$\hat{Cpm}$ 是 Cpm 的估计值

$\chi^2_{\alpha/2,\gamma}$ 是具有 γ 个自由度的卡方分布的第 (α/2) 个分位数。

$$\gamma = \frac{N\left(1+\left(\frac{\bar{x}-T}{s}\right)^2\right)^2}{1+2\left(\frac{\bar{x}-T}{s}\right)^2}$$

N 是观测数

$\bar{x}$ 是观测的均值

T 是目标值

s 是 sigma 估计值

对于总 Sigma 能力，s 是“总 Sigma”估计值。对于“组内 Sigma”能力，s 将被“组内 Sigma”估计值取代。

提示：有关 Cp、Cpk 和 Cpm 的置信区间的详细信息，请参见 Pearn and Kotz (2006)。

Cpl 和 Cpu

Cpl 和 Cpu 的上下置信限使用 Chou et al. (1990) 有关 T 统计量的说明。

Cpl 的 100(1 - α)% 置信限 （用 CPL_L 和 CPL_U 表示）满足以下等式：

$$\Pr[t_{n-1}(\delta_L) \geq 3\hat{C}_{pl}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_L = 3CPL_L\sqrt{n}$$

$$\Pr[t_{n-1}(\delta_U) \leq 3\hat{C}_{pl}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_U = 3CPL_U\sqrt{n}$$

其中

$t_{n\text{-}1}(\delta)$ 服从有 n - 1 个自由度和非中心参数 δ 的非中心 t 分布

$\hat{C}_{pl}$ 是 Cpl 的估计值

Cpu 的 100(1 - α)% 置信限 （用 CPU_L 和 CPU_U 表示）满足以下等式：

$$\Pr[t_{n-1}(\delta_L) \geq 3\hat{C}_{pu}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_L = 3CPU_L\sqrt{n}$$

$\Pr[t_{n-1}(\delta_U) \le 3\hat{C}_{pu}\sqrt{n}] = \alpha/2$ 其中 $\delta_U = 3CPU_U\sqrt{n}$

其中

$t_{n-1}(\delta)$ 服从有 n - 1 个自由度和非中心参数 δ 的非中心 t 分布

$\hat{C}_{pu}$ 是 Cpu 的估计值

非正态分布的能力指标：百分位数和 Z 得分方法

本节说明如何为非正态分布计算能力指标。说明了两种方法：百分位数（亦称 ISO/分位数）方法和 Z 得分（亦称 Bothe/Z 得分）方法。为某个非正态过程变量选择分布时，可以拟合参数分布或非参数分布。您可以使用百分位数或 Z 得分方法来计算相关过程变量的能力指标。不过，除非数据量很大，否则非参数拟合可能无法精确反映分布尾部的行为。

注意：对于百分位数或 Z 得分方法，若数据服从正态分布，则能力公式简化为基于正态性的能力指标的公式。

这两种方法的描述使用以下符号：

LSL = 下规格限

USL = 上规格限

T = 目标值

百分位数（ISO/ 分位数）方法

百分位数方法用拟合分布的中位数替换了标准能力公式中的均值，用相应的百分位数极差替换了 6σ 值范围。AIAG (2005) 中对该方法进行了说明。

用 P_α 表示拟合分布的第 α*100 百分位数。那么，百分位数方法能力指标定义如下：

$$P_{pk} = \min\left(\frac{P_{0.5} - LSL}{P_{0.5} - P_{0.00135}}, \frac{USL - P_{0.5}}{P_{0.99865} - P_{0.5}}\right)$$

$$P_{pl} = \frac{P_{0.5} - LSL}{P_{0.5} - P_{0.00135}}$$

$$P_{pu} = \frac{USL - P_{0.5}}{P_{0.99865} - P_{0.5}}$$

$$P_p = \frac{USL - LSL}{P_{0.99865} - P_{0.00135}}$$

$$C_{pm} = \frac{\min\left(\frac{T - LSL}{P_{0.5} - P_{0.00135}}, \frac{USL - T}{P_{0.99865} - P_{0.5}}\right)}{\sqrt{1 + \left(\frac{\mu - T}{\sigma}\right)^2}}$$

Z 得分（Bothe/Z 得分）方法

Z 得分方法将规格限转换为在标准正态尺度上具有相同概率的值。它计算与正态分布（其风险水平与拟合的非正态分布相同）对应的能力测度。

用 *F* 表示具有下规格限 (LSL) 和上规格限 (USL) 的某个过程变量的拟合分布。等效的标准正态规格限定义如下：

$$\begin{bmatrix} LSL_F = \Phi^{-1}(F(LSL)) \\ USL_F = \Phi^{-1}(F(USL)) \end{bmatrix}$$

那么， Z 得分方法能力指标定义如下：

$$P_{pk} = \min(-LSL_F/3, USL_F/3)$$

$$P_{pl} = -LSL_F/3$$

$$P_{pu} = USL_F/3$$

$$P_p = (USL_F - LSL_F)/6$$

注意：由于 Cpm 是一个基于目标的测度，所以无法使用 Z 得分方法计算。

注意：对于很符合要求的数据，*F(LSL)* 或 *F(USL)* 可能相应地非常接近 0 或 1，以致于无法计算 LSL_F 或 USL_F。在这些情况下，JMP 默认情况下会从“Z 得分”方法自动切换为“百分位数”方法。这会提供更富有意义的能力指标。要禁用该默认设置，请选择“文件”>“首选项”>“平台”>“过程能力”。

分布的参数化

本节为“过程能力”平台中使用的分布提供密度函数 *f*，同时还为除 Johnson 和 SHASH 分布之外的所有分布提供期望值和方差。

正态

$$f(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left[-\frac{1}{2\sigma^2}(x-\mu)^2\right],\ -\infty < x < \infty,\ -\infty < \mu < \infty,\ \sigma > 0$$

$$E(X) = \mu$$

$$Var(X) = \sigma^2$$

Beta

$$f(x|\alpha, \beta) = \frac{1}{B(\alpha, \beta)}x^{\alpha-1}(1-x)^{\beta-1},\ 0 \le x \le 1,\ \alpha > 0,\ \beta > 0$$

$$E(X) = \frac{\alpha}{\alpha+\beta}$$

$$Var(X) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

其中，$B(\cdot)$ 是 Beta 函数。

指数

$$f(x|\sigma) = \frac{1}{\sigma}\exp(-x/\sigma),\ x > 0,\ \sigma > 0$$

$$E(X) = \sigma$$

$$Var(X) = \sigma^2$$

Gamma

$$f(x|\alpha, \sigma) = \frac{1}{\Gamma(\alpha)\sigma^{\alpha}}x^{\alpha-1}\exp(-x/\sigma),\ x > 0,\ \alpha > 0,\ \sigma > 0$$

$$E(X) = \alpha\sigma$$

$$Var(X) = \alpha\sigma^2$$

其中，$\Gamma(\cdot)$ 是 gamma 函数。

Johnson

Johnson Su

$$f(x|\gamma, \delta, \sigma, \theta) = \frac{\delta}{\sigma}\left[1+\left(\frac{x-\theta}{\sigma}\right)^2\right]^{-1/2}\phi\left[\gamma+\delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)\right],\ -\infty < x, \theta, \gamma < \infty,\ \theta > 0,\ \delta > 0$$

Johnson Sb

$$f(x|\gamma, \delta, \sigma, \theta) = \phi\left[\gamma + \delta \ln\left(\frac{x-\theta}{\sigma-(x-\theta)}\right)\right]\left(\frac{\delta\sigma}{(x-\theta)(\sigma-(x-\theta))}\right), \theta < x < \theta + \sigma, \sigma > 0$$

Johnson SI

$$f(x|\gamma, \delta, \sigma, \theta) = \frac{\delta}{|x-\theta|}\phi\left[\gamma + \delta \ln\left(\frac{x-\theta}{\sigma}\right)\right], x > 0 \text{ 若 } \sigma = 1, x < 0 \text{ 若 } \sigma = -1$$

其中，$\phi(\cdot)$ 是标准正态概率密度函数。

对数正态

$$f(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \frac{\exp\left[\frac{-(\log(x)-\mu)^2}{2\sigma^2}\right]}{x}, x > 0, -\infty < \mu < \infty, \sigma > 0$$

$$E(X) = \exp(\mu + \sigma^2/2)$$

$$Var(X) = \exp(2(\mu + \sigma^2)) - \exp(2\mu + \sigma^2)$$

正态混合

分布的 “两正态混合”和 “三正态混合”选项共享以下参数化：

$$f(x|\mu_i, \sigma_i, \pi_i) = \sum_{i=1}^{k} \frac{\pi_i}{\sigma_i}\phi\left(\frac{x-\mu_i}{\sigma_i}\right)$$

$$E(X) = \sum_{i=1}^{k} \pi_i\mu_i$$

$$Var(X) = \sum_{i=1}^{k} \pi_i(\mu_i^2 + \sigma_i^2) - \left(\sum_{i=1}^{k} \pi_i\mu_i\right)^2$$

其中，μ_i、σ_i 和 π_i 分别是第 i 个组的均值、标准差和比例，并且 $\phi(\cdot)$ 是标准正态概率密度函数。对于“两正态混合”，k 等于 2。对于“三正态混合”分布，k 等于 3。为混合中的每个组估计单独的均值、标准差和占总体的比例。

SHASH

$$f(x|\gamma, \delta, \sigma, \theta) = \frac{\delta\cosh(w)}{\sqrt{\sigma^2 + (x-\theta)^2}}\phi[\sinh(w)], -\infty < \gamma, x, \theta < \infty, 0 < \delta, 0 < \sigma$$

其中

$\phi(\cdot)$ 是标准正态 pdf

$$w = \gamma + \delta \sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)$$

注意：当 $\gamma = 0$ 且 $\delta = 1$ 时，SHASH 分布等价于位置为 θ 且尺度为 σ 的正态分布。

Weibull

$$f(x|\alpha,\beta) = \frac{\beta}{\alpha^{\beta}} x^{\beta-1} \exp\left[-\left(\frac{x}{\alpha}\right)^{\beta}\right],\ \alpha > 0,\ \beta > 0$$

$$E(X) = \alpha\Gamma\left(1+\frac{1}{\beta}\right)$$

$$Var(X) = \alpha^2\left\{\Gamma\left(1+\frac{2}{\beta}\right) - \Gamma^2\left(1+\frac{1}{\beta}\right)\right\}$$

其中，$\Gamma(\cdot)$ 是 gamma 函数。

第 8 章

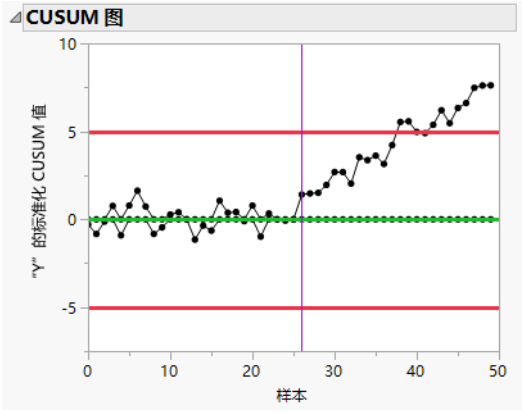
CUSUM 控制图

创建具有决策限的表格式 CUSUM 控制图

累积和 (CUSUM) 控制图支持您检测过程中的较小偏移。这些图有助于检测随时间出现的偏移（如逐渐漂移），以及不一定伴有突然偏移的偏移。“CUSUM 控制图”平台创建具有决策限的 CUSUM 图，类似于 Shewhart 图。该图亦称**表格式 CUSUM 图**。要创建 V 形模板累积和控制图，请参见第 291 页的“V 形模板 CUSUM 控制图”。

“CUSUM 控制图”平台还提供有关平均运行链长 (ARL) 的信息。**平均运行链长**是预计在出现失控信号之前出现的平均样本数或观测数。您可以使用平均运行链长评估在给定特定参数和假定方差一定的前提下 CUSUM 图的性能。

图 8.1 CUSUM 控制图



目录

“CUSUM 控制图”平台概述 221
CUSUM 控制图示例 221
启动“CUSUM 控制图”平台 223
“CUSUM 控制图”平台报表 224
CUSUM 控制图的控制面板 224
CUSUM 图 225
“CUSUM 控制图”平台选项 225
“平均运行链长 (ARL)”报表 227
CUSUM 控制图的更多示例 228
CUSUM 控制图中的“数据单位”选项的示例 228
带子组的 CUSUM 图的示例 229
“CUSUM 控制图”平台的统计详细信息 230
CUSUM 控制图构造的统计详细信息 230
偏移检测的统计详细信息 231
平均运行链长的统计详细信息 231

“CUSUM 控制图”平台概述

表格式 CUSUM 图包含两个单侧决策限图，这两个图叠加在一个图上。该图包含的决策限可指示过程何时失控，并且在图上怀疑出现偏移的位置放置一条偏移线。要使用“CUSUM 控制图”平台，您必须确定您认为重要的均值中的最小变化。您可以采用标准差单位或数据单位查看 CUSUM 控制图。有关表格式 CUSUM 图的详细信息，请参见 Woodall and Adams (1998) 和 Montgomery (2013)。

累积和控制图的另一种形式是 V 形模板图。要创建 V 形模板 CUSUM 图，请参见第 291 页的“V 形模板 CUSUM 控制图”。

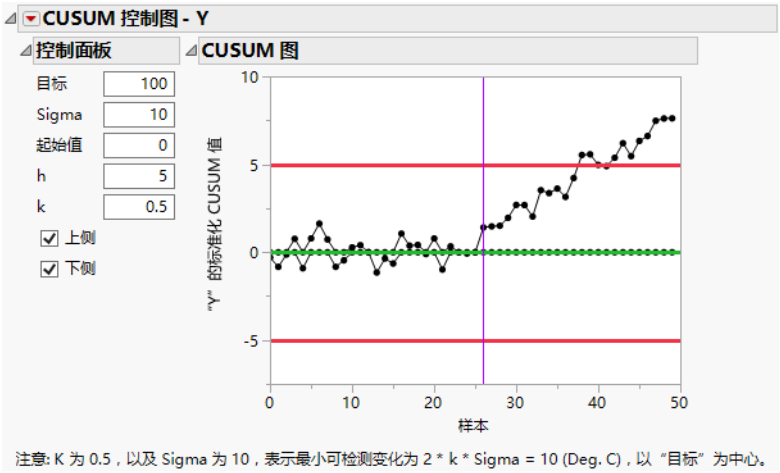
注意：“CUSUM 控制图”平台中的汇总结果不一定总能与 V 形模板 CUSUM 平台中的汇总结果吻合。具体而言，在同时选定“上侧”和“下侧”选项的情况下，双侧 V 形模板 CUSUM 图的汇总结果与 CUSUM 控制图的汇总结果不吻合。不过，“CUSUM 控制图”平台的单侧汇总报表与“V 形模板 CUSUM”平台的单侧汇总报表确实吻合。

CUSUM 控制图示例

您想要检测发动机温度中较小的偏移。数据表包含发动机恒温器的温度测量值。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Engine Temperature Sensor.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **CUSUM 控制图**。
3. 选择 Y 并点击 Y。
4. 点击**确定**。
5. 在“目标”框中，键入 100。
6. 在“Sigma”框中，键入 10。

图 8.2 “CUSUM 控制图”报表



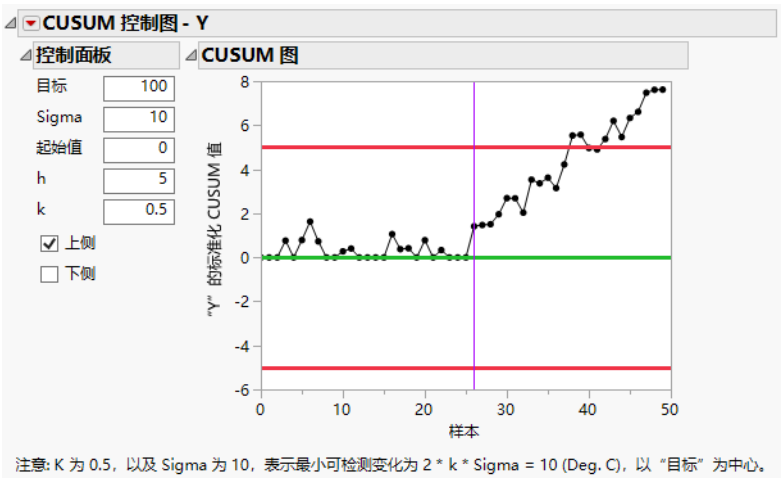
CUSUM 图上的垂线指示温度测量值中的偏移大约从样本 26 开始。

注意：通过运行 Engine Temperature Sensor.jmp 中的“IMR 图”表脚本，您可以将该结果与“单值移动极差”控制图作比较。IMR 图不触发任何 Nelson 检验。

单侧 CUSUM 控制图的示例

继续上一个示例，假定您只关注不断增大的温度变化。要将图 8.2 中的 CUSUM 控制图改为单侧图，请取消选中“下侧”复选框。执行上述操作时，负累积和对应的点将从图中删除。留下的 CUSUM 控制图只包含正累积和对应的点。

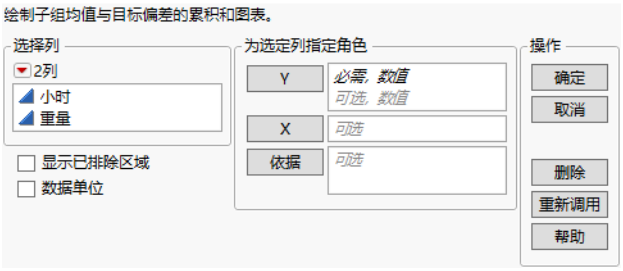
图 8.3 “单侧 CUSUM 控制图”报表



启动“CUSUM 控制图”平台

通过选择**分析** > **质量和过程** > **控制图** > **CUSUM 控制图**启动 “CUSUM 控制图”平台。

图 8.4 “CUSUM 控制图”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“CUSUM 控制图”平台启动窗口包含以下选项：

Y 标识您要绘图的变量。

注意：数据表中的行必须按照观测的收集顺序进行排序。

X 标识子组变量。CUSUM 图的水平轴由子组变量标记。若该列的某个值多次出现，则每个 X 值处的平均响应将标绘在 CUSUM 图上。

依据 为“依据”变量的每个水平生成单独报表。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

显示排除区域 （仅当指定 X 变量时才适用。）指定完全排除的子组显示在 CUSUM 控制图的水平轴上。

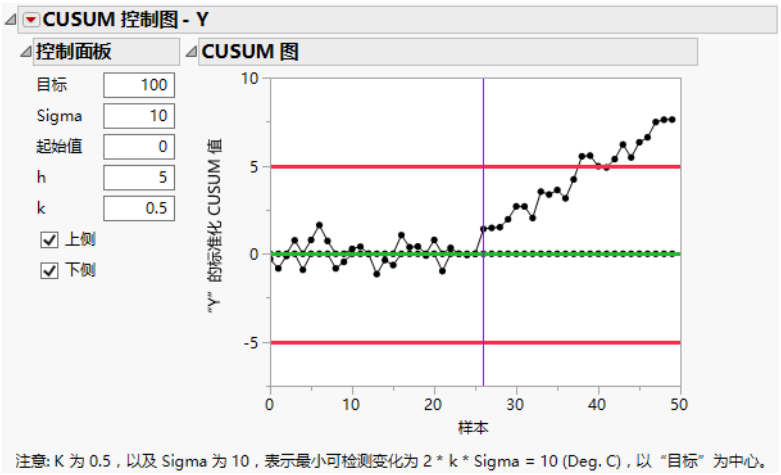
数据单位 指定在报表中使用数据单位而不是标准差单位。默认情况下，图和参数以标准差单位显示。不过，若您在启动窗口中选择“数据单位”选项，则图和参数以所分析的数据列的单位显示。

使用标准差单位时， h 和 k 参数的值不依赖于过程标准差。这可以算作一个优势。

“CUSUM 控制图”平台报表

默认情况下，“CUSUM 控制图”平台生成一个报表，其中包含参数控制面板和 CUSUM 图。

图 8.5 “CUSUM 控制图”报表



CUSUM 控制图的控制面板

“控制面板”报表包含图参数的当前值。当前值位于框中，在框中可以更新参数值。同时提供针对“上侧”和“下侧”的框。若在启动窗口中指定了“数据单位”选项，该设置将在“控制面板”中复选框的下方表示出来。

以下选项显示在 “控制面板”报表中：

目标 均值的已知值。这是图中中心线的值。默认情况下，该参数设置为 Y 列的 “规格限”列属性中的 “目标”值。若 Y 列不具有 “规格限”列属性中的 “目标”值，则该参数设置为 Y 列的总体平均值。

注意：若即使 Y 列在 “规格限”列属性中有 “目标”值也要将 Y 列的总体平均值用作中心线的值，请选择“使用过程均值作为中心线”平台首选项。该首选项位于“文件”>“首选项”>“平台”>“CUSUM 控制图”中。

Sigma 标准差的已知值。默认情况下，该参数设置为 Y 列的平均移动极差。若有 X 变量，Sigma 参数将设置为汇总数据的平均移动极差。

起始值 第一个样本前的累积和的值。以非零值为起始的累积和将增加 CUSUM 图在样本开始附近的灵敏度。该参数亦称快速初始响应 (FIR) 值。默认情况下，该参数设置为 0。

h 或 H 定义限值的参数的值。若未在启动窗口中选定“数据单位”选项，这就是 h 参数。若在启动窗口中选定“数据单位”选项，这就是 H 参数。请注意， H 等于 h 与 Sigma 的乘积。默认情况下， h 等于 5， H 等于 5 与 Sigma 的乘积。

k 或 K 用于定义均值中值得检测的最小变化的参数的值。若未在启动窗口中选定“数据单位”选项，这就是 k 参数。若在启动窗口中选定“数据单位”选项，这就是 K 参数。请注意，K 等于 k 与 Sigma 的乘积。默认情况下， k 等于 0.5， K 等于 Sigma 的一半。

上侧 显示或隐藏图上累积和的正值。这些值是 C^+ 值。

下侧 显示或隐藏图上累积和的负值。这些值是 C^- 值。

使用数据单位 出现这一文字即意味着已经在启动窗口中选定了“数据单位”选项，而且 CUSUM 图中的值被中心化而不是标准化。

CUSUM 图

“CUSUM 图”报表包含带有由图参数的当前值确定的决策限的累积和控制图。样本（若指定了 X 变量则为子组）在水平轴上表示。垂直轴表示累积和的正值和负值的中心化值。若未在启动窗口中选定“数据单位”选项，则垂直轴表示标准化响应值的累积和。若在启动窗口中选定“数据单位”选项，则垂直轴表示未标准化响应值的累积和。

“CUSUM 控制图”平台选项

“CUSUM 控制图”红色小三角菜单包含以下选项：

显示限值 显示或隐藏 CUSUM 图中的上下决策限。

显示中心线 显示或隐藏 CUSUM 图中的中心线。

显示偏移线 （仅当在数据中检测到偏移才可用。）显示或隐藏 CUSUM 图中指定偏移的垂直线。在偏移的起点处绘制偏移线。

- 当 C^+ 的值超过图中上限时，会出现正偏移。偏移的起点定义为 C^+ 的最近零值之后的第一个点。
- 当 C^- 的值落在图中下限的下方时，会出现负偏移。偏移的起点定义为 C^- 的最近零值之后的第一个点。

显示 ARL 显示或隐藏“平均运行链长 (ARL)”报表。请参见第 227 页的 “ “平均运行链长 (ARL)”报表”。

ARL 刻画器 显示或隐藏标绘平均运行链长与参数 h 和 k 的比较情况的刻画器。若在启动窗口中指定了 “数据单位”选项，刻画器将标绘平均运行链长与参数 H 和 K 的比较情况。

指定偏移的平均运行链长 (ARL) 是出现失控信号之前期望的平均试验次数。例如， 0 处的 ARL 表示在过程受控时出现假警报信号之前期望的平均试验数。若过程受控，则偏移大小为 0。

“ARL 刻画器”支持您探索各种参数设置如何影响相应的 CUSUM 图的性能。随着 “控制面板”报表中的参数的更新，“ARL 刻画器”也随之更新。理想的 CUSUM 图有较高的 ARL(0) 值和较低的 ARL(Δ) 值，其中 Δ 是关注的大小偏移。

“ARL 刻画器”依赖 “控制面板”报表中 “上侧”和 “下侧”选项的设置：

- 若同时选定“上侧”和“下侧”选项，刻画器则表示跨越 CUSUM 图上的上决策限或下决策限的平均运行链长。
- 若仅选定“上侧”选项，则刻画器表示 CUSUM 图上的上决策限的平均运行链长。
- 若仅选定“下侧”选项，则刻画器表示 CUSUM 图上的下决策限的平均运行链长。

有关“ARL 刻画器”旁边的红色小三角菜单中的选项的详细信息，请参见《刻画器指南》。

控制面板 显示或隐藏参数当前值的报表。该报表支持您更改参数值和 CUSUM 图的侧。

参数报表 显示或隐藏参数当前值的报表。

检验超出界限 在 CUSUM 图中高于上限或低于下限的任何点周围显示或隐藏一个红色圆圈。

保存汇总 创建包含 CUSUM 图中各个子组的统计量的新数据表。以下统计量保存到新数据表中：子组的数目和大小、子组样本均值、偏移开始的指示符、指示偏移开始之间的每个区间的值、上下累积和与相应的相邻试验计数，以及下控制限值和上控制限值。

调节图表 显示或隐藏支持您为 Y 变量设置可接受范围的控件。调整可接受范围的最小值和最大值，然后点击**完成**。此时， CUSUM 图将基于 k 参数的新值进行更新。

- 若在启动窗口中指定了“数据单位”选项，则插补的 k 参数是最小值和最大值的平均值除以 Sigma 参数的结果。
- 若未指定“数据单位”选项，则插补的 K 参数是最小值和最大值的平均值。

通过设置 Y 变量的可接受范围，可以设置 CUSUM 图的实际显著性。当检验区间较为频繁时，这尤其有用，这会导致实际平均运行链长短得多。

重置为默认值 将所有参数重置回其默认值。

警告脚本 支持您编写和运行指示数据未通过特殊原因检验的脚本。结果可以写入日志或大声说出，并且提供一个选项来包含对检验失败原因的解释。您还可以使用自定义脚本选项将结果发送到电子邮件。

在调用警告脚本时，以下变量在发出的脚本和随后的 JSL 脚本中都可用：

`qc_col` 是列名

`qc_test` 是失败的检验

qc_sample 是样本号

提示：指定警告脚本后，当启用 “检验超出界限”选项时，将调用警告脚本。

请参见 《Scripting Guide》，了解关于编写定制警告脚本的详细信息。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“平均运行链长 (ARL)”报表

“平均运行链长 (ARL)”报表包含 ARL 值的一个表和一个图。指定偏移的平均运行链长 (ARL) 是出现失控信号之前期望的平均试验次数。例如，0 处的 ARL 表示在过程受控时看到假警报信号之前期望的平均试验数。若过程受控，则偏移大小为 0。

ARL 报表中的表和图支持您探索各种参数设置如何影响相应的 CUSUM 图的性能。随着“控制面板”报表中的 h 和 k 参数的更新，ARL 报表也随之更新。理想的 CUSUM 图有较高的 ARL(0) 值和较低的 ARL(Δ) 值，其中 Δ 是关注的大小偏移。

“平均运行链长 (ARL)”报表依赖 “控制面板”报表中的 “上侧”和 “下侧”选项的设置。若仅选定一个选项， ARL 报表将使用相应的单侧 CUSUM 图的计算。若同时选定两个选项，ARL 报表将使用双侧 CUSUM 图的计算。请注意，双侧 ARL 值通过以下方程与正负单侧 ARL 值相关：

$$\frac{1}{\text{ARL}} = \frac{1}{\text{正 ARL}} + \frac{1}{\text{负 ARL}}$$

ARL 表

ARL 表显示 0 到 3 之间（以 0.25 为增量）的偏移 (Δ) 的平均运行链长。若指定了“数据单位”选项，则偏移用 2*K/Sigma^2 来表示。若未指定“数据单位”选项，则偏移用 2*k/Sigma 来表示。

ARL 图

ARL 图显示介于 0 和 3 之间的偏移 (Δ) 的平均运行链长。该图包含与 ARL 图左侧的 ARL 表相同的数据点。

CUSUM 控制图的更多示例

- “CUSUM 控制图中的“数据单位”选项的示例”
- “带子组的 CUSUM 图的示例”

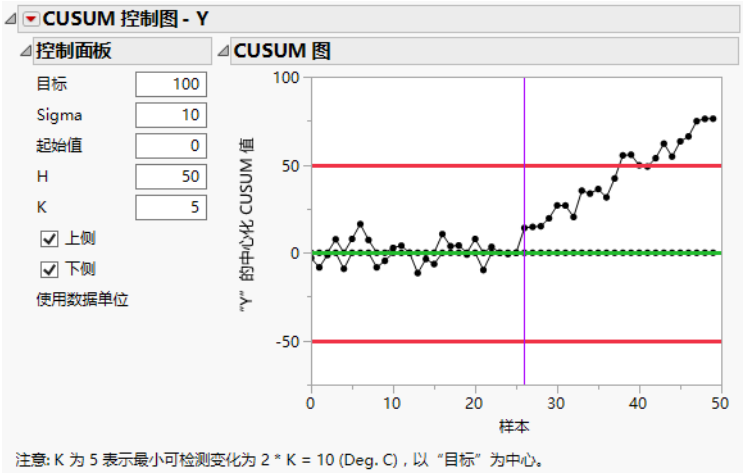
CUSUM 控制图中的“数据单位”选项的示例

本例使用 “数据单位”选项并重现第 221 页的 “CUSUM 控制图示例”中的分析。您想要检测发动机温度中较小的偏移。数据表包含发动机恒温器的温度测量值。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Engine Temperature Sensor.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **CUSUM 控制图**。
3. 选择 Y 并点击 Y。
4. 选择 “数据单位”旁边的框。
5. 点击**确定**。
6. 在 “目标”框中，键入 100。
7. 在 “Sigma”框中，键入 10。

   请注意，“起始值”下的选项为 H 和 K 而不是 h 和 k。这些参数现在以数据列单位指定，而不是以标准差单位指定。

图 8.6 “CUSUM 控制图”报表



类似于在使用 sigma 单位的示例中那样，CUSUM 图上的垂线指示温度测量值中的偏移大约从样本 26 开始。

带子组的 CUSUM 图的示例

利用一种机器来装填二冲程机油添加剂到 8 盎司油罐。装填过程被视为处于统计控制中。该过程经过设置，满罐的平均重量 (μ_0) 为 8.10 盎司。之前的分析显示装填重量的标准差 (σ_0) 为 0.05 盎司。

选取了四个油罐的子组样本，每小时称重一次，连续称了 12 小时。 Oil1 Cusum.jmp 数据表中的每个观测都包含一个重量值及其关联的小时值。观测经过排序，小时值采用升序排序。您希望能够检测过程中的 2σ 偏移。

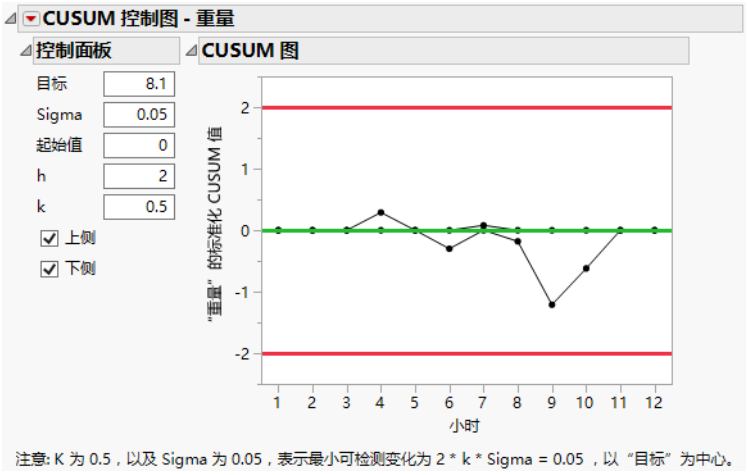
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Oil1 Cusum.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **CUSUM 控制图**。
3. 选择**体重**并点击 Y。
4. 选择**小时**并点击 X。
5. 点击**确定**。
6. 在 “目标” 框中，键入 8.1。

   这是过程的目标均值。
7. 在 “Sigma” 框中，键入 0.05。

   这是过程的已知标准差。
8. 在 “h” 框中，键入 2。

   这样可以定义决策限在每个方向上都是 2 个标准差。

图 8.7 带子组的 CUSUM 控制图



CUSUM 图不显示上决策限或下决策限之外的任何点。没有证据表明该过程出现了偏移。

注意：Montgomery (2013) 声明“只有存在某种显著的规模经济或者存在其他某种有效原因需要取大于 1 的样本大小，此时才应该对 CUSUM 使用大于 1 的子组大小。”在表格式 CUSUM 图中使用有理子组不一定总能改进该图的性能。

“CUSUM 控制图”平台的统计详细信息

- “CUSUM 控制图构造的统计详细信息”
- “偏移检测的统计详细信息”
- “平均运行链长的统计详细信息”

CUSUM 控制图构造的统计详细信息

本节定义构造 CUSUM 图时所用的统计量。其中一些统计量还保存在使用“保存汇总”命令创建的数据表中。

单侧上下累积和

C^+ 和 C^- 的定义依赖于“数据单位”选项的设置。

注意：在“保存汇总”数据表中，C^+ 和 C^- 分别标记为**上累积和**和**下累积和**。

采用标准化单位的累积和

若未选定“数据单位”选项，每一步的 C^+ 和 C^- 定义如下：

$$C_i^+ = \max\left(0, \frac{x_i - T}{\sigma} - k + C_{i-1}^+\right)$$

$$C_i^- = \min\left(0, \frac{x_i - T}{\sigma} + k + C_{i-1}^-\right)$$

其中：

x_i 是第 i 步的响应的值

T 是过程目标

σ 是过程的标准差

k 是参考值，以标准差为单位

若为“起始值”指定了某个值，该值将用作初始 C^+ 值，而该值的负值用作初始 C^- 值。否则，C^+ 和 C^- 的初始值为零。

采用数据单位的累积和

若选定 “数据单位”选项，每一步的 C^+ 和 C^- 定义如下：

$$C_i^+ = \max(0, (x_i - T) - K + C_{i-1}^+)$$

$$C_i^- = \min(0, (x_i - T) + K + C_{i-1}^-)$$

其中：

x_i 是第 i 步的响应的值

T 是过程目标

σ 是过程的标准差

k 是参考值，以数据为单位

若为“起始值”指定了某个值，该值将用作初始 C^+ 值，而该值的负值用作初始 C^- 值。否则，C^+ 和 C^- 的初始值为零。

正负试验的计数器

每一步的 N^+ 都是自 C^+ 的最近一个零值以来的步骤数。每一步的 N^- 都是自 C^- 的最近一个零值以来的步骤数。

注意：在“保存汇总”数据表中，N^+ 和 N^- 分别标记为**正试验数**和**负试验数**。

偏移检测的统计详细信息

当 C^+ 的值超过图中上限时，会出现正偏移。偏移的起点定义为 C^+ 的最近零值之后的第一个点。

当 C^- 的值超过图中下限时，会出现负偏移。偏移的起点定义为 C^- 的最近零值之后的第一个点。

平均运行链长的统计详细信息

使用 Goel and Wu (1971) 中所述的积分方程法（含 24 个高斯点）计算单侧平均运行链长 (ARL) 值。若“起始值”大于 0，则根据 Lucas and Crosier (1982) 的附录 A.1 中的方法计算值。

请注意，双侧 ARL 值通过以下方程与正负单侧 ARL 值相关：

$$\frac{1}{\text{ARL}} = \frac{1}{\text{正 ARL}} + \frac{1}{\text{负 ARL}}$$

Lucas and Crosier (1982) 描述了 CUSUM 图的“起始值”的属性，其中的初始 CUSUM S_0 设置为非零值。这有时称为快速初始响应 (FIR) 功能。他们给出的平均运行链长计算表明 FIR 功能在过程受控时影响极小，所以这导致对初始失控状况的响应速度比标准 CUSUM 图更快。

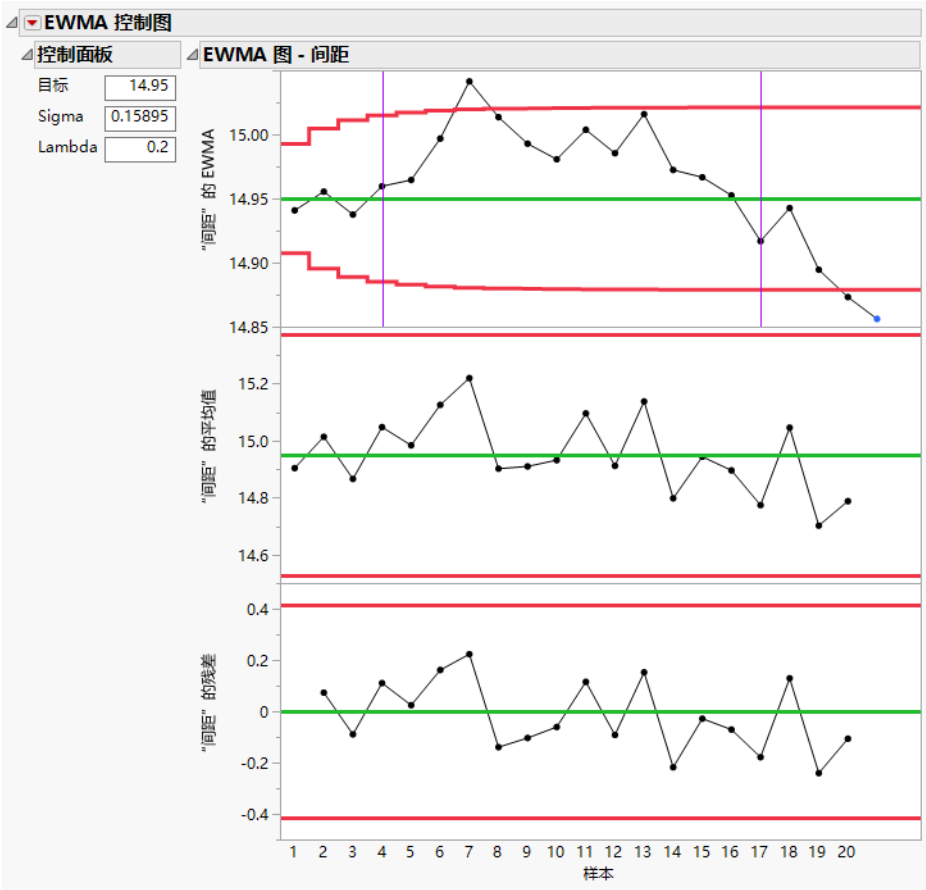
第 9 章

EWMA 控制图

创建指数加权控制图

指数加权移动平均 (EWMA) 图可用于检测过程中的较小偏移。指数加权移动平均 (EWMA) 图上的各点是所有以前的子组均值（包括当前子组样本的均值）的加权平均值。权重随着时间的回溯呈指数级下降。

图 9.1 “EWMA 控制图”报表



目录

“EWMA 控制图”平台概述 235
“EWMA 控制图”平台示例 235
启动“EWMA 控制图”平台 237
“EWMA 控制图”平台报表 238
EWMA 控制图的控制面板 238
“EWMA 图”报表 239
“EWMA 控制图”平台选项 240
EWMA 控制图的“平均运行链长 (ARL)”报表 242
“EWMA 控制图”平台的统计详细信息 242

“EWMA 控制图”平台概述

指数加权移动平均 (EWMA) 图可用于检测过程中的较小偏移。指数加权移动平均 (EWMA) 图上的各点是所有以前的子组均值（包括当前子组样本的均值）的加权平均值。权重随着时间的回溯呈指数级下降。有关指数加权移动平均图的详细信息，请参见 Montgomery (2013)。

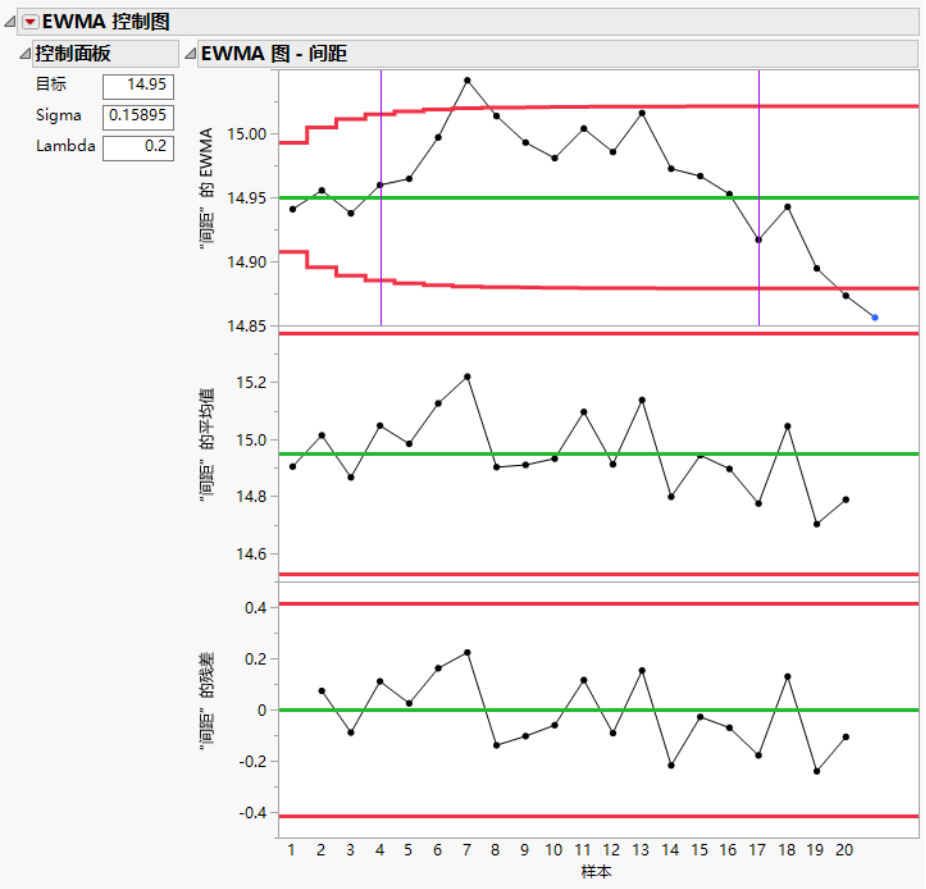
“EWMA 控制图”平台将 EWMA 图与 X 图和残差图配对。若未指定“子组”变量，则 X 图是单个测量值图。若指定了“子组”变量而且至少有一个子组的大小大于 1，则 X 图是均值图。

“EWMA 控制图”平台示例

在样本数据表 Clips1.jmp 中，关注的测量是生产的金属夹两端之间的间距。要监控平均间距的变化过程，每天选取五个金属夹一组的子组样本。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Clips1.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **EWMA 控制图**。
3. 选择**间距**并点击 **Y**。
4. 选择**样本**并点击**子组**。
5. 点击**确定**。

图 9.2 EWMA 控制图



EWMA 图中的紫色垂直线表示偏移。在样本 4 和 17 处检测到偏移开始。

启动 “EWMA 控制图” 平台

通过选择**分析** > **质量和过程** > **控制图** > **EWMA 控制图**启动 “EWMA 控制图” 平台。

图 9.3 “EWMA 控制图” 启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“EWMA 控制图” 平台启动窗口包含以下选项：

Y 标识您要绘图的变量。

注意：若未指定 “子组” 变量，数据表中的行必须按照观测的收集顺序进行排序。

子组 标识子组变量。 EWMA 图的水平轴由子组变量标记。

依据 为 “依据” 变量的每个水平生成单独报表。若指定了多个 “依据” 变量，将为 “依据” 变量水平的每种可能组合生成单独的报表。

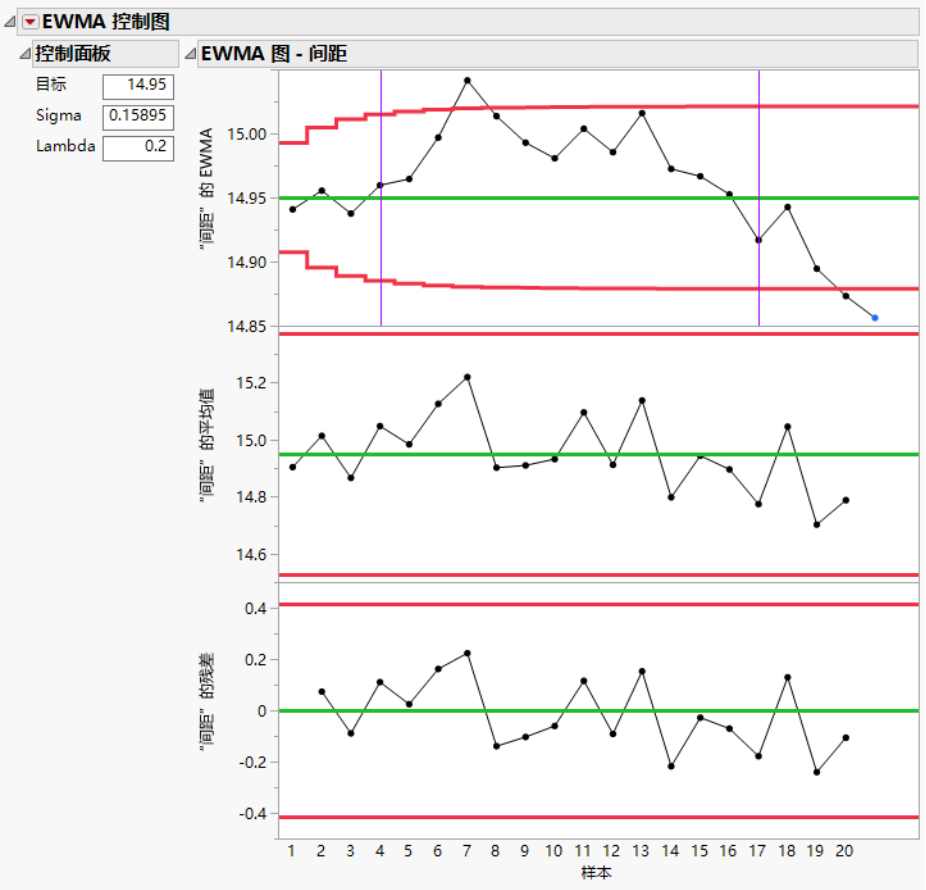
中心数据 指定通过从每个观测中减去目标以使数据中心化。

显示排除区域 （仅当指定 “子组” 变量时才适用。）指定在 EWMA 控制图中显示数据表中完全排除的子组。

“EWMA 控制图”平台报表

默认情况下，“EWMA 控制图”平台生成包含参数控制面板、EWMA 图、X 图和残差图的报表。

图 9.4 “EWMA 控制图”报表



EWMA 控制图的控制面板

“控制面板”包含图参数的当前值。当前值位于框中，在框中可以更新参数值。

以下选项显示在 “控制面板”中：

目标 均值的已知值。这是图中中心线的值。默认情况下，该参数设置为 Y 列的 “规格限”列属性中的 “目标”值。若 Y 列不具有 “规格限”列属性中的 “目标”值，则该参数设置为 Y 列的总体平均值。

在以下情况下无法编辑 “目标”值：

- 若在启动窗口中指定了“中心数据”选项。
- 若 *Y* 变量具有包含“目标”值的“规格限”列属性。不过，您可以选择红色小三角菜单中的“使用总体均值作为目标”选项，在“规格限”列属性中的“目标”值与 *Y* 变量的总体均值之间进行切换。

Sigma 标准差的已知值。默认情况下，该参数设置为 Y 列的平均移动极差。若存在 “子组”变量，则将 Sigma 参数设置为子组均值移动极差的平均值。

注意：默认的 Sigma 值计算不包括排除的行。

Lambda 用于加权先验样本的平滑常数的值。默认情况下，该参数设置为 0.2。

*** 使用中心化数据** 出现该文本表示已在启动窗口中选择 “中心化数据”选项， EWMA 图和 X 图中的值以 “目标”值中心化。

“EWMA 图”报表

“EWMA 图”报表包含三个图表：EWMA 图、 X 图和残差图。

EWMA 图

EWMA 图是一个指数加权移动平均 (EWMA) 图，其决策限由图参数的当前值确定。样本（若指定了“子组”变量则为子组）在水平轴上表示。垂直轴表示指数加权移动平均值。若在启动窗口中指定了“中心数据”选项，则垂直轴表示已减去目标值的指数加权移动平均值。每个样本或子组在图上都有一个点。另外还有一个点是预测点，显示为蓝色。

注意：若最后一个样本（或子组）同时被隐藏和排除，则不会绘制将最后一个样本（或子组）连接到预测点的线。

X 图

X 图是值的单个测量值图（若指定了至少一个子组大小大于 1 的“子组”变量，则为均值的均值图）。样本（若指定了至少一个子组大小大于 1 的“子组”变量，则为子组）在水平轴上表示。垂直轴表示测量值（或子组均值）。若在启动窗口中选定“中心数据”选项，则垂直轴表示已减去目标值的测量值（或子组均值）。每个样本或子组在图上都有一个点。有关 X 图上的限值的详细信息，请参见“控制图生成器”一章中第 82 页的 “ “控制图生成器”的统计详细信息”。

残差图

残差图是样本值（若指定了“子组”变量，则为子组均值）与上一个样本（或子组）的 EWMA 值之间的差值图。该图支持您直观地检查自相关。第 i 个残差计算为 $r_i = X_i - EWMA_{i-1}$，其中 X_i 表示第 i 个样本值（或子组均值），$EWMA_{i-1}$ 表示第 (i-1) 个 EWMA 值。残差图上的限值为 ±3*$ResidSigma$，其中 $ResidSigma$ 是残差的标准差。

“EWMA 控制图”平台选项

“EWMA 控制图”红色小三角菜单包含以下选项：

显示限值 显示或隐藏 EWMA 图、 X 图和残差图中的上下决策限。

显示中心线 显示或隐藏 EWMA 图、 X 图和残差图中的中心线。

显示偏移线 显示或隐藏 EWMA 图中指定为偏移的垂直线。在偏移的起点处绘制偏移线。偏移开始定义为 EWMA 值在特定方向上穿过中心线后的第一个点。

- 当 EWMA 值超过图中上限时，会出现正偏移。偏移的起点定义为“目标”线下方的最近 EWMA 值之后的第一个点。
- 当 EWMA 值落在图中下限的下方时，会出现负偏移。偏移的起点定义为“目标”线上方的最近 EWMA 值之后的第一个点。

检验超出界限 在 EWMA 图和均值图中高于上限或低于下限的任何点周围显示或隐藏一个红色圆圈。

显示 ARL 显示或隐藏 “平均运行链长 (ARL)”报表。请参见第 242 页的 “EWMA 控制图的“平均运行链长 (ARL)”报表”。

控制面板 显示或隐藏参数当前值的报表。该报表支持您更改 EWMA 图中的参数值。

参数报表 显示或隐藏参数当前值的报表。

常数限值 指定使用渐近表达式计算 EWMA 图限值，以便 EWMA 图上的限值为常数。

警告：当子组间的样本大小不等时，“常数限值”选项无效。

保存汇总 创建包含 EWMA 中各个子组的统计量的新数据表。以下统计量保存到新数据表中：子组编号、子组标签、子组大小、子组均值、偏移开始的指示符、指示偏移开始之间的每个区间的值、每个子组的指数加权移动平均值、正负依次相连运行计数的数量，以及下控制限值和上控制限值。预测值保存在汇总表的最后一行中。

重置为默认值 将所有参数重置回其默认值。若在 Lambda 平台首选项中指定了 Lambda 值，则 Lambda 值将重置为平台首选项中指定的值。

注意：当 Y 变量具有包含 “目标”值的 “规格限”列属性时，“重置为默认值”选项会将“目标”设置为 “规格限”列属性中的 “目标”值。

空子组后重新开始 EWMA 指定存在空子组时如何处理移动平均值和限值的计算。若子组的所有观测都为缺失值或位于排除的行中，则子组可以为空。若选定该选项，则计算将在空子组后面的第一个非缺失子组中重新开始。重新开始计算将移动平均值重置为总均值。若未选定该选项， EWMA 计算将继续使用最新的非缺失子组移动平均值。

使用总体均值作为目标 （仅当 *Y* 变量具有包含 “目标”值的 “规格限”列属性时才可用。）将 EWMA 图中的“目标”设置为 *Y* 变量的总均值。若未选定该选项， EWMA 图中的 “目标”将设置为 “规格限”列属性中的 “目标”值。

叠加图 指定是否在 EWMA 图上叠加各个位置值。若选定该选项，则不再显示位置图。相反，位置图中的点在 EWMA 图上显示为未连接的灰色 X。位置图中的限值不会显示在 EWMA 图上，除非选择了 “在叠加图上显示 X 限值”选项。

在叠加图上显示 X 限值 （仅当选定 “叠加图”选项时才可用。）当选定 “叠加图”选项时，在 EWMA 图上显示或隐藏位置图中的限值。当位置图限值显示在 EWMA 图上时，这些限值显示为虚线。

注意：仅当选定 “在叠加图上显示 X 限值”选项时，“检验超出界限”选项才应用于 “叠加图”中的位置图值。

Lambda 滑块 显示或隐藏滑块控件，该控件支持您以交互方式更改 Lambda 参数的值。

显示 X 图 显示或隐藏 EWMA 图下方的位置图。

注意：选定 “叠加图”选项后，“显示 X 图”选项将显示或隐藏叠加在 EWMA 图上的各个位置值和限值。

显示残差图 显示或隐藏残差图。

警告脚本 支持您编写和运行指示数据未通过特殊原因检验的脚本。结果可以写入日志或大声说出，并且提供一个选项来包含对检验失败原因的解释。您还可以使用自定义脚本选项将结果发送到电子邮件。

在调用警告脚本时，以下变量在发出的脚本和随后的 JSL 脚本中都可用：

`qc_col` 是列名

`qc_test` 是失败的检验

`qc_sample` 是样本号

提示：指定警告脚本后，当启用 “检验超出界限”选项时，将调用警告脚本。

请参见 《Scripting Guide》，了解关于编写定制警告脚本的详细信息。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

EWMA 控制图的“平均运行链长 (ARL)”报表

“平均运行链长 (ARL)”报表包含 ARL 值的一个表和一个图。指定偏移的平均运行链长 (ARL) 是出现失控信号之前期望的平均试验次数。例如，0 处的 ARL 表示在过程受控时看到假警报信号之前期望的平均试验数。若过程受控，则偏移大小为 0。

ARL 报表中的表和图支持您探索各种 Lambda 参数设置如何影响相应的 EWMA 图的性能。表和图还支持您比较 EWMA 图与 Shewhart 图（如“EWMA 图”报表中的 X 图）的性能。当 Lambda 设置为 1 时，Shewhart ARL 列等价于 EWMA ARL 列。

0 位置处的 EWMA ARL 的值取决于 “常数限值”选项的设置：

- 若选定“常数限值”选项，则假定该过程受控制时间足够长，以致于开始值的影响可以忽略不计。这种情况也被称为 EWMA 图的**稳态**，此时使用 Crowder (1987) 中所述的方法计算 EWMA ARL(0) 的值。
- 若未选定“常数限值”选项，则使用 Knoth (2004) 中所述的方法计算 EWMA ARL(0) 的值。此情况也被称为 EWMA 图的**零状态**。

ARL 报表

ARL 报表显示 0 到 3 之间（以 0.25 为增量）的偏移 (Δ) 的平均运行链长。该偏移表示为 2*k*/Sigma，其中 *k* 是控制图使用的 sigma 乘数。该表包含 EWMA 图和 Shewhart 图的 ARL 值。

ARL 图

ARL 图显示介于 0 和 3 之间的偏移 (Δ) 的平均运行链长。该图包含与 ARL 图左侧的 ARL 表相同的数据点。实线对应 EWMA ARL 值，虚线对应 Shewhart ARL 值。

“EWMA 控制图”平台的统计详细信息

本节定义构造 EWMA 图时所用的统计量。其中一些统计量还保存在使用“保存汇总”命令创建的数据表中。

EWMA 图上的第 i 个点计算为 $EWMA_i = \lambda X_i - (1 - \lambda)EWMA_{i-1}$，其中 λ 表示 Lambda 参数，$X_i$ 表示第 i 个样本值（或子组均值），$EWMA_{i-1}$ 表示第 (i-1) 个 EWMA 值。当 $i = 1$ 时，将 $EWMA_0$ 定义为“目标”值。

注意：若选定“空子组后重新开始 EWMA”选项，则空子组后面的 $EWMA_{i-1}$ 值为“目标”值。若未选定“空子组后重新开始 EWMA”选项，则空子组后面的 $EWMA_{i-1}$ 值是最近的非空子组的 *EWMA* 值。

若未选定 “常数限值”选项， EWMA 控制限计算如下：

$$\text{下控制限} = T - K\sigma\sqrt{\frac{\lambda}{n_i(2-\lambda)}[1-(1-\lambda)^{2i}]}$$

$$\text{上控制限} = T + K\sigma\sqrt{\frac{\lambda}{n_i(2-\lambda)}[1-(1-\lambda)^{2i}]}$$

其中：

T = 目标值

K = sigma 乘数，且默认设置为 3

σ = Sigma 值

i = 样本 （或子组）数

n_i = 子组 i 的大小 （若未指定子组，则为 1）

若选定 “常数限值”选项， EWMA 控制限计算如下：

$$\text{下控制限} = T - K\sigma\sqrt{\frac{\lambda}{n(2-\lambda)}}$$

$$\text{上控制限} = T + K\sigma\sqrt{\frac{\lambda}{n(2-\lambda)}}$$

其中：

T = 目标值

K = sigma 乘数，且默认设置为 3

σ = Sigma 值

n = 子组大小 （若未指定子组，则为 1）

有关构造指数加权移动平均图的详细信息，请参见 Montgomery (2013)。

第 10 章

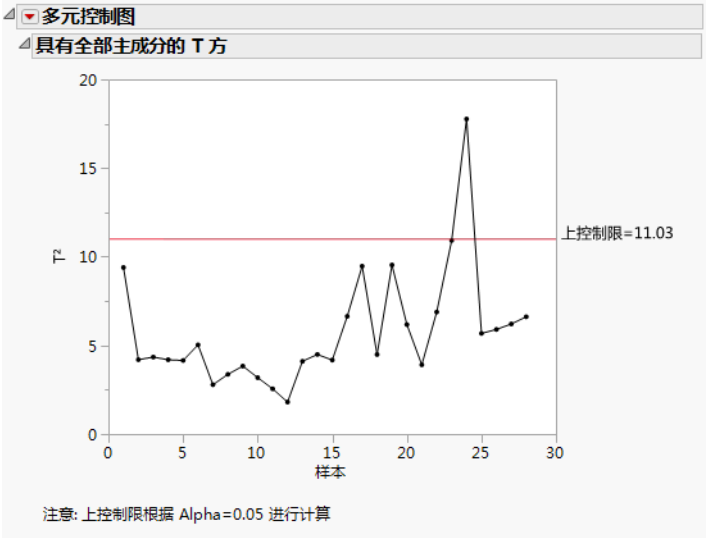
多元控制图

同时监控多个过程特征

多元控制图用于监控两个或更多相互关联的过程变量。一元控制图用于监控单个独立过程特征；而当过程变量相互关联时，则需要多元控制图。使用“多元控制图”平台可以创建 Hotelling T^2 图。您可以使用该平台确定某个过程是否稳定以及通过收集新数据来监控过程。

有关监控和诊断复杂过程的信息，请参见第 269 页的“模型驱动的多元控制图”一章。

图 10.1 多元控制图的示例



目录

多元控制图概述 247
多元控制图的示例 247
启动“多元控制图”平台 250
多元控制图 250
“多元控制图”平台选项 252
T 方已分割 252
变点检测 253
主成分 254
多元控制图的更多示例 254
监控使用子组数据的过程的示例 254
“T 方已分割”的示例 257
“变点检测”的示例 259
多元控制图的统计详细信息 260
单个观测的统计详细信息 260
合理子组中观测的统计详细信息 262
变点检测的统计详细信息 264

多元控制图概述

多元控制图用于监控两个或更多相互关联的过程变量。一元控制图用于监控单个独立过程特征；而当过程变量相互关联时，则需要多元控制图。Hotelling T^2 图（简称 T^2 图）是一种多元控制图。T^2 图可以检测均值偏移或检测几个相关变量之间的关系。观测可以是过程变量的单个观测，也可以将它们划分为合理的子组。

您可以使用当前或历史数据构造多元控制图。若使用当前数据构造控制图，则将控制图称为“阶段 I”图；若使用历史数据集的目标统计量构造控制图，则称之为“阶段 II”图。在“阶段 I”中，您查看过程是否稳定并确定一个历史数据集来计算过程的目标统计量。在“阶段 II”中，多元控制图使用“阶段 I”中的目标统计量来监控新过程观测。

要构造“阶段 II”多元控制图，请首先确定一个时间段，在该时间段内过程是稳定的、有能力的。

1. 绘制 “阶段 I”控制图来验证过程在该时间段内是稳定的。

   “阶段 I”中使用的数据提供了历史数据集
2. 保存该历史数据集的目标统计量。
3. 使用 “阶段 II”控制图基于在第 2 步中保存的目标统计量监控正在进行的过程。

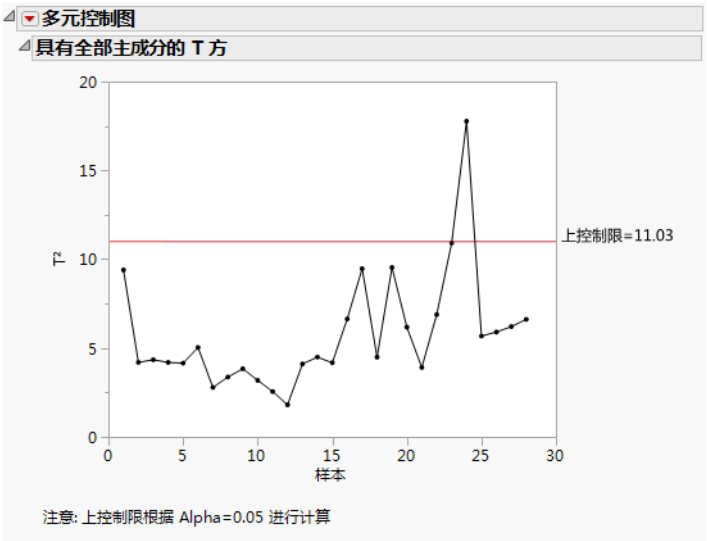
多元控制图的示例

本例说明如何为未分子组的数据构造控制图。该数据是有关蒸汽涡轮发动机的测量数据。有关使用子组数据的示例，请参见第 254 页的 “监控使用子组数据的过程的示例”。

第 1 步：确定过程是否稳定

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Steam Turbine Historical.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **多元控制图**。
3. 选择所有列并点击 **Y，列**。
4. 点击**确定**。

图 10.2 初始多元控制图



该过程似乎处于合理的统计控制中，因为只有一个失控的点。因此，根据该数据创建目标是合理的。

第 2 步：保存目标统计量

1. 点击“多元控制图”旁边的红色小三角并选择**保存目标统计量**。
   这将创建一个包含过程的目标统计量的新数据表。

图 10.3 蒸汽涡轮数据的目标统计量

	Ref_Stats	燃料	汽流	汽温	MW	冷却温度	压力
1	_SampleSize	28	28	28	28	28	28
2	_NumSample	1	1	1	1	1	1
3	_Mean	237595.78571	179015.78571	846.39285714	20.647142857	53.871428571	29.139285714
4	_Std	7247.6859825	4374.3063819	2.9481857034	0.5341650261	0.2088010623	0.0497347461
5	_Corr_燃料	1	0.8714382899	-0.549875041	0.8558570808	-0.270049819	-0.469928462
6	_Corr_汽流	0.8714382899	1	-0.629023927	0.9852529223	-0.223127002	-0.533056185
7	_Corr_汽温	-0.549875041	-0.629023927	1	-0.595214609	0.2475387217	0.2192147319
8	_Corr_MW	0.8558570808	0.9852529223	-0.595214609	1	-0.207305813	-0.50447312
9	_Corr_冷却温度	-0.270049819	-0.223127002	0.2475387217	-0.207305813	1	0.3617461646
10	_Corr_压力	-0.469928462	-0.533056185	0.2192147319	-0.50447312	0.3617461646	1

2. 将新数据表另存为 Steam Turbine Targets.jmp。

既然已经设定了目标统计量，接下来可以创建用于监控过程的多元控制图。

第 3 步：监控过程

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/Steam Turbine Current.jmp。
   该样本数据表包含过程中的最近观测。

2. 选择**分析 > 质量和过程 > 控制图 > 多元控制图**。
3. 选择所有列并点击 **Y，列**。
4. 点击**获取目标**。
5. 打开您保存的 Steam Turbine Targets.jmp 表。
6. 点击**确定**。

默认 alpha 水平设置为 0.05。将它更改为 0.001。

7. 点击 “多元控制图”旁边的红色小三角并选择**设置 Alpha 水平 > 其他**。
8. 键入 0.001 并点击**确定**。

图 10.4 蒸汽涡轮控制图

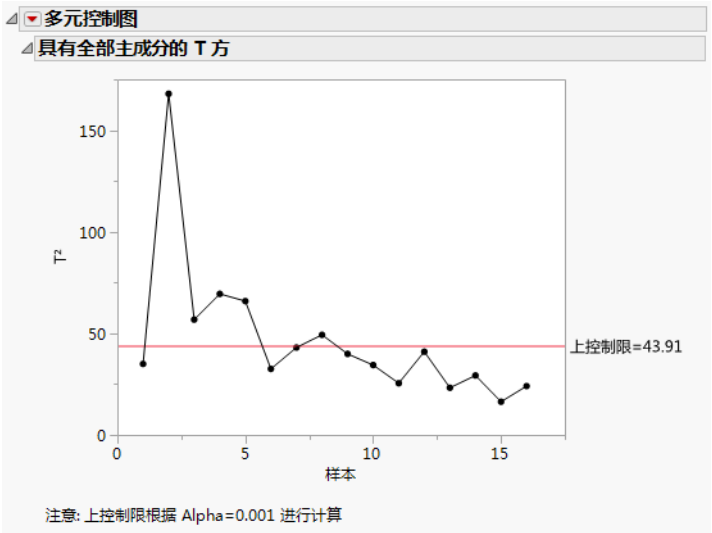
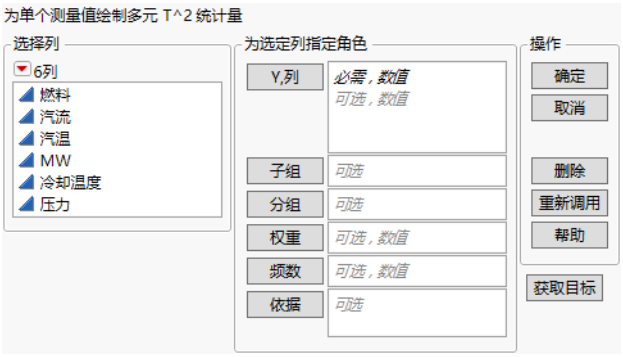


图 10.4 显示失控情况出现在观测 2、3、4、5 和 8 中。该结果暗示这些观测与 Steam Turbine Historical.jmp 中的历史数据不符，需要进一步调查过程。为了找到可指定的原因，您可能要查看各个一元控制图或执行另一个一元过程。

启动“多元控制图”平台

通过选择**分析** > **质量和过程** > **控制图** > **多元控制图**启动 “多元控制图”平台。

图 10.5 多元控制图启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“多元控制图”平台启动窗口包含以下选项：

Y，列 指定要分析的列。

子组 输入带有子组数据的列。从层次上看，该组嵌套在**分组**中。

分组 输入指定最高层次级别的组成员关系的列。

权重 标识数据表列，该列的变量为数据分配权重 （如重要性或影响）。

频数 标识数据表列，该列的值为每行分配频数。当您的数据表包含汇总数据时，这很有用。

依据 标识用于创建报表的一个列，该报表包括变量的每个水平的单独分析。

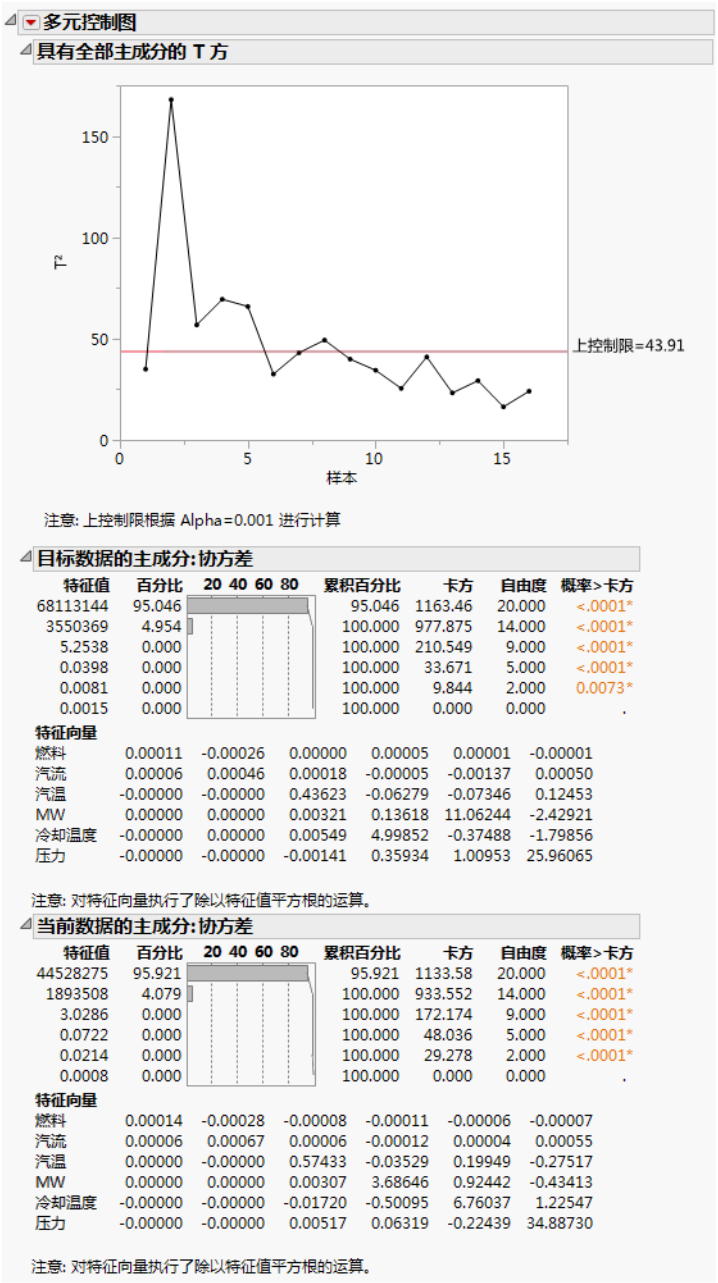
获取目标 点击以选择包含过程的历史目标的 JMP 表。

多元控制图

使用多元控制图可快速标识过程中的偏移，并针对特殊原因指示监控您的过程。

按第 247 页的“多元控制图的示例”中的说明操作以生成图 10.6 中所示的结果。

图 10.6 多元控制图



提示：有关其他选项的信息，请参见第 252 页的““多元控制图”平台选项”。

多元控制图绘制 Hotelling T^2 统计量。控制限的计算基于是否指定了目标而有所不同。要了解如何计算 T^2 统计量和 UCL（上控制限），请参见第 260 页的“多元控制图的统计详细信息”。有关控制限的详细信息，请参见 Tracy et al. (1992)。

在该示例中，两个数据集的“主成分”报表指示第一个特征值（对应于第一个主成分）解释了变量中大约 95% 的总变异。两个“特征向量”表中的值指示第一个主成分主要受变量“燃料”和“汽流”影响。您可以使用该信息仅基于第一个成分构造有可能更敏感的控制图。有关“主成分”报表的详细信息，请参见第 254 页的“主成分”。

“多元控制图”平台选项

“多元控制图”红色小三角菜单包含以下选项：

T 方图 显示 T^2 图。 Hotelling T^2 图是将相关性考虑在内的均值图的多元扩展。

T 方已分割 基于 Y 的主成分构造多元控制图。指定 T^2 的主成分个数。请参见第 252 页的“T 方已分割”。

设置 Alpha 水平 设置用于计算控制限的 α 水平。默认值为 α=0.05。

显示协方差 显示“合并协方差”报表。协方差是两个变量之间线性关系的一种测度。

显示相关性 显示“合并相关性”报表。

显示逆协方差 显示“合并逆协方差”报表。若逆协方差是奇异的，则报告广义逆协方差矩阵。

显示逆相关性 显示“合并逆相关性”报表。若逆相关性是奇异的，则报告广义逆相关性矩阵。

显示均值 显示“组均值”报表，其中包含每个组的均值。

保存 T 方 在包含 T^2 值的数据表中创建新列。

保存 T 方公式 在数据表中创建新列。在列中储存计算 T^2 值的公式。

保存目标统计量 创建包含过程的目标统计值的新数据表。目标统计量包括：样本大小、样本数、均值、标准差和所有相关性。

变点检测 （不可用于子组数据。）显示按行号绘制检验统计量的“变点检测”图，并指示出现变点的行号。请参见第 253 页的“变点检测”。

主成分 显示列出特征值及其相应特征向量的报表。主成分帮助您了解在您监控的变量中哪些对过程变异的贡献最大。请参见第 254 页的“主成分”。

保存主成分 在数据表中创建包含统一尺度的主成分的新列。

T 方已分割

若您正在监控大量相关的过程特征，可以使用“T 方已分割”选项来构造基于主成分的控制图。若很少数目的主成分解释了测量值中的大部分变异，则基于这些重要成分的多元控制图可能比基于原始的更高维数据的图更灵敏。

当您的协方差矩阵为病态时，“T 方已分割”选项也很有用。在这种情况下，具有小特征值的成分解释很少的变异，它们对 T^2 可能有严重的误导性影响。研究过程行为时，将这些较不重要的成分分离出来是很有用的。

选择“T 方已分割”选项后，您需要确定要使用多少个主成分。

该选项创建两个多元控制图：具有大的主成分的 T 方和具有小的主成分的 T 方。假定您首次选择该选项时输入 *r* 作为主成分个数。具有大的主成分的图基于 *r* 个主成分（对应于 *r* 个最大的特征值）。这些是解释最大变异量的 *r* 个成分，如“主成分: 基于协方差”报表的“百分比”和“累积百分比”列中所示。具有小的主成分的图基于其余主成分。

对于指定的子组，它在大的主成分图中的 T^2 值和在小的主成分图中的 T^2 值之和为“具有全部主成分的 T^2”报表中显示的总 T^2 统计量。有关如何计算已分割的 T^2 值的详细信息，请参见 Kourti and MacGregor (1996)。

提示：考虑使用“模型驱动的多元控制图”平台来分解 T^2 统计量。请参见第 269 页的“模型驱动的多元控制图”一章。

变点检测

当数据集由多元单个观测组成时，可以建立控制图以检测均值向量、协方差矩阵或这两者中的偏移。该方法分割数据并针对偏移计算似然比检验统计量。在控制图上标绘的该统计量是观测的似然比检验统计量除以以下项的积所得：

- 假定无偏移的近似期望值。
- 上控制限的近似值。

除以近似上控制限允许针对有效的上控制限 1 标绘点。“变点检测”图清晰显示发生控制图统计量最大值的偏移时的变点。JMP 中的“变点检测”方法基于 Sullivan and Woodall (2000)，相关说明请参见第 264 页的“变点检测的统计详细信息”。

注意：“变点检测”方法设计用于显示数据中的单个偏移。递归应用该方法可以检测多个偏移。

对于“变点检测”图，请注意以下事项：

- 1.0 以上的值指示数据中可能有偏移。
- 通过将关注值（均值向量或协方差矩阵）的似然比统计量除以归一化因子，可以得到“变点检测”图的控制图统计量。
- 对于“变点检测”图，数据的变点发生在具有最大检验统计值的观测。

对于散点图矩阵，请注意以下事项：

- 该图显示样本均值向量中的偏移。

- 在第 259 页的“变点检测”的示例”中，数据被分为两组。前 24 个观测归为第一组。其余观测归为第二组。

主成分

“主成分”报表包含以下信息：

特征值 协方差矩阵的特征值。

百分比 相应特征向量所解释的变异百分比。随之还显示条形图。

累积百分比 对应于特征值的特征向量所解释的累积变异百分比。

卡方 提供用于确定数据中剩余的相关性是否为随机的检验。这是 Bartlett 球形检验。当该检验拒绝原假设时，意味着数据中的剩余结构与该特征值有关。

自由度 与卡方检验关联的自由度。

概率 > 卡方 检验的 *p* 值。

特征向量 对应于特征值的特征向量表。请注意，每个特征向量被除以其相应特征值的平方根。

有关主成分的详细信息，请参见《多元方法》。

多元控制图的更多示例

- “监控使用子组数据的过程的示例”
- ““T 方已分割”的示例”
- ““变点检测”的示例”

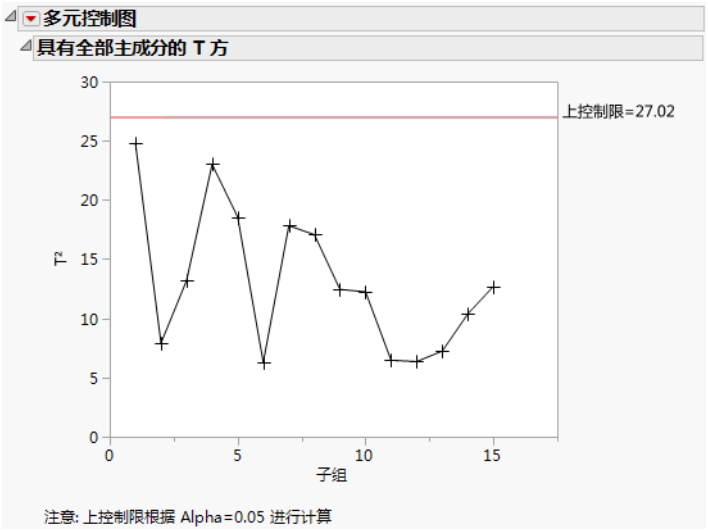
监控使用子组数据的过程的示例

监控使用子组数据的多元过程的工作流程类似于监控单个数据的工作流程。请参见第 247 页的“多元控制图的示例”。您创建一个初始控制图来保存目标统计量，然后使用这些目标来监控过程。

第 1 步：确定过程是否稳定

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Aluminum Pins Historical.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **多元控制图**。
3. 选择所有**直径**和**长度**列，然后点击 **Y，列**。
4. 选择**子组**并点击**子组**。
5. 点击**确定**。

图 10.7 子组数据的多元控制图，步骤 1



该过程看起来处于统计控制中，使用该数据创建目标是合适的。

第 2 步：保存目标统计量

1. 点击“多元控制图”旁边的红色小三角并选择**保存目标统计量**。
   这将创建一个包含过程的目标统计量的新数据表。
2. 将新数据表另存为 Aluminum Pins Targets.jmp。
   既然已经设定了目标统计量，接下来可以创建用于过程监控的多元控制图。

第 3 步：监控过程

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Aluminum Pins Current.jmp。
   该样本数据表包含过程中的最近观测。
2. 选择**分析** > **质量和过程** > **控制图** > **多元控制图**。
3. 选择所有**直径**和**长度**列，然后点击 **Y，列**。
4. 选择**子组**并点击**子组**。
5. 点击**获取目标**。
6. 打开您保存的 Aluminum Pins Targets.jmp 表。
7. 点击**确定**。
8. 点击 “多元控制图”旁边的红色小三角并选择**显示均值**。
   “显示均值”选项给出每个子组的均值。您然后可以观察哪些组最不相似。

图 10.8 子组数据的多元控制图，步骤 3

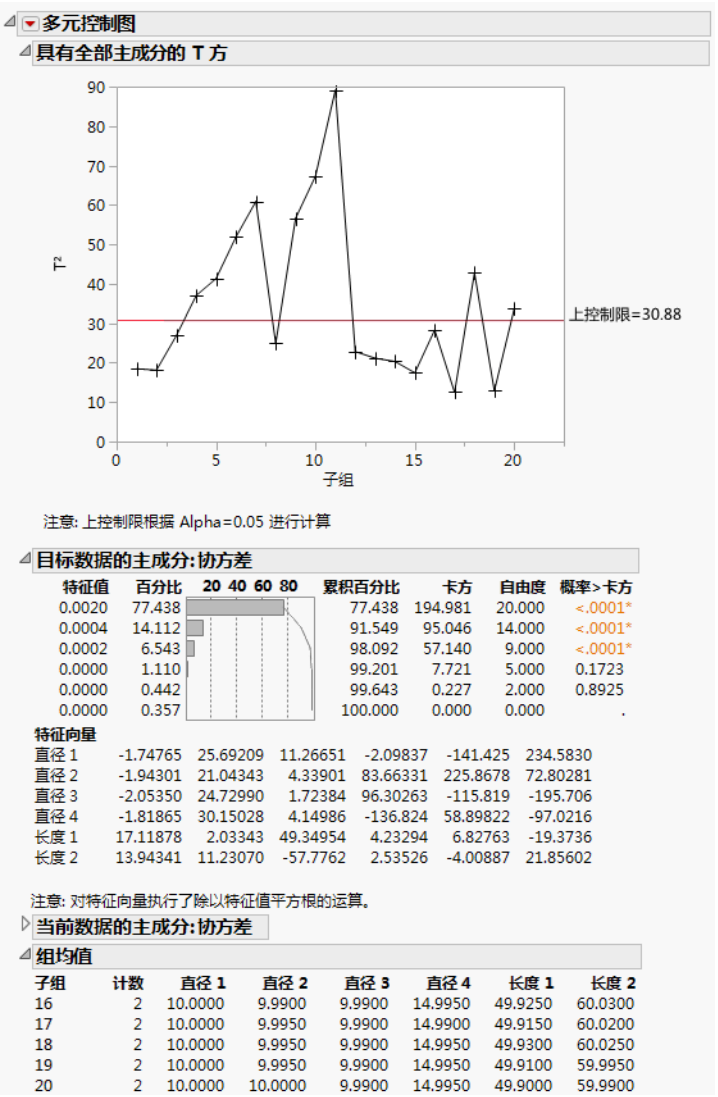


图 10.8 显示不稳定出现在子组 4-7、9-11、18 和 20 中。该结果暗示这些观测与 Aluminum Pins Historical.jmp 中的历史数据不符，需要进一步调查过程。要确定过程在这些点失控的原因，您可能要查看各个一元控制图或执行另一个一元过程。

监控该过程的替代方法是基于大的主成分。在本示例中，对于历史数据，前三个主成分解释了大约 98% 的变异。基于此，您可以构造前三个主成分的图。然后，使用这三个成分监控当前数据。用于监控过程的图的控制限应基于相应历史数据的图。

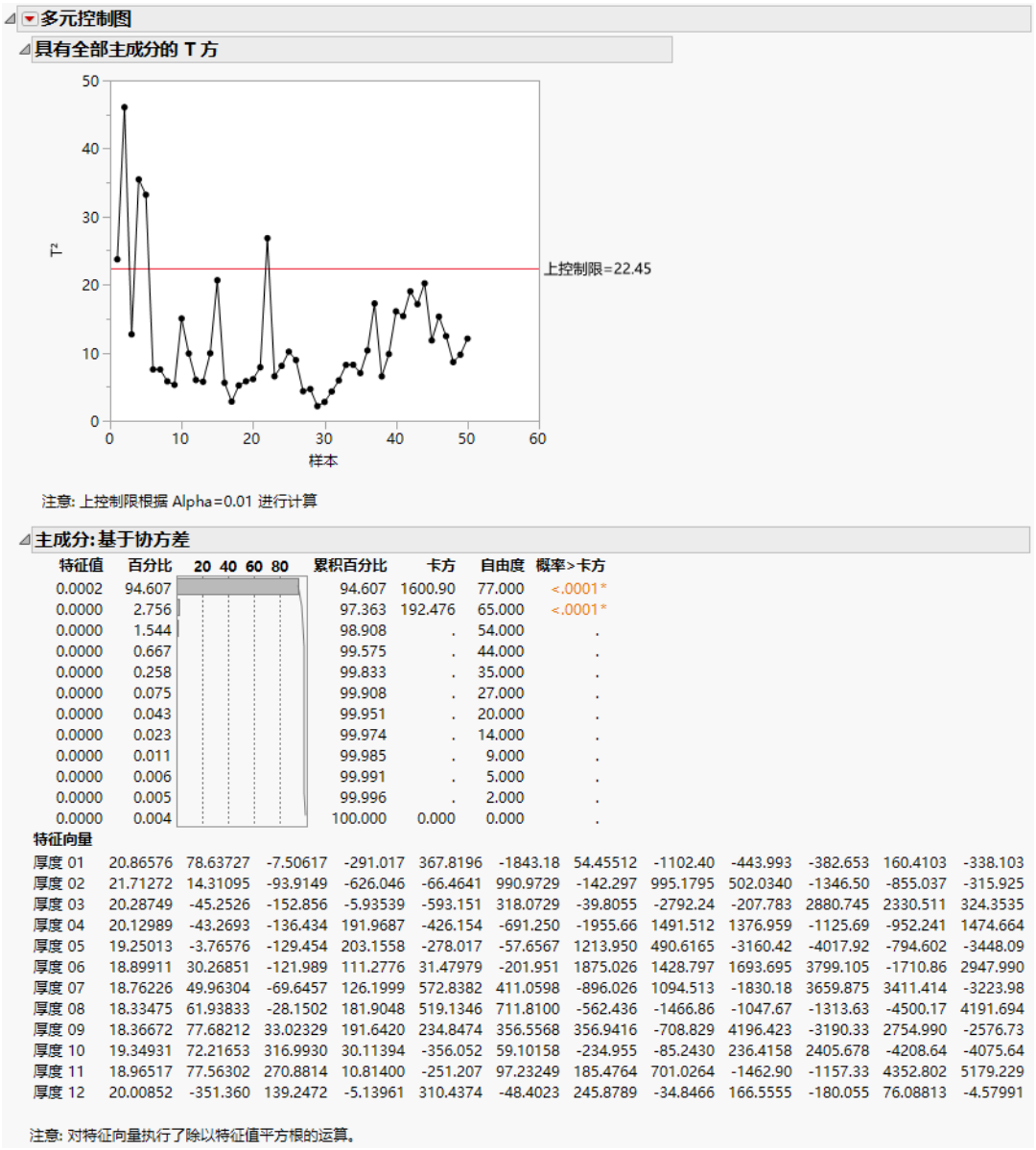
“T 方已分割”的示例

在研究过程行为时，使用“T 方已分割”将更重要的成分与不太重要的成分分离。在本示例中，每 50 个钢筋条的涂层在条上 12 个间距相等的位置进行测量。您想检查测量值中的变异并确定是否需要进一步调查导致变异的原因。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Thickness.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **多元控制图**。
3. 选择所有**厚度**列并点击 **Y，列**。
4. 点击**确定**。

   当前 alpha 水平设置为 0.05，它对应于 5% 的误报率。您想将误报率设置为 1%。
5. 要更改 alpha 水平，点击“多元控制图”旁边的红色小三角，选择**设置 Alpha 水平**，然后选择 **0.01**。

图 10.9 Thickness.jmp 的初始多元控制图



特征值	百分比	累积百分比	卡方	自由度	概率>卡方
0.0002	94.607	94.607	1600.90	77.000	<.0001*
0.0000	2.756	97.363	192.476	65.000	<.0001*
0.0000	1.544	98.908	.	54.000	.
0.0000	0.667	99.575	.	44.000	.
0.0000	0.258	99.833	.	35.000	.
0.0000	0.075	99.908	.	27.000	.
0.0000	0.043	99.951	.	20.000	.
0.0000	0.023	99.974	.	14.000	.
0.0000	0.011	99.985	.	9.000	.
0.0000	0.006	99.991	.	5.000	.
0.0000	0.005	99.996	.	2.000	.
0.0000	0.004	100.000	0.000	0.000	.

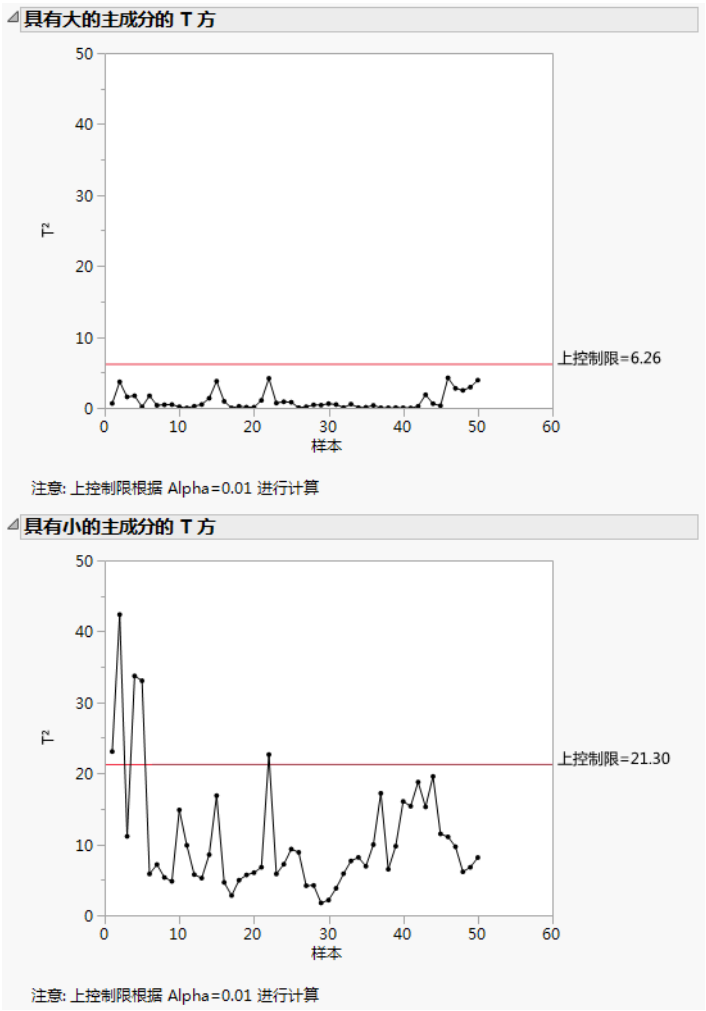
特征向量

厚度 01	20.86576	78.63727	-7.50617	-291.017	367.8196	-1843.18	54.45512	-1102.40	-443.993	-382.653	160.4103	-338.103
厚度 02	21.71272	14.31095	-93.9149	-626.046	-66.4641	990.9729	-142.297	995.1795	502.0340	-1346.50	-855.037	-315.925
厚度 03	20.28749	-45.2526	-152.856	-5.93539	-593.151	318.0729	-39.8055	-2792.24	-207.783	2880.745	2330.511	324.3535
厚度 04	20.12989	-43.2693	-136.434	191.9687	-426.154	-691.250	-1955.66	1491.512	1376.959	-1125.69	-952.241	1474.664
厚度 05	19.25013	-3.76576	-129.454	203.1558	-278.017	-57.6567	1213.950	490.6165	-3160.42	-4017.92	-794.602	-3448.09
厚度 06	18.89911	30.26851	-121.989	111.2776	31.47979	-201.951	1875.026	1428.797	1693.695	3799.105	-1710.86	2947.990
厚度 07	18.76226	49.96304	-69.6457	126.1999	572.8382	411.0598	-896.026	1094.513	-1830.18	3659.875	3411.414	-3223.98
厚度 08	18.33475	61.93833	-28.1502	181.9048	519.1346	711.8100	-562.436	-1466.86	-1047.67	-1313.63	-4500.17	4191.694
厚度 09	18.36672	77.68212	33.02329	191.6420	234.8474	356.5568	356.9416	-708.829	4196.423	-3190.33	2754.990	-2576.73
厚度 10	19.34931	72.21653	316.9930	30.11394	-356.052	59.10158	-234.955	-85.2430	236.4158	2405.678	-4208.64	-4075.64
厚度 11	18.96517	77.56302	270.8814	10.81400	-251.207	97.23249	185.4764	701.0264	-1462.90	-1157.33	4352.802	5179.229
厚度 12	20.00852	-351.360	139.2472	-5.13961	310.4374	-48.4023	245.8789	-34.8466	166.5555	-180.055	76.08813	-4.57991

图 10.9 中的总控制图指示特殊原因影响了钢筋条 1、2、4、5 和 22。通过查看“主成分”报表，您可以看到第一个主成分解释了 12 个厚度测量值中几乎 95% 的变异。您想进一步调查与该主成分相关的变异。

6. 点击“多元控制图”旁边的红色小三角并选择**已分割 T 方**。
7. 通过点击**确定**接受 1 个主成分的默认值。

图 10.10 “T 方已分割”控制图



对比“主成分”报表，“具有大的主成分的 T 方”图（仅反映第一个成分的变异）显示不出来存在特殊原因的证据。“具有小的主成分的 T 方”图显示特殊原因存在于其余更小的成分中。这些更小的成分没有解释很多变异，很可能表示随机噪声。因此，您可以认定厚度测量值中的变异不是值得关注的主要原因。

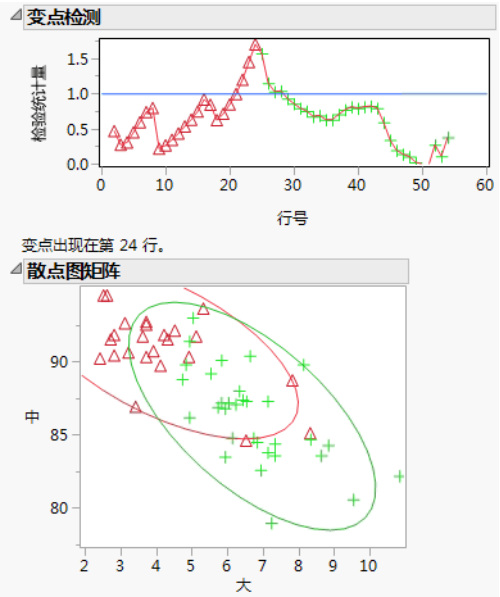
“变点检测”的示例

使用变点检测查找数据中发生偏移的点。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Gravel.jmp。
2. 选择**分析** > **质量和过程** > **控制图** > **多元控制图**。

3. 选择**大**和**中**，然后点击 **Y，列**。
4. 点击**确定**。
5. 点击“多元控制图”旁边的红色小三角并选择**变点检测**。

图 10.11 Gravel.jmp 的变点检测



提示：根据您的数据，您可能需要拖动轴来查看两个组的密度椭圆。

在“变点检测”图中，1.0 以上的值指示数据中可能有偏移。至少一个偏移是明显的，变点出现在第 24 个观测处，偏移紧接在该处之后发生。两个组的 95% 预测区域有几乎相同的大小、形状和方向，这直观地指出样本协方差矩阵是相似的。

多元控制图的统计详细信息

- “单个观测的统计详细信息”
- “合理子组中观测的统计详细信息”
- “变点检测的统计详细信息”

单个观测的统计详细信息

考虑未分子组的测量值，即自然子组大小为 $n = 1$ 的情况。用 m 表示观测数，用 p 表示测量的变量数。为每个观测计算 T^2 统计量并标绘它。T^2 统计量和上控制限 (UCL) 的计算取决于目标统计量的来源。在“阶段 I”图中，限值基于在控制图上标绘的同一数据。在“阶段 II”图

中，限值基于从历史数据集计算得出的目标统计量。有关 T^2 统计量和 Hotelling T^2 控制图的控制限计算的详细信息，请参见 Montgomery (2013)。

“阶段 I”控制图的计算

在 “阶段 I”控制图中，按以下方式定义第 i 个观测的 T^2 统计量：

$$T_i^2 = (Y_i - \overline{Y})' S^{-1} (Y_i - \overline{Y})$$

其中：

Y_i 是第 i 个观测的 p 个测量值的列向量

$\overline{Y}$ 是 p 个变量的样本均值的列向量

S^{-1} 是样本协方差矩阵的逆矩阵

i 个观测中每个观测的 T_i^2 值都是在多元控制图上标绘的点。

计算 “阶段 I”控制限时，上控制限基于 beta 分布。具体而言，按以下方式定义上控制限 (UCL)：

$$UCL = \frac{(m-1)^2}{m} \beta_{\left[1-\alpha; \frac{p}{2}; \frac{m-p-1}{2}\right]}$$

其中：

p = 变量数

m = 观测数

$\beta_{\left[1-\alpha; \frac{p}{2}; \frac{m-p-1}{2}\right]}$ = Beta $\left(\frac{p}{2}, \frac{m-p-1}{2}\right)$ 分布的第 (1–α) 分位数

“阶段 II”控制图的计算

在 “阶段 II”控制图中，将历史数据集定义为 X。然后按以下方式定义第 i 个观测的 T^2 统计量：

$$T_i^2 = (Y_i - \overline{X})' S_X^{-1} (Y_i - \overline{X})$$

其中：

Y_i 是第 i 个观测的 p 个测量值的列向量

$\overline{X}$ 是 p 个变量的样本均值的列向量，从历史数据集计算

S_X^{-1} 是样本协方差矩阵的逆矩阵，从历史数据集计算

i 个观测中每个观测的 T_i^2 值都是在多元控制图上标绘的点。

计算 “阶段 II” 控制限时，新观测独立于历史数据集。在这种情况下，上控制限 (UCL) 是 F 分布的函数，部分依赖于从中计算目标的历史数据集中的观测数。按以下方式定义上控制限：

$$UCL = \begin{cases} \frac{p(m+1)(m-1)}{m(m-p)} F_{[1-\alpha, p, m-p]} & \text{若 } m \le 100 \\ \frac{p(m-1)}{m-p} F_{[1-\alpha, p, m-p]} & \text{若 } m > 100 \end{cases}$$

其中：

p = 变量数

m = 历史数据集中的观测数

$F_{[1-\alpha, p, m-p]}$ $F(p, m-p)$ 分布的 = (1–α) 分位数

合理子组中观测的统计详细信息

考虑监控 p 个变量且获取大小为 $n > 1$ 的 m 个子组的情况。为每个子组计算 T^2 统计量并标绘它。T^2 统计量和上控制限 (UCL) 的计算取决于目标统计量的来源。在 “阶段 I” 图中，限值基于在控制图上标绘的同一数据。在 “阶段 II” 图中，限值基于从历史数据集计算得出的目标统计量。有关 T^2 统计量和 Hotelling T^2 控制图的控制限计算的详细信息，请参见 Montgomery (2013)。

“阶段 I” 控制图的计算

对于 “阶段 I” 控制图，按以下方式定义第 j 个子组的 T^2 统计量：

$$T_j^2 = (\bar{Y}_j - \bar{\bar{Y}})' S_p^{-1} (\bar{Y}_j - \bar{\bar{Y}})$$

其中：

$\bar{Y}_j$ 是第 j 个子组的 p 个测量值的 n 个列向量的均值

$\bar{\bar{Y}} = \frac{1}{m} \sum_{j=1}^{m} \bar{Y}_j$ 是子组均值的均值

S_j 是第 j 个子组中 n 个观测的样本协方差矩阵

$S_p = \frac{1}{m} \sum_{j=1}^{m} S_j$ 是合并的协方差矩阵，计算为子组内协方差矩阵的均值

按以下方式定义 “阶段 I” 的上控制限 (UCL)：

$$UCL = \frac{p(m-1)(n-1)}{mn-m-p+1} F_{[1-\alpha,\, p,\, mn-m-p+1]}$$

其中：

p = 变量数

n = 每个子组的样本大小

m = 子组数

$F_{[1-\alpha,\, p,\, mn-m-p+1]} = F(p, mn-m-p+1)$ 分布的第 (1–α) 分位数

“阶段 II” 控制图的计算

在 “阶段 II” 控制图中，将从中计算目标统计值的历史数据集定义为 X。然后按以下方式定义第 j 个子组的 T^2 统计量：

$$T_j^2 = (\overline{Y}_j - \overline{X})' S_p^{-1} (\overline{Y}_j - \overline{X})$$

其中：

$\overline{Y}_j$ 是第 j 个子组的 p 个测量值的 n 个列向量的均值

$\overline{X}_k$ 是第 k 个子组 （来自历史数据集）中的 p 个测量值的 n 个列向量的均值

$\overline{X} = \frac{1}{m} \sum_{k=1}^{m} \overline{X}_k$ 是观测的总均值

S_k 是第 k 个子组 （来自历史数据集）中的 n 个观测的样本协方差矩阵

$S_p = \frac{1}{m} \sum_{k=1}^{m} S_k$ 是合并的协方差矩阵，计算为子组内协方差矩阵的均值

按以下方式定义 “阶段 II” 上控制限 (UCL)：

$$UCL = \frac{p(m+1)(n-1)}{mn-m-p+1} F_{[1-\alpha,\, p,\, mn-m-p+1]}$$

其中：

p = 变量数

n = 子组样本大小

m = 历史数据集中的子组数

$F_{[1-\alpha,\, p,\, mn-m-p+1]} = F(p, mn-m-p+1)$ 分布的第 (1–α) 分位数

合理子组中观测的检验统计量的加和性

将 mn 个独立正态分布的观测样本分为 m 个大小为 n 的合理子组时，将 T^2_M 定义为第 j 个子组的均值 $\overline{Y}_j$ 与目标值之间的间距。（对于合理子组中的观测， T^2_M 与上一节中的 T^2 等价。）您还可以计算与每个子组中的内部变异性有关的 T^2 统计量和围绕目标值的总变异性。 T^2 统计量的分量是可相加的，很像平方和。具体来说，以下关系对于 m 个子组均成立：

$$T^2_{A_j} = T^2_{M_j} + T^2_{D_j}$$

在所有以下定义中，S_p 的定义方式与上一节中相同，根据控制图是“阶段 I”还是“阶段 II”控制图而定。同样，对于“阶段 I”控制图将 μ 定义为 $\overline{Y}$，对于“阶段 II”控制图将其定义为 $\overline{X}$。

按以下方式定义与第 j 个子组的目标值的间距：

$$T^2_{M_j} = n(\overline{Y}_j - \mu)' S_P^{-1} (\overline{Y}_j - \mu)$$

按以下方式定义第 j 个子组的内部变量性：

$$T^2_{D_j} = \sum_{i=1}^{n} (Y_{ji} - \overline{Y}_j)' S_P^{-1} (Y_{ji} - \overline{Y}_j)$$

其中 Y_{ji} 是第 j 个子组的 p 个测量值的第 i 个列向量。

按以下方式定义第 j 个子组的总变异性：

$$T^2_{A_j} = \sum_{i=1}^{n} (Y_{ji} - \mu)' S_P^{-1} (Y_{ji} - \mu)$$

其中 Y_{ji} 是第 j 个子组的 p 个测量值的第 i 个列向量。

注意：当您从“多元控制图”红色小三角菜单选择**保存 T 方**或**保存 T 方公式**选项时，在每行中保存的三个值对应于上述三个定义中的一个 i 值。

变点检测的统计详细信息

本讨论采纳 Sullivan and Woodall (2000) 方法。

假定

用 $N_p(\mu_i, \Sigma_i)$ 表示均值向量为 μ_i、协方差矩阵为 Σ_i 的 p 维多元分布。假定 x_i 是来自这类分布的 m（其中 $m > p$）个独立观测。

$$x_i \sim N_p(\mu_i, \Sigma_i), \quad i = 1, \ldots, m$$

若过程是稳定的，则表示 μ_i 和协方差矩阵 Σ_i 等于一个公值，使 x_i 服从 $N_p(\mu, \Sigma)$ 分布。

假定 m_1 和 m_1+1 个观测之间的均值向量和 / 或协方差矩阵中出现单个变化，则以下条件成立：

- 观测 *1* 到 m_1 具有相同的均值向量和相同的协方差矩阵 (μ_a,Σ_a)。
- 观测 $m_1 + 1$ 到 m 具有相同的均值向量和相同的协方差矩阵 (μ_b,Σ_b)。
- 出现以下之一：
  - 若变化影响均值，$\mu_a \neq \mu_b$。
  - 若变化影响协方差矩阵，$\Sigma_a \neq \Sigma_b$。
  - 若变化同时影响均值和协方差矩阵，$\mu_a \neq \mu_b$ 且 $\Sigma_a \neq \Sigma_b$。

概述

似然比检验方法用于确定均值向量和协方差矩阵中的一个还是两者均发生变化。使用似然比检验统计量来计算近似上控制限为 1 的控制图统计量。为所有可能的 m_1 值标绘该控制图统计量。若任何观测的控制图统计量超过上控制限 1，这表示发生了偏移。假定正好有一个偏移，该偏移被视为紧跟在具有最大控制图统计量值的观测之后。

似然比检验统计量

按以下方式定义前 m_1 个观测的对数似然函数两倍的最大值：

$$l_1 = -m_1 k_1 \log[2\pi] - m_1 \log\left[\left|S_1\right|_{k_1}\right] - m_1 k_1$$

l_1 的等式使用以下符号：

- S_1 是前 m_1 个观测的协方差矩阵的最大似然估计值。
- $k_1 = \text{Min}[p, m_1\text{-}1]$ 是 p x p 矩阵 S_1 的秩。
- 符号 $\left|S_1\right|_{k_1}$ 表示矩阵 S_1 的广义行列式，将该值定义为 k_1 个正特征值 λ_j 的积：

$$\left|S_1\right|_{k_1} = \prod_{j=1}^{k_1} \lambda_j$$

  当 S_1 具有满秩时，广义行列式等于普通行列式。

用 l_2 表示后面 $m_2 = m - m_1$ 个观测的对数似然函数两倍的最大值，用 l_0 表示所有 m 个观测的对数似然函数两倍的最大值。指定 l_2 和 l_0 所用的表达式类似于 l_1 的表达式。

似然比检验统计量将 $l_1 + l_2$ 的和与 l_0 进行比较。$l_1 + l_2$ 的和是该对数似然的两倍，后者假定 m_1 处可能有偏移。值 l_0 是假定无偏移的对数似然的两倍。若 l_0 远小于 $l_1 + l_2$，则假定过程是不稳定的。

按以下方式定义似然比检验统计量，它检验某个变化是否在观测 $m_1 + 1$ 处开始：

$$\begin{aligned} \text{lrt}[m_1] &= (l_1 + l_2 - l_0) \\ &= (m_1(p - k_1) + m_2(p - k_2))(1 + \log(2\pi)) \\ &\quad + m\log[|S|] - m_1\log\left[|S_1|_{k_1}\right] - m_2\log\left[|S_2|_{k_2}\right] \end{aligned}$$

似然比检验统计量服从自由度为 $p(p + 3)/2$ 的渐近卡方分布。对数似然比值大指示过程是不稳定的。

控制图统计量

模拟结果表明 lrt[m_1] 的期望值随观测在序列中的位置而变化，特别取决于 p 和 m。请参见 Sullivan and Woodall (2000)。

lrt[m_1] 期望值的近似公式从模拟推导而来。为了降低期望值与 p 的相关性，用 lrt[m_1] 除以其渐近期望值 $p(p + 3)/2$。

表示 lrt[m_1] 的近似期望值除以 $p(p+3)/2$ 的公式定义如下：

$$\text{ev}[m,p,m_1] = \begin{cases} a_p + m_1 b_p, & \text{若 } m_1 < p + 1 \\ a_p + (m - m_1)b_p, & \text{若 } m - m_1 < p + 1 \\ 1 + \dfrac{m - 2p - 1}{(m_1 - p)(m - p - m_1)}, & \text{否则} \end{cases}$$

其中

$$a_p = -\frac{0.08684(p - 14.69)(p - 2.036)}{(p - 2)}$$

且

$$b_p = \frac{0.1228(p - 1.839)}{(p - 2)}$$

对于 $p = 2$，当 m_1 或 $m_2 = 2$ 时，ev[m,p,m_1] 的值为 1.3505。

注意：上述公式对于 $p > 12$ 或 $m < (2p + 4)$ 是不准确的。在这些情况中，应使用模拟来获取近似期望值。

按以下方式计算一个近似上控制限，在假定过程是稳定时，该控制限对应产生假的失控信号的概率大约为 0.05：

$$\text{上控制限}[m,p] \cong (3.338 - 2.115\log[p] + 0.8819(\log[p])^2 - 0.1382(\log[p])^3)$$

$$+ \quad (0.6389 - 0.3518\log[p] + 0.01784(\log[p])^3)\log[m]。$$

请注意，该公式依赖于 m 和 p。

将控制图统计量定义为似然比检验统计量对数的两倍除以 $p(p+3)$、除以其近似期望值以及控制限的近似值。因为除以上控制限的近似值，因此可以针对上控制限 1 标绘控制图统计量。按以下计算近似控制图统计量：

$$\hat{y}[m_1] = \frac{2\text{lrt}[m_1]}{p(p+3)(\text{ev}[m,p,m_1]\text{上控制限}[m,p])}$$

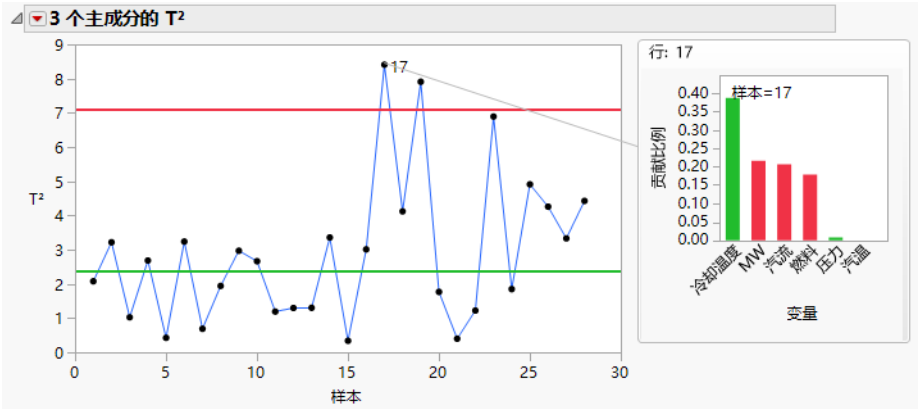
第 11 章

模型驱动的多元控制图

复杂过程的监控和诊断

模型驱动的多元控制图用于监控单个控制图中多个过程的参数。“模型驱动的多元控制图”(MDMVCC) 平台支持您基于主成分或偏最小二乘模型生成控制图。对于一组连续变量，MDMVCC 平台使用主成分生成控制图。对于保存的主成分或偏最小二乘得分函数，MDMVCC 平台基于这些提供的模型生成控制图。使用 MDMVCC 平台以交互方式探索和理解导致失控信号的底层成分。

图 11.1 模型驱动的多重控制图



目录

模型驱动的多元控制图概述 271
模型驱动的多元控制图的示例 271
启动“模型驱动的多元控制图”平台 273
数据格式 274
“模型驱动的多元控制图”报表 275
“模型驱动的多元控制图”平台选项 275
图选项 276
得分图 277
“模型驱动的多元控制图”平台的更多示例 277
包含历史数据的 MDMVCC 的示例 278
包含 PLS 模型的 MDMVCC 的示例 279
“模型驱动的多元控制图”平台的统计详细信息 280
监控统计量 280
限值 282
贡献 284
得分图分组比较 286
得分图载荷 286

模型驱动的多元控制图概述

模型驱动的多元控制图 (MDMVCC) 平台具有两项主要功能：监控和诊断。

- 使用多元控制图来监控多元过程。
- 您可以交互方式下钻，以调查各个变量对诊断过程的总体信号的贡献。

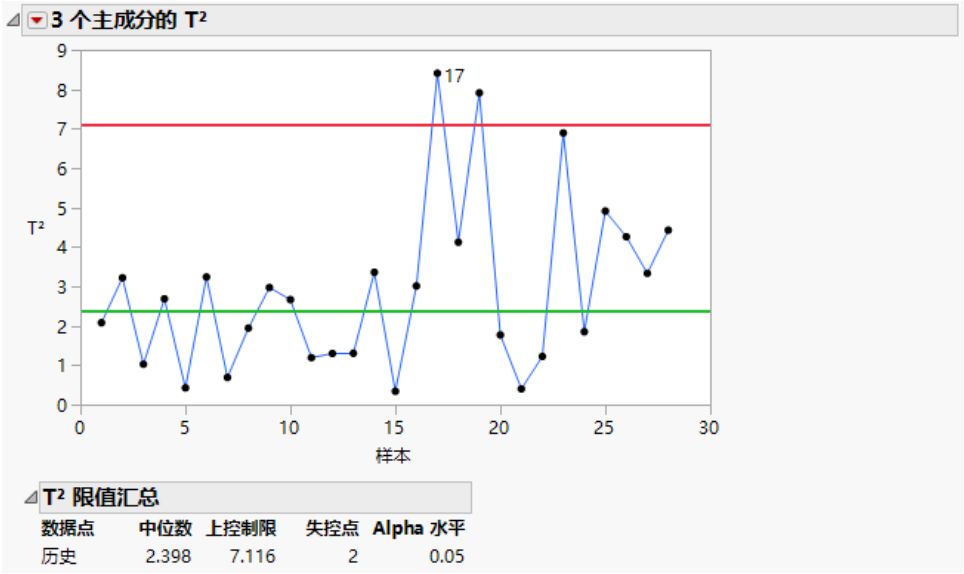
有关多元控制图的详细信息，请参见 Kourti and MacGregor (1996)。

模型驱动的多元控制图的示例

本例使用包含来自蒸汽涡轮系统的过程变量的 Steam Turbine Historical.jmp 样本数据表。您需要为六个受监控的变量生成一个控制图。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Steam Turbine Historical.jmp。
2. 选择**分析** > **质量和过程** > **模型驱动的多元控制图**。
3. 选择全部六列，点击**过程**并点击**确定**。

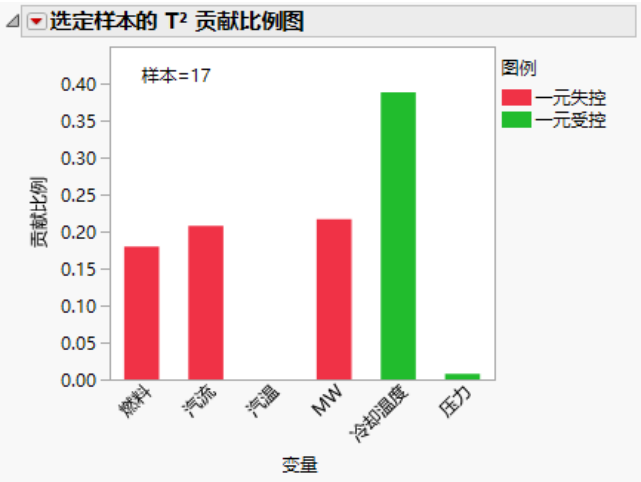
图 11.2 “蒸汽涡轮”报表



请注意该过程在样本 16 之后偏移。

4. 选择样本 17 数据点。右击并选择**行** > **行标签**。
5. 悬停在样本 17 数据点上以查看该点的 T^2 贡献比例图。点击该图以在报表窗口中打开该图。

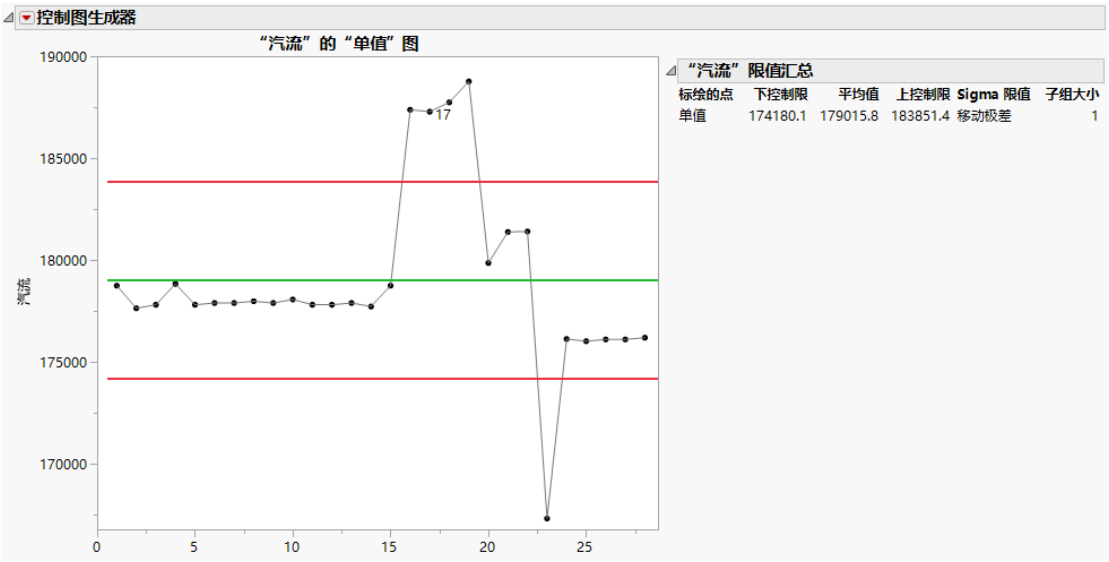
图 11.3 样本 17 的贡献比例图



请注意，**冷却温度**贡献了 40% 的 T^2 值。**冷却温度**直条为绿色，指示样本 17 位于**冷却温度**的一元控制限内。**汽流**和 MW 各贡献了 20% 的 T^2 值。它们都是红色的，这表示样本 17 超出了每个变量的一元控制限。**蒸汽温度**对 T^2 值的贡献为零。在本例中，您找到了变量，其中多元失控样本可以追溯到失控的一元变量。然而，情况并非总是如此。在多元过程控制中，您可能观察到 T^2 图上有某个失控点，但发现该样本在所有变量的一元水平上都处于控制之中。

6. 悬停在贡献比例图中的**汽流**直条上以查看**汽流**的一元控制图。点击该图以便在新的报表窗口中打开。

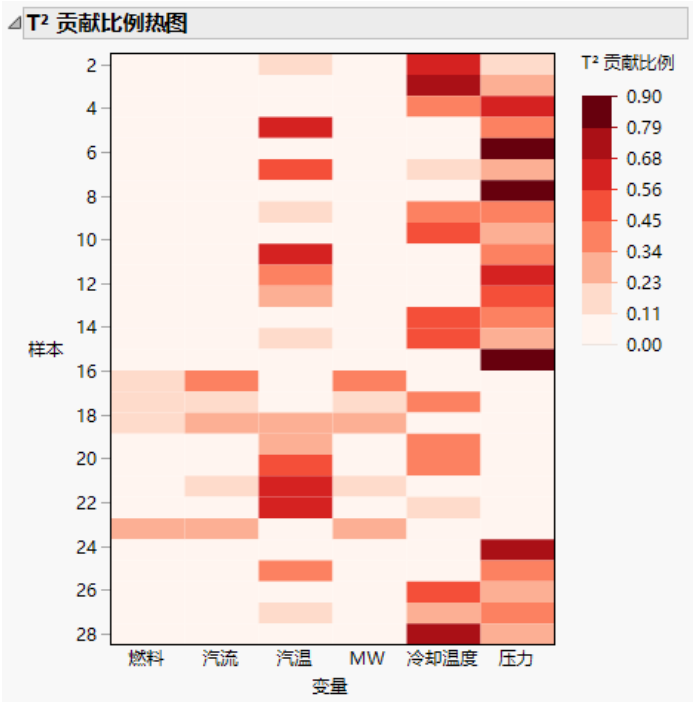
图 11.4 “气流”的单值图



单值图表明：汽流可能在样本 17 附近发生了扰动。

7. 在 PCA “模型驱动的多元控制图”报表窗口中，点击 “3 个主成分的 T^2 ”红色小三角并选择**贡献比例热图**。

图 11.5 贡献比例热图

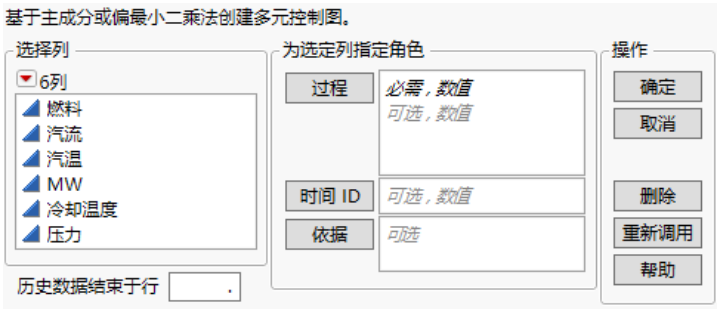


贡献比例热图显示，与其他行相比，第 16、17 和 18 行以及第 23 行（再次）的贡献比例发生了偏移。通常**气流温度**、**低温**和**压力**对每行的 T^2 值的贡献最多。

启动“模型驱动的多元控制图”平台

通过选择**分析** > **质量和过程** > **模型驱动的多元控制图**启动模型驱动的多元控制图 (MDMVCC) 平台。

图 11.6 “模型驱动的多元控制图” 启动窗口



有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“模型驱动的多元控制图” 平台启动窗口包含以下选项：

过程 分配过程列。请参见第 274 页的 “数据格式”。

时间 ID 分配用于标识样本的列。若未指定时间 ID，则用行号标识观测。若 “时间 ID” 列是一个时间，则该时间标识每个样本。若不是时间，则用 “时间 ID” 的数值标识每个样本。

依据 为依据变量的每个水平生成单独的报表。若分配了多个依据变量，则为依据变量的每种可能组合都生成一个单独的报表。

历史数据结束于行 指定行号以指示历史数据的结束位置。这将支持您基于历史数据计算图表限值。历史数据和当前数据都绘制在图表上。历史数据也称为阶段 I 数据，当前数据也称为阶段 II 数据。

数据格式

MDMVCC 平台接受采用以下三种允许的数据格式的数据：

原始数据 使用连续过程数据生成基于数据主成分的控制图。控制图的默认维基于占过程变异 85% 的主成分数量。该数字基于主成分特征值的累积百分比。

主成分 使用以前从主成分分析 (PCA) 中保存的主成分列。控制图的默认维是指定为过程变量的成分数。

偏最小二乘得分数据 使用之前从偏最小二乘 (PLS) 分析中保存的得分列来生成基于得分列的控制图。控制图的默认维是指定为过程变量的得分数。

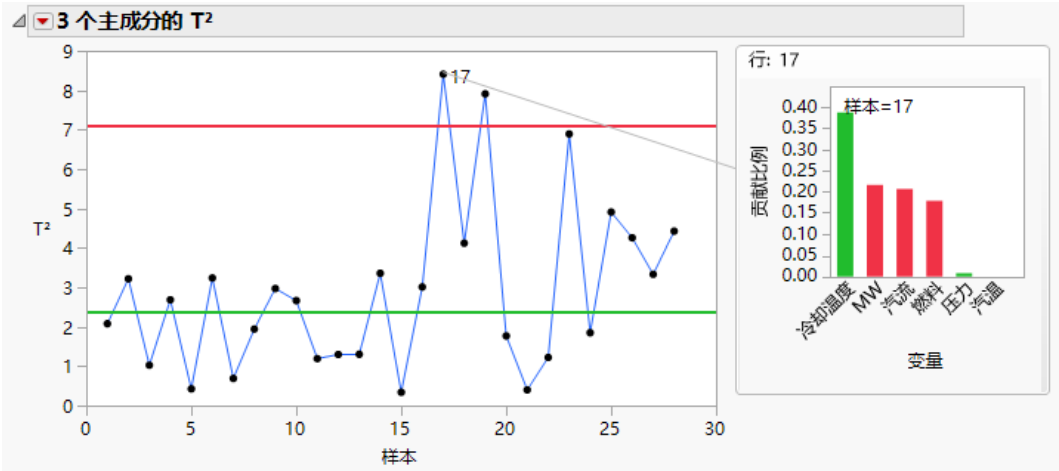
注意：

- 不支持使用频数或权重列生成的 PCA 或 PLS 模型。
- 使用历史数据生成的 PCA 或 PLS 模型必须使用 MDMVCC 启动窗口中指定的相同数量的历史数据行。
- 不支持在 “多元” 平台内构建的 PCA 模型。

“模型驱动的多元控制图”报表

最初的“模型驱动的多元控制图”报表显示 T^2 控制图。图上的悬停标签本身就是图表。点击悬停标签图可打开更大的图形版本。根据图表的不同，它们会在单独的报表窗口中打开，或者在 MDMVCC 报表的“诊断过程”部分中打开。您可以使用 graphlet 以交互方式下钻数据。

图 11.7 使用悬停 Graphlet 的 MDMVCC 报表



“模型驱动的多元控制图”平台选项

显示历史汇总统计量 显示或隐藏指定为历史数据的行或所有行（若未指定历史数据行）的汇总统计量。汇总统计量包括过程变量的一元均值和标准差。对于基于 PCA 的图形，显示特征值和特征向量。对于基于 PLS 得分的图表，显示得分的标准差和得分载荷。

监控过程

显示监控图 显示或隐藏选定的过程监控图。

设置成分 支持您设置 T^2、DModX 或 SPE 图的成分数。成分数可以是从 1 到高达 PCA 驱动分析的有效特征向量数；对于 PLS 驱动分析，成分数可以是从 1 到高达 PLS 模型因子的数目。

设置 α 水平 支持您调整用于所有控制图限值的 alpha 水平。

T^2 图 显示或隐藏 T^2 图。T^2 统计量是多元变异的汇总，它测量观测距离 PCA 或 PLS 模型中心有多远。

标准化 DModX 图 显示或隐藏标准化 DModX 值的图。DModX 测量每个观测与 PCA 或 PLS 模型的距离。较高的 DModX 值指示偏离数据的基本相关结构的观测。

平方预测误差图 显示或隐藏平方预测误差 (SPE) 图。SPE 测量 PCA 或 PLS 模型的残差平方和。较高的 SPE 值指示偏离数据的基本相关结构的观测。

得分图 显示或隐藏主成分或偏最小二乘因子的得分图。请参见第 277 页的“得分图”。

诊断过程 （在至少有一个诊断图处于活动状态时可用。）显示或隐藏诊断图。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

图选项

以下选项应用于 T^2、标准化 DModX、平方预测误差和得分图。一旦选定，这些图就会显示在报表窗口的“诊断和过程”部分中。

显示限值汇总 （不可用于得分图。）在图表下方的报表表中显示或隐藏控制图限值和汇总数据。

贡献热图 显示或隐藏按每个观测的变量贡献着色的热图。

贡献比例热图 显示或隐藏按表示为每个观测的总体值比例的变量贡献着色的热图。

选定样本的贡献图 （仅当选定一个或多个点时才可用。）显示或隐藏来自各个成分对每个选定样本总体值的贡献的条形图。

选定样本的贡献比例图 （仅当选定一个或多个点时才可用。）显示或隐藏表示为对每个选定样本的总体值贡献比例的各个成分贡献的条形图。

选定样本的均值贡献比例图 （仅当选定两个或更多点时才可用。）显示或隐藏来自各个成分对每个选定样本总体值的贡献平均值的条形图。

注意：当值或均值超出均值的 3 个 sigma 之外时，贡献图直条将显示为红色，否则显示为绿色。

为贡献图提供了以下选项：

对直条排序 支持您将直条按照从最大贡献到最小贡献排序，或者对于多个图则按照最大平均贡献到最小平均贡献排序。

添加直条标签 支持您按值、列名来标记直条，或删除标签（无标签）。

选定项的控制图 显示选定列的控制图。

散点图矩阵 （选定两个或更多直条时可用。）显示选定变量的散点图矩阵。

删除图 从报表窗口中删除图。

选定样本的标准化得分图 （仅当选定一个或多个点时才可用于得分图。）显示或隐藏选定的每个样本的标准化得分条形图。

得分椭圆覆盖率 （仅可用于包含两个成分的得分图。）将具有指定覆盖率的椭圆添加到得分图。该椭圆基于历史数据。当阶段 I 和阶段 II 数据都存在时，每个阶段都有一个椭圆，并且阶段 II 椭圆为虚线。

连接点 （仅可用于得分图。）连接得分图中的数据点。

显示载荷 （仅可用于得分图。）使用双标图箭头显示得分图上的 PCA 载荷。

保存列 对于每个图，都可通过三个选项将值保存到数据表中：

保存值 将值 (T^2、标准化 DModX、 SPE 或得分）保存到新的数据表列。

保存贡献 将贡献保存到新的数据表列。

保存贡献比例 将贡献比例保存到新的数据表列。

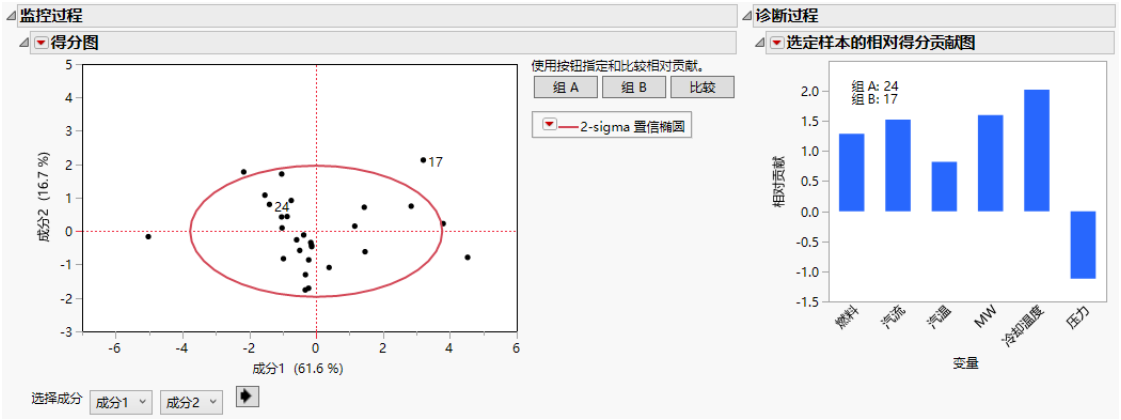
得分图

得分图显示主成分或偏最小二乘因子图。使用图下面的控件更改得分图中显示的成分。

使用图右侧的按钮分配和比较两组点之间的相对贡献。相对贡献显示两个或更多样本之间的差异。相对贡献显示基础过程变量中的哪些变化导致了样本组之间的差异。一个用途是调查受控样本与失控样本之间的差异。

组 A 为参考组，组 B 为对照组。每组都包含一个或多个点。若要分配参考组，请选择一个或多个点，然后点击“组 A”。若要分配对照组，请选择一个或多个点，然后点击“组 B”。若要显示“相对得分贡献图”，请点击“比较”。

图 11.8 包含第 17 行相对于第 24 行的相对贡献图的得分图



“模型驱动的多元控制图”平台的更多示例

- “包含历史数据的 MDMVCC 的示例”

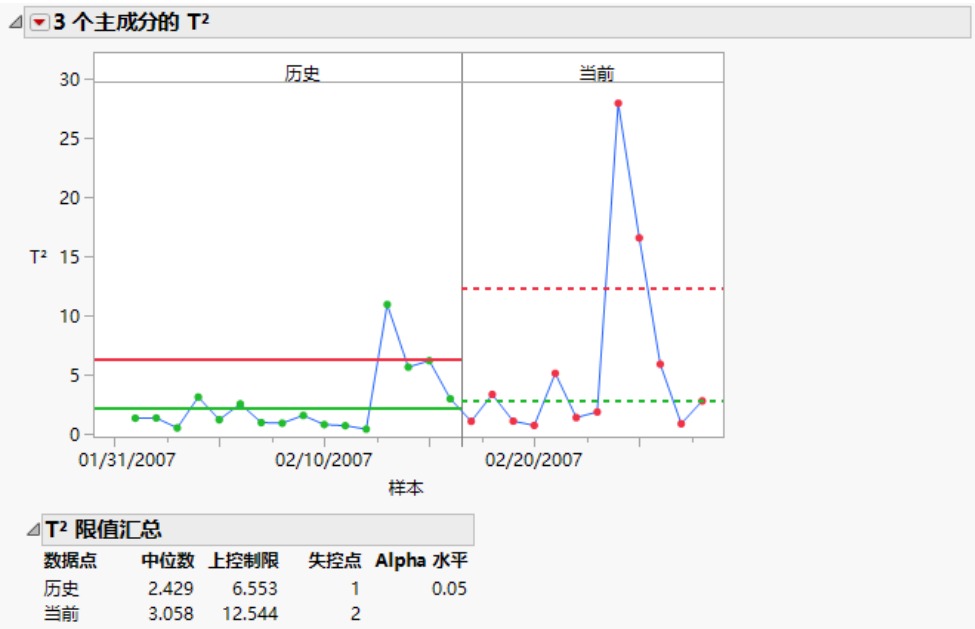
- “包含 PLS 模型的 MDMVCC 的示例”

包含历史数据的 MDMVCC 的示例

本例演示如何使用历史数据设置当前数据的监控限值。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Flight Delays.jmp。
2. 选择**分析** > **质量和过程** > **模型驱动的多元控制图**。
3. 从 AA 一直选择到 WN 并点击**过程**。
4. 选择**航班日期**并点击**时间 ID**。
5. 为**历史数据结束于行**输入 16。
6. 点击**确定**。

图 11.9 历史和当前数据的 T^2 图



请注意有两组限值。一组适用于历史数据。另一组限值应用于当前数据。有关如何使用历史数据计算两组限值的详细信息，请参见第 282 页的“限值”。

提示：启用“自动重新计算”可支持图表在您向数据表添加其他观测时自动更新。点击“PCA 模型驱动的多元控制图”红色小三角时，“自动重新计算”选项处于“重新运行”之下。

包含 PLS 模型的 MDMVCC 的示例

本例演示了如何使用 PLS 模型来监控多元过程。考虑一个有 14 个输入和 5 个质量变量的过程。您有一个 PLS 模型来解释过程，您想用该模型来监控过程中的偏差。

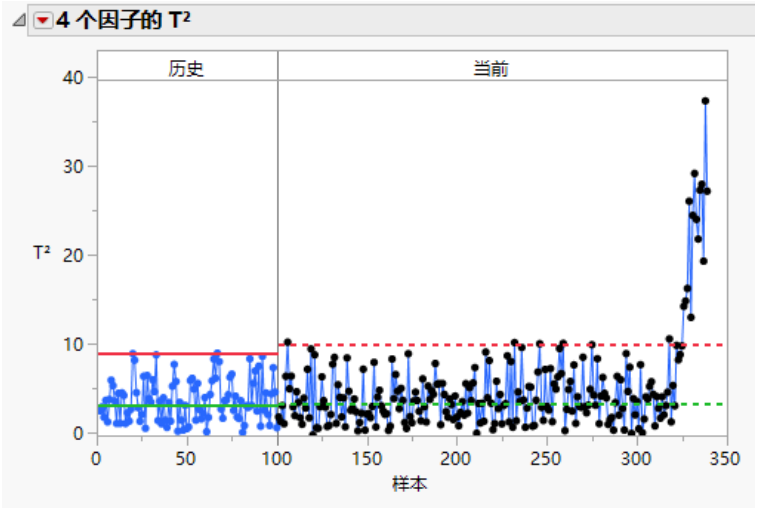
1. 选择**帮助** > **样本数据库**，然后打开 Polyethylene Process.jmp。

   该数据表包含 14 个过程变量和 5 个质量或输出变量。前 100 行是历史数据，用于构建描述该过程的 PLS 模型。这些行着蓝色。其余 239 行是自构建模型以来收集的数据。

   偏最小二乘模型找到 4 个得分函数来描述该过程。这些函数保存到数据表从 **X 得分 1 公式**到 **X 得分 4 公式**的各列中。要构建 PLS 模型，请使用表脚本**将当前数据设置为已排除**来排除历史数据之后收集的 239 行数据。然后使用**拟合偏最小二乘**表脚本构建 PLS 模型，将质量变量与过程变量关联起来。

2. 选择**分析** > **质量和过程** > **模型驱动的多元控制图**。
3. 从 **X 得分 1 公式**一直选择到 **X 得分 4 公式**，然后点击**过程**。
4. 将**历史数据结束于行**设置为 100。
5. 点击**确定**。

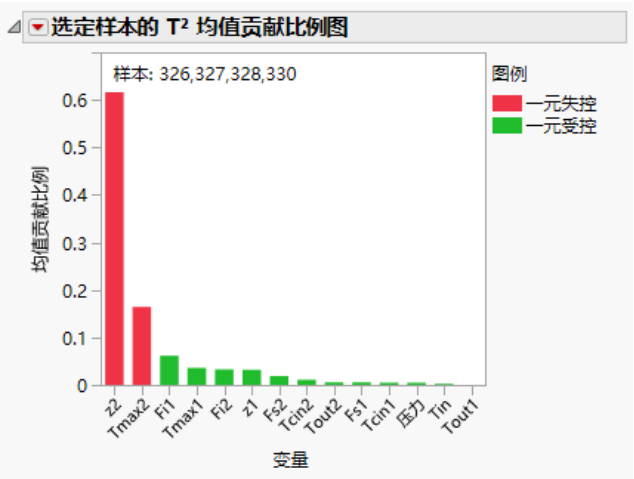
图 11.10 T^2 图



蓝色数据点表示历史数据。黑色数据点为建立控制图之后收集的数据点。该过程在样本编号 326 附近开始发生了扰动。

6. 悬停在上控制限上方的样本数据点上，以查看贡献图。
7. 选择上控制限上方的第一个数据点聚类。点击“4 个因子的 T^2”红色小三角并选择**选定样本的均值贡献比例图**。
8. 点击 “选定样本的 T2 均值贡献比例图”旁边的红色小三角，然后选择**为直条排序**。

图 11.11 均值贡献比例图



请注意，贡献图是根据 PLS 模型输入变量绘制的。看起来 z2 和 Tmax2 引起了过程扰动。Tmax2 和 z2 相关。Tmax2 是反应器温度，z2 是 Tmax2 温度的位置。

注意：这些因子的说明记录在“注释”列属性中。

“模型驱动的多元控制图”平台的统计详细信息

监控统计量

T^2

每个 i 观测的 T_i^2 值都绘制在 T^2 控制图上。对于历史和当前数据，具有 k 个成分的 PCA 或 PLS 模型的 T^2 值定义如下：

$$T_i^2 = t_i^T S_k^{-1} t_i$$

其中：

t_i = 第 i 个观测的 k 得分的向量

$\mathbf{S}_k$ = 历史观测的 k 得分的对角线样本协方差矩阵

对于 PCA 模型，$\mathbf{S}_k$ 是对角线特征值矩阵。

当数据在数据预处理步骤中已中心化时，k 个历史得分向量的均值为 0。该步骤发生在基于相关性或协方差的 PCA 以及进行中心化的 PLS 中。对于 **X** 未中心化的预处理选项，则假设数据已

经由用户中心化，所以每个 k 得分向量的均值为 0。有关 Hotelling's T^2 的详细信息，请参见 Montgomery (2013)。

SPE

对于 PCA 和 PLS 模型，预处理的 **X** 矩阵可分解为：

$$X = T_k P_k^T + E$$

其中，$T_k = (t_1,...,t_k)$ 是 k 维得分矩阵，$P_k = (p_1,...,p_k)$ 对于 PCA 模型是具有前 k 个特征向量的矩阵，而对于 PLS 模型是载荷矩阵。该 PCA 或 PLS 模型的平方预测误差用于 SPE 控制图。

每个 i 观测的 SPE_i 值都绘制在 SPE 控制图上。平方预测误差定义为：

$$SPE_i = e_i^T e_i = \sum_{j=1}^{p} e_{ij}^2$$

其中

e_i = 观测 i 的残差向量

p = 变量数

DModX

每个 i 观测的 $DModX_i$ 值都绘制在 DModX 控制图上。与模型的标准化距离 (DModX) 定义为：

$$DModX_i = \frac{SPE_i/(df_1)}{\left(\sum_{i=1}^{n} SPE_i\right)/(df_2)} = \frac{\sum_{j=1}^{p} e_{ij}^2/(df_1)}{\sum_{i=1}^{n}\sum_{j=1}^{p} e_{ij}^2/(df_2)} = \frac{\sum_{j=1}^{p} e_{ij}^2}{d}$$

其中

e_{ij} = 观测 i 和变量 j 的残差

$df_1 = p-k$

$df_2 = (n-k-1)(p-k)$ （若数据已中心化）；或为 $(n-k)(p-k)$ （若数据未中心化）

n = 历史数据观测数

k = PCA/PLS 成分数

p = 变量数

注意： $DModX_i$ 等于以 $1/d$ 为尺度的 SPE_i。

限值

当启动窗口中未指定历史行数时，所有数据都被视为历史数据。请参见第 273 页的“启动“模型驱动的多元控制图”平台”。

T^2

历史数据的上控制限 (UCL) 基于 Beta 分布并定义如下：

$$UCL = \frac{(n-1)^2}{n}\beta\left[1-\alpha;\frac{k}{2};\frac{n-k-1}{2}\right]$$

其中：

n = 历史数据观测数

k = PCA 或 PLS 成分数

$\beta\left[1-\alpha;\frac{k}{2};\frac{n-k-1}{2}\right]$ = $\text{Beta}\left[\frac{k}{2};\frac{n-k-1}{2}\right]$ 分布的第 (1−α) 分位数。

当前数据的上控制限基于 F 分布并定义如下：

$$UCL = \frac{k(n+1)(n-1)}{n(n-k)}F[1-\alpha;k;(n-k)]$$

其中：

n = 历史数据观测数

k = PCA 或 PLS 成分数

$F(1-\alpha; k; n-k) = F(k; n-k)$ 分布的第 (1−α) 分位数。

DModX

对于 PCA 和 PLS 模型，上控制限基于 F 分布。PCA 模型的 DModX 上控制限定义为：

$$UCL = F[1-\alpha;df_1;df_2]$$

其中：

$df_1 = p-k$

$df_2 = (n-k-1)(p-k)$ （若数据已中心化）；或为 $(n-k)(p-k)$ （若数据未中心化）

n = 历史数据观测数

k = PCA 成分数

p = 变量数

$F(1-\alpha; n-p-1; p-k) = F(n-p-1; p-k)$ 分布的第 (1−α) 分位数

PLS 模型的 DModX 上控制限定义为：

$$UCL = F[1-\alpha;h;nh]$$

其中：

$h = (2\hat{\mu}_{SPE}^2)/(\hat{\sigma}_{SPE}^2)$

$\hat{\mu}_{SPE}$ = SPE 的历史样本均值

$\hat{\sigma}_{SPE}^2$ = SPE 的历史样本方差

n = 历史数据观测数

$F(1-\alpha; h; nh) = F(h; nh)$ 分布的第 $(1-\alpha)$ 分位数。

SPE

PCA 模型的 SPE 上控制限定义为：

$$UCL = \theta_1\left[1 - \frac{\theta_2 h_0(1-h_0)}{\theta_1^2} + \frac{z_{1-\alpha}(2\theta_2 h_0^2)^{1/2}}{\theta_1}\right]^{1/h_0}$$

其中：

$h_0 = 1 - 2\theta_1\theta_3/(3\theta_2^2)$

$\theta_1 = \sum_{a=1}^{k} \lambda_a$

$\theta_2 = \sum_{a=1}^{k} \lambda_a^2$

$\theta_3 = \sum_{a=1}^{k} \lambda_a^3$

λ_a = 第 a 个特征值

k = PCA 成分数

$z_{1-\alpha}$ = 标准正态分布的第 $(1-\alpha)$ 分位数

有关用于 PCA 模型的 SPE 控制限的详细信息，请参见 Jackson and Mudholkar (1979)。

对于 PLS 模型，上控制限基于卡方分布并定义如下：

$$UCL = g\chi^2_{[1-\alpha;h]}$$

其中

$$g = (\hat{\sigma}^2_{SPE})/(2\hat{\mu}_{SPE})$$

$$h = (2\hat{\mu}^2_{SPE})/(\hat{\sigma}^2_{SPE})$$

$\hat{\mu}_{SPE}$ = SPE 的历史样本均值

$\hat{\sigma}^2_{SPE}$ = SPE 的历史样本方差

$X^2(1-\alpha; h) = X^2(h)$ 分布的第 $(1-\alpha)$ 分位数

g 和 h 参数通过矩量法估计。有关用于 PCA 模型的 SPE 控制限的详细信息，请参见 Nomikos (1995)。

贡献

T^2

具有 p 个变量和 k 个成分的 PCA 或 PLS 模型的 T^2 贡献计算如下：

$$\begin{aligned} T_i^2 &= t_i^T S_k^{-1} t_i \\ &= \sum_{a=1}^{k} \frac{t_{ia}^2}{s_a} \\ &= \sum_{a=1}^{k} \frac{t_{ia}}{s_a} \sum_{j=1}^{p} r_{ja} x_{ij} \\ &= \sum_{j=1}^{p} \left(\sum_{a=1}^{k} \frac{t_{ia}}{s_a} r_{ja} x_{ij} \right) \end{aligned}$$

其中：

t_i = 第 i 个观测的 k 得分的向量

$\mathbf{S}_k$ = 历史观测的 k 得分的对角线样本协方差矩阵。对于 PCA 模型，$\mathbf{S}_k$ 是对角线特征值矩阵。

s_a = $\mathbf{S}_k$ 的第 a 个对角线元素

r_{ja} = 对于 PCA 模型是第 a 个特征向量的第 j 个元素，对于 PLS 模型则是 $\mathbf{R}_k$ 载荷矩阵的第 a 列。 $\mathbf{R}_k$ 是用于将得分矩阵 $\mathbf{T}_k$ 关联到 $\mathbf{X}$ 矩阵的矩阵，使得 $\mathbf{T}_k=\mathbf{X}\mathbf{R}_k$。

x_{ij} = 第 i 个观测的第 j 个变量的值。

注意：最后一个总和中的 p 项数是变量贡献。

每个变量的贡献是其对每个得分的贡献之和，按标准化得分值来加权。若标准化得分值较大，且变量贡献较大，则认为变量对 T_i^2 的贡献较大。

$$\sum_{j=1}^{p} Cont(T_i^2)_j = T_i^2$$

变量 j 的贡献比例定义为：

$$ContProp(T_i^2)_j = \frac{Cont(T_i^2)_j}{\sum_{j=1}^{p} Cont(T_i^2)_j}$$

注意：计算 T^2 贡献比例时，JMP 会将负贡献归零。在 PCA 和 PLS 模型中，由于变量在 **X** 投影时的交互作用，可能会产生负贡献。负贡献清零，以确认表示占总的正贡献较大比例的变量贡献。

有关 PCA 贡献和负贡献的详细信息，请参见 Kourti and MacGregor (1996)。有关 PLS 贡献的详细信息，请参见 Li et al.(2009)。

DModX

对于 PCA 和 PLS 模型，变量 j 对 DModX_i 的贡献定义如下：

$$Cont(DModX_i)_j = \frac{e_{ij}}{\sqrt{d}}$$

请注意，由于

$$DModX_i = \sum_{j=1}^{p} \frac{e_{ij}^2}{d}$$

变量 j 的贡献比例定义为：

$$ContProp(DModX_i)_j = \frac{e_{ij}^2/d}{DModX_i}$$

SPE

对于 PCA 和 PLS 模型，变量 j 对 SPE_i 的贡献定义如下：

$$Cont(SPE_i)_j = e_{ij}$$

请注意，由于

$$SPE_i = \sum_{j=1}^{p} e_{ij}^2$$

变量 j 的贡献比例定义为：

$$ContProp(SPE_i)_j = \frac{e_{ij}^2}{SPE_i}$$

得分贡献

得分贡献计算与 T^2 贡献相同，但仅针对得分图中选定的维进行计算。

得分图分组比较

对于得分图分组比较，变量 j 的相对得分贡献是组 B 的平均贡献与组 A 的平均贡献之差：

$$\sum_{i \in B} \frac{Cont(T_i^*)_j}{n_b} - \sum_{i \in A} \frac{Cont(T_i^*)_j}{n_a}$$

其中

T^*_i = 得分矩阵的第 i 行，矩阵中的列与得分图中显示的维相对应。

A = 组 A 中的一组观测

B = 组 B 中的一组观测

n_a = 组 A 中的观测数

n_b = 组 B 中的观测数

得分图载荷

得分图上显示的载荷基于 PCA 特征向量或 PLS X 得分载荷（**R** 矩阵）。这些载荷按得分的最大绝对值统一尺度。统一尺度是为了在得分图上绘制载荷。这些载荷演示了每个变量对每个得分的大致影响。

第 12 章

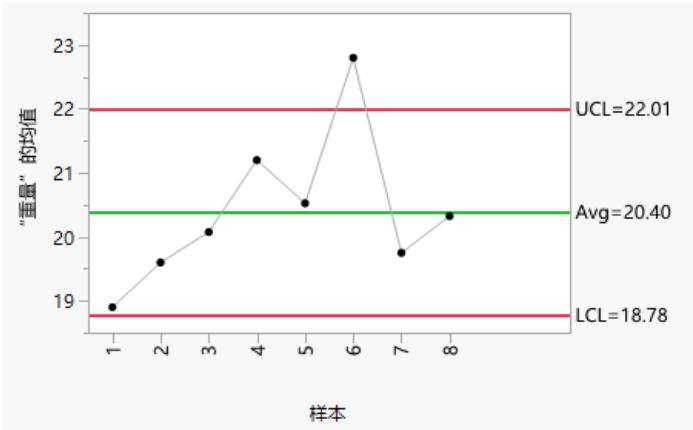
传统控制图

创建计量控制图和计数控制图

控制图是一种用于监控过程变异的图形化分析工具。过程中的自然变异可使用一组控制限来量化。控制限可帮助区分一般原因变异和特殊原因变异。通常会采取措施来确定和消除特殊原因变异。对过程中的一般原因变异进行量化也很重要，因为这决定了过程能力。

JMP 中的传统控制图平台提供各种控制图，以及运行图、V 形模板 CUSUM 图和加权移动平均图。为了支持对过程进行完善，多数控制图选项都支持在同一个图上为项目的不同阶段显示单独的控制图。

图 12.1 控制图示例



目录

传统控制图示例 289
传统控制图类型 290
计量控制图 290
计数控制图 292
启动“传统控制图”平台 293
过程信息 294
限值指定 296
指定的统计量 297
传统控制图报表 297
V 形模板 CUSUM 图报表 299
解释双侧 V 形模板 CUSUM 图 299
解释单侧 CUSUM 图 301
“传统控制图”平台选项 301
传统控制图的窗口选项 301
传统控制图的图形选项 304
V 形模板 CUSUM 控制图的图表选项 306
保存和检索限值 306
排除、隐藏和删除的样本 309
“控制图”平台的更多示例 310
预先汇总图示例 310
V 形模板 CUSUM 图示例 312
单侧 CUSUM 图示例 314
UWMA 图示例 315
“控制图”平台的统计详细信息 315
中位数移动极差图的控制限 316
能力分析的统计详细信息 316
V 形模板 CUSUM 控制图的统计详细信息 321
加权移动平均图的统计详细信息 324

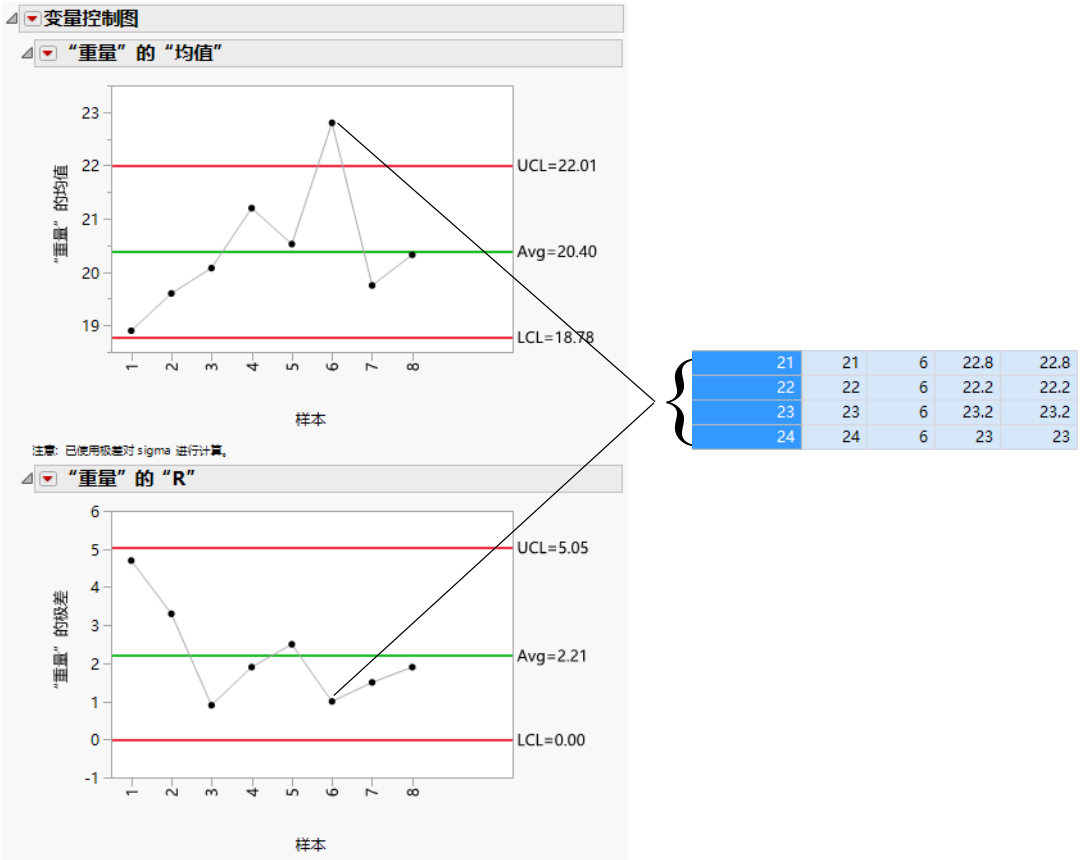
传统控制图示例

本例使用 Quality Control 样本数据文件夹中的 Coating.jmp 样本数据表（数据取自 ASTM Manual on Presentation of Data and Control Chart Analysis）。关注的质量特征是**重量**列。选定了大小为 4 的子组样本。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Coating.jmp。
2. 选择**分析** > **质量和过程** > **传统控制图** > **均值**。

   请注意选定的图表类型为**均值**和 R。
3. 选择**重量**并点击**过程**。
4. 选择**样本**并点击**样本标签**。
5. 点击**确定**。

图 12.2 涂层数据的计量图



过程的均值图和 R 图显示在图 12.2 中。样本 6 指示该过程为失控状态。要检查样本值，请点击任一控制图上的样本 6 汇总点。相应行在数据表中突出显示。

注意：若随均值图选定了 S 图，则均值图的限值基于标准差。否则，均值图的限值基于极差。

传统控制图类型

传统控制图大致分为两类：

- “计量控制图” — 单值极差、均值、运行图、Levey-Jennings、预先汇总、CUSUM、UWMA 和 EWMA。
- “计数控制图” — P、NP、C 和 U

计量控制图

计量控制图根据在图上标绘的子组汇总统计量来分类：

- **单值极差**选项提供其他图表类型：
  - 单个测量值图显示单个测量值。仅当每个子组样本仅有一个测量值时这些图才适用。
  - 移动极差图显示两个或更多相继测量值的移动极差。请参见第 291 页的“移动极差（平均）图”。
- **均值**图显示子组均值（平均值）。该选项提供其他图表类型：
  - R 图显示子组极差（最大值 – 最小值）
  - S 图显示子组标准差。

  对于在连续尺度上测量的质量特征，一个典型的分析会显示出过程均值及其变异性，并且会有一个均值图在其对应的 R 图或 S 图上方对齐。
- **运行图**将数据显示为一系列连接点。当通过窗口或脚本使用了**样本标签**角色后，运行图还可以标绘组均值。
- Levey-Jennings 图显示过程均值，以及基于长期 sigma 的控制限。控制限位于距离中心线 3 *s* 远的位置。Levey-Jennings 图的标准差 *s* 计算方法与“分布”平台中的标准差计算方法相同。
- **预先汇总**图显示子组均值和标准差。请参见第 291 页的“预先汇总图”。
- CUSUM 图显示子组或单个测量值距离目标值的累积和。请参见第 291 页的“V 形模板 CUSUM 控制图”。
- UWMA 图显示指定数量的测量值的等加权移动平均值。请参见第 292 页的“等加权移动平均图”。
- EWMA 图显示具有指定权重的所有测量值的指数加权移动平均值。请参见第 292 页的“指数加权移动平均图”。

移动极差（平均）图

在移动平均图中，取平均值的数量可以是单个观测而不是子组均值。不过，单个测量值的移动平均图不同于单个测量值或标绘有单个测量值的移动极差的控制图。

移动极差（平均）图显示两个或更多相继测量值的移动极差。针对您在**极差跨度**框中输入的相邻测量值个数计算移动极差。默认极差跨度为 2。由于移动极差彼此相关，所以应慎重地解释这些图。

同时提供移动极差中位数图。若同时选择移动极差中位数图和单个测量值图，单个测量值图上的限值将移动极差中位数而不是平均移动极差用作 sigma。

预先汇总图

若您的数据由同一过程单位的重复测量值构成，则可以将这些测量值合并为该单位的一个测量值。除非数据中的每个过程或测量单位都有重复的测量值，否则不推荐使用预先汇总。

通过预先汇总，可基于所选的样本大小或样本标签将过程列汇总到样本均值或标准差中。然后基于在启动窗口中所选的选项绘制汇总数据。通过选中启动窗口中相应的框，您还可以追加能力分析。

预先汇总启动窗口包含以下图表类型选项：

- 基于组均值的单值
- 基于组标准差的单值
- 基于组均值的移动极差
- 基于组标准差的移动极差
- 基于组均值的移动极差中位数
- 基于组标准差的移动极差中位数

还有一个选项用于设置用于移动极差图表类型的极差跨度。

V 形模板 CUSUM 控制图

V 形模板累积和 (CUSUM) 控制图显示子组或单个测量值距离目标值的累积和。V 形模板 CUSUM 图可帮助您通过检测过程均值中的持续小偏移，确定过程是否处在统计控制状态中。相比之下，标准 Shewhart 控制图可检测测量值中的突发大变化，如两个或三个 sigma 偏移，但这种图对于发现更小变化（如一个 sigma 偏移）不够有效。

CUSUM 菜单选项提供用于 V 形模板累积和图的选项。除 KSigma 之外，您还可以指定：

- 双侧图的 V 形模板原点与 V 形模板上臂或下臂之间的垂直距离 h。对于单侧图，H 即决策区间。将 H 选作标准误差的倍数。
- 参考值 k，其中 k 大于零。

累积和控制图的另一种形式是表格式 CUSUM 图。要创建表格式 CUSUM 图，请参见第 219 页的“CUSUM 控制图”一章。推荐使用表格式 CUSUM 图而不是 V 形模板图有多种原因，具体包括：

- V 形模板必须随每个观测移动，而不是简单地置于最后一个观测上。
- V 形模板过程中的累积和可能远离图形中心，即便是对于对准目标过程也是如此。

警告：Montgomery (2013) 强烈“建议不使用 V 形模板过程。”

等加权移动平均图

均匀加权移动平均 (UWMA) 图上的各点是 w 个最近子组均值（包括当前子组均值）的平均值。在获取新的子组样本后，下一个移动平均值的计算方式是：去除之前 w 个子组均值中最旧的均值，代之以最新的子组均值。常数 w 称为移动平均的**跨度**。

除 KSigma 和 Alpha 之外，在 UWMA 启动窗口中您还可以指定：

- 移动平均跨度（或 w），这指示计算移动平均值需要包含多少个子组。移动平均跨度 (w) 越大，UWMA 线越光滑，它所反映的偏移量级就越小。这意味着较大的 w 值可防止较小的偏移。请参见第 324 页的“UWMA 图的控制限”。

指数加权移动平均图

指数加权移动平均 (EWMA) 图上的各点是所有以前的子组均值（包括当前子组样本的均值）的加权平均值。权重随着时间的回溯呈指数级下降。

注意：指数加权移动平均 (EWMA) 图亦称几何移动平均 (GMA) 图。

除 KSigma 和 Alpha 之外，在 EWMA 启动窗口中您还可以指定：

- 一个“权重”参数，该参数是指定给当前子组样本均值的权重 ($0 < \text{权重} \leqslant 1$)。较小的权重值可用于防止较小的偏移。请参见第 324 页的“EWMA 图的控制限”。

提示：请参见第 233 页的“EWMA 控制图”一章，了解更新的“EWMA 控制图”平台。

计数控制图

在之前的各种图中，测量值数据均为过程变量。该类型的数据往往是连续的，所以这些图基于适用于连续数据的原理。另一种类型的数据是计数数据，其中关注的变量是每个子组的缺陷数或瑕疵数的离散计数。离散计数数据适用于计数图，因为这些数据基于二项模型和 Poisson 模型。由于是按子组来测量计数，所以在比较各图时，确定各图之间的子组中项数是否相似这点很重要。计数图和计量图一样，按照在图上标绘的子组样本统计量来分类。

确定使用哪一种计数图

每项若非合格即为不合格：

p 图 显示缺陷项的**比例**。

np 图 显示缺陷项的**数目**。

对于每项计算缺陷数：

c 图 显示缺陷**数**。

u 图 显示缺陷**比例**。

对于计数图，将包含缺陷计数或缺陷比例的列指定为 “过程”变量。数据被解释为计数，除非该列包含 0 到 1 之间的非整数值。

- P 图显示子组样本（大小可能会发生变化）中不合格（缺陷）项的比例。由于 P 图的每个子组都包含 N_i 项，而且任一项若非合格即为不合格，所以子组中的不合格项的最大数目为 N_i。
- NP 图显示子组样本中不合格（缺陷）项的数目。由于 NP 图的每个子组都包含 N_i 项，而且任一项若非合格即为不合格，所以子组 i 中的不合格项的最大数目为 N_i。

  注意：要对 P 图或 NP 图使用 Sigma 列属性，该值需要等于比例。 JMP 将 sigma 计算为比例和样本大小的函数。

- C 图显示子组样本（通常但不一定总是包含一个检查单位）中不合格项（缺陷项）的数目。

  警告: 对于 C 图，若您未指定“样本大小”或“常数大小”，则将“样本标签”作为样本大小。

- U 图显示在可能包含不同数目检查单位的每个子组样本中不合格项（缺陷项）的比例。

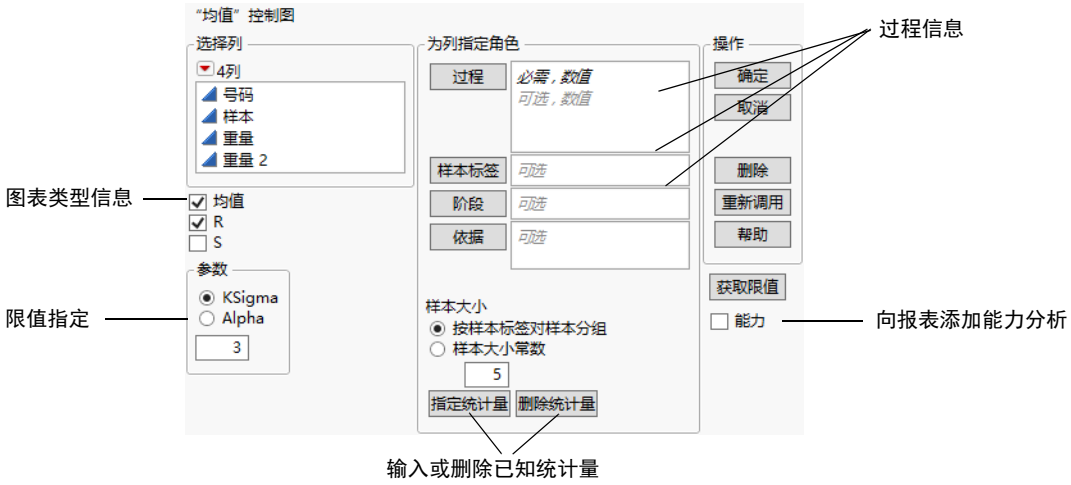
  警告: 对于 U 图，若您未指定“单位大小”或“常数大小”，则将“样本标签”作为单位大小。

启动“传统控制图”平台

通过选择**分析 > 质量和过程 > 传统控制图**启动“传统控制图”平台时，会显示类似于图 12.3 的启动窗口。特定控件随您选择的图表类型而变化。初始窗口显示以下几类信息：

- “过程信息”用于选择测量变量
- 图表类型信息（详细信息，请参见第 290 页的“传统控制图类型”）
- 第 296 页的“限值指定”
- 第 297 页的“指定的统计量”

图 12.3 “均值控制图”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

过程信息

启动窗口显示当前数据表中列的列表。在此，您可以指定要分析的变量和子组样本大小。

过程

过程角色选择要绘图的变量：

- 对于计量图，将测量值指定为过程。
- 对于计数图，将缺陷计数或缺陷比例指定为过程。数据被解释为计数，除非其中包含 0 到 1 之间的非整数值。

注意：数据表中的行必须按照观测的收集顺序进行排序。即便指定了**样本标签**变量，您仍须对观测进行相应的排序。

样本标签

样本标签角色支持您指定某个变量，该变量的值不仅可标注水平轴，还可标识不等子组大小。若未指定样本标签变量，样本将通过其子组的样本号来标识。

- 若样本子组的大小相同，请选择**样本大小常数**选项并在文本框中输入大小。若输入了“样本标签”变量，其值将用于标注水平轴。不论样本中是否有缺失值，样本大小都用在限值计算中。
- 若各个样本子组中的行数不等或子组中有缺失值，而您有一列用来标识各个样本，请选择**按样本标签对样本分组**选项，并输入样本标识列作为样本标签。

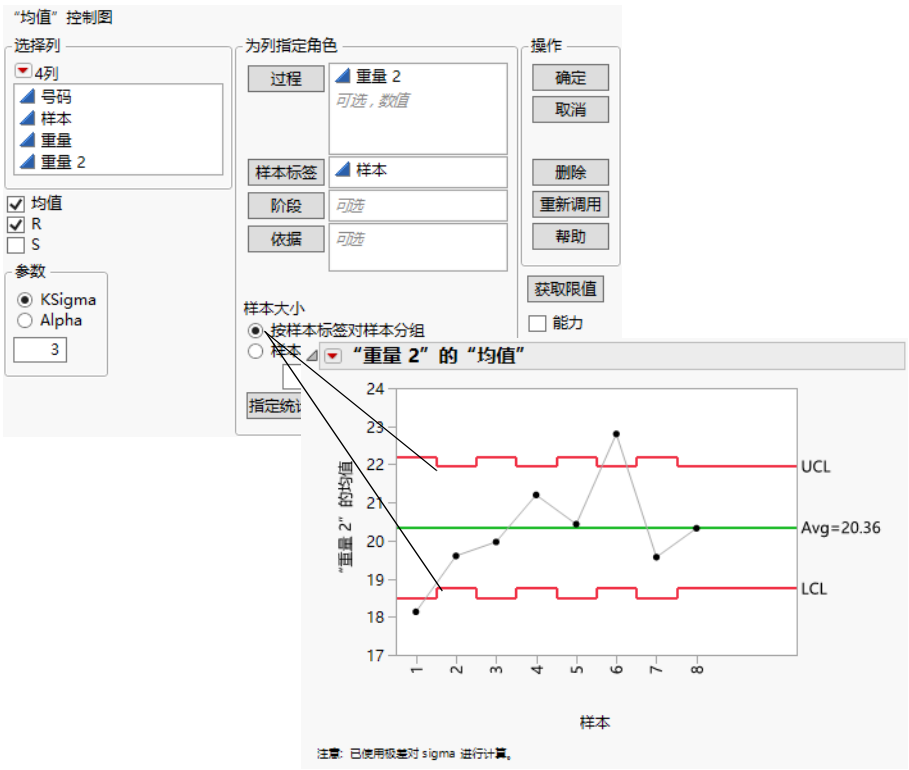
对于计数图（P、NP、C 和 U 图），该变量是子组样本大小。启动窗口中还会显示其他选项，包括**样本大小**、**常数大小**或**单位大小**，具体取决于您所做的选择。在计量图中，该变量标识样本。若图表类型为**单值极差**，则会显示**极差跨度**文本框。**极差跨度**指定基于其计算移动极差的相邻测量值数。

注意：

- 数据表中的行必须按照观测的收集顺序进行排序。即便指定了**样本标签**变量，您仍须对观测进行相应的排序。
- **常数大小**值的非整数部分将被截断。若具有常数非整数子组样本大小，您必须指定一列常数值。

图 12.4 中的演示使用 Quality Control 样本数据文件夹中的 Coating.jmp 样本数据，显示了一个子组样本大小不等的过程的均值图。

图 12.4 子组样本大小不等的计量图



阶段

阶段角色支持您指定标识不同阶段或部分的列。一个**阶段**就是数据表中的一组相邻观测。例如，阶段可能对应于时间期间，在这些期间内，生产中引入了新过程，而后完成了陆续的变更。使用阶段可为指定的“阶段”变量的每个水平生成新的 sigma、一组限值、区域和结果检验。

在均值、R、S、单值极差、P、NP、C、U、预先汇总和 Levey-Jennings 图的窗口中，将显示**阶段**变量按钮。若指定了阶段变量，将逐行检查该阶段变量，以标识每行所属的阶段。保存至限值文件可显示为每个阶段计算的 sigma 和特定限值。

依据

依据角色标识一个变量，用于为列中出现的每个值生成单独分析。

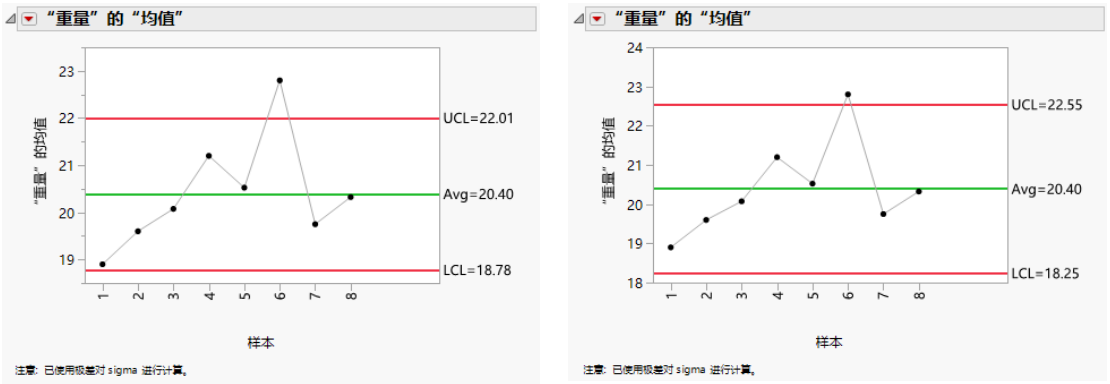
限值指定

您可以通过以下方式指定控制限计算方式：输入 *k* (K Sigma) 的值、输入 α(Alpha) 的概率，或者从过程列属性或以前创建的“限值表”中检索限值。第 306 页的“保存和检索限值”一节对“限值表”和**获取限值**按钮进行了说明。必须指定 K Sigma 或 Alpha。K Sigma 的窗口默认值为 3。

KSigma

KSigma 参数选项支持以样本标准误差的倍数形式指定控制限。KSigma 在期望值（显示为中心线）上下 *k* 个样本标准误差的位置指定控制限。要指定 *k*（sigma 数目），点击 KSigma 对应的单选按钮，然后在文本框中输入一个正数 *k* 值。一般选择 *k* 为 3，即三个标准差。图 12.5 中显示的示例对 Coating.jmp 数据的两个均值图进行了比较，这两个图分别使用 KSigma = 3 和 KSigma = 4 绘制控制线。

图 12.5 K Sigma =3 （左图）和 K Sigma=4 （右图）时的控制限



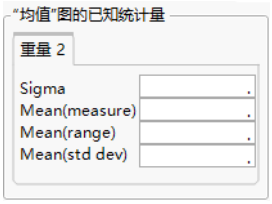
Alpha

Alpha 参数选项以概率形式指定控制限（亦称**概率界限**），这是通过单个子组统计量超过其控制限的概率 α 来表示的（假定过程受控）。要指定 alpha，请点击 Alpha 单选按钮并输入所需的概率。α 的合理选择为 0.01 或 0.001。对于假定服从正态分布且具有已知受控参数的均值图，与 KSigma 为 3 等价的 Alpha 值是 0.0027。

指定的统计量

指定某个过程变量后，若点击控制图启动窗口上的**指定统计量**（若可用）按钮，窗口底部将追加一个选项卡，其中包含可编辑的字段。该选项卡支持您输入过程变量的历史统计量（即，来自历史数据的统计量）。“控制图”平台使用这些条目来构造控制图。下例显示 1 作为过程变量的标准差，20 作为测量值均值。

图 12.6 “指定统计量”示例



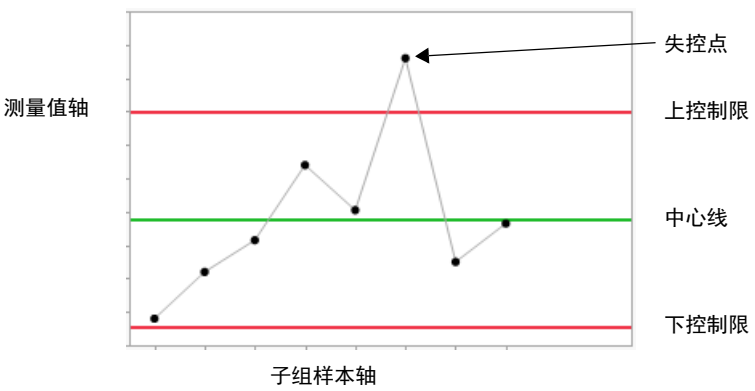
注意：若均值是用户指定的，其在图中标注为 μ0。

若选定了“控制图”启动窗口上的**能力**选项（图 12.3），平台启动时会显示一个窗口，提示您输入规格限。选定的控制图标准差会发送到该窗口中，并显示为“指定 Sigma”的值，这是默认选项。输入规格限并点击“确定”后，能力输出会显示在同一窗口中控制图的旁边。有关如何计算能力指标的信息，请参见“过程能力”一章中第 209 页的“正态分布的能力指标”。

传统控制图报表

该分析会生成一个图表，可用于确定某个过程是否处于统计控制状态中。报表随您选择的图表类型而变化。图 12.7 显示简单控制图的组成部分。控制图随着在数据表中添加或更改数据而动态更新。

图 12.7 控制图示例



注意：数据表中已排除的任何行也会在运行图、P 图、U 图和 C 图中隐藏。

控制图具有以下特征：

- 图上标绘的每个点都代表单个过程测量值或汇总统计量。在图 12.7 中，各点表示测量值样本的平均值。

  应**合理**选择子组；也就是说，在选择子组时，应该让在子组**之间**观测到真正的过程变化的概率最大化。通常，这需要了解确定最有效分组策略的过程。请参见 Wheeler (2004)； Woodall and Adams (1998)。
- 控制图的垂直轴尺度按照与汇总统计量相同的单位来调整。
- 控制图的水平轴标识子组样本并按照时间来排序。随时间推移观测过程对于评估过程是否在变化非常重要。
- 绿线是中心线，即表示数据的平均值。当过程处于统计控制状态时，中心线指示汇总统计量的平均值（期望值）。测量值应均匀分布在中心线两侧。如若不然，则很可能表明过程平均值发生了变化。
- 两条红线分别为上控制限和下控制限，相应地标记为“上控制限”和“下控制限”。当过程处于统计控制状态时，这些限值可给出汇总统计量期望的变异范围。若过程仅显示出常规变异，则所有点都会随机散布在该范围内。在图 12.7 中，有一个测量值位于上控制限上方。这表明该测量值可能受到特殊原因影响，或者可能是一个缺陷。
- 控制限（或是 CUSUM 图的 V 形模板）之外的点指示存在特殊原因变异。

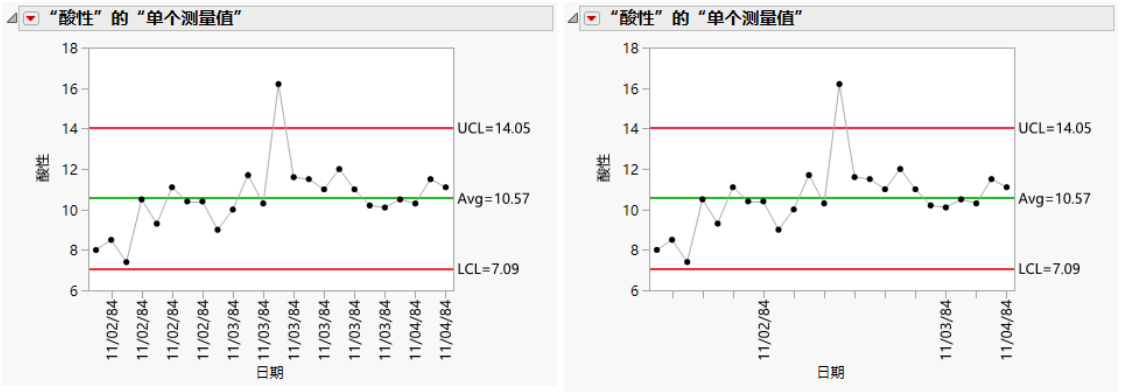
使用每个平台中的选项可以创建控制图，使这些控制图随着接收和记录样本或是随着向数据表添加样本而动态更新。

当控制图指示异常变异时，应采取措施使过程恢复统计控制状态。若异常变异指示过程有所改善，则应对变异原因进行研究和实践。

双击水平或垂直轴时会显示相应的“轴规格”窗口，供您指定格式、坐标轴值、刻度数、网格线、参考线和其他要在轴上显示的选项。

例如，Pickles.jmp 数据表列出了三天每天进行抽样的测量值。在图 12.8 中，默认情况下，水平轴上每隔一个刻度就显示一个标签。有时这会让标签重复出现，如图 12.8 中的左图所示。若指定标签使用 8 作为增量，水平轴会每天标注一次，如右图所示。

图 12.8 示例：添加了标签的 X 轴刻度标



提示：有关警告和规则的信息，请参见本指南中的第 50 页的“检验”和“控制图生成器”一章中第 53 页的“Westgard 规则”。

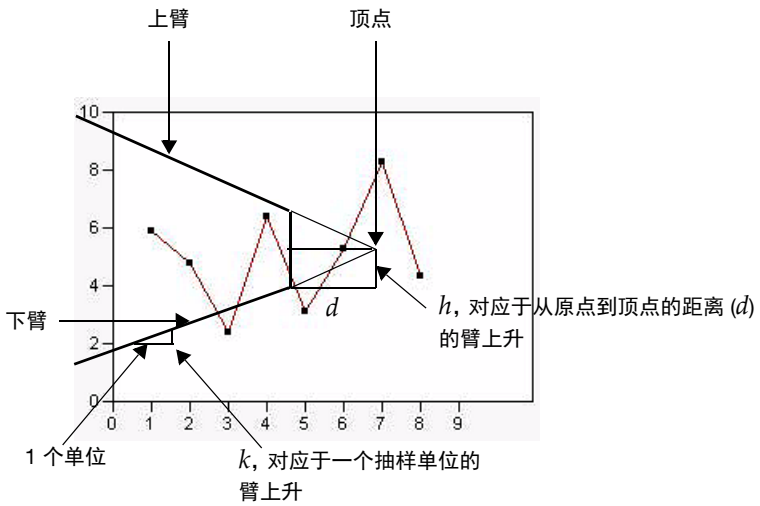
V 形模板 CUSUM 图报表

解释双侧 V 形模板 CUSUM 图

注意：另见第 312 页的“V 形模板 CUSUM 图示例”。

要解释双侧 CUSUM 图，请将各点与构成 V 形模板的限值进行比较。V 形模板是叠加在累积和图上的侧边呈 V 字的形状。V 形模板是通过标绘 V 形限值形成的。V 形模板的原点是最近标绘的点，其双臂沿着水平轴向后延展，如图 12.9 所示。随着不断收集数据，累积和序列随之更新，原点重新定位在最新点。

图 12.9 双侧 CUSUM 图的 V 形模板



由于过程均值中的偏移会使标绘点的斜率产生变化，所以在 CUSUM 图上很容易检测出这种偏移。斜率变化位置处的点是发生偏移的点。若以前标绘的一个或多个点穿过了 V 形模板的上臂或下臂，情况就会**失控**。穿过下臂的点指示过程均值增大，穿过上臂的点指示向下偏移。

CUSUM 图与 Shewhart 图之间有着重要的差异：

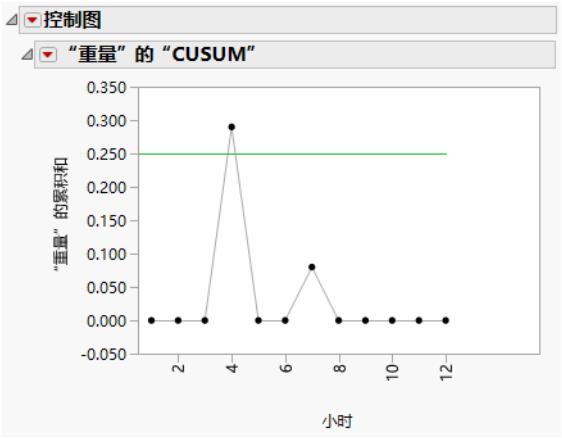
- Shewhart 控制图基于单个子组样本的信息标绘各点。在 CUSUM 图中，每个点都基于来自选取到当前子组（并包含当前子组）的所有样本的信息。
- 在 Shewhart 控制图上，水平控制限定义某个点是否指示失控状况。在 CUSUM 图上，限值既可采用 V 形模板的形式也可采用水平决策区间的形式。
- Shewhart 控制图上的控制限通常指定为 3σ 限值。在 CUSUM 图上，从平均运行链长确定限值。

CUSUM 图对于检测过程均值中的小偏移更加高效。Lucas (1976) 指出 V 形模板检测 1σ 偏移的速度比 Shewhart 控制图快三倍。

解释单侧 CUSUM 图

使用单侧 CUSUM 图可标识接近或超出关注一侧的数据。

图 12.10 单侧 CUSUM 图的示例



决策区间或水平线在您于启动窗口中输入的 H 值处设置。在本例中，决策区间为 0.25。若有任何值超过决策区间 0.25，则表示存在偏移或失控状况。在本例中，观测 4 看起来是发生偏移的位置。还请注意，单侧 CUSUM 图不显示 V 形模板。

“传统控制图”平台选项

传统控制图包含可影响平台各组成部分的红色小三角菜单：

- 位于最顶部的标题栏上的菜单影响整个平台窗口。其中的项根据您选择的图表类型而变化。请参见第 301 页的“传统控制图的窗口选项”。
- 图表类型标题栏上有一个菜单，其中的选项单独影响各个图。请参见第 304 页的“传统控制图的图形选项”。

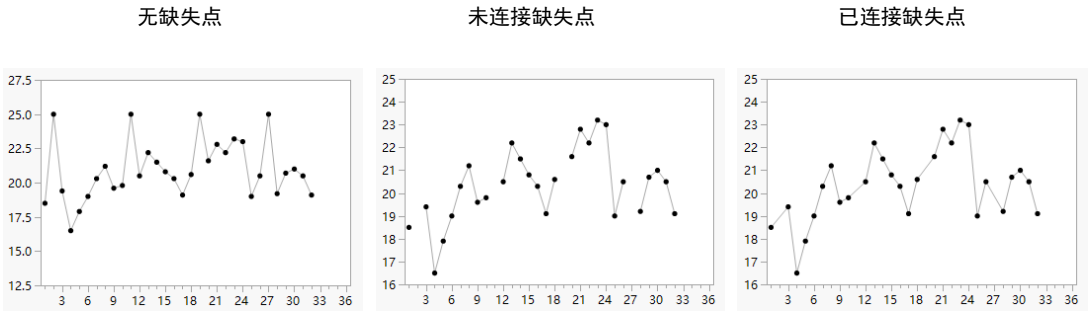
传统控制图的窗口选项

窗口标题栏上的红色小三角菜单列出影响报表窗口的选项。若同时请求**均值**和 **R**，您可以选中每种图形类型以便显示或隐藏它。可用的特定选项取决于您请求的控制图类型。不可用选项显示为灰色的菜单项。

显示限值图例 显示或隐藏图右侧的平均值、上控制限值和下控制限值。

忽略缺失值进行连接 在某些样本包含缺失值时连接各点。在图 12.11 中，左图没有缺失点。中间图样本 2、11、19 和 27 缺失，相应地也未连接这些点。若您选择了**忽略缺失值进行连接**选项 （默认设置），则会显示右图。

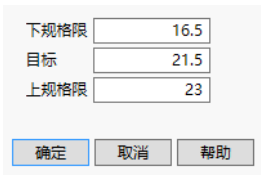
图 12.11 “忽略缺失值进行连接”选项的示例



使用中位数 对于运行图，若您在单个运行图的红色小三角菜单中选择**显示中心线**选项，则会绘制一条穿过列中心值的线。中心线由主运行图红色小三角菜单中的**使用中位数**设置决定。若选定**使用中位数**，则中位数将用作中心线。否则使用均值。将限值保存至文件时，总均值和中位数都将保存。

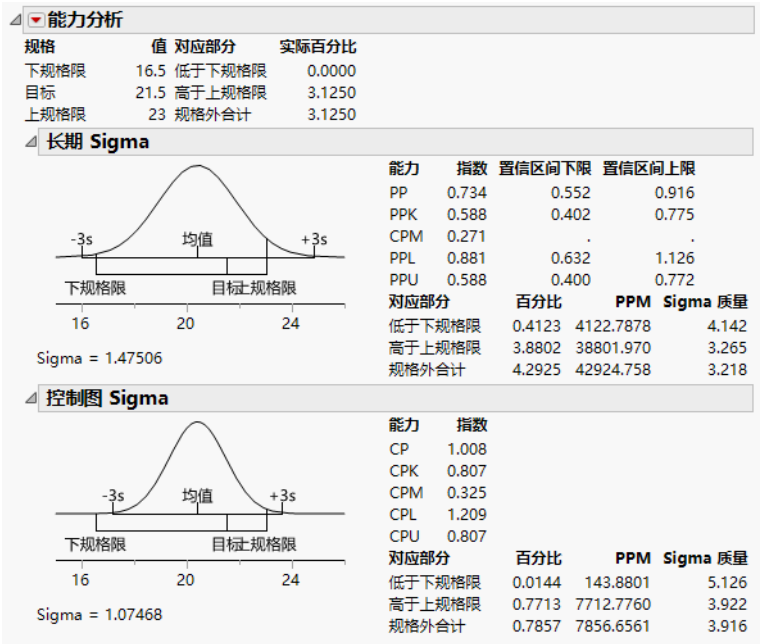
能力 （指定阶段变量时不可用。）对数据执行“能力分析”。首先会显示一个弹出窗口，您可在其中输入过程变量的下规格限、目标和上规格限的值。

图 12.12 “能力分析”窗口



Coating.jmp 的“能力分析”报表示例显示在图 12.13 中，其中的下规格限设置为 16.5、目标设置为 21.5，上规格限设置为 23。

图 12.13 Coating.jmp 的“能力分析”报表



更多信息，请参见第 316 页的“能力分析的统计详细信息”。

保存 Sigma 将 sigma 的计算值另存为 JMP 数据表的过程变量列中的列属性。

保存限值 通过以下方式之一保存控制限：

在列中 将控制限另存为现有数据表中响应变量的列属性。若限值为常数，则保存报表中每种图表类型的下控制限值、平均值和上控制限值。该选项不可用于阶段图。此外，若样本大小对于每个图表并不一致，则该选项无效。

在新表中 将每个图表的标准差和均值保存到新的数据表中。若限值为常数，则同时保存每个图表的下控制限值、平均值和上控制限值。若存在阶段，则为每个阶段保存一组新值。您可以保存该数据表以便将来使用这些限值。在“控制图”启动窗口中，点击**获取限值**，然后选择保存的数据表。请参见第 306 页的“保存和检索限值”一节。

保存汇总 创建包含样本标签、样本大小、标绘的统计量、中心线和控制限的新数据表。包含在表中的特定统计量取决于图表类型。

警告脚本 支持您编写和运行指示数据未通过特殊原因检验的脚本。结果可写入日志或语音播报。请参见本指南的“控制图生成器”一章中第 50 页的“检验”。请参见《Scripting Guide》，了解关于编写定制警告脚本的详细信息。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

传统控制图的图形选项

当您点击图表名称旁边的图标时，图表选项的红色小三角菜单随即显示。某些选项也在您右击图表时显示在**图表选项**中。并非以下所有选项都适用于所有控制图类型。

箱线图 在均值图中标绘的子组均值之上叠加箱线图。箱线图显示子组的最大值、最小值、第 75 百分位数、第 25 百分位数和中位数。除非您取消选择**显示点**选项，否则会显示子组均值的标记。显示的控制限仅适用于子组均值。**箱线图**选项仅可用于 $\bar{X}$ 图。该图最适合较大的子组样本 （子组中有超过 10 个样本）。

针 使用垂直线段将标绘的各点与中心线连接。

连接点 显示或隐藏连接数据点的线。

显示点 显示或隐藏表示汇总统计量的点。最初会显示这些点。若**箱线图**选项有被选用，您就可以使用该选项隐藏表示子组均值的标记。

连接线颜色 显示 JMP 调色板，供您选择用于连接各点的线段的颜色。

中心线颜色 显示 JMP 调色板，供您选择用于绘制中心线的线段的颜色。

限值颜色 显示 JMP 调色板，供您选择用于上下限值线的线段的颜色。

线条粗细 支持您选择控制线的宽度。选项包括**细**、**中**或**粗**。

点标记 支持您选择在图上使用的标记。

显示中心线 显示或隐藏控制图中的中心线。

显示控制限 显示或隐藏图的控制限及其图例。

限值精度 设置标签的小数限值。

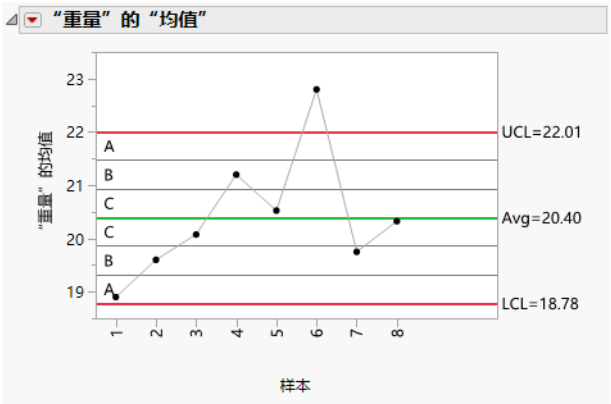
检验 显示一个子菜单，支持您在检验为阳性时选择要在图上标记的检验。检验仅适用于控制限为 3σ 限值的图。检验 1 到 4 适用于均值图、单值图和计数图。检验 5 到 8 仅适用于均值图、预先汇总图和单个测量值图。若检验不适用于某个图，“检验”选项将灰显。若样本大小不等，“检验”选项将灰显。若样本在图处于打开状态时发生了变化，各样本大小变得相等，而且已选定区域或检验选项，那么将立即应用区域或检验并显示在图上。这些特殊检验亦称 Western Electric **规则**。有关特殊原因检验的详细信息，请参见“控制图生成器”一章中第 50 页的 “检验”。

Westgard 规则 Westgard 规则是可以帮助您确定过程是否受控的控制规则。不同的检验缩写为针对特定检验的决策规则。请参见“控制图生成器”一章中第 53 页的 “Westgard 规则”中的文本和图表。

检验超出界限 将超出限值的所有点标记为 “*”。该检验适用于带有限值的所有图，不论样本大小是否不变，也不论 *k* 的大小或是限值宽度如何。例如，若您具有不等样本大小，并且要标记 R 图中超出限值的所有点，您可以使用该命令。

显示区域 显示或隐藏**区域线**。这些区域标记为 A、B 和 C，如下面的 Coating.jmp 样本数据中的**重量**均值图所示。控制图检验将区域线用作边界。这七条区域线彼此相隔一个 sigma 的距离，在中心线两侧对称分布。

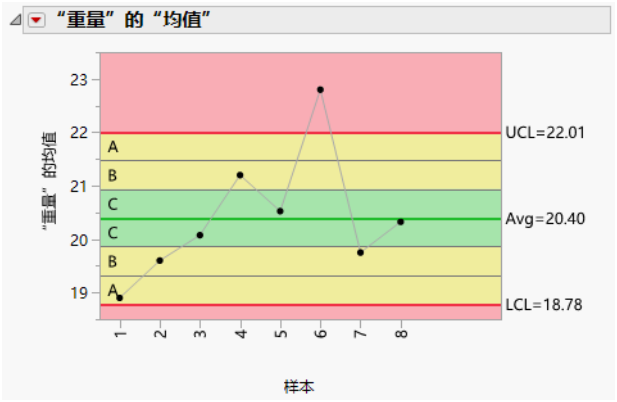
图 12.14 显示区域



区域着色 为三个区域范围和这些区域之外的范围显示或隐藏默认的绿色、黄色和红色。绿色代表距离中心线一个 sigma 远的区域，黄色代表距离中心线两个和三个 sigma 远的区域，红色代表超出三个 sigma 远的区域。着色区域可显示出或不显示出区域线。

提示：要更改用于对分区进行着色的颜色，请在控制图中右击并选择 “定制”。在 “定制图形” 窗口中，可以为三个区域中的每个区域指定颜色。

图 12.15 区域着色



OC 曲线 打开一个包含操作特征 (OC) 曲线的新窗口，使用直接来自活动控制图的所有计算所得值。请参见 “质量实用工具”一章中第 361 页的 “‘操作特征曲线’实用工具”。

V 形模板 CUSUM 控制图的图表选项

V 形模板颜色 （仅当选定“显示 V 形模板”选项时可用。）支持您选择 V 形模板的线条颜色。

显示偏移 显示或隐藏您在启动窗口中输入的偏移。

显示 V 形模板 根据您在 “CUSUM 控制图”启动窗口中指定的统计量显示或隐藏 V 形模板。

显示参数 显示或隐藏汇总 CUSUM 绘图参数的报表。

显示 ARL 显示或隐藏平均运行链长 (ARL) 信息。平均运行链长是在指示失控状况之前所抽取的期望样本个数：

- ARL (Delta)，有时表示为 ARL1，是用于检测指定 Delta 大小的偏移的平均运行链长。
- ARL(0)，有时表示为 ARL0，是指定参数的受控平均运行链长 (Montgomery 2013)。

保存和检索限值

JMP 可对控制图使用以前设定的控制限：

- 上控制限和下控制限，以及中心线值。
- 用于计算限值的参数（比如均值和标准差等）。

控制限或限值参数值必须位于 JMP 数据表（称为**限值表**）中，或者作为列属性存储在过程列中。指定**控制图**命令后，您可以使用控制图启动窗口中的**获取限值**按钮检索“限值表”。

提示：要一次性向若干列添加规格限，请参见“质量实用工具”一章中第 357 页的 “‘管理规格限’实用工具”。

最简单的“限值表”创建方法是保存“控制图”平台计算的结果。每个控制图对应的红色小三角菜单中的**保存限值**命令可自动保存样本值中的限值。表中保存的数据类型随着分析窗口中的控制图类型而变化。您还可以使用来自任意源的值并自行创建“限值表”。

所有 “限值表”都必须具有：

- 用于标识每行的特殊关键字列。
- 每个变量有一列，这些变量的值为已知的标准参数或限值。该列名必须与“控制图”平台要分析的数据表中的相应过程变量名相同。

下表说明了针对传统控制图和用“控制图生成器”创建的图的限值关键字及其关联的控制图。

表 12.1 限值表关键字及相应图表和含义

关键字	适用图表	含义
_KSigma	全部，“控制图生成器”和“V 形模板 CUSUM”除外	使用统计量标准差倍数计算控制限；若限值用 alpha 水平表示，则设置为缺失
_Alpha	全部，控制图生成器除外	用于计算控制限的 I 类误差概率
_Range Span	IM、MR、MMR	用于计算移动极差的连续测量值数。不适用于“控制图生成器”平台，在该平台中极差跨度始终等于 2。
_Sample Size	全部，Levey-Jennings 和预先汇总除外	子组大小
_Std Dev	均值、R、S、IM、MR、G、T、V 形模板 CUSUM 和 Levey-Jennings	已知过程标准差
_U	C、U	已知每单位不合格事件的平均数
_P	NP、P	已知平均不良率值
_LCL、_UCL	均值、IM、P、NP、C、U、G、T 和 Levey-Jennings	均值图、单个测量值图、任何计数图或稀有事件图的下控制限和上控制限
_AvgR	R、MR	平均极差或平均移动极差
_LCLR、_UCLR	R、MR	R 图或 MR 图的下控制限 R 图或 MR 图的上控制限
_AvgS、_LCLS、_UCLS	S 图	S 图的平均标准差、上控制限和下控制限

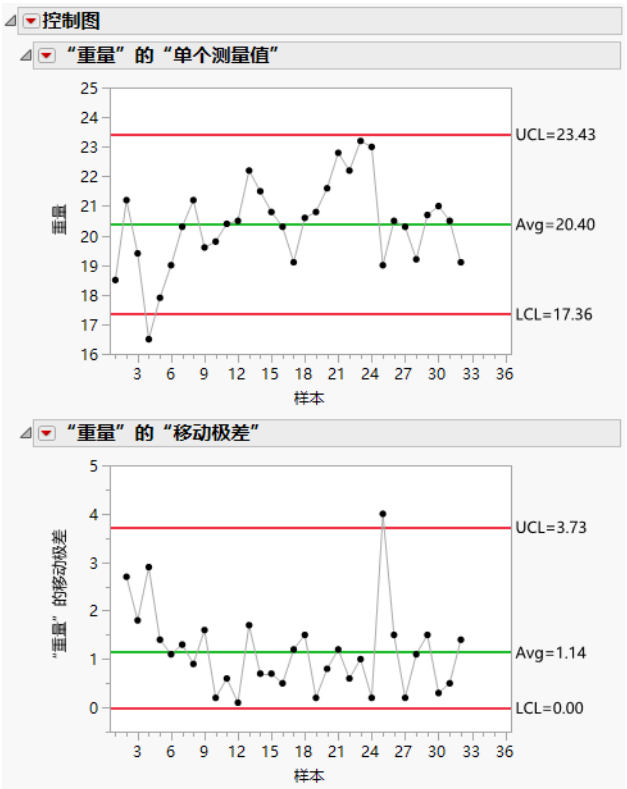
表 12.1 限值表关键字及相应图表和含义 (续)

关键字	适用图表	含义
_AvgR_PreMeans _AvgR_PreStdDev _LCLR_PreMeans _LCLR_PreStdDev _UCLR_PreMeans _UCLR_PreStdDev _Avg_PreMeans _Avg_PreStdDev _LCL_PreMeans _LCL_PreStdDev _UCL_PreMeans _UCL_PreStdDev	IM、MR	基于预先汇总组均值或标准差的均值、上控制限和下控制限。
_Data Units _Two Sided _Headstart _Beta _Delta	V 形模板 CUSUM	V 形模板 CUSUM 图的规格

您可以将限值保存在新数据表中或另存为响应列的属性。若使用**在新表中**命令保存控制限，写入该表的限值关键字取决于当前显示的图表类型。

图 12.16 显示了使用 Coating.jmp 将控制限保存至数据表的示例。包含值 _Mean、_LCL 和 _UCL 的行用于单个测量值图。带有 R 后缀的值（_AvgR、_LCLR 和 _UCLR）用于移动极差图。若使用该“限值表”再次创建这些图，“控制图”平台会通过 _LimitsKey 列中的关键字识别合适的限值。

图 12.16 在数据表中保存限值的示例



	_LimitsKey	重量
1	_KSigma	3
2	_Alpha	0.0026997961
3	_Range Span	2
4	_Std Dev	1.0120139729
5	_Mean	20.396875
6	_LCL	17.360833081
7	_UCL	23.432916919
8	_AvgR	1.1419354839
9	_LCLR	0
10	_UCLR	3.7301687077

请注意：可以在控制图启动窗口中指定 _KSigma、_Alpha 和 _Range Span 的值。JMP 始终最先查看该窗口中的值。在该窗口中指定的值优先于在当前“限值表”中指定的值。

含有未知关键字的行和使用已排除行状态标记的行将被忽略。除 _Range Span、_KSigma、_Alpha 和 _Sample Size 之外，未指定的任何所需值都从数据中估计得出。

排除、隐藏和删除的样本

下表总结了不同条件对样本和子组的影响：

表 12.2 排除、隐藏和删除的样本

创建图之前排除所有样本行。	样本不包含在限值计算中，但显示在图中。
创建图之后排除样本。	样本包含在限值计算中，并且显示在图中。若在图处于打开状态时排除样本，输出不会发生任何变化。
创建图之前隐藏样本。	样本包含在限值计算中，但不显示在图中。

表 12.2 排除、隐藏和删除的样本 (续)

创建图之后隐藏样本。	样本包含在限值计算中，但不显示在图中。样本标记从图中消失，样本标签仍显示在轴上，但限值保持不变。
创建图之前排除并隐藏所有样本行。	样本不包含在限值计算中，并且不显示在图中。
创建图之后排除并隐藏所有样本行。	样本包含在限值计算中，但不显示在图中。样本标记从图中消失，样本标签仍显示在轴上，但限值保持不变。
创建图之前在删除样本的情况下对数据集取子集。	样本不包含在限值计算中，轴不包含样本值，并且样本标记不显示在图中。
创建图之后在删除样本的情况下对数据集取子集。	样本不包含在限值计算中，并且不显示在图中。样本标记从图中消失，样本标签从轴上移除，图发生偏移，并且限值发生变化。

补充说明：

- 隐藏和排除操作仅对样本中第一个观测的行状态有效。例如，若样本中的第二个观测隐藏，而第一个观测未隐藏，则样本仍会显示在图上。

  注意：预先汇总图中排除的行将从计算中排除，而不管它们在样本中处于哪个位置。

- 排除 / 隐藏规则的一个例外：隐藏和排除的行均包括在针对特殊原因检验的点计数中。排除的行可使用特殊原因旗标来标记。无法标记隐藏点。若特殊原因检验的旗标位于隐藏点上，则该旗标不会显示在图中。
- 由于存在特定规则（第 309 页上的表 12.2），控制图不支持“自动重新计算”脚本。

“控制图”平台的更多示例

- “预先汇总图示例”
- “V 形模板 CUSUM 图示例”
- “单侧 CUSUM 图示例”
- “UWMA 图示例”

预先汇总图示例

本例使用 Coating.jmp 数据表。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Coating.jmp。
2. 选择**分析** > **质量和过程** > **传统控制图** > **预先汇总**。
3. 选择**重量**并点击**过程**。

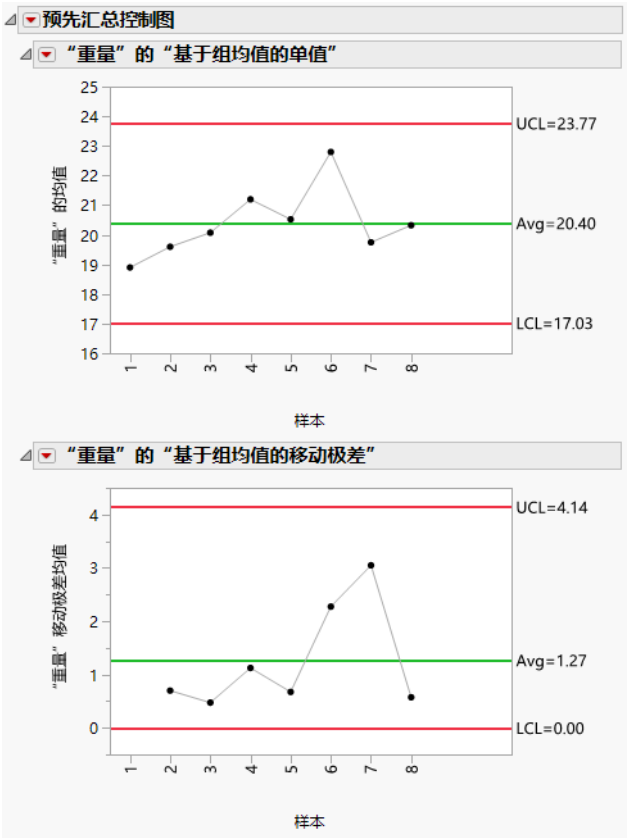
4. 选择**样本**并点击**样本标签**。
5. 同时选择**基于组均值的单值**和**基于组均值的移动极差**。在您选择“样本标签”变量后，将自动选定**按样本标签对样本分组**按钮。

   使用**预先汇总**图时，您可以选择**基于组均值**选项和 / 或**基于组标准差**选项。若同时选择两种单值极差图表类型，每个选项都会创建两个图（一个单个测量值图，亦称均值图，以及一个移动极差图）。

   基于组均值选项计算每个样本的均值，然后绘制均值并基于该均值创建单个测量值和移动极差图。

   基于组标准差选项计算每个样本的标准差，并将标准差标绘为单个点。针对标准差的单个测量值和移动极差图随即显示。
6. 点击**确定**。

图 12.17 预先汇总数据绘制示例



尽管均值图和 S 图上所用点与基于组均值的单值图和基于组标准差的单值图的所用点相同，但限值不同，因为这些限值按单值图计算。

使用 Coating.jmp 数据表还可以通过另一种方式生成预先汇总图：

1. 选择**表** > **汇总**。
2. 选择**样本**并点击**组**。
3. 选择**权重**，然后点击**统计量** > **均值**和**统计量** > **标准差**。
4. 点击**确定**。
5. 选择**分析** > **质量和过程** > **传统控制图** > **单值极差**。
6. 选择**均值 （重量）**和**标准差 （重量）**，然后点击**过程**。
7. 点击**确定**。

生成的图和预先汇总图一致。

V 形模板 CUSUM 图示例

利用一种机器来装填二冲程机油添加剂到 8 盎司油罐。装填过程被视为处于统计控制中。该过程经过设置，满罐的平均重量 (μ_0) 为 8.10 盎司。之前的分析显示装填重量的标准差 (σ_0) 为 0.05 盎司。

选取了四个油罐的子组样本，每小时称重一次，连续称了 12 小时。 Oil1 Cusum.jmp 数据表中的每个观测都包含一个重量值及其关联的小时值。观测经过排序，小时值采用升序排序。

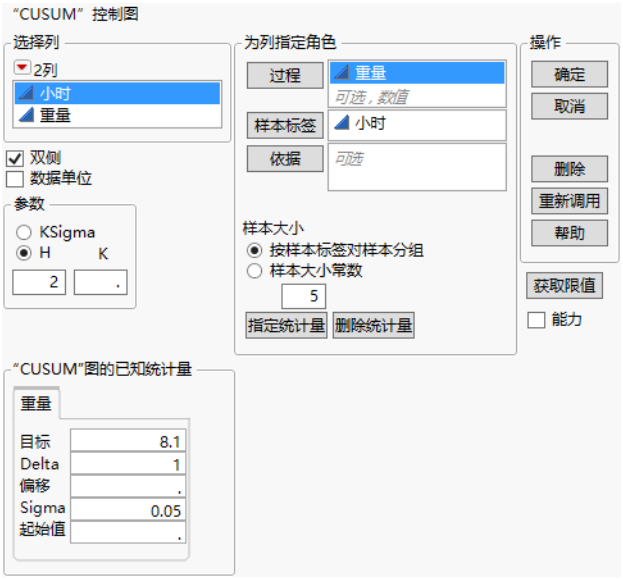
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Oil1 Cusum.jmp。
2. 选择**分析** > **质量和过程** > **传统控制图** > CUSUM。
3. 选择**重量**并点击**过程**。
4. 选择**小时**并点击**样本标签**。
5. 选中**双侧**复选框 （若尚未选中）。
6. 在 “参数”区域中，点击 H 按钮并键入 2。
7. 点击**指定统计量**。
8. 在**目标**旁边键入 8.1。

   8.1 是满罐的平均重量 （单位为盎司）。这是目标均值。
9. 在 Delta 旁边键入 1。

   1 是要检测的最小偏移的绝对值，该值作为过程标准差或标准误差的倍数。
10. 在 Sigma 旁边键入 0.05。

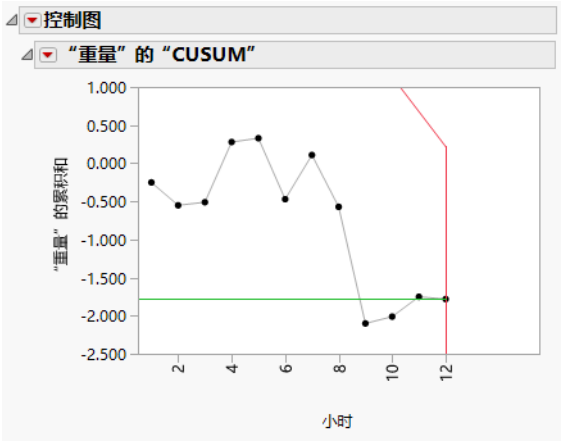
    0.05 是以盎司为单位的装填重量的已知标准差 (σ_0)。

图 12.18 已完成的启动窗口



11. 点击**确定**。

图 12.19 Oil1 Cusum.jmp 数据的双侧 CUSUM 图



您可以通过将点与 V 形模板加以比较来解释该图。V 形模板的右边缘以最近一个点（第 12 个小时）为中心。由于没有任何点穿越 V 形模板的两臂，所以没有证据表明过程中发生偏移。请参见第 299 页的“V 形模板 CUSUM 图报表”。

单侧 CUSUM 图示例

请考虑第 312 页的 “V 形模板 CUSUM 图示例”中使用的数据，其中使用一种机器来装填 8 盎司机油罐。为了削减成本，生产商现在更关注显著的超量装填情况 （而不太关注装填不满的情况）。使用单侧 CUSUM 图可标识所有超量装填情况。超过均值 (8.1) 0.25 盎司的所有情况均被视为有问题。

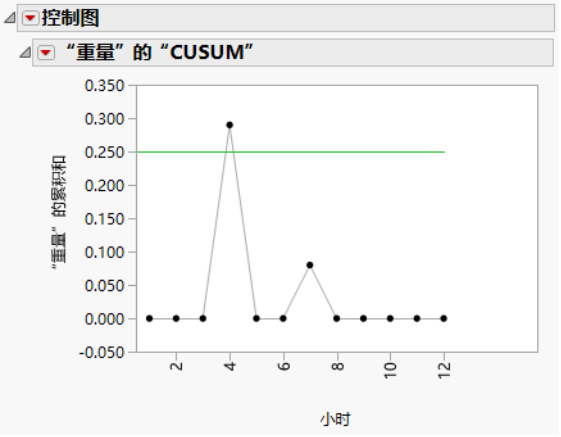
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Oil1 Cusum.jmp。
2. 选择**分析** > **质量和过程** > **传统控制图** > CUSUM。
3. 取消选择**双侧**。
4. 选择**重量**并点击**过程**。
5. 选择**小时**并点击**样本标签**。
6. 点击 H 并键入 0.25。
7. 点击**指定统计量**。
8. 在**目标**旁边键入 8.1。

   8.1 是满罐的平均重量 （单位为盎司）。这是目标均值。
9. 在 Delta 旁边键入 1。

   1 是要检测的最小偏移的绝对值，该值作为过程标准差或标准误差的倍数。
10. 在 Sigma 旁边键入 0.05。

    0.05 是以盎司为单位的装填重量的已知标准差 (σ_0)。
11. 点击**确定**。

图 12.20 Oil1 Cusum.jmp 数据的单侧 CUSUM 图



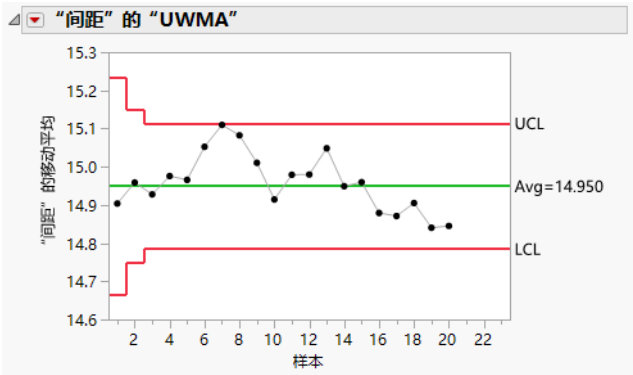
决策区间设置在您输入的 H 值 (0.25) 的位置。您可以看到，在第四个小时处，出现了显著的超量装填情况。

UWMA 图示例

在样本数据表 Clips1.jmp 中，关注的测量是生产的金属夹两端之间的间距。要监控平均间距的变化过程，每天选取五个金属夹一组的子组样本。将检查跨度为 3 的移动平均的 UWMA 图。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Clips1.jmp。
2. 选择**分析** > **质量和过程** > **传统控制图** > UWMA。
3. 选择**间距**并点击**过程**。
4. 选择**样本**并点击**样本标签**。
5. 将**移动平均跨度**改为 3。
6. 点击**确定**。

图 12.21 Clips1 数据的 UWMA 图



第一天的点是该日五个子组样本值的均值。第二天的标绘点是第一天与第二天的子组样本均值的平均值。其余各天的点是每一天与之前两天的子样本均值的平均值。

金属夹平均间距看起来在下降，但没有任何样本点落在 3σ 限值之外。

“控制图”平台的统计详细信息

- “中位数移动极差图的控制限”
- “能力分析的统计详细信息”
- “V 形模板 CUSUM 控制图的统计详细信息”
- “加权移动平均图的统计详细信息”

注意：有关其他类型的图（如均值图和 R 图、P 和 NP 图等）的详细信息，请参见“控制图生成器”一章中第 82 页的“ “控制图生成器”的统计详细信息”。

中位数移动极差图的控制限

中位数移动极差图的控制限计算如下：

下控制限 $_{MMR} = \max(0, MMR - kd_3(n)\hat{\sigma})$

上控制限 $_{MMR} = MMR + kd_3(n)\hat{\sigma}$

其中：

MMR 是非缺失移动极差的中位数。

$\hat{\sigma} = MMR/0.954$

$d_3(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的标准差

能力分析的统计详细信息

本节包含有关“能力分析”报表中统计信息的计算的详细信息。

变异统计量

所有能力分析都使用相同的公式。选项之间的区别在于如何计算 sigma (σ)：

长期 Sigma 使用总 Sigma。该选项用于 P_{pk} 统计量， sigma 计算如下：

$$\hat{\sigma} = \sqrt{\sum_{i=1}^{n} \frac{(x_i - \bar{x})^2}{n-1}}$$

注意：默认情况下，“长期 Sigma”报表中的能力指标使用其他 Sigma 报表中使用的 Cp 标签。要使用“长期 Sigma”报表中的 Ppk 标签，请选择**文件 > 首选项 > 平台 > 分布 > 添加 PpK 能力标签**首选项。

控制图 Sigma 使用由控制图设置确定的 sigma。

- 若使用控制启动窗口中的“指定统计量”按钮指定 Sigma 的值，则指定的值将用于计算能力指标。
- 在使用移动极差 （平均值）选项的单值极差图中， sigma 的值计算如下：

$$\hat{\sigma} = \frac{\bar{R}}{d_2(n)}$$

其中：

$\bar{R}$ 是移动极差的平均值。

$d_2(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的期望值，其中 n 是 “极差跨度”选项的值。

– 在使用移动极差 （平均值）选项的单值极差图中， sigma 的值计算如下：

$$\hat{\sigma} = \frac{MMR}{d_4(n)}$$

其中：

MMR 是非缺失移动极差的中位数。

$d_4(n)$ 是服从单位标准差正态分布的 n 个自变量的极差的中位数，其中 n 是 “极差跨度”选项的值。

– 在使用 R 选项的均值图中， sigma 的值计算如下：

$$\hat{\sigma} = \frac{\frac{R_1}{d_2(n_1)} + \ldots + \frac{R_N}{d_2(n_N)}}{N}$$

其中：

R_i = 第 i 个子组的极差

n_i = 第 i 个子组的样本大小

$d_2(n_i)$ = 服从单位标准差正态分布的 n_i 个自变量的极差期望值

$N = n_i \geq 2$ 的子组数

– 在使用 S 选项的均值图中， sigma 的值计算如下：

$$\hat{\sigma} = \frac{\frac{s_1}{c_4(n_1)} + \ldots + \frac{s_N}{c_4(n_N)}}{N}$$

其中：

n_i = 第 i 个子组的样本大小

$c_4(n_i)$ = 服从单位标准差正态分布的 n_i 个自变量的标准差期望值

$N = n_i \geq 2$ 的子组数

s_i = 第 i 个子组的样本标准差

正态分布的能力指标

本节提供有关正态数据的能力指标计算的详细信息。

对于具有均值 μ 和标准差 σ 的过程特征，基于总体的能力指标定义如下：

$$\text{Cp} = \frac{USL - LSL}{6\sigma}$$

$$\text{Cpl} = \frac{\mu - LSL}{3\sigma}$$

$$Cpu = \frac{USL - \mu}{3\sigma}$$

$$Cpk = \min(Cpl, Cpu)$$

$$Cpm = \frac{\min(T - LSL, USL - T)}{3\sigma\sqrt{1 + \left(\frac{T - \mu}{\sigma}\right)^2}}$$

其中：

LSL 是下规格限。

USL 是上规格限。

T 是目标值。

对于基于样本的能力指标，参数由其估计值替代。σ 的估计值使用您在“能力分析”窗口中指定的方法。请参见第 316 页的“变异统计量”。

若任一规格限缺失，包含缺失规格限的能力指标将报告为缺失。

提示：能力指标 1.33 通常被视为可以接受的最小值。对于正态分布，能力指标 1.33 对应的是不合格品预期数为每 100,000 中有 6 个。

能力指标的置信区间

注意：能力指标的置信区间仅显示在“长期 Sigma”报表中。

Cp 的 100(1 - α)% 置信区间计算如下：

$$\left(\hat{C}P\sqrt{\frac{\chi^2_{\alpha/2,\,n-1}}{n-1}},\ \hat{C}P\sqrt{\frac{\chi^2_{1-\alpha/2,\,n-1}}{n-1}}\right)$$

其中：

$\hat{C}P$ 是 Cp 的估计值。

$\chi^2_{\alpha/2,\,n-1}$ 是具有 *n* - 1 个自由度的卡方分布的第 (α/2) 分位数。

n 是观测数。

Cpk 的 100(1 - α)% 置信区间计算如下：

$$\left(\hat{C}_{pk}\left[1 - \Phi^{-1}{}_{1-\alpha/2}\sqrt{\frac{1}{9n\hat{C}^2_{pk}} + \frac{1}{2(n-1)}}\right],\ \hat{C}_{pk}\left[1 + \Phi^{-1}{}_{1-\alpha/2}\sqrt{\frac{1}{9n\hat{C}^2_{pk}} + \frac{1}{2(n-1)}}\right]\right)$$

其中：

$\hat{C}_{pk}$ 是 Cpk 的估计值。

$\Phi^{-1}{}_{1-\alpha/2}$ 是标准正态分布的第 (1 - α/2) 分位数。

n 是观测数。

CPM 的 100(1 - α)% 置信区间计算如下：

$$\left(\hat{\text{CPM}} \sqrt{\frac{\chi^2_{\alpha/2,\,\gamma}}{\gamma}},\ \hat{\text{CPM}} \sqrt{\frac{\chi^2_{1-\alpha/2,\,\gamma}}{\gamma}} \right)$$

其中：

$\hat{\text{CPM}}$ 是 CPM 的估计值。

$\chi^2_{\alpha/2,\,\gamma}$ 是具有 γ 个自由度的卡方分布的第 (α/2) 分位数。

$$\gamma = \frac{n\left(1 + \left(\frac{\bar{x} - T}{s}\right)^2\right)^2}{1 + 2\left(\frac{\bar{x} - T}{s}\right)^2}$$

n 是观测数。

$\bar{x}$ 是观测均值。

T 是目标值。

s 是长期 Sigma 估计值。

注意：仅当目标值位于下规格限和上规格限的中心时才计算 Cpm 的置信区间。

CPL 和 CPU 的上下置信限使用 Chou et al. (1990) 有关 T 统计量的说明。

CPL 的 100(1 - α)% 置信限 （用 CPL_L 和 CPL_U 表示）满足以下方程：

$$\Pr[t_{n-1}(\delta_L) \geq 3\hat{C}_{pl}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_L = 3CPL_L\sqrt{n}$$

$$\Pr[t_{n-1}(\delta_U) \leq 3\hat{C}_{pl}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_U = 3CPL_U\sqrt{n}$$

其中：

$t_{n\text{-}1}(\delta)$ 是具有 n - 1 个自由度和非中心参数 δ 的非中心 t 分布。

$\hat{C}_{pl}$ 是 Cpl 的估计值。

CPU 的 100(1 - α)% 置信限 （用 CPL_L 和 CPL_U 表示）满足以下方程：

$$\Pr[t_{n-1}(\delta_L) \geq 3\hat{C}_{pu}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_L = 3CPU_L\sqrt{n}$$

$$\Pr[t_{n-1}(\delta_U) \leq 3\hat{C}_{pu}\sqrt{n}] = \alpha/2 \qquad \text{其中} \quad \delta_U = 3CPU_U\sqrt{n}$$

其中：

$t_{n-1}(\delta)$ 是具有 n - 1 个自由度和非中心参数 δ 的非中心 t 分布。

$\hat{C}_{pu}$ 是 Cpu 的估计值。

非正态分布的能力指标

本节说明如何为非正态分布计算能力指标。这些广义能力指标定义如下：

$$\text{Cp} = \frac{USL - LSL}{P_{0.99865} - P_{0.00135}}$$

$$\text{Cpk} = \min(C_{pl}, C_{pu})$$

$$\text{Cpm} = \frac{\min\left(\frac{T - LSL}{P_{0.5} - P_{0.00135}}, \frac{USL - T}{P_{0.99865} - P_{0.5}}\right)}{\sqrt{1 + \left(\frac{\mu - T}{\sigma}\right)^2}}$$

$$\text{Cpl} = \frac{P_{0.5} - LSL}{P_{0.5} - P_{0.00135}}$$

$$\text{Cpu} = \frac{USL - P_{0.5}}{P_{0.99865} - P_{0.5}}$$

其中：

LSL 是下规格限。

USL 是上规格限。

T 是目标值。

P_α 是拟合分布的第 α*100 百分位数。

对于 Cpm 的计算，使用拟合分布的方差的预期值和平方根来估计 μ 和 σ。有关“参数估计值”报表中的参数与拟合分布的预期值和方差之间关系的详细信息，请参见《基本分析》。

Sigma 质量统计量

每个部分 （低于下规格限、高于上规格限和规格外合计）的 Sigma 质量统计量计算如下：

$$\text{Sigma 质量} = \Phi^{-1}{}_{1 - \text{Pct}/100} + 1.5$$

其中：

Pct 是报表中 “百分比”列中的值。

$\Phi^{-1}{}_{1-\mathrm{Pct}/100}$ 是标准正态分布的第 (1 - Pct/100) 分位数。

注意：尽管“低于下规格限的百分比”和“高于上规格限的百分比”加总为“规格外合计百分比”值，但“Sigma 质量低于下规格限”值和“Sigma 质量高于上规格限”值的计算结果之和不等于“Sigma 质量规格外合计”值。这是因为计算 Sigma 质量涉及查找正态分布分位数，因此不是相加得到的。

基准 Z 统计量

基准 Z 统计量仅适用于基于正态分布的能力分析。基准 Z 统计量计算如下：

$$\text{Z 基准} = \Phi^{-1}{}_{1-P(LSL)-P(USL)}$$

$$\text{Z 下规格限} = \frac{\mu - LSL}{\sigma} = 3 * \text{Cpl}$$

$$\text{Z 上规格限} = \frac{USL - \mu}{\sigma} = 3 * \text{Cpu}$$

其中：

LSL 是下规格限。

USL 是上规格限。

μ 是样本均值。

σ 是样本标准差。

$\Phi^{-1}{}_{1-P(LSL)-P(USL)}$ 是标准正态分布的第 (1 - *P(LSL)* -*P(USL)*) 分位数。

P(LSL) = 概率 ($X < LSL$) = 1 - Φ(Z 下规格限)。

P(USL) = 概率 ($X > USL$) = 1 - Φ(Z 上规格限)。

Φ 是标准正态累积分布函数。

V 形模板 CUSUM 控制图的统计详细信息

这些公式中使用了以下符号：

- μ 表示总体均值，亦称过程均值或过程水平。
- μ_0 表示总体的目标均值（或目标）。有时用符号 $\bar{X}_0$ 来表示 μ_0。请参见 American Society for Quality Statistics Division (2004)。您可以将 μ_0 用作启动窗口“CUSUM 图的已知统计量”区域中的“目标”。
- σ 表示总体标准差。$\hat{\sigma}$ 表示 σ 的估计值。

- σ_0 表示已知的标准差。您可以将 σ_o 用作启动窗口“CUSUM 图的已知统计量”区域中的 Sigma。
- n 表示 CUSUM 图的名义样本大小。
- δ 表示要检测的 μ 中的偏移，表示为标准差的倍数。您可以将 δ 用作启动窗口“CUSUM 图的已知统计量”区域中的 Delta。
- Δ 表示要检测的 μ 中的偏移，用数据单位表示。若样本大小 n 在各子组之间保持不变，则适用以下计算：

  $\Delta = \delta\sigma_{\bar{X}} = (\delta\sigma)/\sqrt{n}$

  您可以将 Δ 用作启动窗口 “CUSUM 图的已知统计量”区域中的 “偏移”。

注意：有些作者使用符号 D 而不是 Δ。

单侧 CUSUM 图

正偏移

若要检测的偏移 δ 为正，第 t 个子组的累积和计算如下：

$$S_t = \max(0, S_{t-1} + (z_t - k))$$

$t = 1, 2, \ldots, n$，其中 $S_0 = 0$，z_t 针对双侧图定义，参数 k（称为**参考值**）为正数。若未在启动窗口中指定参数 k，则将 k 设置为 δ/2。将 CUSUM S_t 称为**上累积和**。S_t 可计算如下：

$$\max\left(0, S_{t-1} + \frac{\bar{X}_i - (\mu_0 + k\sigma_{\bar{X}_t})}{\sigma_{\bar{X}_t}}\right)$$

序列 S_t 对子组均值中距离 μ_0 超过 k 个标准误差的偏差进行累积。若 S_t 超过正值 h（称为**决策区间**），则指示偏移或失控状况。

负偏移

若要检测的偏移为负，第 t 个子组的累积和计算如下：

$$S_t = \max(0, S_{t-1} - (z_t + k))$$

$t = 1, 2, \ldots, n$，其中 $S_0 = 0$，z_t 针对双侧图定义，参数 k（称为**参考值**）为正数。若未在启动窗口中指定参数 k，则将 k 设置为 δ/2。 CUSUM S_t 称为**下累积和**。S_t 可计算如下：

$$\max\left(0, S_{t-1} - \frac{\bar{X}_i - (\mu_0 - k\sigma_{\bar{X}_t})}{\sigma_{\bar{X}_t}}\right)$$

序列 S_t 对子组均值中距离 μ_0 为 k 个标准误差以内的偏差绝对值进行累积。若 S_t 超过正值 h（称为**决策区间**），则指示偏移或失控状况。

请注意，不论 δ 正负如何，S_t 始终为正，h 始终为正。对于设计用于检测负偏移的图，有些作者定义了 S_t 的相反版本，即：在 S_t 小于负限值时指示偏移。

Lucas and Crosier (1982) 描述了 CUSUM 图的快速初始响应 (FIR) 功能的属性，其中的初始 CUSUM S_0 设置为“起始值”。他们给出的平均运行链长计算表明 FIR 功能在过程受控时影响极小，所以这导致对初始失控状况的响应速度比标准 CUSUM 图更快。您可以在启动窗口“CUSUM 图的已知统计量”区域中提供“初始值”。

常数样本大小

若子组样本大小为常数 (= n)，最好按照与数据相同的单位来调整和计算累积和。累积和计算如下：

$$S_t = \max(0, S_{t-1} + (\bar{X}_t - (\mu_0 + k\sigma/\sqrt{n})))$$

其中， $\delta > 0$

$$S_t = \max(0, S_{t-1} - (\bar{X}_t - (\mu_0 - k\sigma/\sqrt{n})))$$

其中 $\delta < 0$。在任何一种情况下，参数 k 按 $k' = k\sigma/\sqrt{n}$ 重新调整尺度。若未在启动窗口中指定参数 k，则将 k 设置为 δ/2。若 S_t 超过 $h' = h\sigma/\sqrt{n}$ 则指示偏移。某些作者用符号 H 来代替 h'。

双侧 CUSUM 图

若 CUSUM 图为双侧，为第 t 个子组标绘的累积和 S_t 定义如下：

$$S_t = S_{t-1} + z_t$$

$t = 1, 2, \ldots, n$。此处的 S_0=0，项 z_t 计算如下：

$$z_t = (\bar{X}_t - \mu_0)/(\sigma/\sqrt{n_t})$$

其中，$\bar{X}_t$ 是第 t 个子组的平均值，n_t 是第 t 个子组的样本大小。若子组样本由单个测量值 x_t 构成，项 z_t 简化为以下计算：

$$z_t = (x_t - \mu_0)/\sigma$$

第一个方程可重写为以下形式：

$$S_t = \sum_{i=1}^{t} z_i = \sum_{i=1}^{t} (\bar{X}_i - \mu_0)/\sigma_{\bar{X}_i}$$

其中，序列 S_t 对距离目标均值 μ_0 的子组平均值标准差进行累积。

在许多应用中，子组样本大小 n_i 为常数 ($n_i = n$)， S_t 的方程可以简化：

$$S_t = (1/\sigma_{\bar{X}}) \sum_{i=1}^{t} (\bar{X}_i - \mu_0) = (\sqrt{n}/\sigma) \sum_{i=1}^{t} (\bar{X}_i - \mu_0)$$

在某些应用中，最好按如下方式计算 S_t：

$$S_t = \sum_{i=1}^{t} (\bar{X}_i - \mu_0)$$

该式按照与数据相同的单位来调整尺度。在这种情况下，过程将 V 形模板参数 h 和 k 的尺度分别重新调整为 $h' = h\sigma/\sqrt{n}$ 和 $k' = k\sigma/\sqrt{n}$。某些作者用符号 F 代替 k'，用 H 代替 h'。

若过程处于受控状态且均值 μ 达到目标 μ_0 或接近该目标，则采用随机漫步模型。因此，点可能离开零，但是不会呈现大的趋势，因为距离 μ_0 的正偏移量和负偏移量趋近于彼此抵消。若 μ 具有正向偏移量，则点呈现向上趋势；若 μ 具有负向偏移量，则点呈现向下趋势。

加权移动平均图的统计详细信息

UWMA 图的控制限

每个子组 i 的 UWMA 图的控制限计算如下：

$$\text{下控制限}_i = \mu_0 - k\frac{\hat{\sigma}}{min(i,w)}\sqrt{\frac{1}{n_i} + \frac{1}{n_{i-1}} + \ldots + \frac{1}{n_{1+max(i-w,0)}}}$$

$$\text{上控制限}_i = \mu_0 + k\frac{\hat{\sigma}}{min(i,w)}\sqrt{\frac{1}{n_i} + \frac{1}{n_{i-1}} + \ldots + \frac{1}{n_{1+max(i-w,0)}}}$$

其中：

w 是跨度参数 （移动平均值中的项数）

n_i 是第 i 个子组的样本大小

k 是标准差的数目

μ_0 是子组均值的加权平均值

$\hat{\sigma}$ 是估计的过程标准差

EWMA 图的控制限

EWMA 图的控制限计算如下：

$$\text{下控制限} = \mu_0 - k\hat{\sigma}r\sqrt{\sum_{j=0}^{i-1}\frac{(1-r)^{2j}}{n_{i-j}}}$$

$$\text{上控制限} = \mu_0 + k\hat{\sigma}r\sqrt{\sum_{j=0}^{i-1}\frac{(1-r)^{2j}}{n_{i-j}}}$$

其中：

r 是 EWMA 权重参数 $(0 < r \leq 1)$

x_{ij} 是第 i 个子组中的第 j 个测量值，其中 $j = 1, 2, 3, \ldots, n_i$

n_i 是第 i 个子组的样本大小

k 是标准差的数目

μ_0 是子组均值的加权平均值

$\hat{\sigma}$ 是估计的过程标准差

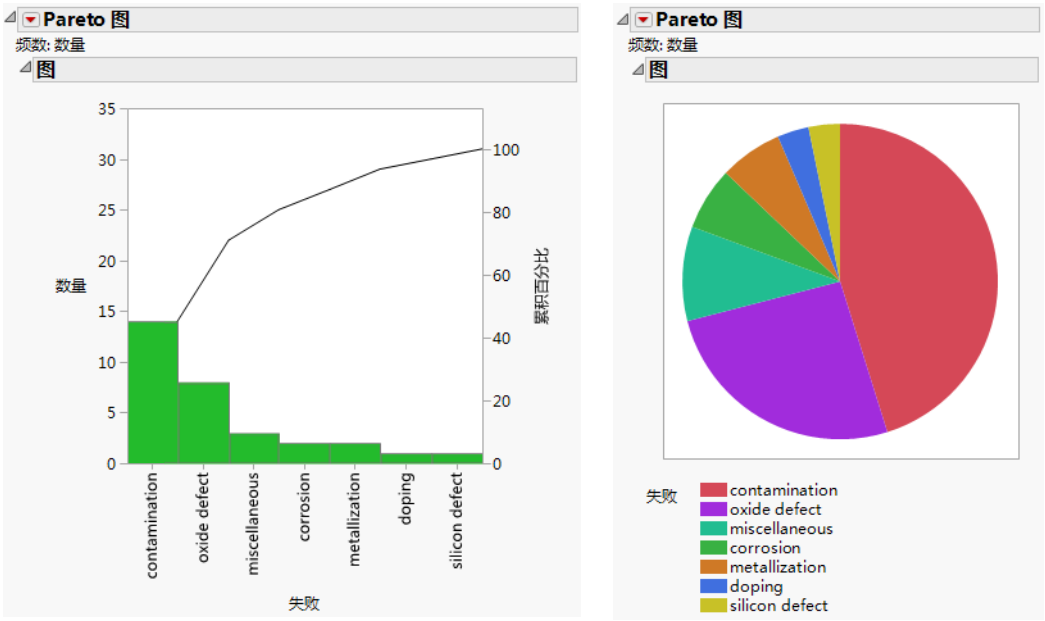
第 13 章

Pareto 图

着重关注重要少数的改善

使用 Pareto 图改进过程或操作的统计质量。Pareto 图是一个显示与质量相关的过程或操作中的问题严重性（频率）的图。Pareto 图可通过突出显示问题的频率和严重性，帮助您决定要首先解决的问题。

图 13.1 Pareto 图示例



目录

“Pareto 图”平台概述 329
“Pareto 图”平台示例 329
启动“Pareto 图”平台 332
Pareto 图报表 333
“Pareto 图”平台选项 334
原因选项 335
“控制图”平台的更多示例 335
合并原因阈值的示例 336
使用组间常数大小的示例 337
使用组间非常数样本大小的示例 338
单因子比较 Pareto 图示例 340
双因子比较 Pareto 图示例 342
“Pareto 图”平台的统计详细信息 343
似然比卡方检验 343

“Pareto 图”平台概述

“Pareto 图”平台所生成的图可显示与质量相关的过程或操作中的问题的相对频率或严重性。Pareto 图最初显示为一个条形图，其中显示以降序排列的问题的分类。值为问题原因的列将被分配 *Y* 角色，并称为**过程变量**。

您还可以生成比较 Pareto 图，用来合并同一过程变量的两个或更多 Pareto 图。单个显示图显示分配了 *X* 角色的列中的每个值，或是两个 *X* 变量的水平组合。分配了 *X* 角色的列称为**分类变量**。

Pareto 图可对单个 *Y*（过程）变量绘图，可不带任何 *X* 分类变量，也可带有一个 *X* 变量或两个 *X* 变量。Pareto 函数不区分数值和字符变量，也不区分建模类型。您可以在条形图与饼图之间切换。所有值都被视为离散值，条形或楔形代表计数或百分比。

“Pareto 图”平台示例

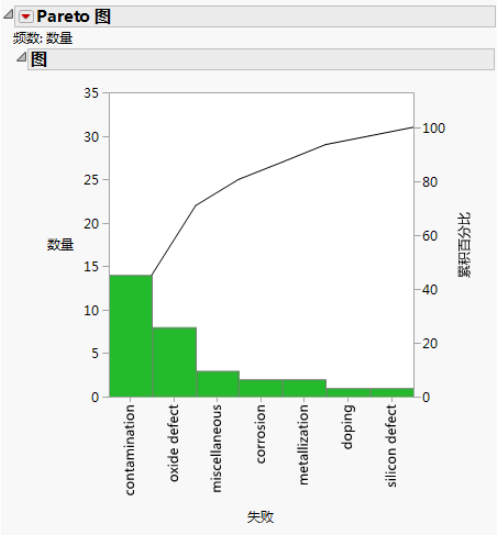
本例使用 Failure.jmp 样本数据表，其中包含失败数据和一个频数列。表中列出集成电路制作过程中失败的原因，以及每种类型缺陷出现的次数。从分析中，您可以确定导致过程失败的最重要因子。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure.jmp。
2. 选择**分析** > **质量和过程** > **Pareto 图**。
3. 选择**失败**并点击 **Y，原因**。

   该列列出失败原因。它是您要检查的变量。
4. 选择**数量**并点击**频数**。

   该列列出每种类型的失败出现的次数。
5. 点击**确定**。

图 13.2 “Pareto 图”报表窗口



左轴表示失败计数，右轴表示每一类别中的失败百分比。直条采用降序排序，出现最频繁的失败原因位于左侧。曲线从左到右指示累积失败情况。

6. 点击 “Pareto 图”红色小三角并选择**添加累积百分比点标签**。

   请注意，“Contamination”占失败总数的 45% 左右。“Oxide Defect”直条上方的点显示“Contamination”和 “Oxide Defect”合起来共占了失败总数的 71% 左右。

7. 点击 “Pareto 图”红色小三角并撤销选择**添加累积百分比点标签**和**显示累积百分比曲线**。
8. 点击标记为**数量**的 y 轴，将它重命名为**计数**。
9. 双击 y 轴以显示 **Y 轴设置**窗口。
   - 在**最大值**字段中，键入 15。
   - 在**增量**字段中，键入 2。
   - 在**轴标签行**面板中，为**主**网格线选择**网格线**。
   - 点击**确定**。
10. 点击 “Pareto 图”红色小三角并选择**类别图例**。

图 13.3 带有显示选项的 Pareto 图

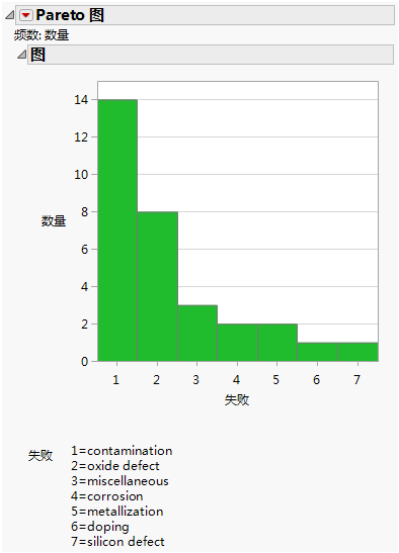
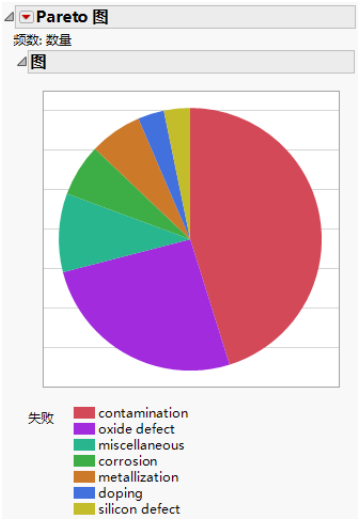


图 13.3 显示不同类型的失败计数，并显示类别图例。垂直计数轴的尺度经过调整，在主刻度标处显示网格线。

11. 要以饼图形式显示数据，请点击“Pareto 图”红色小三角并选择**饼图**。

图 13.4 显示为饼图的 Pareto 图



很明显，“Contamination”和“Oxide Defect”是失败的主要原因。

启动 “Pareto 图” 平台

通过选择**分析** > **质量和过程** > **Pareto 图**启动 “Pareto 图” 平台。

图 13.5 “Pareto 图” 启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“Pareto 图” 启动窗口包含以下选项：

Y，原因 标识其值为问题原因的列。该列称为过程变量，是您想要检查的变量。

X，分组 标识分组因子。分组变量生成一个 Pareto 图窗口，其中并排显示每个值的图。您可以不分配分组变量、分配一个分组变量（请参见第 340 页的 “单因子比较 Pareto 图示例”）或两个分组变量 （请参见第 342 页的 “双因子比较 Pareto 图示例”）。

权重 分配某个变量，赋予观测不同的权重。

频数 标识其值含有频数的列。

依据 标识一个变量，用于为列中出现的每个值生成单独分析。

合并原因阈值 支持您通过指定最小比率或计数来指定合并原因阈值。选择该选项，然后选择**尾部百分比**或**计数**并输入阈值。“尾部百分比” 选项根据指定的占合计百分比（合并的小组计数 / 组的总计数）合并较小的计数组。“计数” 选项支持您指定特定的计数阈值。有关示例，请参见第 336 页的 “合并原因阈值的示例”。

每单位分析 支持您比较各组的缺陷率。JMP 计算缺陷率以及缺陷率的 95% 置信区间。选择该选项，然后选择**常数**或**频数列中的值**并相应地输入样本大小值或原因代码。“常数” 选项支持您在启动窗口中指定常数样本大小。“频数列中的值” 选项支持您通过特殊原因代码为某个组指定唯一的样本大小，以便将行指定为原因行。

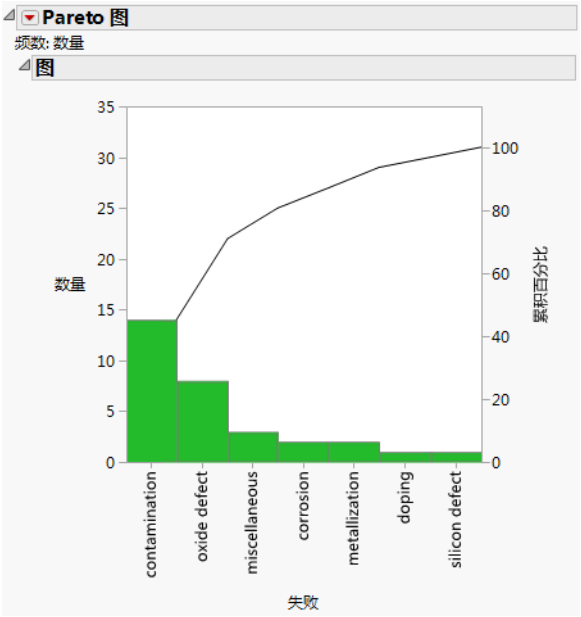
尽管可以在 Pareto 图中对原因进行组合，但这些分析的计算不会相应更改。

有关示例，请参见第 337 页的 “使用组间常数大小的示例” 和第 338 页的 “使用组间非常数样本大小的示例”。

Pareto 图报表

Pareto 图同时显示一个显示数据中变量百分比的条形图以及一个显示变量累积百分比的线图。

图 13.6 Pareto 图示例



Pareto 图可对单个 *Y*（过程）变量绘图，可不带任何 *X* 分类变量，也可带有一个 *X* 变量或两个 *X* 变量。 Pareto 图不区分数值和字符变量，也不区分建模类型。所有值都被视为离散值，直条代表计数或百分比。下面的列表说明 Pareto 图的布局构成：

- 不带 *X* 分类变量的 *Y* 变量生成单个图，其中每个 *Y* 变量值都对应一个直条。有关示例，请参见第 329 页的““Pareto 图”平台示例”。
- 带有一个 *X* 分类变量的 *Y* 变量生成一排 Pareto 图。*X* 变量的每个水平都对应一个图，每个 *Y* 水平都对应一个直条。这些图称为比较 Pareto 图的**单元格**。*X*（分类）变量的每个水平都对应一个单元格。由于只有一个 *X* 变量，所以这称为**单因子比较 Pareto 图**。有关示例，请参见第 340 页的“单因子比较 Pareto 图示例”。
- 带有两个 *X* 变量的 *Y* 变量生成按行列排列的 Pareto 图。第一个 *X* 变量的每个水平都对应一行，第二个 *X* 变量的每个水平都对应一列。由于有两个 *X* 变量，所以这称为**双因子比较 Pareto 图**。在行中，第一个 *X* 变量的每个值都有一个 Pareto 图（如前所述）。左上角单元格称为**关键单元格**。其中的直条按降序排列。其他单元格中的直条采用与关键单元格中的直条相同的顺序。您可以重新排列单元格的行列顺序。移至左上角的单元格变为新的关键单元格，所有其他单元中的直条都随之重新排列。有关示例，请参见第 342 页的“双因子比较 Pareto 图示例”。

- 每个直条采用在相关的数据表中为其分配 Y 水平的行对应的颜色。否则，对于其 Y 水平不具有所分配颜色的行的所有直条使用单一颜色。若 Y 水平的行具有不同颜色，则该 Y 水平的直条采用数据表中该 Y 水平的第一行的颜色。

您可以使用“Pareto 图”红色小三角菜单中的选项更改尺度类型和直条排列方式，还可以将这些直条转换为饼图。请参见第 334 页的“‘Pareto 图’平台选项”。

“Pareto 图”平台选项

“Pareto 图”红色小三角菜单包含用来定制图外观的选项。**原因**子菜单中也有一些选项影响 Pareto 图的单个直条。以下命令影响 Pareto 图的整体外观：

百分比尺度 显示或隐藏左垂直轴显示的计数和百分比。

数量图例 在图区域中显示或隐藏总样本大小。

类别图例 显示或隐藏添加标签的直条和单独的类别图例。

饼图 显示或隐藏条形图和饼图。

水平方向重新排序，垂直方向重新排序 有一个或多个分组变量时，对分组的 Pareto 图重新排序。

取消组合图形 将一组 Pareto 图拆分为多个单独的图。

计数分析 执行每单位缺陷分析。支持您在组间和组内比较缺陷率并执行比率检验：

每单位比率 比较各组间的缺陷率。若指定了样本大小，则每单位的缺陷数 (DPU) 列和每百万部件数 (PPM) 列将添加至报表。

检验组内比率 执行似然比卡方检验以确定各种原因之间的比率在某组内是否相同。请参见第 343 页的“‘Pareto 图’平台的统计详细信息”。

检验组间比率 执行似然比卡方检验以确定每种原因的比率在组间是否相同。请参见第 343 页的“‘Pareto 图’平台的统计详细信息”。

显示累积百分比曲线 显示或隐藏直条上方的累积百分比曲线以及右垂直轴上的累积百分比轴。

显示累积百分比轴 显示或隐藏右垂直轴上的累积百分比轴。

显示累积百分比点 显示或隐藏累积百分比曲线上的点。

添加累积百分比点标签 显示或隐藏累积曲线上的点的标签。

累积百分比曲线颜色 更改累积百分比曲线的颜色。

原因 包含影响一个或多个各图直条的选项。有关这些选项的说明，请参见第 335 页的“原因选项”。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

原因选项

您可以通过点击某个直条将其突出显示。使用 Ctrl+ 点击可选择多个不相邻的直条。选择直条后，您可以访问红色小三角菜单上影响 Pareto 图直条的命令。这些选项位于红色小三角菜单上的**原因**子菜单中。右击图区域中的任意位置后也会提供这些选项。以下选项适用于突出显示的直条而不是整个图：

合并原因 合并选定 （突出显示）的直条。您可以选择**已选定**、**最后原因**、**第一原因**，或从变量列表中进行选择。

图 13.7 “合并原因”窗口

分离原因 将选定的直条分离成其原始的构成直条。

移至最前 将一个或多个突出显示的直条移至左侧 （第一个）位置。

移至最后 将一个或多个突出显示的直条移至右侧 （最后一个）位置。

着色 显示用于对一个或多个突出显示直条着色的调色板。

标记 在实行**显示累积百分比点**命令时，显示用于为累积百分比曲线上的点分配标记的标记选项板。

标签 在所有突出显示直条的上方显示直条的值。

“控制图”平台的更多示例

- “合并原因阈值的示例”
- “使用组间常数大小的示例”

- “使用组间非常数样本大小的示例”
- “单因子比较 Pareto 图示例”
- “双因子比较 Pareto 图示例”

合并原因阈值的示例

本例使用 Failure.jmp 样本数据表，其中包含失败数据和一个频数列。表中列出集成电路制作过程中失败的原因，以及每种类型缺陷出现的次数。为该示例指定的阈值为 2。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure.jmp。
2. 选择**分析** > **质量和过程** > **Pareto 图**。
3. 选择**失败**并点击 **Y，原因**。
4. 选择**数量**并点击**频数**。
5. 选择**合并原因阈值**，然后选择**计数**。
6. 输入 2 作为阈值。
7. 点击**确定**。

图 13.8 阈值计数为 2 的 Pareto 图

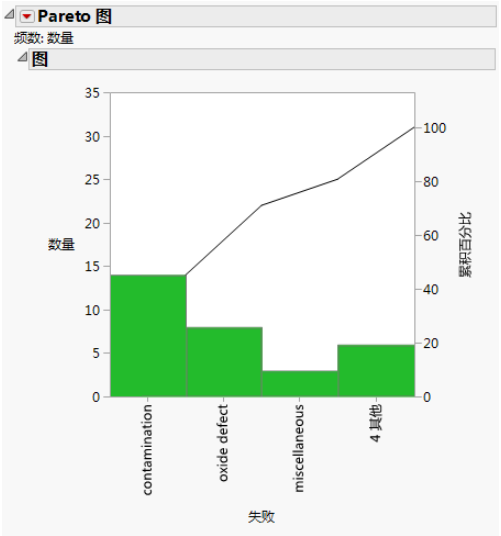
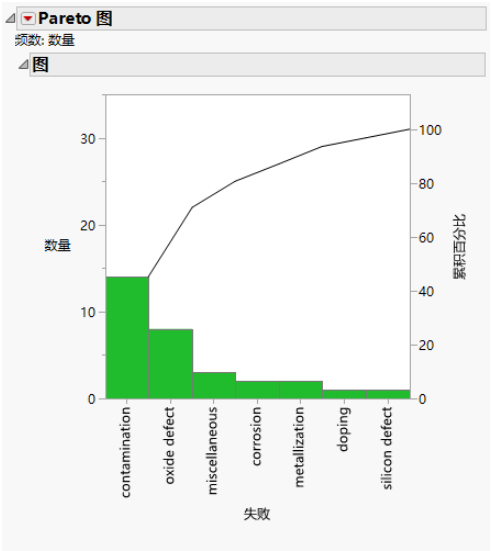


图 13.8 显示在指定计数 2 后标绘的图。计数为 2 或小于 2 的所有原因都合并为标签为“4 其他”的最后一个直条。

8. 要将合并的直条分离成图 13.9 中所示的原始类别，请选择**原因** > **分离原因**。

图 13.9 原因分离后的 Pareto 图

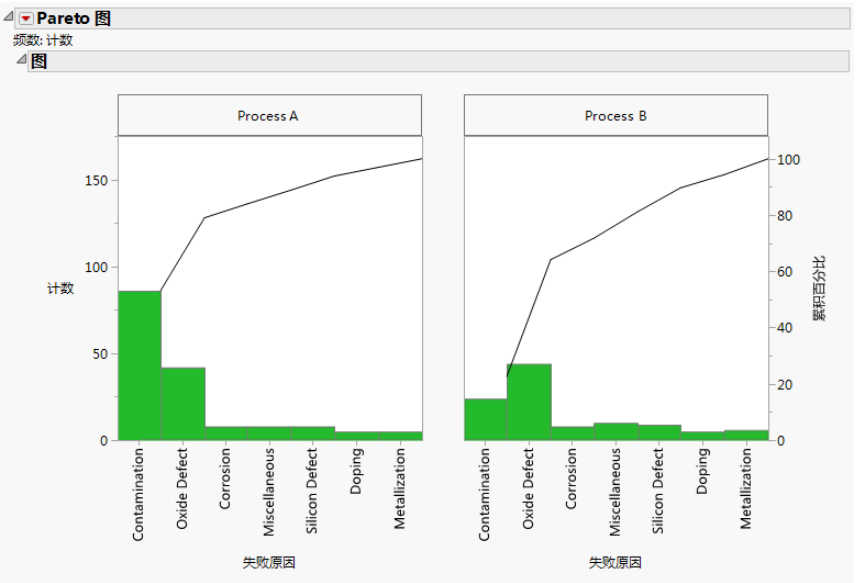


使用组间常数大小的示例

本例使用 Failures.jmp 样本数据表，其中包含失败数据和一个频数列。表中列出集成电路制作过程中失败的原因，以及两个过程中出现每种类型缺陷的次数。为该示例指定的常数样本大小为 1000。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failures.jmp。
2. 选择**分析** > **质量和过程** > **Pareto 图**。
3. 选择**失败原因**并点击 **Y，原因**。
4. 选择**工序**并点击 **X，分组**。
5. 选择**计数**并点击**频数**。
6. 选择**每单位分析**，然后选择**常数**。
7. 在**样本大小**中输入 1000。
8. 点击**确定**。

图 13.10 “Pareto 图”报表窗口



Process A 指示“Contamination”是首要失败原因，Process B 指示“Oxide Defect”为首要失败原因。

9. 点击红色小三角并选择**计数分析 > 检验组间比率**。

图 13.11 “检验组间比率”的结果

检验组间比率

检验组间比率: 工序

原因	DPU 差异	-.8 -.6 -.4 -.2 0 .2 .4 .6 .8	标准误差	卡方	自由度	概率>卡方
Contamination	0.0620		0.0105	37.0810	1	<.0001*
Oxide Defect	-0.0020		0.0093	0.0465	1	0.8292
Corrosion	0.0000		0.0040	0.0000	1	1.0000
Miscellaneous	-0.0020		0.0042	0.2227	1	0.6370
Silicon Defect	-0.0010		0.0041	0.0589	1	0.8083
Doping	0.0000		0.0032	0.0000	1	1.0000
Metallization	-0.0010		0.0033	0.0910	1	0.7629
总合计	0.0080		0.0023	11.7882	1	0.0006*

请注意，组间 （Process A 和 B）“Contamination”的 DPU 大约为 0.06。

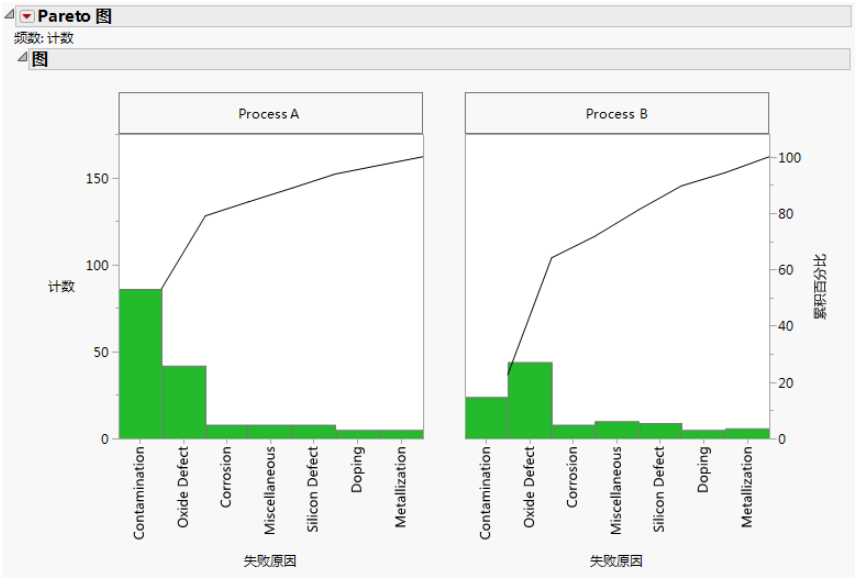
使用组间非常数样本大小的示例

本例使用 Failuressize.jmp 样本数据表，其中包含失败数据和一个频数列。表中列出集成电路制作过程中失败的原因，以及两个过程中出现每种类型缺陷的次数。在其他原因 （氧化层缺陷、硅缺陷等）中，有一种原因标记为 size。将大小指定为原因代码，即可将行指定为大小行。

1. 选择**帮助 > 样本数据库**，然后打开 Quality Control/Failuressize.jmp。
2. 选择**分析 > 质量和过程 > Pareto 图**。

3. 选择**失败原因**并点击 **Y，原因**。
4. 选择**工序**并点击 **X，分组**。
5. 选择**计数**并点击**频数**。
6. 选择**每单位分析**，然后选择**频数列中的值**。
7. 在**代码原因**中输入 size。
8. 点击**确定**。

图 13.12 “Pareto 图”报表窗口



9. 点击 “Pareto 图”红色小三角并选择**计数分析** > **每单位比率**和**计数分析** > **检验组间比率**。

图 13.13 “每单位比率”和“检验组间比率”的结果

每单位比率

工序	原因	计数	DPU	PPM	95% 下限	95% 上限
Process A	Contamination	86	0.8515	851485.15	0.6811	1.0516
	Oxide Defect	42	0.4158	415841.58	0.2997	0.5621
	Corrosion	8	0.0792	79207.92	0.0342	0.1561
	Miscellaneous	8	0.0792	79207.92	0.0342	0.1561
	Silicon Defect	8	0.0792	79207.92	0.0342	0.1561
	Doping	5	0.0495	49504.95	0.0161	0.1155
	Metallization	5	0.0495	49504.95	0.0161	0.1155
	总合计	162	0.2291	229137.20	0.1952	0.2673
	size	101				
Process B	Contamination	24	0.1655	165517.24	0.1061	0.2463
	Oxide Defect	44	0.3034	303448.28	0.2205	0.4074
	Corrosion	8	0.0552	55172.41	0.0238	0.1087
	Miscellaneous	10	0.0690	68965.52	0.0331	0.1268
	Silicon Defect	9	0.0621	62068.97	0.0284	0.1178
	Doping	5	0.0345	34482.76	0.0112	0.0805
	Metallization	6	0.0414	41379.31	0.0152	0.0901
	总合计	106	0.1044	104433.50	0.0855	0.1263
	size	145				

检验组间比率

检验组间比率: 工序

原因	DPU 差异	-.8 -.6 -.4 -.2 0 .2 .4 .6 .8	标准误差	卡方	自由度	概率>卡方
Contamination	0.6860		0.0978	63.0776	1	<.0001*
Oxide Defect	0.1124		0.0788	2.1195	1	0.1454
Corrosion	0.0240		0.0341	0.5202	1	0.4707
Miscellaneous	0.0102		0.0355	0.0847	1	0.7710
Silicon Defect	0.0171		0.0348	0.2500	1	0.6171
Doping	0.0150		0.0270	0.3251	1	0.5685
Metallization	0.0081		0.0278	0.0871	1	0.7679
总合计	0.1247		0.0207	40.7524	1	<.0001*

请注意，使用样本大小 101 来计算组 A 中原因的 DPU；但使用样本大小 145 来计算组 B 中原因的 DPU。

若有两个分组变量（比如，日期和工序），“每单位比率”将为每种原因列出每种日期和工序组合的 DPU 或比率。不过，“检验组间比率”仅检验组间的总差异。

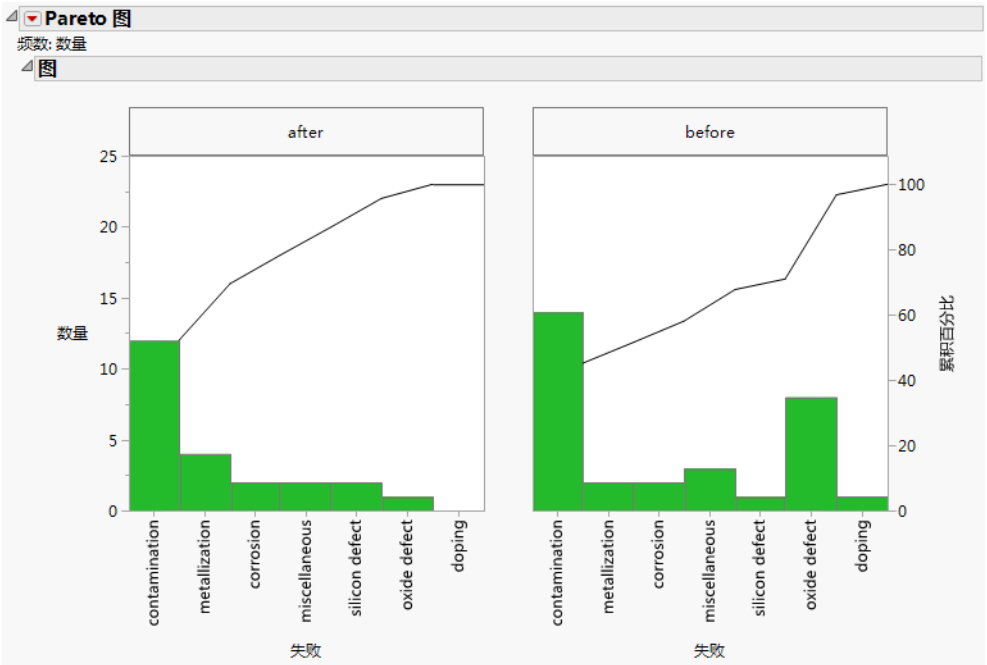
单因子比较 Pareto 图示例

本例使用 Failure2.jmp 样本数据表。该表记录清洗扩散炉中的管道之前生产的一组电容样本中的失败数据，以及在清洗扩散炉之后生产的一组样本中的数据。对于每类失败数据，变量**清洗**使用值“before”或“after”来标识样本。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure2.jmp。
2. 选择**分析** > **质量和过程** > **Pareto 图**。
3. 选择**失败**并点击 **Y，原因**。
4. 选择**清洗**并点击 **X，分组**。
5. 选择**数量**并点击**频数**。
6. 点击**确定**。

图 13.14 为变量**清洗**的每个值显示了并排图。

图 13.14 单因子比较 Pareto 图

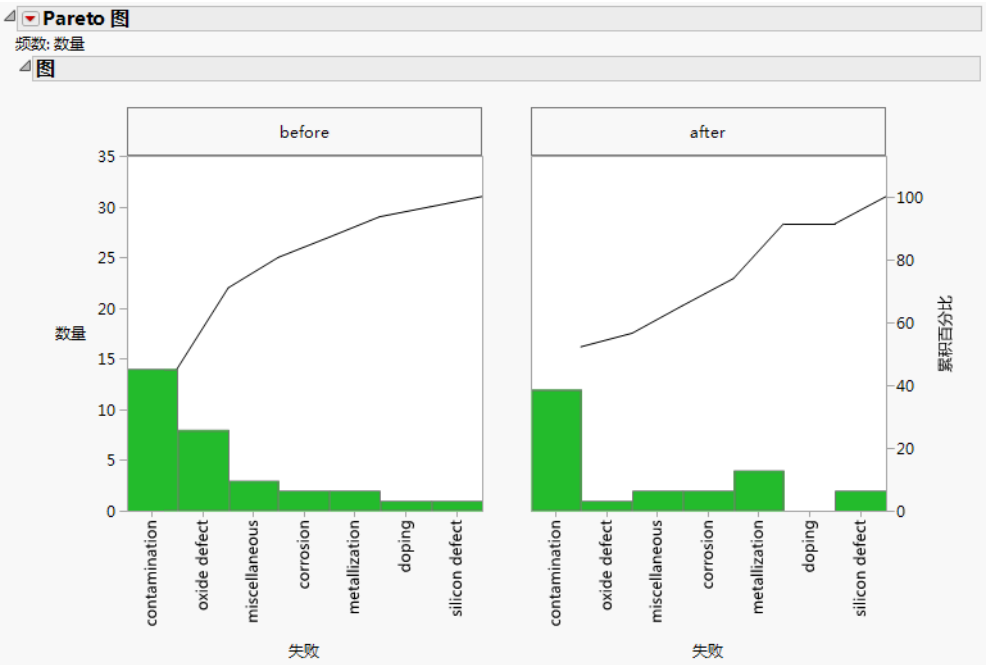


这两个图的水平轴和垂直轴采用相同的尺度。第一个图中的直条以降序显示 y 轴值，由此决定所有单元格的顺序。

7. 通过点击第一个图块中的标题 (after) 并将其拖至下一图块的标题 (before)，可重新排列这些图的顺序。

   通过比较这两个图，可以看出在清洗之后氧化层缺陷减少了。不过，采用图 13.15 所示的“before-after”图形式表示时，各图更容易解释。请注意，原因顺序会改变以反映基于第一个单元格的顺序。

图 13.15 单元格已重新排序的单因子比较 Pareto 图



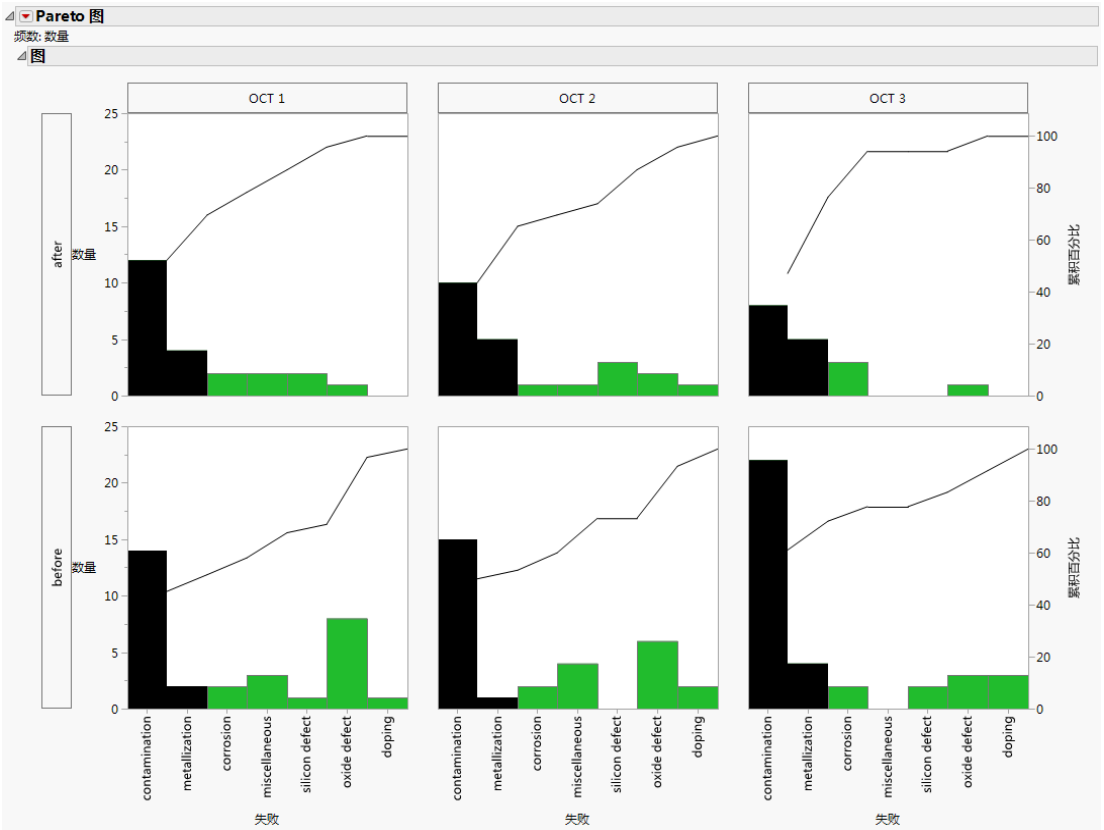
双因子比较 Pareto 图示例

本例使用 Failure3.jmp 样本数据表。该数据监控在清洗扩散炉之前和之后三天的电容生产过程的生产样本。数据表中有一个称为**日期**的列，其中包含值 OCT 1、 OCT 2 和 OCT 3。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure3.jmp。
2. 选择**分析** > **质量和过程** > Pareto 图。
3. 选择**失败**并点击 **Y，原因**。
4. 选择**清洗**和**日期**，然后点击 **X，分组**。
5. 选择**数量**并点击**频数**。
6. 点击**确定**。

   图 13.16 显示了具有双因子图布局的 Pareto 图，其中显示了两个 *X* 变量的每个水平。左上角单元格称为**关键单元格**。其中的直条按降序排列。其他单元格中的直条采用与关键单元格中的直条相同的顺序。
7. 点击关键单元格中的 Contamination 和 Metallization，相应类别的直条在其他所有单元格中会突出显示。

图 13.16 双因子比较 Pareto 图



Pareto 图突出显示了**重要的少数**。在双因子比较图的每个单元格中，表示两个最频繁出现的问题的直条处于选定状态。Contamination 和 Metallization 是所有单元格中重要的两个类别，但在清洗扩散炉后，Contamination 这个问题已不再重要。

“Pareto 图”平台的统计详细信息

似然比卡方检验

符号

在“Pareto 图”平台中计算的似然比卡方检验统计量使用以下符号：

- n_{ij} 是组 j 中的原因 i 的计数。
- E_j 是组 j 的期望计数。这是每个组跨不同原因的均值计数。
- E_i 是原因 i 的期望计数。这是每个原因跨不同组的均值计数。

组内似然比卡方检验统计量

$$G_j^2 = 2\sum_{i=1}^{K} n_{ij}\ln(n_{ij}/E_j)$$

组间似然比卡方检验统计量

$$G_i^2 = 2\sum_{j=1}^{J} n_{ij}\ln(n_{ij}/E_i)$$

第 14 章

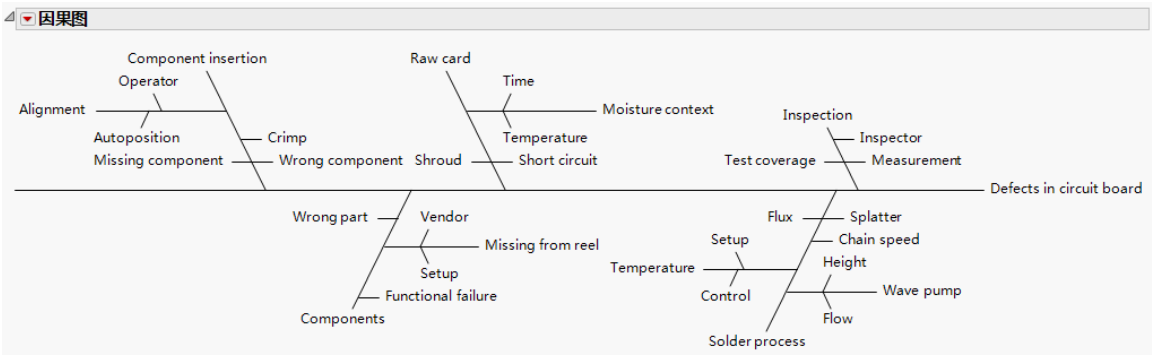
因果图

确定根本原因

使用 “关系图”平台可构建因果图，亦称**石川因果图**或**鱼骨图**。使用这些关系图可以：

- 整理导致某种效应的原因（问题根源）
- 执行头脑风暴分析
- 标识变量，为进一步实验做准备

图 14.1 因果图示例



目录

因果图概述 347
因果图示例 347
准备数据 347
启动“关系图”平台 348
因果图 349
右击菜单 349
“因果图”菜单选项 351
保存关系图 352
将关系图另存为数据表 352
将关系图另存为记录 352
将关系图另存为脚本 353

因果图概述

使用 “关系图”平台可构建因果图，亦称**石川因果图**或**鱼骨图**。使用这些关系图可以：

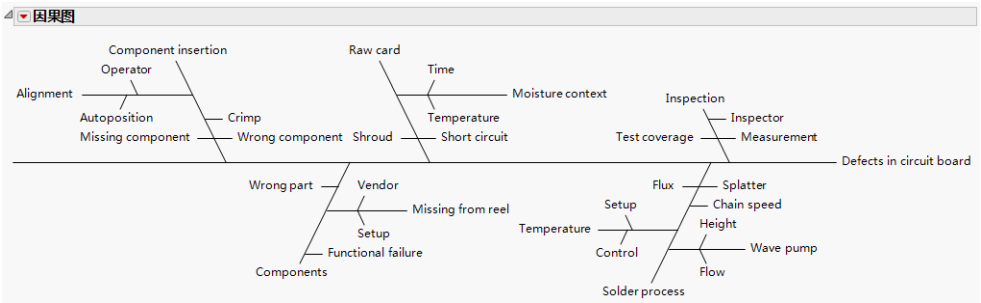
- 整理导致某种效应的原因（问题根源）
- 执行头脑风暴分析
- 标识变量，为进一步实验做准备

因果图示例

您的数据中记录着电路板中的缺陷。您想要通过关系图来检查缺陷的主要因子和可能原因。

1. 选择**帮助** > **样本数据库**，然后打开 Ishikawa.jmp。
2. 选择**分析** > **质量和过程** > **关系图**。
3. 选择**父项**并点击 **X，父级项**。
4. 选择**子项**并点击 **Y，子级项**。
5. 点击**确定**。

图 14.2 Ishikawa.jmp 关系图



主要因子包括 Inspection、Solder process、Raw card、Components 和 Component insertion。从每个主要因子中又可细分出可能的分支原因，如从 Inspection 因子可细分出 Inspector、Measurement 和 Test coverage。

您可以一次关注一个方面，进一步检查每个主要因子的可能原因或变异来源。

准备数据

生成关系图之前，请先从数据表中的两列数据入手。

图 14.3 Ishikawa.jmp 数据表的示例

	父项	子项
1	Defects in circuit board	Inspection
2	Defects in circuit board	Solder process
3	Defects in circuit board	Raw card
4	Defects in circuit board	Components
5	Defects in circuit board	Component insertion
6	Inspection	Measurement
7	Inspection	Test coverage
8	Inspection	Inspector
9	Solder process	Splatter
10	Solder process	Flux
11	Solder process	Chain speed
12	Solder process	Temperature
13	Solder process	Wave pump
14	Temperature	Setup

请注意，**父项**值 Defects in circuit board（效应）有五个主要因子，这些因子列在**子项**列中。其中一个主要因子是 Inspection，该因子本身包含的原因列在**子项**列中。父项值可包含子项，而子项又可以包含自己的子项（因此可同时列在**父项**列和**子项**列中。）

启动“关系图”平台

通过选择**分析** > **质量和过程** > **关系图**启动 “关系图”平台。

图 14.4 “关系图”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

提示：若想不基于数据表创建基本关系图，请将 **Y，子级项**和 **X，父级项**字段留空，并点击**确定**。然后使用右击菜单中的选项编辑节点。请参见第 349 页的“右击菜单”。

Y，子级项 表示为父项因子做贡献的子项因子。

X，父级项 表示包含子项因子的父项因子 （包括效应）。

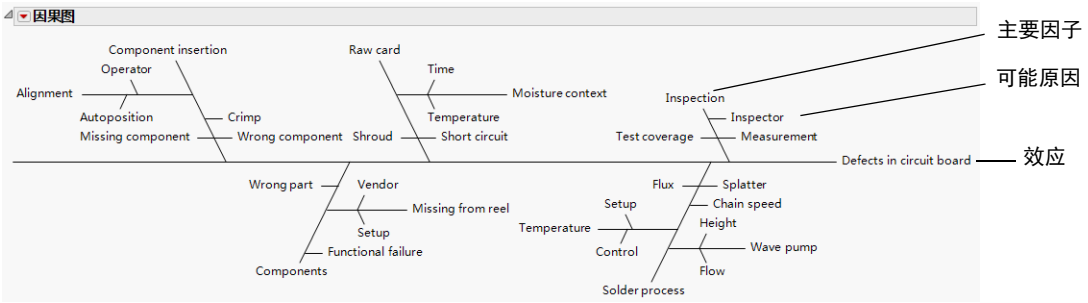
标签 在关系图节点中包含 “标签”列中的文本。

依据 为 “依据”变量的每个值生成单独的关系图。

因果图

在图 14.5 中作为效应（也称问题）的 Defects in circuit board 显示在右侧的中心线。主要贡献因子显示在分支末尾（Inspection、Solder process、Raw Card 等）。可能原因显示为每个主要因子的分支。

图 14.5 因果图



右击菜单

右击某一突出显示的节点可执行修改文本、插入新节点、更改关系图类型等操作。请注意以下事项：

- 右击某个标题可更改字体和颜色、位置、可见性或格式。
- 点击并突出显示某个节点可对其重命名。
- 点击并拖动可移动某个节点。

“文本”菜单

“文本”菜单包含以下选项：

字体 选择文本或数值字符的字体。

颜色 选择文本或数值字符的颜色。

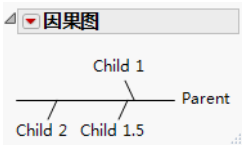
向左旋转、向右旋转、水平 旋转文本或数字使其水平显示、向左旋转 90 度或向右旋转 90 度。

“插入”菜单

使用**插入**菜单可向现有节点插入项。“插入”菜单包含以下选项：

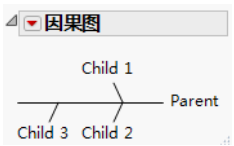
之前 在突出显示的节点右侧插入新节点。例如，图 14.6 在 Child 2 之前插入了 Child 1.5。

图 14.6 之前插入



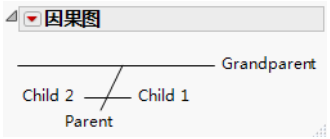
之后 在突出显示的节点左侧插入新节点。例如，图 14.7 在 Child 2 之后插入了 Child 3。

图 14.7 之后插入



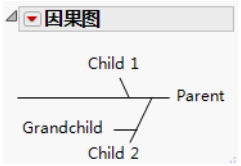
之上 在当前节点之上的一级插入新节点。例如，图 14.8 在 Parent 之上的一级插入了 Grandparent。

图 14.8 之上插入



之下 在当前节点之下的一级插入新节点。例如，图 14.9 在 Child 2 之下的一级插入了 Grandchild。

图 14.9 之下插入



“移动”菜单

使用 “移动”菜单可移动节点或分支。“移动”菜单包含以下选项：

第一个 将突出显示的节点移至其父项下的第一个位置。

最后一个 将突出显示的节点移至其父项下的最后一个位置。

另一侧 将突出显示的节点移至其父项线的另一侧。

强制靠左 使所有水平绘制的元素显示在其父项的左侧。

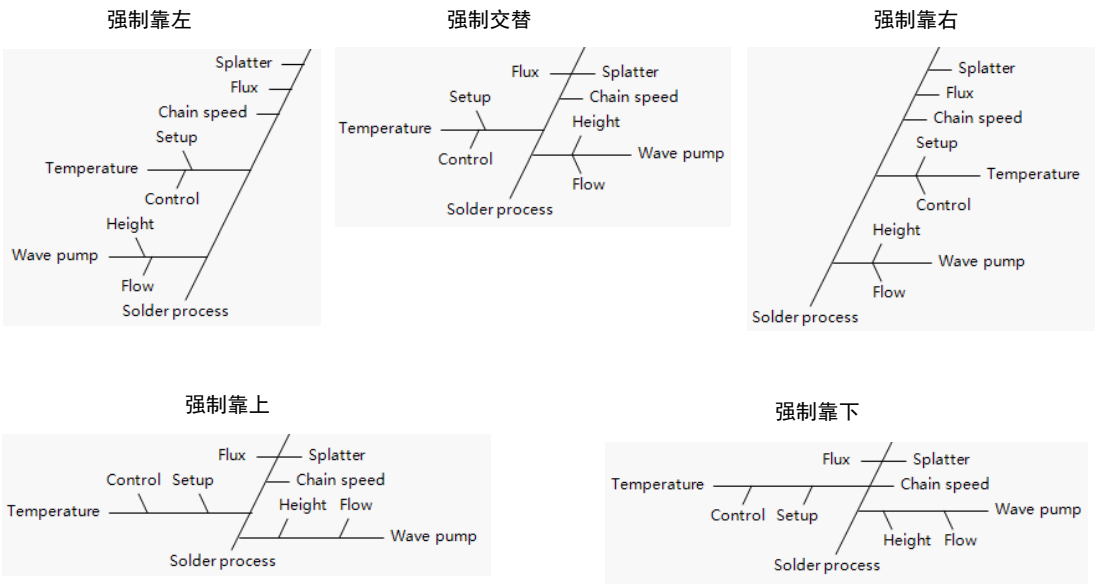
强制靠右 使所有水平绘制的元素显示在其父项的右侧。

强制靠上 使所有垂直绘制的元素显示在其父项上方。

强制靠下 使所有垂直绘制的元素显示在其父项下方。

强制交替 在父项线两侧交替绘制子项。

图 14.10 强制选项



其他菜单选项

突出显示节点的右击菜单还包含以下选项：

更改类型 将整个图表类型改为**鱼骨图**、**层次**或**嵌套**。

不可编辑 禁用除**移动**和**更改类型**之外的其他所有命令。

文本换行宽度 指定文本换行时的标签宽度。

制成数据表 将当前突出显示的节点转换为数据表。通过突出显示整个关系图（效应）转换所有节点。

关闭 显示突出显示的节点。

删除 删除突出显示的节点及其所有子项。

“因果图”菜单选项

“因果图”红色小三角菜单包含以下选项：

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

保存关系图

可通过不同方式保存您的关系图。请从以下方法中选择一种：

- 将关系图另存为数据表
- 将关系图另存为记录
- 将关系图另存为脚本

将关系图另存为数据表

需要注意关于该方法的以下事项：

- 若有其他过程需要更新该数据表，则这是一种可供选择的理想方法。
- 几乎不能进行定制，因为数据表无法表示定制内容。

将关系图另存为数据表：

1. 突出显示整个关系图。
2. 右击并选择**制成数据表**。
3. 保存新的数据表。

将关系图另存为记录

需要注意关于该方法的以下事项：

- 该选项对于临时的工作不失为好的选择。例如，您可以手动生成关系图，将其另存为记录，稍后再重新打开该记录，继续生成和编辑关系图。
- 所有定制内容仅在记录中存在，记录本身不会连接到数据表。

将关系图另存为记录：

1. 突出显示整个关系图。
2. 右击并选择**编辑 > 记录**。
3. 保存新记录。

将关系图另存为脚本

需要注意关于该方法的以下事项：

- 若有其他过程需要更新该数据表，则这是一种可供选择的理想方法。
- 若是基于数据表创建的关系图，则会显示简单的脚本，可针对该数据表重新启动但不显示任何定制内容。
- 若并不是使用数据表创建的关系图或者是从记录创建的关系图，则会显示较为复杂的脚本，其中包含添加和定制关系图各区域所需的所有命令。

将关系图另存为脚本：

1. 点击“因果图”旁边的红色小三角并选择**保存脚本 > 至脚本窗口**。
2. 保存新脚本。

第 15 章

质量实用工具

管理规格限并创建 OC 曲线

本章介绍“分析”>“质量和过程”菜单中的实用工具。具体而言，本章讨论了“管理规格限”实用工具和“OC 曲线”实用工具。

目录

“管理规格限”实用工具 357
“管理规格限”实用工具的示例 357
“管理规格限”选项 360
“操作特征曲线”实用工具 361
启动“OC 曲线”实用工具 361
控制图的 OC 曲线选项 361
用于验收抽样的 OC 曲线选项 362

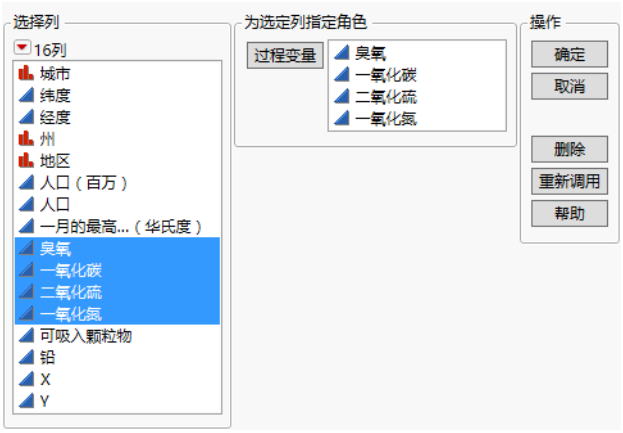
“管理规格限”实用工具

“管理规格限”实用工具支持您同时快速添加或编辑若干列的许多规格限。这些规格限随后会用在将来的任何分析中。您还可以指定每个过程的重要性值并指示限值是否应该作为参考线显示在图形中。

“管理规格限”实用工具的示例

1. 选择**帮助** > **样本数据库**，然后打开 Cities.jmp。
2. 选择**分析** > **质量和过程** > **管理规格限**。
3. 指定您想要对其设置规格限的列。对于本例，选择**臭氧**、**一氧化碳**、**二氧化硫**和**一氧化氮**，然后点击**过程变量**。

图 15.1 指定列



4. 点击**确定**。
5. 添加规格限。您可以通过从 JMP 数据表中加载现有限值 （**从限值表加载**）或手动输入限值来完成该操作。对于本例，手动输入以下限值：
   - **臭氧**：下规格限 0.12，上规格限 0.2
   - **一氧化碳**：下规格限 6，上规格限 12
   - **二氧化硫**：下规格限 0.015，上规格限 0.06
   - **一氧化氮**：下规格限 0.02，上规格限 0.04
6. 点击 “管理规格限”旁边的红色小三角并选择**显示全部限值**。

   所有列的规格限都将显示在图形中，供将来的任何分析使用。若您只想为单个列显示规格限，请选中这些列旁边的**显示限值**框。

图 15.2 设置规格限

7. 选择您希望如何保存规格限。对于本例，点击**保存至列属性**。这会将规格限另存为相应数据表中的列属性。您还可以将其保存到新数据表中 （高型或宽型）。

   在 Cities.jmp 数据表 “列” 面板中，请注意星号指示 “规格限” 列属性显示在**臭氧**、**一氧化碳**、**二氧化硫**和**一氧化氮**的旁边。

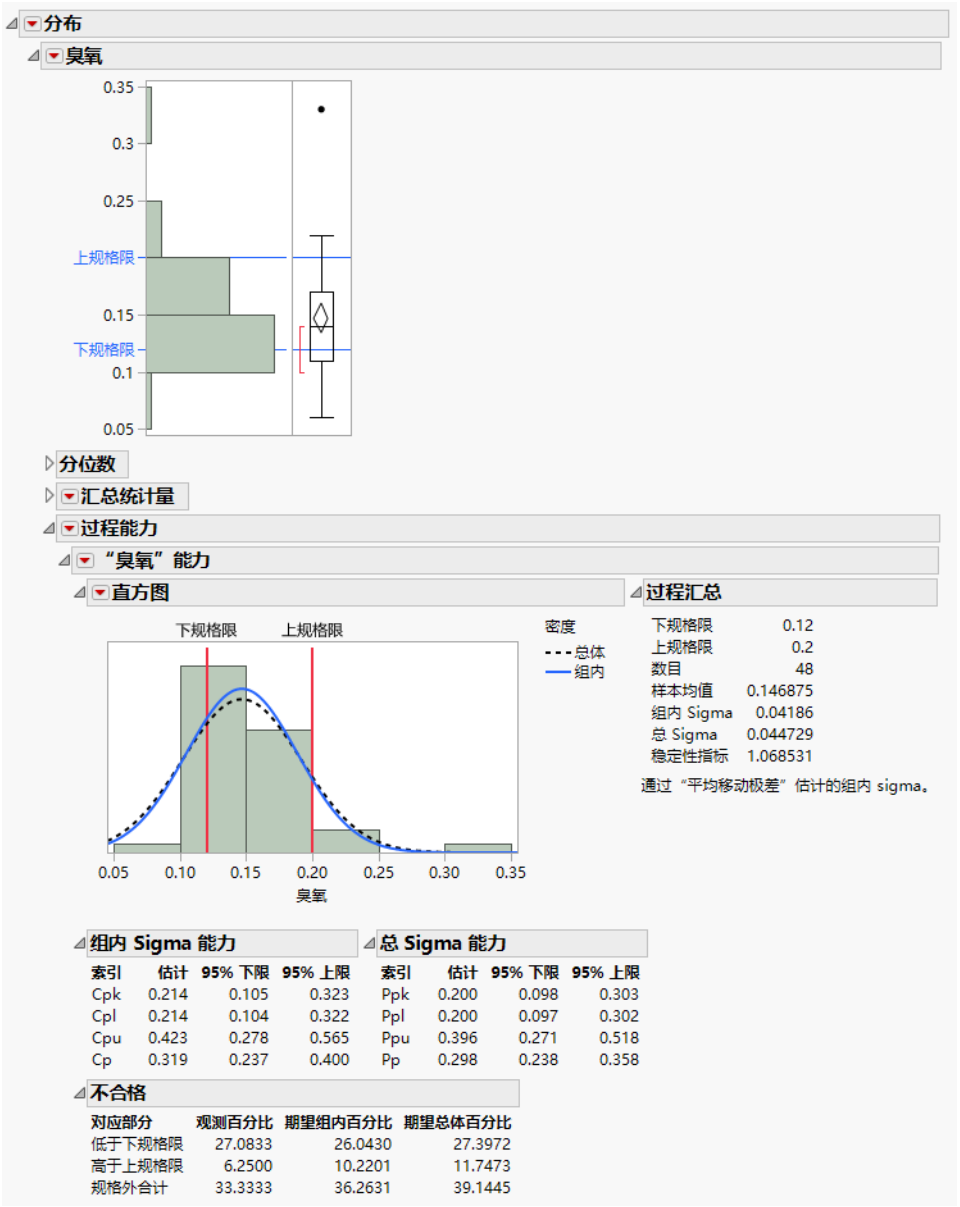
8. 要查看超出数据表中限值的值，点击 “管理规格限” 旁边的红色小三角并选择**对超出规格限的值着色**。转到 Cities.jmp 数据表，您可以看到超出规格限的所有值现在均已着色。

9. 现在，您可以运行任何分析。对于本例，选择**分析 > 分布**。

10. 选择**臭氧**、**一氧化碳**、**二氧化硫**和**一氧化氮**，然后点击 **Y，列**。

11. 点击**确定**。

图 15.3 分布中的 “臭氧”的规格限



您添加到**臭氧**列的规格限显示在直方图中。由于该列包含“规格限”列属性，“分布”报表还包含“能力分析”报表。

“管理规格限”选项

在您设置规格限的窗口中，提供用于保存和加载规格限的按钮以及 “管理规格限”红色小三角菜单中的选项。

报表表列

“管理规格限”报表顶部的表为启动窗口中指定的每个过程列都包含一行。该表支持您指定规格限并为每列提供以下选项：

显示限值 指定在指定列的 “规格限”列属性中选定 “显示为图形参考线”选项。

过程重要性 指定每列的过程重要性值。过程重要性值提供按您喜欢的顺序对过程排序的机制。过程重要性值用于调整许多平台图形中的标记大小。

单位 指定每列的单位。

按钮

从限值表加载 从 JMP 数据表加载规格限。

保存至列属性 将规格限另存为关联的数据表中的列属性。

保存至高型规格限表 将规格限保存至高型的新数据表。

保存至宽型规格限表 将规格限保存至宽型的新数据表。

红色小三角选项

显示全部限值 选中所有列的“显示限值”下的框。若为某列选定了“显示限值”，则会在“规格限”列属性中选定 “显示为图形参考线”选项。“显示为图形参考线”选项将您指定的规格限和目标显示为选定分析图中的参考线。

注意：若选定 “显示限值”下的所有框，则 “显示全部限值”选项将取消选定 “显示限值”下的所有框。

修约小数 设置您想要规格限化整到的小数位数。

对超出规格限的值着色 对数据表中超出列的规格限的所有值着色。

请参见《使用 JMP》中的获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

“操作特征曲线”实用工具

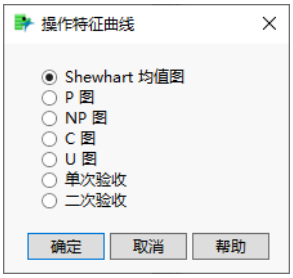
使用 “操作特征 (OC) 曲线”实用工具，您可以为控制图和计数验收抽样计划构造 OC 曲线。OC 曲线可用于均值、P、NP、C 和 U 图。对于指定的控制图，OC 曲线显示未能检测到特定大小偏移的概率。此外，还有用于单次和二次计数验收抽样计划的 OC 曲线。对于指定的验收抽样计划， OC 曲线显示验收批次的概率如何随批次质量而变化。

注意：控制图的 OC 曲线是双侧曲线。针对负偏移和正偏移绘制 OC 曲线。通常 OC 曲线显示一条绝对偏移曲线。

启动“OC 曲线”实用工具

通过选择**分析 > 质量和过程 > OC 曲线**，启动 “操作特征曲线”。选择关注的 OC 曲线并点击“确定”以启动。

图 15.4 “OC 曲线”启动窗口



控制图的 OC 曲线选项

对于控制图，可以使用 OC 曲线来探索规格如何影响 β，这是无法检测到关注的指定偏移的概率。使用 OC 曲线上的文本框、滑块或下控制限、偏移和上控制限等控点可设置和调整规格。规格如下所示：

下控制限 指定控制图中的下控制限。

上控制限 指定控制图中的上控制限。

样本大小 （不可用于 C 图。）指定用于控制图测度的样本大小。

Sigma （仅可用于均值图。）指定控制图 sigma。

偏移 指定要检测的偏移。

Beta 指定在给定控制图规格的情况下，未能检测到指定偏移的概率。Beta 随规格的更改而更新。

用于验收抽样的 OC 曲线选项

对于计数验收抽样，可以使用 OC 曲线来探索抽样计划和假设的产品质量如何影响批次验收的概率。使用 OC 曲线上的文本框、滑块或不良率控点可设置和调整抽样计划。

单次抽样 OC 曲线选项

抽样类型 支持您选择 “批抽样”或 “二项抽样”。

批抽样 支持您基于固定批大小指定和探索验收计划。

二项抽样 支持您指定和探索适用于二项分布的连续过程或其他情况的验收计划。

批大小 (N) （仅可用于 “批抽样”。）指定要从中抽样的批的大小。

样本大小 (n) 指定待检单元的数量。

可接受失效 (c1) 指定允许失效的数目。若观测到的缺陷数量大于 c1，则该批被拒收。

不良率 指定该批的预期不良率。

验收概率 指定根据定义的抽样计划验收该批的概率。“验收概率”随着规格的更改而更新，但您也可以直接调整该值。当您直接调整 “验收概率”时，“不良率”值随之更新。

二次抽样 OC 曲线选项

第一个样本 包含第一个样本的以下规格：

检查数 (n1) 指定第一个样本中检验的单元数。

可接受失效 (c1) 指定第一个样本中允许失效的数目。

第二个样本 包含第二个样本的以下规格：

检查数 (n2) 指定第二个样本中检验的单元数。

可接受失效 (c1+c2) 指定允许失效的总数。

不良率 指定该批的预期不良率。

验收概率 指定根据定义的抽样计划验收该批的概率。“验收概率”随着规格的更改而更新，但您也可以直接调整该值。当您直接调整 “验收概率”时，“不良率”值随之更新。

附录 A

参考书目

Agresti, A., and Coull, B. (1998).“Approximate is Better Than ‘Exact’ for Interval Estimation of Binomial Proportions.”*The American Statistician* 52:119–126.

American Society for Quality Statistics Division (2004).*Glossary and Tables for Statistical Quality Control*.4th ed. Milwaukee, WI: ASQC Quality Press.

American Society for Testing and Materials (1976).*ASTM Manual on Presentation of Data and Control Chart Analysis*.Philadelphia: ASTM.

Automotive Industry Action Group (2002).*Measurement Systems Analysis Reference Manual*.3rd ed. Troy, MI: Automotive Industry Action Group.

AIAG (2005).*Statistical Process Control*.2nd ed. Troy, MI: Automotive Industry Action Group.

Barrentine, L. B. (1991).*Concepts for R&R Studies*.Milwaukee, WI: ASQC Quality Press.

Bissell, A. F. (1990).“How Reliable Is Your Capability Index?”*Journal of the Royal Statistical Society, Series C* 39:331–340.

Burdick, P. K., Borror, C. M., and Montgomery, D. C. (2005).*Design and Analysis of Gauge R&R Studies: Making Decisions with Confidence Intervals in Random Mixed ANOVA Models*.Philadelphia, PA and Alexandria, VA: SIAM and ASA.

Chou, Y.-M., Owen, D. B., and Borrego, S. A. (1990).“Lower confidence limits on process capability indices.”*Journal of Quality Technology* 22:223–229.

Crowder, S. V. (1987).“Average Run Lengths of Exponentially Weighted Moving Average Charts.”*Journal of Quality Technology* 29:401–408.

David, H. A. (1951).“Further Applications of Range to the Analysis of Variance.”*Biometrika* 38:393–407.

Fleiss, J. L. (1981).*Statistical Methods for Rates and Proportions*.New York: John Wiley & Sons.

Goel, A. L., and Wu, S. M. (1971).“Determination of A.R.L. and a Contour Nomogram for Cusum Charts to Control Normal Mean.”*Technometrics* 13:221–230.

Harter, H. L. (1960).“Tables of Range and Studentized Range.”*Annals of Mathematical Statistics* 31:1122–1147.

Hoffman, D. (2003).“Negative Binomial Control Limits for Count Data with Extra-Poisson Variation.”*Pharmaceutical Statistics* 2:127–132.

Jackson, J. E., and Mudholkar, G. S. (1979).“Quality Control Procedures for Residuals Associated with Principal Component Analysis.”*Annals of Mathematical Statistics* 25:290–302.

Knoth, S. (2004).“Fast Initial Response Features for EWMA Control Charts.”*Statistical Papers* 46:47–64.

Kourti, T., and MacGregor, J. F. (1996).“Multivariate SPC Methods for Process and Product Monitoring.”*Journal of Quality Technology* 28:409–428.

Levey, S., and Jennings, E. R. (1950).“The Use of Control Charts in the Clinical Laboratory.”*American Journal of Clinical Pathology* 20:1059–1066.

Li, G., Qin, S., Ji, Y., Zhou, D.(2009).“Total PLS Based Contribution Plots for Fault Diagnosis.”*Acta Automatica Sinica* 35:759 - 765.

Lucas, J. M. (1976).“The Design and Use of V–Mask Control Schemes.”*Journal of Quality Technology* 8:1–12.

Lucas, J. M., and Crosier, R. B. (1982).“Fast Initial Response for CUSUM Quality Control Schemes: Give Your CUSUM a Head Start.”*Technometrics* 24:199–205.

Meeker, W. Q., and Escobar, L. A. (1998).*Statistical Methods for Reliability Data*.New York: John Wiley & Sons.

Montgomery, D. C. (2013).*Introduction to Statistical Quality Control*.7th ed. New York: John Wiley & Sons.

Nair, V. N. (1984).“Confidence Bands for Survival Functions with Censored Data: A Comparative Study.”*Technometrics* 26:265–275.

Nelson, L. (1984).“The Shewhart Control Chart—Tests for Special Causes.”*Journal of Quality Technology* 15:237–239.

Nelson, L. (1985).“Interpreting Shewhart X Control Charts.”*Journal of Quality Technology* 17:114–116.

Nomikos, P., and MacGregor, J. F., (1995).“Multivariate SPC Charts for Monitoring Bach Processes.”*Technometrics* 37:41–59.

Pearn, W. L., and Kotz, S. (2006).*Encyclopedia and Handbook of Process Capability Indices*.Vol. 12 of Series on Quality, Reliability, and Engineering Statistics.New Jersey: World Scientific.

Portnoy, S. (1971).“Formal Bayes Estimation with Application to a Random Effects Model.”*The Annals of Mathematical Statistics* 42:1379–1402.

Sahai, H. (1974).“Some Formal Bayes Estimators of Variance Components in the Balanced Three-Stage Nested Random Effects Model.”*Communication in Statistics – Simulation and Computation* 3:233–242.

Slifker, J. F., and Shapiro, S. S. (1980).“The Johnson System: Selection and Parameter Estimation.”*Technometrics* 22:239–246.

Sullivan, J. H., and Woodall, W. H. (2000).“Change-point detection of mean vector or covariance matrix shifts using multivariate individual observations.”*IIE Transactions* 32:537–549.

Tracy, N. D., Young, J. C., and Mason, R. L. (1992).“Multivariate Control Charts for Individual Observations.”*Journal of Quality Technology* 24:88–95.

Westgard, J. O. (2002).*Basic QC Practices*.2nd ed. Madison, WI: Westgard QC.

Wheeler, D. J. (2004).*Advanced Topics in Statistical Process Control*.2nd ed. Knoxville, TN: SPC Press.

Wheeler, D. J. (2006).*EMP III Using Imperfect Data*.Knoxville, TN: SPC Press.

Wludyka, P., and Sa, P. (2004).“A robust I-Sample analysis of means type randomization test for variances for unbalanced designs.”*Journal of Statistical Computation and Simulation* 74:701–726.

Woodall, W. H., and Adams, B. M. (1998).*Handbook of Statistical Methods for Engineers and Scientists.*2nd ed. Chapter 7.New York: McGraw-Hill.

附录 B

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation.All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

  - You must give any other recipients of the Work or Derivative Works a copy of this License; and
  - You must cause any modified files to carry prominent notices stating that You changed the files; and
  - You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
  - If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
  - You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

可靠性和生存方法

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020–2021。JMP® 16 可靠性和生存方法.Cary, NC: SAS Institute Inc.

JMP® 16 可靠性和生存方法

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

可靠性和生存方法

1 了解 JMP 13

文档和其他资源

JMP 文档中的格式规范 15

JMP 帮助 16

JMP 文档库 16

学习 JMP 的其他资源 22

JMP 教程 22

样本数据表 22

学习统计和 JSL 术语 23

学习 JMP 的提示与技巧 23

JMP 工具提示 23

JMP 用户社区 23

免费在线统计思维课程 23

JMP 新用户欢迎套件 24

统计学知识门户 24

JMP 培训 24

用户编写的 JMP 手册 24

“JMP 起始页”窗口 24

JMP 技术支持 24

2 可靠性和生存介绍 25

寿命和失效分析

3 寿命分布 27

拟合寿命数据的分布

“寿命分布”平台概述 29

“寿命分布”平台的示例 29

启动“寿命分布”平台 32

“寿命分布”报表 34
事件图 35
比较分布 37
寿命分布：统计学 41
“寿命分布”报表选项 46
混合 48
拟合竞争风险混合模型 51
Weibayes 报表 52
“竞争原因”报表 53
竞争原因工作流程 53
竞争原因模型 53
原因组合 54
竞争原因：统计学 55
单个原因 56
“竞争原因”报表选项 56
“寿命分布 - 比较组”报表 57
比较组：统计学 58
单个组 59
“寿命分布 - 比较组”报表选项 59
“寿命分布”平台的更多示例 60
忽略竞争原因 60
更改尺度 62
在不同组之间检查同一分布 63
Weibayes 估计值 65
拟合混合模型示例 67
拟合竞争风险混合模型示例 70
“寿命分布”平台的统计详细信息 72
分布 72
竞争原因详细信息 84
关于中位数秩回归的注释 89
4 以 X 拟合寿命 91
拟合事件时间数据的单因子模型
“以 X 拟合寿命”平台概述 93
“以 X 拟合寿命”平台的示例 93

启动“以 X 拟合寿命”平台 96
“以 X 拟合寿命”报表 98
数据汇总 98
散点图 98
非参数叠加 100
比较 101
结果 104
定制关系 114
“以 X 拟合寿命”平台选项 115
“以 X 拟合寿命”平台的更多示例 116
电容示例 116
定制关系示例 118

5 累积损坏 121
对具有可变应力的产品随时间劣化建模

“累积损坏”平台概述 123
“累积损坏”平台的示例 123
启动“累积损坏”平台 127
事件时间 128
应力模式 129
“累积损坏”报表 131
事件图 131
“应力模式”报表 131
模型列表 132
模型结果 132
“累积损坏”平台选项 133
应力模式选项 133
“累积损坏”平台的更多示例 134
示例：模拟新数据 134

6 复发分析 137
对随时间复发事件的频数或成本建模

“复发分析”平台概述 139
“复发分析”平台的示例 139
启动“复发分析”平台 142

“复发分析”平台选项 143
拟合模型 144
“复发分析”平台的更多示例 147
膀胱癌复发示例 147
Diesel Ship Engines 示例 150

7 退化 155
对产品随时间劣化建模
“退化”平台概述 157
“退化”平台的示例 157
启动“退化”平台 159
退化报表 160
模型规格 162
简单线性路径 163
非线性路径 164
逆预测 172
预测图形 173
“退化”平台选项 174
模型报表 176
模型列表 176
报表 176
破坏性退化 177
稳定性分析 182

8 破坏性退化 185
对产品随时间劣化建模
“破坏性退化”平台的示例 187
启动“破坏性退化”平台 192
指定两个 Y 列 193
“破坏性退化”图选项和模型 193
图选项 194
模型 195
“破坏性退化”报表 198
模型列表 199

“模型”分级显示项 199
“破坏性退化”平台选项 201
“破坏性退化”平台的统计详细信息 202

9 可靠性预测 203
使用生产和失效数据预测产品失效

“可靠性预测”平台概述 205
使用“可靠性预测”平台的示例 205
启动“可靠性预测”平台 208
“可靠性预测”报表 210
“观测数据”报表 210
“寿命分布”报表 211
“预测”报表 211
“可靠性预测”平台选项 215
使用“可靠性预测”平台的其他示例 215

10 可靠性增长 219
随着变更的实现对系统可靠性建模

“可靠性增长”平台概述 221
使用“可靠性增长”平台的示例 222
启动“可靠性增长”平台 225
启动窗口角色 225
数据表结构 227
“可靠性增长”报表 229
模型描述和可行性 229
“观测数据”报表 229
“可靠性增长”平台选项 232
模型报表 233
Crow AMSAA 233
使用修改 MLE 的 Crow AMSAA 239
固定参数 Crow AMSAA 240
分段 Weibull NHPP 241
重新初始化 Weibull NHPP 245
分段 Weibull NHPP 变点检测 247
“可靠性增长”平台的更多示例 248

使用区间删失数据进行分段 NHPP Weibull 模型拟合 248
时间采用日期格式的“分段 Weibull NHPP 变点检测” 250
“可靠性增长”平台的统计详细信息 252
“Crow-AMSAA”报表的统计详细信息 252
“分段 Weibull NHPP 变点检测”报表的统计详细信息 253

11 可靠性框图 255
设计系统可靠性
“可靠性框图”平台概述 257
使用“可靠性框图”平台的示例 257
添加元件 259
对齐形状 261
连接形状 262
配置元件 263
“可靠性框图”窗口 264
“预览”窗口 265
“可靠性框图”平台选项 266
“工作区”选项 269
配置设置 269
分布配置 269
指定非参数分布 271
用于“设计”和“逻辑库”项的选项 272
刻画器 273
元件重要性和失效时间 275
元件图 277
打印代数可靠性公式 279
生成代数表达式数据表 280
复制和删除 280

12 可修复系统模拟 281
估计复杂系统的中断时间
“可修复系统模拟”平台概述 283
示例：使用“可修复系统模拟”平台 283
“可修复系统模拟”窗口 285
系统关系图 285

“形状”工具栏 286
“配置”面板 287
“添加事件”面板 289
“添加操作”面板 290
“可修复系统模拟”平台选项 292
用于方框项的选项 293
分布选项 293
指定非参数或估计分布 294
事件设置 295
操作设置 295
可修复系统模拟结果 296
结果表 296
结果分析器 297
可修复系统模拟结果选项 298
RSS 结果分析器点估计刻画器 299

13 生存分析 301
分析生存时间数据
“生存分析”平台概述 303
生存分析的示例 303
启动“生存”平台 305
生存图 305
“生存”平台选项 306
指数、Weibull 和对数正态图以及拟合 308
拟合分布图 312
竞争原因 312
“生存”平台的更多示例 313
生存分析的示例 313
竞争原因的示例 315
区间删失的示例 317
生存分析的统计报表 318

14 拟合参数生存 321
使用回归模型拟合生存数据
“拟合参数生存”平台概述 323

参数回归生存拟合的示例 323
启动“拟合参数生存”平台 325
“参数生存拟合”报表 326
“参数生存 - 所有分布”报表 328
“参数竞争原因”报表 329
“拟合参数生存”选项 329
非线性参数生存模型 330
拟合参数生存的更多示例 331
Arrhenius 加速失效对数正态模型 331
区间删失的加速失效时间模型 334
使用“非线性”平台分析删失数据 336
左删失数据 336
使用“非线性”平台的 Weibull 损失函数 338
使用“非线性”平台拟合简单生存分布 340
“拟合参数生存”平台的统计详细信息 343
生存分布的损失公式 343

15 拟合比例风险 345
使用半参数回归模型拟合生存数据

“拟合比例风险”平台概述 347
“拟合比例风险”平台的示例 347
具有两个水平的一个名义型效应的风险比 348
启动“拟合比例风险”平台 350
“拟合比例风险”报表 351
“拟合比例风险”选项 352
使用多个效应和多个水平的示例 353
多个效应和多个水平的风险比 355

A 参考书目 357
B 技术许可声明 361

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 15
JMP 帮助 16
JMP 文档库 16
学习 JMP 的其他资源 22
JMP 教程 22
样本数据表 22
学习统计和 JSL 术语 23
学习 JMP 的提示与技巧 23
JMP 工具提示 23
JMP 用户社区 23
免费在线统计思维课程 23
JMP 新用户欢迎套件 24
统计学知识门户 24
JMP 培训 24
用户编写的 JMP 手册 24
“JMP 起始页”窗口 24
JMP 技术支持 24

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助 > JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助 > JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 16 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助 > JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

可靠性和生存介绍

寿命和失效分析

《可靠性和生存方法》介绍了 JMP 中提供的一些方法和工具，这些方法和工具可帮助您评估和改进产品或系统的可靠性，并帮助分析人和产品的生存数据。

- “寿命分布”平台支持您分析产品、元件或系统的寿命以提高质量和可靠性。该分析帮助您确定最适合产品的材料和生产过程，从而提高产品的质量和可靠性。请参见第 3 章，“寿命分布”。
- “以 X 拟合寿命”平台帮助您分析仅有一个因子情况下的寿命事件。您可以选择使用各种变换对事件与因子之间的关系建模，或是创建自定义数据变换。请参见第 4 章，“以 X 拟合寿命”。
- “累积损坏”平台支持您分析应力水平可能随时间变化的加速寿命试验。请参见第 5 章，“累积损坏”。
- “复发分析”平台分析事件时间，这些事件对于每个单元可多次出现。通常情况下，在某个单元发生故障，接受维修，维修后重新投入使用时就会发生这些事件。请参见第 6 章，“复发分析”。
- “退化”平台可分析退化数据以预测伪失效时间。随后可在其他可靠性平台中分析这些伪失效时间，以便估计失效分布。您可以包括解释因子，可以执行稳定性分析以设置产品失效日期，还可以拟合定制破坏性退化模型。请参见第 7 章，“退化”。
- “破坏性退化”平台可对产品特征的失效数据建模，测量这些数据需要破坏产品。这导致每个产品单元都有一个观测。您还可以包括加速因子。有多个通用退化模型可用。请参见第 8 章，“破坏性退化”。
- “可靠性预测”平台可帮助您预测将来失效的次数。该分析使用生产日期、失效日期和产量来估计寿命分布的参数。请参见第 9 章，“可靠性预测”。
- “可靠性增长”平台可随着单个可修复系统不断接受系统设计的改进而对系统可靠性的变化建模。请参见第 10 章，“可靠性增长”。
- JMP PRO “可靠性框图”平台显示系统元件之间的可靠性关系；若为元件指定了可靠性分布，则通过分析来获取可靠性行为。请参见第 11 章，“可靠性框图”。
- JMP PRO “可修复系统模拟”平台支持您以交互方式定义可修复系统元件之间的关系。它还可以模拟系统的关闭时间。请参见第 12 章，“可修复系统模拟”。
- “生存”平台计算一个或多个组的生存估计值。可将其用作完整分析或用作探索性分析，以便获取关于更复杂的模型拟合的信息。请参见第 13 章，“生存分析”。
- “拟合参数生存”平台使用可涉及位置和尺度效应的线性回归模型拟合事件时间变量。可使用若干分布执行该拟合。请参见第 14 章，“拟合参数生存”。

- “拟合比例风险”平台拟合 Cox 比例风险模型，该模型假定预测变量与危险率函数之间存在相乘关系。请参见第 15 章，“拟合比例风险”。

第 3 章

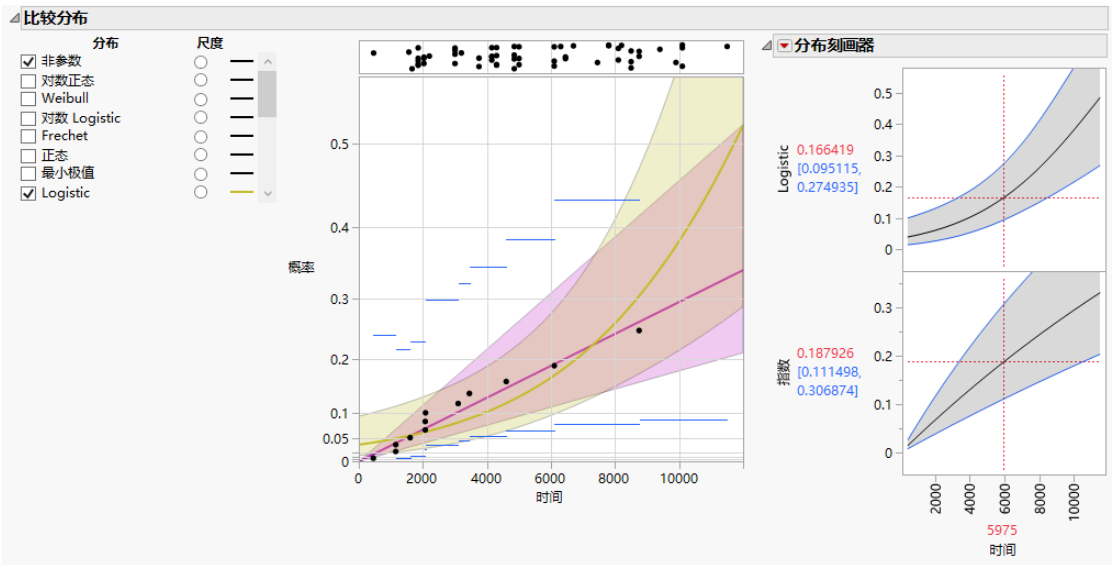
寿命分布

拟合寿命数据的分布

“寿命分布”平台对事件时间数据建模。该平台可容纳右删失和区间删失数据。使用“寿命分布”可执行以下操作：

- 比较多个分布拟合以确定哪一种分布对您的数据拟合效果最好。
- 构造 Bayes 拟合。
- 对零失效数据建模。
- 比较各组以分析组差异。
- 分析多种失效原因。
- 估计混合模型的成分并估计某个观测来自给定成分的概率。
- 估计竞争风险混合模型的成分并估计成分对观测的影响。

图 3.1 分布拟合和比较



目录

“寿命分布”平台概述 29
“寿命分布”平台的示例 29
启动“寿命分布”平台 32
“寿命分布”报表 34
事件图 35
比较分布 37
寿命分布：统计学 41
“寿命分布”报表选项 46
混合 48
拟合竞争风险混合模型 51
Weibayes 报表 52
“竞争原因”报表 53
竞争原因工作流程 53
竞争原因模型 53
原因组合 54
竞争原因：统计学 55
单个原因 56
“竞争原因”报表选项 56
“寿命分布 - 比较组”报表 57
比较组：统计学 58
单个组 59
“寿命分布 - 比较组”报表选项 59
“寿命分布”平台的更多示例 60
忽略竞争原因 60
更改尺度 62
在不同组之间检查同一分布 63
Weibayes 估计值 65
拟合混合模型示例 67
拟合竞争风险混合模型示例 70
“寿命分布”平台的统计详细信息 72
分布 72
竞争原因详细信息 84
关于中位数秩回归的注释 89

“寿命分布”平台概述

寿命数据分析或**寿命分布分析**是对某个产品、元件或系统的寿命建模以预测寿命或失效时间的过程。例如，您可以观察随时间变化的失效率，以预测某个计算机元件何时失效。该方法支持您比较产品的材料和生产过程，支持您提高该产品的质量和可靠性。针对保修期和广告诉求的决策也会更加准确。

利用“寿命分布”平台，您可以分析某些时间观测未知的**删失**数据。而且，若失效可能是多种原因造成的，您可以分析**竞争原因**以估计哪一种原因影响更大。

您可以使用**可靠性检验计划**和**可靠性验证**计算器为可靠性研究选择适当的样本大小。这些计算器位于**实验设计 > 样本大小与功效**中。请参见《实验设计指南》。

“寿命分布”平台的示例

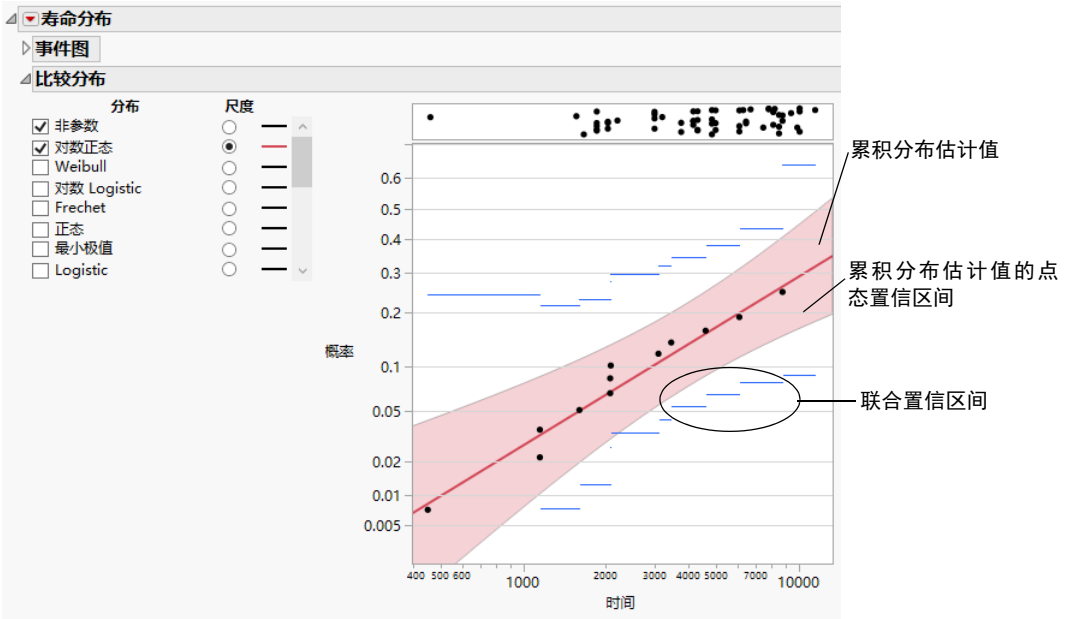
假定您有 70 台发动机风扇的失效时间数据，其中有些失效时间删失。您想要拟合失效时间的分布，然后估计各种可靠性测量值。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Fan.jmp。
2. 选择**分析 > 可靠性和生存 > 寿命分布**。
3. 选择**时间**并点击 **Y，事件时间**。
4. 选择**删失**并点击**删失**。
5. 点击**确定**。

   “寿命分布”报表窗口随即显示。
6. 在 “比较分布”报表中，选择**对数正态**分布和相应的**尺度**单选按钮。

   报表窗口中会显示一个概率图。

图 3.2 概率图



在概率图中，数据点通常落在红线附近，指出对数正态拟合是合理的。

“比较分布”报表下显示“统计量”报表。该报表提供模型比较报表、非参数和参数估计值、刻画器等内容。

图 3.3 “统计学”报表

提供对数正态分布的参数估计值。刻画器有助于直观演示拟合分布以及估计概率和分位数。例如，“分位数刻画器”指出估计的失效时间中位数为 25,418.67 小时。

启动“寿命分布”平台

通过选择**分析 > 可靠性和生存 > 寿命分布**来启动“寿命分布”平台。有关“选择列”红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。

图 3.4 “寿命分布”启动窗口

启动窗口选项卡

启动窗口包括两个选项卡：

- “寿命分布”选项卡对未分组数据建模。可生成以下类型的报表：
  - 若您未指定“失效原因”角色，则显示“寿命分布”报表。从该报表中，您可以比较常用分布并检查统计量。请参见第 34 页的““寿命分布”报表”。
  - 若您的数据包含零失效，则显示 Weibayes 报表。请参见第 52 页的“Weibayes 报表”。
  - 若您指定了“失效原因”角色，则显示“竞争原因”报表。除了“寿命分布”报表中的功能，您还可以比较单个失效原因。请参见第 53 页的““竞争原因”报表”。

注意：您可以检查“寿命分布”报表和“竞争原因”报表中的固定参数和 Bayes 模型。

- “比较组”选项卡支持您指定“分组”变量。“比较组”报表使用单个指定分布比较不同的组。例如，您可以对按供应商分组的元件进行 Weibull 拟合比较。相比之下，“寿命分布”选项卡比较的是单个组的若干拟合分布。请参见第 57 页的““寿命分布 - 比较组”报表”。

启动窗口选项

启动窗口包含以下选项：

Y，事件时间 一个或多个响应列。指定的响应列数取决于数据表中的删失结构。

- 若指定一个变量，它被解释为事件时间（如，失效时间）或右删失时间。使用“删失”列可指示右删失响应。有关右删失的详细信息，请参见第 36 页的“单个事件时间列”。
- 若指定两个变量，这两个变量将被解释为区间删失观测。第一个 Y 变量提供下限，第二个 Y 变量提供每个单元的上限。有关使用两个响应列表示不同类型删失的示例，请参见图 3.5。有关使用两个响应列的删失的详细信息，请参见第 36 页的“两个事件时间列”。

图 3.5 两个响应变量的删失数据类型

	开始时间	结束时间	计数	删失类型
1	•	50	50	Left
2	•	100	6	Left
3	25	•	30	Right
4	75	•	10	Right
5	80	150	4	Interval
6	100	250	7	Interval

- 若指定三个或更多变量，报表会包含单独的分析，该分析是将每个指定变量用作事件时间数据计算得出的。

分组 （仅显示在“比较组”选项卡中。）您要比较的各组所在的列。有关示例，请参见第 63 页的“在不同组之间检查同一分布”。

删失 标识右删失观测的列。从“选择列”列表下方的“删失代码”菜单中选择用于标识右删失观测的值。仅当输入一个 Y 时才使用“删失”列。

失效原因 包含多个失效原因的列。若选择了**失效原因**列，则会向该窗口添加一个部分。该部分包含多个复选框，这些复选框允许失效模式使用零泛滥分布、阈值分布、缺陷子总体分布、固定参数模型或 Bayes 模型进行分析。以下选项也可用：

分布 指定为每个失效原因拟合的初始分布。选择一种要为所有原因拟合的分布；选择**单个最佳**可让平台为每个原因自动选择最佳拟合；选择**手动选择**可在 JMP 创建“寿命分布”报表后为每个失效原因手动选择要拟合的分布。您还可以在“寿命分布”报表本身中更改分布拟合。

比较准则 （仅当您选择**单个最佳**分布拟合时才显示。）指定 JMP 选择最佳分布的方法：校正 Akaike 信息准则 (AICc)、Bayes 信息准则 (BIC) 或 -2 倍的对数似然（-2 对数似然）。请参见《拟合线性模型》。您可以随后在“模型比较”报表中更改该方法。请参见第 41 页的“模型比较”。

失效原因列中的删失指示符 标识在**失效原因**列中用来指示观测未失效的值。要指定这样的指示符，请选择该选项，然后在出现的框中输入指示符。您还可以从框右侧的列表中选择值。

有关多重失效原因的讨论，请参见 Meeker and Escobar (1998, ch. 15)。第 60 页的“忽略竞争原因”演示了如何分析多种原因。

频数 当行中的信息表示多个单元时，用于包含频数或观测计数的列。若行中的值为 0 或正数，则这个值表示该行的频数或观测计数。

标签 包含行号之外的标识符的列。这些标签显示在事件图中的 *y* 轴上。

依据 一个可选变量，其水平定义的行用于创建单独的模型。

删失代码 标识“删失”列中指定右删失观测的值。选定某个“删失”列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中进行选择。您还可以在框中输入不同值。若 “删失”列包含 “值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

选择置信区间方法 定义用于计算参数置信区间的方法。默认为 Wald，但您可以改为选择“似然”。不过，刻画器中提供的所有置信区间都基于 Wald 方法。这是为了减少计算时间。请参见第 72 页的 “估计和置信区间”。

按原因的失效分布 （仅当指定 “原因”时才显示在 “寿命分布”选项卡中。）指定哪些系列的分布应可用于对单个原因的寿命分布建模。从 “分布”菜单中选择初始分布、“单个最佳”或 “手动选择”。请参见第 33 页的 “失效原因”。

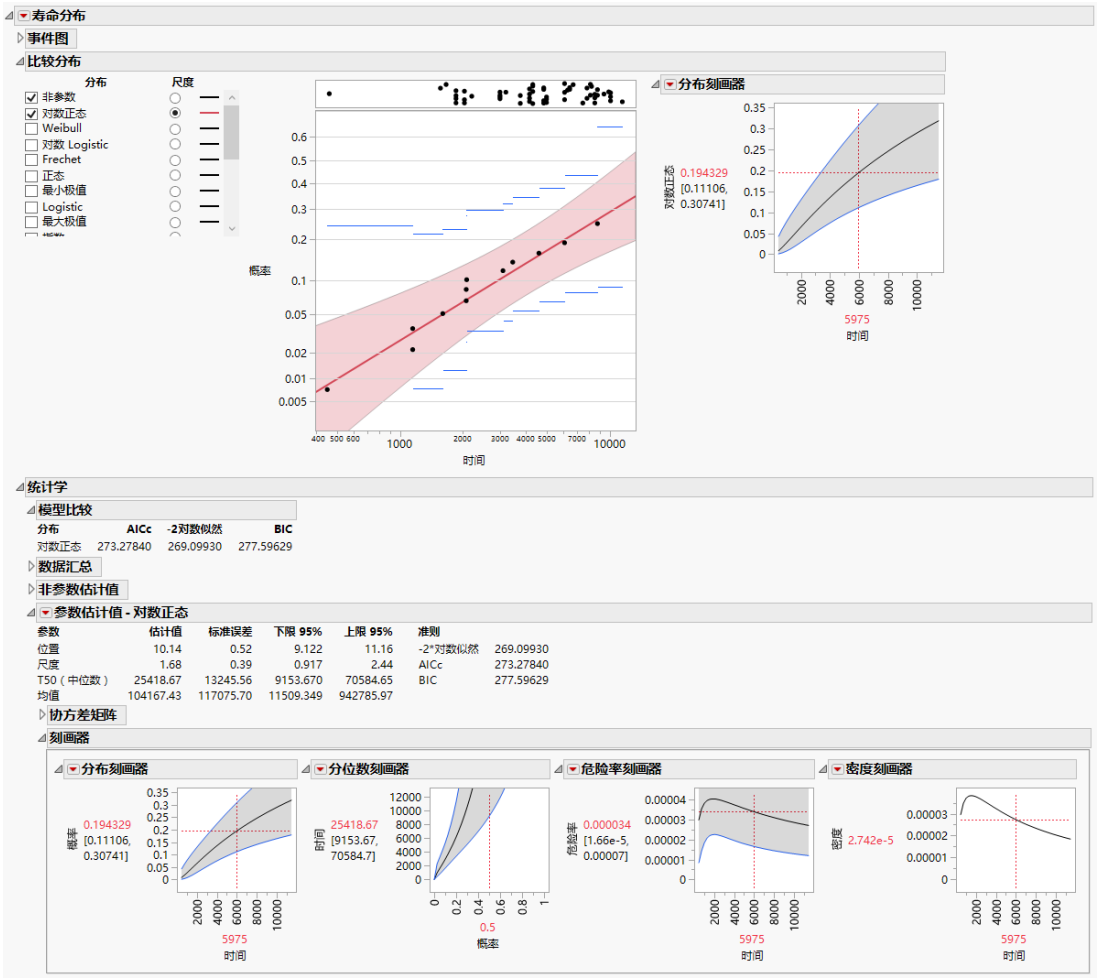
“寿命分布”报表

提示：若您发现报表窗口过长，请从“寿命分布”红色小三角菜单中选择**选项卡式报表**。

若未在启动窗口中选择 “失效原因”，则会显示 “寿命分布”报表。使用该报表可分析可能有一些时间观测删失的寿命数据。“寿命分布”报表包含以下内容和选项：

- 第 35 页的“事件图”
- 第 37 页的“比较分布”
- 第 41 页的“寿命分布：统计学”
- 第 46 页的“ “寿命分布”报表选项”

图 3.6 Fan.jmp “寿命分布”报表的示例



事件图

点击“事件图”展开图标可查看失效或删失时间图。对于数据表中的每一行，“事件图”都会显示一条水平线，指出行中的单元是否删失。若单元删失，水平线会指出删失的性质。

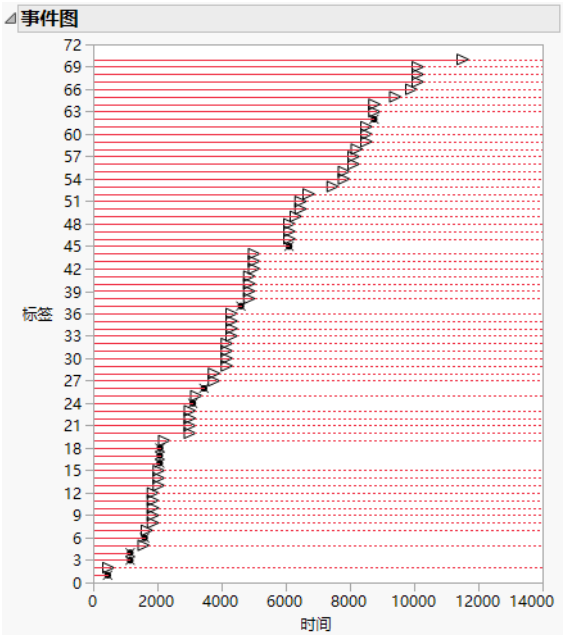
- 已知行中的单元正常运行时的时间期间用实线表示。
- 不清楚行中的单元是否正常运行时的时间期间用虚线表示。
- 一旦已知行中的单元失效，水平线即终止。

单个事件时间列

在 Fan.jmp 样本数据表中，有一个**时间**列指示失效时间。若失效时间未知，则在**删失**列中记录值“Censored”。所有删失单元都被认定为右删失。图 3.7 显示了这些数据的事件图。

注意：要构造图 3.7 中的图，请选择**帮助 > 样本数据库**并打开 Reliability/Fan.jmp。点击**寿命分布 — 指数**脚本旁边的绿色小三角。点击**事件图**展开图标。

图 3.7 右删失数据的事件图



第 3 行中的单元在**时间** 1150 处失效。它的寿命通过在“时间”1150 处终止的一条水平实线表示。失效时间用“x”标记。

第 5 行中的单元右删失。已知它正常运行的最后**时间**为 1560。已知单元正常运行时的时间期间通过在“时间”1560 处终止的一条水平实线来表示。在“时间”1560 处标绘了一个右箭头。该水平线继续以虚线表示，指示失效时间未知但大于 1560。

两个事件时间列

Censor Labels.jmp 样本数据表中有两列：**开始时间**和**结束时间**。**开始时间**指示行中的单元最后知道的正常运行时间。**结束时间**指示该行中的单元最早知道的失效时间。

开始时间和**结束时间**值指示关于单元的以下信息：

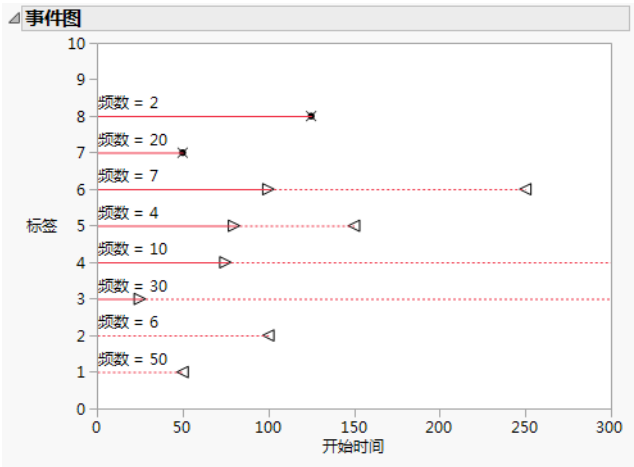
- 第 1 行和第 2 行中的单元左删失。已知这些单元在**结束时间**列中的时间之前失效，但其精确失效时间未知。

- 第 3 行和第 4 行中的单元右删失。已知这些单元最后正常运行的时间是**开始时间**列中的时间，但其失效时间未知。
- 第 5 行和第 6 行中的单元区间删失。已知这些单元在**开始时间**与**结束时间**定义的区间内失效。
- 第 7 行和第 8 行中的单元未删失。它们的失效时间由**开始时间**和**结束时间**列中的值指定，这两个值是相同的。

图 3.8 显示了这些数据的事件图。

注意：要构造图 3.8 中的图，请选择**帮助 > 样本数据库**并打开 Censor Labels.jmp。点击**寿命分布**脚本旁边的绿色小三角。

图 3.8 混合删失数据的事件图



“事件图”中的线条模式表示各种类型的删失：

- 模式 指示右删失。单元在其最后一次检查时失效。
- 模式 指示左删失。单元在开始检验之后而在指示的时间之前失效，但不清楚其最后有效的时间。
- 模式 指示单元在两个箭头标记的时间区间内失效。
- 模式 指示无删失。单元在 x 标记的时间失效。

比较分布

“比较分布”报表支持您拟合和比较不同的失效时间分布。显示两个列表：

分布 选择响应的分布。根据数据特征显示不同的分布。有关可用分布的详细信息，请参见第 39 页的 “可用的参数分布”。

尺度 选择概率轴的尺度。概率尺度对应于 “尺度”按钮左侧列出的分布。使用该尺度，拟合模型通过一条线来表示。假定您拟合给定的分布，随后使用该分布统一轴的尺度。通常情况下，若各点沿线条分布，这指示该分布提供了合理的拟合。

默认图显示未删失数据的非参数估计值 (Kaplan-Meier-Turnbull) 及其置信区间。置信区间通过水平蓝线来表示。

提示：要定制非参数估计值图，请选择**文件 > 首选项 > 平台 > 寿命分布**，然后选择一个或多个以下首选项：**显示着色点态区间**、**显示着色联合区间**或**显示楼梯样式函数**。

默认情况下，图顶部有一个面板显示右删失观测的时间。

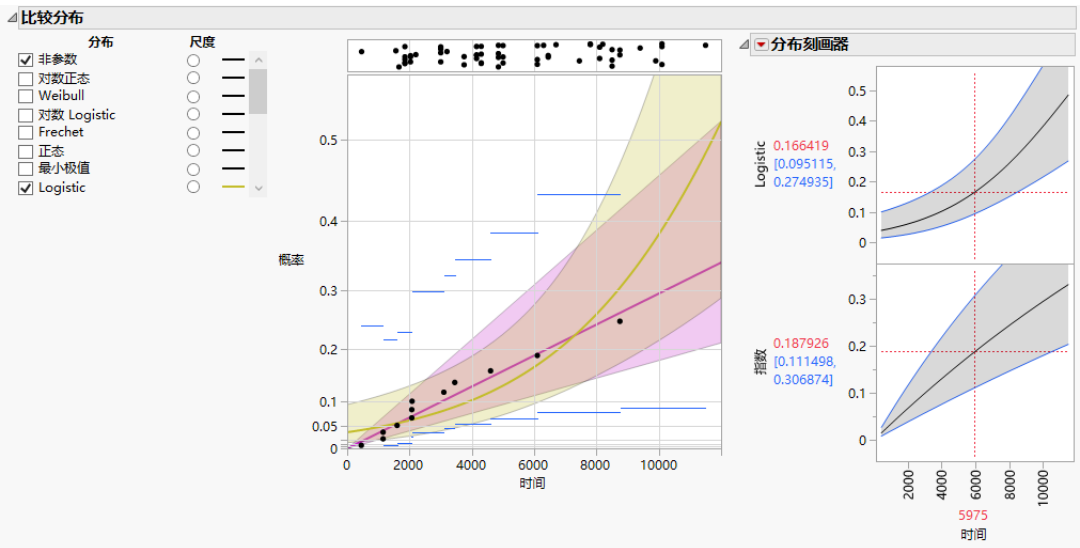
提示：要隐藏包含右删失观测的标记的面板，请选择**文件 > 首选项 > 平台 > 寿命分布**，然后取消选中**为右删失观测显示标记**。

对于您选择的每个分布，“比较分布”报表会随之更新以显示以下内容：

- 估计的累积分布曲线，显示在概率图上
- 着色区域，指示累积分布的置信区间
- 分布刻画器，显示给定时间期间的失效累积概率

图 3.9 显示了“比较分布”报表的示例，其中显示了 Logistic（黄色）和指数（品红色）分布。该图使用“指数”分布统一尺度。

图 3.9 “比较分布”报表和 “分布刻画器”



可用的参数分布

本节说明“比较分布”报表中提供的分布。

注意：针对“竞争原因”报表的分布在第 56 页的“可用于竞争原因比较分布报表的分布”中说明。

可用的分布在第 73 页的“参数分布”中列出并进行了详细说明。共有四组主要的参数分布：

- 第 39 页的“基本失效时间分布”
- 第 40 页的“阈值分布”
- 第 40 页的“缺陷子总体分布”
- 第 40 页的“零泛滥分布”

提示：要限制默认情况下哪些分布可用，请选择**文件 > 首选项 > 平台 > 寿命分布**，然后取消选中您不想显示的分布。列出的分布包括“阈值”、“缺陷子总体”、“零泛滥”、“对数广义 Gamma”和“广义 Gamma”分布。默认情况下选中所有分布，因此所有分布都可用。

用于确定哪些分布显示在“比较分布”面板中的规则取决于特定的实现方式。基本的准则是：若这些分布未在“首选项”中禁用且在给定情形下适用，那么它们就可用。

基本失效时间分布

只要所有失效时间都为正，基本失效时间分布即可用。这些分布包括：

- 对数正态
- Weibull
- 对数 Logistic
- Fréchet
- 正态
- 最小极值
- Logistic
- 最大极值
- 指数
- 对数广义 Gamma
- 广义 Gamma

注意：若存在负或零失效时间，则只有“正态”、“最小极值”、“Logistic”、“最大极值”和“对数广义 Gamma”可用。

阈值分布

阈值 (TH) 分布始终可用。阈值分布是带有阈值参数的对数 - 位置 - 尺度分布。阈值参数使分布偏离 0。这些分布假定所有单元一直生存到阈值所示的时间。阈值分布适用于拟合中度到重度偏移的分布。阈值分布包括：

- 阈值对数正态
- 阈值 Weibull
- 阈值对数 Logistic
- 阈值 Fréchet

缺陷子总体分布

当所有失效时间均为正时，缺陷子总体 (DS) 分布可用。这些分布适用于只有一部分总体具有导致失效的特定缺陷的情况。使用缺陷子总体分布选项可对仅在子总体上出现的失效建模。缺陷子总体分布包括：

- 缺陷子总体对数正态
- 缺陷子总体 Weibull
- 缺陷子总体对数 Logistic
- 缺陷子总体 Fréchet

零泛滥分布

当事件时间数据以零作为“寿命分布”平台中的最小值时，以下零泛滥分布可用：

- 零泛滥对数正态（ZI 对数正态）
- 零泛滥 Weibull (ZI Weibull)
- 零泛滥对数 Logistic（ZI 对数 Logistic）
- 零泛滥 Fréchet (ZI Fréchet)

当一定比例的单元在时间 0 失效时，可使用零泛滥分布。当数据包含零的个数超出标准模型应有的个数时，就会造成零泛滥。

零失效数据

在零失效数据的情况下，以上分布默认均不可用。要获取可使用“Bayes 估计”选项的那些分布的 Bayes 拟合，请选择**文件 > 首选项 > 平台 > 寿命分布**，然后取消选中 **Weibayes 仅适用于零失效数据**。请参见第 45 页的“Weibayes 仅适用于零失效数据”。

允许 Bayes 估计的参数分布

Bayes 估计可用于以下参数分布：

- 对数正态

- Weibull
- 对数 Logistic
- Fréchet
- 正态
- 最小极值
- Logistic
- 最大极值

第 83 页的“Bayes 估计的先验分布”中提供了可用作这些分布的超参数的先验分布的分布列表。

寿命分布：统计学

“统计学”报表包括以下子报表：

- 第 41 页的“模型比较”
- 第 41 页的“数据汇总”
- 第 41 页的“非参数估计值”
- 第 42 页的“参数估计值 - <分布名称>”（为您在“比较分布”报表中选择的每个分布显示一个报表）

模型比较

“模型比较”报表为每个拟合分布都提供 AICc、-2 对数似然和 BIC 统计量。以上每个统计量的值越小指示拟合效果越好。有关这些统计量的详细信息，请参见《拟合线性模型》。

最初，这些行按 AICc 排序。要更改用于对该报表排序的统计量，请点击“寿命分布”红色小三角并选择**比较准则**。有关该选项的详细信息，请参见第 46 页的“ “寿命分布”报表选项”。

数据汇总

“数据汇总”报表显示观测的总单元数、未删失单元数，以及右删失、左删失和区间删失单元的数目。

非参数估计值

“非参数估计值”报表显示每个观测的非参数估计值。对于指定为单个 “事件时间”列的右删失数据，该报表提供以下信息：

中点估计 已调整中点的 Kaplan-Meier 估计值。

95% 下限， 95% 上限 点态 95% 置信区间。您可以通过从报表选项中选择 “更改置信水平”来更改置信水平。

联合 95% 下限 (Nair)，联合 95% 上限 (Nair) 联合 95% 置信区间。您可以通过从报表选项中选择“更改置信水平”来更改置信水平。请参见 Nair (1984) 和 Meeker and Escobar (1998)。

Kaplan-Meier 估计值 标准 Kaplan-Meier 估计值。

若失效时间由两个“事件时间”列表示，该报表提供 Turnbull 估计值（在称为“估计值”的列中）、点态置信区间和联合置信区间 (Nair)。

有关非参数估计值的详细信息，请参见第 73 页的“非参数拟合”。

参数估计值 - < 分布名称 >

为拟合的每个分布都显示称为“参数估计值 - <分布名称>”的报表。该报表提供分布的参数估计值、估计值的标准误差和置信区间。出现在“模型比较”报表中的准则会显示在“准则”下方。

注意：只要提供均值估计值，它的置信区间就计算为 Wald 区间，即便您在启动窗口中将“似然”选作“置信区间方法”也是如此。在这种情况下，符号“均值（Wald 置信区间）”会显示在“参数”列中，指示均值的置信区间为 Wald 区间。

有关如何将分布参数化的详细信息，请参见第 73 页的“参数分布”。

“参数估计值”报表包含以下报表：

- 第 42 页的“协方差矩阵”
- 第 42 页的“刻画器”
- 通过从“参数估计值”红色小三角菜单中选择报表选项，可添加其他报表。其中包括“固定参数”、“Bayes 估计”、“定制估计”（估计概率、估计分位数）和“剩余寿命均值”报表。请参见第 43 页的“‘参数估计值’选项”。

协方差矩阵

对于每个分布，“协方差矩阵”报表都显示估计值的协方差矩阵。

刻画器

为每个分布显示四种刻画器：

- “分布刻画器”显示作为时间的函数的累积失效概率。
- “分位数刻画器”显示作为累积概率的函数的失效时间。
- “危险率刻画器”显示作为时间的函数的危险率。
- “密度刻画器”显示分布的密度函数。

刻画器包含以下红色小三角选项：

置信区间 “分布”、“分位数”和“危险率”刻画器显示所绘函数的基于 Wald 的置信曲线。该选项显示或隐藏置信曲线。

重置因子网格 为每个因子显示一个窗口，允许您为因子的当前设置输入特定值，锁定该设置，以及控制网格的各方面。请参见 《刻画器指南》。

因子设置 提供包含若干选项的菜单。请参见 《刻画器指南》。

注意：刻画器中提供的置信区间基于 Wald 方法，即便您在启动窗口中将“似然”选作“置信区间方法”也是如此。这是为了减少计算时间。

“参数估计值”选项

“参数估计值”红色小三角菜单包含以下选项：

保存概率估计值 将估计的失效概率和置信区间保存至数据表。

保存分位数估计值 将估计的分位数和置信区间保存至数据表。

保存危险率估计值 将估计的危险率值和置信区间保存至数据表。

显示似然等高线 显示或隐藏对数似然函数的等高线图。若已选择 Weibull 分布，则针对 alpha-beta 参数化显示另外一个等高线图。该选项仅适用于具有两个参数的分布。

显示似然刻画器 显示或隐藏对数似然函数的刻画器。该选项不适用于阈值 (TH) 分布。

固定参数 打开一个报表，可在其中指定参数的值。输入固定参数值，选中对应的复选框，然后点击**更新**。JMP 基于新参数重新估计其他参数、协方差和刻画器，并在 “固定参数”报表中显示它们。无约束模型的分布刻画器显示在固定参数模型的分布刻画器下方。有关竞争原因情形下的示例，请参见第 84 页的 “将固定参数模型指定为原因的分布”。

对于 Weibull 分布，**固定参数**选项支持您选择 Weibayes 方法。有关示例，请参见第 65 页的 “Weibayes 估计值”。Weibayes 选项不适用于区间删失数据。

Bayes 估计 为基于先验分布的三种指定方法（先验位置和尺度、先验分位数和参数以及先验失效概率）的特定分布执行 Bayes 参数估计。请参见第 43 页的 “Bayes 估计 - < 分布名称 >”。该选项仅适用于以下分布：对数正态、 Weibull、对数 Logistic、 Fréchet、正态、最小极值、Logistic、最大极值。

定制估计 提供计算器来支持您预测失效概率、生存概率以及特定时间和失效概率值的分位数。每个计算的数量都包括置信区间，置信区间可以是双侧也可以是单侧（任意方向）。显示两个报表：“估计概率”和 “估计分位数”。请参见第 46 页的 “定制估计”。

剩余寿命均值 提供一个计算器来支持您估计单元的剩余寿命均值。在“剩余寿命均值计算器”中，输入 “时间”并按 Enter 键以查看估计值。点击加号可输入其他时间。该计算器仅适用于以下分布：对数正态、Weibull、对数 Logistic、Fréchet、正态、最小极值、Logistic、最大极值和指数。

Bayes 估计 - < 分布名称 >

对于特定分布，您可以拟合 Bayes 模型。这是使用拒绝抽样或 Markov Chain Monte Carlo (MCMC) 算法完成的。具体而言，该平台尝试一个基本的拒绝抽样器。若拒绝抽样器生成有效

结果，则报告这些结果。若拒绝抽样器不能生成有效的结果，则平台使用随机漫步 Metropolis-Hastings 算法，并在“**Bayes** 估计”报表的顶部添加注释。请参见 Robert and Casella (2004)。

从“参数估计值 - <分布名称>”报表分级显示项中，选择 **Bayes 估计**。这将打开名为“Bayes 估计 - <分布名称>”的分级显示项。初始报表是一个控制面板，您可在其中指定先验参数并控制模拟的各个方面。

以下步骤对该工作流程进行了说明：

- 从“Bayes 估计”红色小三角菜单中选择一种先验规格方法，并设置先验参数值。请参见第 44 页的““Bayes 估计”红色小三角选项”。
- 指定模拟选项。请参见第 45 页的“Bayes 估计 - 结果 <N>”。
- 选择“拟合模型”以拟合模型。请参见第 45 页的“Bayes 估计 - 结果 <N>”。

“Bayes 估计”红色小三角选项

您可以从“Bayes 估计”红色小三角菜单中选择以下先验规格方法：

先验位置和尺度 支持您为一般参数（位置和尺度参数）的先验分布指定超参数。选择“先验分布”红色小三角菜单以便为每个参数选择一种分布。您可以为先验超参数输入新值。提供的初始值是与 MLE 一致的估计值。请参见第 83 页的“Bayes 估计的先验分布”。

先验分位数和参数 支持您指定有关分位数和尺度参数（若参数拟合为 Weibull，则指定 Weibull β）的先验信息。分位数由“概率”旁边的值定义。“概率”默认值为 0.10，但您可以指定对应于所关注分位数的值。采用每个先验分布的范围的 99% 上下限的形式，指定有关先验信息的信息。请参见 Meeker and Escobar (1998)。提供的初始值是与 MLE 一致的估计值。请参见第 83 页的“Bayes 估计的先验分布”。

先验失效概率 支持您指定有关两个不重复时间点的失效概率的先验信息。您可以指定两个时间点。每个时间点的先验分布为 Beta 分布。您可以使用以下两种同步方法之一指定先验分布：

1. 按估计值和误差百分比指定失效概率。可使用概率估计值和估计值误差来指定每个 Beta 先验分布的先验信息。请参见 Kaminskiy and Krivtsov (2005)。
2. 指定失效概率估计值范围。您可以采用以下方式指定两个 Beta 分布的 99% 范围：

– 对于每个失效时间，输入 99% 下限与 99% 上限的初始值。

– 点击图中的垂直线段并将它们拖放到您的两个时间点。调整每个标记的垂直散布以指定 99% 限值。

模拟选项

对于您在“Bayes 估计”红色小三角菜单中选择的任何先验规格方法，都会在面板底部显示以下选项：

Monte Carlo 迭代次数 控制老化过程后从后验分布中抽取的样本大小。

随机种子 设置模拟的初始状态。默认情况下，它是时钟时间。该数字应为大于 1 的正整数。若指定 1，则使用当前时钟时间。

显示先验散点图 选择该选项可从先验分布中抽取随机样本并在散点图中标绘结果。选择 “拟合模型”后，该散点图将显示在“Bayes 估计 - 结果 <N>”报表中名为“先验散点图”的分级显示项中。

叠加似然等高线 在 “Bayes 估计结果”报表中的散点图上叠加基于似然的等高线。

拟合模型 基于 JMP 使用您指定的值拟合的先验分布，估计后验寿命分布。添加名为 “Bayes 估计 - 结果 <N>”的报表，其中， N 是一个对 “Bayes 结果”报表连续编号的整数。

Bayes 估计 - 结果 <N>

一旦使用某一个红色小三角菜单选项指定了先验值，请选择“拟合模型”。会为每个先验选择都提供一个“Bayes 估计 - 结果 <N>”报表。该报表包含以下标题：

先验 记录您为拟合 Bayes 模型而在“Bayes 估计”报表中输入的指定内容。“先验”报表还指定随机种子。

后验估计 显示五个边缘统计量，这些统计量描述一般参数 （位置和尺度参数）的后验分布。边缘统计量包括：中位数、0.025 分位数（下限）、0.975 分位数（上限）、均值以及从 Monte Carlo 样本中计算得出的标准差。使用 “先验分位数和参数”规格生成后验估计值时，该表还包括分位数和斜率 β （对于 Weibull 分布）的后验估计值。

要基于一般参数的后验估计值计算其他派生变量的统计量，请点击 “导出 Monte Carlo 样本”链接。

先验散点图 当您在点击 “拟合模型”前选择 “显示先验散点图”时出现。显示与分布的先验规格方法关联的参数或等价量的先验散点图。

后验散点图 显示与分布的先验规格方法关联的参数或等价量的后验散点图。

刻画器 显示基于后验分布中的样本的两个刻画器。

“分布刻画器”中显示的在给定时间 *t* 的值计算如下：

- 对于后验分布中的每组抽样参数值，计算累积分布函数在时间 *t* 的值。
- 预测值是这些计算值的中位数。
- 置信上限和置信下限分别为这些计算值的 0.025 和 0.975 分位数。

“分位数刻画器”中显示的图和置信限以类似方式获取。对于给定的 “概率”值 *p*，从与后验参数值关联的分布中计算与 *p* 对应的分位数。

Weibayes 仅适用于零失效数据

在零失效情形下，没有任何单元失效。所有观测都右删失。若您有零失效数据，则可能执行 Bayes 估计或 Weibayes 推断。请参见第 52 页的“Weibayes 报表”。

注意：默认情况下，使用 Weibayes 方法分析零失效数据。若想要对零失效数据执行更广泛的 Bayes 分析，请选择**文件 > 首选项 > 平台 > 寿命分布**，然后取消选中 **Weibayes 仅适用于零失效数据**。

定制估计

“定制估计”选项生成两个报表：“估计概率”和“估计分位数”。“估计概率”报表包含一个计算器，该计算器支持您预测特定时间值的失效和生存概率。“估计分位数”报表包含一个计算器，该计算器支持您预测特定失效概率值的分位数。为每个估计的数量都同时显示基于 Wald 和基于似然的置信区间。这些区间的置信水平由“寿命分布”红色小三角菜单中的更改置信水平选项确定。

估计概率

在“估计概率”计算器中，输入“时间”的值。按 Enter 键可查看失效概率和生存概率的估计值以及相应的置信区间。要计算多个概率估计值，请点击加号，在框中输入另一个“时间”值，然后按 Enter 键。点击减号可删除最后一个条目。

“估计概率”计算器包含一个选项 “侧”，该选项支持您更改区间格式。选择以下子选项之一：

双侧 提供失效概率和生存概率的双侧置信区间。

失效概率上限 提供包含失效概率上限和生存概率下限的单侧置信区间。

失效概率下限 提供包含失效概率下限和生存概率上限的单侧置信区间。

估计分位数

在“估计分位数”报表中，输入“失效概率”的值。按 Enter 键可查看分位数估计值和相应的置信区间。要计算多个分位数估计值，请点击加号，在框中输入另一个值，然后按 Enter 键。点击减号可删除最后一个条目。

“估计分位数”计算器包含一个选项“侧”，该选项支持您更改区间格式。选择以下子选项之一：

双侧 提供分位数的双侧置信区间。

下限 提供包含分位数下限的单侧置信区间。

上限 提供包含分位数上限的单侧置信区间。

“寿命分布”报表选项

“寿命分布”红色小三角菜单包含以下选项：

拟合所有分布 拟合除阈值 (TH) 分布之外的其他所有分布。在“模型比较”报表中比较各个分布。请参见第 37 页的 “比较分布”。

提示：选择**比较准则**选项可更改用于查找最佳分布的准则。

拟合所有非负的分布 拟合所有非负分布（指数、对数正态、对数 Logistic、Fréchet、Weibull 和广义 Gamma）。在“模型比较”报表中比较各个分布。请参见第 37 页的“比较分布”。请注意以下事项：

- 该选项不拟合缺陷子总体分布或阈值分布。
- 若数据包含负值，则该选项不会生成任何结果。
- 若数据包含零，该选项会拟合四种零泛滥 (ZI) 分布：零泛滥对数正态、零泛滥 Weibull、零泛滥对数 Logistic 和零泛滥 Fréchet。有关零泛滥分布的详细信息，请参见第 82 页的“零泛滥分布”。

拟合所有缺陷子总体分布 拟合所有缺陷子总体 (DS) 分布：缺陷子总体对数正态、缺陷子总体 Weibull、缺陷子总体对数 Logistic 和缺陷子总体 Fréchet。有关缺陷子总体分布的详细信息，请参见第 82 页的“缺陷子总体分布”。

拟合混合模型 拟合除阈值 (TH) 分布以外的各种分布混合而成的分布。请参见第 48 页的“混合”。

拟合竞争风险混合模型 拟合数据的竞争风险混合分布。请参见第 51 页的“拟合竞争风险混合模型”。

显示点 显示或隐藏概率图中的数据点。“寿命分布”平台使用阶梯函数的中点估计值来构造概率图。取消选择**显示点**时，中点估计值会被 Kaplan-Meier 估计值取代。

显示事件图频数标签 （仅当指定了“频数”变量时才显示。）显示或隐藏“事件图”中的“频数”标签。

显示生存曲线 在“比较分布”概率图和“分布刻画器”图上的失效概率和生存曲线之间切换。

显示分位数函数 显示或隐藏叠加了选定分布所对应图的“分位数刻画器”。分位数图还显示在时间值处标绘的点。该图显示在“比较分布”报表下方。若在“比较分布”、“分位数刻画器”和“危险率刻画器”图中的任一个图中选择分布，这些分布还将显示在另外两个图中。

显示危险率函数 显示或隐藏叠加了选定分布所对应图的“危险率刻画器”。该图显示在“统计量”报表上方。若在“比较分布”、“分位数刻画器”和“危险率刻画器”图中的任一个图中选择分布，这些分布还将显示在另外两个图中。

显示统计学 显示或隐藏“统计学”报表。请参见第 41 页的“寿命分布：统计学”。

选项卡式报表 在各个选项卡上显示图形和数据，而不是以默认的分级显示样式。

显示置信区域 显示或隐藏图中的着色置信区域。

区间类型 确定为“比较分布”图中的非参数拟合显示的置信区间的类型。选择点态或联合置信区间。

更改置信水平 支持您更改整个平台的置信水平。所有图和报表都会相应更新。

比较准则 支持您选择用于对“模型比较”报表中的各个模型排名的准则。对于全部三个准则，值越小均指示拟合效果越好。 Burnham and Anderson (2002) 和 Akaike (1974) 讨论了如何使用 AICc 和 BIC 进行模型选择。请参见 《拟合线性模型》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

若您指定了“依据”变量，则为“依据”变量的每个水平显示单独的“寿命分布”报表。

保存 “依据”组结果 对于每个“依据”组，将所有“参数估计值”报表中针对该组显示的估计值都另存为新表中一个单独的行。

对所有组执行相同的分析 将针对当前组的所有选定选项都应用到其他所有 “依据”组。

混合

“拟合混合模型”选项将在报表中添加“混合”分级显示项，您可以在其中拟合数据的混合分布。有关示例，请参见第 67 页的“拟合混合模型示例”。

混合分布的概率函数 F(x) 定义如下：

$$F(x) = \sum_{i=1}^{k} w_i F_i(x)$$

其中，$F_i(x)$ 是受支持的分布之一，k 是混合模型中的成分个数，w_i 是总和为 1 的正权重。“拟合混合模型”选项尝试标识从每个成分分布 $F_i(x)$ 中抽取的观测聚类。它估计混合模型的参数以及从任意给定成分中抽取观测的概率。

模型拟合和混合起始值方法

拟合方法基于有关相应聚类的假设，该方法称为 “起始值方法”。假定您指定了 k 个分布。有三种 “起始值方法”：

- “单个聚类”假定所有观测在一定程度上都受到所有成分分布的影响。没有任何密度单独影响一部分观测。
- “可分隔的聚类”假定成分分布对某些观测的影响比对其他观测更深。对于可分隔的聚类，k 个密度中的每一个都具有可标识的模式并定义一个聚类。

- “重叠聚类”假定了介于“单个聚类”与“可分隔的聚类”之间的一种情形。某些密度单独影响部分观测，但其他密度联合影响部分观测。在这种情况下，数据中有 *m* 个聚类，其中 *m* 小于 *k*（总密度数）。

拟合过程包含以下步骤：

1. 定义观测聚类。
2. 基于 “起始值方法”将聚类分配到密度：
   - 对于“可分隔的聚类”，分配给指定的成分密度的最高似然聚类通过检查可能的排列来确定。
   - 对于“重叠聚类”，分配给指定的成分密度的最高似然聚类通过检查可能的聚类排列和观测组合来确定。

注意：假定您使用给定的“起始值方法”拟合了某个模型，然后又选择了另一种“起始值方法”。若无法基于似然值实现更好的拟合，则不添加任何新模型。

“混合”控制面板

该控制面板包含以下项：

成分 列出您可以用作拟合混合分布的成分的分布。

数量 选择混合分布中具有给定分布的成分个数。“数量”值的总和为 *k*，即混合分布中密度的个数。

起始值方法 选择反映您关于混合分布的假设的方法。请参见第 48 页的“模型拟合和混合起始值方法”。

叠加 显示未删失数据值的非参数估计值 (Kaplan-Meier-Turnbull)。拟合混合模型时，该图会随之更新以显示模型和 95% 水平置信带。这些置信带的置信水平由 “寿命分布”红色小三角菜单中的 “更改置信水平”选项确定。“图例”显示在图右侧。

执行 点击**执行**可拟合所需的混合模型。“模型列表”会随您拟合的模型更新，同时添加具有混合模型名称的报表。

“拟合混合模型”报表

模型列表

“模型列表”报表会列出您拟合的混合分布。该报表提供每个混合分布的参数数目、实际观测数，以及 AICc、-2对数似然和 BIC 统计量。有关这些统计量的详细信息，请参见《拟合线性模型》。

请注意以下几点：

- 以上每个统计量的值越小指示拟合效果越好。
- 各行按 AICc 排序。

- **比较准则**红色小三角选项不影响“模型列表”中模型的顺序。
- AICc、-2对数似然和 BIC 统计量还显示在“模型比较”表中。这样您就可以将混合分布与您的数据的其他分布进行比较。请参见第 41 页的“模型比较”。

“混合模型”报表

“模型列表”报表之后跟有您已经拟合的每个混合分布的报表。每个报表的标题使用指定的成分及其数量描述对应的混合模型。报表会列出参数、参数估计值、标准误差和 95% Wald 置信区间。在启动窗口中将“似然”选作“置信区间方法”并不会影响这些区间。

为混合模型中的每个分布提供了参数估计值。“参数”列还包括名为“对应部分 <i>”的参数，其中 $i = 1, 2, ..., k\text{-}1$。这些是混合模型的权重 w_i 的估计值。由于这些权重的总和为 1，所以可以根据前 $k - 1$ 个权重计算出第 k 个权重。

“密度叠加”图

“密度叠加”图显示混合模型中每个成分的密度函数的估计值。图右侧的图例支持您选择要显示的密度函数。

“混合模型”报表选项

该红色小三角菜单包含以下选项：

删除 删除模型报表以及 “模型列表”中该模型对应的条目。

显示刻画器 显示混合分布 F 的四种刻画器。有关这些刻画器的红色小三角选项的说明，请参见第 50 页的 “混合模型刻画器选项”。

- “分布刻画器”显示作为时间的函数的累积失效概率。
- “分位数刻画器”显示作为累积概率的函数的失效时间。
- “危险率刻画器”显示作为时间的函数的危险率。
- “密度刻画器”显示分布的密度函数。

保存预测 对于每个混合模型密度，会在数据表中保存一列，其中包含某个观测属于该密度的概率。对于计算中使用的公式，请参见第 88 页的 ““拟合混合模型”保存预测公式”。

混合模型刻画器选项

每个混合模型报表的刻画器都包含以下红色小三角选项：

置信区间 “分布”、“分位数”和“危险率”刻画器显示所绘函数的基于 Wald 的 95% 置信曲线。该选项显示或隐藏置信曲线。这些曲线的置信水平由“寿命分布”红色小三角菜单中的“更改置信水平”选项确定。

注意：为了减少计算时间，刻画器中提供的置信区间基于 Wald 方法，即便您在启动窗口中将 “似然”选作 “置信区间方法”也是如此。

重置因子网格 为每个因子显示一个窗口，允许您为因子的当前设置输入特定值，锁定该设置，以及控制网格的各方面。请参见《刻画器指南》。

因子设置 提供一个菜单，其中包含与刻画器设置、脚本和关联刻画器相关的选项。请参见《刻画器指南》。

拟合竞争风险混合模型

“拟合竞争风险混合模型”选项支持您拟合竞争风险混合分布。该选项估计给定观测因为由每个成分混合分布表示的原因而失效的概率。有关示例，请参见第 70 页的“拟合竞争风险混合模型示例”。

竞争风险混合概率分布函数 $F(x)$ 定义如下：

$$F(x) = 1 - \prod_{i=1}^{k} (1 - F_i(x))$$

其中，$F_i(x)$ 表示第 i 个风险的累积失效分布，k 是混合模型中成分（风险）的数目。“拟合竞争风险混合模型”选项尝试标识从每个成分分布 $F_i(x)$ 中抽取的观测聚类。它估计混合模型的参数以及从任意给定成分中抽取观测的概率。

“竞争风险混合模型”报表的结构类似于“混合模型”报表。请参见第 48 页的“混合”。不过，“拟合竞争风险混合模型”报表不显示“密度叠加”图，而是显示“分布叠加”图。

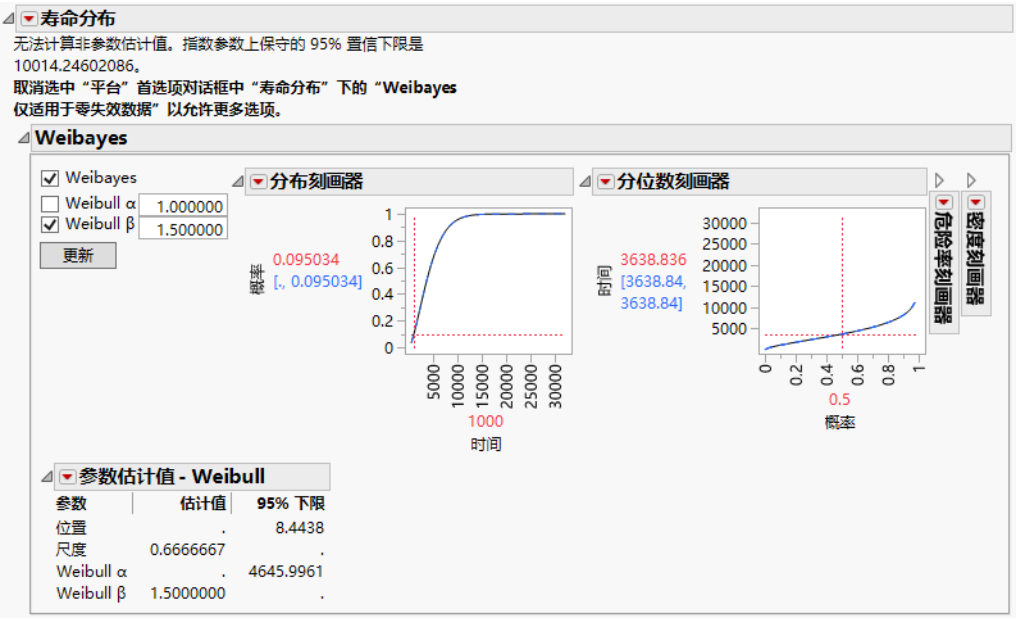
“分布叠加”图

“分布叠加”图显示每个混合成分和组合混合（聚合）的累积分布函数。图右侧的图例支持您选择要显示的累积分布函数。

Weibayes 报表

若您的数据包含零失效（右删失），而您又没有关闭首选项 Weibayes 仅适用于零失效数据，则会显示一个特殊的 Weibayes 报表。图 3.10 显示了 Reliability 文件夹中的 Weibayes No Failures.jmp 样本数据表的 Weibayes 报表。

图 3.10 Weibayes 报表



分析最开始假定一个指数模型。报表顶部的注释提供了指数分布参数的置信下限。这个置信下限是使用 Meeker and Escobar (1998, sec. 7.7) 中说明的方法计算得出的。

报表的 Weibayes 部分执行 Weibayes 分析。有关该过程的说明，请参见 Nelson (1985)。

要获取 Weibayes 估计值，确保选定 Weibayes 和 Weibull β 选项。更改 Weibull β 值并点击“更新”。估计值和刻画器随之更新。刻画器中显示的值使用保守置信界限。有关示例，请参见第 65 页的“无失效数据的 Weibayes 示例”。

注意：若您取消选择 Weibayes 选项，JMP 将显示固定参数 MLE。

若您的数据表包含至少一个失效，则会显示完整的“寿命分布”报表，但最大似然推断可能不会生成有用的结果。在这种情况下，Weibayes 分析可能更合适。有关示例，请参见第 66 页的“有一个失效数据的 Weibayes 示例”。

注意：要在包含零失效数据时使用常用的“寿命分布”选项执行 Bayes 推断，请选择**文件 > 首选项 > 平台 > 寿命分布**，然后在启动分析之前取消选择 **Weibayes 仅适用于零失效数据**。

“竞争原因”报表

提示：若您发现报表窗口过长，请从“竞争原因”红色小三角菜单中选择**选项卡式报表**。

若您在启动窗口中分配了“失效原因”列，则会显示“竞争原因”报表。使用该报表可分析竞争原因以确定哪些原因有影响。有关该报表的示例，请参见第 60 页的“忽略竞争原因”。 有关技术细节，请参见第 84 页的“竞争原因详细信息”。

“竞争原因”报表包含以下内容和选项：

- 第 54 页的“原因组合”
- 第 55 页的“竞争原因：统计学”
- 第 56 页的“单个原因”
- 第 56 页的“‘竞争原因’报表选项”

竞争原因工作流程

遵照以下步骤将便于您使用 “竞争原因”报表：

1. 为方便起见，点击“竞争原因”红色小三角并选择“选项卡式报表”选项。
2. 选择 “单个原因”选项卡。对于每个失效原因，请使用其自身的 “寿命分布”分级显示项中的选项选择一种分布拟合。请参见第 56 页的 “单个原因”。
3. 选择“原因组合”选项卡。对于每种失效原因，在每个**分布**列表中指定所需分布。请参见第 54 页的 “原因组合”。
4. 点击**更新模型**。
5. 选择 “统计学”选项卡可探索和保存模型结果。请参见第 55 页的 “竞争原因：统计学”。

注意：若更改模型并再次点击**更新模型**，针对“统计学”分级显示项中的竞争原因模型报表进行的定制可能丢失。

竞争原因模型

在竞争原因情形下，聚合失效函数可以表示为：

$$F(x) = 1 - \prod_{i=1}^{k} [1 - F_i(x)]$$

其中，$F_i(x)$ 是第 i 个原因的累积失效分布，k 是总原因数。函数 $F_i(x)$ 特定于原因。它反映单单由于原因 i 而引起的失效的概率，并不解释其他失效原因。

备选公式定义如下：

$$F(x) = \sum_{i=1}^{k} \tilde{F}_i(x)$$

其中，每个 $\tilde{F}_i$ 都是一个单调递增函数，函数值介于 [0, 1] 的区间内。函数 $\tilde{F}_i$ 称为子分布。这种形式的聚合失效分布用于预测与单个原因关联的失效比例，考虑因其他原因导致的失效。

原因组合

“原因组合”报表支持您拟合和比较各种原因的不同失效时间分布 ($F_i(x)$)。根据您在启动窗口中做出的选择会提供不同的分布。不允许失效时间为负数和零。

默认图基于 “线性”尺度并显示以下内容：

- 未删失数据值的非参数估计值 (Kaplan-Meier-Turnbull) 及其置信区间。置信区间通过水平蓝线来表示。
- 为每个原因拟合累积失效分布 ($F_i(x)$)。初始分布是您在启动窗口中选定的分布。若您选择了“单个最佳”，则为每个组计算最佳分布并且显示这些拟合。（该选择可能要花费较长时间。）若选择了“手动选择”，在初始分布对于所有组都是“非参数”并且显示非参数拟合。图右侧会显示一个图例。
- “聚合”累积失效分布 $F(x)$ 表示为一条黑线。基于选定的原因分布计算该函数。若为某个原因指定了非参数分布，则聚合累积失效分布仅延伸到该原因的最终时间观测所在的位置。

当您与该报表交互时，将重新计算聚合模型的统计量。

“原因组合”报表包含以下元素：

尺度 指定图的垂直轴的概率尺度。第 62 页的“更改尺度”演示了更改尺度如何影响分布拟合。

忽略 选中某个框可删除相应原因的拟合。在固定某个特定原因后使用该选项。“聚合”模型随之更新以反映已删除因该原因导致的失效。第 60 页的“忽略竞争原因”演示了忽略原因之后的效果。

原因 列出“原因”列中的原因。

分布 列出每个原因可用的分布。要更改特定失效原因的分布，请从**分布**列表中选择该分布。点击**更新模型**可在图上显示新的分布拟合。“原因汇总”报表也随之更新。

计数 提供具有给定失效原因的观测数。

更新模型 显示图中选定的分布；使用选定模型更新“原因汇总”报表；并将选定模型添加为“单个原因”报表中最近的模型。

竞争原因：统计学

“竞争原因”的“统计学”报表包含以下报表：

- 第 55 页的“ “原因汇总”报表”
- 第 55 页的“刻画器”

“原因汇总”报表

“原因汇总”报表显示当前选择所定义的拟合的信息（在“原因汇总”报表中“分布”下方）。该报表显示每个原因的失效数目以及每个失效原因的分布拟合的参数估计值。当您在“原因组合”报表中更改分布拟合并点击“更新模型”后，“原因汇总”报表随之更新。

提供以下信息：

- “原因”列显示原因的标签或删失代码。
- “计数”列列出每个原因的失效数目。
- “计数”列中的数值条目指示原因有足够多的失效事件可供考虑。事件不足两个的原因被视为右删失。该列还标识缺失原因。
- “分布”列指定每个原因的选定分布。
- 根据选定的分布，各列显示分布的参数：
  - 位置列指定各个分布的位置参数。
  - 尺度列指定各个分布的尺度参数。
  - Weibull α 和 Weibull β 列显示 alpha 和 beta 的 Weibull 估计值。
  - 其他列显示其他选定分布的参数。
  - 若存在收敛问题，将显示“收敛”列。

“原因汇总”红色小三角菜单选项支持您将聚合失效分布的概率、分位数、危险率和密度估计值保存到数据表。

刻画器

“分布刻画器”、“分位数刻画器”、“危险率刻画器”和“密度刻画器”帮助您直观演示聚合失效分布。“分布刻画器”、“分位数刻画器”和“危险率刻画器”显示 95% 水平置信带。请参见第 42 页的“刻画器”。

注意：若为某个原因指定了非参数分布，则不提供“危险率刻画器”和“密度刻画器”。此外，也不在“分布刻画器”和“分位数刻画器”中提供置信限。

单个原因

“单个原因”报表包含每一个单个原因的“寿命分布 - 失效原因:<分布名称>”报表。每个“寿命分布 - 失效原因:<分布名称>”报表都显示报表标题中指明的单个失效原因的图和分布拟合统计量。

只要您点击 “原因组合”报表中的 “更新模型”，您选择的所有新的原因分布都会添加到该原因的 “寿命分布 - 失效原因 : < 分布名称 >”报表中。在 “寿命分布 - 失效原因 : < 分布名称 >”报表中，会发生以下情况：

- 在“比较分布”报表中选定该分布。
- 该分布添加到“模型比较”报表。
- 为该分布添加了“参数”报表。

每个“寿命分布 - 失效原因 <分布名称>”报表都是一个“寿命分布：统计学”报表，如第 41 页的“寿命分布：统计学”所述。不过，所有置信区间都是 Wald 区间。在启动窗口中将“似然”选作“置信区间方法”并不会影响这些区间。

可用于竞争原因比较分布报表的分布

若在启动窗口中指定“失效原因”，您可以指定想在为“单个原因”生成的“比较分布”报表中允许哪些模型和分布分组。您可以选择 ZI（零泛滥）分布、TH（阈值）分布和 DS（缺陷子总体）分布。您还可以选择固定参数和 Bayes 模型。

注意：若您已在“首选项”中禁用了任何分布，这些分布则不会出现。此外，还要遵从用于控制对“寿命分布”显示哪些分布的规则。请参见第 39 页的“可用的参数分布”。

“竞争原因”报表选项

“竞争原因”红色小三角菜单包含以下选项：

选项卡式报表 在各个选项卡上显示图形和数据，而不是以默认的分级显示样式。

单个原因的选项卡式报表 在各个选项卡中显示 “寿命分布 - 失效原因 : < 分布名称 >”报表，而不是堆叠显示各个寿命分布报表。

显示点 显示或隐藏 “竞争组合”图中的数据点。“寿命分布”平台使用阶梯函数的中点估计值来构造概率图。取消选择**显示点**时，中点估计值会被 Kaplan-Meier 估计值取代。

显示子分布 显示每一个单个原因子分布 $\tilde{F}_i$ 的刻画器。请参见第 57 页的 “原因单个子分布刻画器”。

显示剩余寿命分布 通过 “分布刻画器”显示剩余寿命分布。这是有条件的，具体取决于存活到给定时间的单元。

剩余寿命均值 估计给定生存时间条件下的某个单元的剩余寿命均值。在“剩余寿命均值计算器”中，输入一个时间值并按 Enter 键以查看估计值和置信限。点击加号可输入其他时间。点击减号可删除最新添加的条目。

要获取置信区间，请从“剩余寿命均值计算器”中选择**配置**选项。选中**使用 Bootstrap 构造置信区间**。输入适当的值，请记住计算可能占用较长时间。请参见第 88 页的“剩余寿命均值计算器”。

导出 Bootstrap 结果 仅在选定了 Bayes 模型并且应用了**更新模型**后才显示。

Bootstrap 样本大小 将 Bayes 估计或 Weibayes 结果用于任何原因时，必须使用参数 Bootstrap 模拟显示在“分布刻画器”中的聚合函数的置信限。使用该选项可指定要用在 Bootstrap 中的样本数。请参见第 87 页的“为原因指定 Bayes 模型”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

原因单个子分布刻画器

对于给定原因，“原因单个子分布刻画器”显示因该原因在时间 t 的失效 $\tilde{F}_i$ 的估计概率。估计概率会将来自竞争原因的失效考虑在内。请参见第 53 页的“竞争原因模型”。

要显示每个原因的子分布的刻画器，点击“竞争原因”红色小三角并选择**显示子分布**。“原因单个子分布刻画器”报表显示在其他刻画器下方。它包含一个刻画器和一个计算器。

注意：当您选择**显示子分布**时，“原因组合”图随之更新以显示所有原因的子分布函数。

从刻画器右侧的列表中选择一个“原因”可查看其刻画器。适用于该刻画器的选项包含在“原因单个子分布刻画器”红色小三角菜单中。请参见第 42 页的“刻画器”。

使用“计算器”面板可查找所有原因在一个或多个“事件时间”值处的子分布函数的值。输入“事件时间”变量的值。按 Enter 键（或在文本框外点击）后，每个原因的值随之更新。要添加“事件时间”值，请点击加号。要删除最新添加的值，请点击减号。

“寿命分布 - 比较组”报表

提示：若您发现报表窗口过长，请从“寿命分布 - 比较组”旁边的红色小三角菜单中选择**选项卡式报表**。

若您在启动窗口中选定了“比较组”选项卡，“寿命分布 - 比较组”报表随即显示。该报表使用单个指定分布比较不同的组。例如，您可以对按供应商分组的元件进行 Weibull 拟合比较。相比之下，“寿命分布”选项卡比较的是单个组的若干拟合分布。

您可以比较 CDF、分位数、危险率和密度等函数。您还可以合并所有组的概率和分位数预测。

有关该报表的示例，请参见第 63 页的“在不同组之间检查同一分布”。

“比较组”报表可包含以下内容和选项：

- 第 37 页的“比较分布”（无“分布刻画器”）
- 第 58 页的“比较组：统计学”
- 第 59 页的“单个组”
- 第 59 页的“‘寿命分布 - 比较组”报表选项”

比较组：统计学

“比较组”的“统计学”报表包含以下报表：

- 第 58 页的“汇总”
- 第 58 页的“组齐性检验”
- 第 41 页的“模型比较”
- 第 59 页的“参数估计值”

汇总

“汇总”报表对每个组和组合数据都包含一行。每行显示失效的单元数，以及左删失、区间删失或右删失的数目。每行还提供相应的均值和标准误差。有关如何计算均值和标准误差的详细信息，请参见“生存分析”一章中第 318 页的“生存分析的统计报表”。

组齐性检验

“组齐性检验”报表包含三个检验的结果，以确定各组的危险率函数是否相等。这些检验在观测随时间加权的方式上有所不同。

对数秩 “对数秩”检验对所有时间点都使用相同的权重。

Peto-Peto Peto-Peto 检验将每个时间点的生存比例用作权重。

Wilcoxon 广义 Wilcoxon 检验将每个时间点的危险数目用作权重。该检验被称为 Klein and Moeschberger (1997) 中的 Gehan 检验。

对于上述每个检验，报表都包含卡方近似值、相关自由度和每个检验的 p 值。较小的 p 值表明各组不同。有关这些检验和失效曲线的比较的详细信息，请参见 Klein and Moeschberger (1997, ch.7)。

参数估计值

“参数估计值”分级显示项为拟合的每个分布包含标题为“参数估计值 - <分布名称>”的报表。对于每个组，“参数估计值 - <分布名称>”报表提供该分布的参数估计值及其 95% 水平置信区间。这些区间的置信水平由“寿命分布 - 比较组”菜单中的“更改置信水平”选项确定。

单个组

“单个组”报表中的各个选项卡包含每一单个组的“寿命分布”报表。有关这些报表的详细信息，请参见第 34 页的“‘寿命分布’报表”和第 46 页的“‘寿命分布’报表选项”。

“寿命分布 - 比较组”报表选项

“比较组”红色小三角菜单中的许多选项也在“寿命分布”红色小三角菜单中提供。请参见第 46 页的“‘寿命分布’报表选项”。

不过，以下选项特定于 “比较组”：

显示分位数函数 显示或隐藏“比较分位数”报表。选择一个分布。对于每个组都标绘一条曲线，以显示时间变量的估计分位数。显示置信带。图例显示在图右侧。一次只能指定一个分布。

显示危险率函数 显示或隐藏“比较风险”报表。选择一个分布。对于每个组都标绘一条曲线，以显示危险率函数。显示置信带。图例显示在图右侧。一次只能指定一个分布。

显示密度函数 显示或隐藏 “比较密度”报表。选择一个分布。对于每个组，都显示密度函数和置信带。图例显示在图右侧。一次只能指定一个分布。

估计概率 添加一个 “估计概率”报表，该报表对应在 “比较分布”下方最新选定的分布。在文本框中输入 “事件时间”变量的值，然后按 Enter 键。要添加 “事件时间”值，请点击加号。要删除最新添加的值，请点击减号。请参见第 60 页的 “‘估计概率’报表”。

估计分位数 （仅在选定 “比较分位数”时显示。）添加一个 “估计分位数”报表，该报表对应在 “比较分位数”下方最新选定的分布。在文本框中输入关注的概率的值，然后按 Enter 键。要添加 “概率”值，请点击加号。要删除最新添加的值，请点击减号。对于每个组和概率值，都显示 “事件时间”分位数以及 95% Wald 和似然置信区间。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

“估计概率”报表

对于每个组和 “事件时间”值，该报表都提供以下内容：

中点估计 指定时间之前的失效的中点调整 Kaplan-Meier 估计值。

95% 下限， 95% 上限 指定时间之前的失效的概率的点态 95% 置信区间。

联合 95% 下限 (Nair)，联合 95% 上限 (Nair) 指定时间之前的失效的概率的联合 95% 置信区间。请参见 Nair (1984) 和 Meeker and Escobar (1998)。

生存概率 指定时间之后的生存的中点调整估计值。

生存概率 95% 下限， 95% 上限 指定时间之后的生存的概率的点态 95% 置信区间。

生存概率联合 95% 下限 (Nair)，联合 95% 上限 (Nair) 指定时间之后的生存的概率的联合 95% 置信区间。请参见 Nair (1984) 和 Meeker and Escobar (1998)。

“寿命分布”平台的更多示例

- “忽略竞争原因”
- “更改尺度”
- “在不同组之间检查同一分布”
- “Weibayes 估计值”
- “拟合混合模型示例”
- “拟合竞争风险混合模型示例”

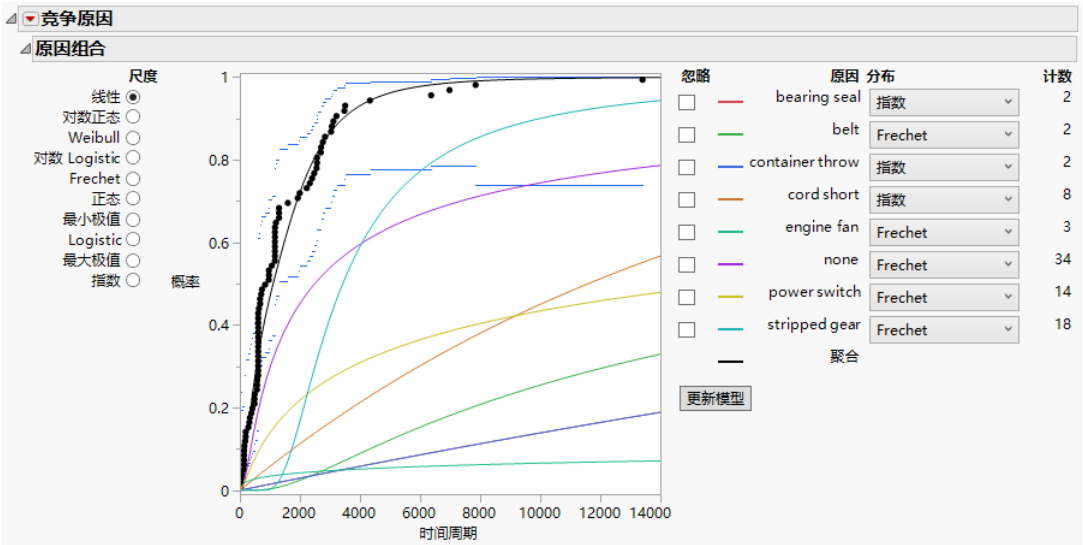
忽略竞争原因

本例演示如何决定竞争原因的最佳拟合。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Blenders.jmp。
2. 选择**分析** > **可靠性和生存** > **寿命分布**。
3. 选择**时间周期**并点击 **Y，事件时间**。
4. 选择**原因**并点击**失效原因**。
5. 选择**删失**并点击**删失**。
6. 将**单个最佳**选作 “分布”。
7. 确保 AICc 为 “比较准则”。
8. 点击**确定**。

   在 “竞争原因”报表中， JMP 显示了每个失效原因的最佳分布拟合。

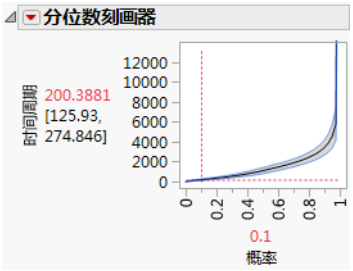
图 3.11 初始 “竞争原因”报表



9. 在 “分位数刻画器”中，为概率键入 0.1。

   出现 10% 的失效的估计时间为 200。

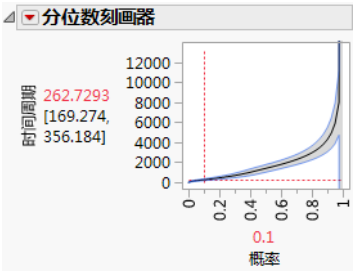
图 3.12 10% 单元失效的估计失效时间



10. 为 bearing seal、belt、container throw、cord short 和 engine fan（具有最少失效计数的原因）选择**忽略**。

    出现 10% 的失效的估计时间现在为 263。

图 3.13 更新的失效时间



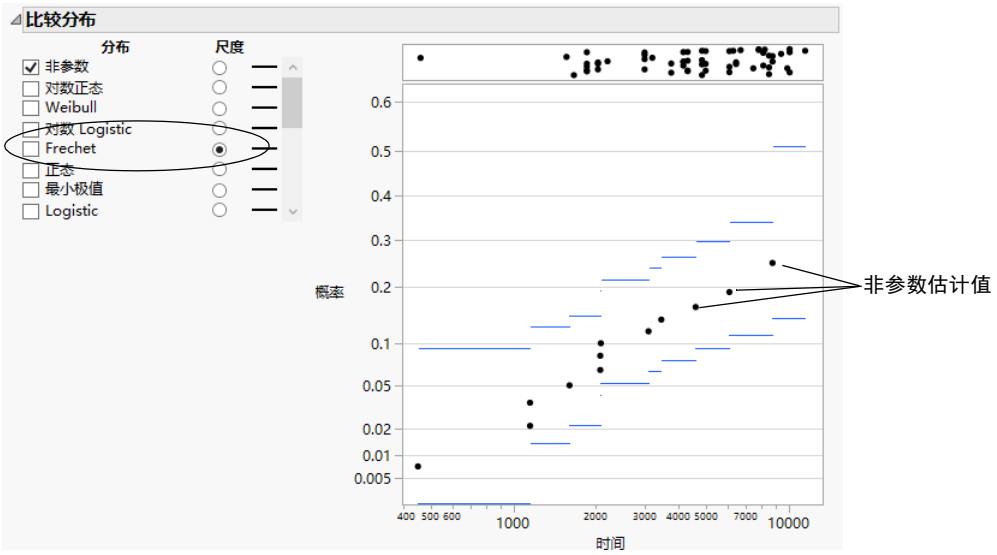
若电源开关和齿轮折断是仅有的失效原因，出现 10% 的失效的估计时间将增加大约 31%。

更改尺度

在初始 “比较分布”报表中，概率轴和时间轴是线性的。但假定您想要查看基于 Fréchet 尺度的分布估计值。[1]

1. 执行第 29 页的““寿命分布”平台的示例”中的第 1 步到第 5 步。
2. 在 “比较分布”报表中，选择 “尺度”列中的 Fréchet。
3. 点击 “寿命分布”红色小三角并选择**区间类型** > **点态**。

图 3.14 具有 Fréchet 概率尺度的非参数估计值



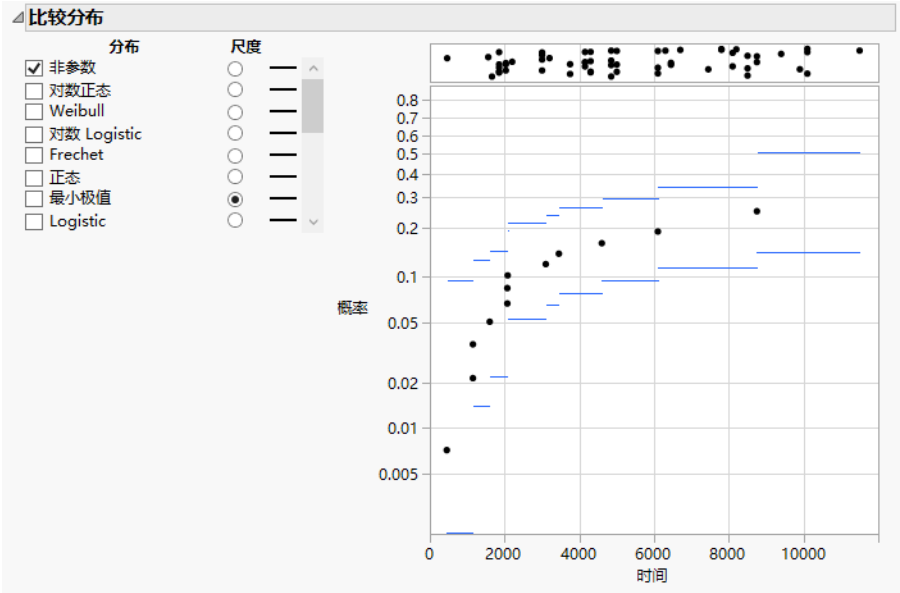
使用 Fréchet 尺度，非参数估计值近似为一条直线，这意味着 Fréchet 拟合可能合理。

1. 使用不同的尺度有时是指在不同类型的**概率纸**上绘制分布。

4. 在 “尺度”列中选择**最小极值**。

非参数估计值不再近似为一条直线。您现在知道最小极值分布不合适。

图 3.15 具有最小极值概率尺度的非参数估计值

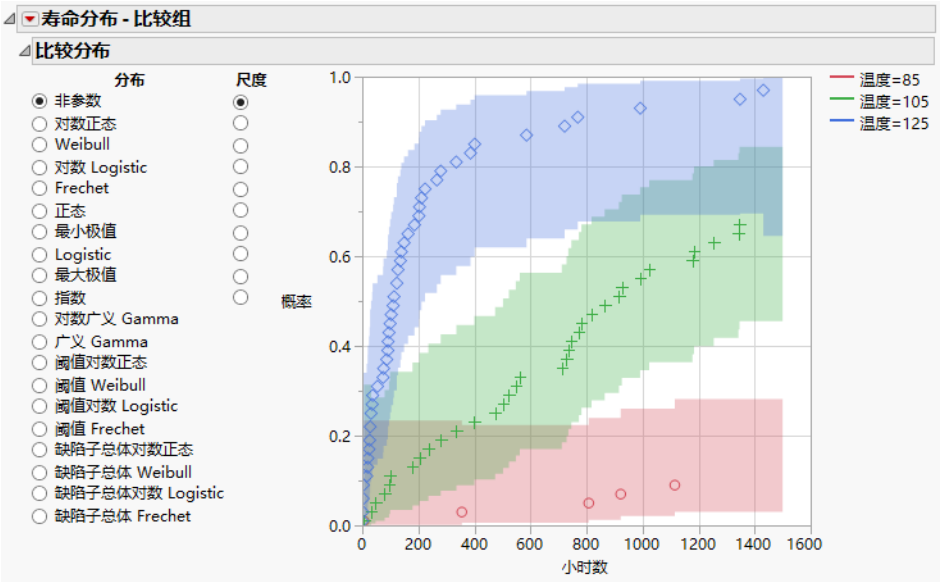


在不同组之间检查同一分布

假定您想要在不同组之间比较同一分布。您想要检查某一种电容在三个不同温度下工作时的失效概率估计值。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Capacitor ALT.jmp。
2. 选择**分析** > **可靠性和生存** > **寿命分布**。
3. 点击 “比较组”选项卡。
4. 选择**小时数**并点击 **Y，事件时间**。
5. 选择**温度**并点击**分组**。
6. 选择**删失**并点击**删失**。
7. 选择**频数**并点击**频数**。
8. 点击**确定**。

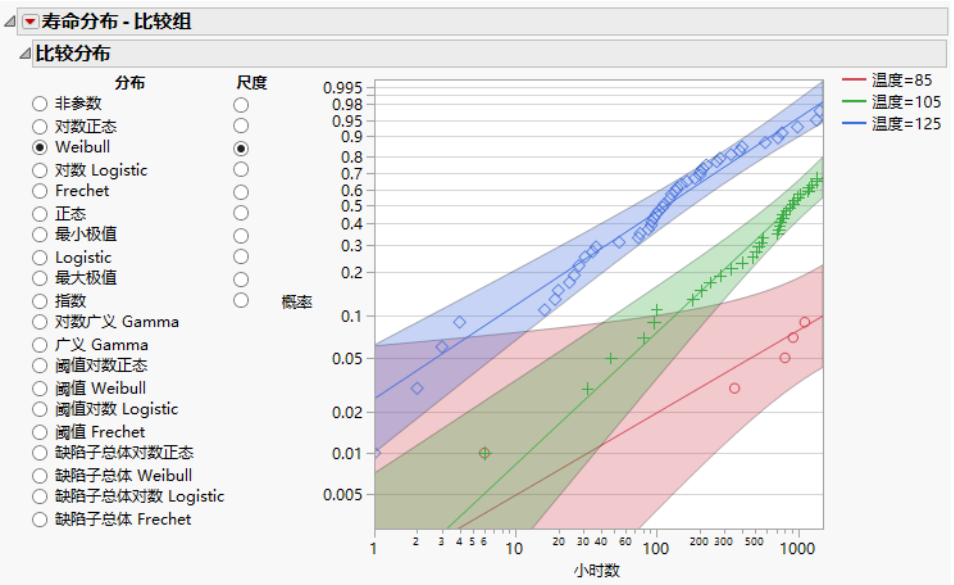
图 3.16 比较各组的分布



默认图形显示非参数估计值。在较高温度下，电容失效的概率较高。您想要尝试拟合参数分布。

9. 为“分布”和“尺度”选择 Weibull。

图 3.17 比较各组的 Weibull 分布



基于 Weibull 概率尺度进行绘图时，各点分布在每条线附近。这表明 Weibull 分布为每一个**温度**组都提供了合理拟合。

Weibayes 估计值

有两种可能的方法可以获取 Weibayes 分析：

- 没有失效（所有观测均右删失）并且已选中首选项 **Weibayes 仅适用于零失效数据**，则显示 Weibayes 报表。请参见第 65 页的“无失效数据的 Weibayes 示例”。
- 有极少的失效。显示完整的“寿命分布”报表。拟合 Weibull 分布。在“参数估计值 - Weibull”报表中，选择“固定参数”选项。然后选择“固定参数”报表中的 Weibayes 选项。请参见第 66 页的“有一个失效数据的 Weibayes 示例”。

无失效数据的 Weibayes 示例

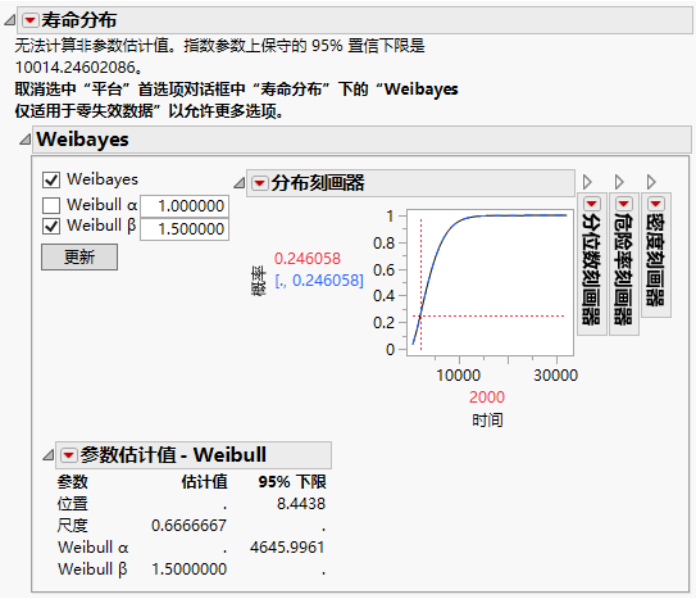
您有一种大体上可靠的产品的相关数据，其中 30 个经过 1,000 个小时的测试后未出现失效。您想要预测 2,000 个小时下的失效概率。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Weibayes No Failures.jmp。
2. 选择**分析 > 可靠性和生存 > 寿命分布**。
3. 选择**时间**并点击 **Y，事件时间**。
4. 选择**删失**并点击**删失**。
5. 选择**频数**并点击**频数**。
6. 将**似然**选作 “置信区间方法”。
7. 点击**确定**。

   特殊的 “寿命分布”报表随即显示。 Weibayes 和 Weibull beta 应已被选定。
8. 键入 1.5 作为已知的 Weibull β 值。

   对本例而言，值 1.5 是合适的。
9. 点击**更新**。
10. 在 “分布刻画器”中，为 “时间”键入 2000。

图 3.18 零失效的 “寿命分布”报表



从“分布刻画器”中，您可以看到在 2,000 个小时处，保守概率为 24.6058%。这意味着失效概率的单尾 95% 保守置信限为 24.6058%。

有一个失效数据的 Weibayes 示例

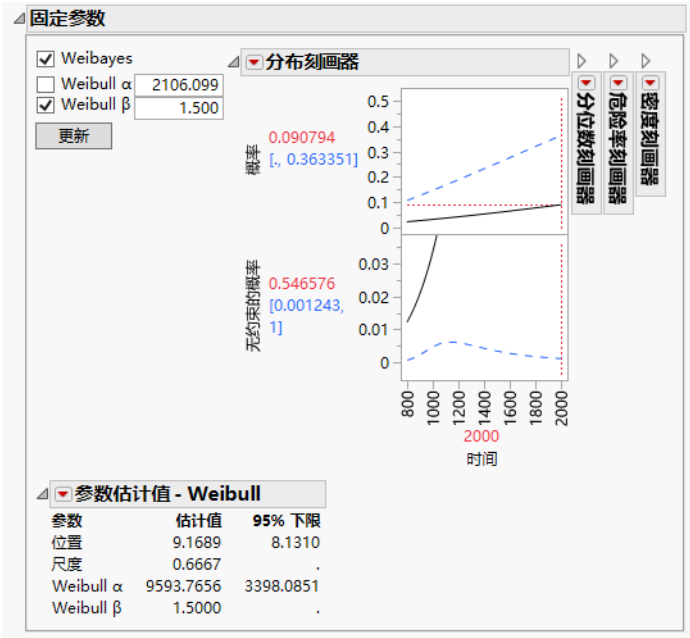
假定您还使用相同的数据，但这次在 800 个小时的时候出现了一个失效。您仍想预测 2,000 个小时下的失效概率。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Weibayes One Failure.jmp。
2. 选择**分析** > **可靠性和生存** > **寿命分布**。
3. 选择**时间**并点击 **Y，事件时间**。
4. 选择**删失**并点击**删失**。
5. 选择**频数**并点击**频数**。
6. 将**似然**选作 “置信区间方法”。
7. 点击**确定**。

   “寿命分布”报表随即显示。
8. 选择 “比较分布”图中的 Weibull 分布。
9. 点击 “参数估计值 - Weibull”旁边的红色小三角并选择**固定参数**。
10. 选择 “固定参数”报表中的 Weibayes 和 Weibull beta。
11. 键入 1.5 作为已知的 Weibull β 值。

12. 点击**更新**。
13. 在 “分布刻画器”中，为 “时间”键入 2000。
14. 悬停在 *Y* 轴顶部上方。光标将变为手形工具。将该轴向下拖动，直到最高处的值达到 0.5。

图 3.19 一个失效的 “寿命分布”报表



在“分布刻画器”中，实线显示 MLE。虚线显示 Weibayes 保守限值。您可以看到在 2,000 个小时处，保守概率为 36.3351%。这意味着失效概率的单尾保守 95% 置信限为 36.3351%。

拟合混合模型示例

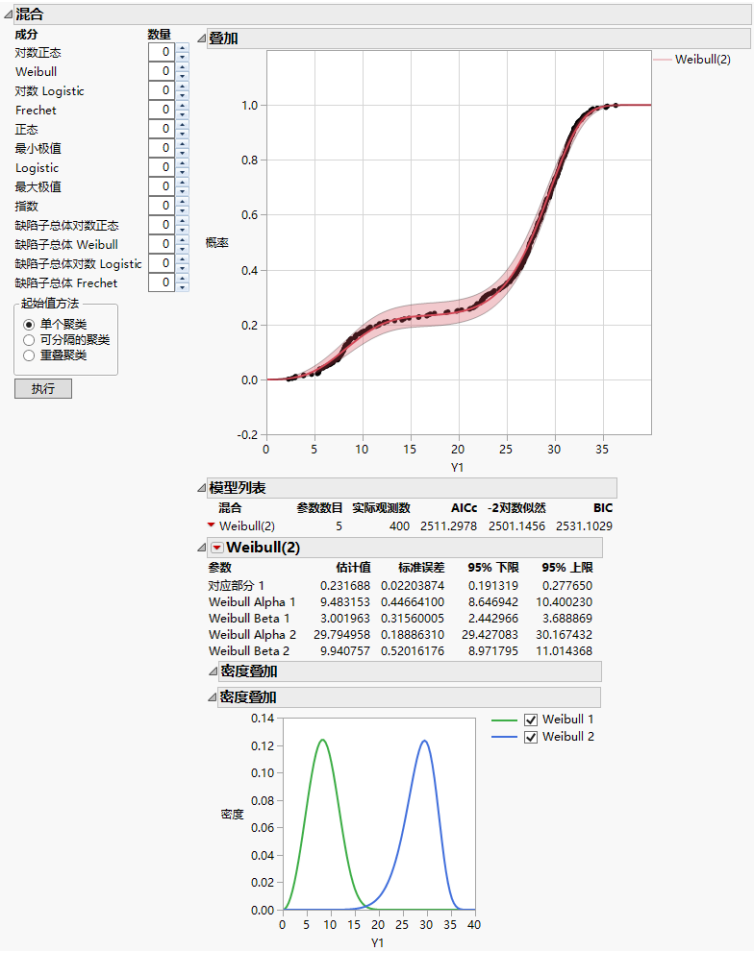
在本例中，您拟合两个混合分布，然后标识属于第二个混合分布的其中一个聚类的观测。

拟合两个混合分布

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Mixture Demo.jmp。
2. 选择**分析** > **可靠性和生存** > **寿命分布**。
3. 选择 Y1 并点击 **Y，事件时间**。
4. 点击**确定**。
5. 点击 “寿命分布”红色小三角并选择**拟合混合模型**。
6. 在 Weibull 旁边的**数量**框中键入 2。
7. 在 “起始值方法”面板中选择**可分隔的聚类**。

8. 点击**执行**。

图 3.20 拟合 “Weibull(2)” 的混合模型

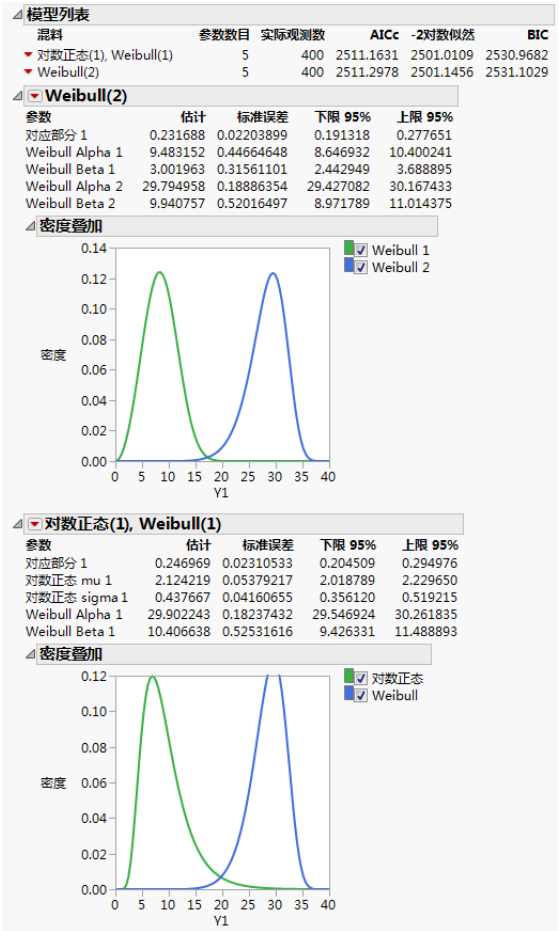


JMP 拟合包含两个 Weibull 成分的混合模型。“对应部分 1”的估计值为 0.231688，指示大约有 23% 的观测服从 Weibull 分布，其 alpha = 9.483153， beta = 3.001963。剩余的 77% 估计来自第二个 Weibull 分布。

为了将该模型与另一个模型进行比较，您可以更改 “成分”选择和成分 “数量”。

9. 在**对数正态**旁边键入 1，并且在 **Weibull** 旁边键入 1。
10. 点击**执行**。

图 3.21 拟合 “对数正态 (1)， Weibull(1)”的混合模型



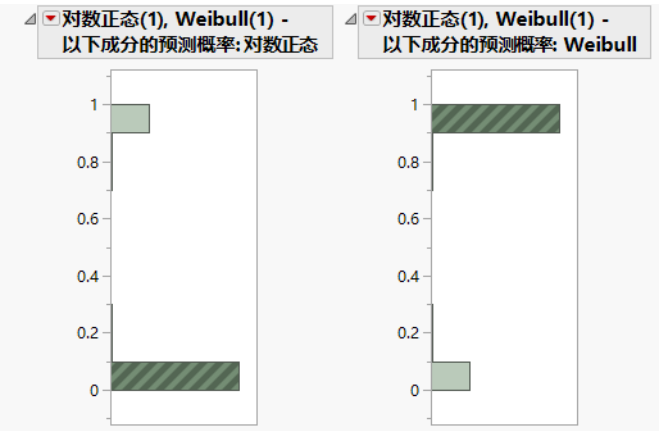
“叠加”图随之更新以显示两个混合模型。“模型列表”中的图和统计量指出 “对数正态 (1)， Weibull(1)”混合模型看起来提供了与 “Weibull(2)”混合模型极为相似的拟合。

标识属于某聚类的观测

1. 点击“对数正态 (1)，Weibull(1)”旁边的红色小三角，然后选择**保存预测**。
   将向数据表添加两列：
   - 对数正态(1)，Weibull(1) - 以下成分的预测概率: 对数正态
   - 对数正态(1)，Weibull(1) - 以下成分的预测概率: Weibull
2. 选择**分析 > 分布**。
3. 从 “选择列”列表中选择这两个新列，然后点击 **Y，列**。
4. 选中**仅显示直方图**。

5. 点击**确定**。
6. 在**对数正态 (1)，Weibull(1) - 以下成分的预测概率 : Weibull** 的直方图中，点击与接近 1 的值对应的直条。

图 3.22 混合概率的直方图



在数据表中，选定了 297 个对应的行。这些是可能来自参数 alpha = 29.90 且 beta = 10.41 的 Weibull 分布的观测。

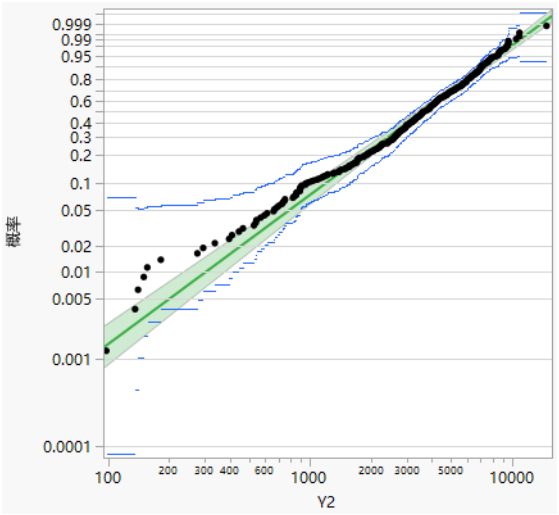
拟合竞争风险混合模型示例

在本例中，您将单个 Weibull 分布的拟合与具有两个 Weibull 拟合的竞争风险混合分布的拟合进行比较。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Mixture Demo.jmp。
2. 选择**分析 > 可靠性和生存 > 寿命分布**。
3. 选择 Y2 并点击 **Y，事件时间**。
4. 点击**确定**。
5. 在 “比较分布”报表中，选择 Weibull 分布和相应的**尺度**单选按钮。

   随即显示单个 Weibull 分布拟合的概率图。请注意， Y2 范围下部拟合不佳。

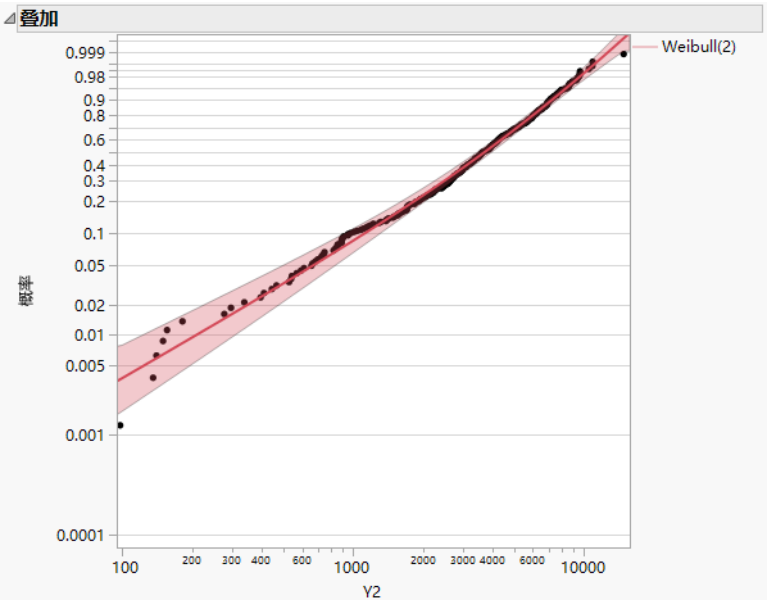
图 3.23 Weibull 分布拟合



6. 点击 “寿命分布”红色小三角并选择**拟合竞争风险混合模型**。
7. 向下滚动至 “竞争风险混合模型”报表。在 Weibull 旁边的**数量**框中键入 2。
8. 点击**执行**。
9. 向上滚动至 “比较分布”报表。在概率图中，右击垂直轴并选择**编辑 > 复制轴设置**。
10. 向下滚动至“竞争风险混合模型”报表。在叠加图中，右击垂直轴并选择**编辑 > 粘贴轴设置**。
11. 为水平轴执行相同操作。在概率图中，右击水平轴并选择**编辑 > 复制轴设置**。
12. 在叠加图中，右击水平轴并选择**编辑 > 粘贴轴设置**。

    随即显示 Weibull(2) 分布拟合的概率图。请注意，两个 Weibull 分布的混合有助于更好地捕捉 Y2 范围下部的分布。

图 3.24 Weibull(2) 竞争风险混合分布的拟合



“寿命分布”平台的统计详细信息

本节涉及以下主题：

- “寿命分布”平台中使用的分布。请参见第 72 页的“分布”。
- 有关“竞争原因”报表的技术信息。请参见第 84 页的“竞争原因详细信息”。
- 有关中位数秩回归的技术信息。请参见第 89 页的“关于中位数秩回归的注释”。

分布

本节提供有关“寿命分布”平台中的分布拟合的详细信息。Meeker and Escobar (1998, ch. 2-5) 是关于非参数和参数细节所依据的理论、应用和讨论的极佳信息来源。

估计和置信区间

所有分布的参数（除非另有说明）都使用最大似然估计值 (MLE) 来估计。唯一的例外是阈值分布。若最小的观测是一个精确失效，则该观测被视为具有小区间的区间删失。参数估计值是从这个稍作修改的数据集估计的 MLE。若不进行这项修改，似然会无界，所以 MLE 可能不存在。该方法类似于 Meeker and Escobar (1998, p. 275) 中所述的方法，只不过最小精确失效删失。这是为确保似然函数有界而对数据进行的最小更改。

“寿命分布”平台提供两种方法来计算分布参数的置信区间。这两种方法标记为“Wald”或“似然”，可在“寿命分布”平台的启动窗口中进行选择。Wald 置信区间用作默认设置。计

算累积分布函数 (cdf) 的置信区间时，最初要计算标准化变量的 Wald 置信区间。接下来，这些区间变换为 cdf 尺度（Nelson, 1982, pp. 332–333 and pp. 346-347)。其他图形和刻画器中给出的置信区间变换为 Wald 区间 (Meeker and Escobar, 1998, chap. 7)。双参数分布的参数的联合置信区间显示在对数似然等高线图中。它们基于参数的近似似然比 (Meeker and Escobar, 1998, chap. 8)。

非参数拟合

非参数拟合说明分布的基本曲线。对于不含删失的数据（仅包含失效）以及观测同时包含失效和右删失的数据，JMP 使用 Kaplan-Meier 估计值。对于混合删失、区间删失或左删失，JMP 使用 Turnbull 估计值。若数据集仅包含右删失数据，则“非参数估计值”报表将指明无法计算非参数估计值。

“寿命分布”平台使用阶梯函数的中点估计值来构造概率图。中点估计值介于当前和前一个 Kaplan-Meier 估计值的中间位置（或二者的平均值）。

参数分布

参数分布提供比非参数分布更简明的分布拟合。失效时间分布的估计值也更平滑。参数模型还可用于外推 （沿时间）到分布的下尾或上尾。

注意：“寿命分布”平台中的许多分布都按照位置和尺度参数化。对于对数正态拟合，还提供中位数。阈值参数还包含在阈值分布中。位置对应于 μ，尺度对应于 σ，阈值对应于 γ。

对数正态

对数正态分布通常用于失效时间数据范围是 10 的若干次幂的情形。该分布常常被视为许多较小的独立同分布的正数随机变量的乘积。这在数据值的对数呈现正态分布时是合理的。由对数正态分布正确建模的数据示例包括：医院费用数据、金属疲劳裂纹扩展，以及细菌在消毒剂控制下的存活时间。 pdf 曲线通常呈现较强的右偏特点。对数正态 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \frac{1}{x\sigma}\phi_{nor}\left[\frac{\log(x)-\mu}{\sigma}\right], \quad x > 0$$

$$F(x;\mu,\sigma) = \Phi_{nor}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

其中

$$\phi_{nor}(z) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{z^2}{2}\right)$$

和

$$\Phi_{nor}(z) = \int_{-\infty}^{z}\phi_{nor}(w)dw$$

分别为标准化正态或 nor(μ = 0, σ = 1) 分布的 pdf 和 cdf。

Weibull

Weibull 分布可用于对具有升高或下降危险率的失效时间数据建模。由于该分布能够极为灵活地基于形状参数 β 的值对众多不同类型的数据建模，所以它经常用在可靠性分析中。该分布已经成功用于描述电子元件、滚柱轴承、电容和陶瓷领域中的失效。可通过更改尺度参数 α 和形状参数 β 来显示 Weibull 分布的各种形状。 Weibull pdf 和 cdf 通常表示如下：

$$f(x;\alpha,\beta) = \frac{\beta}{\alpha^{\beta}}x^{(\beta-1)}\exp\left[-\left(\frac{x}{\alpha}\right)^{\beta}\right]; \qquad x > 0, \alpha > 0, \beta > 0$$

$$F(x;\alpha,\beta) = 1-\exp\left[-\left(\frac{x}{\alpha}\right)^{\beta}\right]$$

其中， α 是尺度参数， β 是形状参数。 Weibull 分布极为通用，因为它可以在 β = 1 时简化为指数分布。在文献和 JMP 中常用的备选参数化是将 σ 用作尺度参数，将 μ 用作位置参数。这些参数很容易通过使用以下定义转换为 α 和 β 参数化：

$$\alpha = \exp(\mu)$$

且

$$\beta = \frac{1}{\sigma}$$

Weibull 分布的 pdf 和 cdf 还表示为对数变换最小极值分布 (SEV)。这将使用位置尺度参数化，且 μ = log(α)， σ = 1/β

$$f(x;\mu,\sigma) = \frac{1}{x\sigma}\phi_{sev}\left[\frac{\log(x)-\mu}{\sigma}\right], \qquad x > 0, \sigma > 0$$

$$F(x;\mu,\sigma) = \Phi_{sev}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

其中

$$\phi_{sev}(z) = \exp[z - \exp(z)]$$

和

$$\Phi_{sev}(z) = 1 - \exp[-\exp(z)]$$

分别为标准化最小极值（μ = 0、σ = 1）分布的 pdf 和 cdf。

对数 Logistic

对数 Logistic 分布的 pdf 在形状上类似于对数正态分布，但尾部较厚。它经常用于对呈现非单调危险率函数特征的数据（如癌症死亡率和金融财富）建模。对数 Logistic pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \frac{1}{x\sigma}\phi_{\text{logis}}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

$$F(x;\mu,\sigma) = \Phi_{\text{logis}}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

其中

$$\phi_{\text{logis}}(z) = \frac{\exp(z)}{[1+\exp(z)]^2}$$

和

$$\Phi_{\text{logis}}(z) = \frac{\exp(z)}{[1+\exp(z)]} = \frac{1}{1+\exp(-z)}$$

分别为标准化 Logistic 或 Logis 分布 ($\mu = 0, \sigma = 1$) 的 pdf 和 cdf。

Fréchet

Fréchet 分布称为对数最大极值分布，或者有时在其参数化为 Weibull 分布的倒数时称为 Fréchet 极大值分布。该分布常用于财务数据。 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \exp\left[-\exp\left(-\frac{\log(x)-\mu}{\sigma}\right)\right]\exp\left(-\frac{\log(x)-\mu}{\sigma}\right)\frac{1}{x\sigma}$$

$$F(x;\mu,\sigma) = \exp\left[-\exp\left(-\frac{\log(x)-\mu}{\sigma}\right)\right]$$

更一般的情况下参数化如下所示：

$$f(x;\mu,\sigma) = \frac{1}{x\sigma}\phi_{\text{lev}}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

$$F(x;\mu,\sigma) = \Phi_{\text{lev}}\left[\frac{\log(x)-\mu}{\sigma}\right]$$

其中

$$\phi_{\text{lev}}(z) = \exp[-z-\exp(-z)]$$

和

$$\Phi_{\text{lev}}(z) = \exp[-\exp(-z)]$$

分别为标准化最大极值 LEV(μ = 0, σ = 1) 分布的 pdf 和 cdf。

正态

正态分布是在多数统计领域中应用最广的分布，因为它相对简单并且易于应用中心极限定理。不过，正态分布极少用于可靠性领域。它对于 μ > 0 且变异系数 (σ / μ) 较小的数据最为有用。由于危险率函数增长没有上限，所以它尤其适用于显现出磨损失效的数据。示例包括白炽灯泡、烤箱加热元件和金属丝的机械强度。 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \frac{1}{\sigma}\phi_{\text{nor}}\left(\frac{x-\mu}{\sigma}\right), \quad -\infty < x < \infty$$

$$F(x;\mu,\sigma) = \Phi_{\text{nor}}\left(\frac{x-\mu}{\sigma}\right)$$

其中

$$\phi_{\text{nor}}(z) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{z^2}{2}\right)$$

和

$$\Phi_{\text{nor}}(z) = \int_{-\infty}^{z} \phi_{\text{nor}}(w)dw$$

分别为标准化正态或 nor(μ = 0, σ = 1) 分布的 pdf 和 cdf。

最小极值 (SEV)

这种非对称 （左偏）分布适用于两种情况。第一种情况是数据指示少量的微弱单元位于分布的下尾处 （该数据指示许多观测的最小数目）。第二种情况是在 σ 相对于 μ 较小时，因为在使用最小极值分布时小于零的概率较小。最小极值分布适用于描述危险率随单元老化而变大的数据。示例包括老年人死亡率和旱季的降雨量。该分布有时称为 Gumbel 分布。pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \frac{1}{\sigma}\phi_{\text{sev}}\left(\frac{x-\mu}{\sigma}\right), \quad -\infty < \mu < \infty, \quad \sigma > 0$$

$$F(x;\mu,\sigma) = \Phi_{\text{sev}}\left(\frac{x-\mu}{\sigma}\right)$$

其中

$$\phi_{\text{sev}}(z) = \exp[z - \exp(z)]$$

和

$$\Phi_{\text{sev}}(z) = 1 - \exp[-\exp(z)]$$

分别为标准化最小极值 SEV($\mu = 0, \sigma = 1$) 分布的 pdf 和 cdf。

Logistic

Logistic 分布的形状类似于正态分布，但尾部较长。Logistic 分布常用于在负数失效时间不构成问题的情况下对寿命数据建模。二值型或有序型响应的 Logistic 回归模型假定 Logistic 分布为潜在分布。 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \frac{1}{\sigma}\phi_{\text{logis}}\left(\frac{x-\mu}{\sigma}\right), \quad -\infty < \mu < \infty \text{ 且 } \sigma > 0$$

$$F(x;\mu,\sigma) = \Phi_{\text{logis}}\left(\frac{x-\mu}{\sigma}\right)$$

其中

$$\phi_{\text{logis}}(z) = \frac{\exp(z)}{[1+\exp(z)]^2}$$

和

$$\Phi_{\text{logis}}(z) = \frac{\exp(z)}{[1+\exp(z)]} = \frac{1}{1+\exp(-z)}$$

分别为标准化 Logistic 或 Logis 分布 ($\mu = 0, \sigma = 1$) 的 pdf 和 cdf。

最大极值 (LEV)

若 σ 相对于 $\mu > 0$ 较小，该右偏分布可用于对失效时间建模。该分布不常用于可靠性领域，但适用于估计自然极端现象，如特大洪水高度或极端风速。 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma) = \frac{1}{\sigma}\phi_{\text{lev}}\left(\frac{x-\mu}{\sigma}\right), \quad -\infty < \mu < \infty \quad \text{且} \quad \sigma > 0$$

$$F(x;\mu,\sigma) = \Phi_{\text{lev}}\left(\frac{x-\mu}{\sigma}\right)$$

其中

$$\phi_{\text{lev}}(z) = \exp[-z - \exp(-z)]$$

和

$$\Phi_{\text{lev}}(z) = \exp[-\exp(-z)]$$

分别为标准化最大极值 LEV($\mu = 0, \sigma = 1$) 分布的 pdf 和 cdf。

指数

单参数和双参数指数分布均用于可靠性领域。双参数指数分布的 pdf 和 cdf 如下：

$$f(x;\theta,\gamma) = \frac{1}{\theta}\exp\left(-\frac{x-\gamma}{\theta}\right), \quad \theta > 0$$

$$F(x;\theta,\gamma) = 1 - \exp\left(-\frac{x-\gamma}{\theta}\right)$$

其中，θ 是尺度参数，γ 既是阈值参数也是位置参数。可靠性分析常常使用单参数指数分布，其 $\gamma = 0$。指数分布适用于描述元件在远远超出期望寿命后显现出磨损的失效时间。该分布具有常数失效率，这意味着对于较小的时间增量，单元的失效与该单元龄无关。指数分布不应用于描述可能遭受疲劳、腐蚀或短期磨损的机械元件的寿命。但是，该分布适用于对特定类型的稳健电子元件建模。它已成功地被用来描述绝缘油和电介质流体的寿命 (Nelson, 1990, p. 53)。

对数广义 Gamma (LogGenGamma)

对数广义 Gamma 分布包含最小极值、最大极值和正态分布。 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma,\lambda) = \begin{cases} \frac{|\lambda|}{\sigma}\phi_{lg}[\lambda\omega + \log(\lambda^{-2});\lambda^{-2}] & \text{若 } \lambda \neq 0 \\ \frac{1}{\sigma}\phi_{nor}(\omega) & \text{若 } \lambda = 0 \end{cases}$$

$$F(x;\mu,\sigma,\lambda) = \begin{cases} \Phi_{lg}[\lambda\omega + \log(\lambda^{-2});\lambda^{-2}] & \text{若 } \lambda > 0 \\ \Phi_{nor}(\omega) & \text{若 } \lambda = 0 \\ 1 - \Phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{若 } \lambda < 0 \end{cases}$$

其中， $-\infty < x < \infty$， $\omega = [x - \mu]/\sigma$，且

$-\infty < \mu < \infty$, $-12 < \lambda < 12$, 且 $\sigma > 0$ 。

请注意

$$\phi_{lg}(z;\kappa) = \frac{1}{\Gamma(\kappa)}\exp[\kappa z - \exp(z)]$$

$$\Phi_{lg}(z;\kappa) = \Gamma_I[\exp(z);\kappa]$$

分别为对数 Gamma 变量的 pdf 和 cdf，并且 $\kappa > 0$ 是形状参数。上面的标准化分布取决于形状参数 κ。

注意：在 JMP 中，广义 Gamma 分布的形状参数 λ 限定在 [-12,12] 之间以便提供数值稳定性。

扩展广义 Gamma (GenGamma)

扩展的广义 Gamma 分布可包含许多其他分布作为特例，如：广义 Gamma、 Weibull、对数正态、 Fréchet、 Gamma 和指数分布。它尤其适用于具有极少删失或没有删失的情况。该分布已成功地用于对人类癌症预后建模。 pdf 和 cdf 分别为：

$$f(x;\mu,\sigma,\lambda) = \begin{cases} \frac{|\lambda|}{x\sigma}\phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{若 } \lambda \neq 0 \\ \frac{1}{x\sigma}\phi_{nor}(\omega) & \text{若 } \lambda = 0 \end{cases}$$

$$F(x;\mu,\sigma,\lambda) = \begin{cases} \Phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{若 } \lambda > 0 \\ \Phi_{nor}(\omega) & \text{若 } \lambda = 0 \\ 1 - \Phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{若 } \lambda < 0 \end{cases}$$

其中 $x > 0$， $\omega = [\log(x) - \mu]/\sigma$，且

$-\infty < \mu < \infty$， $-12 < \lambda < 12$， 且 $\sigma > 0$。

请注意

$$\phi_{lg}(z;\kappa) = \frac{1}{\Gamma(\kappa)}\exp[\kappa z - \exp(z)]$$

$$\Phi_{lg}(z;\kappa) = \Gamma_I[\exp(z);\kappa]$$

分别为标准化对数 Gamma 变量的 pdf 和 cdf，并且 $\kappa > 0$ 是形状参数。

上面的标准化分布取决于形状参数 κ。Meeker and Escobar (1998, ch. 5) 提供了关于扩展的广义 Gamma 分布的详细解释。

注意：在 JMP 中，广义 Gamma 分布的形状参数 λ 限定在 [-12,12] 之间以便提供数值稳定性。

含有阈值参数的分布

阈值分布是带有阈值参数的对数 - 位置 - 尺度分布。上述有些分布通过添加阈值参数 （表示为 γ）进行了推广。添加该阈值参数后，分布的左端点偏移了 0 点位置。阈值参数有时称为偏移、最小值或保证参数，因为所有单元都至少生存到阈值时间。请注意，尽管添加阈值参数使分布在时间轴上的位置发生偏移，但分布的形状和散布程度并不受影响。阈值分布适用于拟合中度到重度偏移的分布。对数 - 位置 - 尺度阈值分布的 pdf 和 cdf 的一般形式如下所示：

$$f(x;\mu,\sigma,\gamma) = \frac{1}{\sigma(x-\gamma)}\phi\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right], \quad x > \gamma$$

$$F(x;\mu,\sigma,\gamma) = \Phi\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right]$$

其中，ϕ 和 Φ 分别是特定分布的 pdf 和 cdf。针对 Weibull、对数正态、Fréchet 和对数 Logistic 分布的特定阈值分布的示例显示如下；其中相应地适当替换了最小极值、正态、最大极值和 logis 的 pdf 和 cdf。

注意：若最小的观测是一个失效（而非删失），则 JMP 会围绕该点创建小区间，并将该观测视为区间删失。这种围绕失效的填充限定了对数似然函数并改善了估计效果。若最小观测删失，则不向观测添加任何额外填充。

阈值 Weibull

三参数 Weibull 分布的 pdf 和 cdf 如下：

$$f(x;\mu,\sigma,\gamma) = \frac{1}{(x-\gamma)\sigma}\phi_{sev}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right], \quad x>\gamma, \sigma>0$$

$$F(x;\mu,\sigma,\gamma) = \Phi_{sev}\left(\frac{\log(x-\gamma)-\mu}{\sigma}\right) = 1-\exp\left[-\left(\frac{x-\gamma}{\alpha}\right)^{\beta}\right], \quad x>\gamma$$

其中，$\mu=\log(\alpha)$，$\sigma=1/\beta$，其中

$$\phi_{sev}(z) = \exp[z-\exp(z)]$$

和

$$\Phi_{sev}(z) = 1-\exp[-\exp(z)]$$

分别为标准化最小极值 SEV($\mu = 0, \sigma = 1$) 分布的 pdf 和 cdf。

阈值对数正态

三参数对数正态分布的 pdf 和 cdf 如下：

$$f(x;\mu,\sigma,\gamma) = \frac{1}{\sigma(x-\gamma)}\phi_{nor}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right], \quad x>\gamma$$

$$F(x;\mu,\sigma,\gamma) = \Phi_{nor}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right]$$

其中

$$\phi_{nor}(z) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{z^2}{2}\right)$$

且

$$\Phi_{\text{nor}}(z) = \int_{-\infty}^{z} \phi_{\text{nor}}(w)dw$$

分别为标准化正态或 N(μ = 0, σ = 1) 分布的 pdf 和 cdf。

阈值 Fréchet

三参数 Fréchet 分布的 pdf 和 cdf 如下：

$$f(x;\mu,\sigma,\gamma) = \frac{1}{\sigma(x-\gamma)}\phi_{\text{lev}}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right], \quad x > \gamma$$

$$F(x;\mu,\sigma,\gamma) = \Phi_{\text{lev}}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right]$$

其中

$$\phi_{\text{lev}}(z) = \exp[-z-\exp(-z)]$$

和

$$\Phi_{\text{lev}}(z) = \exp[-\exp(-z)]$$

分别为标准化最大极值 LEV(μ = 0, σ = 1) 分布的 pdf 和 cdf。

阈值对数 Logistic

三参数对数 Logistic 分布的 pdf 和 cdf 如下：

$$f(x;\mu,\sigma,\gamma) = \frac{1}{\sigma(x-\gamma)}\phi_{\text{logis}}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right], \quad x > \gamma$$

$$F(x;\mu,\sigma,\gamma) = \Phi_{\text{logis}}\left[\frac{\log(x-\gamma)-\mu}{\sigma}\right]$$

其中

$$\phi_{\text{logis}}(z) = \frac{\exp(z)}{[1+\exp(z)]^2}$$

和

$$\Phi_{\text{logis}}(z) = \frac{\exp(z)}{[1+\exp(z)]} = \frac{1}{1+\exp(-z)}$$

分别为标准化 Logistic 或 Logis 分布 (μ = 0, σ = 1) 的 pdf 和 cdf。

缺陷子总体分布

在可靠性实验中，有时仅有总体的一小部分存在导致失效的特定缺陷。由于并不是所有单元都可能导致失效，所以使用常规失效分布不合适并可能生成误导的结果。使用缺陷子总体分布选项可对仅在子总体上出现的失效建模。提供以下缺陷子总体分布：

- 缺陷子总体对数正态
- 缺陷子总体 Weibull
- 缺陷子总体对数 Logistic
- 缺陷子总体 Fréchet

缺陷子总体分布的 pdf 和 cdf 定义如下：

$$f(t) = \left[p\frac{1}{t\sigma}\right]\phi\left[\frac{(\log(t)-\mu)}{\sigma}\right]$$

$$F(t) = p\Phi\left[\left(\frac{(\log(t)-\mu)}{\sigma}\right)\right]$$

其中：

p 是缺陷子总体分数

t 是寿命事件的测量时间

通过使用相应缺陷子总体的 pdf 和 cdf 计算通常的最大似然估计，以此方式估计 μ 和 σ。

$\phi(z)$ 和 $\Phi(z)$ 分别为标准分布的密度和累积分布函数。例如，对于 Weibull 分布，

$\phi(z) = \exp(z-\exp(z))$ 且 $\Phi(z) = 1 - \exp(-\exp(z))$。

有关缺陷子总体模型的详细信息，请参见 Tobias and Trindad (2012, p. 321)。

缺陷子总体模型在 Meeker and Escobar (1998, ch. 11) 中亦称**有限失效总体**模型。

零泛滥分布

若一定比例 (p) 的数据在 $t = 0$ 时失效，就会使用零泛滥分布。当数据包含零的个数超出标准模型应有的个数时，就会造成零泛滥。在“寿命分布”平台中当事件时间数据以零作为最小值时，有四种零泛滥分布可用：这些分布包括：

- 零泛滥对数正态（ZI 对数正态）
- 零泛滥 Weibull (ZI Weibull)
- 零泛滥对数 Logistic（ZI 对数 Logistic）
- 零泛滥 Fréchet (ZI Fréchet)

零泛滥分布的 pdf 和 cdf 定义如下：

$$f(t) = \left[(1-p)\frac{1}{t\sigma}\right]\phi\left[\frac{(\log(t)-\mu)}{\sigma}\right]$$

$$F(t) = p + (1-p)\Phi\left[\left(\frac{(\log(t)-\mu)}{\sigma}\right)\right]$$

其中：

p 是零数据值的比例

t 是寿命事件的测量时间

μ 和 σ 通过在从原始数据中删除零值后计算通常的最大似然估计来估计

$\phi(z)$ 和 $\Phi(z)$ 分别为标准分布的密度和累积分布函数。例如，对于 Weibull 分布，

$\phi(z) = \exp(z-\exp(z))$ 且 $\Phi(z) = 1 - \exp(-\exp(z))$。

有关零泛滥分布的详细信息，请参见 Lawless (2003, p. 34)。替换 $p = 1 - p$ 和 $S_1(t) = 1 - \Phi(t)$，以获取如上所示的形式。

有关可靠性分布的详细信息，请参见 Tobias and Trindade (1995, p. 232)。该参考文献提供了混合分布的一般形式。使用 Tobias and Trindade 中的参数化，可通过替换 $\alpha = p$、$F_d(t) = 1$ 和 $F_N(t) = \Phi(t)$ 获得上述形式。

Bayes 估计的先验分布

以下分布可用于 “先验位置尺度”：

- 正态/对数正态分布，带有位置 (mu) 和尺度 (sigma) 这两个超参数。相关定义，请参见第 73 页的“对数正态”和第 76 页的“正态”。
- 均匀分布，带有“下限”和“上限”这两个超参数，用于定义“均匀”分布的支持范围。
- Gamma 分布，带有“形状”和“尺度”这两个超参数。使用 k/theta 参数化和概率密度函数。
- 点质量分布，带“位置”超参数。这是退化的先验；对于我们所指定先验分布的参数，只有一个可能值。这唯一的可能值等于为该“位置”超参数输入的值。

以下分布可用于“先验分位数参数”：

- 正态/对数正态分布（99% 的概率范围），使用分布的 0.005 和 0.995 百分位数指定先验分布。JMP 停止支持 mu 和 sigma。
- 均匀分布，带有“下限”和“上限”这两个超参数，用于定义“均匀”分布的支持范围。
- 对数均匀分布，带有“下限”(a) 和“上限”(b)。该分布是在 Log(a) 和 Log(b) 之间的对数尺度上均匀的分布。
- 点质量分布，带“位置”超参数。这是退化的先验；对于我们所指定先验分布的参数，只有一个可能值。这唯一的可能值等于为该“位置”超参数输入的值。

以下分布可用于 “先验失效概率”：

- Beta 分布，由概率密度函数来表征。
  - 使用估计值和误差百分比（均值和方差）指定 Beta 先验值。均值等于在“估计值”中输入的数字，方差等于 (误差百分比 / 100 * 估计值)^2。

– 使用分布的 0.005 和 0.995 百分位数指定 Beta 先验值。JMP 停止支持超参数。

竞争原因详细信息

对于竞争原因模型，聚合分布的闭型定义如下：

$$F(x) = 1 - \prod_{i=1}^{k} [1 - F_i(x)]$$

其中，$F_i(x), i = 1, \ldots, k$ 是与原因对应的单个失效分布。置信限是现成的，因为所有涉及的估计值都是 MLE。

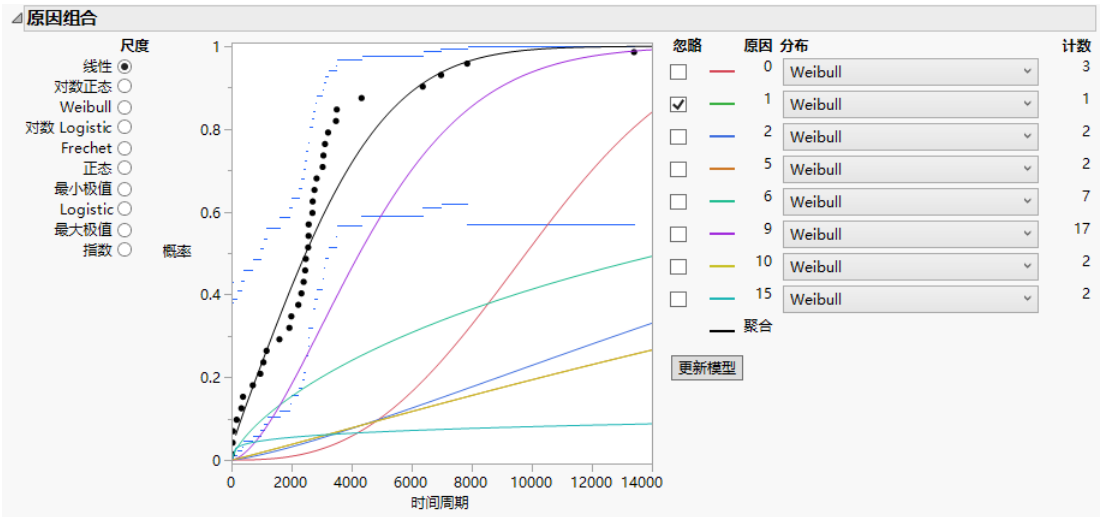
将固定参数模型指定为原因的分布

若为某个原因指定了固定参数模型，您必须在该原因的“单个原因”报表中固定该参数。在“寿命分布 - 失效原因: <名称>”报表中的所需“参数估计值”报表中固定该参数。当您点击“更新模型”时，固定参数将成为聚合分布的一部分。

本例演示了如何在聚合分布中包含固定参数模型：

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Appliance.jmp。
2. 选择**分析 > 可靠性和生存 > 寿命分布**。
3. 选择**时间周期**并点击 **Y，事件时间**。
4. 选择**原因编号**并点击**失效原因**。
5. 将**似然**选作 “置信区间方法”。
6. 选择**允许失效模式使用固定参数模型**。
7. 点击**确定**。

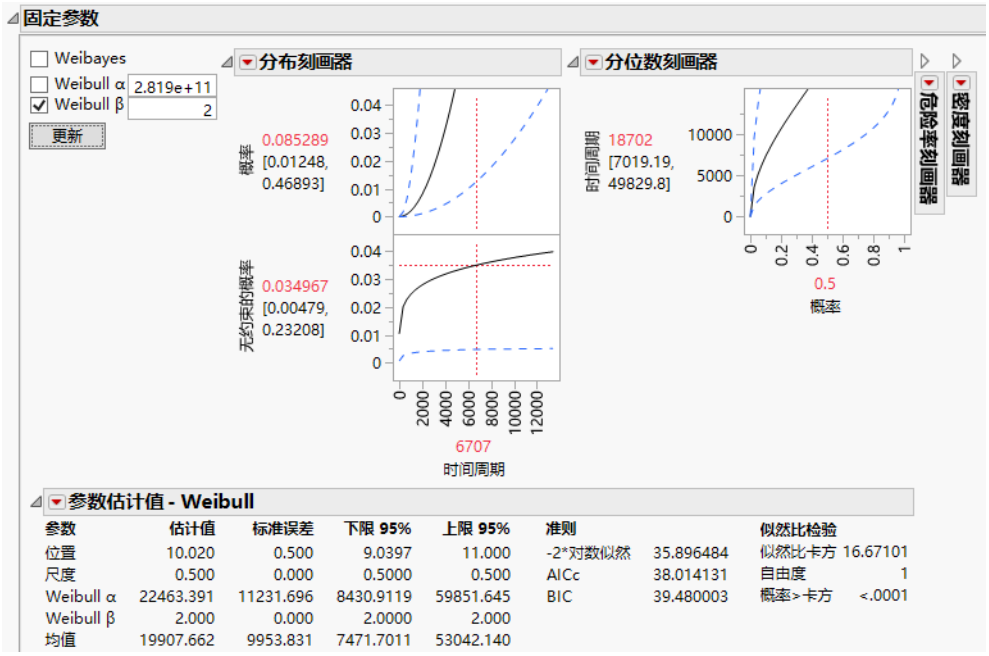
图 3.25 忽略了 “原因 1”的固定参数模型



默认情况下，忽略 “原因 = 1”，因为没有足够的数据。不过，您不希望忽略该原因。

8. 打开“原因 1”对应的“单个原因”报表。该报表称为“寿命分布 - 失效原因: 1 失效计数: 1”。
9. 点击 “参数估计值 - Weibull”旁边的红色小三角并选择**固定参数**。
10. 选择 Weibull beta 并键入 2。
11. 点击**更新**。

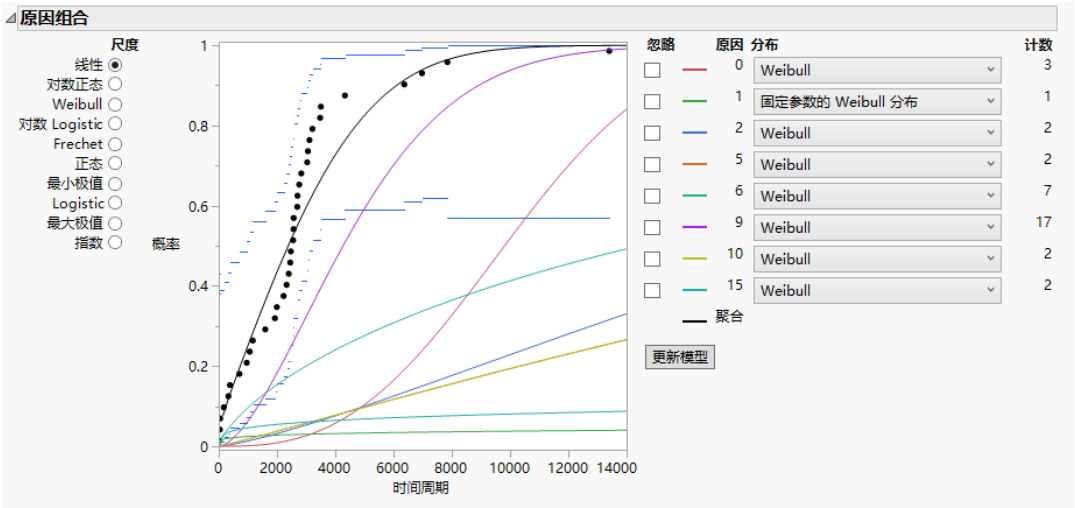
图 3.26 指定了 Weibull Beta 的固定参数模型



在 “参数估计值 - Weibull”报表中，假定 β 等于 2， alpha 参数估计为 22463.391。现在您可以将其用于 “原因 =1”的失效分布。

12. 向上滚动到报表窗口顶部的 “原因组合”。
13. 为 “原因 1”取消选择**忽略**。
14. 对于 “原因 1”的分布，选择**固定参数的 Weibull 分布**。
15. 点击**更新模型**。

图 3.27 显示 “原因 1”的更新模型



现在聚合模型在总体竞争原因模型中使用“原因 1”的“固定参数的 Weibull 分布”结果。

为原因指定 Bayes 模型

为某个原因指定 Bayes 模型的步骤类似于第 84 页的“将固定参数模型指定为原因的分布”中所述的步骤。在所需的“Bayes 估计”报表中定义该模型，该报表位于针对单个原因的“寿命分布”报表的“统计学”下方的相应“参数估计值”分级显示项中。请参见第 43 页的“Bayes 估计 - <分布名称>”。

要将 Bayes 模型并入聚合模型，其他原因的非 Bayes 分布必须能接受基于模拟的框架的处理。例如，假定某个模型具有两个失效原因。一个原因使用 Weibull 分布建模，另一个使用 Bayes 方法来估计第二个 Weibull 分布的参数。第一个 Weibull 分布的参数（由向量 θ_1 表示）使用最大似然来估计。第二个 Weibull 分布的参数 θ_2 使用 Bayes 方法来估计。

聚合混合分布 （表示为 $F(x, \theta_1, \theta_2)$）的分位数和中位数通过如下方式来获取：

- 对第一个 Weibull 分布执行参数 Bootstrap，生成来自最大似然估计值 $\hat{\theta}_1$ 的渐近分布的随机样本。将来自 $\hat{\theta}_1$ 的渐近分布的抽样值表示为 θ_1^*。
- 从 θ_2 的后验分布中抽取某个样本，表示为 θ_2^*。
- 对于 θ_1^* 和 θ_2^* 的每一组值，获取 $F(x, \theta_1, \theta_2)$ 的估计值（表示为 $F^*(x, \theta_1, \theta_2)$）。
- 值 $F^*(x, \theta_1, \theta_2)$ 用于获取聚合分布的分位数和中位数的估计值。这些值是在给定的 x 值处显示在“分布刻画器”中的值。

为原因指定 Weibayes 模型

为某个原因指定 Weibayes 模型的步骤类似于第 84 页的“将固定参数模型指定为原因的分布”中所述的步骤。 在针对该原因的“寿命分布”报表中的“统计学”下方的“参数估计值 -

Weibull”分级显示项中，选择“固定参数”选项。在“固定参数”报表中，选中“Weibayes”选项。Weibayes 模型被视为 Bayes 模型，并且从参数 alpha 的后验分布中抽取 Bootstrap 样本。请参见 Liu and Wang (2013)。

剩余寿命均值计算器

使用红色小三角菜单中的“配置”选项可以为计算剩余寿命均值时使用的模拟失效时间的个数设置一个值。用 m 来表示该值。

要获取时间 t 处的剩余寿命均值的估计值，需要从聚合分布（以生存到时间 t 为条件）中抽取 m 个样本。然后计算这些样本的平均值。

要计算剩余寿命均值的置信限，您必须在“配置”窗口中选择该框。随后您可以选择设置 Bootstrap 样本数。用 n 来表示该值。

要计算置信区间，需要从 MLE 的渐近分布或使用 Bayes 推论推导出的后验分布中抽取 n 个参数估计值样本。对于每个参数值样本，都形成一个聚合分布，从中抽取 m 个样本来计算剩余寿命均值。使用 n 个剩余寿命均值的样本来构造置信区间。

“拟合混合模型”保存预测公式

本节提供的公式用于计算“拟合混合模型”报表选项“保存预测”所保存的列中的值。

考虑以下符号：

$\hat{p}_i$ 是混合比例 w_i 的估计值

$\hat{F}_i$ 是估计概率分布函数 F_i

$\hat{f}_i$ 是 F_i 的估计概率密度函数

- 若观测 y 未删失，保存值则由以下公式计算得出：

$$\frac{\hat{p}_i \hat{f}_i(y)}{\sum_{i=1}^{k} \hat{p}_i \hat{f}_i(y)}$$

- 若观测删失，则通过用以下值替换未删失观测的公式中的估计密度值来得到保存值：

$\hat{F}_i(y)$ 用于右删失

$1 - \hat{F}_i(y)$ 用于左删失

$\hat{F}_i(y_{high}) - \hat{F}_i(y_{low})$ 用于区间删失

关于中位数秩回归的注释

当没有删失行和“权重”列时，“寿命分布”平台提供 Weibull 参数估计值的中位数秩回归 (MRR)。在“寿命分布”报表中显示 MRR 估计值之前，必须满足以下条件：

- 已选定下面的“寿命分布”平台首选项：无删失观测时，将报告基于中位数秩回归的 Weibull 参数估计值。
- 分析中没有删失观测。换言之，所有观测都表示失效时间。
- 未指定“权重”列。

若满足上述所有条件，则修改“参数估计值 - Weibull”报表以显示 MRR 估计值和通常的最大似然估计值 (MLE)。参数估计值表中的列标题以 MLE 或 MRR 作为前缀，表示生成每列估计值所用的估计方法。没有可用于 MRR 估计值的置信区间。

警告：不推荐使用 MRR 估计值。中位数秩回归使用最小二乘法估计而不是最大似然来生成估计值。目前还没有公认的方法对删失数据应用最小二乘估计。而且，Genschel and Meeker (2010) 显示 MLE 方法更为精确。他们提供的模拟结果基于真实估计值是已知的这一事实；但实际上，MRR 估计值的准确性却是未知的。由于 MRR 是一个松散定义的权宜下特殊估计过程，不同的软件包会对相同的数据生成不同的 MRR 估计值。

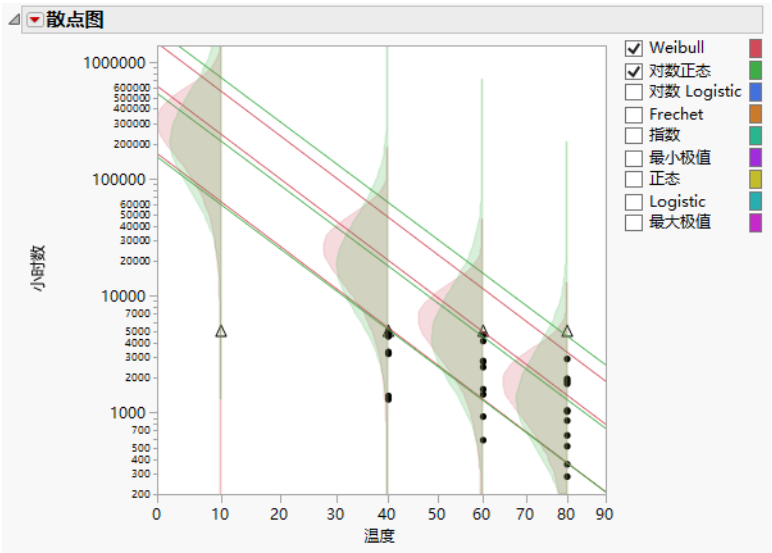
第 4 章

以 X 拟合寿命

拟合事件时间数据的单因子模型

“以 X 拟合寿命”平台帮助您分析仅有一个因子情况下的寿命事件。您可以选择使用各种变换对事件与因子之间的关系建模，或是创建自定义数据变换。您还可以灵活地比较同一因子水平下的不同分布，以及在不同因子水平之间比较同一分布。

图 4.1 显示不同分布和因子水平的散点图



目录

“以 X 拟合寿命”平台概述 93
“以 X 拟合寿命”平台的示例 93
启动“以 X 拟合寿命”平台 96
“以 X 拟合寿命”报表 98
数据汇总 98
散点图 98
非参数叠加 100
比较 101
结果 104
定制关系 114
“以 X 拟合寿命”平台选项 115
“以 X 拟合寿命”平台的更多示例 116
电容示例 116
定制关系示例 118

“以 X 拟合寿命”平台概述

“以 X 拟合寿命”平台提供加速寿命试验分析所需的工具。加速试验通常用于工业，用来在相对较短的时间范围内提供关于产品或元件的失效时间信息。常见的加速因子包括温度、电压、压力和使用率。可以外推结果，以获取在加速因子的较低正常操作水平下的失效时间估计值。这些结果用于评估可靠性、检测和更正失效模式、比较制造商和认证元件。

“以 X 拟合寿命”平台包括许多常用的变换，用于对事件与所关注因子之间的物理和化学关系建模。示例包括：使用 Arrhenius（摄氏温度、华氏温度和绝对温度）关系的时间加速因子和电压加速机制进行变换。线性、对数、Logit、倒数、平方根、Box-Cox、位置、位置和尺度以及定制加速模型也包含在该平台中。

您可以使用“实验设计”>“加速寿命试验设计”平台来设计加速寿命试验实验。请参见《实验设计指南》。

Meeker and Escobar (1998, p. 495) 提供了以下用于分析加速寿命数据的策略：

1. 以图形方式检查数据。其中一种有用的直观演示数据的方法就是检查加速因子与失效时间变量的散点图。
2. 针对加速因子不同水平下的数据逐一拟合分布。对不同的假设分布重复执行该步骤。
3. 用失效时间变量与加速因子之间看似合理的关系拟合总体模型。
4. 将步骤 3 中的模型与步骤 2 中的各个分析进行比较，评估总体模型是否失拟。
5. 执行残差和各种诊断分析以验证模型假设。
6. 评估数据的貌似真性以进行推断。

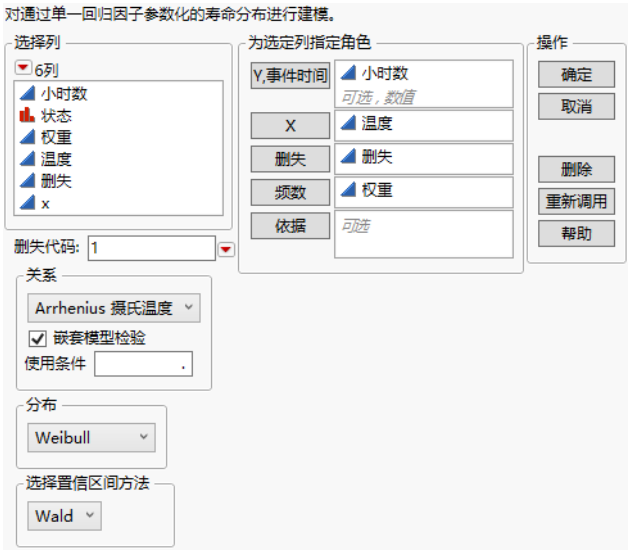
“以 X 拟合寿命”平台的示例

本例使用 Meeker and Escobar (1998) 中的 Devalt.jmp 样本数据表，该数据表位于样本数据的 Reliability 文件夹中。其中包含某设备在加速工作温度下的失效时间数据。在正常工作温度 10 摄氏度时未记录任何失效时间观测；其他所有观测显示为 40、 60 和 80 摄氏度这三个加速温度水平下的失效时间或删失值。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Devalt.jmp。
2. 选择**分析** > **可靠性和生存** > **以 X 拟合寿命**。
3. 选择**小时数**并点击 **Y，事件时间**。
4. 选择**温度**并点击 **X**。
5. 选择**删失**并点击**删失**。
6. 保留**删失代码**为 1。
7. 选择**权重**并点击**频数**。
8. 保留 Arrhenius **摄氏温度**作为关系，并保持选定**嵌套模型检验**选项。

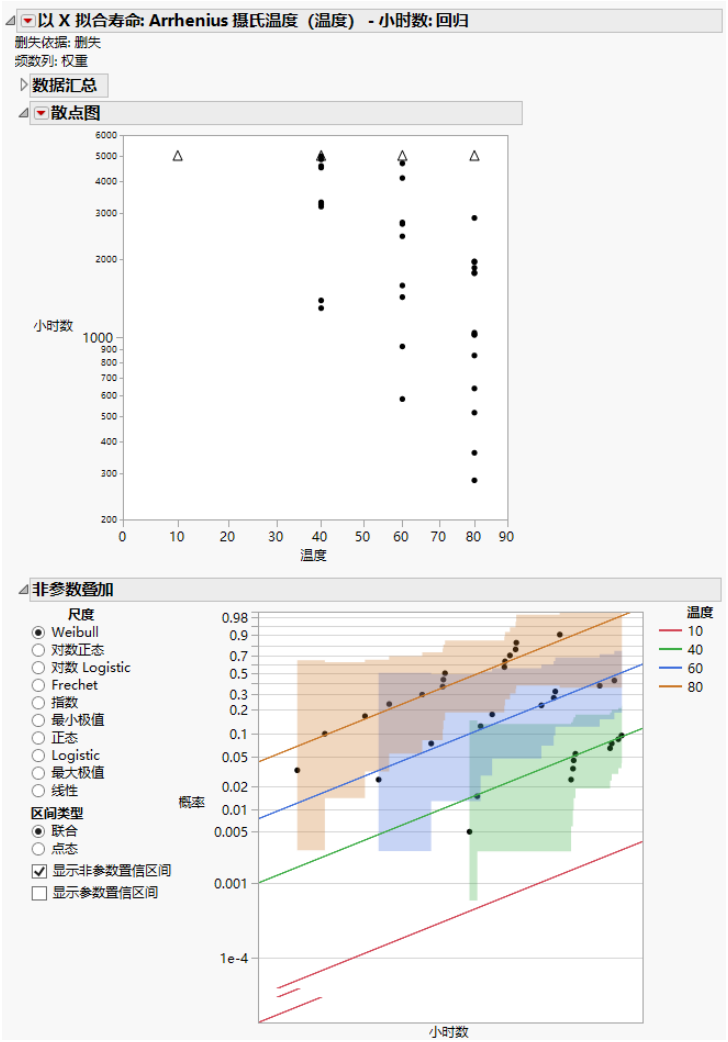
9. 将 Weibull 选作分布。
10. 将 Wald 选作置信区间方法。

图 4.2 “以 X 拟合寿命”启动窗口



11. 点击**确定**。

图 4.3 Devalt.jmp 数据的 “以 X 拟合寿命”报表窗口



该报表窗口显示汇总数据、诊断图、比较数据和结果，包括详细的统计学和预测刻画器。为每个选定分布都显示单独的结果部分。每个指定分布都包含“分布刻画器”、“分位数刻画器”、“危险率刻画器”、“密度刻画器”和“加速因子刻画器”。

启动 “以 X 拟合寿命” 平台

通过选择**分析 > 可靠性和生存 > 以 X 拟合寿命**启动 “以 X 拟合寿命” 平台。有关 “选择列” 红色小三角菜单中选项的详细信息，请参见 《使用 JMP》。

图 4.4 “以 X 拟合寿命” 启动窗口

“以 X 拟合寿命” 启动窗口包含以下选项：

Y，事件时间 标识事件时间（如，失效时间）或删失时间。在区间删失时，指定两个 Y 变量，其中一个 Y 变量提供每个单元的下限，另一个 Y 变量提供每个单元的上限。有关删失的详细信息，请参见 “寿命分布” 一章中第 35 页的 “事件图”。

X 标识加速因子。

删失 标识删失观测。从 “选择列” 列表下方的 “删失代码” 菜单中选择用于标识右删失观测的值。仅当输入一个 Y 时才使用 “删失” 列。

频数 标识有多个单元时的频数或观测计数。若值为 0 或正整数，则该值表示记录多个单元时每行的频数或观测计数。

依据 标识用于创建报表的一个列，该报表包括变量的每个水平的单独分析。

删失代码 标识 “删失” 列中指定右删失观测的值。选定某个 “删失” 列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中

进行选择。您还可以在框中输入不同值。若 “删失”列包含 “值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

关系 标识事件与加速因子之间的关系。表 4.1 定义每种关系的模型。

表 4.1 关系模型选项

关系	模型
Arrhenius 摄氏温度	μ = b0 + b1 * 11604.5181215503 / (X + 273.15)
Arrhenius 华氏温度	μ = b0 + b1 * 11604.5181215503 / ((X + 459.67) / 1.8)
Arrhenius 绝对温度	μ = b0 + b1 * 11604.5181215503 / X
电压	μ = b0 + b1 * log(X)
线性	μ = b0 + b1 * X
对数	μ = b0 + b1 * log(X)
Logit	μ = b0 + b1 * log(X / (1 - X))
倒数	μ = b0 + b1 / X
平方根	μ = b0 + b1 * sqrt(X)
Box-Cox	μ = b0 + b1 * BoxCox(X)
位置	表示 μ 对于 X 的每个水平都不同
位置和尺度	表示 μ 和 σ 对于 X 的每个水平均不同（等同于 X 作为“依据”变量时的“寿命分布”拟合）
定制	用户定义的 μ 和 σ

若您选择 Box-Cox，文本编辑框显示在 “使用条件”选项之下。使用该框可指定 lambda 值。指定的 λ 的 BoxCox(X) 变换定义如下：

$$x_i^{(\lambda)} = \begin{cases} \dfrac{x_i^{\lambda} - 1}{\lambda} & \text{若 } \lambda \neq 0 \\ \ln(x_i) & \text{若 } \lambda = 0 \end{cases}$$

若想要对模型使用 “定制”关系，请参见第 114 页的 “定制关系”。

嵌套模型检验 向报表窗口附加非参数叠加图、嵌套模型检验和多重概率图。

使用条件 支持您为加速因子的说明变量 X 输入值。您还可以在启动该平台后使用 “以 X 拟合寿命”红色小三角菜单中的 “设置时间加速使用条件”选项设置使用条件值。

分布 一次指定一种分布 （Weibull、对数正态、对数 Logistic、 Fréchet、最小极值、正态、Logistic、最大极值或指数分布）。对数正态是默认设置。

选择置信区间方法 显示用于计算参数置信区间的方法。默认为 Wald，但您可以改为选择 “似然”。Wald 方法是一种近似方法，运行较快。似然方法提供更精确的参数，但计算时间较长。

注意：“置信区间方法”首选项支持您选择 “似然”方法作为默认的置信区间方法。您可以在**首选项 > 平台 > 以 X 拟合寿命**中更改该首选项。

“以 X 拟合寿命”报表

初始报表窗口包括以下部分：

- 第 98 页的 “数据汇总”
- 第 98 页的 “散点图”
- 第 100 页的 “非参数叠加”
- 第 101 页的 “比较”

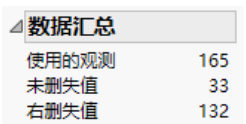
  可以查看和比较分布、分位数、危险率、密度和加速因子刻画器，以及比较准则下方的准则值。
- 第 104 页的 “结果”

  可以检查和比较每一选定分布的参数估计值、协方差矩阵、嵌套模型检验和诊断。您还可以执行定制估计并获取分布参数的 Bayes 估计。

数据汇总

“数据汇总”部分显示总观测数、未删失值的数目和删失值（右删失、左删失和区间删失）的数目。图 4.5 显示了 Devalt.jmp 样本数据表的汇总数据。

图 4.5 数据汇总示例



散点图

说明变量与寿命事件的散点图显示在报表窗口的顶部。对于 Devalt.jmp 样本数据，散点图显示“温度 - 小时数”图。表 4.2 指示如何在报表窗口的散点图中表示每种失效类型。要增大图上标记的大小，右击该图，先选择**标记大小**，再选择所列的一个标记大小。

图 4.6 “温度 - 小时数”散点图

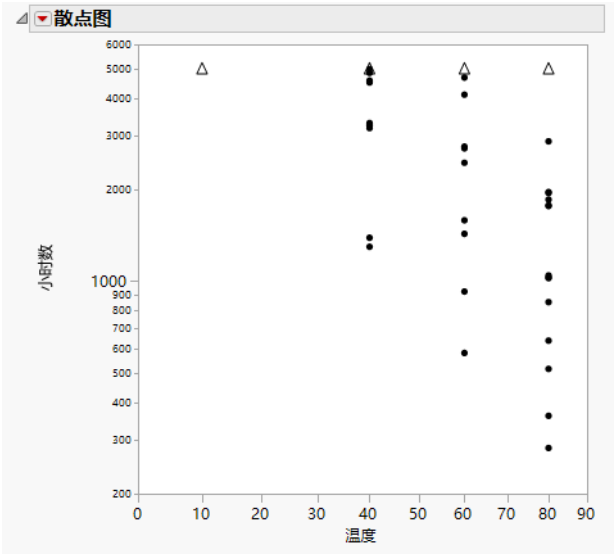


表 4.2 失效和删失观测的散点图表示

事件	散点图表示
失效	点
右删失	向上的三角
左删失	向下的三角
区间删失	向下的三角落在向上的三角之上，二者通过一条实线连接

散点图选项

“散点图”红色小三角菜单包含以下选项：

添加密度曲线 通过在加速因子范围内输入任意值，指定您所需的密度曲线，一次指定一条曲线。您随后可以通过选中在添加曲线后出现的相应复选框来选择不同的分布。

删除密度曲线 显示之前输入的密度曲线值。通过选中适当的复选框来删除曲线。

显示密度曲线 显示密度曲线。若拟合的是**位置**或**位置和尺度**模型，或是在启动窗口中选定了**嵌套模型检验**，则将显示所有给定说明变量水平的密度曲线。创建曲线之后，使用**显示密度曲线**选项可在图中切换曲线的显示。

添加多条分位数线 指定所需的分位数线，一次指定三条。您可以通过连续选择**添加多条分位数线**添加更多分位数。默认分位数值为 0.1、0.5 和 0.9。无效的分位数值（如缺失值）将被忽略。您可以按需仅输入一个分位数值，将其他条目留空。

显示分位数线置信区间带 显示或隐藏分位数线周围的置信区间。

设置分位数线置信区间带的水平 指定分位数线周围置信区间的置信水平。

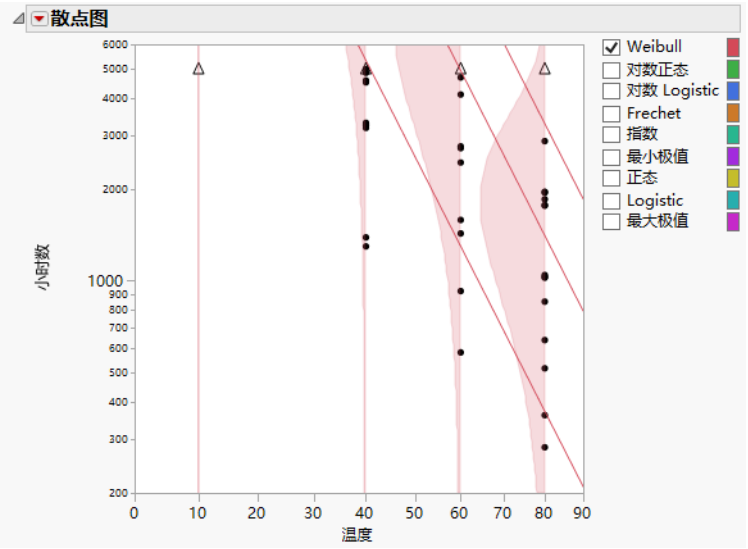
删除多条分位数线 显示之前输入的分位数值。通过选中适当的复选框来删除分位数线。

转置坐标轴 交换水平轴和垂直轴。

使用变换尺度 散点图的默认视图包含变换尺度。选择该选项可在水平轴的线性尺度和非线性尺度之间切换。

图 4.6 显示了初始散点图；图 4.7 显示了选定**显示密度曲线**和**添加多条分位数线**选项后生成的散点图，其中显示了 Weibull 分布的各种**温度**水平下的曲线和分位数线。您还可以查看其他分布所有**温度**水平下的密度曲线。可以一次选定一个分布，也可以通过选中所需分布名称左侧的框来同时查看这些分布。

图 4.7 选定了密度曲线和分位数线选项的散点图



非参数叠加

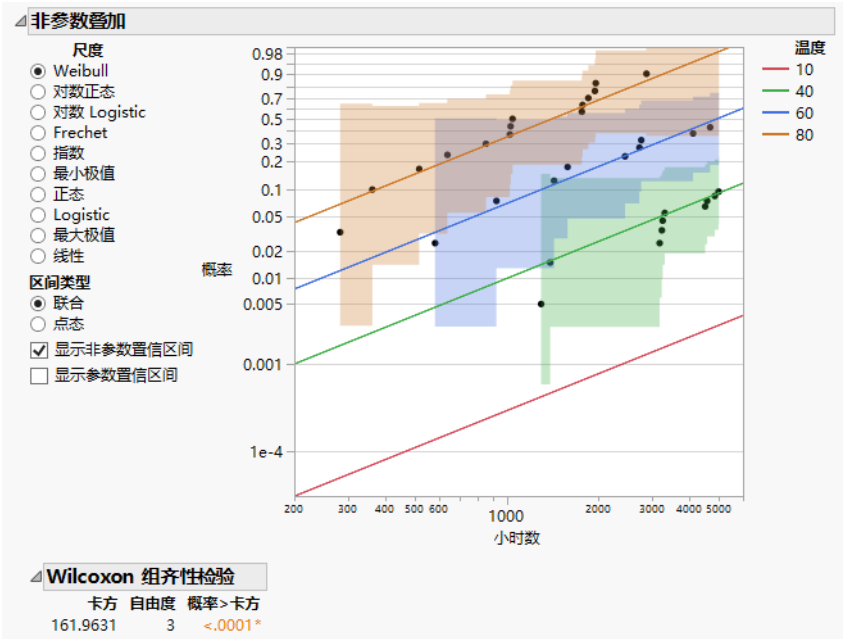
“非参数叠加”图在散点图之后显示。通过检查该图可以很容易检测出各分组之间的差异。您可以选择不同的尺度来查看这些差异。您可以更改非参数拟合概率图上的区间类型，即改为**联合**或**点态**（选定**显示非参数置信区间**时显示的结果）。还可以选择是显示参数置信区间还是显示非参数置信区间。

点态估计值在图上显示 95% 点态置信带，而联合置信区间在图上显示所有分组的联合置信带。Meeker and Escobar (1998, ch. 3) 讨论了点态和联合置信区间，以及在寿命分析中涉及联合置信区间的动因。

Wilcoxon 组齐性检验

对于本例，图 4.8 中显示的 Wilcoxon 组齐性检验指出分组之间存在差异。高卡方值和低 p 值与“非参数叠加”图中观察到的各个温度组之间的差异是一致的。

图 4.8 Devalt.jmp 的 “非参数叠加”图和 Wilcoxon 检验



比较

图 4.9 中的 “比较”报表部分针对您在 “非参数叠加”部分中所选的分布显示一些刻画器，并包括以下选项卡：

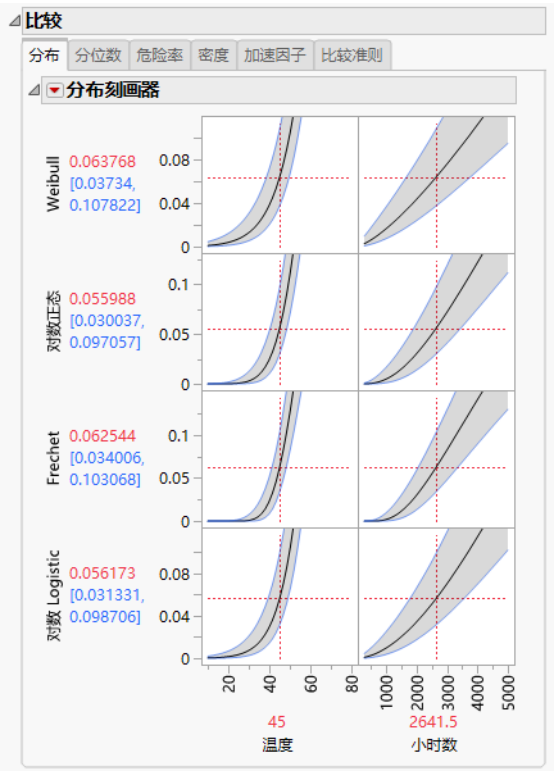
- 分布
- 分位数
- 危险率
- 密度
- 加速因子
- 比较准则

要显示特定的刻画器，请在“非参数叠加”部分中选择适当的分布选项。

刻画器

前五个选项卡显示所选分布的刻画器。前四个刻画器中显示的曲线同时对应于事件时间和解释变量。“加速因子”刻画器选项卡只与加速因子（解释变量）相对应。图 4.9 显示了 Weibull、对数正态、Fréchet 和对数 Logistic 分布的“分布刻画器”。

图 4.9 分布刻画器

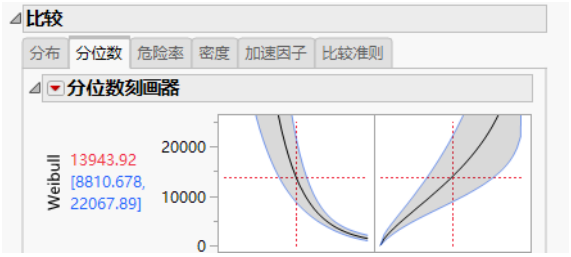


可比较的结果显示在**分位数**、**危险率**和**密度**选项卡上。分布、分位数、危险率、密度和加速因子这些刻画器的行为类似于其他平台中的“预测刻画器”。例如，可拖动**温度**和**小时数**的垂线来查看每个分布值如何随着温度和时间变化。有关“预测刻画器”的详细解释，请参见《刻画器指南》。

分位数

您可以使用“分位数”刻画器进行外推。假定数据通过 Weibull 分布表示。通过查看图 4.11 中的 Weibull“加速因子刻画器”，您可以看到对于使用条件温度 10 摄氏度，45 摄氏度时的加速因子为 17.42132。选择**分位数**选项卡可查看 Weibull 分布的“分位数刻画器”。选择并拖动概率图中的垂线，让概率读数为 0.5。通过查看图 4.10（其中的“概率”设置为 0.5），您可以发现 45 摄氏度时与失效概率 0.5 对应的分位数为 13943.92 小时。所以，在 10 摄氏度时，预计 50% 的单元在 13943.92 * 17.42132 = 242921 小时的时候失效。

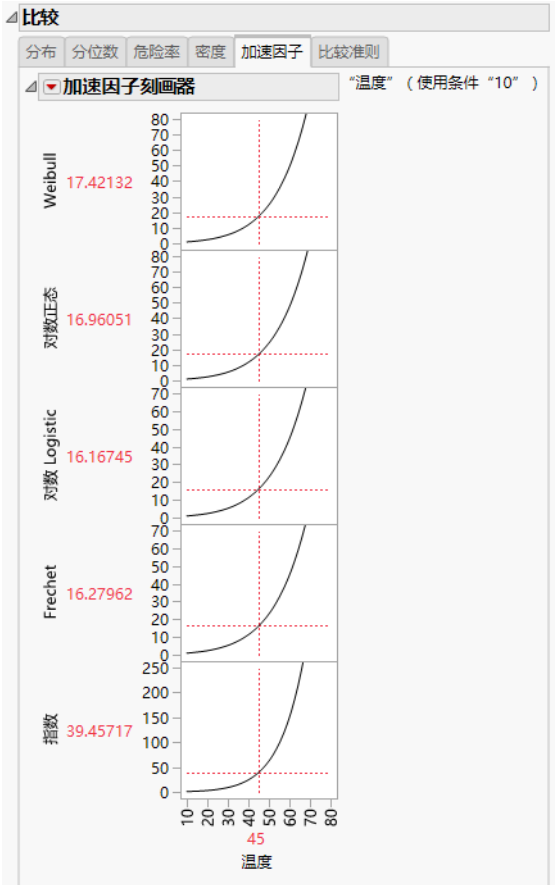
图 4.10 Devalt.jmp 的 Weibull “分位数刻画器”



加速因子

选择**加速因子**选项卡可显示每个指定分布的事件时间变量的“加速因子刻画器”。要生成图 4.11，请从“以 X 拟合寿命”红色小三角菜单中选择**拟合所有分布**。通过从“以 X 拟合寿命”红色小三角菜单中选择**设置时间使用条件**并输入所需值，修改解释变量的使用条件值。请注意，解释变量和使用条件值显示在刻画器标题旁边。

图 4.11 Devalt.jmp 的 “加速因子刻画器”



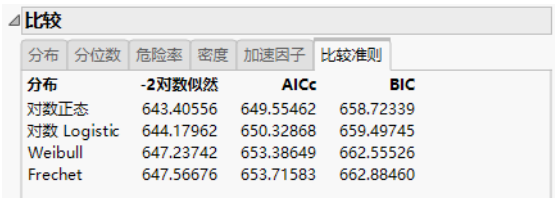
“加速因子刻画器”支持您将加速试验条件与使用条件值比较时，估计该加速试验条件和参数分布假设下的失效时间。通常将时间加速图解释为使用条件的第 *p* 分位数与加速试验条件的第 *p* 分位数的比率。仅当分布为对数正态、Weibull、对数 Logistic 或 Fréchet 且尺度参数对于所有水平保持不变时，该关系才适用。有关分布的参数化的详细信息，请参见“寿命分布”一章中第 72 页的“分布”。

注意：“加速因子刻画器”在以下情况下不显示：解释变量为离散变量；解释变量被视为离散变量；定制公式不使用统一的尺度因子；或是分布为正态、最小极值、Logistic 或最大极值分布。

比较准则

比较准则选项卡显示所关注分布的 -2 对数似然、AICc 和 BIC 准则。图 4.12 显示了 Weibull、对数正态、对数 Logistic 和 Fréchet 分布的这些值。提供更好数据拟合的分布显示在“比较”报表的顶部，这些分布按 AICc 排序。

图 4.12 “比较准则”报表选项卡


比较

分布 | 分位数 | 危险率 | 密度 | 加速因子 | 比较准则

分布	-2对数似然	AICc	BIC
对数正态	643.40556	649.55462	658.72339
对数 Logistic	644.17962	650.32868	659.49745
Weibull	647.23742	653.38649	662.55526
Frechet	647.56676	653.71583	662.88460

该报表表明对数正态和对数 Logistic 分布提供最佳数据拟合，因为这些分布具有最低的准则值。有关准则的详细信息，请参见《拟合线性模型》。

结果

报表窗口的“结果”部分显示了比“比较”报表更为详细的统计学和预测刻画器。为每个选定分布都显示单独的结果部分。图 4.13 显示了 Devalt.jmp 的 Weibull 结果、嵌套模型检验和诊断图的一部分内容。

每个指定分布都包含统计结果、诊断图以及分布、分位数、危险率、密度和加速因子这些刻画器。“定制估计”选项卡支持您使用 “Wald”和“似然区间”方法估计特定的失效概率和分位数。若在平台启动窗口中选择了“Box-Cox 关系”，还会显示“灵敏度”选项卡。该选项卡显示相对似然和 B10 寿命如何作为 Box-Cox lambda 的函数发生变化。

图 4.13 Devalt.jmp 数据的 Weibull 分布嵌套模型检验

统计学

对于每个参数分布，都有一个“统计学”部分显示参数估计值、协方差矩阵、置信区间、汇总统计量和诊断图。您可以通过从每个参数分布的“统计学”红色小三角菜单中选择以上任意或全部选项，从而保存概率、分位数和危险率估计值。估计值以及相应的置信上限和下限会另存为数据表中的列。

嵌套模型检验

若您在平台启动窗口中选定了“嵌套模型检验”选项，则会包括“嵌套模型检验”。“嵌套模型检验”包括以下模型的统计量和诊断图：

不同位置和尺度 假定位置和尺度参数对于解释变量的所有水平都不同。该选项等同于依据解释变量的水平来拟合分布。“不同位置和尺度”模型包含多个位置参数和多个尺度参数（图 4.14）。

不同位置 假定位置参数不同，但尺度参数对于解释变量的所有水平都相同。“不同位置”模型包含多个位置参数，但仅有一个尺度参数（图 4.15）。

回归 显示在初始“以 X 拟合寿命”报表窗口中的默认模型（图 4.16）。

无效应 假定解释变量不影响响应。该选项等同于使用选定分布拟合所有数据值。“无效应”模型有一个位置参数和一个尺度参数（图 4.17）。

默认显示“不同位置和尺度”、“不同位置”和“回归”分析结果。回归参数估计值和位置参数公式默认显示在“估计值”部分下方。通过选中“嵌套模型检验”标题下方的“无效应”左侧的复选框可显示“无效应”模型的诊断图。

要独立于其他模型查看每个模型的结果，请点击所关注的加下划线的模型（列在“嵌套模型检验”下方），然后取消选中其他模型的复选框。

若在启动窗口中未选中“嵌套模型检验”选项，则不评估“不同位置和尺度”和“不同位置”模型。在这种情况下，会为您选择的每个分布的回归模型提供估计值，并且“Cox-Snell 残差 P-P 图”是唯一的诊断图。

注意：为 Weibull 分布拟合“不同位置和尺度”或“不同位置”模型时，Weibull 分布的这两个参数化都会显示在“估计值”表中，这与图 4.14 和图 4.15 中的情况相同。有关 Weibull 参数化的详细信息，请参见“生存分析”一章中第 310 页的“Weibull”。

诊断

图 4.13 中显示的“多重概率图”用来验证加速变量不同水平的分布假设。若每个水平对应的线条未能穿过该水平的数据点，则分布假设可能不成立。并排的比较诊断图可用于直观地比较不同模型的有效性。有关多重概率图的讨论，请参见 Meeker and Escobar (1998, sec. 19.2.2)。每个多重概率图在图例下方都有一个选项，该选项支持您显示或隐藏图中每条线的着色参数置信区间。

“Cox-Snell 残差 P-P 图”用于验证数据的分布假设。若数据点大幅度地偏离对角线，则分布假设可能不成立。“Cox-Snell 残差 P-P 图”红色小三角菜单包含名为**保存残差**的选项，该选项支持您将残差数据保存到数据表中。有关 Cox-Snell 残差的说明，请参见 Meeker and Escobar (1998, sec. 17.6.1)。

“拟合值 - 残差图”用来验证加速变量不同水平的分布假设。这些图在垂直轴上显示标准化残差，在水平轴上显示 X 变量。“拟合值 - 残差”红色小三角菜单包含名为**保存残差**的选项，该选项支持您将标准化残差数据保存到数据表中。

图 4.14 Devalt.jmp 数据 Weibull 分布的 “不同位置和尺度”模型

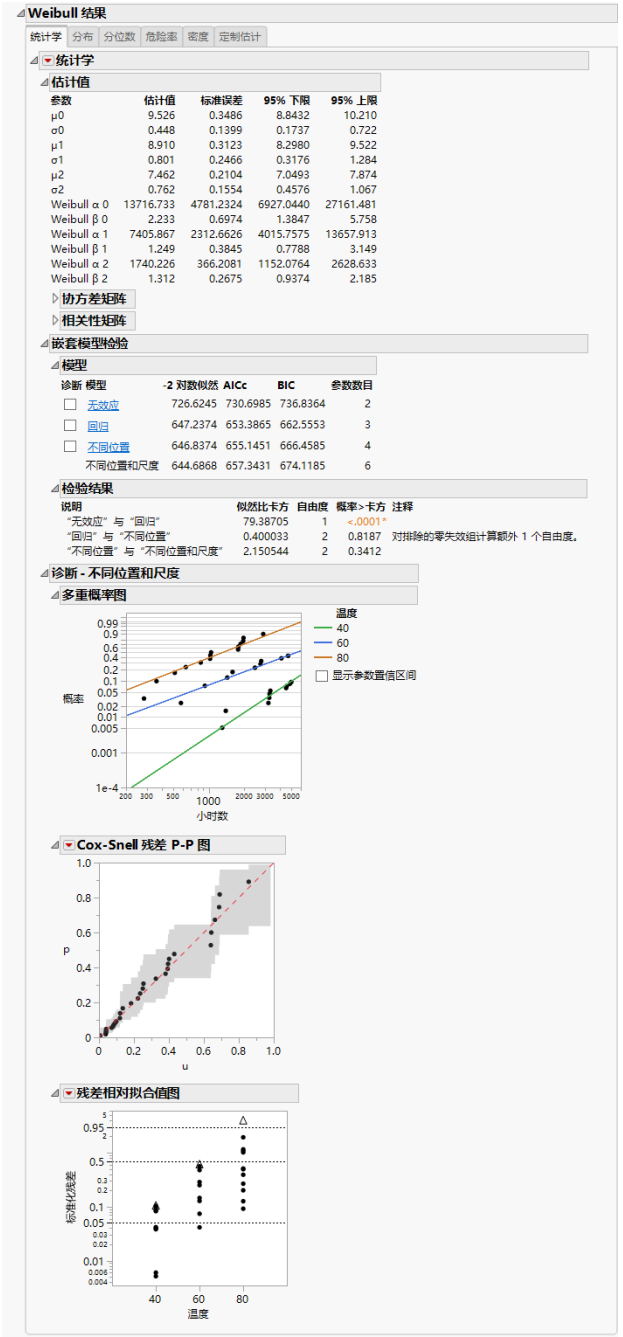


图 4.15 Devalt.jmp 数据 Weibull 分布的 “不同位置”模型

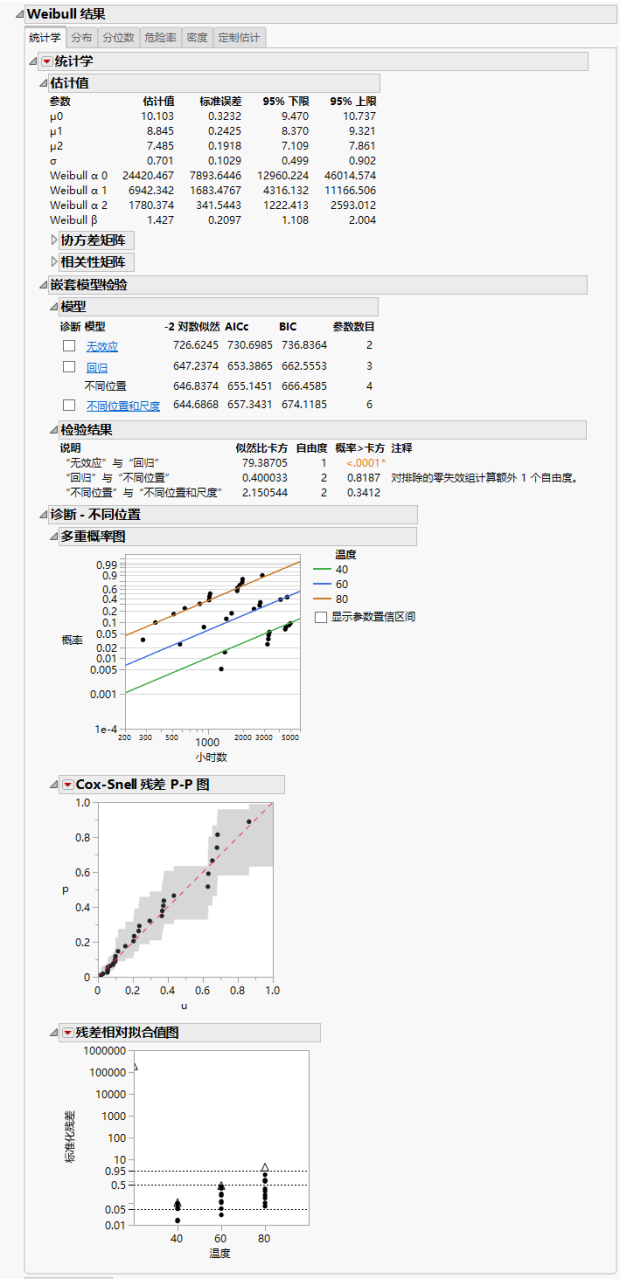


图 4.16 Devalt.jmp 数据 Weibull 分布的 “回归”模型

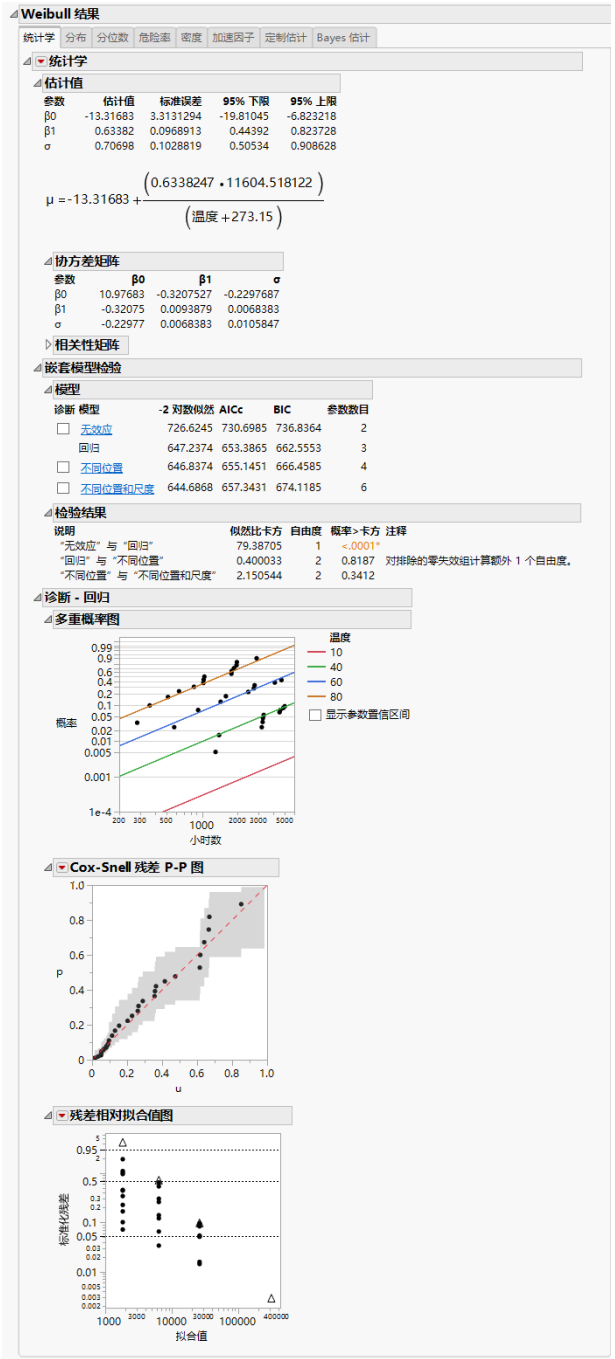
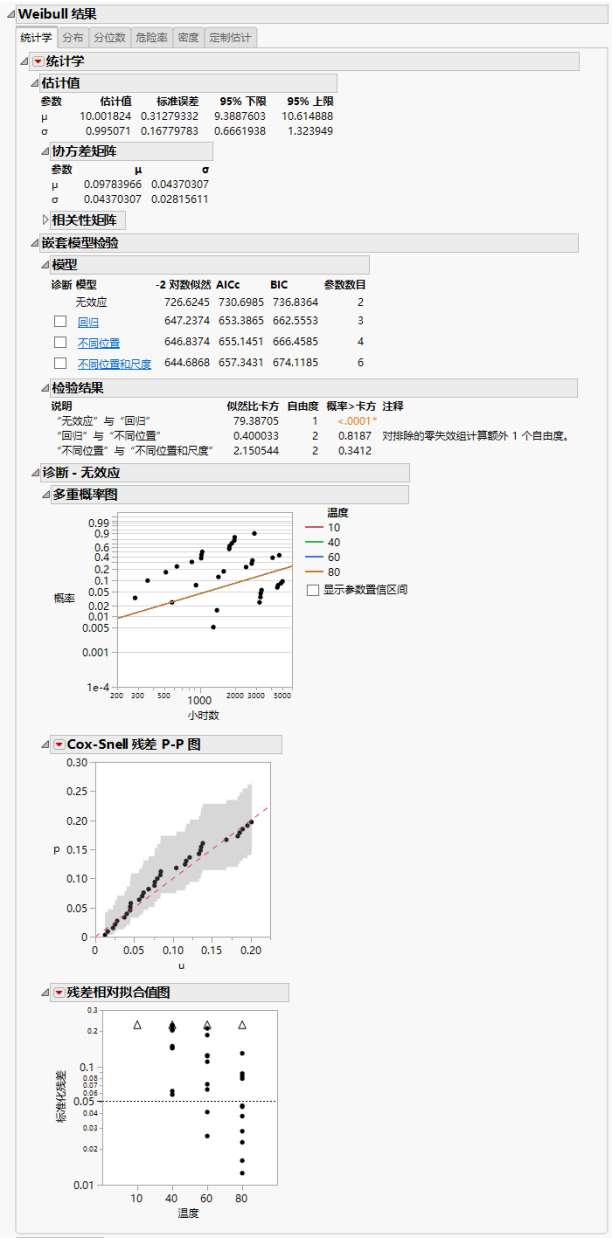


图 4.17 Devalt.jmp 数据 Weibull 分布的 “无效应”模型



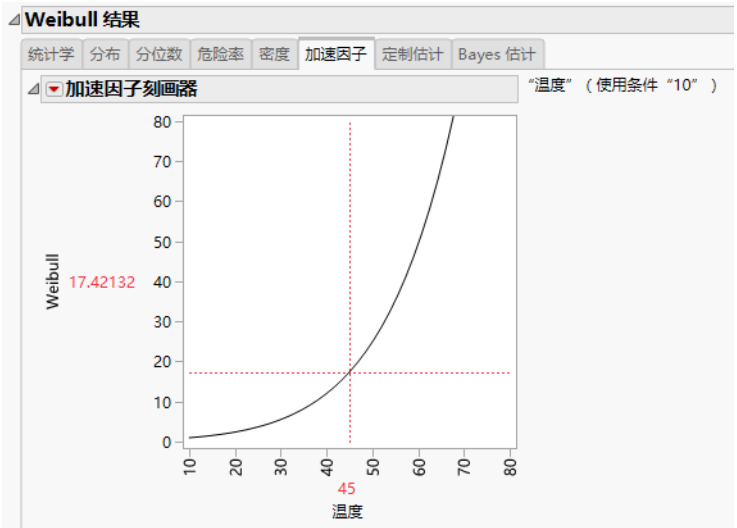
刻画器和曲面图

除统计汇总和诊断图之外，“以 X 拟合寿命”报表窗口还包括您指定的每个分布的刻画器和曲面图。要查看 Weibull 时间加速因子和解释变量刻画器，请点击“Weibull 结果”下方的**分布**

选项卡。要查看曲面图，请点击“Weibull”标题（在刻画器下方）左侧的展开图标。刻画器和曲面图的行为类似于其他平台。请参见《刻画器指南》。

该报表窗口还包括标记为**加速因子**的选项卡。点击**加速因子**选项卡可显示“加速因子刻画器”。该刻画器显示的是该报表窗口的“比较”部分中**加速因子**选项卡下方显示的 Weibull 图的放大版本。图 4.18 显示了 Devalt.jmp 的 Weibull 分布的“加速因子刻画器”。通过从“以 X 拟合寿命”红色小三角菜单中选择**设置时间加速使用条件**选项，可以修改解释变量的使用条件水平。

图 4.18 Devalt.jmp 的 Weibull “加速因子刻画器”



定制估计

对于每个参数分布，都有一个“定制估计”部分，其中包含两个报表：“估计分位数”和“估计概率”。对于具有正值支持的分布，“定制估计”部分还包含“估计剩余寿命均值”(MRLF) 报表。

估计分位数

“估计分位数”报表包含一个计算器，该计算器支持您预测特定失效概率值的分位数。在“估计分位数”计算器中，输入“概率”和 X 变量的值。按 Enter 键可查看分位数估计值和相应的置信区间。要计算多个分位数估计值，请点击加号，输入另一个“概率”值和/或另一个 X 值，然后按 Enter 键。点击减号可删除最后一个条目。若在任一个列中输入不止一个值，该表将包含“概率”和 X 值的所有组合。

默认情况下显示基于 Wald 的区间。点击“区间类型”分级显示项中的**似然置信区间**，以便切换到基于似然的置信区间。这些区间的置信水平由“以 X 拟合寿命”红色小三角菜单中的**更改置信水平**选项确定。

估计概率

“估计概率”报表包含一个计算器，该计算器支持您预测特定时间值的失效和生存概率。在“估计概率”计算器中，输入“时间”和 X 变量的值。按 Enter 键可查看失效概率估计值和相应的置信区间。要计算多个失效概率估计值，请点击加号，输入另一个“时间”值和/或另一个 X 值，然后按 Enter 键。点击减号可删除最后一个条目。若在任一个列中输入不止一个值，该表将包含“时间”和 X 值的所有组合。

默认情况下显示基于 Wald 的区间。点击“区间类型”分级显示项中的**似然置信区间**，以便切换到基于似然的置信区间。这些区间的置信水平由“以 X 拟合寿命”红色小三角菜单中的**更改置信水平**选项确定。

估计剩余寿命均值 (MRLF)

“估计剩余寿命均值”(MRLF) 报表包含一个计算器，该计算器支持您预测特定时间值的剩余寿命均值。在“估计剩余寿命均值”(MRLF) 计算器中，输入“生存时间”和 X 变量的值。按 Enter 键可查看剩余寿命均值的估计值与相应的基于 WAD 的置信区间。这些区间的置信水平由“以 X 拟合寿命”红色小三角菜单中的**更改置信水平**选项确定。要计算多个剩余寿命均值的估计值，请点击加号，输入另一个“生存时间”值和/或另一个 X 值，然后按 Enter 键。点击减号可删除最后一个条目。若在任一个列中输入不止一个值，该表将包含“生存时间”和 X 值的所有组合。

注意：当生存时间等于零时，计算的剩余寿命均值 (MRLF) 的值等于平均失效时间 (MTTF) 的值。

Bayes 估计

对于“指数”之外的每个参数分布，都有一个“Bayes 估计”部分支持您获取 Bayes 参数估计值。若“以 X 拟合寿命”启动窗口中的“关系”为“定制”、“无效应”、“位置”或“位置和尺度”，则“Bayes 估计”部分不可用。

使用拒绝抽样或 Markov Chain Monte Carlo (MCMC) 算法执行“以 X 拟合寿命”平台中的“Bayes 估计”。具体而言，该平台尝试一个基本的拒绝抽样器。若拒绝抽样器生成有效结果，则报告这些结果。若拒绝抽样器不能生成有效的结果，则平台使用随机漫步 Metropolis-Hastings 算法，并在“Bayes 估计”报表的顶部添加注释。请参见 Robert and Casella (2004)。

初始报表是一个控制面板，您可在其中指定参数的先验分布并控制模拟的各个方面。要获取参数的后验估计值，请指定先验分布和模拟选项，然后点击“拟合模型”。

要指定参数的先验分布，您必须指定有关分布的分位数的信息，以及斜率 β_1 和尺度 σ 参数。（对于 Weibull 分布，您指定 Weibull β 而不是 σ。）该分位数由两个值定义：分位数的概率和指定分位数处的 X 变量的值。“概率”默认值为 0.10，但您可以指定对应于所关注分位数的值。指定有关先验分布的范围的信息。对于“正态”和“对数正态”先验分布，采用 99% 上下限的形式指定范围。对于“均匀”和“对数均匀”先验分布，采用上下限的形式指定范围。请参见 Meeker and Escobar (1998)。提供的初始值是与报表的“统计”部分中最大似然估计值一致的估计值。

以下用于模拟的选项显示在先验分布规格表下方：

Monte Carlo 迭代次数 设置老化过程完成后从后验分布中抽取的样本大小。

随机种子 设置模拟的初始状态。默认情况下，它是时钟时间。该数字应为大于 1 的正整数。若指定 1，则使用当前时钟时间。

“Bayes 估计 - 结果 <N>”报表

指定先验分布和模拟选项后，点击**拟合模型**执行模拟。为每个模拟都提供一个“Bayes 估计 - 结果 <N>”报表。该报表包含以下标题：

先验 显示您在“Bayes 估计”报表中输入的用于运行模拟的规格。“先验”报表还包含随机种子。

后验估计 显示五个边缘统计量，这些统计量描述 β_0、β_1、σ 和分位数的后验分布。边缘统计量包括：中位数、0.025 分位数（下限）、0.975 分位数（上限）、均值以及从 Monte Carlo 样本中计算得出的标准差。若指定 Weibull 分布，该表则包含 Weibull β 而不是 σ 的后验估计值。

要基于一般参数的后验估计值计算其他派生变量的统计量，请点击 “导出 Monte Carlo 样本”链接。

后验散点图 显示来自 Monte Carlo 模拟的值的两个散点图。左侧的散点图显示与在 “先验”报表中所指定的后验参数相同的值。右侧的散点图显示与在“后验估计值”报表中所指定的后验参数相同的值。

刻画器 显示基于后验分布中的样本的两个刻画器。在指定的 X 和时间变量值处，使用以下步骤计算刻画器中显示的值：

- 对于后验分布中的每组抽样参数值，计算累积分布函数和分位数函数在指定的 X 和时间变量值处的值。
- 累积分布函数和分位数函数的预测值是计算值的中位数。
- 置信上限和置信下限分别为这些计算值的 0.025 和 0.975 分位数。这些限值的置信水平由“以 X 拟合寿命”红色小三角菜单中的**更改置信水平**选项确定。

分布刻画器 将参数累积分布函数显示为 X 变量和时间的函数。

分位数刻画器 将参数分位数函数显示为 X 变量和指定概率的函数。

“Bayes 估计 - 结果 <N>”选项

“Bayes 估计 - 结果 <N>”红色小三角菜单包含以下选项：

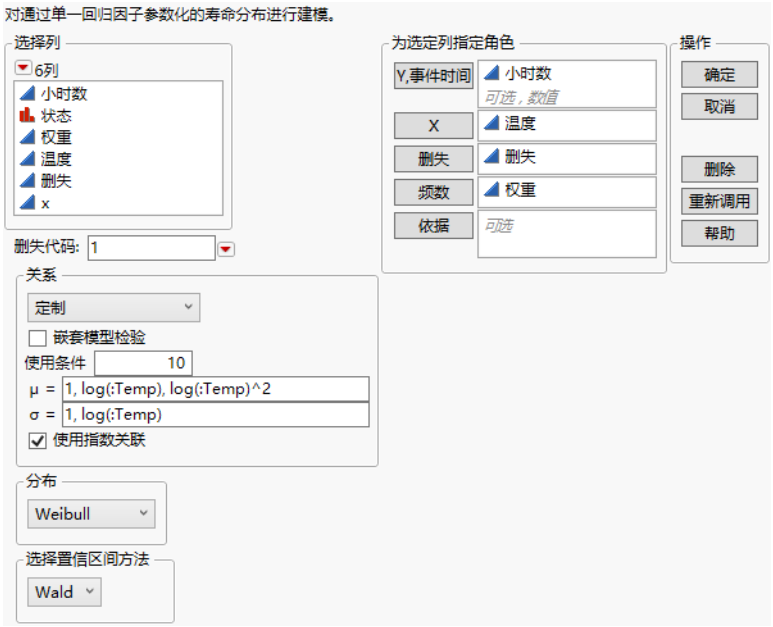
删除 从 “以 X 拟合寿命”报表中删除当前 “Bayes 估计”报表。

导出 Monte Carlo 样本 将 Monte Carlo 模拟的结果保存到新数据表中。您可以使用该表计算后验估计值的统计量。

定制关系

若您想要使用定制变换来对寿命事件与加速因子之间的关系建模，请使用**定制**选项。该选项位于启动窗口中的“关系”下方的列表中。在位置 (μ) 和尺度 (σ) 参数的输入字段中输入逗号分隔值。对于 Devalt.jmp 样本数据，举例来说，可以为 μ 输入 “1, log(:Temp), log(:Temp)^2”，可以为 σ 输入“1, log(:Temp)”，其中 1 指示截距包含在模型中。选中**使用指数关联**复选框可确保 sigma 参数为正。

图 4.19 在 “以 X 拟合寿命”启动窗口中指定**定制**关系



选择**确定**后，将创建位置和尺度变换并包含在“估计值”报表部分的底部。

图 4.20 定制关系的 Weibull 估计值和公式

估计值

参数	估计值	标准误差	下限 95%	上限 95%
β0	-0.603333	9.9924940	-20.18826	18.98160
β1	8.047095	4.8377341	-1.43469	17.52888
β2	-1.412340	0.5989816	-2.58632	-0.23836
λ0	-2.317331	2.0821317	-6.39823	1.76357
λ1	0.474054	0.5015868	-0.50904	1.45715

$$\mu = -0.6033328 + 8.047095 \cdot \text{Log}(\text{温度}) + -1.41234 \cdot \text{Log}(\text{温度})^2$$

$$\sigma = \text{Exp}\left(-2.317331 + 0.4740541 \cdot \text{Log}(\text{温度})\right)$$

有关如何使用定制变换的示例，请参见第 118 页的“定制关系示例”。类似于第 93 页的“‘以 X 拟合寿命”平台的示例”（其中指定了 Arrhenius 摄氏温度关系），继续执行分析。

“以 X 拟合寿命”平台选项

“以 X 拟合寿命”红色小三角菜单包含以下选项：

拟合对数正态 拟合数据的对数正态分布。

拟合 Weibull 拟合数据的 Weibull 分布。

拟合对数 Logistic 拟合数据的对数 Logistic 分布。

拟合 Fréchet 拟合数据的 Fréchet 分布。

拟合指数 拟合数据的指数分布。

拟合最小极值 拟合数据的最小极值 (SEV) 分布

拟合正态 拟合数据的正态分布。

拟合 Logistic 拟合数据的 Logistic 分布。

拟合最大极值 拟合数据的最大极值 (LEV) 分布。

拟合所有分布 拟合数据的所有分布。

设置时间加速使用条件 支持您在弹出窗口中为加速因子的解释变量输入使用条件值。

更改置信水平 支持您在弹出窗口中为图和统计量输入所需的置信水平。默认置信水平为 0.95。

选项卡式报表 支持您指定希望报表窗口如何显示。提供两个选项：“选项卡式总体报表”和“选项卡式单个报表”。默认情况下选中 “选项卡式单个报表”。您可以选择一个选项、两个选项或不选任何选项。

显示曲面图 在报表的单个分布结果部分中显示或隐藏分布的曲面图。曲面图显示在单个分布的 “分布”、“分位数”、“危险率”和 “密度”部分中，默认情况下显示曲面图。

显示点 在 “非参数叠加”图和 “多重概率图”中显示或隐藏数据点。默认情况下在图中显示点。若取消选中该选项，则改为显示阶梯函数。

散点图 显示解释变量与寿命事件的散点图。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“以 X 拟合寿命”平台的更多示例

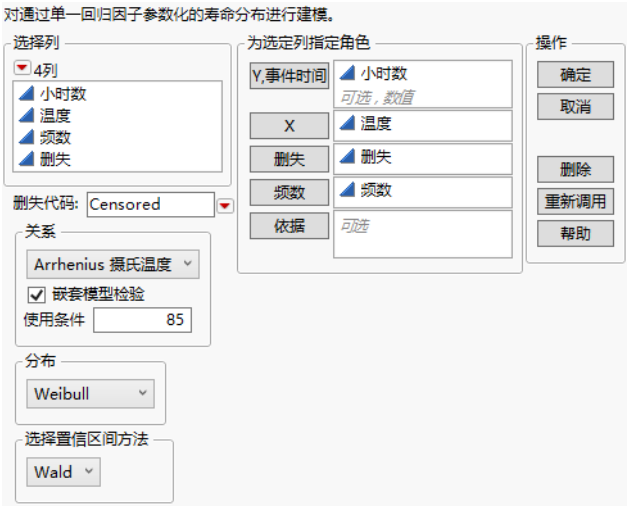
- “电容示例”
- “定制关系示例”

电容示例

本例使用 Capacitor ALT.jmp，该数据表位于样本数据的 Reliability 文件夹中。其中包含用于可靠性研究的模拟数据，这些数据提供了三个温度水平下的删失观测。观测显示为温度水平 85、105 和 125 摄氏度下的删失值。

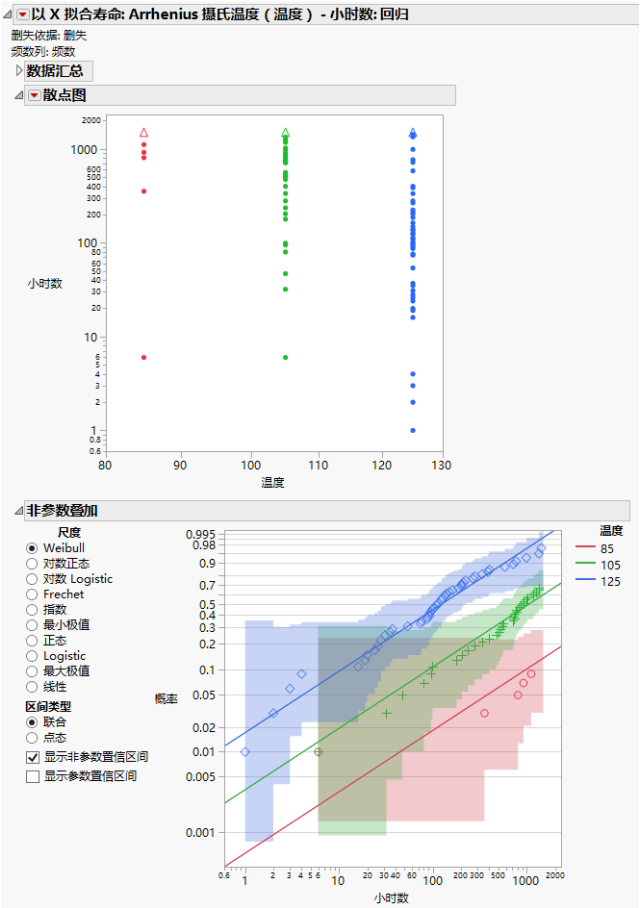
1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Capacitor ALT.jmp。
2. 选择**分析** > **可靠性和生存** > **以 X 拟合寿命**。
3. 选择**小时数**并点击 **Y，事件时间**。
4. 选择**温度**并点击 **X**。
5. 选择**删失**并点击**删失**。
6. 保留**删失代码**为 “Censored”。
7. 选择**频数**作为**频数**。
8. 保留 Arrhenius **摄氏温度**作为关系，并保持选定**嵌套模型检验**选项。
9. 将 Weibull 选作分布。
10. 将 Wald 选作置信区间方法。

图 4.21 “以 X 拟合寿命”启动窗口



11. 点击**确定**。

图 4.22 Capacitor ALT.jmp 数据的 “以 X 拟合寿命”报表窗口



该报表窗口显示汇总数据、诊断图、比较数据和结果，包括详细的统计学和预测刻画器。为每个选定分布都显示单独的结果部分。每个指定分布都包含 “分布刻画器”、“分位数刻画器”、“危险率刻画器”、“密度刻画器”和 “加速因子刻画器”。

定制关系示例

要创建以 Log(温度) 为 Weibull 位置参数的二次模型以及以 Log(温度) 为 Weibull 尺度参数的对数线性模型，请执行以下步骤：

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Devalt.jmp。
2. 选择**分析 > 可靠性和生存 > 以 X 拟合寿命**。
3. 选择**小时数**并点击 **Y，事件时间**。
4. 选择**温度**并点击 **X**。

5. 选择**删失**并点击**删失**。
6. 选择**权重**并点击**频数**。
7. 从列表中选择**定制**作为关系。
8. 在 μ 的输入字段中，输入 “1, log(:Temp), log(:Temp)^2”。
   （1 指示截距包含在模型中。）
9. 在 σ 的输入字段中，输入 “1, log(:Temp)”。
10. 选中**使用指数关联**复选框。
11. 取消选中**嵌套模型检验**对应的复选框。
12. 在 “使用条件”的输入字段中，输入 10。
13. 将 Weibull 选作分布。
    图 4.23 显示了使用**定制**选项的完成的启动窗口。

注意: 没有为非常数尺度模型选中“嵌套模型检验”复选框。该选项不支持嵌套模型检验结果。

14. 点击**确定**。

图 4.23 在 “以 X 拟合寿命”启动窗口中指定定制关系

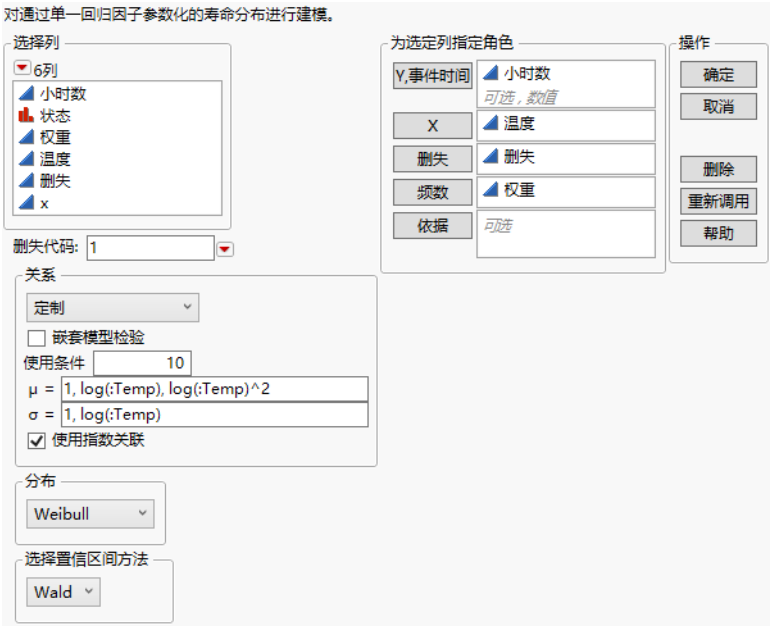


图 4.24 显示了位置和尺度变换，这些变换随后创建并包含在 “估计值”报表部分的底部。

类似于第 93 页的““以 X 拟合寿命”平台的示例”（其中指定了 Arrhenius **摄氏温度**关系），继续执行分析。

图 4.24 定制关系的 Weibull 估计值和公式

估计值

参数	估计值	标准误差	下限 95%	上限 95%
β0	-0.603333	9.9924940	-20.18826	18.98160
β1	8.047095	4.8377341	-1.43469	17.52888
β2	-1.412340	0.5989816	-2.58632	-0.23836
λ0	-2.317331	2.0821317	-6.39823	1.76357
λ1	0.474054	0.5015868	-0.50904	1.45715

$$\mu = -0.6033328 + 8.047095 \bullet \text{Log}\left(\text{温度}\right) + -1.41234 \bullet \text{Log}\left(\text{温度}\right)^2$$

$$\sigma = \text{Exp}\left(-2.317331 + 0.4740541 \bullet \text{Log}\left(\text{温度}\right)\right)$$

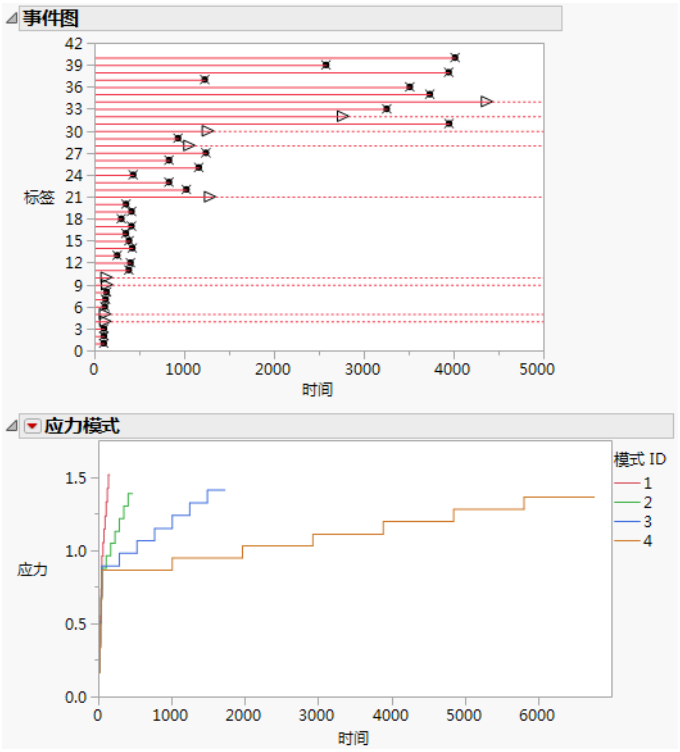
第 5 章

累积损坏

对具有可变应力的产品随时间劣化建模

累积损坏模型（其中包括步进应力模型）支持您分析应力水平可能随时间变化的加速寿命试验。许多不同的力（负载、温度和压力）都可以构成应力。典型的累积损坏实验包含多个检验单元。每个单元都有一个初始应力水平，可以在整个实验过程中更改该应力水平。响应是失效时间或事件时间。该平台绘制失效事件，并支持您用多个分布拟合您的数据。

图 5.1 “累积损坏”报表的示例



目录

“累积损坏”平台概述 123
“累积损坏”平台的示例 123
启动“累积损坏”平台 127
事件时间 128
应力模式 129
“累积损坏”报表 131
事件图 131
“应力模式”报表 131
模型列表 132
模型结果 132
“累积损坏”平台选项 133
应力模式选项 133
“累积损坏”平台的更多示例 134
示例：模拟新数据 134

“累积损坏”平台概述

累积损坏实验（亦称**变动应力实验**）是应力水平可随时间变化的加速寿命试验。许多不同的力（负载、温度和压力）都可以构成应力。典型的累积损坏实验包含多个检验单元。每个单元都有一个初始应力水平，可以在整个实验过程中更改该应力水平。

最常见的累积损坏实验是**步进应力实验**。步进应力实验使用应用了不同应力水平的多个单元。可使用温度、压力或电压之类的因子构成应力。每个单元都有一个初始应力水平。在指定的时间点，基于不同的应力水平模式调整应力水平。在两次应力水平变化期间，应力水平保持不变。

“累积损坏”平台还包括另外三种变动应力模式模型：

- 在序进应力实验中，应力水平在起始值处开始，然后按指定的斜率随时间线性增加。
- 在正弦曲线应力实验中，应力水平以正弦波定义的周期方式波动。
- 在分段序进应力实验中，在指定时间点定义应力水平，这类似于步进应力的情况。不过，不要求应力水平在时间点之间保持不变。应力水平在时间点之间从开始应力水平线性改变为结束应力水平。若有一对开始和结束应力水平相等，则区间等同于步进应力区间。

有关变动应力和步进应力模型的详细信息，请参见 Nelson (2004, ch. 10)。

“累积损坏”平台的示例

在不同的应力条件下对 40 个检验单元进行了步进应力实验。您的目标是在给定应力水平 0.75 的前提下，估计在 10,000 这个时间单位时的失效概率。这些数据表基于 Nelson (2004, ch. 10)。

Reliability/CD Step Stress Pattern.jmp 数据表包含名为**模式 ID** 的列，该列标识四种不同的应力模式。特定步的应力水平是**电压**与**厚度**之比。（请注意，这两列是隐藏的。）每种应力模式的**厚度**都保持不变。不过，**电压**设置为不同水平并在每种模式中增高。

Reliability/CD Step Stress.jmp 数据表包含失效时间数据。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/CD Step Stress.jmp 和 Reliability/CD Step Stress Pattern.jmp。

   CD Step Stress 表包含失效时间数据：

   – **时间**列提供失效时间。
   – **模式 ID** 列标识应力模式。
   – **删失**列指示失效时间是精确时间还是删失时间。

   表中每行都对应一个检验单元。

   CD Step Stress Pattern 表包含四种应力模式（标识为 1 到 4）。应力因子**应力**的水平在**模式 ID** 列的每个值内各有不同。**持续时间**列表示应力因子的特定水平持续了多少个时间单位。
2. 选择**分析** > **可靠性和生存** > **累积损坏**。

启动窗口有两个部分：一个针对失效时间数据（事件时间），另一个针对应力模式数据（应力模式）。

3. 点击 “事件时间”面板中的**选择表**。

   将显示 “事件时间数据表”窗口，该窗口提示您指定失效时间数据所在的数据表。

4. 选择 CD Step Stress，然后点击**确定**。

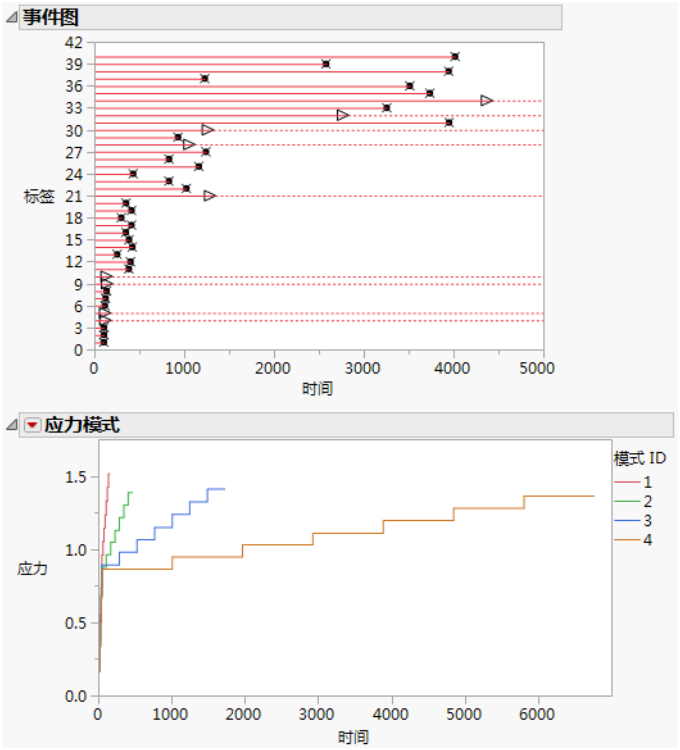
   该表中的列现在填充 “事件时间”面板中的**选择列**列表。

5. 选择**时间**，然后点击**事件时间**。
6. 选择**删失**并点击**删失**。
7. 为**模式 ID** 选择**模式 ID**。
8. 点击 “应力模式”面板中的**选择表**。
9. 选择 CD Step Stress Pattern，然后点击**确定**。

   该表中的列现在填充 “应力模式”面板中的**选择列**列表。

10. 选择**持续时间**，然后点击**应力持续时间**。
11. 选择**应力**，然后点击**应力**。
12. 选择**模式 ID**，然后点击**模式 ID**。
13. 点击**确定**。

图 5.2 事件图和应力模式图



初始报表包含事件图和定义的应力模式图。全部四种应力模式都沿着前 40 个时间单位快速提升应力水平，这之后它们的提升速度显著不同。

14. 点击 “累积损坏”红色小三角并选择**全部拟合**。

图 5.3 “模型列表”报表

模型列表

分布	-2 对数似然	参数数目	AICc	BIC
指数	398.48312	2	402.80745	405.86088
Weibull	398.4822	3	405.14887	409.54884
对数 Logistic	400.93264	3	407.59931	411.99928
对数正态	402.3488	3	409.01547	413.41544
Frechet	410.92577	3	417.59243	421.99241

从 “模型列表”报表中，您可以确定最佳拟合分布是 “指数”分布。

15. 在 “结果”报表中，滚动至 “指数”报表。
16. 在 “分布刻画器”报表中，将**应力**的当前值设置为 0.75。
17. 将**时间**的当前值设置为 10000。

图 5.4 采用指定设置的指数分布的分布刻画器

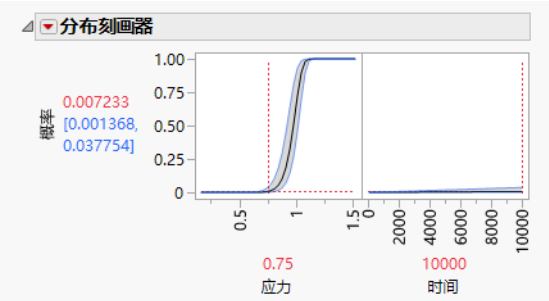


图 5.4 显示在 10000 个时间单位时，处于稳定不变应力 (0.75) 下的某个检验单元的失效预测概率为 0.007233， 95% 置信区间介于 0.001368 到 0.037754 之间。

启动 “累积损坏” 平台

通过选择**分析 > 可靠性和生存 > 累积损坏**来启动 “累积损坏” 平台。您必须为 “累积损坏” 平台提供两个数据表作为输入。第一个数据表提供待检验的每个单元的事件时间数据。第二个数据表定义用于每个单元的应力模式。

图 5.5 “累积损坏” 启动窗口

该启动窗口为每种步进应力数据格式都提供了一个单独的选项卡。有关各种应力模式的详细信息，请参见第 129 页的 “应力模式”。有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

每个步进应力格式选项卡都包含两个面板，用于指定模型变量：

- “事件时间”面板对所有步进应力格式都是相同的。该面板类似于“以 X 拟合寿命”平台启动窗口。
- “应力模式”面板用于描述应力类型。因此，其格式取决于选定的步进应力选项卡。

事件时间

“累积损坏”启动窗口中的 “事件时间”面板包含以下选项：

事件时间 标识事件时间 （如，失效时间）或删失时间。对于区间删失，您的数据表应包含两列，分别提供每个单元的失效时间的下限和上限。输入两个删失列作为“事件时间”。有关删失的详细信息，请参见 “寿命分布”一章中第 35 页的 “事件图”。

删失 标识右删失观测。从 “选择列”列表下方的 “删失代码”菜单中选择用于标识右删失观测的值。仅当输入一个 Y 时才使用 “删失”列。

频数 标识有多个单元时的频数或观测计数。若值为 0 或正整数，则这个值表示具有给定行设置的频数或观测计数。

模式 ID 包含的值用于指定 “应力模式”数据表中用于给定行的应力模式。

删失代码 标识“删失”列中指定右删失观测的值。选定某个“删失”列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中进行选择。您还可以在框中输入不同值。若 “删失”列包含 “值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

关系 标识事件与应力因子之间的关系。表 5.1 定义每种关系的模型。

表 5.1 关系模型选项

关系	模型
Arrhenius 摄氏温度	μ = b0 + b1 * 11604.5181215503 / (X + 273.15)
Arrhenius 华氏温度	μ = b0 + b1 * 11604.5181215503 / ((X + 459.67) / 1.8)
Arrhenius 绝对温度	μ = b0 + b1 * 11604.5181215503 / X
逆幂（默认）	μ = b0 + b1 * log(X)
线性	μ = b0 + b1 * X
对数	μ = b0 + b1 * log(X)
Logit	μ = b0 + b1 * log(X / (1 - X))
倒数	μ = b0 + b1 / X
平方根	μ = b0 + b1 * sqrt(X)

表 5.1 关系模型选项 (续)

关系	模型
Box-Cox	μ = b0 + b1 * BoxCox(X)
定制	用户定义（仅在“步进应力”面板中可用。）

BoxCox(X) 变换定义如下：

$$x_i^{(\lambda)} = \begin{cases} \dfrac{x_i^{\lambda} - 1}{\lambda} & \text{若 } \lambda \neq 0 \\ \ln(x_i) & \text{若 } \lambda = 0 \end{cases}$$

若选择**定制**，则出现更多控件，要求您通过定义“定制”变换对寿命事件和应力因子之间的关系建模。

若想要对模型使用“定制”关系，请参见“以 X 拟合寿命”一章中第 114 页的“定制关系”。

分布 指定初始的失效时间分布。从 Weibull、Lognormal、Loglogistic、Fréchet 或“指数”中进行选择。Weibull 是默认设置。有关分布的详细信息，请参见“寿命分布”一章中第 72 页的 “分布”。

应力模式

使用第二个面板指定实验中使用的应力模式。

步进应力模式

步进应力模式的应力水平可在任意时间点变化。必须以时间升序指定每个应力步的持续时间及关联的应力水平。

“步进应力”选项卡中的 “应力模式”面板包含以下选项：

应力持续时间 包含每个应力步的按时间单位计量的长度的列。

应力 包含应力设置水平的列。

模式 ID 包含应力模式的唯一标识符的列。使用该列可将 “应力模式”数据表中的应力模式与“事件时间”数据表中的观测进行匹配。

模式持续 指定如何处理在定义的应力模式中的最终时间期间后出现的失效。该面板包含以下选项：

终止 在定义的应力模式中的最终时间期间之后某个时间出现的失效将生成错误，导致模型无法拟合。

扩展 对于在定义的应力模式中的最终时间期间之后某个时间出现的失效，其采用与最终时间期间中的水平相同的应力水平。

重复 在定义的应力模式中的最终时间期间之后某个时间出现的失效采用相同的应力水平，就如同重复使用该应力模式一样。例如，若在定义的应力模式中的最终时间期间之后 10 个时间单位出现失效，则该失效时间的应力水平设置为定义的应力模式开始之后 10 个时间单位时的应力水平。

注意：“步进应力模式”的默认 “模式持续”设置为 “终止”。

序进应力模式

序进应力模式将应力定义为时间的线性函数。通过截距 （时间为 0 时的应力水平）和斜率 （每一时间单位应力水平的增长）来定义每个模式。每个模式都在应力模式数据表中的一行内说明。

截距 包含每个模式的截距的列。

斜率 包含每个模式的斜率的列。

模式 ID 包含应力模式的唯一标识符的列。使用该列可将 “应力模式”数据表中的应力模式与 “事件时间”数据表中的观测进行匹配。

正弦曲线应力模式

正弦曲线应力模式将应力定义为一个周期函数。该模式通过水平、振幅、周期和相位来定义。每个模式都在应力模式数据表中的一行内说明。模式定义如下：

$S(t) = \text{水平} + \text{振幅} * \sin(\text{相位} + (2*\pi*t) / \text{周期})$

水平 包含每个模式的水平的列。

振幅 包含每个模式的振幅的列。

周期 包含每个模式的周期的列。

相位 包含每个模式的相位的列。

模式 ID 包含应力模式的唯一标识符的列。使用该列可将 “应力模式”数据表中的应力模式与 “事件时间”数据表中的观测进行匹配。

分段序进应力模式

分段序进应力模式将应力定义为时间的分段线性函数。应力水平的线段在跨越时间上既可断开，也可连续。线段还可以是扁平的，这样即可将步进应力和序进应力合并起来。在应力模式数据表中，通过线段的持续时间和应力设置的开始和结束水平来定义线段。

应力持续时间 包含每个应力步的按时间单位计量的长度的列。

应力斜坡 包含步开始和结束时的应力水平的两列。

模式 ID 包含应力模式的唯一标识符的列。使用该列可将“应力模式”数据表中的应力模式与“事件时间”数据表中的观测进行匹配。

模式持续 使用“模式持续”面板，您可以指定在定义的应力模式中的最终时间期间之后出现的应力水平。该面板包含以下选项：

终止 在定义的应力模式中的最终时间期间之后某个时间出现的失效将生成错误，导致模型无法拟合。

扩展 对于在定义的应力模式中的最终时间期间之后某个时间出现的失效，其采用与最终时间期间中的水平相同的应力水平。

重复 在定义的应力模式中的最终时间期间之后某个时间出现的失效采用相同的应力水平，就如同重复使用该应力模式一样。例如，若在定义的应力模式中的最终时间期间之后 10 个时间单位出现失效，则该失效时间的应力水平设置为定义的应力模式开始之后 10 个时间单位时的应力水平。

注意：“分段序进应力模式”的默认“模式持续”设置为“终止”。

“累积损坏”报表

点击“确定”后，“累积损坏”报表窗口随即出现。默认情况下，“累积损坏”报表包含一个事件图、一个应力模式报表、一个模型列表和模型结果。

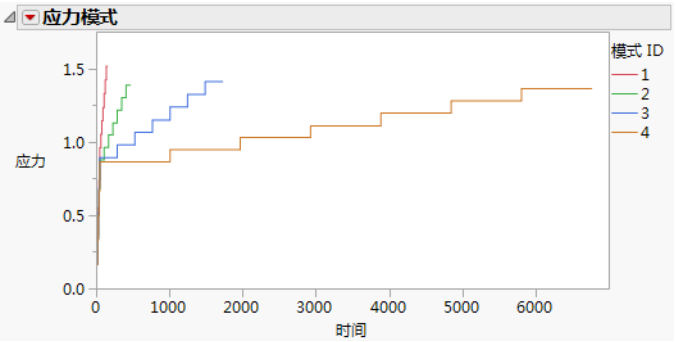
事件图

“累积损坏”中的“事件图”显示失效时间或删失。请参见“寿命分布”一章中第 35 页的“事件图”。

“应力模式”报表

“应力模式”报表为每个应力模式 ID 显示一段时间内的应力水平图。该报表中的“模拟”选项支持您模拟新数据。图 5.6 显示“应力模式”报表图的示例。请参见第 133 页的“应力模式选项”。

图 5.6 “应力模式”报表



模型列表

“模型列表”报表为每个拟合分布都提供 -2 对数似然、参数个数、AICc 和 BIC 统计量。这些统计量（参数个数除外）中的较小值指示拟合效果更好。有关这些统计量的详细信息，请参见《拟合线性模型》。

模型结果

“结果”报表为每个拟合分布提供单独的报表。每个报表都包含以下信息：

参数估计值 显示估计值、标准误差和基于 Wald 的 95% 置信区间。

拟合方程显示在“参数估计值”表的下方。该方程考虑到拟合参数估计值以及在启动窗口中指定的关系。

分布刻画器 将累积失效概率显示为时间的函数。

分位数刻画器 将失效时间显示为累积概率的函数。

危险率刻画器 将危险率显示为时间的函数。

密度刻画器 显示分布的密度函数。

标准化残差的概率图 在概率尺度轴上显示模型的标准化残差图。

“标准化残差的概率图”红色小三角菜单提供一个“保存残差”选项，该选项可将标准化残差保存到失效时间数据表中的三个新列。这三列分别为“左残差”、“右残差”和“残差权重”。

Cox-Snell 残差 P-P 图 显示用于验证数据的分布假设的残差图。

“Cox-Snell 残差 P-P 图”红色小三角菜单提供一个“保存残差”选项，该选项可将 Cox-Snell 残差保存到失效时间数据表中的三个新列。这三列分别为“左残差”、“右残差”和“残差权重”。有关 Cox-Snell 残差的讨论，请参见 Meeker and Escobar (1998, sec. 17.6.1)。

“累积损坏”平台选项

“累积损坏”红色小三角菜单包含以下选项：

全部拟合 拟合在启动窗口中未选定的分布。可用分布列在第 129 页的 “分布” 下方。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

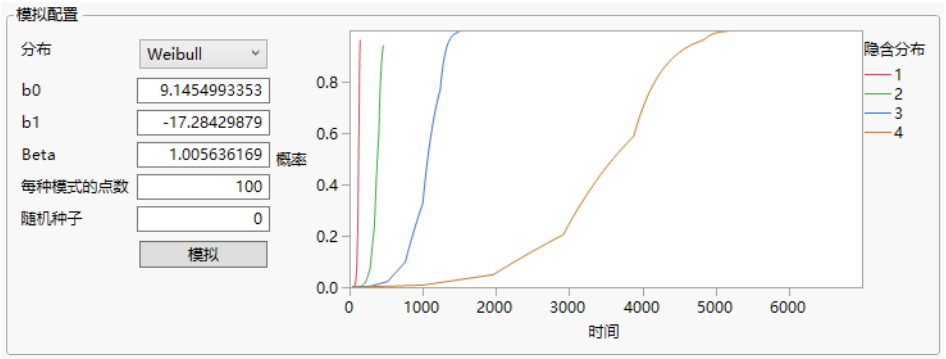
应力模式选项

“应力模式”红色小三角菜单包含“模拟”选项，该选项显示或隐藏“模拟配置”面板。

“模拟”控制面板

图 5.7 显示“模拟配置”面板。“分布”的初始值和参数设置由“累积损坏”启动窗口中指定的分布的参数估计值确定。该图为“应力模式”数据表中定义的应力模式显示一段时间内的估计失效分布函数。

图 5.7 “模拟配置”面板



“模拟配置”面板支持您基于分布和应力模式模拟新失效时间数据。平台启动时使用的 “应力模式”数据表中所定义的应力模式也用于模拟。该面板包含以下选项：

分布 要用于模拟的分布。可用的分布与 “累积损坏”启动窗口中的分布相同。有关分布的详细信息，请参见 “寿命分布”一章中第 72 页的 “分布”。

b0 分布的位置参数的截距。

b1 分布的位置参数的斜率。

lambda （仅可用于 Box-Cox 关系。） Box-Cox 关系的 lambda 值。

b2、 s0、 s1 等 （仅适用于 “定制”关系。 ）其他参数在 “累积损坏”启动窗口的 “定制”关系中定义。

Beta （仅适用于 Weibull 分布。） Weibull 分布的 Beta 参数。

sigma （仅适用于对数正态、对数 Logistic 和 Fréchet 分布。）分布的 sigma 参数。

每种模式的点数 在每种应力模式的模拟中生成的点数。

随机种子 （可选）确保再现模拟结果的非零随机种子。

终止 （当 “累积损坏”启动窗口中的指定 “模式持续”为 “终止”时不可用。 ）一个时间，超过该时间生存检验单元即删失。

模拟

“模拟配置”面板中的这个图为一段时间内的每种应力模式显示隐含分布。点击“模拟”按钮可生成包含模拟结果的新的 JMP 数据表。

“累积损坏”平台的更多示例

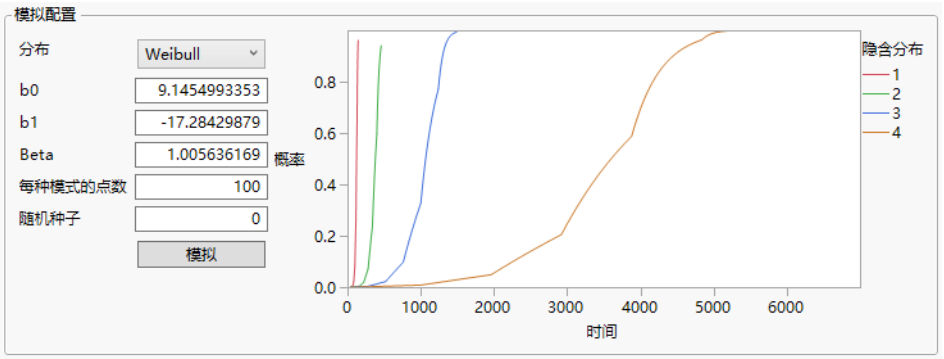
本节包含使用 “累积损坏”平台的更多示例。

示例：模拟新数据

本例演示如何使用 “累积损坏”报表窗口中的 “模拟配置”面板生成新的步进应力数据。本例使用与第 123 页的 “‘累积损坏’平台的示例”中示例相同的数据。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/CD Step Stress.jmp 和 Reliability/CD Step Stress Pattern.jmp。
2. 在 CD Step Stress 数据表中，运行脚本**累积损坏**。
3. 点击 “应力模式”红色小三角并选择**模拟**。

图 5.8 “模拟配置”面板



“模拟配置”面板显示在“应力模式”报表中。选择的“分布”是 Weibull。b0 和 b1 的拟合值用作模拟的初始值。

4. 为**分布**选择“指数”。
5. 为 **b0** 输入 10。
6. 为 **b1** 输入 -18。
7. （可选）为**随机种子**输入 14678。
8. 点击**模拟**。

图 5.9 模拟的部分结果

	左 “时间”	右 “时间”	模式 ID
1	140.67948106	140.67948106	1
2	139.34770497	139.34770497	1
3	136.95430716	136.95430716	1
4	136.30050091	136.30050091	1
5	145	•	1
6	118.5123739	118.5123739	1
7	119.03562567	119.03562567	1
8	145	•	1
9	127.19420127	127.19420127	1
10	118.48391055	118.48391055	1
11	103.54004804	103.54004804	1
12	145	•	1
13	121.04292162	121.04292162	1
14	136.15486483	136.15486483	1
15	143.65666248	143.65666248	1
16	123.04501594	123.04501594	1
17	145	•	1
18	124.26692664	124.26692664	1
19	120.96407573	120.96407573	1

图 5.9 显示了模拟数据表的部分列表。**模式 ID** 为 1 的应力模式仅定义了至多 145 个时间单位。由于启动窗口中的“模式配置”设置设置为“终止”，针对应力模式 1，模拟将删失 145 处的所有模拟值。

第 6 章

复发分析

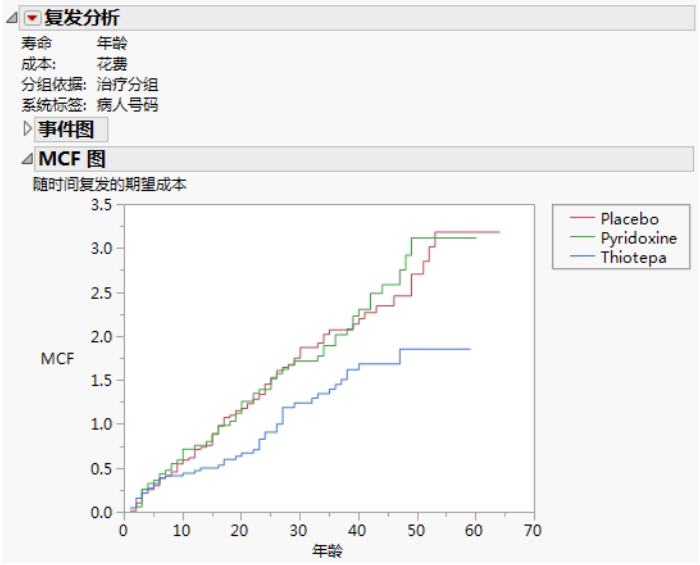
对随时间复发事件的频数或成本建模

“复发分析”平台分析事件时间，这些事件对于每个单元、项或人可多次出现。在工业设置中，在某个单元发生故障，接受维修，维修后重新投入使用时就会发生这些事件。这些单元会被跟踪，直到其停止使用。

在医疗设置中，复发分析可用于分析长期疾病持续治疗的数据，比如接受治疗的膀胱癌患者肿瘤复发的数据。

分析目标是要获取平均累积函数 (MCF)，它显示作为时间函数的每单元总成本。成本可以是维修次数，或者是实际的维修成本。

图 6.1 复发分析示例



目录

“复发分析”平台概述 139
“复发分析”平台的示例 139
启动“复发分析”平台 142
“复发分析”平台选项 143
拟合模型 144
“复发分析”平台的更多示例 147
膀胱癌复发示例 147
Diesel Ship Engines 示例 150

“复发分析”平台概述

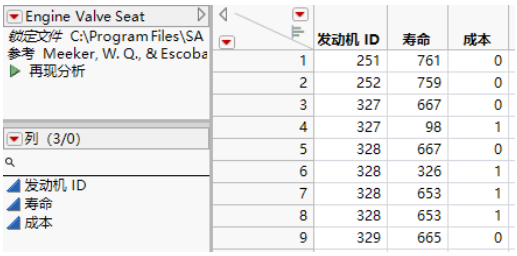
复发事件数据涉及随单元老化而进行的维修的累积频数或成本。在 JMP 中，“复发分析”平台分析复发事件数据。

复发分析数据中每一行对应一个观测的事件，末行包含最后观测到的单元龄。可以包含任意数目的单元或系统。此外，这些单元或系统还可包含任意数目的复发。

“复发分析”平台的示例

典型单元可以是一个系统，如发动机或装置的组件。例如，考虑样本数据表 Engine Valve Seat.jmp，它记录了机车发动机中的气门座更换情况。请参见 Meeker and Escobar (1998, p. 395) 和 Nelson (2003)。图 6.2 列出了该数据的部分内容。**发动机 ID** 列标识特定的机车单元。**寿命**是从发动机气门座投入使用直到被更换之间的天数。请注意，一个发动机可对应多个行，这些行包含对应于多次维修的发动机各次更换时的寿命及其成本。在此，**成本**=0 指示最后观测到的机车寿命。

图 6.2 “Engine Valve Seat”数据表的部分视图

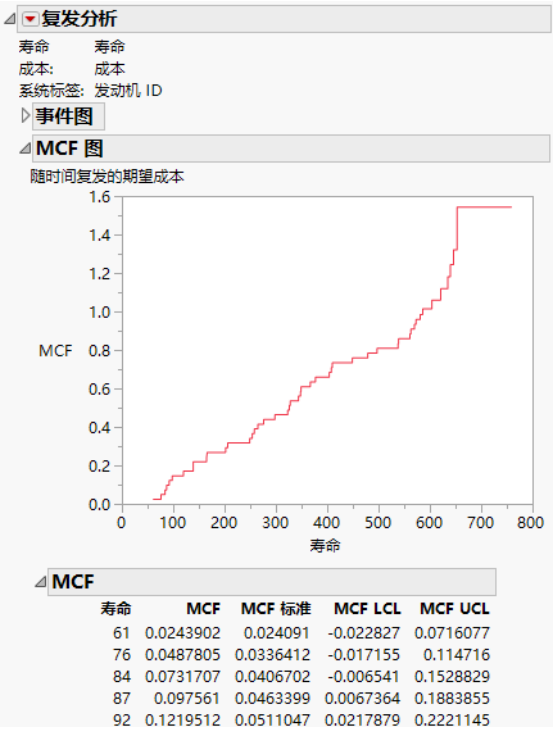


	发动机 ID	寿命	成本
1	251	761	0
2	252	759	0
3	327	667	0
4	327	98	1
5	328	667	0
6	328	326	1
7	328	653	1
8	328	653	1
9	329	665	0

按照图 6.5 所示完成启动窗口。

当您点击**确定**后，“复发”平台将显示图 6.3 和图 6.4 中的报表。MCF 图显示样本平均累积函数。对于每个寿命，它是平均累积成本的非参数估计值或每单元事件数。该函数随单元老化和总成本增加而增长。图 6.3 中的图显示平均一次维修事件大约能维持单元工作 580 天。

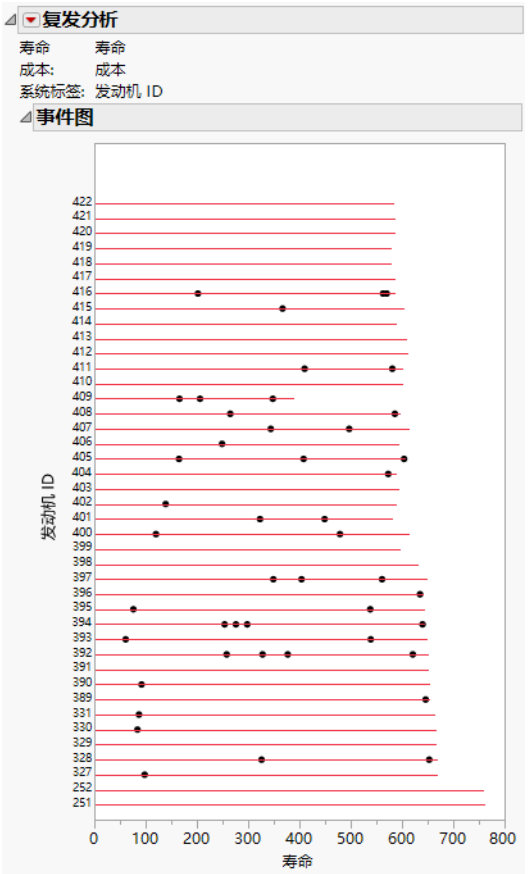
图 6.3 复发分析的 MCF 图和部分表内容



寿命	MCF	MCF 标准	MCF LCL	MCF UCL
61	0.0243902	0.024091	-0.022827	0.0716077
76	0.0487805	0.0336412	-0.017155	0.114716
84	0.0731707	0.0406702	-0.006541	0.1528829
87	0.097561	0.0463399	0.0067364	0.1883855
92	0.1219512	0.0511047	0.0217879	0.2221145

图 6.4 中的事件图显示了每个单元的时间线。每次维修时都有相应的标记，每条线都延伸到该单元最后观测到的寿命。例如，最后观测到单元 409 是在 389 天时，此时已经更换三次气门。

图 6.4 气门座更换的事件图



启动“复发分析”平台

通过选择**分析** > **可靠性和生存** > **复发分析**来启动 “复发分析”平台。

图 6.5 “复发分析”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y, 寿命 , 事件时间戳 指定事件发生时的单元寿命或事件时间戳。若 Y 列为事件时间戳，则您必须指定开始和结束时间戳，以便 JMP 计算寿命。

标签，系统 ID 标识每个事件和删失寿命的单元。

成本 标识必须包含以下值之一的列：

- 1，指示已出现某个事件（单元失效或经过维修、更换或调整）。指定指示符 (1) 后，MCF 即成为作为寿命的函数的每单元平均累积事件数。
- 事件成本（维修、更换或调整的成本）。指定成本后，MCF 即成为作为寿命的函数的每单元平均累积成本，而且“事件图”中的标记大小依据成本值来调整。
- 0，指示单元停止使用，或者不再是研究对象。所有单元（每个系统 ID）在该列中都必须有一个包含 0 的行，并且 **Y,寿命 , 事件时间戳**列包含最后的观测寿命。若每个单元在该表中并不是只有一个最后观测寿命（其中“成本”列单元格为 0），则会显示错误消息。

注意：复发分析的成本指示符正好与**寿命分布**或**生存分析**中显示的删失指示符相反。对于成本变量，值为 1 指示某个事件（如维修）；值为 0 指示该单元已停止使用。对于删失变量，值为 1 指示删失值，值为 0 指示该单元发生的事件或失效 （非删失值）。

分组 为该列标识的不同组生成单独的 MCF 估计值。

原因 指定多个失效模式。

开始时间戳 指定包含原始时间戳的列。若您将开始时间作为事件记录，请改为选择**第一个事件是开始时间戳**选项。 JMP 会减去该列中的值来计算寿命。

结束时间戳 指定包含服务结束时间戳的列。若为所有单元都指定了结束时间，则在此指定该列。若没有为所有单元都指定结束时间，则改为指定**默认结束时间戳**选项。但若您有某条记录中的 “成本”等于 0， JMP 会将该记录用作结束时间戳，您无需指定该角色。

寿命尺度 指定用于建模的时间单位。例如，若您的时间戳以秒编码，您可以将其改为小时。

“复发分析”平台选项

“复发分析”红色小三角菜单包含以下选项：

MCF 图 显示或隐藏平均累积函数 (MCF) 图。

MCF 置信限 显示或隐藏与平均累积函数 (MCF) 的近似 95% 置信限对应的线条。

事件图 显示或隐藏事件图。若在启动窗口中指定了 “成本”列，则 “事件图”中的标记大小将依据 “成本”列中的值来调整。

日历事件图 （仅当事件由时间戳而不是寿命指定时才可用。）显示或隐藏 “日历事件图”，该图显示的事件的日历日期位于水平轴上。该图显示在 “事件图”旁边，每个图中的单元都垂直对齐。

按寿命标绘到达间隔图 显示或隐藏 “寿命 - 到达间隔”图，该图在垂直轴上标绘连续事件之间的时间，在水平轴上标绘事件时间。您可以使用该图确定数据中各事件之间的时间是否有变化。对于复发分析，到达间隔时间应该独立且随时间的分布完全相同。有关到达间隔图的详细信息，请参见 Tobias and Trindade (2012, p. 420)。

标绘 MCF 差值图 （仅当您指定分组变量时才可用。）显示或隐藏 MCF 差值图，包括该差值的 95% 置信区间。当置信区间线没有与零线相交时， MCF 存在显著差异。仅当您指定分组变量时，该选项才可用。

MCF 图 （每个组） （仅当您指定分组变量时才可用。）显示或隐藏一个报表，该报表包含分组变量每个水平的平均累积函数 (MCF) 图。

若还将**标签** , **系统 ID** 变量指定为**分组**变量，该选项可用于获取每个单元的 MCF 图。

拟合模型 支持您拟合复发强度和累积函数的模型。请参见第 144 页的 “拟合模型”。

拟合模型

“拟合模型”选项用于拟合复发强度和累积函数的模型。有四种模型可用于描述强度和累积函数。您拟合的模型可以具有常数参数，也可以具有作为效应的函数的参数。

从“复发分析”平台红色小三角菜单中选择“拟合模型”，生成图 6.6 所示的“复发模型规格”窗口。

图 6.6 复发模型规格

您可以选择具有以下强度和累积函数的四种模型之一：

幂非齐次 Poisson 过程

$$I(t) = \left(\frac{\beta}{\theta}\right)\left(\frac{t}{\theta}\right)^{\beta-1}$$

$$C(t) = \left(\frac{t}{\theta}\right)^{\beta}$$

比例强度 Poisson 过程

$$I(t) = \delta t^{\delta-1} e^{\gamma}$$

$$C(t) = t^{\delta} e^{\gamma}$$

对数线性非齐次 Poisson 过程

$$I(t) = e^{\gamma + \delta t}$$

$$C(t) = \frac{I(t) - I(0)}{\delta} = \frac{e^{\gamma + \delta t} - e^{\gamma}}{\delta}$$

齐次 Poisson 过程

$$I(t) = e^{\gamma}$$

$$C(t) = te^{\gamma}$$

其中， t 是产品的寿命。

表 6.1 将每个模型参数定义为尺度参数或形状参数。

表 6.1 尺度和形状参数

模型	尺度参数	形状参数
幂 NHPP	θ	β
比例强度 PP	γ	δ
对数线性 NHPP	γ	δ
齐次 PP	γ	无

请注意以下事项：

- 对于“复发模型规格”窗口（图 6.6），若您包括尺度效应或形状效应，则表 6.1 中的尺度和形状参数将建模为效应函数。要拟合具有常数尺度和形状参数的模型，请不要包含任何尺度效应或形状效应。
- 与其他模型相比，“齐次 Poisson 过程”是一个特例。对于单项模型而言，“幂 NHPP”和“比例强度 Poisson 过程”是等价的；但对于复杂模型而言，比例强度模型看起来具有更可靠的拟合。

点击**运行模型**可拟合模型并查看模型报表。

图 6.7 模型报表

“拟合的复发模型”红色小三角菜单包含以下选项：

刻画器 显示或隐藏用来显示强度和累积函数的刻画器。

边缘效应 为分类效应的每个水平计算参数函数，而将其他效应保留为中性值。这有助于您查看参数函数在不同分组之间有哪些差异。仅当您指定分类效应时，该选项才可用。

检验齐性 检验过程是否具有齐性。该选项不可用于 “齐次 Poisson 过程”模型。

效应似然比检验 为模型中的每个效应生成一个检验。该选项仅在模型中包含效应时才可用。

特定的强度和累积 计算与特定时间和效应值关联的强度和累积值。置信区间是边侧似然区间。

特定的累积时间 计算与特定复发次数和效应值关联的时间。

保存强度公式 将强度公式保存至数据表。

保存累积公式 将累积公式保存至数据表。

发布强度公式 创建 “强度”公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见《预测和专业建模》。

发布累积公式 创建 “累积”公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见《预测和专业建模》。

从模型模拟 支持您模拟来自估计的复发模型的新数据。该选项创建模拟观测 （基于在 “从模型模拟”窗口中指定的选项）的新数据表。请参见第 147 页的 “从模型模拟”。

删除拟合 删除模型报表。

从模型模拟

从“拟合复发模型”红色小三角菜单中选择“从模型模拟”选项后，随即显示“从模型模拟”窗口。该窗口包含以下模拟规格：

最大事件数 指定要在系统 ID 的每个水平中模拟的最大事件数。

最大寿命 指定模拟事件的最大时间。

单元数 指定系统 ID 的每个水平中的单元数。

注意：若模型包含“截距”项和“常数”项之外的项，模拟数据表将包含回归项的所有水平组合的这一单元数。若回归项是连续的，它将划分到 5 个水平中。

点击“确定”后，随即显示包含结果的新数据表。该表包含一个脚本，该脚本支持您在“复发”平台中分析模拟观测。

“复发分析”平台的更多示例

- “膀胱癌复发示例”
- “Diesel Ship Engines 示例”

膀胱癌复发示例

样本数据文件 Bladder Cancer.jmp 包含来自“退伍军人管理局合作泌尿外科研究组”(Veteran's Administration Co-operative Urological Research Group) 的癌变膀胱肿瘤复发数据。请参见 Andrews and Herzberg (1985, table 45)。所有患者都患有浅表性膀胱肿瘤，在加入试验后即摘除了肿瘤。每位患者随后被分配到三个治疗组中的一个：安慰剂药丸、吡哆醇（维生素 B6）药丸或用三胺硫磷进行周期性化疗。下面的肿瘤复发分析探索了疾病的进展，以及三种治疗之间是否存在差异。

使用图 6.8 所示的选项启动该平台。

图 6.8 “Bladder Cancer”启动窗口

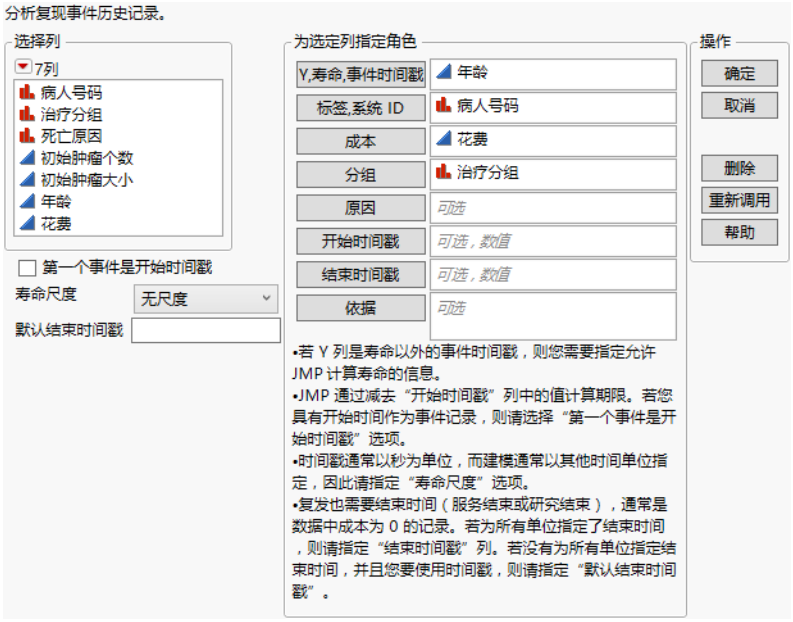
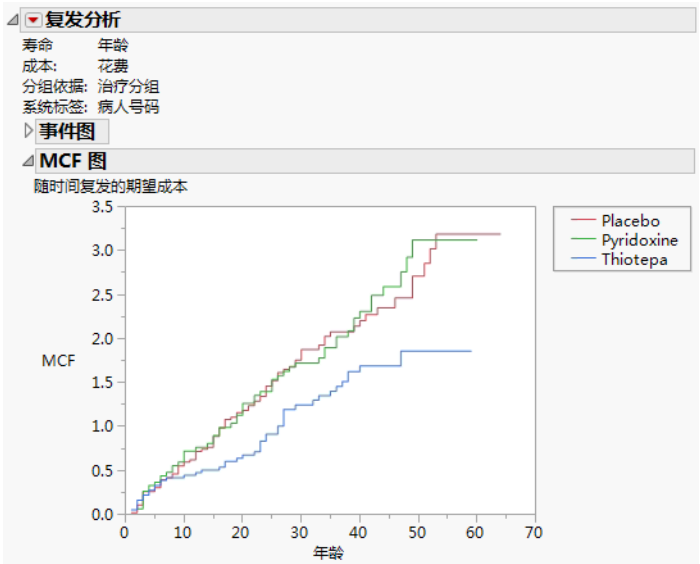


图 6.9 显示了三种治疗的 MCF 图。

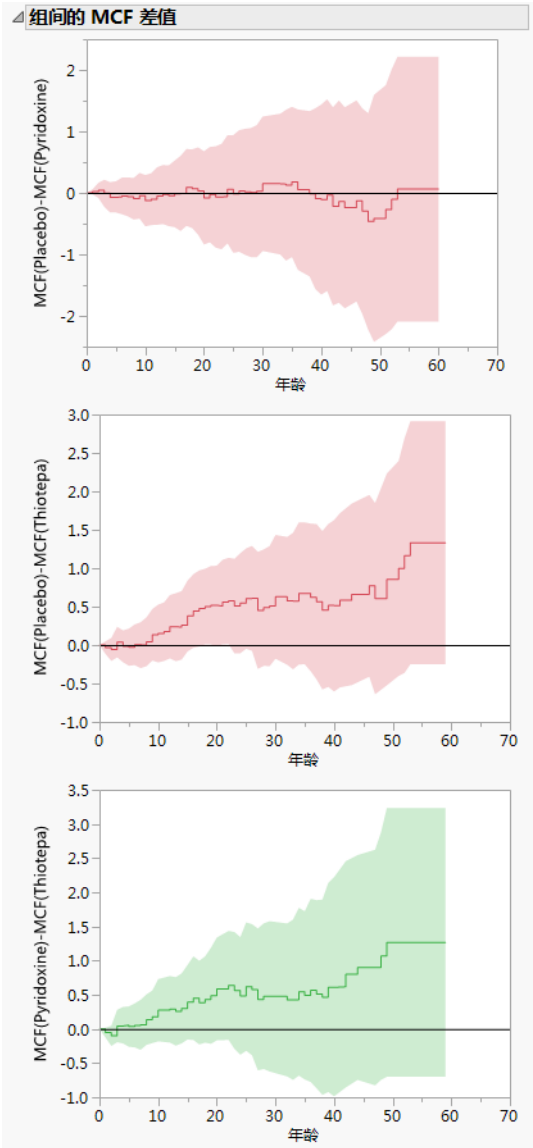
图 6.9 “Bladder Cancer”的 MCF 图



请注意，全部三条 MCF 曲线本质上都是直线。因此，斜率（再现的比率）不随时间变化，这意味着患者看起来不随着疾病进展而变好或变坏。

若要检查治疗之间是否存在差异，请点击“复发分析”红色小三角并选择**标绘 MCF 差值图**。

图 6.10 MCF 差值



要确定治疗之间是否在统计上存在显著差异，请检查差值图上的置信限。若限值不包括 0，则治疗明确地存在差异。图 6.10 中的图形显示治疗之间不存在显著差异。

Diesel Ship Engines 示例

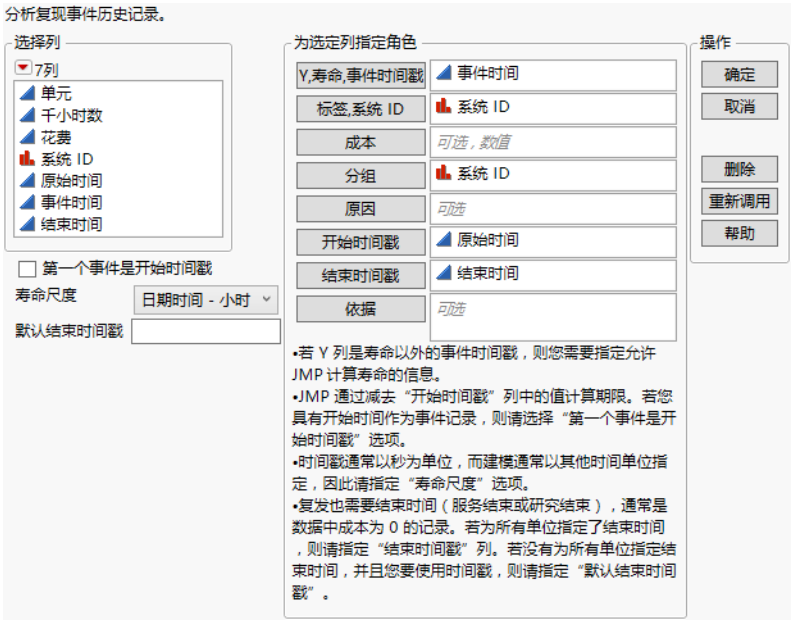
样本数据表 Diesel Ship Engines.jmp 包含关于延长服务期的两艘船（Grampus4 和 Halfbeak4）的发动机维修时间的数据。请参见 Meeker and Escobar (1998)。您想要检查维修进度，了解将来经过多久就需要一次维修。这些观测可以帮助您决定应该何时让发动机停止服务。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Diesel Ship Engines.jmp。
2. 确保行 57 和 129 设置为 “已排除”。

> **注意：**若未设置为 “已排除”，请选择行 57 和 129，再选择**行 > 排除 / 撤销排除**。

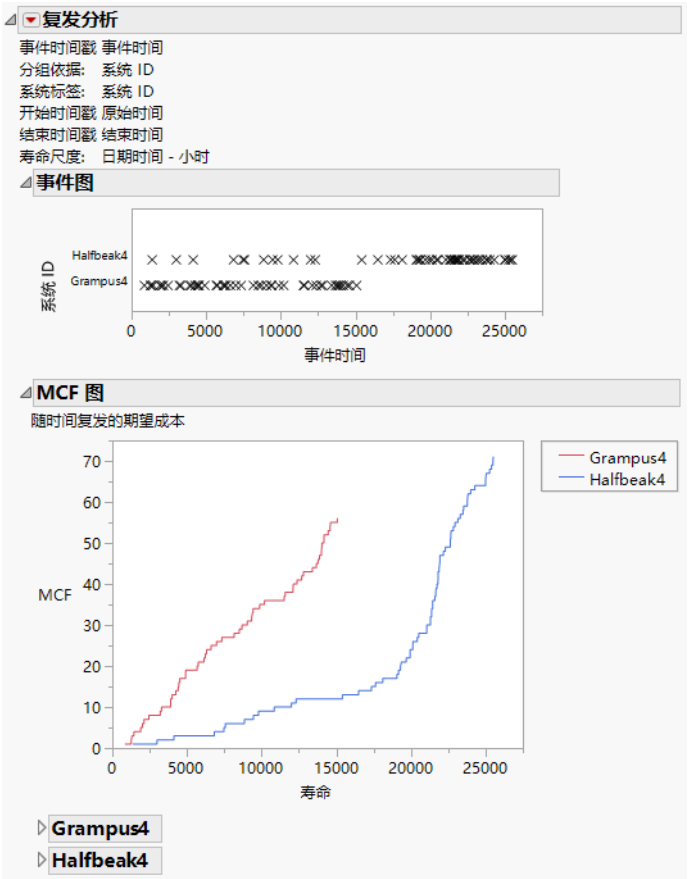
3. 选择**分析 > 可靠性和生存 > 复发分析**。
4. 按照图 6.11 所示完成启动窗口。

图 6.11 “Diesel Ship Engines”启动窗口



5. 点击**确定**。

图 6.12 “Diesel Ship Engines”报表

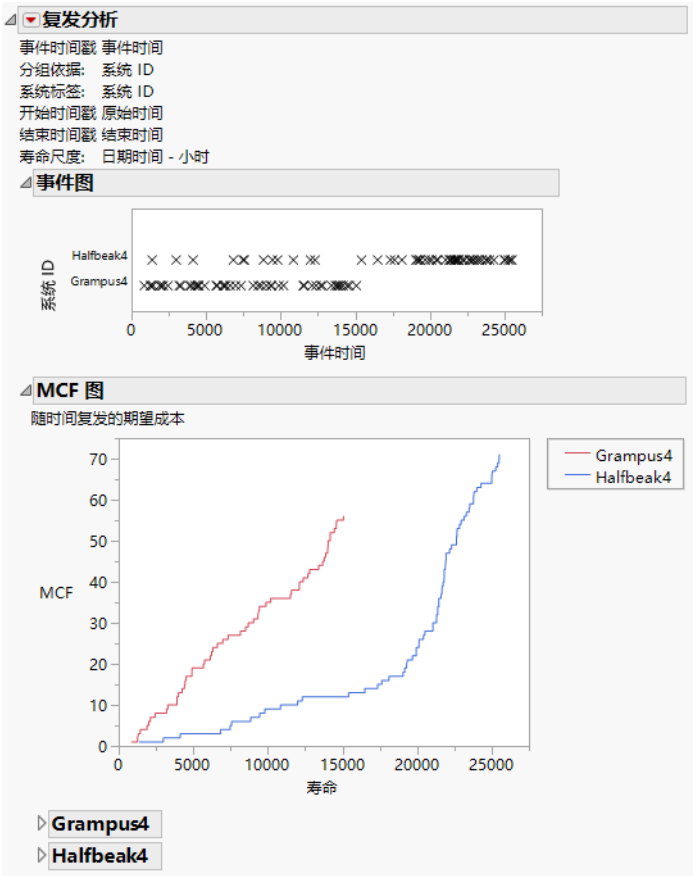


通过查看事件图，您可以看到对 Grampus4 发动机的维修次数相对较一致。对 Halfbeak4 发动机的维修只是偶尔发生，但看起来在 19,000 小时标记处维修次数突然增加。这种增加在 MCF 图中甚至更为明显。

继续您的分析，通过拟合参数模型来帮助预测将来的性能。

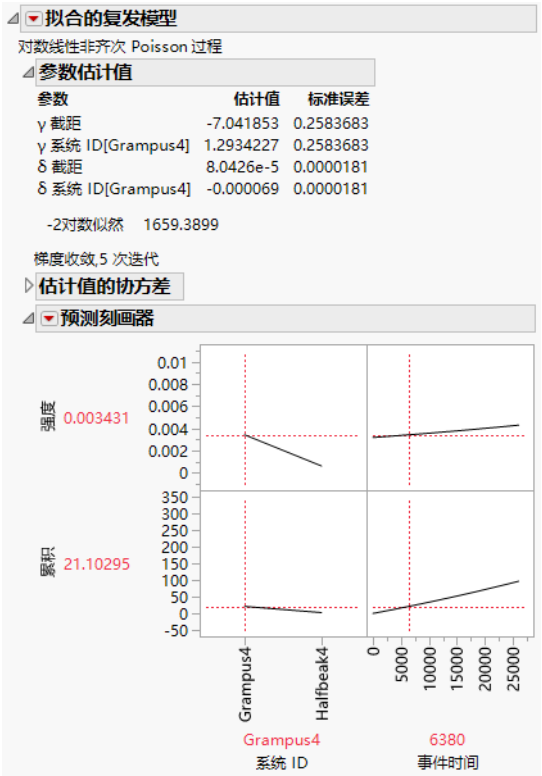
6. 点击“复发分析”红色小三角并选择**拟合模型**。
7. 在“复发模型规格”中，选择**对数线性非齐次 Poisson 过程**。
8. 添加**系统 ID** 列，同时作为“尺度效应”和“形状效应”。
9. 点击**运行模型**。

图 6.13 “Diesel Ship Engines”拟合模型



10. 点击 “拟合的复发模型”红色小三角并选择**刻画器**。

图 6.14 “Diesel Ship”刻画器



比较 Grampus4 与 Halfbeak4 发动机的将来维修次数。更改事件时间值以查看对将来累积维修次数的影响。

- 要查看在使用 30,000 个小时后需要多少次维修，请为事件时间输入 30,000。Grampus4 发动机将需要大约 114 次维修。要查看 Halfbeak4 的值，请点击虚线，并将其从 Grampus4 拖至 Halfbeak4。Halfbeak4 发动机将需要大约 140 次维修。
- 要查看在使用 80,000 个小时后需要多少次维修，请为事件时间输入 80,000。Halfbeak4 发动机将需要大约 248,169 次维修。点击虚线，并将其从 Halfbeak4 拖至 Grampus4。Grampus4 发动机将需要大约 418 次维修。

您可以得出结论：Halfbeak4 发动机将来需要的维修次数明显多于 Grampus4 发动机。

第 7 章

退化

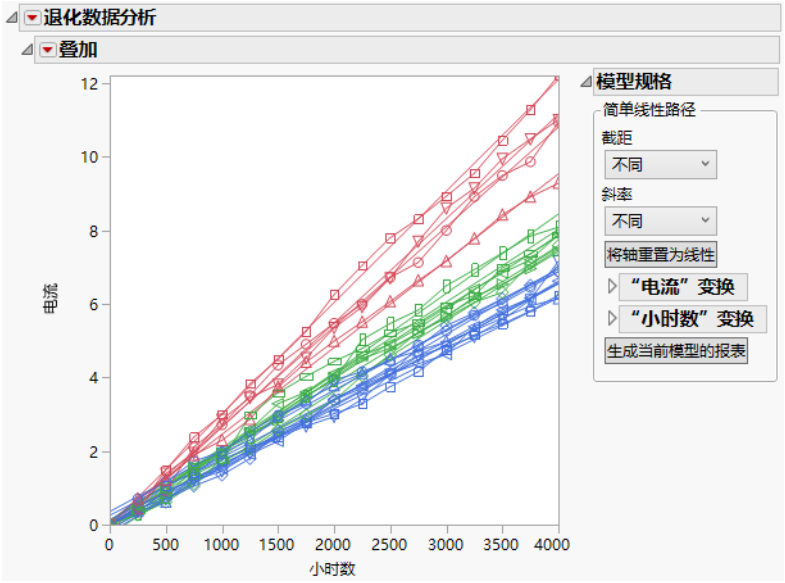
对产品随时间劣化建模

使用“退化”平台，您可以分析退化数据以预测伪失效时间。随后可在其他可靠性平台中分析这些伪失效时间，以便估计失效分布。

可以对线性和非线性退化路径建模。您可以指定加速因子来分析加速退化数据。

您还可以执行稳定性分析，这在设置医药产品失效日期时很有用。

图 7.1 退化分析示例



目录

“退化”平台概述 157
“退化”平台的示例 157
启动“退化”平台 159
退化报表 160
模型规格 162
    简单线性路径 163
    非线性路径 164
逆预测 172
预测图形 173
“退化”平台选项 174
模型报表 176
    模型列表 176
    报表 176
破坏性退化 177
稳定性分析 182

“退化”平台概述

在可靠性分析中，主要目标是对所研究产品的失效时间建模。在许多情况下，由于产品随时间退化（弱化）就会出现这些失效。但有时并不出现失效。在这些情况下，对产品随时间退化建模有助于预测失效时间。当数据中包括加速因子时，您可以拟合加速退化模型。

“退化”平台可对遵从线性或非线性退化路径的数据建模。若路径是非线性的，则可以使用变换使路径线性化。若不能线性化路径，您可以指定非线性模型。

您还可以使用“退化”平台执行稳定性分析。可拟合三种线性模型，并估计失效日期。稳定性分析用于设置医药产品的失效日期。

“退化”平台的示例

本例使用 Meeker and Escobar (1998) 中的 GaAs Laser.jmp 数据表，其中包含几台砷化镓激光器的工作电流增量百分比的测量值。当增量百分比达到 10% 时，激光器即被视为已失效。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/GaAs Laser.jmp。
2. 选择**分析** > **可靠性和生存** > **退化**。
3. 选择**电流**并点击 **Y，响应**。
4. 选择**小时数**并点击**时间**。
5. 选择**单元**并点击**标签，系统 ID**。
6. 在**上规格限**文本框中键入 10。
7. 点击**确定**。

图 7.2 初始退化报表

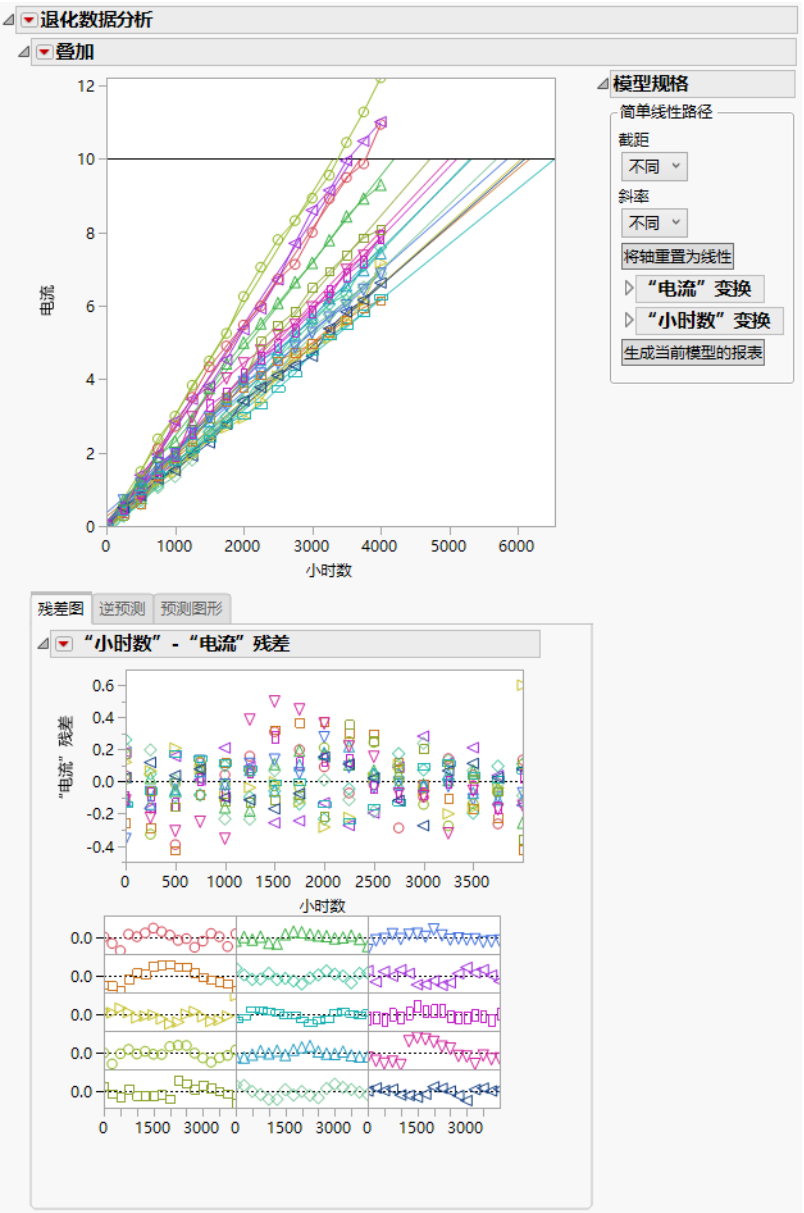


图 7.2 显示初始“退化”报表。叠加图显示数据中的每个单元的**时间**测量值 - **电流**测量值。**电流** = 10 处的水平线对应于 10% 处的上规格限。值高于该限值的单元被视为已失效。到研究的末期，15 个单元中有 3 个达到了这一点。“逆预测”分级显示项根据您指定的模型显示每个单元失效时的预测**小时数**值。

默认模型使用线性变换为每个单元拟合单独的斜率和截距。您可以使用“模型规格”分级显示项拟合其他模型。

图 7.2 中的“残差图”选项卡根据您指定的模型显示残差。上图显示针对**小时数**标绘的所有单元的残差，全部叠加在一张图上。下图在一个矩形阵列中显示分别对应每个单元的单独残差图。

启动“退化”平台

通过选择**分析 > 可靠性和生存 > 退化**来启动“退化”平台。图 7.3 显示了使用 GaAs Laser.jmp 数据表（位于 Reliability 文件夹下）的“退化”启动窗口。有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

图 7.3 “退化”启动窗口

分析类型

启动窗口拆分为三个选项卡，表示三种不同类型的分析：

重复测量退化 执行线性或非线性退化分析。该选项仅允许使用一个“Y，响应”变量。不允许删失。

破坏性退化 若单元在测量过程中遭到破坏，则选择该分析类型。该选项允许删失。请参见第 177 页的“破坏性退化”。

注意：“破坏性退化”平台为破坏性检验提供一系列灵活的预定义模型。请参见第 185 页的“破坏性退化”一章。

稳定性检验 为设置医药产品的失效日期执行稳定性分析。该选项仅允许使用一个“Y，响应”变量。请参见第 182 页的“稳定性分析”。

启动窗口选项

启动窗口包含以下选项：

Y，响应 分配包含退化测量值的列。

时间 分配包含时间值的列。

X （仅适用于“重复测量退化”选项卡和“破坏性退化”选项卡。）分配说明变量。使用该角色可指定加速退化模型中的加速因子。

标签，系统 ID （仅适用于“重复测量退化”选项卡和“稳定性检验”选项卡。）分配指定单元 ID 的列。

频数 分配提供每行频数的列。

删失 （仅适用于“破坏性退化”选项卡。）分配指定某个单元是否删失的列。

依据 分配一个变量，以便为该变量的每个水平生成单独的分析。

删失代码 （仅适用于“破坏性退化”选项卡。）指定“删失”列中用来指定右删失观测的值。选定某个“删失”列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中进行选择。您还可以在框中输入不同值。若“删失”列包含“值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

上规格限 指定上规格限。（除“稳定性检验”选项卡外均可选。）

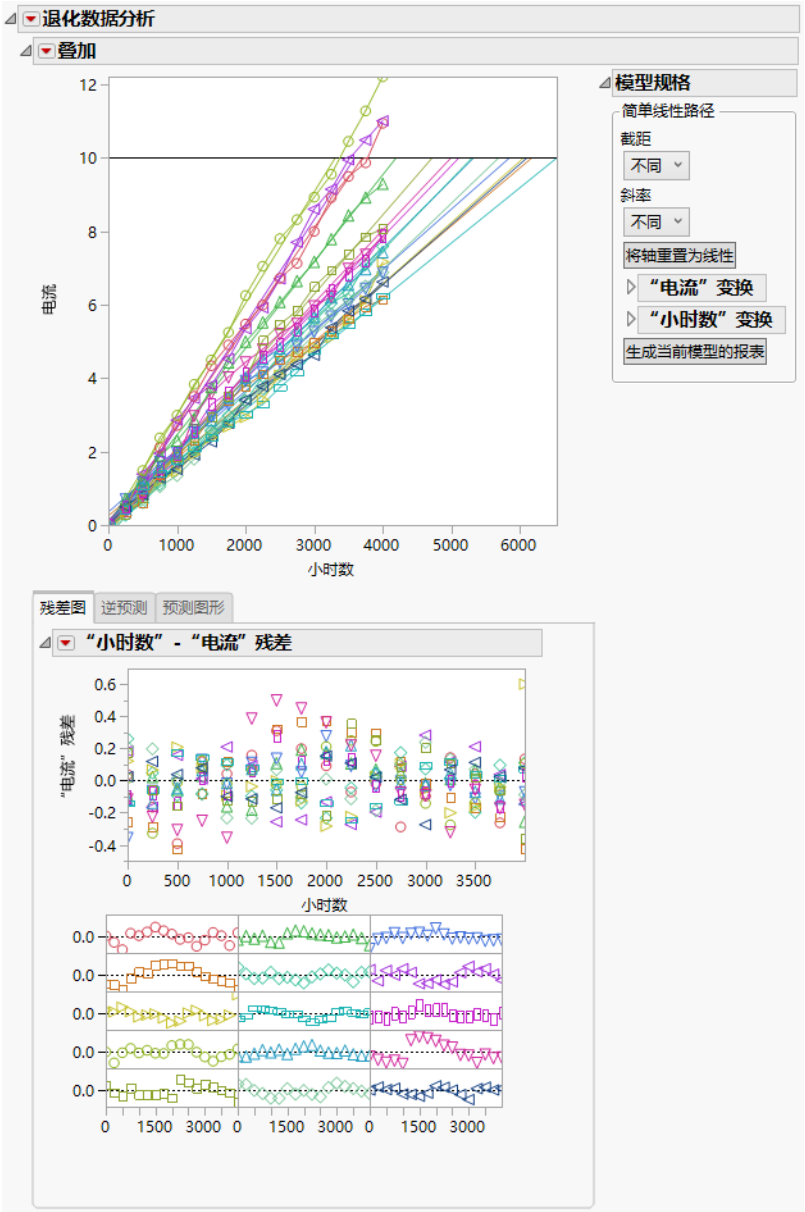
下规格限 指定下规格限。（除“稳定性检验”选项卡外均可选。）

删失时间 （仅适用于“重复测量退化”选项卡和“稳定性检验”选项卡。）分配在您使用“逆预测”时表示伪失效删失的时间值。请参见第 172 页的 “逆预测”。

退化报表

“重复测量退化”方法和“破坏性退化”方法显示拟合默认模型的报表。如图 7.4 中的“模型规格”分级显示项所示，该模型通过响应和时间列的线性变换使用模型自己的截距和斜率拟合每个单元。为“标签,系统 ID”变量的每个值拟合单独的截距和斜率；或者，若仅指定了 X 变量，则为 X 变量的每个水平拟合单独的截距和斜率。“稳定性检验”方法拟合三个模型。

图 7.4 “变换”分级显示项处于打开状态的初始“重复测量退化”报表



要重现本例，请参见第 157 页的“ “退化”平台的示例”。

“重复测量退化”报表、“破坏性退化”报表和 “稳定性检验”报表包括以下信息：

叠加

“时间”变量-“Y，响应”变量的叠加图。在本例中为**小时数-电流**图。“叠加”图红色小三角菜单包含保存估计值选项，该选项可创建包含所有单元的估计斜率和截距的新数据表。

模型规格

指定模型并生成该模型的报表。请参见第 162 页的“模型规格”。（仅适用于“重复测量退化”方法和“破坏性退化”方法。）

“稳定性检验”分级显示项

比较模型并估计失效日期。请参见第 182 页的“稳定性分析”。（仅适用于“稳定性检验”方法。）

报表

显示三个不同模型和最佳模型的分析结果。请参见第 182 页的“稳定性分析”。（默认情况下适用于“稳定性检验”方法。点击“生成当前模型的报表”后，还适用于“重复测量退化”方法和“破坏性退化”方法。）

选项卡式报表

残差图 显示一个叠加了所有单元的单个残差图，以及排列在矩形网格中的分别对应每个单元的单独残差图。红色小三角菜单包含以下选项：

保存残差 将当前模型的残差保存到新数据表中。

随机散布 在时间方向上为点添加随机噪声。当有许多点聚集在一起时，该选项有助于直观演示数据。

分隔组 在组之间添加间距以便直观分隔各组。仅当在平台启动窗口上指定了 X 变量时，才显示该选项。

随机散布尺度 更改随机散布和组分隔的幅度。仅在选定“随机散布”时才显示该选项。

逆预测 支持您预测 Y 变量达到指定值的时间。请参见第 172 页的 “逆预测”。

预测图形 支持您预测指定的 “时间”值对应的 Y 变量。请参见第 173 页的 “预测图形”。

模型规格

您可以使用 “模型规格”分级显示项指定您想要对退化数据拟合的模型。有两种模型规格：

简单线性路径 用于对线性退化路径或可变换为线性路径的非线性路径建模。请参见第 163 页的 “简单线性路径”。

非线性路径 用于对非线性退化路径（特别是那些无法变换为线性路径的非线性路径）建模。
请参见第 164 页的 “非线性路径”。

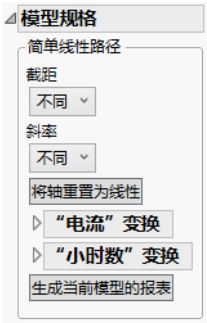
要在两种规格之间切换，请使用“退化数据分析”红色小三角菜单中的“退化路径样式”子菜单。

简单线性路径

要对线性退化路径建模，请从“退化数据分析”红色小三角菜单中选择**退化路径样式 > 简单线性路径**。

使用“简单线性路径”模型规格可指定您想要对退化路径拟合的线性模型的形式。您可以对线性路径或可变换为线性路径的非线性路径建模。

图 7.5 “简单线性路径”模型规格



“简单线性路径”模型规格包含以下选项：

截距 指定模型中截距的类型：

不同 对 ID 变量的每个水平拟合不同的截距。

在组中相同 对 X 变量同一水平内的 ID 变量的每个水平拟合相同的截距，但对不同水平拟合不同的截距。

相同 对 ID 变量的所有水平拟合相同的截距。

零 针对 ID 变量的所有水平，将截距限定为 0。

斜率 指定模型中斜率的格式：

不同 对 ID 变量的每个水平拟合不同的斜率。

在组中相同 对 X 变量同一水平内的 ID 变量的每个水平拟合相同的斜率，但对不同水平拟合不同的斜率。

相同 对 ID 变量的所有水平拟合相同的斜率。

将轴重置为线性 将 “叠加”图的轴恢复其初始设置。

<Y，响应 > 变换 若针对 Y 变量的变换可使退化路径线性化，请在此选择变换（线性、ln(x)、exp(x)、x^2、sqrt(x) 或定制）。有关“定制”选项的详细信息，请参见第 164 页的“定制变换”。

< 时间 > 变换 若针对“时间”变量的变换可使退化路径线性化，请在此选择变换（线性、ln(x)、x^2、sqrt(x) 或定制）。有关“定制”选项的详细信息，请参见第 164 页的“定制变换”。

生成当前模型的报表 针对当前模型设置创建一个报表。这将包括“模型汇总”报表以及提供参数估计值的“估计值”报表。请参见第 176 页的“模型报表”。

定制变换

若需要执行并未提供的变换，可以使用“定制”选项。例如，要使用 $\exp(-x^2)$ 变换响应变量，请按照图 7.6 中的“尺度”框所示输入变换。此外，在“反转尺度”框中输入逆变换。

注意：JMP 自动尝试对逆变换求解。若可以对逆变换求解，则在“反转尺度”框中自动输入逆变换。若无法对逆变换求解，您必须手动输入逆变换。

图 7.6 “定制”变换选项

使用文本框为变换命名。完成后，点击**使用并保存**按钮可应用该变换。若创建了多个定制变换，则从菜单中选择所需的变换。点击**删除**按钮可删除定制变换。

非线性路径

要对非线性退化路径建模，请从“退化数据分析”红色小三角菜单中选择**退化路径样式 > 非线性路径**。若无法使用变换将退化路径线性化，或是您想拟合数据的定制非线性模型，该选项很有用。

为便于解释 “非线性路径模型规格”，请打开 Device B.jmp 数据表。该数据包含在四个温度水平上测量的 34 个单元的功率下降测量值。按以下步骤执行：

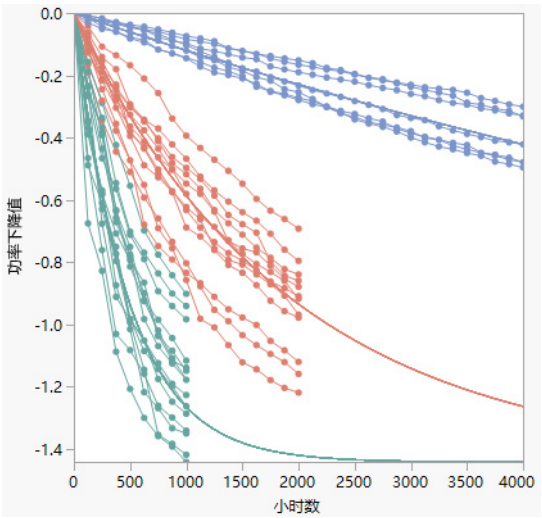
1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Device B.jmp。
2. 选择**分析 > 可靠性和生存 > 退化**。

3. 选择**功率下降值**并点击 **Y，响应**。
4. 选择**小时数**并点击**时间**。
5. 选择**摄氏度数**并点击 **X**。

   温度设置是实验中的加速因子。
6. 选择**设备**并点击**标签**，**系统 ID**。
7. 点击**确定**。

随即显示数据的初始叠加图。

图 7.7 “Device B”的叠加图



退化路径在开始的几百个小时呈线性，之后开始弯曲。要拟合非线性模型，请从“退化数据分析”红色小三角菜单中选择**退化路径样式 > 非线性路径**，以显示“非线性路径模型规格”分级显示项（图 7.8）。

注意：要查看图 7.8 中显示的“编辑”按钮，您必须在启动“退化”平台之前选择交互式公式编辑器首选项（**文件 > 首选项 > 平台 > 退化 > 使用交互式公式编辑器**）。

图 7.8 初始 “非线性模型规格” 分级显示项

创建模型的第一步是在最初标记为**空**的菜单上选择一个选项：

- 有关 “反应率” 模型的详细信息，请参见第 166 页的 “反应率模型” 。
- 有关 “恒定比率” 模型的详细信息，请参见第 166 页的 “恒定比率模型” 。
- 有关使用 “预测列” 的详细信息，请参见第 167 页的 “预测列” 。

反应率模型

当退化发生在单个化学反应中并且反应率只是温度的函数时，“反应率” 选项适用。从图 7.8 所示的菜单中选择**反应率**或**反应率 I 型**。尽管类似于 “反应率” 模型，但 “反应率 I 型” 模型包含一个偏移项，该项可更改与响应值符号相关的基本假设。

“设置” 窗口提示您选择温度尺度和基线温度。基线温度用于生成参数值的初始估计值。基线温度应该能够代表研究中使用的温度。

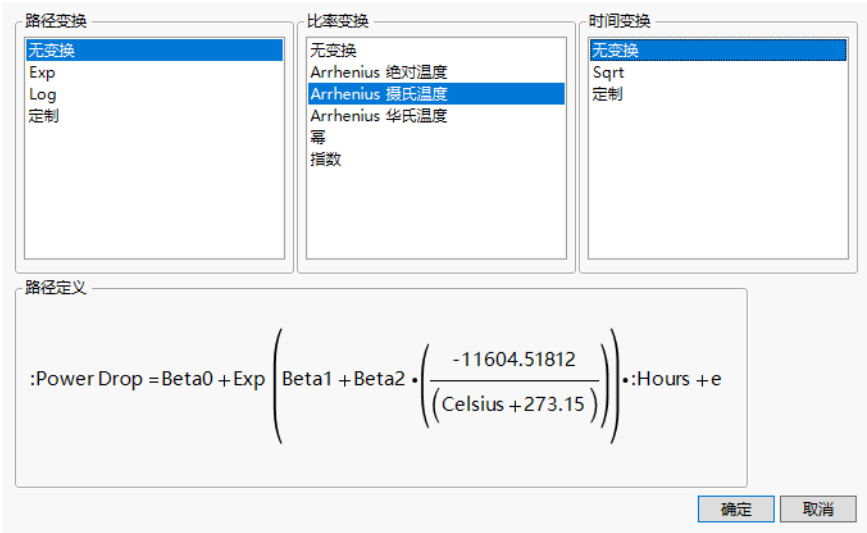
对于本例，请选择**反应率**，然后将 “摄氏温度” 选作 “温度单位” 。点击**确定**返回该报表。有关 “模型规格” 所有功能的详细信息，请参见第 168 页的 “ “模型规格” 详细信息” 。

恒定比率模型

“恒定比率” 选项用于针对相对于时间呈线性（或相对于变换响应或时间后的时间呈线性）的退化路径建模。反应率只是温度的函数。

从图 7.8 所示的菜单中选择**恒定比率**。“恒定比率模型设置” 窗口提示您输入对 “路径” 、“比率” 和 “时间” 的变换。

图 7.9 恒定比率变换



一旦对每个变换做出选择，关联的公式即显示在图 7.9 所示的左下角。

做出所有选择后，点击**确定**返回该报表。有关“模型规格”所有功能的详细信息，请参见第 168 页的““模型规格”详细信息”。

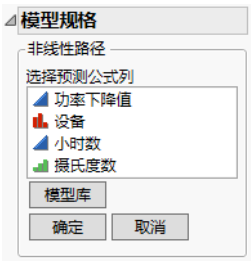
预测列

“预测列”选项支持您使用存储为数据表列中的公式的定制模型。最简单的方法是在启动“退化”平台前创建公式列。若您想要使用“非线性模型库”中的内置模型之一，您也可以在“退化”平台中创建公式列。

有关如何创建定制模型并将其存储为列公式的详细信息，请参见第 170 页的“拟合定制模型”或《预测和专业建模》。

从显示在图 7.8 的“表达式”区域下方的列表中选择**预测列**。“模型规格”分级显示项将更改以提示您选择包含该模型的列。

图 7.10 列选择



此时执行以下三个操作之一：

- 若您想要使用的模型已作为数据表某列中的公式存在，请在此选择相应列，然后点击**确定**。您将返回“非线性路径模型规格”。有关该规格的所有功能的详细信息，请参见第 168 页的““模型规格”详细信息”。
- 若您想要使用的模型在数据表中不存在，您可以点击**模型库**按钮使用内置模型之一。有关使用“模型库”按钮的详细信息，请参见第 171 页的“模型库”或《预测和专业建模》。创建模型后，从“退化数据分析”红色小三角菜单中选择**重新运行 > 重新运行分析**。然后，返回图 7.10 所示的列选择。选择包含该模型的列，然后点击**确定**。您将返回“非线性路径模型规格”。有关该规格的所有功能的详细信息，请参见第 168 页的“模型规格”详细信息”。
- 若您想要使用的模型不在数据表中，而您不想使用内置模型之一，那么您尚不能使用该模型规格。首先，创建模型，重新启动“退化”平台，然后返回列选择（图 7.10）。选择包含该模型的列，然后点击**确定**。您将返回“非线性路径模型规格”。有关该规格的所有功能的详细信息，请参见第 168 页的“模型规格”详细信息”。

“模型规格”详细信息

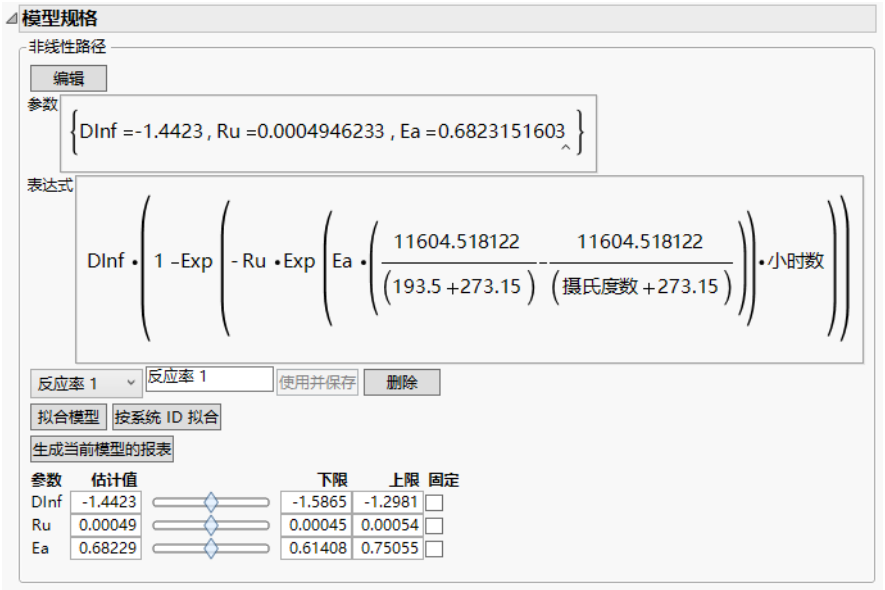
在您选定一种模型类型并提供了所需信息后，系统会让您返回“非线性路径模型规格”窗口。

注意：要查看图 7.11 中显示的“编辑”按钮，您必须在启动“退化”平台之前选择交互式公式编辑器首选项（**文件 > 首选项 > 平台 > 退化 > 使用交互式公式编辑器**）。

有一个模型现在显示在使用**参数**语句的脚本框中。参数的初始值是从数据中估计得到的。有关创建使用参数的模型的详细信息，请参见第 170 页的“拟合定制模型”或《预测和专业建模》。

若需要，在文本框中输入名称为模型命名。对于本例，请使用名称“Device RR”。之后，点击**使用并保存**按钮进入该模型并激活其他按钮和功能。图 7.11 显示点击“使用并保存”按钮后出现的“模型规格”窗口。

图 7.11 模型规格



- **拟合模型**按钮用数据拟合模型。
- **按系统 ID 拟合**用于按**标签**，**系统 ID** 的每个水平拟合模型。
- **删除**按钮用于从模型菜单中删除模型。
- **生成当前模型的报表**按钮针对当前模型设置创建报表。请参见第 176 页的“模型报表”。

初始参数值显示在底部，同时显示滑块，用来直观演示参数的变化如何影响模型。拟合线显示在叠加图上。移动参数滑块以查看更改会如何影响拟合线。

下面是“反应率”模型的参数 (Meeker and Escobar, 1998)：

- Dinf (D_∞) — 渐近退化水平
- Ru (R_U) — 使用温度 ($temp_U$) 下的反应率
- Ea (Ea) — 特定反应的活化能

以上参数计算如下：

$$D(t; \text{temp}) = D_\infty \times \{1\text{-}\exp[-R_U \times AF(\text{temp}) \times t]\}$$

其中，R_U 是使用温度 $temp_U$ 下的反应率，$R_U \times AF(\text{temp})$ 是常规温度 temp 下的反应率，并且对于 $\text{temp} > temp_U$，$AF(\text{temp}) > 1$

且

$$AF(\text{temp}) = \frac{R(temp)}{R(temp_U)} = \exp\left[Ea\left(\frac{11604.5181215503}{(temp_U K)} - \frac{11604.5181215503}{(temp K)}\right)\right]$$

其中，$temp_U K$ 和 $tempK$ 是用 Kelvin 尺度表示的温度。

要计算参数的最优值，请点击**拟合模型**或**按系统 ID 拟合**按钮。

要固定某个参数的值，请在**固定**下方选中该参数对应的框。一旦固定，该参数在模型拟合过程中将保持不变。

使用“公式编辑器”输入模型

您可以使用“公式编辑器”输入模型。点击**编辑**按钮打开“公式编辑器”，输入参数和模型。有关在“公式编辑器”中输入参数和公式的详细信息，请参见《使用 JMP》。

注意：要查看图 7.12 中显示的“编辑”按钮，您必须在启动“退化”平台之前选择交互式公式编辑器首选项（**文件 > 首选项 > 平台 > 退化 > 使用交互式公式编辑器**）。

图 7.12 备择 “模型规格”报表

拟合定制模型

若您想要拟合定制模型，您必须首先创建包含初始参数估计值的公式列。该方法所需的步骤比拟合内置模型多一些，但它允许拟合任何非线性模型。此外，您可以提供定制损失函数，并且可以为拟合过程指定一些其他选项。

1. 打开数据表。
2. 在数据表中创建新列。
3. 为新列打开 “公式编辑器”。
4. 从左下角的列表中选择**参数**。
5. 点击**新建参数**。
6. 输入参数名称。
7. 输入参数的初始值。

   重复执行步骤 4 到 6，以创建模型中的所有参数。
8. 使用数据表列、参数和公式编辑器函数构建模型公式。

9. 点击**确定**。

包含分组变量的模型的参数

在公式编辑器中，当您添加参数时，可以看到**扩展到类别，选择列**复选框。该选项用于同时添加若干参数（例如，为分类变量的每个水平各添加一个参数）。选择该选项时，会出现一个窗口，支持您选择一列。选择之后，“参数”列表中将显示名为 D_ 列的新参数，其中 D 是您提供给参数的名称。在公式中使用该参数时，将插入一个 Match 表达式，它为分类变量的每个水平包含一个单独的参数。

模型库

“模型库”可支持您创建包含参数和初始值的公式列。点击“模型规格”下的**模型库**打开该库。选择列表中的某个模型即可在**公式**框中查看其公式。

点击**显示图形**可为单参数模型显示二维理论曲线，为双参数模型显示三维曲面图。对于包含两个以上说明 (*X*) 变量的模型，将不提供图形。使用滑块更改参数的默认起始值。您还可以点击这些值并直接输入新值。

重置按钮用于将参数的起始值设置回其默认值。

点击**显示点**可将实际数据点叠加到图上。随即打开一个窗口，要求您将列分配给 *X* 和 *Y* 角色以及可选的“分组”角色。“分组”角色允许您为分类变量的每个水平拟合模型。若您在此指定“分组”角色，则同时还要在平台启动窗口的“标签，系统 ID”角色中指定相同列。

对于大多数模型，起始值是常数。图中的散点有助于您调整参数值，以查看使用不同的参数值时模型拟合的好坏程度。

点击**生成公式**在数据表中创建新列。该列包含的公式是指定的 *X* 变量的函数，该列使用图形窗口中指定的参数值。

注意：若在点击**显示图形**或**显示点**按钮之前点击**生成公式**，系统会要求您提供 *X* 和 *Y* 角色，以及可选的“分组”角色。之后，您会返回图中，这样您可通过选项来调整参数起始值。调整为满意的参数起始值后，再次点击**生成公式**创建新列。

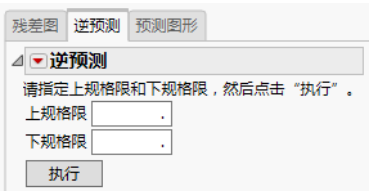
一旦在数据表中创建公式，点击“退化数据分析”红色小三角并选择**重新运行 > 重新运行分析**。然后，返回第 167 页上的图 7.10 所示的列选择。选择包含该模型的列，然后点击**确定**。您将返回“非线性路径模型规格”。有关该规格的所有功能的详细信息，请参见第 168 页的“‘模型规格’详细信息”。

注意：您可以通过修改名为 NonlinLib.jsl 的内置脚本定制“非线性模型库”中包括的模型。该脚本位于 JMP 所在文件夹的 Resources/Builtins 文件夹 (Windows) 或 Application Package (macOS) 中。

逆预测

使用“逆预测”选项卡预测 Y 变量达到指定值的时间。这些时间有时称为伪失效时间。

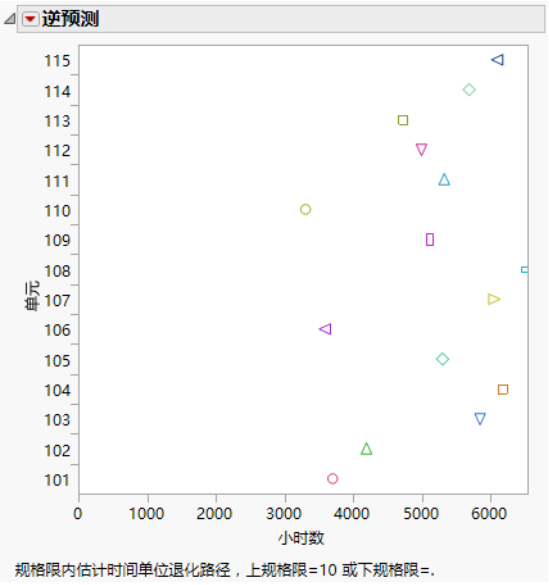
图 7.13 “逆预测”选项卡



输入“下规格限”或“上规格限”。通常，若 Y 变量随时间减少，则输入“下规格限”。若 Y 变量随时间增加，则输入“上规格限”。

对于 GaAs Laser 示例，请为“上规格限”输入 10 并点击**执行**。将生成一个图，显示单元达到 10% 的工作电流增量时的估计时间。

图 7.14 “逆预测”图



“逆预测”红色小三角菜单包含以下选项：

保存交叉时间 将伪失效时间保存到新数据表。该表包含“寿命分布”或“以 X 拟合寿命”脚本，可用于拟合伪失效时间的分布。若启用了“逆预测区间”选项之一，该表还包含区间。

设置上规格限 设置上规格限。

设置下规格限 设置下规格限。

设置删失时间 设置删失时间。该图更新，将 “删失时间”显示为垂直虚线。若选定 “逆预测区间”>“无区间”，超过 “删失时间”的观测显示在从 “删失时间”开始的水平线上。若选定 “置信区间”或 “预测区间”，水平线将无限延伸至其上限超出删失时间的观测的右侧。“删失时间”反映在使用 “保存交叉时间”和 “生成伪失败数据”构造的数据表中。

在数据进行内插 指定在点与点之间使用线性插值（而不是拟合模型）来预测单元何时跨越规格限。该行为取决于单元是否包含超过规格限的观测。

- 若单元包含的观测超过规格限，逆预测则为规格限周围观测之间的线性插值。
- 若单元不包含超过规格限的观测，则逆预测删失，并且它的值等于该单元的最大观测时间。

逆预测区间 指定在 “逆预测”图上显示伪失效时间的置信区间或预测区间。若启用了区间，使用 “保存交叉时间”选项时所创建的数据表中也会包含区间。

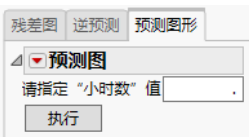
逆预测 Alpha 指定用于区间的 alpha 水平。

逆预测单侧或双侧区间 指定单侧或双侧区间。

预测图形

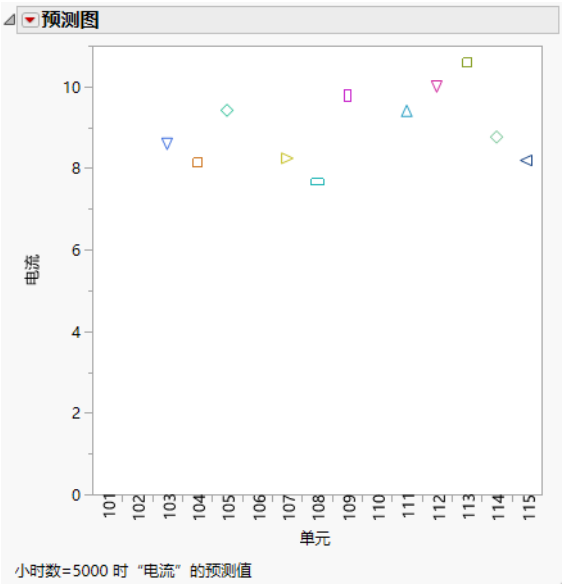
使用“预测图形”选项卡可预测指定的“时间”值对应的 Y 变量。

图 7.15 “预测图”选项卡



对于 GaAs Laser 示例，未收集 4000 小时后的数据。若想要预测 5000 小时后工作电流的增量百分比，请输入 5000 并点击**执行**。将生成一个图，显示所有单元在 5000 小时后的估计降幅百分比。

图 7.16 预测图



“预测图”红色小三角菜单包含以下选项：

保存预测 将预测的 Y 值保存到数据表中。若启用了 “纵向预测区间”选项之一，该表还包含区间。

纵向预测区间 在 “预测图”上显示估计的 Y 值的置信区间或预测区间。若启用了区间，使用 “保存预测”选项时所创建的数据表中也会包含区间。

纵向预测时间 指定要预测 Y 的时间值。

纵向预测 Alpha 指定用于区间的 alpha 水平。

“退化”平台选项

“退化数据分析”红色小三角菜单包含以下选项：

路径定义 假定给定时间的 Y 变量服从某种分布。您可以对该分布的均值、位置参数或中位数建模。

均值路径 指定均值为要建模的参数。

位置参数路径 指定其分布的位置参数是要建模的参数。

中位数路径 指定其分布的中位数是要建模的参数。

选择 “位置参数”或 “中位数路径”选项后，“模型规格”中将出现一个菜单，该菜单支持您选择响应的分布。

退化路径样式 包含用于选择要拟合的退化路径样式的选项。

简单线性路径 支持您拟合线性退化路径和可变换为线性路径的非线性路径。请参见第 163 页的“简单线性路径”。

非线性路径 支持您拟合非线性退化路径。请参见第 164 页的“非线性路径”。

图形选项 包含用于修改报表中图形外观的选项。

连接数据标记 在“叠加”图上显示或隐藏连接各点的线。

显示拟合线 在“叠加”图上显示或隐藏拟合线。

显示规格限 显示或隐藏“叠加”图上的规格限。

显示残差图 显示或隐藏残差图。

显示逆预测图 显示或隐藏逆预测图

显示曲线区间 在“叠加”图上显示或隐藏拟合线的置信区间。

曲线区间 Alpha 支持您更改用于置信区间曲线的 alpha。

显示中位数曲线 当“路径定义”设置为“位置参数路径”时，在图上显示或隐藏中位数线。

显示图例 显示或隐藏“叠加”图上使用的标记的图例。

无选项卡式列表 在选项卡或堆叠报表中显示或隐藏“残差图”、“逆预测”和“预测图形”。

预测设置 包含用于修改模型预测中使用的设置的选项。

上规格限 指定上规格限。

下规格限 指定下规格限。

删失时间 设置删失时间。请参见第 172 页的“逆预测”。

基线 为非线性退化路径中的 X 变量指定正常使用条件。具有该值的路径通常生成“叠加”图。

逆预测 指定逆预测的区间类型、alpha 水平以及单侧或双侧区间。要进行逆预测，您还必须指定下规格限或上规格限。请参见第 172 页的“逆预测”。

纵向预测 指定纵向预测的“时间”值、区间类型和 alpha 水平。请参见第 173 页的“预测图形”。

应用 包含用于进一步分析退化数据的选项。

生成伪失败数据 创建一个数据表，其中提供每个单元跨越规格限的预测时间。该表包含“寿命分布”或“以 X 拟合寿命”脚本，可用于拟合伪失效时间的分布。

检验稳定性 执行稳定性分析。请参见第 182 页的“稳定性分析”。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

模型报表

点击**生成当前模型的报表**按钮后，将在两个位置添加汇总报表：

- 向 “模型列表”报表添加一个条目。请参见第 176 页的 “模型列表”。
- 向 “报表”报表添加一个条目。请参见第 176 页的 “报表”。

模型列表

“模型列表”报表提供每个拟合模型的汇总统计量和其他选项。图 7.17 显示了包含三个模型的汇总的 “模型列表”的示例。

图 7.17 模型列表

模型列表

显示		模型类型	报表	参数数目	-2对数似然	AICc	BIC	误差平方和	自由度	说明
○	1	简单线性路径	☑	30	-179.299	-110.995	-13.0606	7.39094	225	截距:不同；斜率:不同；Y:线性；X:线性
○	2	简单线性路径	☑	16	-115.57	-81.2841	-26.9096	9.489362	239	截距:相同；斜率:不同；Y:线性；X:线性
◉	3	简单线性路径	☑	2	756.146	760.1936	767.2286	289.6476	253	截距:相同；斜率:相同；Y:线性；X:线性

显示 选择您想在 “叠加”图、“残差图”、“逆预测”图和 “预测图形”中表示的模型。

模型类型 提供路径类型，即线性或非线性。

报表 选中复选框可显示模型的报表。有关报表的详细信息，请参见第 176 页的 “报表”。

参数数目 提供为模型估计的参数个数。

-2 对数似然 是 -2 倍的对数似然。请参见 《拟合线性模型》。

AICc 提供校正的 Akaike 准则。请参见 《拟合线性模型》。

BIC 提供 Bayes 信息准则。请参见 《拟合线性模型》。

误差平方和 提供模型的误差平方和。

自由度 提供误差自由度。

说明 提供模型说明。

报表

“报表”分级显示项节点提供有关每个模型拟合的详细信息。该报表包括 “模型汇总”报表和 “估计值”报表。

“模型汇总”报表包含以下信息：

<Y，响应 > 尺度 针对响应变量的变换。

< 时间 > 尺度 针对时间变量的变换。

误差平方和 误差平方和。

参数数目 为模型估计的参数个数。

自由度 误差自由度。

R 方 R 方。

均方误差 均方误差。

“估计值”报表包含以下信息：

参数 参数的名称。

估计值 参数的估计值。

标准误差 参数估计值的标准误差。

t 比 参数的 t 统计量，用估计值除以标准差计算得出。

概率 >|t| 参数的双侧检验的 *p* 值。

破坏性退化

要测量产品特性，有时必须破坏产品。例如，在测量断裂强度时，必须给产品施加应力直至其断裂。常规退化分析不再适用于这些情形。您可以采用以下两种方式之一处理这些情形：

- 若您的失效时间模型是标准模型，则可以通过“破坏性退化”平台进行处理。请参见第 185 页的“破坏性退化”一章。
- 若您使用的是时间或响应变量的定制变换以及定制非线性模型，则使用“退化”平台。在启动窗口中，选择“破坏性退化”选项卡。

加速破坏性退化的示例

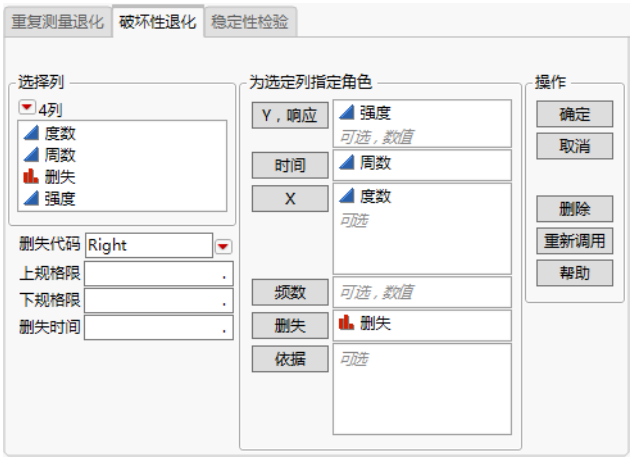
本例拟合定制非线性模型。数据包含某种粘结剂强度的测量值。对产品施加应力，直到粘结剂断裂，并且记录所需的断裂强度。由于单元在正常使用条件下不太可能断裂，所以在加速因子（温度）的若干水平下对单元进行了检验。您想要估计在 25°C 的使用条件下经过 52 周（一年）后单元的强度（以牛顿为单位）。

完成启动窗口

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Adhesive Bond.jmp。
2. 选择**分析 > 可靠性和生存 > 退化**。
3. 选择**破坏性退化**选项卡。

4. 选择**强度**并点击 **Y，响应**。
5. 选择**周数**并点击**时间**。
6. 选择**度数**并点击 **X**。
7. 选择**删失**并点击**删失**。

图 7.18 已完成的启动窗口



8. 点击**确定**。

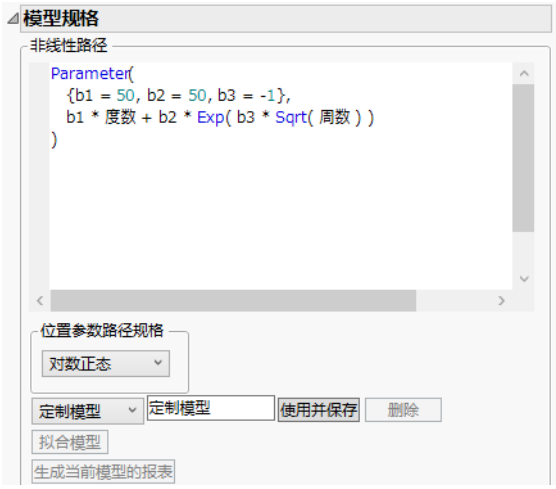
定义和拟合模型

1. 从“模型规格”分级显示项的“位置参数路径规格”面板的菜单中选择**对数正态**。
   这将为响应**强度**指定对数正态变换。
2. 点击“退化数据分析”红色小三角并选择**退化路径样式 > 非线性路径**。
   这会向报表中添加一个脚本窗口。您在该窗口中插入用于指定模型的脚本。
3. 复制下面的 JSL 公式，并将其粘贴到“非线性路径”下方的脚本窗口中：

```
Parameter(
   {b1 = 50, b2 = 50, b3 = -1},
   b1 * :度数 + b2 * Exp( b3 * Sqrt( :周数 ) )
)
```

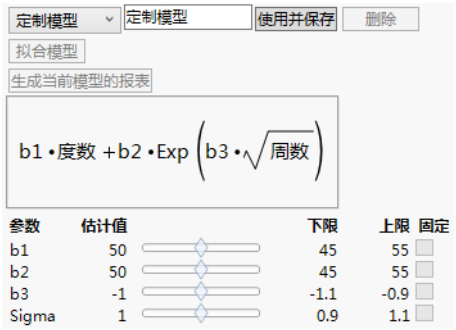
   该脚本以参数 b1、b2 和 b3 的形式为“强度”定义了模型。该脚本指定了参数的初始值。
4. 在报表底部的文本框中，将“空”改为“定制模型”。

图 7.19 非线性路径脚本



5. 点击**使用并保存**。

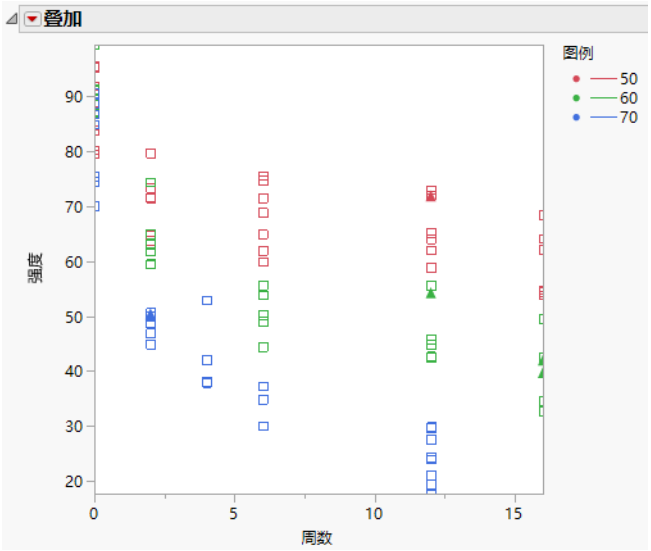
图 7.20 更新的 “模型规格” 分级显示项



该报表将更新，以包括用于更改参数初始值的控件。参数的初始值设置为脚本编辑器的公式中指定的值。

6. 点击**拟合模型**。

图 7.21 拟合模型图



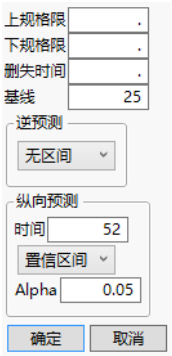
“模型规格”分级显示项中的“参数”面板随之更新，以显示拟合模型的参数估计值。模型拟合显示在“叠加”图中。您可以拖动轴以显示各点，如图 7.21 中所示。图例对曲线进行了标识。

获取预测值和预测区间

接下来，您获取在基准使用条件（25°C）下“强度”在 52 周之后的预测值和预测区间。

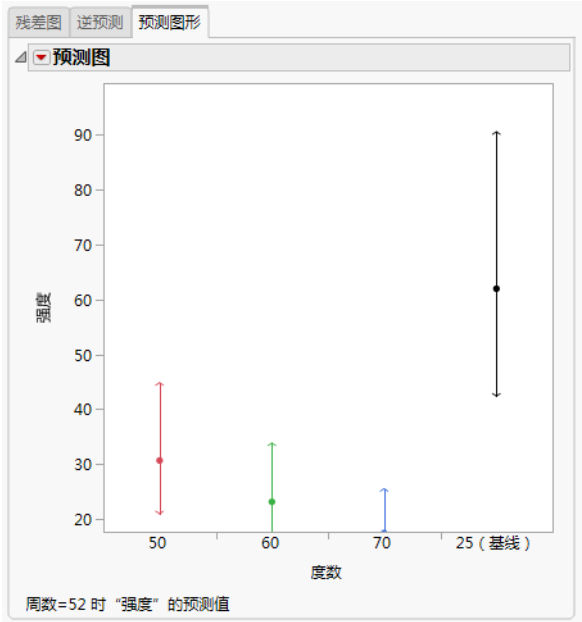
1. 点击“退化数据分析”红色小三角并选择**预测设置**。
2. 在**预测设置**窗口中：
   - 为“基线”输入 25。
   - 在“纵向预测”面板的“时间”中输入 52。
   - 从“纵向预测”面板的菜单中选择“预测区间”。

图 7.22 预测设置



3. 点击**确定**。
4. 选择报表中的**预测图形**选项卡。

图 7.23 周数 = 52 时的预测图



"预测图"显示在轴标签 25 （基线）之上的预测值及其 95% 水平预测区间。

5. 点击 "预测图"红色小三角并选择**保存预测**。

度数的三个值以及期望的基线 （25 度）时的预测值将保存到数据表中。在 25° C 时，粘结剂 1 年之后的预测强度为 61.96，预测区间介于 42.42 到 90.50 之间。

稳定性分析

稳定性分析用于设置医药产品的失效日期。拟合了三个线性退化模型，并且根据国际协调会议 (ICH) 指导原则估计了失效日期。ICH 指导原则用于确定批次是否可以合并以确定失效日期的一般框架 (ICH Q1E 2003)。有关具体实施细节，请参见 Chow（2007，附录 B）中的 STAB 宏和 FDA 指导原则。

“稳定性检验”报表汇总了三种退化模型及其对应的最早交叉时间。选择最佳模型并显示在“叠加”图中。这些模型按复杂程度依次列出。

第一个模型拟合每个批次的不同截距和不同斜率。（在如下所述的过程步骤和模型比较中，这是**完全模型**。）当该模型用于估计失效日期时，均方误差 (MSE) 不会在批次之间合并。使用各均方误差为每个批次计算预测区间，并且使用最先跨越规格限的区间来估计失效日期。当提供两个规格限时，最早交叉时间基于 95% 双侧预测区间；当仅提供一个规格限时，最早交叉时间基于 95% 单侧预测区间。

第二个模型拟合每个批次的不同截距，但在所有批次之间拟合共同斜率。第三个模型在所有批次之间拟合共同截距和共同斜率。若该模型合适，它将提供距今最远的失效日期。

图 7.24 稳定性检验汇总

稳定性检验

在显著性水平 0.25 处接受的最佳模型具有不同的截距和相同的斜率。该模型建议的最早交叉时间为“23.47505”，置信度为 95%。

ICH 指导方针指示过期时间: 23.47505。

显示	截距	斜率	最早交叉时间
○	不同	不同	23.11596
◉	不同	相同	23.47505
○	相同	相同	29.98567

“稳定性检验”报表的“模型比较”部分汇总了每种稳定性模型的显著性检验。“图例”说明每个源。确定失效日期的最佳模型的过程考虑了“源 C”和“源 B”的 p 值。下面按相反顺序列出各个源，以适应过程中各步的顺序。

提示：ICH 指导原则规定使用 0.25 的显著性水平来确定适当的模型。您可以按照以下步骤将每个源的 p 值与不同的显著性水平进行比较。若这导致选择不同的模型，您可以使用“稳定性检验”汇总报表中表格的“显示”列中的单选按钮。

源 E 指定响应的平方和减去源 D 的平方和。源 E 的“均方”列中的值是源 E 平方和的值除以源 E 自由度的结果，源 E 自由度等于完全模型 （不同截距和不同斜率）中的参数个数。

源 D 指定完全模型 （不同截距和不同斜率）的误差平方和与相应的均方误差。

源 C 指定等斜率检验。这是第二个模型 （不同截距和共同斜率）对照完全模型 （不同截距和不同斜率）的检验。

- 若 p 值小于 0.25，则假定斜率在不同批次之间是不同的。过程停止，并使用完全模型（不同截距和不同斜率）来估计失效日期。

– 若 p 值大于等于 0.25，则假定斜率在不同批次之间是相同的，随后使用源 B 计算斜率。

源 B 指定等截距检验。这是第三个模型（共同截距和共同斜率）对照第二个模型（不同截距和共同斜率）的检验。

– 若 p 值小于0.25，则假定截距在不同批次之间是不同的，并且使用第二个模型（不同截距和共同斜率）来估计失效日期。

– 若 p 值大于等于 0.25，则假定截距在不同批次之间是相同的，并且使用第三个模型（共同截距和共同斜率）来估计失效日期。

源 A 指定第三个模型（共同截距和共同斜率）对照完全模型（不同截距和不同斜率）的检验。在确定失效日期的最佳模型的过程中不使用该检验。

图 7.25 稳定性模型比较

模型比较

源	自由度	平方和	均方	F 统计量	概率>F
A	6	61.48956	10.24826	10.10836	<.0001*
B	3	60.48866	20.16289	19.88764	<.0001*
C	3	1.000906	0.333635	0.329081	0.8043
D	28	28.38752	1.01384		
E	8	360214.5	45026.81		

图例

源	截距	斜率	截距	斜率
A	不同	不同	相同	相同
B	不同	相同	相同	相同
C	不同	不同	不同	相同
D 残差				
E 整体模型				

“报表”部分包含通过 “检验稳定性”选项拟合的模型。以下四个模型通过 “检验稳定性”选项拟合：

- 不同截距、不同斜率的模型，其中的 MSE（均方误差）在批次之间合并。（这是“汇总”报表中第一个模型的替代模型。）
- 不同截距，共同斜率的模型。（这是“汇总”报表中的第二个模型。）
- 共同截距，共同斜率的模型。（这是“汇总”报表中的第三个模型。）
- 不同截距、不同斜率的模型，其中的 MSE（均方误差）不在批次之间合并。（这是“汇总”报表中的第一个模型。）

警告：除了“检验稳定性”选项拟合的四个模型之外，“报表”部分还包含在运行“检验稳定性”选项之前“退化”平台中拟合的所有模型。

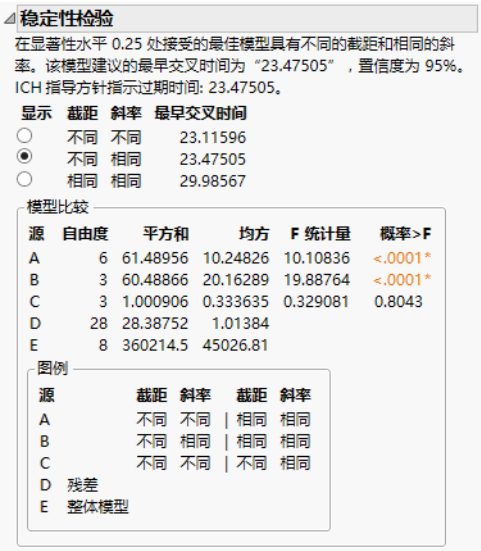
示例

使用 Stability.jmp 样本数据表中的数据为新产品建立失效数据。该数据包含四个批次的产品浓度测量值。浓度为 95 被视为产品的可使用性的结束。

要执行稳定性分析，请执行以下步骤：

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Stability.jmp。
2. 选择**分析** > **可靠性和生存** > **退化**。
3. 选择**稳定性检验**选项卡。
4. 选择**浓度 （毫克 / 千克）**并点击 **Y，响应**。
5. 选择**时间**并点击**时间**。
6. 选择**批次编号**并点击**标签，系统 ID**。
7. 对 “下规格限” 输入 95。
8. 点击**确定**。

图 7.26 稳定性模型



等斜率检验的 p 值为 0.8043。由于该值大于显著性水平 0.25，所以未拒绝该检验，您可以得出结论：退化斜率在不同批次之间是相等的。

等截距和等斜率检验的 p 值 <.0001。由于该值小于显著性水平 0.25，所以拒绝该检验，您可以得出结论：截距在不同批次之间是不同的。

由于未拒绝等斜率检验，但拒绝了等截距检验，所以所选模型是具有 “不同截距” 和 “共同斜率” 的模型。该模型是在报表中选定的模型，并且提供估计的失效日期为 23.475。

第 8 章

破坏性退化

对产品随时间劣化建模

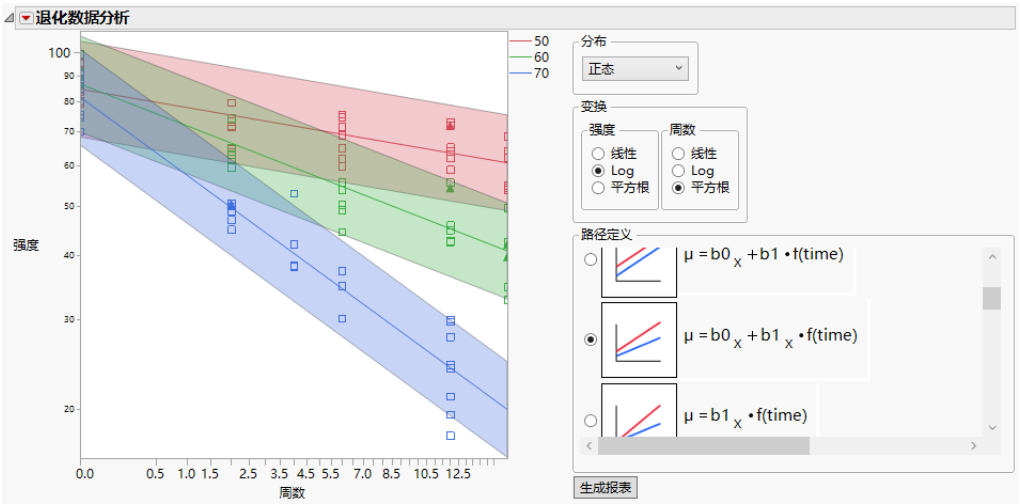
要测量产品特性，有时必须破坏产品。例如，在测量断裂强度时，必须给产品施加应力直至其断裂。由于检验是破坏性的，所以每个产品单元只有一个观测。在这种情况下，您可以使用“破坏性退化”平台对产品可靠性建模。

该平台针对通常非负的响应如何随时间变化建模。观测被假定是独立的，并且它们测量响应值和失效时间。平台提供了一大批灵活的预定义模型。这些模型包括位置-尺度和对数-位置-尺度分布，它们的位置参数是时间的函数。而且这些模型允许使用解释变量和其他参数。若指定解释变量，该平台将拟合加速破坏性退化模型。

注意：若您所需的模型未列在提供的模型中，您可以使用“退化”平台。请参见“退化”一章中第 177 页的“破坏性退化”。

有关破坏性退化和可靠性的详细信息，请参见 Escobar et al. (2003) 和 Meeker and Escobar (1998)。

图 8.1 Adhesive Bond.jmp 的破坏性退化模型示例



目录

“破坏性退化”平台的示例 187
启动“破坏性退化”平台 192
指定两个 Y 列 193
“破坏性退化”图选项和模型 193
图选项 194
模型 195
“破坏性退化”报表 198
模型列表 199
“模型”分级显示项 199
“破坏性退化”平台选项 201
“破坏性退化”平台的统计详细信息 202

“破坏性退化”平台的示例

这一加速破坏性退化模型的示例模仿的是 Escobar et al. (2003) 中的示例。数据包含某种粘结剂强度的测量值（测量单位为牛顿）。温度被视为加速因子。对产品施加应力，直到粘结剂断裂，并且记录所需的断裂强度。由于单元在正常温度下不容易断裂，所以在较大温度范围内的若干水平下对单元进行了检验。强度小于 50 牛顿被视为失效。您想要估计在 35 摄氏度的参考温度下，经过 260 周（5 年）后强度低于 50 牛顿的单元的比例。

本例分三个阶段：

- 第 187 页的“执行初始分析”
- 第 188 页的“更改模型并生成报表”
- 第 190 页的“使用刻画器进行预测”

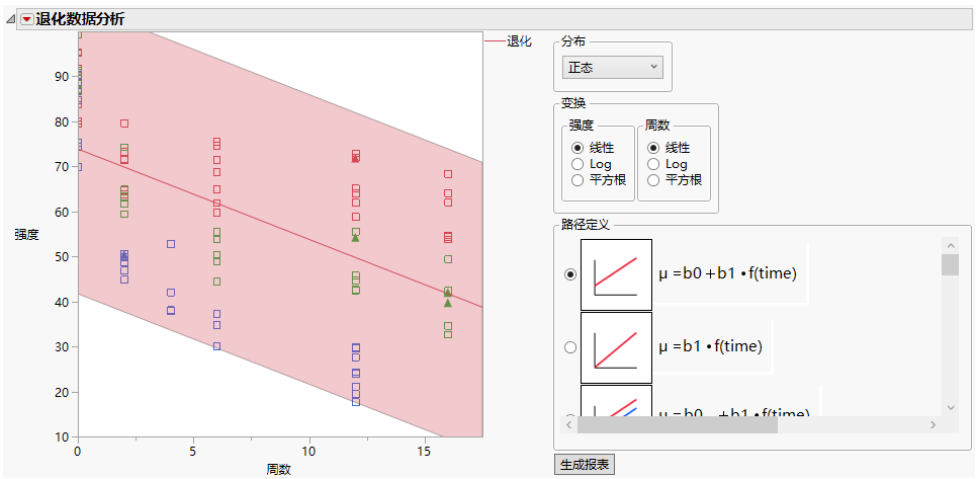
执行初始分析

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Adhesive Bond.jmp。
2. 选择**分析** > **可靠性和生存** > **破坏性退化**。
3. 选择**强度**并点击 **Y，响应**。
4. 选择**周数**并点击**时间**。
5. 选择**度数**并点击 **X**。

   温度是实验中的加速因子。
6. 选择**删失**并点击**删失**。

   请注意，**删失代码**设置为 “Right”。
7. 点击**确定**。

图 8.2 初始退化图



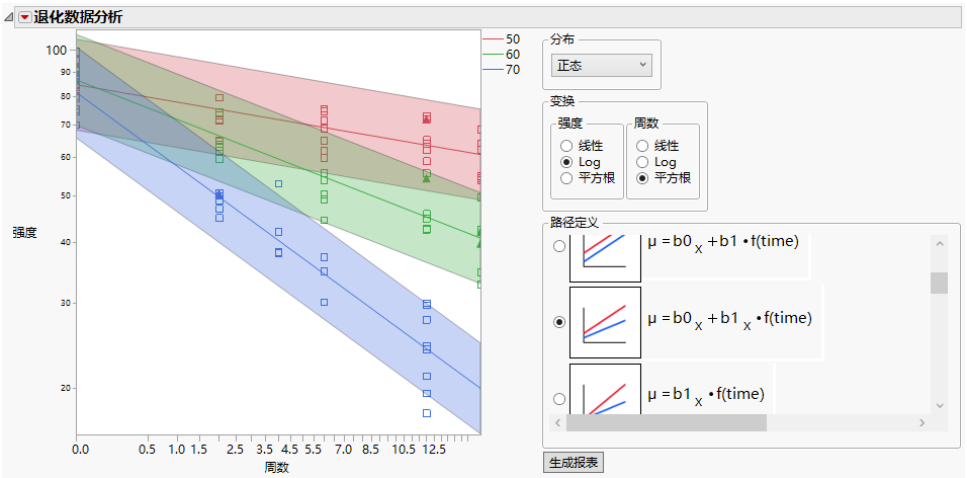
平台指定默认模型。默认模型假定通过单个正态分布（其位置参数是时间的线性函数）来描述数据。

更改模型并生成报表

1. 为 Y（**强度**）变换选择 **Log**。
2. 对 “时间”（**周数**）变换选择**平方根**。
3. 对 “路径定义”选择 $\mu = b0_x + b1_x * f(time)$。
   下标 “x”表示加速变量，即本例中的**度数**。

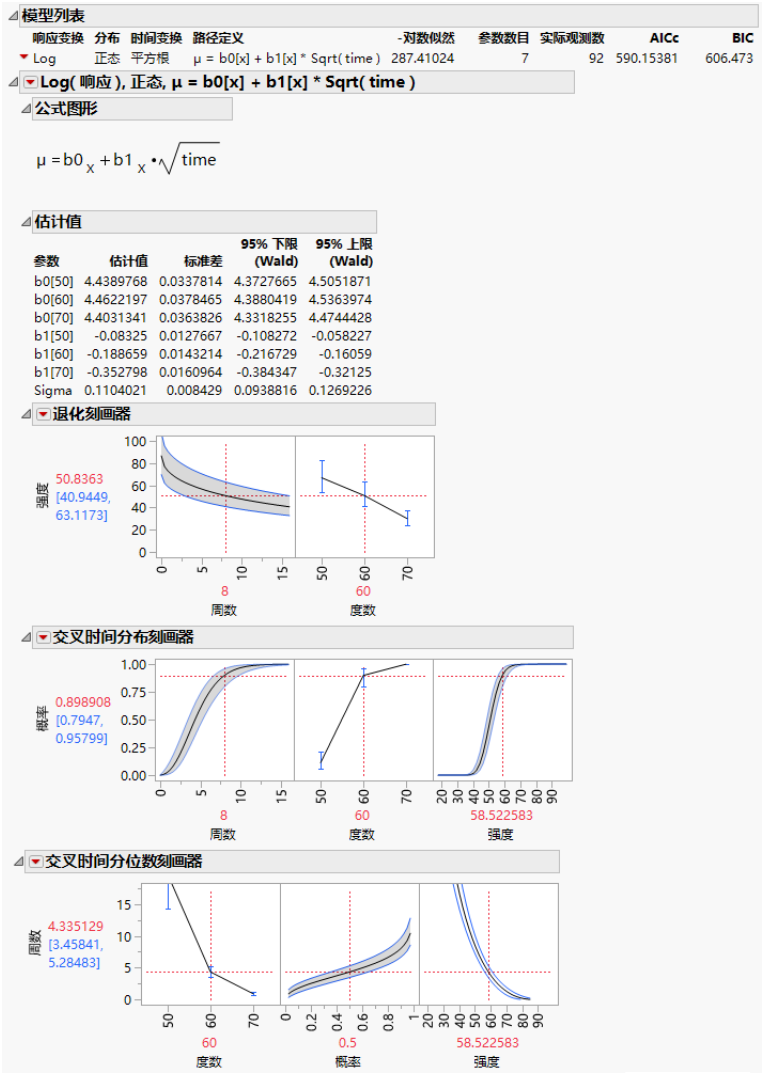
注意：该模型在所有参数中呈线性关系。

图 8.3 显示模型的图



4. 点击**生成报表**。

图 8.4 基本模型的报表



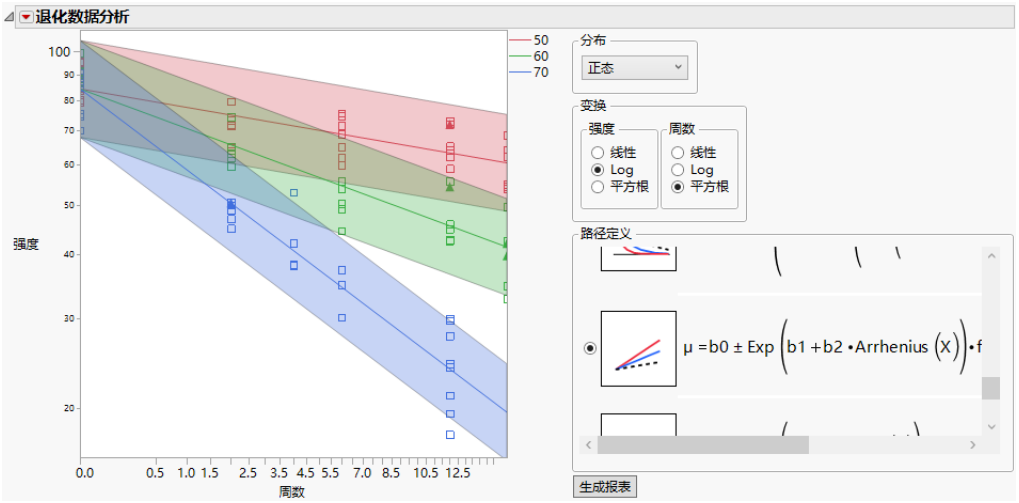
斜率 b1 在**度数**的三个值下的估计值表明在较高温度下会更快地出现退化。依赖化学过程的失效机制通常使用针对温度的 Arrhenius 模型就可以很好地建模。出于此原因，您现在需要拟合一个模型，将 Arrhenius 变换应用到用 “摄氏温度”尺度测量的**度数**。

5. 对 “路径定义”选择 μ = b0 ± Exp(b1 + b2*Arrhenius(X))*f(time)。

注意：该模型在相对参数上不呈线性关系。

6. 选择**摄氏温度**并点击**确定**。

图 8.5 显示应用了 Arrhenius 变换的模型的图



7. 点击**生成报表**。

图 8.6 包括应用了 Arrhenius 变换的第二个模型的报表

模型列表

响应变换	分布	时间变换	路径定义	-对数似然	参数数目	实际观测数	AICc	BIC
Log	正态	平方根	μ = b0[x] + b1[x] * Sqrt(time)	287.41024	7	92	590.15381	606.473
Log	正态	平方根	μ = b0 - Exp(b1 + b2 * Arrhenius(X)) * Sqrt(time)	288.09559	4	92	584.65095	594.27834

Log(响应), 正态, μ = b0[x] + b1[x] * Sqrt(time)

Log(响应), 正态, μ = b0 - Exp(b1 + b2 * Arrhenius(X)) * Sqrt(time)

公式图形

$$\mu = b0 - \text{Exp}\left(b1 + b2 \cdot \text{Arrhenius}\left(X\right)\right) \cdot \sqrt{\text{time}}$$

估计值

参数	估计值	标准差	95% 下限 (Wald)	95% 上限 (Wald)
b0	4.4354686	0.0205344	4.3952219	4.4757154
b1	22.825503	1.4028768	20.075915	25.575091
b2	-0.704728	0.0414641	-0.785996	-0.62346
Sigma	0.1114956	0.0085011	0.0948337	0.1281574

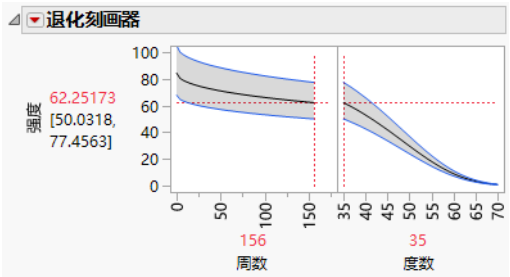
使用刻画器进行预测

由于 Arrhenius 模型显示拟合效果更好（可以通过图 8.6 中该模型较小的 AICc 和 BIC 值看出），您可以使用该模型继续您的分析。

回想一下，强度小于 50 牛顿被视为失效。您关注的是已经在 35 摄氏度的参考温度下使用了 156 周 （三年）的单元。更改刻画器中的设置以反映这些值。点击每个图的水平轴下方的红色值，然后输入新值。

1. 在 Arrhenius 模型的“退化刻画器”中，将**周数**设置为 156，**度数**设置为 35。

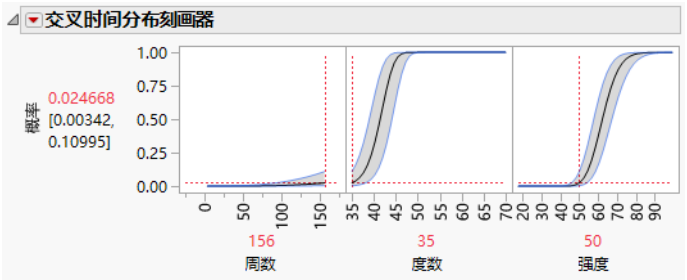
图 8.7 退化刻画器



这些设置下的预测**强度**为 62.25173， 95% 预测区间介于 50.0318 到 77.4563 之间。在这些设置或更温和的设置下，不太可能出现失效。

2. 在 “交叉时间分布刻画器”中，将**周数**设置为 156，**度数**设置为 35，**强度**设置为 50。

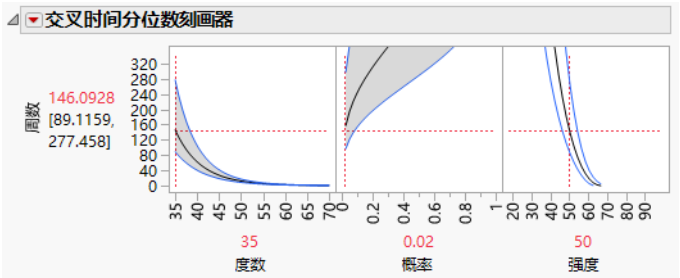
图 8.8 交叉时间分布刻画器



在 35 摄氏度的温度下，经过 156 周后，“强度”值低于 50 的概率为 0.024668。95% 置信区间介于 0.00342 到 0.10995 之间。在这些条件或更温和的条件下的失效概率大约为 2%。

3. 在 “交叉时间分位数刻画器”中，将**度数**设置为 35，**概率**设置为 0.02，**强度**设置为 50。
4. 调整 “交叉时间分位数刻画器”的垂直轴，以便最大值大约为 350。

图 8.9 交叉时间分位数刻画器

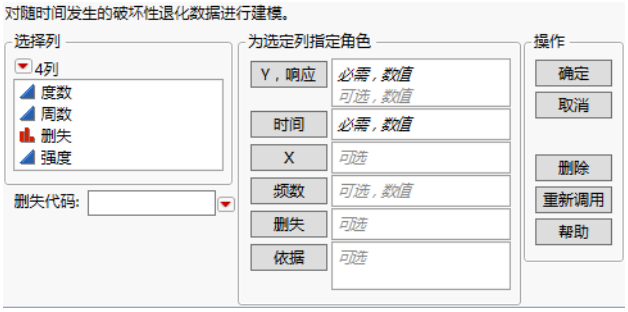


在 35 摄氏度的温度下有 2% 的单元失效所对应的周数估计为 146.0928。95% 置信区间介于 89.1159 到 277.458 之间。

启动“破坏性退化”平台

通过选择**分析 > 可靠性和生存 > 破坏性退化**启动 “破坏性退化”平台。图 8.10 显示了使用 Adhesive Bond.jmp 数据表 （位于 Reliability 文件夹下）的 “破坏性退化”启动窗口。

图 8.10 “破坏性退化”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“破坏性退化”启动窗口包含以下选项：

Y，响应 标识包含退化测量值的列。若响应值为区间删失数据，您可以输入两列。请参见第 193 页的 “指定两个 Y 列”。

时间 标识包含时间值的列。

X 标识可选的解释变量。若 Y 的分布不仅随时间变化，还受其他某个变量的影响，则可将其他变量添加为 X。使用该角色可指定加速破坏性退化模型中的加速因子。

频数 标识有多个单元时的频数或观测计数。若值为 0 或正整数，则这个值表示具有给定行设置的频数或观测计数。

删失 标识用来标识删失的响应测量值的可选列。若 Y 角色中只有一列，该列指示给定行中的响应 Y 是精确测量值还是右删失测量值。右删失观测是精确测量值未知但肯定比相应行中的 Y 值要大的观测。

从 “删失代码”菜单中选择用于标识右删失观测的值。与 “删失”列中的其他值对应的行被视为未删失。包含缺失删失代码值的行将从分析中排除。JMP 尝试检测删失代码并将其显示在列表中。

依据 标识可选的 “依据”变量。为该变量的每个水平生成单独的分析。

删失代码 标识 “删失”列中指定右删失观测的值。选定某个 “删失”列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中进行选择。您还可以在框中输入不同值。若 “删失”列包含 “值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

指定两个 Y 列

若某些退化测量值为区间删失或左删失数据，您可以指定两个 Y 列。对于给定的行，两个 Y 列中的值确定删失类型。

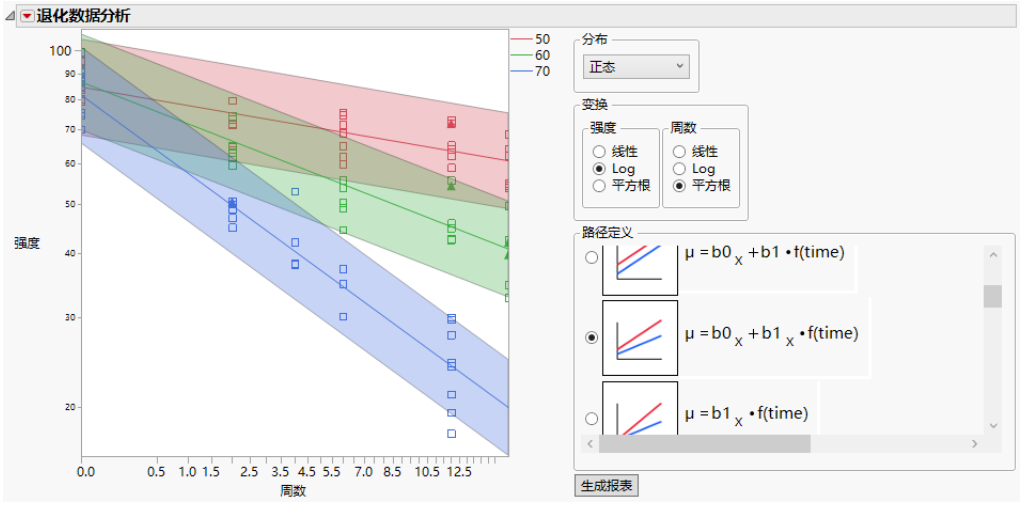
- 若两个 Y 值相等且均不缺失，则共同的测量值被视为精确测量值。
- 若两个 Y 值不相等且均不缺失，则测量值为区间删失数据并假定介于两个值之间。
- 若只是第一个值缺失，则测量值为左删失数据并假定小于第二个值。
- 若只是第二个值缺失，则测量值为右删失数据并假定大于第一个值。

注意：在“破坏性退化”平台中拟合左删失测量值的唯一方法就是使用两个 Y 列。

“破坏性退化”图选项和模型

“退化数据分析”图显示当前基于“分布”、“变换”和“路径定义”的选择所指定的模型的数据和图形化表示。图 8.11 中所示的图针对 Adhesive Bond.jmp 数据表，表示一个包含可选 X 变量的模型。

图 8.11 “破坏性退化”图和选项



请注意以下关于图的事项：

- 数据值通过标记来表示。

表 8.1 标记说明

图标	说明
□	精确测量值。
▼	左删失观测，指示删失测量值位于小三角之下。 **注意：**在“退化”平台中，仅当观测为区间删失数据时才会出现左删失。请参见第 193 页的“指定两个 Y 列”。
▲	右删失观测，指示删失测量值位于小三角之上。
▼ ▲	区间删失观测，指示删失值位于指定区间内。

注意：默认情况下，标记无颜色。要对其着色以匹配模型颜色方案，请选择**行 > 按列设定颜色或标记**。选择 X 列。取消选择**连续尺度**。从 “颜色”菜单中选择 **JMP 默认**。

- 为 X 的每个水平都显示一个着色带。若模型不包含 X 变量，则显示一个着色带。对于给定的“时间”值，着色带的上下限是 Y 的拟合分布的 0.025 和 0.975 百分位数。着色带的颜色对应于 X 值，正如图右上方的图例所示。

注意：标记颜色对应于在数据表中指定的颜色状态。

- 着色带中央的实曲线是针对 X 的相应值 Y 的拟合分布随时间变化的中位数。若模型不包含 X 变量，则曲线标绘 Y 随时间变化的中位数。

图选项

分布 选择位置 - 尺度或对数 - 位置 - 尺度分布。

注意：在为响应列指定对数转换时，不建议拟合对数 - 位置 - 尺度分布模型。

变换 为响应 Y 和时间变量选择变换函数。

注意：若将 “Log”变换应用到包含非正值的列，则将从模型拟合中省略包含非正值的行。若将 “平方根”变换应用到包含负值的列，则将从模型拟合中省略包含负值的行。

路径定义 选择回归模型的线性或非线性路径。有关每个模型的详细信息，请参见第 195 页的“模型”。

生成报表 创建指定模型的报表。首次选择“生成报表”后，会创建一个“模型列表”分级显示项。选择“生成报表”拟合其他模型时，“模型列表”分级显示项随之更新，并为每个模型添加一个分级显示项。

模型

下表提供了“路径定义”列表中每个模型的方程。有关每个模型的说明，请点击相应的链接。

注意：每个方程左侧显示的缩略图显示位置参数 μ 随时间变化的一般行为图。在报表的主图中，根据您的“分布”和“变换”选择，有关估计中位数的图可能不同于缩略图。

表 8.2 模型方程

模型	方程
“带截距的共同路径”	μ = b0 + b1 * f(time)
“不带截距的共同路径”	μ = b1 * f(time)
“共同斜率”	μ = $b0_X$ + b1 * f(time)
“带截距的单个路径”	μ = $b0_X$ + $b1_x$ * f(time)
“不带截距的单个路径”	μ = $b1_X$ * f(time)
“共同截距”	μ = b0 + $b1_X$ * f(time)
“一阶动力学 1 型”	μ = b0 - b1 * Exp[-b2 * Exp[b3 * [Arrhenius(X_0) - Arrhenius(X)]] * f(time)]
“一阶动力学 2 型”	μ = b0 * [1 - Exp[-b1 * Exp[b2 * [Arrhenius(X_0) - Arrhenius(X)]] * f(time)]]
“一阶动力学 3 型”	μ = b0 + b1 * Exp[-b2 * Exp[b3 * [Arrhenius(X_0) - Arrhenius(X)]] * f(time)]
“一阶动力学 4 型”	μ = b0 * Exp[-b1 * Exp[b2 * [Arrhenius(X_0) - Arrhenius(X)]] * f(time)]
“Arrhenius 速率”	μ = b0 ± Exp[b1 + b2 * Arrhenius(X)] * f(time)
“多项式速率”	μ = b0 ± Exp[b1 + b2 * Log(X)] * f(time)

表 8.2 模型方程 (续)

模型	方程
“指数速率”	μ = b0 ± Exp[b1 + b2 * X] * f(time)

在变换时间中呈线性关系的模型

带截距的共同路径

该模型拟合单个分布，其中的位置参数随变换时间线性变化。该模型拟合共同截距和共同斜率，不论是否有 X 变量。

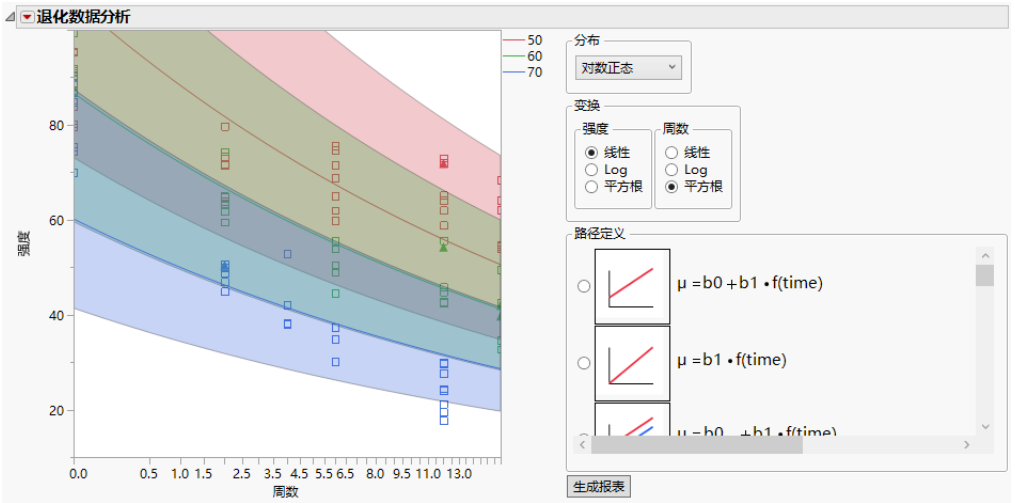
不带截距的共同路径

该模型拟合单个分布，其中的位置参数随变换时间线性变化，但参数值在时间 0 处为 0。基于“分布”和“变换”的选择，中位数曲线可能不是直线，可能未穿过原点。该模型拟合 0 截距和共同斜率，不论是否有 X 变量。

共同斜率

在该模型中，位置参数是变换时间的线性函数，X 各值的截距不同，但具有共同斜率。基于“分布”和“变换”的选择，该模型拟合可能显示为曲线。例如，选择“对数正态”分布将生成图 8.12 中的图。

图 8.12 使用对数正态分布的共同斜率模型



带截距的单个路径

在该模型中，位置参数是变换时间的线性函数，X 各值的截距和斜率均不同。

不带截距的单个路径

在该模型中，位置参数是变换时间的线性函数，X 各值的截距为 0，且斜率不同。

共同截距

在该模型中，位置参数是变换时间的线性函数，X 各值的截距相同，但斜率不同。

一阶动力学模型

提供四个一阶动力学模型，其中的位置参数是基于温度的 Arrhenius 变换的非线性函数。以上每个位置模型为可选解释变量 X 的每个值拟合不同的模型。

首次选择以上任意模型时，您必须指定温度的测量值尺度和参考温度值 X_0。指定的参考温度影响对这些模型中的 *b2* 的解释。*b2* 参数是 X_0 时的比率常数。X_0 的值用于构造时间加速因子 (Meeker and Escobar 1998)。若您随后选择另一个一阶动力学模型或 Arrhenius 速率模型（请参见第 198 页的“Arrhenius 速率”），该平台会记住并使用这些指定内容。

一阶动力学 1 型

在该模型中，*b1* 和 *b2* 为正。在线性尺度上，当时间趋向于无穷大时，曲线在 *b0* 处具有上渐近线。

一阶动力学 2 型

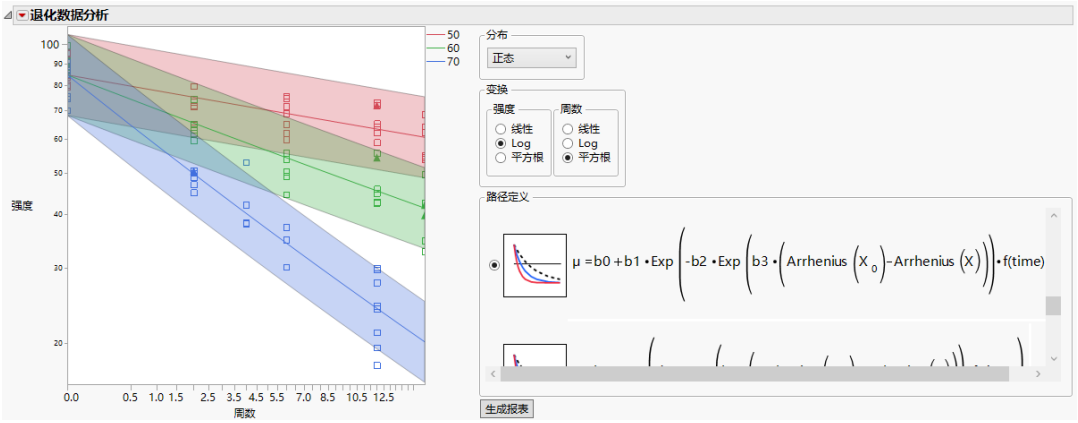
在该模型中，位置参数在时间 0 处为 0。*b0* 和 *b1* 为正。在线性尺度上，当时间趋向于无穷大时，曲线在 *b0* 处具有上渐近线。您可以认为 2 型模型是 1 型模型的垂直偏移版本。

一阶动力学 3 型

在该模型中，*b1* 和 *b2* 为正。由于 *b1* 前面的符号正好与 1 型模型中 *b1* 前面的符号相反，该模型是 1 型模型的反转版本。在线性尺度上，当时间趋向于无穷大时，曲线在 *b0* 处具有下渐近线。

鉴于数据随时间呈现负斜率，拟合模型可生成类似图 8.13 的图。该图针对 Adhesive Bond.jmp。选定的温度测量值尺度为“摄氏温度”，典型使用条件下的指定参考温度为 35 度。

图 8.13 一阶动力学 3 型模型的示例



一阶动力学 4 型

该模型是 3 型模型的垂直偏移版本。在线性尺度上，当时间趋向于无穷大时，曲线在 *0* 处具有下渐近线。

速率模型

提供三个模型，其中位置参数是变换后的 X 变量的指数函数。以上每一个位置模型为可选解释变量 X 的每个值拟合共同的截距和不同的斜率模型。对于以上每个模型，在线性尺度上，位置参数在变换后的时间值中呈线性关系。

Arrhenius 速率

该模型涉及 Arrhenius 变换构成的指数函数乘以变换时间。选择该模型后，您需要指定温度的测量值尺度，除非您已经提供了该信息。

多项式速率

该模型涉及 X 取对数后的线性函数构成的指数函数乘以变换时间。

指数速率

该模型涉及 X 的线性函数构成的指数函数乘以变换时间。

“破坏性退化”报表

“破坏性退化”报表为您拟合的每个模型包含一个分级显示项。在您拟合某个模型时，“模型列表”将使用该模型对应的行进行更新。

注意：所有模型都使用最大似然法拟合。

图 8.14 “模型列表”和“模型”分级显示项

模型列表

响应变换	分布	时间变换	路径定义	-对数似然	参数数目	实际观测数	AICc	BIC
Log	正态	平方根	μ = b0[x] + b1[x] * Sqrt(time)	287.41024	7	92	590.15381	606.473
Log	正态	平方根	μ = b0 - Exp(b1 + b2 * Arrhenius(X)) * Sqrt(time)	288.09559	4	92	584.65095	594.27834

Log(响应), 正态, μ = b0[x] + b1[x] * Sqrt(time)

Log(响应), 正态, μ = b0 - Exp(b1 + b2 * Arrhenius(X)) * Sqrt(time)

模型列表

“模型列表”中的前四列反映您在图选项中所做的选择。-对数似然、AICc 和 BIC 统计量是可用于模型比较的基于信息的测度。有关这些测度的说明，请参见《拟合线性模型》。

只要所比较的模型具有相同的“实际观测数”，“模型列表”中这前三个基于信息的测度在不同的模型之间就可进行比较。若情况并非如此，则需要谨慎操作，因为不同的模型可能使用不同的子集。“实际观测数”可能因选择的分布或变换而减少。选择对数-位置-尺度分布将排除所有非正的 Y 值。此外，“Log”和“平方根”变换会排除所有非正值。

“模型列表”表的每行都包含具有以下选项的红色小三角菜单：

滚动至 使报表窗口滚动至相应的模型分级显示项。

删除 从“模型列表”中删除模型并从报表中删除相应的模型分级显示项。

“模型”分级显示项

每个模型的分级显示项都包含具有以下选项的红色小三角菜单：

删除 从报表中删除模型分级显示项并从“模型列表”中删除模型。

每个模型的分级显示项都包含以下报表：

公式图形 显示位置参数的方程。

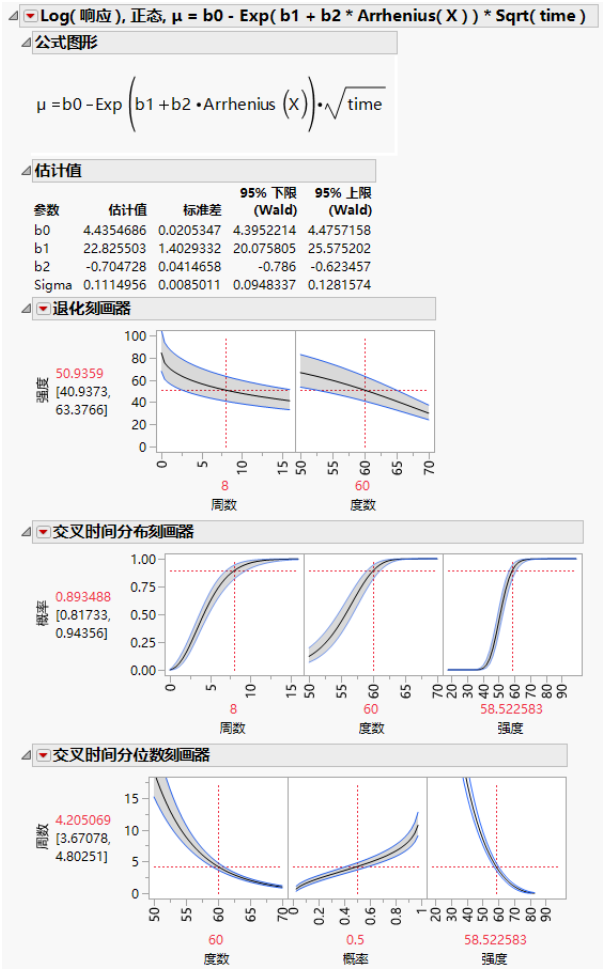
估计值 显示参数估计值及其标准差。该报表还包含基于 Wald 的 95% 参数置信区间。

注意：若指定参考温度 (X_0)，该值还显示在“估计值”报表的顶部。

分布、分位数和逆预测 有关刻画器的详细信息，请参见第 200 页的“刻画器”。

残差图 有关残差图的详细信息，请参见第 201 页的“残差图”。

图 8.15 “模型”分级显示项中的报表



刻画器

有三个刻画器显示在模型报表中。

退化刻画器

“退化刻画器”显示给定模型的“退化数据分析”图的刻画器视图。响应为退化响应 Y。该刻画器包括一个针对“时间”变量标绘的图和一个针对可选的解释变量 X（若您指定了该变量）标绘的图。针对“时间”标绘的图将 Y 的拟合分布的中位数显示为一条实曲线。虚曲线显示 Y 的拟合分布的 0.025 和 0.975 百分位数。

交叉时间分布刻画器

使用“交叉时间分布刻画器”可确定退化测量值在某个时间点降至给定阈值以下的概率。

该刻画器将响应 Y 的估计累积分布函数标绘为“时间”、可选的 X 变量以及 Y 的函数。针对“时间”和 Y 标绘的图显示 Wald 置信区间。

交叉时间分位数刻画器

使用“交叉时间分位数刻画器”可确定指定比例的测量值降至给定阈值以下的时间点。

该刻画器将估计的“时间”标绘为可选的 X 变量、Y 的分位数值（概率）以及 Y 的函数。针对“概率”和 Y 标绘的图显示 Wald 置信区间。

残差图

四个残差图显示在模型报表中。使用这些图来验证模型的分布假设。有关标准化残差的详细信息，请参见第 202 页的““破坏性退化”平台的统计详细信息”。

Cox-Snell 残差 P-P 图

若数据点大幅度地偏离对角线，则分布假设可能不成立。“Cox-Snell 残差 P-P 图”红色小三角菜单包含名为保存残差的选项，该选项支持您将残差数据保存到数据表中。有关 Cox-Snell 残差的说明，请参见 Meeker and Escobar (1998, sec. 17.6.1)。

标准化残差的概率图

若数据点大幅度地偏离对角线，则分布假设可能不成立。“标准化残差的概率图”红色小三角菜单包含名为保存残差的选项，该选项支持您将残差数据保存到数据表中。

标准化残差相对时间

使用“标准化残差相对时间”图来检查一段时间内变异性的不同。“标准化残差相对时间”红色小三角菜单包含名为保存残差的选项，该选项支持您将残差数据保存到数据表中。

标准化残差相对预测值

使用“标准化残差相对预测值”图来检查一组预测值范围内变异性的不同。“标准化残差相对预测值”红色小三角菜单包含名为保存残差的选项，该选项支持您将残差数据保存到数据表中。

“破坏性退化”平台选项

“退化数据分析”红色小三角菜单包含以下选项：

图形选项 提供用于修改平台图形的选项。

着色 启用或禁用着色。默认情况下，每个着色带针对给定时间值的上下限分别对应于 Y 分布的 0.025 和 0.975 分位数。

着色覆盖率 若启用了着色，您可以输入一个比例，用于增大或减小着色覆盖率。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

“破坏性退化”平台的统计详细信息

破坏性退化模型可表示如下：

$$g(Y) \sim F(\mu, \sigma)$$

$$\mu = h(f(Time), X)$$

其中：

- $g(Y)$ 是变换后的 Y 变量
- F 是选择的概率分布
- μ 是由 h 定义的位置参数
- h 是将变换后的“时间”变量与解释变量 X 关联的函数
- σ 是分布的尺度参数
- f(Time) 是变换后的“时间”变量
- X 是可选的解释变量

标准化残差通过以下方式获取：

- 对于位置-尺度分布，标准化残差为 $(y-\mu)/\sigma$。
- 对于对数-位置-尺度分布，标准化残差为 $(log(y)-\mu)/\sigma$。

第 9 章

可靠性预测

使用生产和失效数据预测产品失效

“可靠性预测”平台可帮助您预测将来失效的次数。JMP 使用生产日期、失效日期和产量估计寿命分布的参数。

使用交互式图形，您可以调整将来产量和合同期限之类的因子，以便估计将来失效。维修成本可并入分析，以便预测所有故障单元的维修总成本。

图 9.1 可靠性预测的示例

目录

“可靠性预测”平台概述 205
使用“可靠性预测”平台的示例 205
启动“可靠性预测”平台 208
“可靠性预测”报表 210
“观测数据”报表 210
“寿命分布”报表 211
“预测”报表 211
“可靠性预测”平台选项 215
使用“可靠性预测”平台的其他示例 215

“可靠性预测”平台概述

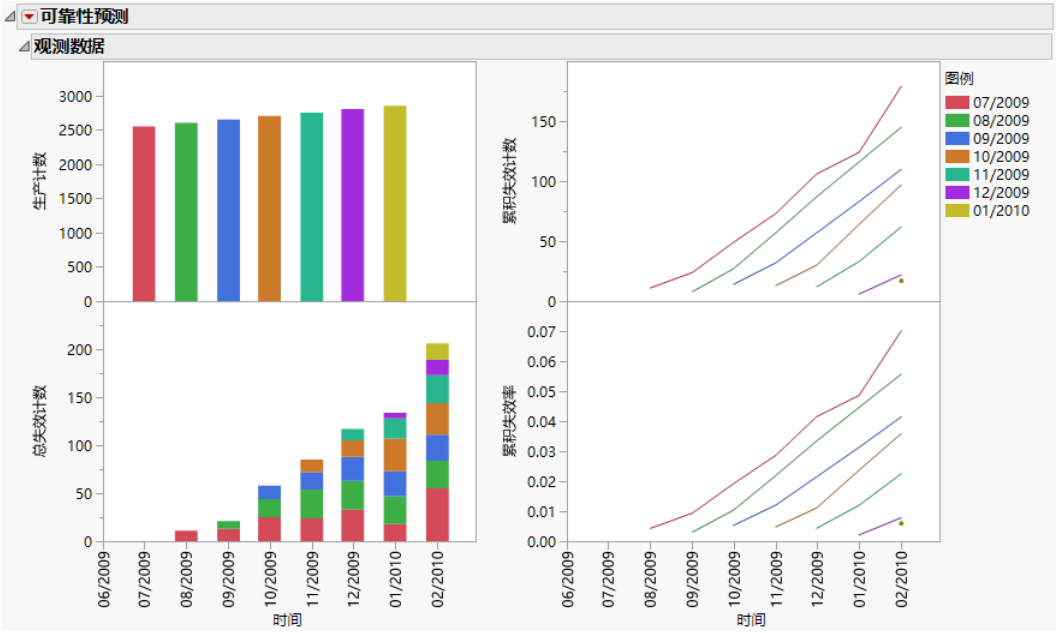
“可靠性预测”平台可帮助您预测将来失效的次数。JMP 使用生产日期、失效日期和产量估计寿命分布的参数。使用交互式图形，您可以调整将来产量和合同期限之类的因子，以便估计将来失效。维修成本可并入分析，以便预测所有故障单元的维修总成本。

使用“可靠性预测”平台的示例

您有七个月的生产和退回数据。您想要使用该信息预测截至 2011 年 2 月将要返修的总单元数。该产品的合同期限为 12 个月。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Small Production.jmp。
2. 选择**分析** > **可靠性和生存** > **可靠性预测**。
3. 在 Nevada **格式**选项卡上，选择**已售数量**并点击**生产计数**。
4. 选择**销售月份**，然后点击**时间戳**。
5. 选择其他列，然后点击**失效计数**。
6. 点击**确定**。

图 9.2 “观测数据”报表

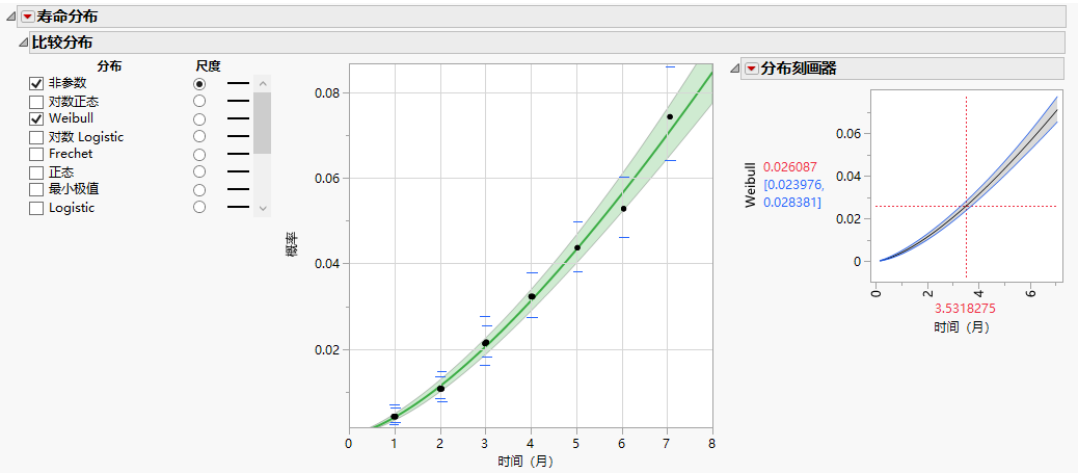


在“观测数据”报表中，左下角显示以前失效的条形图。累积失效显示在右侧的线图中。请注意，生产水平相当一致。随着时间的推移，生产的数量累积多了，更多的产品可能失效，因此累积失效率逐渐上升。一致的生产水平也导致逐月上升的类似的累积失效率和计数。

7. 点击**寿命分布**展开图标。

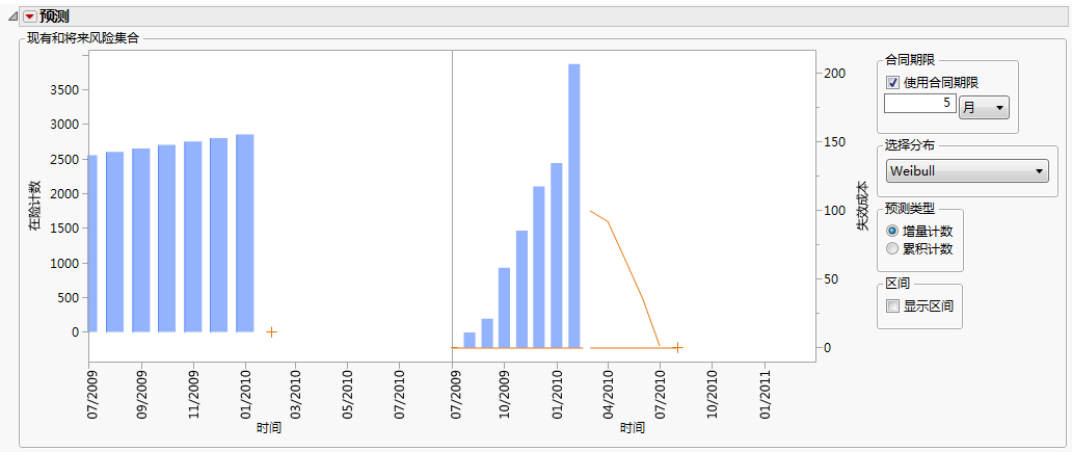
JMP 使用“寿命分布”平台拟合生产和失效数据的 Weibull 分布 （图 9.3）。JMP 随后使用拟合的 Weibull 分布来预测未来 5 个月的退回数量 （图 9.4）。

图 9.3 “寿命分布”报表



“预测”报表在左侧图上显示以前的生产数据（图 9.4）。在右侧图上，您看到以前的失效数随时间平稳增加。

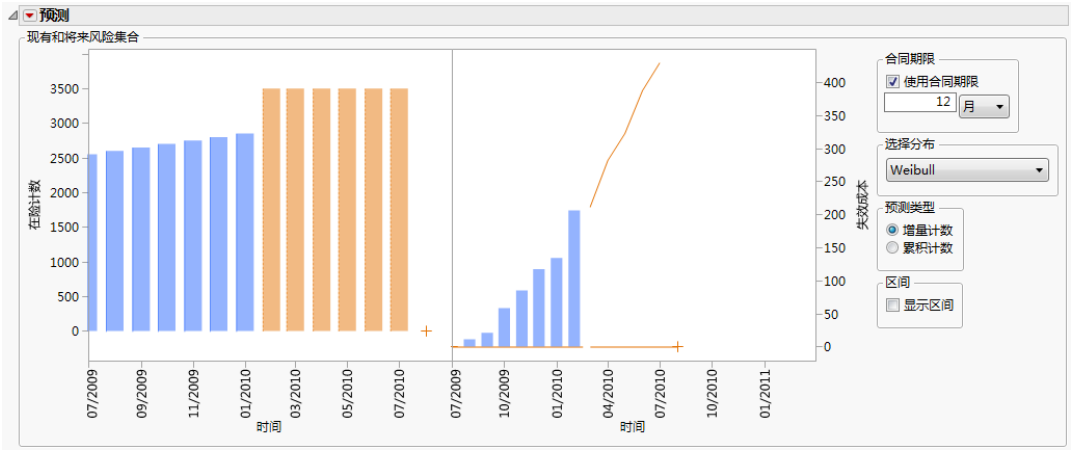
图 9.4 “预测”报表



8. 在“预测”报表中，为“合同期限”键入 12。

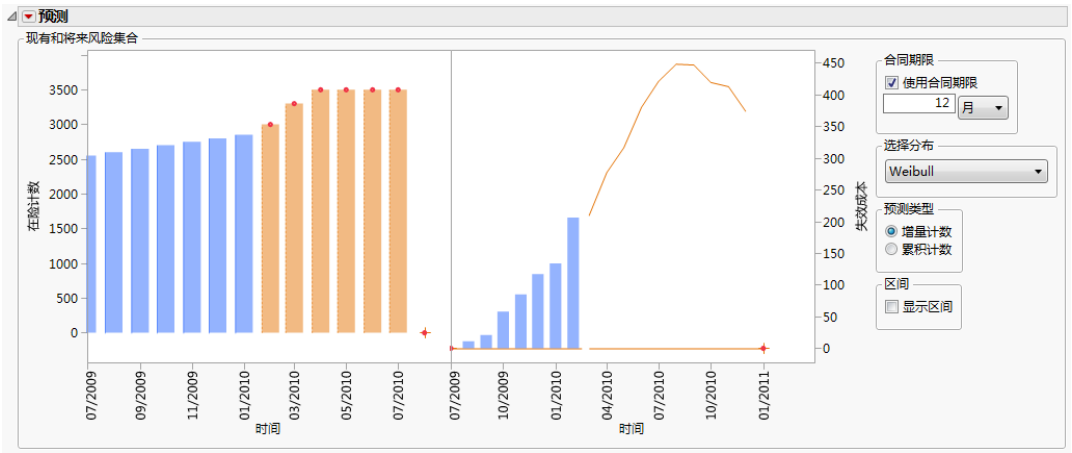
9. 在左侧的“预测”图中，将动画热区拖到 2010 年 7 月，然后将它向上拖到大约 3500 的位置。
   橙色直条显示在左侧图上来表示将来的生产。右侧图中每月退回数逐渐增加一直到 2010 年 8 月。

图 9.5 生产和失效估计值



10. 在左侧图中，将 2010 年 2 月热区拖到大约 3000 的位置，然后将 2010 年 3 月热区拖到大约 3300 的位置。
11. 在右侧图中，将右边的热区拖到 2011 年 2 月。
    JMP 估计退回数逐渐上升一直到 2010 年 8 月，然后逐渐下降直到 2011 年 2 月。

图 9.6 将来的生产计数和预测的失效数



启动“可靠性预测”平台

通过选择**分析** > **可靠性和生存** > **可靠性预测**来启动 “可靠性预测”平台。

图 9.7 “可靠性预测”启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

该启动窗口针对每种合同数据格式各包含一个选项卡：“Nevada”、“日期”和“事件时间”（针对失效时间数据）。以下小节介绍了这些格式。

Nevada 格式

合同数据通常以 Nevada 格式储存：装运或生产日期以及指定期间内的失效计数的形状类似内华达州。图 9.8 显示了 Small Production.jmp 样本数据表。

图 9.8 Nevada 格式的示例

	已售数量	销售月份	08/2009	09/2009	10/2009	11/2009	12/2009	01/2010	02/2010
1	2550	07/2009	11	13	25	24	33	18	55
2	2600	08/2009	0	8	19	30	30	29	29
3	2650	09/2009	0	0	14	18	25	26	27
4	2700	10/2009	0	0	0	13	17	34	33
5	2750	11/2009	0	0	0	0	12	21	29
6	2800	12/2009	0	0	0	0	0	6	16
7	2850	01/2010	0	0	0	0	0	0	17

“Nevada 格式”选项卡包含以下选项：

区间删失失效 考虑退回数量是区间删失数据。区间介于最后记录的时间和观测到失效的时间之间。默认选择该选项。

寿命时间单位 所有时间戳的实际日期 - 时间格式，包括退回计数的列标题。平台在预测步进增量时使用该设置。

生产计数 标识生产的单元数的列。

时间戳 包含生产日期的列。

失效计数 包含失效的单元数的列。

组 ID 观测按其分组的列。每个组具有各自的分布拟合和预测。还包括组合的预测。

日期格式

日期格式侧重于生产日期和失效日期。一个数据表指定每个时间期间的生产计数。另一个表提供失效日期、失效计数以及相应的失效产品生产时间。

图 9.9 显示了 Small Production part1.jmp 和 Small Production part2.jmp 样本数据表。

图 9.9 日期格式的示例

生产数据

	已售数量	销售月份
1	2550	07/2009
2	2600	08/2009
3	2650	09/2009
4	2700	10/2009
5	2750	11/2009
6	2800	12/2009
7	2850	01/2010

失效数据

	返修月份	返修数量	销售月份
1	08/2009	11	07/2009
2	09/2009	13	07/2009
3	10/2009	25	07/2009
4	11/2009	24	07/2009
5	12/2009	33	07/2009
6	01/2010	18	07/2009
7	02/2010	55	07/2009

“日期格式”选项卡分为“生产数据”和“失效数据”两部分。

生产数据

选择表 选择包含单元数和生产日期的表。

失效数据

选择表 选择包含失效数据 （例如失效单元数、生产日期和失效日期）的表。

左删失 标识删失观测的列。

时间戳 将生产观测关联到失效观测的列，指示失效的单元来自哪个批次。

删失代码 “删失”列中指定右删失观测的值。选定某个 “删失”列后， JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中进行选择。您还可以在框中输入不同值。若 “删失”列包含 “值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

其他选项与“Nevada 格式”选项卡上的选项相同。请参见第 208 页的“Nevada 格式”。

事件时间格式

事件时间格式显示生产数据和失效数据。与 Nevada 格式和日期格式不同，“事件时间”数据不包括日期时间信息，如生产日期或失效日期。这种格式还支持在数据中包含任意删失模式。请参见第 213 页的“使用事件时间数据定义风险集合”。

图 9.10 显示 Small Production Time to Event.jmp 样本数据表。有关使用事件时间数据的示例，请参见第 215 页的“使用“可靠性预测”平台的其他示例”。

图 9.10 事件时间格式的示例

开始时间 结束时间 失效计数

	时间（月）	时间右侧	频数	时间戳
1	0	1.0184804928	11	07/2009
2	1.0184804928	2.0369609856	13	07/2009
3	2.0369609856	3.022587269	25	07/2009
4	3.022587269	4.0410677618	24	07/2009
5	4.0410677618	5.0266940452	33	07/2009
6	5.0266940452	6.045174538	18	07/2009
7	6.045174538	7.0636550308	55	07/2009
8	7.0636550308	•	2371	07/2009

“事件时间格式”选项卡包含以下选项：

预测开始时间 预测开始的时间。输入要放在水平轴上的第一个值。要启用预测，您必须为“寿命时间单位”选择 “数值”并将 “预测开始时间”设置为 0。

删失代码 删失观测的代码。仅当您分配**删失**变量时才可用。

其他选项与“Nevada 格式”选项卡上的选项相同。请参见第 208 页的“Nevada 格式”。

“可靠性预测”报表

“可靠性预测”报表包含“观测数据”报表、“寿命分布”报表和“预测”报表。使用这些报表可以查看当前数据，比较分布以找到合适的拟合，然后调整影响预测的因子。您还可以将预测保存至数据表以便用于将来的财务预测。有关报表的示例，请参见第 203 页的图 9.1 “可靠性预测的示例”。

“可靠性预测”红色小三角菜单包含按日期过滤观测数据并以另一格式保存数据的选项。请参见第 215 页的“ “可靠性预测”平台选项”。

“观测数据”报表

“观测数据”报表使您能快速查看 Nevada 数据和日期数据 （图 9.11）。

- 条形图显示指定生产期间的生产计数和失效计数。
- 线图按生产期间显示累积预测。

注意：“事件时间”数据不包括日期时间信息（如生产日期或失效日期），因此平台不为该格式创建“观测数据”报表。

“寿命分布”报表

通过“寿命分布”报表，您可以比较分布并使用刻画器来找到合适的拟合。在“预测”报表中选择一个分布时，“寿命分布”报表会相应进行更新。有关该报表的详细信息，请参见第 27 页的“寿命分布”一章。（“寿命分布”一章中所述的某些选项在“可靠性预测”平台的“寿命分布”报表中不可用。）

“预测”报表

“预测”报表提供帮助您预测失效数的交互式图形。通过拖动热区，您可以添加预期生产计数并查看它们是如何影响预测的。

调整风险集合

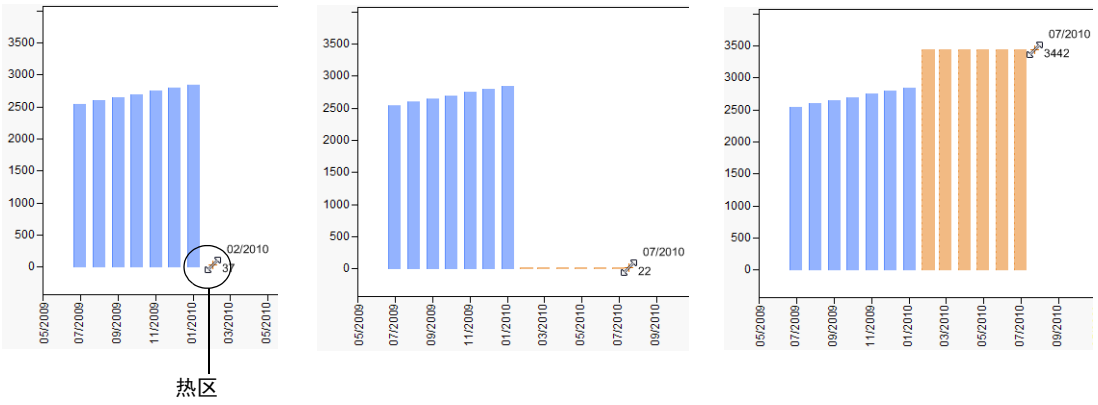
将来的生产

在左侧图中，蓝色直条表示以前的生产计数。要添加预期生产，请执行以下步骤：

1. 将热区拖到右侧以添加一个或多个生产期间。

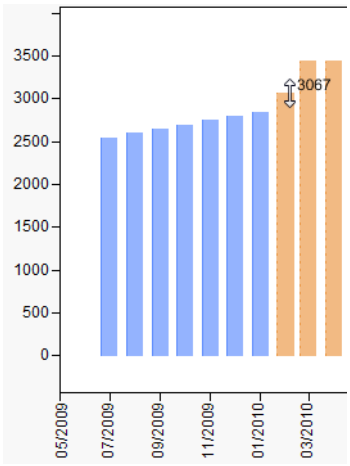
   橙色直条表示将来的生产。

图 9.11 添加生产期间



2. 向上或向下拖动每个直条以更改每个期间的生产计数。

图 9.12 调整生产计数



提示：要调整将来的生产计数，点击“预测”红色小三角并选择**以电子表格方式配置风险集合**。请参见第 214 页的“以电子表格方式配置风险集合”。

现有生产

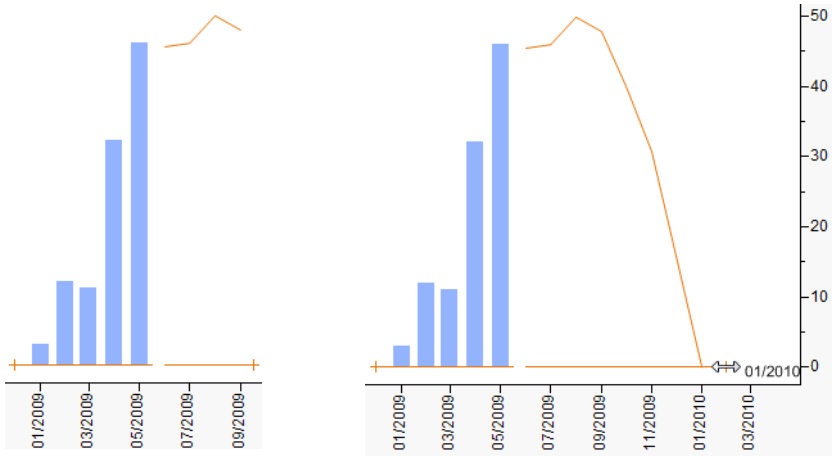
要从预测结果中删除某个风险集合，请右击蓝色直条并选择**排除**。要将该数据返回风险集合，右击并选择**包括**。

提示：要调整现有生产计数，点击“预测”红色小三角并选择**以电子表格方式配置风险集合**。请参见第 214 页的“以电子表格方式配置风险集合”。

预测失效

当您在左侧图中调整生产时，右侧图会相应更新来估计将来的失效数（图 9.13）。通过拖动热区可更改预测周期，随后橙色线条会缩短或伸长来显示估计的失效计数。

图 9.13 调整预测周期



使用事件时间数据定义风险集合

您可以使用“事件时间格式”选项卡获取任意风险集合的预测。在启动窗口的“事件时间格式”选项卡中，您必须为“寿命时间单位”选择“数值”并将“预测开始时间”设置为 0。按需输入“事件时间”、“删失”、“频数”和“组 ID”等列。

左图提供表示现有生产的直条的位置。将位于 0 的热区拖至左侧以创建现有风险集合。将位于 1 的热区拖至右侧以创建将来的生产风险集合。

您可以使用“以电子表格方式配置风险集合”选项指定现有和将来的风险集合。对于事件时间数据，您必须为“未来风险”区域中的现有风险集合输入负时间值。请参见第 214 页的“以电子表格方式配置风险集合”。

有关使用事件时间数据的示例，请参见第 215 页的“使用“可靠性预测”平台的其他示例”。

“预测”图选项

要进一步探索您的数据，您可以更改合同期限、分布类型和其他选项。

- 要预测其他合同期限的失效数，请在**使用合同期限**旁边输入数字。根据需要更改时间单位。
- 要更改分布拟合，请从**选择分布**列表中选择一个分布。随后将在将来风险的图形上拟合分布。分布拟合将显示在“寿命分布”报表图中，并添加一个新的刻画器。更改“寿命分布”报表中的分布拟合不会更改“预测”图中的拟合。
- 若您更关注一段时间内的总失效数，请选择**累积计数**。否则，JMP 以增量方式显示失效数，这可以使趋势更易于识别。
- 要显示预期失效数的 95% 置信限，请选择**显示区间**。

“预测”报表选项

“预测”红色小三角菜单包含以下选项：

动画 控制热区的闪现。您也可以在现有风险集合中右击蓝色直条并选择或取消选择**动画**。

交互式配置风险集合 确定您是否可以在图中拖动热区。

以电子表格方式配置风险集合 允许您指定生产计数和期间（而非将它们添加到交互式图形中）。您也可以从分析中排除生产期间。

- 要从分析中删除现有时间期间，请在“现有风险”区域中突出显示该期间并点击鼠标，然后选择**排除**。或选择**包括**使该期间重新回到预测。
- 要编辑生产，请在相应的“将来风险”字段中双击，然后输入新值。
- 要将生产期间添加到预测，请在“将来风险”区域中右击，然后选择一个**附加**选项。（**附加一行**用于添加一行；**附加 N 行**允许您指定行数。）

随着您更改这些值，图形会相应更新。

注意：若使用 “事件时间格式”选项卡启动了该平台，则必须在 “将来风险”区域中使用负时间值输入现有风险集合的值。

导入将来风险集合 允许您从另一打开的数据表导入将来的生产数据。新的预测随后显示在将来风险图中。导入的数据必须具有时间戳列和数量列。

显示区间 显示或隐藏图形中的 95% 置信限。该选项的作用与选择图形旁边的**显示区间**的作用相同。

选择**显示区间**后，**预测区间类型**选项会出现在菜单中。选择以下期间类型之一：

插件区间 仅考虑给定固定分布下的预测误差。

预测区间 考虑使用估计误差估计分布时（例如，使用不固定的分布）的预测误差。

若您选择 “预测区间”，则**预测区间设置**选项会出现在菜单中。图形中最初显示近似区间。选择 Monte Carlo **样本大小**或**随机种子**来指定这些值。要使用系统时钟，请输入缺失数字。

使用合同期限 确定是否在预测中考虑指定的合同期限。该选项的作用与选择图形旁边的**使用合同期限**的作用相同。

使用失效成本 在将来风险图中显示失效成本而非失效计数。

选择**使用失效成本**后，**设置失效成本**选项会出现在菜单中。该选项支持您设置每个失效的成本。若在启动窗口中指定了 “分组”变量，则可使用 “设置失效成本”窗口为每组中的失效指定单独的成本。

保存预测数据表 在新数据表中保存累积退回数和增加的退回数，并保存您在启动窗口中选择的变量。对于分组的分析，表名称包括组 ID 和 “聚合”一词，现有退回数还包含在聚合的数据表中。

“可靠性预测”平台选项

“可靠性预测”红色小三角菜单包含以下选项：

按事件时间格式保存数据 在事件时间格式的表中保存 Nevada 数据或日期数据。

显示图例 显示或隐藏 “观测数据”报表的图例。对 “事件时间”数据不可用。

显示图形过滤器 显示或隐藏 “图形过滤器”，以便您可以选择要在 “观测数据”图中显示哪些生产期间。未选择的期间对应的直条变淡。取消选择这些期间可以按原始状态显示图形。对 “事件时间”数据不可用。

请参见《使用 JMP》获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

使用“可靠性预测”平台的其他示例

您有七个月的生产和退回数据，这些数据以事件时间格式存储。您想要使用该信息预测接下来 6 个月返修的总单元数。该产品的合同期限为 4 个月。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Small Production Time to Event.jmp。
2. 选择**分析 > 可靠性和生存 > 可靠性预测**。
3. 在**事件时间格式**选项卡上，选择**时间 （月）**和**时间右侧**。点击**事件时间**。
4. 选择**频数**并点击**频数**。
5. 点击**确定**。
6. 在 “预测”报表中，为 “合同期限”键入 4。
7. （可选。）点击**寿命分布**展开图标。

   “寿命分布”报表显示生产和失效数据的 Weibull 拟合。“可靠性预测”平台随后使用拟合的 Weibull 分布预测返修。

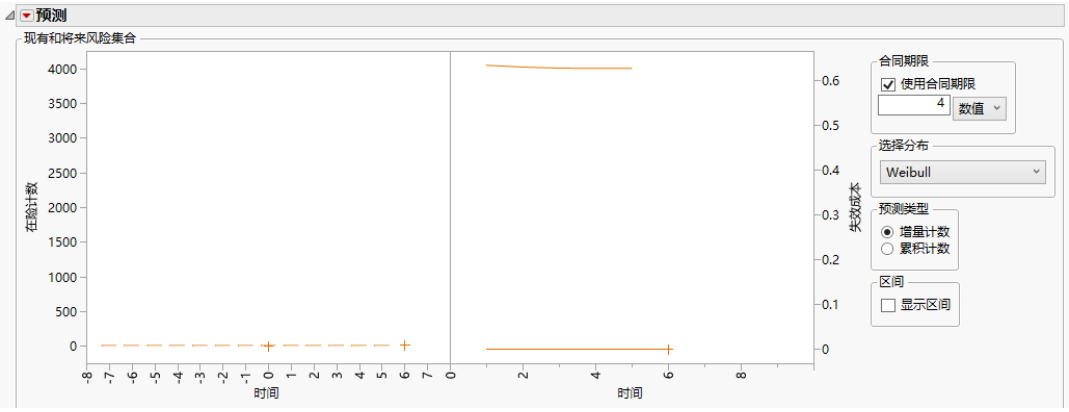
指定当前和将来的生产计数

8. 设置垂直轴设置以便为过去和将来的生产计数做准备。右击左侧 “预测”图中的垂直轴，然后选择**轴设置**。指定以下轴设置：
   1. 为 “最小值”键入 -250。
   2. 为 “最大值”键入 4250。

3. 为 “增量”键入 500。
4. 点击**确定**。

9. 设置水平轴以覆盖前 8 个月和后 8 个月。右击左侧 “预测”图中的水平轴，然后选择**轴设置**。指定以下轴设置：
   1. 为“最小值”键入 -8。
   2. 为 “最大值”键入 8。
   3. 点击**确定**。
10. 设置生产计数为过去 7 个月。在左侧 “预测”图中，将最左侧的动画热点向左拖至 -7。
11. 设置未来 5 个月的生产估计值。在左侧 “预测”图中，将最右侧的动画热点向右拖至 5。

图 9.14 拖动动画热点后的风险集合



12. 在 “预测”红色小三角菜单中，选择**以电子表格方式配置风险集合**。
13. 将 “将来风险”面板的底部向下拖，以便可以看到 “将来风险”表中的全部 13 行。

注意：当使用事件时间格式的数据时，必须使用负的数值型时间值在 “将来风险”表中指定现有生产数量。

14. 键入 “计数”值，如图 9.15 中所示。

图 9.15 将来风险计数规格

将来风险

	时间	计数
1	-7	2550
2	-6	2600
3	-5	2650
4	-4	2700
5	-3	2750
6	-2	2800
7	-1	2850
8	0	3022
9	1	3307
10	2	3502
11	3	3502
12	4	3502
13	5	3502

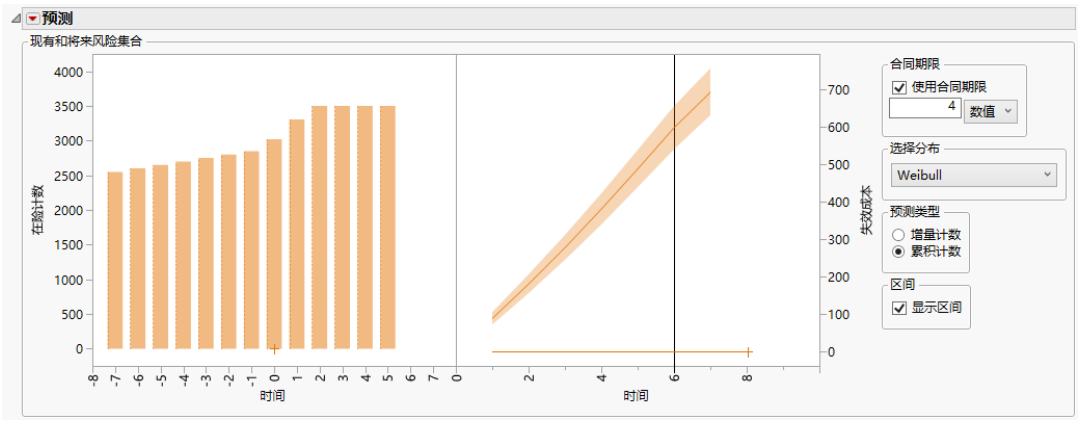
可选

15. 在 “预测类型”面板中，选择**累积计数**。
16. 在 “区间”面板中，选择**显示区间**。
17. 右击右侧“预测”图中的水平轴，然后选择**轴设置**。使用以下步骤在月份 6 处添加一条参考线：
    1. 为“值”键入 6。
    2. 点击**添加**。
    3. 点击**确定**。

预计在未来 6 个月内将返修的单元总数约为 600，置信区间大约介于 550 至 650 之间。

提示：您可以使用“预测”红色小三角菜单中的**保存预测数据表**选项查看预测和区间的精确值。

图 9.16 预测结果



第 10 章

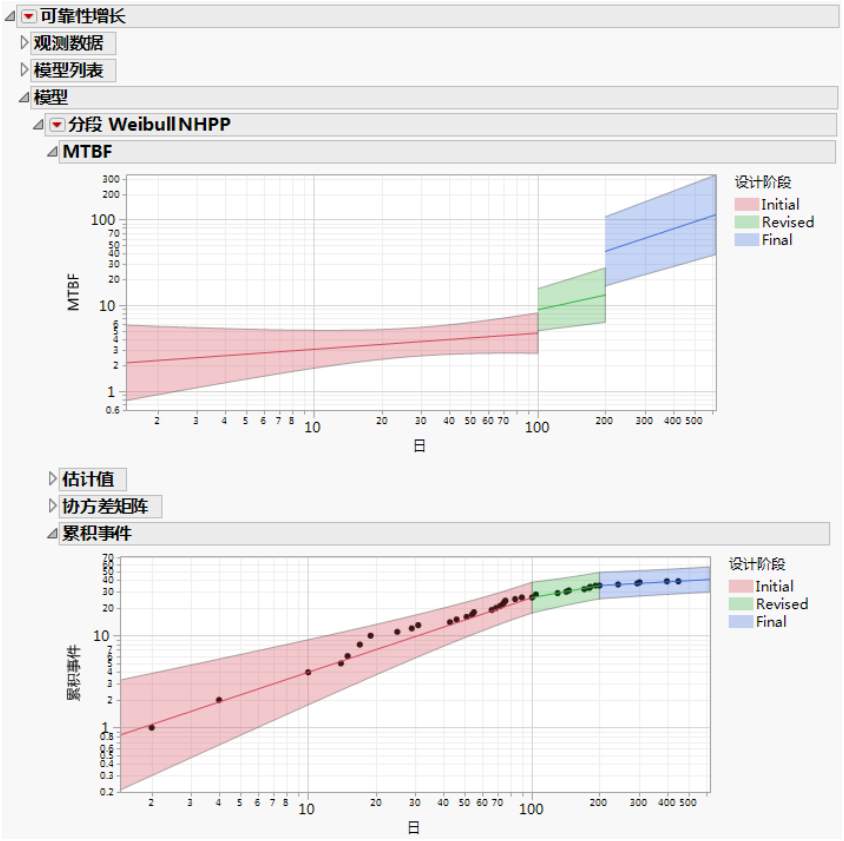
可靠性增长

随着变更的实现对系统可靠性建模

“可靠性增长”平台可随着单个可修复系统不断接受系统设计的改进而对系统可靠性的变化建模。可靠性增长检验计划尝试通过在发现失效时集成设计改进来增加系统的平均失效间隔时间 (MTBF)。

“可靠性增长”平台拟合 Crow-AMSAA 模型。这些是具有 Weibull 强度函数的非齐次 Poisson 过程。不同的模型可以包含可靠性增长计划的各个阶段。该平台还拟合多个系统的模型。

图 10.1 三阶段可靠性增长模型的图示例



目录

“可靠性增长”平台概述 221
使用“可靠性增长”平台的示例 222
启动“可靠性增长”平台 225
启动窗口角色 225
数据表结构 227
“可靠性增长”报表 229
模型描述和可行性 229
“观测数据”报表 229
“可靠性增长”平台选项 232
模型报表 233
Crow AMSAA 233
使用修改 MLE 的 Crow AMSAA 239
固定参数 Crow AMSAA 240
分段 Weibull NHPP 241
重新初始化 Weibull NHPP 245
分段 Weibull NHPP 变点检测 247
“可靠性增长”平台的更多示例 248
使用区间删失数据进行分段 NHPP Weibull 模型拟合 248
时间采用日期格式的“分段 Weibull NHPP 变点检测” 250
“可靠性增长”平台的统计详细信息 252
“Crow-AMSAA”报表的统计详细信息 252
“分段 Weibull NHPP 变点检测”报表的统计详细信息 253

“可靠性增长”平台概述

“可靠性增长”平台拟合 MIL-HDBK-189 (1981) 中所述的 Crow-AMSAA 模型。Crow-AMSAA 模型是具有 Weibull 强度的非齐次 Poisson 过程 (NHPP)；它也被称为幂次定律过程。这样的模型允许失效强度随时间变化。失效强度由 beta 和 lambda 这两个参数来定义。

对于单原型数据，该平台拟合四类模型并执行自动变点检测。提供以下报表：

- 简单 Crow-AMSAA 模型，该模型的两个参数均使用最大似然估计。
- 使用修改 MLE 的 Crow-AMSAA，该模型中针对偏倚修正了 beta 的最大似然估计值。
- 固定参数 Crow-AMSAA 模型，该模型中允许用户固定一个或两个参数。
- 分段 Weibull NHPP 模型，该模型中为每个检验阶段估计参数（考虑以前阶段的失效历史记录）。
- 重新初始化 Weibull NHPP 模型，该模型中为每个检验阶段估计两个参数（忽略以前阶段的失效历史记录）。
- 自动估计变点和相关的分段 Weibull NHPP 模型，针对具有不同失效强度的可靠性增长情况可定义两个不同的检验阶段。

对于多原型数据，该平台拟合以下几类模型：

- 分段 Weibull NHPP 模型，其中多阶段研究中的每个系统遵循相同的分段 Weibull NHPP 模型。系统之间的差异被认定来自同一模型的个体实现的随机性。在该模型中，每个阶段都对应一个 beta 参数，该模型还包含一个 lambda 参数。
- 具有“不同截距”模型的分段 Weibull NHPP，其中多阶段研究中的每个系统遵循不同的分段 Weibull NHPP 模型。在该模型中，每个阶段都对应一个 beta 参数，每个系统都对应一个 lambda 参数。
- 独特阶段 Weibull NHPP 模型，其中多阶段研究中的每个系统在每个阶段中都遵循相同的 Crow-AMSAA 模型。在该模型中，每个阶段都对应一个 beta 参数和一个 lambda 参数。
- 独特 Weibull NHPP 模型，其中多阶段研究中的每个系统在每个阶段中都遵循不同的 Crow-AMSAA 模型。在该模型中，研究中的每一系统和阶段的组合都对应一个 beta 参数和一个 lambda 参数。
- 独特系统 Weibull NHPP 模型，其中研究中的每个系统都遵循具有不同参数的不同 Crow-AMSAA 模型。
- 相同系统 Weibull NHPP 模型，其中研究中的每个系统都遵循同一个 Crow-AMSAA 模型。系统之间的差异被认定来自同一模型的个体实现的随机性。

通过交互式刻画器，您可以探索 MTBF、失效强度和累积失效在一段时间内的变化情况。当您怀疑在检验期间强度发生了变化时，可以使用变点检测选项来估计变点及其相应的模型。

使用“可靠性增长”平台的示例

假定您要检验新型涡轮发动机的原型。检验计划历时一年多，分为三个阶段。

数据包含在 Reliability 子文件夹下的 TurbineEngineDesign1.jmp 样本数据表中。对于出现的每个失效，在列**日**中记录了距离检验开始的天数。在列**纠正次数**中记录了给定日的失效数或相等价的所需纠正次数。

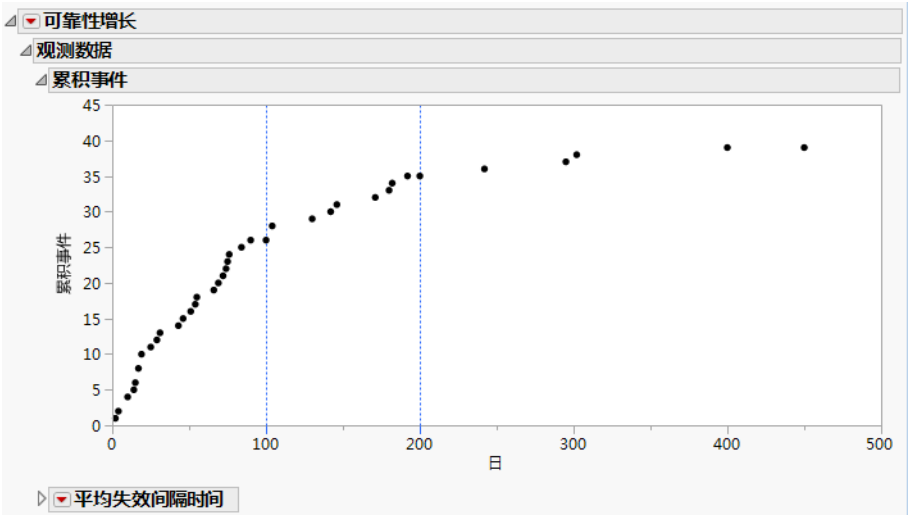
计划的前 100 天称为初始检验阶段。使用积极的设计变更来解决失效问题，这使得对设计进行了大量修改。接着是另一个 100 天的阶段，在此阶段，通过对子系统的设计变更来解决修改后设计中的失效问题。第三个阶段是最后检验阶段，它持续 250 天。在这个最后阶段，通过局部设计变更来解决失效问题。

根据指定的天数来终止每个检验阶段，因此阶段是定时截尾。具体而言，给定的阶段在下一阶段的开始时间处定时截尾。但是，失效时间是精确的 （未删失）。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/TurbineEngineDesign1.jmp。
2. 选择**分析 > 可靠性和生存 > 可靠性增长**。
3. 在**事件时间格式**选项卡上，选择**日**，然后点击**事件时间**。
4. 选择**纠正次数**并点击**事件计数**。
5. 选择**设计阶段**并点击**阶段**。
6. 点击**确定**。

   随即显示 “可靠性增长”报表。“累积事件”图显示按天划分的累积失效数。蓝色垂直虚线显示三个阶段之间的两个转换。

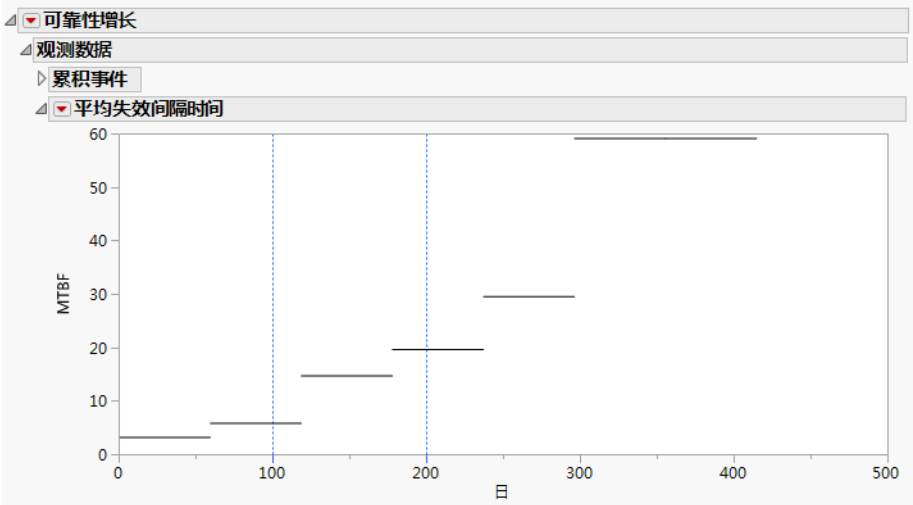
图 10.2 “观测数据”报表



7. 点击**平均失效间隔时间**展开图标。

   这将提供一个图，其中的水平线位于在预定大小的区间上计算的平均失效间隔时间处。使用红色小三角菜单中的选项可以指定区间大小。

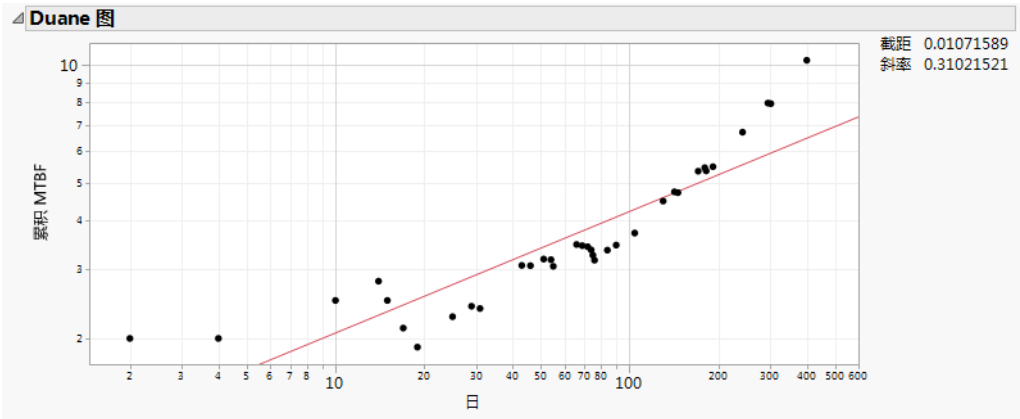
图 10.3 平均失效间隔时间图



8. 点击 Duane 图展开图标。

   这将提供一个图，它在垂直轴上显示累积 MTBF 估计值，在水平轴上显示事件时间变量。若数据符合 Crow-AMSAA 模型，则在重对数坐标纸上绘图时点应落在一条直线附近。

图 10.4 Duane 图



9. 点击 “可靠性增长”红色小三角并选择**拟合模型 > 分段 Weibull NHPP**。

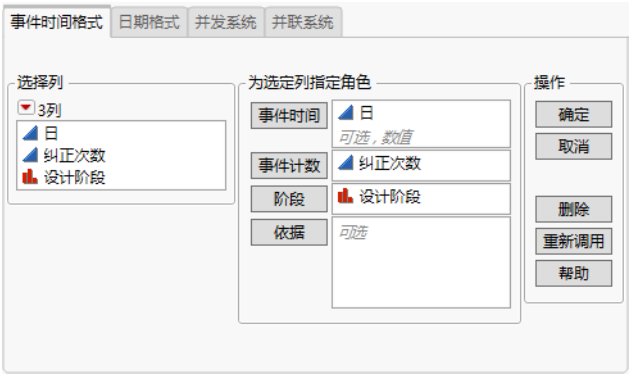
这将为检验计划的三个阶段拟合 Weibull NHPP 模型，将这些阶段视为单个可靠性增长计划的多个阶段 （图 10.5）。“分段 Weibull NHPP”红色小三角菜单中的选项可提供各种其他图和报表。

图 10.5 “分段 Weibull NHPP”报表

启动“可靠性增长”平台

通过选择**分析 > 可靠性和生存 > 可靠性增长**来启动 “可靠性增长” 平台。使用 TurbineEngineDesign1.jmp 中的数据的启动窗口如图 10.6 中所示。

图 10.6 “可靠性增长” 启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

该启动窗口为以下四种数据格式各包含一个选项卡：“事件时间格式”、“日期格式”、“并发系统”和 “并联系统”。

- “日期格式”假定使用日期 / 时间格式记录时间，用来指示一个绝对日期或时间。在“日期格式”选项卡上，为一个或多个时间列分配“时间戳”角色。
- 其他三种数据格式假定将时间记录为时间单位数，例如自系统最初启动以来的天数或小时数。检验开始时间假定为时间 0。在“事件时间格式”选项卡上，为一个或多个时间列分配“事件时间”角色。

分析中不包括“事件时间”、“时间戳”、“事件计数”、“阶段”或“系统 ID”列中具有缺失数据的行。

注意：“可靠性增长”平台的所有数据格式都要求以非降序指定时间或时间间隔。

启动窗口角色

可用于“可靠性增长”启动窗口中的变量的角色由指定的数据格式选项卡确定。本节对这些角色进行了说明。

事件时间

“事件时间”是检验开始和发生事件（失效或检验终止）之间间隔的时间单位数。检验开始时间假定为时间 0。请注意，“事件时间”角色仅在“事件时间格式”选项卡上可用。

允许两个规则（请参见第 227 页的“精确失效时间与区间删失”。）：

- 可以输入单个列。在这种情况下，假定该列给出发生事件时精确的经过时间。
- 可以输入两列，给出区间开始时间和结束时间。若区间的开始时间和结束时间不同，假定“事件计数”列中给出的相应事件发生在该区间内的某个未知时间。我们称之为数据是区间删失的。若区间的开始时间和结束时间相同，则假定相应的事件发生在该特定时间点，因此时间是精确的（未删失）。

平台要求时间列以非递减的顺序排序。当提供了区间开始时间和结束时间的两列时，这些区间不能重叠（端点除外）。事件计数为零的区间可以忽略（这些事件计数严格位于某个阶段内），因为它们不影响似然函数。

时间戳

“时间戳”是一个绝对时间（例如，日期）。与“事件时间”一样，“时间戳”允许使用一列或两列来输入时间。请注意，“时间戳”角色仅在“日期格式”选项卡上可用。

对于输入为“时间戳”的时间，认定表的第一行给出检验开始时间：

- 输入单个列时，对应于检验开始的事件计数为 0 的时间戳应显示在第一行。
- 输入给出时间区间的两列时，第一列中的第一个条目应是检验开始时间戳。（请参见第 226 页的“阶段”。）

其他详细信息与第 225 页的“事件时间”一节中关于“事件时间格式”的那些信息类似。另见第 227 页的“精确失效时间与区间删失”。

事件计数

“事件计数”是在指定时间或指定时间区间内发生的事件数，通常为被纠正措施（纠正次数）解决的失效数。若没有将任何列作为“事件计数”输入，则假定每行的“事件计数”为 1。

系统 ID

“系统 ID”标识多原型数据的观测的原型。“系统 ID”角色仅在“并发系统”和“并联系统”数据格式中可用。它是这两种数据格式所要求的角色。

阶段

可靠性增长计划通常涉及积极检验的几个期间或阶段。可以在可选的“阶段”列中指定这些检验阶段。对于“事件时间格式”和“日期格式”数据格式，“阶段”变量可以是任意数据或建模类型。对于“并联系统”数据格式，“阶段”变量的数据类型必须是“数值”。有关构造多阶段数据的详细信息，请参见第 227 页的“检验阶段”。有关示例，请参见第 248 页的“使用区间删失数据进行分段 NHPP Weibull 模型拟合”。

依据

这将为列中显示的每个值生成单独的分析。

数据表结构

“事件时间格式”和“日期格式”允许您分别为为“事件时间”或“时间戳”输入单个列**或**两列。“并发系统”和“并联系统”支持您输入与“系统 ID”变量的每个水平对应的一列。Sample Data 文件夹的 Reliability 文件夹中包含多原型数据表的示例：Concurrent Systems.jmp 用于“并发系统”，前缀为 Parallel Systems 的四个表用于“并联系统”。

本节将介绍如何使用这两种方法来指定检验结构。

精确失效时间与区间删失

在某些检验情况下，所检验系统会定期接受失效检查。在这种情况下，已知在时间区间内发生失效，但是不知道精确的失效时间。我们称失效时间是**区间删失的**。

“可靠性增长”平台接受精确、未删失的失效时间数据以及区间删失的数据。将单个列作为“事件时间”或“时间戳”输入时，时间会被视为精确时间（未删失）。

输入两列时，平台会将它们视为定义时间区间的开始点和结束点。若区间的开始时间和结束时间不同，则在该区间内发生的失效的时间视为区间删失的。若结束点相同，则假定相应失效的时间是精确的并且等于该共同的时间值。因此，您可以使用两个时间列来表示精确失效时间和区间删失的失效时间。

特别是，**精确失效时间**可以用以下两种方式之一表示：时间用单个时间列给出，或用两个时间列给出具有相同端点的区间。

“可靠性增长”平台中的模型拟合依赖于似然函数。似然函数会考虑是否存在区间删失。因此，允许同时存在区间删失的失效时间和精确失效时间。

定数截尾和定时截尾

当达到特定失效数或某一时间段过去后，检验计划可以要求检验终止。例如，检验计划可以在发生 50 次失效后终止检验。另一计划可以在 6 个月过后终止检验。

若基于指定的失效次数终止检验，我们称该检验为**定数截尾**。若基于指定的时间区间终止检验，我们称该检验为**定时截尾**。“可靠性增长”平台中使用的似然函数反映检验阶段是定数截尾还是定时截尾。

检验阶段

可靠性增长检验通常涉及几个检验**阶段**。例如，正在开发的系统或检验计划在特定时间点可能经历大量的变更。数据表指示每个阶段的开始时间以及每个阶段是定数截尾还是定时截尾，如下所述。

单个检验阶段

有单个检验阶段时，平台会根据数据表最后一行中的时间和事件计数条目来推断检验是定数截尾还是定时截尾。

- 若最后一行包含具有非零事件计数的精确失效时间，则该检验被视为定数截尾。
- 若最后一行包含具有零事件计数的精确失效时间，则该检验被视为定时截尾。
- 若最后一行包含具有非零宽度的区间，则该检验被视为定时截尾并且终止时间等于该区间的右端点。

注意：要指示某个检验已定时截尾，请确保包含数据表中的最后一行来显示检验终止时间。若您要输入单个列作为“事件时间”或“时间戳”，则最后一行应显示零事件计数。若您要输入两列作为“事件时间”或“时间戳”，则最后区间的右端点应是检验终止时间。在这种情况下，若在最后区间中没有失效，您应输入零事件计数。

多个检验阶段

使用“事件时间格式”时，除第一个之外的任何阶段的开始时间应包含在时间列中。使用“日期格式”时，所有阶段的开始时间应包含在时间列中。若在阶段开始时间没有发生任何事件，则“事件计数”列中的相应条目应为零。对于用两列给定的时间，使用具有相同端点的区间和零事件计数反映阶段开始时间可能是必要的。

在多阶段检验情形下，平台会根据某阶段变更前的最后一行中的条目推断除最后一个阶段外的每个阶段是定数截尾还是定时截尾。假定阶段 A 结束且阶段 B 在时间 t_B 开始。在这种情况下，对应于阶段 B 的第一行包含时间 t_B 的条目。

- 若阶段 A 中最后一个失效的失效时间是精确的且该时间不同于 t_B，则阶段 A 被视为定时截尾。终止时间等于 t_B。
- 若阶段 A 中最后一个失效的失效时间是精确的且等于 t_B，则阶段 A 被视为定数截尾。
- 若阶段 A 中的最后一个失效是区间删失的，则阶段 A 被视为定时截尾且终止时间等于 t_B。

平台会根据数据表最后一行中的条目推断最后阶段是定数截尾还是定时截尾。

- 若最后一行包含具有非零事件计数的精确失效时间，则该检验被视为定数截尾。
- 若最后一行包含具有零事件计数的精确失效时间或具有非零宽度的区间，则该检验被视为定时截尾。在包含区间的情况下，终止时间被视为右端点。

“可靠性增长”报表

默认显示“观测数据”报表。若在启动窗口中指定了“并联系统”数据格式，默认还显示“模型描述和可行性”报表。

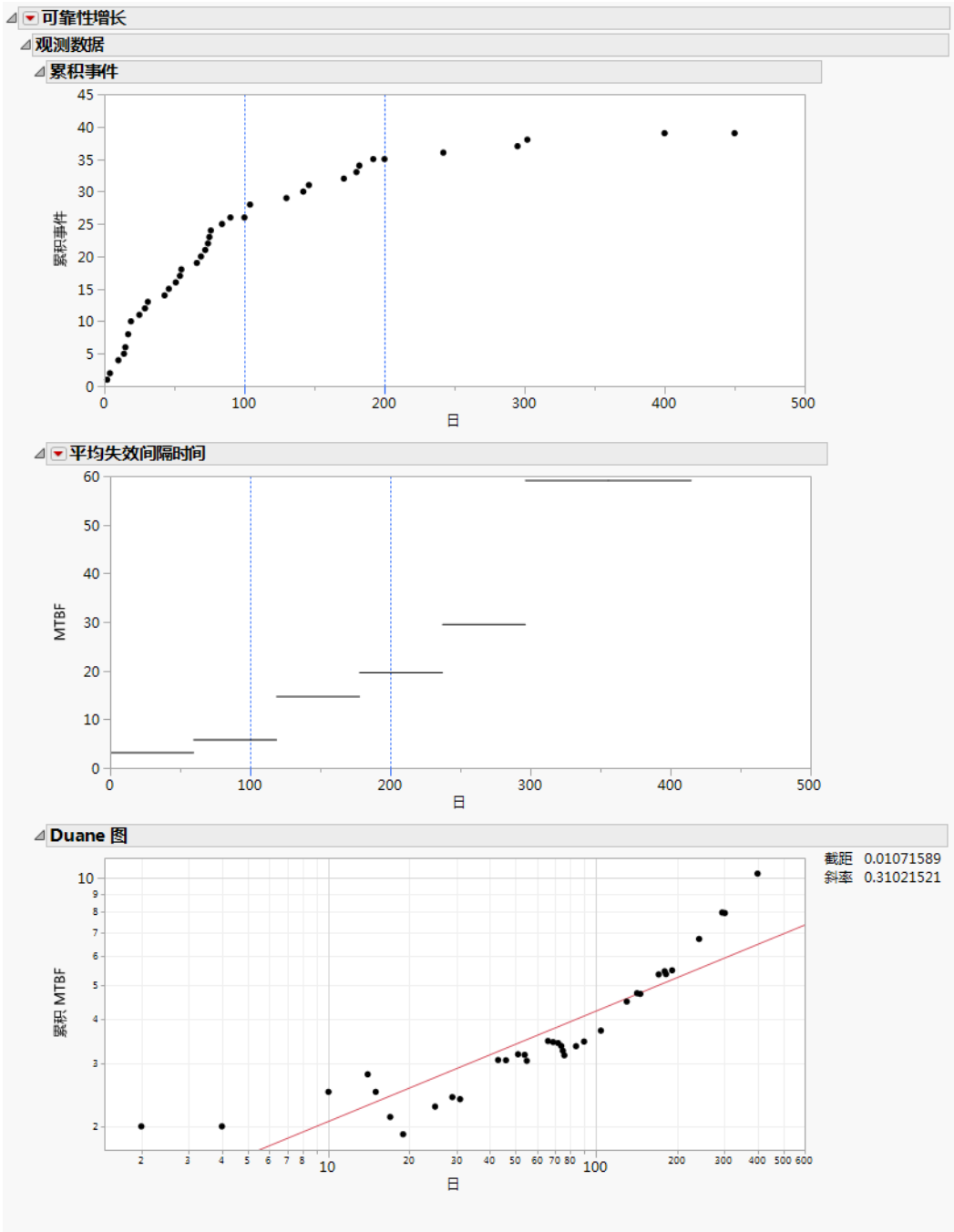
模型描述和可行性

“模型描述和可行性”报表列出并联系统模型的名称和描述。它还包含显示模型是否可用于当前数据的一列。若所列模型不可用，则在该表最右侧的列中报告原因。

“观测数据”报表

“观测数据”报表包含累积事件图、平均失效间隔时间图和 Duane 图。这些图如图 10.7 所示，在该图中打开了“平均失效间隔时间”报表和“Duane 图”报表。要生成该报表，请按第 222 页的“使用“可靠性增长”平台的示例”中的说明操作。

图 10.7 “观测数据”报表


可靠性增长
观测数据
累积事件
累积事件
日
平均失效间隔时间
MTBF
日
Duane 图
截距 0.01071589
斜率 0.31021521
累积 MTBF
日

累积事件图

该图显示事件如何随时间累积。“累积事件”图上每个点的垂直坐标等于截至该点的水平坐标给出的时间发生的总事件数。

只要拟合模型，就会在“累积事件”图上表示该模型。具体而言，模型估计的累积事件用一条曲线表示，用纯色置信带表示 95% 置信区间。图右侧的复选框可用于控制所显示的模型。

对于“并联系统”数据格式，“累积事件”图在图中为“系统 ID”变量的每个水平都提供单独的面板。它还在图底部包含一个面板，用来叠加“系统 ID”变量的所有水平的累积事件。“系统 ID”变量的各个水平用颜色来指定。

“平均失效间隔时间”图

“平均失效间隔时间”图显示在相等长度的小时间区间上获得的平均失效间隔时间。它们与阶段没有关系。长度相等的区间的默认数跟行数有关系。

“平均失效间隔时间”图选项

点击“平均失效间隔时间”红色小三角并选择“选项”，这将打开一个窗口，允许您指定要进行平均的区间。

提供两种平均类型：

- 等区间平均 MTBF（平均失效间隔时间），允许您指定共同的区间大小。
- 定制平均 MTBF，允许您指定时间区间的截止点。
  - 在表单元格内双击以更改其值。
  - 在表中右击以打开一个菜单，可以使用该菜单添加和删除行。

Duane 图

“Duane 图”显示针对“事件时间”变量标绘的累积 MTBF 估计值，两个轴均采用 log10 尺度。若数据符合 Crow-AMSAA 模型，则在重对数坐标纸上绘图时点应成一条直线。

注意：仅当失效时间是精确的且使用“事件时间格式”时，“Duane 图”才可用。该图不适用于区间删失数据或在“日期格式”下输入的数据。

图上显示的线是以 log10 尺度表示的累积 MTBF 对以 log10 尺度表示的“事件时间”变量做回归得到的最小二乘回归线。

注意：“Duane 图”不反映“阶段”变量。对于“事件时间”变量定义了阶段变更的一些行，在构造图和拟合回归线时会忽略它们。

截距和斜率

截距和斜率值显示在图的右侧。

- 因为历史原因给出的截距值为**自然**对数尺度下进行拟合所得到的截距。具体而言，在自然对数尺度下的“累积 MTBF”对自然对数尺度下的“事件时间”进行回归。截距值是该回归方程在 log(1) = 0 处预测的值，其中 log 是自然对数。要获取以 10 为底的对数尺度下拟合的截距，请将截距值除以 log(10)。请参见 Tobias and Trindade (2012, ch. 13)。
- 斜率值是自然尺度或对数尺度下拟合的斜率。它遵循对数的属性。

“可靠性增长”平台选项

“可靠性增长”红色小三角菜单包含以下选项：

拟合模型 若在启动窗口中指定了“事件时间格式”、“日期格式”或“并发系统”数据格式，则该菜单包含用于拟合各种非齐次 Poisson 过程 (NHPP) 模型的选项。下文对这些选项进行了详细说明。根据在启动窗口中所做的选择，可能的选项是：

- 第 233 页的“Crow AMSAA”
- 第 239 页的“使用修改 MLE 的 Crow AMSAA”
- 第 240 页的“固定参数 Crow AMSAA”
- 第 241 页的“分段 Weibull NHPP”
- 第 245 页的“重新初始化 Weibull NHPP”
- 第 247 页的“分段 Weibull NHPP 变点检测”

拟合并联系统模型 若在启动窗口中指定了“并联系统”数据格式，则该菜单包含用于拟合各种多原型数据模型的选项。该菜单中的可能选项取决于启动窗口中所做的选择。

请参见《使用 JMP》获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

模型列表

拟合模型后，将显示“模型列表”报表。该报表提供各种统计测度来描述模型拟合。当拟合其他模型时，它们会添加到“模型”列表，这提供了便利的模型比较汇总。模型按 AICc 值的升序方式排序。“模型列表”报表包含以下统计量：

参数数目 模型中的参数数目。

-2 对数似然 似然函数是在给定估计的模型参数时对观测数据可能性的度量。一般来讲，似然值越大，模型拟合效果越好。由此判定：-2 倍的对数似然 （-2 对数似然）的较小值指示模型拟合效果较好。

AICc 校正的 Akaike 信息准则。

BIC Bayes 信息准则。

有关 -2对数似然、AICc 和 BIC 的详细信息，请参见《拟合线性模型》。

模型报表

本节说明拟合可用模型时生成的报表以及这些模型报表中提供的选项。

- “Crow AMSAA”
- “使用修改 MLE 的 Crow AMSAA”
- “固定参数 Crow AMSAA”
- “分段 Weibull NHPP”
- “重新初始化 Weibull NHPP”
- “分段 Weibull NHPP 变点检测”

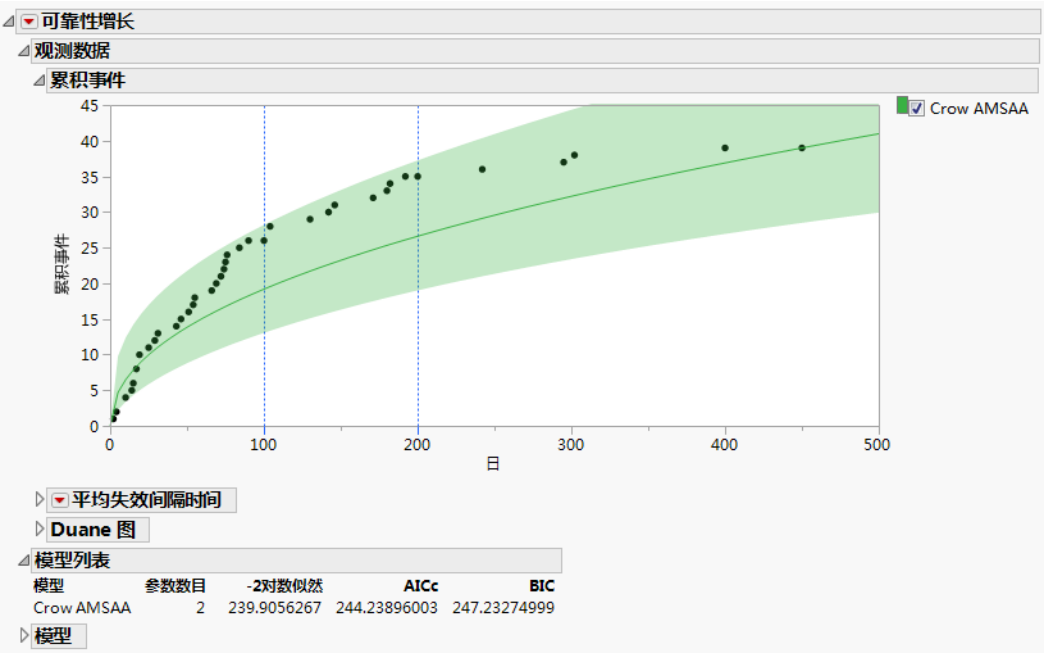
Crow AMSAA

该选项拟合 Crow-AMSAA 模型 (MIL-HDBK-189 1981)。Crow-AMSAA 模型是一个非齐次 Poisson 过程，其失效强度是时间 t 的函数，满足 $\rho(t) = \lambda\beta t^{b-1}$。其中，$\lambda$ 是尺度参数，β 是增长参数。该函数也称为 Weibull 强度，该过程本身也称为幂次定律过程（Rigdon and Basu 2000；Meeker and Escobar 1998）。请注意，“复发”平台拟合幂次非齐次 Poisson 过程。“幂次非齐次 Poisson 过程”等价于 Crow-AMSAA 模型，尽管它使用不同的参数化方法。请参见“复发分析”一章中第 144 页的“拟合模型”。

强度函数是适用于可修复系统的概念。它在时间 t 处的值是 t 附近的小区间中失效概率的限值除以该区间的长度；当区间长度趋近零时取该限值。您可以将强度函数视为测量系统在给定时间失效的可能性。若 $\beta < 1$，系统将随时间改进。若 $\beta > 1$，系统将随时间退化。若 $\beta = 1$，失效的发生率为常数。

选择 Crow AMSAA 选项时，“累积事件图”会相应更新以显示模型估计的累积事件曲线。对于每个时间点，该曲线周围的着色带定义了在该时间真实累积事件数的 95% 置信区间。“模型列表”报表也会相应更新。图 10.8 显示了 TurbineEngineDesign1.jmp 中数据的“观测数据”报表。

图 10.8 Crow AMSAA 的 “累积事件”图和 “模型列表”报表



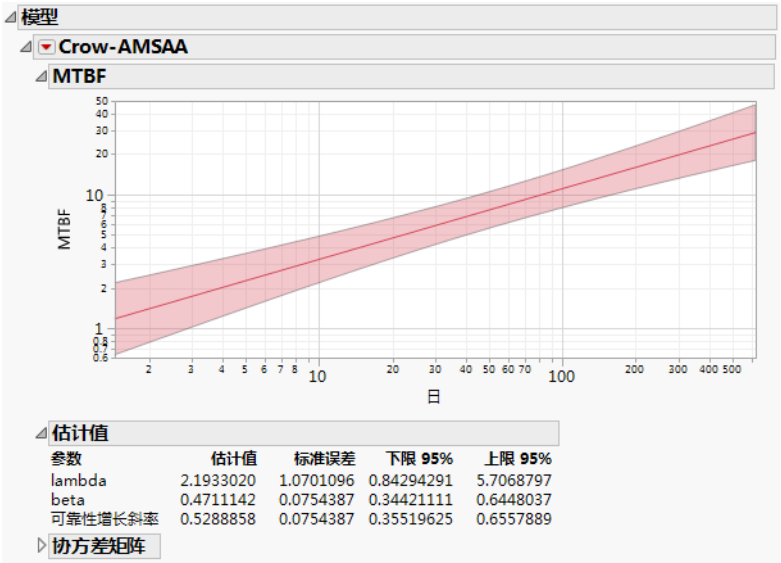
“Crow-AMSAA”报表

“Crow-AMSAA”报表在“模型”报表内打开。若使用“事件时间格式”，则“Crow-AMSAA”报表显示两个轴均采用对数尺度的 MTBF 图。请参见第 234 页的“MTBF 图”。

MTBF 图

默认显示平均失效间隔时间 (MTBF) 图（图 10.9）。对于每个时间点，MTBF 图周围的着色带定义了在时间 *t* 处的真实 MTBF 的 95% 置信区间。显示的图中的两个轴均采用对数尺度。在该尺度下，MTBF 图为线性的。

图 10.9 MTBF 图



要了解为何使用对数尺度时 MTBF 图是线性的，请考虑以下因素。平均失效间隔时间是强度函数的倒数。对于 Weibull 强度函数，MTBF 为 $1/(\lambda\beta t^{b-1})$，其中 t 表示自检验开始以来经历的时间。由此可见，MTBF 的对数是 $\log(t)$ 的线性函数，其斜率为 1 - β。通过用估计值替换参数 λ 和 β 来定义估计的 MTBF。因此估计的 MTBF 的对数是 $\log(t)$ 的线性函数。

估计值

lambda (λ)、beta (β) 和可靠性增长斜率 (1 - β) 的最大似然估计值显示在图下方的“估计值”报表中。（图 10.9）。此外还给出了 λ、β 和 1 - β 的标准误差和 95% 置信区间。有关计算的详细信息，请参见第 252 页的“Crow-AMSAA 模型的参数估计值”。

协方差矩阵

拟合模型参数估计值的协方差估计值矩阵。默认关闭该报表。

Crow-AMSAA 选项

本节说明拟合“Crow AMSAA”模型时在“Crow-AMSAA”红色小三角菜单中提供的选项。

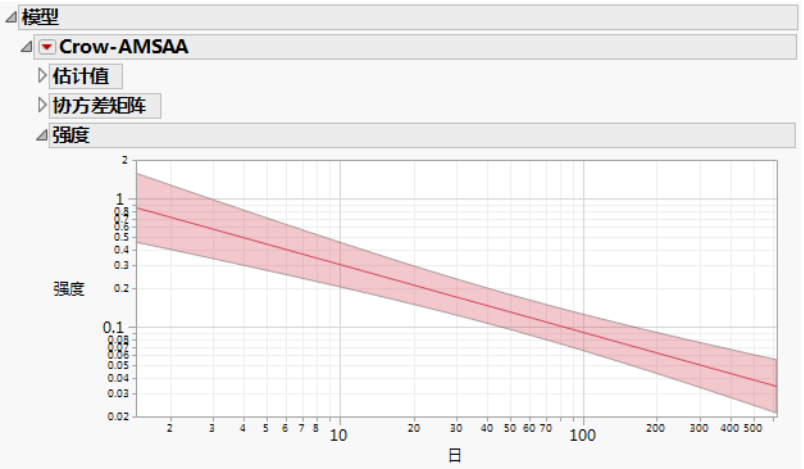
显示 MTBF 图

该选项显示或隐藏 MTBF 图。请参见第 234 页的“MTBF 图”。

显示强度图

该图显示估计的强度函数。Weibull 强度函数由 $\rho(t) = \lambda\beta t^{b-1}$ 给出，因此 log(强度) 是 $\log(t)$ 的线性函数。两个轴均采用对数尺度。

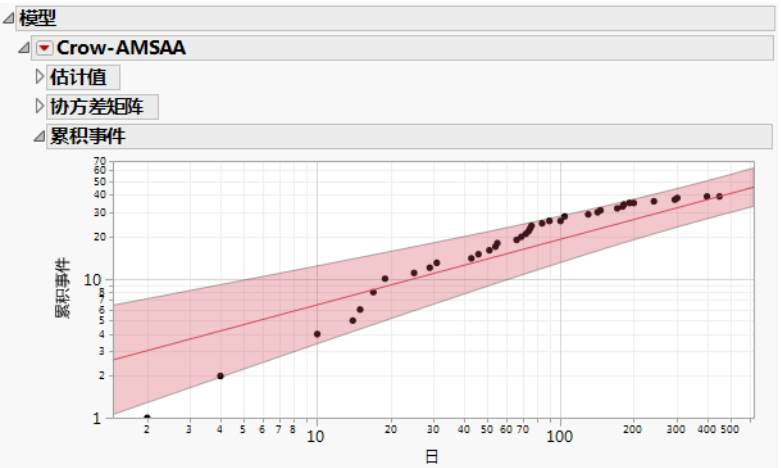
图 10.10 强度图



显示累积事件图

该图显示估计的累积事件数。观测的累积事件数也显示在该图上。两个轴均采用对数尺度。

图 10.11 累积事件图



对于 Crow-AMSAA 模型，时间 t 处的累积事件数由 λt^b 给出。由此可见，累积事件数的对数是 $\log(t)$ 的线性函数。因此，针对采用对数尺度的轴绘图时，估计的累积事件图是线性的。

显示刻画器

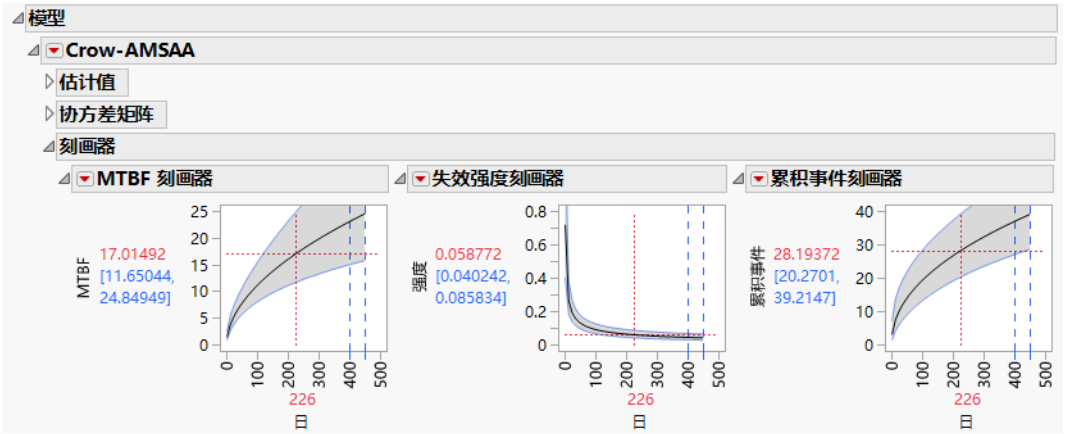
显示三个刻画器，分别显示估计的 MTBF、失效强度和累积事件。这些刻画器不使用对数尺度。通过在任何刻画器中拖动红色垂直虚线，您可以探索各个时间点的模型估计值；选定的时

间点值用红色显示在图下面。通过在图中点击时按 Ctrl 键，您还可以将时间轴设置为特定值。蓝色垂直虚线表示最后观测到失效的时间点。

刻画器还显示估计量的 95% 置信带。对于指定的时间设置，估计量（用红色表示）和 95% 置信限（用黑色表示）显示在刻画器的左侧。请参见第 253 页的“刻画器”。

请注意，您可以通过从任意刻画器的红色小三角菜单中选择**因子设置 > 关联刻画器**来关联这些刻画器。有关刻画器的用法和解释的详细信息，请参见《拟合线性模型》。另见《刻画器指南》。

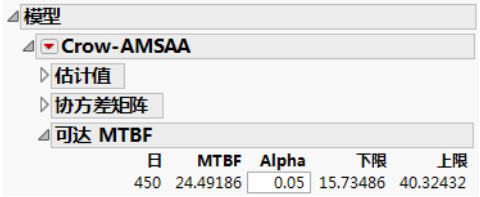
图 10.12 刻画器



可达 MTBF

检验结束时相应点的 MTBF 的置信区间通常很受关注。对于未删失的失效时间数据，该报表给出“可达 MTBF”的估计值及其 95% 置信区间。您可以通过输入 Alpha 的值指定 100*(1-α)% 置信区间。该报表如图 10.13 中所示。对于删失数据，仅报告检验终止时的估计 MTBF。

图 10.13 “可达 MTBF”报表



来自 NHPP 有无限多可能的失效时间序列；观测的数据仅表示其中之一。假定该检验在第 n 次失效时定数截尾。在“可达 MTBF”报表中计算的置信区间考虑到 n 个失效时间是随机的。若检验是定时截尾，则失效次数和失效时间都是随机的。正因为如此，“可达 MTBF”的置信区间不同于“MTBF 刻画器”在最后观测到的失效时间提供的置信区间。请参见 Crow (1982) 和 Lee and Lee (1978)。

检验是定数截尾时，“可达 MTBF”的置信区间是精确的。但是，当检验是定时截尾时，无法获取精确的区间。这时在区间包含“可达 MTBF”的概率至少为 1-α 的意义上，该限值是保守的。

拟合优度

“拟合优度”报表检验数据服从 Crow-AMSAA 模型这一原假设。根据输入的是一个还是两个时间列，执行 Cramér-von Mises 检验（请参见第 238 页的“具有未删失失效时间的数据的 Cramér-von Mises 检验”）或卡方检验（请参见第 238 页的“区间删失的失效时间的卡方拟合优度检验”）。

具有未删失失效时间的数据的 Cramér-von Mises 检验

在启动窗口中将数据输入为单个“事件时间”或“时间戳”列时，拟合优度检验是 Cramér-von Mises 检验。对于 Cramér-von Mises 检验，大的检验统计量值会导致拒绝原假设并得出模型拟合效果不好的结论。该检验使用 beta 的无偏估计值，在报表中会给出该值。检验统计量的值位于“Cramér-von Mises”标题下方。

“p 值”标题下方的条目指示若数据来自 Crow-AMSAA 模型，检验统计量与观测到的一样大的不可能性有多大。平台计算得到的 *p* 值最大为 0.25。若检验统计量小于对应于 *p* 值等于 0.25 的值，则该报表指示其 *p* 值 >=0.25。有关该检验的详细信息，请参见 Crow (1975)。

图 10.14 显示了对 TurbineEngineDesign1.jmp 中的数据进行 Crow-AMSAA 模型拟合的拟合优度检验。计算出的检验统计量对应于小于 0.01 的 *p* 值。我们可以得出结论：Crow-AMSAA 模型未能很好地拟合数据。

图 10.14 “拟合优度”报表 - Cramér-von Mises 检验

区间删失的失效时间的卡方拟合优度检验

在启动窗口中将数据输入为两个“事件时间”或“时间戳”列时，执行卡方拟合优度检验。卡方检验基于在定义的时间区间内比较观测的失效数和期望的失效数。大的检验统计量值导致拒绝原假设并得出模型未拟合的结论。

在“可靠性增长”平台中，卡方拟合优度检验专为区间删失数据设计，在数据表中指定的时间区间覆盖了该检验的整个时间期间。这意味着区间的开始时间是前一区间的结束时间。特别是，未发生失效的区间应包括在数据表中。若一些区间是不相邻的或一些区间具有相同的开始时间和结束时间，则算法会进行相应的调整。但是，生成的检验仅是近似正确的。

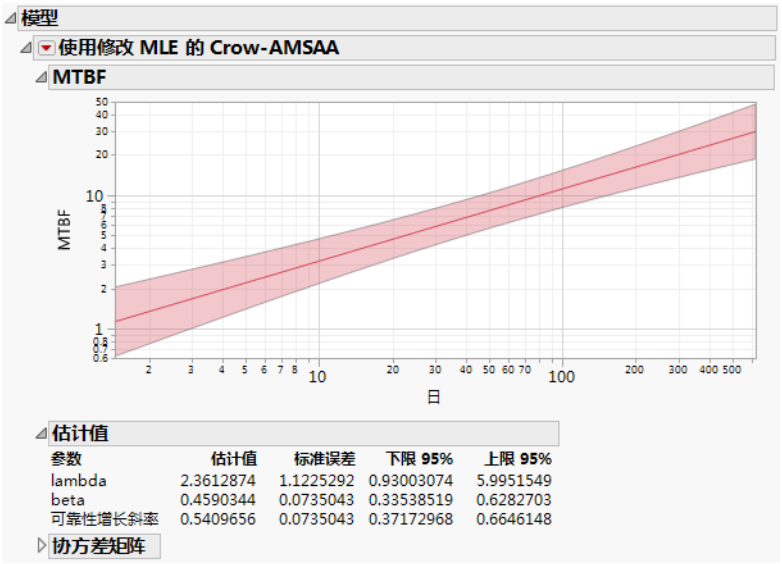
使用修改 MLE 的 Crow AMSAA

在 Crow-AMSAA 模型中，β 的最大似然估计值 (MLE) 有偏。该选项可拟合 β 经过偏倚调整的 Crow AMSAA 模型。

注意：仅当在启动窗口将数据输入为单个“事件时间”或“时间戳”列时该选项才可用。它不适用于区间删失数据。

图 10.15 显示了使用修改 MLE 的 Crow-AMSAA 拟合 TurbineEngineDesign1.jmp 中的数据。

图 10.15 “使用修改 MLE 的 Crow-AMSAA”报表



参数	估计值	标准误差	下限 95%	上限 95%
lambda	2.3612874	1.1225292	0.93003074	5.9951549
beta	0.4590344	0.0735043	0.33538519	0.6282703
可靠性增长斜率	0.5409656	0.0735043	0.37172968	0.6646148

针对 β 的修正偏倚的估计值的公式取决于该检验是定数截尾还是定时截尾。请参见第 252 页的“‘使用修改 MLE 的 Crow-AMSAA’的参数估计值”。

选择“使用修改 MLE 的 Crow-AMSAA”选项时，“累积事件”图会更新以显示该模型。“模型列表”随之更新。将打开“使用修改 MLE 的 Crow-AMSAA”报表以显示“使用修改 MLE 的 Crow-AMSAA”拟合的 MTBF 图、估计值和协方差矩阵；该图将在第 246 页的“MTBF 图”一节中进行介绍。

除了“显示 MTBF 图”外，可用的选项包括：“显示强度图”、“显示累积事件图”、“显示刻画器”、“可达 MTBF”和“拟合优度”。第 233 页的“Crow AMSAA”中介绍了这些报表。有关如何使用修改的 MLE 构造这些报表的详细信息，请参见第 252 页的“‘使用修改 MLE 的 Crow-AMSAA’的参数估计值”。 下面将详细说明针对修改的 MLE 选项的“拟合优度”和“可达 MTBF”报表。

拟合优度

因为仅当将数据输入为单个“事件时间”或“时间戳”列时“使用修改 MLE 的 Crow-AMSAA”选项才可用，因此“拟合优度”检验是 Cramér-von Mises 检验。因为在该检验中使用的 β 估计值已修正偏倚，因此该检验结果与 Crow-AMSAA 模型的拟合优度检验结果相同。

可达 MTBF

可达 MTBF 使用修改的 MLE 进行估计。但是，可达 MTBF 的置信区间使用真实 MLE 且与 Crow AMSAA 模型给出的区间相同。

固定参数 Crow AMSAA

该选项允许您指定 Crow-AMSAA 拟合的参数值。若在选择“固定参数 Crow-AMSAA”选项前未获得“Crow-AMSAA”报表，则同时提供“Crow-AMSAA”报表和“固定参数 Crow-AMSAA”报表。

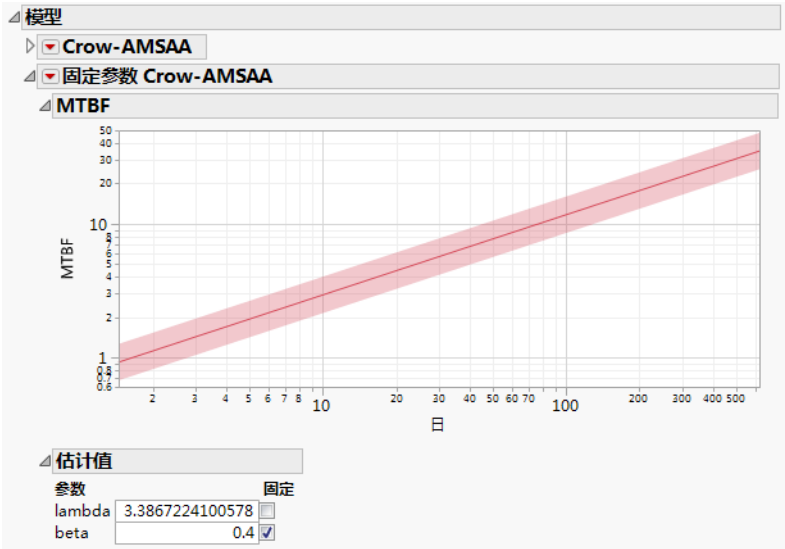
选择“固定参数 Crow-AMSAA”选项时，“累积事件”图会更新以显示该模型。“模型列表”随之更新。将打开“固定参数 Crow-AMSAA”报表以显示 Crow-AMSAA 拟合的 MTBF 图；该图将在第 246 页的“MTBF 图”一节中进行介绍。

除了“显示 MTBF 图”外，可用的选项包括：“显示强度图”、“显示累积事件图”和“显示刻画器”。有关这些图的构造和解释，请参见第 233 页的“Crow AMSAA”。

估计值

初始参数估计值是 Crow-ASMAA 拟合的 MLE。通过选中所需参数旁边的复选框并输入所需的值，可以固定其中一个参数。该模型将重新估计并且 MTBF 图会相应更新来描述该模型。图 10.16 显示了对 TurbineEngineDesign1.jmp 中的数据进行的固定参数 Crow-AMSAA 拟合，其中 beta 的值设置为 0.4。

图 10.16 “固定参数 Crow-AMSAA”报表

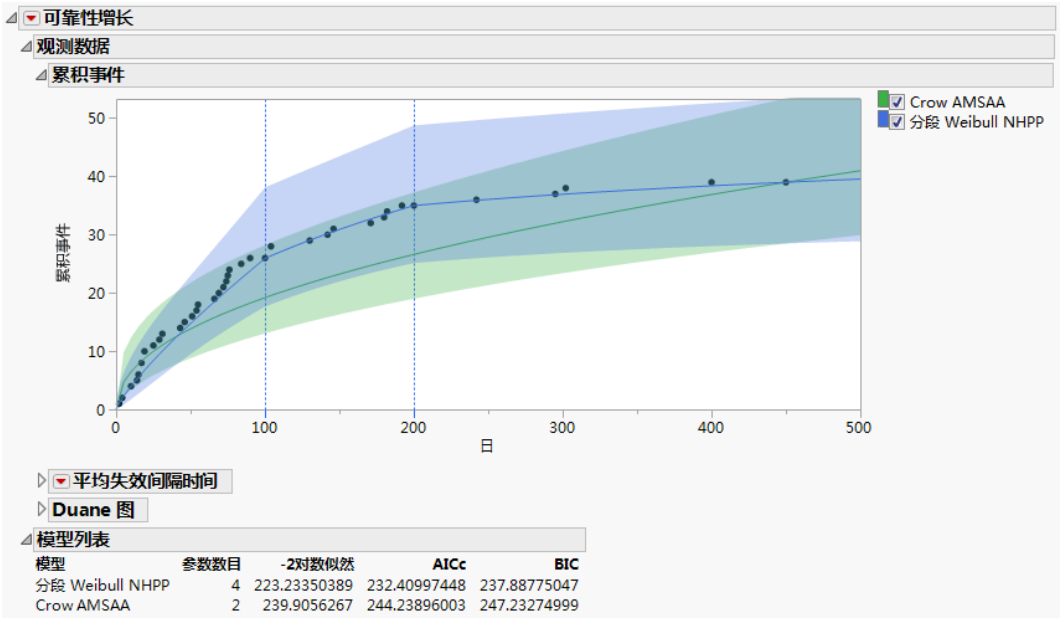


分段 Weibull NHPP

当在启动窗口中输入至少指定了两个值的“阶段”列时，可以拟合分段 Weibull NHPP 模型。在阶段开始时的累积事件数与前一阶段结束时的事件数相一致这一约束条件下，对每个阶段进行 Crow-AMSAA 模型拟合。为了正确显示阶段转换时间，除第一阶段外的每个阶段的第一行必须给出该阶段的开始时间。请参见第 228 页的“多个检验阶段”。

运行报表时，“累积事件”图会更新以显示分段模型。蓝色垂直虚线显示每个阶段的转换时间。“模型列表”随之更新。请参见图 10.17，其中已对 TurbineEngineDesign1.jmp 中的数据进行了 Crow-AMSAA 模型和分段 Weibull NHPP 模型拟合。请注意，在“模型列表”报表中会比较这两个模型。

图 10.17 “累积事件”图和“模型列表”报表

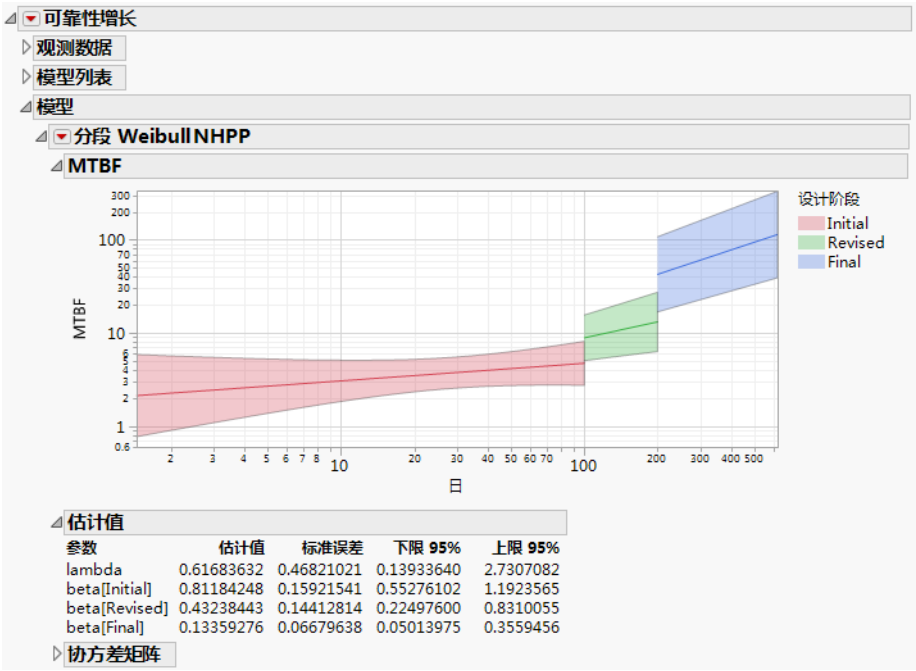


模型	参数数目	-2对数似然	AICc	BIC
分段 Weibull NHPP	4	223.23350389	232.40997448	237.88775047
Crow AMSAA	2	239.9056267	244.23896003	247.23274999

“分段 Weibull NHPP”报表

默认情况下，“分段 Weibull NHPP”报表显示估计的 MTBF 图。不同阶段用不同颜色区分开来。“估计值”报表和“协方差矩阵”报表显示在图的下方。

图 10.18 “分段 Weibull NHPP”报表



MTBF 图

选择“分段 Weibull NHPP”选项时，默认显示 MTBF 图和“估计值”报表（图 10.18）。使用“事件时间格式”时，轴的尺度为对数。有关该图的详细信息，请参见第 234 页的“MTBF 图”。

估计值

“估计值”报表给出模型参数的估计值。请注意，仅给出对应于第一个阶段的 λ 值的估计值。在分段模型中，一个阶段结束时的累积事件数必须与后续阶段开始时的累积事件数相一致。因为这些约束，第一个阶段的 λ 估计值和 β 的估计值决定其余的 λ。

用于计算估计值、它们的标准误差和置信限的方法类似于简单 Crow-AMSAA 模型所使用的方法。请参见第 252 页的“Crow-AMSAA 模型的参数估计值”。似然函数反映其他参数和有关累积事件数的约束。

协方差矩阵

拟合模型参数估计值的协方差估计值矩阵。默认关闭该报表。

分段 Weibull NHPP 选项

本节说明“分段 Weibull NHPP”红色小三角菜单中可用的选项。

显示 MTBF 图

该选项显示或隐藏 MTBF 图。请参见第 234 页的“MTBF 图”。

显示强度图

强度图显示在设计阶段上估计的强度函数和置信带。强度函数在阶段转换处通常是不连续的。颜色编码有助于区分不同阶段。若使用“事件时间格式”，轴的尺度为对数。请参见第 235 页的“显示强度图”。

显示累积事件图

累积事件图显示在设计阶段上估计的累积事件数以及置信带。该模型要求一个阶段结束时的累积事件数与后续阶段开始时的累积事件数相一致。颜色编码有助于区分不同阶段。若使用“事件时间格式”，轴的尺度为对数。请参见第 236 页的“显示累积事件图”。

显示刻画器

显示三个刻画器，分别显示估计的 MTBF、失效强度和累积事件。这些刻画器不使用对数尺度。有关解释和使用这些刻画器的详细信息，请参见第 236 页的“显示刻画器”一节。

请务必注意，由于刻画器图的默认分辨率，在“MTBF 刻画器”或“失效强度刻画器”中不能清晰显示不连续性。在阶段转换附近，刻画器迹显示几乎垂直但是略有倾斜的线条；该线条表示不连续性（图 10.19）。像这类在阶段转换处的线条不应该用于估计。获得更高分辨率的显示，使这些线条显得更垂直：

1. 在刻画器图中点击的同时按 Ctrl 键。
2. 为该窗口中的“标绘的点数”输入较大的值。（请参见图 10.20，在其中我们指定了 500 作为“标绘的点数”。）

图 10.19 刻画器

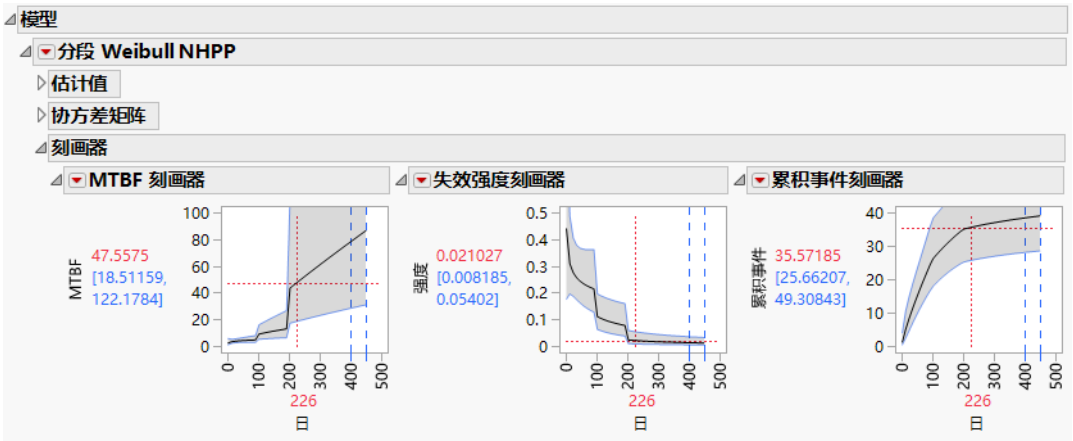


图 10.20 “因子设置”窗口

重新初始化 Weibull NHPP

该选项对每个检验阶段的数据进行独立的增长模型拟合。当在检验或工程方面影响增长率的因子在两个阶段间变化很大时，采用这种方式拟合模型会很有用。在这样的情况下，您可能要独立比较检验阶段。当在启动窗口中输入至少指定了两个阶段的“阶段”列时，可以使用“重新初始化 Weibull NHPP”选项。

对于拟合该模型的算法，包含新阶段第一次出现的每个行必须包含开始日期。

- 假定将单个列输入为“事件时间”或“时间戳”，则具有零事件计数的新阶段的开始时间必须显示在该阶段的第一行中。有关示例，请参见 Reliability 子文件夹中的样本数据表 ProductionEquipment.jmp。
- 若输入两列，则其左端点是该开始时间的区间必须显示在第一行中且包含相应的事件计数。有关示例，请参见 Reliability 子文件夹中的样本数据表 TurbineEngineDesign2.jmp。另请参见第 248 页的“使用区间删失数据进行分段 NHPP Weibull 模型拟合”。

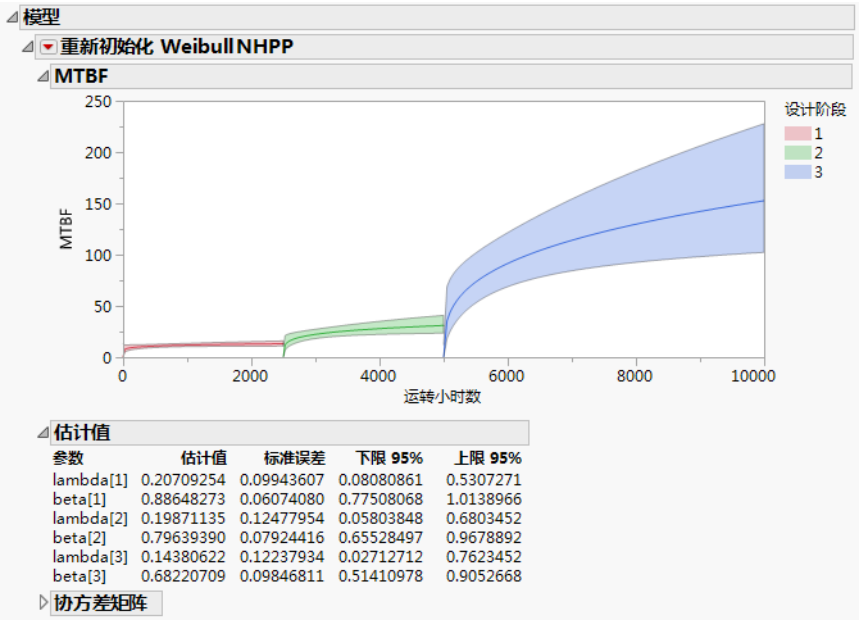
请参见第 228 页的“多个检验阶段”。

对每个阶段的数据进行独立的 Crow-AMSAA 模型拟合。运行该报表时，“累积事件”图会更新以显示重新初始化的模型。蓝色垂直虚线显示每个阶段的转换点。“模型列表”随之更新。

“重新初始化 Weibull NHPP”报表

默认情况下，“重新初始化 Weibull NHPP”报表显示估计的 MTBF 图。不同阶段用不同颜色区分开来。“估计值”报表和“协方差矩阵”报表显示在图的下方。（请参见图 10.21，它使用 Reliability 子文件夹中的 ProductionEquipment.jmp 样本数据文件。）

图 10.21 “重新初始化 Weibull NHPP”报表



MTBF 图

选择“重新初始化 Weibull NHPP”选项时，默认打开 MTBF 图。有关该图的详细信息，请参见第 234 页的“MTBF 图”。

估计值

“估计值”报表给出每个阶段的 λ 和 β 估计值。对于给定的阶段，仅使用该阶段的数据估计 λ 和 β。计算假定阶段从时间 0 开始并反映该阶段是定数截尾还是定时截尾，这由数据表结构定义。（请参见第 227 页的“检验阶段”。）此外还显示标准误差和 95% 置信限。这些值按第 252 页的“Crow-AMSAA 模型的参数估计值”中所述的那样计算。

协方差矩阵

拟合模型参数估计值的协方差估计值矩阵。默认关闭该报表。

重新初始化 Weibull NHPP 选项

本节说明 “重新初始化 Weibull NHPP”红色小三角菜单中可用的选项。

显示 MTBF 图

该选项显示或隐藏 MTBF 图。请参见第 234 页的“MTBF 图”。

显示强度图

强度图显示阶段的估计强度函数以及置信带。因为仅根据阶段内的数据计算强度函数，它们在阶段转换处是不连续的。颜色编码有助于区分不同阶段。请参见第 235 页的“显示强度图”。

显示累积事件图

“重新初始化 Weibull NHPP”模型的累积事件图按以下方式在设计阶段上描述估计的累积事件数和置信带。假定 *t* 表示检验的第一个阶段开始以来经历的时间。在时间 *t* 有效的阶段的模型在时间 *t* 处计算。特别是，有效阶段的模型的计算时间点不是自特定阶段开始以来经历的时间处，而是自检验的**第一个**阶段开始以来经历的时间处。

在阶段转换处，累积事件函数是不连续的。“累积事件”图将一个阶段开始时的估计累积事件数与前一阶段结束时的累积数相匹配。该匹配允许用户比较观测的累积事件函数与估计的累积事件函数。颜色编码有助于区分不同阶段。

显示刻画器

显示三个刻画器，分别显示估计的 MTBF、失效强度和累积事件。请注意，“累积事件刻画器”按“累积事件图”一节中所述的那样构造。特别是，一个阶段开始时的累积事件数与前一阶段结束时的事件数相一致。请参见第 244 页的“显示刻画器”。

分段 Weibull NHPP 变点检测

“分段 Weibull NHPP 变点检测”选项尝试找到一个可靠性模型发生变化的时间点。若您怀疑在检验期间可靠性增长有变化，则该选项会很有用。请注意，检测仅查找单个变点，它对应于两个潜在阶段。

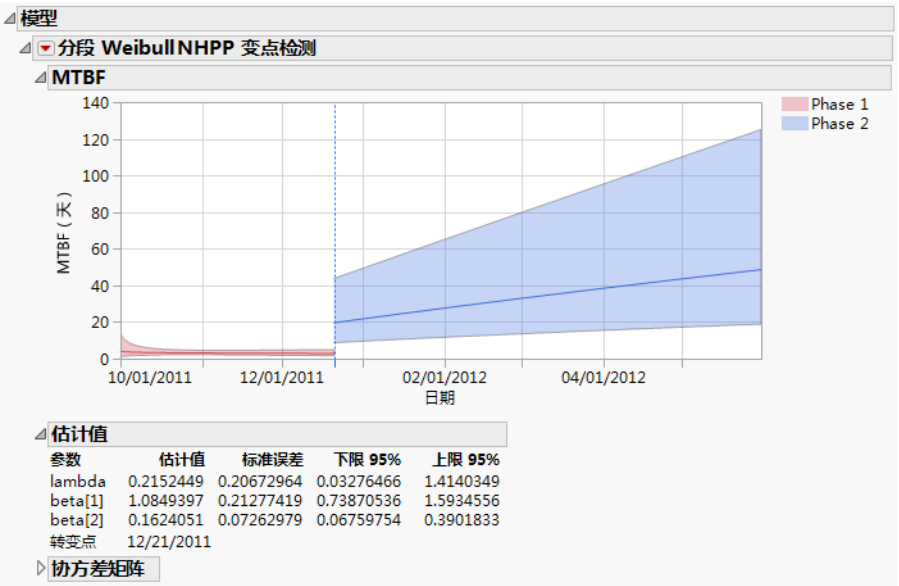
仅当在启动窗口中**未**输入“阶段”并且以下之一成立时，该选项才可用：

- 在启动窗口中将单个列输入为“事件时间”或“时间戳”（指示失效时间是精确的）。
- 在启动窗口的“并发系统”面板中将两列输入为“事件时间”。

选择“分段 Weibull NHPP 变点检测”选项时，估计的模型图和置信带会添加到“观测数据”下的“累积事件”报表中。“模型列表”会相应更新，给出在估计的变点条件下的统计量。在“模型”下方，提供了“分段 Weibull NHPP 变点检测”报表。

默认“分段 Weibull NHPP 变点检测”报表显示 MTBF 图、“估计值”和“协方差矩阵”。（请参见图 10.22，它使用 Reliability 子文件夹下的 BrakeReliability.jmp 中的数据。）请注意，显示在“估计值”报表底部的变点被估计为 12/21/2011。标准误差和置信区间将变点视为已知的。该图和“估计值”报表在第 241 页的“分段 Weibull NHPP”一节中进行介绍。

图 10.22 “分段 Weibull NHPP 变点检测”报表



可用选项包括：“显示 MTBF 图”、“显示强度图”、“显示累积事件图”、“显示刻画器”。这些选项在第 246 页的“重新初始化 Weibull NHPP 选项”一节中进行介绍。

估计变点所用的过程在第 253 页的“‘分段 Weibull NHPP 变点检测”报表的统计详细信息”一节中进行介绍。

“可靠性增长”平台的更多示例

- “使用区间删失数据进行分段 NHPP Weibull 模型拟合”
- “时间采用日期格式的‘分段 Weibull NHPP 变点检测’”

使用区间删失数据进行分段 NHPP Weibull 模型拟合

Reliability 子文件夹下的样本数据文件 TurbineEngineDesign2.jmp 包含有关在检验计划的三个阶段中涡轮发动机设计失效的数据。前两列给出发生失效的时间区间。这些区间记录为自检验以来的天数。精确失效时间未知，只知道失效发生在这些区间内。

通常以每周作为一个时间区间来提供失效数报表。没有发生失效并严格位于一个阶段内的区间不包括在数据表中（例如，表中未显示区间 106 到 112）。因为这些区间对似然函数没有影响，在估计模型参数时不需要它们。

但是，要拟合“分段 Weibull NHPP”或“重新初始化 Weibull NHPP”模型，请务必在“事件时间”或“时间戳”列中提供所有阶段的开始时间。

在本例中，三个阶段分别从第 0 天（“初始”阶段）、第 91 天（“修改”阶段）和第 200 天（“最后”阶段）开始。在“初始”阶段和“修改”阶段开始的几周内有发生失效。但是，在第 196 天到第 231 天没有报告失效。因此，开始天数和结束天数等于 200 的区间已包括在表中（第 23 行），它反映 0 次失效。这是必要的，这样 JMP 才能知道“最后”阶段的开始时间。

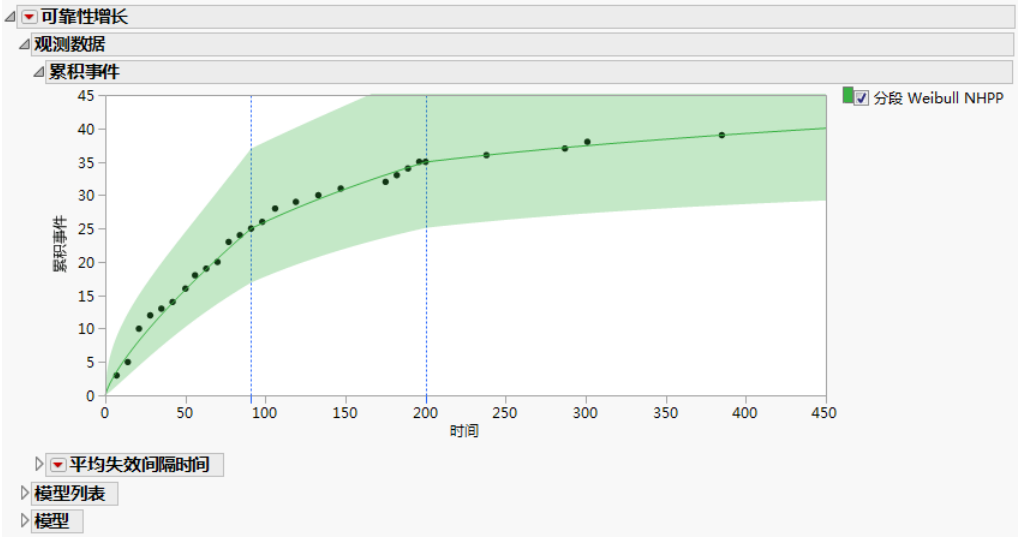
该检验在第 385 天终止。这是具有定时截尾阶段的区间删失失效时间的示例。

注意：阶段开始时间对于正确显示“分段 Weibull NHPP”模型的转换时间是必需的，对于估计“重新初始化 Weibull NHPP”模型是必需的。对于区间删失数据，算法定义阶段的开始时间为包含该阶段标明的第一次出现的行中记录的开始日期。在我们的示例中，若第 23 行不在表中，则“最后”阶段的开始时间会被视为 231。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/TurbineEngineDesign2.jmp。
2. 选择**分析 > 可靠性和生存 > 可靠性增长**。
3. 在**事件时间格式**选项卡上，选择列**区间开始**和**区间结束**，然后点击**事件时间**。
4. 选择**纠正次数**并点击**事件计数**。
5. 选择**设计阶段**并点击**阶段**。
6. 点击**确定**。
7. 点击 “可靠性增长”红色小三角并选择**拟合模型 > 分段 Weibull NHPP**。

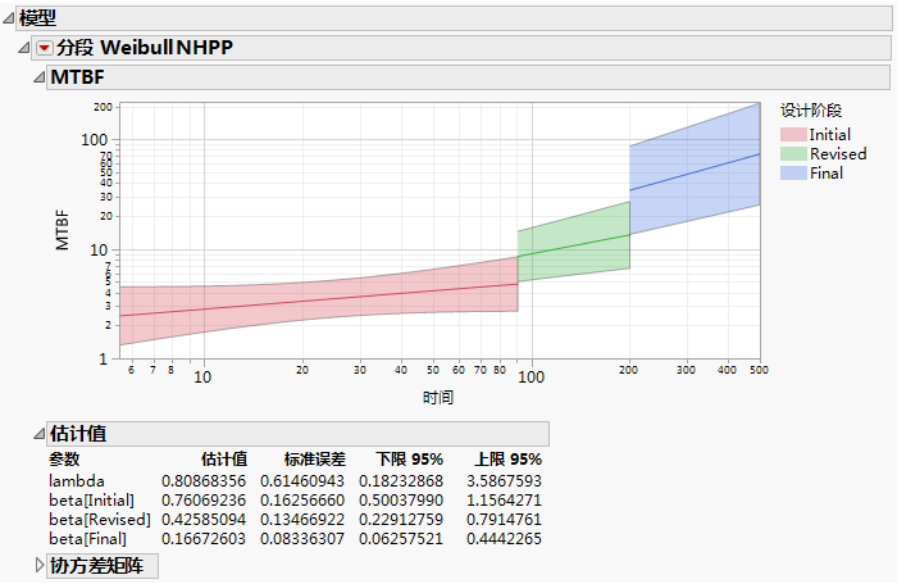
   “观测数据”报表中的 “累积事件图”如图 10.23 中所示。蓝色垂直虚线指示阶段转换点。列**设计阶段**中第一次出现“Revised”是在第 14 行。因此，“修改”阶段的开始时间取第 14 行中的**区间开始**值，即第 91 天。类似，列**设计阶段**中第一次出现 “Final”是第 23 行，因此 “最后”阶段的开始时间取第 23 行中的**区间开始**值，即第 200 天。

图 10.23 累积事件图



“分段 Weibull NHPP”报表位于“模型”分级显示项节点下方。在这里我们看到平均失效间隔时间在三个阶段上递增。从“估计值”报表中，我们看到 beta 估计值在三个检验阶段上递减。

图 10.24 MTBF 图



参数	估计值	标准误差	下限 95%	上限 95%
lambda	0.80868356	0.61460943	0.18232868	3.5867593
beta[Initial]	0.76069236	0.16256660	0.50037990	1.1564271
beta[Revised]	0.42585094	0.13466922	0.22912759	0.7914761
beta[Final]	0.16672603	0.08336307	0.06257521	0.4442265

时间采用日期格式的“分段 Weibull NHPP 变点检测”

Reliability 子文件夹下的文件 BrakeReliability.jmp 包含刹车系统的纠正次数数据。**日期**列给出第二列中指定的**纠正次数**实施的日期。对于该数据，失效时间是已知的。请注意，**日期**列必须采用升序顺序。

检验开始时间是**日期**列中的第一个条目 09/29/2011，对应的**纠正次数**值设置为 0。需要用它指示检验的开始时间。若这个第一行中有**纠正次数**的非零值，则相应日期将被视为检验开始时间。但是，在分析中**纠正次数**的值将被视为 0。

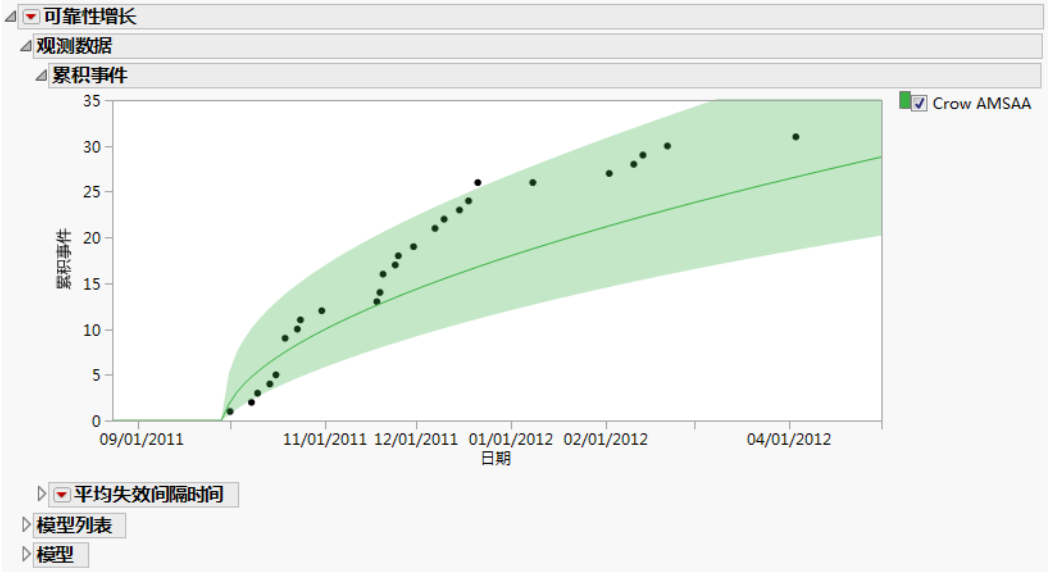
在最后一行给出检验终止时间为 05/31/2012。因为最后一行中的**纠正次数**值为 0，检验被视为在 5/31/2012 定时截尾。若这个最后一行中有**纠正次数**的非零值，则检验被视为定数截尾。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/BrakeReliability.jmp。
2. 选择**分析** > **可靠性和生存** > **可靠性增长**。
3. 选择**日期格式**选项卡。
4. 选择**日期**并点击**时间戳**。
5. 选择**纠正次数**并点击**事件计数**。
6. 点击**确定**。

7. 点击 “可靠性增长”红色小三角并选择**拟合模型 > Crow AMSAA**。

   “观测数据”报表中的 “累积事件图”会相应更新以显示模型。该模型看起来拟合数据的效果不好。

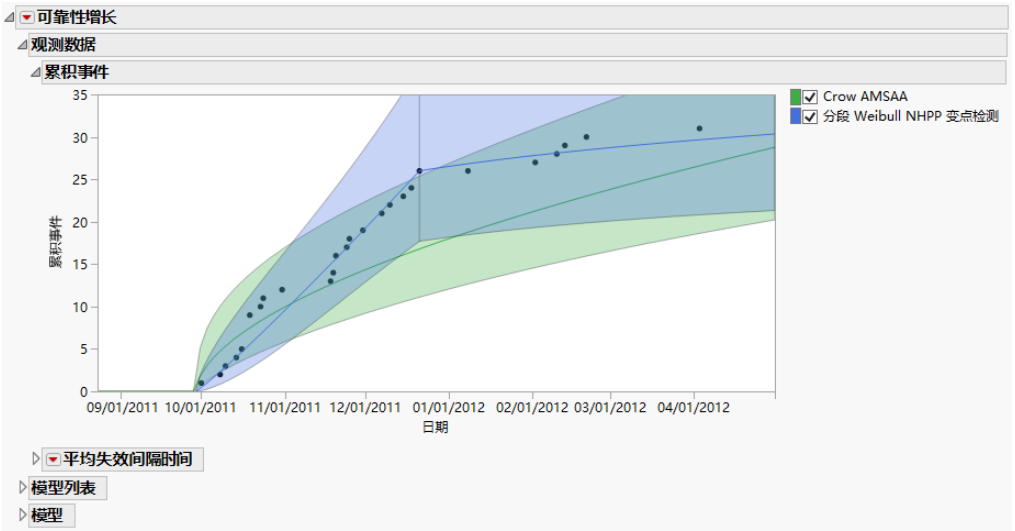
图 10.25 具有 Crow AMSAA 模型的 “累积事件”图



8. 点击 “可靠性增长”红色小三角并选择**拟合模型 > 分段 Weibull NHPP 变点检测**。

   “观测数据”报表中的 “累积事件图”会相应更新以显示使用变点检测的分段模型拟合。这两个模型均显示在图 10.26 中。尽管数据颇为稀疏，分段模型看起来可以更好地拟合数据。

图 10.26 具有两个模型的 “累积事件”图



“可靠性增长”平台的统计详细信息

- “‘Crow-AMSAA’报表的统计详细信息”
- “‘分段 Weibull NHPP 变点检测’报表的统计详细信息”

“Crow-AMSAA”报表的统计详细信息

本节包含 “Crow-AMSAA”报表中显示的参数估计值和刻画器的详细信息。

Crow-AMSAA 模型的参数估计值

除“使用修改 MLE 的 Crow-AMSAA”选项外，λ 和 β 的估计值是最大似然估计值，这些估计值按如下方式进行计算。使用 Meeker and Escobar (1998) 中的方法得到的该似然函数。它根据 $param_1 = \log(\lambda)$ 和 $param_2 = \log(\beta)$ 重新进行了参数化。这样就可以使用不受约束的优化算法，即在 - ∞ 到 + ∞ 范围上进行搜索的算法。此外还会得到 $param_1$ 和 $param_2$ 的 MLE。

λ 和 β 的标准误差根据 Fisher 信息矩阵获得。使用 Wald 统计量基于 MLE 的渐近分布计算 $param_1$ 和 $param_2$ 的置信限。然后使用指数函数将这些估计值和它们的置信限转换回原始单位。

“使用修改 MLE 的 Crow-AMSAA”的参数估计值

对于“使用修改 MLE 的 Crow-AMSAA”选项，已针对偏倚修正了 β 的估计值。针对 β 的修正偏倚的估计值的公式取决于该检验是定数截尾还是定时截尾。

用 $\hat{\beta}$ 表示 β 的 MLE，假定 n 是观测数，T 是总检验时间。

β 的修正偏倚的估计值 （**修改的 MLE**）是 $\bar{\beta}$，其中：

$$\bar{\beta} = \left(\frac{n-2}{n}\right)\hat{\beta}$$ 用于定数截尾的检验

$$\bar{\beta} = \left(\frac{n-1}{n}\right)\hat{\beta}$$ 用于定时截尾的检验

根据似然函数给出的表达式 （但基于调整的 beta 值）计算 λ 的修改 MLE （用 $\bar{\lambda}$ 表示）：

$$\bar{\lambda} = n/T^{\bar{\beta}}$$

使用 Fisher 信息矩阵估计参数的协方差矩阵。（请参见第 252 页的 “Crow-AMSAA 模型的参数估计值”。）但是，在生成的公式中用 MLE 替换 λ 和 β 的修正偏倚的估计值。图中的所有置信带和报表中的置信区间均基于该过程。

刻画器

对于 Crow-AMSAA 模型，通过用来自 MLE 的参数替换理论表达式中的参数 λ 和 β，得到刻画器中给出的 MTBF、强度和累积事件的估计值。选择“使用修改 MLE 的 Crow-AMSAA”选项时，使用修改的 MLE。通过将 delta 方法应用到相关表达式的对数，得到置信限。

例如，考虑累积事件函数。自检验开始以来在时间 t 处的累积事件数由 $N(t) = \lambda t^b$ 给出。它满足 $\log(N(t)) = \log(\lambda) + \beta\log(t)$。用 MLE（或修改的 MLE）替换 $\log(N(t))$ 中的参数 λ 和 β 来估计 $\log(N(t))$。将 delta 方法应用到该表达式来获得其方差的估计值。该估计值用于构造基于 Wald 的 95% 置信区间。然后使用指数函数转换所得的置信限以给出在 t 时间处估计的累积事件数的置信限。

“分段 Weibull NHPP 变点检测”报表的统计详细信息

按以下方式估计变点：

- 使用连续事件时间，定义不相交的区间。
- 这样的区间中的每个点可以视为定义具有两个阶段的分段 Weibull NHPP 模型的一个变点。只要该点定义的两个阶段均至少包含两个事件，算法就可以计算该模型的参数的 MLE。还可以计算该模型的对数似然。
- 在每个不相交的区间内，使用带约束的优化例程来求得似然函数的局部最优值。
- 比较这些局部最优值，并选择对应于最大值的点作为估计的变点。

请注意，该过程不同于 Guo et al. (2010) 中所述的基于网格的方法。

第 11 章

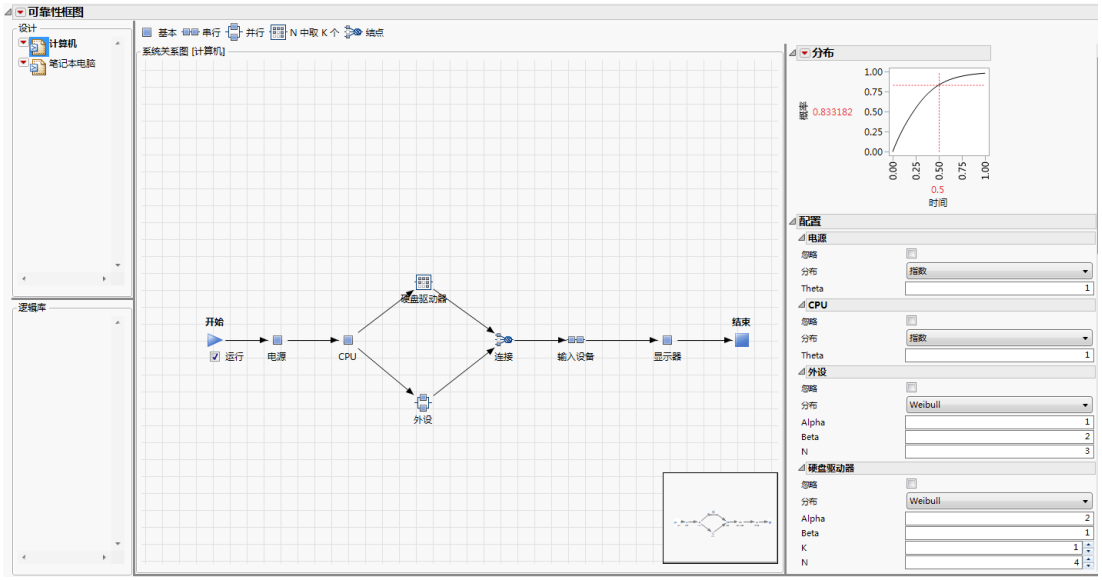
JMP PRO 可靠性框图

设计系统可靠性

“可靠性框图”平台仅在 JMP Pro 中提供。

“可靠性框图”平台显示系统元件之间的可靠性关系。若为元件指定了可靠性分布，则平台会通过分析对可靠性行为建模。“可靠性框图”（也称为“依赖性关系图”）说明元件可靠性如何影响系统的成功或失效。

图 11.1 “可靠性框图”的示例



目录

“可靠性框图”平台概述 257
使用“可靠性框图”平台的示例 257
添加元件 259
对齐形状 261
连接形状 262
配置元件 263
“可靠性框图”窗口 264
“预览”窗口 265
“可靠性框图”平台选项 266
“工作区”选项 269
配置设置 269
分布配置 269
指定非参数分布 271
用于“设计”和“逻辑库”项的选项 272
刻画器 273
元件重要性和失效时间 275
元件图 277
打印代数可靠性公式 279
生成代数表达式数据表 280
复制和删除 280

JMP PRO “可靠性框图”平台概述

使用“可靠性框图”平台，您可以绘制系统及其相关元件的关系图，以显示元件可靠性如何影响整个系统的成功。可靠性框图中的每个方框表示系统中的一个元件，它与系统中的其他元件相连。

可靠性框图可以包含以串联或并联方式连接的元件。任何串联元件中的失效将导致整个系统失效。在并联连接（或冗余）的系统中，只有所有并联元件失效时整个系统才失效。此外，关系图可以包含“N 中取 K 个”元件，其中 n 中的 k 个元件必须正常使用，系统才能正常使用。

“可靠性框图”模板在关系图左侧放置“开始”方框，在关系图右侧放置“结束”方框。使用“形状”工具，您可以对系统绘制关系图，使分析从“开始”方框开始，然后连接各元件直到“结束”方框。

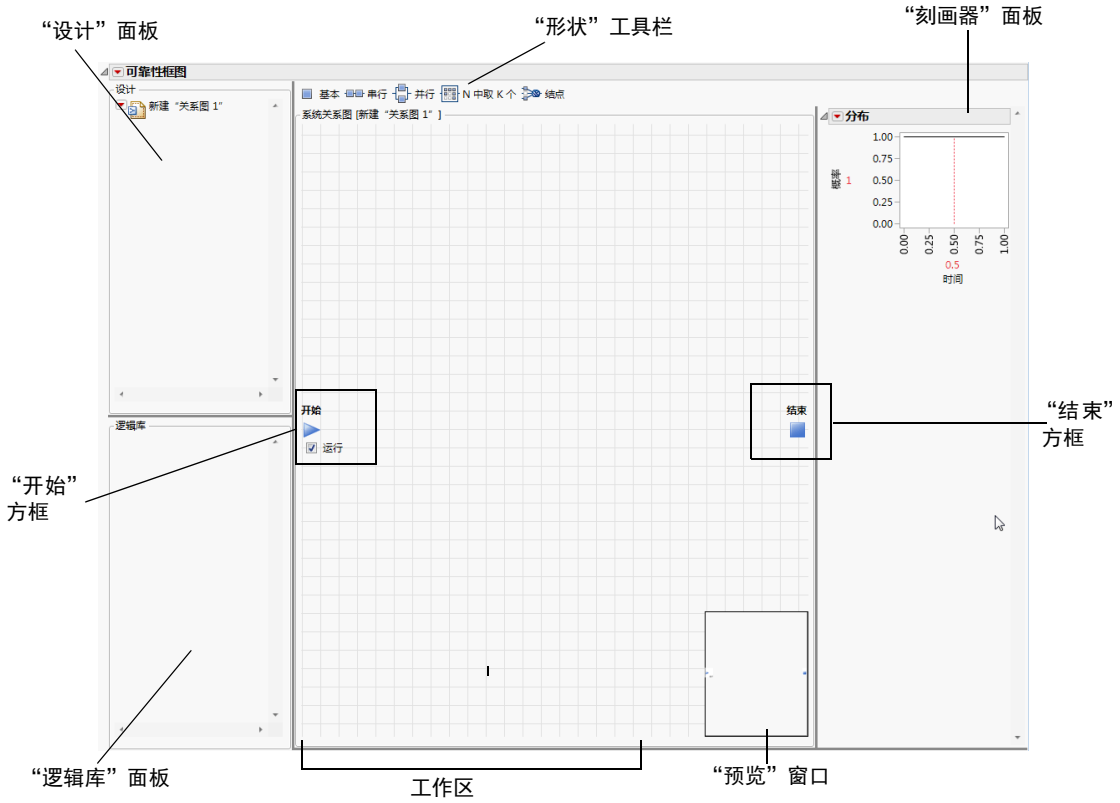
JMP PRO 使用“可靠性框图”平台的示例

在本例中，您将了解如何创建新的 “可靠性框图”。

1. 选择**分析 > 可靠性和生存 > 可靠性框图**。

   随即显示空白的 “可靠性框图”窗口。

图 11.2 新的 “可靠性框图”



注意：默认显示 “分布”刻画器。

2. 在 “设计”面板中，选择 “新建 ‘关系图 1’ ”并将其重命名为 “计算机”。

   工作区现在被命名为 “系统关系图 [计算机]”。

3. 取消选择 “开始”方框下的**运行**。

   在选择**运行**的情况下，平台将在每次系统关系图更改后更新关系图的可靠性计算。这些更改可包括添加或删除元件、更改元件的配置以及添加或删除连接。

   在取消选择**运行**的情况下，平台在任何更改后不更新可靠性计算。

提示：当您对大型系统绘制关系图时请取消选择**运行**。关系图绘制完成后再选择**运行**。

4. 继续执行第 259 页的 “添加元件”的操作。

JMP PRO 添加元件

位于工具栏中的 “可靠性框图”绘制元素称为**形状**。**元件**一词是指表示系统组成部分的形状。

1. 点击“形状”工具栏上的“基本”图标 基本，将该形状拖到系统关系图中“开始”方框的右侧。
2. 选择该标签，用 “电源”替换 “新建基本 1”，然后按 Enter 键。

图 11.3 “基本”形状

点击标签或形状后，连接箭头随即出现。这些箭头将在点击模板中的其他位置后消失。

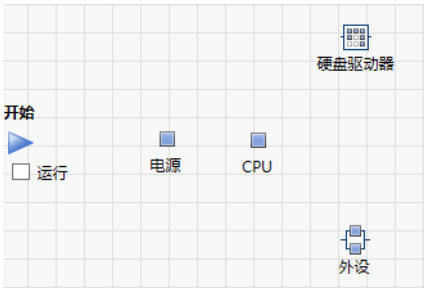
3. 将第二个 “基本”形状拖到 “电源”形状的右侧。
4. 选择标签并键入 “CPU”。

图 11.4 “系统关系图”示例

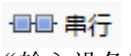
注意：您稍后将在第 261 页的“对齐形状”一节中对齐形状。

5. 将一个 “并联”形状 并行 拖到 “CPU”形状右下方的位置。
6. 选择标签并键入 “外设”。
7. 将 “N 中取 K 个”形状 N 中取 K 个 拖到 “CPU”形状右上方的位置。
8. 选择标签并键入 “硬盘驱动器”。

图 11.5 部分系统关系图

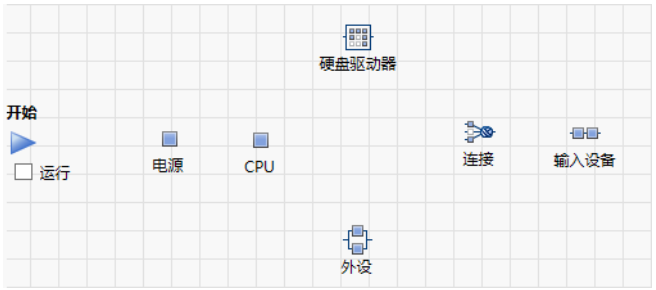


9. 将 “结点”形状

拖到以前形状的右侧。
10. 选择标签并键入 “连接”。
11. 将 “串联”形状 

拖到 “结点”形状右侧的位置。
12. 选择标签并键入 “输入设备”。

图 11.6 部分系统关系图



13. 将 “基本”形状拖到 “输入设备”形状右侧的位置。
14. 选择标签并键入 “显示器”。

图 11.7 显示所有形状的系统关系图

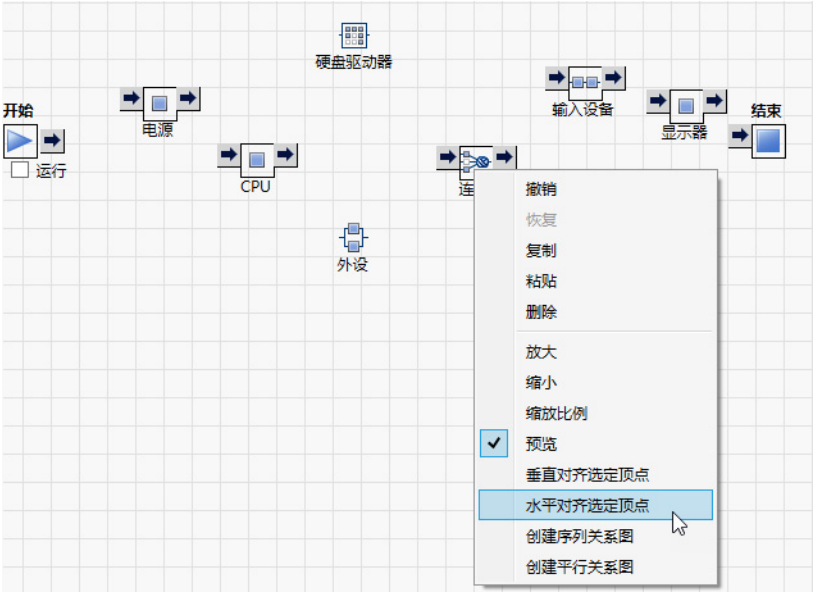
15. 继续执行第 261 页的 “对齐形状”的操作。

JMP PRO 对齐形状

1. 要垂直对齐“硬盘驱动器”和“外设”对应的形状，请选择以下元件：
   - 硬盘驱动器
   - 外设

   提示：要选择形状，请围绕形状拖动光标或按 Shift 键并点击每个形状。
2. 选择形状后，右击其中一个形状并选择**垂直对齐选定顶点**。
3. 要水平对齐其余形状，请选择以下元件：
   - 开始
   - 电源
   - CPU
   - 连接
   - 输入设备
   - 显示器
   - 结束
4. 选择形状后，右击其中一个形状并选择**水平对齐选定顶点**。

图 11.8 水平对齐形状



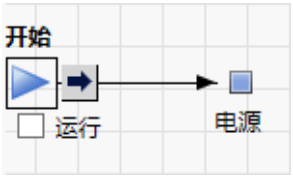
5. 继续执行第 262 页的 “连接形状”的操作。

JMP PRO 连接形状

要连接形状，请选择某个形状以显示其连接箭头。假定您想要将形状 A 连接到形状 B。选择形状 A。将右箭头拖至形状 B 以指示形状 A 在形状 B 之前。将左箭头拖至形状 B 以指示形状 B 在形状 A 之前。要连接关系图中的形状，请选择右箭头以连接到序列中的下一个形状。

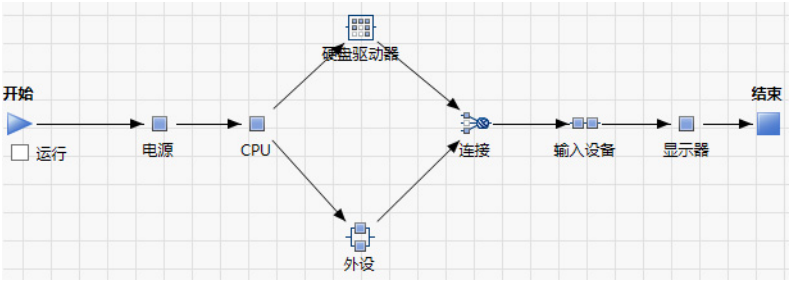
1. 选择“开始”方框（蓝箭头）以显示连接箭头。
2. 选择单个连接箭头 并将箭头拖到 “电源”元件。

图 11.9 连接形状



3. 对于以下每个元件，点击第一个元件，选择其右连接箭头，然后将该箭头拖至第二个元件：
    1. 电源 → CPU
    2. CPU → 硬盘驱动器
    3. CPU → 外设
    4. 硬盘驱动器 → 连接
    5. 外设 → 连接
    6. 连接 → 输入设备
    7. 输入设备 → 显示器
    8. 显示器 → “结束”方框

图 11.10 完成的系统关系图



4. 继续执行第 263 页的 “配置元件”的操作。

JMP PRO 配置元件

1. 在“配置”面板中，输入元件的“配置”设置。请参见第 269 页的“配置设置”。

表 11.1 配置设置

成分	设置
电源	• 分布—指数 • Theta—1
CPU	• 分布—指数 • Theta—1
外设	• 分布—Weibull • Alpha—1 • Beta—2 • N—3
硬盘驱动器	• 分布—Weibull • Alpha—2 • Beta—1 • K—1 • N—4
连接	最小可用—1
输入设备	• 分布—Fréchet • 位置—0 • 尺度—1 • N—2
显示器	• 分布—指数 • Theta—1

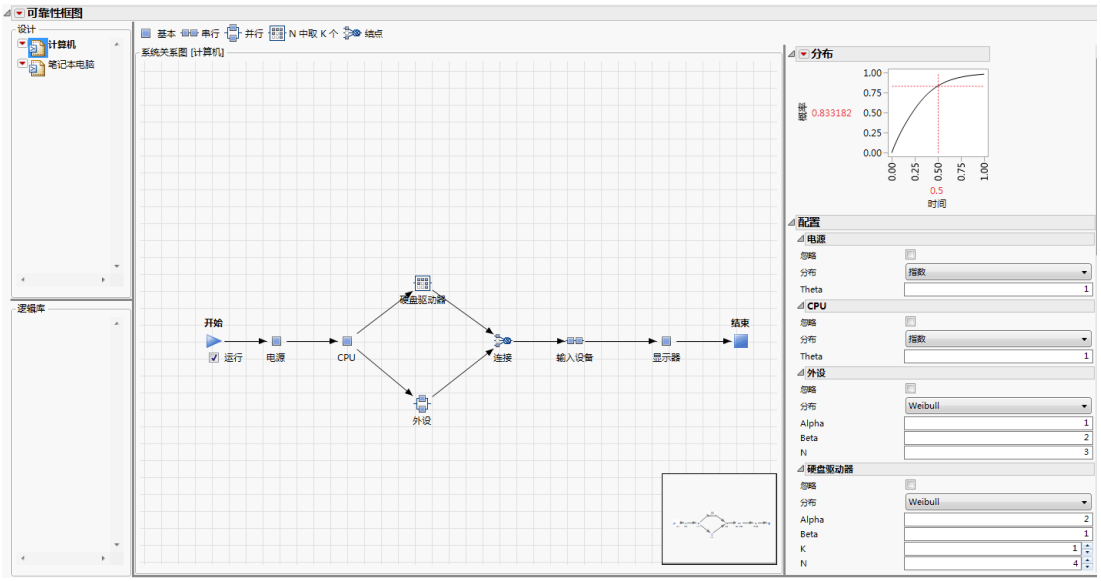
“可靠性框图”已完成 （图 11.11）。

2. 选择**运行**。

   该系统的可靠性信息随之更新。这反映在 “刻画器”窗格的 “分布”图中。

3. 要将 “可靠性框图”保存为 JMP 脚本语言 (JSL) 文件，请选择**文件** > **保存**并将其命名为 exampleRBDcomplete.jsl。

图 11.11 “可靠性框图”示例



JMP PRO “可靠性框图”窗口

“可靠性框图”窗口包含以下面板：

- 设计：列出使用“可靠性框图”平台创建的系统关系图。
- 逻辑库：列出可在创建新系统关系图时重新使用的子系统设计。

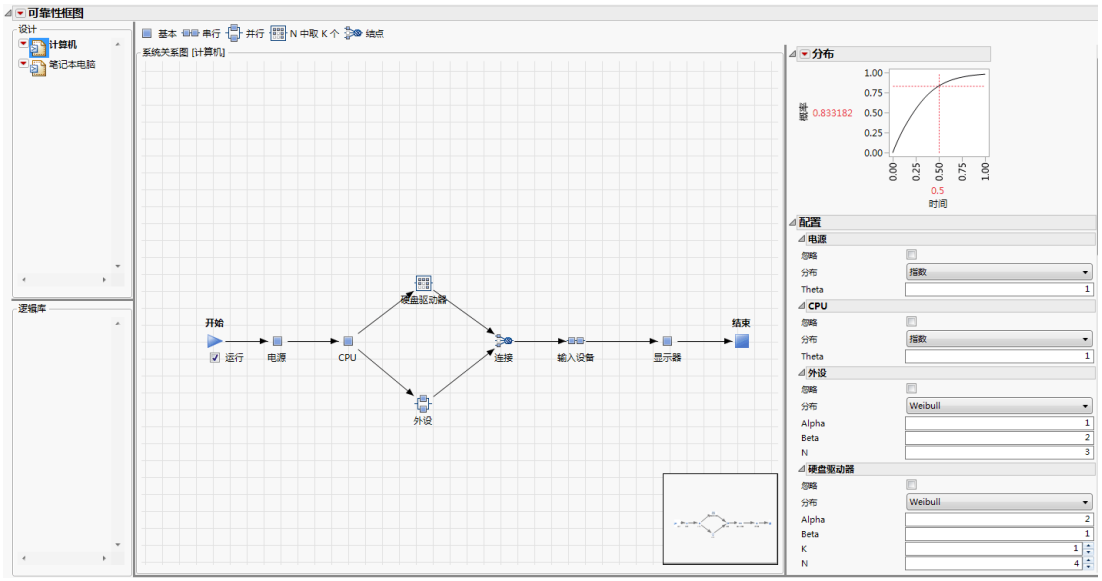
注意：在“设计”和“逻辑库”面板中显示的每个系统关系图都包含它自己的红色小三角菜单。以上每个红色小三角菜单中的选项都在第 272 页的“用于“设计”和“逻辑库”项的选项”中说明。

- 工作区：显示“形状”工具栏、“系统关系图”和“预览”窗口。

提示：要隐藏“预览”窗口，请在“工作区”中右击，然后取消选择“预览”。

- “刻画器”面板：显示“分布”刻画器、“配置”设置，以及各种系统和元件的刻画器和图，供您从“关系图”红色小三角菜单中选择。

图 11.12 “可靠性框图”的示例



“形状”工具栏包括以下绘制工具：

基本 将单个方框形状添加到系统关系图。

串行 添加串联方框形状，该形状表示串联在一起的一组元件。所有元件都必须正常使用，整个系统才能正常使用。

并行 添加并联方框形状，表示并联的一组元件。必须至少有一个元件正常使用，整个系统才能正常使用。

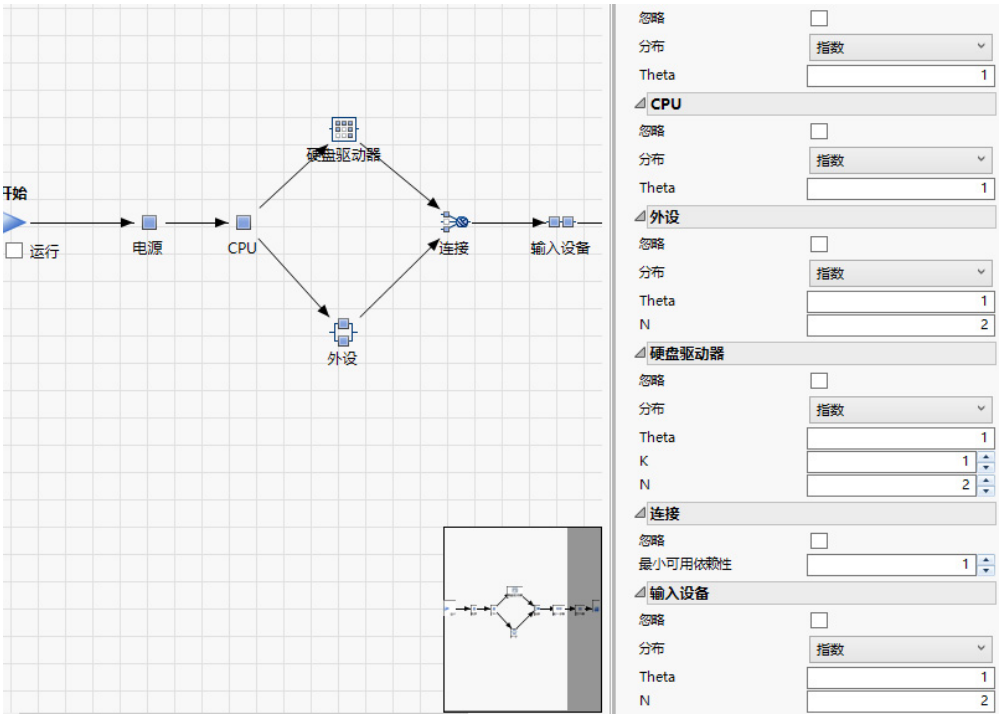
N 中取 K 个 添加“*n* 中取 *k* 个”方框形状，您要在其中指定 *k* 和 *n*。必须至少有 k 个元件正常使用，整个系统才能正常使用。

结点 向系统关系图添加一个结点形状，以便连接具有不同“分布”属性设置的“并联”或“N 中取 K 个”元件。

JMP PRO “预览”窗口

若系统关系图太大而无法显示在工作区中，“预览”窗口可支持您重新放置显示在工作区中的系统关系图部分。在“预览”窗口中，查看区由白色背景指示。拖动查看区可将“工作区”的视图位置变换为系统关系图的另一部分。

图 11.13 在白色背景中显示部分关系图的 “预览”窗口



JMP PRO “可靠性框图”平台选项

“可靠性框图”红色小三角菜单包含以下用于 “可靠性框图”窗口的选项：

“保存”和 “另存为” 允许您保存一个可靠性框图，或使用新名称将现有可靠性框图另存为 JMP 脚本语言 (JSL) 脚本，每当您在 JMP 中打开该脚本时它将自动执行。请参见《Scripting Guide》，了解有关 “自动提交”脚本的详细信息。

注意：“保存”和 “另存为”红色小三角选项等同于 “文件”>“保存”和 “文件”>“另存为”。这些选项位于红色小三角菜单中，以便于使用。

显示设计关系图 若选择了**显示设计比较**，则使用该选项可以在 “工作区”中隐藏或显示系统关系图。

显示设计比较 为选定的系统关系图显示“分布叠加”刻画器和“剩余寿命分布叠加”刻画器。请参见第 267 页的 “显示设计比较”。

导入元件分布设置 支持您从数据表中导入系统关系图的配置设置。该表必须包含用于关系图类别、关系图名称、元件名称、分布以及一个或多个参数的列。参数个数取决于指定分布。

注意：导入表中的字符串必须与系统关系图中的字符串完全匹配。

新建设计项 允许您向“设计”面板添加新的设计关系图。请参见第 268 页的“添加新设计项”。

新建逻辑库项 允许您向“逻辑库”面板添加新项。使用“逻辑库”面板，您可以创建子系统关系图以供在多个系统设计中使用。请参见第 268 页的“添加新逻辑库项”。

注意：逻辑库项仅可用于包含在 JSL 文件的“设计”面板中的关系图。它们不能用于其他 JSL 文件的其他关系图中。

JMP PRO 显示设计比较

显示设计比较选项显示“分布叠加”图和“剩余寿命分布叠加”图。这些图显示在系统关系图下方的工作区中。请参见图 11.14 和图 11.15。

- “分布叠加”图显示“设计”面板中每个设计的系统失效概率。
- “剩余寿命分布叠加”图显示“设计”面板中每个设计的失效条件概率，前提是系统存续到指定的生存时间。在图右侧的框中输入“生存时间”。或者，选择原点处的小矩形，并将其拖至右侧以便动态设置“生存时间”。

复选框支持您选择在图中展示哪些设计。这样您就可以比较一部分设计的失效概率。

提示：要在“工作区”中隐藏系统关系图，请点击“可靠性框图”红色小三角并取消选择**显示设计关系图**。或者，要显示更多设计比较，可向上重新定位水平拆分条。

图 11.14 “分布叠加”示例

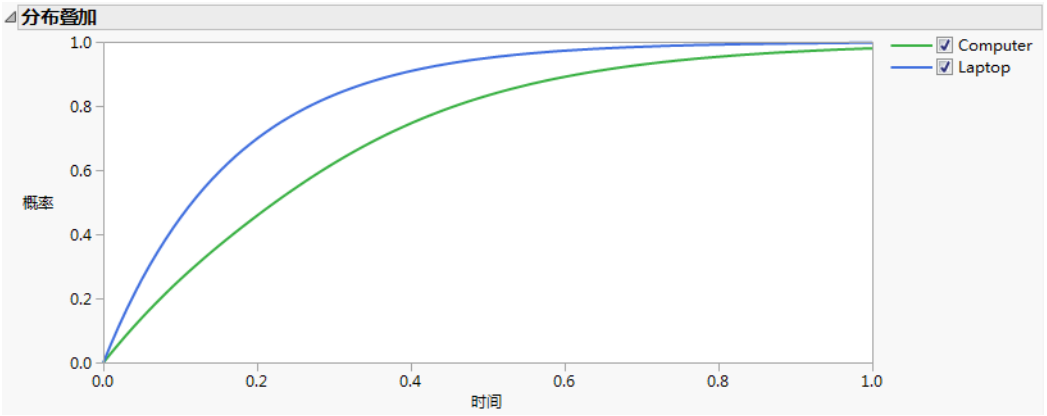
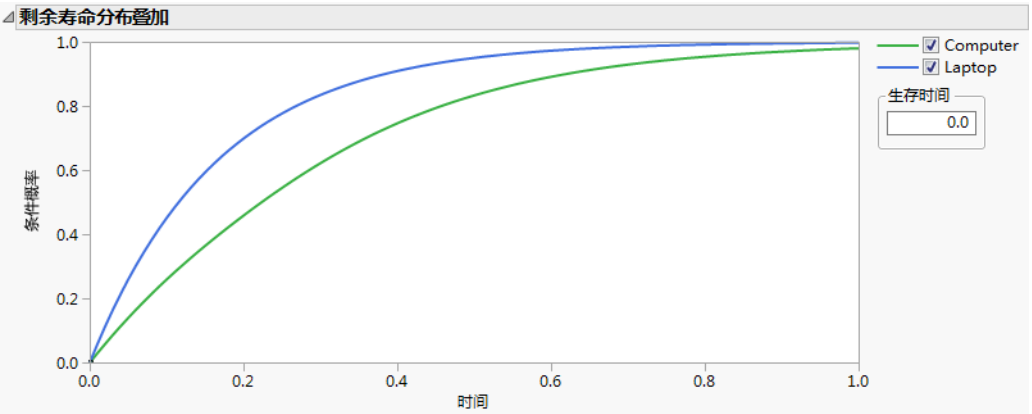


图 11.15 “剩余寿命分布叠加”示例



JMP PRO 添加新设计项

新建设计项命令可将新系统设计添加到 “设计”面板列表中。

- 点击“可靠性框图”红色小三角并选择**新建设计项**。
- 名为“新建关系图 X”的设计（其中 X 是标识关系图名称的编号）将添加到“设计”面板列表中。
- 要命名该设计，请选择标签并输入名称。
- 使用“形状”工具栏绘制系统关系图。（请参见第 257 页的“使用“可靠性框图”平台的示例”。）

  提示：您还可以从另一个框图模板复制和粘贴关系图主体，但不能复制“开始”和“结束”形状。粘贴关系图后，您必须重新连接 “开始”和 “结束”形状。

- 配置元件。
- 保存该文件。

要在“系统关系图”窗口中显示某个设计并查看其“刻画器”面板，可在“设计”面板中双击其图标。

JMP PRO 添加新逻辑库项

新建逻辑库项选项可将新的子系统添加到 “逻辑库”面板列表。

提示：将一个系统设计从“设计”面板拖到“逻辑库”面板可将它作为子系统添加到“逻辑库”。

- 点击“可靠性框图”红色小三角并选择**新建逻辑库项**。

- 名为“新建关系图 X”的子系统（其中 X 是标识关系图名称的编号）将添加到“逻辑库”面板列表中。
- 要命名该子系统，请选择标签并输入名称。
- 使用“形状”工具栏绘制子系统关系图。
- 配置元件。
- 创建从“开始”方框到“结束”方框的连接。
- 保存该文件。

JMP PRO “工作区”选项

“工作区”面板支持几个用于调整面板视图的右击命令。以下是一些可用的命令：

放大 允许您放大系统关系图。

缩小 允许您缩小系统关系图。

提示：在 Windows 上，您可以按 Ctrl 键并使用鼠标滚轮放大或缩小关系图。

缩放比例 打开“设置缩放比例”窗口，以便您可以使用尺度因子进行放大或缩小。尺度因子的范围从 0.75 (75%) 到 5.0 (500%)。

预览 允许您在“工作区”面板中隐藏或显示“预览”窗口。

垂直对齐选定顶点 允许您在垂直线上对齐形状。

水平对齐选定顶点 允许您在水平线上对齐形状。

创建串联关系图 允许您使用关系图中的节点创建串联关系图。串联的顺序由创建节点的顺序决定。仅当关系图不包含任何箭头时该选项才可用。

创建并联关系图 允许您使用关系图中的节点创建并联关系图。仅当关系图不包含任何箭头时该选项才可用。

JMP PRO 配置设置

可为“可靠性框图”中的每个元件都分配一个失效分布。可用失效分布列在第 270 页上的表 11.2 中。要查看这些失效分布的公式和参数化信息，请参见“寿命分布”一章中第 72 页的“分布”。

JMP PRO 分布配置

向关系图添加某个元件时，该元件的分级显示项显示在“配置”分级显示项下方。

注意：您可以通过选中“忽略”旁边的框从分析中忽略某个元件。

选择某个分布并输入必需的参数值。

表 11.2 分布和参数

属性类型	所需的输入信息
指数	Theta
Weibull	Alpha、Beta
对数正态	位置、尺度
对数 Logistic	位置、尺度
Fréchet	位置、尺度
广义 Gamma	mu、sigma、lambda
缺陷子总体 Weibull	Alpha、Beta、缺陷概率
缺陷子总体对数正态	位置、尺度、缺陷概率
缺陷子总体对数 Logistic	位置、尺度、缺陷概率
缺陷子总体 Fréchet	位置、尺度、缺陷概率
非参数	数据或数据文件

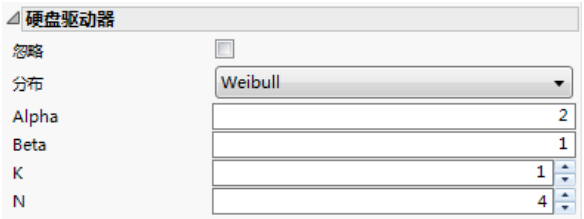
要查看选定设计或子系统中元件的 “配置”设置，请执行以下操作：

- 要查看特定元件的“配置”设置，请选择该元件的形状。
- 要查看多个元件的“配置”设置，请使用箭头工具或按 Ctrl 键并点击来选择多个元件的形状。
- 要查看所有元件的“配置”设置，请通过点击工作区中的空白部分取消选择所有形状。

对于关系图中的每个元件，您可以通过选中 “省略”旁边的框在计算中忽略该元件，或者：

- 从分布列表中，选择适当的分布。对于“非参数”之外的所有选择，输入分布的参数值。对于“非参数”，请参见第 271 页的“指定非参数分布”。
- 对于“串联”和“并联”元件，您还必须输入 N 的值（N 是串联或并联形状中包含的元件总数）。
- 对于“N 中取 K 个”元件，您还必须输入 K（系统正常使用所需的最少正常使用元件数），以及 N（形状中的元件总数）。
- 对于“结点”元件，输入最小可用依赖性。“结点”形状支持您配置“*n* 中取 *k* 个元件”形状，其中连接的形状具有不同的分布。最小依赖项数目为 *k*，即可正常使用的以便系统运行的通向结点的最小路径数。

图 11.16 示例：针对 “N 中取 K 个” 形状的 Weibull 配置

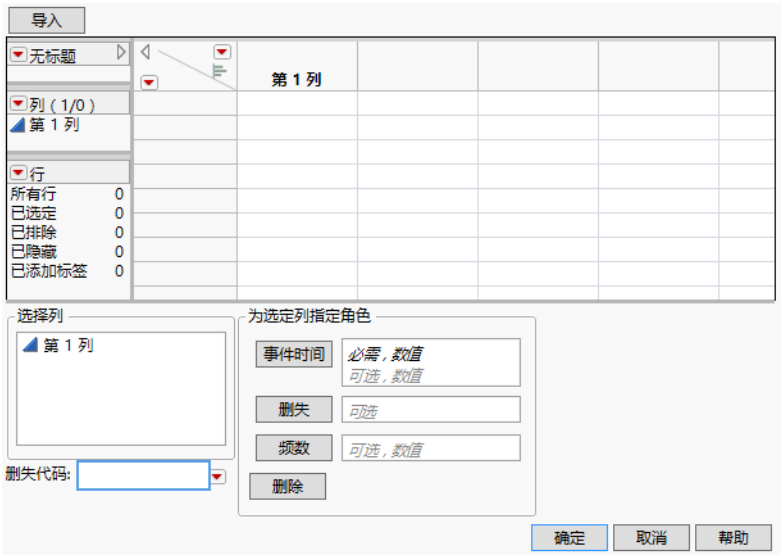


JMP PRO 指定非参数分布

“分布”下的“非参数”选项支持您近似任意分布。输入数据或导入包含大型数据集的文件。该数据用于近似分布。

选择“非参数”后，点击“数据”旁边的 … 图标。随即显示“提供数据”窗口，您可以在其中输入数据或导入数据文件。一旦导入或输入数据，即使用数据计算元件的非参数分布。

图 11.17 “提供数据”窗口



要从文件导入数据，请执行以下操作。

1. 打开包含要导入的数据的 JMP 数据表。
2. 在 “提供数据”窗口中，点击**导入**。
   随即显示 “选择数据表”窗口。
3. 从 “数据表”列表中，选择该数据表。
4. 点击**确定**。

5. 在数据网格下方的面板中，指定哪些列表示 “事件时间”数据以及 “删失”和 “频数”数据 （若适用）。
6. 在 “提供数据”窗口中，点击**确定**。

要手动输入数据，请执行以下操作。

1. 创建用于“事件时间”数据的列以及“删失”和“频数”数据的列（若适用）。
2. 在列中输入数据。
3. 在数据网格下方的面板中，指定哪些列表示 “事件时间”数据以及 “删失”和 “频数”数据 （若适用）。
4. 在 “提供数据”窗口中，点击**确定**。

JMP PRO 用于“设计”和“逻辑库”项的选项

可用于 “设计”和 “逻辑库”项的选项如下所列：

- “显示配置”显示或隐藏“配置”分级显示项。请参见第 269 页的“配置设置”。
- 用于显示各种刻画器的选项。请参见第 273 页的“刻画器”和以下信息：
  - 第 273 页的“ “分布”刻画器”
  - 第 273 页的“ “剩余寿命分布”刻画器”
  - 第 274 页的“ “可靠性”刻画器”
  - 第 274 页的“ “分位数”刻画器”
  - 第 274 页的“ “密度”刻画器”
  - 第 274 页的“ “危险率”刻画器”
  - 第 274 页的“ “累积危险率”刻画器”
- 用于显示元件重要性测度和平均失效时间的图的选项。请参见第 275 页的“元件重要性和失效时间”和以下信息：
  - 第 275 页的“Birnbaum 元件重要性”
  - 第 276 页的“剩余寿命 BCI”
  - 第 276 页的“元件综合重要性”
  - 第 277 页的“平均失效时间”
- 用于显示系统关系图中的元件叠加图的选项。请参见第 277 页的“元件图”和以下信息：
  - 第 277 页的“元件分布函数”
  - 第 277 页的“元件可靠性函数”
  - 第 278 页的“元件密度函数”
  - 第 278 页的“元件危险率函数”

– 第 279 页的 “元件累积危险率函数”
- 第 279 页的 “打印代数可靠性公式”
- 第 280 页的 “生成代数表达式数据表”
- 第 280 页的 “复制和删除”

JMP PRO 刻画器

提供多种刻画器来帮助您分析系统的可靠性属性。您可以查看列在 “设计”和 “逻辑库”面板中的每个系统关系图的刻画器。本节对这些刻画器进行说明。这些刻画器显示在报表的 “刻画器”面板中。

刻画器的红色小三角选项

每个刻画器都有一个包含以下命令的红色小三角菜单：

重置因子网格 显示一个窗口，您可在其中输入用于控制显示的当前设置和值。有关设置因子网格的详细信息，请参见 《刻画器指南》。

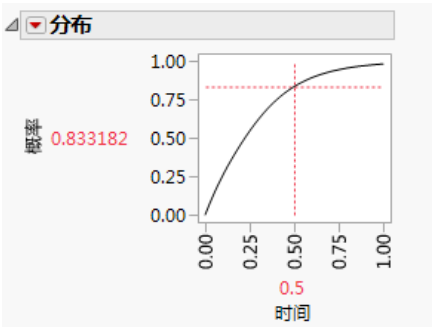
因子设置 选择该选项可配置刻画器的设置并链接这些刻画器。请参见 《刻画器指南》，了解有关 “因子设置”的详细信息。

注意：调整刻画器的 X 和 Y 轴可查看所需的图形部分。

JMP PRO “分布”刻画器

“分布”刻画器显示系统作为时间函数的失效概率。默认显示系统的 “分布”刻画器。

图 11.18 分布

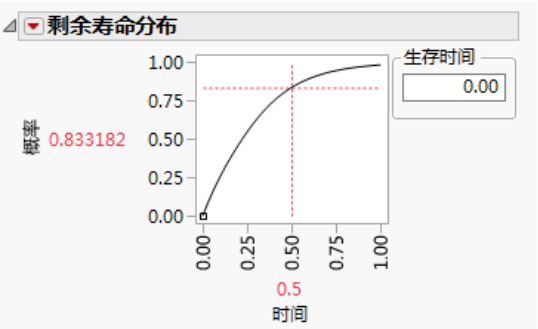


JMP PRO “剩余寿命分布”刻画器

系统的 “剩余寿命分布”刻画器显示系统的失效概率，前提是系统生存至指定的时间（指定为 “生存时间”）。默认情况下，“生存时间”设置为 0，此时剩余寿命分布函数等同于分布函数。

输入**生存时间**的值，指示系统无故障生存的时间。作为输入值的替代方法，您可以选择图中原点处的小矩形并将它拖到右侧，以便动态地设置“生存时间”。

图 11.19 剩余寿命分布



JMP PRO “可靠性”刻画器

系统的“可靠性”刻画器显示系统作为时间函数的正常使用的概率。可靠性函数亦称生存函数。

JMP PRO “分位数”刻画器

系统的“分位数”刻画器显示作为失效概率函数的时间。请注意，“分位数”函数是“分布”函数的反函数。

JMP PRO “密度”刻画器

系统的“密度”刻画器显示与系统的失效分布函数关联的概率密度函数。

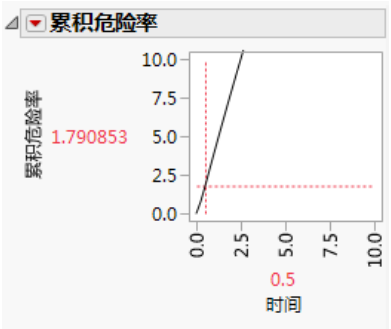
JMP PRO “危险率”刻画器

系统的“危险率”刻画器显示给定时间的瞬间失效率。

JMP PRO “累积危险率”刻画器

系统的“累积危险率”刻画器显示作为时间的函数的累积危险率函数。

图 11.20 累积危险率



JMP PRO 元件重要性和失效时间

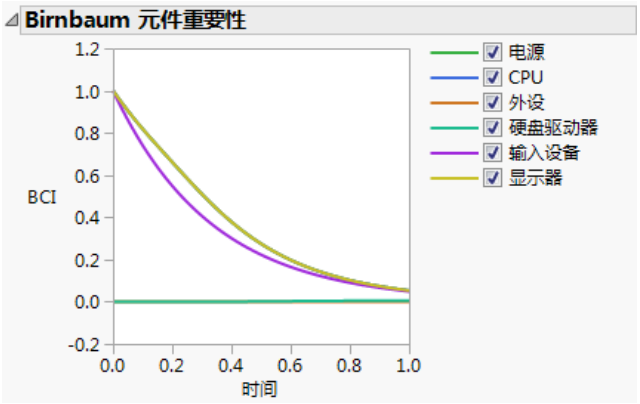
针对列在“设计”和“逻辑库”面板中的每个系统关系图都提供用于分析元件重要性和平均失效时间的图。本节说明元件重要性测度和平均失效时间。这些图和统计量显示在报表的“刻画器”面板中。

注意：使用每个元件重要性图的图例中的复选框，您可以选择要在该图中查看哪些元件。

JMP PRO Birnbaum 元件重要性

选择**显示 BCI** 可为选定系统关系图的每个元件显示 Birnbaum 元件重要性测度的叠加图。在给定时间处的元件 BCI 是当元件失效时系统失效的概率。具有较大 BCI 的元件对于系统可靠性至关重要。

图 11.21 Birnbaum 元件重要性

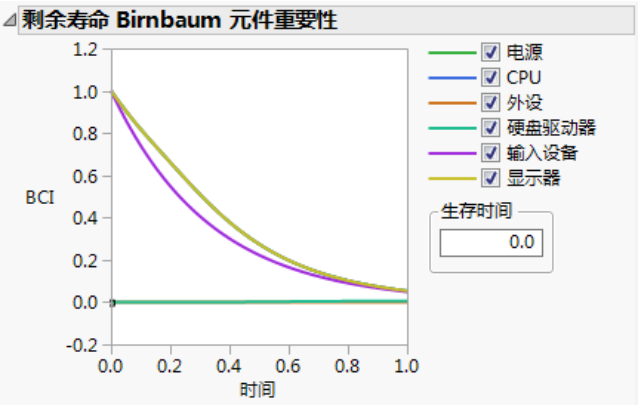


JMP PRO 剩余寿命 BCI

选择**显示剩余寿命 BCI** 可显示剩余寿命 Birnbaum 元件重要性的叠加图。剩余寿命 BCI 是若元件失效而造成系统失效的概率，前提是系统生存至指定的时间（指定为“生存时间”）。默认情况下，“生存时间”设置为 0，此时剩余寿命 BCI 等同于 Birnbaum 元件重要性。

输入**生存时间**的值，指示系统无故障生存的时间。作为输入值的替代方法，您可以选择图中原点处的小矩形并将它拖到右侧，以便动态地设置“生存时间”。

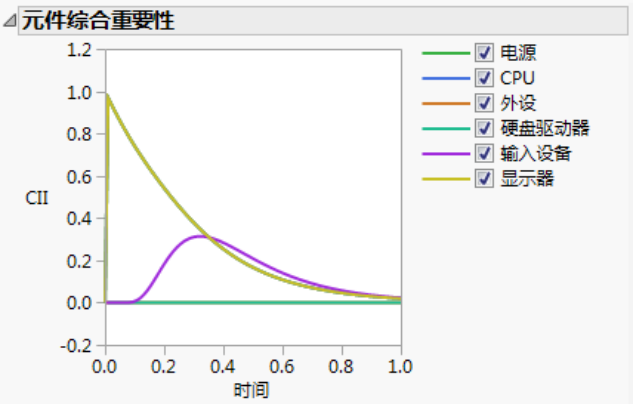
图 11.22 剩余寿命 Birnbaum 元件重要性



JMP PRO 元件综合重要性

选择**显示元件综合重要性**可为“可靠性框图”的元件显示综合重要性测度的叠加图。每个元件的综合重要性测度都考虑到元件的失效率以及瞬间失效的可能性。请参见 Si et al. (2012)。

图 11.23 元件综合重要性



JMP PRO 平均失效时间

选择**显示 MTTF** 可查看系统的“平均失效时间 (MTTF)”。

注意：用于计算“平均失效时间”的公式取决于指定的失效分布和系统中每个元件的“配置”设置。

JMP PRO 元件图

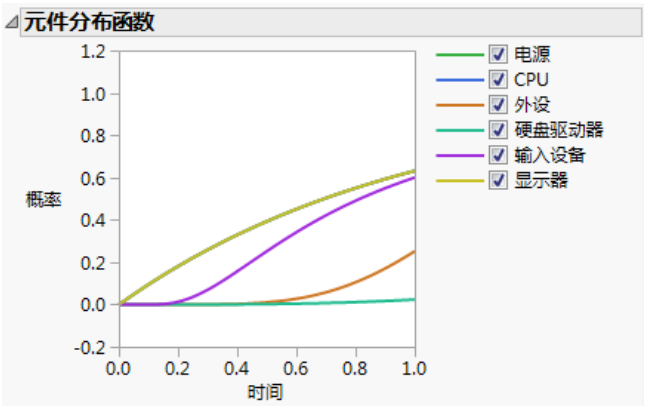
系统关系图通常包含许多单个元件。可以在叠加图中检查这些元件中每一个元件的可靠性函数。您可以查看列在“设计”和“逻辑库”面板中的每个系统关系图的元件叠加图。本节说明元件叠加图。这些图显示在报表的“刻画器”面板中。

注意：使用每个元件图的图例中的复选框，您可以选择要在该图中查看哪些元件。

JMP PRO 元件分布函数

选择**显示元件分布函数**可显示“可靠性框图”元件的分布函数的叠加图。

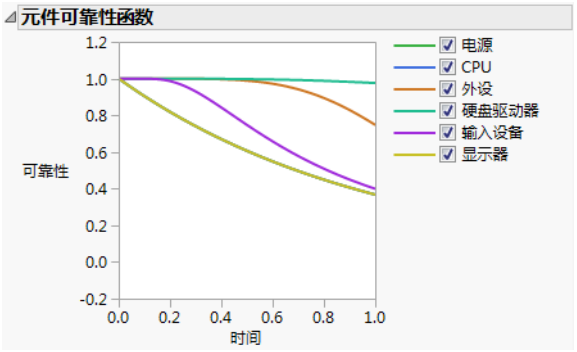
图 11.24 元件分布函数



JMP PRO 元件可靠性函数

选择**显示元件可靠性函数**可显示“可靠性框图”元件的可靠性函数的叠加图。

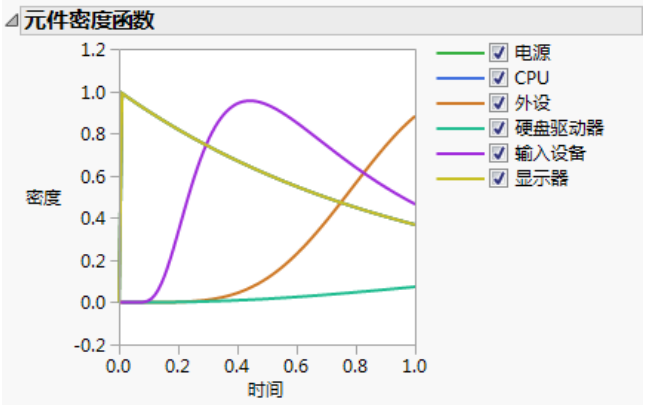
图 11.25 元件可靠性函数



JMP PRO 元件密度函数

选择**显示元件密度函数**可显示“可靠性框图”元件的密度函数的叠加图。

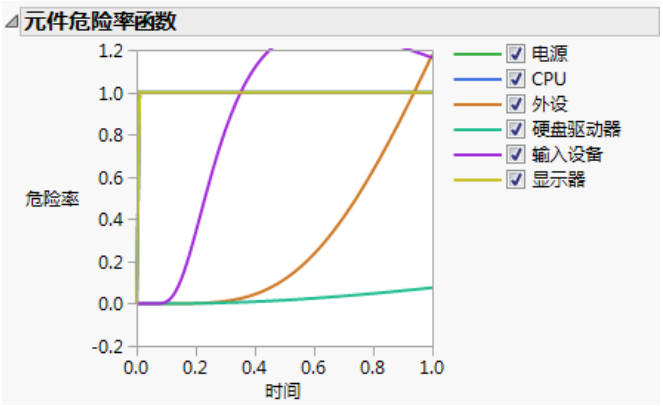
图 11.26 元件密度函数



JMP PRO 元件危险率函数

选择**显示元件危险率函数**可显示“可靠性框图”元件的危险率函数的叠加图。

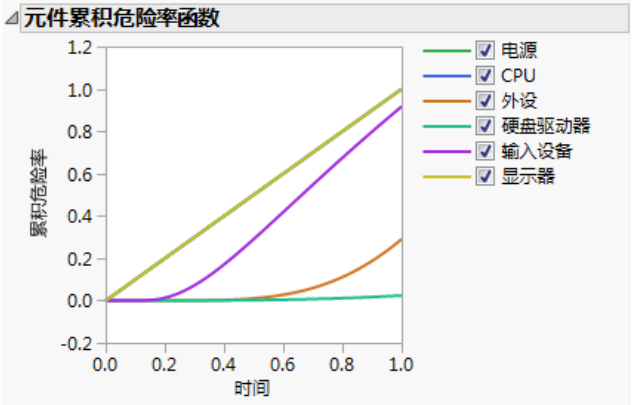
图 11.27 元件危险率函数



JMP PRO 元件累积危险率函数

选择**显示元件累积危险率函数**可显示“可靠性框图”元件的累积危险率函数的叠加图。

图 11.28 元件累积危险率函数



JMP PRO 打印代数可靠性公式

“将代数可靠性打印至日志窗口”选项可以将所选系统关系图的可靠性公式打印至“日志”窗口。

1. （仅限 Windows）选择**视图 > 日志**。

   或

   （仅限 macOS）选择**窗口 > 日志**。
2. 打开您创建的 exampleRBDcomplete.jsl 文件。
3. 在 “设计”面板中，选择**计算机**。

4. 点击 “计算机”红色小三角并选择**将代数可靠性打印至日志窗口**。
“日志”窗口显示选定框图的代数可靠性的公式。

图 11.29 “日志”窗口中的代数可靠性

```
/*:

可靠性框图[]代数可靠性:
R["电源"] * R["CPU"] * R["外设"] * R["输入设备"] * R["显示器"]
 + R["电源"] * R["CPU"] * F["外设"] * R["硬盘驱动器"] * R["输入设备"] * R["显示器"]
```

注意：可靠性公式使用“R”表示元件的可靠性，使用“F”表示元件失效的概率。

JMP PRO 生成代数表达式数据表

“生成代数表达式数据表”选项创建一个新的数据表，其中包含作为列公式的代数可靠性表达式。该表为系统中每个元件都包含一列。最后一列包含系统可靠性代数表达式，作为使用之前的列作为参数的 JSL 列公式。

JMP PRO 复制和删除

列在“设计”和“逻辑库”面板中的每个系统关系图都包含一些选项，可协助管理设计和逻辑库项。本节说明复制和删除操作。

JMP PRO 复制设计或逻辑库项

复制选项可创建与选定的系统设计或逻辑库子系统相同的新系统设计或逻辑库子系统。

- 从红色小三角菜单中选择**复制**，以便复制设计或逻辑库项。
新的设计或逻辑库项会添加到 “设计”或 “逻辑库”列表中。
- 保存该文件。

JMP PRO 删除设计或逻辑库项

删除选项可从面板中删除选定的系统设计或逻辑库子系统。

- 从红色小三角菜单中选择**删除**，以便删除设计或逻辑库项。
指定的设计或逻辑库项将从 “设计”或 “逻辑库”列表中删除。
- 保存该文件。

第 12 章

JMP PRO 可修复系统模拟

估计复杂系统的中断时间

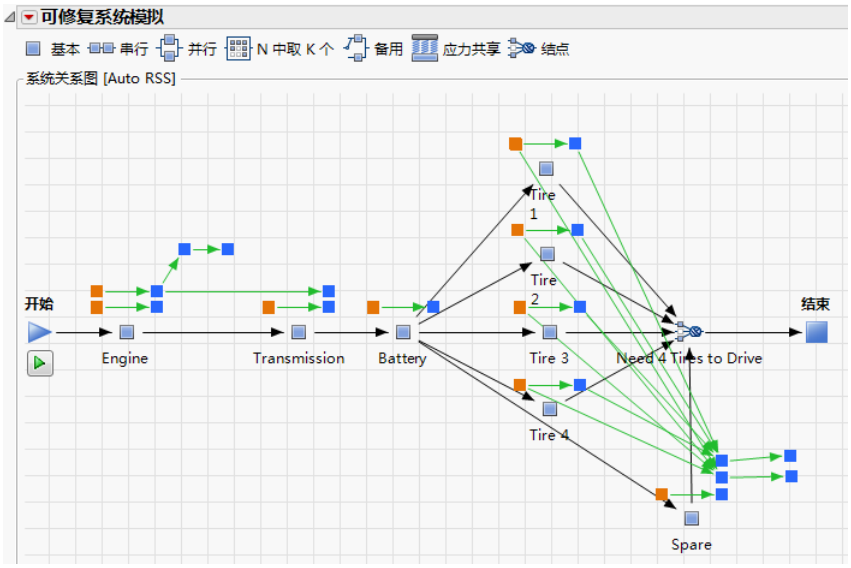
“可修复系统模拟”平台仅在 JMP Pro 中提供。

“可修复系统模拟”(RSS) 平台支持您探索复杂可修复系统内部的可靠性。**可修复系统**由因老化而需要维修的各个元件构成。可修复系统的状态会随着其元件失效并因此需要修复而发生变化。

可修复系统渗透到现代生活中的很多方面。从冰箱到房屋到电厂和电话通信网络，可修复系统的规模大小不一。

RSS 平台支持您模拟不同的系统配置以便优化您的可修复系统。例如，您可以在保持安全系统操作的同时，延长昂贵元件的使用寿命或最大化生产输出。

图 12.1 示例：可修复系统模拟关系图



目录

“可修复系统模拟”平台概述 283
示例：使用“可修复系统模拟”平台 283
“可修复系统模拟”窗口 285
系统关系图 285
“形状”工具栏 286
“配置”面板 287
“添加事件”面板 289
“添加操作”面板 290
“可修复系统模拟”平台选项 292
用于方框项的选项 293
分布选项 293
指定非参数或估计分布 294
事件设置 295
操作设置 295
可修复系统模拟结果 296
结果表 296
结果分析器 297
可修复系统模拟结果选项 298
RSS 结果分析器点估计刻画器 299

JMP PRO “可修复系统模拟”平台概述

“可修复系统模拟”(RSS) 平台支持您模拟可修复系统并分析其**中断时间**。中断时间是系统因为计划内维护或意外故障而未处于“开”状态的总时间。

通过系统关系图来表示可修复系统。系统关系图中的方框形状表示连接到其他元件的一个或多个元件。在方框中执行操作的元件被称为**正常使用**。可以串连或并连元件。串连元件时，系统将在串连中的任何元件失效后失效。并连元件时，系统将在并连中的所有元件失效后失效。

若“开始”方框和“结束”方框之间至少有一个路径未通过方框形状失效，即称元件路径为**未中断**。在模拟过程中，只要有未中断的元件路径，系统即一直保持“开”状态。若方框形状失效中断了元件路径，则系统设置为“关闭”状态。若方框形状失效但元件路径未中断，则系统会保持“开”状态。

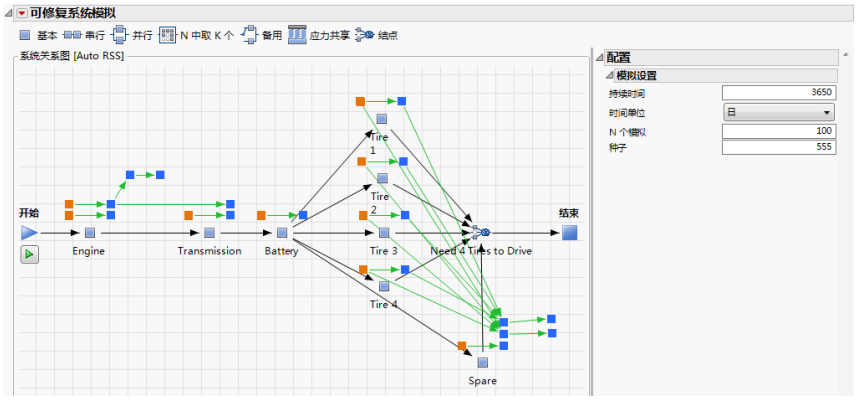
您可以添加唯一事件和操作以便表征各个元件的可修复本质的特性。**事件**是指系统中特定事件的发生，如元件失效、计划的维护或开始模拟迭代。事件触发一个或多个**操作**的执行，这些操作可表述元件的行为。操作可改变元件或整个系统的状态。

JMP PRO 示例：使用“可修复系统模拟”平台

在本例中，您运行 100 次小汽车系统的模拟迭代，并且分析该系统在 10 年间的估计中断时间。

1. 选择**帮助 > 样本数据**，点击**打开样本脚本目录**，然后打开 Car Repair Simulation.jsl。

图 12.2 小汽车修复系统关系图



随即显示绘制小汽车系统关系图的 RSS 窗口。在该关系图中，左侧第一个方框是 “开始”方框。注意在 “配置”面板中， RSS 窗口的右侧，模拟设置为运行 3650 天。

2. （可选）在 “种子”旁边输入 555。

   由于模拟涉及随机元件失效，该操作确保您可以得到以下所示的精确结果。

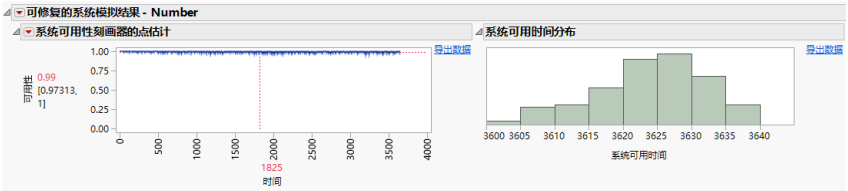
3. 点击 “开始”方框下的绿色小三角模拟小汽车系统。

随即显示包含模拟结果的数据表。

4. 点击 “Launch Repairable Systems Simulation Results Explorer”脚本旁边的绿色箭头。

   随即显示包含“可修复系统模拟结果”报表的窗口。有关如何解释这些结果的详细信息，请参见第 296 页的 “可修复系统模拟结果”。

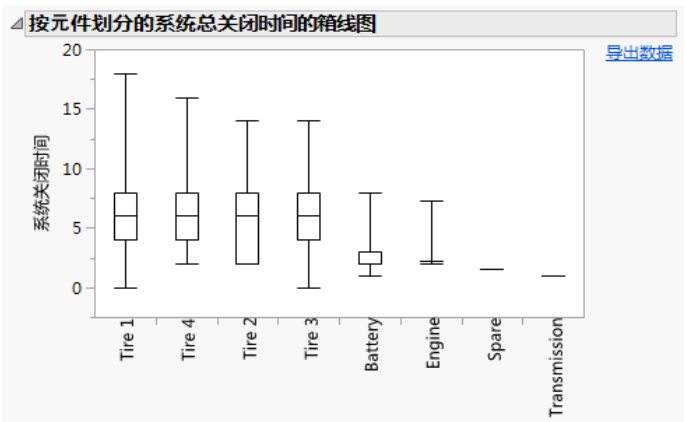
图 12.3 部分 RSS 报表



您预测该汽车在未来 10 年内可用的天数为 3,600 到 3,640 天之间。由于“系统可用性的点估计”图中显示的值接近 1，所以您推断该汽车在未来 10 年的大多数时间内都可以驾驶。而您关注的是哪些元件导致了大多数的系统关闭。

5. 点击 “可修复系统模拟结果 - Number”旁边的红色小三角，然后选择**按元件划分的系统总关闭时间的箱线图**。

图 12.4 部分显示的 “按元件划分的系统总关闭时间的箱线图”



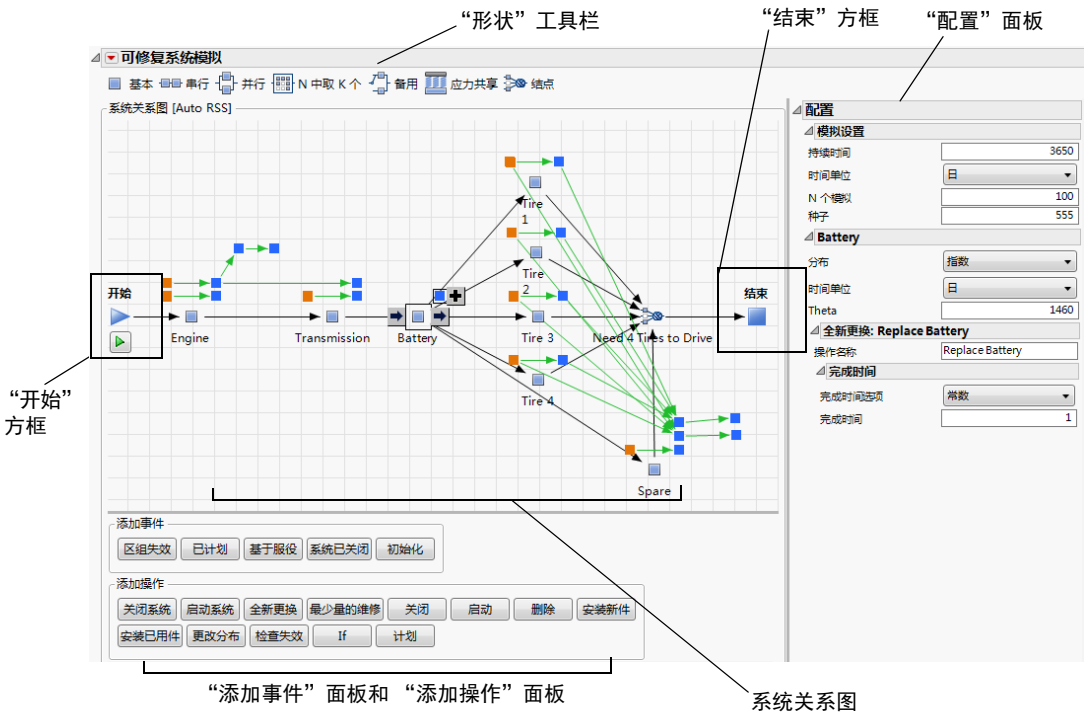
您推断轮胎导致了大多数的汽车系统关闭。要提高平均的系统总可用性，您可以考虑使用更耐磨的轮胎或另外增加一个备胎。

JMP PRO “可修复系统模拟”窗口

通过选择**分析 > 可靠性和生存 > 可修复系统模拟**启动 “可修复系统模拟”平台。“可修复系统模拟”窗口分为以下几个面板：

- 第 285 页的“系统关系图”
- 第 286 页的“ “形状”工具栏”
- 第 287 页的“ “配置”面板”
- 第 289 页的“ “添加事件”面板”
- 第 290 页的“ “添加操作”面板”

图 12.5 “可修复系统模拟”窗口



JMP PRO 系统关系图

系统关系图是您可以在其中绘制可修复系统的空间。新的系统关系图包含“开始”方框和“结束”方框。

以下选项显示在当您右击“系统关系图”时出现的弹出式菜单中：

运行模拟 使用“配置”面板中的当前“模拟设置”运行模拟。若为“种子”指定了非零值，模拟结果可再现。

运行多线程模拟 运行系统的多线程模拟。运行多线程模拟时，模拟结果不可再现。

关系图操作 包含用于控制关系图外观的选项。

删除 （在选定区组、事件或动作时可用。）从关系图中删除选定项。

显示区组名称 显示或隐藏显示在关系图中区组下方的名称。

放大 将“缩放比例”值增大 0.9 倍。

缩小 将“缩放比例”缩小 0.9 倍。

提示：在 Windows 上，您可以按 Ctrl 键并使用鼠标滚轮放大或缩小关系图。

缩放比例 支持用户在 0 到 10000 的尺度内设置缩放比例。最初的“缩放比例”值为 1。“缩放比例”值随着您放大关系图而增大。

预览 显示或隐藏“系统关系图”右下角的“预览”窗口。

显示区组间的事件动作链接 显示或隐藏表示从一个区组到另一个区组的动作链接的绿箭头。若关闭该选项，选定事件的动作链接和动作仍然显示。

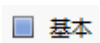
垂直对齐选定顶点 （仅当选定不止一个区组时才可用。）更新选定区组的水平位置，以使其垂直对齐。

水平对齐选定顶点 （仅当选定不止一个区组时才可用。）更新选定区组的垂直位置，以使其水平对齐。

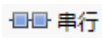
右击“系统关系图”中的某项时，更多的子菜单项会显示在弹出式菜单中。显示的子菜单项取决于项类型。这些选项支持您添加、删除或更改附加到区组项的事件和动作。

JMP PRO “形状”工具栏

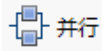
“形状”工具栏包含用于表示系统关系图中元件的方框形状。通过点击方框形状对应的图标并将其拖到系统关系图上，可以将方框形状添加到系统关系图中。“形状”工具栏包括以下方框形状图标：



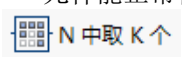
添加表示单个元件的单个方框形状。



添加串联方框形状，该形状表示串联在一起的一组相同元件。所有元件都必须能正常使用，方框才能正常使用。



添加一个并联方框形状，该形状表示并联在一起的一组相同元件。必须至少有一个元件能正常使用，方框才能正常使用。



添加一个“n 中取 k 个”方框形状，该形状表示并联在一起的一组 n 个相同元件。n 中必须至少有 k 个元件能正常使用，方框才能正常使用。

添加一个备用方框形状，该形状表示并联在一起的一组 n 个相同元件。只有 k 个元件**活动**，或是在系统中执行操作。其余**不活动**的元件充当备用元件，这些元件将在 k 个活动元件中有任何元件失效后激活。该方框必须至少包含 k 个正常使用的已激活元件才能正常使用。

添加一个应力共享方框形状，该形状表示并联在一起的 n 个相同元件。元件一次失效一个，其余元件以更快的速率失效。该方框必须至少有一个正常使用的元件，并且该方框必须成功地在其余元件之间重新分配应力，该方框才能正常使用。

添加一个结点方框形状，该形状表示指向结点的方框形状组合。必须至少有指定数量的指向结点的方框形状正常使用，结点才能正常使用。

有关可用于方框形状的设置的信息，请参见第 287 页的““配置”面板”。

JMP PRO “配置”面板

“模拟设置”报表显示在“配置”面板的顶部。每个选定的元件的元件设置显示在“模拟设置”分级显示项下方。选定的事件和动作的配置设置显示在该动作或事件所属的方框形状的下方。

JMP PRO 模拟设置

在“模拟设置”报表中更改模拟的设置，该报表在“配置”面板中提供。提供以下设置：

持续时间 在每次迭代中模拟的时间长度。

时间单位 用于模拟的时间单位。

注意：循环事件和非立即操作使用在“模拟设置”中指定的时间单位。

N 个模拟 模拟中的迭代次数。

种子 （可选）确保再现模拟结果的随机种子。默认情况下，“种子”设置为 0，即不生成再现结果。将分析保存到脚本中时，您输入的随机种子将保存到该脚本中。

警告：若通过右击并选择**运行多线程模拟**来运行模拟，结果不可再现，即便您指定了随机种子也是如此。

JMP PRO 方框设置

选择系统关系图中的某个方框形状以查看其在 “配置”面板中的设置。

每个方框形状 （结点方框除外）都具有 “启用系统免除”选项。若为某个方框选定该选项，则在调用 “启动系统”操作时不会启用该方框。

每个方框形状 （结点方框除外）都具有可确定方框形状的各个元件的随机失效率的失效分布。基本方框的失效分布确定方框的失效率，因为基本方框仅表示一个元件。有关可用的失效分布

选项的详细信息，请参见第 293 页的 “分布选项”。在 “分布”选项下，可以为方框指定 “时间单位”和依赖于分布的参数。

串联和并联

当串联方框中的某一元件失效时，该串联方框即失效。当并联方框的所有元件失效时，该并联方框才失效。以下选项可用于串联和并联方框：

N 指定方框中包含的相同元件数。

N 中取 K 个

“*N* 中取 *K* 个”方框包含 *n* 个相同元件。若正常使用的元件不足 *k* 个时，该方框将失效。以下选项可用于 “*N* 中取 *K* 个”方框：

K 指定方框保持正常使用所需的正常使用元件的最小数目。

N 指定方框中包含的相同元件数。

备用

备用方框包含不活动的从属元件 （称为备用元件） 。活动元件在备用方框内执行操作。不活动元件不在备用方框内执行操作，但会随着活动元件失效而激活，一次激活一个不活动元件。激活过程有时并不成功。在激活某个备用成分时，元件开关可能失效。若 *n* 个相同元件中有不足 *k* 个处于活动状态，则备用方框失效。以下选项可用于备用方框：

K 指定最初活动的元件数。这也是方框保持正常使用所需的活动元件的最小数目。

N 指定方框中包含的相同元件总数。 *k* 和 *n* 之间的差值等于备用元件数。

开关类型 指定在任何活动元件失效后用于激活单个备用元件的机制。

单一开关 方框中存在单一开关。若某个备用元件激活失败，则该方框也随之失效。

各个开关 每个备用元件都对应一个开关。若某个备用元件激活失败，则无法激活该备用元件。备用方框尝试激活下一个备用元件，直到某个备用元件被激活。若其余所有开关都无法正常使用并且活动元件不足 *k* 个，则该方框失效。

开关可靠性 指定当有任何活动元件失效时成功激活备用元件的概率。

备用类型 指定备用元件的状态和失效分布。

冷 备用元件在激活后才会老化。

暖 备用元件根据其不活动期间的次失效分布而老化。备用元件在激活后将根据主失效分布而老化。使用次失效分布可模拟没有在备用方框中执行操作的备用元件上的折算应力。

应力共享

应力共享方框在其元件之间均等分布应力。随着元件失效，仍能正常使用的元件承受的应力会越来越高，因此失效率会增加。

N 指定方框中包含的相同元件总数。

开关可靠性 指定在仍能正常使用的元件之间成功重新分配应力的概率。若重新分配应力失败，则该方框将失效。

应力共享类型 指定如何在正常使用的元件之间共享应力。

基本（默认） 指定在仍能正常使用的元件之间均等共享应力。这种应力共享类型称为**负载共享**。单个元件的特征寿命与共享工作负载的元件数成比例。

定制 指定元件根据“共享公式”选项中显示的 JSL 代码共享应力。“共享公式”定义应力如何随着元件失效而变化。

结点

结点方框不具有失效分布。仅当连接的正常使用的方框形状数目不足指定的最小数目时，结点方框才失效。以下选项可用于结点方框：

最小可用 指定为使结点保持正常使用而必须指向该结点方框的正常使用方框的最小数目。

JMP PRO “添加事件”面板

事件表示模拟中的离散发生次数。您可以使用事件触发操作。对于单个事件可触发的操作数并没有限制。“添加事件”面板中提供以下选项：

注意：某些方框形状不一定具有所有可用的事件类型。

方框失效 在方框失效时出现。若元件路径中断，则系统将设置为 “关闭”状态。

已计划 在指定的循环区间出现。“最大出现次数”选项对该事件在模拟中出现的次数设置限值。默认情况下，“最大出现次数”设置为缺失。在这种情况下，该事件会持续出现，直到模拟结束。请参见第 295 页的 “事件设置”。

基于服役 在元件达到指定寿命时出现。元件寿命是元件一直正常使用和处于活动状态的累积时间。当某个元件失效或其方框被关闭后，寿命即停止增加。“最大出现次数”对该事件在模拟中发生的次数设置限值。默认情况下，该事件会持续出现，直到模拟结束。请参见第 295 页的 “事件设置”。

系统已关闭 在系统设置为 “关闭”或 “关”状态时出现。系统状态的更改可能是有意的也可能是无意的。

初始化 在每次模拟迭代开始时出现。使用该事件可安排在系统运行前必须完成的操作。

创建事件

1. 要创建事件，请在“系统关系图”中选择一个方框形状，以便在“系统关系图”底部显示“添加事件”和“添加操作”面板。
2. 选择 “添加事件”面板中的某一选项，为选定的方框形状定义事件。

图 12.6 方框失效事件

一个橙色方块，表示在元件上方显示的新事件。请注意，选定的事件的右侧会显示一个连接箭头。

JMP PRO “添加操作”面板

操作可以定义元件和由连接的事件或连接的操作所触发的系统行为。一完成前一操作即触发连接的操作。“添加操作”面板中提供以下选项：

注意：某些方框形状不一定具有所有可用的操作类型。

表 12.1 操作选项

操作名称	可使用该操作的方框	起始行为	完成行为
关闭系统	全部		关闭所有方框。若系统尚未进入“关闭”状态，则将系统设置为“关”状态。
启动系统	全部		启动所有尚未失效的方框。若元件路径未中断，则系统设置为“开”状态。该操作不会启用已选定“启用系统免除”选项的方框。
全新更换	全部	若元件路径中断，则关闭原始方框并将系统设置为“关闭”状态。	方框变为新的方框。其寿命将重置，方框将被启动。若元件路径未中断，则系统设置为“开”状态。
最少量的维修	“基本”和“串联”	等同于“全新更换”操作的起始行为。	启动该方框。若元件路径未中断，则系统设置为“开”状态。不重置该方框的寿命。
开启区组	全部	若方框处于“关闭”状态或当前已从系统中删除，则取消该操作。	启动该方框会使“启动计数”增加 1。
关闭区组	全部		关闭该方框。若元件路径中断，则系统设置为“关闭”状态。

表 12.1 操作选项 (续)

操作名称	可使用该操作的方框	起始行为	完成行为
删除区组	全部		从系统中删除该方框。若元件路径中断，则系统设置为“关闭”状态。
安装新件	全部	等同于“全新更换”操作的起始行为。	关闭该方框。重置该方框的寿命。
安装已用件	基本	等同于“全新更换”操作的起始行为。	关闭该方框。您要为该方框指定新的寿命和失效分布。
更改分布	基本	关闭该方框。若元件路径中断，则系统设置为“关闭”状态。	更改该方框的失效分布。使用该行为可模拟一段时间内的操作变化或针对该方框的累积损坏。
检查失效	全部		若该方框已失效或已从系统中删除，则触发连接的操作。
条件	全部		若指定的条件脚本成立，则触发连接的操作。
计划	全部		触发指定间隔处的连接操作。您可以限制已计划间隔的数目，或允许该操作持续到模拟迭代结束。

创建操作

1. 要创建操作，请在“系统关系图”中选择一个方框形状，以便显示位于“系统关系图”底部的“添加事件”和“添加操作”面板。
2. 选择 “添加操作”面板中的某一选项，为选定的方框形状定义操作。

   将在选定的方框形状上方创建一个蓝色操作方块。出现连接的事件时即触发操作。创建触发您定义的操作的事件。
3. 选择 “添加事件”面板中的某一选项，为选定的方框形状定义事件。

   将在方框形状上方创建一个橙色事件方块。请注意事件右侧的连接箭头。
4. 点击连接箭头并将其拖至您在第 2 步中创建的蓝色操作方块。

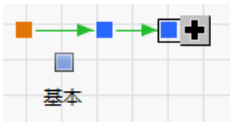
   将有一个绿色箭头将该事件与操作方块连接起来。当该事件出现在模拟迭代中时，它就会触发连接的操作。
5. 选择您在第 2 步中创建的蓝色操作方块。

   通过使用选定事件的右侧出现的加号，您可以连接完成前一操作后即触发的其他操作。
6. 点击加号并将其拖至系统关系图中的空白区域。

随即显示可用操作的列表。

7. 从操作列表中选择某一选项，以创建与第一个操作连接的操作。

图 12.7 创建操作



将有一个绿色箭头将这两个操作连接起来。完成第一个操作后即触发第二个操作。

JMP PRO “可修复系统模拟”平台选项

“可修复系统模拟”红色小三角菜单包含以下选项：

“保存”和“另存为” 支持您将“可修复系统模拟”保存到一个 JMP 脚本语言 (JSL) 脚本，在 JMP 中打开该脚本时会自动执行该脚本。请参见《Scripting Guide》，了解有关“自动提交”脚本的详细信息。

注意：“保存”和“另存为”红色小三角选项分别等同于选择**文件 > 保存**和**文件 > 另存为**。这些选项位于红色小三角菜单中，以便于使用。

导入元件分布设置 支持您从数据表中导入系统关系图的配置设置。该表必须包含用于元件名称、分布以及一个或多个参数的列。参数个数取决于指定分布。

注意：导入表中的字符串必须与系统关系图中的字符串完全匹配。

JMP PRO 用于方框项的选项

要查看选定设计或子系统中元件的设置，请执行以下操作：

- 要查看某个方框形状的“配置”设置，请在系统关系图中选择该方框形状。
- 要查看不止一个方框形状的“配置”设置，请使用箭头工具，在按 Ctrl 的同时点击多个方框形状来选择多个方框形状。

JMP PRO 分布选项

每个方框形状根据其指定的失效分布随机失效。此外，您可以指定分布的时间单位。方框形状的时间单位可以不同于您在模拟设置中指定的时间单位选项。“启动计数”选项基于单个方框形状设置为“开”状态的次数。

可用失效分布列在表 12.2 中。

表 12.2 分布和其他参数

属性类型	所需的输入信息
指数	Theta
Weibull	Alpha、Beta
对数正态	位置、尺度
对数 Logistic	位置、尺度
Fréchet	位置、尺度
广义 Gamma	mu、sigma、lambda
缺陷子总体 Weibull	Alpha、Beta、缺陷概率
缺陷子总体对数正态	位置、尺度、缺陷概率
缺陷子总体对数 Logistic	位置、尺度、缺陷概率
缺陷子总体 Fréchet	位置、尺度、缺陷概率
非参数	数据或数据文件
估计	估计的分布、数据或数据文件

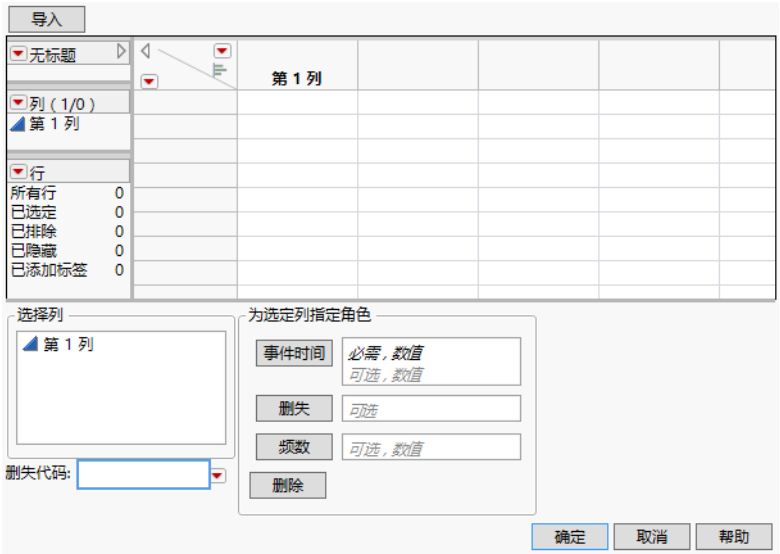
要查看这些失效分布的公式和参数化信息，请参见“寿命分布”一章中第 72 页的“分布”。

JMP PRO 指定非参数或估计分布

“分布”下的“非参数”和“估计”选项支持您近似任意分布。您可以手动输入数据或导入包含数据的文件。这些数据用于近似分布。

选择“非参数”或“估计”后，点击“数据”旁边的 图标。随即显示“提供数据”窗口，您可以在其中输入数据或导入数据文件。导入或输入数据后，即可使用这些数据计算元件的分布。

图 12.8 “提供数据”窗口



要从文件导入数据，请执行以下操作：

1. 点击 ... 图标前，打开包含要导入数据的 JMP 数据表。
2. 点击 “数据”旁边的 ... 图标。
3. 在 “提供数据”窗口中，点击**导入**。
   随即显示 “选择数据表”窗口。
4. 从 “数据表”列表中，选择一个数据表。
5. 点击**确定**。
6. 在数据网格下方的面板中，指定表示事件时间数据的列。
7. 按需指定 “删失”列和 “频数”列。
8. 在 “提供数据”窗口中，点击**确定**。

要手动输入数据，请执行以下操作：

1. 创建“事件时间”数据列。
2. 按需创建 “删失”列和 “频数”数据列。
3. 在这些列中输入数据。
4. 在数据网格下方的面板中，指定哪些列表示 “事件时间”数据以及 “删失”和 “频数”数据 （若适用）。请参见上面的第 5 步和第 6 步。
5. 在 “提供数据”窗口中，点击**确定**。

JMP PRO 事件设置

选择系统关系图中的某个事件以查看其在 “配置” 面板中的设置。您可以更改 “事件名称” 设置以便将该事件与其他事件区分开来。只有 “已计划” 和 “服役” 事件具有其他设置，这些设置如下所列：

已计划

已计划事件具有以下选项：

循环区间 指定事件各次发生之间的时间间隔长度。

最大出现次数 指定该事件可出现的最大次数。

服役

服役事件具有以下选项：

服役 指定发生该事件之前等待的元件运行时间的间隔。

注意：仅当系统设置为 “开” 状态并且元件可正常使用时，才会积累元件运行时间。

最大出现次数 指定该事件可出现的最大次数。

JMP PRO 操作设置

选择系统关系图中的某个操作以查看其在 “配置” 面板中的设置。您可以更改 “操作名称” 设置以便将该操作与其他操作区分开来。默认情况下，操作立即完成。“完成时间选项” 菜单中提供非立即完成时间选项。

图 12.9 操作设置

以下选项显示在 “完成时间选项” 列表中：

立即 （默认） 指定在开始和完成行为之间无时间传递。

常数 指定时延始终为指定的 “完成时间”。

选择 指定从指定的逗点分隔值列表中随机选取时延。

均匀 指定从具有指定的最小值和最大值的均匀分布中随机选取时延。

三角 指定从具有指定的最小值、众数和最大值的三角形分布中随机选取时延。

正态 指定从具有指定的均值和标准差的正态分布中随机选取时延。

JMP PRO 可修复系统模拟结果

- “结果表”
- “结果分析器”
- “可修复系统模拟结果选项”
- “RSS 结果分析器点估计刻画器”

JMP PRO 结果表

当您点击“开始”方框下方的绿色箭头以运行模拟后，模拟结果将显示在数据表中。该表说明在每次模拟迭代中出现的事件和后续操作。

结果表包含以下列：

模拟 ID 标识事件或操作所属的模拟迭代。

时间 提供事件或操作在模拟中发生的确切时间。

对象 提供事件或操作连接到的方框形状的名称；若为系统事件和操作，则提供系统的名称。

表述 提供发生的事件或操作的名称。

状态 提供对象在该确切时间的状态。每个操作的初始化和终止都分别用“开始”和“完成”来表示。

注释 提供关于某个操作的补充说明。每个模拟迭代的末尾都用“结束”来表示。

图 12.10 RSS 结果表

	模拟 ID	时间	对象	表述	状态	注释
1	1	0	Spare	Initialization Sp...	开始	
2	1	0	Spare	Initialization Sp...	关	
3	1	0	Spare	Initialization Sp...	完成	
4	1	65.024584796	Tire 2	Tire 2 is unrepai...	关闭	
5	1	65.024584796	系统	关闭系统	关闭	Unintended
6	1	65.024584796	Engine	关闭按系统分区组	关	
7	1	65.024584796	Transmission	关闭按系统分区组	关	
8	1	65.024584796	Battery	关闭按系统分区组	关	
9	1	65.024584796	Tire 1	关闭按系统分区组	关	
10	1	65.024584796	Tire 3	关闭按系统分区组	关	
11	1	65.024584796	Tire 4	关闭按系统分区组	关	
12	1	65.024584796	Tire 2	Replace Tire 2	已删除	
13	1	65.024584796	Tire 2	Replace Tire 2	开始	

请注意，图 12.10 中的第一个条目是 Initialization Spare in Trunk 操作的开始，该操作具有立即完成时间。该操作关闭 Spare 元件，然后在**时间**仍为 0 时完成。

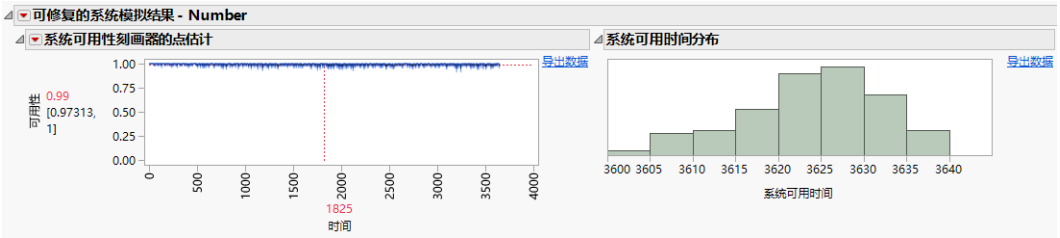
下一事件是在第一次模拟迭代经过大约 65 天后发生的 Tire 2 is Unrepairable。在第五行中，系统意外设置为“关闭”状态，因为只有三个轮胎能正常使用。Need 4 Tires to Drive 结点要求至少有四个轮胎（包括备胎）才能正常使用。Tire 2 is Unrepairable 事件同时触发了 Replace Tire 2 操作和 Use Spare 操作。第 11 行中触发了 Drive with Spare 操作，系统被设置为“开”状态。

请注意，Tire 2 失效的同时被 Spare 更换。在系统被设置为“关闭”状态和系统恢复“开”状态两个时间点之间未占用任何时间。由于后续操作具有立即完成时间，Tire 2 is Unrepairable 事件未造成任何系统中断时间。结果分析器显示系统中断时间会在操作为“非立即”时累积。

JMP PRO 结果分析器

点击结果数据表中的“Launch Repairable Systems Simulation Results Explorer”脚本旁边的绿色箭头后，将会显示一个包含模拟结果分析的报表。

图 12.11 部分 RSS 分析器报表



默认情况下，该报表包含两类图形。第一类图形（显示在图 12.11 的左侧）是点估计图形。点估计图形显示针对水平轴上的模拟迭代时间标绘的垂直轴上的聚合模拟结果。水平轴的范围是您在模拟设置中指定的持续时间。某一给定时间可用的系统的估计概率在垂直轴旁边显示为红色。点估计的下方是该点估计的 95% 置信区间。

提示：要查看模拟迭代中特定时间的系统可用性的点估计，请点击水平轴下方的红色数字。指定模拟持续时间范围中的某个时间，然后按 Enter 键。

第二类图形 （显示在图 12.11 的右侧）是模拟迭代中的系统可用时间的直方图。直方图中的箱代表以下之一：

- 总模拟时间。
- 模拟持续时间的比例。

提示：选择图形旁边的**导出数据**选项以创建包含图形使用数据的数据表。

默认情况下，该报表包含以下图形：

系统可用性的点估计刻画器 显示模拟迭代期间，系统处于 “开”或 “关”状态的估计概率随时间变化的刻画器图形。

系统可用时间分布 显示每次模拟迭代系统可用总时间的直方图。

系统可用性分布 显示每次模拟迭代系统可用总时间除以模拟持续时间的直方图。该分布与“系统可用时间分布”相同，只不过该分布显示为模拟持续时间的比例。

系统服役概率的点估计刻画器 显示模拟迭代期间，系统处于“开”状态的估计概率随时间变化的刻画器图形。

系统服役时间分布 显示每次模拟迭代系统处于“开”状态的总时间的直方图。

系统服役概率分布 显示每次模拟迭代系统处于“开”状态的总时间除以模拟持续时间的直方图。该分布与“系统服役时间分布”相同，只不过该分布显示为模拟持续时间的比例。

系统计划外中断的点估计刻画器 显示模拟迭代期间，系统处于“关闭”状态的估计概率随时间变化的刻画器图形。

系统计划外中断分布 显示每次模拟迭代系统处于“关闭”状态的总时间的直方图。

系统计划外中断百分比分布 显示每次模拟迭代系统处于“关闭”状态的总时间除以模拟持续时间的直方图。该分布与“系统计划外中断分布”相同，只不过该分布显示为模拟持续时间的比例。

JMP PRO 可修复系统模拟结果选项

“可修复系统模拟结果”红色小三角菜单包含以下选项：

元件可用性的点估计 对于系统关系图中的每个方框形状，该选项都显示以下报表：

- 可用性的点估计刻画器
- 可用时间分布
- 可用性分布
- 计划外中断的点估计刻画器
- 计划外中断时间分布
- 计划外中断分布

按周期划分的系统可用性 显示一个图形，该图形支持您检查指定时间期间的平均系统可用性。使用“按时间”面板可将时间期间定义为时间的间隔。使用“按事件”面板可将时间期间定义为一个事件的指定出现次数。点击**执行**后，该报表显示每个期间的平均系统可用性条形图。

按元件划分的系统总关闭时间的箱线图 显示个别元件导致系统处于“关闭”或“关”状态的总时间的箱线图。元件按照从最长中断时间到最短中断时间的贡献来排序，与 Pareto 图类似。

按元件划分的系统计划外中断总时间的箱线图 显示各个元件导致系统处于“关闭”状态的总时间的箱线图。元件按照从最长中断时间到最短中断时间的贡献来排序，与 Pareto 图类似。

请参见《使用 JMP》获取有关下列选项的信息：

恢复 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

JMP PRO RSS 结果分析器点估计刻画器

“可修复系统模拟结果分析器”报表中的每个点估计刻画器都有一个红色小三角菜单，其中包含以下选项：

置信区间 显示或隐藏点估计图形的 95% 置信区间。

重置因子网格 打开 “因子设置”窗口，可在该窗口中修改点估计图形的参数。有关设置因子网格的详细信息，请参见 《刻画器指南》。

因子设置 提供影响“因子网格”的其他选项。请参见《刻画器指南》，了解有关“因子设置”的详细信息。

第 13 章

生存分析

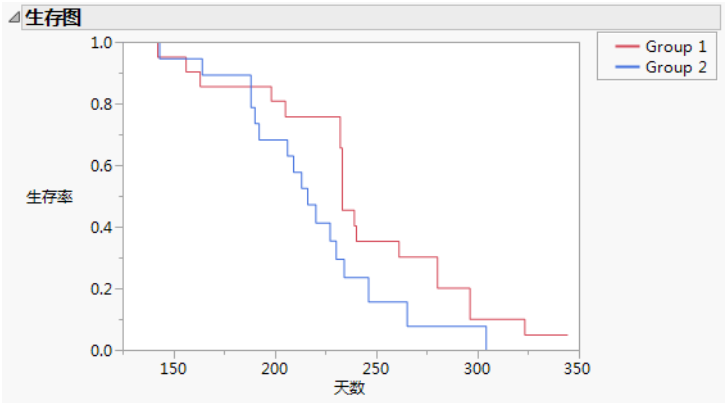
分析生存时间数据

生存数据包含直到发生特定事件的持续时间，有时称为**事件时间**响应数据。事件通常是指失效，如发动机故障或患者死亡。若在观测的研究结束前未发生事件，则我们称观测为**删失的**。

“生存”平台拟合表示事件时间（或失效时间）的单个 Y 变量。使用“生存”平台来检查失效时间的分布。

提示：要拟合解释性模型，请使用“拟合参数生存”平台或“拟合比例风险”平台。

图 13.1 生存图的示例



目录

“生存分析”平台概述 303
生存分析的示例 303
启动“生存”平台 305
生存图 305
“生存”平台选项 306
指数、Weibull 和对数正态图以及拟合 308
拟合分布图 312
竞争原因 312
“生存”平台的更多示例 313
生存分析的示例 313
竞争原因的示例 315
区间删失的示例 317
生存分析的统计报表 318

“生存分析”平台概述

出于以下两个原因，需要使用专门的方法来分析生存数据：

1. 生存时间通常具有专门的非正态分布，如指数、Weibull 和对数正态。
2. 一些数据可能是删失的。

将非参数 Kaplan-Meier 方法用于一组或多组**完整**或**右删失**数据来计算生存函数。完整数据没有删失值。右删失是指您不知道精确的生存时间但是知道它大于指定的值。当研究结束但并非所有单元都发生故障或者在完成研究前患者必须退出研究时，就会发生右删失。若不在分析中引入偏倚，将无法忽略删失的观测。生存模型的元素包括：

- 一个时间，指示还要多长时间单元（或患者）才会经历事件或删失。时间是模型响应 (Y)。
- 一个指示符，表示观测值是经历了事件还是删失的。JMP 使用以下约定：删失单元的代码为 1，非删失事件的代码为 0。
- 解释变量（若使用回归模型）。
- 区间删失是指数据点位于两个值构成的区间中的情形。若需要区间删失，则两个 Y 变量指定界定事件时间的上限和下限。

用于可靠性和生存数据的常见术语包括：寿命、生存、失效时间、事件时间和持续时间。

“生存”平台计算一个或多个组的乘积限 (Kaplan-Meier) 生存函数估计值。可将其用作完整分析或用作探索性分析，以便获取关于更复杂的模型拟合的信息。“Kaplan-Meier 生存”平台完成以下工作：

- 显示每个组的估计生存函数图。整个样本的估计生存函数图是可选项。
- 计算并列出每个组以及合并样本的生存函数估计值。
- 显示指数、Weibull 和对数正态诊断失效图，以图形方式检查使用这些分布是否适合进行进一步的回归建模。一经要求便可得到参数估计值。
- 计算对数秩和广义 Wilcoxon 卡方统计量，以在不同组之间进行估计的生存函数的齐性检验。
- 分析竞争原因、提示失效原因变量，并为对应于每个原因的删失模式估计 Weibull 失效时间分布。

生存分析的示例

采用一个实验来探索两个治疗组中接触致癌物的老鼠的生存时间特点。数据位于 Rats.jmp 样本数据表中。该示例中的事件是死亡。目标是查看一个治疗组中的老鼠是否比另一个治疗组中的老鼠活得更长 （活更多天数）。

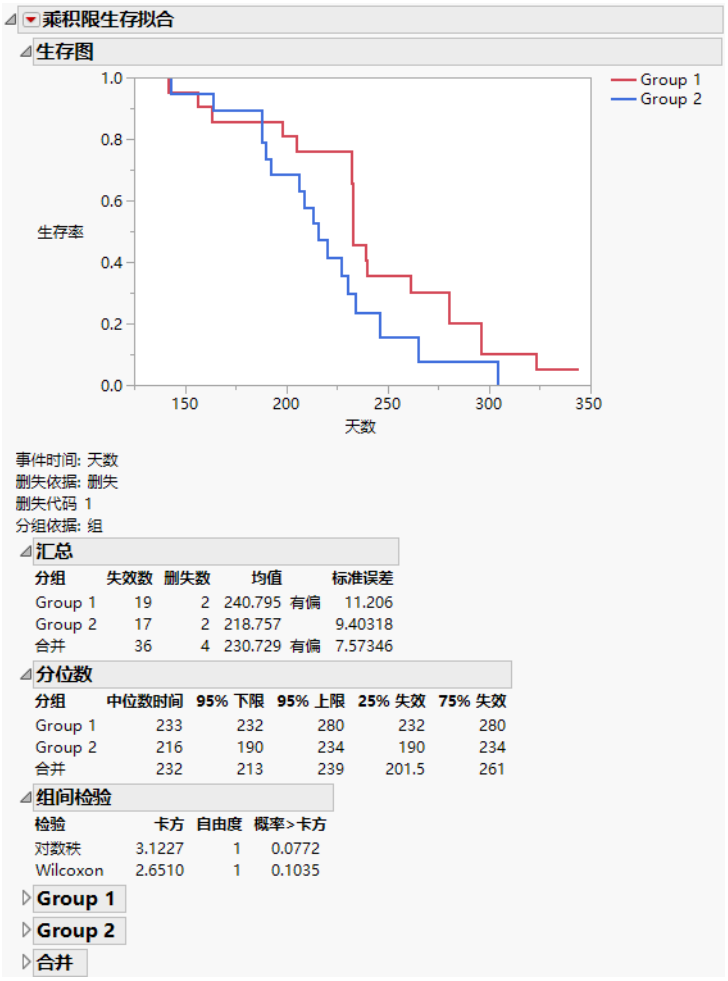
1. 选择**帮助 > 样本数据库**，然后打开 Rats.jmp。

   “天数”列中的数据是生存时间。请注意，一些观测是删失的。

2. 选择**分析 > 可靠性和生存 > 生存**。

3. 选择**天数**并点击 **Y，事件时间**。
4. 选择**组**并点击**分组**。
5. 选择**删失**并点击**删失**。
6. 点击**确定**。

图 13.2 Rats.jmp 数据的 “生存图”



看起来治疗组 1 中的老鼠比治疗组 2 中的老鼠活得更长。

启动 “生存” 平台

通过选择**分析** > **可靠性和生存** > **生存**来启动 “生存” 平台。

图 13.3 “生存” 启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“生存” 启动窗口包含以下选项：

Y，事件时间 标识事件时间或删失时间。若您有区间删失，指定两个 Y 变量，分别表示上限和下限。

分组 将数据分为单独拟合的组。

删失 标识删失的值。在删失代码框中输入标识删失的值。该列在以下条件下可以包含两个以上的不同值：

- 所有删失行具有在 “删失代码” 框中所输入的值。
- 非删失行具有 “删失代码” 框中之外的值。

频数 指示当记录多个单元时其值是每行的观测频数的列。若值为 0 或正整数，则该值表示每行的观测频数或计数。

依据 为分类或分组变量的每个水平执行单独的分析。

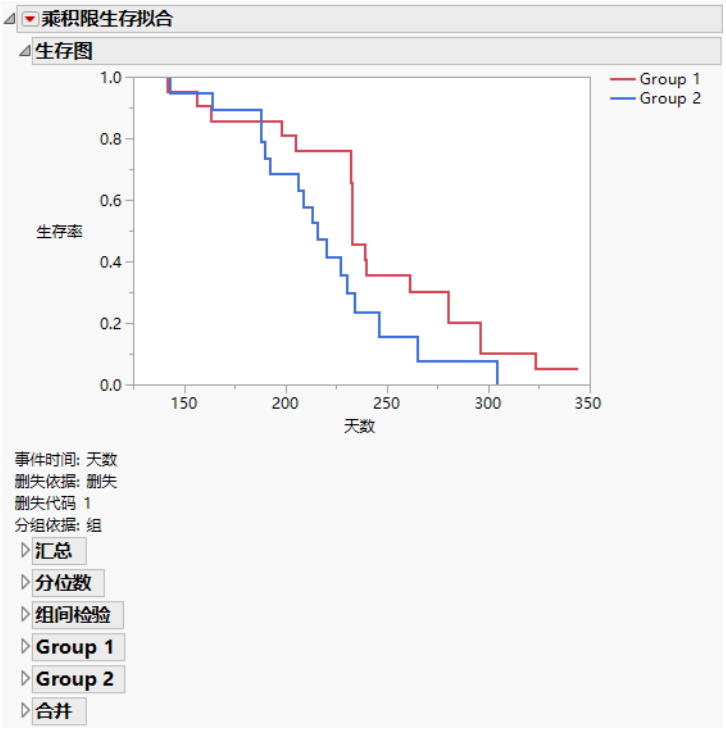
标绘失效图而非生存图 显示失效概率图而非与它相反的生存概率图。

删失代码 标识 “删失” 列中指定右删失观测的值。默认值为 1。

生存图

“生存” 平台显示每个组的估计生存函数的叠加阶梯图。图例按颜色和线条类型标识各组。

图 13.4 生存图



图下面的报表显示生存时间的汇总统计量和分位数。计算各组中每个观测的估计生存时间。计算合并样本的生存时间。当有多个组时，统计检验比较生存曲线。

若在时间 0 出现任何失效，则当您从“乘积限生存拟合”红色小三角菜单中请求分布拟合时，将显示“在时间 0 失效”报表。该报表包含分组变量每个水平的 0 时失效计数表。若未指定分组变量，则表中有一行标记为“合并”。表中还包含一个标签为**概率时间>0**的列。该列是每组中失效时间非零的观测的比例。

“生存”平台选项

“乘积限生存拟合”红色小三角菜单包含以下选项：

生存图 显示每个组的叠加生存图。

失效图 按照可靠性文献的传统，显示每个组的叠加失效图 （在一段时间上失效的比例） 。失效图反转垂直轴来显示失效数而非生存者数。

注意: “失效图”选项替换了旧版本的 JMP 中的“反转 Y 轴”选项（该选项在脚本中仍提供）。

图选项 包含以下选项：

注意：前七个选项（显示点、显示 Kaplan Meier、显示合并、显示置信区间、显示联合置信区间、显示着色点态置信区间和显示着色联合置信区间）和最后两个选项（拟合生存置信区间、拟合失效置信区间）与初始生存图和失效图有关。其他五个（阶梯中部位置分位数点、连接分位数点、拟合分位数、拟合分位数置信区间线、拟合着色分位数置信区间）只与分布图有关。

显示点 显示生存图每步的样本点。失效显示在阶梯底部，删失由阶梯之上的点表示。

显示 Kaplan Meier 显示 Kaplan-Meier 曲线。默认启用该选项。

显示合并 在“生存图”中显示合并组的生存曲线。

显示置信区间 在“生存图”上显示各组的 95% 点态置信带，当该选项与显示合并选项一起使用时，则显示合并图的 95% 点态置信带。

显示点，显示合并 当您选择“显示点”和“显示合并”选项时，总样本或合并样本的生存图显示为灰线。点也显示在每个组的图阶梯中。

显示联合置信区间 在图上显示所有组的联合置信带。Meeker and Escobar (1998, ch. 3) 讨论了点态和联合置信区间，以及在寿命分析中涉及联合置信区间的动因。

阶梯中部位置分位数点 更改绘图位置以使用**修改的** Kaplan-Meier *绘图位置*。这些绘图位置等同于取 Kaplan-Meier 曲线的阶梯中部位置而非阶梯底部位置。推荐使用该选项，因此默认启用它。

连接分位数点 在图中显示线条。默认启用该选项。

拟合分位数 在拟合的 Weibull、对数正态或指数图上显示直线拟合。默认启用该选项。

拟合分位数置信区间线 显示拟合的 Weibull、对数正态或指数图的 95% 置信带。

拟合着色分位数置信区间 将拟合的 95% 置信带显示为着色区域或虚线。

拟合生存置信区间 在“生存图”上显示拟合分布的置信区间。

拟合失效置信区间 在“失效图”上显示拟合分布的置信区间。

指数图 为每个组绘制时间 - 累积指数失效概率图。近似呈线性的线条从经验上表明使用指数模型适合进行进一步的分析。例如，在图 13.5 中，“指数图”中的 Group 1 和 Group 2 对应的线条是曲线而非直线。这表明指数分布不适合该数据。请参见第 308 页的“指数、Weibull 和对数正态图以及拟合”。

指数拟合 生成“指数参数”表并在“指数图”中生成指数累积分布函数的线性拟合（图 13.5）。参数 Theta 对应于失效时间均值。请参见第 308 页的“指数、Weibull 和对数正态图以及拟合”。

Weibull 图 为每个组绘制 log(时间)- 累积 Weibull 失效概率图。具有近似平行的直线的 Weibull 图指示 Weibull 生存分布模型可能适合用于进一步分析。请参见第 308 页的“指数、Weibull 和对数正态图以及拟合”。

Weibull 拟合 在“Weibull 图”中生成 Weibull 累积分布函数的线性拟合和两个常用形式的 Weibull 估计值。这些估计值显示在“极值参数估计值”表和“Weibull 参数估计值”表中（图 13.5）。Alpha 参数是失效时间分布的 0.632 分位数。“极值”表显示同一拟合的另一参

数化，其中 Lambda = ln(Alpha)，Delta = 1/Beta。请参见第 308 页的“指数、Weibull 和对数正态图以及拟合”。

对数正态图 为每个组绘制 log(时间)- 累积对数正态失效概率图。具有近似平行的直线的对数正态图指示对数正态分布适合用于进一步分析。请参见第 308 页的“指数、Weibull 和对数正态图以及拟合”。

对数正态拟合 在“对数正态图”中生成对数正态累积分布函数的线性拟合以及“对数正态参数估计值”表，如图 13.5 中所示。 Mu 和 Sigma 对应于服从正态分布的时间变量自然对数的均值和标准差。请参见第 308 页的“指数、 Weibull 和对数正态图以及拟合”。

拟合分布图 与拟合选项一起使用以显示对应于以下拟合分布的三个图：“生存”、“密度”和“危险率”。若您未执行拟合，则不显示任何图。请参见第 312 页的“拟合分布图”。

竞争原因 使用指定的原因来指示失效事件并使用其他原因来指示删失，以便执行 Weibull 模型的估计。拟合的分布在“生存图”中显示为虚线。请参见第 312 页的“竞争原因”。

估计生存概率 估计您指定的时间值所对应的生存概率。

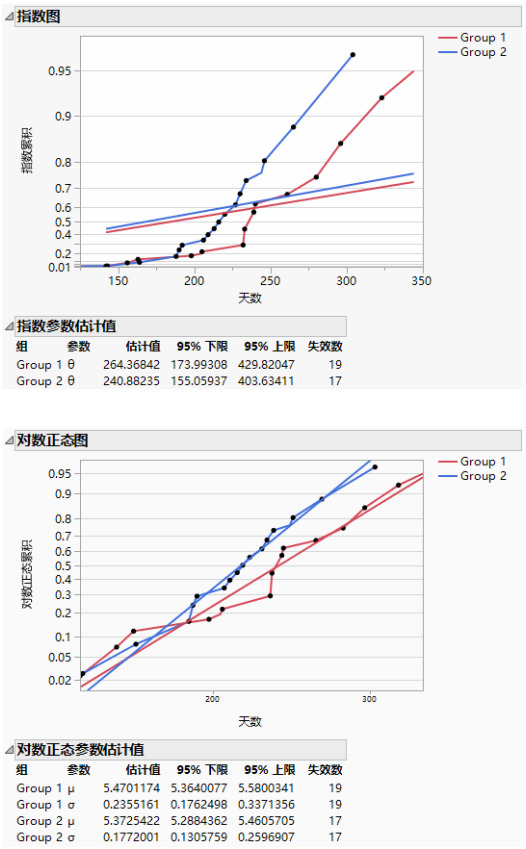
估计时间分位数 估计您指定的每个生存概率所对应的时间分位数。

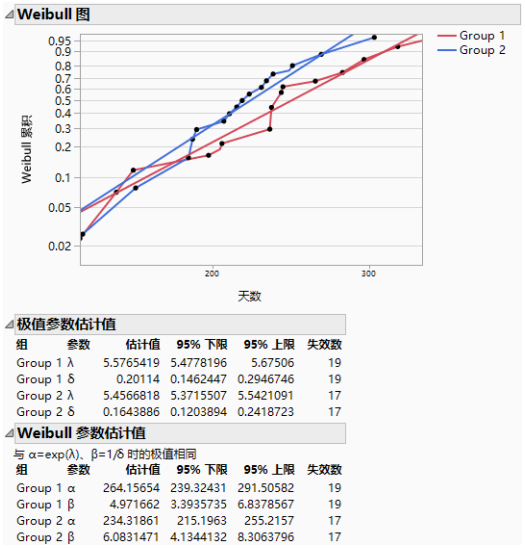
保存估计值 创建一个数据表，它包含生存估计值和失效估计值、置信区间以及其他分布统计量。

指数、Weibull 和对数正态图以及拟合

对于“生存”平台中的三种受支持分布中的每一种分布，都提供相应的图命令和拟合命令。使用图命令可查看事件标记是否遵循直线轨迹。当分布拟合适合数据时，标记倾向于遵循直线轨迹。然后，使用拟合命令来估计参数。

图 13.5 指数、 Weibull 和对数正态图以及报表





下表显示要绘制什么来生成该分布的直线拟合：

表 13.1 分布的直线拟合

分布图	水平轴	垂直轴	解释
指数	时间	-log(S)	斜率是 1/theta
Weibull	log(时间)	log(-log(S))	斜率是 beta
对数正态	log(时间)	Probit(1-S)	斜率是 1/sigma

注意： S = 生存分布的乘积限估计值。

指数

指数分布是用于对事件时间数据建模的最简分布。指数分布只有一个参数 theta。它是危险率为常数的分布，不记得生存时间长短对事件可能性的影响。参数 theta 是预期寿命。

Weibull

Weibull 分布是用于对事件时间数据建模的最常用的分布。Weibull 分布可有两个或三个参数。“生存”平台拟合两参数 Weibull 分布。不同作者有很多不同方式来参数化该分布（表 13.2）。JMP 报告其中两种参数化：Weibull alpha-beta 参数化和基于最小极值分布的参数化。

alpha-beta 参数化（显示在“Weibull 参数估计值”报表中），在可靠性文献 (Nelson 1990) 中广泛使用。将 alpha 解释为 63.2% 的单元失效时的分位数。beta 参数确定危险率如何随时间变化。若 beta > 1，危险率将随时间提高；若 beta < 1，则危险率随时间下降；若 beta = 1，则危险率不随时间变化。危险率函数为常数的 Weibull 分布等同于指数分布。

lambda-delta 极值参数化显示在“极值参数估计值”报表中。该参数化有时在统计意义上是需要的，因为它将 Weibull 分布置于位置-尺度的设置中 (Meeker and Escobar 1998, p. 86)。位置参数为 lambda，尺度参数为 delta。与 alpha-beta 参数化相关的是，lambda 等于 alpha 的自然对数，delta 等于 beta 的倒数。因此，delta 参数确定危险率如何随时间变化。若 delta > 1，危险率将随时间降低；若 delta < 1，则危险率随时间提高；若 delta = 1，则危险率不随时间变化。危险率函数为常数的 Weibull 分布等同于指数分布。

表 13.2 依照 JMP 中的 alpha 和 beta 的各种 Weibull 参数

JMP Weibull	alpha	beta
Wayne Nelson	alpha=alpha	beta=beta
Meeker 和 Escobar	eta=alpha	beta=beta
Tobias 和 Trindade	c = alpha	m = beta
Kececioglu	eta=alpha	beta=beta
Hosmer 和 Lemeshow	exp(X beta)=alpha	lambda=beta
Blishke 和 Murthy	beta=alpha	alpha=beta
Kalbfleisch 和 Prentice	lambda = 1/alpha	p = beta
JMP 极值	lambda=log(alpha)	delta=1/beta
Meeker 和 Escobar s.e.v.	mu=log(alpha)	sigma=1/beta

对数正态

对数正态分布也是极常用的用于对事件时间数据建模的分布。对数正态分布等同于对值取对数后服从的正态分布。若您要拟合数据的正态分布，可以取它的 exp() 并使用对数正态分布对数据建模。请参见“拟合参数生存”一章中第 331 页的“拟合参数生存的更多示例”。

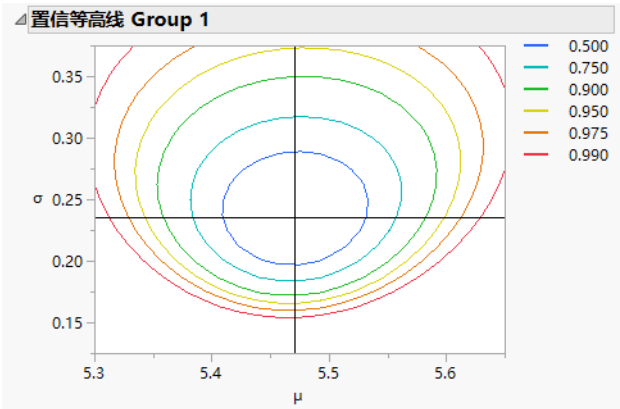
其他选项

要查看指数、Weibull 和对数正态拟合的其他选项，请按 Shift 键，点击“乘积限生存拟合”旁边的红色小三角并选择所需的拟合。

使用这些选项可以执行以下任务：

- 设置限值的置信水平。
- 设置 theta（指数拟合）、sigma（对数正态拟合）或 beta（Weibull 拟合）的约束值。请参见第 311 页的“WeiBayes 分析”。
- 获取 Weibull 和对数正态拟合（没有约束值时）的置信等高线图。

图 13.6 置信等高线图



WeiBayes 分析

在拟合相应分布时，JMP 可以约束 Theta（指数）、Beta (Weibull) 和 Sigma（对数正态）参数的值。在 WeiBayes 情形中需要该功能，例如：

- 有很少的失效或没有失效
- 有 beta 的现有历史值
- 仍需要估计 alpha

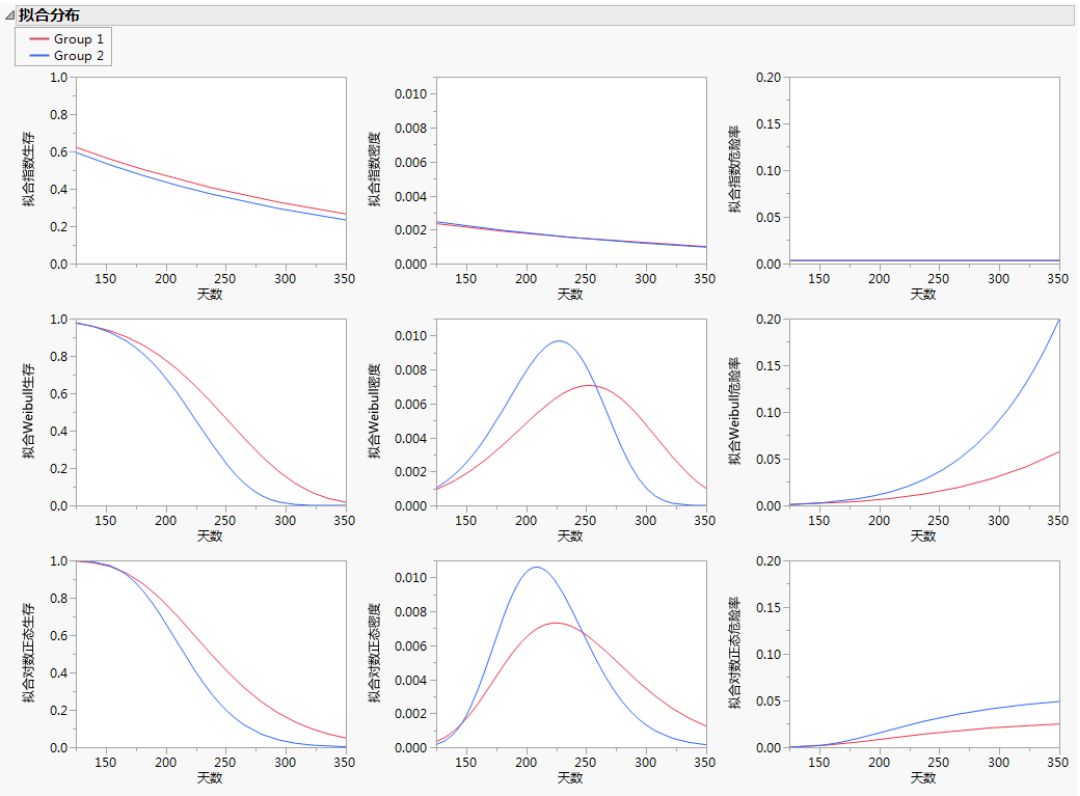
有关 WeiBayes 情形的详细信息，请参见 Abernethy (1996)。

在没有失效时，标准方法是在末尾添加失效。然后，估计值将反映 alpha 值的下限类型而非真实估计值。不过，WeiBayes 功能允许真实估计值。

拟合分布图

使用“拟合分布图”选项可以查看指数、Weibull 和对数正态分布的生存、密度和危险率图。这些图共享相同的轴尺度以便可以轻松比较各分布。

图 13.7 三种分布的拟合分布图



通过使用图形脚本，可以将这些图转换为其他图形。要复制图形，请在要复制的图中右击并选择**编辑 > 复制框架内容**。在目标图中右击并选择**编辑 > 粘贴框架内容**。

竞争原因

有时系统中有多个失效原因。例如，假定制造过程有几个阶段，任何阶段的失效都可能导致整个系统失效。若不同原因是独立的，可以用每个原因的生存分布估计值对失效时间建模。通过将所有不来自给定原因的事件时间视为删失观测，执行针对该原因的删失估计。

“竞争原因”红色小三角菜单包含以下选项：

忽略原因 删除指定的原因值并重新计算生存估计值。

保存原因坐标值 将名为 log(–log(Surv)) 的新列添加到当前表。该信息通常用于针对具有分组变量 （例如用于失效类型的代码）的时间变量绘图。

Weibull 线 将 Weibull 线添加到图中。

危险率图 添加危险率图。

模拟 创建一个新数据表，它包含通过数据估计的 Weibull 分布的时间和原因信息。

“生存”平台的更多示例

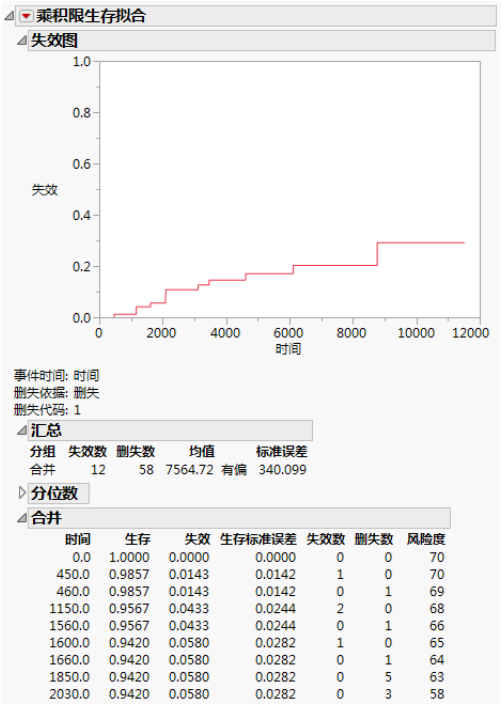
- “生存分析的示例”
- “竞争原因的示例”
- “区间删失的示例”

生存分析的示例

Nelson (1982, p. 133) 和 Meeker and Escobar (1998, app. C1) 对柴油发动机风扇的失效进行了研究。

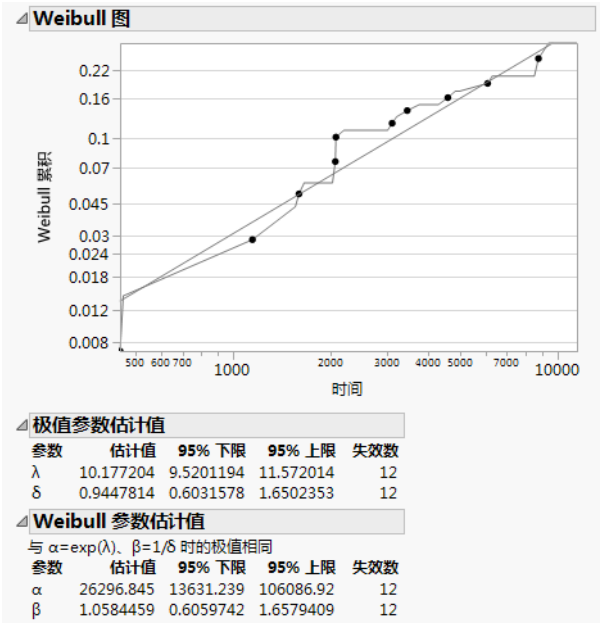
1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Fan.jmp。
2. 选择**分析** > **可靠性和生存** > **生存**。
3. 选择**时间**并点击 **Y，事件时间**。
4. 选择**删失**并点击**删失**。
5. 选中**标绘失效图而非生存图**复选框。
6. 点击**确定**。

图 13.8 风扇初始输出



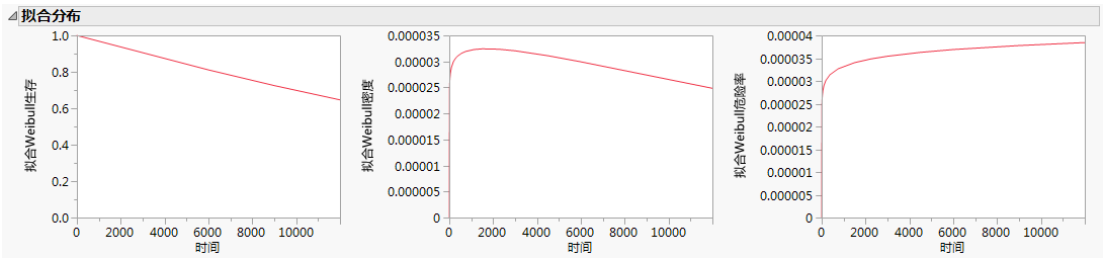
请注意失效概率随时间增加。通常下一步是探索分布拟合（如 Weibull 模型）。点击“乘积限生存拟合”旁边的红色小三角并选择 Weibull 图和 Weibull 拟合。

图 13.9 风扇数据的 Weibull 输出



因为拟合是合理的且 Beta 估计值接近 1，您可以推断出它看起来类似指数分布，该分布的危险率为常数。点击“Weibull 图”红色小三角并选择**拟合分布图**。随即显示 Weibull 拟合的三个视图。

图 13.10 拟合分布图



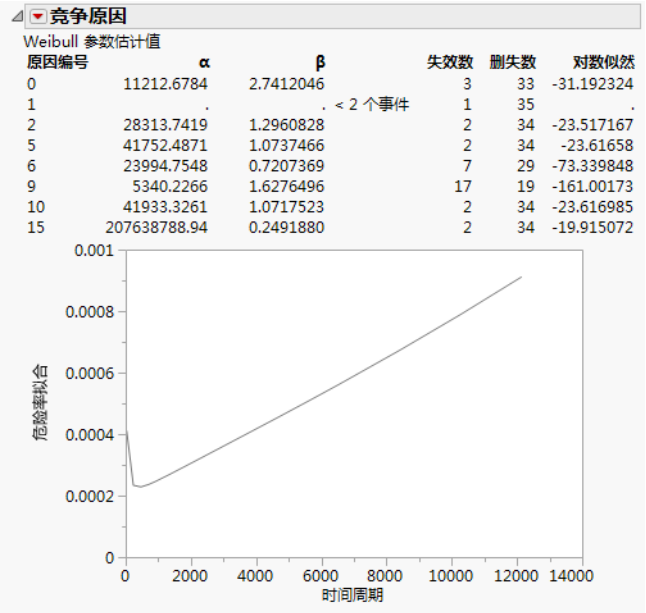
竞争原因的示例

Nelson (1982) 讨论了有很多失效原因的小电器的失效时间。数据的一个组（第 2 组）用 JMP 样本数据表 Appliance.jmp 来表示。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Appliance.jmp。
2. 选择**分析 > 可靠性和生存 > 生存**。
3. 选择**时间周期**并点击 **Y，事件时间**。

4. 点击**确定**。
5. 点击“乘积限生存拟合”旁边的红色小三角并选择**竞争原因**。
6. 点击**原因编号**，然后点击**确定**。
7. 点击“竞争原因”红色小三角并选择**危险率图**选项。

图 13.11 “竞争原因”报表和危险率图

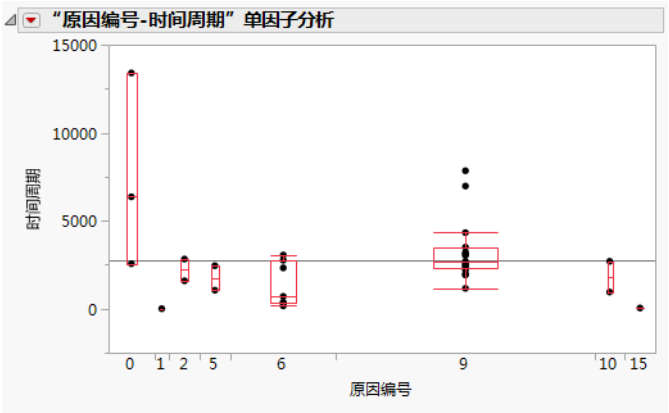


整个系统的生存分布仅是生存概率之积。“竞争原因”表显示每个失效原因的 Alpha 和 Beta 的 Weibull 估计值。

在本例中，大多数失效是由于原因 9 造成的。原因 1 只出现过一次且不能生成好的 Weibull 估计值。原因 15 出现的时间很短而且导致很小的 beta 和很大的 alpha。回想一下 alpha 是失效时间的 63.2% 分位数估计值，这意味着前期失效的原因通常有很大的 alpha。若这些原因不导致前期失效，则也通常不会导致后期失效。

图 13.12 显示了在“分位数”选项有效时按“原因编号”划分的“时间周期”的“以 X 拟合 Y”图。该图进一步说明 alpha 和 beta 与失效分布之间的关系。

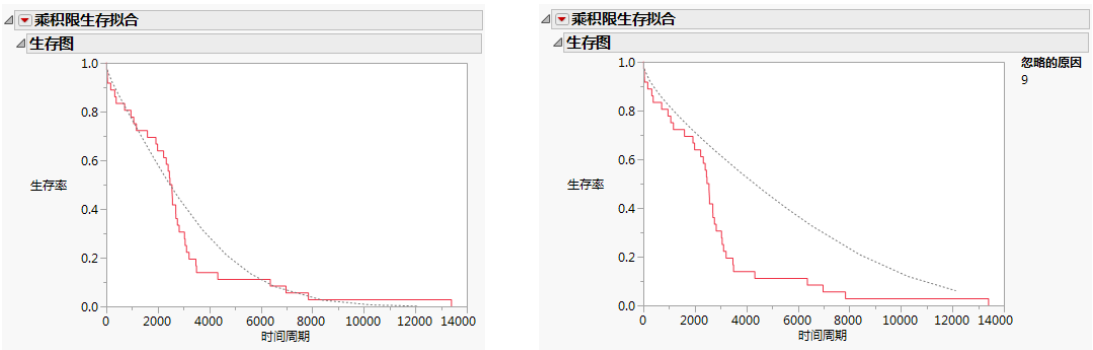
图 13.12 按 “原因编号”划分的 “时间周期”的 “以 X 拟合 Y”图



在本例中，回想一下原因 9 导致了大多数失效。若更正了原因 9 的问题，这将如何影响其他原因导致的生存概率呢？选择**忽略原因**选项可删除原因值并重新计算生存估计值。

图 13.13 显示了具有所有竞争原因的生存图和没有原因 9 的生存图。您可以看到不含原因 9 的生存率（用虚线表示）在 2,000 个周期前没有大的改进。之后，该生存率变得好许多，甚至在 10,000 个周期后仍有改进。

图 13.13 具有忽略原因的生存图



区间删失的示例

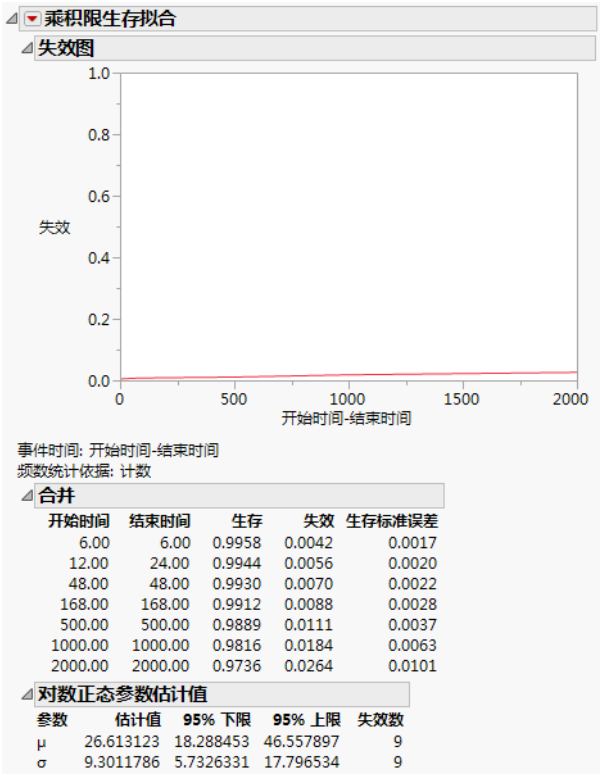
有区间删失数据时，您只知道事件在某个时间区间内发生。 Turnbull 方法用于获取生存函数的非参数估计值。

在 Nelson (1990, p. 147) 的这一示例中，在不同时间测试和检查了微处理器单元并记录了故障单元的数目。其中一列中的缺失值表示您不知道下限或上限，因此事件相应为左删失或右删失。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Microprocessor Data.jmp。
2. 选择**分析** > **可靠性和生存** > **生存**。

3. 选择**开始时间**和**结束时间**并点击 **Y，事件时间**。
4. 选择**计数**并点击**频数**。
5. 选中**标绘失效图而非生存图**旁边的复选框。
6. 点击**确定**。
7. 点击“乘积限生存拟合”旁边的红色小三角并选择**对数正态拟合**。

图 13.14 区间删失输出



显示生成的 Turnbull 估计值。Turnbull 估计值可能有无法估计生存概率的时间间隙。在本例中，这样的间隙出现在 6 和 12、24 和 48、48 和 168 之间等位置。

此时，选择一个分布以查看它的拟合估计值 — 在本例中，拟合的是对数正态分布。请注意失效图显示这些数据的失效率很小。

生存分析的统计报表

对于非区间删失的数据，初始报表显示“汇总”和“分位数”数据（图 13.15）。“汇总”数据显示每个组（存在组的情况下）和整个研究的失效数以及删失观测数。还针对删失调整了均值

和标准差。有关这些统计量的计算详细信息，请参见 SAS Institute Inc. (2020) 中的“LIFETEST 过程”一章。(2020)。

“分位数”数据显示单个组和合并组的失效时间统计量，其中包括中位数生存时间（带 95% 置信上限和置信下限）。中位数生存时间是一半对象失效时的时间（天数）。还包括四分位数生存时间（25% 和 75%）。

图 13.15 一元生存分析的汇总统计量

乘积限生存拟合

事件时间: 天数
删失依据: 删失
删失代码 1
分组依据: 组

汇总

分组	失效数	删失数	均值		标准误差
Group 1	19	2	240.795	有偏	11.206
Group 2	17	2	218.757		9.40318
合并	36	4	230.729	有偏	7.57346

分位数

分组	中位数时间	95% 下限	95% 上限	25% 失效	75% 失效
Group 1	233	232	280	232	280
Group 2	216	190	234	190	234
合并	232	213	239	201.5	261

“汇总”报表给出生存时间均值的估计值以及该均值的标准误差。估计的生存时间均值定义如下：

$$\hat{\mu} = \sum_{i=1}^{D} \hat{S}(t_{i-1})(t_i - t_{i-1})$$ 其标准误差为 $$\hat{\sigma}(\hat{\mu}) = \sqrt{\frac{m}{m-1}\sum_{i=1}^{D-1}\frac{A_i^2}{n_i(n_i-d_i)}}$$

其中

$$\hat{S}(t_i) = \prod_{j=1}^{i}\left(1 - \frac{d_j}{n_j}\right)$$

$$A_i = \sum_{j=i}^{D-1}\hat{S}(t_j)(t_{j+1} - t_j)$$

$$m = \sum_{j=1}^{D} d_j$$

$\hat{S}(t_i)$ 是时间 t_i 处的生存分布

D 是不同的事件时间数

n_i 是恰好在 t_i 前的生存单元数

d_i 是在 t_i 处失效的单元数

t_0 定义为 0

有多个组时，“组间检验”表会提供各组的齐性统计检验。Kalbfleisch and Prentice (1980, ch. 1), Hosmer and Lemeshow (1999, ch. 2) 和 Klein and Moeschberger (1997, ch. 7) 讨论了生存曲线的统计量和比较。

图 13.16 组间检验

组间检验

检验	卡方	自由度	概率>卡方
对数秩	3.1227	1	0.0772
Wilcoxon	2.6510	1	0.1035

检验 命名两个统计检验，它们假设生存函数在不同组之间是相同的。

卡方 提供统计检验的卡方近似。

对数秩检验赋予较大生存时间更大的权重，当要比较的组中的危险率函数之比近似为常数时，它会更加有用。危险率函数是给定时间处的瞬间失效率。它也称为**死亡率**或**死亡力**。

Wilcoxon 检验赋予前期生存时间更大的权重，若误差分布为 Logistic 分布，则该检验是最佳秩检验。请参见 Kalbfleisch and Prentice (1980)。

自由度 提供统计检验的自由度。

概率 > 卡方 列出当生存函数对于所有组相同时，得到卡方值大于计算值的概率。

图 13.17 显示了一个组的乘积限生存函数估计值的示例。

图 13.17 “生存估计值”表的示例

Group 1

天数	生存	失效	生存标准误差	失效数	删失数	风险度
0.000	1.0000	0.0000	0.0000	0	0	21
142.000	0.9524	0.0476	0.0465	1	0	21
156.000	0.9048	0.0952	0.0641	1	0	20
163.000	0.8571	0.1429	0.0764	1	0	19
198.000	0.8095	0.1905	0.0857	1	0	18
204.000	0.8095	0.1905	0.0857	0	1	17
205.000	0.7589	0.2411	0.0941	1	0	16
232.000	0.6577	0.3423	0.1053	2	0	15
233.000	0.4554	0.5446	0.1114	4	0	13
239.000	0.4048	0.5952	0.1099	1	0	9
240.000	0.3542	0.6458	0.1072	1	0	8
261.000	0.3036	0.6964	0.1031	1	0	7
280.000	0.2024	0.7976	0.0902	2	0	6
296.000	0.1012	0.8988	0.0678	2	0	4
323.000	0.0506	0.9494	0.0493	1	0	2
344.000	0.0506	0.9494	0.0493	0	1	1

注意：记录的最后时间是删失观测时，该报表会指示**有偏**均值估计值。有偏均值估计值是真实均值的下限。

第 14 章

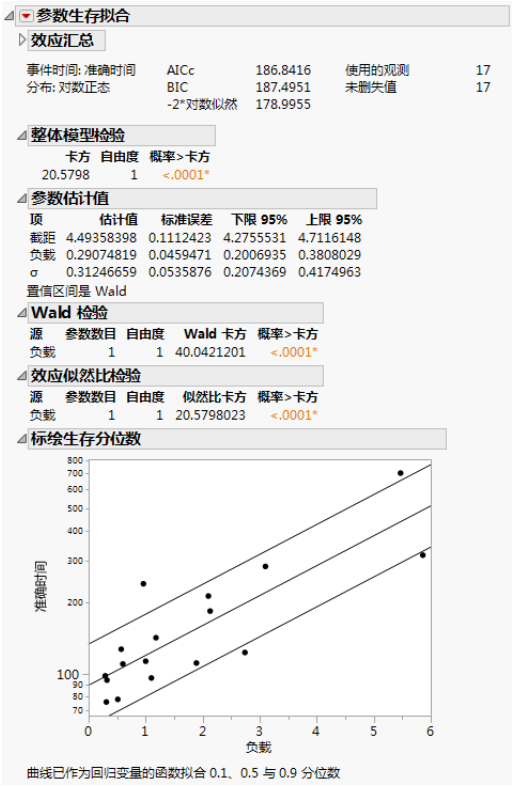
拟合参数生存

使用回归模型拟合生存数据

生存时间可表示为一个或多个变量的函数。在这种情况下，使用回归平台在拟合线性回归模型的同时将生存分布和删失考虑在内。“拟合参数生存”平台使用可涉及位置和尺度效应的线性回归模型拟合事件时间 Y（具有删失）。使用 Weibull、对数正态、指数、Fréchet、对数 Logistic、最小极值 (SEV)、正态、最大极值 (LEV) 和 Logistic 分布执行拟合。

注意：“拟合参数生存”平台是略微经过定制化的“拟合模型”平台。您还可以使用“非线性”平台拟合参数生存模型。

图 14.1 “参数生存拟合”的示例



目录

“拟合参数生存”平台概述 323
参数回归生存拟合的示例 323
启动“拟合参数生存”平台 325
“参数生存拟合”报表 326
“参数生存 - 所有分布”报表 328
“参数竞争原因”报表 329
“拟合参数生存”选项 329
非线性参数生存模型 330
拟合参数生存的更多示例 331
Arrhenius 加速失效对数正态模型 331
区间删失的加速失效时间模型 334
使用“非线性”平台分析删失数据 336
左删失数据 336
使用“非线性”平台的 Weibull 损失函数 338
使用“非线性”平台拟合简单生存分布 340
“拟合参数生存”平台的统计详细信息 343
生存分布的损失公式 343

“拟合参数生存”平台概述

生存时间可表示为一个或多个变量的函数。在这种情况下，使用回归平台在拟合线性回归模型的同时将生存分布和删失考虑在内。“拟合参数生存”平台使用可涉及位置和尺度效应的线性回归模型拟合事件时间 Y（具有删失）。使用 Weibull、对数正态、指数、Fréchet、对数 Logistic、最小极值、正态、最大极值和 Logistic 分布执行拟合。

参数回归生存拟合的示例

数据表 Comptime.jmp 包含计算机程序执行时间的分析数据，其执行时间的对数正态分布依赖于效应**负载**。

注意：Comptime.jmp 中的数据来自 Meeker and Escobar (1998, p. 434)。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Comptime.jmp。
2. 选择**分析** > **可靠性和生存** > **拟合参数生存**。
3. 选择**执行时间**并点击**事件时间**。
4. 选择**负载**并点击**添加**。
5. 将**分布**从 Weibull 更改为**对数正态**。
6. 点击**运行**。

图 14.2 计算时间输出

参数生存拟合

效应汇总

事件时间: 准确时间	AICc	186.8416	使用的观测	17
分布: 对数正态	BIC	187.4951	未删失值	17
	-2*对数似然	178.9955		

整体模型检验

卡方	自由度	概率>卡方
20.5798	1	<.0001*

参数估计值

项	估计值	标准误差	下限 95%	上限 95%
截距	4.49358398	0.1112423	4.2755531	4.7116148
负载	0.29074819	0.0459471	0.2006935	0.3808029
σ	0.31246659	0.0535876	0.2074369	0.4174963

置信区间是 Wald

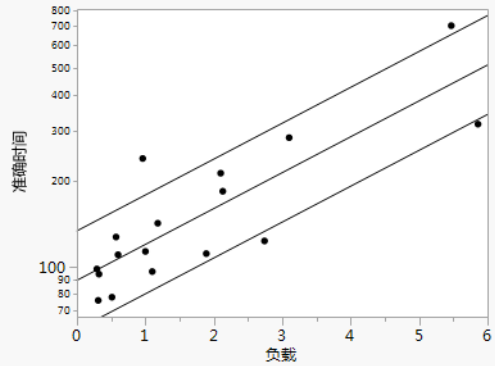
Wald 检验

源	参数数目	自由度	Wald 卡方	概率>卡方
负载	1	1	40.0421201	<.0001*

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
负载	1	1	20.5798023	<.0001*

标绘生存分位数



曲线已作为回归变量的函数拟合 0.1、0.5 与 0.9 分位数

只有一个效应时，三个生存概率的生存分位数图显示为该效应的函数。

当 90% 的作业是在系统负载 5 下完成时需要使用时间分位数。请参见 Meeker and Escobar (1998, p. 438)。

7. 点击“参数生存拟合”红色小三角并选择**估计分位数**。
8. 在**负载**下面的第一行中键入 5。
9. 在 p 下面的第一行中键入 0.9。
10. 点击**执行**。

图 14.3 时间分位数估计

分位数估计值

负载	p	分位数	95% 下限	95% 上限
5	0.9	571.21575	401.29076	813.09482

报表估计 90% 的作业在系统负载 5 下的执行时间不超过 571 秒。

启动“拟合参数生存”平台

通过选择**分析 > 可靠性和生存 > 拟合参数生存**来启动 “拟合参数生存”平台。

图 14.4 “拟合参数生存”启动窗口

提示：要更改 alpha 水平，请点击“模型规格”红色小三角并选择**设置 Alpha 水平**。

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“拟合参数生存”启动窗口包含以下选项：

事件时间 包含事件时间或删失时间。在区间删失时，指定两个 *Y* 变量，其中一个 *Y* 变量提供每个单元的下限，另一个 *Y* 变量提供每个单元的上限。

删失 指定一列用指示符来标识右删失观测。从 “删失代码”菜单中选择用于标识右删失观测的值。仅当输入一个 “事件时间”列时才使用 “删失”列。

频数 指定记录多个单元时包含每行的观测频数或计数的列。

原因 指定包含多个失效原因的列。该列对于估计竞争原因特别有用。为每个原因值执行单独的参数拟合。可以使用数值或分类（标签）值对失效事件编码。

依据 为分类或分组变量的每个水平执行单独的分析。

位置和尺度效应 指定位置和尺度效应。有关“构造模型效应”选项的详细信息，请参见《拟合线性模型》。

特质 指示拟合方法。应始终选择“参数生存”。

分布 选择适合您的数据所需的响应分布。选择“所有分布”选项可拟合所有分布并比较拟合效果。若选择“所有分布”，该报表显示分布拟合的对比。请参见第 328 页的“‘参数生存 - 所有分布’报表”。

注意：默认情况下，“所有分布”选项针对“分布”菜单中的“所有分布”上方显示的对数 - 位置 - 尺度分布拟合模型。选择**首选项 > 平台 > 拟合参数生存 > 在所有分布中包含位置尺度分布**以更改“所有分布”选项的行为，使其同时包括位置 - 尺度分布。

删失代码 标识“删失”列中指定右删失观测的值。选定某个“删失”列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中进行选择。您还可以在框中输入不同值。若“删失”列包含“值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

“参数生存拟合”报表

若在启动窗口中选择“所有分布”，则为每个分布显示“参数生存拟合”报表。若在启动窗口中指定“原因”列，则为每个原因显示“参数生存拟合”报表。否则，仅显示一个“参数生存拟合”报表。每个“参数生存拟合”报表都包含以下信息：

效应汇总 显示一个交互式报表，该报表允许您在模型中添加或删除效应。请参见《拟合线性模型》。

模型拟合详细信息 “事件时间”显示所指定的 Y 列，“分布”显示所拟合的分布。“AICc”、“BIC”和“-2 对数似然”都是模型拟合的测度。可以使用这些测度来进行与其他模型拟合的比较。“使用的观测”和“未删失值”是数据的汇总统计量。请参见《拟合线性模型》。

整体模型检验 将完整拟合与仅截距拟合作比较。若只有一个截距项，则拟合结果与“寿命分布”平台的拟合结果相同。

参数估计值 显示回归参数的估计值。

JMP PRO 一个链接，用于启动在“参数估计值”表下方显示的“广义回归”平台。该链接支持您使用“广义回归”平台执行变量选择，在以下条件下显示该链接：

- 模型没有尺度效应。
- 在启动窗口中未指定“原因”列。

– 在启动窗口中指定的“分布”为“正态”、“对数正态”或 Weibull。

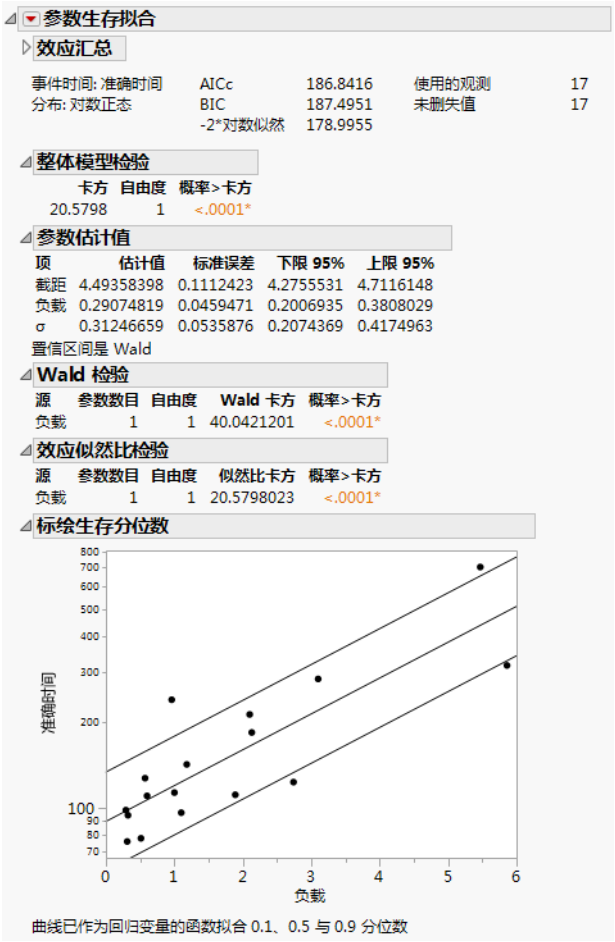
备择参数 （仅适用于 Weibull 分布。）显示 Weibull 分布的 α 和 β 参数化的参数估计值。有关该参数化的详细信息，请参见“寿命分布”一章中第 74 页的“Weibull”。

Wald 检验 为模型中的每一项显示 Wald 卡方检验。

效应似然比检验 比较拟合模型的对数似然与分别从模型删除每个项时的对数似然。

标绘生存分位数 显示使用 0.1、0.5 和 0.9 分位数标绘的数据点。

图 14.5 “参数生存拟合”报表



“参数生存 - 所有分布”报表

仅当在启动窗口中选择“所有分布”时才会显示“参数生存 - 所有分布”报表。默认情况下，该报表包含“模型比较”报表和“分布叠加”图。“分位数函数叠加”图在“参数生存 - 所有分布”旁边的红色小三角菜单中提供。

模型比较 列出拟合分布的拟合统计量（AICc 和 BIC）的表。具有最小 AICc 和 BIC 值的分布标注在最右侧的列中。若有一个分布同时具有最小的 AICc 和 BIC 值，则该分布将标注为“最佳”。默认打开与具有最小 AICc 的分布对应的“参数生存拟合”报表。有关这些统计量的详细信息，请参见《拟合线性模型》。

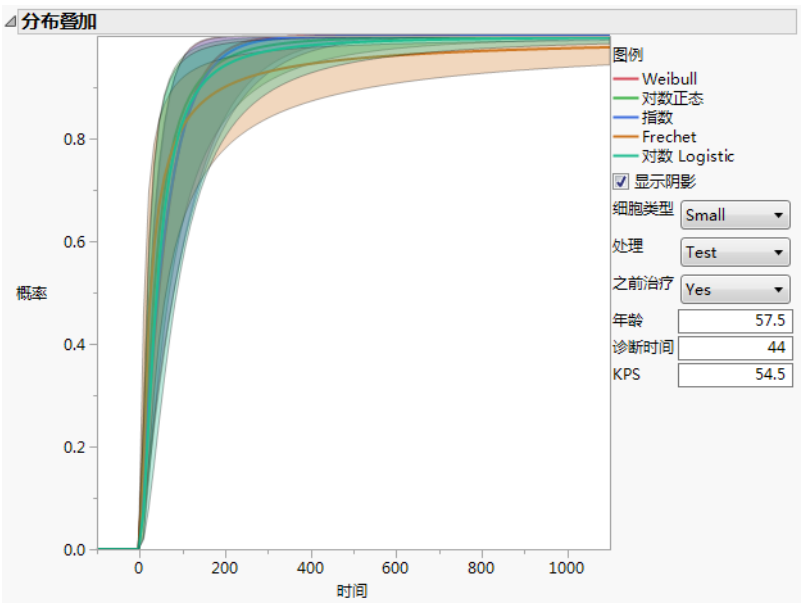
分布叠加 指定效应值处的拟合分布的叠加分布函数图。

分位数函数叠加 指定效应值处的拟合分布的叠加分位数函数图。

“分布叠加”图和“分位数函数叠加”图都为在同一图形上叠加的每个拟合分布显示曲线。默认情况下，每条曲线具有基于 Wald 的着色置信区间。每个图的右侧都有一个图例、一个显示置信区间着色的选项，以及支持您指定不同效应值的控件。图 14.6 显示了“分布叠加”图的一个示例。

注意：通过从“拟合模型”启动窗口的红色小三角菜单中选择“设置 Alpha 水平”，您可以更改着色置信区间的 α 水平。默认的 α 水平为 0.05。

图 14.6 “分布叠加”图



“参数竞争原因”报表

若您在启动窗口中指定了 “原因”列，则会显示 “参数竞争原因”报表。若还在启动窗口中指定了 “所有分布”，该报表将标记为 “参数竞争原因 - 所有分布”。“参数竞争原因”报表包含以下信息：

按原因汇总 列出每个原因的拟合统计量（AICc 和 BIC）的表。若在启动窗口中为 “分布”选项选定了 “所有分布”，则该表将包括每个分布拟合中每个原因的拟合统计量。

模型比较 （仅当在启动窗口中为 “分布”选项选定 “所有分布”时才可用。）显示列出拟合分布的拟合统计量 （AICc 和 BIC）的表。具有最小 AICc 和 BIC 值的分布标注在最右侧的列中。若有一个分布同时具有最小的 AICc 和 BIC 值，则该分布将标注为 “最佳”。默认打开与具有最小 AICc 的分布对应的 “参数生存拟合”报表。

有关 AICc 和 BIC 统计量的详细信息，请参见《拟合线性模型》。

“拟合参数生存”选项

“参数生存拟合”红色小三角菜单包含以下选项：

似然比检验 生成一些检验，用于比较拟合模型的对数似然与分别从模型删除每个项时的对数似然。

Wald 检验 为检验每个参数是否为 0 的 Wald 检验生成卡方检验统计量和 *p* 值。

似然置信区间 为每个参数指定 “参数估计值”表中所示的置信区间的类型。选定该选项后，即显示边侧似然置信区间；否则显示 Wald 区间。在该报表中，“参数估计值”表的下方会注明区间类型。若边侧似然置信区间的计算时间不长，则默认启用该选项。

注意：通过从 “拟合模型”启动窗口的红色小三角菜单中选择 “设置 Alpha 水平”，您可以更改置信区间的 α 水平。默认的 α 水平为 0.05。

估计值的相关性 生成模型效应之间以及模型效应与拟合分布的参数之间的相关性矩阵。

估计值的协方差 生成模型效应之间以及模型效应与拟合分布的参数之间的协方差矩阵。

估计生存概率 对给定的时间值估计失效和生存概率。指定效应值以及一个或多个时间值。JMP 然后计算条目的所有可能组合的 95% 置信限下的生存和失效概率。

估计分位数 估计给定概率的分位数。指定效应值以及一个或多个分位数概率。 JMP 然后计算条目的所有可能组合的时间分位数和 95% 置信限。

注意：对于 “估计生存概率”选项和 “估计分位数”选项，您可以更改默认的 0.05 alpha 水平。

残差概率图 显示标准化残差的概率图。

保存残差 将残差保存至数据表中的新列。对于区间删失观测，有两列残差保存到数据表中。

分布刻画器 显示失效概率针对各个说明变量和响应变量的响应曲面。

分位数刻画器 显示响应变量针对说明变量和失效概率的响应曲面。

按水平组合划分的分布图 显示用于评估模型拟合的三个概率图。这些图针对 X 水平的每个组合显示不同的线。

不同位置 假定尺度参数相等而位置参数不同的概率图。这对于评估平行性假设很有用。

不同位置和尺度 假定不同尺度和位置参数的概率图。这对于评估分布对于数据是否合适很有用。“指数”分布不显示该图。

回归 分布参数是 X 变量的函数的概率图。

保存概率公式 将估计的概率公式保存到数据表中的新列。

保存分位数公式 将估计的分位数公式保存到数据表中的新列。选择该选项将显示一个弹出窗口，要求您输入所关注的分位数的概率值。

发布概率公式 创建预测公式并将其另存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

发布分位数公式 创建分位数公式并将其另存为 “公式存储库”平台中的公式列脚本。若未打开 “公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

模型对话框 重启启动窗口。

效应汇总 显示交互式“效应汇总”报表，该报表支持您在模型中添加或删除效应。请参见《拟合线性模型》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

非线性参数生存模型

在以下情形下将 “非线性”平台用于生存模型：

- 模型是非线性的。
- 您需要除 Weibull、对数正态、指数、Fréchet、对数 Logistic、最小极值、正态、最大极值或 Logistic 之外的一种分布。

- 您的删失不是通常的右删失、左删失或区间删失。

“非线性”平台具有估计所指定的损失函数中参数的能力，因此成为拟合最大似然模型的强大工具。有关“非线性”平台的完整信息，请参见《预测和专业建模》。

要在数据删失时拟合非线性模型，您必须首先使用公式编辑器来创建一个参数方程来表示针对删失观测调整的损失函数。然后通过“非线性”平台使用最大似然估计参数。

拟合参数生存的更多示例

- “Arrhenius 加速失效对数正态模型”
- “区间删失的加速失效时间模型”
- “使用“非线性”平台分析删失数据”
- “左删失数据”
- “使用“非线性”平台的 Weibull 损失函数”
- “使用“非线性”平台拟合简单生存分布”

Arrhenius 加速失效对数正态模型

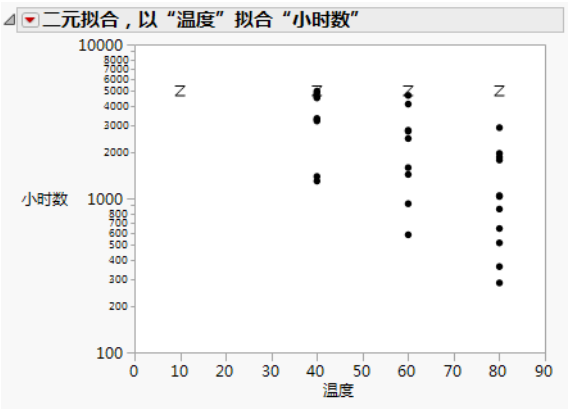
在 Devalt.jmp 数据中，通过加热对单元施加压力以便使它们很快发生故障，从而获取足够的失效数来拟合分布。

注意：Devalt.jmp 中的数据来自 Meeker and Escobar (1998, p. 493)。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Devalt.jmp。

   首先，使用 “二元”平台来查看时间使用对数尺度的温度 - 小时数图。
2. 选择**分析** > **以 X 拟合 Y**。
3. 选择**小时数**并点击 **Y，响应**。
4. 选择**温度**并点击 **X，因子**。
5. 点击**确定**。

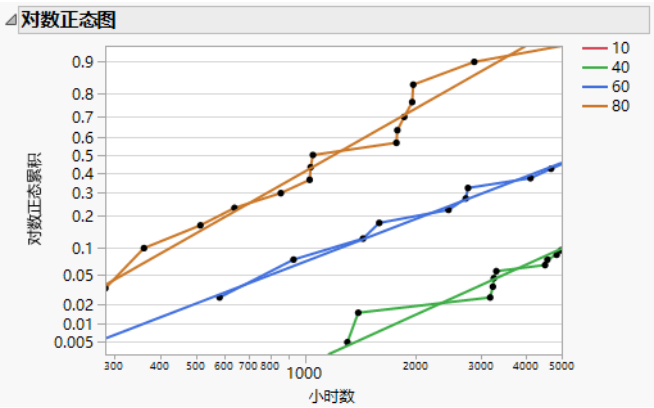
图 14.7 温度对数 - 小时数的二元图



接下来，使用生存平台针对每个温度生成数据的对数正态图。

6. 选择**分析 > 可靠性和生存 > 生存**。
7. 选择**小时数**并点击 **Y，事件时间**。
8. 选择**删失**并点击**删失**。
9. 选择**温度**并点击**分组**。
10. 选择**权重**并点击**频数**。
11. 点击**确定**。
12. 点击 “乘积限生存拟合”旁边的红色小三角并选择**对数正态图**和**对数正态拟合**。
13. 点击**确定**。

图 14.8 对数正态图



接下来，使用 “拟合参数生存”平台来使用温度的一个效应拟合一个模型。

14. 选择**分析 > 可靠性和生存 > 拟合参数生存**。

15. 选择**小时数**并点击**事件时间**。
16. 选择 **x** 并点击**添加**。
17. 选择**删失**并点击**删失**。
18. 选择**权重**并点击**频数**。
19. 将**分布**类型改为**对数正态**。
20. 点击**运行**。

图 14.9 “Devalt”参数输出

参数生存拟合

效应汇总

事件时间: 小时数	AICc	649.5546	使用的观测	165
分布: 对数正态	BIC	658.7234	未删失值	33
删失依据: 删失	-2*对数似然	643.4056	右删失值	132
频数列: 权重				

整体模型检验

卡方	自由度	概率>卡方
80.7141	1	<.0001*

参数估计值

项	估计值	标准误差	95 下限%	95 上限%
截距	-13.468649	2.8871953	-19.12745	-7.809851
x	0.62787903	0.0828422	0.4655112	0.7902468
σ	0.97782331	0.1326468	0.7178403	1.2378063

置信区间是 Wald

使用广义回归的变量选择

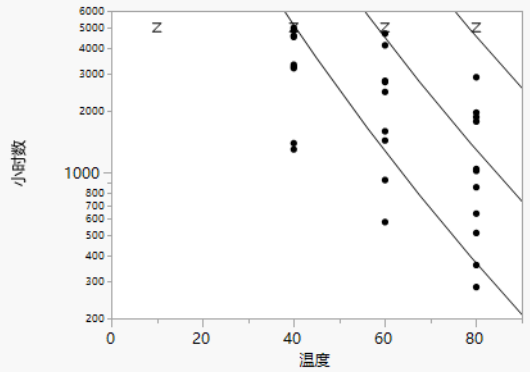
Wald 检验

源	参数数目	自由度	Wald 卡方	概率>卡方
x	1	1	57.4444814	<.0001*

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
x	1	1	80.714074	<.0001*

标绘生存分位数



曲线已作为回归变量的函数拟合 0.1、0.5 与 0.9 分位数
部分数据点为删失时间

结果显示数据的回归拟合：

– 若只有一个效应并且它是连续的，则显示作为该效应的函数的生存图。线分别对应 0.1、0.5 和 0.9 生存概率。
– 若效应列有用其他一列表示的公式（如本示例中的情况），图按内部列来进行绘制。在本示例中，效应是列 x，但是按**温度**（x 是其函数）来绘制图。

最后，获取外推到温度为 10 摄氏度（对应时间为 30000 和 10000 小时）的生存概率估计值。

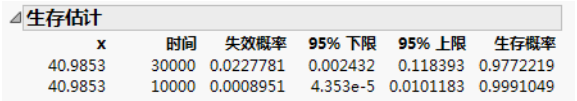
21. 点击 “参数生存拟合”红色小三角并选择**估计生存概率**。
22. 将图 14.10 中所示的值输入 “估计生存对话框”。

    10 度的 Arrhenius 变换是 40.9853 （即效应值）。

图 14.10 估计生存概率

23. 点击**执行**。

图 14.11 生存概率



“生存估计值”报表显示估计值和置信区间。

区间删失的加速失效时间模型

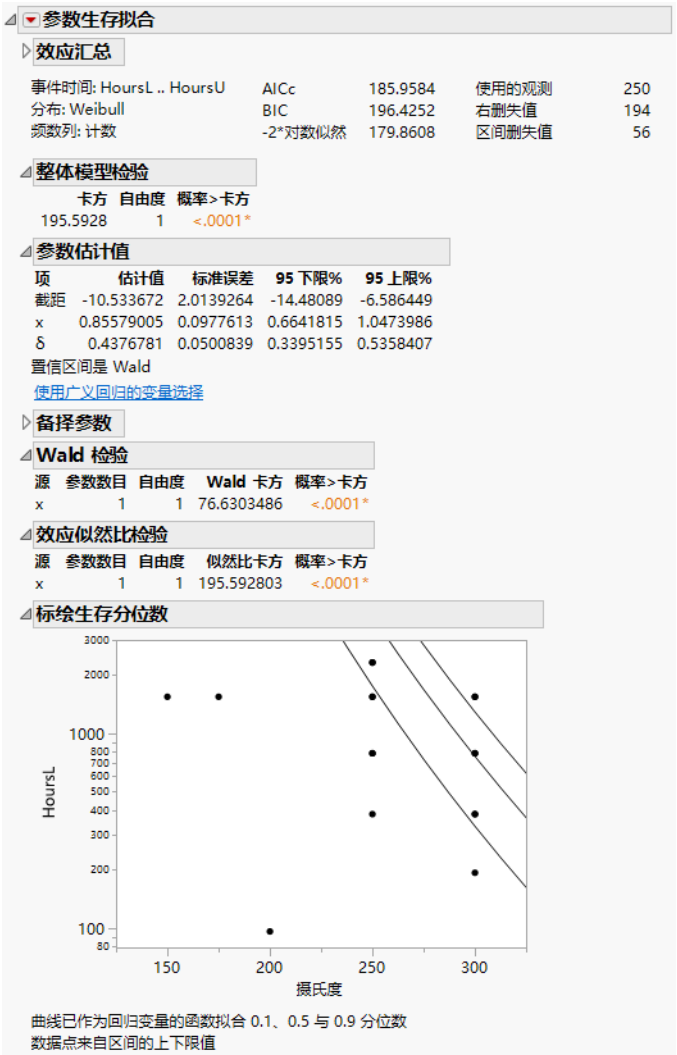
ICdevice02.jmp 数据显示在检查间隔之间发生的失效数。该模型使用两个 y 变量，包含失效时间的上限和下限。右删失时间显示为上限缺失。

注意：ICdevice02.jmp 中的数据来自 Meeker and Escobar (1998, p. 640)。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/ICdevice02.jmp。
2. 选择**分析 > 可靠性和生存 > 拟合参数生存**。
3. 选择 HoursL 和 HoursU 并点击**事件时间**。

4. 选择**计数**并点击**频数**。
5. 选择 x 并点击**添加**。
6. 点击**运行**。

图 14.12 ICDevice 输出



生成的回归显示度数-时间图。

使用“非线性”平台分析删失数据

您可以使用“非线性”平台分析左删失数据。对于左删失数据，零是删失值，因为它也表示观测的最小已知时间。

注意：对于必须为正数或零的响应（零表示删失点），Tobit 模型在经济领域很流行。

请注意有关 Tobit2.jmp 数据表的以下信息：

- 响应变量是产品耐用性的测度，不能小于零（**耐久性**在零处左删失）。
- **寿命**和**流动性**是自变量。
- 该表还包含模型和 tobit 损失函数。残差形式的模型为**耐久性 - (b0+b1*寿命+b2*流动性)**。要查看与 Tobit **损失**关联的公式，请右击该列并选择**公式**。

拟合 Tobit 模型：

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Tobit2.jmp。
2. 选择**分析 > 专业建模 > 非线性**。
3. 选择**模型**并点击 **X，预测变量公式**。
4. 选择 **Tobit 损失**并点击**损失**。
5. 点击**确定**。
6. 点击**执行**。
7. 点击**置信限**。

图 14.13 “解”报表

解

损失	误差自由度	平均损失	均方根损失
28.92596097	16	1.8078726	1.3445715

参数	估计值	近似标准误差	置信下限	置信上限
b0	15.277120063	16.0327167	-22.059913	54.7012259
b1	-0.134007501	0.21893135	-0.7365041	0.32841861
b2	-0.045135584	0.05826851	-0.1858906	0.09330941
Sigma	5.5693492202	1.72814365	3.31613117	11.5291865

求解方法: 分析 NR

估计值的相关性

左删失数据

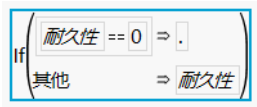
上一节中的 Tobit 模型假定服从在零处删失的正态分布。零观测被视为左删失。在“拟合参数生存”平台中，使用两个响应列指定左删失观测。在本例中，您创建一个指示出现左删失的观测的新列。对于左删失观测，新列包含缺失值。否则，它包含**耐久性**的观测值。新列随后用作区间删失观测的左侧，而现有列**耐久性**用作区间的右侧。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Tobit2.jmp。

创建左删失列：

2. 选择**列** > **新建列**。
3. 在**列名**中键入 durable0。
4. 选择**列属性**并点击**公式**。
5. 选择**条件** > If 并选择**耐久性**。
6. 选择**比较** > a == b，键入 0 并按 Enter 键。
7. 选择标记为 “else 子句” 的框，然后选择**耐久性**。
   图 14.14 显示了完整的公式。

图 14.14 durable0 的列公式



8. 点击**确定**。
9. 点击**确定**。

拟合 Tobit 模型：

10. 选择**分析** > **可靠性和生存** > **拟合参数生存**。
11. 选择 durable0 并点击**事件时间**。
12. 选择**耐久性**并点击**事件时间**。
    您必须使用两个响应列来指定左删失观测。删失的方向由 “事件时间” 角色中列的顺序确定。
13. 选择**寿命**和**流动性**并点击**添加**。
14. 将**分布**类型改为**正态**。
15. 点击**运行**。

图 14.15 Tobit 模型结果

参数生存拟合

效应汇总

事件时间: durable0.. 耐久性		AICc	68.51859	使用的观测	20
分布: 正态		BIC	69.83485	未删失值	7
		-2*对数似然	57.85192	左删失值	13

整体模型检验

卡方	自由度	概率>卡方
1.1325	2	0.5677

参数估计值

项	估计值	标准误差	下限 95%	上限 95%
截距	15.2771208	16.032721	-16.14643	46.700676
寿命	-0.1340075	0.2189314	-0.563105	0.2950901
流动性	-0.0451356	0.0582685	-0.15934	0.0690686
σ	5.56935051	1.7281446	2.1822494	8.9564517

置信区间是 Wald

使用广义回归的变量选择

Wald 检验

源	参数数目	自由度	Wald 卡方	概率>卡方
寿命	1	1	0.37466429	0.5405
流动性	1	1	0.60002606	0.4386

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
寿命	1	1	0.3931753	0.5306
流动性	1	1	0.57102176	0.4499

该报表显示估计的 Tobit 模型拟合。请注意在报表的右上部分中有 13 个左删失观测。这是一种很好的检查您是否已经正确指定左删失的方式。

使用“非线性”平台的 Weibull 损失函数

在本例中，使用 Weibull、对数正态和指数分布拟合生存时间模型。模型拟合包括只有两个效应的简单生存模型，包含所有效应的更复杂模型，以及为离散效应**细胞类型**创建虚拟变量以包括在完全模型中。

“非线性”模型拟合通常对于您指定的模型参数初始值很敏感。在本例中，找到合理初始值的一个方法是首先使用“非线性”平台仅拟合线性模型。该模型收敛时，参数的解值变为非线性模型的初始参数值。

1. 选择**帮助 > 样本数据库**，然后打开 VA Lung Cancer.jmp。

   第一个模型和所有损失函数已创建为数据表中的公式。**模型**列具有以下公式：

   ```
   Log( 时间 ) - (b0 + b1 * 年龄 + b2 * 诊断时间 )
   ```

2. 选择**分析 > 专业建模 > 非线性**。
3. 选择**模型**并点击 **X，预测变量公式**。
4. 点击**确定**。
5. 点击**执行**。

图 14.16 “非线性拟合”控制面板中的初始参数值

报表计算该模型的最小二乘参数估计值。

6. 点击**保存估计值**。

   公式列中的参数估计值设置为该初始非线性拟合过程估计的那些值。

   Weibull 列包含第 344 页的 “Weibull 损失函数”中所述的 Weibull 公式。

   要继续拟合过程，请执行以下步骤：

7. 再次选择**分析 > 专业建模 > 非线性**。
8. 选择**模型**并点击 **X，预测变量公式**。
9. 选择 Weibull **损失**并点击**损失**。
10. 点击**确定**。

    随即显示图 14.17 中左侧的 “非线性拟合”控制面板。现在损失函数中新增一个称为 sigma 的参数。因为它在分数的分母中，起始值 1 对于 sigma 是合理的。使用除默认值之外的任何损失函数时，默认选中控制面板上的**损失为负的对数似然**框。

11. 点击**执行**。

    拟合过程收敛情况如图 14.17 中右侧所示。

图 14.17 具有定制损失函数的非线性模型

拟合过程通过使 Weibull 似然函数的负对数最大化来估计参数。

12.（可选）点击**置信限**以在 "解" 表中显示参数的 95% 置信上限和下限。

图 14.18 "解" 报表

注意: 因为置信限是边侧似然置信区间而非标准渐近置信区间，计算它们可能需要一些时间。

还可以运行具有预定义的指数和对数正态损失函数的模型。在拟合另一模型前，将参数估计值重置为最小二乘估计值，否则它们可能不收敛。要重置参数估计值，请执行以下步骤：

13.（可选）点击 "非线性拟合" 红色小三角并选择**恢复原始参数**。

使用 "非线性" 平台拟合简单生存分布

以下示例显示如何使用最大似然法估计除删失状态外没有其他效应时的时间删失数据的分布。

Loss Function Templates 文件夹包含具有指数、极值、对数 Logistic、对数正态、正态以及单参数和双参数的 Weibull 损失函数公式的模板。要使用这些损失函数，请将您的时间和删失值复制到损失函数模板的**时间**和**删失**列。要运行模型，请选择**非线性**并将损失列分配为**损失**变量。因为响应模型和删失状态均包含在损失函数中且没有其他效应，您不需要预测列（模型变量）。

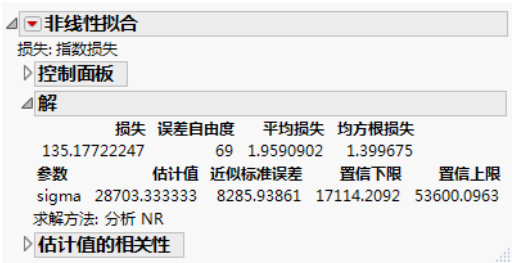
指数、Weibull 和极值损失函数

Fan.jmp 数据表可用于说明 Nelson (1982) 中所述的指数损失函数、Weibull 损失函数和极值损失函数。该数据来自对 70 个累积使用了 344,440 小时的柴油机风扇的研究。这些风扇是在不同的时间投入使用的。响应是风扇的失效时间或运行时间（若数据删失）。

提示：要查看 Fan.jmp 数据表中损失函数的公式，请右击**指数**、Weibull 和**极值**列，然后选择**公式**。

1. 选择**帮助** > **样本数据库**，然后打开 Reliability/Fan.jmp。
2. 选择**分析** > **专业建模** > **非线性**。
3. 选择**指数**并点击**损失**。
4. 点击**确定**。
5. 确保选中**损失为负的对数似然**复选框。
6. 点击**执行**。
7. 点击**置信限**。
8. 重复这些步骤，但是选择的是 Weibull 和**极值**而非**指数**。

图 14.19 非线性拟合结果


非线性拟合
损失: 指数损失
控制面板
解

损失	误差自由度	平均损失	均方根损失
135.17722247	69	1.9590902	1.399675

参数	估计值	近似标准误差	置信下限	置信上限
sigma	28703.333333	8285.93861	17114.2092	53600.0963

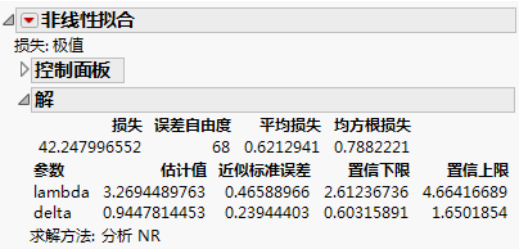
求解方法: 分析 NR
估计值的相关性

非线性拟合
损失: Weibull
控制面板
解

损失	误差自由度	平均损失	均方根损失
135.15271994	68	1.98754	1.4098014

参数	估计值	近似标准误差	置信下限	置信上限
Alpha	26296.817205	12251.3882	13629.9746	106086.918
Beta	1.0584462448	0.26825073	0.60609377	1.65792545

求解方法: 分析 NR


非线性拟合
损失: 极值
控制面板
解

损失	误差自由度	平均损失	均方根损失
42.247996552	68	0.6212941	0.7882221

参数	估计值	近似标准误差	置信下限	置信上限
lambda	3.2694489763	0.46588966	2.61236736	4.66416689
delta	0.9447814453	0.23944403	0.60315891	1.6501854

求解方法: 分析 NR

对数正态损失函数

Locomotive.jmp 数据可用于说明对数正态损失。当数据范围为 e 的若干次幂时，对数正态分布很有用。

提示：要在 Locomotive.jmp 数据表中查看损失函数的公式，请右击**对数正态**列，然后选择**公式**。

对数正态损失函数可能对参数的初始值很敏感。因为对数正态分布与正态分布相似，您可以创建是**时间**的自然对数的新变量并使用**分布**来查找该列的均值和标准差。然后，使用这些值作为“非线性”平台的起始值。在本例中，**时间**的自然对数的均值是 4.72，标准差为 0.35。

1. 选择**帮助 > 样本数据库**，然后打开 Reliability/Locomotive.jmp。
2. 选择**分析 > 专业建模 > 非线性**。
3. 选择**对数正态**并点击**损失**。
4. 点击**确定**。

5. 在 Mu 旁边的框中键入 4.72。
6. 在 sigma 旁边的框中键入 0.35。
7. 点击**执行**。
8. 点击**置信限**。

图 14.20 “解”报表

解

损失	误差自由度	平均损失	均方根损失
237.09354534	94	2.5222718	1.5881662

参数	估计值	近似标准误差	置信下限	置信上限
Mu	5.1169247193	0.10416697	4.93735853	5.35966436
sigma	0.7054940302	0.0932136	0.55446171	0.93294904

求解方法: 数值 NR

▷ 估计值的相关性

对数正态参数 Mu 和 Sigma 的最大似然估计值分别为 5.11692 和 0.7055。相应的对数正态分布的中位数估计值是 5.11692 的反对数 ($e^{5.11692}$)，它大约为 167，表示机车发动机的一般寿命。

“拟合参数生存”平台的统计详细信息

本节包含 “拟合参数生存”平台的统计详细信息。

生存分布的损失公式

以下公式是拟合常见参数模型的负对数似然公式。每个公式使用计算器 `if` 条件函数（未删失时执行第一个条件，右删失时执行 `Else` 子句）。您可以从**样本数据**的 Loss Function Templates 文件夹下的表中复制这些公式并将它们粘贴到您的数据表。

指数损失函数

在此处显示的指数损失函数中， sigma 表示指数分布的均值，**时间**是失效时的寿命。

$$-\text{IfMZ}\left(\begin{array}{ll} \text{删失} == 0 \Rightarrow & -\text{Log}\left(\text{sigma}\right) - \dfrac{\text{时间}}{\text{sigma}} \\ \text{else} \Rightarrow & -\left(\dfrac{\text{时间}}{\text{sigma}}\right) \end{array}\right)$$

指数分布的一个特点是瞬间失效率不随时间变化。这意味着无论对象处于研究中多长时间，在给定的时间长度内任何对象的失效机会是相同的。

Weibull 损失函数

Weibull 密度函数通常提供寿命分布的合适模型。您可以将 “生存”平台用于数据的初始调查以确定 Weibull 损失函数对您的数据是否合适。

$$-\text{IfMZ}\left(\begin{array}{ll} \text{删失} == 0 \Rightarrow & \frac{\text{模型}}{\text{sigma}} - \text{Exp}\left(\frac{\text{模型}}{\text{sigma}}\right) - \text{Log}(\text{sigma}) \\ \text{else} \Rightarrow & -\text{Exp}\left(\frac{\text{模型}}{\text{sigma}}\right) \end{array}\right)$$

在 Loss Function Templates 文件夹中有单参数函数、双参数函数和极值函数的示例。

对数正态损失函数

下面显示的公式是对数正态损失函数，其中**正态分布 (模型 /sigma)** 是标准正态分布函数。危险率函数在 $t = 0$ 处的值为 0，逐渐增大到最大值，然后减小。危险率函数随着 t 变大逐渐趋近 0。

$$-\text{If}\left(\begin{array}{ll} \text{删失} == 0 \Rightarrow & -0.5*\left(\frac{\text{模型}}{\textbf{sigma}}\right)^2 - 0.5*\text{Log}(2*\text{Pi}) - \text{Log}(\textbf{sigma}) \\ \text{else} \Rightarrow & \text{Log}\left(1 - \text{正态分布}\left(\frac{\text{模型}}{\textbf{sigma}}\right)\right) \end{array}\right)$$

对数 Logistic 损失函数

若 Y 服从 Logistic 分布，则 Exp(Y) 服从对数 Logistic 分布。

$$-\text{IfMZ}\left(\begin{array}{ll} \text{删失} == 0 \Rightarrow & \frac{\text{模型}}{\text{sigma}} - 2*\text{Log}\left(1 + \text{Exp}\left(\frac{\text{模型}}{\text{sigma}}\right)\right) - \text{Log}(\text{sigma}) \\ \text{else} \Rightarrow & -\text{Log}\left(1 + \text{Exp}\left(\frac{\text{模型}}{\text{sigma}}\right)\right) \end{array}\right)$$

第 15 章

拟合比例风险

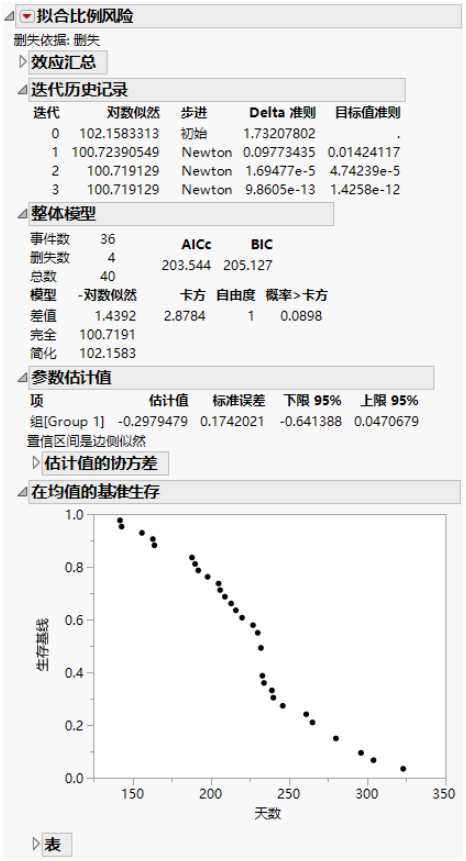
使用半参数回归模型拟合生存数据

“拟合比例风险”平台拟合 Cox 比例风险模型，该模型假定协变量（预测变量）与危险率函数之间存在相乘关系。

比例风险模型对于具有协变量的生存数据来说是常见的回归模型。该模型是半参数模型。估计线性模型，但不估计危险率函数的形式。不支持时变协变量。

注意：“拟合比例风险”平台是略微经过定制化的“拟合模型”平台。

图 15.1 “拟合比例风险”的示例



目录

“拟合比例风险”平台概述 347
“拟合比例风险”平台的示例 347
具有两个水平的一个名义型效应的风险比 348
启动“拟合比例风险”平台 350
“拟合比例风险”报表 351
“拟合比例风险”选项 352
使用多个效应和多个水平的示例 353
多个效应和多个水平的风险比 355

“拟合比例风险”平台概述

比例风险模型是 D. R. Cox (1972) 提出的特殊半参数回归模型，用来检查解释变量对生存时间的影响。假定总体的每个成员的生存时间遵循各自的危险率函数。

比例风险模型在某种意义上来说是非参数的，因为它涉及未指定的任意基准危险率函数。它又是参数的，因为它采用了协变量的参数形式。基准危险率函数采用与时间无关的协变量模型的函数来统一尺度以给出一般危险率函数。与 Kaplan-Meier 分析不同，比例风险计算每个协变量的参数估计值和标准误差。使用最大似然法估计与解释变量关联的回归参数 (β) 及其标准误差。还从参数估计值计算条件风险比（或危险比）。

使用经验法生成比例风险的生存估计值。请参见 Lawless (1982)。它们表示生存函数 $S(t)$ 的经验累积危险率函数估计值 $H(t)$，可以表示为 $S_0 = \exp(-H(t))$。危险率函数定义如下：

$$\mathrm{H}(t) = \sum_{j:t_j < t} \frac{d_j}{\sum_{l \in R_j} e^{x_l \beta}}$$

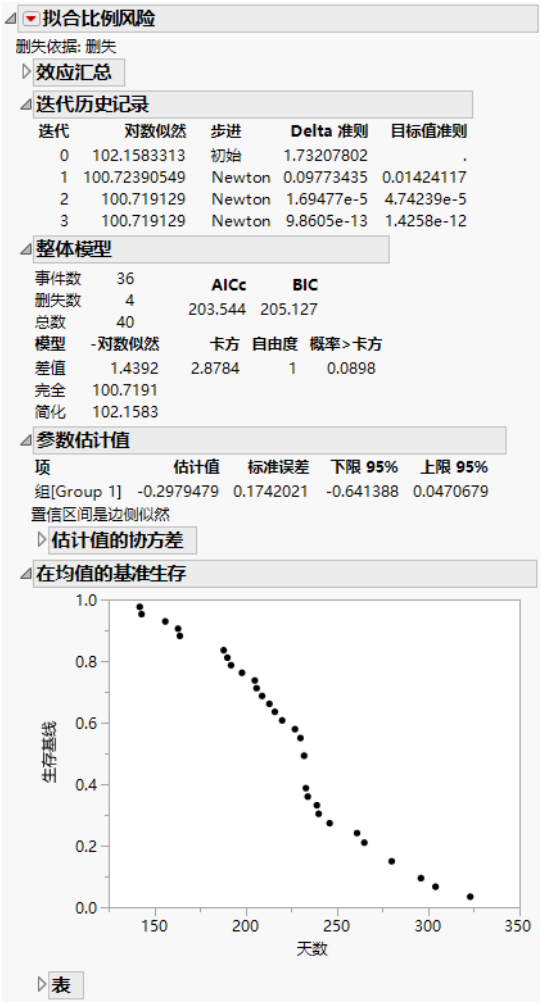
响应中有结值（表示在给定的时间事件处有多个失效）时，使用 Breslow 似然。

“拟合比例风险”平台的示例

本例说明具有两个水平的一个名义型效应。有关具有多个效应和多个水平的示例，请参见第 353 页的 “使用多个效应和多个水平的示例”。

1. 选择**帮助** > **样本数据库**，然后打开 Rats.jmp。
2. 选择**分析** > **可靠性和生存** > **拟合比例风险**。
3. 选择**天数**并点击**事件时间**。
4. 选择**删失**并点击**删失**。
5. 选择**组**并点击**添加**。
6. 点击**运行**。

图 15.2 Rats.jmp 数据的 “拟合比例风险”报表



在 Rats.jmp 数据中，只有两个组。因此，在“参数估计值”报表中，不包括零的置信区间指示组间存在着 alpha 水平的显著差异。此外，在“效应似然比检验”报表中，检验“整体模型”检验表中所示的组间没有差异这个原假设与检验**组**的回归系数为零这个原假设是等效的。

具有两个水平的一个名义型效应的风险比

要显示效应的风险比，请从红色小三角菜单中选择**风险比**选项。在本例中，只有一个效应，该效应只有两个水平。比较“Group 2”与“Group 1”的风险比并将结果显示在“‘组’的风险比”报表中。请参见图 15.3。 该表中的风险比按以下方式得到：计算“Group 2”的参数估计值的指数并除以“Group 1”的参数估计值的指数。

请注意以下事项：

- “Group 1”参数估计值显示在“参数估计值”表中（图 15.2）。
- 通过取“Group 1”参数估计值的负值来计算“Group 2”参数估计值。
- “倒数”显示“1/风险比”的值。

提示：要查看倒数值，请在“风险比”报表中右击，然后选择**列 > 倒数**。

对于本示例， Group 2/Group 1 的风险比计算如下：

exp[-(-0.2979479)] /exp(-0.2979479) = 1.8146558

该风险比值表明“Group 2”的死亡风险是“Group 1”的 1.81 倍。

图 15.3 “‘组’的风险比”表

风险比

“组”的风险比率

水平1	/水平2	风险比	概率>卡方	95 下限%	95 上限%
Group 2	Group 1	1.8146558	0.0872	0.9167104	3.5921661
Group 1	Group 2	0.5510687	0.0872	0.2783836	1.0908571

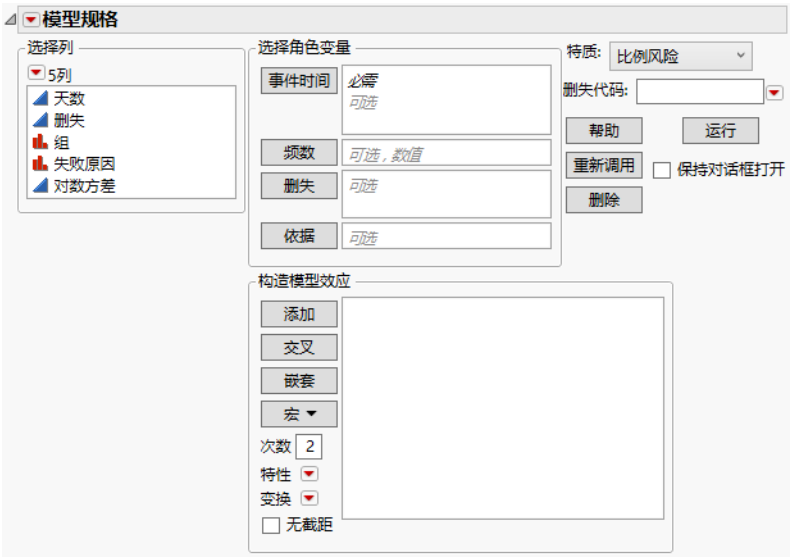
用于比值置信限效应的正态近似: 组

有关如何在有多个效应或分类效应包含两个以上水平时计算风险比的信息，请参见第 355 页的“多个效应和多个水平的风险比”。

启动“拟合比例风险”平台

通过选择**分析** > **可靠性和生存** > **拟合比例风险**来启动 “拟合比例风险”平台。

图 15.4 “拟合比例风险”启动窗口



提示：要更改 alpha 水平，请点击“模型规格”红色小三角并选择**设置 Alpha 水平**。

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

“拟合比例风险”启动窗口包含以下选项：

事件时间 包含事件时间或删失时间。

删失 删失列中的非零值指示删失观测。（删失值的编码通常等于 1。）未删失值必须编码为 0。

警告：“删失”列角色不接受 “缺失值代码”列属性。

频数 当记录多个单元时其值是每行的观测频数或计数的列。

依据 为分类或分组变量的每个水平执行单独的分析。

构造模型效应 使效应进入模型。有关 “构造模型效应”选项的详细信息，请参见 《拟合线性模型》。

特质 指示拟合方法。应始终选择 “比例风险”。

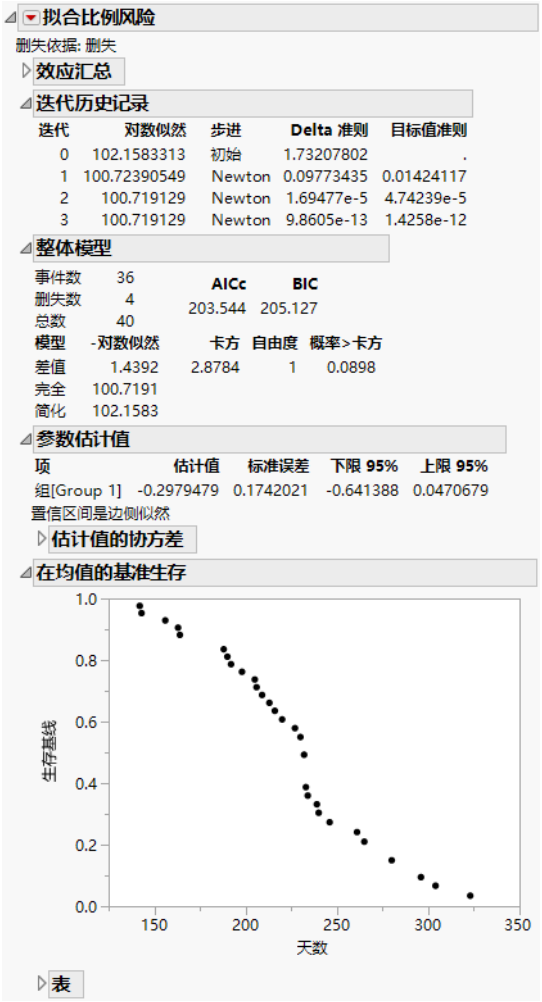
删失代码 标识“删失”列中指定右删失观测的值。选定某个 “删失”列后，JMP 会尝试自动检测删失代码并在框中显示该代码。要对此进行更改，您可以点击红色小三角并从值列表中

进行选择。您还可以在框中输入不同值。若“删失”列包含“值标签”列属性，值标签将显示在值列表中。缺失值会从分析中排除。

“拟合比例风险”报表

计算比例风险模型的参数估计值是一个迭代过程。拟合完成时，将显示图 15.5 中的报表。

图 15.5 “拟合比例风险”报表



迭代历史记录 列出在模型计算期间得到的迭代结果。

整体模型 显示有和没有协变量时的模型的似然函数的负数（– 对数似然）。它们的正差值的两倍用于执行以下假设的卡方检验：不同效应之间的生存时间没有差异。自由度 (DF) 等于完全模型和简化模型之间参数数目的差值。请参见 《拟合线性模型》。

参数估计值 显示协变量的参数估计值、其标准误差以及 95% 置信上限和置信下限。连续列不包括零的置信区间指示效应是显著的。分类列中某水平不包括零的置信区间指示该水平和所有水平的平均值之间的差异是显著的。

效应似然比检验 显示协变量效应的参数估计值是零这个原假设的似然比卡方检验。

在均值的基准生存 标绘数据中每个事件时间的基准函数估计值。在此处标绘“表”报表中的值。

“拟合比例风险”选项

“拟合比例风险”红色小三角菜单包含以下选项：

似然比检验 生成一些检验，用于比较拟合模型的对数似然与分别从模型删除每个项时的对数似然。

Wald 检验 为检验每个参数是否为 0 的 Wald 检验生成卡方检验统计量和 *p* 值。

似然置信区间 为每个参数指定 “参数估计值”表中所示的置信区间的类型。选定该选项后，即显示边侧似然置信区间；否则显示 Wald 区间。在该报表中，“参数估计值”表的下方会注明区间类型。若边侧似然置信区间的计算时间不长，则默认启用该选项。

注意：通过从 “拟合模型”启动窗口的红色小三角菜单中选择 “设置 Alpha 水平”，您可以更改置信区间的 α 水平。默认的 α 水平为 0.05。

风险比 显示效应的风险比。对于连续列，计算单位风险比和范围风险比。单位风险比为 Exp(估计值)，范围风险比为 Exp[估计值 *($x_{\text{最大值}} - x_{\text{最小值}}$)]。单位风险比显示回归变量的一个单位上的风险变化，范围风险比显示回归变量的整个范围上的风险变化。对于分类列，在单独的报表中显示每个效应的风险比。请注意，对于包含 *k* 个水平的分类变量，仅使用 *k* -1 个设计变量或水平。

提示：要在 “风险比”报表中查看倒数值，请在报表中右击并选择**列 > 倒数**。

模型对话框 显示当前分析的已完成的启动窗口。

效应汇总 显示交互式“效应汇总”报表，该报表支持您在模型中添加或删除效应。请参见《拟合线性模型》。

请参见《使用 JMP》获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存“依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

使用多个效应和多个水平的示例

本例对样本数据 VA Lung Cancer.jmp 使用比例风险模型。该数据是从随机化的临床试验收集的，在这些试验中对无法手术的男性肺癌患者实施了标准或新的（检验）化学疗法（**治疗**）。该试验的主要目的是评估治疗类型是否对生存时间有影响；特别关注肿瘤类型（**细胞类型**）的影响。有关该数据集的详细信息，请参见 Prentice (1973) 和 Kalbfleisch and Prentice (2002)。

对于比例风险模型，协变量包括以下项：

- 患者以前是否进行过这样的治疗（**之前治疗**）
- 患者的年龄（**年龄**）
- 从肺癌诊断到开始研究的时间（**诊断时间**）
- 常规医疗状况测度 (KPS)

年龄、**诊断时间**和 KPS 是连续测量值，**细胞类型**、**治疗**和**之前治疗**是分类（名义型）变量。**细胞类型**的四个名义型水平包括“Adeno”、“Large”、“Small”和“Squamous”。

本例说明具有多个效应且一个名义型效应具有两个以上水平的模型结果。还演示了风险比，给出计算连续效应的风险比和具有两个以上水平的效应的风险比的示例。

1. 选择**帮助** > **样本数据库**，然后打开 VA Lung Cancer.jmp。
2. 选择**分析** > **可靠性和生存** > **拟合比例风险**。
3. 选择**时间**作为**事件时间**。
4. 选择**删失**作为**删失**。
5. 选择**细胞类型**、**治疗**、**之前治疗**、**年龄**、**诊断时间**和 KPS，然后点击**添加**。
6. 点击**运行**。
7. （可选）点击“在均值的基准生存”标题栏上的展开图标以关闭图，然后点击“效应汇总”上的展开图标以关闭报表。

图 15.6 具有多个效应和水平的比例风险模型的报表窗口

拟合比例风险

删失依据: 删失

效应汇总

迭代历史记录

整体模型

		AICc	BIC
事件数	128	967.484	989.719
删失数	9		
总数	137		

模型	-对数似然	卡方	自由度	概率>卡方
差值	30.7046	61.4091	8	<.0001*
完全	475.1794			
简化	505.8840			

参数估计值

项	估计值	标准误差	下限 95%	上限 95%
细胞类型[Adeno]	0.57719588	0.1849861	0.2036319	0.9316196
细胞类型[Large]	-0.2114757	0.1741197	-0.56618	0.1191116
细胞类型[Small]	0.24538322	0.1592518	-0.070502	0.5550322
治疗[Standard]	-0.1449679	0.1036051	-0.348984	0.0578618
之前治疗[No]	-0.0361633	0.1160663	-0.259353	0.1970409
年龄	-0.0085494	0.0093042	-0.026447	0.0100846
诊断时间	-0.000092	0.0091251	-0.020233	0.016114
KPS	-0.0326217	0.0055052	-0.043442	-0.021834

置信区间是边侧似然

估计值的协方差

在均值的基准生存

效应似然比检验

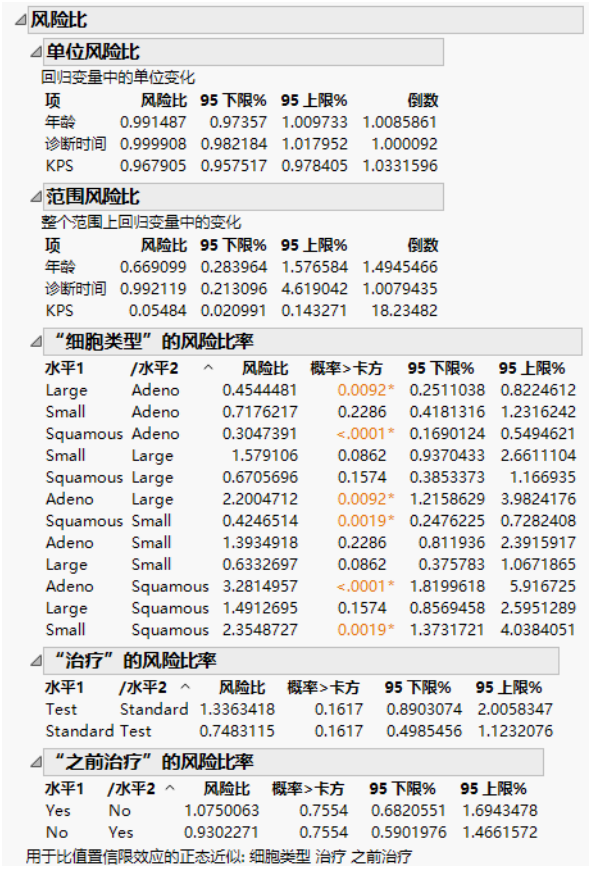
源	参数数目	自由度	似然比卡方	概率>卡方
细胞类型	3	3	18.6003364	0.0003*
治疗	1	1	1.96251687	0.1612
之前治疗	1	1	0.09639285	0.7562
年龄	1	1	0.82813759	0.3628
诊断时间	1	1	0.00010177	0.9920
KPS	1	1	34.616163	<.0001*

请注意以下有关结果的事项：

- 在“整体模型”报表中，较低的“概率>卡方”值 (<.0001) 指示当模型中至少包含一个效应时生存时间存在差异。
- 在“效应似然比检验”报表中，“概率>卡方”值指示 KPS 和**细胞类型**的至少一个水平是显著的；不过，**治疗**、**之前治疗**、**年龄**和**诊断时间**这些效应不显著。

8. 点击 “拟合比例风险”红色小三角并选择**风险比**。

图 15.7 “风险比”报表



风险比

单位风险比

回归变量中的单位变化

项	风险比	95 下限%	95 上限%	倒数
年龄	0.991487	0.97357	1.009733	1.0085861
诊断时间	0.999908	0.982184	1.017952	1.000092
KPS	0.967905	0.957517	0.978405	1.0331596

范围风险比

整个范围上回归变量中的变化

项	风险比	95 下限%	95 上限%	倒数
年龄	0.669099	0.283964	1.576584	1.4945466
诊断时间	0.992119	0.213096	4.619042	1.0079435
KPS	0.05484	0.020991	0.143271	18.23482

“细胞类型”的风险比率

水平1	/水平2	风险比	概率>卡方	95 下限%	95 上限%
Large	Adeno	0.4544481	0.0092*	0.2511038	0.8224612
Small	Adeno	0.7176217	0.2286	0.4181316	1.2316242
Squamous	Adeno	0.3047391	<.0001*	0.1690124	0.5494621
Small	Large	1.579106	0.0862	0.9370433	2.6611104
Squamous	Large	0.6705696	0.1574	0.3853373	1.166935
Adeno	Large	2.2004712	0.0092*	1.2158629	3.9824176
Squamous	Small	0.4246514	0.0019*	0.2476225	0.7282408
Adeno	Small	1.3934918	0.2286	0.811936	2.3915917
Large	Small	0.6332697	0.0862	0.375783	1.0671865
Adeno	Squamous	3.2814957	<.0001*	1.8199618	5.916725
Large	Squamous	1.4912695	0.1574	0.8569458	2.5951289
Small	Squamous	2.3548727	0.0019*	1.3731721	4.0384051

“治疗”的风险比率

水平1	/水平2	风险比	概率>卡方	95 下限%	95 上限%
Test	Standard	1.3363418	0.1617	0.8903074	2.0058347
Standard	Test	0.7483115	0.1617	0.4985456	1.1232076

“之前治疗”的风险比率

水平1	/水平2	风险比	概率>卡方	95 下限%	95 上限%
Yes	No	1.0750063	0.7554	0.6820551	1.6943478
No	Yes	0.9302271	0.7554	0.5901976	1.4661572

用于比值置信限效应的正态近似: 细胞类型 治疗 之前治疗

多个效应和多个水平的风险比

图 15.7 显示了连续效应（**年龄**、**诊断时间**、KPS）以及名义型效应（**细胞类型**、**治疗**、**之前治疗**）的风险比。为了便于说明，我们侧重 VA Lung Cancer.jmp 样本数据的连续效应**年龄**和具有四个水平的名义型效应（**细胞类型**）。

对于 VA Lung Cancer.jmp 样本数据中的连续效应**年龄**，按以下方式计算风险比：

单位风险比

$$\exp(\beta) = \exp(-0.0085494) = 0.991487$$

范围风险比

$$\exp[\beta(x_{max} - x_{min})] = \exp(-0.0085494 * 47) = 0.669099$$

对于名义型效应**细胞类型**，计算所有水平对并将结果显示在“‘细胞类型’的风险比”表中。请注意，对于包含 k 个水平的分类变量，仅使用 k -1 个设计变量或水平。在“参数估计值”表中，

只显示了**细胞类型**四个水平中的三个水平（“Adeno”、“Large”和“Small”）的参数估计值。未显示“Squamous”水平，但是它是其他估计值之和的负数。此处是“**细胞类型**的风险比”计算的两个示例：

Large/Adeno = $\exp(\beta_{Large})/\exp(\beta_{Adeno}) = \exp(-0.2114757)/\exp(0.57719588) = 0.4544481$

Squamous/Adeno = $\exp[-(\beta_{Adeno} + \beta_{Large} + \beta_{Small})]/\exp(\beta_{Adeno})$

$= \exp[-(0.57719588 + (-0.2114757) + 0.24538322)]/\exp(0.57719588) = 0.3047391$

“倒数”显示“1/风险比”的值。

附录 A

参考书目

Abernethy, R. B. (1996).*The New Weibull Handbook*.2nd ed. North Palm Beach, FL: Robert B. Abernethy.

Akaike, H. (1974).“A New Look at the Statistical Model Identification.”*IEEE Transactions on Automatic Control* AC-19:716–723.

Andrews, D. F., and Herzberg, A. M. (1985).*A Collection of Problems from Many Fields for the Student and Research Worker*.New York: Springer-Verlag.

Burnham, K. P., and Anderson, D. R. (2002).*Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*.2nd ed. New York: Springer-Verlag.

Chow, S.-C.(2007).*Statistical Design and Analysis of Stability Studies*.Boca Raton, FL: Chapman & Hall/CRC.

Cox, D. R. (1972).“Regression Models and Life-Tables.” *Journal of the Royal Statistical Society, Series B* 34:187–220.

Crow, L. H. (1975).*Reliability Analysis for Complex, Repairable Systems*.Technical Report No. 138, December 1975, US Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD.

Crow, L. H. (1982).“Confidence Interval Procedures for the Weibull Process with Applications to Reliability Growth.”*Technometrics* 24:67–72.

Escobar, L. A., Meeker, W. Q., Kugler, D. L., and Kramer, L. L. (2003).“Accelerated Destructive Degradation Tests: Data, Models, and Analysis.”In *Mathematical and Statistical Methods in Reliability*, edited by B. H. Lindqvist and K. A. Doksum, 319–338.London: World Scientific Publishing Company.

Genschel, U., and Meeker, W. Q. (2010).“A Comparison of Maximum Likelihood and Median-Rank Regression for Weibull Estimation.”*Quality Engineering* 22:236–255.

Guo, H., Mettas, A., Sarakakis, G., and Niu, P. (2010).“Piecewise NHPP Models with Maximum Likelihood Estimation for Repairable Systems.”In *2010 Proceedings - Annual Reliability and Maintainability Symposium (RAMS)*.New York: IEEE Press.

Hosmer, D. W., Jr., and Lemeshow, S. (1999).A*pplied Survival Analysis: Regression Modeling of Time-to-Event Data*.New York: John Wiley & Sons.

ICH Q1E.(2003).*Evaluation for Stability Data*.Tripartite International Conference on Harmonization Guideline Q1E, Geneva. https://database.ich.org/sites/default/files/Q1E_Guideline.pdf.

Kalbfleisch, J. D., and Prentice, R. L. (1980). *The Statistical Analysis of Failure Time Data*.New York: John Wiley & Sons.

Kalbfleisch, J. D., and Prentice, R. L. (2002).*The Statistical Analysis of Failure Time Data*.2nd ed. Hoboken, NJ: John Wiley & Sons.

Kaminskiy, M. P., and Krivtsov, V. V. (2005).“A Simple Procedure for Bayesian Estimation of the Weibull Distribution.”*IEEE Transactions on Reliability* 54:612–616.

Klein, J. P., and Moeschberger, M. L. (1997).*Survival Analysis: Techniques for Censored and Truncated Data*.New York: Springer-Verlag.

Lawless, J. F. (1982).*Statistical Models and Methods for Lifetime Data*.New York: John Wiley & Sons.

Lawless, J. F. (2003).*Statistical Models and Methods for Lifetime Data*.2nd ed. New York: John Wiley & Sons.

Lee, L., and Lee S. K. (1978).“Some Results on Inference for the Weibull Process.”*Technometrics* 20:41–45.

Liu, P., and Wang, P. (2013).“Competing Failure Modes Modeling with Limited Wearout Failures.”In *2013 Proceedings Annual Reliability and Maintainability Symposium (RAMS)*.New York: IEEE Press.

Meeker, W. Q., and Escobar, L. A. (1998).*Statistical Methods for Reliability Data*.New York: John Wiley & Sons.

Nair, V. N. (1984).“Confidence Bands for Survival Functions with Censored Data: A Comparative Study.”*Technometrics* 26:265–275.

Nelson, W. B. (1982).*Applied Life Data Analysis*.New York: John Wiley & Sons.

Nelson, W. B. (1985).“Weibull Analysis of Reliability Data with Few or No Failures.”*Journal of Quality Technology* 17:140–146.

Nelson, W. B. (1990).*Accelerated Testing: Statistical Models, Test Plans, and Data Analyses*.New York: John Wiley & Sons.

Nelson, W. B. (2003).*Recurrent Events Data Analysis for Product Repairs, Disease Recurrences, and Other Applications*.Philadelphia: Society for Industrial Mathematics.

Nelson, W. B. (2004).*Accelerated Testing: Statistical Models, Test Plans, and Data Analysis*.New York: John Wiley & Sons.

Prentice, R. L. (1973).“Exponential Survivals with Censoring and Explanatory Variables.”*Biometrika* 60:279–288.

Rigdon, S. E., and Basu, A. P. (2000).*Statistical Methods for the Reliability of Repairable Systems*.New York: John Wiley & Sons.

Robert, C. P., and Casella, G. (2004).*Monte Carlo Statistical Methods*.2nd ed. New York: Springer-Verlag.

SAS Institute Inc.(2020).“The LIFETEST Procedure.”In *SAS/STAT 15.2 User's Guide*.Cary, NC: SAS Institute Inc.
https://go.documentation.sas.com/api/docsets/statug/15.2/content/lifetest.pdf.

Si, S., Dui, H., Zhao, X., Zhang, S., and Sun, S. (2012).“Integrated Importance Measure of Component States Based on Loss of System Performance.”*IEEE Transactions on Reliability*.61:192–202.

Tobias, P. A., and Trindade, D. C. (1995).*Applied Reliability*.2nd ed. New York: Van Nostrand Reinhold.

Tobias, P. A., and Trindade, D. C. (2012).*Applied Reliability*.3rd ed. Boca Raton, FL: Chapman & Hall/CRC.

US Department of Defense (1981).*Military Handbook: Reliability Growth Management* (MIL-HDBK-00189).Washington, DC: US Department of Defense.

附录 B

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

版本 16

消费者研究

“真正的发现之旅，并不在于寻求新的景观，
而在于拥有新的眼光。”

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.0

本手册的准确文献资料出处为：SAS Institute Inc. 2020–2021。JMP® 16 消费者研究.Cary, NC: SAS Institute Inc.

JMP® 16 消费者研究

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved.Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government.Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007).If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation.The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

2021 年 3 月

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement.For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

充分利用 JMP

无论您是新手还是经验丰富的用户，都有需要了解的 JMP 知识。

访问 JMP.com 获取以下资源：

- JMP 入门知识网络讲座直播和录像
- 新功能和高级技巧视频演示和网络讲座
- 关于注册参加 JMP 培训的详细信息
- 您所在地区举办的研讨会安排
- 其他人使用 JMP 的成功案例
- JMP 用户社区、提供给用户的资源（包括插件和脚本示例）、论坛、博客、会议信息等

https://www.jmp.com/getstarted

目录

消费者研究

1 了解 JMP 9

文档和其他资源

JMP 文档中的格式规范 11

JMP 帮助 12

JMP 文档库 12

学习 JMP 的其他资源 18

JMP 教程 18

样本数据表 18

学习统计和 JSL 术语 19

学习 JMP 的提示与技巧 19

JMP 工具提示 19

JMP 用户社区 19

免费在线统计思维课程 19

JMP 新用户欢迎套件 20

统计学知识门户 20

JMP 培训 20

用户编写的 JMP 手册 20

“JMP 起始页”窗口 20

JMP 技术支持 20

2 消费者研究介绍 21

客户和行为研究方法概述

3 分类响应分析 23

分析调查数据和其他计数数据

“分类”平台示例 25

启动“分类”平台 31

响应角色 31

列角色 36

其他启动窗口选项 36
“分类”报表 37
“分类”平台选项 38
报表选项 38
统计检验选项 39
其他“分类”平台选项 40
交叉表选项 42
比较字母 43
超类别 44
超类别选项 44
“分类”平台的更多示例 45
响应齐性检验的示例 45
多重响应检验的示例 46
超类别示例 48
单元格卡方检验的示例 51
使用比较字母“比较每个样本”的示例 52
使用比较字母“比较每个单元格”的示例 53
使用比较字母的用户指定比较的示例 55
对齐的响应的示例 56
条件关联和相对风险的示例 57
评级员一致性的示例 59
重复测量的示例 60
多重响应的示例 61
带比较字母的得分均值的示例 66
结构化报表的示例 67
无响应的多重响应示例 69
设置首选项 71
“分类”平台的统计详细信息 72
Rao-Scott 校正 72
4 选择模型 73
针对选择实验拟合模型
“选择建模”平台概述 75
“选择”平台的示例 76

“无选择”的单表格式 76
多表格式 79
启动“选择”平台 84
针对“一个表，堆叠”的启动窗口 85
针对“多个表，交叉引用”的启动窗口 87
“选择模型”报表 91
效应汇总 92
参数估计值 93
似然比检验 93
Bayes 参数估计值 93
“选择”平台选项 94
支付意愿 97
更多示例 99
示例：做出设计决策 99
细分示例 107
使用“选择”平台进行 Logistic 回归的示例 111
配对的案例对照研究的 Logistic 回归示例 114
将数据转换为两个分析表的示例 115
将数据转换为一个分析表的示例 120
“选择”平台的统计详细信息 122
特殊数据表规则 122
效用和概率 123
梯度 123

5 MaxDiff 125

针对 MaxDiff 实验拟合模型

“MaxDiff 建模”平台概述 127
MaxDiff 平台的示例 127
单表格式 128
多表格式 130
启动 MaxDiff 平台 133
针对“一个表，堆叠”的启动窗口 134
针对“多个表，交叉引用”的启动窗口 135
“MaxDiff 模型”报表 138

效应汇总 139
MaxDiff 结果 140
参数估计值 141
Bayes 参数估计值 142
似然比检验 143
MaxDiff 平台选项 143
“比较”报表 145
保存 Bayes 链 145

6 提升 147
对市场活动对消费者行为带来的增量影响进行建模
“提升”平台概述 149
“提升”平台的示例 150
启动“提升”平台 152
“提升模型”报表 153
提升模型图形 153
“提升”报表选项 155

7 多因子分析 157
分析小组成员之间的一致性
“多因子分析”平台概述 159
多因子分析的示例 159
启动“多因子分析”平台 162
数据格式 163
“多因子分析”报表 164
汇总图 164
一致性图 165
“多因子分析”平台选项 165
“多因子分析”平台的统计详细信息 167

A 参考书目 169
B 技术许可声明 171

第 1 章

了解 JMP
文档和其他资源

了解 JMP 文档，例如手册规范、每个 JMP 文档的说明、“帮助”系统以及哪里可以查找到其他的支持。

目录

JMP 文档中的格式规范 11
JMP 帮助 12
JMP 文档库 12
学习 JMP 的其他资源 18
JMP 教程 18
样本数据表 18
学习统计和 JSL 术语 19
学习 JMP 的提示与技巧 19
JMP 工具提示 19
JMP 用户社区 19
免费在线统计思维课程 19
JMP 新用户欢迎套件 20
统计学知识门户 20
JMP 培训 20
用户编写的 JMP 手册 20
“JMP 起始页”窗口 20
JMP 技术支持 20

JMP 文档中的格式规范

以下规范有助于将书面材料与您在屏幕上看到的信息相联系：

- 样本数据表名称、列名、路径名称、文件名、文件扩展名和文件夹采用 Helvetica（或无衬线联机）字体显示。
- 代码采用 `Lucida Sans Typewriter`（或固定宽度联机）字体显示。
- 代码输出采用 *`Lucida Sans Typewriter`* 斜体（或固定宽度斜体联机）字体显示，并且相对于之前的代码缩进显示。
- **Helvetica 粗体**格式（或粗体无衬线联机）表示为了完成某个任务而选择的项：
  - 按钮
  - 复选框
  - 命令
  - 可供选择的列表名称
  - 菜单
  - 选项
  - 选项卡名称
  - 文本框
- 下列项采用**黑体**字体显示：
  - 重要的或具有特定 JMP 定义的字词
  - 变量
- 仅适用于 JMP Pro 的功能使用 JMP Pro 图标 JMP PRO 加以注释。对于 JMP Pro 功能概述，请访问 https://www.jmp.com/software/pro/。

注意：特殊信息和适用的局限性将在“注意”中显示。

提示：实用信息将在“提示”中显示。

JMP 帮助

使用 “帮助”菜单中的 “JMP 帮助”，您可以搜索有关 JMP 功能、统计方法以及 “JMP 脚本语言”（或 JSL）的信息。打开 “JMP 帮助”有以下几种方式：

- 在 Windows 系统中，通过选择**帮助 > JMP 帮助**搜索和查看“JMP 帮助”。
- 在 Windows 系统中，按 F1 键可在默认浏览器中打开“帮助”系统。
- 获取有关数据表或报表窗口特定部分的帮助。从**工具**菜单选择“帮助”工具 ，然后点击数据表或报表窗口中的任意位置，查看该部分的帮助。
- 在 JMP 窗口中，点击**帮助**按钮。

注意：“JMP 帮助”可供具有 Internet 连接的用户使用。没有 Internet 连接的用户可以通过选择**帮助 > JMP 文档库**搜索 PDF 文件格式的所有手册。详细信息，请参见第 12 页的“JMP 文档库”。

JMP 文档库

“帮助”系统内容也在一个称为 JMP 文档库的 PDF 文件中提供。选择**帮助 > JMP 文档库**可以打开该文件。若您更喜欢搜索 JMP 库中对应于每个文档的单个 PDF 文件，可以从 https://www.jmp.com/documentation 下载文件。

下表说明了 JMP 库中每个文档的用途和内容。

文档标题	文档用途	文档内容
发现 JMP	若您不熟悉 JMP，则从该文档开始学习。	介绍 JMP 以及指导如何创建和分析数据。还可学习如何共享结果。
使用 JMP	了解 JMP 数据表以及如何执行基本操作。	涵盖了贯穿整个 JMP 的常规 JMP 概念和功能，包括导入数据、修改列属性、数据排序以及连接至 SAS。

文档标题	文档用途	文档内容
基本分析	使用该文档执行基本分析。	介绍以下“分析”菜单平台： • 分布 • 以 X 拟合 Y • 制表 • 文本分析器 涵盖如何通过“分析”>“以 X 拟合 Y”执行二元分析、单因子方差分析和列联分析。还包括如何使用 Bootstrapping 来近似估计抽样分布以及如何使用“模拟”平台执行参数再抽样。
基本绘图	为您的数据找到最理想的图形展示方法。	介绍以下“图形”菜单平台： • 图形生成器 • 三维散点图 • 等高线图 • 气泡图 • 平行图 • 方格图 • 散点图矩阵 • 三元图 • 矩形树图 • 图表 • 叠加图 手册还涵盖如何创建背景地图和定制地图。
刻画器指南	学习如何使用交互式刻画工具，它们可以让您查看任何响应曲面的横截面。	涵盖“图形”菜单中列出的所有刻画器。还包括了对噪声因子的分析以及使用随机输入运行模拟。
实验设计指南	学习如何设计实验和确定适当的样本大小。	涵盖“实验设计”菜单中的所有主题。

文档标题	文档用途	文档内容
拟合线性模型	学习“拟合模型”平台及其许多特质。	介绍以下特质，它们都位于“分析”菜单“拟合模型”平台中： • 标准最小二乘法 • 逐步 • 广义回归 • 混合模型 • 多元方差分析 • 对数线性方差 • 名义型 Logistic • 有序型 Logistic • 广义线性模型

文档标题	文档用途	文档内容
预测和专业建模	学习其他建模技巧。	介绍以下“分析”>“预测建模”菜单平台： • 神经 • 分割 • Bootstrap 森林法 • 提升树 • K 最近邻 • 朴素 Bayes • 支持向量机 • 模型比较 • 模型筛选 • 生成验证列 • 公式存储库 介绍以下“分析”>“专业建模”菜单平台： • 拟合曲线 • 非线性 • 函数数据分析器 • 高斯过程 • 时间序列 • 配对 介绍以下“分析”>“筛选”菜单平台： • 建模工具 • 响应筛选 • 过程筛选 • 预测变量筛选 • 关联分析 • 过程历史分析器

文档标题	文档用途	文档内容
多元方法	获悉同时分析多个变量的技巧。	介绍以下“分析”>“多元方法”菜单平台： • 多元 • 主成分 • 判别 • 偏最小二乘 • 多重对应分析 • 结构化方程模型 • 因子分析 • 多维尺度化 • 项目分析 介绍以下“分析”>“聚类”菜单平台： • 层次聚类 • K 均值聚类 • 正态混合 • 潜在类分析 • 聚类变量
质量和过程方法	了解用于评估和改进过程的工具。	介绍以下“分析”>“质量和过程”菜单平台： • 控制图生成器和单个控制图 • 测量系统分析 • 变异性/计数量具图 • 过程能力 • 模型驱动的多元控制图 • 传统控制图 • Pareto 图 • 关系图 • 管理规格限 • OC 曲线

文档标题	文档用途	文档内容
可靠性和生存方法	学习评估和改进产品或系统的可靠性以及分析人或产品的生存数据。	介绍以下“分析”>“可靠性和生存”菜单平台： • 寿命分布 • 以 X 拟合寿命 • 累积损坏 • 复发分析 • 退化 • 破坏性退化 • 可靠性预测 • 可靠性增长 • 可靠性框图 • 可修复系统模拟 • 生存 • 拟合参数生存 • 拟合比例风险
消费者研究	学习研究消费者偏好的方法以及如何通过获得的洞察力创造更好的产品和服务。	介绍以下“分析”>“消费者研究”菜单平台： • 分类 • 选择 • MaxDiff • 提升 • 多重因子分析
Scripting Guide	学习如何使用功能强大的 JMP 脚本语言 (JSL)。	涵盖多方面主题，例如编写和调试脚本、操作数据表、构造显示框以及创建 JMP 应用程序。
JSL Syntax Reference	获悉许多 JSL 函数及其变元和发送至对象和显示框的消息。	包括 JSL 命令的语法、示例和注释。

学习 JMP 的其他资源

除了阅读 JMP 帮助外，还可以使用以下资源学习 JMP：

- “JMP 教程”
- “样本数据表”
- “学习统计和 JSL 术语”
- “学习 JMP 的提示与技巧”
- “JMP 工具提示”
- “JMP 用户社区”
- “免费在线统计思维课程”
- “JMP 新用户欢迎套件”
- “统计学知识门户”
- “JMP 培训”
- “用户编写的 JMP 手册”
- ““JMP 起始页”窗口”

JMP 教程

可以选择**帮助** > **教程**来访问 JMP 教程。**教程**菜单中的第一项是**教程目录**。这将打开一个新的窗口，其中包含按类别分组的所有教程。

若您不熟悉 JMP，则从**初学者教程**开始。它分步介绍了 JMP 界面并解释了使用 JMP 的基本操作。

其余教程有助于您了解 JMP 的特定方面，例如设计实验以及将样本均值与常数比较。

样本数据表

JMP 文档系列中的所有示例使用的都是样本数据。选择**帮助** > **样本数据库**以打开样本数据目录。

要查看按字母顺序列出的样本数据表或查看不同分类下的样本数据，选择**帮助** > **样本数据**。

样本数据表安装在以下目录：

在 Windows 上：C:\Program Files\SAS\JMP\16\Samples\Data

在 macOS 上：\Library\Application Support\JMP\16\Samples\Data

在 JMP Pro 中，样本数据安装在 JMPPRO（而不是 JMP）目录中。

要查看使用样本数据的示例，选择**帮助** > **样本数据**并导航到“教学资源”部分。要了解教学资源的更多信息，请访问 http://jmp.com/tools。

学习统计和 JSL 术语

有关统计术语的帮助，请选择 “帮助 > 统计索引”。有关 JSL 脚本和示例的帮助，请选择**帮助 > 脚本索引**。

统计索引 提供统计术语定义。

脚本索引 使您可以搜索有关 JSL 函数、对象和显示框的信息。您还可以从 “脚本索引”编辑和运行样本脚本以及获取有关命令的帮助。

学习 JMP 的提示与技巧

首次启动 JMP 时会看到“今日提示”窗口。该窗口提供使用 JMP 的一些小技巧。

要关闭“今日提示”，清除**启动时显示提示**复选框。要再次查看它，请选择**帮助 > 今日提示**。或者可以使用“首选项”窗口将其关闭。

JMP 工具提示

若您将鼠标悬停在下列项之上， JMP 会提供说明性工具提示 （或**悬停标签**）：

- 菜单或工具栏选项
- 图形中的标签
- 报表窗口中的文本结果（在结果上以圆圈的方式移动光标可显示提示）
- “主窗口”中的文件或窗口
- “脚本编辑器”中的代码

提示：在 Windows 上，可在“JMP 首选项”中隐藏工具提示。选择**文件 > 首选项 > 常规**，然后取消选择**显示菜单提示**。在 macOS 上该选项不可用。

JMP 用户社区

“JMP 用户社区”提供了多种选项，帮助您更好地学习 JMP 以及与其他 JMP 用户建立联系。学习资源库包含一页的指南、教程和演示，您可以从这里入手开始学习。之后您可以注册各种 JMP 培训课程以获得 JMP 进阶学习。

其他资源包括论坛、文件交换库（样本数据，脚本等）、网络学习视频以及社交网络小组等。

要访问网站上的 JMP 资源，选择**帮助 > JMP 用户社区**或访问 https://community.jmp.com。

免费在线统计思维课程

通过本免费在线课程的相关主题（例如探索性数据分析、质量方法以及相关和回归）学习实用统计技能。课程包括短片、演示、练习等等。访问 https://www.jmp.com/statisticalthinking。

JMP 新用户欢迎套件

“JMP 新用户欢迎套件”旨在帮助您快速熟悉 JMP 的基本操作。您将完成它的 30 个演示短片与活动，树立使用本软件的信心，以及与全球最大 JMP 用户网上社区建立联系。访问 https://www.jmp.com/welcome。

统计学知识门户

“统计学知识门户”使简明的统计解释与启发性的示例和图形相结合，帮助访问者奠定牢固的基础来逐步增强统计能力。访问 https://www.jmp.com/skp。

JMP 培训

SAS 提供各种由经验丰富的 JMP 专家团队领导的主题培训。提供公开课、网上实况课程和现场课程。您也可以选择网上在线学习订阅以在您方便的时候进行学习。访问 https://www.jmp.com/training。

用户编写的 JMP 手册

您可以从以下 JMP 网站获取 JMP 用户编写的有关使用 JMP 的其他手册。访问 https://www.jmp.com/books。

“JMP 起始页”窗口

若您不熟悉 JMP 或数据分析，可以从“JMP 起始页”窗口开始操作。在该窗口中，各种选项进行了分类并伴有说明，通过点击按钮即可启动。“JMP 起始页”窗口包括分析、图形、表和文件菜单中的许多选项。该窗口还列出 JMP Pro 功能和平台。

- 要打开“JMP 起始页”窗口，请选择**视图**（在 macOS 中选择**窗口**）> **JMP 起始页**。
- 在 Windows 中，要在打开 JMP 时自动显示“JMP 起始页”，请选择**文件** > **首选项** > **常规**，然后从“初始 JMP 窗口”列表中选择 **JMP 起始页**。在 macOS 中，选择 **JMP** > **首选项** > **初始 JMP 起始页窗口**。

JMP 技术支持

JMP 技术支持由在 SAS 和 JMP 接受过培训的统计学家和工程师提供，其中很多人具有统计学或其他技术学科的研究生学位。

许多技术支持选项在 https://www.jmp.com/support 中提供，包括技术支持电话。

第 2 章

消费者研究介绍

客户和行为研究方法概述

《消费者研究》提供一整套工具来分析消费者和行为研究数据。您收集关于客户如何使用产品或服务、客户对您提供的产品或服务是否满意，以及客户可能需要的新功能等方面的信息。对这些信息的深入分析让您能够创造出更好的产品和服务、令客户更加满意，并给自己的组织带来更丰厚的收入。用于分析这些消费者研究活动的工具位于“消费者研究”菜单中。可使用以下平台分析您的数据：

- “分类”平台支持您对分类响应数据（包括多重响应数据）进行制表、绘图以及比较。您可以使用该平台分析调查数据以及其他分类响应数据，如：缺陷记录和研究参与者的人口统计数据。使用“分类”平台，您可以分析以许多不同方式组织的数据表中的响应。请参见第 3 章，“分类响应分析”。
- “选择”平台专用于市场研究实验，其最终目标是发现消费者的偏好结构。然后我们根据得到的消费者对不同特性的偏好结果来设计并提供他们喜欢的产品或服务。请参见第 4 章，“选择模型”。
- “MaxDiff”平台可替代使用标准偏好尺度来确定评级项的相对重要性。MaxDiff 实验强制响应者报告其最喜欢和最不喜欢的选项，从而强制响应者按偏好程度对选项排名。请参见第 5 章，“MaxDiff”。
- “提升”平台支持您通过将特价信息仅发送给可能积极响应的个人来最大化市场营销预算的影响，即便是在您的数据集非常大而且有许多可能的行为或人口统计预测变量的情况下。您可以使用提升模型进行此类预测。开发该方法是为了帮助优化市场营销决策、确定个性化的药物方案，或是更加宽泛地来讲 — 确定很可能响应某种行为的个人的特征。请参见第 6 章，“提升”。
- “多因子分析”平台支持您分析感官数据分析中的小组成员之间的一致性。您可以使用 MFA 对研究进行分析，研究中针对相同或不同特性通过不同的手段、个人或在不同情形下对项进行测量。请参见第 7 章，“多因子分析”。

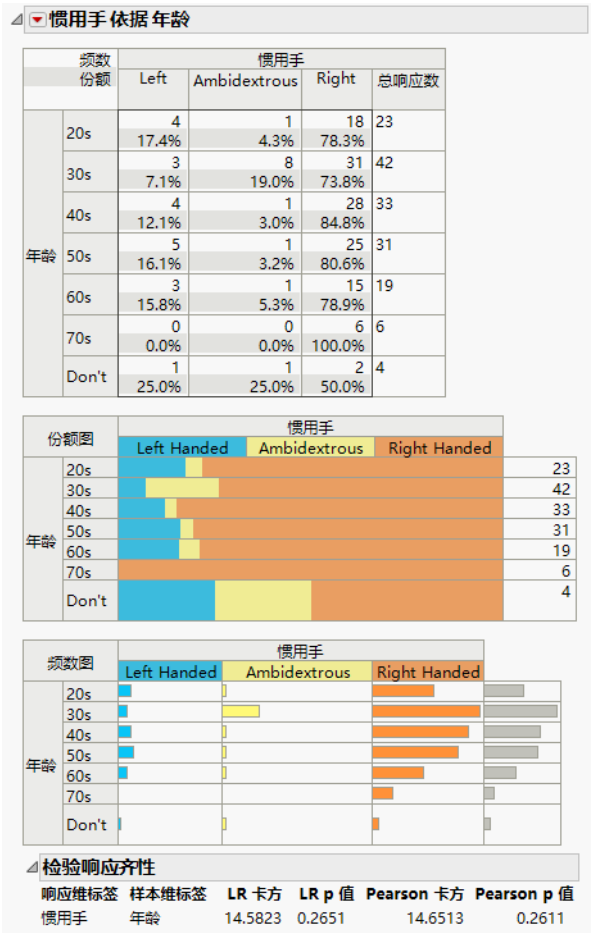
第 3 章

分类响应分析

分析调查数据和其他计数数据

“分类”平台支持您对分类响应数据（包括多重响应数据）进行制表、绘图以及比较。您可以分析调查数据以及其他分类响应数据，如：缺陷记录和研究参与者的人口统计数据。支持许多不同的数据类型和格式。

图 3.1 分类分析示例



惯用手 依据 年龄

频数 / 份额		惯用手 Left	Ambidextrous	Right	总响应数
年龄	20s	4 17.4%	1 4.3%	18 78.3%	23
	30s	3 7.1%	8 19.0%	31 73.8%	42
	40s	4 12.1%	1 3.0%	28 84.8%	33
	50s	5 16.1%	1 3.2%	25 80.6%	31
	60s	3 15.8%	1 5.3%	15 78.9%	19
	70s	0 0.0%	0 0.0%	6 100.0%	6
	Don't	1 25.0%	1 25.0%	2 50.0%	4

份额图

频数图

检验响应齐性

响应维标签	样本维标签	LR 卡方	LR p 值	Pearson 卡方	Pearson p 值
惯用手	年龄	14.5823	0.2651	14.6513	0.2611

目录

“分类”平台示例 25
启动“分类”平台 31
响应角色 31
列角色 36
其他启动窗口选项 36
“分类”报表 37
“分类”平台选项 38
报表选项 38
统计检验选项 39
其他“分类”平台选项 40
交叉表选项 42
比较字母 43
超类别 44
超类别选项 44
“分类”平台的更多示例 45
响应齐性检验的示例 45
多重响应检验的示例 46
超类别示例 48
单元格卡方检验的示例 51
使用比较字母“比较每个样本”的示例 52
使用比较字母“比较每个单元格”的示例 53
使用比较字母的用户指定比较的示例 55
对齐的响应的示例 56
条件关联和相对风险的示例 57
评级员一致性的示例 59
重复测量的示例 60
多重响应的示例 61
带比较字母的得分均值的示例 66
结构化报表的示例 67
无响应的多重响应示例 69
设置首选项 71
“分类”平台的统计详细信息 72
Rao-Scott 校正 72

“分类”平台示例

本例使用 Color Preference Survey.jmp 样本数据表，其中包含关于颜色首选项的调查数据。您可以使用 “分类”平台汇总来自不同问题类型的结果。

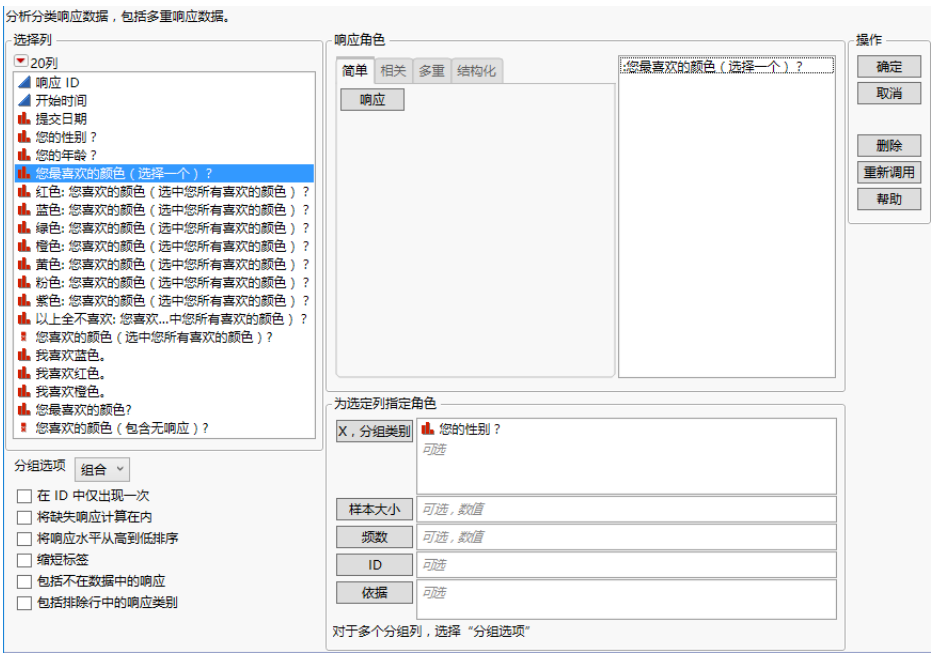
- 单个响应问题
- 三个对齐的排名问题
- 多重响应问题
- 由次要问题分段的单个响应问题

您将在“分类”启动窗口中设置四个分析（每种问题类型对应一个分析）。或者，您可以一次使用一个启动选项卡。

1. 选择**帮助** > **样本数据库**，然后打开 Color Preference Survey.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择**您最喜欢的颜色？（选择一个）**，然后点击**响应**。
4. 选择**您的性别？**并点击 **X，分组类别**。

注意：若您想要按性别运行最喜爱颜色的单个分析，请立即点击**确定**。

图 3.2 “分类”启动窗口中的已完成 “简单”选项卡



5. 选择**相关**选项卡。

6. 选择**我喜欢蓝色**。一直到**我喜欢橙色**。并点击**对齐的响应**。

   这将输入三个问题以及要一同分析的评级尺度。性别分组类别应用到该分析。

7. 选择**多重**选项卡。
8. 选择**红色：您喜欢的颜色 （选中您所有喜欢的颜色）？**一直到**以上全不喜欢：您喜欢的颜色（选中您所有喜欢的颜色）？**，然后点击**多重响应**。

   这将输入多重响应问题“您喜欢什么颜色？”，其中每个响应都位于要分析的单值列中。性别分组类别应用到该分析。

9. 选择**结构化**选项卡。
10. 将**您最喜欢的颜色 （选择一个）？**拖至侧面绿色箭头，将**您的年龄？**拖至顶部绿色箭头。

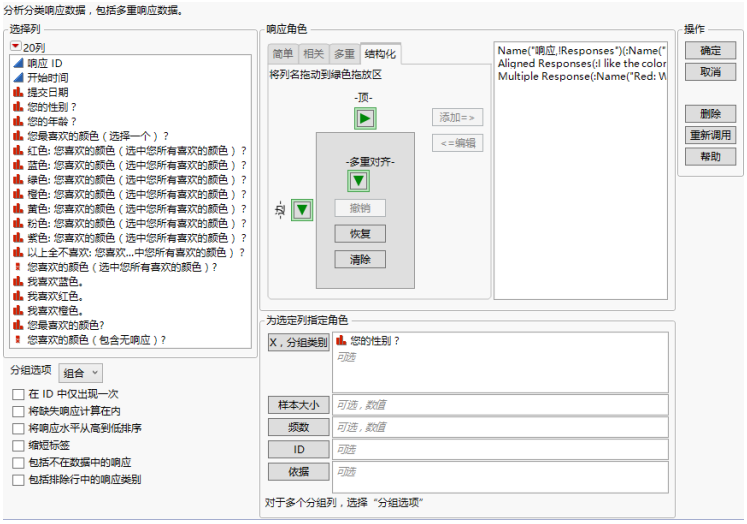
图 3.3 “分类”启动窗口中的“结构化”选项卡

11. 点击**添加**。

    在“X，分组类别”角色中指定的**您的性别？**列不应用到结构化分析中。

提示：点击“结构化”选项卡中的绿色箭头以显示列列表，从中可点击以添加列。可以使用顶部的向下箭头或侧面的右箭头嵌套多个列。

图 3.4 “完整示例”的已完成启动窗口



12. 点击**确定**。

每个分析的结果都垂直堆叠在单个报表窗口中。

“简单”选项卡报表

第一部分显示简单响应调查问题“您最喜欢的颜色？”（按性别分组）的结果。每个响应者需要从列表中选择一种颜色：红色、蓝色、绿色、橙色、黄色、粉色、紫色或以上全不是。

图 3.5 简单响应：按 “性别”划分最喜欢的颜色

分类

您最喜欢的颜色（选择一个）？依据 您的性别？

频数 / 份额		您最喜欢的颜色（选择一个）？								
		Blue	Green	None of the	Orange	Pink	Purple	Red	Yellow	总响应数
您的性别？	Female	11 29.7%	4 10.8%	6 16.2%	0 0.0%	1 2.7%	10 27.0%	3 8.1%	2 5.4%	37
	Male	19 44.2%	8 18.6%	5 11.6%	2 4.7%	1 2.3%	3 7.0%	5 11.6%	0 0.0%	43

分析中显示两个性别最喜欢的颜色都是蓝色。对于男性，43 位男性响应者选择蓝色的频数为 19。这对应于 44.2% 的份额（或 19/43）。对于女性，37 位女性中有 11 位（频数）或 29.7%（份额）选择了蓝色。

提示：您可以使用“值颜色”列属性定义图中的颜色。请参见《使用 JMP》。

“相关”选项卡报表

报表的第二部分显示三个相关问题（询问响应者对某种颜色的喜爱程度的排名）的结果。可以将这些问题的响应对齐，因为这些问题使用了相同的从“完全同意”到“完全不同意”的评级尺度。

图 3.6 对齐响应：按性别对颜色排名

对齐的响应(我喜欢蓝色。, 我喜欢红色。, 我喜欢橙色。) 依据 您的性别？

				频数 / 份额	响应 Strongly	Disagree	Neutral	Agree	Strongly	总响应数
响应	我喜欢蓝色。	您的性别？	Female		1	1	4	13	18	37
					2.7%	2.7%	10.8%	35.1%	48.6%	
			Male		0	2	0	17	24	43
					0.0%	4.7%	0.0%	39.5%	55.8%	
	我喜欢红色。	您的性别？	Female		1	6	9	14	7	37
					2.7%	16.2%	24.3%	37.8%	18.9%	
			Male		1	4	3	24	11	43
					2.3%	9.3%	7.0%	55.8%	25.6%	
	我喜欢橙色。	您的性别？	Female		6	5	16	7	2	36
					16.7%	13.9%	44.4%	19.4%	5.6%	
			Male		2	6	12	15	8	43
					4.7%	14.0%	27.9%	34.9%	18.6%	

				份额图	响应 Strongly	Disagree	Neutral	Agree	Strongly	
响应	我喜欢蓝色。	您的性别？	Female							37
			Male							43
	我喜欢红色。	您的性别？	Female							37
			Male							43
	我喜欢橙色。	您的性别？	Female							36
			Male							43

该分析显示蓝色对于男性和女性都是排名最高的颜色。与红色和蓝色相比，橙色对于男性和女性都具有较高数量的中性结果。值颜色列属性用于定义份额图中使用的颜色。

注意：若使用“简单”选项卡单独分析对齐的问题时，您会得到三个单独的报表，其中包含与相关报表相同的信息。使用“相关”选项卡对齐这些报表，以便结果相互之间易于比较。

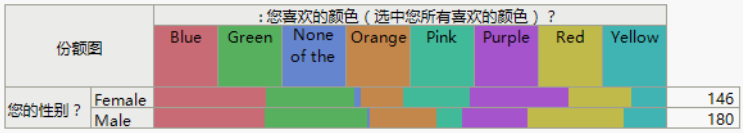
“多重”选项卡报表

报表中的第三部分显示按性别分组的多项选择调查问题的结果。该问题是“您喜欢的颜色（选中您所有喜欢的颜色）？”。每个可能的响应都收集到一个单独的列中。若测试对象选择了某个响应，则对应列中会有一个值。否则，该列为空。

图 3.7 多重响应：按性别排列的喜欢的颜色

结构化

频数 份额 比率		:您喜欢的颜色（选中您所有喜欢的颜色）？										
		Blue	Green	None of the	Orange	Pink	Purple	Red	Yellow	总响应数	总案例数	响应的总案例数
您的性别？	Female	32 21.9% 86.5%	25 17.1% 67.6%	2 1.4% 5.4%	12 8.2% 32.4%	19 13.0% 51.4%	28 19.2% 75.7%	18 12.3% 48.6%	10 6.8% 27.0%	146	37	37
	Male	39 21.7% 90.7%	36 20.0% 83.7%	1 0.6% 2.3%	23 12.8% 53.5%	9 5.0% 20.9%	23 12.8% 53.5%	34 18.9% 79.1%	15 8.3% 34.9%	180	43	43



从结果中可以观察到：蓝色和绿色是男性和女性都高度喜爱的颜色。与男性相比，女性喜爱粉色和紫色的程度更高。

在多重响应问题中，允许进行多项选择。一个案例代表一个响应者，“总案例数”即响应者总数。在我们的数据中，有 37 个女性案例和 43 个男性案例。响应的总案例数是选择一个或多个响应的案例数（在我们的数据中即所有人）。

“总响应数”列列出所做的选择总数。对于女性，37 位女性选择了 146 种颜色。对于男性，43 位男性选择了 180 种颜色。表中包含选择每种颜色的响应者数目、占总响应数的份额或百分比，以及占总案例数的比率或百分比。

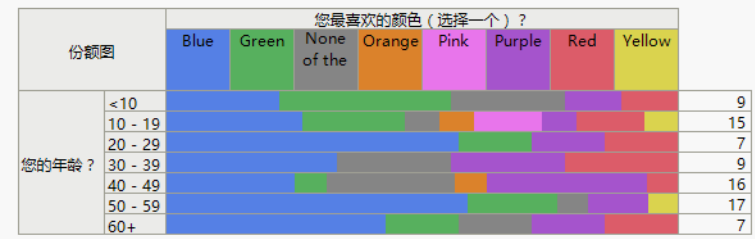
“结构化”选项卡报表

最后一部分显示从“结构化”选项卡生成的表，在该选项卡中，您可以将最喜欢的颜色问题设置为按年龄组分组所关注的响应。在结构化格式中，所关注的主响应的水平构成表的各行，分组变量的水平构成各列。

图 3.8 结构化：按年龄排列的最喜欢的颜色

您最喜欢的颜色（选择一个）？依据 您的年龄？

频数 份额		您的年龄？						
		<10	10 - 19	20 - 29	30 - 39	40 - 49	50 - 59	60+
您最喜欢的颜色（选择一个）？	Blue	2 22.2%	4 26.7%	4 57.1%	3 33.3%	4 25.0%	10 58.8%	3 42.9%
	Green	3 33.3%	3 20.0%	1 14.3%	0 0.0%	1 6.3%	3 17.6%	1 14.3%
	None of the	2 22.2%	1 6.7%	0 0.0%	2 22.2%	4 25.0%	1 5.9%	1 14.3%
	Orange	0 0.0%	1 6.7%	0 0.0%	0 0.0%	1 6.3%	0 0.0%	0 0.0%
	Pink	0 0.0%	2 13.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%
	Purple	1 11.1%	1 6.7%	1 14.3%	2 22.2%	5 31.3%	2 11.8%	1 14.3%
	Red	1 11.1%	2 13.3%	1 14.3%	2 22.2%	1 6.3%	0 0.0%	1 14.3%
	Yellow	0 0.0%	1 6.7%	0 0.0%	0 0.0%	0 0.0%	1 5.9%	0 0.0%
	总响应数	9	15	7	9	16	17	7



10 至 19 岁年龄组有两名响应者，他们最喜欢粉色。其他年龄组没有任何响应者最喜欢粉色。从表的最下面一行可以看出，每个年龄组中的响应者数量很少。您可以收集更多的数据，以得出关于不同年龄组最喜欢的颜色的结论。

启动 “分类” 平台

通过选择**分析** > **消费者研究** > **分类**来启动 “分类” 平台。

图 3.9 “简单” 选项卡的 “分类” 平台启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

响应角色

启动窗口包括用于三组响应角色的选项卡（“简单”、“相关” 和 “多重”）以及一个供您创建定制数据汇总的 “结构化” 选项卡。响应角色对应于您要分析的响应类型。每个选项卡上的选项对应于响应在数据表中的结构。

“简单” 选项卡

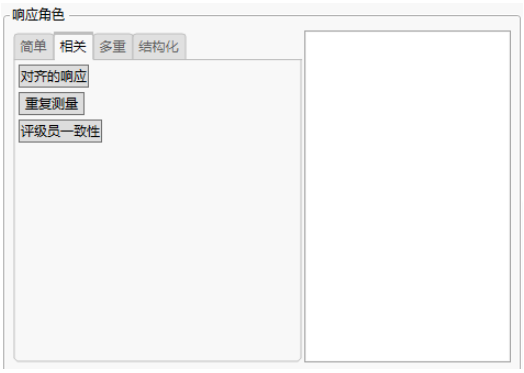
使用 “简单” 选项卡可分析单个响应，例如只允许单一答案的调查问题。要分析的数据包含在单个列中。

响应 汇总单个列中的数据。若选定多列，则分类报表包含针对每一单列的单独报表。

“相关”选项卡

使用“相关”选项卡分析跨多个列的一组相关响应。

图 3.10 “分类”平台启动窗口的“相关”选项卡



对齐的响应 在单个报表中汇总响应水平相同的多个列中的数据。该选项适用于很多问题都具有同一组响应的调查数据。您可以快速汇总和一起比较所有问题的响应趋势。

重复测量 汇总多个列中的数据，其中的每一列都包含不同时点针对同一问题的响应。若某人在多个时点都作出响应，则样本被称为重叠。若有重叠样本，则使用 Kish 校正。请参见 Kish (1965, sec. 12.4)。

评级员一致性 汇总多个列中的数据，其中的每一列都是不同的个人（评级员）对同一问题或项目的评级。

“多重”选项卡

使用“多重”选项卡分析多重响应，其中的响应记录在一列或多列中。“多重”选项卡上的选项特定于数据在数据表中的组织方式。一组多重响应可以来自一个调查，其中的响应集允许多项选择（“选中所有适用选项”问题）；也可以来自一组缺陷数据，其中一个项可有多个缺陷。

图 3.11 “分类”平台启动窗口的“多重”选项卡

多重响应 汇总来自多重响应的数据，其中每个可能的响应都记录在其各自的列中。每一列都可以包含空白，空白对应于未选中的项。

图 3.12 多重响应列格式

	红色: 您喜欢的颜色（选中您所有喜欢的颜色）？	蓝色: 您喜欢的颜色（选中您所有喜欢的颜色）？	绿色: 您喜欢的颜色（选中您所有喜欢的颜色）？	橙色: 您喜欢的颜色（选中您所有喜欢的颜色）？	黄色: 您喜欢的颜色（选中您所有喜欢的颜色）？	粉色: 您喜欢的颜色（选中您所有喜欢的颜色）？	紫色: 您喜欢的颜色（选中您所有喜欢的颜色）？	以上全不喜欢: 您喜欢的颜色（选中您所有喜欢的颜色）？
1	Red	Blue				Pink	Purple	
2	Red	Blue	Green					
3	Red	Blue	Green	Orange			Purple	
4								None of the above
5		Blue	Green	Orange			Purple	
6	Red	Blue	Green				Purple	
7	Red	Blue		Orange				
8	Red		Green	Orange				
9	Red	Blue	Green				Purple	
10	Red	Blue	Green	Orange	Yellow	Pink	Purple	

由 ID 标识的多重响应 汇总以堆叠格式记录数据的多重响应中的数据。有一个响应列，同时附带另一列，其中包含测试对象的 ID。

图 3.13 由 ID 标识的多重响应格式

	响应 ID	您最喜欢的颜色?
1	2	Red
2	2	Blue
3	2	
4	2	
5	2	
6	2	Pink
7	2	Purple
8	2	
9	3	Red
10	3	Blue

分隔的多重响应 汇总来自多重响应的数据，其中响应位于单个列中，并且每个响应由逗号、分号或制表符分隔。

图 3.14 分隔的多重响应格式

	您喜欢的颜色（选中您所有喜欢的颜色）?
1	Red, Blue, , , , Pink, Purple,
2	Red, Blue, Green, , , , ,
3	Red, Blue, Green, Orange, , , Purple,
4	, , , , , , , None of the above
5	, Blue, Green, Orange, , , Purple,
6	Red, Blue, Green, , , , Purple,
7	Red, Blue, , Orange, , , ,
8	Red, , Green, Orange, , , ,
9	Red, Blue, Green, , , , Purple,
10	Red, Blue, Green, Orange, Yellow, Pink, Purple,

提示：对于包含所述的多重响应的列，请使用 “多重响应” 列属性。请参见 《使用 JMP》。

指示符组 汇总存储在指示符列中的多重响应。在该数据表中，每个可能的响应都对应一列，每一列都是一个指示符（例如，0 和 1）。空值指示缺少响应。请参见第 64 页的 “指示符组”。

响应频数 汇总存储在包含频数统计的列中的多重响应。该数据格式是汇总的 “指示符组” 格式。请参见第 65 页的 “响应频数”。

自由文本 汇总文本数据。“自由文本” 选项可在 “分类” 报表窗口中启动 “文本资源管理器” 报表。请参见 《基本分析》。

“结构化” 选项卡

使用 “结构化” 选项卡构造定制交叉表。

图 3.15 “结构化” 选项卡的 “分类” 平台启动窗口

“结构化” 选项卡有三个拖放区，用于将列分配给角色。将列名拖动到绿色拖放区箭头。或者，点击列列表的绿色箭头以选择列。生成的结构化表将表的一侧上最里面的项视作响应，将其他所有项都视作分组因子。

拖放区

顶 将一个或多个数据表列分配给交叉表的列。分配一列后，使用向下箭头在已分配的列中嵌套其他列，或使用向右箭头添加其他列组。

边 将一个或多个数据表列分配给交叉表的行。分配一列后，使用向右箭头在已分配的行中嵌套其他列，或使用向下箭头添加其他行组。

多重对齐 分配两个或更多具有对齐响应的列。

控件

撤销 点击以撤销列分配。

恢复 点击以恢复最近撤销的分配。

清除 点击以删除拖放区中的所有分配。

添加=> 点击以将结构化表添加到结构化表列表。

<=编辑 点击可对结构化表列表中选定的表进行更改。

列角色

分类平台启动窗口具有以下列角色：

X，分组类别 （不适用于“结构化”选项卡。）将某列指定为分组类别。为每组都汇总响应。若使用不止一个分组列，则默认情况下表将被嵌套。使用启动窗口中的“分组”选项以更改汇总。

样本大小 分配一列，用该列的值定义频数所适用的组中各个单元的数目。样本大小用于具有汇总数据的多个响应角色。请参见第 61 页的 “多重响应的示例”。

频数 分配一列，用该列的值定义要分析的每行的频数。频数角色用于汇总数据。

ID 分配用于标识响应者的列。选定“由 ID 标识的多重响应”时才需要该选项，该选项不用于任何其他响应类型。

依据 为“依据”变量的每个水平生成单独报表。若指定了多个“依据”变量，将为“依据”变量水平的每种可能组合生成单独的报表。

其他启动窗口选项

更多选项位于启动窗口的左下角。或者，可以从平台报表窗口中的“分类”红色小三角菜单中选择这些选项。

分组选项 指定了多个分组列时，定义如何在分析中使用分组变量。

组合 分析分组变量组合的响应。分组列表中的第一列是交叉表中最外层的组。

单独 单独分析每个分组变量的响应。

二者 为分组变量的组合以及逐一为每个分组变量提供报表。

在 ID 中仅出现一次 将计数限定为某个参与者的非重复响应水平数。使用 ID 角色将某列指定为 ID。选定“由 ID 标识的多重响应”时该选项才适用，该选项不用于任何其他响应类型。

将缺失响应计算在内 指定将缺失值作为类别包含在内。缺失值可为空单元格，或是在“缺失值代码”列属性中定义的缺失代码。若某列仅包含缺失值，无论是否选择该选项，都会统计缺失值。对于多重响应，若所有响应类别都缺失（或者对于指标，若所有响应类别都为零），则认为响应缺失。

注意：若未选定该选项，将从分析中排除缺失值。

将响应水平从高到低排序 从高到低对响应排序。（默认排序为从低到高。）该选项仅适用于响应，不适用于分组类别。

提示：使用“值顺序”列属性定义特定的类别排序。请参见《使用 JMP》。

缩短标签 通过删除所有标签共有的前缀和后缀来缩短值标签。

注意：该选项仅适用于值标签，不适用于列名。

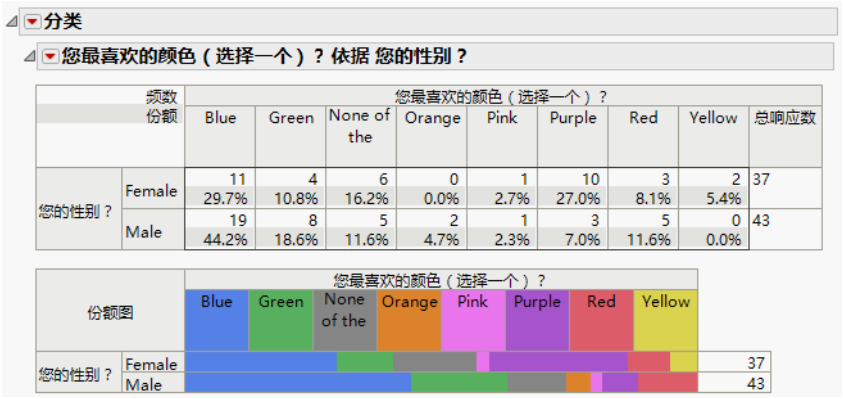
包括不在数据中的响应 指定报表中包括不含任何响应的类别。必须在 “值标签”列属性中指定没有响应的类别。该选项仅适用于响应。对于分组变量表，只包括具有响应的类别。

包括排除行中的响应类别 指定仅在排除的行中显示的响应类别将包含在报表中。这些类别的计数为 0。

“分类”报表

初始 “分类”报表为每组选定的响应显示一个交叉表和一个份额图。

图 3.16 初始分类报表



表的左上角列出该表中每个单元格中包含的数量（“频数”、“份额”和 “比率”（若适用））。使用 “分类”红色小三角菜单中的选项删除或添加这些量。

- 为每个类别都提供了“频数统计”（标记为“频数”），且总频数（总响应数）显示在表右侧。若有多重响应，表右侧的汇总列还会提供案例数或行数（总案例数）以及响应者数目（响应的总案例数）。
- “响应的份额”（份额）通过用每个计数（频数）除以总响应数来确定。
- “比率”是响应频数（频数）除以案例的总数（总案例数）的结果。此数量仅对多重响应显示，不显示在图 3.16 中。

在图 3.16 中，针对 “您最喜欢的颜色？”这一问题的响应数按性别来制表。考虑带有女性结果的表的第一行：

- 表的第一个单元格包含 11 个响应。这是将蓝色选作自己最喜欢的颜色的女性响应者的计数。
- 女性共有 37 个响应。在这 37 个女性响应中，只有 29.7% (11/37) 将蓝色选作她们最喜欢的颜色。

“分类”平台选项

“分类”红色小三角菜单包含支持您定制报表和请求各种统计检验的选项。可以使用的特定选项由响应角色、是否使用分组类别以及在启动窗口中选定的选项来确定。

- “报表选项”
- “统计检验选项”
- “其他“分类”平台选项”
- “交叉表选项”

报表选项

使用以下选项可以自定义报表的外观：

交叉表选项

频数 在交叉表中显示或隐藏频数 （标记为 “频数”）。频数是每个类别中的响应计数。

响应的份额 在交叉表中显示或隐藏响应的份额 （标记为 “份额”）。响应的份额是响应在每个类别中的百分比。

每种情况的比率 （仅适用于多重响应。 ）显示或隐藏交叉表中每种情况的比率 （标记为 “比率”）。每种情况的比率是基于情况总数（无论是否为响应者）的每个类别中的响应的百分比。

每种情况的响应率 （仅适用于多重响应。）在交叉表中显示或隐藏 “每种情况的响应率”。每种情况的响应率是基于响应情况数的每个类别中的响应百分比。

图表选项

份额图 显示或隐藏一个分割条形图。直条长度与每种类型的响应百分比成正比。右侧的列显示每个分组类别中的响应数。若未使用分组类别，则右侧列显示响应总数。

频数图 显示或隐藏频数图。其中的直条反映每组内的响应频数。整个图中的刻度都是一致的。最右侧的灰条表示每个分组类别中的响应总数。

转置的频数图 显示或隐藏转置的频数图。其中的直条反映每组内的响应频数。响应对应行，分组水平对应图中的列。每个分组水平的合计通过图底部行中的灰条来表示。

提示：您可以使用“值颜色”列属性更改份额图中的颜色。请参见《使用 JMP》。

交叉表查看选项

交叉表 显示或隐藏交叉表。交叉表将响应类别显示为列标题，将分组水平 （若使用）显示为行标签。表的左上角单元格显示该表中每个单元格中的项的标签 （“频数”、“份额”、“比率”和 “每种情况的响应率”）。若报表包含转置的交叉表，则该选项还将从报表中删除转置的交叉表。

交叉表转置 显示或隐藏转置的交叉表。转置的交叉表将响应类别显示为行标签，将分组水平（若使用）显示为列标题。表的左上角单元格显示该表每个单元格中的项的标签（“频数”、“份额”、“比率”和“每种情况的响应率”）。若报表包含交叉表，则该选项还将从报表中删除该交叉表。

统计检验选项

可用的统计检验选项取决于响应角色以及是否在分析中使用分组变量。选项包括针对多重响应、响应齐性、关联、相对风险和一致性等方面的检验。选项取决于响应数据类型和分组变量（X，分组类别）数据类型。

检验多重响应 （仅适用于具有一个或多个分组类别的多重响应数据。）请参见第 46 页的“多重响应检验的示例”。包含跨各个分组类别的响应的独立性检验：

计数检验，Poisson 显示或隐藏使用 Poisson 回归的比率独立性检验。每单位频数通过样本分类变量来建模。结果是一个似然比卡方检验，检验每个单独响应的比率在各分组水平之间是否不同。

齐性检验，二项 显示或隐藏每个单独的响应水平的似然比卡方独立性检验。每个响应类别都服从二项分布（选定或未选定）。

排除非响应 为多重响应类别的计数和齐性检验排除非响应。如果某行在所有响应类别中都缺少数据，则将其视为无响应。

检验响应齐性 （仅适用于没有多重响应建模类型且指定了一个或多个分组变量的响应变量。）显示或隐藏包含依赖于您的分组变量的响应齐性检验的报表：

- 当分组变量不是多重响应时，则需要检验跨分组变量水平的响应的独立性。提供似然比和 Pearson 卡方检验。请参见第 45 页的“响应齐性检验的示例”。
- 当分组变量是多重响应时，则需要检验跨分组变量**各个**水平的响应的独立性。提供 Rao-Scott 卡方检验。

单元格卡方 为卡方独立性检验显示或隐藏表中每个单元格的 *p* 值。较小的 *p* 值指示单元格中的观测值大于或小于期望值（假设前提是：行独立于列）。根据该计数是大于还是小于期望值对 *p* 值着色。请参见第 51 页的“单元格卡方检验的示例”。

比较每个样本 （仅适用于具有一个或多个分组变量的单个响应。）显示或隐藏不同水平组之间响应的独立性检验报表，其中包括配对似然比和 Pearson 卡方。请参见第 52 页的“使用比较字母“比较每个样本”的示例”。

比较每个单元格 （仅适用于具有一个或多个分组变量的单个和多重响应。）针对响应的每个水平与分组变量各水平组合而成的其他所有水平之间的独立性对比，显示或隐藏配对似然比卡方、 Pearson 卡方和 Fisher 精确检验。请参见第 53 页的“使用比较字母“比较每个单元格”的示例”。

相对风险 （在分组变量有两个水平、响应有两个水平或为多重响应，并且已选定“在 ID 中仅出现一次”选项的情况下，该选项才可用。）针对响应的每个水平，显示或隐藏二水平分组变量的相对风险。请参见第 57 页的“条件关联和相对风险的示例”。

条件关联 （仅当已选定“在 ID 中仅出现一次”选项时可用。）显示或隐藏在给定某一响应水平的前提下的另一个响应水平的条件概率。请参见第 57 页的 “条件关联和相对风险的示例”。

一致性统计量 （仅适用于“评级员一致性”响应。）显示或隐藏一致性 Kappa 系数和 Bowker 对称性检验。请参见第 59 页的 “评级员一致性的示例”。

转换报表 （仅适用于“重复测量”响应。）针对响应随时间发生的变化显示或隐藏转换计数和比率矩阵。请参见第 60 页的 “重复测量的示例”。

检验选项 该菜单中可用的选项取决于选定的分析。

卡方检验选择 指定为单个响应计算的齐性卡方检验。您可以从“LR 和 Pearson”、“仅 LR”或“仅 Pearson”中进行选择。其中，LR 是指似然比。

显示警告 在卡方检验中显示小样本的警告。

按显著性排序 对报表重新排序，使得最显著的报表显示在最上部。

隐藏不显著 隐藏不显著的报表。

FDR 调整后的 p 值 显示或隐藏基于方法 Benjamini and Hochberg (1995) 的假发现率 *p* 值。

其他“分类”平台选项

以下选项支持您向报表添加汇总统计信息、保存报表和设置报表格式。

“汇总统计量”选项

总响应数 显示或隐藏交叉表和份额图中的响应的频数统计总和。指定分组变量时，合计涵盖每个分组类别。

响应水平 显示或隐藏交叉表和份额图中的响应列的类别。

显示超类别 （仅当定义一个或多个超类别时才可用。）显示或隐藏交叉表和份额图中的超类别对应的列。有关超类别的详细信息，请参见第 44 页的 “超类别”。

提示：该选项显示或隐藏超类别。要隐藏超类别中的单个类别，请使用“超类别”列属性中的“隐藏”选项。或者，使用“响应水平”选项隐藏所有响应水平，以便只有超类别保持可见。

总案例数 （仅适用于多重响应列。）显示或隐藏交叉表中包含每组中的案例（参与者）数的列。

响应的总案例数 （仅适用于多重响应列。）显示或隐藏交叉表中包含至少响应一次的案例（参与者）数的列。没有做出任何响应的参与者不包括在内。响应的总案例数小于等于总案例数。

得分均值 显示或隐藏交叉表和份额图中包含响应总均值或每个分组类别的均值的列。基于为每个响应类别分配的数值计算该均值。

- 对于数值类别，数值是实际值。
- 对于非数值类别，该值是通过“值得分”列属性分配给类别的值。

– 对于没有值得分的类别，默认从 1 到类别数为该值赋值。

请参见第 66 页的 “带比较字母的得分均值的示例”。

得分均值比较 在交叉表中显示或隐藏 “比较均值” 列。该列使用非合并的 Satterthwaite *t* 检验进行配对比较，来比较各个分组类别的得分均值。请参见 SAS Institute Inc. (2020) 中的 “TTEST 过程” 一章。使用字母显示比较结果。有关比较字母的详细信息，请参见第 43 页的 “比较字母”。有关指定比较组的详细信息，请参见第 55 页的 “使用比较字母的用户指定比较的示例”。

得分标准差 显示或隐藏交叉表和份额图中包含响应总标准差或每个分组类别的标准差的列。

按得分均值排序 （仅当指定了多个响应且分析中没有分组变量时才显示。）按得分均值对响应报表排序。

保存选项

保存表 包含将报表特定部分保存到新数据表中的选项。每个选项都为每个报表创建单独的数据表。该菜单中可用的选项取决于选定的分析。保存的表都包含一个 “源” 脚本。

注意：超类别不包含在新表中。

保存频数 将交叉表中的频数统计保存到新数据表中。

保存响应的份额 将交叉表中的响应份额保存到新数据表中。

保存列联表 将完整的交叉表保存到新数据表中。

保存每种情况的比率 将交叉表中的每种情况的比率保存到新数据表中。

保存转置频数 将交叉表中的转置频数统计保存到新数据表中。

保存转置的响应份额 将交叉表中的转置响应份额保存到新数据表中。

保存转置的每种情况的比率 将交叉表中转置的每种情况的比率保存到新数据表中。

保存检验比率 将 “检验多重响应” 选项的结果保存到新数据表中。

保存齐性检验结果 将 “检验响应齐性” 选项的结果保存到新数据表中。

保存得分均值 将每个样本组的得分均值保存到新数据表中。

保存 t 检验和 p 值 将 “得分均值比较” 报表中的 *t* 检验和 *p* 值保存到新数据表中。

保存 Excel 文件 （仅在 Windows 上可用。）创建含有交叉表结构的 Microsoft Excel 电子表格。该选项将所有表映射到一个工作表，其中行代表响应类别，列代表样本水平，多个表的样本水平共享标题。若每个表单元格中有多个元素，您可以在 Microsoft Excel 中将其拆分为多个单元格或合并为一个单元格。

报表格式选项

过滤 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

内容汇总 在 “分类” 报表的顶部显示或隐藏 “内容汇总” 报表。“内容汇总” 报表包含链接到关联报表的汇总中的所有检验和**得分均值**。

显示报表中使用的列 （仅可用于 SPSS 或 SAS 名称。）显示或隐藏“报表中使用的列”信息。该选项仅影响具有以下列属性的列：SPSS 名称、 SAS 名称、 SPSS 标签或 SAS 标签。

提示：当您从 SAS 或 SPSS 导入调查数据时，“名称”和“标签”列属性将自动添加至您的 JMP 表。您可以使用 “其他”列属性添加 “SAS 名称”、“SPSS 名称”、“SAS 标签”或“SPSS 标签”列属性。例如，若您使用“SAS 名称”或“SPSS 名称”列属性存储某个调查问题，该列名可以为短名称。

设置元素格式 支持您指定频数、份额、比率和均值的格式。

行内排列 支持您在窗口内排列多个报表。输入您想要在窗口中横向查看的报表数。

设置首选项 支持您在当前 JMP 会话以及将来的 JMP 会话中为将来启动 “分类”平台设置首选项。请参见第 71 页的 “设置首选项”。

类别选项 包含在启动窗口中也提供的一些选项 （分组选项、将缺失响应计算在内、将响应水平从高到低排序、缩短标签以及包括不在数据中的响应）。若在此选定这些选项，该平台使用新设置更新。有关类别选项的详细信息，请参见第 36 页的 “其他启动窗口选项”。

强制交叉表着色 强制在交叉表报表上着色，即使首选项设置为不着色。若未选定该选项，则根据 “表隔行着色”首选项的当前设置为交叉表报表着色。

重新启动对话框 支持您返回启动窗口并对所指定的分析进行编辑。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

交叉表选项

显示字母 在交叉表中显示或隐藏列字母 ID。这些字母用在许多齐性检验中，当执行这些检验时，字母会自动显示。

指定比较组 支持您为齐性检验指定特定比较组。使用用斜线分隔的组比较字母表示每个组。用逗号分隔多个组。例如，要检验 A 与 E， B 与 D， C 与 F，应按照“A/E, B/D, C/F”来指定各组。为定义的比较组提供 “比较每个单元格”报表。请参见第 55 页的 “使用比较字母的用户指定比较的示例”。

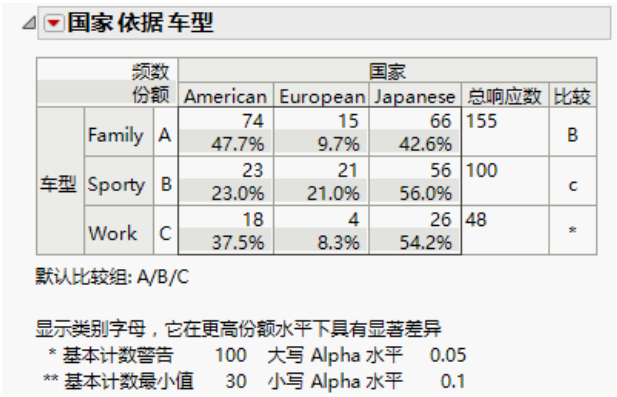
删除 从报表窗口中删除报表。

警告：无法撤销 “删除”选项。

比较字母

“比较每个单元格”、“比较每个样本”和 “得分均值比较”选项都使用比较字母来标识样本水平。若超过 26 个水平，则会在字母之后追加数字。若启用某个比较选项，这些字母将显示在交叉表的样本水平标题右侧。这些字母用于标识 “比较”列中的水平。

图 3.17 带比较字母的交叉表



国家依据 车型

频数 份额			国家 American	国家 European	国家 Japanese	总响应数	比较
车型	Family	A	74 47.7%	15 9.7%	66 42.6%	155	B
	Sporty	B	23 23.0%	21 21.0%	56 56.0%	100	c
	Work	C	18 37.5%	4 8.3%	26 54.2%	48	*

默认比较组: A/B/C

显示类别字母，它在更高份额水平下具有显著差异
* 基本计数警告 100 大写 Alpha 水平 0.05
** 基本计数最小值 30 小写 Alpha 水平 0.1

“比较”列中的字母指示两个水平之间的差异。包含字母的行是一个水平，“比较”列中的字母指示第二个水平。若两个样本水平存在显著差异，响应份额较小的样本水平的字母会被放进另一个水平的比较单元格中。与小写字母相比，大写字母指示水平间存在更大的差异。大写字母的默认 alpha 水平（显著性水平）为 0.05，小写字母的默认 alpha 水平为 0.10。例如，在图 3.17 中，请注意以下事项：

- Family 轿车表的第一行在“比较”列中有一个 B。字母 B 与 Sporty 轿车有关。这指示在 0.05 这个显著性水平下，Sporty（运动型）和 Family（家用型）轿车的原产国有差异。B 位于 Family 轿车这一行中，这是因为 Sporty 轿车的总响应数 (100) 小于 Family 轿车的总响应数 (155)。
- Sporty 行的“比较”列中的 c 指示在比较 Sporty 和 Work（工作用车）轿车时，在 0.10 水平下两种车型的原产国之间有显著差异。c 位于 Sporty 轿车行中，因为 Sporty 轿车的总响应数 (100) 大于 Work 轿车的总响应数 (48)。

小计数警告在比较单元格中用星号指示。一个星号指示该水平的响应数小于 100。两个星号指示水平的响应数小于 30。在图 3.17 中，请注意 Work 的行在“比较”列中有一个星号。由于 Work 轿车的总响应数 (48) 小于 100 而出现响应计数小的警告。

您可以在“分类”平台首选项中更改 alpha 水平和警告计数阈值。有关更改首选项的详细信息，请参见第 71 页的“设置首选项”。

提示：若您只想在报表中包含一组比较字母，请在首选项中将**小写 Alpha 水平**设置为 0。

以下示例演示比较字母的用法：

- 第 52 页的“使用比较字母“比较每个样本”的示例”
- 第 53 页的“使用比较字母“比较每个单元格”的示例”
- 第 66 页的“带比较字母的得分均值的示例”

超类别

超类别一词是指响应类别的聚合。例如，使用五分评级制时，您可能想要了解前两个评级的响应百分比（前两个框）。使用“超类别”列属性定义响应组。

将“超类别”列属性添加到响应列时，不会向数据表中添加其他列。“超类别”列属性会改为将一个额外的类别列添加到“分类”平台报表中的交叉表和频数图。可以为单个响应列创建多个超类别。份额图不显示超类别，超类别不应用于分组列。

要创建超类别，请执行以下操作：

1. 在数据表中选择您要聚合的类别所在的列。
2. 选择**列 > 列信息**。
3. 点击**列属性**，然后选择**超类别**。
4. （可选）要更改超类别的默认名称，请输入 “超类别名称”。
5. 从 “列的类别”列表中选择一个或多个类别。
6. 点击**添加**。
7. （可选）选择所需超类别，然后点击 “超类别”红色小三角菜单查看更多选项。

超类别选项

“列属性”窗口的 “超类别”红色小三角菜单中提供以下选项：

隐藏 隐藏交叉表和频数图中超类别中的类别。

提示：若想要灵活地显示或隐藏报表中的单个类别，则不要使用 “隐藏”选项；改为使用“分类”红色小三角菜单中的 “响应水平”选项。

净值 （仅适用于多重响应列。）当各个响应者出现在多个超类别中时，防止对其重复计数。

添加均值 在报表中包含均值统计量。

添加标准差 在报表中包含标准差统计量。

全部添加 在报表中包含总响应数。默认情况下，始终包含 “总响应数”列。

注意：除了“重复测量”和“评级员一致性”，所有响应效应都支持超类别。

“分类”平台的更多示例

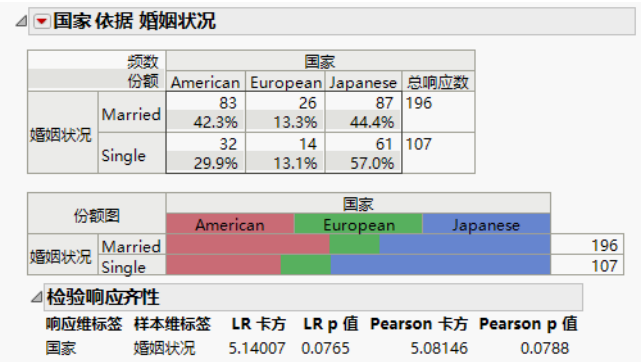
- “响应齐性检验的示例”
- “多重响应检验的示例”
- “超类别示例”
- “单元格卡方检验的示例”
- “使用比较字母“比较每个样本”的示例”
- “使用比较字母“比较每个单元格”的示例”
- “使用比较字母的用户指定比较的示例”
- “对齐的响应的示例”
- “条件关联和相对风险的示例”
- “评级员一致性的示例”
- “重复测量的示例”
- “多重响应的示例”
- “带比较字母的得分均值的示例”
- “结构化报表的示例”
- “无响应的多重响应示例”

响应齐性检验的示例

本例使用 Car Poll.jmp 样本数据表，其中包含从轿车所有权的调查中收集的数据。数据中包括有关调查的个人的人口统计信息以及他们轿车的相关信息。您想要了解婚姻状况与轿车原产地之间的关系。还想要检验响应齐性。也就是说，您想要通过检验来判定轿车原产地的分布针对已婚和单身响应者来说分布是否相同。

1. 选择**帮助** > **样本数据库**，然后打开 Car Poll.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择**国家**并点击 “简单”选项卡上的**响应**。
4. 选择**婚姻状况**并点击 **X，分组类别**。
5. 点击**确定**。
6. 点击 “分类”红色小三角并选择**检验响应齐性**。

图 3.18 检验响应齐性



国家 依据 婚姻状况

频数 / 份额		国家 American	European	Japanese	总响应数
婚姻状况	Married	83 42.3%	26 13.3%	87 44.4%	196
	Single	32 29.9%	14 13.1%	61 57.0%	107

检验响应齐性

响应维标签	样本维标签	LR 卡方	LR p 值	Pearson 卡方	Pearson p 值
国家	婚姻状况	5.14007	0.0765	5.08146	0.0788

份额图指示已婚组所拥有的轿车在美国车和日本车之间均匀拆分。在单身组中，日本车是最常拥有的车型。

响应齐性检验提供了两个版本的检验结果。Pearson 检验和似然比检验都有卡方检验统计量和相关的 p 值。这两种方法的响应齐性检验的 p 值约为 0.08。

多重响应检验的示例

本例使用 Consumer Preferences.jmp 样本数据表。该表包含关于人们的态度和观点的调查中的数据，以及有关口腔卫生的问题。您可以使用 “检验多重响应”选项检验每个刷牙时间 （**刷牙分隔**）的响应率在各组（**刷牙**）之间是否相同。其中，各组是按照响应者的刷牙频率来定义的。

1. 选择**帮助 > 样本数据库**，然后打开 Consumer Preferences.jmp。
2. 滚动到右侧，直到您看到**刷牙分隔**列。

图 3.19 Consumer Preferences 数据表

	起床后刷牙	饭后刷牙	睡前刷牙	其他时间刷牙	刷牙其他	刷牙分隔
1	1	0	0	0		Wake,
2	0	1	0	0		After Meal,
3	1	0	1	0		Wake,Before Sleep,
4	0	1	0	0		After Meal,
5	1	0	1	0		Wake,Before Sleep,
6	1	1	1	0		Wake,After Meal,Before Sleep,
7	1	1	1	0		Wake,After Meal,Before Sleep,

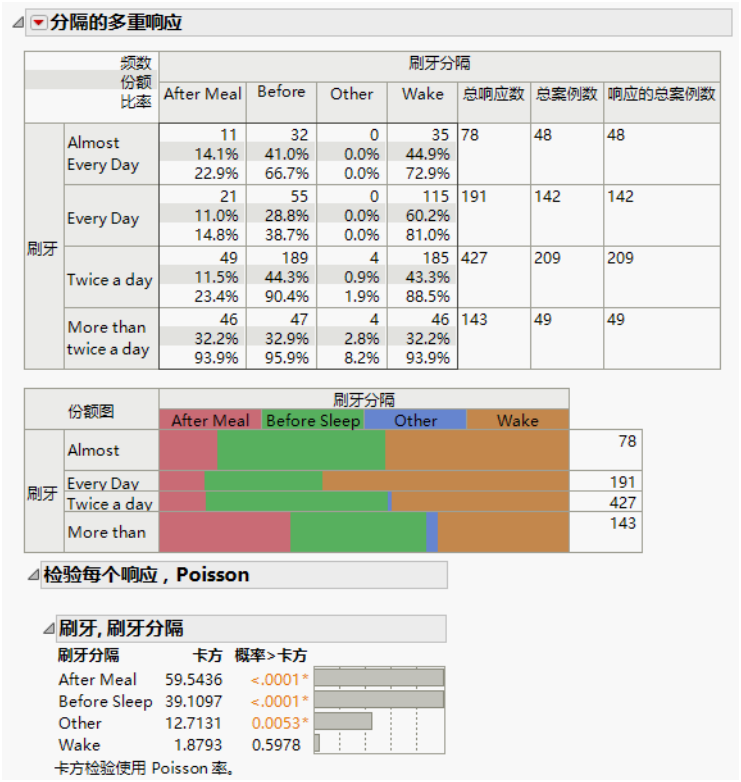
请注意，**刷牙分隔**列包含多重响应问题的响应。每个响应由逗号分隔。**刷牙分隔**之前的四列包含以四种数据格式显示的相同信息。每列（**起床后刷牙**、**饭后刷牙**、**睡前刷牙**与**其他时间刷牙**）都包含一个响应。若选择了该响应，则列值为 1，否则为 0。

3. 选择**分析 > 消费者研究 > 分类**。
4. 选择**多重**选项卡。
5. 选择**刷牙分隔**并点击 “多重”选项卡上的**分隔的多重响应**。

提示：或者，您可以选择**起床后刷牙、饭后刷牙、睡前刷牙与其他时间刷牙**，然后点击“多重”选项卡上的**指示符组**。

6. 选择**刷牙**并点击 **X，分组类别**。
7. 点击**确定**。
8. 点击“分类”红色小三角，然后选择**检验多重响应 > 计数检验 , Poisson**。

图 3.20 检验多重响应， Poisson



p 值显示 After Meal、Before Sleep 和 Other 的响应率在各个刷牙组之间显著不同。Wake 的响应率在各个刷牙组之间没有显著不同。“概率 > 卡方”列右侧的条形图绘制以 -Log10(p) 为尺度的 *p* 值。从交叉表中，您可以看到多数人都在起床后刷牙，不论他们刷几次。

9. 点击“分类”红色小三角并选择**检验多重响应 > 齐性检验，二项**。

图 3.21 检验多重响应，二项

“齐性检验，二项”选项始终生成比“计数检验， Poisson”选项更大的检验统计量（因此生成的 p 值较小）。二项分布不仅比较发生响应的比率（报告在睡醒后刷牙的人数），而且还比较未发生响应的比率（未报告在睡醒后刷牙的人数）。

在本例中，每个响应（After Meal、Before Sleep、Wake 和 Other）的响应者比例在各个组之间都不同。每个响应的 p 值都小于 0.05。

提示：JMP 可通过“多重响应”建模类型或“多重响应”列属性来检测出多重响应列。

超类别示例

本例使用 Color Preference Survey.jmp 样本数据表，其中包含关于人们的颜色首选项的调查数据。本例演示超类别的用法。

1. 选择**帮助 > 样本数据库**，然后打开 Color Preference Survey.jmp。
2. 选择列**我喜欢橙色**。右击列标题并选择**列属性 > 超类别**。
3. 在“超类别”下，为列类别选择 Agree 和 Strongly agree。
4. 在“超类别名称”下，输入 Top 2 Box。
5. 点击**添加**。

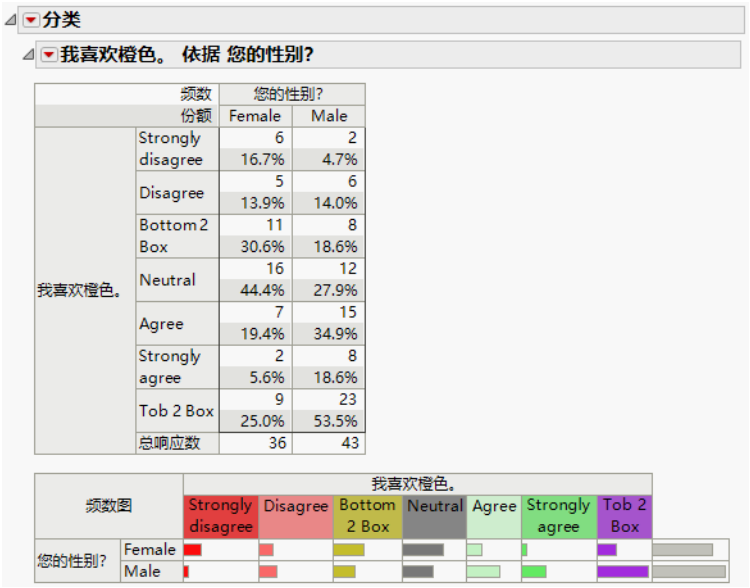
   提示：点击名称左侧的小三角以显示包含在超类别中的类别。

6. 在“超类别”下，为列类别选择 Disagree 和 Strongly disagree。
7. 在“超类别名称”下，输入 Bottom 2 Box。
8. 点击**添加**。

图 3.22 超类别列属性

9. 点击**确定**。
10. 选择**分析 > 消费者研究 > 分类**。
11. 选择**结构化**选项卡。
12. 点击侧面绿色小三角并选择**我喜欢橙色**。
13. 点击顶部绿色小三角并选择**您的性别？**
14. 点击**添加 =>**，然后点击**确定**。
15. 点击“分类”红色小三角并选择**转置的频数图**。

图 3.23 带有超类别的结构化分类报表


分类

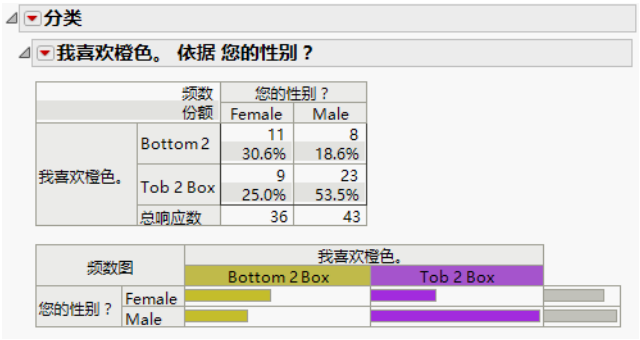
我喜欢橙色。 依据 您的性别?

频数 / 份额		您的性别? Female	您的性别? Male
我喜欢橙色。	Strongly disagree	6 16.7%	2 4.7%
	Disagree	5 13.9%	6 14.0%
	Bottom 2 Box	11 30.6%	8 18.6%
	Neutral	16 44.4%	12 27.9%
	Agree	7 19.4%	15 34.9%
	Strongly agree	2 5.6%	8 18.6%
	Tob 2 Box	9 25.0%	23 53.5%
	总响应数	36	43

交叉表包含两个额外行，每个超类别一行。这些超类别还包含在频数图中。请注意，Top 2 Box 行中的频数统计是“Agree”和“Strongly agree”类别中计数的总和。Bottom 2 Box 行中的频数统计是“Disagree”和“Strongly disagree”类别中计数的总和。

16. 点击 “分类”红色小三角并取消选择**响应水平**。

图 3.24 具有超类别且无响应水平的结构化分类报表


分类

我喜欢橙色。 依据 您的性别？

频数 / 份额		您的性别？ Female	您的性别？ Male
我喜欢橙色。	Bottom 2	11 30.6%	8 18.6%
	Tob 2 Box	9 25.0%	23 53.5%
	总响应数	36	43

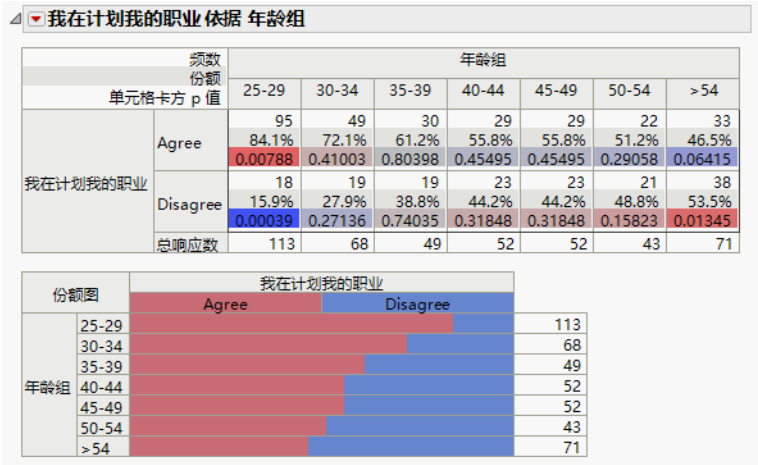
通过删除响应水平，您的输出现在只包含超类别。请注意总计针对所有水平。在本例中，中性响应不包含在超类别中。

单元格卡方检验的示例

本例使用 Consumer Preferences.jmp 样本数据表。该表包含关于人们的态度和观点的调查数据，以及有关口腔卫生的问题。您要探索各年龄组之间对于 “我在计划我的职业”这一陈述的响应的分布。

1. 选择**帮助** > **样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择**我在计划我的职业**并点击 “简单”选项卡上的**响应**。
4. 选择**年龄组**并点击 **X，分组类别**。
5. 点击**确定**。
6. 点击 “分类”红色小三角并选择**交叉表转置**。
7. 点击 “分类”红色小三角并选择**单元格卡方**。

图 3.25 单元格卡方



我在计划我的职业 依据 年龄组

频数 份额 单元格卡方 p 值		年龄组						
		25-29	30-34	35-39	40-44	45-49	50-54	>54
我在计划我的职业	Agree	95 84.1% 0.00788	49 72.1% 0.41003	30 61.2% 0.80398	29 55.8% 0.45495	29 55.8% 0.45495	22 51.2% 0.29058	33 46.5% 0.06415
	Disagree	18 15.9% 0.00039	19 27.9% 0.27136	19 38.8% 0.74035	23 44.2% 0.31848	23 44.2% 0.31848	21 48.8% 0.15823	38 53.5% 0.01345
	总响应数	113	68	49	52	52	43	71

较小的 p 值指示观测的单元格计数与期望的单元格计数之间存在显著差异。p 值按照显著性水平来着色，计数显著高于期望值的单元格着深红色，计数显著低于期望值的单元格着深蓝色。期望的单元格计数基于观测的行与列合计。预期的单元格计数计算为行总数乘以列总数再除以总计数的结果。

例如，在 25 - 29 年龄组中选择同意的比率相等的原假设下，预期响应数为 (287*113)/448 = 72.4；观测值为 95。这个观测值（p 值为 0.00788）显著大于期望值。25 - 29 年龄组中同意“我在计划我的职业”的响应数高于期望值（若对该问题的响应与年龄无关）。

使用比较字母“比较每个样本”的示例

本例使用 Consumer Preferences.jmp 样本数据表。该表包含关于人们的态度和观点的调查数据，以及有关口腔卫生的问题。您要探索各年龄组之间对于 “我在计划我的职业”这一陈述的响应的分布。

1. 选择**帮助 > 样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**分析 > 消费者研究 > 分类**。
3. 选择**我在计划我的职业**并点击 “简单”选项卡上的**响应**。
4. 选择**年龄组**并点击 **X，分组类别**。
5. 点击**确定**。
6. 点击 “分类”红色小三角并选择**比较每个样本**。

图 3.26 比较每个样本

我在计划我的职业 依据 年龄组

			频数 / 份额	我在计划我的职业		
			Agree	Disagree	总响应数	比较
年龄组	25-29	A	95 84.1%	18 15.9%	113	b,C,D,E,F,G
	30-34	B	49 72.1%	19 27.9%	68	d,e,F*
	35-39	C	30 61.2%	19 38.8%	49	*
	40-44	D	29 55.8%	23 44.2%	52	*
	45-49	E	29 55.8%	23 44.2%	52	*
	50-54	F	22 51.2%	21 48.8%	43	*
	>54	G	33 46.5%	38 53.5%	71	B*

默认比较组: A/B/C/D/E/F/G

显示类别字母，它在更高份额水平下具有显著差异
* 基本计数警告 100 大写 Alpha 水平 0.05
** 基本计数最小值 30 小写 Alpha 水平 0.1

比较每个样本

字母比较使用 Pearson 卡方 LR p 值
年龄组, 我在计划我的职业 每对的 LR 卡方 p 值

	A	B	C	D	E	F	G
A	1.0000	0.0552	0.0020	0.0001	0.0001	<.0001	<.0001
B	0.0552	1.0000	0.2183	0.0641	0.0641	0.0261	0.0020
C	0.0020	0.2183	1.0000	0.5781	0.5781	0.3312	0.1108
D	0.0001	0.0641	0.5781	1.0000	1.0000	0.6540	0.3083
E	0.0001	0.0641	0.5781	1.0000	1.0000	0.6540	0.3083
F	<.0001	0.0261	0.3312	0.6540	0.6540	1.0000	0.6276
G	<.0001	0.0020	0.1108	0.3083	0.3083	0.6276	1.0000

Pearson p 值 每对的 Pearson 卡方 p 值

	A	B	C	D	E	F	G
A	1.0000	0.0523	0.0015	<.0001	<.0001	<.0001	<.0001
B	0.0523	1.0000	0.2170	0.0638	0.0638	0.0255	0.0022
C	0.0015	0.2170	1.0000	0.5783	0.5783	0.3314	0.1119
D	<.0001	0.0638	0.5783	1.0000	1.0000	0.6540	0.3087
E	<.0001	0.0638	0.5783	1.0000	1.0000	0.6540	0.3087
F	<.0001	0.0255	0.3314	0.6540	0.6540	1.0000	0.6276
G	<.0001	0.0022	0.1119	0.3087	0.3087	0.6276	1.0000

交叉表汇总了各年龄组之间关于“我在计划我的职业”的这一陈述。该表的单元格包含各个年龄组中同意或不同意这一陈述的响应者频数（计数）和份额（百分比）。此外，交叉表还包括比较字母。每个组在“比较”列中都标有一个字母，该字母位于组标签右侧。“比较”列中的字母支持您解释组间独立性统计测试的结果。

“比较每个样本”报表提供来自配对 Pearson 和卡方似然比卡方检验的 *p* 值。该 *p* 值在比较字母标记的对称矩阵中报告。

对于本例，我们进行了以下观测：

- 25 - 29 年龄组的比较列包含 b 到 g 之间的所有字母。因此，与其他所有组相比，25 - 29 年龄组对于“我在计划我的职业”这一陈述的响应率有显著不同。由于字母 b 小写，25 - 29 年龄组与 30 - 34 年龄组之间的差异在 0.10 水平下显著。所有其他字母都是大写字母，表明在 0.05 水平下差异显著。
- >54 的年龄组（用字母 G 表示）与 30 - 34 年龄组（用 B 表示）显著不同。比较字母位于组 G 的单元格中，因为组 G 的响应者数目比组 B 高（71 对 68）。
- 比较单元格中的单个星号是小样本警告。单个星号表明组中有 30 个以上 100 个以下的响应。
- 两个星号（本例未观测到）则表明组大小小于 30。

使用比较字母“比较每个单元格”的示例

本例使用 Consumer Preferences.jmp 样本数据表。该表包含关于人们的态度和观点的调查数据，以及有关口腔卫生的问题。您要探索员工任期组之间关于工作满意度的响应的分布。

1. 选择**帮助** > **样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择**工作满意度**，并在 “简单”选项卡上点击**响应**。
4. 选择**员工任期**并点击 **X，分组类别**。
5. 点击**确定**。
6. 点击 “分类”红色小三角并选择**比较每个单元格**。
7. 点击**比较每个单元格 - 详细信息**灰色展开图。

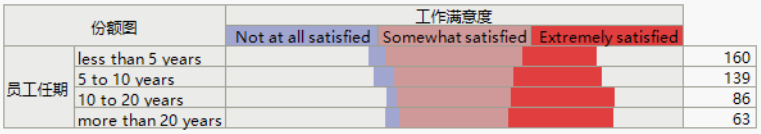
图 3.27 比较每个单元格

工作满意度 依据 员工任期

频数 份额 比较			工作满意度 Not at all	 Somewhat	 Extremely	 总响应数
员工任期	less than 5 years	A	11 6.9%	97 60.6% d	52 32.5%	160
	5 to 10 years	B	13 9.4%	72 51.8%	54 38.8%	139
	10 to 20 years	C	4 4.7% *	43 50.0% *	39 45.3% a*	86
	more than 20 years	D	4 6.3% *	30 47.6% *	29 46.0% a*	63

默认比较组: A/B/C/D

显示类别字母，它在更高份额水平下具有显著差异
* 基本计数警告 100 大写 Alpha 水平 0.05
** 基本计数最小值 30 小写 Alpha 水平 0.1



比较每个单元格 - 详细信息

字母比较使用 Fisher 精确检验
员工任期, 工作满意度

LR 配对

		AB	BB	AC	BC	CC	AD	BD	CD	DD
Not at all satisfied	1.0000	0.4323	1.0000	0.4781	0.1806	1.0000	0.8872	0.4658	0.6515	1.0000
Somewhat satisfied	1.0000	0.1246	1.0000	0.1092	0.7931	1.0000	0.0783	0.5820	0.7739	1.0000
Extremely satisfied	1.0000	0.2526	1.0000	0.0476	0.3366	1.0000	0.0607	0.3377	0.9341	1.0000

Pearson 配对

		AB	BB	AC	BC	CC	AD	BD	CD	DD
Not at all satisfied	1.0000	0.4316	1.0000	0.4870	0.1948	1.0000	0.8878	0.4763	0.6496	1.0000
Somewhat satisfied	1.0000	0.1246	1.0000	0.1086	0.7931	1.0000	0.0774	0.5820	0.7740	1.0000
Extremely satisfied	1.0000	0.2524	1.0000	0.0465	0.3360	1.0000	0.0585	0.3364	0.9341	1.0000

Fisher 精确配对

		AB	BB	AC	BC	CC	AD	BD	CD	DD
Not at all satisfied	1.0000	0.5235	1.0000	0.5853	0.2990	1.0000	1.0000	0.5912	0.7221	1.0000
Somewhat satisfied	1.0000	0.1302	1.0000	0.1373	0.8909	1.0000	0.0983	0.6493	0.8685	1.0000
Extremely satisfied	1.0000	0.2764	1.0000	0.0531	0.4034	1.0000	0.0649	0.3574	1.0000	1.0000

表中提供配对似然比卡方、Pearson 卡方和 Fisher 独立性精确检验的 *p* 值。这些表用比较字母来标记。比较字母显示在组标签右侧的交叉表中。各组之间不同的响应率在交叉表单元格中使用比较字母来指示。

与任期超过 20 年的员工相比，任期不到 5 年的员工在“比较满意”上的响应率要稍微高一些。这一发现用交叉表第一行中“Somewhat satisfied”单元格中的字母 d 来表示。此外，任期不到 5 年的员工在 Extremely satisfied（极其满意）上的比率要低于任期 20 年的员工组。此结果

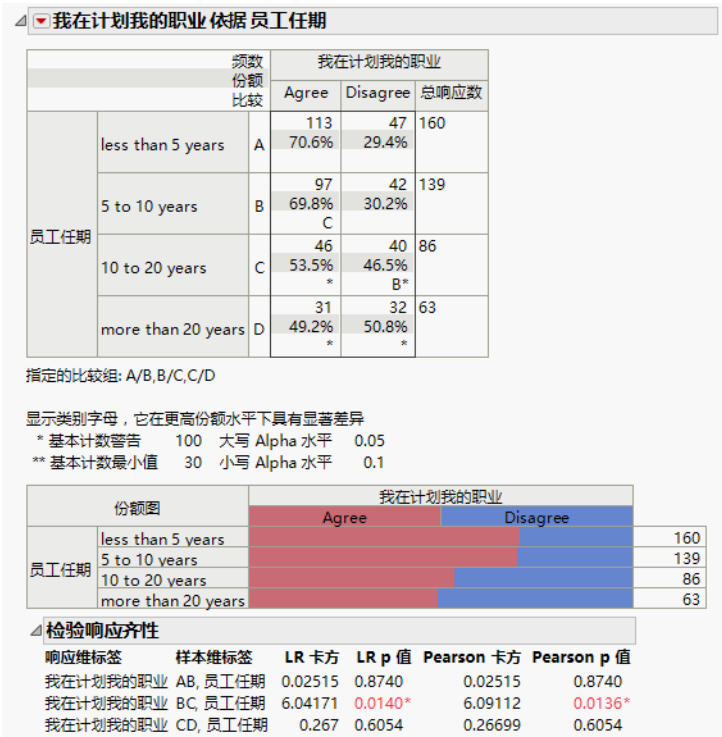
由交叉表最后一行的 Extremely satisfied 单元格中的字母 a 表明。这些字母位于响应份额最高的单元格中。

使用比较字母的用户指定比较的示例

本例使用 Consumer Preferences.jmp 样本数据表。该表包含关于人们的态度和观点的调查数据。您需要定义特定的比较组，在这些组之间比较对于“我在计划我的职业”这一陈述的响应。

1. 选择**帮助 > 样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**分析 > 消费者研究 > 分类**。
3. 选择**我在计划我的职业**并点击 “简单”选项卡上的**响应**。
4. 选择**员工任期**并点击 **X，分组类别**。
5. 点击**确定**。
6. 点击 “我在计划我的职业 依据 员工任期”旁边的红色小三角，然后选择**显示字母**。
7. 点击 “我在计划我的职业 依据 员工任期”旁边的红色小三角，然后选择**指定比较组**。
8. 输入 A/B、 B/C、 C/D 并点击**确定**。
9. 点击 “分类”红色小三角并选择**检验响应齐性**。

图 3.28 指定比较示例



我在计划我的职业 依据 员工任期

		频数 份额 比较	我在计划我的职业 Agree	 Disagree	 总响应数
员工任期	less than 5 years	A	113 70.6%	47 29.4%	160
	5 to 10 years	B	97 69.8% C	42 30.2%	139
	10 to 20 years	C	46 53.5% *	40 46.5% B*	86
	more than 20 years	D	31 49.2% *	32 50.8% *	63

指定的比较组: A/B,B/C,C/D

显示类别字母，它在更高份额水平下具有显著差异
* 基本计数警告 100 大写 Alpha 水平 0.05
** 基本计数最小值 30 小写 Alpha 水平 0.1

检验响应齐性

响应维标签	样本维标签	LR 卡方	LR p 值	Pearson 卡方	Pearson p 值
我在计划我的职业	AB, 员工任期	0.02515	0.8740	0.02515	0.8740
我在计划我的职业	BC, 员工任期	6.04171	0.0140*	6.09112	0.0136*
我在计划我的职业	CD, 员工任期	0.267	0.6054	0.26699	0.6054

响应齐性检验比较组 A 与组 B、组 B 与组 C、组 C 与组 D。组 B（5 到 10 年）响应者同意“我在计划我的职业”的这一陈述的次数比组 C（10 到 20 年）多。该一致性比率的差异的 Pearson *p* 值为 0.0136。

对齐的响应的示例

本例使用 Consumer Preferences.jmp 样本数据表。该表包含关于态度和观点的调查数据。

1. 选择**帮助 > 样本数据库**，然后打开 Consumer Preferences.jmp。
2. 滚动以查看列**我在计划我的职业**。

图 3.29 Consumer Preferences 数据表 （部分表）

	我在计划我的职业	我想环游世界	我的家需要重大改善	我在工作之外获得巨大收益	我想让我的债务获得控制	我来自大家庭
1	Agree	Disagree	Disagree	Agree	Disagree	Agree
2	Agree	Agree	Agree	Agree	Disagree	Agree
3	Disagree	Agree	Agree	Agree	Agree	Agree
4	Agree	Agree	Agree	Agree	Agree	Agree
5	Disagree	Agree	Disagree	Agree	Agree	Disagree

请注意，一共有六列，所有列的响应都相同：同意和不同意。这六列的响应已对齐。对于本例，我们分析列**我在计划我的职业**和**我想环游世界**。首先，使用标准化属性设置这两列的值颜色、值顺序和建模类型：

3. 选择列标题**我在计划我的职业**和**我想环游世界**。右击并选择**标准化特性**。
4. 选择**列属性 > 值颜色**。右击“Agree”颜色椭圆并将其设置为绿色，将“Disagree”设置为红色。
5. 选择**列属性 > 值顺序**。点击**反转**。
6. 选择**特性 > 建模类型**和**建模类型 > 有序型**。
7. 点击**确定**。

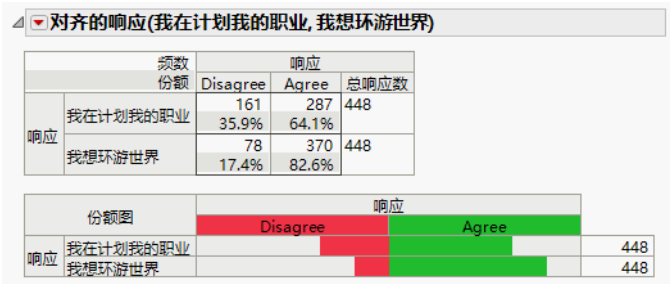
接下来，使用“分类”平台来分析这两列。

8. 选择**分析 > 消费者研究 > 分类**。
9. 选择**我在计划我的职业**和**我想环游世界**。
10. 选择**相关**选项卡，然后点击“相关”选项卡上的**对齐的响应**。

提示：若要使用“结构化”选项卡执行相同操作，请将两列拖入“多重对齐”拖放区（绿色箭头）。

11. 点击**确定**。

图 3.30 对齐的响应报表



请注意，份额图是一个方向条形图。您看到的条形图类型取决于列的建模类型：

- 有序型列会生成方向条形图。本例中的两列均为有序型。
- 名义型列会生成堆叠条形图。

条件关联和相对风险的示例

本例使用 AdverseR.jmp 样本数据表，其中包含来自临床试验的不良反应。使用该数据可探索不良事件的条件关联，之后探索与控制组比照后的治疗组中事件的相对风险。

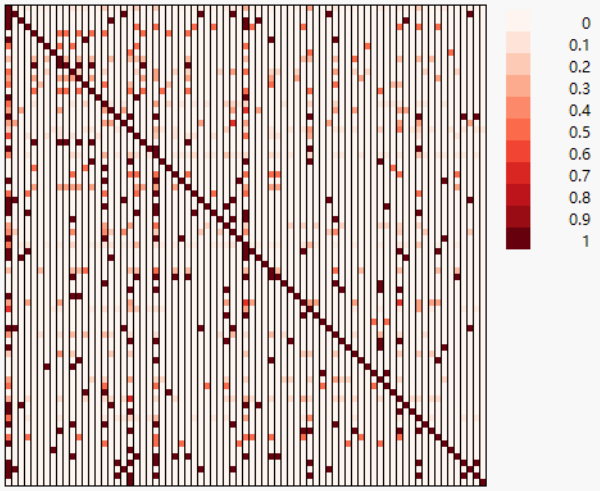
1. 选择**帮助 > 样本数据库**，然后打开 AdverseR.jmp。
2. 选择**分析 > 消费者研究 > 分类**。
3. 选择 “多重”选项卡。
4. 选择**不良反应**并点击 “多重”选项卡上的**由 ID 标识的多重响应**。
5. 选择**治疗组**并点击 **X，分组类别**。
6. 选择**病人编号**，然后点击 ID。
7. 在其他启动窗口选项之下，选择**在 ID 中仅出现一次**并点击**确定**。
8. 点击 “分类”红色小三角并选择**条件关联**。

图 3.31 “条件关联”报表 （部分报表）

由 ID 标识的多重响应(不良反应) 依据 治疗组

条件关联: 不良反应

给定每行中的响应，显示使该响应也在列中显示的比率。



条件关联

	ABDOMINAL PAIN	ABNORMAL VISION	ALOPECIA	AMBLYOPIA
ABDOMINAL PAIN	1.0000	0.0323	0.0323	0.0645
ABNORMAL VISION	1.0000	1.0000	0.0000	0.0000
ALOPECIA	1.0000	0.0000	1.0000	0.0000
AMBLYOPIA	1.0000	0.0000	0.0000	1.0000
ANEMIA	0.0000	0.0000	0.0000	0.0000
ANGINA PECTORIS	1.0000	0.0000	0.0000	0.0000
ANOREXIA	0.5000	0.0000	0.0000	0.0000
ANXIETY	0.0000	0.0000	0.0000	0.0000
ASTHENIA	0.2500	0.0000	0.0000	0.0000
ASTHMA	0.0000	0.0000	0.0000	0.0000
BACK PAIN	0.5000	0.0000	0.0000	0.0000
BRONCHITIS	0.3333	0.0000	0.0000	0.0000
CHEST PAIN	0.5000	0.0000	0.0000	0.0000

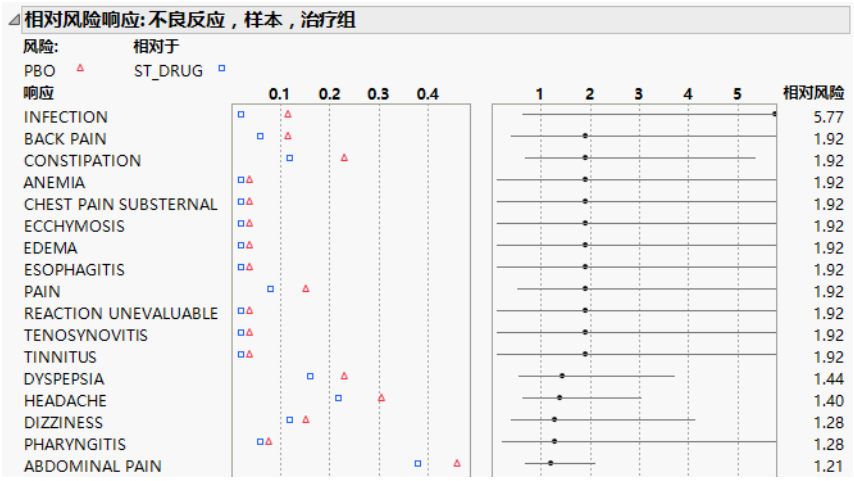
条件关联矩阵提供在有某一不良反应的前提下，存在另一不良反应的条件概率。概率涉及所有组。在患者有异常疼痛的前提下，视觉异常的概率为 0.0323。

提示：悬停在条件概率的热图上方。

9. 点击 “分类”红色小三角并选择**相对风险**。
10. 在窗口中选择 PBO 并点击**确定**。

    右击窗口中的 “相对风险”报表并选择**按列排序**。
11. 选择**相对风险**并点击**确定**。

图 3.32 “相关风险”报表 （部分报表）



“相对风险”选项通过计算分组变量各个水平的风险比率来得到不同响应的相对风险。默认“相对风险”报表列出响应名称、分组变量每个水平的风险（比率）、具有 95% 置信区间的相对风险图，以及相对风险估计值。在此您可以按治疗组比较不良反应的相对风险。PBO 组感染的相对风险是 ST_DRUG 组的 5.7 倍。不过，置信区间非常宽，并且包括相对风险 1.0。当分组变量每个水平的风险相同时就会出现相对风险 1.0。

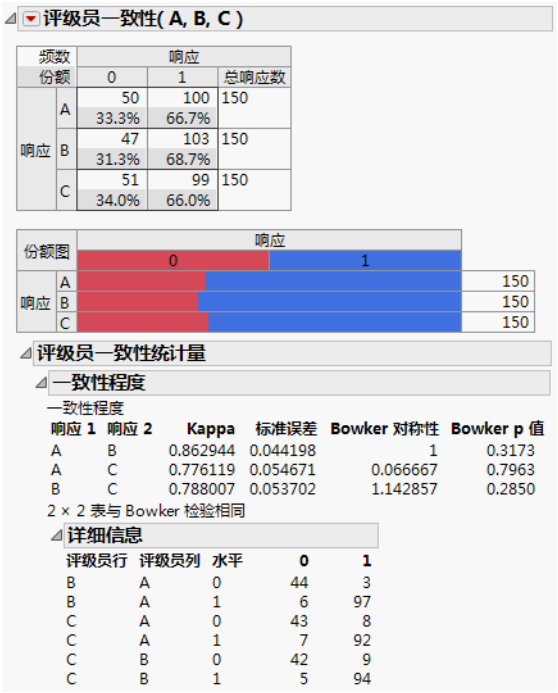
右击并选择**列 > 95% 下限**和**列 > 95% 上限**，将相对风险估计值的 95% 置信区间添加至报表表。

评级员一致性的示例

本例使用 Attribute Gauge.jmp 样本数据，其中包含三个操作员对 50 个部件评级三次的评级数据 (0/1)。

1. 选择**帮助 > 样本数据库**，然后打开 Attribute Gauge.jmp。
2. 选择**分析 > 消费者研究 > 分类**。
3. 选择 “相关”选项卡。
4. 选择 A、 B 和 C，然后点击 “相关”选项卡上的**评级员一致性**。
5. 点击**确定**。

图 3.33 “评级员一致性”报表



评级员一致性(A, B, C)

频数		响应		
份额		0	1	总响应数
响应	A	50 33.3%	100 66.7%	150
	B	47 31.3%	103 68.7%	150
	C	51 34.0%	99 66.0%	150

份额图		响应		
		0	1	
响应	A			150
	B			150
	C			150

评级员一致性统计量

一致性程度

一致性程度

响应 1	响应 2	Kappa	标准误差	Bowker 对称性	Bowker p 值
A	B	0.862944	0.044198	1	0.3173
A	C	0.776119	0.054671	0.066667	0.7963
B	C	0.788007	0.053702	1.142857	0.2850

2 × 2 表与 Bowker 检验相同

详细信息

评级员行	评级员列	水平	0	1
B	A	0	44	3
B	A	1	6	97
C	A	0	43	8
C	A	1	7	92
C	B	0	42	9
C	B	1	5	94

正如 Kappa 统计量所显示的那样，评级员之间的一致性程度很高。Kappa 统计量的取值范围在 0（完全不一致）到 1.0（完全一致）之间。详细信息部分为每对评级员都提供 2x2 表格。Bowker 对称性检验用来检验以下原假设：单元格比例对于所有单元格对都是对称的（对于所有 i 和 j，$p_{ij} = p_{ji}$）。在此，Bowker 检验的 p 值全部大于 0.05，指示没有很强的证据表明评级员之间不对称。

重复测量的示例

本例使用 Presidential Elections.jmp 样本数据表，其中包含 1980 到 2012 年间美国各州的总统选举结果。使用该数据表探索重复测量，在此我们将选举结果视为重复测量。

1. 选择**帮助 > 样本数据库**，然后打开 Presidential Elections.jmp。
2. 选择**分析 > 消费者研究 > 分类**。
3. 选择 “相关”选项卡。
4. 选择 **1980 年获胜党**一直到 **2012 年获胜党**，然后点击 “相关”选项卡上的**重复测量**。
5. 点击**确定**。
6. 靠近报表窗口的底部，点击灰色 “转变报表”展开图标以打开 “转变报表”。

图 3.34 重复测量转变报表

转变报表

全部 自	到	转变数			转变比率			
全部 1980 年获胜党	1984 年获胜党		Democrat	Republican		Democrat	Republican	1984 年获胜党
		Democrat	1	5	Democrat	0.1667	0.8333	
		Republican	0	44	Republican	0.0000	1.0000	
全部 1984 年获胜党	1988 年获胜党		Democrat	Republican		Democrat	Republican	1988 年获胜党
		Democrat	1	0	Democrat	1.0000	0.0000	
		Republican	9	40	Republican	0.1837	0.8163	
全部 1988 年获胜党	1992 年获胜党		Democrat	Republican		Democrat	Republican	1992 年获胜党
		Democrat	10	0	Democrat	1.0000	0.0000	
		Republican	22	18	Republican	0.5500	0.4500	
全部 1992 年获胜党	1996 年获胜党		Democrat	Republican		Democrat	Republican	1996 年获胜党
		Democrat	29	3	Democrat	0.9063	0.0938	
		Republican	2	16	Republican	0.1111	0.8889	
全部 1996 年获胜党	2000 年获胜党		Democrat	Republican		Democrat	Republican	2000 年获胜党
		Democrat	20	11	Democrat	0.6452	0.3548	
		Republican	0	19	Republican	0.0000	1.0000	
全部 2000 年获胜党	2004 年获胜党		Democrat	Republican		Democrat	Republican	2004 年获胜党
		Democrat	18	2	Democrat	0.9000	0.1000	
		Republican	1	29	Republican	0.0333	0.9667	
全部 2004 年获胜党	2008 年获胜党		Democrat	Republican		Democrat	Republican	2008 年获胜党
		Democrat	19	0	Democrat	1.0000	0.0000	
		Republican	9	22	Republican	0.2903	0.7097	
全部 2008 年获胜党	2012 年获胜党		Democrat	Republican		Democrat	Republican	2012 年获胜党
		Democrat	26	2	Democrat	0.9286	0.0714	
		Republican	0	22	Republican	0.0000	1.0000	

“转变报表”是重复测量分析所特有的报表。该报表包括后续时间点之间差异的计数和比率。1980 年至 1984 年期间，有 5 个民主党州转变为共和党州，比率为 0.8333（即 6 个州中的 5 个州）。1980 年，这些州投票选举了民主党，但到 1984 年却改投共和党。2008 年至 2012 年期间，28 个民主党州中有 2 个民主党州转变为共和党州，比率为 0.0714。在 2008 年和 2012 年的两次选举中，其他所有州的投票选举的政党都没有变化。

多重响应的示例

本节中的示例使用的样本数据表包含相同的信息，但这些数据表使用了五种不同的布局。数据来自对两种不同条件下的三个不同场合的生产线的检验。每组生产条件（批次）都生成 50 个待检单元。检验员记录了七种缺陷。每个单元可能有 0、1 或不止一个缺陷。一个单元可能有多个同一类型的缺陷。

多重响应

Failure3MultipleField.jmp 样本数据表中每个单元对应一行，缺陷包含在多个列中，每列输入一个缺陷。在本例中，共有三个缺陷列。因此，任一个单元都至多有三个缺陷。

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure3MultipleField.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择 “多重”选项卡。
4. 选择**失败 1**、**失败 2** 和**失败 3**，然后点击 “多重”选项卡上的**多重响应**。

5. 选择**清洗**和**日期**，然后点击 **X，分组类别**。
6. 点击**确定**。

图 3.35 “多重响应”报表

多重响应(失败 1, 失败 2, 失败 3) 依据 清洗*日期

频数 份额 比率				失败									
				contamination	corrosion	doping	metallization	miscellaneous	oxide defect	silicon defect	总响应数	总案例数	响应的总案例数
清洗	after	日期	OCT 1	12	2	0	4	2	1	2	23	50	23
				52.2%	8.7%	0.0%	17.4%	8.7%	4.3%	8.7%			
				24.0%	4.0%	0.0%	8.0%	4.0%	2.0%	4.0%			
			OCT 2	10	1	1	5	1	2	3	23	50	23
				43.5%	4.3%	4.3%	21.7%	4.3%	8.7%	13.0%			
				20.0%	2.0%	2.0%	10.0%	2.0%	4.0%	6.0%			
			OCT 3	8	3	0	5	0	1	0	17	50	17
				47.1%	17.6%	0.0%	29.4%	0.0%	5.9%	0.0%			
				16.0%	6.0%	0.0%	10.0%	0.0%	2.0%	0.0%			
	before	日期	OCT 1	14	2	1	2	3	8	1	31	50	28
				45.2%	6.5%	3.2%	6.5%	9.7%	25.8%	3.2%			
				28.0%	4.0%	2.0%	4.0%	6.0%	16.0%	2.0%			
			OCT 2	15	2	2	1	4	6	0	30	50	30
				50.0%	6.7%	6.7%	3.3%	13.3%	20.0%	0.0%			
				30.0%	4.0%	4.0%	2.0%	8.0%	12.0%	0.0%			
			OCT 3	22	2	3	4	0	3	2	36	50	36
				61.1%	5.6%	8.3%	11.1%	0.0%	8.3%	5.6%			
				44.0%	4.0%	6.0%	8.0%	0.0%	6.0%	4.0%			

份额图				失败							
				contamination	corrosion	doping	metallization	miscellaneous	oxide defect	silicon defect	
清洗	after	日期	OCT 1								23
			OCT 2								23
			OCT 3								17
	before	日期	OCT 1								31
			OCT 2								30
			OCT 3								36

交叉表中的每个批次对应一行，每个缺陷类型对应一列。每个批次中的频数、份额和每个缺陷的比率都显示在表单元格中。例如，对于 10 月 1 日清洗过后的批次，有 12 个污染缺陷，表示为 12/23 或缺陷数占该批次的 52.2%。这 12 个污染缺陷来自 50 个单元。对于每个**清洗**和**日期**组合，都有总共 50 个单元。每个单元可有一个或多个缺陷。因此，每单元污染比率为 24%。对于 10 月 1 日清洗前的批次，“响应的总案例数”为 28。由于其中三个案例报告了两个缺陷，因此“总响应数”为 31。

由 ID 标识的多重响应

在 Failure3ID.jmp 样本数据表中，每个批次中的每个缺陷类型都对应一行，有一列包含每个缺陷类型出现的次数，并且有一个包含每个批次的 ID 列。

图 3.36 Failure3ID 数据表 （部分显示）

	失败	数量	清洗	日期	样本大小	ID
1	contamination	14	before	OCT 1	50	OCT 1 before
2	corrosion	2	before	OCT 1	50	OCT 1 before
3	doping	1	before	OCT 1	50	OCT 1 before
4	metallization	2	before	OCT 1	50	OCT 1 before
5	miscellaneous	3	before	OCT 1	50	OCT 1 before
6	oxide defect	8	before	OCT 1	50	OCT 1 before
7	silicon defect	1	before	OCT 1	50	OCT 1 before
8	doping	0	after	OCT 1	50	OCT 1 after
9	corrosion	2	after	OCT 1	50	OCT 1 after
10	metallization	4	after	OCT 1	50	OCT 1 after

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure3ID.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择 “多重”选项卡。
4. 选择**失败**并点击 “多重”选项卡上的**由 ID 标识的多重响应**。
5. 选择**清洗**和**日期**，然后点击 **X，分组类别**。
6. 选择**样本大小**并点击**样本大小**。
7. 选择**数量**并点击**频数**。
8. 选择 ID 并点击 ID。
9. 点击**确定**。

生成的报表与图 3.35 中显示的报表相同，只有交叉表中的“响应的总案例数”列除外。在此对缺陷计数进行了汇总。汇总表中不包含零缺陷的单元数的记录。因此，“响应的总案例数”对每个批次而言都是整批的大小，即 50。

分隔的多重响应

在 Failures3Delimited.jmp 样本数据表中，每个单元对应一行，并且有一列记录缺陷，缺陷之间由逗号分隔。请注意：在图 3.37 所示的部分显示的数据表中，某些单元没有观测到任何缺陷，所以**失败**列为空。

图 3.37 Failure3Delimited.jmp 数据表 （部分显示）

	失败	清洗	日期	ID	ID 标签
1		before	OCT 1	1	OCT 1 before
2	oxide defect	before	OCT 1	1	OCT 1 before
3	contamination,oxide def...	before	OCT 1	1	OCT 1 before
4		before	OCT 1	1	OCT 1 before
5	contamination	before	OCT 1	1	OCT 1 before
6	oxide defect	before	OCT 1	1	OCT 1 before
7	contamination	before	OCT 1	1	OCT 1 before
8		before	OCT 1	1	OCT 1 before
9		before	OCT 1	1	OCT 1 before
10	metallization,contaminat...	before	OCT 1	1	OCT 1 before

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/ Failures3Delimited.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择 “多重”选项卡。
4. 选择**失败**并点击 “多重”选项卡上的**分隔的多重响应**。
5. 选择**清洗**和**日期**，然后点击 **X，分组类别**。
6. 点击**确定**。

点击**确定**时，您还会得到图 3.35 中所示的报表。

注意：若指定了多个分隔列，将为每列生成单独的分析。

指示符组

在 Failures3Indicators.jmp 样本数据表中，每个单元对应一行，每个缺陷类型都对应一个指示符列。若未观测到该缺陷，则每个缺陷列中的数据条目都为 0；若观测到该单元存在该缺陷，则数据条目为 1。

图 3.38 Faliure3Indicators.jmp 数据表 （部分显示）

	清洗	日期	ID	ID 标签	污染	腐蚀	掺杂	金属化	杂项	氧化层缺陷	硅缺陷
1	before	OCT 1	1	OCT 1 before	0	0	0	0	0	0	0
2	before	OCT 1	1	OCT 1 before	0	0	0	0	0	1	0
3	before	OCT 1	1	OCT 1 before	1	0	0	0	0	1	0
4	before	OCT 1	1	OCT 1 before	0	0	0	0	0	0	0
5	before	OCT 1	1	OCT 1 before	1	0	0	0	0	0	0

1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failures3Indicators.jmp。

2. 选择**分析** > **消费者研究** > **分类**。
3. 选择 “多重”选项卡。
4. 选择**污染**一直到**硅缺陷**，然后点击 “多重”选项卡上的**指示符组**。
5. 选择**清洗**和**日期**，然后点击 **X，分组类别**。
6. 点击**确定**。

点击**确定**时，您会得到图 3.35 中所示的报表。

响应频数

在 Failure3Freq.jmp 样本数据表中，每个批次都对应一行，每个缺陷类型都对应一列，还有一列包含批次大小。缺陷列中的数据条目是批次中出现每个缺陷的次数。

图 3.39 Failure3Freq.jmp 数据表

	清洗	日期	污染	腐蚀	掺杂	金属化	杂项	氧化层缺陷	硅缺陷	样本大小
1	after	OCT 1	12	2	0	4	2	1	2	50
2	after	OCT 2	10	1	1	5	1	2	3	50
3	after	OCT 3	8	3	0	5	0	1	0	50
4	before	OCT 1	14	2	1	2	3	8	1	50
5	before	OCT 2	15	2	2	1	4	6	0	50
6	before	OCT 3	22	2	3	4	0	3	2	50

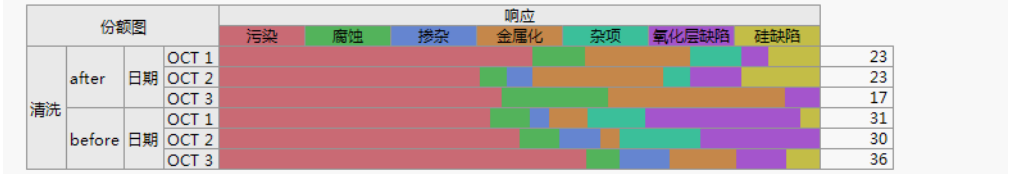
1. 选择**帮助** > **样本数据库**，然后打开 Quality Control/Failure3Freq.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 选择 “多重”选项卡。
4. 选择频数变量 （**污染**一直到**硅缺陷**）。
5. 在 “多重”选项卡上，点击**响应频数**。
6. 选择**清洗**和**日期**，然后点击 **X，分组类别**。
7. 选择**样本大小**并点击**样本大小**。
8. 点击**确定**。

图 3.40 缺陷率输出

样本大小: 样本大小

响应频数(污染, 腐蚀, 掺杂, 金属化, 杂项, 氧化层缺陷, 硅缺陷) 依据 清洗*日期

				频数 份额 比率	响应									
					污染	腐蚀	掺杂	金属化	杂项	氧化层缺陷	硅缺陷	总响应数	总案例数	响应的总案例数
清洗	after	日期	OCT 1		12	2	0	4	2	1	2	23	50	50
					52.2%	8.7%	0.0%	17.4%	8.7%	4.3%	8.7%			
					24.0%	4.0%	0.0%	8.0%	4.0%	2.0%	4.0%			
			OCT 2		10	1	1	5	1	2	3	23	50	50
					43.5%	4.3%	4.3%	21.7%	4.3%	8.7%	13.0%			
					20.0%	2.0%	2.0%	10.0%	2.0%	4.0%	6.0%			
			OCT 3		8	3	0	5	0	1	0	17	50	50
					47.1%	17.6%	0.0%	29.4%	0.0%	5.9%	0.0%			
					16.0%	6.0%	0.0%	10.0%	0.0%	2.0%	0.0%			
	before	日期	OCT 1		14	2	1	2	3	8	1	31	50	50
					45.2%	6.5%	3.2%	6.5%	9.7%	25.8%	3.2%			
					28.0%	4.0%	2.0%	4.0%	6.0%	16.0%	2.0%			
			OCT 2		15	2	2	1	4	6	0	30	50	50
					50.0%	6.7%	6.7%	3.3%	13.3%	20.0%	0.0%			
					30.0%	4.0%	4.0%	2.0%	8.0%	12.0%	0.0%			
			OCT 3		22	2	3	4	0	3	2	36	50	50
					61.1%	5.6%	8.3%	11.1%	0.0%	8.3%	5.6%			
					44.0%	4.0%	6.0%	8.0%	0.0%	6.0%	4.0%			



生成的输出结果与图 3.35 中的结果相同，只有交叉表中的“响应的总案例数”列除外。在此对缺陷计数进行了汇总。汇总表中不包含零缺陷的单元数的记录。因此，“响应的总案例数”对每个批次而言都是整批的大小，即 50。

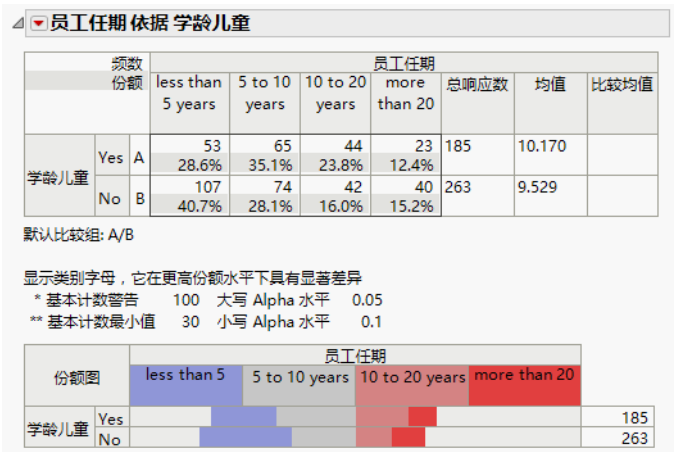
带比较字母的得分均值的示例

本例使用 Consumer Preferences.jmp 样本数据表探索员工任期与有学龄儿童之间的关系。**员工任期**列是值为 1、2、3 和 4 的数值列。已使用“值标签”列属性为这些值指定了值标签。要使用“分类”平台中的“得分均值”选项评估员工任期均值，您需要为列值分配“值得分”。有关列属性的详细信息，请参见《使用 JMP》。

1. 选择**帮助** > **样本数据库**，然后打开 Consumer Preferences.jmp。
2. 在该数据表中，右击**员工任期**列标题，然后选择**列属性** > **值得分**。
3. 为**值**输入 1，为**得分**输入 3，然后点击**添加**。
4. 为**值**输入 2，为**得分**输入 7.5，然后点击**添加**。
5. 为**值**输入 3，为**得分**输入 15，然后点击**添加**。
6. 为**值**输入 4，为**得分**输入 25，然后点击**添加**。
7. 点击**确定**。
8. 选择**分析** > **消费者研究** > **分类**。

9. 选择**员工任期**，并在 “简单”选项卡上点击**响应**。
10. 选择**学龄儿童**并点击 **X，分组类别**。
11. 点击**确定**。
12. 点击 “分类”红色小三角并选择**得分均值**。
13. 点击 “分类”红色小三角并选择**得分均值比较**。

图 3.41 包含得分均值的分类报表



有学龄儿童的员工的员工任期均值为 10.17，没有学龄儿童的员工的员工任期均值为 9.53。这两个均值在统计上没有不同，因为交叉表中的“比较均值”列为空。若有差异，会有字母显示。若未使用值得分，则有学龄儿童的员工的均值为 2.20，没有学龄儿童的员工的均值为 2.057。

提示：在使用得分均值选项时，要注意数据的记录形式。若您的数据记录为带值标签的编码数值数据，则均值计算基于该数值数据。若数值没有意义，请使用“值得分”列属性将有意义的值分配给响应水平。

结构化报表的示例

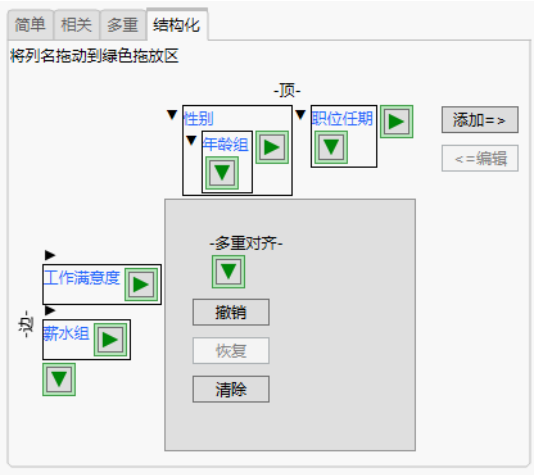
本例使用 Consumer Preferences.jmp 样本数据表来比较不同年龄组和不同职位任期的工作满意度和薪水情况。使用 “结构化”选项卡创建报表。

1. 选择**帮助 > 样本数据库**，然后打开 Consumer Preferences.jmp。
2. 选择**分析 > 消费者研究 > 分类**。
3. 选择 “结构化”选项卡。
4. 将**性别**拖至 “结构化”选项卡上的表**顶**处的绿色拖放区。
5. 将**年龄组**拖至**性别**正下方的绿色拖放区。

6. 将**职位任期**拖至表**顶**处**性别**旁边的绿色拖放区。
7. 将**工作满意度**拖至表**边**处的绿色拖放区。
8. 将**薪水组**拖至表**边**处**工作满意度**下方的绿色拖放区。

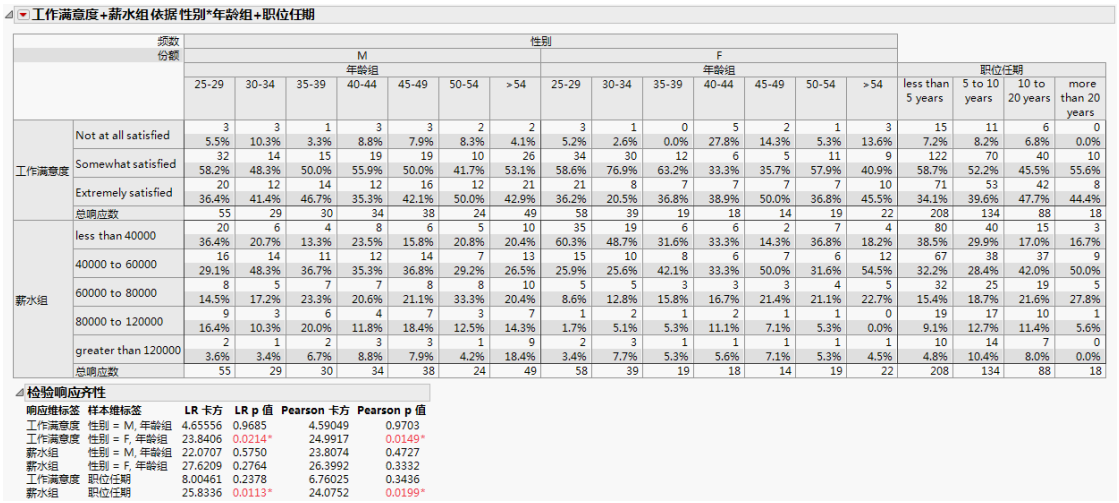
提示：点击绿色拖放区箭头选择列。

图 3.42 “结构化”选项卡报表设置



9. 点击**添加 =>**。
10. 点击**确定**。
11. 点击“分类”红色小三角并选择**检验响应齐性**。

图 3.43 “结构化”选项卡报表示例



工作满意度+薪水组 依据 性别*年龄组+职位任期

频数 / 份额		性别																	
		M							F							职位任期			
		年龄组							年龄组										
		25-29	30-34	35-39	40-44	45-49	50-54	>54	25-29	30-34	35-39	40-44	45-49	50-54	>54	less than 5 years	5 to 10 years	10 to 20 years	more than 20 years
工作满意度	Not at all satisfied	3	3	1	3	3	2	2	3	1	0	5	2	1	3	15	11	6	0
		5.5%	10.3%	3.3%	8.8%	7.9%	8.3%	4.1%	5.2%	2.6%	0.0%	27.8%	14.3%	5.3%	13.6%	7.2%	8.2%	6.8%	0.0%
	Somewhat satisfied	32	14	15	19	19	10	26	34	30	12	6	5	11	9	122	70	40	10
		58.2%	48.3%	50.0%	55.9%	50.0%	41.7%	53.1%	58.6%	76.9%	63.2%	33.3%	35.7%	57.9%	40.9%	58.7%	52.2%	45.5%	55.6%
	Extremely satisfied	20	12	14	12	16	12	21	21	8	7	7	7	7	10	71	53	42	8
		36.4%	41.4%	46.7%	35.3%	42.1%	50.0%	42.9%	36.2%	20.5%	36.8%	38.9%	50.0%	36.8%	45.5%	34.1%	39.6%	47.7%	44.4%
	总响应数	55	29	30	34	38	24	49	58	39	19	18	14	19	22	208	134	88	18
薪水组	less than 40000	20	6	4	8	6	5	10	35	19	6	6	2	7	4	80	40	15	3
		36.4%	20.7%	13.3%	23.5%	15.8%	20.8%	20.4%	60.3%	48.7%	31.6%	33.3%	14.3%	36.8%	18.2%	38.5%	29.9%	17.0%	16.7%
	40000 to 60000	16	14	11	12	14	7	13	15	10	8	6	7	6	12	67	38	37	9
		29.1%	48.3%	36.7%	35.3%	36.8%	29.2%	26.5%	25.9%	25.6%	42.1%	33.3%	50.0%	31.6%	54.5%	32.2%	28.4%	42.0%	50.0%
	60000 to 80000	8	5	7	7	8	8	10	5	5	3	3	3	4	5	32	25	19	5
		14.5%	17.2%	23.3%	20.6%	21.1%	33.3%	20.4%	8.6%	12.8%	15.8%	16.7%	21.4%	21.1%	22.7%	15.4%	18.7%	21.6%	27.8%
	80000 to 120000	9	3	6	4	7	3	7	1	2	1	2	1	1	0	19	17	10	1
		16.4%	10.3%	20.0%	11.8%	18.4%	12.5%	14.3%	1.7%	5.1%	5.3%	11.1%	7.1%	5.3%	0.0%	9.1%	12.7%	11.4%	5.6%
	greater than 120000	2	1	2	3	3	1	9	2	3	1	1	1	1	1	10	14	7	0
		3.6%	3.4%	6.7%	8.8%	7.9%	4.2%	18.4%	3.4%	7.7%	5.3%	5.6%	7.1%	5.3%	4.5%	4.8%	10.4%	8.0%	0.0%
	总响应数	55	29	30	34	38	24	49	58	39	19	18	14	19	22	208	134	88	18

检验响应齐性

响应维标签	样本维标签	LR 卡方	LR p 值	Pearson 卡方	Pearson p 值
工作满意度	性别 = M, 年龄组	4.65556	0.9685	4.59049	0.9703
工作满意度	性别 = F, 年龄组	23.8406	0.0214*	24.9917	0.0149*
薪水组	性别 = M, 年龄组	22.0707	0.5750	23.8074	0.4727
薪水组	性别 = F, 年龄组	27.6209	0.2764	26.3992	0.3332
工作满意度	职位任期	8.00461	0.2378	6.76025	0.3436
薪水组	职位任期	25.8336	0.0113*	24.0752	0.0199*

“结构化”选项卡报表包含您在“结构化”选项卡中指定的表。响应齐性检验针对每一种分组变量组合。我们可以看到，对于男性，各年龄组的工作满意度没有差别（Pearson p 值 = 0.9703）。对于女性，各年龄组的工作满意度有差别（Pearson p 值 = 0.0149）。中年女性往往对其工作最不满意。可将份额图和频数图添加到报表中，用来形象演示结果。

无响应的多重响应示例

本例使用 Color Preference Survey.jmp 样本数据表，其中包含关于人们的颜色首选项的调查数据。您可以使用“分类”平台来汇总没有响应的多重响应问题的结果。无响应是指参与者未回答问题。调查通常使用“以上全不是”或“不适用”作为响应选择。这提供了一种方法来区分未找到合适响应的响应者与未回答问题的响应者。

1. 选择**帮助** > **样本数据库**，然后打开 Color Preference Survey.jmp。
2. 选择**分析** > **消费者研究** > **分类**。
3. 点击**结构化**选项卡。
4. 选择**您喜欢的颜色 （包含无响应）？**并将其拖入表**边**的绿色箭头拖放区。

> **注意：**该列包含多重响应数据，其中各响应水平用逗号分隔。有三行未包含任何响应。该列是为本例构造的，与 “您喜欢什么颜色”各列中的响应不对齐。

5. 点击**顶**绿色箭头并选择**您的性别？**
6. 点击**添加 =>**。
7. 点击**确定**。

图 3.44 初始交叉表

您喜欢的颜色（包含无响应）？依据 您的性别？

		频数 份额 比率	您的性别？	
			Female	Male
您喜欢的颜色（包含无响应）？	Blue		31 21.5% 83.8%	38 21.5% 88.4%
	Green		24 16.7% 64.9%	36 20.3% 83.7%
	None of the above		2 1.4% 5.4%	0 0.0% 0.0%
	Orange		12 8.3% 32.4%	23 13.0% 53.5%
	Pink		19 13.2% 51.4%	9 5.1% 20.9%
	Purple		28 19.4% 75.7%	23 13.0% 53.5%
	Red		18 12.5% 48.6%	33 18.6% 76.7%
	Yellow		10 6.9% 27.0%	15 8.5% 34.9%
	总响应数		144	177
	总案例数		37	43
	响应的总案例数		36	41

请注意，总案例数高于总响应案例数。有 37 名女性填写了调查，但只有 36 人回答了这个问题。

8. 点击“分类”红色小三角并选择**检验多重响应 > 排除非响应**。
9. 点击“分类”红色小三角并选择**检验多重响应 > 齐性检验，二项**。

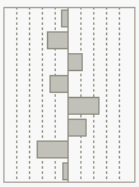
图 3.45 二项齐性检验结果

检验每个响应，二项

注意: 排除非响应行。

您的性别？, 您喜欢的颜色（包含无响应）？

您喜欢的颜色（包含无响应）？	卡方	概率>卡方
Blue	0.8918	0.3450
Green	5.0612	0.0245*
None of the above	3.1020	0.0782
Orange	4.0516	0.0441*
Pink	7.9930	0.0047*
Purple	4.1132	0.0426*
Red	8.0999	0.0044*
Yellow	0.6819	0.4089



假设多个响应可以全部选择 (CATA)。

分析支持蓝色和黄色在不同性别之间有相似的偏好水平，而其他颜色往往在不同性别之间有或多或少的偏好。例如，红色似乎更受男性而不是女性的喜爱。检验的 p 值为 0.0044，这为不同性别的偏好存在差异这一结论提供了支持。从图 3.44 中的交叉表，您发现 76.7% 的男性喜欢红色，而喜欢红色的女性只有 48.6%。

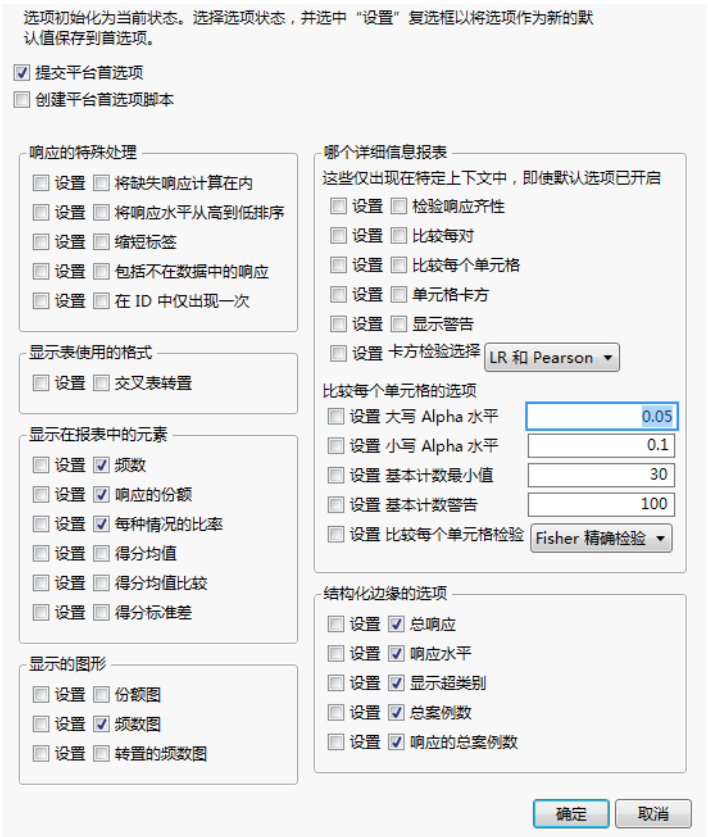
输出顶部的注释指出分析中排除了无响应行。输出底部的注释表明，假设多重响应是一个“选中所有适用项”类型的问题。这意味着响应者可以选择针对问题的多个响应。仅使用具有响应的那些行执行检验。不包括没有对问题作出响应的三行。

提示：您可以使用“多重”选项卡获取相同的输出。将**您喜欢的颜色（包含无响应）？**选作“分隔的多重响应”并将**您的性别？**选作分组变量。

设置首选项

“分类”红色小三角菜单提供 “设置首选项”选项，该选项支持您指定设置和首选项。

图 3.46 “设置首选项”窗口



为您要设置的选项选择**设置**框。若想要默认显示该选项，请选中该选项框；若不想默认显示该选项，请取消选中该选项框。要提交您对平台首选项的更改，请选中**提交平台首选项**框。要将您所做的更改另存为首选项脚本，请选中**创建平台首选项脚本**框。启动“分类”平台时，将使用与当前首选项集关联的首选项创建“分类”报表。

运行保存的脚本可将这些首选项提交给平台首选项。您可以使用平台首选项脚本在多个用户之间共享首选项集，或保存这些设置以便用于特定项目。

“分类”平台的统计详细信息

该部分包含 “分类”平台的统计详细信息。

Rao-Scott 校正

Rao-Scott 校正应用于多重响应的响应齐性检验。请参见 Lavassani et al.(2009)。

若存在多重响应，您可以有重叠样本，这意味着单个参与者可提供不止一个响应。Pearson 卡方检验不适用于多重响应，因为多重响应违反了 Pearson 卡方检验的独立性假设。此外，使用边缘合计计算的期望值受多重响应影响，因为若不允许多重响应则合计会变大。

Rao-Scott 卡方统计量定义如下：

$$\chi_C^2 = \frac{\chi^2}{\bar{\delta}}$$

其中

χ^2 是标准 Pearson 卡方统计量

$\bar{\delta} = 1 - \frac{m_{++}}{n_+ C}$ 是校正因子

校正因子包含以下量：

m_{++} 是多重响应的合计数

n_+ 是参与者合计数

C 是响应水平数 （交叉表中的列数）。

自由度为 $(R\text{-}1)C$ 或行数减 1 后与列数的乘积。

第 4 章

选择模型

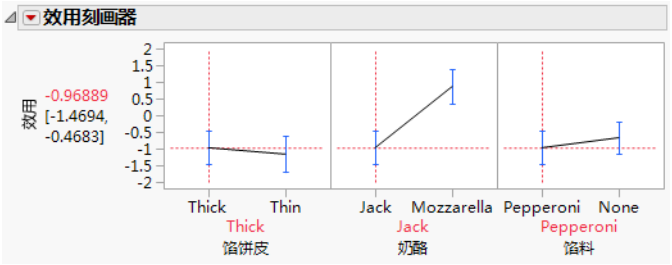
针对选择实验拟合模型

使用“选择”平台可分析市场研究过程中执行的选择实验的结果。借助选择实验可以发现您的潜在客户偏好哪种产品或服务特性。您可以使用上述信息来设计具有消费者最喜爱的特性的产品或服务。

“选择”平台支持您执行以下操作：

- 使用有关测试对象（客户）特点以及产品特性的信息。
- 分析选择实验，响应者可在实验中选择“以上都不是”。
- 整合来自一个、两个或三个源的数据。
- 使用集成的刻画器来了解、可视化和优化响应（效用）曲面。
- 获取测试对象级别的得分以便对数据进行细分或聚类。
- JMP PRO 使用 Bayes 方法估计特定于测试对象的系数。
- 使用修正偏倚的最大似然估计量 (Firth, 1993)。

图 4.1 “选择”平台的效用刻画器



目录

“选择建模”平台概述 75
“选择”平台的示例 76
“无选择”的单表格式 76
多表格式 79
启动“选择”平台 84
针对“一个表，堆叠”的启动窗口 85
针对“多个表，交叉引用”的启动窗口 87
“选择模型”报表 91
效应汇总 92
参数估计值 93
似然比检验 93
Bayes 参数估计值 93
“选择”平台选项 94
支付意愿 97
更多示例 99
示例：做出设计决策 99
细分示例 107
使用“选择”平台进行 Logistic 回归的示例 111
配对的案例对照研究的 Logistic 回归示例 114
将数据转换为两个分析表的示例 115
将数据转换为一个分析表的示例 120
“选择”平台的统计详细信息 122
特殊数据表规则 122
效用和概率 123
梯度 123

“选择建模”平台概述

选择建模最早由 McFadden (1974) 提出，它是一种功能强大的分析方法，用于估计个体从现有备择方案中做出特定选择的概率。选择建模也被称为联合选择建模、离散选择分析和条件 Logistic 回归。

选择实验研究客户对于一组产品或流程（针对服务的情况）特性的偏好。响应者会面临若干产品特性组合（称为**特征**）。选择实验会提供给每个响应者一小组特征集合（称为**选择集**），让他们选择自己最偏好的选项。通常会给每个响应者都提供若干选择集。使用“选择”平台可分析选择实验的结果。

注意：您可以使用“选择设计”平台来设计您的选择实验。请参见《实验设计指南》。

由于客户对于特性有不同评价，许多市场研究人员将市场细分视为分析选择实验的重要步骤。不这样做，您设计出的产品或流程取悦的可能就是根本不存在的“一般”客户，从而忽略了**真正**存在的细分市场的偏好。

有关选择建模的背景信息，请参见 Louviere et al. (2015)、Train (2009) 和 Rossi et al. (2005) 一书。

“选择”平台

“选择建模”平台使用条件 Logistic 回归的形式估计偏好某种配置的概率。不同于简单 Logistic 回归，选择建模使用线性模型对选择与否进行建模时，不仅仅基于测试对象的特征，同时还基于响应特性。在选择建模中，响应者可能会在两辆车中进行选择，而这些车可通过以下 10 种不同特性的组合来描述：价格、载客人数、茶杯架数、颜色、GPS 设备、汽油里程数、防盗系统、可拆卸的座位、安全装置数和保险费用。

“选择”平台允许响应者在一组特征中**不**做选择。**无选择**选项被视作只有一个特性（选择“以上都不是”）的产品，响应者可以选择。随后可根据模型假设通过多种方式解释“无选择”特性的参数估计值。“选择”平台还支持您获取测试对象级别信息，这些信息有助于对偏好模式进行细分。

您可以获取修正偏倚的最大似然估计量，如 Firth (1993) 所述。已证实该方法与没有修正偏倚的 MLE 相比能够得到更好的估计值和检验。此外，修正偏倚的 MLE 可缓解 Logistic 模型中常见的分离问题。有关 Logistic 回归中的分离问题的讨论，请参见 Heinze and Schemper (2002)。

注意：“选择”平台不适用于涉及排名、评分或嵌套分层选择的拟合模型。您可以使用 SAS/ETS 中的 PROC MDC 进行这些分析。

在开发产品和服务时运用选择设计

尽管客户满意度调查可以揭示产品或服务哪些方面做得不够好，但是对于特定的产品特性，它们无从得知消费者的偏好。工程师设计产品时，通常会进行数百或数千个小的设计决策。若客

户测试是可行的，同时又可以找到调查参与者（测试对象），这时您就可以通过选择实验来指导一些设计决策。

设计产品的过程中，在调查实施、建模和原型设计等方面成本的降低，推进了针对客户对许多特性和备选方案的评估。选择建模可用在“六西格玛”计划中，用来改进消费品，或是更概括地说，制造人们所需的产品。选择实验收集关于客户偏好的数据，而选择建模分析将这些偏好揭示出来。

细分

市场研究人员有时想分别分析不同测试对象的偏好结构，以查看是否存在不同表现行为的群体。但是，对于一般的估计，通常没有足够的数据来进行类似的分析。若有足够的数据，可以在“响应数据”中将测试对象标识符指定为“依据组”，或者您还可以引入测试对象标识符作为测试对象级的模型项。不过若测试对象人数很多的话，此方法的成本会很高。

若没有足够数据指定“依据组”，则可以使用“按对象保存梯度”选项对测试对象进行聚类的方法在 JMP 中进行细分。该选项创建一个新数据表，其中包含每个测试对象各参数的平均 Hessian 比例梯度。有关示例，请参见第 107 页的“细分示例”。有关梯度值的详细信息，请参见第 123 页的“梯度”。

JMP PRO 在 JMP Pro 中，您可以请求“选择”平台使用分层 Bayes 方法以便于市场细分。Bayes 建模提供特定于测试对象的模型参数估计值（亦称成分效用值）。可通过层次聚类或其他某类聚类分析来分析这些参数，从而揭示细分市场。

“选择”平台的示例

在比萨饼偏好的研究中，每个响应者都要从四个选择集中进行选择，每个选择集又含有两个特征。“选择”平台可以分析以单表格式或多数据格式呈现的数据。在多表格式中，关于响应、选择集和测试对象的信息保存在不同的数据表中。在单表格式中，该信息包含在单个数据表中。

- 第 76 页的“‘无选择’的单表格式”显示如何分析采用单表格式的一部分可用数据。
- 第 79 页的“多表格式”显示如何将不同表中的信息集中到一个选择分析中。

“无选择”的单表格式

在本例中，某些响应者对任一个特征都不表示偏好。这些响应者 “无选择” 。若某个响应者不表示偏好，则为该响应者输入的选择指示符为缺失。

1. 选择**帮助** > **样本数据库**，然后打开 Pizza Combined No Choice.jmp。

   通过**试尝者**和**试尝**的组合来定义选择集。请注意，**指示符**列中某些选择集的值缺失。

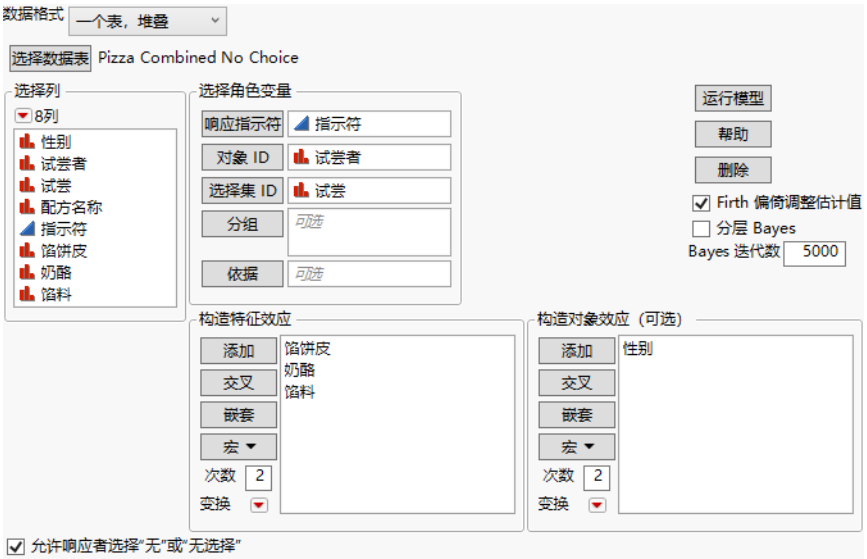
2. 选择**分析** > **消费者研究** > **选择**。

   “一个表，堆叠”数据格式是默认格式。

3. 点击**选择数据表**。

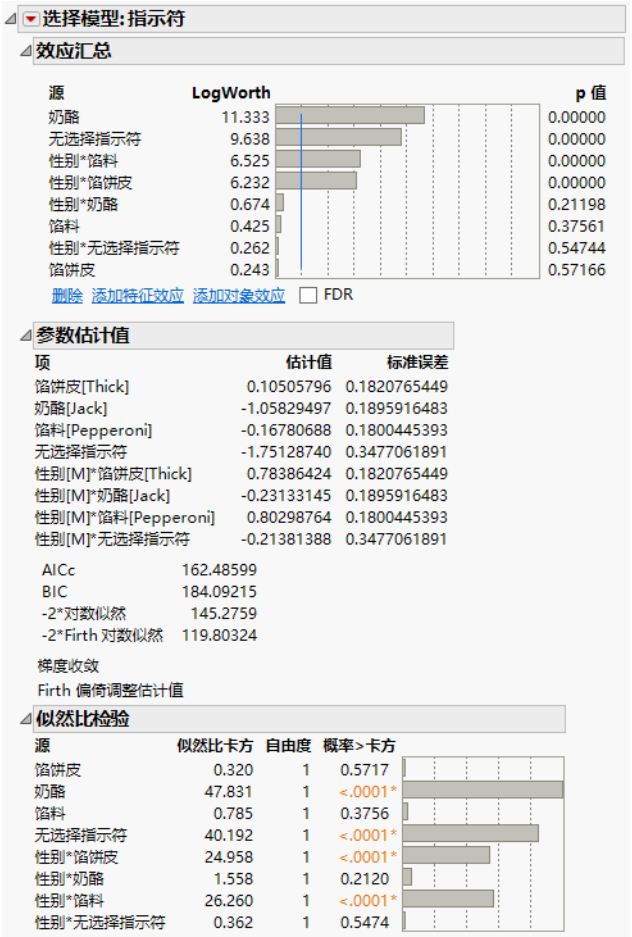
4. 选择 Pizza Combined No Choice，然后点击**确定**。
5. 完成启动窗口：
   - 选择**指示符**，然后点击**响应指示符**。
   - 选择**试尝者**，然后点击**对象 ID**。
   - 选择**试尝**，然后点击**选择集 ID**。
   - 选择**馅饼皮**、**奶酪**和**馅料**，然后在“构造特征效应”面板中点击**添加**。
   - 选择**性别**，然后在“构造对象效应（可选）”面板中点击**添加**。

图 4.2 完成的启动窗口



6. 选中**允许响应者选择 “无”或 “无选择”**旁边的框。
7. 点击**运行模型**。

图 4.3 报表将 “无选择”显示为一个效应


选择模型: 指示符

效应汇总

源	LogWorth	p 值
奶酪	11.333	0.00000
无选择指示符	9.638	0.00000
性别*馅料	6.525	0.00000
性别*馅饼皮	6.232	0.00000
性别*奶酪	0.674	0.21198
馅料	0.425	0.37561
性别*无选择指示符	0.262	0.54744
馅饼皮	0.243	0.57166

删除 添加特征效应 添加对象效应 FDR

参数估计值

项	估计值	标准误差
馅饼皮[Thick]	0.10505796	0.1820765449
奶酪[Jack]	-1.05829497	0.1895916483
馅料[Pepperoni]	-0.16780688	0.1800445393
无选择指示符	-1.75128740	0.3477061891
性别[M]*馅饼皮[Thick]	0.78386424	0.1820765449
性别[M]*奶酪[Jack]	-0.23133145	0.1895916483
性别[M]*馅料[Pepperoni]	0.80298764	0.1800445393
性别[M]*无选择指示符	-0.21381388	0.3477061891

AICc	162.48599
BIC	184.09215
-2*对数似然	145.2759
-2*Firth 对数似然	119.80324

梯度收敛

Firth 偏倚调整估计值

似然比检验

源	似然比卡方	自由度	概率>卡方
馅饼皮	0.320	1	0.5717
奶酪	47.831	1	<.0001*
馅料	0.785	1	0.3756
无选择指示符	40.192	1	<.0001*
性别*馅饼皮	24.958	1	<.0001*
性别*奶酪	1.558	1	0.2120
性别*馅料	26.260	1	<.0001*
性别*无选择指示符	0.362	1	0.5474

“效应汇总”报表按显著性高低显示效应。**奶酪**是显著性最高的效应，随后是“无选择指示符”，它被视为模型效应。对象效应交互作用**性别** * **馅料**和**性别** * **馅饼皮**也很显著，这表明不同**性别**的细分市场对**馅料**和**馅饼皮**的偏好不同。

要深入了解“无选择”响应的本质，请选择并查看结果为“无选择”的那些选择集。

8. 在数据表中，右击**指示符**列中缺失响应的某个单元格，然后选择**选择匹配单元格**。
9. 在“行”面板中，右击**已选定**，再选择**数据视图**。

图 4.4 带有 “无选择”响应的选择集

	性别	试尝者	试尝	配方名称	指示符	馅饼皮	奶酪	馅料
1	F	2	4	TrimPepperjack	•	Thin	Jack	Pepperoni
2	F	2	4	TrimOni	•	Thin	Mozzare...	Pepperoni
3	M	7	2	Trimella	•	Thin	Mozzare...	None
4	M	7	2	TrimJack	•	Thin	Jack	None
5	M	7	3	Trimella	•	Thin	Mozzare...	None
6	M	7	3	TrimJack	•	Thin	Jack	None
7	F	8	2	ThickElla	•	Thick	Mozzare...	None
8	F	8	2	ThickJack	•	Thick	Jack	None
9	M	11	4	ThickOni	•	Thick	Mozzare...	Pepperoni
10	M	11	4	ThickJackoni	•	Thick	Jack	Pepperoni
11	F	14	2	TrimOni	•	Thin	Mozzare...	Pepperoni
12	F	14	2	TrimPepperjack	•	Thin	Jack	Pepperoni
13	F	18	3	ThickJack	•	Thick	Jack	None
14	F	18	3	ThickElla	•	Thick	Mozzare...	None
15	F	24	3	ThickJack	•	Thick	Jack	None
16	F	24	3	TrimOni	•	Thin	Mozzare...	Pepperoni
17	M	29	1	TrimPepperjack	•	Thin	Jack	Pepperoni
18	M	29	1	ThickJackoni	•	Thick	Jack	Pepperoni

在图 4.4 所示的表中，考虑前七个选择集中的特征 （这些选择集由第 1 到 14 行中的**试尝者**和**试尝**组合来定义）。每个选择集中唯一的不同就是**奶酪**。有迹象表明某些响应者可能无法检测出奶酪之间的差异。不过，分析中已经考虑了 “无选择指示符”，因此可以得出：即使存在这些迹象，**奶酪**仍是显著的。

要了解如何进一步分析此类型的数据，请参见第 82 页的“查找最优特征”。

多表格式

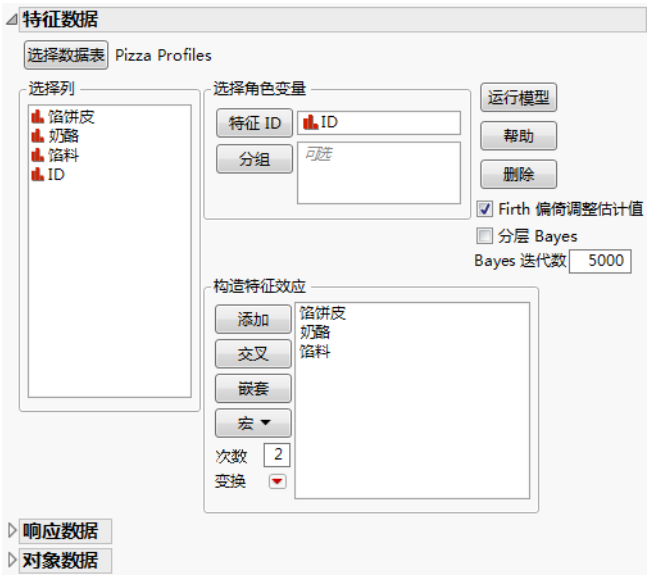
在本例中，供响应者选择的比萨饼包含三个特性 （每个特性有两个水平）。按照本研究的设计，响应者不得不作出某项选择。分析中使用三个数据表：Pizza Profiles.jmp、Pizza Responses.jmp 和 Pizza Subjects.jmp。尽管您总是可将数据整合到单个表中，但若您想要在分析中加入其他特征和测试对象变量，多表方法比单表分析更为方便。

1. 选择**帮助** > **样本数据库**，然后打开 Pizza Profiles.jmp、Pizza Responses.jmp 和 Pizza Subjects.jmp。
   - 特征数据表 Pizza Profiles.jmp 列出了您要提供给测试对象的所有比萨饼选择组合。每个选择组合都有自己的一个 ID。
   - 响应数据表 Pizza Responses.jmp 包含设计和结果。在实验中，为每个测试对象提供四个选择集，每个选择集都包含两个选择特征（**选择1** 和**选择2**）。测试对象为每个选择集选择一种偏好（**选择**）。有关如何构造选择设计的信息，请参见《实验设计指南》。请注意，**选择**列中的每个值都对应于包含特性信息的“特征”数据表中的 ID 值。
   - 对象数据表 Pizza Subjects.jmp 包括一个**试尝者** ID 列和测试对象的一个特征 — **性别**。Pizza Subjects.jmp 数据表中的每个**试尝者**值都与 Pizza Responses.jmp 数据表中的**试尝者**列中的值相对应。
2. 选择**分析** > **消费者研究** > **选择**以打开启动窗口。

注意：可以从三个打开的数据表中的任一个执行该操作。

3. 从“数据格式”菜单中，选择**多个表，交叉引用**。
   窗口分为三个不同部分，每个部分对应一个数据源。
4. 在“特征数据”下点击**选择数据表**。
   将显示“特征数据表”窗口，该窗口提示您指定特征数据所在的数据表。
5. 选择 Pizza Profiles，然后点击**确定**。
6. 选择 ID 并点击**特征 ID**。
7. 选择**馅饼皮**、**奶酪**和**馅料**并点击**添加**。

图 4.5 特征数据



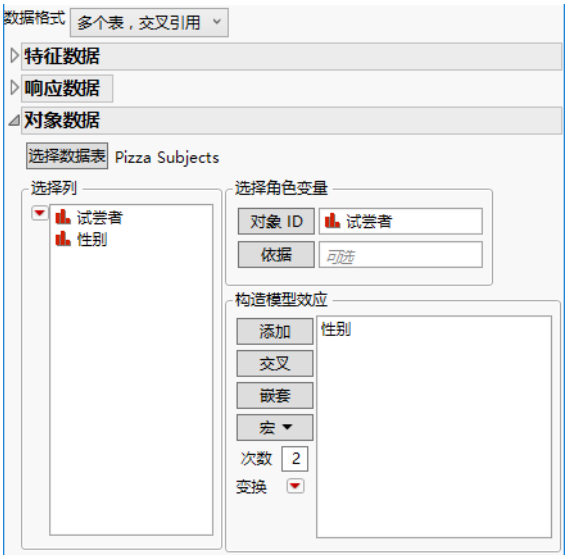
8. 点击“响应数据”旁边的展开图标以打开该分级显示项，然后点击**选择数据表**。
9. 选择 Pizza Responses，然后点击**确定**。
10. 输入“特征 ID”选择：
    - 选择**选择**，然后点击**已选择的特征 ID**。
    - 选择**选择1** 和**选择2**，然后点击**特征 ID 选择**。
    - 选择**试尝者**，然后选择**对象 ID**。

图 4.6 “响应数据”窗口

选择 1 和**选择 2** 是提供给测试对象的四个选择集中的特征。**选择**列包含在**选择 1** 和**选择 2** 中做出的偏好选择。

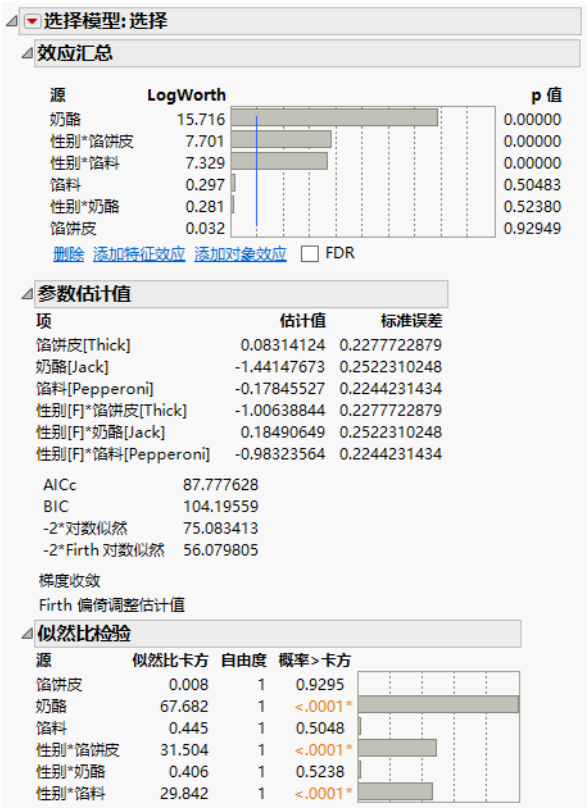
11. 点击“对象数据”旁边的展开图标以打开该分级显示项，然后点击**选择数据表**。
12. 选择 Pizza Subjects，然后点击**确定**。
13. 选择**试尝者**，然后点击**对象 ID**。
14. 选择**性别**并点击**添加**。

图 4.7 “对象数据”窗口



15. 点击**运行模型**。

图 4.8 “选择模型”结果



模型有六个效应。效应**馅饼皮**、**奶酪**和**馅料**是产品特性。交互作用效应 — **性别 * 馅饼皮**、**性别 * 奶酪**和**性别 * 馅料** — 是测试对象和产品特性的交互作用效应。分析这些交互作用效应，您可以打造不同的产品以满足不同细分市场的偏好。

注意：对于选择模型，对象效应无法输入为主效应。它们只显示为交互作用项。

“效应汇总”报表和“似然比检验”报表显示**性别**和**馅饼皮**以及**性别**和**馅料**之间较强的交互作用。请注意，主效应**馅饼皮**和**馅料**不显著。若您不包含测试对象级别的效应，则可能忽略与市场细分相关的重要信息。

查找最优特征

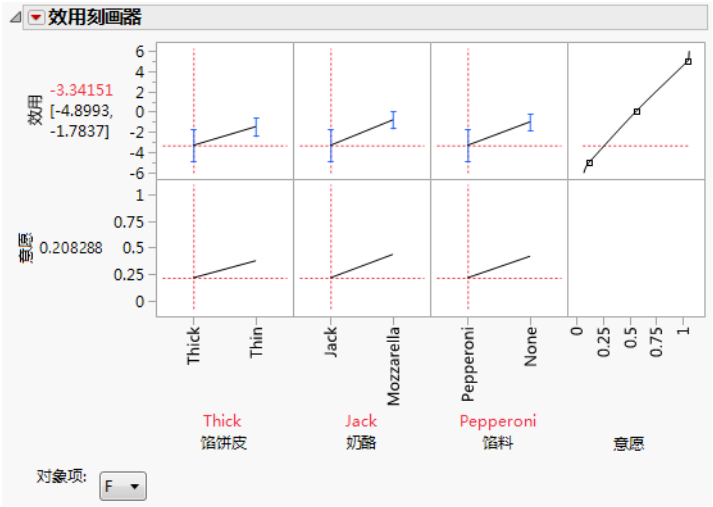
下一步，您需要使用“效用刻画器”探索结果并找出特性的最优设置。

1. 点击“选择模型 : 选择”红色小三角并选择**效用刻画器**。

   该刻画器下方的“对象项”菜单指示它显示的是女性的结果。

2. 点击“效用刻画器”红色小三角并选择**优化和意愿 > 意愿函数**。

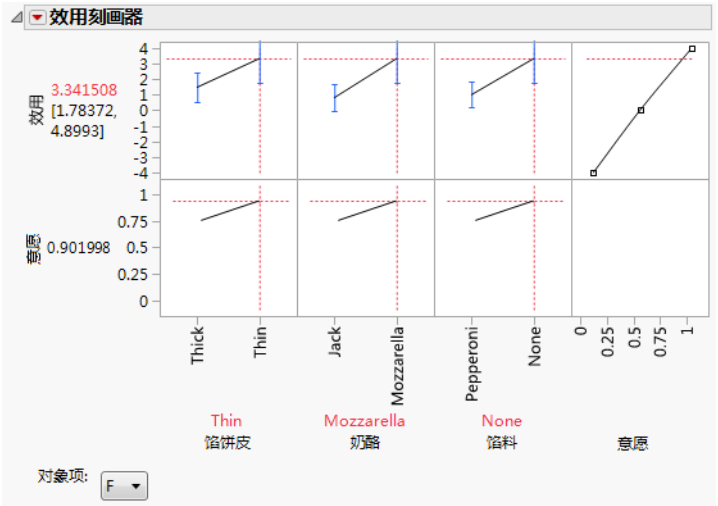
图 4.9 具有意愿函数的效用刻画器



将效用最大化的意愿函数添加到刻画器中。请参见《刻画器指南》。

3. 点击“效用刻画器”红色小三角，然后选择**优化和意愿 > 最大化意愿**。

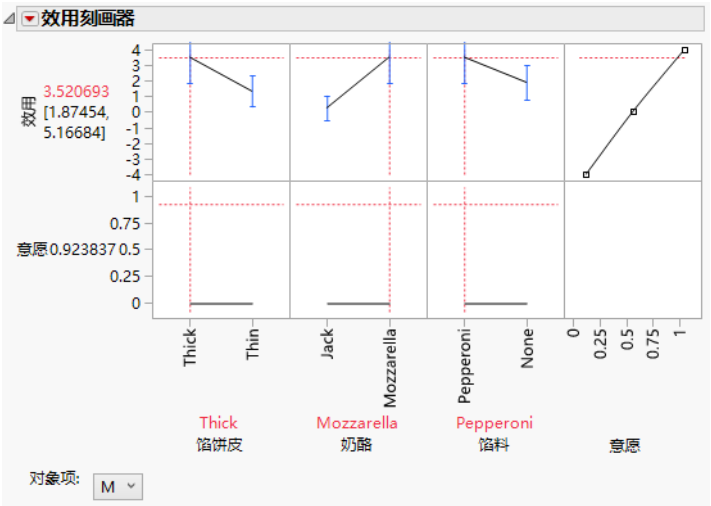
图 4.10 具有针对女性的最优设置的效用刻画器



针对女性的最优设置如下：薄馅饼皮、马苏里拉奶酪且无馅料。

4. 从“对象项”菜单中，选择 M。
5. 点击“效用刻画器”红色小三角，然后选择**最优化和意愿 > 最大化意愿**。

图 4.11 性别设置为男性的效用刻画器



针对男性的最优设置如下：厚馅饼皮、马苏里拉奶酪加意大利辣香肠馅料。

在本例中，了解由性别确定的细分市场的偏好，这样您就可以提供两种比萨饼选择来满足两类客户的需求。

启动“选择”平台

通过选择**分析 > 消费者研究 > 选择**启动 “选择”平台。

“选择”平台的数据可组合到一个数据表中，也可分别位于两个或三个不同数据表中。在“选择”启动窗口中，指定在“数据格式”列表中使用一个还是多个数据表。

一个表，堆叠

对于“一个表，堆叠”格式，数据包含在单个数据表中。为测试对象提供的每一个特征都对应一行，并且有一个指示符指示是否选择了该特征。Pizza Combined No Choice.jmp 样本数据表包含采用单个表格式的选择实验的结果。请参见第 76 页的“‘无选择’的单表格式”。

有关该格式启动窗口的详细信息，请参见第 85 页的“针对‘一个表，堆叠’的启动窗口”。

多个表，交叉引用

对于 “多个表，交叉引用”格式，数据包含在两个或三个不同数据表中。需要一个特征数据表和一个响应数据表。测试对象数据表是可选的。请注意以下几点：

- 特征数据表必须包含一列，其中每个特征都有一个唯一标识符，并且必须包含针对特征水平变量的列。特征标识符在响应数据表中用于标识显示的特征和选定的特征。

- 可选的测试对象数据表必须包含一列，其中每个测试对象都有一个唯一的测试对象标识符，并且必须包含针对测试对象水平变量的列。在响应表中使用测试对象标识符来标识测试对象。

这种格式的启动窗口包含三个部分：特征数据、响应数据和测试对象数据。每个部分都对应不同的数据表。您可以按需展开或折叠每个部分。

Pizza Profiles.jmp、Pizza Responses.jmp 和 Pizza Subjects.jmp 样本数据表包含使用这三个表的选择实验的结果。其中包含一个特征表、一个响应表和一个对象信息表。请参见第 79 页的 “多表格式”。

有关该格式的启动窗口的详细信息，请参见第 87 页的 “针对 “多个表，交叉引用” 的启动窗口”。

针对 “一个表，堆叠” 的启动窗口

图 4.12 针对 “一个表，堆叠” 数据格式的启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

选择数据表 选择或打开包含组合数据的数据表。选择 “其他” 可打开当前尚未打开的文件。

响应指示符 包含指示偏好选择的值所在的列。1 指示偏好的特征，0 指示其他特征。若为响应者提供了 “无偏好” 的选项，则针对选择了无偏好的选择集，输入缺失值。请参见第 86 页的 “允许响应者选择 “无” 或 “无选择””。

对象 ID 用于标识研究参与者的标识符。

选择集 ID 针对给定的偏好决定，为测试对象提供的选择集指示符。

分组 一列，在该列与 “选择集 ID” 列一同使用时，可唯一指定每个选择集。例如，若针对**调查** = A 的某个选择集的**选择集** ID = 1，而针对**调查** = B 的另一个选择集的**选择集** ID = 1，则**调查**应用作一个分组列。

依据 为依据变量的每个水平生成单独的报表。若分配了多个依据变量，则为依据变量水平的每种可能组合都生成一个单独的报表。

构造特征效应 添加从特征中的特性构造的效应。

有关 “构造特征效应” 面板的信息，请参见 《拟合线性模型》。

注意：选择模型注意到连续特征和对象效应的列编码属性。

构造对象效应 （可选） 添加由测试对象相关因子构造的效应。

有关 “构造对象效应” 面板的信息，请参见 《拟合线性模型》。

Firth 偏倚调整估计值 计算修正偏倚的 MLE，以便生成比没有修正偏倚的 MLE 更准确的估计值和检验。这些估计值还可缓解 Logistic 模型中常见的分离问题。有关 Logistic 回归中的分离问题的讨论，请参见 Heinze and Schemper (2002)。

JMP PRO **分层 Bayes** 使用 Bayes 方法估计特定于测试对象的参数。请参见第 93 页的 “Bayes 参数估计值”。

JMP PRO **Bayes 迭代数** （仅在选定 “分层 Bayes” 时适用。）用来估计特定于测试对象的参数的自适应 Bayes 算法的迭代总次数。该数字包括废弃的老化迭代期。老化迭代次数等于启动窗口中指定的 “Bayes 迭代数” 的一半。

允许响应者选择 “无” 或 “无选择” 在模型中为包含缺失值的响应行输入 “无选择指示符”。对于 “一个表，堆叠” 数据格式，“无选择” 行必须在 “响应指示符” 列中包含 （数值）缺失值。该选项显示在启动窗口的底部。

针对 “多个表，交叉引用” 的启动窗口

图 4.13 针对 “多个表，交叉引用” 数据格式的启动窗口

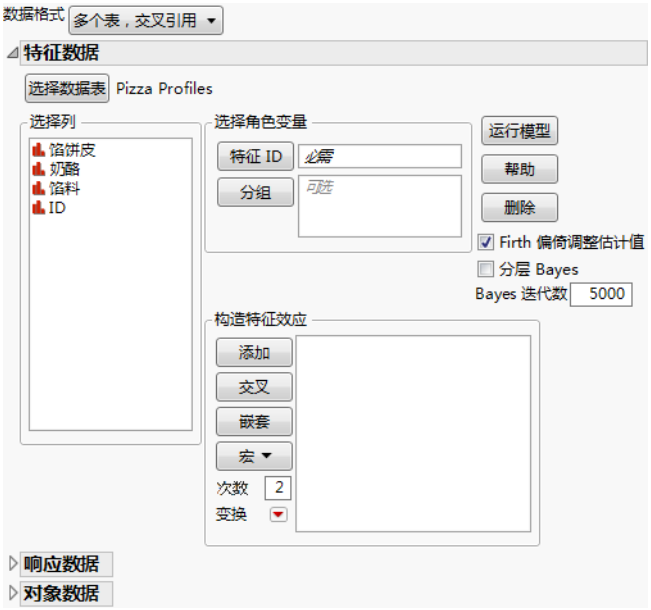


图 4.13 显示了 “多个表” 格式的启动窗口，其中将 Pizza Profiles.jmp 用作特征表。

若采用 “多个表，交叉引用”，启动窗口包含三个部分：

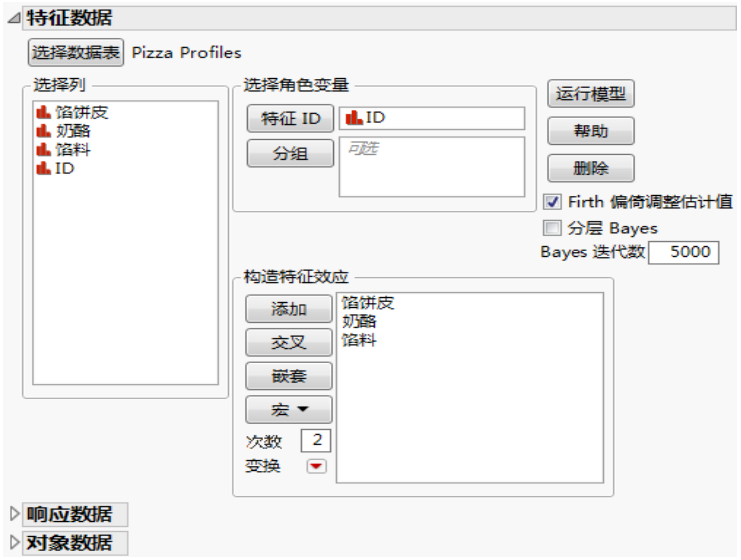
- 第 87 页的 “特征数据”
- 第 89 页的 “响应数据”
- 第 90 页的 “对象数据”

特征数据

特征数据表描述的是与每个选择关联的特性。在数据表中，每个特性列定义一列。每个特征对应一行。同时每个特征在表中有一列唯一的标识符。图 4.14 显示 Pizza Profiles.jmp 数据表和完成的 “特征数据” 面板。

图 4.14 特征数据表和完成的 “特征数据”分级显示项

	馅饼皮	奶酪	馅料	ID
1	Thick	Mozzarella	Pepperoni	ThickOni
2	Thick	Mozzarella	None	ThickElla
3	Thick	Jack	Pepperoni	ThickJackoni
4	Thick	Jack	None	ThickJack
5	Thin	Mozzarella	Pepperoni	TrimOni
6	Thin	Mozzarella	None	Trimella
7	Thin	Jack	Pepperoni	TrimPepperjack
8	Thin	Jack	None	TrimJack



选择数据表 选择或打开包含特征数据的数据表。选择 “其他”可打开当前尚未打开的文件。

特征 ID 每行特性组合 （特征）的标识符。若**特征 ID** 列不能唯一标识特征数据表中的每行，您需要添加**分组**列。添加**分组**列，直到**分组**和**特征 ID** 列的组合能够唯一标识该行或特征。

分组 一列，在该列与 “特征 ID”列一同使用时，可唯一指定每个选择集。例如，若对于**调查** = A **特征 ID** = 1，对于**调查** = B 另有一个**特征 ID** = 1，则**调查**将用作**分组**列。

构造特征效应 添加从特征中的特性构造的效应。

有关 “构造特征效应”面板的信息，请参见 《拟合线性模型》。

注意：选择模型注意到连续特征和对象效应的列编码属性。

Firth 偏倚调整估计值 计算修正偏倚的 MLE，以便生成比没有修正偏倚的 MLE 更准确的估计值和检验。这些估计值还可缓解 Logistic 模型中常见的分离问题。有关 Logistic 回归中的分离问题的讨论，请参见 Heinze and Schemper (2002)。

分层 Bayes 使用 Bayes 方法估计特定于测试对象的参数。请参见第 93 页的“Bayes 参数估计值”。

Bayes 迭代数 （仅在选定“分层 Bayes”时适用。）用来估计对象效应的自适应 Bayes 算法的迭代总次数。该数字包括废弃的老化迭代期。老化迭代次数等于启动窗口中指定的“Bayes 迭代数”的一半。

响应数据

响应数据表包含一个测试对象标识符列、列出每个选择集内特征的特征标识符的列，还有一列包含偏好特征标识符。每个测试对象和选择集都对应一行。当数据包含不止一组选择集时，可使用分组变量来区分各个选择集。图 4.15 显示 Pizza Responses.jmp 数据表和完成的“响应数据”面板。

当数据中包含多个组时，分组变量可用于对选择指标进行分组匹配。

图 4.15 响应数据表和完成的 “响应数据”分级显示项

	试尝者	选择1	选择2	选择
1	1	ThickJack	TrimPepperjack	TrimPepperjack
2	1	TrimPepperjack	ThickElla	ThickElla
3	1	TrimOni	Trimella	TrimOni
4	1	ThickElla	ThickJack	ThickElla
5	2	Trimella	ThickJackoni	Trimella
6	2	TrimJack	ThickElla	ThickElla
7	2	Trimella	TrimPepperjack	Trimella
8	2	TrimPepperjack	TrimOni	TrimOni
9	3	TrimOni	ThickJackoni	TrimOni
10	3	TrimPepperjack	ThickElla	ThickElla
11	3	ThickJackoni	TrimPepperjack	ThickJackoni
12	3	ThickOni	Trimella	ThickOni
13	4	ThickElla	ThickOni	ThickElla
14	4	TrimPepperjack	ThickJack	ThickJack

选择数据表 选择或打开包含响应数据的数据表。选择 “其他”可打开当前尚未打开的文件。

已选择的特征 ID 特征数据表中的特征 ID，该 ID 代表测试对象选择的特征。

分组 一列，在该列与**已选择的特征 ID** 列一同使用时，可唯一指定每个选择集。

特征 ID 选择 可能的特征集的特征 ID。必须至少有两个特征。

对象 ID 用于标识研究参与者的标识符。

频数 包含频数的列。若 *n* 是给定行的频数变量的值，则该行在计算中使用 n 次。若它小于 1 或缺失，则 JMP 不使用它来计算任何分析。

权重 包含数据表中每个观测的权重值的列。仅当权重值大于零时才在分析中包含该权重。

依据 为依据变量的每个水平生成单独的报表。若分配了多个 “依据”变量，则为 “依据”变量水平的每个可能组合生成单独分析。

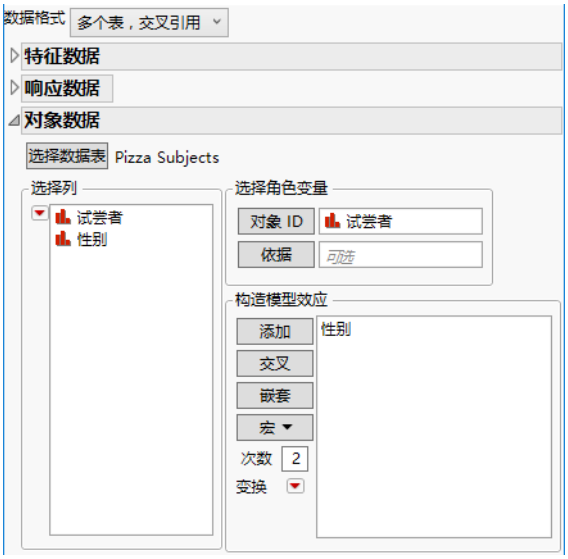
允许响应者选择 “无”或 “无选择” 在模型中为包含缺失值的响应行输入 “无选择指示符”。对于 “多个表，交叉引用”数据格式，“无选择”行必须在响应数据表的 “已选择的特征 ID”列中包含 （分类）缺失值。该选项显示在 “响应数据”面板的底部。

对象数据

对象数据表是可选的，这取决于您是否要对对象效应建模。该表中有一列包含在响应表中使用的测试对象标识符，还有若干列包含测试对象的特性或特征。您可以将对象数据放入响应数据表，但您应在 “对象数据”分级显示项中指定对象效应。图 4.16 显示 Pizza Subjects.jmp 数据表和完成的 “测试对象数据”面板。

图 4.16 对象数据表和完成的 “对象数据”分级显示项

	试尝者	性别
1	1	M
2	2	F
3	3	M
4	4	F



选择数据表 选择或打开包含对象数据的数据表。选择 “其他”可打开当前尚未打开的文件。

对象 ID 测试对象的唯一标识符。

依据 为依据变量的每个水平生成单独的报表。若分配了多个依据变量，则为依据变量水平的每种可能组合都生成一个单独的报表。

构造模型效应 添加由对象数据表中的列构造的效应。

有关 “构造模型效应”面板的信息，请参见 《拟合线性模型》。

“选择模型”报表

- “效应汇总”
- “参数估计值”
- “似然比检验”
- “Bayes 参数估计值”

效应汇总

若您的模型包含不止一个效应而且可以快速计算效应，则显示“效应汇总”报表。（若该报表不显示，请从红色小三角菜单中选择“似然比检验”，同时显示这两个报表。）该报表列出模型估计的效应，并提供这些效应的 LogWorth（或 FDR LogWorth）值的图。该报表还提供一些控件，这些控件可支持您在模型中添加或删除效应。模型拟合报表会基于在“效应汇总”报表中所做的更改自动更新。请参见《拟合线性模型》。

注意：“效应汇总”报表不适用于使用“分层 Bayes”拟合的模型。

“效应汇总”表列

“效应汇总”表包含以下列：

源 列出按 *p* 值升序排序的模型效应。

LogWorth 显示每个模型效应的 LogWorth，该值定义为 $-\log_{10}(p \text{ 值})$。该变换调整 *p* 值以提供适用于绘图的尺度。超过 2 的值在 0.01 水平下显著 （因为 $-\log_{10}(0.01) = 2$）。

FDR LogWorth 显示每个模型效应的假发现率 LogWorth，该值定义为 $-\log_{10}(\text{FDR P 值})$。这是用于标绘和评估显著性的最佳统计量。选中 FDR 复选框可将 LogWorth 列替换为 FDR LogWorth 列。

条形图 显示 LogWorth（或 FDR LogWorth）值的条形图。该图在整数值处显示垂直虚线，在 2 处显示蓝色参考线。

p 值 显示每个模型效应的 *p* 值。这是与“似然比检验”报表中显示的显著性检验对应的 *p* 值。

FDR p 值 显示使用 Benjamini-Hochberg 方法计算的每个模型效应的假发现率 *p* 值。该方法调整 *p* 值以控制多重检验的假发现率。选中 FDR 复选框可将 **p 值**列替换为 FDR **p 值**列。

有关 FDR 校正的详细信息，请参见 Benjamini and Hochberg (1995)。有关假发现率的详细信息，请参见 《预测和专业建模》或 Westfall et al.(2011)。

“效应汇总”表选项

汇总表下的选项支持您添加和删除效应：

删除 从模型中删除选定的效应。要删除一个或多个效应，请选择与效应对应的行并点击 “删除”按钮。

添加特征效应 打开一个面板，其中包含针对 “一个表，堆叠”数据格式的数据表中所有列的列表，以及针对“多个表，交叉引用”数据格式的特征数据表中的列的列表。选择您要添加到模型中的列，然后点击列选择列表下的 “添加”将这些列添加到模型中。点击 “关闭”关闭该面板。

添加对象效应 打开一个面板，其中包含针对 “一个表，堆叠”数据格式的数据表中所有列的列表，以及针对“多个表，交叉引用”数据格式的对象数据表中的列的列表。选择您要添加

到模型中的列，然后点击列选择列表下的 “添加”将这些列添加到模型中。点击 “关闭”关闭该面板。

参数估计值

“参数估计值”报表提供与“项”列中所列的效应关联的效用系数的估计值和标准误差。与特性关联的系数有时称为**成分效用值**。在启动窗口中选定“Firth 偏倚调整估计值”选项后，参数估计值基于 Firth 修正偏倚的最大似然估计量。这些估计值要比没有修正偏倚的 MLE 更准确。有关效用的详细信息，请参见第 123 页的“效用和概率”。

比较准则

AICc（校正的 Akaike 信息准则）、BIC（Bayes 信息准则）、–2对数似然和 –2Firth 对数似然拟合统计量作为报表的一部分显示，并可用来比较模型。请参见《拟合线性模型》。

在启动窗口中选定“Firth 偏倚调整估计值”选项时，报表中将包含 –2Firth 对数似然拟合统计量。请注意，默认选中该选项。使用或不使用 Firth 偏倚调整估计值不会影响 AICc 得分或 –2 对数似然结果。

注意：对于上述每一个统计量，较小的值指示较好的拟合。

似然比检验

若模型在 5 秒内拟合，则默认显示“似然比检验”报表。若该报表不显示，可以从“选择模型”红色小三角菜单中选择 “似然比检验”选项。该报表提供以下信息：

源 列出模型中的效应。

似然比卡方 针对相应效应检验的似然比卡方统计量的值。

自由度 卡方检验的自由度。

概率 > 卡方 卡方检验的 *p* 值。

条形图 显示似然比卡方值的条形图。

JMP PRO Bayes 参数估计值

（仅可用于分层 Bayes。）“Bayes 参数估计值”报表提供模型效应的结果。估计值基于“分层 Bayes”拟合，该拟合将测试对象级别的协变量整合到似然函数中并且直接估计它们对于参数的效应。使用 Bayes 过程结合 Metropolis-Hastings 算法估计测试对象级别的协变量。请参见 Train (2001)。为每个模型效应计算后验均值和方差。该算法还会提供模型效应参数的特定于测试对象的估计值。请参见第 96 页的“保存对象估计值”。

在估计过程中，向每个个体分配他/她自己的参数估计值向量，基本上将估计值视为随机效应和协变量。个体的系数向量假定来自具有任意均值和协方差矩阵的多元正态分布。在给定选择集

中的特性时，给定测试对象的效用参数的似然函数基于选择集内每个测试对象的偏好的多项式 Logit 模型。给定测试对象的系数向量的先验分布呈正态分布，其均值为 0，对角协方差矩阵中每个测试对象的方差都相同。协方差矩阵假定来自逆 Wishart 分布，其尺度矩阵为具有相同对角线元素的对角矩阵。

对于每个测试对象，链最初的老化迭代次数将被废弃。默认情况下，该次数等于启动窗口中指定的“Bayes 迭代数”的一半。

图 4.17 “Bayes 参数估计值”报表

选择模型: 选择

Bayes 参数估计值

项	后验均值	后验标准差	对象标准差
馅饼皮[Thick]	0.8816482	2.31954708	10.69219592
奶酪[Jack]	-23.7817993	12.95816152	8.32854831
馅料[Pepperoni]	-3.9113860	3.20179657	8.91276668

迭代总次数	5000
老化迭代次数	2500
响应者数目	32
老化后的平均对数似然	-5.281536

项 模型项。

后验均值 项系数的参数估计值。对于老化期之后的每次迭代，计算特定于测试对象的系数估计值的均值。后验均值是这些均值的平均值。

提示：选择红色小三角选项“保存 Bayes 链”，查看每次迭代的单个估计值。

后验标准差 老化过程后经过几次迭代，特定于测试对象的估计值的均值的标准差。

对象标准差 特定于测试对象的估计值的标准差。

提示：选择红色小三角选项“保存测试对象估计值”以查看单个估计值。

迭代总次数 执行的迭代总次数，包括老化期。

老化迭代次数 老化迭代次数。该次数等于启动窗口中指定的“Bayes 迭代数”的一半。

响应者数目 测试对象数。

老化后的平均对数似然 基于在老化期后获取的值计算的对数似然函数的平均值。

“选择”平台选项

“选择模型”红色小三角菜单包含以下选项。

注意：使用“分层 Bayes”时，测试对象级别的估计值基于 Monte Carlo 抽样。出于此原因，每次运行以下选项所获取的结果都不相同。

似然比检验 请参见第 93 页的 “似然比检验”。

显示 MLE 参数估计值 （可用于分层 Bayes）显示模型项系数的非 Firth 最大似然估计值和标准误差。这些估计值用作分层 Bayes 算法的起始值。

联合因子检验 （不可用于分层 Bayes）通过对包含该因子的所有效应构造似然比检验来检验模型中的每个因子。有关联合因子检验的详细信息，请参见 《拟合线性模型》。

置信区间 （不可用于分层 Bayes）显示或隐藏 “参数估计值”报表中每个参数的置信区间。

置信限 （可用于分层 Bayes）为 “Bayes 参数估计值”报表中的每个参数显示置信限。基于后验分布的 2.5 和 97.5 分位数构造限值。

估计值的相关性 若未选定 “分层 Bayes”，则显示最大似然参数估计值之间的相关性。

对于分层 Bayes，则显示参数估计值后验均值的相关性矩阵。基于老化后的迭代次数计算相关性。像处理数据表中的列那样来处理老化后每次迭代的后验均值。通过计算这些列的相关性矩阵来获取 “估计值的相关性”表。

边缘效应 显示或隐藏模型中的每个主效应的边缘概率和边缘效用。边缘概率是指在所有其他特性设置为其均值或默认水平下，一个人选择特性 A 而不是特性 B 的概率。

在图 4.18 中，相对于采用蒙特里杰克奶酪的同类比萨饼的测试对象倾向于选择马苏里拉奶酪、厚馅饼皮加意大利辣香肠馅料的比萨饼的概率为 0.9470。

图 4.18 边缘效应的示例

效用刻画器 显示或隐藏不同因子设置下的预测效用。该效用值通过线性模型预测得出。有关 “效用刻画器”的示例，请参见第 82 页的 “查找最优特征”。有关效用的详细信息，请参见第 123 页的 “效用和概率”。有关 “效用刻画器”选项的详细信息，请参见 《刻画器指南》。

概率刻画器 支持您在若干潜在产品之间比较选择概率。该预测概率定义如下：

$$(\exp(U))/(\exp(U)+\exp(U_b))$$

其中，U 为当前设置的效用，U_b 为基线设置的效用。这意味着基线设置的概率为 0.5。请参见第 123 页的 “效用和概率”。

有关 “概率刻画器”的示例，请参见第 103 页的 “与基线比较”。有关 “概率刻画器”选项的详细信息，请参见 《刻画器指南》。

多项选择刻画器 提供您指定的概率刻画器的数目。这样您就可以将每个刻画器设置为给定特征的设置，从而方便您比较相对于其他特征，测试对象倾向于选择给定特征的概率。有关使用 “多项选择刻画器”的示例，请参见第 105 页的 “多项选择的比较”。有关 “多项选择刻画器”选项的详细信息，请参见 《刻画器指南》。

比较 执行两个特定备择选择特征之间的比较。支持您选择要比较的因子和值。您可以比较特定的配置：包括通过选中**任意**复选框来比较左侧或右侧的所有设置。若有对象效应，您可以选择要比较的对象效应水平。使用“任意”并不会对所有不同特征的组合进行比较，而是所有两两特征的组合进行比较，一次比较一个特征，对于其他因子维持左侧显示的设置。

图 4.19 “效用比较”窗口

支付意愿 要求您的模型包含连续的价格列。计算客户愿意为一个新功能支付的超过基准功能成本的最大价格上涨（下降）。使用针对每个背景设置的基线设置计算结果。

保存效用公式 当分析针对多个数据表时，创建包含用于效用的公式列的新数据表。新数据表的行是测试对象特征和被测特征的组合，列是被测特征和测试对象效应列。当分析针对一个数据表时，将添加新的“效用公式”列。

按对象保存梯度 （不可用于分层 Bayes。）构建一个新表，每个测试对象对应一行，其中包含测试对象水平的每个参数的似然函数的平均步长（Hessian 比例梯度）。这对应于使用 Lagrangian 乘数检验将该对象与其余对象分离出来。以后可以使用内置脚本对这些值聚类，以表明数据中存在的独特的市场细分。请参见第 123 页的“梯度”。有关示例，请参见第 107 页的“细分示例”。

JMP PRO **保存对象估计值** （可用于分层 Bayes。）创建一个表，其中每行都包含每个效应的特定于测试对象的参数估计值。每个效应的特定于测试对象的参数效应的分布以“Bayes 参数估计值”报表中给定项的估计值为中心。“对象接受率”提供 Metropolis-Hastings 步骤期间得到新参数估计值的接受率。通常，接受率 0.20 被视为良好。请参见第 93 页的“Bayes 参数估计值”。

JMP PRO **保存 Bayes 链** （可用于分层 Bayes。）创建一个表，其中提供关于计算特定于测试对象的 Bayes 估计时使用的迭代链的信息。请参见第 98 页的“保存 Bayes 链”。

模型对话框 显示“选择”启动窗口，可以使用该窗口来修改和重新拟合模型。您可以指定新数据集、新 ID 和新模型效应。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为“依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定“依据”变量时才可用。

支付意愿

支付意愿一词是指客户愿意为新功能支付的价格，计算该价格以匹配客户的基准功能效用。例如，假定客户愿意为配有 40 GB 硬盘驱动器的计算机支付 $1,000。通过将硬盘驱动器功能设置为 80 GB，再求解提供与 $1000 40 GB 硬盘驱动器相同效用的价格，即可计算 80 GB 硬盘驱动器的支付意愿。

“支付意愿”启动窗口选项

选择 “支付意愿”选项后即显示 “支付意愿”启动窗口。通过在运行 Laptop Profile.jmp 中的**选择**数据表脚本所生成的报表中选择 “支付意愿”选项，即可显示图 4.20 中的启动窗口。

因子 分析的变量。这些可以是产品功能或特定于测试对象的特性。

基线 每个因子的基线设置。若因子为分类因子，则从列表中选择基线值。若因子为数值因子，则输入基线值。

角色 因子的类型。

- **特征因子** 实验中您想要定价的产品或服务功能。
- **价格因子** 实验中的价格因子。价格因子必须为连续型，而且对于每个支付意愿分析都只能指定一个价格因子。
- **背景常数** 您希望在支付意愿计算中保持不变的一个因子。通常，它们是特定于测试对象的变量。
- **背景变量** 针对该变量的每个因子水平，您希望在支付意愿的计算中保持不变。通常，它们是测试对象级别的因子。将测试对象因子指定为背景变量而不是背景常数，这样可为所有变量水平提供支付意愿估计值。

在报表表中包括基线设置 向 “支付意愿”报表添加价格变化为零的基线设置。

提示：若生成输出表，使用该选项可显示所有基线设置和特性设置。

同时显示输出数据表 创建包含 “支付意愿”报表的数据表。

图 4.20 “支付意愿”启动窗口

一旦完成首次支付意愿计算，该平台会记住您选定的基线值和指定的角色。这样您就可以进行多次支付意愿比较，而不必重新输入基线信息。若没有称为“价格”的因子，但在分析中使用了连续因子，则将该连续因子自动指定为“支付意愿”窗口中的“价格”因子。常见的传统意义上的成本变量（非价格）包括差旅时间或距离等。

“支付意愿”报表

“支付意愿”报表显示每个因子的基线值以及基线效用值。对于每个因子，该报表都显示特征设置、估计的价格变化和新价格。若没有交互作用或二阶效应，则还会显示标准误差和置信区间。使用 delta 方法计算这些统计量。

图 4.21 “支付意愿”报表

支付意愿

因子	基线值
硬盘	40 GB
速度	1.5 GHz
电池寿命	4 hours
价格	1000

基线效用
-3.4736

因子	特征设置	价格变动	标准误差	95% 下限	95% 上限	新价格
硬盘	80 GB	$959.67	381.743	$211.47	$1,707.87	$1,959.67
速度	2.0 GHz	$496.86	218.515	$68.58	$925.14	$1,496.86
电池寿命	6 hours	$387.88	191.088	$13.35	$762.40	$1,387.88

价格变动的标准差按 Delta 方法计算。

JMP PRO 保存 Bayes 链

（可用于使用分层 Bayes 拟合的模型。）您可以使用“Bayes 链”数据表确定估计值是否已经稳定。创建的表包含的行数等于 “Bayes 迭代数”（在启动窗口中指定）加 1。第一行 （迭代 1）

提供起始值。后续行按顺序显示迭代结果。该表有一列用于迭代计数、模型对数似然，并且包含与每个模型效应对应的列：

迭代 提供迭代次数，第一行显示起始值。

对数似然 这次迭代的模型对数似然。您可以针对迭代绘制对数似然，以便查看老化期和调节期的行为。

< 模型效应 > 的自适应 Sigma 提供相应效应的逆 Wishart 分布尺度矩阵的对角线元素的平方根估计值。

< 模型效应 > 的接受率 提供相应效应的样本接受率。

< 模型效应 > 的均值 提供相应效应的估计均值。

< 模型效应 > 的方差 提供相应效应的估计方差。

更多示例

- “示例：做出设计决策”
- “细分示例”
- “使用“选择”平台进行 Logistic 回归的示例”
- “配对的案例对照研究的 Logistic 回归示例”
- “将数据转换为两个分析表的示例”
- “将数据转换为一个分析表的示例”

示例：做出设计决策

您可以使用“选择建模”平台来确定产品特性的相对重要性。即使对于特定的产品来讲，我们知道哪些特性对于消费者来讲是重要的，而有关这些特性的偏好取舍的信息也可能是未知的。通过获取这些信息，市场研究人员或产品设计者可以在考虑消费者偏好的同时，权衡产品的不同特性，从而得到最优的产品。本例演示了该方法在产品设计上的优势。

已知对于笔记本电脑设计很重要的四个特性：硬盘大小、处理器速度、电池寿命和销售价格。接下来的示例用已经收集好的数据来确定四个笔记本电脑特性中（**硬盘**、**速度**、**电池寿命**和**价格**）哪一个是最重要的。同时我们还将通过该示例评估是否有**性别**或**职位**效应与这些特性相关。

本例在四个章节中进行了说明：

- 第 100 页的“完成启动窗口”
- 第 102 页的“分析模型”
- 第 103 页的“与基线比较”
- 第 105 页的“多项选择的比较”

完成启动窗口

1. 选择**帮助** > **样本数据库**，然后打开 Laptop Runs.jmp。

注意：若您不想执行本节中的手动步骤，请点击脚本**选择 - 性别**旁边的绿色小三角运行该模型，然后转至第 102 页的“分析模型”。

2. 点击**打开特征表和对象表**脚本旁边的绿色小三角。

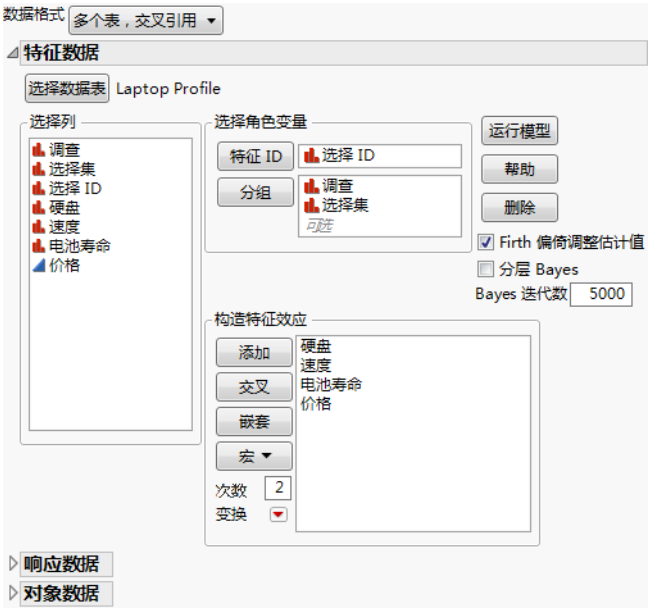
   该脚本可打开 Laptop Profile.jmp 和 Laptop Subjects.jmp 数据表。

3. 选择**分析** > **消费者研究** > **选择**。

   注意：可以从三个打开的数据表中的任一个执行该操作。

4. 从“数据格式”列表中，选择**多个表，交叉引用**。
5. 点击“特征数据”下的**选择数据表**，然后选择 Laptop Profile。选择**选择 ID** 并点击**特征 ID**。
6. 选择**硬盘**、**速度**、**电池寿命**和**价格**，然后点击**添加**。
7. 选择**调查**和**选择集**，然后点击**分组**。

图 4.22 针对笔记本电脑研究的“特征数据”窗口



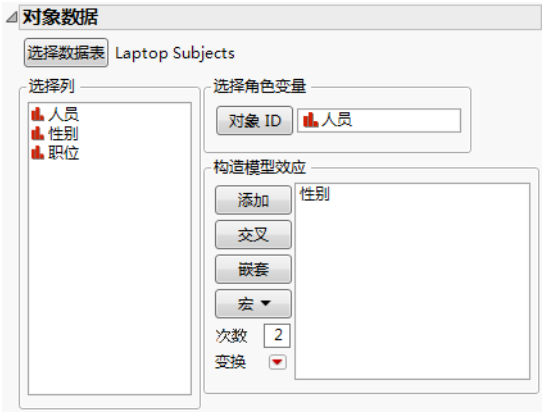
8. 打开**响应数据**分级显示项。
9. 从**选择数据表**列表中，选择 Laptop Runs。
10. 完成“响应数据”表：
    - 选择**响应**，然后点击**已选择的特征 ID**。

- 选择**选择1** 和**选择2**，然后点击**特征 ID 选择**。
- 选择**调查**和**选择集**，然后点击**分组**。
- 选择**人员**，然后点击**对象 ID**。

图 4.23 针对笔记本电脑研究的 “响应数据”窗口

11. 打开**对象数据**分级显示项。
12. 从**选择数据表**列表中，选择 Laptop Subjects。
13. 选择**人员**，然后点击**对象 ID**。
14. 选择**性别**并点击**添加**。

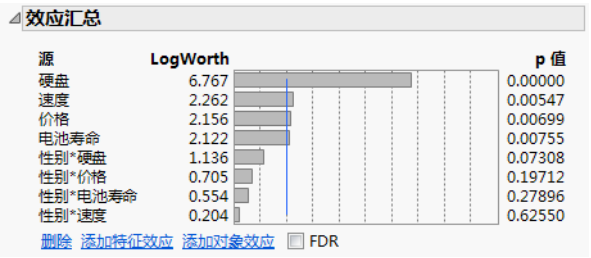
图 4.24 针对笔记本电脑研究的 “对象数据”窗口



分析模型

1. 点击**运行模型**。

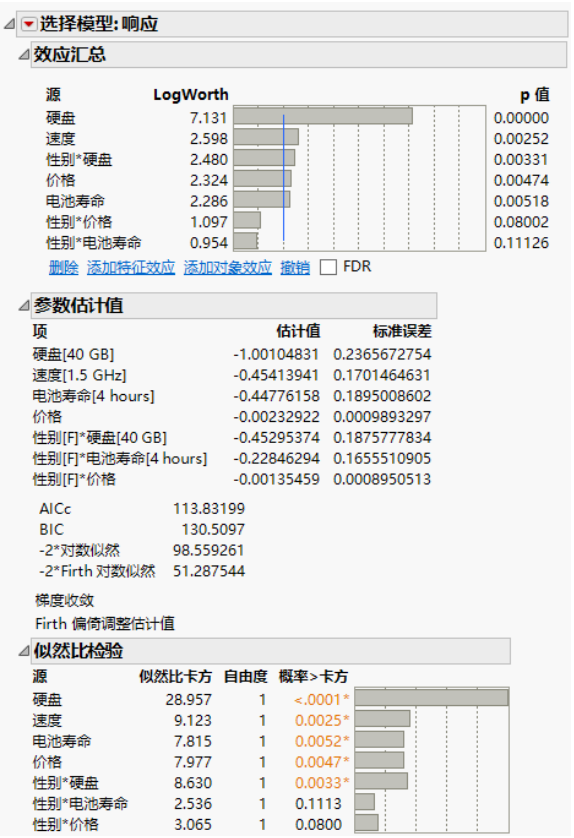
图 4.25 笔记本电脑效应汇总



“效应汇总”报表显示**硬盘**是最显著的效应。您可以通过删除 *p* 值大于 0.15 的项精简该模型。此过程要求一次删除一项。在此，**性别 * 速度**是显著性最低的效应， *p* 值为 0.625。

2. 在“效应汇总”报表中，选择**性别 * 速度**并点击**删除**。

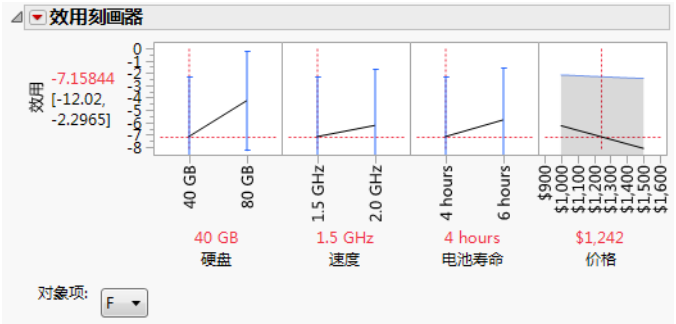
图 4.26 笔记本电脑结果



一旦从模型中删除了**性别 * 速度**，所有效应都具有小于或等于 0.15 的 *p* 值。因此，您可以将该模型用作您的最终模型。

3. 点击 “选择模型 : 响应” 红色小三角并选择**效用刻画器**。

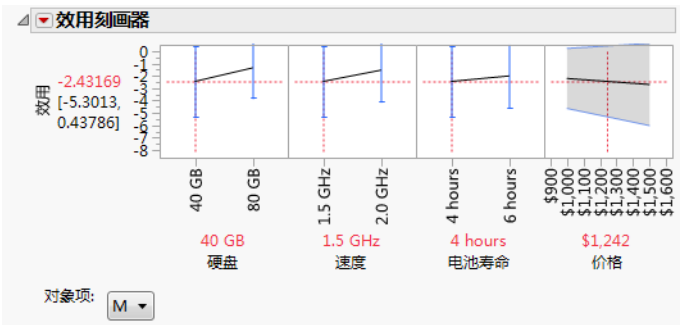
图 4.27 针对女性的笔记本电脑刻画器结果



提示：若您的效用刻画器未能如图 4.27 所示，请点击 “效用刻画器” 红色小三角并选择**外观 > 适应 Y 轴**。

4. 从 “对象项” 旁边的列表中，选择 M。

图 4.28 开发过程中针对男性的笔记本电脑刻画器结果



性别和**硬盘**之间的交互作用效应高度显著，*p* 值为 0.0033（第 102 页上的图 4.26）。在 “效用刻画器” 中，同时检查**性别**两个水平的**硬盘**的斜率。您可以看到女性的斜率比男性的斜率陡峭。

与基线比较

假定您正在开发一种新产品。您想要知道相对于老产品或竞争对手的产品，客户更倾向于选择该新产品的可能性。使用 “概率刻画器” 将特征与基线特征作比较。

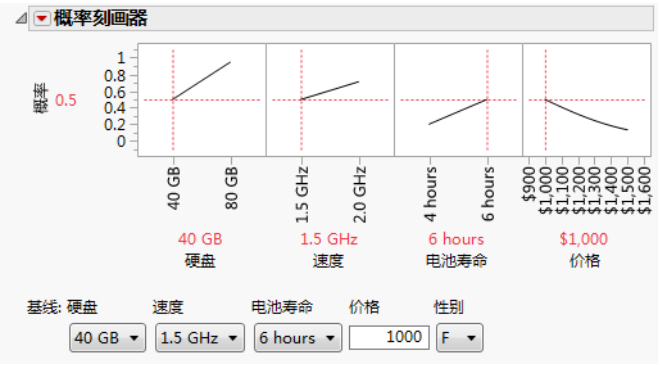
在该示例中，您的公司当前正在生产的笔记本电脑规格为：40 GB 硬盘，1.5 GHz 处理器，电池寿命为 6 小时，价格为 1,000 美元。您想通过尽可能少地调整产品特性来改进产品使其更受客户

欢迎。您将当前产品配置设置为基线。JMP 调整了概率，将偏好该基线配置的概率调整为 0.5。然后您将其他配置的概率与该基线概率作比较。

1. 执行以下操作之一：
   - 执行第 100 页的“完成启动窗口”中的步骤。然后完成第 102 页的“分析模型”中的第 1 步和第 2 步。
   - 在 Laptop Runs.jmp 样本数据表中，点击**选择简化模型**脚本旁边的绿色小三角。
2. 点击“选择模型 : 响应”红色小三角并选择**概率刻画器**。

   请注意，“概率刻画器”的设置是**性别** = F。您可以随后进行更改。
3. 使用该刻画器下面的菜单和文本框，在“基线”区域中，将“基线”设置指定为 40 GB、1.5 GHz、6 小时和 1000。
4. 接下来在“概率刻画器”中做同样的设置。要将“价格”设置为 $1000，请点击最右侧刻画器单元格下方“价格”之上的 $1242，然后键入 1000。接着在文本框外点击。

图 4.29 带有“价格”文本输入区的概率刻画器

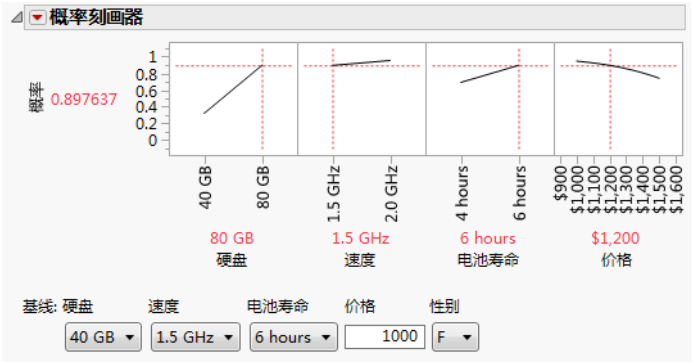


该配置的概率为 0.5。

5. 在“概率刻画器”中，移动**硬盘**滑块至 80 GB。

   请注意，进行这种更改后，**价格**增长对选择概率的影响相对不那么敏感了。
6. 点击刻画器中的**价格**单元格上方的 $1000 标签，键入 $1,200，然后在文本框外点击。

图 4.30 考虑基准产品效应的笔记本电脑的概率刻画器结果



“硬盘”大小从 40 GB 增长到 80 GB 且价格增长到 $1200，与女性的偏好概率会相应地从 0.50 增长到 0.90。将基准中的**性别**效应更改为 M。偏好概率为 0.71。

多项选择的比较

使用 “多项选择刻画器”比较产品特征。

- 您当前在生产一种低端笔记本电脑，它的硬盘小、处理器慢且电池寿命短。定价为 $1000。
- A 公司生产的产品具有处理器速度快、电池寿命长且价格 ($1200) 合理等特性。
- B 公司生产的产品硬盘空间最大、速度最快、但是价格高 ($1500) 且电池寿命短。

您想通过提高某一方面的性能并提升价格来赢得市场份额。

1. 执行以下操作之一：
   - 执行第 100 页的“完成启动窗口”中的步骤。然后完成第 102 页的“分析模型”中的第 1 步和第 2 步。
   - 在 Laptop Runs.jmp 样本数据表中，点击**选择简化模型**脚本旁边的绿色小三角。
2. 点击 “选择模型 : 响应”红色小三角并选择**多项选择刻画器**。

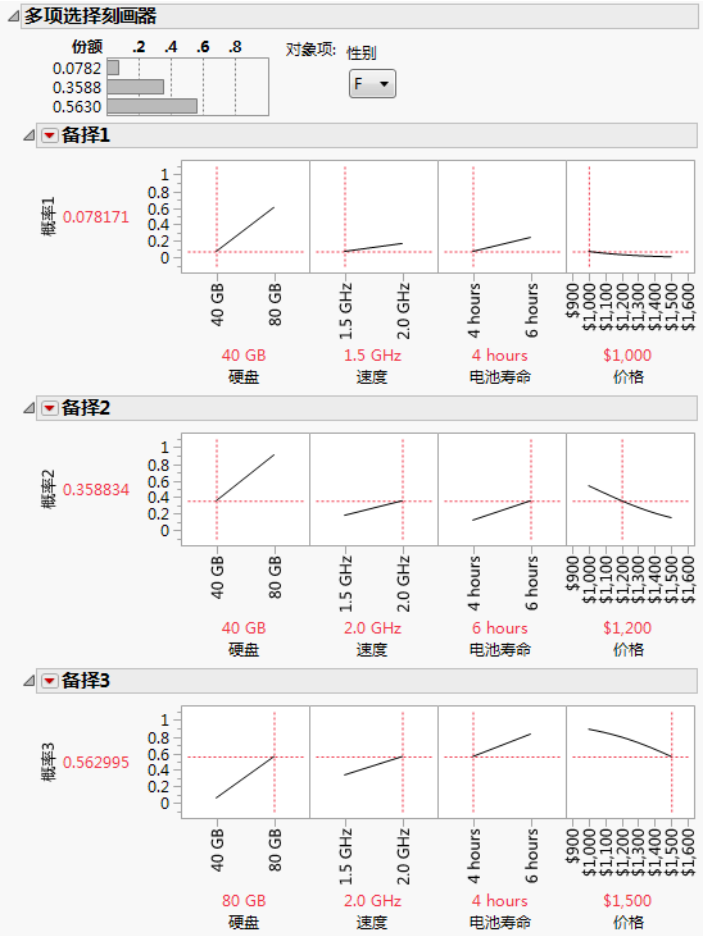
   将显示一个窗口，询问要刻画的备择选择数。接受默认数字 3。
3. 点击**确定**。

   将显示三个备择刻画器。请注意，刻画器设置为 “性别 = F”。

   每个刻画器中因子都设置为其默认值。备择 1 表示您要开发的产品。备择 2 表示 A 公司的产品。备择 3 表示 B 公司的产品。
4. 对于备择 1，将**硬盘**设置为 40 GB，**速度**设置为 1.5 GHz，**电池寿命**设置为 4 小时，**价格**设置为 $1,000。
5. 对于备择 2，将**硬盘**设置为 40 GB，**速度**设置为 2.0 GHz，**电池寿命**设置为 6 小时，**价格**设置为 $1,200。

6. 对于备择 3，将**硬盘**设置为 80 GB，**速度**设置为 2.0 GHz，**电池寿命**设置为 4 小时，**价格**设置为 $1,500。

图 4.31 针对女性的 “多项选择刻画器”

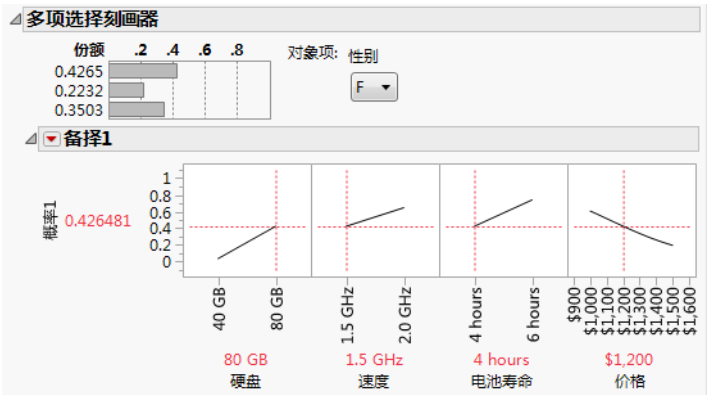


您可以看到 B 公司占有最大的份额 0.5630。很显然，以您公司现有的设置，只会有很少的女性买您的产品。

您想通过升级您公司笔记本电脑在某方面的性能并提高价格来增加市场份额。从备择 1 中硬盘特征的直线斜率可以看出增加硬盘空间最有助于提高市场份额。

7. 对于备择 1，将**硬盘**设置为 80 GB，**价格**设置为 $1,200。

图 4.32 改进笔记本电脑之后的 “多项选择刻画器”



通过增加硬盘空间，同时提高笔记本电脑的价格，预计您公司可以获得的女性市场份额约为 43%。该份额超过了 B 公司高性能笔记本电脑的份额，远远高于图 4.31 中所示的初始低端设置下的市场份额。

探索可增加男性市场份额的设置。若同时提高 “硬盘” 大小和 “速度”，您可以获得男性市场份额中的 44%。

细分示例

在本例中，您尝试确定不同比萨饼偏好的细分市场。

要查看如何完成本示例的启动窗口，请参见示例第 79 页的 “多表格式” 中的第 1 步到第 15 步。否则，遵照以下说明操作。

定义聚类

1. 选择**帮助** > **样本数据库**，然后打开 Pizza Responses.jmp。
2. 点击**选择**脚本旁边的绿色小三角。
3. 点击 “选择模型 : 选择” 红色小三角并选择**按对象保存梯度**。

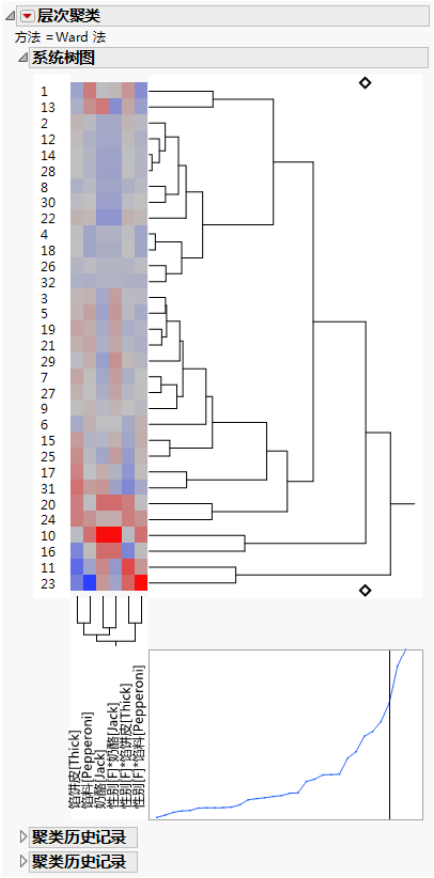
   将显示一个数据表，其中包含为每个主效应和测试对象的交互作用保存的梯度力。

图 4.33 比萨饼数据按测试对象划分的梯度 （部分视图）

	试尝者	馅饼皮[Thick]	奶酪[Jack]	馅料[Pepperoni]	性别[F]*馅饼皮[Thick]	性别[F]*奶酪[Jack]	性别[F]*馅料[Pepperoni]
1	1	-0.00959	-0.00168	0.014876	0.009585	0.001685	-0.01488
2	2	0.002373	-0.00758	-0.00239	0.002373	-0.00758	-0.00239
3	3	0.002129	-0.0079	0.003031	-0.00213	0.007899	-0.00303
4	4	-0.00106	-0.00485	-0.00901	-0.00106	-0.00485	-0.00901
5	5	0.002828	-0.00945	0.00725	-0.00283	0.009453	-0.00725
6	6	-0.0073	-0.00089	0.003761	-0.0073	-0.00089	0.003761

4. 点击**层次聚类**脚本旁边的绿色小三角。

图 4.34 比萨饼数据的测试对象聚类的系统树图



该脚本针对梯度表中除**对象**之外的所有列运行层次聚类分析。点击两个菱形中的任一个，可以看到这些行已分成三个聚类。

5. 点击“层次聚类”红色小三角菜单并选择**保存聚类**。

   名为**聚类**的新列将添加到包含梯度的数据表中。为每个测试对象指定了一个**聚类**值，该类的所有测试对象都具有相似梯度力。有关其他“层次聚类”选项的讨论，请参见《多元方法》。

   您可以删除梯度列，因为它们仅用于获取聚类。

6. 选择除了**试尝者**和**聚类**之外的其他所有列。右击所选列，然后选择**删除列**。

7. 点击**重新合并数据**脚本旁边的绿色小三角 （图 4.33）。

   聚类信息合并到对象数据表中。测试对象数据表中的列现在为**试尝者**、**性别**和**聚类**。

图 4.35 具有聚类列的对象数据

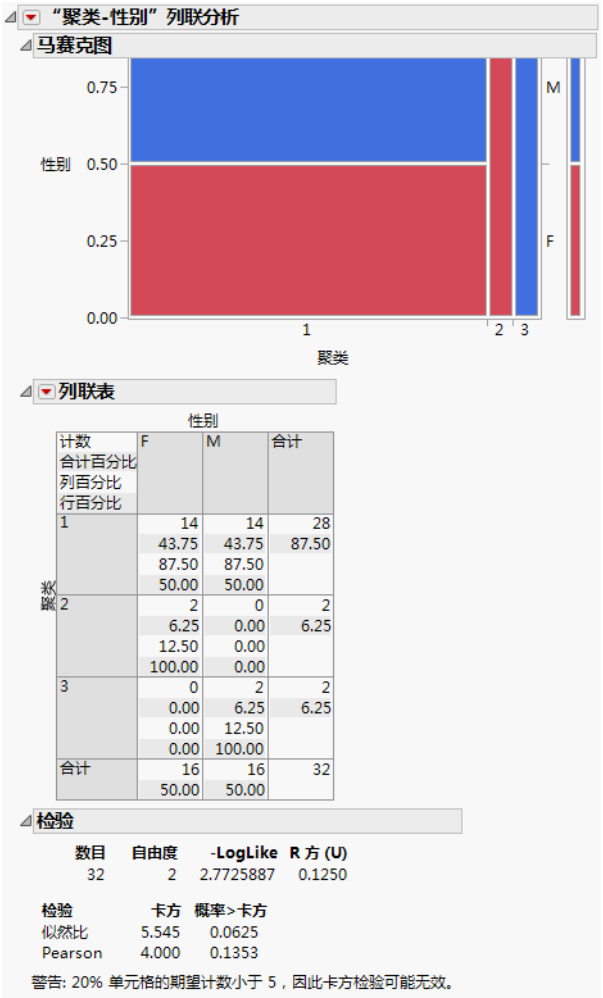
	试尝者	馅饼皮[Thick]	奶酪[Jack]	馅料[Pepperoni]	性别[F]*馅饼皮[Thick]	性别[F]*奶酪[Jack]	性别[F]*馅料[Pepperoni]	聚类
1	1	-0.00959	-0.00168	0.014876	0.009585	0.001685	-0.01488	1
2	2	0.002373	-0.00758	-0.00239	0.002373	-0.00758	-0.00239	1
3	3	0.002129	-0.0079	0.003031	-0.00213	0.007899	-0.00303	1
4	4	-0.00106	-0.00485	-0.00901	-0.00106	-0.00485	-0.00901	1
5	5	0.002828	-0.00945	0.00725	-0.00283	0.009453	-0.00725	1
6	6	-0.0073	-0.00089	0.003761	-0.0073	-0.00089	0.003761	1
7	7	0.006003	-0.00815	0.000308	-0.006	0.008151	-0.00031	1
8	8	-0.0055	-0.00887	-0.00274	-0.0055	-0.00887	-0.00274	1
9	9	0.000438	-0.00271	0.002402	-0.00044	0.00271	-0.0024	1
10	10	-0.00217	0.032641	0.017043	-0.00217	0.032641	0.017043	2

该表现在可以用于进一步分析。

探索聚类

1. 点击列面板中**聚类**变量左侧的图标，然后选择**名义型**。
2. 选择**分析 > 以 X 拟合 Y**。
3. 选择**性别**，然后点击 **Y，响应**。
4. 选择**聚类**，然后点击 **X，因子**。
5. 点击**确定**。

图 4.36 “聚类 - 性别”列联分析



观察聚类中的模式：

- 聚类 1 在男性和女性之间平分
- 聚类 2 仅包含女性
- 聚类 3 仅包含男性

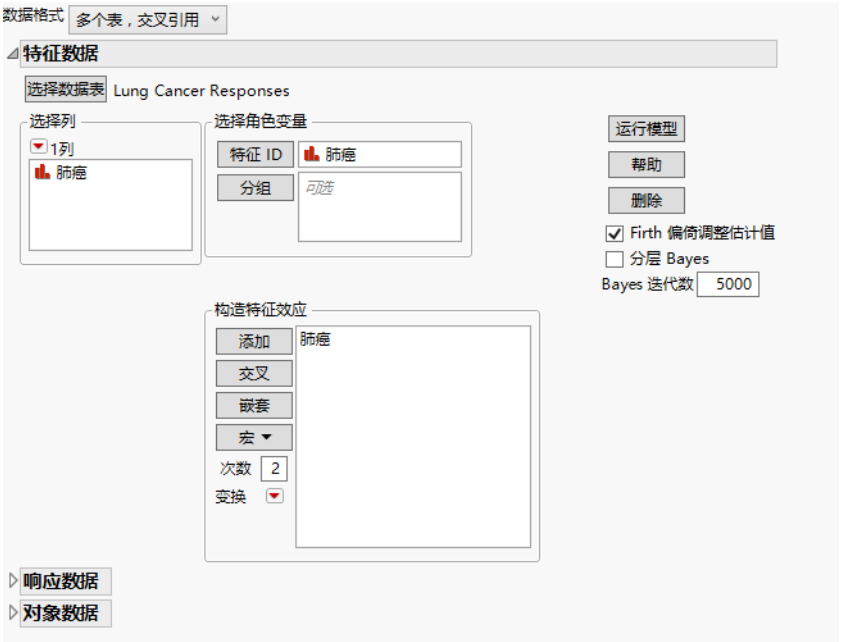
若需要，您现在可以通过添加**聚类**变量来重新拟合并分析该模型。

使用“选择”平台进行 Logistic 回归的示例

使用“选择”平台

1. 选择**帮助** > **样本数据库**，然后打开 Lung Cancer Responses.jmp 和 Lung Cancer Choice.jmp。请注意：Lung Cancer Responses.jmp 仅有一列（**肺癌**）和两行（Cancer 和 NoCancer）。
2. 选择**分析** > **消费者研究** > **选择**。
3. 从“数据格式”旁边的列表中，选择**多个表，交叉引用**。
4. 点击**选择数据表**，选择 Lung Cancer Responses，然后点击**确定**。
5. 选择**肺癌**，然后点击**特征 ID**。
6. 选择**肺癌**，然后点击**添加**。
7. 取消选中“Firth 偏倚调整估计值”框。

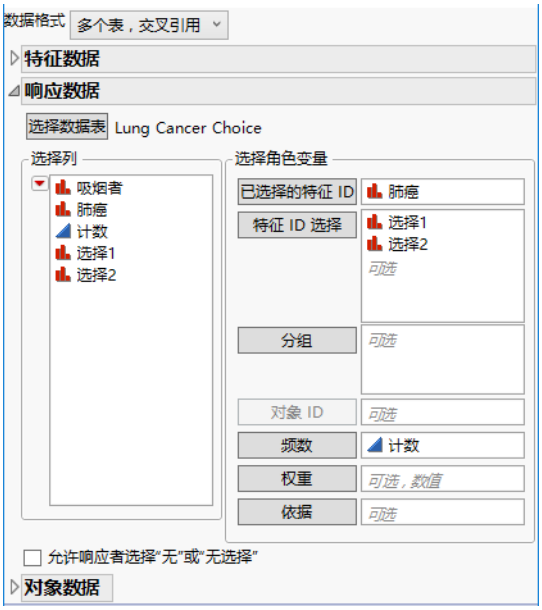
图 4.37 完成的“特征数据”面板



8. 打开**响应数据**分级显示项。
9. 点击**选择数据表**，选择 Lung Cancer Choice，然后点击**确定**。
10. 完成**响应数据**分级显示项：
    - 选择**肺癌**，然后点击**已选择的特征 ID**。
    - 选择**选择1** 和**选择2**，然后点击**特征 ID 选择**。

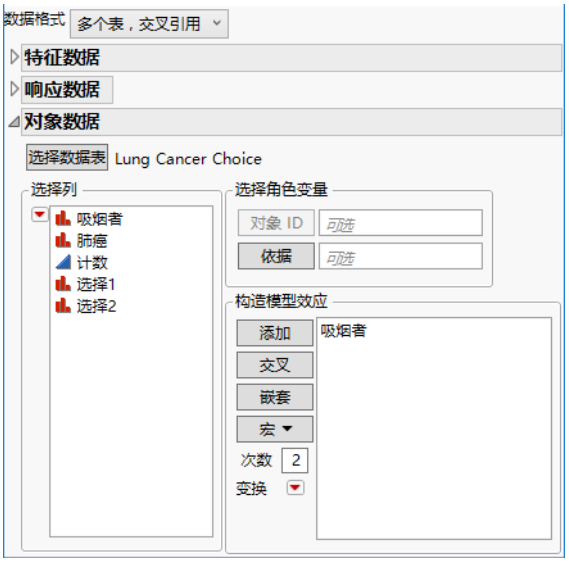
– 选择**计数**并点击**频数**。

图 4.38 完成的 “响应数据”面板



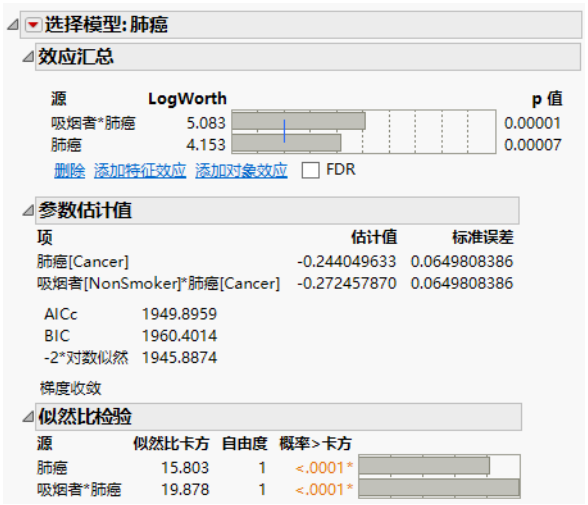
11. 打开**对象数据**分级显示项。
12. 点击**选择数据表**，选择 Lung Cancer Choice，然后点击**确定**。
13. 选择**吸烟者**并点击**添加**。

图 4.39 完成的“特征数据”面板



14. 点击**运行模型**。

图 4.40 选择建模 Logistic 回归结果



使用“拟合模型”平台。

1. 选择**帮助 > 样本数据库**，然后打开 Lung Cancer.jmp。
2. 选择**分析 > 拟合模型**。

   由于数据表包含一个模型脚本，“模型规格”窗口将自动完成。**名义型** Logistic 特质已被选定。

3. 点击**运行**。

图 4.41 拟合模型名义型 Logistic 回归结果

名义型 Logistic 拟合: 肺癌

效应汇总

梯度收敛,4 次迭代
频数: 计数

迭代

整体模型检验

模型	-对数似然	自由度	卡方	概率>卡方
差值	9.93901	1	19.87802	<.0001*
完全	972.94369			
简化	982.88270			

R 方 (U)	0.0101
AICc	1949.9
BIC	1960.4
观测数 (或权重和)	1418

拟合详细信息

参数估计值

项	估计值	标准误差	卡方	概率>卡方
截距	-0.4880993	0.1299617	14.11	0.0002*
吸烟者[NonSmoker]	-0.5449157	0.1299617	17.58	<.0001*

对数优势比: Cancer/NoCancer

估计值的协方差

效应似然比检验

源	参数数目	自由度	似然比卡方	概率>卡方
吸烟者	1	1	19.878016	<.0001*

需要注意的是，“选择”模型中**吸烟者*肺癌**的似然比卡方检验与 Logistic 模型中吸烟者的似然比卡方检验一致。图 4.40 和图 4.41 中显示的报表支持以下结论：吸烟对患肺癌有很大的影响。请参见《拟合线性模型》。

配对的案例对照研究的 Logistic 回归示例

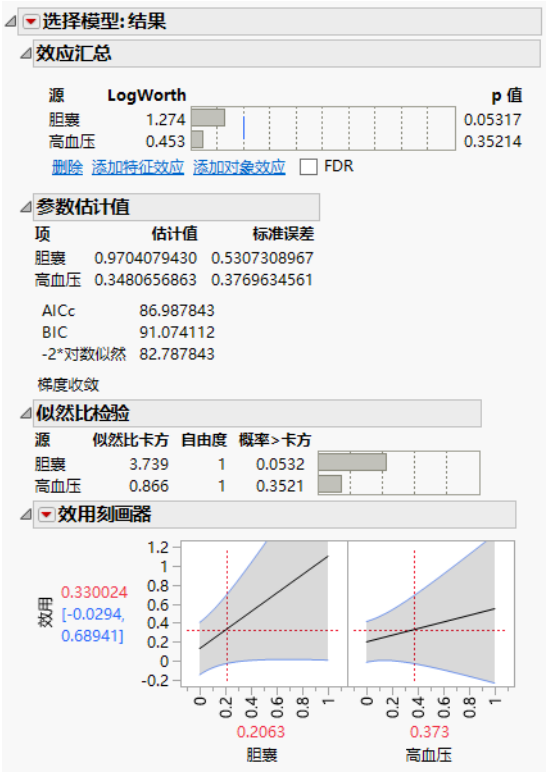
本节提供的示例使用 “选择”平台对包含 63 个配对的子宫内膜癌患者的研究结果进行 Logistic 回归。数据来自 Breslow and Day (1980) 中报告的关于洛杉矶地区子宫内膜癌数据的研究。病例对照分析的目标是在控制高血压的作用之后确定胆囊疾病患者患癌的相对风险。结果 1 表示存在子宫内膜癌， 0 表示没有患癌。是否存在胆囊和高血压也通过 0 或 1 来表示。

有关使用 “选择”平台执行 Logistic 回归的详细信息，请参见第 122 页的 “Logistic 回归”。

1. 选择**帮助 > 样本数据库**，然后打开 Endometrial Cancer.jmp。
2. 选择**分析 > 消费者研究 > 选择**。
3. 检查所选的 “数据格式”是否为**一个表，堆叠**。
4. 点击**选择数据表**按钮。
5. 选择 Endometrial Cancer 作为特征数据表。点击**确定**。
6. 选择**结果**，然后点击**响应指示符**。
7. 选择**配对**，然后点击**分组**。

8. 选择**胆囊**和**高血压**，然后在“构造特征效应”窗口中点击**添加**。
9. 取消选中 **Firth 偏倚调整估计值**复选框。
10. 点击**运行模型**。
11. 点击“选择模型 : 结果”红色小三角并选择**效用刻画器**。

图 4.42 子宫内膜癌数据的 Logistic 回归



分析结果给出了每个因子的似然比检验。我们可以看到**胆囊**在 0.05 水平下几乎是显著的（p 值 = 0.0532）。接下来可以使用“效用刻画器”可视化因子对响应的影响。

将数据转换为两个分析表的示例

考虑来自 Daganzo 的数据，数据表为 Daganzo Trip.jmp。该数据集包含每个测试对象采用三种不同交通方式的旅行时间及其首选交通方式。

添加交通方式选择和测试对象

1. 选择**帮助 > 样本数据库**，然后打开 Daganzo Trip.jmp 数据表。

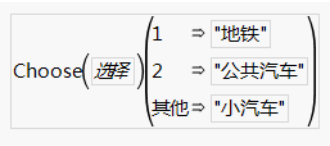
图 4.43 Daganzo Trip 表的部分内容

	地铁	公共汽车	小汽车	选择
1	16.481	16.196	23.89	2
2	15.123	11.373	14.182	2
3	19.469	8.822	20.819	2
4	18.847	15.649	21.28	2
5	12.578	10.671	18.335	2

列出的每个**选择**编号必须首先转换为相应的旅行模式名称。通过使用公式编辑器中的 Choose 函数可以轻松进行转换：

2. 选择**列 > 新建列**。
3. 指定列名为**交通方式选择**，指定建模类型为**名义型**。
4. 点击**列属性**，然后选择**公式**。
5. 在函数列表中点击**条件**，选择 Choose，然后按逗号两次以获取该函数的额外参数。
6. 为 Choose 表达式 (expr) 点击**选择**，然后双击每个子句输入框依次输入 “**地铁**”、“**公共汽车**”和 “**小汽车**”（带引号）。

图 4.44 Daganzo 数据的 “交通方式选择”列的 Choose 函数



7. 在 “公式编辑器”窗口中点击**确定**。
8. 在 “新建列”窗口中点击**确定**。

新生成的**交通方式选择**列显示在数据表中。因为每行包含的数据为每个测试对象所做的选择，应创建一列序列数据来标识测试对象。

9. 选择**列 > 新建列**。
10. 指定 “列名”为**测试对象**。
11. 点击 “初始化数据”旁边的**缺失 / 空**，然后选择**序列数据**。
12. 点击**确定**。

图 4.45 添加了新的 “交通方式选择”列和 “测试对象”列的 Daganzo Trip 部分数据

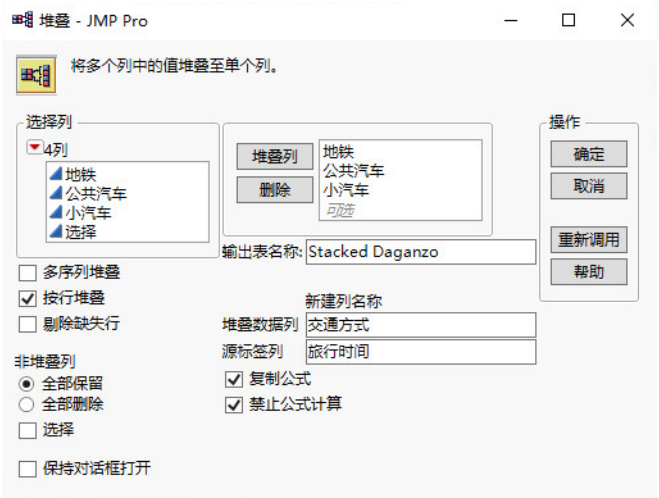
	地铁	公共汽车	小汽车	选择	交通方式选择	测试对象
1	16.481	16.196	23.89	2	公共汽车	1
2	15.123	11.373	14.182	2	公共汽车	2
3	19.469	8.822	20.819	2	公共汽车	3
4	18.847	15.649	21.28	2	公共汽车	4
5	12.578	10.671	18.335	2	公共汽车	5
6	11.513	20.582	27.838	1	地铁	6
7	10.651	15.537	17.418	1	地铁	7

堆叠数据

为了构造特征数据，每个备择选择需要在单独的行中表示。

1. 选择**表 > 堆叠**。
2. 选择**地铁**、**公共汽车**和**小汽车**，然后点击**堆叠列**。
3. 对于输出表名称，键入 Stacked Daganzo。为“堆叠数据列”键入**旅行时间**，为“源标签列”键入**交通方式**。

图 4.46 针对 Daganzo 数据的“堆叠”对话框



4. 点击**确定**。

图 4.47 Stacked Daganzo 表的部分内容

	选择	交通方式选择	测试对象	交通方式	旅行时间
1	2	公共汽车	1	地铁	16.481
2	2	公共汽车	1	公共汽车	16.196
3	2	公共汽车	1	小汽车	23.89
4	2	公共汽车	2	地铁	15.123
5	2	公共汽车	2	公共汽车	11.373
6	2	公共汽车	2	小汽车	14.182
7	2	公共汽车	3	地铁	19.469

创建特征数据表

对于特征数据表，您需要**测试对象**、**交通方式**和**旅行时间**列。

1. 选择**测试对象**、**交通方式**和**旅行时间**列，然后选择**表 > 子集**。
2. 选择**所有行**和**选定列**，然后点击**确定**。

生成的数据表的部分内容显示在图 4.48 中。请注意默认表名称为“‘Stacked Daganzo’的子集”。

图 4.48 Stacked Daganzo 数据的部分子集表

	测试对象	交通方式	旅行时间
1	1	地铁	16.481
2	1	公共汽车	16.196
3	1	小汽车	23.89
4	2	地铁	15.123
5	2	公共汽车	11.373
6	2	小汽车	14.182
7	3	地铁	19.469

创建响应数据表

对于响应数据表，您需要**测试对象**和**交通方式选择**列，同时还需要一列对应于每个可能的选择。

3. 从 Daganzo Trip.jmp 数据中，选择**测试对象**和**交通方式选择**列。
4. 选择**表 > 子集**。
5. 选择**所有行**和**选定列**，然后点击**确定**。

   请注意默认表名称为“‘Daganzo Trip’的子集”。
6. 选择**列 > 新建列**。
7. 对于列前缀，键入**选择**。
8. 选择**数据类型 > 字符**。
9. 为“要添加的列数”输入 3。
10. 点击**确定**。

    数据表中添加了**选择 1**、**选择 2** 和**选择 3** 列。
11. 在**选择 1** 的第一行中键入“公共汽车”（不带引号）。右击该单元格并选择**填充 > 填充至表末**。
12. 在**选择 2** 的第一行中键入“地铁”（不带引号）。右击该单元格并选择**填充 > 填充至表末**。
13. 在**选择 3** 的第一行中键入“小汽车”（不带引号）。右击该单元格并选择**填充 > 填充至表末**。

图 4.49 具有选择集的 Daganzo 数据的部分子集表

	交通方式选择	测试对象	选择 1	选择 2	选择 3
1	公共汽车	1	公共汽车	地铁	小汽车
2	公共汽车	2	公共汽车	地铁	小汽车
3	公共汽车	3	公共汽车	地铁	小汽车
4	公共汽车	4	公共汽车	地铁	小汽车
5	公共汽车	5	公共汽车	地铁	小汽车
6	地铁	6	公共汽车	地铁	小汽车
7	地铁	7	公共汽车	地铁	小汽车

拟合模型

现在您将原始 Daganzo Trip.jmp 表分为了两个单独的表，可以运行 “选择”平台了。

1. 选择**分析 > 消费者研究 > 选择**。
2. 从 “数据格式”列表中，选择**多个表，交叉引用**。
3. 指定模型，如图 4.50 中所示。

图 4.50 具有多个表的 Daganzo 数据的选择对话框

4. 点击**运行模型**。

   生成的参数估计值现在表示**旅行时间**的效用系数。

图 4.51 Daganzo 数据的旅行时间的参数估计值

选择模型: 交通方式选择

参数估计值

项	估计值	标准误差
Travel Time	-0.341768586	0.0745222259

AICc	68.766653
BIC	70.595342
-2*对数似然	66.683319
-2*Firth 对数似然	61.490004

梯度收敛

Firth 偏倚调整估计值

似然比检验

源	似然比卡方	自由度	概率>卡方
旅行时间	43.178	1	<.0001*

负的系数意味着旅行时间的增加对于消费者效用或满意度有负面影响。似然比检验结果表明包含“旅行时间”效应的“交通方式”模型是显著的。

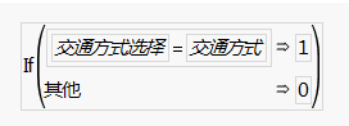
将数据转换为一个分析表的示例

无需创建两个或三个表，从更加切合实际的角度考虑，可以将数据转换成一个表。对于单表格式，按之前的示例那样添加对象效应。我们需要添加一个响应指示符列，而非生成三个不同列来表示选择集（**选择 1**、**选择 2**、**选择 3**）。变换数据以便用于单表情形：

1. 通过第 115 页的“将数据转换为两个分析表的示例”中所示的“堆叠数据”步骤创建或打开 Stacked Daganzo.jmp。
2. 选择**列** > **新建列**。
3. 键入**响应**作为列名。
4. 点击**列属性**，然后选择**公式**。
5. 在函数列表中选择**条件**，然后选择 If。
6. 为表达式 (expr) 选择列**交通方式选择**。
7. 输入“=”并选择**交通方式**。
8. 为 Then 子句键入 1，为 Else 子句键入 0。
9. 在“公式编辑器”窗口中点击**确定**。在“新建列”窗口中点击**确定**。

   完成的公式应类似于图 4.52。

图 4.52 Stacked Daganzo 数据的响应指示符的公式



10. 选择**测试对象**、**旅行时间**和**响应**列，然后选择**表** > **子集**。

11. 选择**所有行**和**选定列**，然后点击**确定**。

    新数据表的部分内容显示在图 4.53 中。

图 4.53 Stacked Daganzo 数据子集的部分内容

	测试对象	旅行时间	响应
1	1	16.481	0
2	1	16.196	1
3	1	23.89	0
4	2	15.123	0
5	2	11.373	1
6	2	14.182	0
7	3	19.469	0

12. 选择**分析** > **消费者研究** > **选择**以打开启动窗口并指定模型，如图 4.54 中所示。

图 4.54 单表式分析中 Stacked Daganzo 数据子集的 “选择对话框”

13. 点击**运行模型**。

图 4.55 单表式分析中 Daganzo 数据的旅行时间的参数估计值

选择模型: 响应

参数估计值

项	估计值	标准误差
旅行时间	-0.341768586	0.0745222259

AICc	68.766653
BIC	70.595342
-2*对数似然	66.683319
-2*Firth 对数似然	61.490004
梯度收敛	
Firth 偏倚调整估计值	

似然比检验

源	似然比卡方	自由度	概率>卡方
旅行时间	43.178	1	<.0001*

请注意该结果与前文图 4.51 中所示的两表模型得到的结果相同。

本章用简单的示例来说明如何使用“选择建模”平台。该平台还可以用于更复杂的模型，如涉及更复杂的转换以及包含交互作用项等等。

“选择”平台的统计详细信息

- “特殊数据表规则”
- “效用和概率”
- “梯度”

特殊数据表规则

默认选择集

若在每次试验中，您可以选择任意响应特征，则可以忽略**特征 ID 选择**项，它位于“选择”启动窗口的“响应数据”部分中的**选择角色变量**下。“选择模型”平台会假定每次的试验都包含所有选择特征。

具有响应数据的对象数据

若您在响应数据表中包含对象数据，请在“对象数据”下选择该表为**选择数据表**。在这种情况下，不需要指定**对象 ID** 列，实际上不使用它。通常假定对象数据在每个对象的多次试验中以一致方式重复。

Logistic 回归

可以使用“选择建模”平台执行普通 Logistic 回归。

注意：与“选择建模”平台相比，“以 X 拟合 Y”和“拟合模型”平台更便于执行 Logistic 回归建模。本节仅用于说明“选择建模”平台可用于 Logistic 回归（若需要的话）。

若您的数据已采用选择模型格式，您可能想使用以下所示的步骤进行 Logistic 回归分析。总的来说，需要三个步骤：

- 创建一个普通的特征数据表，每个响应水平对应一行。
- 将解释变量放入响应数据。
- 为对象数据表再次指定响应数据表。

有关使用“选择”平台执行 Logistic 回归的示例，请参见第 111 页的“使用“选择”平台进行 Logistic 回归的示例”和第 114 页的“配对的案例对照研究的 Logistic 回归示例”。

效用和概率

通过“选择”模型的参数估计值我们能够获得消费者**效用**或线性效用函数中的边缘效用。效用反映了消费者对具有特定特性的产品的满意度水平，由模型的参数估计值确定。

选择模型的统计公式可表示如下：

用 $X[k]$ 表示一个具有截距的对象特性设计行。

用 $Z[j]$ 表示一个没有截距的选择特性设计行。

则第 k 个对象相对 m 个选择中的第 j 个选择做出给定选择的概率为：

$$P_i[jk] = \frac{\exp(\beta'(X[k] \otimes Z[j]))}{\sum_{l=1}^{m} \exp(\beta'(X[k] \otimes Z[l]))}$$

其中：

- $\otimes$ 是 Kronecker 逐行乘积
- 分子用于计算实际选择的第 j 个备选
- 分母对提供给该测试对象的 m 个选择求和

梯度

在选择“按对象保存梯度”选项后，您所获得的梯度值是优化过程中用于生成估计值的按测试对象聚合的 Newton-Raphson 步长。在估计中，总梯度为零，$\Delta = H^{-1}g = 0$，其中 g 是最大似然估计中的对数似然函数的总梯度，H^{-1} 是 Hessian 函数的反函数或对数似然函数的二阶偏导数负数的反函数。

但是，取消聚合 Δ 将生成以下结果：

$\Delta = \Sigma_{ij}\Delta_{ij} = \Sigma H^{-1}g_{ij} = 0$，

其中，i 是测试对象索引，j 是每个测试对象的选择响应索引，Δij 是基于每次试验的偏 Newton-Raphson 步长，g_{ij} 是按试验排列的对数似然函数的梯度。

然后，每个测试对象的平均梯度步长可以按以下公式计算：

$$\overline{\Delta}_i = \Sigma_j \frac{\Delta_{ij}}{n_i},$$

其中 n_i 是每个测试对象的试验次数。$\overline{\Delta}_i$ 与测试对象 i 对最终参数估计的影响力度有关。若各组测试对象确实有不同的偏好结构，其对参数估计能够产生较强的影响力度，因此可用于对测试对象聚类。$\overline{\Delta}_i$ 即为保存的梯度力。接下来您便可以使用“聚类”平台对这些值进行聚类。

第 5 章

MaxDiff

针对 MaxDiff 实验拟合模型

MaxDiff（最大化差异度量）可用于替代标准偏好尺度来确定被评测项的相对重要性。MaxDiff 强制响应者报告其最偏好和最不偏好的选项。这样生成的排序往往比使用标准偏好尺度生成的排名更为明确。

MaxDiff 平台支持您执行以下操作：

- 使用有关响应者（测试对象）特点以及产品特性的信息。
- 整合来自一个、两个或三个源的数据。
- 获取测试对象级别的得分以便对数据进行细分或聚类。
- JMP PRO 使用 Bayes 方法估计特定于测试对象的系数。
- 使用修正偏倚的最大似然估计量 (Firth, 1993)。

图 5.1 MaxDiff 的所有比较报表

“所有水平比较”报表

差值（行-列） 差值的标准误差 Wald P 值	All Dressed	Barbecue	Biscuits and Gravy	Dill Pickle	Gyro	Ketchup	Reuben	Sour Cream	Southern Barbecue	Truffle Fries
All Dressed	0	-1.4951	-0.3222	-0.0804	0.94665	0.12371	0.24744	-0.228	-1.0068	-0.3615
	0	0.36838	0.29974	0.29447	0.33571	0.29728	0.30148	0.28158	0.3418	0.31136
		9.22e-5	0.28477	0.78538	0.00569	0.67812	0.41355	0.41975	0.00393	0.24809
Barbecue	1.49507	0	1.1729	1.41468	2.44172	1.61877	1.7425	1.26703	0.48824	1.13355
	0.36838	0	0.36267	0.3539	0.41499	0.36911	0.37117	0.37654	0.39164	0.39213
	9.22e-5		0.00161	0.00012	4.32e-8	2.64e-5	7.69e-6	0.00105	0.21514	0.00462
Biscuits and Gravy	0.32217	-1.1729	0	0.24179	1.26882	0.44588	0.56961	0.09413	-0.6847	-0.0393
	0.29974	0.36267	0	0.28249	0.34456	0.29752	0.30639	0.30716	0.34038	0.31345
	0.28477	0.00161		0.39388	0.00036	0.1368	0.06567	0.75983	0.0467	0.90032
Dill Pickle	0.08038	-1.4147	-0.2418	0	1.02703	0.20409	0.32782	-0.1477	-0.9264	-0.2811
	0.29447	0.3539	0.28249	0	0.35251	0.27938	0.30328	0.30707	0.35532	0.30871
	0.78538	0.00012	0.39388		0.00432	0.46661	0.28208	0.63156	0.01038	0.36444
Gyro	-0.9467	-2.4417	-1.2688	-1.027	0	-0.8229	-0.6992	-1.1747	-1.9535	-1.3082
	0.33571	0.41499	0.34456	0.35251	0	0.32802	0.34758	0.33327	0.38939	0.35477
	0.00569	4.32e-8	0.00036	0.00432		0.01356	0.04668	0.00062	2.01e-6	0.00035
Ketchup	-0.1237	-1.6188	-0.4459	-0.2041	0.82294	0	0.12373	-0.3517	-1.1305	-0.4852
	0.29728	0.36911	0.29752	0.27938	0.32802	0	0.30893	0.31915	0.34808	0.31157
	0.67812	2.64e-5	0.1368	0.46661	0.01356		0.68956	0.27279	0.00154	0.12223
Reuben	-0.2474	-1.7425	-0.5696	-0.3278	0.69921	-0.1237	0	-0.4755	-1.2543	-0.609
	0.30148	0.37117	0.30639	0.30328	0.34758	0.30893	0	0.30932	0.35324	0.31817
	0.41355	7.69e-6	0.06567	0.28208	0.04668	0.68956		0.12709	0.00056	0.05821
Sour Cream and Onion	0.22804	-1.267	-0.0941	0.14766	1.17469	0.35175	0.47548	0	-0.7788	-0.1335
	0.28158	0.37654	0.30716	0.30707	0.33327	0.31915	0.30932	0	0.34896	0.31379
	0.41975	0.00105	0.75983	0.63156	0.00062	0.27279	0.12709		0.02764	0.67137
Southern Barbecue	1.00683	-0.4882	0.68465	0.92644	1.95348	1.13053	1.25426	0.77879	0	0.6453
	0.3418	0.39164	0.34038	0.35532	0.38939	0.34808	0.35324	0.34896	0	0.34904
	0.00393	0.21514	0.0467	0.01038	2.01e-6	0.00154	0.00056	0.02764		0.06715
Truffle Fries	0.36152	-1.1335	0.03935	0.28114	1.30817	0.48523	0.60896	0.13348	-0.6453	0
	0.31136	0.39213	0.31345	0.30871	0.35477	0.31157	0.31817	0.31379	0.34904	0
	0.24809	0.00462	0.90032	0.36444	0.00035	0.12223	0.05821	0.67137	0.06715	

目录

“MaxDiff 建模”平台概述 127
MaxDiff 平台的示例 127
单表格式 128
多表格式 130
启动 MaxDiff 平台 133
针对“一个表，堆叠”的启动窗口 134
针对“多个表，交叉引用”的启动窗口 135
“MaxDiff 模型”报表 138
效应汇总 139
MaxDiff 结果 140
参数估计值 141
Bayes 参数估计值 142
似然比检验 143
MaxDiff 平台选项 143
“比较”报表 145
保存 Bayes 链 145

“MaxDiff 建模”平台概述

MaxDiff 亦称**最优最差缩放** (BWS)，是基于选择的测量方法。MaxDiff 要求响应者报告**最优**和**最差**选择，而不要求响应者从若干备选特征中报告最优选择。相对于响应者仅报告一个最优选择的方法，MaxDiff 方法可以提供关于偏好的更多信息。有关 MaxDiff 研究课题的背景信息，请参见 Louviere et al. (2015)。有关选择建模的背景信息，请参见 Louviere et al. (2015)、Train (2009) 和 Rossi et al. (2005) 一书。

MaxDiff 分析使用随机效用理论框架。假定某一选择对响应者存在基本价值或**效用**。MaxDiff 平台估计这些效用，除此之外还估计相对于其他选择更偏好某一选择的概率。上述估计需要使用条件 Logistic 回归来完成。请参见 McFadden (1974)。

注意：可使用“MaxDiff 设计”平台设计单因子 MaxDiff 研究。请参见《实验设计指南》。

细分和 Bayes 测试对象级别的效应

市场研究人员有时想分别分析不同测试对象的偏好结构，以查看是否存在不同表现行为的群体。若有足够的数据，可以在“响应数据”中指定“依据组”，或者您还可以引入测试对象标识符作为测试对象级的模型项。不过若测试对象人数很多的话，此方法的成本会很高。文献中讨论的其他细分方法包括 Bayesian 和混合法。

若没有足够数据指定“依据组”，则可以使用响应数据和**按对象保存梯度**选项对测试对象进行聚类的方法在 JMP 中进行细分。该选项创建一个新数据表，其中包含每个测试对象各参数的平均 Hessian 比例梯度。有关示例，请参见“选择模型”一章中第 107 页的“细分示例”。有关梯度值的详细信息，请参见“选择模型”一章中第 123 页的“梯度”。

JMP PRO MaxDiff 还提供分层 Bayes 方法来估计测试对象级别的效应。该方法适用于市场细分。

MaxDiff 平台的示例

有 30 位响应者参加了 MaxDiff 研究，目的在于比较七种不同口味的薯片。每个选择集都包含三种特征（薯片口味）。对于每个选择集，响应者的最优选择记录为 1，其最差选择记录为 -1。中间选择记录为 0。

MaxDiff 平台可以分析以单表格式或多表格式呈现的数据。在多表格式中，关于响应、选择集和测试对象的信息保存在不同的数据表中。在单表格式中，该信息包含在单个数据表中。

- 第 128 页的“单表格式” 显示如何分析采用单表格式的一部分可用数据。请注意，您可以向单个表添加其他特征和对象数据，以便执行更全面的分析。
- 第 130 页的“多表格式”显示如何将不同表中的信息集中到一个 MaxDiff 分析中。

单表格式

1. 选择**帮助** > **样本数据库**，然后打开 Potato Chip Combined.jmp。
2. 选择**分析** > **消费者研究** > MaxDiff。

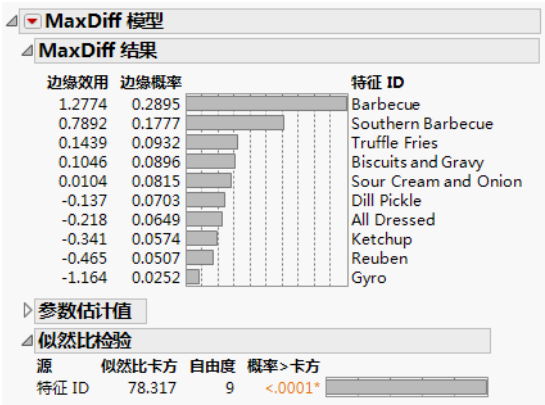
   请注意，默认数据格式设置为 “一个表，堆叠”。
3. 点击**选择数据表**。
4. 选择 Potato Chip Combined，然后点击**确定**。
5. 将角色分配给列以完成启动对话框：
   - 选择**响应**，然后点击**响应指示符**。
   - 选择**响应者**，然后点击**对象 ID**。
   - 选择**选择集 ID**，然后点击**选择集 ID**。
   - 选择**特征 ID**，然后在“构造特征效应”面板中点击**添加**。

图 5.2 完成的 MaxDiff 启动窗口

请注意，当您将 “响应”列指定为 “响应指示符”变量时，“最差”选择的设置改为 -1。

6. 点击**运行模型**。

图 5.3 Potato Chip Combined.jmp 的 MaxDiff 报表



MaxDiff 模型

MaxDiff 结果

边缘效用	边缘概率	特征 ID
1.2774	0.2895	Barbecue
0.7892	0.1777	Southern Barbecue
0.1439	0.0932	Truffle Fries
0.1046	0.0896	Biscuits and Gravy
0.0104	0.0815	Sour Cream and Onion
-0.137	0.0703	Dill Pickle
-0.218	0.0649	All Dressed
-0.341	0.0574	Ketchup
-0.465	0.0507	Reuben
-1.164	0.0252	Gyro

参数估计值

似然比检验

源	似然比卡方	自由度	概率>卡方
特征 ID	78.317	9	<.0001*

该报表指示**特征 ID** 显著，这表明对不同种类薯片的偏好显著不同。Barbecue 薯片的“边缘效用”是最大的。相对于其他口味的薯片，被访者更喜欢 Barbecue 薯片的估计概率为 0.2895。

7. 点击“MaxDiff 模型”红色小三角并选择 **“所有水平比较”报表**。

图 5.4 “所有比较”报表

“所有水平比较”报表

差值（行-列） 差值的标准误差 Wald P 值	All Dressed	Barbecue	Biscuits and	Dill Pickle	Gyro	Ketchup	Reuben	Sour Cream	Southern Barbecue	Truffle Fries
All Dressed	0	-1.4951	-0.3222	-0.0804	0.94665	0.12371	0.24744	-0.228	-1.0068	-0.3615
	0	0.36838	0.29974	0.29447	0.33571	0.29728	0.30148	0.28158	0.3418	0.31136
		9.22e-5	0.28477	0.78538	0.00569	0.67812	0.41355	0.41975	0.00393	0.24809
Barbecue	1.49507	0	1.1729	1.41468	2.44172	1.61877	1.7425	1.26703	0.48824	1.13355
	0.36838	0	0.36267	0.3539	0.41499	0.36911	0.37117	0.37654	0.39164	0.39213
	9.22e-5		0.00161	0.00012	4.32e-8	2.64e-5	7.69e-6	0.00105	0.21514	0.00462
Biscuits and Gravy	0.32217	-1.1729	0	0.24179	1.26882	0.44588	0.56961	0.09413	-0.6847	-0.0393
	0.29974	0.36267	0	0.28249	0.34456	0.29752	0.30639	0.30716	0.34038	0.31345
	0.28477	0.00161		0.39388	0.00036	0.1368	0.06567	0.75983	0.0467	0.90032
Dill Pickle	0.08038	-1.4147	-0.2418	0	1.02703	0.20409	0.32782	-0.1477	-0.9264	-0.2811
	0.29447	0.3539	0.28249	0	0.35251	0.27938	0.30328	0.30707	0.35532	0.30871
	0.78538	0.00012	0.39388		0.00432	0.46661	0.28208	0.63156	0.01038	0.36444
Gyro	-0.9467	-2.4417	-1.2688	-1.027	0	-0.8229	-0.6992	-1.1747	-1.9535	-1.3082
	0.33571	0.41499	0.34456	0.35251	0	0.32802	0.34758	0.33327	0.38939	0.35477
	0.00569	4.32e-8	0.00036	0.00432		0.01356	0.04668	0.00062	2.01e-6	0.00035
Ketchup	-0.1237	-1.6188	-0.4459	-0.2041	0.82294	0	0.12373	-0.3517	-1.1305	-0.4852
	0.29728	0.36911	0.29752	0.27938	0.32802	0	0.30893	0.31915	0.34808	0.31157
	0.67812	2.64e-5	0.1368	0.46661	0.01356		0.68956	0.27279	0.00154	0.12223
Reuben	-0.2474	-1.7425	-0.5696	-0.3278	0.69921	-0.1237	0	-0.4755	-1.2543	-0.609
	0.30148	0.37117	0.30639	0.30328	0.34758	0.30893	0	0.30932	0.35324	0.31817
	0.41355	7.69e-6	0.06567	0.28208	0.04668	0.68956		0.12709	0.00056	0.05821
Sour Cream and Onion	0.22804	-1.267	-0.0941	0.14766	1.17469	0.35175	0.47548	0	-0.7788	-0.1335
	0.28158	0.37654	0.30716	0.30707	0.33327	0.31915	0.30932	0	0.34896	0.31379
	0.41975	0.00105	0.75983	0.63156	0.00062	0.27279	0.12709		0.02764	0.67137
Southern Barbecue	1.00683	-0.4882	0.68465	0.92644	1.95348	1.13053	1.25426	0.77879	0	0.6453
	0.3418	0.39164	0.34038	0.35532	0.38939	0.34808	0.35324	0.34896	0	0.34904
	0.00393	0.21514	0.0467	0.01038	2.01e-6	0.00154	0.00056	0.02764		0.06715
Truffle Fries	0.36152	-1.1335	0.03935	0.28114	1.30817	0.48523	0.60896	0.13348	-0.6453	0
	0.31136	0.39213	0.31345	0.30871	0.35477	0.31157	0.31817	0.31379	0.34904	0
	0.24809	0.00462	0.90032	0.36444	0.00035	0.12223	0.05821	0.67137	0.06715	

每一项比较都是在标记行的薯片类型与标记列的薯片类型之间的估计效用差异。较小 *p* 值的颜色为深蓝或红色，具体取决于差异符号。例如，根据横贯 Gyro 行的蓝色，您可以看到

Gyro 薯片的效用显著低于其他所有薯片类型。Barbecue 薯片的效用比其他所有薯片类型都高，尽管与 Southern Barbecue 薯片的差别不大。

注意：由于 “所有比较报表”的 *p* 值未针对多重比较进行修正，所以仅将其用作指导。

多表格式

这一版本的薯片研究使用三个数据表：Potato Chip Profiles.jmp、Potato Chip Responses.jmp 和 Potato Chip Subjects.jmp。尽管您总是可将数据整合到单个表中，但若您想要在分析中加入其他特征和测试对象变量，多表方法比单表分析更为方便。

完成启动窗口

1. 选择**帮助** > **样本数据库**，然后打开 Potato Chip Responses.jmp 样本数据表。

注意：若您不想执行用于完成启动窗口操作的步骤，请点击**“口味”的** MaxDiff 脚本旁边的绿色小三角。然后前进到第 132 页的“探索模型”。

2. 点击**打开特征表和对象表**脚本旁边的绿色小三角。
   - 特征数据表 Potato Chip Profiles.jmp 列出研究中的所有薯片类型（**口味**）以及关于原产地的信息（**产地**）。每个选择都具有一个**特征 ID**。
   - 对象数据表 Potato Chip Subjects.jmp 列出响应者。其中还提供关于每个响应者的其他信息：**国籍**和**性别**。
   - 响应数据表 Potato Chip Responses.jmp 列出响应者。对于每个响应者，列出每组特征的**调查 ID** 和**选择集 ID**，以及每个选择集的**特征 ID** 值。该表还包含**最佳特征**列和**最差特征**列中的响应数据。
3. 从以上三个数据表中的任意表中选择**分析** > **消费者研究** > MaxDiff。
4. 从 “数据格式”列表中，选择多个表，交叉引用。

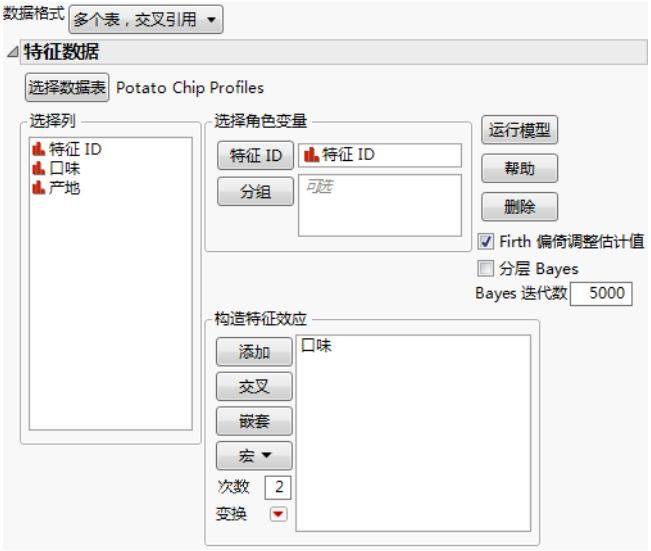
   共有三个不同的分级显示项，每个分级显示项对应一个数据源。
5. 在 “特征数据”下点击**选择数据表**。

   将显示 “特征数据表”窗口，该窗口提示您指定特征数据所在的数据表。
6. 选择 Potato Chip Profiles，然后点击**确定**。

   该表中的列显示在**选择列**中。
7. 从 “选择列”列表中选择**特征 ID**，然后点击**选择角色变量**下的**特征 ID**。
8. 选择**口味**，然后点击**构造模型效应**下的**添加**。

   请注意，**产地**是您可以添加到效应列表的另一个特征效应。

图 5.5 完成 “特征数据” 分级显示项



9. 打开响应数据分级显示项。点击**选择数据表**。
10. 选择 Potato Chip Responses，然后点击**确定**。
11. 将角色分配给列以完成启动对话框：
    - 选择**最佳特征**，然后点击**最佳选择**。
    - 选择**最差特征**，然后点击**最差选择**。
    - 选择**选择 1**、**选择 2** 和**选择 3**，然后点击**特征 ID 选择**。
    - 选择**响应者**，然后点击**对象 ID**。

图 5.6 完成的 “响应数据”分级显示项

12. 打开对象数据分级显示项。点击**选择数据表**。
13. 选择 Potato Chip Subjects，然后点击**确定**。
14. 选择**响应者**，然后点击**对象 ID**。
15. 选择**国籍**和**性别**，然后点击**构造模型效应**下的**添加**。

图 5.7 完成的 “对象数据”分级显示项

探索模型

1. 点击**运行模型**。

图 5.8 “MaxDiff 模型”报表

“效应汇总”报表显示模型中的项并提供其显著性的 *p* 值。请注意，**口味**是一个特征效应，而且每个**国籍 * 口味**和**性别 * 口味**都是测试对象与特征效应之间的交互作用。

“似然比检验”报表表明**口味**对于被访者的偏好选择是显著的。

启动 MaxDiff 平台

通过选择**分析 > 消费者研究 > MaxDiff** 启动 MaxDiff 平台。

MaxDiff 平台的数据可组合到一个数据表中，也可分别位于两个或三个不同数据表中。在“MaxDiff”启动窗口中，指定在“数据格式”列表中使用一个还是多个数据表。

一个表，堆叠

对于“一个表，堆叠”格式，数据包含在单个数据表中。提供给选择集内某个对象的每个特征都对应一行，并且该选择集内的最佳和最差特征都对应一个指示符。Potato Chip Combined.jmp 样本数据表包含以单表格式显示的 MaxDiff 实验的结果。请参见第 128 页的“单表格式”。

有关该格式启动窗口的详细信息，请参见第 134 页的“针对“一个表，堆叠”的启动窗口”。

多个表，交叉引用

对于“多个表，交叉引用”格式，数据包含在两个或三个不同数据表中。需要一个特征数据表和一个响应数据表。测试对象数据表是可选的。请注意以下几点：

- 特征数据表必须包含一列，其中每个特征都有一个唯一标识符，并且必须包含针对特征水平变量的列。响应数据表中使用特征标识符来标识每个选择集的最佳和最差特征响应。
- 可选的测试对象数据表必须包含一列，其中每个测试对象都有一个唯一的测试对象标识符，并且必须包含针对测试对象水平变量的列。在响应数据表中使用测试对象标识符来标识测试对象。

这种格式的启动窗口包含三个部分：特征数据、响应数据和测试对象数据。每个部分都对应不同的数据表。您可以按需展开或折叠每个部分。

Potato Chip Profiles.jmp、Potato Chip Responses.jmp 和 Potato Chip Subjects.jmp 样本数据表包含使用三个表的 MaxDiff 实验的结果。请参见第 130 页的“多表格式”。

有关该格式启动窗口的详细信息，请参见第 135 页的“针对“多个表，交叉引用”的启动窗口”。

针对“一个表，堆叠”的启动窗口

通过选择**分析** > **消费者研究** > MaxDiff 启动 MaxDiff 平台。从“数据格式”菜单中为一个表选择**一个表，堆叠**。

图 5.9 针对 “一个表，堆叠”数据格式的启动窗口

有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

选择数据表 选择或打开包含组合数据的数据表。选择 “其他”可打开当前尚未打开的文件。

响应指示符 包含偏好数据的列。使用 1、-1 或 0 中的两个值表示最佳选择和最差选择，第三个值表示最佳和最差之外的特征。默认编码用 1 指示“最佳”选择，用 -1 指示“最差”选择。

对象 ID 用于标识研究参与者的标识符。

选择集 ID 针对给定的偏好决定，为测试对象提供的特征集指示符。

分组 该列与“选择集 ID”一同使用时，可唯一指定每个选择集。例如，若针对**调查** = A 的某个选择集的**选择集 ID** = 1，而针对**调查** = B 的另一个选择集的**选择集 ID** = 1，则**调查**应用作一个**分组**列。

依据 为依据变量的每个水平生成单独的报表。若分配了多个依据变量，则为依据变量的每种可能组合都生成一个单独的报表。

构造特征效应 添加由特征的特性构造的效应。

有关“构造特征效应”面板的信息，请参见《拟合线性模型》。

构造对象效应（可选） 添加由测试对象相关因子构造的效应。

有关“构造对象效应”面板的信息，请参见《拟合线性模型》。

Firth 偏倚调整估计值 计算修正偏倚的 MLE，以便生成比没有修正偏倚的 MLE 更准确的估计值和检验。这些估计值还可缓解 Logistic 模型中常见的分离问题。有关 Logistic 回归中的分离问题的讨论，请参见 Heinze and Schemper (2002)。

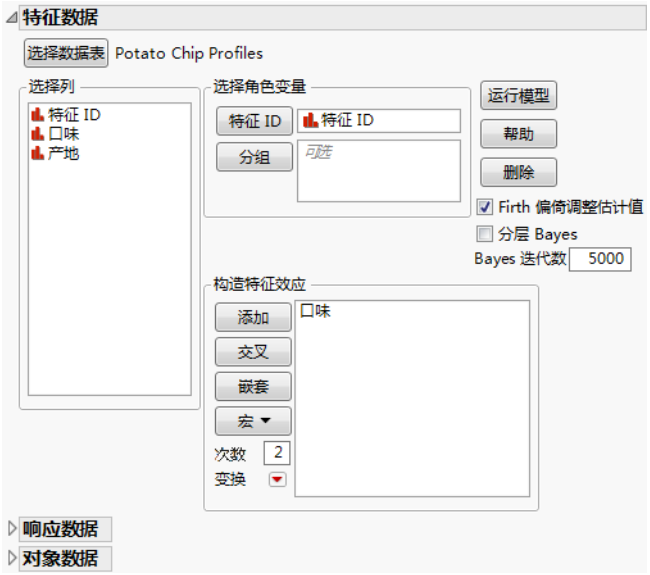
JMP PRO **分层 Bayes** 使用 Bayes 方法估计特定于测试对象的参数。请参见第 142 页的“Bayes 参数估计值”。

JMP PRO **Bayes 迭代数** （仅在选定“分层 Bayes”时适用。）用来估计对象效应的自适应 Bayes 算法的迭代总次数。该数字包括废弃的老化迭代期。老化迭代次数等于启动窗口中指定的“Bayes 迭代数”的一半。

针对“多个表，交叉引用”的启动窗口

通过选择**分析 > 消费者研究 > MaxDiff** 启动 MaxDiff 平台。从“数据格式”菜单中为多个表选择**多个表，交叉引用**。

图 5.10 针对“多个表，交叉引用”数据格式的启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

若采用 “多个表，交叉引用”，则启动窗口包含三个部分：

- 第 136 页的“特征数据”
- 第 136 页的“响应数据”
- 第 137 页的“对象数据”

特征数据

特征数据表描述的是与每个选择关联的特性。每个选择可以包含很多不同的特性，每个特性通过数据表中的一列来显示。每个可能的选择都对应一行，而且它们都有唯一的 ID。

选择数据表 选择或打开包含特征数据的数据表。选择 “其他”可打开当前尚未打开的文件。

特征 ID 每行选择组合的标识符。若**特征 ID** 列不能唯一标识特征数据表中的每行，您需要添加**分组**列。添加**分组**列，直到**分组**和**特征 ID** 列的组合能够唯一标识该行或特征。

分组 一列，在该列与 “选择集 ID”列一同使用时，可唯一指定每个选择集。例如，若对于**调查** = A **特征 ID** = 1，对于**调查** = B 另有一个**特征 ID** = 1，则**调查**将用作**分组**列。

构造特征效应 添加从特征中的特性构造的效应。

有关 “构造特征效应”面板的信息，请参见 《拟合线性模型》。

Firth 偏倚调整估计值 计算修正偏倚的 MLE，以便生成比没有修正偏倚的 MLE 更准确的估计值和检验。这些估计值还可缓解 Logistic 模型中常见的分离问题。有关 Logistic 回归中的分离问题的讨论，请参见 Heinze and Schemper (2002)。

JMP PRO **分层 Bayes** 使用 Bayes 方法估计特定于测试对象的参数。请参见第 142 页的“Bayes 参数估计值”。

JMP PRO **Bayes 迭代数** （仅在选定“分层 Bayes”时适用。）用来估计对象效应的自适应 Bayes 算法的迭代总次数。该数字包括废弃的老化迭代期。老化迭代次数等于启动窗口中指定的 “Bayes 迭代数”的一半。

响应数据

图 5.11 显示使用 Potato Chip Responses.jmp 填充的“响应数据”分级显示项。

图 5.11 “响应数据”分级显示项

响应数据表包含研究结果。其中提供了每个试验的选择集 ID，以及测试对象选定的最佳和最差特征。响应数据通过选择集列和选择响应列链接到特征数据。当数据中包含多个组时，分组变量可用于对选择指标进行分组匹配。

选择数据表 选择或打开包含响应数据的数据表。选择 “其他”可打开当前尚未打开的文件。

最佳选择 “响应”表列，其中包含研究参与者指定为 “最佳”特征的特征 ID。

最差选择 “响应”表列，其中包含研究参与者指定为 “最差”特征的特征 ID。

特征 ID 选择 对于每个选择集，包含可能选择集的**特征** ID 的列。至少有三个特征。

分组 一列，在该列与**已选择的特征** ID 列一同使用时，可唯一指定每个选择集。

对象 ID 用于研究参与者的唯一标识符。

频数 包含频数的列。若 *n* 是给定行的频数变量的值，则该行在计算中使用 n 次。若它小于 1 或缺失，则 JMP 不使用它来计算任何分析。

权重 包含数据表中每个观测的权重值的列。仅当权重值大于零时才在分析中包含该权重。

依据 为依据变量的每个水平生成单独的报表。若分配了多个依据变量，则为依据变量的每种可能组合都生成一个单独的报表。

对象数据

图 5.12 显示使用 Potato Chip Subjects.jmp 填充的“对象数据”分级显示项。

图 5.12 “对象数据”分级显示项

注意：对象数据表是可选的，这取决于是否对对象效应建模。

对象数据表包含“对象 ID”，以及每个测试对象的一个或多个特性或特征列。对象数据表包含与测试对象数相同的行数，还有一个标识符列，该列与响应数据表中的列可以相互匹配。

注意：您可以将对象数据放入响应数据表，但您需要在“对象数据”分级显示项中指定对象效应。

选择数据表 选择或打开包含对象数据的数据表。选择 “其他”可打开当前尚未打开的文件。

对象 ID 测试对象的唯一标识符。

依据 为依据变量的每个水平生成单独的报表。若分配了多个依据变量，则为依据变量的每种可能组合都生成一个单独的报表。

构造模型效应 添加由对象数据表中的列构造的效应。

有关“构造模型效应”面板的信息，请参见 《拟合线性模型》。

“MaxDiff 模型”报表

“MaxDiff 模型”窗口默认显示以下一些报表，具体取决于您在启动窗口中的选择。

- “效应汇总”
- “MaxDiff 结果”
- “参数估计值”
- “Bayes 参数估计值”
- “似然比检验”

有关平台选项的说明，请参见第 143 页的“MaxDiff 平台选项”。

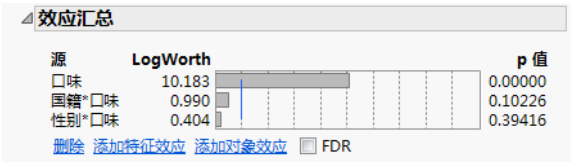
效应汇总

若模型包含不止一个效应，则显示“效应汇总”报表。该报表列出模型估计的效应，并提供这些效应的 LogWorth（或 FDR LogWorth）值的图。该报表还提供一些控件，这些控件可支持您在模型中添加或删除效应。模型拟合报表会基于在“效应汇总”报表中所做的更改自动更新。请参见《拟合线性模型》。

注意：“效应汇总”报表不适用于使用“分层 Bayes”拟合的模型。

图 5.13 显示通过运行 Potato Chip Responses.jmp 中的脚本“口味”的 MaxDiff 获得的“效应汇总”报表。

图 5.13 “效应汇总”报表



“效应汇总”表列

“效应汇总”表包含以下列：

源 列出按 *p* 值升序排序的模型效应。

LogWorth 显示每个模型效应的 LogWorth，该值定义为 $-\log_{10}(p \text{ 值})$。该变换调整 *p* 值以提供适用于绘图的尺度。超过 2 的值在 0.01 水平下显著（因为 $-\log_{10}(0.01) = 2$）。

FDR LogWorth 显示每个模型效应的假发现率 LogWorth，该值定义为 $-\log_{10}(\text{FDR P 值})$。这是用于标绘和评估显著性的最佳统计量。选中 FDR 复选框可将 LogWorth 列替换为 FDR LogWorth 列。

条形图 显示 LogWorth（或 FDR LogWorth）值的条形图。该图在整数值处显示垂直虚线，在 2 处显示蓝色参考线。

p 值 显示每个模型效应的 *p* 值。这是与“似然比检验”报表中显示的显著性检验对应的 *p* 值。

FDR p 值 显示使用 Benjamini-Hochberg 方法计算的每个模型效应的假发现率 *p* 值。该方法调整 *p* 值以控制多重检验的假发现率。选中 FDR 复选框可将 p 值列替换为 FDR p 值列。

有关 FDR 校正的详细信息，请参见 Benjamini and Hochberg (1995)。有关假发现率的详细信息，请参见《预测和专业建模》或 Westfall et al.(2011)。

“效应汇总”表选项

汇总表下方的选项支持您添加和删除效应：

删除 从模型中删除选定的效应。要删除一个或多个效应，请选择与效应对应的行并点击 “删除”按钮。

添加特征效应 打开一个列对话框，其中包含针对 “一个表，堆叠”数据格式的数据表中所有列的列表，以及针对“多个表，交叉引用”数据格式的特征数据表中的列的列表。选择您要添加到模型中的列，然后点击列选择列表下的 “添加”将这些列添加到模型中。点击 “关闭”关闭该面板。

添加对象效应 打开一个列对话框，其中包含针对 “一个表，堆叠”数据格式的数据表中所有列的列表，以及针对“多个表，交叉引用”数据格式的测试对象数据表中的列的列表。选择您要添加到模型中的列，然后点击列选择列表下的 “添加”将这些列添加到模型中。点击 “关闭”关闭该面板。

MaxDiff 结果

图 5.14 显示通过运行 Potato Chip Responses.jmp 中的脚本**没有对象效应的** MaxDiff 获得的 “MaxDiff 结果”报表。

图 5.14 “MaxDiff 结果”报表

MaxDiff 结果

边缘效用	边缘概率	口味
1.2774	0.2895	Barbecue
0.7892	0.1777	Southern Barbecue
0.1439	0.0932	Truffle Fries
0.1046	0.0896	Biscuits and Gravy
0.0104	0.0815	Sour Cream and Onion
-0.137	0.0703	Dill Pickle
-0.218	0.0649	All Dressed
-0.341	0.0574	Ketchup
-0.465	0.0507	Reuben
-1.164	0.0252	Gyro

对于在启动窗口中指定的每个特征效应，显示以下信息：

边缘效用 相应效应水平的感知值的指示符。较大值表明特性的效用大。

边缘概率 测试对象表示其偏好相应效应水平而不是其他所有水平的估计概率。对于每个效应，边缘概率加总为 1。

条形图 显示边缘概率的条形图。

“效应”列 提供效应名称及其水平的列表。各水平定义边缘效用和边缘概率估计值应用到的特性。

参数估计值

该报表提供有关参数估计值、拟合准则和拟合算法的详细信息。

图 5.15 显示通过运行 Potato Chip Responses.jmp 中的脚本“口味”的 MaxDiff 获得的“参数估计值”报表。

图 5.15 “参数估计值”报表

参数估计值

项	估计值	标准误差
口味[All Dressed]	-0.15616234	0.2227718641
口味[Barbecue]	1.22210326	0.2978972284
口味[Biscuits and Gravy]	0.16398758	0.2249076601
口味[Dill Pickle]	-0.17356828	0.2151668801
口味[Gyro]	-1.11927509	0.2827016652
口味[Ketchup]	-0.47308393	0.2319900823
口味[Reuben]	-0.50927309	0.2294979936
口味[Sour Cream and Onion]	0.21115573	0.2450361368
口味[Southern Barbecue]	0.70149945	0.2695322771
国籍[Canadian]*口味[All Dressed]	-0.04368106	0.2239442590
国籍[Canadian]*口味[Barbecue]	-0.16180196	0.2978036763
国籍[Canadian]*口味[Biscuits and Gravy]	0.05734172	0.2233312021
国籍[Canadian]*口味[Dill Pickle]	-0.09824391	0.2188896544
国籍[Canadian]*口味[Gyro]	0.43257276	0.2907051874
国籍[Canadian]*口味[Ketchup]	-0.38035261	0.2349720398
国籍[Canadian]*口味[Reuben]	-0.34677939	0.2342643217
国籍[Canadian]*口味[Sour Cream and Onion]	0.56678250	0.2355493683
国籍[Canadian]*口味[Southern Barbecue]	-0.00712532	0.2720518372
性别[Female]*口味[All Dressed]	-0.26955535	0.2106170508
性别[Female]*口味[Barbecue]	0.39881410	0.2961368976
性别[Female]*口味[Biscuits and Gravy]	0.09184462	0.2245258406
性别[Female]*口味[Dill Pickle]	-0.11461679	0.2131955199
性别[Female]*口味[Gyro]	-0.40538827	0.2950786212
性别[Female]*口味[Ketchup]	-0.07317652	0.2135446256
性别[Female]*口味[Reuben]	0.20780054	0.2227994637
性别[Female]*口味[Sour Cream and Onion]	0.27353428	0.2337547848
性别[Female]*口味[Southern Barbecue]	0.07570279	0.2627935035

AICc	397.44423
BIC	456.27172
-2*对数似然	327.00945
-2*Firth 对数似然	244.39836

梯度收敛

Firth 偏倚调整估计值

项 列出模型中的项。

估计值 与相应项关联的参数的估计值。在离散选择实验中，参数估计值有时称为**成分效用值**。每个成分效用值是与给定项关联的效用的系数。默认情况下，这些估计值基于 Firth 修正偏倚的最大似然估计量，因此比没有修正偏倚的 MLE 更准确。

标准误差 参数估计值的标准差的估计值。

比较准则

AICc（校正的 Akaike 信息准则）、BIC（Bayes 信息准则）、–2对数似然和 –2Firth 对数似然拟合统计量作为报表的一部分显示，并可用来比较模型。请参见《拟合线性模型》，了解有关这些测度中的前三个测度的详细信息。

仅当在启动窗口中选中“Firth 偏倚调整估计值”复选框时才在报表中包含 –2Firth 对数似然值。该选项在默认情况下是选中的。

对于上述每一个统计量，较小的值指示较好的拟合。

JMP PRO Bayes 参数估计值

（仅当在启动窗口中选定“分层 Bayes”的情况下才出现。）“Bayes 参数估计值”报表提供模型效应的结果。估计值基于“分层 Bayes”拟合，该拟合将测试对象级别的协变量整合到似然函数中并且直接估计它们对于参数的效应。使用 Train (2001) 中所述的算法版本估计测试对象级别的协变量，该算法包含自适应 Bayes 和 Metropolis-Hastings 方法。为每个模型效应计算后验均值和方差。该算法还会提供模型效应参数的特定于测试对象的估计值。请参见第 144 页的“保存对象估计值”。

在估计过程中，向每个个体分配他/她自己的参数估计值向量，基本上将估计值视为随机效应和协变量。个体的系数向量假定来自具有任意均值和协方差矩阵的多元正态分布。在给定选择集中的特性时，给定测试对象的效用参数的似然函数基于选择集内每个测试对象的偏好的多项式 Logit 模型。给定测试对象的系数向量的先验分布呈正态分布，其均值为 0，对角协方差矩阵中每个测试对象的方差都相同。协方差矩阵假定来自逆 Wishart 分布，其尺度矩阵为具有相同对角线元素的对角矩阵。

对于每个测试对象，链最初的老化迭代次数将被废弃。默认情况下，该次数等于启动窗口中指定的“Bayes 迭代数”的一半。

图 5.16 “Bayes 参数估计值”报表

Bayes 参数估计值

项	后验均值	后验标准差	对象标准差
产地[Canada]	-0.391376793	0.1782358356	0.2110773916
国籍[Canadian]*产地[Canada]	-0.069583581	0.1821677887	0.2094958905
性别[Female]*产地[Canada]	-0.275446124	0.2082716547	0.2146128346

迭代总次数	5000
老化迭代次数	2500
响应者数目	30
老化后的平均对数似然	-196.5012

项 模型项。

后验均值 项系数的参数估计值。对于老化期之后的每次迭代，计算特定于测试对象的系数估计值的均值。后验均值是这些均值的平均值。

> **提示：**选择红色小三角选项“保存 Bayes 链”，查看每次迭代的单个估计值。

后验标准差 老化过程后经过几次迭代，特定于测试对象的估计值的均值的标准差。

对象标准差 关于后验均值的特定于测试对象的估计值的标准差。

> **提示：**选择红色小三角选项“保存测试对象估计值”以查看单个估计值。

迭代总次数 执行的迭代总次数，包括老化期。

老化迭代次数 废弃的老化迭代次数。该次数等于启动窗口中指定的“Bayes 迭代数”的一半。

响应者数目 测试对象数

老化后的平均对数似然 基于在老化期后获取的值计算的对数似然函数的平均值。

似然比检验

图 5.17 显示通过运行 Potato Chip Responses.jmp 中的脚本“口味”的 MaxDiff 获得的“似然比检验”报表。

图 5.17 似然比检验

似然比检验

源	似然比卡方	自由度	概率>卡方
口味	66.757	9	<.0001*
国籍*口味	14.609	9	0.1023
性别*口味	9.480	9	0.3942

源 列出模型中的效应。

似然比卡方 针对相应效应检验的似然比卡方统计量的值。

自由度 卡方检验的自由度。

概率 > 卡方 卡方检验的 *p* 值。

条形图 显示似然比卡方值的条形图。

MaxDiff 平台选项

“MaxDiff 模型”红色小三角菜单包含以下选项：

JMP PRO **显示 MLE 参数估计值** （可用于分层 Bayes。）显示模型项系数的非 Firth 最大似然估计值和标准误差。这些估计值用作分层 Bayes 算法的起始值。

联合因子检验 （不可用于分层 Bayes。）通过对包含该因子的所有效应构造似然比检验来检验模型中的每个因子的整体效应。有关联合因子检验的详细信息，请参见 《拟合线性模型》。

置信区间 （不可用于分层 Bayes）显示或隐藏 “参数估计值”报表中每个参数的置信区间。

置信限 （可用于分层 Bayes）为“Bayes 参数估计值”报表中的每个参数显示置信限。基于后验分布的 2.5 和 97.5 分位数构造限值。

估计值的相关性 若未选定 “分层 Bayes”，则显示或隐藏最大似然参数估计值之间的相关性。

对于分层 Bayes，则显示或隐藏参数估计值后验均值的相关性矩阵。基于老化后的迭代次数计算相关性。像处理数据表中的列那样来处理老化后每次迭代的后验均值。通过计算这些列的相关性矩阵来获取 “估计值的相关性”表。

比较 执行两个特定备择选择特征之间的比较。您可以选择感兴趣的因子值进行比较。您可以比较特定的配置：包括通过选中**任意**复选框来比较左侧或右侧的所有设置。使用**任意**并不会

对所有不同特征的组合进行比较，而是所有两两特征的组合进行比较，一次比较一个特征，对于其他因子维持左侧显示的设置。请参见第 145 页的 “比较”报表”。

“所有水平比较”报表 显示 “所有水平比较”报表，其中包含一个表，表中含有关于特征的所有配对比较的信息。若您是针对对象效应建模，则必须指定对象效应的组合，并且该表针对于对象效应的组合。表中每个单元格都显示行水平和列水平的效用差异、差值标准误差，以及无差异检验的 Wald *p* 值。

警告：未针对多重比较修正 *p* 值。仅将这些结果用作指导。

Wald *p* 值用彩色标记。饱和蓝色（对应“红色”）指示差值（行 - 列）为负（对应“正”）。红色和蓝色的强度指示显著程度。

保存效用公式 当分析针对多个数据表时，创建包含用于效用的公式列的新数据表。新数据表的行是测试对象特征和被测特征的组合，列是被测特征和测试对象效应列。当分析针对一个数据表时，将添加新的 “效用公式”列。

按对象保存梯度 （不可用于分层 Bayes。）构建一个新表，每个测试对象对应一行，其中包含测试对象水平的每个参数的似然函数的平均步长 （Hessian 比例梯度） 。这对应于使用 Lagrangian 乘数检验将该对象与其余对象分离出来。以后可以使用内置脚本对这些值聚类，以表明数据中存在的独特的市场细分。请参见 “选择模型”一章中第 107 页的 “细分示例”。

注意：当某个对象梯度不可估计时，它被设置为在 “按对象划分的梯度”表中缺失。

JMP PRO **保存对象估计值** （可用于分层 Bayes。）创建一个表，其中每行都包含每个效应的特定于测试对象的参数估计值。每个效应的特定于测试对象的参数效应的分布以“Bayes 参数估计值”报表中给定项的估计值为中心。“对象接受率”提供 Metropolis-Hastings 步骤期间得到新参数估计值的接受率。通常，接受率 0.20 被视为良好。请参见第 142 页的 “Bayes 参数估计值”。

JMP PRO **保存 Bayes 链** （可用于分层 Bayes。 ）创建一个表，其中提供关于计算特定于测试对象的 Bayes 估计时使用的迭代链的信息。请参见第 145 页的 “保存 Bayes 链”。

模型对话框 显示生成当前分析的 MaxDiff 启动窗口，使用当前分析可以修改和重新拟合模型。您可以指定新数据集、新 ID 和新模型效应。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

保存 “依据”组脚本 包含使您可以保存脚本的选项，可将为 “依据”变量的所有水平重新生成平台报表的脚本保存到多个不同的位置。仅当在启动窗口中指定 “依据”变量时才可用。

“比较”报表

指定配对比较时显示 “比较”报表。其中包含以下列：

因子 显示您指定的测试对象因子的水平。

比较 1 显示第一个比较成分中的特征变量的因子和水平。

比较 2 显示第二个比较成分中的特征变量的因子和水平。

效用 1 显示在 “因子”列中指定的测试对象的第一个成分的估计效用。

效用 2 显示在 “因子”列中指定的测试对象的第二个成分的估计效用。

概率 1 显示对于 “因子”列中指定的测试对象，相对于第二个成分更偏好第一个成分的预测概率。

概率 2 显示对于 “因子”列中指定的测试对象，相对于第一个成分更偏好第二个成分的预测概率。

优势比 1 概率 1 与概率 2 之比。

优势比 2 概率 2 与概率 1 之比。

比较差异 效用 1 与效用 2 之差。

标准差 比较差异估计值的样本标准误差。

JMP PRO 保存 Bayes 链

（可用于使用分层 Bayes 拟合的模型。）您可以使用 “Bayes 链”数据表确定估计值是否已经稳定。创建的表包含的行数等于 “Bayes 迭代数”（在启动窗口中指定）加 1。第一行 （迭代 1）提供起始值。后续行按顺序显示迭代结果。该表有一列用于迭代计数、模型对数似然，并且包含与每个模型效应对应的列：

迭代 提供迭代次数，第一行显示起始值。

对数似然 这次迭代的模型对数似然。您可以针对迭代绘制对数似然，以便查看老化期和调节期的行为。

< 模型效应 > 的自适应 Sigma 提供相应效应的逆 Wishart 分布尺度矩阵的对角线元素的平方根估计值。

< 模型效应 > 的接受率 提供相应效应的样本接受率。

< 模型效应 > 的均值 提供相应效应的估计均值。

< 模型效应 > 的方差 提供相应效应的估计方差。

第 6 章

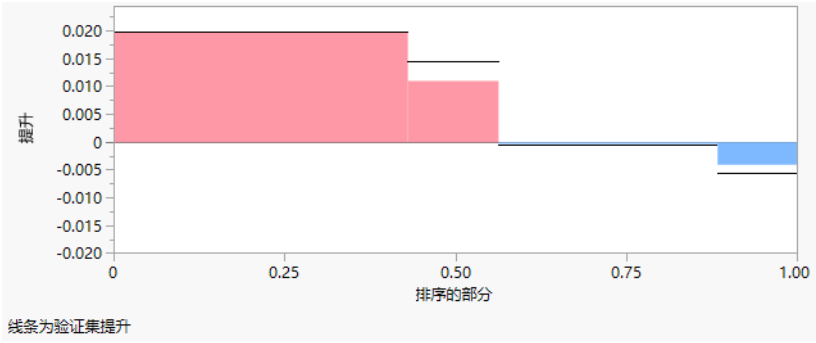
JMP PRO 提升

对市场活动对消费者行为带来的增量影响进行建模

“提升”平台仅在 JMP Pro 中提供。

使用提升建模来优化营销决策、编制个性化的医疗方案，或者更普遍来讲，用于识别可能会对某些干预做出响应的个体特征。提升建模也称为增量建模、真正的提升建模或净增量建模，它与传统建模方法的区别在于：该方法主要探索处理和其他变量之间的交互作用，其更关注那些可能对市场活动或治疗有积极响应的个体。

图 6.1 提升护发用品营销活动效果的示例



目录

“提升”平台概述 149
“提升”平台的示例 150
启动“提升”平台 152
“提升模型”报表 153
提升模型图形 153
“提升”报表选项 155

JMP PRO “提升”平台概述

使用“提升”平台对某项市场活动或**治疗**对个体的增量影响建模。提升模型帮助我们识别最可能响应某项市场活动的个体群组。特定群体的识别有助于制定高效、有针对性的决策，进而合理优化资源分配，对个体产生一定的影响。请参见 Radcliffe and Surry (2011)。

“提升”平台拟合分割模型。尽管传统分割模型选择拆分来优化分类，但提升模型选择拆分来最大化处理间的差异。

提升分割模型解释了个体按处理因子分组的事实。为了确定拆分，模型拟合每个因子的所有可能的二元拆分。拟合的模型类型取决于响应的类型。若响应变量是连续型，就构建拆分、处理以及两者交互作用的线性函数。若响应变量是分类型，就构建拆分、处理以及两者交互作用的 Logistic 函数。在以上两种情况中，交互项用于衡量两个拆分中个体组在提升中的差异。将选中最显著的拆分，之后该过程重复进行。

“提升”平台根据每个二元拆分模型中交互作用项的显著性来选择最显著的拆分。不过，仅基于 p 值的预测变量选择会导致偏向于具有许多水平的预测变量的偏倚，从而生成针对单个预测变量的许多模型。为此，JMP 调整了 p 值以考虑水平数或考虑的模型数。基于 Monte Carlo 模拟使用这种校正。请参见 Sall (2002)。拆分由最小调整后的 p 值确定，用于检验所有预测变量中所有可能的二元拆分在模型之间的交互作用效应的显著性。报表中给出了每个调整 p 值的 logworth 值，即 $-\log_{10}$(调整 p 值)。

JMP PRO “提升”平台的示例

Hair Care Product.jmp 样本数据表来自一个营销活动，该活动旨在提高男性和女性对染发产品的购买量。为了设计该研究并跟踪购买行为，我们选取了一家大型美发连锁店的 126,184 名持有“会员卡”的会员为研究对象。并从中随机抽取一半的会员，向他们发送该产品的促销信息。接下来我们追踪了所有持有会员卡的会员在随后三个月对该产品的购买行为。

数据表包含一个**促销**列，表示会员是否收到了促销资料。**购买**列表示会员在试验期内是否购买了产品。对于每个会员，我们收集了以下信息：**性别**、**年龄**、**头发颜色（自然）**、**美国地区**和**住处**（会员是否居住在城市地区）。数据表还包含一个**验证**列，包含大约 33% 的测试对象。

对于分类响应，“提升”平台会把值排序中的第一个水平当做关注的响应进行建模。这是**购买**列具有 “值顺序”列属性的原因所在，该属性确保 “是”响应排在第一位。

1. 选择**帮助** > **样本数据库**，然后打开 Hair Care Product.jmp。
2. 选择**分析** > **消费者研究** > **提升**。
3. 从 “选择列”列表：
   - 选择**促销**，然后点击**处理**。
   - 选择**购买**，然后点击 **Y，响应**。
   - 选择**性别**、**年龄**、**头发颜色**、**美国地区**和**住处**，然后点击 **X，因子**。
   - 选择**验证**并点击**验证**。
4. 点击**确定**。
5. 在显示的报表中的图形下方，点击**执行**。

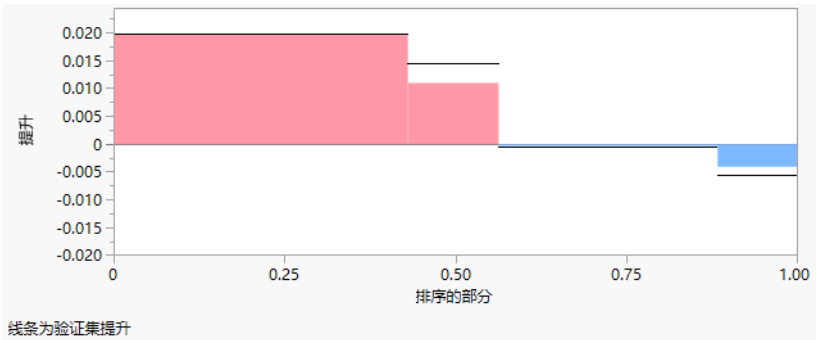
   基于验证集的最优拆分数确定为 3。请注意，左垂直尺度已锁定，以便保持右垂直轴上显示的结果的总体比率。

图 6.2 三次拆分后的图形

图中的右侧垂直轴指示购买的比例比不购买的比例小。该图形显示购买的提升发生在具有黑色、红色或棕色头发的女性群体中或具有金发的更年轻的女性（**年龄** <42）会员中。对于男性会员和年龄更大的淡黄色头发的女性会员（**年龄** ≥ 42），促销具有负面影响。

6. 点击“‘购买’的提升模型”红色小三角并选择**提升图形**。

图 6.3 提升图形



请注意，对于后面的两组测试对象（男性组和**年龄** ≥ 42 的非金发的女性组），促销具有负面影响。“提升图形”中显示的水平线绘制了验证集的图形。具体来说，首先根据验证集对决策树进行评估，然后“提升图形”根据估计的提升值构造提升图形。

JMP PRO 启动 “提升” 平台

通过选择**分析** > **消费者研究** > **提升**启动 “提升” 平台。

图 6.4 “提升” 平台启动窗口

有关 “选择列” 红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

Y，响应 分配一个或多个要分析的列。

X，因子 分配一个或多个要用作因子的列。

处理 分配一个分类处理列。若处理列包含两个以上水平，则将第一个水平视为一个处理水平，其余水平合并为第二个处理水平。

权重 分配一个数值列，其中包含数据表中每个观测的权重。仅当行的权重大于零时才在分析中包含该行。

频数 指定一个频数变量。这对于汇总数据很有用。

验证 分配用于定义验证集的数值列。该列应包含至多三个非重复值。

- 若有两个值，较小的值定义训练集，较大的值定义验证集。
- 若有三个值，这些值按大小递增的顺序定义训练集、验证集和测试集。
- 若验证列有三个以上的水平，则包含最小三个值的行定义验证集。其他所有行都从分析中排除。

“提升” 平台使用验证列来训练和调整模型，或训练、优化和评估模型。有关验证的详细信息，请参见 《预测和专业建模》。

若在 “选择列” 列表中没有选择任何列的情况下点击 “验证” 按钮，您可以向数据表添加一个验证列。请参见 《预测和专业建模》。

依据 为“依据”变量的每个水平生成单独报表。若分配了多个“依据”变量，则为“依据”变量水平的每个可能组合生成单独报表。

验证部分 要用作验证集的数据部分。输入 0 和 1 之间的值。

信息性缺失 若选定，则支持对分类预测变量进行缺失值分类，并支持对连续预测变量进行缺失值的信息性处理。

有序型限制顺序 若选定，对于有序变量，在拆分时则会进行限定以维持有序变量水平的排序。

JMP PRO “提升模型”报表

- “提升模型图形”
- ““提升”报表选项”

JMP PRO 提升模型图形

该图形的纵坐标表示响应。横坐标对应的是按拆分节点排列的观测值。对于每个节点，黑色的水平线表示平均响应。在每个拆分内，都有一个由红线或蓝线表示的处理子拆分。这些线分别表示该拆分内两个处理组各自的响应均值。处理列的值排序决定了这些线的放置顺序。拆分节点时，图形会随之更新并在水平轴下方显示拆分，同时会添加垂直线用于分隔这些拆分。

在图形下方是控制按钮：**拆分**、**剪除**和**执行**。仅当存在验证集时才显示“执行”按钮。除此之外，还显示了“处理”列的名称及其两个水平，称之为“处理1”和“处理2”。若“处理”列包含两个以上的水平，除了将第一个水平视为单独的水平外，其他所有水平统一合并为“处理2”。

“处理”列信息的右侧是一个报表，它显示与预测有关的汇总值。（请记住预测不是提升建模的目标。）随着节点的拆分报表会相应更新。若使用验证集，则显示训练集和验证集的值。

R 方 与提升树关联的回归模型的 R 方。请注意回归模型包括与处理列的交互作用。与 R 方越接近 0 相比， R 方越接近 1 表示数据拟合效果更好。

注意：较低的 R 方值表明可能有一些模型中不包含的变量可以解释无法解释的变异。不过，若数据受到大量内在变异影响，即便提升模型是有效的， R 方值也可能比较低。

RMSE 与树关联的回归模型的均方根误差 (RMSE)。仅为连续响应提供 RMSE。请参见《拟合线性模型》。

数目 观测数。

拆分数 发生拆分的次数。

AICc 使用相关的回归模型计算出的校正的 Akaike 信息准则 (AICc)。仅为连续响应提供 AICc。请参见《拟合线性模型》。

JMP PRO 提升决策树

决策树显示了构建提升模型过程中对变量进行的拆分。图 6.5 中显示了一个使用 Hair Care Product.jmp 样本数据表的示例。每个节点包含以下信息：

处理 显示处理列的名称，它包含两个水平。

比率 （仅对两水平分类响应显示。）对于每个处理水平，在该节点中响应者占测试对象的比例。

均值 （仅对连续响应显示。）对于每个处理水平，该节点中测试对象的平均响应。

计数 在指定的处理水平上该节点中的测试对象数。

t 比 在该节点中测试对象在各个处理水平上对响应差异进行显著性检验的 *t* 比。若响应是分类型，该检验将视其为连续 （值 0 和 1）。

处理差异 各个处理水平上响应均值的差异，这是以下列条件为假定前提的提升：

- 处理列的值排序中的第一个水平表示处理。
- 响应变量也定义成差异值越大处理的影响越大。

LogWorth 基于给定节点进行下一步拆分的 Logworth 的值。

图 6.5 第一次拆分的节点

“候选项”报表

每个节点还包含一个提供更多信息的 “候选项”报表：

项 模型项。

LogWorth 对给定项的所有可能拆分中最大的 Logworth 值。拆分对应的 Logworth 通过对调整 p 值取 $-\log_{10}$ 来计算。

F 比 当响应为连续型时，会提供线性回归模型中与交互作用项相关的 F 比。回归模型中指定响应为处理、二元拆分及两者交互作用的线性函数。响应为分类型时，则会提供名义型 Logistic 模型中交互作用项的卡方值。

Gamma 当响应为连续型时，该值为在计算 F 比时使用的线性回归模型中的交互作用项的系数。当响应为分类型时，该值反映了根据 Firth 调整的对数优势比构造的交互作用估计值。

截断点 若模型项为连续型，则该点为拆分点；若模型项为分类型，该点描述了第一个拆分节点（左节点）的内容。

JMP PRO “提升”报表选项

除了下面所述的选项之外，“提升”报表的其他所有红色小三角选项均在“分割”平台的相关文档中进行了介绍。有关这些选项的详细信息，请参见《预测和专业建模》。

JMP PRO 最小拆分大小

该选项显示一个窗口，您可以在其中输入一个数字或占总样本大小的比例来定义允许的最小拆分大小。要指定一个数字，请输入大于等于 1 的值。要指定占样本大小的比例，请输入小于 1 的值。“提升”平台的默认值设置为 25 或用行数除以 2,000 后的整数部分（取两者之间较大的值）。

JMP PRO 列提升贡献

该表和图显示了列对于提升树的贡献。列的贡献通过计算与其拆分关联的 F 比值之和得来。回想一下我们之前介绍的，这些值用于测量回归模型中处理和拆分节点变量交互作用项的显著性。

JMP PRO 提升图形

考虑训练集中的观测数。对某一观测值带来的提升定义为针对不同的处理水平进行预测的响应概率或响应均值的差异。这些提升值按降序排序。“提升图形”在纵坐标显示提升值。该图形在横坐标显示不同提升值下观测所占的比例。

JMP PRO 保存列

保存差异 保存观测节点的各个处理水平上响应均值的估计差值。这是对提升的估计。

保存差异公式 保存差异或提升的公式。

发布差分公式 创建差分公式并将其另存为“公式存储库”平台中的公式列脚本。若未打开“公式存储库”报表，该选项将创建 “公式存储库”报表。请参见 《预测和专业建模》。

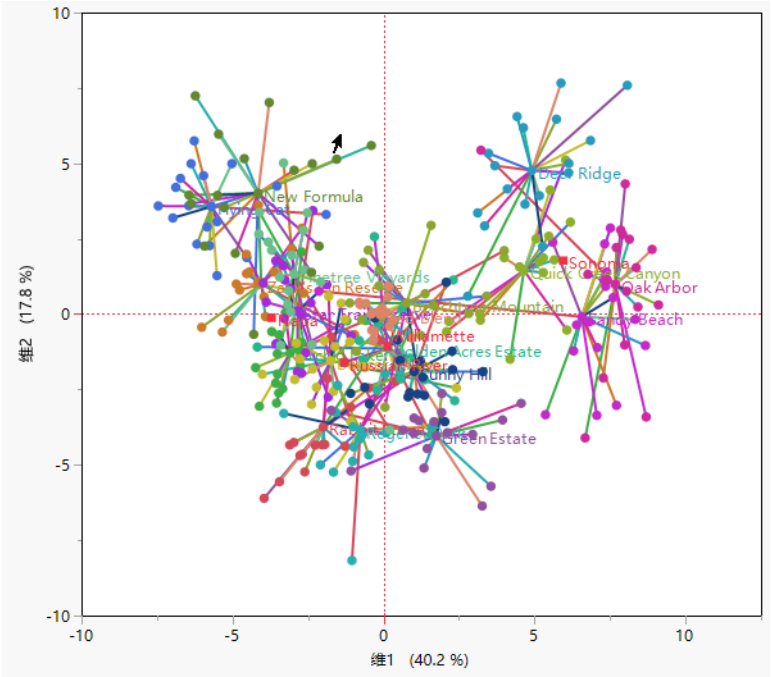
第 7 章

多因子分析

分析小组成员之间的一致性

多因子分析 (MFA) 是一种与主成分分析 (PCA) 紧密相关的分析方法。MFA 使用特征值分解将相同项的多个测量值变换为正交主成分。这些成分可以帮助您了解各项如何相似以及它们又如何不同。MFA 使用多个表或一致性 PCA 方法。

图 7.1 多因子分析中的一致性图



目录

“多因子分析”平台概述 159
多因子分析的示例 159
启动“多因子分析”平台 162
数据格式 163
“多因子分析”报表 164
汇总图 164
一致性图 165
“多因子分析”平台选项 165
“多因子分析”平台的统计详细信息 167

“多因子分析”平台概述

多因子分析 (MFA) 是一种与主成分分析 (PCA) 紧密相关的分析方法。不过， MFA 不同于 PCA，因为它将不止一个表的测量值进行了组合。这样的表有时称为子表或子矩阵。每个子表都有相同的行数，这些行表示待检验的项或产品。在 JMP 中，子表在单个数据表中表示为列组。每个列组都称为一个区组。请注意以下关于区组的信息：

- 区组中的列数可能有所不同。例如，在感官分析中，一个区组就表示一个小组成员。某些小组成员评级的产品特性可能比其他小组成员要少。
- 每个列区组都可以表示完全不同的测量值。MFA 统一每个区组的尺度，以便对所有测量值进行全局分析。

MFA 的主要目标是要查找相似的产品分组（数据表中的行）。次要目标是要标识离群值小组成员。离群值小组成员结果与组中其余部分如此不同，以致于这些结果会改变研究结果。可以使用补充变量来调查各项为何分组在一起。

您可以使用 MFA 对研究进行分析，研究中针对相同或不同特性通过不同的手段、个人或在不同情形下对项进行测量。MFA 经常用在感官分析中，用来解释小组成员之间的不同测量值。传统的感官分析可能需要数小时的先期培训，以确保小组成员的测量值彼此之间保持一致。例如，考虑一种果汁产品，其感官测量值可描述为“水果味”、“甜味”和“提神”。在传统的感官分析中，每个小组成员不得不接受培训和测试，以确保针对不同的感官测量值的报告在小组成员之间是一致的。MFA 支持研究人员对未培训的小组成员执行类似 PCA 的分析。

当您使用 MFA 时，每次都测量相同的项，所以测量值可组织到内部一致的组或区组中。对于感官分析，行是测量的项，列是每个小组成员记录的感官方面（每个小组成员对应一个区组）。缺失观测替换为列均值。

有关多因子分析的详细信息，请参见 Abdi et al. (2013)。

多因子分析的示例

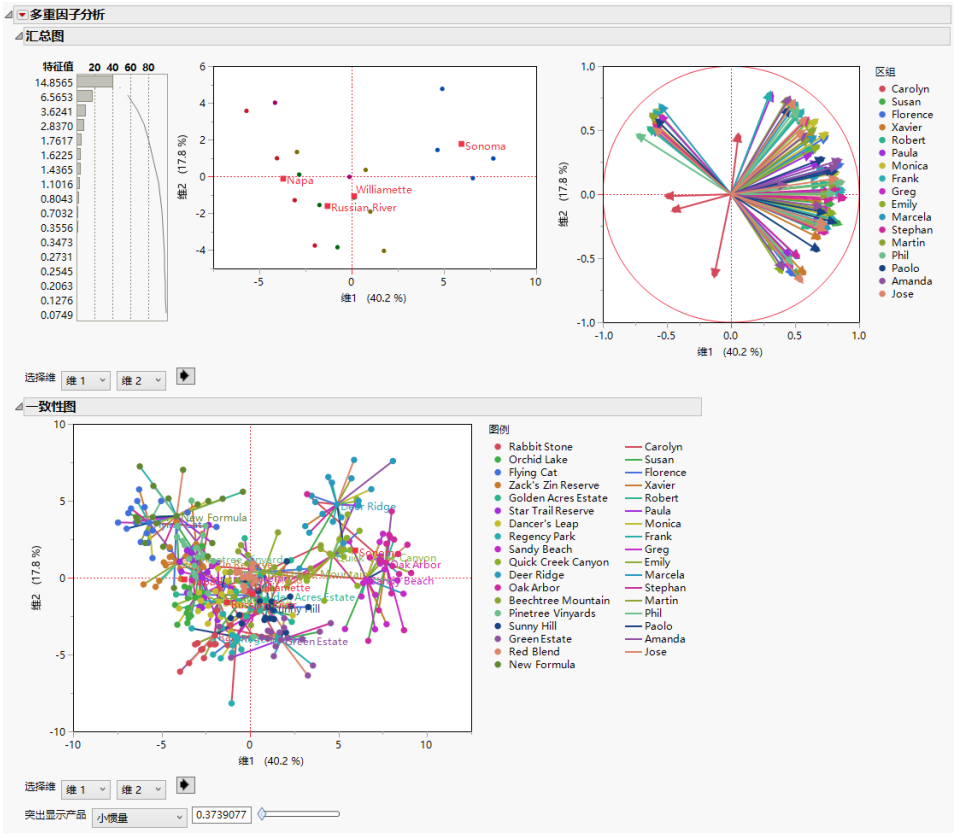
本例使用葡萄酒特征的模拟感官小组研究中的数据。参与者对 16 种葡萄酒的若干特征进行了评级，级别从 1（无强度）一直到 10（显著强度）。您想要更好地了解这 16 种葡萄酒的异同程度。

1. 选择**帮助** > **样本数据库**，然后打开 Wine Sensory Data.jmp。
2. 选择**分析** > **消费者研究** > **多因子分析**。
3. 选择**葡萄园**，然后点击**产品 ID**。
4. 选择**地区**，然后点击 **Z，补充**。
5. 选择从 Carolyn 到 Jose 之间的所有列组，然后点击**添加区组**。

注意：针对每个小组成员，该数据表中的列分组到一个区组中。对于未分组数据，选择某个区组的列，点击**添加区组**，然后为每个区组重复该操作。

6. 点击**运行模型**。

图 7.2 初始“多因子分析”报表



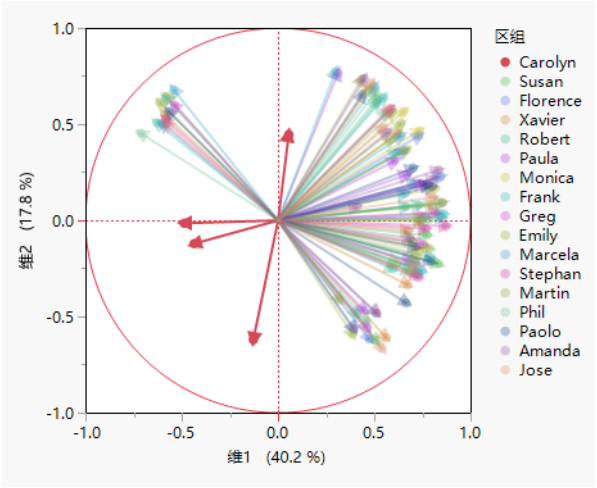
提示：要将“一致性图”中的图例设置为两列，双击该图例并将“图例设置”中的“换行/列的项数目”设置为 18。

请注意汇总图中的以下信息：

- 在前两个维的因子得分图中，葡萄酒倾向于按其所在地区聚集在一起。
- 在载荷图中，左下象限中的射线对应于 Carolyn。这些射线指示 Carolyn 与其他评级员之间存在差异。

7. 在载荷图旁边的图例中，点击 Carolyn 以突出显示其结果。

图 7.3 包含突出显示的 Carolyn 的结果的载荷图

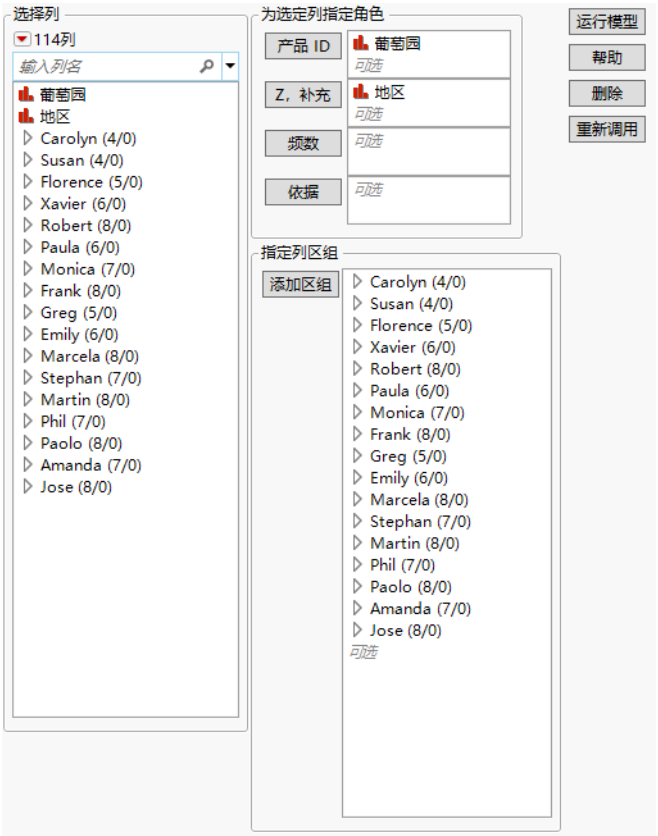


Carolyn 的结果不同于其他小组成员。您最好重新运行分析，而不包含她的结果。请参见第 164 页上的图 7.6，了解不含 Carolyn 的分析结果。

启动“多因子分析”平台

通过选择**分析** > **消费者研究** > **多因子分析**启动 “多因子分析”平台。

图 7.4 “多因子分析”启动窗口



有关“选择列”红色小三角菜单中选项的详细信息，请参见《使用 JMP》。

产品 ID 指定要分析的项或产品列。

Z，补充 指定要用作补充变量的列。您对这些变量之间的关联感兴趣，但这些变量并不包括在计算中。

频数 标识一列，该列的数值为分析中的每一行都分配一个频数。

依据 一个或多个列，其水平定义不同的分析。对于指定列的每个水平，都使用您已经指定的其他变量分析相应行。结果显示在不同的报表中。若分配了多个依据变量，则为依据变量水平的每种可能组合都生成一个单独的报表。

添加区组 用来向区组添加一个或多个列：

- 将单个列添加为单个区组。
- 将列组添加为区组。
- 将单个列添加到选定区组中。

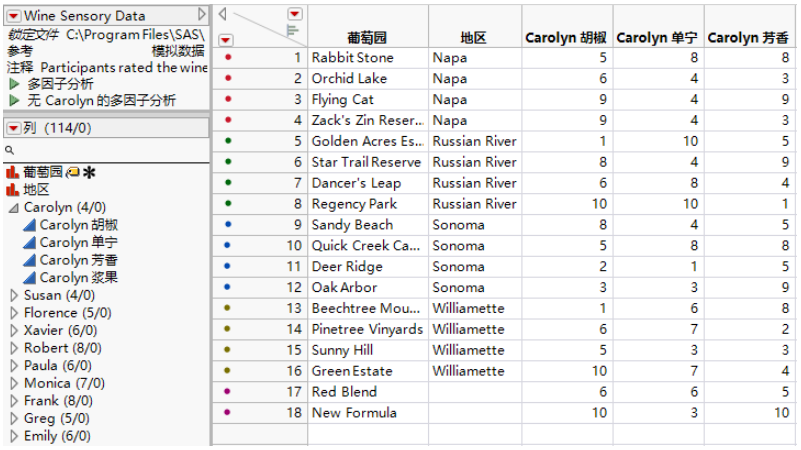
提示：若您在运行该平台之前将各列分组到区组中，您就可以选择多个列组并在一个操作中将其转换为区组。否则，您必须为每个区组选择每个列组，然后点击 “添加区组”，一次添加一个区组。双击某个区组名称对其进行更改。

数据格式

“多因子分析”平台使用包含列组或子表的数据表。每个列组（称为区组）都可有不同数目的列。每个列区组都可以表示不同的测量值。列或区组不必位于 JMP 列组中。不过，当列分组到数据表的区组中时，该平台更容易启动。

数据表行表示要测量的项。每个项的观测必须位于单个行中。例如，图 7.5 显示了一个测量来自不同葡萄园的 16 种葡萄酒的特性的表。列面板显示每个小组成员的列组或子表。没有为第 17 和 18 行中的**葡萄园**分配**地区**。该分析可用于探索葡萄园最常分配到的地区。

图 7.5 多因子分析的数据表的部分视图



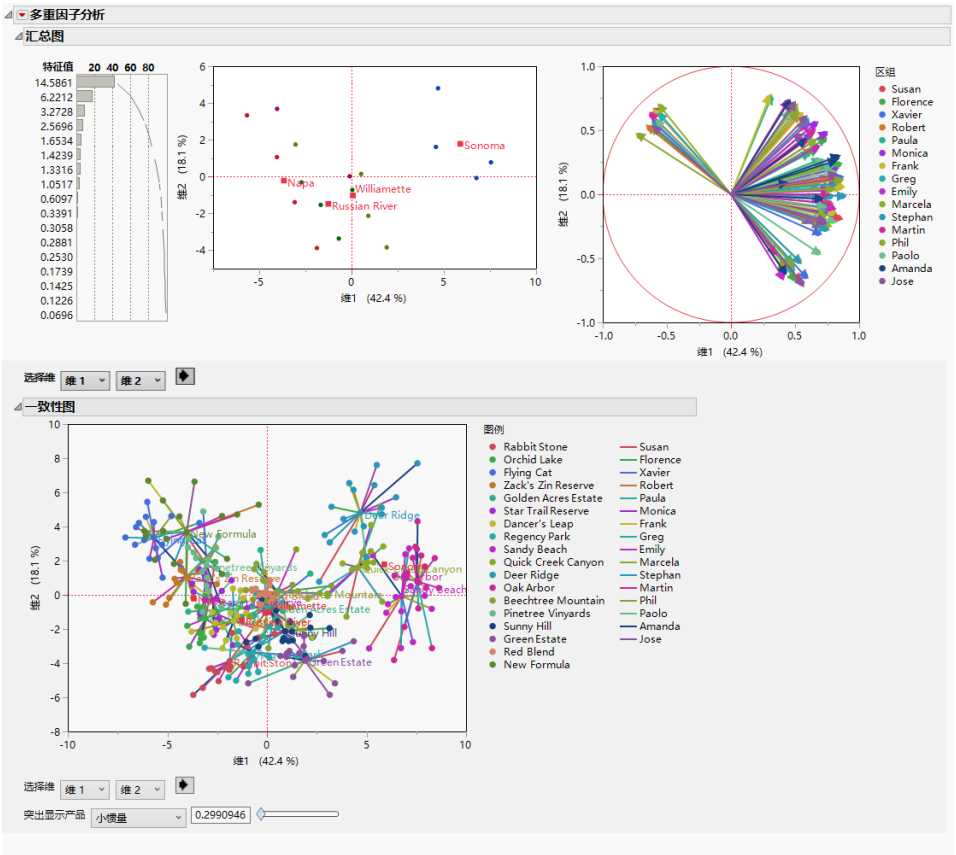
	葡萄园	地区	Carolyn 胡椒	Carolyn 单宁	Carolyn 芳香
1	Rabbit Stone	Napa	5	8	8
2	Orchid Lake	Napa	6	4	3
3	Flying Cat	Napa	9	4	9
4	Zack's Zin Reser...	Napa	9	4	3
5	Golden Acres Es...	Russian River	1	10	5
6	Star Trail Reserve	Russian River	8	4	9
7	Dancer's Leap	Russian River	6	8	4
8	Regency Park	Russian River	10	10	1
9	Sandy Beach	Sonoma	8	4	5
10	Quick Creek Ca...	Sonoma	5	8	8
11	Deer Ridge	Sonoma	2	1	5
12	Oak Arbor	Sonoma	3	3	9
13	Beechtree Mou...	Williamette	1	6	8
14	Pinetree Vinyards	Williamette	6	7	2
15	Sunny Hill	Williamette	5	3	3
16	Green Estate	Williamette	10	7	4
17	Red Blend		6	6	5
18	New Formula		10	3	10

注意：缺失观测替换为列均值。若缺失观测导致某列无变异，那么该列将从分析中排除。缺失规则应用于所有变量，包括补充变量。

“多因子分析”报表

初始 “多因子分析”报表显示一个特征值表、汇总图和一致性图。

图 7.6 “多因子分析”报表



汇总图

“汇总图”报表包含三个部分：

- 第一部分显示一致性 PCA 的特征值，同时显示每个成分所解释的方差的累积百分比图。一致性 PCA 用于获取数据区组的公共表示。一致性 PCA 是指加权子表的主成分分解，用于获取数据区组的公共表示。
- 中间部分是因子得分图，每行（项）都对应一个标记。若使用补充变量，则每个变量水平都对应一个加注了标签的标记。在该图中聚集在一起的项被视为相似。
- 第三部分是一个标绘每个区组的因子载荷的载荷图。

提示：在载荷图图例中，点击某个区组可使其在图中突出显示。

选择维 控制在得分和载荷图中标绘的维。第一个控件选择水平维，第二个控件选择垂直维。

一致性图

“一致性图”报表包含一个图，它将各小组成员的响应与所有小组成员针对每项的平均响应进行叠加。您可以使用该图调查小组成员之间的响应一致性。例如，若给定的某个小组成员的多个分数一致远离其他小组成员的平均分，则可以考虑将该小组成员从分析中排除。悬停在数据点上方以查看该点的区组标签。点击图例中的产品 ID 使其在一致性图中突出显示。

选择维 控制在一致性图上标绘的维。第一个控件选择水平维，第二个控件选择垂直维。

突出显示产品 根据项的惯量得分控制其透明度。小惯量指示小组成员一致性较好的项。大惯量指示小组成员一致性较差的项。

小惯量 突出显示惯量小于等于文本框中的值的项。要调整惯量值的截止值，请使用滑块或在文本框中输入一个值。

大惯量 突出显示惯量大于等于文本框中的值的项。要调整惯量值的截止值，请使用滑块或在文本框中输入一个值。

最小和最大惯量 突出显示具有最小和最大惯量的项。若选定该选项，文本框和滑块对结果无影响。

“多因子分析”平台选项

“多因子分析”红色小三角菜单包括以下选项。

区组权重 显示或隐藏每个区组的第一个特征值以及该特征值的权重。该权重是第一个特征值的逆平方根。

特征值 显示或隐藏特征值表，该表对应于一致性维，且按照从最大到最小的顺序对应。

特征向量 显示或隐藏每个一致性维的特征值表 （采用从左到右的顺序）。使用这些系数构成原始变量的线性组合可生成一致性主成分变量。

变量载荷 显示或隐藏每列的载荷。在主成分分析中，载荷表示变量与成分的相关性。接近零的值表示变量对一致性维几乎没有影响。

变量部分贡献 显示或隐藏包含变量的部分贡献的表。部分贡献表示每个变量为一致性维贡献的方差百分比。

变量平方余弦 显示或隐藏包含变量的平方余弦的表。对于每个变量，各个一致性维的平方余弦值之和等于 1 (100%)。平方余弦表示变量和维之间的方差重叠。

提示：对于变量载荷、变量部分贡献和变量平方余弦，接近零的值指示变量与一致性维的相关性很微弱。远离零的值指示强烈关联。可通过表值的透明度突出显示这些影响。

汇总图 显示或隐藏汇总图。汇总图包括特征值图 / 得分图和载荷图。

一致性图 显示或隐藏一致性图。请参见第 165 页的 “一致性图”。

双标图 显示或隐藏得分图和载荷图的叠加图。使用控件为该图选择任意两个维。

部分轴图 显示或隐藏部分轴图。该图显示单独区组分析和一致性主成分的 PCA 得分之间的相关性。使用控件为该图选择任意两个维。点击图例中的某个区组可使该区组在图中突出显示。

显示选项

箭头线 支持您显示或隐藏载荷图和部分轴图上的箭头。若变量数为 1000 或小于 1000，则显示箭头。若变量数超过 1000，默认情况下不显示箭头。

显示标签 显示或隐藏一致性图和双标图中所有点上的区组名称标签。显示或隐藏部分轴图中所有点上的列名标签。

提示：使用行标签标识一致性图上的重心和载荷图上的数据点。

RV 相关性 显示或隐藏区组之间平方相关性系数的矩阵。

Lg 系数 显示或隐藏区组之间相似性测度的矩阵。

区组偏相关性和一致性相关性 显示或隐藏区组偏得分和一致性主成分得分之间的相关性系数矩阵。该矩阵是个矩形，因为仅显示一致维之间的相关性。

区组部分贡献 显示或隐藏区组内的变量贡献之和。

区组部分惯量 显示或隐藏区组贡献乘以主成分的特征值再除以 100 后的结果。

区组平方余弦 显示或隐藏区组惯量的平方除以用于计算特征值的平方和。这些值的范围介于 0 和 1 之间。“区组平方余弦”可被视为每个主成分解释的区组方差的百分比。

保存单项得分 将项一致性主成分保存到数据表中的新列。若使用了一个或多个分类补充变量，该选项还将分类补充变量每个水平的单个得分保存到新数据表中。

保存单值平方余弦 将项的平方余弦保存到数据表中的新列。若使用了一个或多个分类补充变量，该选项还将分类补充变量的平方余弦保存到新数据表中。

保存单值部分贡献 将项的部分贡献保存到数据表中的新列。若使用了一个或多个分类补充变量，该选项还将分类补充部分贡献保存到新数据表中。

保存区组部分得分 将区组部分得分保存到新数据表中。

保存部分轴坐标 将部分轴坐标保存到新数据表中。

请参见《使用 JMP》中的获取有关下列选项的信息：

本地数据过滤器 显示或隐藏支持您过滤特定报表中使用的数据的本地数据过滤器。

重新运行 包含使您可以重复或重新启动分析的选项。在支持该功能的平台中，“自动重新计算”选项立即在相应报表窗口中反映您对数据表所做的更改。

保存脚本 包含的选项支持您保存可将报表重现到若干目标的脚本。

“多因子分析”平台的统计详细信息

多因子分析将各个子表的信息组合到一组正交列中，这些列描述该表行中的各项。基本过程如下：

- 对每个子表执行 PCA。
- 记录每个子表的第一个特征值以创建权重矩阵。
- 并排拼接各子表，对矩阵进行中心化和标准化。
- 通过奇异值分解对拼接表执行广义 PCA。广义 PCA 用于通过子表权重来限制解。

这将生成广义右奇异向量、广义左奇异向量和奇异值这三个矩阵。随后使用这些矩阵得出跨各个子表的一致性的成分得分、特征值和成分载荷。这三个矩阵是将许多列从原始测量值分解为若干可解释的维的结果，这些维解释所测量的对象之间的相似性和差异。

计算

对于 MFA， **X** 矩阵的奇异值分解可定义如下：

$$X = P\Delta Q^T \qquad \text{带有约束} \qquad P^T MP = Q^T AQ = I$$

矩阵用法如下所示：

X 是子表的 $n \times p$ 中心化和标准化矩阵。在消费者研究中，有 n 个产品和 p 个小组成员评级。

Q 是右奇异向量的 $p \times q$ 矩阵，这些向量经过 MFA 奇异值加权，可获取 q 个主成分的载荷。

Δ 是广义 PCA 的奇异值的 $q \times q$ 对角矩阵。与 PCA 一样，平方奇异值的量值（或特征值）表示每个主成分在组合分析中的重要性。

P 是左奇异向量的 $n \times q$ 矩阵，这些向量经过 MFA 奇异值加权，可获取折衷的 q 个主成分。

M 是量权重的 $n \times n$ 对角矩阵。

A 是区组或小组成员权重的 $p \times p$ 对角矩阵。

有关多因子分析的详细信息，请参见 Abdi et al. (2013)。

量权重

JMP 计算使用 $N - 1$ 进行量权重计算。这些计算影响单个和区组部分得分。

附录 A

参考书目

Abdi, H., Williams, L. J., and Valentin, D. (2013).“Multiple factor analysis: principal component analysis for multitable and multiblock data sets.”*WIREs Comp Stat*, 5:149–179. doi: 10.1002/wics.1246.

Benjamini, Y., and Hochberg, Y. (1995).“Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.”*Journal of the Royal Statistical Society, Series B* 57:289–300.

Breslow, N. E., and Day, N. E. (1980).*The Analysis of Case-Control Studies*.Statistical Methods in Cancer Research, IARC Scientific Publications, vol. 1, no. 32.Lyon: International Agency for Research on Cancer.

Firth, D. (1993).“Bias Reduction of Maximum Likelihood Estimates.”Biometrika 80:27–38.

Heinze, G., and Schemper, M. (2002).“A Solution to the Problem of Separation in Logistic Regression.”*Statistics in Medicine* 21:2409–2419.

Kish, L. (1965).*Survey Sampling*.New York: John Wiley & Sons.

Lavassani, K. M., Movahedi, B., and Kumar, V. (2009).“Developments in Analysis of Multiple Response Survey Data in Categorical Data Analysis: The Case of Enterprise System Implementation in Large North American Firms.”*Journal of Applied Quantitative Methods* 4:45–53.

Louviere, J. J., Flynn, T. N., and Marley, A. A. (2015).*Best-Worst Scaling: Theory, Methods and Applications*.Cambridge: Cambridge University Press.

McFadden, D. (1974).“Conditional Logit Analysis of Qualitative Choice Behavior.” in *Frontiers in Econometrics* edited by P. Zarembka, 105–142.New York: Academic Press.Accessed April 25, 2016. https://eml.berkeley.edu/reprints/mcfadden/zarembka.pdf.

Radcliffe, N. J., and Surry, P. D. (2011). *Real-World Uplift Modeling with Significance-Based Uplift Trees*.Portrait Technical Report TR-2011-1, Stochastic Solutions.Accessed January 13, 2017. http://www.stochasticsolutions.com/pdf/sig-based-up-trees.pdf.

Rossi, P. E., Allenby, G. M., and McCulloch, R. (2005).*Bayesian Statistics and Marketing*. Chichester, UK: John Wiley & Sons.

Sall, J. (2002).“Monte Carlo Calibration of Distributions of Partition Statistics.”SAS Institute Inc.,Cary, NC.Accessed July 29, 2015. https://www.jmp.com/content/dam/jmp/documents/en/white-papers/montecarlocal.pdf.

SAS Institute Inc.(2020).“The TTEST Procedure.”In *SAS/STAT 15.2 User's Guide*.Cary NC: SAS Institute Inc. https://go.documentation.sas.com/api/docsets/statug/15.2/content/ttest.pdf.

Train, K. E. (2001).“A Comparison of Hierarchical Bayes and Maximum Simulated Likelihood for Mixed Logit.”Department of Economics, University of California, Berkley.Accessed January 13, 2017. https://eml.berkeley.edu/~train/compare.pdf.

Train, K. E. (2009).*Discrete Choice Methods and Simulation*.2nd ed. Cambridge University Press.

Westfall, P. H., Tobias, R. D., and Wolfinger, R. D. (2011).*Multiple Comparisons and Multiple Tests Using SAS*.2nd ed. Cary, NC: SAS Institute Inc.

附录 B

技术许可声明

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved.Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth.Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org).All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.The origin of this software must not be misrepresented; you must not claim that you wrote the original software.If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3.Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4.The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents.See www.mathworks.com/patents.MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks.Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter.All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard.All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California.All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3.Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group.NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois.All rights reserved.DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS “AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated.All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives.The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1.Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3.The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear.The contents of the NOTICE file are for informational purposes only and do not modify the License.You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License.You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2020, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file.All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Version 16

JSL Syntax Reference

"The real voyage of discovery consists not in seeking new landscapes, but in having new eyes."

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.1

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020–2021. *JMP® 16 JSL Syntax Reference*. Cary, NC: SAS Institute Inc.

JMP® 16 JSL Syntax Reference

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

March 2021

August 2021

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement. For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

Get the Most from JMP

Whether you are a first-time or a long-time user, there is always something to learn about JMP.

Visit JMP.com to find the following:

- live and recorded webcasts about how to get started with JMP
- video demos and webcasts of new features and advanced techniques
- details on registering for JMP training
- schedules for seminars being held in your area
- success stories showing how others use JMP
- the JMP user community, resources for users including examples of add-ins and scripts, a forum, blogs, conference information, and so on

https://www.jmp.com/getstarted

Contents

JSL Syntax Reference

1 Learn about JMP 9

Documentation and Additional Resources

Formatting Conventions in JMP Documentation 11
JMP Help 12
JMP Documentation Library 12
Additional Resources for Learning JMP 18
JMP Tutorials 18
Sample Data Tables 18
Learn about Statistical and JSL Terms 19
Learn JMP Tips and Tricks 19
JMP Tooltips 19
JMP User Community 20
Free Online Statistical Thinking Course 20
JMP New User Welcome Kit 20
Statistics Knowledge Portal 20
JMP Training 20
JMP Books by Users 21
The JMP Starter Window 21
JMP Technical Support 21

2 JSL Functions 23

Summary of Functions, Operators, and Messages

Assignment Functions 25
Character Functions 29
Character Pattern Functions 44
Comment Functions 56
Comparison Functions 57
Conditional and Logical Functions 62
Constant Functions 75
Date and Time Functions 76
Discrete Probability Functions 85
Display Functions 91
Expression Functions 127

File Functions ... 129
Financial Functions ... 150
Graphics Functions ... 155
HTTP Functions ... 175
List Functions ... 175
MATLAB Integration Functions ... 182
MATLAB JSL Function Interfaces ... 182
Matrix Functions ... 189
Numeric Functions ... 213
Optimization Functions ... 217
Probability Functions ... 220
Programming Functions ... 249
Python Integration Functions ... 267
R Integration Functions ... 273
Random Functions ... 278
Row Functions ... 289
Row State Functions ... 295
SAS Integration Functions ... 298
SQL Functions ... 317
Statistical Functions ... 319
Transcendental Functions ... 334
Trigonometric Functions ... 340
Utility Functions ... 343

3 JSL Messages ... 377
Summary of Messages for Objects and Display Boxes

Alpha Shape ... 380
Associative Arrays ... 380
Classes ... 381
Data Tables ... 383
Columns ... 414
Rows ... 423
Data Filter ... 424
Data Feed (Windows Only) ... 427
Display Boxes ... 430
All Display Boxes ... 430
Axis Boxes ... 440
Border Boxes ... 445
Data Browser Boxes ... 446
Data Filter Source Boxes ... 446

Frame Boxes ... 446
Display 3D Boxes ... 448
Excerpt Boxes ... 449
Filter Col Selector ... 449
Global Boxes ... 449
Hier Boxes ... 449
Matrix Boxes ... 449
Nom Axis Boxes ... 450
Number Col Boxes ... 451
Number Col Edit Boxes ... 454
Number Edit Box ... 454
Outline Boxes ... 455
Panel Boxes ... 456
Plot Col Boxes ... 456
Slider Boxes and Range Slider Boxes ... 457
String Col Boxes ... 458
Tab Boxes ... 459
Table Boxes ... 460
Text Boxes ... 463
Tree Node and Tree Box ... 464
Triangulation ... 466
Windows ... 468
Dynamic Link Libraries (DLLs) ... 470
HTML 5 ... 472
Web Report ... 472
Images ... 473
JMP Applications ... 476
JMP App ... 477
JMP App Module ... 478
JMP App Module Instance ... 479
MATLAB ... 479
Namespaces ... 483
Platforms ... 485
Bubble Plot ... 490
DOE ... 491
Partition ... 491
Response Screening ... 492
Tabulate ... 492
Python Integration Messages ... 493
R Integration Messages ... 497

SAS Integration Messages 500
Metadata Server Objects 500
SAS Server Objects 501
Stored Processes 510
SAS Results 516
Schedule 519
Segments 520
Sockets 520
SQL 523
Other Objects 527
Zip Archives 527
Journals 528

A SQL Functions Available for JMP Queries 529

Numeric SQL Functions 531
Date-Time SQL Functions 532
String SQL Functions 535
System SQL Functions 536
Aggregate SQL Functions 536

B References 539

C Technology License Notices 541

Chapter 1

Learn about JMP

Documentation and Additional Resources

Learn about JMP documentation, such as book conventions, descriptions of each JMP document, the Help system, and where to find additional support.

Contents

Formatting Conventions in JMP Documentation . . . 11
JMP Help . . . 12
JMP Documentation Library . . . 12
Additional Resources for Learning JMP . . . 18
JMP Tutorials . . . 18
Sample Data Tables . . . 18
Learn about Statistical and JSL Terms . . . 19
Learn JMP Tips and Tricks . . . 19
JMP Tooltips . . . 19
JMP User Community . . . 20
Free Online Statistical Thinking Course . . . 20
JMP New User Welcome Kit . . . 20
Statistics Knowledge Portal . . . 20
JMP Training . . . 20
JMP Books by Users . . . 21
The JMP Starter Window . . . 21
JMP Technical Support . . . 21

Formatting Conventions in JMP Documentation

These conventions help you relate written material to information that you see on your screen:

- Sample data table names, column names, pathnames, filenames, file extensions, and folders appear in Helvetica (or sans-serif online) font.
- Code appears in `Lucida Sans Typewriter` (or monospace online) font.
- Code output appears in *`Lucida Sans Typewriter`* italic (or monospace italic online) font and is indented farther than the preceding code.
- **Helvetica bold** formatting (or bold sans-serif online) indicates items that you select to complete a task:
  - buttons
  - check boxes
  - commands
  - list names that are selectable
  - menus
  - options
  - tab names
  - text boxes
- The following items appear in italics:
  - words or phrases that are important or have definitions specific to JMP
  - book titles
  - variables
- Features that are for JMP Pro only are noted with the JMP Pro icon JMP PRO. For an overview of JMP Pro features, visit https://www.jmp.com/software/pro.

Note: Special information and limitations appear within a Note.

Tip: Helpful information appears within a Tip.

JMP Help

JMP Help in the Help menu enables you to search for information about JMP features, statistical methods, and the JMP Scripting Language (or *JSL*). You can open JMP Help in several ways:

- Search and view JMP Help on Windows by selecting **Help > JMP Help**.
- On Windows, press the F1 key to open the Help system in the default browser.
- Get help on a specific part of a data table or report window. Select the Help tool from the **Tools** menu and then click anywhere in a data table or report window to see the Help for that area.
- Within a JMP window, click the **Help** button.

Note: The JMP Help is available for users with Internet connections. Users without an Internet connection can search all books in a PDF file by selecting **Help > JMP Documentation Library**. See "JMP Documentation Library" for more information.

JMP Documentation Library

The Help system content is also available in one PDF file called *JMP Documentation Library*. Select **Help > JMP Documentation Library** to open the file. You can also download the Documentation PDF Files add-in if you prefer searching individual PDF files of each document in the JMP library. Download the available add-ins from https://community.jmp.com.

The following table describes the purpose and content of each document in the JMP library.

Document Title	Document Purpose	Document Content
Discovering JMP	If you are not familiar with JMP, start here.	Introduces you to JMP and gets you started creating and analyzing data. Also learn how to share your results.
Using JMP	Learn about JMP data tables and how to perform basic operations.	Covers general JMP concepts and features that span across all of JMP, including importing data, modifying columns properties, sorting data, and connecting to SAS.

Document Title	Document Purpose	Document Content
Basic Analysis	Perform basic analysis using this document.	Describes these Analyze menu platforms: • Distribution • Fit Y by X • Tabulate • Text Explorer Covers how to perform bivariate, one-way ANOVA, and contingency analyses through Analyze > Fit Y by X. How to approximate sampling distributions using bootstrapping and how to perform parametric resampling with the Simulate platform are also included.
Essential Graphing	Find the ideal graph for your data.	Describes these Graph menu platforms: • Graph Builder • Scatterplot 3D • Contour Plot • Bubble Plot • Parallel Plot • Cell Plot • Scatterplot Matrix • Ternary Plot • Treemap • Chart • Overlay Plot The book also covers how to create background and custom maps.
Profilers	Learn how to use interactive profiling tools, which enable you to view cross-sections of any response surface.	Covers all profilers listed in the Graph menu. Analyzing noise factors is included along with running simulations using random inputs.

Document Title	Document Purpose	Document Content
Design of Experiments Guide	Learn how to design experiments and determine appropriate sample sizes.	Covers all topics in the DOE menu.
Fitting Linear Models	Learn about Fit Model platform and many of its personalities.	Describes these personalities, all available within the Analyze menu Fit Model platform: • Standard Least Squares • Stepwise • Generalized Regression • Mixed Model • MANOVA • Loglinear Variance • Nominal Logistic • Ordinal Logistic • Generalized Linear Model

Document Title	Document Purpose	Document Content
Predictive and Specialized Modeling	Learn about additional modeling techniques.	Describes these Analyze > Predictive Modeling menu platforms: • Neural • Partition • Bootstrap Forest • Boosted Tree • K Nearest Neighbors • Naive Bayes • Support Vector Machines • Model Comparison • Model Screening • Make Validation Column • Formula Depot Describes these Analyze > Specialized Modeling menu platforms: • Fit Curve • Nonlinear • Functional Data Explorer • Gaussian Process • Time Series • Matched Pairs Describes these Analyze > Screening menu platforms: • Modeling Utilities • Response Screening • Process Screening • Predictor Screening • Association Analysis • Process History Explorer

Document Title	Document Purpose	Document Content
Multivariate Methods	Read about techniques for analyzing several variables simultaneously.	Describes these Analyze > Multivariate Methods menu platforms: • Multivariate • Principal Components • Discriminant • Partial Least Squares • Multiple Correspondence Analysis • Structural Equation Models • Factor Analysis • Multidimensional Scaling • Item Analysis Describes these Analyze > Clustering menu platforms: • Hierarchical Cluster • K Means Cluster • Normal Mixtures • Latent Class Analysis • Cluster Variables
Quality and Process Methods	Read about tools for evaluating and improving processes.	Describes these Analyze > Quality and Process menu platforms: • Control Chart Builder and individual control charts • Measurement Systems Analysis • Variability / Attribute Gauge Charts • Process Capability • Model Driven Multivariate Control Chart • Legacy Control Charts • Pareto Plot • Diagram • Manage Spec Limits • OC Curves

Document Title	Document Purpose	Document Content
Reliability and Survival Methods	Learn to evaluate and improve reliability in a product or system and analyze survival data for people and products.	Describes these Analyze > Reliability and Survival menu platforms: • Life Distribution • Fit Life by X • Cumulative Damage • Recurrence Analysis • Degradation • Destructive Degradation • Reliability Forecast • Reliability Growth • Reliability Block Diagram • Repairable Systems Simulation • Survival • Fit Parametric Survival • Fit Proportional Hazards
Consumer Research	Learn about methods for studying consumer preferences and using that insight to create better products and services.	Describes these Analyze > Consumer Research menu platforms: • Categorical • Choice • MaxDiff • Uplift • Multiple Factor Analysis
Scripting Guide	Learn about taking advantage of the powerful JMP Scripting Language (JSL).	Covers a variety of topics, such as writing and debugging scripts, manipulating data tables, constructing display boxes, and creating JMP applications.
JSL Syntax Reference	Read about many JSL functions on functions and their arguments, and messages that you send to objects and display boxes.	Includes syntax, examples, and notes for JSL commands.

Additional Resources for Learning JMP

In addition to reading JMP help, you can also learn about JMP using the following resources:

- "JMP Tutorials"
- "Sample Data Tables"
- "Learn about Statistical and JSL Terms"
- "Learn JMP Tips and Tricks"
- "JMP Tooltips"
- "JMP User Community"
- "Free Online Statistical Thinking Course"
- "JMP New User Welcome Kit"
- "Statistics Knowledge Portal"
- "JMP Training"
- "JMP Books by Users"
- "The JMP Starter Window"

JMP Tutorials

You can access JMP tutorials by selecting **Help > Tutorials**. The first item on the **Tutorials** menu is **Tutorials Directory**. This opens a new window with all the tutorials grouped by category.

If you are not familiar with JMP, start with the **Beginners Tutorial**. It steps you through the JMP interface and explains the basics of using JMP.

The rest of the tutorials help you with specific aspects of JMP, such as designing an experiment and comparing a sample mean to a constant.

Sample Data Tables

All of the examples in the JMP documentation suite use sample data. Select **Help > Sample Data Library** to open the sample data directory.

To view an alphabetized list of sample data tables or view sample data within categories, select **Help > Sample Data**.

Sample data tables are installed in the following directory:

On Windows: C:\Program Files\SAS\JMP\16\Samples\Data

On macOS: \Library\Application Support\JMP\16\Samples\Data

In JMP Pro, sample data is installed in the JMPPRO (rather than JMP) directory.

To view examples using sample data, select **Help > Sample Data** and navigate to the Teaching Resources section. To learn more about the teaching resources, visit https://jmp.com/tools.

Learn about Statistical and JSL Terms

For help with statistical terms, select Help > Statistics Index. For help with JSL scripting and examples, select **Help > Scripting Index**.

Statistics Index Provides definitions of statistical terms.

Scripting Index Lets you search for information about JSL functions, objects, and display boxes. You can also edit and run sample scripts from the Scripting Index and get help on the commands.

Learn JMP Tips and Tricks

When you first start JMP, you see the Tip of the Day window. This window provides tips for using JMP.

To turn off the Tip of the Day, clear the **Show tips at startup** check box. To view it again, select **Help > Tip of the Day**. Or, you can turn it off using the Preferences window.

JMP Tooltips

JMP provides descriptive tooltips (or *hover labels*) when you hover over items, such as the following:

- Menu or toolbar options
- Labels in graphs
- Text results in the report window (move your cursor in a circle to reveal)
- Files or windows in the Home Window
- Code in the Script Editor

Tip: On Windows, you can hide tooltips in the JMP Preferences. Select **File > Preferences > General** and then deselect **Show menu tips**. This option is not available on macOS.

JMP User Community

The JMP User Community provides a range of options to help you learn more about JMP and connect with other JMP users. The learning library of one-page guides, tutorials, and demos is a good place to start. And you can continue your education by registering for a variety of JMP training courses.

Other resources include a discussion forum, sample data and script file exchange, webcasts, and social networking groups.

To access JMP resources on the website, select **Help > JMP User Community** or visit https://community.jmp.com.

Free Online Statistical Thinking Course

Learn practical statistical skills in this free online course on topics such as exploratory data analysis, quality methods, and correlation and regression. The course consists of short videos, demonstrations, exercises, and more. Visit https://www.jmp.com/statisticalthinking.

JMP New User Welcome Kit

The JMP New User Welcome Kit is designed to help you quickly get comfortable with the basics of JMP. You'll complete its thirty short demo videos and activities, build your confidence in using the software, and connect with the largest online community of JMP users in the world. Visit https://www.jmp.com/welcome.

Statistics Knowledge Portal

The Statistics Knowledge Portal combines concise statistical explanations with illuminating examples and graphics to help visitors establish a firm foundation upon which to build statistical skills. Visit https://www.jmp.com/skp.

JMP Training

SAS offers training on a variety of topics led by a seasoned team of JMP experts. Public courses, live web courses, and on-site courses are available. You might also choose the online e-learning subscription to learn at your convenience. Visit https://www.jmp.com/training.

JMP Books by Users

Additional books about using JMP that are written by JMP users are available on the JMP website. Visit https://www.jmp.com/books.

The JMP Starter Window

The JMP Starter window is a good place to begin if you are not familiar with JMP or data analysis. Options are categorized and described, and you launch them by clicking a button. The JMP Starter window covers many of the options found in the Analyze, Graph, Tables, and File menus. The window also lists JMP Pro features and platforms.

- To open the JMP Starter window, select **View** (**Window** on macOS) **> JMP Starter**.
- To display the JMP Starter automatically when you open JMP on Windows, select **File > Preferences > General**, and then select **JMP Starter** from the Initial JMP Window list. On macOS, select **JMP > Preferences > Initial JMP Starter Window**.

JMP Technical Support

JMP technical support is provided by statisticians and engineers educated in SAS and JMP, many of whom have graduate degrees in statistics or other technical disciplines.

Many technical support options are provided at https://www.jmp.com/support, including the technical support phone number.

Chapter 2

JSL Functions

Summary of Functions, Operators, and Messages

This topic provides abbreviated descriptions for many of JMP's functions, operators, and general object messages. For complete information about functions, see the JMP Scripting Index. In JMP, select **Help > Scripting Index**.

For information about platform messages, see the *Scripting Guide*.

Contents

Assignment Functions . . . 25
Character Functions . . . 28
Character Pattern Functions . . . 43
Comment Functions . . . 53
Comparison Functions . . . 54
Conditional and Logical Functions . . . 59
Constant Functions . . . 71
Date and Time Functions . . . 72
Discrete Probability Functions . . . 80
Display Functions . . . 86
Expression Functions . . . 120
File Functions . . . 122
Financial Functions . . . 144
Graphics Functions . . . 149
HTTP Functions . . . 168
List Functions . . . 169
MATLAB Integration Functions . . . 175
    MATLAB JSL Function Interfaces . . . 175
Matrix Functions . . . 182
Numeric Functions . . . 206
Optimization Functions . . . 210
Probability Functions . . . 213
Programming Functions . . . 242
Python Integration Functions . . . 260
R Integration Functions . . . 266
Random Functions . . . 271
Row Functions . . . 281
Row State Functions . . . 287
SAS Integration Functions . . . 290
SQL Functions . . . 309
Statistical Functions . . . 312
Transcendental Functions . . . 326
Trigonometric Functions . . . 332
Utility Functions . . . 335

Assignment Functions

JSL also provides operators for in-place arithmetic, or *assignment operators*. These operations are all done in place, meaning that the result of the operation is assigned to the first argument. The most basic assignment operator is the = operator (or the equivalent function `Assign`). For example, if *a* is 3 and you do `a+=4`, then *a* becomes 7.

The first argument to an assignment function must be something capable of being assigned (an *L-value*). You cannot do something like `3+=4`, because 3 is just a value and cannot be reassigned. However, you can do something like `a+=4`, because *a* is a variable whose value you can set.

`Add To(`*a*`, `*b*`)`

a+=*b*

Description

Adds *a* and *b* and places the sum into *a*.

Returns

The sum.

Arguments

a Must be a variable.

b Can be a variable, a list, a number, or a matrix.

Notes

The first argument must be a variable, because its value must be able to accept a value change. A number as the first argument produces an error.

For `Add To()`: Only two arguments are permitted. If one or no argument is specified, `Add To()` returns a missing value. Any arguments after the first two are ignored.

For `a+=b`: More than two arguments can be strung together. JMP evaluates pairs from right to left, and each sum is placed in the left-hand variable. All arguments except the last must be a variable.

Example

```
a+=b+=c
```

JMP adds *b* and *c* and places the sum into *b*. Then JMP adds *a* and *b* and places the sum into *a*.

`Assign(a, b)`

a=b

Description

Places the value of *b* into *a*.

Returns

The new value of *a*.

Arguments

a Must be a variable.

b Can be a variable, number, or matrix.

Notes

a must be a variable, because it must be able to accept a value change. A number as the first argument produces an error. If *b* is some sort of expression, it's evaluated first and the result is placed into *a*.

`Divide To(a, b)`

a/=b

Description

Divides *a* by *b* and places the result into *a*.

Returns

The quotient.

Arguments

a Must be a variable.

b Can be a variable, number, or matrix.

`Multiply To(a, b)`

a=b*

Description

Multiplies *a* and *b* and places the product into *a*.

Returns

The product.

Arguments

a Must be a variable.

b Can be a variable, number, or matrix.

Notes

The first argument must be a variable, because its value must be able to accept a value change. A number as the first argument produces an error.

For `Multiply To()`: Only two arguments are permitted. If one or no argument is specified, `Multiply To()` returns a missing value. Any arguments after the first two are ignored.

For *a*`*=`*b*: More than two arguments can be strung together. JMP evaluates pairs from right to left, and each product is placed in the left-hand variable. All arguments except the last must be a variable.

Example

```
a*=b*=c
```

JMP multiplies *b* and *c* and places the product into *b*. Then JMP multiplies *a* and *b* and places the product into *a*.

`PostDecrement(`*a*`)`

a`--`

Description

Post-decrement. Subtracts 1 from *a* and places the difference into *a*.

Returns

a-1

Argument

a Must be a variable.

Notes

If *a*`--` or `Post Decrement(`*a*`)` is inside another expression, the expression is evaluated first, and then the decrement operation is performed. This expression is mostly used for loop control.

`Post Increment(`*a*`)`

a`++`

Description

Post-increment. Adds 1 to *a* and places the sum into *a*.

Returns

a+1

Argument

a Must be a variable.

Notes

If `a++` or `Post Increment(a)` is inside another expression, the expression is evaluated first, and then the increment operation is performed. Mostly used for loop control.

Subtract To(*a*, *b*)

a-=*b*

Description

Subtracts *b* from *a* and places the difference into *a*.

Returns

The difference.

Arguments

a Must be a variable.

b Can be a variable, number, or matrix.

Notes

The first argument must be a variable, because its value must be able to accept a value change. A number as the first argument produces an error.

For SubtractTo(): Only two arguments are permitted. If fewer than two or more than two arguments is specified, SubtractTo() returns a missing value.

For *a*-=*b*: More than two arguments can be strung together. JMP evaluates pairs from right to left, and each difference is placed in the left-hand variable. All arguments except the last must be a variable.

Example

```
a-=b-=c
```

JMP subtracts *c* from *b* and places the difference into *b*. Then JMP subtracts *b* from *a* and places the difference into *a*.

Character Functions

Most character functions take character arguments and quoted return character strings, although some take numeric arguments or return numeric data. Arguments that are literal character strings must be enclosed in quotation marks.

BLOB To Char(*blob*, <Encoding="*enc*">)

Description

Reinterpret binary data as a quoted Unicode string.

Returns

A quoted string.

Required Argument

blob A binary large object.

Optional Argument

Encoding A quoted string that specifies an encoding. The default encoding for the character string is "utf-8". "utf-16le", "utf-16be", "us-ascii", "iso-8859-1", "ascii~hex", "shift-jis", and "euc-jp" are also supported.

Notes

The optional argument "ascii~hex" is intended to make conversions of blobs containing normal ASCII data simpler when the data might contain CR, LF, or TAB characters (for example) and those characters do not need any special attention.

BLOB To Matrix(*blob*, *type*, *bytes*, *endian*, <*nCols*>)

Description

Creates a matrix by converting each byte in the blob to numbers.

Returns

A matrix that represents the blob.

Required Arguments

blob A blob or reference to a blob.

type A quoted string that contains the named type of number. The options are "int", "uint", or "float".

bytes Byte size of the data in the blob. Options are 1, 2, 4, or 8.

endian The quoted endian-ness of your system: "Big" (the first byte is most significant), "Little" (the first byte is the least significant), or "Native" (the machine's native format).

Optional Argument

<*nCols*> The number of columns in the matrix. The default value is 1.

Char(*x*, <*width*>, <*decimal*>, < <<Use Locale(*Boolean*)>)

Description

Converts an expression or numeric value into a quoted character string.

Returns

A quoted string.

Required Argument

x an expression or a numeric value. An expression must be quoted with Expr(). Otherwise, its evaluated value is converted to a quoted string.

Optional Arguments

width A number that sets the maximum number of characters in the quoted string.

decimal A number that sets the maximum number of places after the decimal that is included in the quoted string.

Use Locale(*Boolean*) Preserves locale-specific numeric formatting.

Example

```
Char( Pi(), 10, 4)
   "3.1416"

Char( Pi(), 3, 4)
   "3.1"
```

Notes

The *width* argument overrides the *decimal* argument.

Char To BLOB(*string*, <encoding="*enc*">)

Description

Converts a quoted string of characters into a binary (blob).

Returns

A binary object.

Required Argument

string A quoted string or a reference to a string.

Optional Argument

encoding A quoted string that specifies an encoding. The default encoding for the blob is "utf-8". "utf-16le", "utf-16be", "us-ascii", "iso-8859-1", "ascii~hex", "shift-jis", and "euc-jp" are also supported.

Notes

Converting BLOBs into printable format escapes \ (in addition to ~ " ! and characters outside of the printable ASCII range) into hex notation (~5C for the backslash character).

```
x = Char To BLOB( "abc\def\!n" );
y = BLOB To Char( x, encoding = "ASCII~HEX" );
If(
   y == "abc~5Cdef~0A", "JMP 12.2 and later behavior",
   y == "abc\def~0A", "Pre-JMP 12.2 behavior"
);
   "JMP 12.2 and later behavior" // output
```

Char To Hex(*value*, <*integer*|encoding="*enc*">)

Hex(*value*, <*integer*|encoding="*enc*"|Base(*number*)|Pad To(*number*)>)

Description

Returns the hexadecimal (or other base number system) text corresponding to the given value and encoding, which can be a number a quoted string or a blob. If the value is a number, IEEE 754 64-bit encoding is used unless one of the optional arguments, *integer*, or *Base*, is provided.

Required Argument

value Any number, quoted string, or blob.

Optional Arguments

`integer` A quoted switch that causes the value to be interpreted as an integer.

`encoding` A quoted string that specifies an encoding. The default encoding is `"utf-8"`. `"utf-16le"`, `"utf-16be"`, `"us-ascii"`, `"iso-8859-1"`, `"ascii~hex"`, `"shift-jis"`, and `"euc-jp"` are also supported.

`Base(`*`number`*`)` An integer value between 2 and 36 inclusive. If base is specified, the function returns the text corresponding to the specified number in that base number system instead of hexadecimal.

`Pad To(`*`number`*`)` A value to specify the padded width of the hex output.

`Collapse Whitespace(`*`text`*`)`

Description

Trims leading and trailing whitespace and replaces interior whitespace with a single space. That is, if more than one white space character is present, the `Collapse Whitespace` function replaces the two spaces with one space.

Returns

A quoted string.

Required Argument

`text` A quoted string.

`Concat(`*`a`*`, `*`b`*`)`

`Concat(`*`A`*`, `*`B`*`)`

a||*b*

A||*B*

Description

For quoted strings: Appends the string *b* to the string *a*. Neither argument is changed.

For lists: Appends the list *b* to the list *a*. Neither argument is changed.

For matrices: Horizontal concatenation of two matrices, A and B.

Returns

For quoted strings: A quoted string composed of the string *a* directly followed by the string *b*.

For lists: A list composed of the list *a* directly followed by the list *b*.

For matrices: A matrix.

Arguments

Two or more quoted strings, quoted string variables, lists, or matrices.

Example

```
a = "Hello"; b = " "; c = "World"; a || b || c;
```

```
   "Hello World"
d = {"apples", "bananas"}; e = {"peaches", "pears"}; Concat( d, e );
   {"apples", "bananas", "peaches", "pears"}
A = [1 2 3]; B = [4 5 6]; Concat( A, B );
   [1 2 3 4 5 6]
```

Notes

More than two arguments can be strung together. Each additional quoted string is appended to the end, in left to right order. Each additional matrix is appended in left to right order.

Concat Items

See "Concat Items({string1, string2, ...}, <delimiter>})".

Concat To(*a*, *b*)

Concat To(*a*, *b*)

a||=*b*

A||=*B*

Description

For quoted strings: Appends the string *b* to the string *a* and places the new concatenated string into *a*.

For matrices: Appends the matrix *b* to the matrix *a* and places the new concatenated matrix into *a*.

For lists: Appends the list *b* to the list and places the new concatenated list into *a*.

Returns

For quoted strings: A string composed of the string *a* directly followed by the string *b*.

For matrices: A matrix.

For lists: A list composed of the list *a* directly followed by the list *b*.

Arguments

Two or more quoted strings, quoted string variables, matrices, or lists. The first variable must be a variable whose value can be changed.

Notes

More than two arguments can be strung together. Each additional quoted string, matrix, or list is appended to the end, in left to right order.

Example

```
a = "Hello"; b = " "; c = "World"; Concat To( a, b, c ); Show( a );
   a = "Hello World"
A = [1 2 3]; B = [4 5 6]; Concat To( A, B ); Show( A );
   A = [1 2 3 4 5 6];
```

```
d = {"apples", "bananas"}; e = {"peaches", "pears"}; Concat to(d,e); Show( d
);
d = {"apples", "bananas", "peaches", "pears"};
```

Contains(*whole*, *part*, <*start*>)

Description

Determines whether *part* is contained within *whole*.

Returns

If *part* is found: For lists, quoted strings, and namespaces, the numeric position where the first occurrence of *part* is located. For associative arrays, 1.

If *part* is not found, 0 is returned in all cases.

Required Arguments

whole A quoted string, list, namespace, or associative array.

part For a quoted string or namespace, a string that can be part of the string *whole*. For a list, an item that can be an item in the list *whole*. For an associative array, a key that can be one of the keys in the map *whole*.

Optional Argument

start A numeric argument that specifies a starting point within *whole*. If *start* is negative, contains searches *whole* for *part* backwards, beginning with the position specified by the length of *whole* – *start*. Note that *start* is meaningless for associative arrays and is ignored.

Example

```
nameList={"Katie", "Louise", "Jane", "Jaclyn"};
r = Contains(nameList, "Katie");
```

The example returns a 1 because “Katie” is the first item in the list.

Contains Item(*x*, <*item* | {*list*} | *pattern*>, <*delimiter*>)

Description

Identifies multiple responses by searching for the specified item, list, pattern, or delimiter. The function can be used on columns with the Multiple Response modeling type or column property.

Returns

Returns a Boolean that indicates whether the word (*item*), one of a list of words (*list*), or pattern (*pattern*) matches one of the words in the text represented by *x*. Words are delimited by the characters in the optional quoted delimiter (*delimiter*) string. A comma is the default delimiter. Blanks are trimmed from the ends of each extracted word from the input text string (*x*).

Example

The following example searches for *"pots"* followed by a comma and then outputs the result.

```
x = "Franklin Garden Supply is a leading online store featuring garden decor,
   statues, pots, shovels, benches, and much more.";
b = Contains Item( x, "pots", "," );
If( b,
   Write( "The specified items were found."  ), Write( "No match." )
);
   The specified items were found.
```

Ends With(*string*, *substring*)

Description

Determines whether the quoted `substring` appears at the end of `string`.

Returns

1 if `quoted string` ends with a quoted `substring`, otherwise 0.

Required Arguments

`string` A quoted string or a quoted string variable. Can also be a list.

`substring` A quoted string or a quoted string variable. Can also be a list.

Equivalent Expression

`Right(`*`string`*`, Length(`*`substring`*`)) == substring`

Hex(*value*, <*integer*|encoding="*enc*"|Base(*number*)|Pad To(*number*)>)

See "Char To Hex(value, <integer|encoding="enc">)".

Hex To BLOB(*string*)

Description

Converts the quoted hexadecimal *`string`* (including whitespace characters) to a blob (binary large object).

Example

```
Hex To BLOB( "4A4D50" );
   Char To BLOB("JMP", "ascii~hex")
```

Hex To Char(*string*, <*encoding*>)

Description

Converts the quoted hexadecimal *`string`* to its character equivalent.

Example

Hex To Char("30") results in "0".

Notes

The default quoted encoding for character string is "utf-8". "utf-16le", "utf-16be", "us-ascii", "iso-8859-1", "ascii~hex", "shift-jis", and "euc-jp" are also supported.

`Hex To Number(`*`string`*`, <Base(`*`number`*`)>)`

Description

Returns the number corresponding to the hexadecimal (or other base number system) text.

Required Argument

`string` A quoted hexadecimal string.

Optional Argument

`Base(`*`number`*`)` An integer between 2 and 36 inclusive. If base is specified, the text is treated as a quoted string representing the number in that base.

Example

```
Hex To Number( "80" );
  128
```

Notes

- 16-digit hexadecimal numbers are converted as IEEE 754 64-bit floating point numbers. Otherwise, the input is treated as a hexadecimal integer.
- Whitespace between bytes (or pairs of digits) and in the middle of bytes is permitted (for example, `FF 1919` and `F F1919`).

`Insert`

See "Insert(source, item, <position>)".

`Insert Into`

See "Insert Into(source, item, <position>)".

`Item(`*`n`*`|[`*`first last`*`], `*`string`*`, <`*`delimiter`*`>, <Unmatched(`*`result string`*`)>, <Include Boundary Delimiters(`*`Boolean`*`)>)`

Description

Returns the *n*th item or the span from the first to last item of the quoted *`string`* according to the quoted string *`delimiters`* given. If you include a third argument, any and all characters in that argument are taken to be delimiters.

Required Arguments

`n` The position of the word being extracted.

`[`*`first last`*`]` A matrix that defines the beginning and end word range to return.

`string` The quoted string that is evaluated.

Optional Arguments

delimiter The character used as a boundary. If *delimiter* is absent, an ASCII space is used. If *delimiter* is the quoted empty string, each character is treated as a separate word.

Unmatched(*result string*) The quoted string to print if no match is found.

Include Boundary Delimiters(*Boolean*) Determines how delimiters on the front and back of string are treated. The default value is false, which means that the delimiters on these boundaries are ignored. If the value is true, delimiters on these boundaries produce an empty element (similar to consecutive delimiters within a string).

Example

Consecutive delimiters are treated as though they have a word between them. In this example, the delimiters are a comma and a space.

```
Item( 4,"the  quick, brown fox", ", " ); // quick is preceded by two spaces
```

The expression is processed as follows:

the<delim[space]><word2><delim[space]>quick<delim[comma]><word 4><delim[space]>brown<delim[space]>fox

Because word4 is empty, this expression returns a quoted empty string.

Item() is the same as Word() except that Item() treats each delimiter character as a separate delimiter, and Word() treats several adjacent delimiters as a single delimiter.

```
Word( 4,"the  quick, brown fox", ", " ); // quick is preceded by two spaces
```

This expression is processed as follows:

the<delim[2 spaces]>quick<delim[comma + space]>brown<delim[space]>fox

It returns "fox".

Left(*string*, *n*, <*filler*>)

Left({*list*}, *n*, <*filler*>)

Description

Returns a truncated or padded version of the original quoted *string* or *list*. The result contains the left *n* characters or list items, padded with any *filler* on the right if the length of *string* is less than *n*.

Length(*string*)

Description

Returns the length of the given quoted string (in characters), list (in items), associative array (in number of keys), BLOB (in bytes), matrix (in elements), namespace (in number of functions and variables), or class (in number of methods, functions, and variables).

Lowercase(*string*)

Description

Converts any uppercase character found in quoted *string* to the equivalent lowercase character.

Matrix to BLOB(*matrix*, *type*, *bytesEach*, *endian(value)*)

Description

Makes a BLOB from a matrix by converting the matrix elements to 1-byte, 2-byte, or 4-byte signed or unsigned integers or 4-byte or 8-byte floating point numbers.

Required Arguments

matrix The matrix.

type The quoted type of BLOB: `int`, `uint`, or `float`.

bytesEach The number of bytes in each int, uint, or float. Integers can be 1, 2, or 4 bytes each, and floats can be 4 or 8 bytes each.

endian(value) The quoted endian-ness of your system: `"Big"` (the first byte is most significant), `"Little"` (the first byte is the least significant), or `"Native"` (the machine's native format).

Munger(*string*, *start position*, *find|length*, *<replacement string>*)

Description

Computes new quoted character strings from the quoted *string* by inserting or deleting characters. It can also produce substrings, calculate indexes, and perform other tasks depending on how you specify its arguments.

Required Arguments

start position A numeric expression that specifies the starting position to search in the quoted string. If the *start position* is greater than the position of the first instance of the find argument, the first instance is disregarded. If the *start position* is greater than the search string's length, `Munger()` uses the string's length as the *start position*.

find|length Specifies the string or number of characters to find.

Optional Argument

replacement string The quoted replacement string. If *replacement string* is omitted, a substring between start position, and position and length, is returned.

Num(*string*)

Description

Converts a quoted character string into a number.

`Regex(source, pattern, (<replacement string>, <format, "GLOBALREPLACE", "IGNORECASE">>)`

Description

Searches for the quoted *pattern* within the quoted *source* string.

Returns

The matched text as a quoted string or numeric missing if there was no match.

Required Arguments

source A quoted string.

pattern A quoted regular expression.

replacement string The replacement string.

Optional Arguments

format A backreference to the capturing group. The default is \0, which is the entire matched quoted string. \n returns the n^{th} match.

"IGNORECASE" The search is case sensitive, unless you specify "IGNORECASE".

"GLOBALREPLACE" Applies the regular expression to the quoted *source* string repeatedly until all matches are found.

`Remove`

See "Remove(source, position, <n>)".

`Remove From`

See "Remove From(source, position, <n>)".

`Repeat(source, a)`

`Repeat(matrix, a, b)`

Description

Returns a copy of *source* concatenated with itself *a* times. Or returns a matrix composed of *a* row repeats and *b* column repeats. The *source* can be text, a matrix, or a list.

`Reverse`

See "Reverse(source)".

`Reverse Into`

See "Reverse Into(source)".

`Right(string, n, <filler>)`

`Right({list}, n, <filler>)`

Description

Returns a truncated or padded version of the original quoted *string* or *list*. The result contains the right *n* characters or list items, padded with any optional *filler* on the left if the length of *string* is less than *n*.

`Shift`

See "Shift(source, <n>)".

`Shift Into`

See "Shift Into(source, <n>)".

`Starts With(string, substring)`

Description

Determines whether the quoted `substring` appears at the start of the quoted *string*.

Returns

1 if *string* starts with *substring*, otherwise 0.

Arguments

string A quoted string or a reference to one. Can also be a list.

substring A quoted string or a reference to one. Can also be a list.

Equivalent Expression

`Left(string, Length("substring")) = = "substring"`

`Substitute`

See "Substitute(string, "substring", "replacementString", ...)".

`Substitute Into`

See "Substitute Into(string, substring, replacementString, ...)".

`Substr(string, start, length)`

Description

Extracts the characters that are the portion of the first argument beginning at the position given by the second argument and ending based on the number of characters specified in the third argument. The first argument can be a character column or value, or an

expression evaluating to same. The starting argument and the length argument can be numbers or expressions that evaluate to numbers.

Example

This example extracts the first name:

```
Substr( "Katie Layman", 1, 5 );
```

The function starts at position 1, reads through position 5, and ignores the remaining characters, which yields "Katie."

`Text Score(`*`text column`*`, `*`text-to-number`*`, <`*`weighting`*`>, <{`*`support vectors`*`}>, <`*`text explorer setup`*`>)`

Description

Used to create scoring formulas in Text Explorer. Not supported for use with the Stem for Combining option.

Returns

Returns a vector of scores.

Required Arguments

`text column` The data table column.

`text-to-number` An associative array that maps lowercase words to numbers.

Optional Arguments

`weighting` The quoted `"Count"`, `"Binary"`, `"Ternary"`, `"LogCount"`, `"LCA"`, or an array of inverse document frequency weights for TFLogIDF. `"Count"` is the default value. The default value is `"Count"`.

`support vectors` A list of vectors that are used in the text scoring. The number and length of the vectors depends on the *`weighting`* argument.

`text explorer setup` An expression that contains a block of Text Explorer setup information.

`Titlecase(`*`string`*`)`

Description

Converts the quoted *`string`* to title case; each word in the string has an initial uppercase character and subsequent lowercase characters.

Returns

A quoted string.

Argument

`string` A quoted string.

Example

The following function capitalizes the name:

```
Titlecase( "veronica layman ")
```

```
"Veronica Layman"
```

`Trim(string,<"Left"|"Right"|"Both">)`

`Trim Whitespace(string,<"Left"|"Right"|"Both">)`

Description

Removes leading and trailing whitespace from the specified string.

Returns

A quoted string.

Required Argument

string A quoted string.

Optional Argument

`"Left"|"Right"|"Both"` A quoted string that specifies whether whitespace is removed from the left, the right, or both ends of the string. If omitted, whitespace is removed from both ends.

Example

For example, the following command returns `"John"`:

```
Trim( "  John    ", Both )
   "John"
```

`Uppercase(string)`

Description

Converts any lowercase character found in the quoted *string* to the equivalent uppercase character.

`Word(n|[first last], string, <delimiter>, <Unmatched(result string)>)`

Description

Returns the *n*th item of the string, where words are substrings separated by any number of any characters in the *delimiter* argument.

Required Arguments

n The position of the word being extracted.

`[first last]` A matrix that defines the beginning and end word range to return.

string The quoted string that is evaluated.

Optional Arguments

delimiter The character used as a boundary. If *delimiter* is absent, an ASCII space is used. If *delimiter* is the empty quoted string, each character is treated as a separate word.

`Unmatched(result string)` The quoted string to print if no match is found.

Examples

This example returns the last name:

```
Word( 2, "Katie Layman" );
   "Layman"
```

See Also

See "Item(n|[first last], string, <delimiter>, <Unmatched(result string)>, <Include Boundary Delimiters(Boolean)>)" for examples of how `Word()` differs from `Item()`.

Words

See "Words(string, <delimiter>)".

XPath Query(*xml*, *xpath_expression*)

Description

Runs an XPath expression on an XML document.

Returns

A list.

Required Arguments

xml A valid XML document.

xpath_expression A quoted XPath 1.0 expression.

Example

Suppose that you created a report of test results in JMP and exported important details to an XML document. The test results are enclosed in `<result>` tags.

The following example stores the XML document in a variable. The `XPath Query` expression parses the XML to find the text nodes inside the `<result>` tags. The results are returned in a list.

```
rpt =
"\[<?xml version="1.0" encoding="utf-8"?>
<JMP><report><title>Production Report</title>
<result>November 21st: Pass</result>
<result>November 22nd: Fail</result>
<note>Tests ran at 3:00 a.m.</note></report>
</JMP> ]\";
results = XPath Query( rpt, "//result/text()" );
   {"November 21st: Pass", "November 22nd: Fail"}
```

Character Pattern Functions

Pat Abort()

Description

Constructs a quoted pattern that immediately stops the pattern match. The matcher does not back up and retry any alternatives. Conditional assignments are *not* made. Immediate assignments that were already made are kept.

Returns

0 when a match is stopped.

Argument

none

Pat Altern(*pattern1*, <*pattern2*, ...>)

Description

Constructs a quoted pattern that matches any one of the pattern arguments.

Returns

A quoted pattern.

Argument

One or more patterns.

Pat Any(*string*)

Description

Constructs a quoted pattern that matches a single character in the argument.

Returns

A quoted pattern.

Argument

pattern A quoted string.

Pat Arb()

Description

Constructs a quoted pattern that matches an arbitrary quoted string. It initially matches the null quoted string. It then matches one additional character each time the pattern matcher backs into it.

Returns

A quoted pattern.

Argument

none

Example

```
p = "the beginning" + Pat Arb() >? stuffInTheMiddle + "the end";
Pat Match( "in the beginning of the story, and not near the end, there are
   three bears", p );
Show( stuffInTheMiddle );
   stuffInTheMiddle = " of the story, and not near "
```

Pat Arb No(*pattern*)

Description

Constructs a quoted pattern that matches zero or more copies of *pattern*.

Returns

A quoted pattern.

Argument

pattern A quoted pattern to match against.

Example

```
adjectives = "large" | "medium" | "small" | "warm" | "cold" | "hot" | "sweet";
rc = Pat Match( "I would like a medium hot, sweet tea please",
                Pat Arbno( adjectives | Pat Any(", ") ) >> adj +
                ("tea" | "coffee" | "milk") );
Show( rc, adj );
   rc = 1;
   adj = " medium hot, sweet ";
```

Pat At(*varName*)

Description

Constructs a quoted pattern that matches the null quoted string and stores the current position in the source string into the specified JSL variable (*varName*). The assignment is immediate, and the variable can be used with `expr()` to affect the remainder of the match.

Returns

A quoted pattern.

Argument

varName The name of a variable to store the result in.

Example

```
p = ":" + Pat At( listStart ) + Expr(
   If( listStart == 1,
      Pat Immediate( Pat Len( 3 ), early ),
      Pat Immediate( Pat Len( 2 ), late )
   )
);
```

```
early = "";
late = "";
Pat Match( ":123456789", p );
Show( early, late );
early = "";
late = "";
Pat Match( "   :123456789", p );
Show( early, late );
```

First this is produced:

```
early = "123"
late = ""
```

and later this:

```
early = ""
late = "12"
```

Pat Break(*string*)

Description

Constructs a quoted pattern that matches zero or more characters that are not in its argument; it stops or breaks on a character in its argument. It fails if a character in its argument is not found (in particular, it fails to match if it finds the end of the quoted source string without finding a break character).

Returns

A quoted pattern.

Argument

string A quoted string.

Pat Concat(*pattern1*, *pattern2* <*pattern3*, ...>)

Pattern1 + *Pattern2* + ...

Description

Constructs a quoted pattern that matches each pattern argument in turn.

Returns

A quoted pattern.

Argument

Two or more quoted patterns.

Pat Conditional(*pattern*, *varName*)

Description

Saves the result of the quoted pattern match, if it succeeds, to a variable named as the second argument (*varName*) after the match is finished.

Returns

A quoted pattern.

Arguments

pattern A quoted pattern to match against.

varName The name of a variable to store the result in.

Example

```
type = "undefined";
rc = Pat Match(
   "green apples",
   Pat Conditional( "red" | "green", type ) + " apples"
);
Show( rc, type );
   rc = 1;
   type = "green";
```

Pat Fail()

Description

Constructs a quoted pattern that fails whenever the matcher attempts to move forward through it. The matcher backs up and tries different alternatives. If and when there are no alternatives left, the match fails and `Pat Match` returns 0.

Returns

0 when a match fails.

Argument

none

Pat Fence()

Description

Constructs a pattern that succeeds and matches the quoted null string when the matcher moves forward through it, but fails when the matcher tries to back up through it. It is a one-way trap door that can be used to optimize some matches.

Returns

1 when the match succeeds, 0 otherwise.

Argument

none

Pat Immediate(*pattern*, *varName*)

Description

Saves the result of the pattern match to a variable named as the second argument (*varName*) immediately.

Returns

A quoted pattern.

Arguments

pattern A quoted pattern to match against.

varName The name of a variable to store the result in.

Example

```
type = "undefined";
rc = Pat Match(
   "green apples",
   ("red" | "green") >> type + " pears"
);
Show( rc, type );
   rc = 0
   type = "green"
```

Even though the match failed, the immediate assignment was made.

Pat Len(*int*)

Description

Constructs a quoted pattern that matches *n* characters.

Returns

A quoted pattern.

Argument

int An integer that specifies the number of characters.

Pat Look Ahead(*pattern*, *Boolean*)

Description

A zero-width pattern match after the current position.

Arguments

pattern The quoted pattern.

Boolean 0 (the default) indicates a match. 1 indicates a negative match or non-match.

Pat Look Behind(*pattern*, *Boolean*)

Description

A zero-width quoted pattern match before the current position.

Arguments

pattern The quoted pattern.

Boolean 0 (the default) indicates a match. 1 indicates a negative match or non-match.

`Pat Match(source text, pattern, <replacement string>, <"NULL">, <"ANCHOR">, <"MATCHCASE">, <"FULLSCAN">)`

Description

`Pat Match` executes the quoted *pattern* against the *source text*. The pattern must be constructed first, either inline or by assigning it to a JSL variable elsewhere.

Returns

1 if the pattern is found, 0 otherwise.

Required Arguments

source text A quoted string or quoted string variable that contains the text to be searched.

pattern A quoted pattern or pattern variable that contains the text to be searched for.

Optional Arguments

replacement string A quoted string that defines text to replace the pattern in the source text.

"NULL" A placeholder for the third argument if ANCHOR, MATCHCASE, or FULLSCAN are necessary *and* there is no replacement text.

"ANCHOR" Starts the pattern match at the beginning of the quoted string. The following match fails because the pattern, "cream", is not found at the beginning of the string:

```
Pat Match( "coffee with cream and sugar", "cream", NULL, ANCHOR );
```

"MATCHCASE" Optional command to consider capitalization in the match. By default, `Pat Match()` is case insensitive.

"FULLSCAN" Optional command to force `Pat Match` to try all alternatives, which uses more memory as the match expands. By default, `Pat Match()` does not use FULLSCAN, and makes some assumptions that allow the recursion to stop and the match to succeed.

`Pat Not Any(string)`

Description

Constructs a pattern that matches a single character that is not in the argument.

Returns

A quoted pattern.

Argument

`string` A quoted string.

Pat Pos(*int*)

Description

Constructs patterns that match the quoted null string if the current position is *int* from the left end of the string, and fail otherwise.

Returns

A quoted pattern.

Argument

int An integer that specifies the position in a quoted string.

Pat R Pos(*int*)

Description

Constructs patterns that match the quoted null string if the current position is *int* from the right end of the string, and fails otherwise.

Returns

A quoted pattern.

Argument

int An integer that specifies the position in a quoted string.

Pat R Tab(*int*)

Description

Constructs a quoted pattern that matches up to position *n* from the end of the quoted string. It can match 0 or more characters. It fails if it would have to move backwards or beyond the end of the string.

Returns

A quoted pattern.

Argument

int An integer that specifies a position in a quoted string.

Pat Regex(*string*)

Description

Constructs a quoted pattern that matches the regular expression in the quoted *string* argument.

Returns

A quoted pattern.

Argument

string A quoted string.

Pat Rem()

Description

Constructs a quoted pattern that matches the remainder of the quoted string. It is equivalent to `Pat R Tab(0)`.

Returns

A quoted pattern.

Argument

none

Pat Repeat(*pattern*, *minimum*, *maximum*, <"GREEDY"|"RELUCTANT">)

Description

Matches the quoted *pattern* between *minimum* and *maximum* times.

Returns

A quoted pattern.

Required Arguments

pattern A pattern to match against.

minimum An integer that must be smaller than maximum.

maximum An integer that must be greater than minimum.

Optional Argument

"GREEDY"|"RELUCTANT" If GREEDY is specified, it tries the maximum first and works back to the minimum. If RELUCTANT is specified, it tries the minimum first and works up to the maximum.

Notes

- `Pat Arbno(p)` is the same as `Pat Repeat(p, 0, infinity, RELUCTANT)`
- `Pat Repeat(p)` is the same as `Pat Repeat(p, 1, infinity, GREEDY)`
- `Pat Repeat(p, n)` is the same as `Pat Repeat(p, n, infinity, GREEDY)`
- `Pat Repeat(p, n, m)` is the same as `Pat Repeat(p, n, m, GREEDY)`

Pat Span(*string*)

Description

Constructs a pattern that matches one or more (not zero) occurrences of characters in its argument. It is greedy; it always matches the longest possible quoted string. It fails rather than matching zero characters.

Returns

A quoted pattern.

Argument

string A quoted string.

Pat String(*string*)

Description

Constructs a pattern that matches its quoted *string* argument.

Returns

A quoted pattern.

Argument

string A quoted string.

Pat Succeed()

Description

Constructs a pattern that always succeeds, even when the matcher backs into it. It matches the quoted null string.

Returns

1 when the match succeeds.

Argument

none

Pat Tab(*int*)

Description

Constructs a pattern that matches forward to position *int* in the quoted source string. It can match 0 or more characters. It fails if it would have to move backwards or beyond the end of the string.

Returns

A pattern.

Argument

int An integer that specifies a position in a quoted string.

Pat Test(*expr*)

Description

Constructs a pattern that succeeds and matches the quoted null string if *expr* is not zero and fails otherwise.

Returns

A quoted pattern.

Argument

expr An expression.

Notes

Usually the argument is wrapped with `expr()` because the test needs to be made on the current value of variables set by `Pat Immediate`, `Pat Conditional`, and `Pat At`. Without `expr`, the test is based on values that were known when the pattern was constructed, which means the test always succeeds or always fails at pattern execution time, which is probably not what you want.

Example

```
nCats = 0;
whichCat = 3;
string = "catch a catnapping cat in a catsup factory";
rc = Pat Match(
   string,
   "cat" + Pat Test(
      Expr(
         nCats = nCats + 1;
         nCats == whichCat;
      )
   ),
   "dog"
);
Show( rc, string, nCats );
   rc = 1
   string = "catch a catnapping dog in a catsup factory"
   nCats = 3
```

`Regex Match(`*source*`, `*pattern*`, <`*replacement string*`>|<"MATCHCASE">, <"NULL">)`

Description

Executes the pattern match in quoted *pattern* against the quoted *source* string.

Returns

A pattern.

Required Arguments

source A quoted string.

pattern A quoted pattern.

Optional Arguments

replacement string The quoted string that specifies the text to replace the source with.

"MATCHCASE" The search is case insensitive unless you specify MATCHCASE.

"NULL" Indicates that the expression contains MATCHCASE but you don't want to specify a replacement.

Examples

```
Regex Match(
   "person=Fred id=77 friend= favorite=tea", // source
   "(\w+)=(\S*) (\w+)=(\S*) (\w+)=(\S*) (\w+)=(\S*)" // pattern
);
   {"person=Fred id=77 friend= favorite=tea", "person", "Fred", "id", "77",
      "friend", "", "favorite", "tea"}

// case-insensitive, no replacement
Regex Match( "beliEve", "([aeiou])(.*?)(\1)" );
   {"eliE", "e", "li", "E"}
// case-sensitive, no replacement
Regex Match( "beliEve", "([aeiou])(.*?)(\1)", NULL, MATCHCASE );
   {"eliEve", "e", "liEv", "e"}
```

Comment Functions

// comment

Description

Comments to end of line.

Notes

Everything after the // is ignored when running the script.

/* comment */

Description

A comment that can appear in the middle of a line of script.

Notes

Anything between the beginning tag /* and the end tag */ is ignored when running the script. This comment style can be used almost anywhere, even inside lists of arguments. If you place a comment inside a double-quoted string, the comment is treated merely as part of the string and not a comment. You cannot place comments in the middle of operators.

Examples

```
+/*comment*/=
:/*comment*/name
```

are invalid and produce errors. The first comment interrupts += and the second interrupts :name.

```
sums = {(a+b /*comment*/), /*comment*/ (c^2)}
```

is valid JSL; the comments are both ignored.

`//!`

Description

If placed on the first line of a script, this comment line causes the script to be run when opened in JMP without opening into the script editor window.

Notes

You can over-ride this comment when opening the file. Select **File > Open**. Hold the Ctrl key while you select the JSL file and click **Open**. Or right-click the file in the Home Window Recent Files list and select **Edit Script**. The script opens into a script window instead of being executed.

`/*debug step*/`

`/*debug run*/`

Description

If placed on the first line of a script, the script is opened in the debugger when it is run.

Notes

All letters must be lowercase. There must be one space between debug and step or run, and there must be no other spaces present. Only one of these lines can be used, and it must be the first line of the script; a first line that is blank followed by this comment negates the debug command.

Comparison Functions

The comparison operators (<, <=, >, >=) work for numbers, quoted strings, and matrices. For matrices, they produce a matrix of results. If you compare mixed arguments, such as strings with numbers or matrices, the result is a missing value. Comparisons involving lists are not allowed and also return missing values.

The equality operators (== and !=) work for numbers, quoted strings, matrices, and lists. For matrices, they produce a matrix of results; for lists, they produce a single result. If you *test equality* of mixed results (for example. strings with numbers or matrices) the result is 0 or unequal.

Range check operators let you check whether something falls between two specified values:

```
a = 1;
Show( 1 <= a < 3 );
b = 2;
Show( 2 < b <= 3 );
   1 <= a < 3 = 1;
   2 < b <= 3 = 0;
```

Expressions with comparison operators are evaluated all at once, not in sequence

All the comparison operators are *eliding operators*. That means JMP treats arguments joined by comparison operators as one big clause, as opposed to the way most expressions are evaluated one operator at a time. Evaluating as a single clause produces different results than the more usual method of evaluating in pieces. For example, the following two statements are different:

```
12 < a < 13;
(12 < a) < 13;
```

The first statement checks whether *a* is between 12 and 13, because all three arguments and both operators are read and evaluated together. The second statement uses parentheses to regroup the operations explicitly to evaluate from left to right, which would be the normal way to evaluate most expressions. Thus it first checks whether 12 is less than *a*, returning 1 if true or 0 if false. Then it checks whether the result is less than 13, which is always true because 0 and 1 are both less than 13.

All the comparison operators are elided when they are used in matched pairs or in the unmatched pairs <... <= and <=... <. What this means is that if you want a comparison statement to be evaluated one comparison operator at a time, you should use parentheses ( ) to control the order of operations explicitly.

`Equal(a, b, ...)`

`a==b==...`

Description

Compares all the listed values and tests if they are all equal to each other.

Returns

1 (true) if all arguments evaluate to the same value.

0 (false) otherwise.

Arguments

Two or more variables, references, matrices, or numbers.

Notes

If more than two arguments are specified, a 1 is returned only if all arguments are exactly the same. This is typically used in conditional statements and to control loops.

The comparison is case-sensitive for quoted string comparisons.

`Greater(a, b, ...)`

`a>b>...`

Description

Compares all the list values and tests if, in each pair, the left value is greater than the right.

Returns

1 (true) if *a* evaluates strictly greater than *b* (and *b* evaluates strictly greater than *c*, and so on).

0 (false) otherwise.

Arguments

Two or more variables, references, matrices, or numbers.

Notes

If more than two arguments are specified, a 1 is returned only if each argument is greater than the one that follows it. This is typically used in conditional statements and to control loops.

`Greater`, `Less`, `Greater Or Equal`, and `Less Or Equal` can also be strung together (as in `0 < x <= 5`). If you do not group with parentheses, JMP evaluates each pair left to right. You can also use parentheses to explicitly tell JMP how to evaluate the expression.

`Greater or Equal(`*a*`, `*b*`, ...)`

a>=*b*>=...

Description

Compares all the list values and tests if, in each pair, the left value is greater than or equal to the right.

Returns

1 (true) if *a* evaluates strictly greater than or equal to *b* (and *b* evaluates strictly greater than or equal to *c*, and so on).

0 (false) otherwise.

Arguments

Two or more variables, references, matrices, or numbers.

Notes

If more than two arguments are specified, a 1 is returned only if each argument is greater than or equal to the one that follows it. This is typically used in conditional statements and to control loops.

`Greater`, `Less`, `Greater Or Equal`, and `Less Or Equal` can also be strung together (as in `0 < x <= 5`). If you do not group with parentheses, JMP evaluates each pair left to right. You can also use parentheses to explicitly tell JMP how to evaluate the expression.

`Is Missing(`*expr*`)`

Description

Returns 1 if the expression yields a missing value and 0 otherwise.

`Less(a, b, ...)`

`a<b<...`

Description

Compares all the list values and tests if, in each pair, the left value is less than the right.

Returns

1 (true) if *a* evaluates strictly less than *b* (and *b* evaluates strictly less than *c*, and so on).

0 (false) otherwise.

Arguments

Two or more variables, references, matrices, or numbers.

Notes

If more than two arguments are specified, a 1 is returned only if each argument is less than the one that follows it. This is typically used in conditional statements and to control loops.

`Greater`, `Less`, `Greater Or Equal`, and `Less Or Equal` can also be strung together (as in `0 < x <= 5`). If you do not group with parentheses, JMP evaluates each pair left to right. You can also use parentheses to explicitly tell JMP how to evaluate the expression.

`Less Less Equal(a, b, c, ...)`

`a<b<=c<=...`

Description

Range check, exclusive below and inclusive above.

Returns

1 (true) if *b* is greater than *a* and less than or equal to *c*.

0 (false) otherwise.

Arguments

a, *b*, *c* Variables, references, matrices, or numbers.

Notes

Returns 1 when two conditions are met: the first argument is less than the second argument, and each remaining argument is less than or equal to its argument on the right. This is typically used in conditional statements and to control loops.

`Less or Equal(a, b, ...)`

`a<=b<=...`

Description

Compares all the list values and tests if, in each pair, the left value is less than or equal to the right.

Returns

1 (true) if *a* evaluates strictly less than or equal to *b* (and *b* evaluates strictly less than or equal to *c*, and so on).

0 (false) otherwise.

Arguments

Two or more variables, references, matrices, or numbers.

Notes

If more than two arguments are specified, a 1 is returned only if each argument is less than or equal to the one that follows it. This is typically used in conditional statements and to control loops.

`Greater`, `Less`, `Greater Or Equal`, and `Less Or Equal` can also be strung together (as in `0 < x <= 5`). If you do not group with parentheses, JMP evaluates each pair left to right. You can also use parentheses to explicitly tell JMP how to evaluate the expression.

`Less Equal Less(a, b, c, ...)`

`a<=b<c<...`

Description

A range check, inclusive below and exclusive above.

Returns

1 (true) if *b* is greater than or equal to *a* and less than *c*.

0 (false) otherwise.

Arguments

a, *b*, *c* Variables, references, matrices, or numbers.

Notes

Returns 1 when two conditions are met: the first argument is less than or equal to the second argument, and each remaining argument is less than its argument on the right. This is typically used in conditional statements and to control loops.

`Not Equal(a, b)`

`a!=b`

Description

Compares *a* and *b* and tests if they are equal.

Returns

0 (false) if *a* and *b* evaluate to the same value.

1 (true) otherwise.

Argument

a, b Any variable or number.

Notes

Mostly used for conditional statements and loop control.

Conditional and Logical Functions

And(*a, b*)

a&*b*

Description

The logical And.

Returns

1 (true) if both *a* and *b* are true.

0 (false) if either *a* or *b* is false or if both *a* and *b* are false.

Missing if either *a* or *b* is a missing value or if both *a* and *b* are missing values.

Arguments

Two or more variables or expressions.

Notes

More than two arguments can be strung together. a&b returns 1 (true) only if all arguments evaluate to true.

AndMZ(*a, b*)

a:&*b*

Description

Returns the logical AND of all arguments. Missing values are treated as zeroes.

Returns

1 (true) if both *a* and *b* are true.

0 (false) if either *a* or *b* is false or if both *a* and *b* are false.

0 (false) if either *a* or *b* is a missing value or if both *a* and *b* are missing values.

Arguments

Two or more variables or expressions.

Notes

More than two arguments can be strung together. *a*:&*b* returns 1 (true) only if all arguments evaluate to true.

Break()

Description

Stops execution of a loop completely and continues to the statement following the loop.

Notes

`Break()` works with `For` and `While` loops and also with `For Each Row`.

Choose(*expr*, *r1*, *r2*, *r3*, ..., *resultElse*)

Description

Evaluates *expr*. If the value of *expr* is 1, *r1* is returned; if 2, the value of *r2* is returned, and so on. If no matches are found, the last argument (*resultElse*) is returned.

Returns

The value whose index in the list of arguments matches *expr*, or the value of the last argument.

Arguments

expr An expression or a value.

r1, *r2*, *r3*, ... An expression or a value.

resultElse The argument that is returned when no matches are found.

Continue()

Description

Ends the current iteration of a loop and begins the loop at the next iteration.

Notes

`Continue()` works with `For` and `While` loops, and also with `For Each Row`.

Filter Each(*names*, *container*, *locals*, *body*)

Description

Iterates over a container, which can be a list, an associative array, or a matrix, and returns a subset of the values in the container based on the evaluation of a Boolean expression at each iteration. The value, key, or element, as well as the index number, are all available at each iteration.

- For associative array containers, the key and value can be accessed using a two-item list.
- For matrix containers, a linear index is provided by default, but a two-item list can be used to access the row and column indices.

These symbols are provided within the body of the loop only. A list of local variables can also be provided. If local variables are defined, they are initialized after the first iteration symbols are set.

Returns

A subset from the original container. The return object has the same type as the original container.

Arguments

names Specification of loop control variable names, specified as a list. The form of the list is determined by the type of container. All of the names are optional, so you need to specify them only if you need to refer to them in the *locals* or *body* arguments. If you do not specify any names, the first argument should be an empty list, which can be specified as {} or List().

For a *list* container, the names list contains a name for each value in the list and a name for the index of each value in the list.

For an *associative array* container, the names list contains a two-item list of names and a name for the index of each item in the associative array.

For a *matrix* container, the names list contains a name for each element in the matrix and a second argument for the index of each element in the matrix. The second argument can be specified as a single name or as a two-item list of names that represent symbols for the row and column indices.

When specifying multiple containers using the Across() keyword, the names list contains a list of names that refer to values in each container. The number of names in the first item in the names list must match the number of containers specified in the Across() keyword.

container A list, associative array, or a matrix. The container can be defined in the argument or it can be a reference to a previously defined object.

This argument can also use the Across() keyword, which enables you to use the function across multiple containers. The multiple containers can be specified as separate arguments or as items in a list. The Across() keyword has an optional Count() argument that enables you to specify how containers of different sizes are handled. The available Count() options are: "Shortest", "Longest", "Enforce Equal", and N, where N is a number. If you specify a number, the function iterates through all containers exactly N times; note that the function loops back to the start of containers that have fewer than N items.

locals A list of variables that are local to the function. This is equivalent to other Local variable initializations in JSL. The initialization of the local variables occurs after the first loop control variables have been set, but the local variables do not get initialized again after that.

body Any number of valid JSL expressions, glued together if there are more than one. The result of the JSL expressions should be a Boolean value. If the result of the expression is true, the container value at the current iteration is included in the result; otherwise, the

container value at the current iteration is not included in the result. You can use the `Continue()` function as an equivalent way to return false and skip to the next iteration. You can also use `Break()` function to stop iteration through the loop and proceed to the next expression that follows the loop. See "Break and Continue".

Example

```
values = Filter Each( {x}, {10, 20, 30}, x > 15 );
Show( values );
  values = {20, 30};
```

For(*init*, *while*, *increment*, *body*)

Description

Repeats the statement(s) in the *body* as long as the *while* condition is true. *init* and *increment* control iterations.

Returns

Null.

Arguments

init Initialization of loop control counter.

while Condition for loop to continue or end. As long as the conditional statement while is true, the loop is iterated one more time. As soon as while is false, the loop is exited.

increment Increments (or decrements) the loop counter after while is evaluated every *time* the loop is executed.

body Any number of valid JSL expressions, glued together if there are more than one.

Example

```
mysum = 0; myprod = 1;
For( i = 1, i <= 10, i++, mysum += i; myprod *= i; );
Show( mysum, myprod );
  mysum = 55;
  myprod = 3628800;
```

For Each(*names*, *container*, *locals*, *body*)

Description

Iterates over a container, which can be a list, an associative array, or a matrix, and provides the value, key, or element at each iteration. The index number is also available at each iteration. For associative array containers, the key and value can be accessed using a two-item list. For matrix containers, a linear index is provided by default, but a two-item list can be used to access the row and column indices. These symbols are provided within the body of the loop only. A list of local variables can also be provided. If local variables are defined, they are initialized after the first iteration symbols are set.

Arguments

names Specification of loop control variable names, specified as a list. The form of the list is determined by the type of container. All of the names are optional, so you need to specify them only if you need to refer to them in the *locals* or *body* arguments. If you do not specify any names, the first argument should be an empty list, which can be specified as {} or List().

For a *list* container, the names list contains a name for each value in the list and a name for the index of each value in the list.

For an *associative array* container, the names list contains a two-item list of names and a name for the index of each item in the associative array.

For a *matrix* container, the names list contains a name for each element in the matrix and a second argument for the index of each element in the matrix. The second argument can be specified as a single name or as a two-item list of names that represent symbols for the row and column indices.

When specifying multiple containers using the Across() keyword, the names list contains a list of names that refer to values in each container. The number of names in the first item in the names list must match the number of containers specified in the Across() keyword.

container A list, associative array, or a matrix. The container can be defined in the argument or it can be a reference to a previously defined object.

This argument can also use the Across() keyword, which enables you to use the function across multiple containers. The multiple containers can be specified as separate arguments or as items in a list. The Across() keyword has an optional Count() argument that enables you to specify how containers of different sizes are handled. The available Count() options are: "Shortest", "Longest", "Enforce Equal", and N, where N is a number. If you specify a number, the function iterates through all containers exactly N times; note that the function loops back to the start of containers that have fewer than N items.

locals A list of variables that are local to the function. This is equivalent to other Local variable initializations in JSL. The initialization of the local variables occurs after the first loop control variables have been set, but the local variables do not get initialized again after that.

body Any number of valid JSL expressions, glued together if there are more than one. You can use the Continue() function to skip to the next iteration. You can also use Break() function to stop iteration through the loop and proceed to the next expression that follows the loop. See "Break and Continue".

Example

```
For Each( {value, index}, {10, 20, 30}, Show( value, index ) );
  value = 10;
  index = 1;
  value = 20;
  index = 2;
```

```
value = 30;
index = 3;
```

For Each Row(<*dt*>, *script*)

Description

Repeats the *script* on each row of the data table.

Returns

Null.

Required Argument

script Any valid JSL expressions.

Optional Argument

dt Positional argument that is a reference to a data table. If this argument is not in the form of an assignment, then it is considered a data table expression.

Example

The following example creates data table references and then iterates over each row in Big Class.jmp. If the value of age in a row is greater than 15, the age is printed to the log.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( dt, If( :age > 15, Show( :age ) ) );
```

If(*condition1*, *result1*, <*condition2*, *result2*>, ..., <*elseResult*>)

Description

Evaluates the first of each pair of arguments and returns the evaluation of the *result* expression associated with the first *condition* argument that evaluates to a nonzero result. The *condition* arguments are evaluated in order. If all of the *condition* arguments evaluate to zero, the optional *elseResult* is evaluated and the result is returned. If no *elseResult* is specified, and none of the conditions are true, a missing value is returned. If all of the *condition* arguments evaluate to missing, a missing value is returned.

IfMax(*expr1*, *result1*, *expr2*, *result2*, ... <*all missing results*>)

Description

Evaluates the first of each pair of arguments and returns the evaluation of the result expression (the second of each pair) associated with the maximum of the expressions. If more than one expression is the maximum, the first maximum is returned. If all expressions are missing and a final result is not specified, missing is returned. If all expressions are missing and a final result is specified, that final result is returned. The test expressions must evaluate to numeric values, but the result expressions can be anything.

Returns

The result expression associated with the maximum of the expressions

IfMin(*expr1*, *result1*, *expr2*, *result2*, ... <*all missing results*>)

Description

Evaluates the first of each pair of arguments and returns the evaluation of the result expression (the second of each pair) associated with the minimum of the expressions. If more than one expression is the minimum, the first minimum is returned. If all expressions are missing and a final result is not specified, missing is returned. If all expressions are missing and a final result is specified, that final result is returned. The test expressions must evaluate to numeric values, but the result expressions can be anything.

Returns

The result expression associated with the minimum of the expressions

IfMZ(*condition1*, *result1*, <*condition2*, *result2*>, ..., <*elseResult*>)

Description

Evaluates the first of each pair of arguments and returns the evaluation of the *result* expression associated with the *condition1* argument that evaluates to a nonzero result. The *condition* arguments are evaluated in order. If all of the *condition* arguments evaluate to zero or missing, the optional *elseResult* is evaluated and the result is returned. If no *elseResult* is specified, and none of the conditions are true, a missing value is returned.

Notes

The test arguments are evaluated in order until the first nonzero result. If all test results return zero or missing, the *elseExpr* argument is evaluated.

IfMZ() is equivalent to If() where missing values for evaluated *condition* arguments are treated as zero.

Interpolate(*x*|*xmatrix*|*xlist*, *x1*, *y1*, *x2*, *y2*, ...)

Interpolate(*x*|*xmatrix*|*xlist*, *xmatrix*, *ymatrix*)

Interpolate({*x*, *y*}, *xvector*, *yvector*, *zmatrix*)

Description

Performs linear interpolation for continuous data. There are many ways to specify the function.

In the simplest cases where the first argument is a single numeric value, the *y* value corresponding to a given *x* value between two sets of points or by matrices *xmatrix* and *ymatrix*. The *x* values, (*x1*, *x2*, ...) or *xmatrix*, must be in ascending order.

If the first argument is a matrix or list of numeric values, the resulting matrix or list is a set of interpolated values.

You can perform *bilinear interpolation* using the four-argument case. Here, the first argument is a list of two points, the second and third arguments are vectors that define the grid of x and y values, and the fourth argument is a matrix of data points. The function then finds the interpolated *z* value within the appropriate quadrant of the *zmatrix*.

Returns

The interpolated value or values. In the three-argument cases, the return object type matches the type of the first argument. In the four-argument case, the return object is a number.

`Is Associative Array(name)`

Description

Returns 1 if the evaluated argument is an associative array, or 0 otherwise.

`Is Empty(global)`

`Is Empty(dt)`

`Is Empty(col)`

Description

Returns 1 if the *global* variable, data table, or data column is undefined or holds the `Empty()` value, or 0 otherwise.

`Is Expr(x)`

Description

Returns 1 if the evaluated argument is an expression, or 0 otherwise.

`Is List`

See "Is List(x)".

`Is Name(x)`

Description

Returns 1 if the evaluated argument is a name, or 0 otherwise.

`Is Namespace(namespace)`

Description

Returns 1 if the namespace argument is a namespace; returns 0 otherwise.

`Is Number(x)`

Description

Returns 1 if the evaluated argument is a number or missing numeric value, or 0 otherwise.

`Is Scriptable(x)`

Description

Returns 1 if the evaluated argument is a scriptable object, or 0 otherwise.

`Is String(x)`

Description

Returns 1 if the evaluated argument is a quoted string, or 0 otherwise.

`Match(x, value1, result1, value2, result2, ..., resultElse)`

Description

If *a* is equal to *value1*, then *result1* is returned. If *a* is equal to *value2*, *result2* is returned, and so on.

Notes

The `Match()` function explicitly checks to see if the compare expression *x* is missing and if the value of *value1* is missing, then it returns the value of `result1`; otherwise it continues to compare the expression *x* to each *valueN* value in each *valueN/resultN* pair, ignoring any missing values. If the expression *x* is equal to any of the *valueN* value, then the corresponding *resultN* value is returned. If no matching *valueN* value is found, then the *resultElse* value is returned.

`MatchMZ(x, value1, expr1, value2, expr2, ..., exprElse)`

Description

Evaluates and returns the *exprN* argument that equals *x* or evaluates and returns the *exprElse* argument if no value equals *x*.

Notes

The `MatchMZ()` function works the same as the `Match()` function except that missing values are treated as 0.

`Not(a)`

`!a`

Description

The logical Not.

Returns

0 (false) if *a*>0.

1 (true) if *a*<=0.

Missing value if *a* is missing.

Argument

a Any variable or number. The variable must have a numeric or matrix value.

Notes

Mostly used for conditional statements and loop control.

`Or(a, b)`

`a|b`

Description

The logical Or.

Returns

1 (true) if either of or both *a* and *b* are true.

0 (false) otherwise.

Missing if either are missing.

Arguments

a, *b* Any variable or number.

Notes

Mostly used for conditional statements and loop control.

`OrMZ(a, b)`

`a:|b`

Description

Returns the logical OR of all arguments with missing values treated as zeroes: 1 if any arguments are nonzero and 0 otherwise.

Returns

1 (true) if either of or both *a* and *b* are true.

0 (false) otherwise.

Arguments

a, *b* Any variable or number.

Notes

– Mostly used for conditional statements and loop control.

– `Or()` returns missing if any evaluated argument is missing. `OrMZ()` returns 0 if any evaluated argument is missing.

Return(<expr1>, <expr2>, ..., <exprN>)

Description

Returns an expression value from a user-defined function.

Example

This example returns the evaluation of both expressions in the `Return()` function as a list. The `Return()` function can have more than one argument. If only one is present, then the value of the expression is returned. If more than one is present, then the values of all the expressions is returned in a list.

```
f = Function( {a, b},
  Return( a - b, a + b )
);
{lo, hi} = f( 10, 1 );
Show( lo, hi );
Show( f( 7, 15 ) );
   lo = 9;
   hi = 11;
   f(7, 15) = {-8, 22};
```

Notes

`Return()` not enclosed by a function, method, or recursive function call causes an error.

Step(x0, x1, y1, x2, y2, ...)

Step(x0, [x1, x2, ...], [y1, y2, ...])

Description

Returns the *y* argument corresponding to the largest *x* argument that is less than or equal to *x0*. The *x* points must be specified in ascending order.

Stop()

Description

Immediately stops a script that is running.

Transform Each(names, container, <Output(type)>, <locals>, body)

Description

Iterates over a container, which can be a list, an associative array, or a matrix, and updates each of the values in the container based on the evaluation of a JSL expression at each iteration. The value, key, or element, as well as the index number, are all available at each iteration. For associative array containers, the key and value can be accessed using a

two-item list. For matrix containers, a linear index is provided by default, but a two-item list can be used to access the row and column indices. These symbols are provided within the body of the loop only. A list of local variables can also be provided. If local variables are defined, they are initialized after the first iteration symbols are set.

Returns

An updated version of the original container. The return object has the same type as the original container, unless the `Output()` keyword is used to designate a different type for the returned container.

Arguments

`names` Specification of loop control variable names, specified as a list. The form of the list is determined by the type of container. All of the names are optional, so you need to specify them only if you need to refer to them in the *`locals`* or *`body`* arguments. If you do not specify any names, the first argument should be an empty list, which can be specified as `{}` or `List()`.

For a *list* container, the names list contains a name for each value in the list and a name for the index of each value in the list.

For an *associative array* container, the names list contains a two-item list of names and a name for the index of each item in the associative array.

For a *matrix* container, the names list contains a name for each element in the matrix and a second argument for the index of each element in the matrix. The second argument can be specified as a single name or as a two-item list of names that represent symbols for the row and column indices.

When specifying multiple containers using the `Across()` keyword, the names list contains a list of names that refer to values in each container. The number of names in the first item in the names list must match the number of containers specified in the `Across()` keyword.

`container` A list, associative array, or a matrix. The container can be defined in the argument or it can be a reference to a previously defined object.

This argument can also use the `Across()` keyword, which enables you to use the function across multiple containers. The multiple containers can be specified as separate arguments or as items in a list. The `Across()` keyword has an optional `Count()` argument that enables you to specify how containers of different sizes are handled. The available `Count()` options are: `"Shortest"`, `"Longest"`, `"Enforce Equal"`, and `N`, where `N` is a number. If you specify a number, the function iterates through all containers exactly N times; note that the function loops back to the start of containers that have fewer than N items.

`Output(type)` Specifies a type for the output. This can be `"List"`, `"Matrix"`, or `"Associative Array"`. By default, the output type matches the type of the input container.

`locals` A list of variables that are local to the function. This is equivalent to other Local variable initializations in JSL. The initialization of the local variables occurs after the

first loop control variables have been set, but the local variables do not get initialized again after that.

body Any number of valid JSL expressions, glued together if there are more than one. The result of the JSL expressions at each iteration is used in the output container. You can use the `Continue()` function to return no value for an iteration and skip to the next iteration. You can also use `Break()` function to stop iteration through the loop and proceed to the next expression that follows the loop. See "Break and Continue".

Example

```
values = Transform Each( {x}, {10, 20}, x + 10 );
Show( values );
   values = {20, 30};
```

While(*expr*, *body*)

Description

Repeatedly tests the *expr* condition and executes the *body* until the *expr* condition is no longer true.

Zero Or Missing(*expr*)

Description

Returns 1 if *expr* yields a missing value or zero, 0 otherwise.

Constant Functions

JMP provides functions for two useful constant functions.

Note: These functions do not take an argument, but the parentheses are required.

e()

Description

Returns the constant *e*, which is 2.7182818284590451...

Pi()

Description

Returns the constant π, which is 3.1415926535897931...

Date and Time Functions

Datetime values are handled internally as numbers of seconds since midnight, January 1, 1904.

The expression `x=01Jan1904` sets *x* to zero, since the indicated date is the base date or "zero date" in JMP. If you examine the values of dates, they should be appropriately large numbers (for example, `5oct1998` is 2990390400).

Abbrev Date(*date*)

Description

Converts the provided *date* to a quoted string.

Returns

A quoted string representation of the date.

Argument

date Can be the number of seconds since the base date (midnight, January 1, 1904), or any date-time operator.

Example

```
Abbrev Date( 29Feb2004 );
```

02/29/2004

As Date(*x*)

Description

Formats the number or expression *x* so that it shows as a date or duration when displayed in a text window. Values that represent one year or more are returned as dates. Values that represent less than a year are returned as durations.

Returns

A date that is calculated from the number or expression provided.

Argument

x Number or expression.

Date Difference(*datetime1*, *datetime2*, *interval name*, <*alignment*>)

Description

Returns the difference in intervals of two date-time values.

Returns

A number.

Required Arguments

datetime1, *datetime2* Date-time values.

interval name A quoted string that contains a date-time interval, such as "month", "day", or "hour".

Optional Arguments

alignment A quoted string:

- "Start" includes full or partial intervals.
- "Actual" counts only whole intervals.
- "Fractional" returns fractional differences using averages for "year", "quarter", and "month" intervals.

Date DMY(*day*, *month*, *year*)

Description

Constructs a date value from the arguments.

Returns

The specified date, expressed as the number of seconds since midnight, 1 January 1904.

Arguments

day The numeric day of month, 1-31. Note that there is no error-checking, so you can enter February 31.

month The numeric month, 1-12.

year The year.

Date Increment(*datetime*, *interval name*, <*increment*>, <*alignment*>)

Description

Adds 1 or more intervals to a starting datetime value.

Returns

Returns the new datetime value.

Required Arguments

datetime The starting datetime value.

interval name A quoted string that contains the name of a datetime interval. "year", "quarter", "month", "week", "day", "hour", "minute", and "second" are supported.

Optional Arguments

increment A number that specifies the number of intervals. The default value is 1.

alignment A quoted string that contains a keyword:

- "Start" truncates the date to the nearest interval prior to adding the increment. For example, it removes the time and outputs the date. "start" is the default value.
- "Actual" retains the full input datetime value.

– `"Fractional"` allows fractional incremental values using averages for the duration of `"Year"`, `"Quarter"`, and `"Month"` intervals.

`Date MDY(`*`month`*`, `*`day`*`, `*`year`*`)`

Description

Constructs a date value from the arguments.

Returns

The specified date, expressed as the number of seconds since midnight, 1 January 1904.

Arguments

`month` The numeric month, 1-12.

`day` The numeric day of month, 1-31. Note that there is no error-checking, so you can enter February 31.

`year` The year.

`Day(`*`datetime`*`)`

Description

Determines the day of the month supplied by the *`datetime`* argument.

Returns

Returns an integer representation for the day of the month of the date supplied.

Arguments

`datetime` The number of seconds since midnight, 1 January 1904. This can also be an expression.

Example

```
d1 = Date DMY( 12, 2, 2003 );
   3127852800

Day( 3127852800 );
   12
Day( d1 );
   12
```

`Day Of Week(`*`datetime`*`)`

Description

Determines the day of the week supplied by the *`datetime`* argument.

Returns

Returns an integer representation for the day of the week of the date supplied. Weeks are Sunday–Saturday.

Arguments

datetime The number of seconds since midnight, 1 January 1904. This can also be an expression.

Day Of Year(*datetime*)

Description

Determines the day of the year supplied by the *datetime* argument.

Returns

Returns an integer representation for the day of the year of the date supplied.

Arguments

datetime The number of seconds since midnight, 1 January 1904. This can also be an expression.

Format(*x*, formatString, *width*|<*width*, *decimal places*>, <"Use Thousands Separator">)

Format(*x*, "Best", <*width*>, <"Use Thousands Separator">)

Format(*x*, ("Fixed Dec"|"Percent"), *width*|<*width*, *decimal places*>, <"Use Thousands Separator">)

Format(*x*, "Pvalue", <*width*>)

Format(*x*, ("Scientific"|"Engineering"|"Engineering SI"), <*width*>|<*width*, *decimal places*>)

Format(*x*,"Precision", *width*|<*width*, *decimal places*>, <"Use Thousands Separator">, <"Keep Trailing Zeroes">, <"Keep All Whole Digits">)

Format(*x*, "Currency", <*currency code*>, <*width*>|<*width*, *decimal places*>, <"Use Thousands Separator">, < <<Use Locale(*Boolean*) >)

Format(*x*, *datetime*, <*width*>)

Format(*x*, ("Latitude DDD"|"Latitude DDM"|"Latitude DMS"|"Longitude DDD"|"Longitude DDM"|"Longitude DDM"), *width*|<*width*, *decimal places*>, ("PUN"|"DIR"|"PUNDIR"))

Format(*x*, "Custom", Formula(), <*width*>)

Description

Converts the value *x* into the quoted *format* that you specify in the subsequent arguments.

Returns

Returns the text that corresponds to the number in the specified format.

Arguments

See *Using JMP* for more information about the arguments. The arguments are also shown in a data table column's Column Info window.

Examples

```
Format( x, "Fixed Dec", 10, 2, "Use Thousands Separator");
Format( x, "Currency", "EUR", 20, <<Use Locale(0)); // ignores computer locale
Format( x, "m/d/y", 10 );
Format( x, "Precision", 10, 2, "Keep trailing zeroes", "Keep All Whole Digits"
  );
Format( x, "Latitude DDD", "PUNDIR"); // "PUN" for punctuation, "DIR" for
  direction, PUNDIR for both
Format( x, "Custom", Formula( Abs( value ) ), 15 );
```

Notes

- You must always precede the number of decimal places with the width.
- If the date format is unknown, an error is written to the log.

Format Date(*x*, *datetime*, <*width*>)

Description

Converts the value of *x* into the quoted *datetime* that you specify in the second argument. Format choices are those shown in the data table Column Info window.

Returns

Returns the number in the specified format.

Arguments

See *Using JMP* for more information about the arguments.

Example

```
Format Date( Today(), "yyyQq" );
  "04/03/2020"
```

Hour(*datetime*, <"12"|"24">)

Description

Determines the hour supplied by the *datetime* argument.

Returns

Returns an integer representation for the hour part of the date-time value supplied.

Arguments

`datetime` The number of seconds since midnight, 1 January 1904. This can also be an expression.

`"12"|"24"` Changes the mode to 12 hours (with am and pm). The default is 24-hour mode.

HP Time()

Description

Returns a high precision time value (in microseconds). This function is only useful relative to another `HP Time()` value. The time value represents the number of microseconds since the start of the JMP session.

Notes

For less precise time values use `Tick Seconds()`.

In Days(*n*)

Description

Returns the number of seconds per *n* days. Divide by this function to express seconds as days.

Informat(*string*, *format*)

Parse Date(*string*, *format*)

Description

Parses a quoted *string* of a given quoted *format* and returns a date/time value. The value is expressed as if surrounded by the `As Date()` function, returning the date in "*ddMonyyyy*" format.

Example

```
Informat( "07152000", "MMDDYYYY" );
```

15Jul2000

Notes

- To see the format options, open the Column Info window on a data table column, select a date/time value for the format, and view the Input Format list.
- If the date format is unknown, an error is written to the log.

See Also

"As Date(x)"

In Hours(*n*)

Description

Returns the number of seconds per *n* hours. Divide by this function to express seconds as hours.

In Minutes(*n*)

Description

Returns the number of seconds per *n* minutes. Divide by this function to express seconds as minutes.

In Weeks(*n*)

Description

Returns the number of seconds per *n* weeks. Divide by this function to express seconds as weeks.

In Years(*n*)

Description

Returns the number of seconds per *n* years. Divide by this function to express seconds as years.

ISO Year(*datetime*)

Description

Returns the ISO year of the *datetime* supplied. ISO years correspond to ISO weeks; they begin on the Monday of the first week that contains at least four days.

Long Date(*date*)

Description

Returns a locale-specific quoted string representation for the *date* supplied, formatted like "Sunday, February 29, 2004" or "Wednesday, November 9, 2011".

MDYHMS(*date*)

Description

Returns a quoted string representation for the *date* supplied, formatted like "2/29/04 00:02:20".

Minute(*datetime*)

Description

Determines the minute supplied by the *datetime* argument, 0-59.

Returns

Returns an integer representation for the minute part of the date-time value supplied.

Month(*date*)

Description

Returns an integer representation for the month of the *date* that is supplied.

Nth Day of Week in the Month(*datetime*)

Description

Determines the day of the week of the *datetime* argument and how many instances of that day of the week have occurred in the month of the *datetime* argument.

Returns

Returns an integer that represents the number of instances of the day of the week of the *datetime* argument that have occurred in the month.

Parse Date()

See "Informat(string, format)".

Quarter(*datetime*)

Description

Returns the annual quarter of a *datetime* value as an integer 1-4.

Second(*datetime*)

Description

Determines the second supplied by the *datetime* argument.

Returns

Returns an integer representation for the second part of the date-time value supplied.

Argument

datetime Number of seconds since midnight, 1 January 1904. This can also be an expression.

Short Date(*date*)

Description

Returns a quoted string representation for the *date* supplied in the format MM/DD/YYYY.

Tick Seconds()

Description

Measures the time taken for a script to run, measured down to the 60th of a second.

Notes

For higher time value resolution (for example, microseconds) use the `HP Time()` function.

`Time Of Day(datetime)`

Description

Returns an integer representation for the time of day of the *datetime* supplied.

`Today()`

Description

Returns the current date and time expressed as the number of seconds since midnight, 1 January 1904. No arguments are available, but the parentheses are still necessary.

`Week Of Year(date, <ruleN>)`

Description

Returns the week of the year that contains a date-time value. Three rules determine when the first week of the year begins.

- With rule 1 (the default), weeks start on Sunday, with the first Sunday of the year being week 2. Week 1 is a partial week or empty.
- With rule 2, the first Sunday begins with week 1, with previous days being week 0.
- With rule 3, the ISO-8601 week number is returned. Weeks start on Monday. Week 1 is the first week of the year with four days in that year. It is possible for the first or last three days of the year to belong to the neighboring year's week number.

`Year(date)`

Description

Returns an integer representation for the year of *date*.

Discrete Probability Functions

`Beta Binomial Distribution(k, p, n, delta)`

Description

Returns the cumulative distribution function (cdf) of the beta binomial distribution. This is the probability that a beta binomially distributed random variable is less than or equal to *k*. The cdf is calculated as the summation of the beta binomial pmf for values of X from 0 to *k*.

Arguments

k The count of interest. *k* must be an integer.

p The probability of success for each trial, which must be between 0 and 1.

n The number of trials, which must be greater than 1.

delta The overdispersion parameter, which must be between Maximum[-*p*/(*n*-*p*-1), -(1-*p*)/(*n*-2+*p*)] and 1. When the overdispersion parameter is zero, the distribution reduces to Binomial(*n*, *p*).

Beta Binomial Probability(*k*, *p*, *n*, *delta*)

Description

Returns the probability mass function (pmf) of the beta binomial distribution. This is the probability that a beta binomially distributed random variable is equal to *k*. The pmf is parameterized as follows:

$$P(X = k;p, n, \delta) = \binom{n}{k} \frac{\Gamma\left(\frac{1}{\delta}-1\right)\Gamma\left[k+p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[n-k+(1-p)\left(\frac{1}{\delta}-1\right)\right]}{\Gamma\left[p\left(\frac{1}{\delta}-1\right)\right]\Gamma\left[(1-p)\left(\frac{1}{\delta}-1\right)\right]\Gamma\left(n+\frac{1}{\delta}-1\right)}$$

Arguments

k The count of interest. *k* must be an integer.

p The probability of success for each trial, which must be between 0 and 1.

n The number of trials, which must be greater than 1.

delta The overdispersion parameter δ, which must be between Maximum[-*p*/(*n*-*p*-1), -(1-*p*)/(*n*-2+*p*)] and 1. When the overdispersion parameter is zero, the distribution reduces to Binomial(*n*, *p*).

Notes

The beta binomial distribution results from assuming that X|π follows a Binomial(n,π) distribution and π follows a Beta(p(1-δ)/δ,(1-p)(1-δ)/δ) distribution. It is useful when the data are a combination of several Binomial distributions that each have different probabilities of success.

Beta Binomial Quantile(*p*, *n*, *delta*, *cumprob*)

Description

Returns the smallest integer quantile for which the cumulative probability of the Beta Binomial(p, n, delta) distribution is larger than or equal to *cumprob*.

Arguments

p The probability of success for each trial. *p* must be between 0 and 1.

n The number of trials, which must be greater than 1.

delta The overdispersion parameter δ, which must be between Maximum[-*p*/(*n*-*p*-1), -(1-*p*)/(*n*-2+*p*)] and 1. When the overdispersion parameter is zero, the distribution reduces to Binomial(*n*, *p*).

cumprob The cumulative probability of the quantile desired. *cumprob* must be between 0 and 1.

Binomial Distribution(*p*, *n*, *k*)

Description

Returns the cumulative distribution function (cdf) of the binomial distribution. This is the probability that a binomially distributed random variable is less than or equal to *k*. The cdf is calculated as the summation of the binomial pmf for values of X from 0 to *k*.

Arguments

p The probability of success for each trial. *p* must be between 0 and 1.

n The number of trials.

k The number of successes, which must be less than or equal to *n*.

Binomial Probability(*p*, *n*, *k*)

Description

Returns the probability mass function (pmf) of the binomial distribution. This is the probability that a binomially distributed variable is equal to *k*. The pmf is parameterized as follows:

$$P(X = k; p, n) = \binom{n}{k} p^k (1-p)^{n-k}$$

Arguments

p The probability of success for each trial. *p* must be between 0 and 1.

n The number of trials.

k The number of successes, which must be less than or equal to *n*.

Binomial Quantile(*p*, *n*, *cumprob*)

Description

Returns the smallest integer quantile for which the cumulative probability of the Binomial(p, n) distribution is larger than or equal to *cumprob*.

Arguments

p The probability of success for each trial. *p* must be between 0 and 1.

n The number of trials.

cumprob The cumulative probability of the quantile desired. *cumprob* must be between 0 and 1.

Gamma Poisson Distribution(*k*, *lambda*, *sigma*)

Description

Returns the cumulative distribution function (cdf) of the gamma-Poisson distribution. This is the probability that a gamma-Poisson distributed random variable is less than or equal to *k*. The cdf is calculated as the summation of the gamma-Poisson pmf for values of X from 0 to *k*.

Arguments

k The count of interest. *k* must be an integer.

lambda The shape parameter λ, which much be greater than 0. This is the mean of the distribution.

sigma The overdispersion parameter σ, which must be greater than or equal to 1. When the overdispersion parameter is 1, the distribution reduces to a Poisson(λ) distribution.

Gamma Poisson Probability(*k*, *lambda*, *sigma*)

Description

Returns the probability mass function (pmf) of the gamma-Poisson distribution. This is the probability that a gamma-Poisson distributed random variable is equal to *k*. The pmf is parameterized as follows:

$$P(X = k;\lambda, \sigma) = \frac{\Gamma\left(k + \frac{\lambda}{\sigma - 1}\right)}{\Gamma(k+1)\Gamma\left(\frac{\lambda}{\sigma - 1}\right)}\left(\frac{\sigma - 1}{\sigma}\right)^{k}\left(\sigma^{-\left(\frac{\lambda}{\sigma - 1}\right)}\right)$$

where Γ(·) is the Gamma function.

Arguments

k The count of interest. *k* must be an integer.

lambda The shape parameter λ, which much be greater than 0. This is the mean of the distribution.

sigma The overdispersion parameter σ, which must be greater than or equal to 1. When the overdispersion parameter is 1, the distribution reduces to a Poisson(λ) distribution.

Notes

The gamma Poisson distribution results from assuming that X|μ follows a Poisson(μ) distribution and μ follows a Gamma(λ/(σ-1),σ-1) distribution. It is useful when the data are a combination of several Poisson(μ) distributions that each have different values of μ.

Gamma Poisson Quantile(*lambda*, *sigma*, *cumprob*)

Description

Returns the smallest integer quantile for which the cumulative probability of the Gamma Poisson(lambda, sigma) distribution is larger than or equal to *cumprob*.

Arguments

lambda The shape parameter λ, which much be greater than 0. This is the mean of the distribution.

sigma The overdispersion parameter σ, which must be greater than or equal to 1. When the overdispersion parameter is 1, the distribution reduces to a Poisson(λ) distribution.

cumprob The cumulative probability of the quantile desired. *cumprob* must be between 0 and 1.

Hypergeometric Distribution(*N*, *K*, *n*, *x*, <*r*>)

Description

Returns the cumulative distribution function (cdf) of the hypergeometric distribution. This is the probability that a hypergeometrically distributed random variable is less than or equal to *x*. The cdf is calculated as the summation of the hypergeometric pmf for values of X from 0 to *x*.

Required Arguments

N The population size.

k The number of items in the category of interest.

n The sample size.

x The count of interest, which must be less than or equal to *n* and *k*.

Optional Argument

r The odds ratio.

Hypergeometric Probability(*N*, *k*, *n*, *x*, <*r*>)

Description

Returns the probability mass function (pmf) of the hypergeometric distribution. This is the probability that a hypergeometrically distributed random variable is equal to *x*. The pmf is parameterized as follows:

$$P(X = x;N, n, k) = \frac{\binom{k}{x}\binom{N-k}{n-x}}{\binom{N}{n}}, n - x \le N - k$$

Required Arguments

N The population size.

k The number of items in the category of interest.

n The sample size.

x The count of interest, which must be less than or equal to *n* and *k*.

Optional Argument

r The odds ratio.

Neg Binomial Distribution(*p*, *n*, *k*)

Description

Returns the cumulative distribution function (cdf) of the negative binomial distribution. This is the probability that a negative binomially distributed random variable is less than or equal to *k*. The cdf is calculated as the summation of the negative binomial pmf for values of X from 0 to *k*.

Arguments

p The probability of success for each trial. *p* must be between 0 and 1.

n The number of successes.

k The number of failures before the n^{th} success.

Neg Binomial Probability(*p*, *n*, *k*)

Description

Returns the probability mass function (pmf) of the negative binomial distribution. This is the probability that a negative binomially distributed random variable is equal to *k*. The pmf is parameterized as follows:

$$P(X = k; p, n) = \binom{n + k - 1}{k} p^n (1 - p)^k$$

Arguments

p The probability of success for each trial. *p* must be between 0 and 1.

n The number of successes.

k The number of failures before the n^{th} success.

Notes

The return value of the pmf is the probability of observing the n^{th} success after *k* failures have occurred.

Poisson Distribution(*lambda*, *k*)

Description

Returns the cumulative distribution function (cdf) of the Poisson distribution. This is the probability that a Poisson distributed random variable with mean *lambda* is less than or equal to *k*. The cdf is calculated as the summation of the Poisson pmf for values of X from 0 to *k*.

Arguments

k The number of events in a given time interval. *k* must be an integer.

lambda The shape parameter λ, which much be greater than 0. This is the mean of the distribution.

`Poisson Probability(`*lambda*`, k)`

Description

Returns the probability mass function (pmf) of the Poisson distribution. This is the probability that a Poisson distributed random variable with mean *lambda* is equal to *k*. The pmf is parameterized as follows:

$$P(X = k;\lambda) = \frac{e^{-\lambda}\lambda^{k}}{k!}$$

Arguments

k The number of events in a given time interval. *k* must be an integer.

lambda The shape parameter λ, which must be greater than 0. This is the mean of the distribution.

`Poisson Quantile(`*lambda*`, cumprob)`

Description

Returns the smallest integer quantile for which the cumulative probability of the Poisson(lambda) distribution is larger than or equal to *cumprob*.

Arguments

lambda The shape parameter λ, which must be greater than 0. This is the mean of the distribution.

cumprob The cumulative probability of the quantile desired. *cumprob* must be between 0 and 1.

Display Functions

`Alpha Shape(Triangulation())`

Description

Returns the alpha shape for the given triangulation.

Border Box(<Left(*pix*)>, <Right(*pix*)>, <Top(*pix*)>, <Bottom(*pix*)>, <Sides(*Boolean*)>, *db*)

Description

Constructs a bordered display box that contains another display box. Optional arguments (Left, Right, Top, Bottom) add space between the border box and what it contains. The other optional argument (Sides) draws borders around the border box on any single side or combination of sides; draws the border in black or the highlight color; makes the background transparent or white or erases the background of a display box that contains it.

Returns

The display box.

Required Argument

db A display box object (for example, a text box or another border box).

Optional Arguments

Left(*pix*) An integer that measures pixels.

Right(*pix*) An integer that measures pixels.

Top(*pix*) An integer that measures pixels.

Bottom(*pix*) An integer that measures pixels.

Sides(*pix*) An integer that maps to settings for the display box.

Notes

The formula for deriving the integer for Sides is: 1*top + 2*left + 4*bottom + 8*right + 16*highlightcolor + 32*whitebackground + 64*erase. Thus, if you want to just draw a black border on the top and bottom, 1+4 = 5. If you want that same box with a white background, 5+32 = 37.

Box Plot Seg(<*data*>, <*frequency*>, <*weight*>, <Vertical(*Boolean*)>)

Description

Returns a display seg that represents a box plot based on the passed x and y values.

Returns

The display box (a box plot).

Optional Arguments

data The data values within the box plot.

frequency The frequency values within the box plot.

weight The weights for observations on continuous Ys.

Vertical(*Boolean*) A vertical (1) or horizontal (0) box plot.

Example

```
win = New Window( "Box Plot Seg Example",
Graph Box(
```

```
      Frame Size( 40, 180 ),
      Y Scale( 0, 100 ),
      Box Plot Seg(
         [20, 30, 40], // data
         [1, 1, 3], // frequencies
         [1, 1, 1], // weights
         1 // vertical
         )
      )
   );
```

Busy Light(< <<Automatic(*Boolean*)>, <Size(*x*, *y*)>, < <<Disable>)

Description

Creates a rotating image to indicate a busy process.

Returns

A rotating image.

Optional Argument and Messages

<<Automatic(*Boolean*) Rotates the image.

Size(*x*, *y*) Specifies the size of the image in pixels.

<<Disable Hides the image.

Example

```
win = New Window( "Example",
   blb = Busy Light( <<Automatic( 1 ), Size( 50, 50 ) ) );
```

Button Box(*title*, *script*, < *<<Set Icon(path)*>, < <<Set Icon Location(left|right)>

Description

Constructs a button with the text *title* that executes *script* when clicked.

Returns

The display box (button box).

Required Arguments

title A quoted string or a string variable.

script A quoted string or a reference to a string that specifies a valid JSL script.

Optional Messages

<<Set Icon("*path*") Displays the image in the quoted pathname on the button. Most common graphic formats are supported, such as GIF, JPG, PNG, BMP, TIF. Since the title argument is optional, you can create a button with only a text title, with only an icon, or with both a text title and an icon. In the last case, the icon is placed next to the text title.

<<Set Icon Location("left"|"right") Allows the position of the icon on a button to be either left or right of the text.

Example

The following example shows a simple button box. When the user clicks the button box, "Pressed" is printed to the log.

```
win = New Window( "Simple Example",
   ex = Button Box( "Press me" )
);
ex << Set Script( Print( "Pressed" ) );
```

Notes

Line-break characters are ignored in button boxes.

Calendar Box(*title*, < <<Date>, < <<Min Date>, < <<Max Date>, < <<Show Time>)

Description

Constructs a pop-up calendar with selectable days and time.

Returns

The display box (calendar box).

Required Argument

title A quoted string or a string variable.

Optional Messages

<<Date The currently selected date.

<<Min Date The earliest date that can be selected.

<<Max Date The latest date that can be selected.

<<Show Time The time that can be specified.

Example

The following example creates a calendar with October 5, 1989 initially selected. The minimum date and maximum date are specified, so the user can select only dates in that range.

```
New Window( "Calendar Box Example", cal = Calendar Box() );
date = Date MDY (10, 5, 1989);
cal << Date( date );
cal << Show Time( 0 ); // omit the time

/* earliest date that can be selected is 60 days before 10/5/1989
"start" truncates the value so the time is not included */
cal << Min Date( Date Increment(date, "Day", -60, "start" ) );

// latest date that can be selected is 60 days after 10/5/1989
cal << Max Date( Date Increment(date, "Day",  60, "start" ) );
```

```
cal << Set Function( Function( {this}, Print( Abbrev Date(this << Get Date())
  ) ) ); // print the abbreviated date to the log
```

`Cell Plot(`*`Y(column(s))`*`, <`*`X(column)`*`>)`

Description

Displays each value in a cell graph.

`Check Box({`*`list`*`}, <`*`script`*`>, < <<Get(`*`n`*`)>, < <<Set(`*`n`*`, `*`Boolean`*`)>, < <<Get Selected>, < <<Enable Item(`*`n`*`, `*`Boolean`*`)>, < <<Item Enabled(`*`check box item`*`)>)`

Description

Constructs a display box to show one or more check boxes.

Returns

The display box (Check Box).

Required Argument

`list` A list of quoted strings or a reference to a list of strings.

Optional Argument

`script` An optional JSL script.

Optional Messages

`<<Get(`*`n`*`)` Returns 1 if the check box item specified by *`n`* is selected, or 0 otherwise.

`<<Set(`*`n`*`, `*`Boolean`*`)` Sets the check box item specified by *`n`* as either selected (1) or cleared (0).

`<<Get Selected` Returns a list of quoted strings that contain the names of the check box items that are selected.

`<<Enable Item(`*`n`*`, `*`Boolean`*`)` Sets the check box item specified by *`n`* as either enabled (1) or disabled (0). The state of a disabled check box cannot be changed.

`<<Item Enabled(`*`check box item`*`)` Returns 0 or 1 depending on whether the specific check box item is enabled.

Example

Create three check boxes labeled “one”, “two”, and “three”. The first check box is selected.

```
New Window( "Example", Check Box( {"one", "two", "three"}, <<Set( 1, 1 ) ) );
```

`Col Box(`*`title`*`, `*`display boxes`*`)`

Description

Returns a column box made up of the specified display boxes.

Arguments

`title` The quoted title of the column.

`display boxes` Display boxes that hold content within the column box.

Example

```
win = New Window( "Example",
   exx = 1;
   exy = 4;
   exz = 8;
   Table Box(
      String Col Box( "strings", {"x", "y", "z"} ),
      Col Box(
         "boxes",
         Slider Box( 0, 10, exx, Show( exx ) ),
         Slider Box( 0, 10, exy, Show( exy ) ),
         Slider Box( 0, 10, exz, Show( exz ) )
      )
   );
);
```

`Col List Box(<Data Table(`*name*`)>, <"All"|"Character"|"Numeric">, <width(`*pixels*`)>, <"Grouped">, <MaxSelected(`*n*`)>, <nLines(`*n*`)>, <MaxItems(`*n*`)>, <MinItems(`*n*`)>, <On Change(`*expr*`)>, < <<Modeling Type("Any"|"Continuous"|"Ordinal"|"Nominal"|Multiple Response"|"Unstructured Text"|"Vector"|"None"|"Row State") >, < <<Set Data Type(Any|Numeric|Character)>, <`*script*`>)`

Description

Constructs a display box to show a list box that allows selection of data table columns.

Returns

The display box (Col List Box).

Optional Arguments

`Data Table(`*name*`)` The quoted name of the data table.

`"All" | "Character" | "Numeric"` Adds all columns of the current data table into the list. Omitting `"All"` results in an empty col list box with the “optional” label. To display “optional character”, specify `"Character"`. To display “optional numeric”, specify `"Numeric"`.

`width(`*pixels*`)` Sets the width of the list box to `pixels`. `pixels` is a number that measures pixels.

`"Grouped"` Displays grouped columns in the box.

`MaxSelected(`*n*`)` Sets whether only one item can be selected. For *n*>1, *n* is ignored.

`nLines(`*n*`)` Sets the length of the list box to n number of lines. *n* is an integer.

script A script.

`MaxItems(`*n*`)` A number that allows only *n* columns to be added to the list.

`MinItems(`*n*`)` A number that only requires at least *n* columns for the list. If *n*=2, the top two slots in the col list box an initial display of “required numeric” (or whatever you set the data type to be).

On Change(*expr*) Evaluates the expression when the selection in the list changes. Dragging between two column list boxes that have this argument results in both expressions being evaluated. The expression for the target being dragged is evaluated first, then the expression for the source is evaluated.

Optional Messages

<<Set Tips({*Tip text 1*, *Tip text 2*, ...}) Quoted strings that set tool tips for items in the list box. A quoted null string or an empty list results in no tips. A list shorter that the list of items in the list box will use the last tip text for the remaining items in the list and the list box.

<<Set Tip(*Tip text*) A quoted string that overrides any tool tips set using Set Tips() function. If there is a tip set for the box, you cannot set tips for each individual item.

Using Set Tip() with no arguments clears the list box tip and allows the individual item tool tips to be displayed.

Notes

- The MaxSelected(*n*) argument only affects whether one or more than one item can be selected. It does not enforce a limit greater than 1.
- Specialty modeling types can be used only in a role (determined by the platform) that explicitly accepts columns of the same type.

Col Span Box(*title*, *display box args*)

Description

Creates spanned columns headers inside a table box. The top column header spans two child column headers.

Returns

The display box (a Col Span Box).

Arguments

title The title that appears in the box.

display box args Display boxes.

Example

```
win = New Window( "Col Span Box",
   <<Modal,
   Table Box(
      Col Span Box(
         "Confidence Limits",
         neb = Number Col Edit Box( "Upper limits", [0, 0] ),
         Number Col Edit Box( "Lower limits", [0, 0] )
      )
   )
);
```

Combo Box(*{items <(tip string)>, ...}*, *<script>*)

Description

Constructs a display box to show a drop-down list.

Returns

The display box (Combo Box).

Arguments

items The items that the user can select.

tip string A quoted string that specifies tooltip text.

script An optional JSL script.

Context Box(*display box*, ...)

Description

Defines a scoped evaluation context. Each Context Box is executed independently of each other.

Returns

A display box.

Arguments

Any number of display boxes.

Contour Seg(Triangulation, [levels], <zColor([*colors*]|{*colors*}, <"Cycle Colors"|"Interpolate Colors">)>, <"Fill"|"Fill Between"|"Fill Below"|"Fill Above">, <Transparency([]|*t*)>)

Description

Returns a display seg that represents contours of a Triangulation.

Required Arguments

Triangulation The columns to include in the Triangulation.

[levels] A matrix of values that control the contour levels that are drawn.

Optional Arguments

zColor([*colors*]|{colors} Colors for each level, specified as a list or matrix.

"Cycle Colors"|"Interpolate Colors Cycle Colors" alternates the colors (for example, red, green, red, green). With Interpolate Colors, the first contour is red, and the last is green. The contours between smoothly blend the colors.

"Fill Below"|"Fill Between"|"Fill Above"|"Fill" "Fill Below" fills the region below the lines. "Fill Between" fills only the middle region. "Fill Above" fills the region above the lines.. "Fill" works like "Fill Above", but the default is to display lines if no Fill options are specified.

Transparency([]|*t*) The transparency specified as a number or matrix.

Example

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
tri = Triangulation( X( :X, :Y ), Y( :POP ) );
{xx, yy} = tri << Get Points();
win = New Window( "Contour Seg Example",
   g = Graph Box(
      X Scale( Min( xx ) - .1, Max( xx ) + .1 ),
      Y Scale( Min( yy ) - .1, Max( yy ) + .1 ),
      Contour Seg(
         tri,
         [0, 400, 1000, 2000, 9000],
         zColor( 5 + [64 32 0 16 48] ),
         Transparency( [1, 1, 1, 1, 1] )
      )
   )
);
```

Notes

The triangulation is computed using the Xs, and the Y is a continuous variable defined at each position. The [levels] in this case defines values of POP that are drawn as lines, one line per level. If any Fill argument is specified, then the filled regions are [level1, level2], [level2, level3], ..., [level-n].

Current Journal()

Description

Gets the display box at the top of the current (topmost) journal.

Returns

Returns a reference to the display box at the top of the current journal.

Current Report()

Description

Gets the display box at the top of the current (topmost) report window.

Returns

Returns a reference to the display box at the top of the current report window.

Current Window()

Description

Returns a reference to the current window.

Data Filter Context Box(*display box*)

Description

Returns a display box that defines the extent of the local data filters that a display tree contains. Data filters and Data Filter Context Boxes can be arranged in a hierarchy and shared among platforms or boxes that the Data Filter Context Boxes contain.

Data Filter Source Box(*display box*)

Description

Defines which graph is the "source" of the selection filter. Selected rows in reports that are within the Data Filter Source box are included for analysis in the other reports that are within a common Data Filter Context Box.

Data Table Box(*data table*)

Description

Returns a table box the represents the specified data table.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Example", Data Table Box( dt ) );
```

Data Table Col Box(*column*)

Description

Returns a column box that corresponds to the specified data table column.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Example",
   Table Box( Data Table Col Box( :name ), Data Table Col Box( :height ) )
);
```

Dialog(*contents*)

Description

`Dialog` is deprecated. Use `New Window()` with the `Modal` argument instead.

Excerpt Box(*report*, *subscripts*)

Description

Returns a display box containing the excerpt designated by the report held at number *report* and the list of display subscripts *subscripts*. The subscripts reflect the current state of the report, after previous excerpts have been removed.

Expr As Picture(*Expr*(...), <Width(*pixels*)>)

Description

Converts expr() to a picture as it would appear in the Formula Editor.

Returns

Reference to the picture.

Required Argument

Expr(...) Place any valid JSL expression that can be displayed as a picture inside expr().

Optional Argument

Width(*pixels*) Sets the width of the box to pix. pix is a number that measures pixels.

Filter Col Selector(<Data Table(*name*)>, <Width(*pixels*)>, <nLines(*n*)>, <*script*>, <OnChange(*expr*)>)

Description

Returns a display box that contains a list of items. Control allows column filtering.

Get Project(*title*|*index*|*display box*|*window*)

Description

Returns a single project.

Examples

The following examples show how to get the window title of various projects.

```
Get Project( 1 ) << Get Window Title;
// first open project
Get Project( "My Project" ) << Get Window Title;
// first project named “My Project”
Get Project( display box ) << Get Window Title;
// parent project of the specified display box
```

Get Project List()

Description

Returns a list of all open projects.

Example

```
Get Project List() << Get Window Title;
// list of the titles of all open projects
```

Get Window(<Project(*title*|*index*|*display box*|*window*)>, <Type(*string*)>, *title*|*index*|*display box*)

Description

Returns a reference to a specific open window by title, index, or display box. When run in a project, Get Window() returns windows from the current project.

Optional Arguments

Project Specifies the title (a quoted string), index, display box, or window from another project.

Type(*string*) To limit the search to particular types of windows, use the Type() argument and one of these quoted strings: "Data Tables", "Journals", "Reports", or "Dialogs".

Examples

The following examples show how to get the window title of various windows.

```
Get Window( 1 ) << Get Window Title;
// first window in the current project

Get Window( "Big Class" ) << Get Window Title;
// Big Class window in the current project

Get Window( ob ) << Get Window Title;
// parent window of specified display box in the current project

Get Window( Project(), 1 ) << Get Window Title;
// first window, no project (global scope)

Get Window( Project( myProject ), "Big Class" << Get Window Title;
// Big Class window in the specified project
```

Get Window List(<Project(*title*|*index*|*display box*|*window*),><Type(*string*)>)

Description

Returns a list of currently open windows. By default, Get Window List() returns references to open windows in the current project. You can return an open window list from something other than the current project by using the Project() argument. To limit the search to particular types of windows, use the Type () argument and one of these quoted strings: "Data Tables", "Journals", "Reports", or "Dialogs".

Optional Arguments

Project Specifies the title, index, display box, or window from another project.

Type To limit the search to particular types of windows, use the Type () argument and one of these quoted strings: "Data Tables", "Journals", "Reports", or "Dialogs".

Examples

```
Get Window List() << Get Window Title;
```

```
// list of open windows in the current project
Get Window List( Type( "Reports" ) ) << Get Window Title;
// list of the titles of open reports in the current project
Get Window List( Project( 0 ), Type( "Reports" ) ); // positional arguments
// list of the titles of open reports outside of a project
Get Window List( 2 );
// second window list
```

Global Box(*global*)

Description

Constructs a box for editing *global* value directly.

Graph()

See "Graph Box(properties, script)".

Graph 3D Box(*properties*)

Description

Constructs a display box with 3-D content.

Returns

The display box.

Arguments

properties Properties can include: Frame Size(x, y), Xname("title"), Yname("title"), Zname("title").

Notes

This display box constructor is experimental.

Graph Box(*properties*, *script*)

Graph(*properties*, *script*)

Description

Constructs a graph with axes.

Returns

The display box (Graph Box).

Arguments

properties Named property arguments: Title("title"), XScale(low, high), YScale(low, high), FrameSize(h, v), XName("x"), YName("y"), SuppressAxes.

script Any script to be run on the graph box.

H Center Box(<*child box*>)

Returns a display box that contains the optional child display box argument. The box is centered in the horizontal space defined by the maximum size of that child display box and all of the other siblings of the center box.

H List Box(<Align("center"|"bottom")>, display box, ...)

Description

Creates a display box that contains other display boxes and displays them horizontally.

Arguments

`Align("center"|"bottom")` Specify center or bottom alignment of the contents in the list box. The contents are bottom aligned by default.

display box Any number of display box arguments can be contained in the list box.

H Scroll Box(<Size(*h*)>, *display box*)

Description

Returns a display box that positions a larger child box using a horizontal scroll bar.

Arguments

`Size(`*h*`)` The horizontal length of the scroll bar.

Notes

The flexible argument is deprecated. Use `Set Stretch` instead. See "V Scroll Box(<Size(v)>, display box)" for an example.

H Sheet Box(<<Hold(*report*), *display boxes*)

Description

Returns a display box that arranges the display boxes provided by the arguments in a horizontal layout. The `<<Hold()` message tells the sheet to own the report(s) that is excerpted.

H Splitter Box(<Size(*h*, *v*)>, *display box*, ...)

Description

Returns a display box that arranges the display boxes provided by the arguments in a horizontal layout (or panel). The splitter enables the user to interactively resize the panel.

Required Argument

display box Any number of display box arguments can be contained in the splitter box.

Optional Argument

`Size(h, v)` Specifies the size of the splitter box in pixels. This size is for the outer splitter box. Inner display boxes are proportionately sized according to the width and height of the outer splitter box.

Optional Messages

`<<Size(n)` Specifies the proportions of the last panel. `<<Size(.25)` resizes the last panel to 25% the splitter box height (or width, for vertical splitter boxes).

`<<Set Sizes({n, n})` Specifies the proportions of each panel. `db<<Set Sizes({.75, .25})` sizes the first panel to 75% and the second panel to 25% of the splitter box height (or width, for vertical splitter boxes).

`<<Close Panel(n, <Boolean>)` Closes the panel that you specify. `<<Close Panel(2)` closes the second panel. With three or more panels, you must include the second Boolean value. That value indicates which panel expands to fill the space left by the closed panel.

- `<<Close Panel(2,0)` closes the second panel; the following sibling takes the extra space.
- `<<Close Panel(2,1)` closes the second panel; the preceding sibling takes the extra space.

`<<Open Panel(n, <Boolean>)` Opens the panel that you specify. With three or more panels, you must include the second Boolean value. Works similar to `<<Close Panel` described above. The panels are initially opened. You use `<<Open Panel` only after using `<<Close Panel`.

`<<Get Sizes()` Returns the proportions of each panel as a list.

Hier Box(text, Hier Box(...), ...)

Description

Constructs a node of a tree (similar to Diagram output) containing text. `Hier Box` can contain additional `Hier Boxes`, allowing you to create a tree. The text can be a quoted string or a `Text Edit Box`.

Hist Seg([*data*], <[*freq column*]>, <[*weight column*]>, <Vertical(*Boolean*)>, <Row States()>

Description

Returns a histogram seg.

Required Argument

data The data in matrix format.

Optional Arguments

freq column The frequency column in matrix format.

weight column The weight column in matrix format.

`Vertical(Boolean)` Displays the histogram vertically by default (or if set to 1). Display the histogram horizontally by setting the value to 0.

`Row States` Specifies a data table reference or row states.

`Icon Box(name)`

Description

Constructs a display box containing an icon, where the argument is a name such as Popup, Locked, Labeled, Sub, Excluded, Hidden, Continuous, Nominal, or Ordinal. The argument can also be a path to an image.

Argument

name A quoted string that is the name of a JMP icon or the path to an icon.

Example

`Icon Box( "Nominal" )` constructs a display box that contains the Nominal icon.

`Icon Box( "$SAMPLE_IMAGES/pi.gif" )` inserts the pi.gif sample image.

Notes

- Some icons are used on both Windows and macOS. Other icons are platform specific.
- Consider installing the Built-In JMP Icons add-in. The add-in lets you view icons interactively and see what they look like in different contexts. Download the add-in from https://community.jmp.com/t5/JMP-Add-Ins/Built-In-JMP-Icons/ta-p/42251.

`If Box(Boolean, display boxes)`

Description

Constructs a display box whose contents are conditionally displayed.

Arguments

Boolean 1 displays the display boxes inside the `If Box`. 0 does not display them.

display boxes Any display box tree.

`If Seg(<State(Boolean)>)`

Description

Returns a display seg that shows or hides display seg children.

Arguments

`State(Boolean)` Determines whether the display seg children are shown (1) or hidden (0).

Example

```
lines = [30 20 80 70, 10 90 90 10, 40 20 60 30];
win = New Window( "Lines Seg Example",
   g = Graph Box( If Seg( true, <<Append( Lines Seg( lines ) ) ) )
```

```
);
```

`Journal Box(string)`

Description

Constructs a display box that displays the quoted string. We recommend that you do not generate the journal text by hand.

`Line Seg(x, y, <Row States(dt|dt, [rows]|dt, {{rows}, ...}|{row states})>, <Sizes(s)>)`

Description

Creates a display seg of connected line segments for the given *x* and *y* values. The optional third argument enables row state assignments from either a data table or independently.

`Lines Seg([x1 y1 x2 y2, ...])`

Description

Returns a display seg with a sequence of line segments for the given *x* and *y* values.

`Lineup Box(<nCol(n)>, <Spacing(pixels, <vspace>), display boxes, ...)`

Description

Constructs a display box to show an alignment of boxes in *n* columns.

`List Box({item, ...}, <Width(pixels)>, <maxSelected(n)>, <nLines(n)>, <script>)`

Description

Creates a display box to show a list box of selection items (quoted strings). The argument can be a list of two-item lists containing the item name and a quoted string that specifies the modeling type or sorting order. Item names are case sensitive by default. The icon appears next to the corresponding item in the list box.

`Marker Seg(x, y, <Row States(dt|dt, [rows]|dt, {{rows}, ...}|{row states})>, <Sizes(s)>)`

Description

Creates a display seg with markers for all of the x and y values. The optional third argument enables row state assignments from either a data table or independently.

Matrix Box(*x*)

Matrix Box(*matrix*, < <<Column Names(*col1*, *col2*, ...)>, < <<Row Names(*row1*, *row2*, ...)>>)

Description

Displays the *matrix* given in the usual array form. Column and row names are quoted strings.

Mouse Box(*display box arguments*)

Description

Returns a box that can make JSL callbacks for dragging and dropping, marking, or clicking and tracking mouse actions.

Arguments

display box arguments Specifies the object that the user interacts with, such as a Text Box or Button Box. See the Scripting Index in the Help menu.

Move to Project(<Source(*project*)|Destination(*project*)>, <Windows({*list of windows to move*})>)

Description

Moves one or more windows into a project or out of a project, or between projects.

Arguments

Source(*project*) The project containing the windows that you want to move.

Destination(*project*) The project to which you want to move the windows.

Windows({*list of windows to move*}) A list of windows to move to the project. If omitted, all windows will be moved. Note that the data table and all of its dependent reports will be moved. However, you need to specify only the data table name or report name in the windows argument to move it.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
report = dt << Run Script( "Bivariate" );
project = New Project();
// move the report and data table to a new project
Move to Project( Destination( project ), Windows( {report} ) );
```

New Image()

New Image(*width*, *height*)

New Image(*filepath*)

New Image(Open(*url*))

New Image(*picture*)

New Image(*matrix*)

New Image(rgb|r|rgba, {*matrix*, ...})

Description

Creates a new image which can then be edited using JSL. Valid formats are png, bmp, jpeg, jpg, tiff, tif, and gif.

Returns

An image.

Arguments

All argument sets are optional, but all arguments within each set are required.

width, height Sets the width and height of the image in pixels.

filepath The quoted filepath to an image.

Open(*url*) Opens the image at the specified URL path.

picture A JSL picture object.

matrix The image as a matrix of JSL color pixels.

rgb|r|rgba, {*matrix*, ...} Specifies the channels ("rgb", "r", or "rgba") and provides a matrix of values (0.0-1.0) for each channel. Examples:

```
New Image( "r", [r matrix] );
New Image( "rgb", {[r matrix], [g matrix], [b matrix]} );
New Image( "rgba", {[r matrix], [g matrix], [b matrix], [a matrix]} );
```

New Project(*arguments*)

Description

Creates a project using the specified script.

Arguments

<<Add Bookmarks({<File(*path*)>, <Folder(*path*, Expanded(*Boolean*)>)>, <Group(*name*, <Expanded(*Boolean*)>, {*contents*}>) Creates bookmarks for frequently used files in the project. The argument is a list of bookmark items, each of which is specified using File(), Folder(), or Group(). Group() accepts File(), Folder(), and Group() as children.

<<Reset Layout Sets the project to use the default layout.

<<Run Script Specifies the data tables and reports that appear in the project.

<<Save(*<path>*) Saves the project. Include a quoted path and file name to save the project to a specific location. Save As is an alias.

<<Set Bookmarks({<File(*path*)>, <Folder(*path*, Expanded(*Boolean*)>)>, <Group(*name*, <Expanded(*Boolean*)>, {*contents*}>) Sets the bookmarks for the project. The argument is a list of bookmark items, each of which is specified using File(), Folder(), or Group(). Group() accepts File(), Folder(), and Group() as children.

<<Set Layout Sets the window layout of the project.

<<Show Bookmarks Shows or hides the bookmarks.

<<Show Log Shows or hides the log.

<<Show Window List Shows or hides the Window List.

Example

The following example creates a project from BigClass.jmp and two reports.

```
project = New Project();
project << Run Script(
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   dt << Run Script( "Bivariate" );
   dt << Run Script( "Distribution" );
);
```

New Window(*title*, *<arguments>*, *display box*)

Description

Makes a new window with the required quoted title and a display box tree.

Optional Arguments

<<Script(*<script>*) Creates a new script window. The optional quoted string *script* is placed inside the script window.

<<Journal Creates an empty journal.

<<Size Window(*x*, *y*) Creates a new window of the specified height and width.

<<Modal Makes the new window a modal window, which prevents any other actions in JMP until the window is closed. If you do not include an **OK** or **Cancel** button, one is added automatically for you. **Note:** If used, this argument must be the second argument, directly after the window title. Available modal window arguments are:

– <<On Open(*expr*) runs expr when the window is created.

Note: In data tables, On Open (or OnOpen) scripts that execute other programs are never run. Set the Evaluate OnOpen Scripts preference to control when the script is run.

– <<On Close(*expr*) runs expr when the window is closed. Returns 0 if the window fails to close.

- – `<<On Validate(expr)` runs `expr` when the **OK** button is pressed. If it returns True, the window is closed otherwise the window remains open.
- – `<<Return Result` changes the window's return value when it closes to match that of the deprecated `Dialog()` function.

`Show Toolbars(Boolean)` Shows or hides the toolbar. The default value is 1. (Windows only.)

`Show Menu(Boolean)` Shows or hides the menu bar. The default value is 1. (Windows only.)

`Suppress AutoHide(Boolean)` Suppresses or uses the auto-hide feature for menus and toolbars. The default value is 1. (Windows only).

Notes

`Dialog()` is deprecated. Use `New Window()` with the `Modal` argument instead.

`Number Col Box(title, numbers)`

Description

Creates a column named *title* (a quoted string) with numeric entries given in list or matrix form.

`Number Col Edit Box(title, {numbers}|[numbers])`

Description

Creates a column named *title* (a quoted string) with numeric entries given in list or matrix form. The numbers can be edited.

`Number Edit Box(initial value, <width>)`

Description

Creates an editable number box that initially contains the *initial value* argument.

Returns

The display box object.

Argument

initial value Any number to use as the initial value. If you use a date or time format, a date and time selector window is created.

<width> Sets the width of the box in characters.

`Outline Box(title, display box, ...)`

Description

Creates a new outline named *title* (a quoted string) containing the listed display boxes.

Page Break Box()

Description

Creates a display box that forces a page break when the window is printed.

Panel Box(*title*, *display box*)

Description

Creates a display box labeled with the quoted string *title* that contains the listed display boxes.

Picture Box(Open(*picture*), *format*)

Description

Creates a display box that contains a graphics picture object.

Returns

A reference to the display box.

Argument

Open(*picture*) Opens the directory that contains the picture.

format Specifies the graphic file format. Specifying the format opens the picture in JMP. If you omit this argument, the picture opens in the default graphics program. Valid formats are the quoted strings "png", "bmp", "jpeg", "jpg", "tiff", "tif", and "gif".

Example

```
New Window( "Example",
   Picture Box( Open( "$SAMPLE_IMAGES/pi.gif", gif ) ) );
```

Platform(*data table*, *script*)

Description

Evaluates the specified script in the context of the specified data table.

Returns

The resulting display box for embedding in a display tree.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Platform example",
   H List Box(
      Platform(
         dt,
         Bubble Plot(
            X( :weight ),
            Y( :height ),
```

```
            Sizes( :age ),
            Title Position( 0, 0 )
         )
      ),
      Platform(
         dt,
         Bubble Plot(
            X( :weight ),
            Y( :age ),
            Sizes( :height ),
            Title Position( 0, 0 )
         )
      )
   )
);
```

Plot Col Box(*title*, *numbers*)

Description

Returns a display box labeled with the quoted string *title* to graph the *numbers*. The numbers can be either a list or a matrix.

Poly Seg(*x values*, *y values*)

Description

Returns a display seg that represents a polygon with vertices based on the *x values* and *y values*.

Example

```
x = [10, 50, 90];
y = [10, 90, 10];
win = New Window( "Poly Seg Example",
g = Graph Box( Poly Seg( x, y ) ) );
frame = g[FrameBox( 1 )];
seg = (frame << Find Seg( "Poly Seg" ));
```

Popup Box({*command1*, *script1*, *command2*, *script2*, ...})

Description

Creates a red triangle menu. The single argument is an expression yielding a list of an even number of items alternating between the quoted command string and the expression that you want evaluated when the command is selected. If the command is an empty string, a separator line is inserted.

Notes

Pressing ALT and right-clicking the red triangle menu opens a window with check boxes for the commands.

`Radio Box({item, ...}, <script>)`

Description

Constructs a display box to show a set of radio buttons. The items are quoted strings. The optional script is run every time a radio button is selected.

`Range Slider Box(minValue, maxValue, lowVariable, highVariable, script)`

Description

`Range Slider Box()` returns a display box that shows a range slider control that ranges from *minValue* to *maxValue*. As the two sliders' positions change, their values are placed into *lowVariable* and *highVariable*, and the script is run.

Returns

The display box (Range Slider Box).

Arguments

minValue, *maxValue* Numbers that set the minimum and maximum value the slider represents.

lowVariable The variable whose value is set and changed by the lower slider.

highVariable The variable whose value is set and changed by the upper slider.

script Any valid JSL commands that are run as the slider is moved.

`Report(obj)`

Description

Returns the display tree of a platform *obj*. This can also be sent as a message to a platform: `obj<<Report.`

`Scene Box(x size, y size)`

Description

Creates an *x*-sized by *y*-sized scene box for 3-D graphics.

`Scene Display List`

Description

Returns a display list for 3-D graphics.

Example

```
ex = Scene Display List();
```

```
ex << Color( .9, .9, .9 );
ex << Text( center, middle, .3, "Hello World" );
exScene = Scene Box( 600, 600 );
exScene << Background Color( 0 );
exScene << Show Arcball( always );
New Window( "See HelloWorld.jsl in sample scripts", exScene );
exScene << Perspective( 45, .2, 20 );
exScene << Translate( 0.0, 0.0, -4.5 );
exScene << Arcball( ex, 1.5 );
exScene << Update;
```

Script Box(<*script*>, <*language*>, <*width*>, <*height*>)

Description

Constructs an editable box that contains the quoted string *script*. The editable box is a script window and can both be edited and run as JSL.

Optional Arguments

script A quoted string that appears in the script box.

language A quoted string that provides syntax highlighting for the specified language. Valid languages are "JSL", "Text", "SAS", "SAS Output", "SASLog", "R", "MATLAB", "JavaScript", "C", "SQL", "Python", "JSON", and "XML".

width An integer that sets the width of the script box.

height An integer that sets the height of the script box.

Example

```
// JSON
New Window( "JSON",
   Script Box(
      "{\!"a\!":1,\!"b\!":\!"test\!"}",
      "JSON"
   )
);
```

Scroll Box(<Size(*h*,*v*)>, *display box*, ...)

Description

Creates a display box that positions a larger child box using scroll bars.

Returns

A reference to the scroll box object.

Required Argument

display box Any number of display box arguments can be contained in the scroll box.

Optional Argument

Size(*h*,*v*) The *h* and *v* arguments specify the size of the box in pixels.

Notes

You can send a scroll box object a message to set the background color:

```
<<Set Background Color( {R, G, B} | <color> )
```

The `Flexible` argument is deprecated. Use the `Set Stretch` message instead. See "V Scroll Box(<Size(v)>, `display box`)" for an example.

You can set the Boolean flags for horizontal (h) and vertical (v) scrolling to enable (1) or disable (0) the scroll bars. If scrolling is disabled in a given direction, the Scroll Box will behave as a regular container in that direction.

```
<<Set Scrollers (h, v)
```

To return the flags for scrolling, use the following message:

```
<<Get Scrollers
```

To set the horizontal (h) and vertical (v) positions (in pixels) for the scrollers on the scroll bar:

```
<<Set Scroll Position (h,v)
```

To return the flags for scroll position, use the following message:

```
<<Get Scroll Position
```

To return the maximum positions for horizontal and vertical scrolling, use the following message:

```
<<Get Scroll Extents
```

Example

The following example shows a window containing a scroll box with the specified settings.

```
win = New Window( "Example",
   sb = Scroll Box(
      Size( 150, 75 ),
      List Box(
         {"First Item", "Second Item",
         "Third Item", "Fourth Item",
         "Fifth Item"},
         width( 200 ),
         max selected( 2 ),
         nLines( 6 )
      )
   )
);
win << Set Window Size( 300, 200 );
sb << Set Scrollers( 1, 1 ); // enable both scroll bars
// position the scrollers on the scroll bar
sb << Set Scroll Position( 0, 20 );
```

Shape Seg({Path(*path*), ...}, <Row States(*dt*|*dt*,[*rows*]|dt,{{*rows*}, ...}|{*row states*})>)

Description

Returns a display seg with a collection of shapes.

Required Argument

`Path` Specifies the path with an Nx3 matrix or with a text representation. A path matrix has three columns for x, y, and flags for each point in the path. The flag values are 0 for control, 1 for move, 2 for a line segment, 3 for a cubic Bézier segment, and are negative if the point also closes the path. Path text supports SVG syntax.

Optional Arguments

`Row States` Specifies a data table reference and optionally rows or the actual row state.

Example

```
win = New Window( "Shape Seg Example",
   Graph Box(
      Shape Seg(
         {Path( [10 10 1, 10 70 0, 70 70 0, 70 10 -3] ),
         Path( "M20,20 C20,60 60,60 60,20 Z" )},
         Row States( {Selected State( 1 ), Color State( "red" )} )
      )
   )
);
```

Sheet Box(<<Hold(*rpt*), *display box*, ...)

Description

Returns a display box that can organize other display boxes vertically or horizontally.

Sheet Panel Box(*title*, *display box*)

Description

Specifies whether the Sheet Box should be horizontal or vertical.

Slider Box(*minValue*, *maxValue*, *variable*, *script*, <Set Width(*n*)>, <Rescale Slider(*min*, *max*)>)

Description

Creates an interactive slider control.

Returns

The display box (Slider Box).

Required Arguments

minValue, *maxValue* Numbers that set the minimum and maximum value the slider represents.

variable The variable whose value is set and changed by the slider box.

script Valid JSL commands that are run as the slider box is moved.

Optional Arguments

Set Width(*n*) Specifies the width of the slider box in pixels.

Rescale Slider(*minValue*, *maxValue*) Resets the maximum and minimum values for the slider box.

Notes

You can send Set Width and Rescale Slider as messages to a slider object. For example:

```
ex = .6;
New Window( "Example", mybox = Slider Box( 0, 1, ex, Show( ex ) ) );
mybox << Set Width( 200 ) << Rescale Slider( 0, 5 );
```

Spacer Box(<Size(*h*, *v*)>, <Color(*color*)>)

Description

Creates a display box that can be used to maintain space between other display boxes, or to fill a cell in a LineUp Box.

Returns

A reference to the display box.

Optional Arguments

Size(*h*, *v*) The *h* and *v* arguments specify the size of the box in pixels.

Color(*color*) Sets the color of the box to the JSL color argument.

Spin Box(*script*)

Description

Returns a display box that shows a button with up and down controls.

Argument

script Invoked with an argument that indicates the direction of the arrow clicked. Negative is down, and positive is up. A value of 1 indicates a single click, while larger values indicate a repeating action.

Example

```
win = New Window( "Example",
   Lineup Box(
      2,
      nb = Number Edit Box( 3 ),
      sb = Spin Box( Function( {value}, nb << Increment( value ) ) )
   )
);
nb << Set Increment( 1 );
```

Splitter Box(<Size(*x*, *y*)>, *display box*, ...)

Description

Returns a display box that can organize other display boxes horizontally or vertically with interactive control of sizes. Child sizes are specified as a proportion of the width or height of the splitter box. The optional *Size* argument is used only for the top-most splitter box. Lower level display boxes are sized like any other child box.

Use `H Splitter Box()` or `V Splitter Box()`.

String Col Box(*title*, {*string*, ...})

Description

Creates a column in the data table containing the quoted *string* items. The column is named after the quoted *title*.

String Col Edit Box(*title*, {*string*, ...})

Description

Creates a column in the data table containing the quoted *string* items. The string boxes are editable. The column is named after the quoted *title*.

Notes

To retrieve the data, use the following message:

```
data = obj << Get;
```

Tab Box(Tab Page Box(Title(*page title 1*), <*options*>, *contents of page 1*), Tab Page Box(Title(*page title 2*), <*options*>, *contents of page 2*), ...);

Description

(Previously called `Tab List Box`.) Creates a tabbed window pane. The arguments are an even number of items alternating between the name of a tab page and the contents of the tab page. The page titles must be quoted.

Arguments

`Tab Page Box` Returns a display box that can be used in a tab box or as a stand-alone container with a title.

- `Title(page title #)` specifies the title of page 1.
- Options include <<`Closeable`(*Boolean*) (specifies whether the box can be closed), <<`Tip`(*string*) (a quoted string that specifies a tooltip), and <<`Icon`(*string*) (a quoted string that specifies the icon).
- *contents of page #* is a quoted string that specifies the text on the tab.

Example

```
New Window( "Example",
```

```
Tab Box(
   t1 = Tab Page Box( Title( "alpha" ), Panel Box( "panel", Text Box( "text"
) ) ),
   t2 = Tab Page Box( Title( "beta" ), Popup Box( {"x", ex = 1, "y", ex = 2}
) ),
)
);
```

Notes

Certain messages that you can send to `Tab Page Box` have been renamed:

- `Set Title` is called `Title`.
- `Set Tip` is called `Tip`.
- `Set Icon` is called `Icon`.
- `Set Closeable` is called `Closeable`.

Tab List Box(*title*, *tabExpr1*, ...)

See "Tab Box(Tab Page Box(Title(page title 1), <options>, contents of page 1), Tab Page Box(Title(page title 2), <options>, contents of page 2), ...);".

Tab Page Box(Title(*string*), <*options*>, *contents*)

Description

Returns a display box that can be used in a tab box or as a stand-alone container with a title.

Required Argument

`Title` A quoted string that specifies the title on the tab.

Optional Messages

`<<Tip(`*string*`)` A quoted string that specifies a tooltip.

`<<Closeable(`*Boolean*`)` Specifies whether the page can be closed.

`<<Icon(`*string*`)` A quoted string that specifies the icon.

`<<Set Font(font name, <size>, <"bold"|"italic"|"underline"|strikeout">, <angle>` Specifies the font properties.

`<<Set Font Name("`*font name*`")` Specifies the name of the font.

`<<Set Font Scale(f)` Specifies the scale factor for the font. The scale factor is applied to the size that is determined by the base font and point size.

`<<Set Font Size(n)` Specifies the font size in pixels.

`<<Set Font Style("plain"|"bold")` Specifies the font style.

Notes

Certain messages that you can send to `Tab Page Box` have been renamed:

- `Set Title` is called `Title`.

- `Set Tip` is called `Tip`.
- `Set Icon` is called `Icon`.
- `Set Closeable` is called `Closeable`.

`Table Box(`*`display box, ...`*`)`

Description

Creates a report table with the *`display boxes`* listed as columns.

`Text Box(`*`text, <arguments>`*`)`

Description

Constructs a box that contains the quoted string *`text`*.

Arguments

`<<Justify Text(`*`position`*`)` Justifies the text left, center, or right as specified in quotes.

`<<Set Wrap(`*`pixels`*`)` Sets the point at which text wraps.

`Text Edit Box(`*`text, <arguments>`*`)`

Description

Constructs an editable box that contains the quoted string *`text`*.

Arguments

`<<Password Style(`*`Boolean`*`)` Displays asterisks in the box rather than the password.

`<<Set Script` Runs the specified script after the text is edited.

`<<Set Width(`*`pixels`*`)` Sets the point at which text wraps.

`This Project()`

Description

Gets the current project when a JSL script is run from that project.

Example

The following example gets the window title of the current project.

```
project = New Project();
project << Save( "$DOCUMENTS/Test Project.jmpprj" );
project << Run Script(
   New Window( "Project Title",
      Text Box(This Project() << Get Window Title())
   );
);
```

Tree Box(<{*rootnodes*}>, <Size(*width*, *height*)>, <MultiSelect(*Boolean*)>)

Description

Constructs a box to show a hierarchical tree composed of tree nodes.

Arguments

{*rootnodes*} Specifies the names for the root nodes created by `Tree Node()` which the box contains.

`Size(`*width*, *height*`)` Specifies the width and height (in pixels) of the box.

`MultiSelect(`*Boolean*`)` Indicates that more than one item in the tree can be selected.

Tree Node(<*data*>)

Description

Creates a node for display in a Tree Box display. Tree Node is used for both parent and child nodes.

Notes

If you send a root node that contains one or more nodes with the `Set Node Select Script` defining a collapse message, then macOS runs the script twice. Windows doesn't run the script. This behavior on macOS doesn't just affect increments. Any script runs twice. It will print to the log twice, create a column twice, try to delete something twice, and so on.

Triangulation(<*dt*>, X(*col1*, *col1*), <Y(*col*)>)

Description

Returns an object containing the Delaunay triangulation of the given point set. The optional Y will be averaged for duplicate points, and all points in the output will be unique.

Examples

```
tri = Triangulation(
  X( [0 0 1 1], [0 1 0 1] ),
  Y( [0 1 2 3] )
);
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
tri = Triangulation( X( :X, :Y ), Y( :POP ) );
```

V Center Box(*display box...*)

Returns a display box that contains the child display box argument. The box is centered in the vertical space defined by the maximum size of that child display box and all of the other siblings of the center box.

V List Box(<Align("Center"|"Right")>, *display box, ...*)

Description

Creates a display box that contains other display boxes and displays them vertically.

Arguments

`Align` Specify right or center alignment of the contents in the list box. The contents are center aligned by default.

`display box` Any number of display box arguments can be contained in the list box.

V Scroll Box(<Size(*v*)>, *display box*)

Description

Returns a display box that places a scroll bar on the bottom and right if the contents are bigger than the size of the scroll box.

Arguments

`Size(`*`v`*`)` The vertical length of the scroll bar.

`display box` Any number of display box arguments can be contained in the scroll box.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Window( "stretchable",
   H Splitter Box(
      Size( 400, 200 ),
      Scroll Box(
         Size( 200, 200 ),
         dt <<Run Script( "Distribution" ),
         <<Set Stretch( "Window", "Window" )
      ),
      Scroll Box(
         Size( 200, 200 ),
         dt <<Run Script( "Bivariate" ),
         <<Set Stretch( "Window", "Window" )
      ),
      <<Set Stretch( "Window", "Window" )
   )
);
```

Notes

The flexible argument is deprecated. Use `Set Stretch` instead.

V Sheet Box(<<Hold(*report*), *display box, ...*)

Description

Returns a display box that arranges the display boxes provided by the arguments in a vertical layout. The <<Hold() message tells the sheet to own the report(s) that is excerpted.

V Splitter Box(<Size(*h*,*v*)>, *display box...*, <*arguments*>)

Description

Returns a display box that arranges the display boxes provided by the arguments in a vertical layout (or panel). The splitter enables the user to interactively resize the panel.

Arguments

Size(*v*) The vertical length of the scroll bar.

display box Any number of display box arguments can be contained in the splitter box.

Optional Arguments

For more information about the optional arguments, see "H Splitter Box(<Size(h, v)>, display box, ...)".

Web Browser Box(*url*)

Description

Creates a display box that contains a web page. Available only on Windows in Internet Explorer.

Returns

A reference to the web browser box object.

Argument

url A quoted string containing the URL to the web page to display.

Example

The following example creates a splitter box with the web browser box on the left and the bubble plot on the right.

```
dt = Open( "$SAMPLE_DATA/PopAgeGroup.jmp" );
New Window( "Example",
  H Splitter Box(
     Size( 800, 300 ),
     wb = Web Browser Box( "http://www.jmp.com" ),
     dt << Run Script( "Bubble Plot Region" )
  )
);
wb << Set Stretch( "Window", "Window" ); // auto stretch horizontally and
  vertically
wb << Set Max Size( 10000, 10000 ); // maximum size in pixels
```

Notes

The <a href> target "_blank" opens the web page in a new Internet Explorer window. The <a href> target "_new" opens the web page in the active Internet Explorer tab.

`Window(<string|int>)`

Returns

Either a list of references to all open windows, or a reference to an explicitly named window.

Arguments

string A quoted string containing the name of a specific open window.

int The number of a specific open window.

Notes

If no argument is provided, a list of all open windows is returned.

If the argument (either a window name or number) does not exist, an empty list is returned.

`Wrap List Box(display box, ...)`

Description

Creates a list box that contains other display boxes and displays them horizontally, but wraps them when printing.

Arguments

display box Any number of display box arguments can be contained in the list box.

Expression Functions

`Arg(expr, i)`

`Arg Expr(expr, i)`

Description

Finds the argument numbered by *i* within the given expression.

Returns

The *i*th argument within the expression *expr*.

`Empty()` if that argument does not exist or is not specified.

Arguments

expr An expression.

i An integer that indicates which argument to return.

Notes

`Arg Expr()` is deprecated. Use `Arg()` instead.

`Eval Expr(`*expr*`)`

Description

Evaluates any expressions within *expr*, but leaves the outer expression unevaluated.

Returns

An expression with all the expressions inside *expr* evaluated.

Argument

expr A valid expression.

`Expr(`*expr*`, `*i*`)`

Description

Returns the argument unevaluated (expression-quoting).

Returns

The argument, unevaluated.

`Empty()` if that argument does not exist or is not specified.

Argument

expr An expression.

i An integer that indicates which argument to return.

`Extract Expr(`*expr*`, `*pattern*`)`

Description

Find *expr* matching *pattern*.

Returns

A pattern that matches the specified *pattern*.

Arguments

expr An expression.

pattern Any pattern.

`Head(`*exprArg*`)`

`Head Expr(`*exprArg*`)`

Description

Returns the head of the evaluated expression, without its arguments.

Notes

`Head Expr()` is deprecated. Use `Head()` instead.

`Head Name(expr)`

`Head Name Expr(expr)`

Description

Returns the head of the evaluated expression as a quoted string.

Notes

`Head Name Expr()` is deprecated. Use `Head Name()` instead.

`N Arg(exprArg)`

Description

Returns the number of arguments of the evaluated expression head.

`N Arg Expr(exprArg)`

Description

Returns the number of arguments of the expression head.

Notes

`N Arg Expr()` is deprecated. Use `N Arg()` instead.

`Name Expr(x)`

Description

Returns the unevaluated expression of *x* rather than the evaluation of *x*.

File Functions

`Close(<dt|query>, <noSave|Save(path)>)`

Description

Closes a data table, query, or JSON file. If no arguments are specified, the current file is closed. If the file has been changed, it is automatically saved. All dependent windows are also closed (for example, report windows that are based on the data table).

Returns

Void.

Arguments

dt An optional reference to a data table, query, or JSON file.

`nosave|Save(path)` An optional switch to either save the file to the specified quoted path before closing or to close the file without saving it.

`Close All(<Project(`*`title`*`|`*`index`*`|`*`display box`*`|`*`window`*`)>, `*`type`*`, <"Invisible"|"Private">, <NoSave|Save>)`

Description

Closes all open resources of *type*.

Required Argument

type A named argument that defines the type of resources that you want to close. The allowable types are: `Data Tables`, `Reports`, and `Journals`.

Optional Argument

`Project(`*`title`*`|`*`index`*`|`*`display box`*`|`*`window`*`)` Closes the specified project window.

`"Invisible"` Closes all invisible data tables.

`"Private"` Closes all private data tables.

`NoSave or Save` Saves the specified types of windows before closing or to close without saving.

`Close Database Connection(`*`db connection handle`*`)`

Description

Closes a database connection returned from Create Database Connection.

`Close Log(`*`Boolean`*`)`

Description

Closes the log window.

`Convert File Path(`*`path`*`, <"Absolute"|"Relative">, <"POSIX"|"Windows">, <Base(`*`path`*`)>, <`*`Search`*`>)`

Description

Converts a file path according to the arguments.

Returns

The converted path.

Required Argument

path A quoted pathname that can be either Windows or POSIX.

Optional Arguments

`"Absolute"|"Relative"` Specifies whether the returned pathname is in absolute or relative terms. The default value is `Absolute`.

`"POSIX"|"Windows"` Specifies whether the returned pathname is in Windows or POSIX style. The default is `POSIX`.

`Base(`*`path`*`)` Specifies the quoted base pathname, useful if `Relative` is specified. The default is the default directory. See "Set Default Directory(path)".

`Search` Searches through the specified directories for the specified string. In the example below, JMP searches $SAMPLE_DATA/ and $SAMPLE_DATA/Time Series/ for Air.jmp. If Air.jmp is in both directories, the first instance is returned. Without the search option, `Convert File Path()` uses the default directory instead.

Example

```
Set File Search Path(
   {Convert File Path( "$SAMPLE_DATA/" ),
   Convert File Path( "$SAMPLE_DATA/Time Series/" )}
);
Convert File Path( "Air.jmp", Search );
   Get File Search Path() = {"/C:/Program Files/SAS/JMPPRO/16/Samples/Data/",
      "/C:/Program Files/SAS/JMPPRO/16/Samples/Data/Time Series/"};
   Convert File Path("Air.jmp", search) = "/C:/Program
      Files/SAS/JMPPRO/16/Samples/Data/Time Series/Air.jmp";
```

Copy Directory(*from path*, *to path*, <Recursive(*Boolean*)>)

Description

Copies files from one directory to another, optionally copying subdirectories. The directory name is created in the *to path* and should not be part of the *to path* argument.

Returns

Returns 1 if the directory is copied; otherwise, returns 0.

Required Arguments

from path Specifies the directory containing the files to copy to the new directory. *from path* is a quoted string.

to path Specifies the path where the new directory should be created and to which the files are copied. *to path* is a quoted string.

Optional Argument

`Recursive(`*Boolean*`)>` Specifies whether to copy the *from path* subdirectory structure to the *to path*.

Notes

`Copy Directory(`*from path*, *to path*, *Boolean*`)` is deprecated.

Copy File(*from path*, *to path*)

Description

Copies one file to a new file using the same or a different name.

Returns

Returns 1 if the file is copied; otherwise, returns 0.

Arguments

from path Specifies the path and file name to copy to the new file. *from path* is a quoted string.

to path Specifies the path and file name for the new file. *from path* is a quoted string.

`Create Database Connection(`*dataSourceName*`|"Connect Dialog",<DriverPrompt(`*Boolean*`)>);`

Description

Creates a connection to the specified database or prompts the user to provide database log in information.

Returns

A handle to the database connection.

Required Argument

dataSourceName The quoted server connection string that contains information such as the data source name and user name.

Optional Arguments

`"Connect Dialog"` Opens the Select Data Source window, from which the user selects the database.

`Driver Prompt(`*Boolean*`)` Enables the ODBC driver to prompt for the connection information if necessary.

Examples

Specify the data source name, user name, and password:

```
Create Database Connection( "dsn=Books;UID=johnsmith;password=Christmas" );
```

Request that the ODBC driver prompt the user to enter connection information, because the quoted connection string does not specify the password:

```
Create Database Connection( "dsn=Books;UID=johnsmith", Driver Prompt( 1 ) );
```

Enable the user to select the data source, specify `"Connect Dialog"`:

```
Create Database Connection( "Connect Dialog" );
```

`Create Directory(`*path*`)`

Description

Creates a new directory at the specified path location.

Returns

Returns 1 if the directory is created; otherwise, returns 0.

Arguments

path Specifies the quoted path where the new directory should be located.

Creation Date(*path*)

Description

Returns the creation date for the specified file or directory.

Returns

Creation date.

Arguments

path Specifies the quoted directory or path and file name for the query.

Delete Directory(*path*, <Allow Undo(*Boolean*)>)

Description

Deletes the specified directory and its contents and any subdirectories.

Returns

Returns 1 if the directory is deleted; otherwise, returns 0.

Arguments

path Specifies the path and directory for deletion.

Allow Undo(*Boolean*) Allows undo operations, for example, moving to the Recycle Bin or Trash Can.

Delete File(*path*, <Allow Undo(*Boolean*)>)

Description

Deletes the specified file.

Returns

Returns 1 if the file is deleted; otherwise, returns 0.

Required Argument

path Specifies the quoted path and file name for deletion.

Optional Argument

Allow Undo(*Boolean*) Allows undo operations, for example, moving to the Recycle Bin or Trash Can.

Directory Exists(*path*)

Description

Verifies the specified directory exists.

Returns

Returns 1 if the directory exists; otherwise returns 0.

Arguments

path Specifies the quoted path and directory for verification.

File Exists(*path*)

Description

Verifies the specified file name exists at the specified path.

Returns

Returns 1 if the file exists; otherwise returns 0.

Arguments

path Specifies the quoted path and file name for verification.

File Size(*path*)

Description

Determines the size of the file within the specified path.

Arguments

path Specifies the quoted path and file name.

Example

```
File Size( "$SAMPLE_DATA/Big Class.jmp" );
  13142
```

Files In Directory(*path*, <Recursive(*Boolean*)>)

Description

Returns a list of filenames in the *path* given.

Returns

List of filenames. If Recursive(*Boolean*) is not specified, directory names are included in the list.

Required Argument

path A valid quoted pathname.

Optional Argument

Recursive(*Boolean*) A keyword that causes all folders in the path (and all folders that they contain, and so on) to be searched for files.

Notes

Files In Directory(*path*, "Recursive"|*Boolean*) is deprecated.

Find All(<Project(*title*|*index*|*display box*|*window*)>, *type*, <"Invisible"|"Private">)

Description

Finds all open files of the specified type.

Required Argument

type A named argument that defines the type of resources that you want to close. The allowable types are: `Data Tables`, `Reports`, and `Journals`.

Optional Argument

`Project(`*title*`|index|`*display box*`|`*window*`)` Closes the specified project window.

`"Invisible"` Closes all invisible data tables.

`"Private"` Closes all private data tables.

Example

The following example finds all open data tables:

```
exdt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
exdt2 = Open( "$SAMPLE_DATA/Animals.jmp" );
windows = Find All( Data Tables );
For( i = 1, i <= N Items( windows ), i++,
  Write( Char( windows[i] << Get Window Title ) || "\!N" )
);
  Big Class
  Animals
```

Get Default Directory()

Description

Retrieves the user's default directory. This path is used for subsequent relative paths.

If the default directory was set using `Set Default Directory()`, JMP returns the specified path as long as `Get Default Directory()` and `Set Default Directory()` are in the same script.

See "Set Default Directory(path)".

Returns

The absolute pathname as a quoted string.

Arguments

none

Notes

`Get Default Directory()` also gets the path of an active saved scripting window.

Get Excel Worksheets(*absolute path*)

Description

Retrieves a list of worksheets that are in the specified Microsoft Excel workbook. If no worksheets are found, an empty list is returned. The path is a quoted string.

Notes

The function supports .xlsx and Excel 1997 or later workbooks.

Get File Search Path()

Description

Retrieves the current list of directories to search for opening files.

This list is configured using the `Set File Search Path()` function. See "Set File Search Path(path|{list of paths})".

Returns

A list of pathnames as quoted strings.

Get Path Variable(*name*)

Description

Retrieves the value of name, a path variable.

Returns

The absolute pathname as a quoted string.

Argument

`name` A quoted string that contains a path variable. (Examples: SAMPLE_DATA and SAMPLE_SCRIPTS)

Google Sheet Export(*dt*, *email*, *spreadsheet URL|spreadsheet ID|new spreadsheet name*, *sheet name*)

Description

Exports a data table to a Google sheet.

Returns

"1" if the export is successful.

Arguments

`dt` The data table.

`email` The quoted Google email address. @gmail.com is unnecessary.

`spreadsheet URL|ID` The quoted spreadsheet's URL or ID (the string that precedes "spreadsheets/d/").

`new spreadsheet name` The quoted name of the new spreadsheet that you are creating.

`sheet name` The quoted name of the sheet (or tab) within the spreadsheet.

Notes

- JMP features such as formulas and List Check column properties are not supported in Google Sheets.
- If the spreadsheet is empty, look in the JMP log for error messages. On Windows, select **View > Log**. On macOS, select **Window > Log**.

See Also

See *Using JMP* for details about security, country restrictions, and more.

Google Sheet Import(*email*, *spreadsheet URL*|*spreadsheet ID*, <*sheet name*|{*sheet name*, *sheet name*}, *Google Sheet Settings*>)

Description

Imports sheets from a Google Sheet.

Returns

A data table (or the first data table imported if several sheets are imported at once).

Required Arguments

email The quoted Google email address. @gmail.com is unnecessary.

spreadsheet URL|*ID* The quoted spreadsheet's URL or ID (which precedes "spreadsheets/d/").

Optional Arguments

sheet name The quoted name of the sheet or sheets that you want to import.

Google Sheet Settings The settings that describe how the data is imported.

See Also

See *Using JMP* for details about security, country restrictions, and more.

Is Directory(*path*)

Description

Returns 1 if the quoted path argument is a directory and 0 otherwise.

Is Directory Writable(*path*)

Description

Returns 1 if the directory specified in the quoted path is writable and 0 otherwise.

Is File(*path*)

Description

Returns 1 if the quoted path is a file and 0 otherwise.

Is File Writable(*path*)

Description

Returns 1 if the file specified in the quoted path is writable and 0 otherwise.

JSON To Data Table(*JSON string*, (<Private(*Boolean*)>|<Invisible(*Boolean*)>), <Guess(Stack(*Boolean*)|"Tall"|"Wide"|"Huge"|"Pandas")>)

Description

Converts a quoted JSON string to a data table.

Returns

A data table reference. The parsing of an empty value, "" string, missing value, or any other invalid value returns an empty data table.

Required Argument

JSON string The quoted JSON string.

Optional Arguments

Private(*Boolean*) Hides the data table completely. Specify this argument if the user doesn't need to interact with the data table.

Invisible(*Boolean*) Hides the data table from view but shows it in the JMP Home Window.

Guess(Stack(*Boolean*)|"Tall"|"Wide"|"Huge"|"Pandas") Stack applies to nodes that repeat within a parent node that is creating rows. By default, extra values are stored in a single table cell separated by commas. If the value is 1, repeating values are stacked in extra rows. Be careful stacking data. It can cause non-obvious data errors.

"Tall" imports the data in a tall data table. Select this option when the JSON file contains many rows. This option is the default setting.

"Wide" imports the data in a wide data table. Select this option when the JSON file contains many columns.

"Huge" imports the data in a tall and wide data table.

"Pandas" imports data in the pandas format.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt = JSON to Data Table(
   "[ { \!"name\!": \!"KATIE\!", \!"age\!": 12, \!"sex\!": \!"F\!",
   \!"height\!": 59, \!"weight\!": 95 }, { \!"name\!": \!"LOUISE\!",
   \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!": 61, \!"weight\!": 123 }, {
   \!"name\!": \!"JANE\!", \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!":
   55, \!"weight\!": 74 } ]"
);
```

Notes

Stacking the data can cause non-obvious data errors. Values that are supposed to be in the same row might not be. Here's an example of stacking data:

```
d =
"\[
{
       "toys": [{
```

```
                "Wheels": 2,
                "Color": "Red"
        },
        {
                "Wheels": 4,
                "Size": "Large"
        }]
}
]\";

JSON to Data Table( d, Guess( Stack( 1 ) ) );
```

Figure 2.1 Example of Stacking Data that Shows Errors

	Wheels	Color	Size
1	2	Red	Large
2	4		

The first toy should be red and have 2 wheels. The second toy should be large and have 4 wheels.

JSON To List(*JSON string*)

Description

Converts quoted JSON string to a JSL list that represents the structure specified by the JSON data. The parsing of an empty value, "" string, missing value, or any other invalid value returns {}.

Example

```
l = JSON To List(
  "[ { \!"name\!": \!"KATIE\!", \!"age\!": 12, \!"sex\!": \!"F\!",
  \!"height\!": 59, \!"weight\!": 95 }, { \!"name\!": \!"LOUISE\!",
  \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!": 61, \!"weight\!": 123 }, {
  \!"name\!": \!"JANE\!", \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!":
  55, \!"weight\!": 74 } ]"
);
Show( l );
```

JSON Literal(*JSON string*)

Description

Returns a valid JSON Boolean or null constant based on the specification of the parameter. The parsing of an empty value, "" string, missing value, or any other invalid value returns Empty().

Last Modification Date(*path*)

Description

Returns the last modification date of the path.

Returns

Last modification date.

Arguments

path The quoted directory or file name.

Load Text File(*path*, <Charset(character set), Force("Throw"|"Alert"|"Silent")>)>, <Line Separator(character)>, <"XMLParse">|<"SASODSXML">|<"JSON">|<BLOB(<arguments>)>)>)

Description

Reads the text file at *path* into a JSL variable.

Returns

A quoted string.

Required Argument

path A quoted pathname that points to a text file. The path can be a URL.

Optional Arguments

`Charset("charater set")` Specifies the character set. Valid character sets are `"Best Guess"`, `"utf-8"`, `"utf-16"`, `"us-ascii"`," `windows-1252"`, `"x-max-roman"`, `"x-mac-japanese"`, `"shift-jis"`, `"euc-jp"`, `"utf-16be"`, and `"gb2312"`.

`Force("Throw"|"Alert"|"Silent")` Specifies what happens if the character set cannot be detected.

`Line Separator(`*character*`)` Specifies the quoted end-of-line character. For example, `"\!n"` specifies a line feed character. `"\!t"` specifies a tab.

`"XMLParse"` Converts an XML file into JSL.

`"SASODSXML"` Parses the text file as SAS ODS default XML.

`"JSON"` Converts JSON into an expression tree.

`BLOB(<`*arguments*`>)` Returns data from the file as a blob rather than a quoted string. The following optional arguments are for reading parts of the file:

- `ReadOffsetFromBegin(`*n*`)` specifies the zero-based offset to begin reading from the beginning of the file.
- `ReadOffsetFromEnd(`*n*`)` specifies the zero-based offset to begin reading from the end of the file.
- `ReadLength(`*n*`)` specifies the number of bytes to read from the file, either from the beginning of the file or from one of the offset values.

– `Base64Compressed(Boolean)` specifies how the blob is converted to a printable representation. 0, the default and recommended setting, uses JMP's ASCII~HEX representation. 1 means the blob is compressed and converted to base 64 when printed.

`Move Directory(from path, to path)`

Description

Moves a directory and its contents (including subdirectories) from the quoted path to another quoted path.

Returns

Returns 1 if the directory is moved; otherwise returns 0.

Arguments

`from path` Specifies the path and directory for relocation.

`to path` Specifies the destination path and directory.

`Move File(from path, to path)`

Description

Moves a file from the quoted path to another quoted path with the same or different file name.

Returns

Returns 1 if the file is moved. Otherwise, it returns 0.

Arguments

`from path` Specifies the quoted path and file name for relocation.

`to path` Specifies the quoted destination path and file name.

Notes

On Windows, when you move a file to a folder that does not exist, Windows creates the folder and returns 1. On macOS, the folder is not created, and an error is returned.

`Open(path, <arguments>)`

Description

Opens the data table or other JMP file or object created from a file named by the *`path`*. If no path is specified, the Open window appears. Also opens JSON and HDF5 files. See the examples in the JMP Scripting Index in the Help menu for more information about which arguments apply to specific file types.

Required Argument

`path` The quoted path to the file that you want to open.

Optional Arguments

`Add to Recent Files(`*`Boolean`*`)` Determines whether the file is added to the Recent Files list in the Home Window.

`Charset(`*`character set`*`)` The available character set options for importing text files are `"Best Guess"`, `"utf-8"`, `"utf-16"`, `"us-ascii"`,`" windows-1252"`, `"x-max-roman"`, `"x-mac-japanese"`, `"shift-jis"`, `"euc-jp"`, `"utf-16be"`, and `"gb2312"`.

`Column Names Start(`*`n`*`)|Column Names are on Line(`*`n`*`)` Specifies the line number that column names start in the imported text file. If the text file uses returns between cells, column names could be on multiple lines.

`Columns(`*`colName`*`=`*`colType`*`(`*`colWidth`*`),...)` Specifies the columns by name in the text file to import into a data table where:

– `colName`: Specifies the column name used in the imported text file.

– `colType("Character"|"Numeric")`: Specifies whether the specified column contains character or numeric data.

– `colWidth(`*`n`*`)`: Specifies the integer width of the specified column.

`Columns(<`*`arguments`*`>)` For ESRI shapefiles (.shp), this argument and its settings indicate the following:

– `Shape=numeric(`*`n`*`)`: Specifies the column number in the imported ESRI shapefile that contains the shape number.

– `Part=numeric(`*`n`*`)`: Specifies the column number in the imported ESRI shapefile that contains the part number.

– `X=numeric(`*`n`*`)`: Specifies the column number in the imported ESRI shapefile that contains the decimal degree for the longitude (range of ±180°).

– `Y=numeric(`*`n`*`)`: Specifies the column number in the imported ESRI shapefile that contains the decimal degree for the latitude (range of ±90°).

`"Column Names Only"` Returns a list of column names.

`Compress Allow List Check(`*`Boolean`*`)` Specifies that JMP can compress data table created from the imported text file.

`Compress Character Columns(`*`Boolean`*`)` Specifies that JMP should compress data table columns that contain character data from the imported text file.

`Compress Numeric Columns(`*`Boolean`*`)` Specifies that JMP should compress data table columns that contain numeric data from the imported text file.

`Concatenate Worksheets(`*`Boolean`*`)` Specifies that JMP should combine the imported Excel worksheets into one data table.

`Create Concatenation Column(`*`Boolean`*`)` Specifies that JMP should combine columns from an imported Excel file into one column.

`Data Starts(`*`n`*`)|Data Starts on Line(`*`n`*`)` Specifies the line number where data starts in the imported text file.

`Debug JSL(`*`Boolean`*`)` Opens the specified JSL script in the Debugger instead of opening it.

`End Of Field("Tab"|"Space"|"Comma"|"Semicolon"|"Other"|"None")` Specifies the quoted character used to delimit the end of a field in the imported text file. To specify multiple characters, separate each character designation by a comma. If you use `"Other"`, designate the delimiter with `EOF Other()` argument.

`End Of Line("CRLF"|"CR"|"LF"|"Semicolon"|"Other")` Specifies the quoted character used to delimit the end of a line in the imported text file. To specify multiple characters, separate each character designation by a comma. If you use `"Other"`, designate the delimiter with `EOL Other()` argument.

`EOF Other(`*`char`*`)` If the imported text file uses an end-of-field character other than the one specified by `End of Field()`, this argument specifies the character used.

`EOL Other(`*`char`*`)` If the imported text file uses an end of line character other than the one specified by `End of Line()`, this argument specifies the character used.

`"Excel Wizard"` Opens Microsoft Excel worksheets in the Excel Import Wizard. If you omit this argument, the worksheets open directly as a data table.

`File Type(`*`type`*`)` An optional quoted string that specifies the type of file that you are opening (for example, "text", "journal", "sas", "script", "png", and "jmp"). This can be useful if your file does not have a file extension, the file extension of the file does not match the contents of the file, or you want to import a JSL BLOB. If you do not specify this string, the file opens in the default program for the file extension.

Note: The path argument should be used for a zip archive. The extension (.zip) is not required. See "Zip Archives" for the messages that you can send to a zip archive. The basic functionality is to get a list of files in the zip archive, to read a file in the zip archive into either a quoted string or a blob, and to write files into the zip archive. Note that reading a zip archive temporarily puts the contents into memory. Reading very large zip archives can cause errors.

`"Force Refresh"` Closes the specified JMP (.jrn, .jsl, .jrp, or .jmpappsource) file without saving and tries to reopen the file from disk. This argument deletes any changes made since the last time the file was opened.

`HTML Table(`*`n`*`, <ColumnNames(`*`n`*`)>, <DataStarts(`*`n`*`)>)` To import a table from an HTML web page, use the URL as the filepath. The optional *`n`* argument specifies which table number, *`n`*, on the web page to open. If you omit the value, only the first table on the page is imported. The optional `ColumnNames(`*`n`*`)` specifies the row that contains column names. The optional `DataStarts(`*`n`*`)` specifies the row on which the data begins.

Tip: If the table you are importing contains images, they are first imported as text. To load the images in your JMP data table, run the automatically generated table script named Load Pictures. A new expression column containing the images is created. See *Using JMP* for more information about expression columns.

`Ignore Columns(col1, ...)` Specifies the quoted column names in the JMP data table or other JMP file that should not be included in the data table.

`"Invisible"` Opens the file as invisible. This quoted keyword applies to these files: data table, JMP file, external, text, Excel, SAS, ESRI shapefile, and HTML. The data table appears only in the JMP Home Window and the Window menu.

`Labels(Boolean)` Specifies the imported text file contains labels or column headers as the first line in the file. The default value is `True`.

`Lines to Read(n)` Specifies the number of lines in the text file to import. JMP starts counting lines after column names are read.

`Number of Columns(n)` Specifies the number of columns contained in the imported text file.

`Run JSL(Boolean)` Runs the specified JSL file instead of opening it. Include a Boolean argument or an expression that contains a Boolean value. If the script begins with `//!`, which automatically runs the script, include the Boolean value `(0)` to open the script instead.

`Password(password)` Specifies the quoted password for a password-protected SAS file to avoid entering it manually. The password is not encrypted. (Password-protected Microsoft Excel files cannot be imported.)

`"PDF Wizard"` Opens a PDF file in the PDF Import Wizard, where you can control how data is imported.

`"Private"` Opens the table as invisible and without showing it in the JMP Home Window or Window menu. For example, you might create a private data table to hold temporary data that the user does not need to see. This quoted keyword applies to the following files: data table, JMP file, external, text, Excel, SAS, ESRI shapefile, or HTML. Creating a private data table speeds the process of getting to the data; it does not save the computer from allocating the memory necessary to hold the data table data.

`Quarantine Action("Allow Scripts"|"Block Scripts"|"Do Not Open"|"Show Dialog")` Specifies whether scripts run when you open downloaded data tables. Also provides an option to display a window that prompts users to examine or open the data table. If they click Examine, the scripts are disabled.

`Scan Whole File(Boolean)` Specifies how long JMP scans the text file to determine data types for the columns. This is a Boolean value. The default value is true; the entire file is scanned until the data type is determined. To import large files, consider setting the value to false, which scans the file for five seconds.

`Select Columns(col1, ...)` Specifies the quoted column names in the JMP data table or other JMP file that should be included in the data table.

"Strip Quotes"|Strip Enclosing Quotes(*Boolean*) If the fields in the text file are quoted, setting this to True removes the quotes, and setting it to False does not remove the quotes. The default value is True.

Table Contains Column Headers(*Boolean*) Indicates the imported text file contains labels or column headers as the first line in the file. The default value is True.

"Text Wizard" Opens the text file in the text import window, where you can select import options. Otherwise, the Text Data Files options in the JMP preferences apply, and the text file is automatically imported as a data table.

Treat Empty Columns as Numeric(*Boolean*) Specifies that JMP should import text file columns of missing data as numeric rather than character. Possible missing value indicators are a period, a Unicode dot, NaN, or a blank quoted string. The default value is False.

Use Apostrophe as Quotation Mark(*Boolean*) Declares apostrophes as quotation marks in importing text files. This option is not recommended unless your data comes from a nonstandard source that places apostrophes around data fields rather than quotation marks. The default value is False.

Use Labels for Var Names(*Boolean*) For SAS data sets, this option specifies to use SAS labels as JMP columns names. The default value is False.

Use for All Sheets(*Boolean*) Indicates that JMP should use the Worksheets settings for all worksheets in the Excel file to be opened as a data table.

Worksheet Settings(*Boolean*, <*options*>) specifies options for importing an Excel file into a JMP data table. Available options are:

- Has Column Headers(*Boolean*) specifies that the Excel file has column headers in the first row.
- Number of Rows in Headers(*n*) specifies the number of rows in the Excel file used as column headers.
- Headers Start on Row(*n*) specifies the row number in the Excel file where the column headers begin. Default is row 1.
- Data Starts on Row(*n*) specifies the row number in the Excel file where the data begins.
- Data Starts on Column(*n*) specifies the column number in the Excel file where the data begins.
- Data Ends on Row(*n*) specifies the row number in the Excel file where data ends.
- Data Ends on Column(*n*) specifies the column number in the Excel file where the data ends.
- Replicated Spanned Rows(*Boolean*) specifies that the Excel file contains spanned columns should be imported into JMP as spanned columns.
- Suppress Hidden Rows(*Boolean*) indicates that JMP should not import rows hidden in the Excel file.

– `Suppress Hidden Columns(`*`Boolean`*`)` indicates that JMP should not import columns hidden in the Excel file.

– `Treat as Hierarchy(`*`Boolean`*`)` indicates that JMP should treat multiple column headers (merged cells) as hierarchies when importing an Excel file. If `True`, the Excel file opens with the merged columns stacked (Tables > Stacked).

`Worksheets("Sheet Name"|{"Sheet Name", "Sheet Name",...}|`*`n`*`)` Opens the specified Excel file worksheet by name, all worksheets in a list of names, or the indexed number of the worksheet. If the worksheets are not specified, all worksheets in the spreadsheet open as separate data tables.

Note: You can import .xls worksheets from a web site by specifying worksheets arguments. .xlsx worksheets cannot be imported from a web site using `Open()`.

`"XML Wizard"` Opens XML files in the XML Import Wizard. If you omit this argument, the XML file opens as a data table.

```
Year Rule | Two Digit Year Rule
  ("1900-1999"|"1910-2009"|"1920-2019"|"1930-2029"|"1940-2039"|
  "1950-2049"|"1960-2059"|"1970-2069"|"1980-2079"|"1990-2089"|"2000-2099")
```

Indicates the year format used in the imported text file. For example, if the earliest date is 1979, use "1970-2069".

`Open Database(dataSourceName|"Connect Dialog", "SELECT ..."|"SQLFILE..."|tableName, <"Invisible"|"Private">, <outputTableName>)`

Description

Opens the database indicated by *`dataSourceName`* (or opened by the user) with the *`SELECT`* , *`SQLFILE`*, or *`tableName`* arguments.

Returns

A data table named *`outputTableName`*.

Required Argument

`dataSourceName``|"Connect Dialog"` dataSourceName specifies the name of the data source. `"Connect Dialog"` shows a window from which the user selects the data source.

Optional Arguments

`"Invisible"` Creates an invisible data table that hides the table from view but lists it in the JMP Home Window and Window menu.

`"Private"` Hides the data table completely. Creating a private data table speeds the process of getting to the data; it does not save the computer from allocating the memory necessary to hold the data table data.

`outputTableName` The name of the data table in JMP.

Example

```
Open Database(
  "DSN=dBASE Files;DBQ=C:/Program Files/SAS/JMPPRO/16/Samples/Import Data/;",
```

```
    // SQL statement
    "SELECT HEIGHT, WEIGHT FROM Bigclass", // selected columns
    "hw" // name of the output data table
);
```

Parse JSON(JSON string)

Description

Converts JSON string into an associative array or list based on the structure of the JSON data. Convert the result to a list if the parsed JSON object contains more than one member. The parsing of an empty value, "" quoted string, missing value, or any other invalid value returns Empty().

Example

The following example converts JSON into a list:

```
j = Parse JSON(
    "[ { \!"name\!": \!"KATIE\!", \!"age\!": 12, \!"sex\!": \!"F\!",
    \!"height\!": 59, \!"weight\!": 95 }, { \!"name\!": \!"LOUISE\!",
    \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!": 61, \!"weight\!": 123 }, {
    \!"name\!": \!"JANE\!", \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!":
    55, \!"weight\!": 74 } ]"
);
Show( j );
    j = {["age" => 12, "height" => 59, "name" => "KATIE", "sex" => "F", "weight"
       => 95], ["age" => 12, "height" => 61, "name" => "LOUISE", "sex" => "F",
       "weight" => 123], ["age" => 12, "height" => 55, "name" => "JANE", "sex"
       => "F", "weight" => 74]};
```

Pick Directory(<"Prompt">, <*path*>, <Show Files(*Boolean*)>)

Description

Prompts the user for a directory, returning the directory path as a quoted string.

Returns

The path for the directory that the user selects.

Optional Arguments

"Prompt" The quoted string appears at the top of the Browse window (Windows) or the Finder window (macOS).

path The quoted string specifies the initial directory that appears in the Pick Directory window.

Show Files(*Boolean*) 1 shows files in the Pick Directory window. 0 hide the files. The default is 0.

Pick File(<"Prompt">, <*initial directory*>, <{*filter list*}>, <*first filter*>, <Save Flag(*Boolean*)>, <default file>), <"Multiple">)

Description

Prompts the user to select one or more files in the Open window.

Returns

The path of the file that the user selects.

Optional Arguments

`"Prompt"` A quoted string. If provided, that string appears at the top of the Open window.

`initial directory` A quoted string that is a valid filepath to a folder. If provided, it specifies where the Open window begins. If not provided, or if it's an empty string, the JMP Default Directory is used.

`filter list` A list of quoted strings that define the filetypes to show in the Open window. See the following example for syntax.

`first filter` An integer that specifies which of the filters in the filter list to use initially. If you use an integer that is too large or small for the list (for example, 4 for a list of 3), the first filter in the list is used.

`Save Flag(`*`Boolean`*`)` A Boolean that specifies whether the Open window or Save window is used. 0 lets the user select a file to open in JMP. 1 lets the user save a new, empty file of the selected type in the selected folder. The default value is 0.

`default file` The quoted name of the file that appears in the window by default.

`"Multiple"` The quoted name that lets the user select multiple files if the `Save Flag` value is 0.

Notes

Although all arguments are optional, they are also positional. For example, you cannot specify a filter list without also specifying the caption and the initial directory.

The buffer size in the computer's physical memory affects the number of files the user can open.

Example

The following script assigns *Select JMP File* as the window title; shows the JMP Samples/Data directory; shows *JMP Files* and *All Files* in the File name list and selects *JMP Files*; displays the Open window; and shows the sample data file name Hollywood Movies.jmp.

```
Pick File(
  "Select JMP File",
  "$SAMPLE_DATA",
  {"JMP Files|jmp;jsl;jrn", "All Files|*"},
  1,
  0,
  "Hollywood Movies.jmp"
```

```
);
```

Rename Directory(*old path name*, *new directory name*)

Description

Renames a directory without moving or copying it.

Returns

Returns 1 if the directory is renamed; otherwise, returns 0.

Arguments

old path name A quoted string that specifies the path and old directory name.

new directory name A quoted string that specifies the new directory name.

Notes

When you specify the *new directory Name*, include only the directory name, not the entire path.

Rename File(*old path name*, *new file name*)

Description

Renames a file without moving or copying it.

Returns

Returns 1 if the file is renamed; otherwise, returns 0.

Arguments

old path name Specifies the path and old file name.

new file name Specifies the new file name.

Notes

When you specify the *new name*, include only the file name, not the entire path.

Save Text File(*path*, *text*|BLOB, <Mode("Replace"|"Append")>)

Description

Returns the path name of the created file. *text* is quoted string.

Set Default Directory(*path*)

Description

Sets the default directory, which is used for resolving relative paths.

See Also

"Get Default Directory()"

Set File Search Path(*path*|{*list of paths*})

Description

Sets the current list of directories to search for opening files. Using {"."} as the path configures JMP to use the current directory.

Example

```
Set File Search Path( {"C:/JMP/13/source", "C:/Program
  Files/SAS/JMPPRO/16/Samples"} );
```

See Also

"Get File Search Path()".

Set Path Variable(*name*, <*path*>)

Description

Sets the path stored in the variable.

Required Argument

name The quoted name of the variable.

Optional Argument

path The path.

TripleS Import(*path*, <*arguments*>)

Description

Imports the specified Triple-S Survey (SSS) file. The SSS format consists of a pair of files: .xml or .sss, and a .csv, .dat, or .asc file. Both sets of files must have the same root name and be in the same folder.

Required Argument

path A quoted string that contains the full path to the .xml or .sss file.

Optional Arguments

"Invisible" A quoted keyword that hides the table from view. The data table appears only in the JMP Home Window and the Window menu. Hidden data tables remain in memory until they are explicitly closed, reducing the amount of memory that is available to JMP. To explicitly close the hidden data table, call Close(*dt*), where *dt* is the data table reference returned by TripleS Import.

Use Labels for Imported Column Names(*Boolean*) Converts the label names to column headings. The default value is 1.

Example

```
dt = TripleS Import( "C:/Data/airlines.sss", "Invisible", Use Labels for
  Imported Column Names( 0 ) );
```

Financial Functions

Double Declining Balance(*cost*, *salvage*, *life*, *period*, <*factor*>)

Description

Returns the depreciation of an asset for a specified period of time. The function uses the double-declining balance method or some other depreciation factor.

Required Arguments

cost The initial cost.

salvage The value at the end of the depreciation.

life The number of periods in the depreciation cycle.

period The length of the period, in the same units as *life*.

Optional Argument

factor The rate at which the balance declines. The default value is 2.

Notes

This function is equivalent to the Excel function DDB.

Future Value(*rate*, *nper*, *pmt*, <*pv*>, <*Type(Boolean)*>)

Description

Returns the future value of an investment that is based on periodic, constant payments and a constant interest rate.

Required Arguments

rate The interest rate.

nper The number of periods.

pmt The constant payment.

Optional Arguments

pv The present value. The default value is 0.

Type(*Boolean*) 0 specifies end-of-period payments. 1 specifies beginning-of-period payments. The default value is 0.

Notes

This function is equivalent to the Excel function FV.

Interest Payment(*rate*, *per*, *nper*, *pv*, <*fv*>, <*Type(Boolean)*>)

Description

Returns the interest payment for a given period for an investment that is based on periodic, constant payments and a constant interest rate.

Required Arguments

rate The interest rate.

per The period for which you want the interest.

nper The total number of periods.

pv The present value.

Optional Arguments

fv The future value. The default value is 0.

Type(Boolean) 0 specifies end-of-period payments, and 1 specifies beginning-of-period payments. The default value is 0.

Notes

This function is equivalent to the Excel function IPMT.

Interest Rate(*nper*, *pmt*, *pv*, <*fv*>, <*Type(Boolean)*>, <*guess*>)

Description

Returns the interest rate per period of an annuity.

Required Arguments

nper The total number of periods.

pmt The constant payment.

pv The present value.

Optional Arguments

fv The future value. The default value is 0.

Type(Boolean) 0 specifies end-of-period payments, and 1 specifies beginning-of-period payments. The default value is 0.

guess Your estimate of the rate The default value is 0.1 (10%).

Notes

This function is equivalent to the Excel function RATE.

Internal Rate of Return(*values*, <*guess*>)

Internal Rate of Return(*guess*, *value1*, *value2*, ...)

Description

Returns the internal rate of return for a series of cash flows in the *values* argument.

Arguments

values A one-dimensional matrix of values. If the second form of the function is used, list each value separately.

guess What you think is near the result. The default value is 0.1 (10%). This argument is required if you specify values. It is required if you specify individual values.

Notes

This function is equivalent to the Excel function IRR.

`Modified Internal Rate of Return(`*values*`, `*finance rate*`, `*reinvest rate*`)`

`Modified Internal Rate of Return(`*finance rate*`, `*reinvest rate*`, `*value1*`, `*value2*`, ...)`

Description

Returns the modified internal rate of return for a series of periodic cash flows. The cost of investment and the interest received on reinvested cash is included.

Arguments

values A one-dimensional matrix of values. If the second form of the function is used, list each value separately.

finance rate The interest rate that you pay on the money in the cash flows.

reinvest rate The interest rate that you receive on the cash flows when you reinvest them.

Notes

This function is equivalent to the Excel function MIRR.

`Net Present Value(`*rate*`, `*values*`)`

`Net Present Value(`*rate*`, `*value1*`, `*value2*`, ...)`

Description

Returns the net present value of an investment by using a discount rate and a series of future payments (negative values) and income (positive values).

Arguments

rate The discount rate.

values A one-dimensional matrix of values. If the second form of the function is used, list each value separately.

Notes

This function is equivalent to the Excel function NPV.

`Number of Periods(`*rate*`, `*pmt*`, `*pv*`, <`*fv*`>, <`*type(0|1)*`>)`

Description

Returns the number of periods for an investment that is based on periodic, constant payments and a constant interest rate.

Required Arguments

rate The interest rate.

pmt The constant payment.

pv The present value.

Optional Arguments

fv The future value. The default value is 0.

type(0|1) 0 specifies end-of-period payments, and 1 specifies beginning-of-period payments. The default value is 0.

Notes

This function is equivalent to the Excel function NPER.

`Payment(`*rate*`, `*nper*`, `*pv*`, <`*fv*`>, <`*type(0|1)*`>)`

Description

Returns the payment for a loan that is based on constant payments and a constant interest rate.

Required Arguments

rate The interest rate.

nper The total number of periods.

pv The present value.

Optional Arguments

fv The future value. The default value is 0.

type(0|1) 0 specifies end-of-period payments, and 1 specifies beginning-of-period payments. The default value is 0.

Notes

This function is equivalent to the Excel function PMT.

`Present Value(`*rate*`, `*nper*`, `*pmt*`, <`*fv*`>, <`*type(0|1)*`>)`

Description

Returns the present value of an investment.

Arguments

rate The interest rate per period.

nper The total number of periods.

pmt The constant payment.

fv The future value. The default value is 0.

type(0|1) 0 specifies end-of-period payments, and 1 specifies beginning-of-period payments. The default value is 0.

Notes

This function is equivalent to the Excel function PV.

Principal Payment(*rate*, *per*, *nper*, *pv*, <*fv*>, <*type(0|1)*>)

Description

Returns the payment on the principal for a given period for an investment that is based on periodic, constant payments and a constant interest rate.

Required Arguments

rate The interest rate per period.

per The period for which you want the interest.

nper The total number of periods.

pv The present value.

Optional Arguments

fv The future value. The default value is 0.

type(0|1) An optional switch. 0 specifies end-of-period payments, and 1 specifies beginning-of-period payments. The default value is 0.

Notes

This function is equivalent to the Excel function PPMT.

Straight Line Depreciation(*cost*, *salvage*, *life*)

Description

Returns the straight-line depreciation of an asset for one period.

Arguments

cost The initial cost of the asset.

salvage The value at the end of the depreciation.

life The number of periods in the depreciation cycle.

Notes

This function is equivalent to the Excel function SLN.

Sum Of Years Digits Depreciation(*cost*, *salvage*, *life*, *per*)

Description

Returns the sum-of-years' digits depreciation of an asset for a specified period.

Arguments

cost The initial cost of the asset.

salvage The value at the end of the depreciation.

life The number of periods in the depreciation cycle.

per The length of the period, in the same units as life.

Notes

This function is equivalent to the Excel function SYD.

Graphics Functions

Add Color Theme({*name*, <*flags*>, {*color*}, <{*position*}>)

Description

Creates a custom color theme that you can apply to components such as markers, data table rows, and treemaps. Add the color theme to the JMP Preferences by including `Add Color Theme(...)` inside `Preferences()`.

Returns

Null.

Required Arguments

`name` The quoted name of the color theme.

`color` Lists of RGB values. These values define the blocks in categorical color themes and the gradients in continuous color themes. Each list of RGB values corresponds to a slider in the preferences Color Themes window.

`position` An optional list of numbers between 0 and 1 with one position per color. Each position corresponds to a slider in the preferences Color Themes window. If you omit the position, the sliders are evenly spaced.

Optional Argument

`flags` A flag for the Continuous or Categorical color theme list and category of color.

```
Continuous, <Continuous>, Sequential
Continuous, <Continuous>, Diverging
Continuous, <Continuous>, Chromatic
Categorical, <Continuous>, Sequential
Categorical, <Continuous>, Diverging
Categorical, Qualitative
Categorical, <Continuous>, Chromatic
```

With the default JMP color themes, Sequential colors transition from left to right or right to left. Diverging colors are lighter in the middle. Chromatic colors consist of blocks or gradients of bright color. All categories except for Qualitative can be both continuous and categorical.

If you omit the flag, the color is shown in the Continuous, Sequential and Categorical, Sequential categories.

Examples

The following example creates a continuous color theme named Blue to Purple. The color is in the Diverging category. RGB values are defined in the four lists.

```
Add Color Theme(
  {"Blue to Purple", {"Continuous", "Diverging"}, {{0, 0, 255},
  {57, 108, 244}, "white", {128, 0, 100}}} );
```

Notes

Any style except for Qualitative can be Continuous and Categorical at the same time. For example, the Cool to Warm Diverging theme is in the Continuous and Categorical theme lists. In JMP, select **Preferences > Graphs** to see examples.

To delete a color theme, use `Remove Color Theme()`.

See Also

"Remove Color Theme(name|{name, <flags>, {color, ...}, <{position, ...}>}>)"

`Arc(x1, y1, x2, y2, startangle, endangle)`

Description

Inscribes an arc in the rectangle described by the arguments.

Returns

Null.

Arguments

x1, *y1* The point at the top left of the rectangle

x2, *y2* The point at the bottom right of the rectangle

startangle, *endangle* The starting and ending angle in degrees, where 0 degrees is 12 o'clock and the arc or slice is drawn clockwise from *startangle* to *endangle*.

`Arrow(<pixellength>, {x1, y1}, {x2, y2})`

Description

Draws an arrow from the first point to the second point. The optional first argument specifies the length of the arrow's head lines (in pixels).

Returns

Null.

Required Arguments

{*x1*, *y1*}, {*x2*, *y2*} Two lists of two numbers that each specify a point in the graph.

Optional Argument

pixellength The length of the arrowhead in pixels.

Notes

The two points can also be enclosed in square brackets: `Arrow(<pixellength>, [x1, x2], [y1, y2])`.

`Back Color(name)`

Description

Sets the color used for filling the graph's background.

Returns

Null.

Argument

`name` A quoted color name or a color index (such as `"red"` or `"3"` for the color red).

`Char To Path(path)`

Description

Converts a path specification from a quoted string to a matrix.

Returns

A matrix.

Arguments

`path` A quoted string that contains the path specification.

`Circle({x, y}, radius|PixelRadius(n), <...>, <"Fill">)`

Description

Draws a circle centered at {x, y} with the specified radius.

Returns

Null.

Required Arguments

`{x, y}` A number that describes a point in the graph

`radius` The length of the circle's radius in relation to the vertical axis. If the vertical axis is resized, the circle is also resized.

`PixelRadius(n)` The length of the circle's radius in pixels. If the vertical axis is resized, the circle is *not* resized.

Optional Argument

`"Fill"` A positional argument indicating that all circles defined in the function are filled with the current fill color. If `"Fill"` is omitted, the circle is empty.

Notes

The center point and the radius can be placed in any order. You can also add additional center point and radius arguments and draw more than one circle in one statement. One point and several radii results in a bull's-eye. Adding another point still draws all previous circles, and then adds an additional circle with the last radius specified. This means that this code:

```
Graph Box(circle({20, 30}, 5, {50, 50}, 15))
```

results in three circles, not two. First, a circle with radius 5 is drawn at 20, 30. Second, a circle with radius 5 is drawn at 50, 50. Third, a circle with radius 15 is drawn at 50, 50.

Color To HLS(*color*)

Description

Converts the *color* argument (including any JMP color) to a list of HLS values.

Returns

A list of the hue, lightness, and saturation components of *color*. The values range between 0 and 1.

Argument

color A number from the JMP color index.

Example

The output from ColorToHLS() can either be assigned to a single list variable or to a list of three scalar variables:

```
hls = Color To HLS( 8 );
{h, l, s} = Color To HLS( 8 );
Show( hls, h, l, s );
   hls = {0.778005464480874, 0.509803921568627, 0.976};
   h = 0.778005464480874;
   l = 0.509803921568627;
   s = 0.976;
```

Color To RGB(*color*)

Description

Converts the *color* argument (including any JMP color) to a list of RGB values.

Returns

A list of the red, green, and blue components of *color*. The values range between 0 and 1.

Argument

color A number from the JMP color index.

Example

The output from ColorToRGB() can either be assigned to a single list variable or to a list of three scalar variables:

```
rgb = Color To RGB( 8 );
{r, g, b} = Color To RGB( 8 );
Show( rgb, r, g, b );
   rgb = {0.670588235294118, 0.0313725490196078, 0.988235294117647};
   r = 0.670588235294118;
   g = 0.0313725490196078;
   b = 0.988235294117647;
```

Contour(*xVector*, *yVector*, *zGridMatrix*, *zContour*, <*messages*>)

Description

Draws contours given a grid of values.

Returns

None.

Required Arguments

xVector The *n* values that describe *zGridMatrix*.

yVector The *m* values that describe *zGridMatrix*.

zGridMatrix An *nxm* matrix of values on some surface.

zContour The values for the contour lines.

Optional Messages

<<zColor(color, option) The colors to use for the contour lines.

<<"Fill"|"Fill Between"|"Fill Below"|"Fill Above" Specifies where the contour fill appears.

<<Transparency(vector) Specifies the transparency level of the fill in a vector.

Contour Function(*zExpr*, *xName*, *yName*, (*z*|*zMatrix*), < <<xGrid(*min*, *max*, *incr*)>, < <<yGrid(*min*, *max*, *incr*)>, < <<zColor(*color*, *option*)>, < <<*zLabeled*>, < <<"Filled">, < <<"FillBetween">, < <<"Ternary">, < <<Transparency(*number*|*matrix*))

Description

Draws sets of contour lines of the expression, a function of the two symbols. The *z* argument can be a single value or an index or matrix of values.

Returns

None.

Arguments

zExpr Any expression (for example, Sine(y)+Cosine(x)).

xName, yName The values to use in the expression.

z|zMatrix A z-value or a matrix of z-values.

Optional Messages

<<xGrid, <<yGrid Defines a box, beyond which the contour lines are not drawn.

<<zColor Defines the color in which to draw the contour lines. The argument can be either a scalar or a matrix but must evaluate to numeric.

<<zLabeled Labels the contours.

<<"Filled" Fills the contour levels using the current fill color.

<<"FillBetween" Fills only between adjacent contours using the current fill color. For nz contours specified, this option fills nz-1 regions for the intervals between the nz values. Using this option is recommended over using the <<Filled option.

<<"Ternary" Clips lines to be within the ternary coordinate system inside ternary plots.

<<Transparency(*number*|*matrix*) Sets the transparency level of the fill. A vector of numbers between 0 and 1 are sequenced through and cycled for the z contours. This option should be used only in conjunction with the <<FillBetween option.

Drag Line(*xMatrix*, *yMatrix*, <*dragScript*>, <*mouseUpScript*>)

Description

Draws line segments between draggable vertices at the coordinates given by the matrix arguments.

Returns

None.

Required Arguments

xMatrix A matrix of *x*-coordinates.

yMatrix A matrix of *y*-coordinates.

Optional Arguments

dragScript Any valid JSL script. The script is run at drag.

mouseUpScript Any valid JSL script. The script is run at mouseup.

Drag Marker(*xMatrix*, *yMatrix*, <*dragScript*>, <*mouseUpScript*>)

Description

Draws draggable markers at the coordinates given by the matrix arguments.

Returns

None.

Arguments

xMatrix A matrix of *x*-coordinates.

yMatrix A matrix of *y*-coordinates.

dragScript Any valid JSL script. The script is run at drag.

mouseUpScript Any valid JSL script. The script is run at mouseup.

Drag Polygon(*xMatrix*, *yMatrix*, <*dragScript*>, <*mouseupScript*>)

Description

Draws a filled polygon with draggable vertices at the coordinates given by the matrix arguments.

Returns

None.

Required Arguments

xMatrix A matrix of *x*-coordinates.

yMatrix A matrix of *y*-coordinates.

Optional Arguments

dragScript Any valid JSL script. The script is run at drag.

mouseUpScript Any valid JSL script. The script is run at mouseup.

Drag Rect(*xMatrix*, *yMatrix*, <*dragScript*>, <*mouseupScript*>)

Description

Draws a filled rectangle with draggable vertices at the first two coordinates given by the matrix arguments.

Returns

None.

Required Arguments

xMatrix A matrix of two *x*-coordinates.

yMatrix A matrix of two *y*-coordinates.

Optional Arguments

dragScript Any valid JSL script. The script is run at drag.

mouseUpScript Any valid JSL script. The script is run at mouseup.

Notes

xMatrix and *yMatrix* should each contain exactly two values. The resulting coordinate pairs should follow the rules for drawing a `Rect()`. The first point (given by the first value in *xMatrix* and the first value in *yMatrix*) must describe the top, left point in the rectangle. The second point (given by the second value in *xMatrix* and the second value in *yMatrix*) must describe the bottom, right point in the rectangle.

Drag Text(*xMatrix*, *yMatrix*, *string*, <*dragScript*>, <*mouseUpScript*>)

Description

Draws the *string* (or all the items if a list is specified) at the coordinates given by the matrix arguments.

Returns

None.

Arguments

xMatrix A matrix of *x*-coordinates.

yMatrix A matrix of *y*-coordinates.

string A quoted string to be drawn in the graph.

dragScript Any valid JSL script. The script is run at drag.

mouseUpScript Any valid JSL script. The script is run at mouseup.

Fill Color(*name*|*color*|*rgbList*)

Description

Sets the color used for filling solid areas.

Returns

None.

Arguments

name|color|rgbList The quoted color name, index, or list of RGB values.

`Fill Pattern(`*name|mask|image*`)`

Description

Sets the pattern for filled areas.

Arguments

name|mask|image The quoted color name, matrix of values between 0 and 1, or image path.

`Get Color Theme Detail(`*name*`)`

Description

Returns a script for the specified color theme.

Example

The following example returns the script for the JMP default color theme:

```
Get Color Theme Details( "JMP Default" );
  {"JMP Default", 9221, {{213, 72, 87}, {57, 177, 67}, {64, 111, 223}...}}
```

`Get Color Theme Names(<`*kind*`>)`

Description

Returns a list of all color theme names or color themes of the specified kind. The kinds include `"Continuous"`, `"Categorical"`, `"Sequential"`, `"Diverging"`, `"Qualitative"`, or `"Chromatic"`.

Example

The following example returns all color themes:

```
Get Color Theme Names();
  {"Green to Black to Red", "Green to White to Red", "White to Black"...}
```

The following example returns the diverging color themes:

```
Get Color Theme Names( "diverging" );
  {"Green to Black to Red", "Green to White to Red", "Blue to Gray to Red"...}
```

`Gradient Function(`*zExpr*`, `*xName*`, `*yName*`, [`*zLow*`, `*zHigh*`], zColor([`*colorLow*`, `*colorHigh*`]), < <<xGrid(`*min*`, `*max*`, `*incr*`)>, < <<yGrid(`*min*`, `*max*`, `*incr*`)> < <<Transparency(`*alpha|vector*`))`

Description

Fills a set of rectangles on a grid according to a color determined by the expression value as it crosses a range corresponding to a range of colors.

Required Arguments

zExpr Any expression (for example, Sine(y)+Cosine(x)).

xName, *yName* The values to use in the expression.

zLow, *zHigh* A matrix of low and high values (2 to 10).

zColor([colorLow, colorHigh) The two colors that are blended together (4 is green, 6 is orange).

Optional Messages

<<*xGrid*, <<*yGrid* Specifies a box, beyond which the gradients are not drawn.

<<*zColor* The color in which to draw the contour lines. The argument can be either a scalar or a matrix but must evaluate to numeric.

<<Transparency(*number*|*matrix*) The transparency level of the fill. A vector of numbers between 0 and 1 are sequenced through and cycled for the z contours.

Example

```
Gradient Function(Log(a * a + b * b),
a, b, [2 10],
zColor([4, 6]));
```

H Line(*x1*, *x2*, *y*)

H Line(*y*)

Description

Draws a horizontal line at *y* across the graph. If you specify start and end points on the *x*-axis (*x1* and *x2*), the line is drawn horizontally at *y* from *x1* to *x2*. You can also draw multiple lines by using a matrix of values in the *y* argument.

H Size()

Description

Returns the horizontal size of the graphics frame in pixels.

Handle(*xPos*, *yPos*, *dragScript*, *mouseUpScript*)

Description

Places draggable marker at coordinates given by *xPos*, *yPos*. The first script is executed at drag and the second at mouseup.

Heat Color(*x*, < <<*theme*>)

Heat Color(*x*)

Description

Returns the JMP color that corresponds to *n* in the color "*theme*".

Returns

An integer that is a JMP color.

Required Argument

x A value between 0 and 1.

Optional Message

theme A quoted color theme that is supported by Cell Plot. The default value is "Blue to Gray to Red".

`HLS Color(h, l, s)`

`HLS Color({h, l, s})`

Description

Converts hue, lightness, and saturation values into a JMP color number.

Returns

An integer that is a JMP color number.

Arguments

Hue, lightness, and saturation, or a list containing the three HLS values. All values should be between 0 and 1.

`In Path(x, y, (pathMatrix|pathText)`

Description

Determines it the point described by *x* and *y* falls in *path*.

Returns

True (1) if the point (x, y) is in the given path, False(0) otherwise.

Arguments

x and *y* The coordinates of a point.

pathMatrix|*pathText* Either a matrix or a quoted string describing a path.

`In Polygon(x, y, xMatrix, yMatrix)`

`In Polygon(x, y, xyPolygon)`

Description

Returns 1 or 0, indicating whether the point (*x*, *y*) is inside the polygon that is defined by the *xMatrix* and *yMatrix* vector arguments.

xMatrix, *yMatrix* can also be combined into a two-column matrix (*xyPolygon*), allowing you to use three arguments instead of four. Also, *x* and *y* can be conformable vectors, and then a vector of 0s and 1s are returned based on whether each (*x*, *y*) pair is inside the polygon.

Level Color(*i*)

Level Color(*i*, *n*)

Level Color(*i*, *n*, <*theme*>)

Level Color(*i*, <*theme*>)

Description

Assigns a JMP color to categorical data in a graphic.

Returns

An integer that is a JMP color.

Required Arguments

i An integer that is greater than or equal to 1 and less than or equal to the number of categories specified by n.

n The number of categories.

Optional Argument

theme A quoted color theme from the Value Color list of the Column Properties window. If not specified, the JMP Default color theme is applied.

Notes

When the second argument is a quoted character string and not *n*, then the second argument determines the color theme.

Line({*x1*, *y1*}, {*x2*, *y2*}, ...), <<ValueSpace(*Boolean*)

Line([*x1*, *x2*, ...], [*y1*, *y2*, ...]), <<ValueSpace(*Boolean*)

Description

Draws a line between points.

Arguments

{*x1*, *y1*}, {*x2*, *y2*}|[*x1*, *x2*, ...], [*y1*, *y2*, ...] Can be any number of lists of two points, separated by commas or a matrix of *x* values and a matrix of *y* values.

<<ValueSpace(*Boolean*) Draws lines that follow the projection when the line represents a movement of the underlying data, such as a bubble trail in a bubble plot. The Boolean value can be a constant or an expression.

Line Style(*name*|*number*)

Description

Sets the line style used to draw the graph.

Argument

n Can be either a quoted style name or the style's number:

– 0 or "Solid"

- 1 or "Dotted"
- 2 or "Dashed"
- 3 or "DashDot"
- 4 or "DashDotDot"

Marker(<*rowState*>, {*x1*, *y1*}, {*x2*, *y2*}, ...)

Marker(<*rowState*>, [*x1*, *x2*, ...], [*y1*, *y2*, ...])

Description

Draws one or more markers at the points described either by lists or matrices. The optional *rowState* argument sets the type of marker.

Marker Size(*n*)

Description

Sets the size used for markers.

Mousetrap(*dragScript*, *mouseUpscript*)

Description

Captures click coordinates to update graph properties. The first script is executed at drag and the second script at mouseup.

New Heat Image(*matrix*, <*colorTheme*>)

Description

Creates a heat map image based on a matrix and color theme or gradient.

Arguments

matrix A matrix of values. The function creates a heat map that has the same dimensions as the input matrix.

colorTheme A color theme or set of gradient properties that define the mapping of values to colors. The example below shows the use of a custom gradient.

Example

```
width = 256;
height = 256;
wstride = 16;
hstride = 16;
b = J( hstride, wstride, Random Normal() );
a = Transform Each( {z, {row, col}},
  J( height, width, 0 ),
  b[Floor( (row - 1) / hstride ) + 1,
  Floor( (col - 1) / wstride ) + 1]
```

```
);
E1 = New Heat Image(
   a,
   gradient(
      {Color Theme( "Blue To Gray To Orange" ),
      Scale Type( "Standard Deviation" ),
      Show Missing Color( "On" ),
      Reverse Gradient( 1 )}
   )
);
New Window( "test", Lineup Box( E1 ) );
```

Normal Contour(*prob*, *meanMatrix*, *stdMatrix*, *corrMatrix*, <*colorsMatrix*>, <*fill=x*>)

Description

Draws normal probability contours for *k* populations and two variables.

Required Arguments

prob A scalar or matrix of probabilities.

meanMatrix A matrix of means of size *k* by 2.

stdMatrix A matrix of standard deviations of size *k* by 2.

corrMatrix A matrix of correlations of size *k* by 1.

Optional Arguments

colorsMatrix Specifies the color(s) for the k contour(s). The colors must be specified as JSL colors (either JSL color integer values or return values of JSL Color functions such as RGB Color or HLS Color).

fill=x Specifies the amount of transparency for the contour fill color.

Oval(*x1*, *y1*, *x2*, *y2*, <*Fill(Boolean)*>)

Oval({*x1*, *y1*}, {*x2*, *y2*}, <*Fill(Boolean)*>)

Description

Draws an oval inside the rectangle whose diagonal has the coordinates (*x1, y1*) and (*x2, y2*). *Fill* is Boolean. If *Fill* is 0, the oval is empty. If *Fill* is nonzero, the oval is filled with the current fill color. The default value for *Fill* is 0.

Path((*pathMatrix*|*pathText*), <*Fill(Boolean)*>)

Description

Draws a stroke along the given path. If a fill is specified, the interior of the path is filled with the current fill color.

Required Arguments

pathMatrix An Nx3 matrix. If you don't specify the path matrix, specify the path text.

pathText A quoted string that contains SVG code.

Optional Arguments

Fill(Boolean) Specifies whether a line is drawn (0) or the path is filled (1). The default value is 0.

Notes

A path matrix has three columns, for x and y, and a flag. The flag value for each point can be 0 for control, 1 for move, 2 for line segment, 3 for cubic Bézier segment, and any negative value to close the path.

Path To Char(*path*)

Description

Converts a path specification from a matrix to a quoted string.

Returns

A quoted string.

Argument

path An Nx3 path matrix.

Notes

A path matrix has three columns, for x and y, and a flag. The flag value for each point can be 0 for control, 1 for move, 2 for line segment, 3 for cubic Bézier segment, and any negative value to close the path.

Pen Color(*n*)

Description

Sets the color used for the pen.

Pen Size(*n*)

Description

Sets the thickness of the pen in pixels.

Pick Color(<*window title*>, <*name*|*index*|*RGBlist*>)

Description

Creates a color picker, which enables the user to select a color to apply to graphs. The operating system color picker lets users select a predefined color or create their own color. You can also specify a default color in your script. If you omit the default color, Black is selected.

Returns

The color that the user selected in the operating system's color picker.

Arguments

window title The quoted title of the color picker window.

name The name of the default JMP color.

index The number of the default JMP color.

RGBlist The RGB values of the default color.

Pie(*left*, *top*, *right*, *bottom*, *startAngle*, *endAngle*)

Description

Draws a filled pie slice. The two points describe a rectangle, within which is a virtual oval. Only the slice described by the start and end angles is drawn.

Pixel Line To(*h*, *v*)

Description

Draws a one-pixel-wide line from the current pixel location to the location given in pixel coordinates. Set the current pixel location using the `Pixel Origin()` and `Pixel Move To()` functions.

Pixel Move To(*h*, *v*)

Description

Moves the current pixel location to a new location given in pixel coordinates.

Pixel Origin(x, *y*)

Description

Sets the origin, in graph coordinates, for subsequent `Pixel Line To()` or `Pixel Move To()` functions.

Polygon({*x1*, *y1*}, {*x2*, *y2*}, ...)

Polygon(*xMatrix*, *yMatrix*)

Description

Draws a filled polygon defined by the listed points.

`Polygon Area({x1, y1}, {x2, y2}, ...)`

`Polygon Area(xMatrix, yMatrix)`

Description

Calculates the area of the specified polygon.

Examples

```
area = Polygon Area( {0, 0}, {0, 10}, {10, 10}, {10, 0} );
area = Polygon Area( [10 20 30], [10 30 20] );
```

`Polygon Centroid({x1, y1}, {x2, y2}, ...)`

`Polygon Centroid(xMatrix, yMatrix)`

Description

Calculates the centroid of the specified polygon.

Examples

```
{cx, cy} = Polygon Centroid( {0, 0}, {0, 10}, {10, 10}, {10, 0} );
centroid = Polygon Centroid( [10 20 30], [10 30 20] );
```

`Pixel Path(h, v, (path matrix|path text), <fill=Boolean)>, <scale=n>, <orient={n, n})`

Description

Draws a stroke along the given pixel-based path if the fill is 0, or paints the interior of the path if the fill is not 0.

Required Arguments

h, *v* Specifies the horizontal and vertical position.

path matrix Contains three columns for *h*, *v*, and flags for each point in the path. The flag values are 0 for control, 1 for move, 2 for line segment, 3 for cubic Bézier segment, and are negative if the point also closes the path. Each flag (one for each point) can be 0, 1, 2 or 3, (or negative, if it closes the shape).

If you don't specify the path matrix, specify the path text.

path text Supports SVG syntax. The path is scaled (`scale`) and translated about its origin according to the optional parameters (`fill` and `orient`), with the orientation specified in the axis space.

Optional Arguments

`fill(Boolean)` 0 fills the shape. 1 draws an empty shape.

`orient` Rotates the object. For example, an orientation of `{Sqrt( 2 ), Sqrt( 2 )}` draws the shape on a 45-degree angle.

`scale` Sizes the object. For example, a scale of 1.0 draws the shape as it is defined; a scale of 2.0 draws it twice as big; and a scale of 0.5 draws it half as big.

`Pixel Text(<properties>, {h, v}, text, ...)`

Description

Moves to the {*h, v*} pixel position and draws text that the quoted *text* argument specifies.

Required Arguments

h The horizontal pixel position.

v The vertical pixel position.

`text` The quoted string to display at the start and end of the pixels.

Optional Arguments

`center justified` Center justifies the text.

`right justified` Right justifies the text.

`erased` Omits pixels from the edges of the rectangle of text. The text is drawn over the graph content in the background color, erasing the graph within the rectangle to make the text more readable.

`boxed` Displays a box around the text.

`counterclockwise` Rotates the text counterclockwise.

`clockwise` Rotates the text clockwise.

`Rect(left, top, right, bottom, <Fill(Boolean)>)`

`Rect({left, top}, {right, bottom}, <Fill(Boolean)>)`

Description

Draws a rectangle whose diagonal has the coordinates (*left*, *top*) and (*right*, *bottom*). *Fill* is Boolean. If *Fill* is 0, the rectangle is empty. If *Fill* is nonzero, the rectangle is filled with the current fill color. The default value for *Fill* is 0.

`Remove Color Theme(name|{name, <flags>, {color, ...}, <{position, ...}>}>)`

Description

Removes a custom color theme from the global list, either by name or by the full color theme object.

Required Arguments

name The quoted name of the color theme.

color The RGB values for the color.

Optional Arguments

flags A number that represents metadata such as whether the theme is continuous or categorical. Run `Get Color Theme Details("name")` on the color theme and use the flag that is returned.

position A number between 0 and 1. There is one number per color that indicates where on the gradient that color is, where 0 is the beginning and 1 is the end.

Example

```
Remove Color Theme( {"Yellow Blue", 0, {{255, 255, 0}, {0, 0, 255}}, {0.0,
  1.0}} );
```

`RGB Color(r, g, b)`

`RGB Color({r, g, b})`

Description

Converts red, green, and blue values into a JMP color number.

Returns

An integer that is a JMP color number.

Arguments

Red, green, and blue, or a list containing the three RGB values. All values should be between 0 and 1.

`Text(<properties>, ({x, y}|{left, bottom, right, top}), text)`

Description

Draws the quoted string *text* at the given point, either the x and y axes or the left, bottom, right, and top axes.

Properties can be any of several named arguments: `Center Justified`, `Right Justified`, `Erased`, `Boxed`, `Counterclockwise`, `Position`, and named arguments. The position, named arguments, and quoted strings can be added in any order. The position and named arguments apply to all the strings.

`Text Color(n)`

Description

Sets the color for text strings.

`Text Font(fontName, <size>, <bold italic underline strikeout>, <angle>)`

Description

Sets the font for text strings. Use without arguments to get the current font properties. Angle is in degrees clockwise.

Quote the font properties. To apply multiple properties, specify them together, as in `"bold italic"`.

`Text Size(n)`

Description

Sets the font size in points for text strings.

`Transparency(alpha)`

Description

Sets the transparency of the current drawing, with *alpha* between 0 and 1 where 0 is clear (no drawing) and 1 is completely opaque (the default).

Notes

Not all operating systems support transparency.

`V Line(x, <y1, y2>)`

Description

Draws a vertical line at *x* across the graph. If you specify start and end points on the *y*-axis (*y1* and *y2*), the line is drawn vertically at *x* from *y1* to *y2*. You can also draw multiple lines by using a matrix of values in the *x* argument.

`V Size()`

Description

Returns the vertical size of the graphics frame in pixels

`X Function(zExpr, yName, <Min(min), Max(max), Fill(Boolean), Inc(upper bound))`

Description

Draws a plot of the function as the *yName* is varied over the *y*-axis of the graph.

`X Origin()`

Description

Returns the *x*-value for the left edge of the graphics frame.

`X Range()`

Description

Returns the distance from the left to right edge of the display box. For example, `X Origin() + X Range()` is the right edge.

`X Scale(xMin, xMax)`

Description

Sets the range for the horizontal scale. The default value for *xMin* is 0. The default value for *xMax* is 100.

XY Function(*x(t)*, *y(t)*, *t*, Min(*min*), Max(*max*), (Inc(*bound*)|Steps(*min*))

Description

Combines an expression of *x(t)* and *y(t)* to draw an x-y curve for the specified range of parameter *t*. Each time *t* it is varied between Min and Max, the *x* and *y* expressions are evaluated using the current value of *t*.

Note: Either *Inc()* or *Steps()* is needed if the default granularity misses details.

Y Function(*yExpr*, *xName*, <Min(*min*), Max(*max*), Fill(*Boolean*), Inc(*bound*)>)

Description

Draws a plot of the function as the symbol is varied over the *x*-axis of the graph.

Y Origin()

Description

Returns the *y*-value for the bottom edge of the graphics frame.

Y Range()

Description

Returns the distance from the bottom to top edges of a display box. For example, Y Origin() + Y Range() is the top edge.

Y Scale(*yMin*, *yMax*)

Description

Sets the range for the vertical scale. If you do not specify a scale, it defaults to 0, 100.

HTTP Functions

Decode 64 Blob(*string*)

Description

Decodes the quoted string using Base-64 encoding.

Encode 64 Blob(*string*)

Description

Encodes the quoted string using Base-64 encoding.

List Functions

As List(*matrix*)

Description

Converts a matrix into a list. Multi-column matrices are converted to a list of row lists.

Returns

A list.

Argument

matrix Any matrix.

Concat Items({*string1*, *string2*, ...}, <*delimiter*>})

Description

Converts a list of quoted string expressions into one string, with each item separated by a delimiter. The delimiter is a blank if unspecified.

Returns

The concatenated quoted string.

Arguments

string Any quoted string.

delimiter An optional quoted string that is placed between each item. The delimiter can be more than one character long.

Example

```
str1 = "one";
str2 = "two";
str3 = "three";

comb = Concat Items({str1, str2, str3});
   "one two three"
comb = Concat Items({str1, str2, str3}, " : ");
   "one : two : three"
del = ",";
comb = Concat Items({str1, str2, str3}, del);
   "one,two,three"
```

Eval List({*list*})

Description

Evaluates expressions inside *list*.

Returns

A list of the evaluated expressions.

Arguments

list A list of valid JSL expressions.

`Insert(source, item, <position>)`

`Insert(source, key, value)`

Description

Inserts a new *item* into the *source* at the given *position*. If *position* is not given, *item* is added to the end.

For an associative array: Adds the *key* into the *source* associative array and assigns *value* to it. If the *key* exists in *source* already, its value is overwritten with the new *value*.

Required Arguments

source A quoted string, list, vector, expression, or associative array.

item|key Any value to be placed within *source*. For an associative array, *key* might or might not be present in *source*.

value A value to assign to the *key*.

Optional Arguments

position Optional numeric value that specifies the position in *source* to place the *item* into.

`Insert Into(source, item, <position>)`

`Insert Into(source, key, value)`

Description

Inserts a new item into the *source* at the given position in place. The *source* must be an L-value.

Arguments

source A variable that contains a quoted string, list, vector, display box, expression, or associative array.

item|key Any value to be placed within source. For an associative array, *key* might be present in *source*.

position Optional numeric value that specifies the position in *source* to place the *item* into.

value A value to assign to the *key*.

`Is List(x)`

Description

Returns 1 if the evaluated argument is a list, or 0 otherwise.

Items(*string*, <*delimiter*>, <Include Boundary Delimiters(*Boolean*)>)

Description

Returns a list of (possibly empty) quoted sub-strings separated by exactly one of any of the characters specified in the *delimiter* argument.

Arguments

`string` The quoted string that is evaluated.

`Delimiter` (Optional) The character used as a boundary. If *delimiter* is absent, an ASCII space is used. If *delimiter* is the empty quoted string, each character is treated as a separate word.

`Include Boundary Delimiters(Boolean)` (Optional) Returns the empty strings between the boundary and the delimiter.

Example

```
Items( "http://www.jmp.com", ":/." );
   {"http", "", "", "www", "jmp", "com"}
Items(",toy,", ",");
   {"toy"}
Items(",toy,", ",", Include Boundary Delimiters( 1 ));
   {"", "toy", ""}
   /* There is no text between the boundary (the beginning of the quoted string)
      and the comma delimiter, so you get an empty string. The same priniciple
      applies to the delimiter at the end of the string. */
```

List(a, b, c, ...)

{a, b, c, ...}

Description

Constructs a list from a set of items.

N Items(source)

Description

Determines the number of elements in the *source* specified.

Returns

For a list or display box, the number of items in the list or display box is returned. For an associative array, the number of keys is returned. For a matrix, the number of elements in the matrix is returned. For a namespace, the number of functions and variables in the namespace is returned. For a class object, the number of methods, functions, and variables is returned.

Arguments

`source` A list, associative array, matrix, display box, or namespace.

`Remove(source, position, <n>)`

`Remove(source, {items})`

`Remove(source, key)`

Description

Deletes the *n* item(s), starting from the indicated *position*. If *n* is omitted, the item at *position* is deleted. If *position* and *n* are omitted, the item at the end is removed. For an associative array: Deletes the *key* and its value.

Returns

A copy of the *source* with the items deleted.

Arguments

`source` A quoted string, list, vector, expression, or associative array.

`position or key` An integer (or list of integers) that points to a specific item (or items) in the list or expression.

`n` (Optional) An integer that specifies how many items to remove.

`Remove From(source, position, <n>)`

`Remove From(source, key)`

Description

Deletes the *n* item(s) in place, starting from the indicated *position*. If *n* is omitted, the item at *position* is deleted. If *position* and *n* are omitted, the item at the end is removed. For an associative array: Deletes the *key* and its value. The *source* must be an L-value.

Returns

The original *source* with the items deleted.

Arguments

`source` A quoted string, list, vector, expression, display box, or associative array.

`position or key` An integer (or list of integers) that points to a specific item (or items) in the list or expression.

`n` (Optional) An integer that specifies how many items to remove.

`Reverse(source)`

Description

Reverse order of elements or terms in the *source*.

Argument

`source` A quoted string, list, matrix, or expression.

Reverse Into(source)

Description

Reverses the order of elements or terms in *source* in place.

Argument

source A quoted string, list, matrix, display box, or expression.

Shift(source, <n>)

Description

Shifts an item or *n* items from the front to the back of the *source*.

Arguments

source A quoted string, list, matrix, or expression.

n (Optional) An integer that specifies the number of items to shift. Positive values shift items from the beginning of the *source* to the end. Negative values shift items from the end of the *source* to the beginning. The default value is 1.

Shift Into(source, <n>)

Description

Shifts items in place.

Arguments

source A quoted string, list, matrix, display box, or expression.

n (Optional) An integer that specifies the number of items to shift. Positive values shift items from the beginning of the *source* to the end. Negative values shift items from the end of the *source* to the beginning. The default value is 1.

Sort List({list}|expr)

Description

Sort the elements or terms of *list* or *expr*.

Sort List Into({list}|expr)

Description

Sort the elements or terms of *list* or *expr* in place.

`Substitute(`*`string`*`, "substring", "replacementString", ...)`

`Substitute({list}, listItem, replacementItem, ...)`

`Substitute(Expr(sourceExpr), Expr(findExpr), Expr(replacementExpr), ...)`

Description

This is a search and replace function. It searches for a specific portion (second argument) of the source (first argument), and replaces it (third argument).

If a quoted string, finds all matches to *`substring`* in the source *`string`*, and replaces them with the *`replacementString`*.

If a list, finds all matches to *`listItem`* in the source *`list`*, and replaces them with the *`replacementItem`*.

If an expression, finds all matches to the *`findExpr`* in the *`sourceExpr`*, and replaces them with the *`replacementExpr`*. Note that all expressions must be enclosed within an `Expr()` function.

Arguments

`string, list, sourceExpr` A quoted string, list, matrix, or expression in which to perform the substitution.

`substring, listItem, findExpr` A quoted string, list item, or expression to be found in the source string, list, or expression.

`replacementString, replacementItem, replacementExpr` A quoted string, list item, or expression to replace the found string, list item, or expression.

`Substitute Into(`*`string`*`, substring, replacementString, ...)`

`Substitute Into(list, listItem, replacementItem, ...)`

`Substitute Into(Expr(sourceExpr), Expr(findExpr), Expr(replacementExpr), ...)`

Description

This is a search and replace function, identical to `Substitute()` except in place. It searches for a specific portion (second argument) of the source (first argument), and replaces it (third argument). The first argument must be an L-value.

If a quoted string, finds all matches to *`substring`* in the source *`string`*, and replaces them with the *`replacementString`*.

If a list, finds all matches to *`listItem`* in the source *`list`*, and replaces them with the *`replacementItem`*.

If an expression, finds all matches to the *`findExpr`* in the *`sourceExpr`*, and replaces them with the *`replacementExpr`*. Note that all expressions must be enclosed within an `Expr()` function.

Arguments

`string, list, sourceExpr` A quoted string, list, matrix, or expression in which to perform the substitution.

`substring, listItem, findExpr` A quoted string, list item, or expression to be found in the source string, list, or expression.

`replacementString, replacementItem, replacementExpr` A quoted string, list item, or expression to replace the found string, list item, or expression.

`Words(string, <delimiter>)`

Description

Extracts the words from the quoted *string* according to the delimiters given. The default delimiter is ASCII whitespace. If you include a second argument, any and all characters in that argument are considered delimiters. If *delim* is an empty string, each character is treated as a separate word.

Examples

```
Words( "the quick brown fox" );
   {"the","quick","brown","fox"}
Words( "Doe, Jane P.",", ." );
   {"Doe","Jane","P"}
```

MATLAB Integration Functions

JMP provides the following interfaces to access MATLAB. The basic execution model is to first initialize the MATLAB connection, perform the required MATLAB operations, and then terminate the MATLAB connection. In most cases, these functions return 0 if the MATLAB operation was successful or an error code if it was not. If the MATLAB operation is not successful, a message is written to the Log window. The single exception to this is `MATLAB Get( )`, which returns a value.

MATLAB JSL Function Interfaces

`MATLAB Connect( <named arguments> )`

Description

Initializes the MATLAB integration interfaces and returns an active MATLAB integration interface connection as a scriptable object.

Returns

MATLAB scriptable object.

Named Arguments

`Echo(Boolean)` Sends the MATLAB source lines to the JMP log. The default value is true.

`MATLAB Control( <named arguments> )`

Description

Sends control operations to signal MATLAB with external events such as source line echoing.

Returns

None.

Arguments

None.

Named Arguments

`Echo(Boolean)` Global. Echo MATLAB source lines to the JMP log.

`Visible(Boolean)` Global. Determine whether to show or hide the active MATLAB workspace.

`MATLAB Execute( { list of inputs }, { list of outputs }, mCode, <named arguments> )`

Description

Submits the MATLAB code to the active global MATLAB connection given a list of inputs. Upon completion, retrieves a list of outputs.

Returns

0 if successful, otherwise nonzero.

Arguments

`{ list of inputs }` Positional, name list. List of JMP variable names to send to MATLAB as inputs.

`{ list of outputs }` Positional, name list. List of JMP variable names to retrieve from MATLAB as outputs.

`mCode` Positional, quoted string. The MATLAB code to submit.

Named Arguments

`Expand(Boolean)` Perform an Eval Insert on the MATLAB code prior to submission.

`Echo(Boolean)` Echo MATLAB source lines to the JMP log. Default is true.

Example

The following example sends the JMP variables x and y to MATLAB, executes the MATLAB statement *z = x * y*, and then gets the MATLAB variable z and returns it to JMP.

```
MATLAB Init();
x = [1 2 3];
```

```
y = [4 5 6];
MATLAB Execute( {x, y}, {z}, "z = x * y;" );
Show( z );
```

MATLAB Get(name)

Description

Gets named variable from MATLAB to JMP.

Returns

Value of named variable.

Arguments

`name` Positional. The name of a JMP variable to be sent to MATLAB.

Example

Suppose that a matrix named `qbx` and a structure named `df` are present in your MATLAB connection.

```
// get the MATLAB variable qbx and placed it into a JMP variable qbx
qbx = MATLAB Get( qbx );

/* get the MATLAB variable df and placed it into a JMP data table
referenced by df */
df = MATLAB Get( df );
```

Table 2.1 shows what JMP data types can be exchanged with MATLAB using the `MATLAB Get( )` function. Getting lists from MATLAB recursively examines each element of the list and sends each base MATLAB data type. Nested lists are supported.

Table 2.1 JMP and MATLAB Equivalent Data Types for `MATLAB Get( )`

MATLAB Data Type	JMP Data Type
Double	Numeric
Logical	Numeric (0 | 1)
String	String
Integer	Numeric
Date/Time	Numeric
Structure	Data Table
Matrix	Numeric Matrix
Numeric Vector	Numeric Matrix

Table 2.1 JMP and MATLAB Equivalent Data Types for `MATLAB Get( )` *(Continued)*

MATLAB Data Type	JMP Data Type
String Vector	List of Strings
Graph	Picture object

`MATLAB Get Graphics( format )`

Description

Get the last graphic object written to the MATLAB graph display window in a specific graphic format. The graphic object can be returned in several graphic formats.

Returns

JMP Picture object.

Argument

`format` Positional. The format the MATLAB graph display window contents are to be converted to. Valid formats are `png`, `bmp`, `jpeg`, `jpg`, `tiff`, `tif`, and `gif`.

`MATLAB Get Version`

Description

Returns the version number of MATLAB being used with the JMP MATLAB interfaces.

`MATLAB Init( <named arguments> )`

Description

Initializes the MATLAB integration interfaces.

Returns

Return code.

Named Arguments

`Echo(Boolean)` Sends MATLAB source lines to the JMP log. This option is global. The default value is true.

`MATLAB Is Connected()`

Description

Determines whether a MATLAB connection is active.

Returns

1 if connected, otherwise 0.

MATLAB JMP Name To MATLAB Name(name)

Description

Maps a JMP variable name to its corresponding MATLAB variable name using MATLAB variable name naming rules.

Returns

A quoted string, mapped MATLAB variable name.

Argument

`name` Positional. The name of a JMP variable to be sent to MATLAB.

MATLAB Send(name, <named arguments>)

Description

Sends the named variable from JMP to MATLAB.

Returns

0 if successful, otherwise nonzero.

Arguments

`name` Positional. The name of a JMP variable to be sent to MATLAB.

Named Arguments

The following optional arguments apply to data tables only:

`Selected(Boolean)` Send selected rows from the referenced data table to MATLAB.

`Excluded(Boolean)` Send only excluded rows from the referenced data table to MATLAB.

`Labeled(Boolean)` Send only labeled rows from the referenced data table to MATLAB.

`Hidden(Boolean)` Send only hidden rows from the referenced data table to MATLAB.

`Colored(Boolean)` Send only colored rows from the referenced data table to MATLAB.

`Markered(Boolean)` Send only markered rows from the referenced data table to MATLAB.

`Row States(Boolean, <named arguments>)` Send row states from referenced data table to MATLAB by adding an additional data column named “RowState”. Create multiple selections by adding together individual settings. The row state consists of individual settings with the following values:

- Selected = 1
- Excluded = 2
- Labeled = 4
- Hidden = 8
- Colored = 16
- Markered = 32

The following optional, named Row States arguments are supported:

`Colors(Boolean)` Send row colors. Adds additional data column named "RowStateColor".

`Markers(Boolean)` Send row markers. Adds additional data column named "RowStateMarker".

Example

```
// create a matrix, assign it to X, and send the matrix to MATLAB
X = [1 2 3];
ml = MATLAB Send( X );

/* open a data table, assign a reference to it to dt, and send the
data table along with its current row states to MATLAB */
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
ml = MATLAB Send( dt, Row States( 1 ) );
```

Table 2.2 shows what JMP data types can be exchanged with MATLAB using the `MATLAB Send( )` function. Sending lists to MATLAB recursively examines each element of the list and sends each base JMP data type. Nested lists are supported.

Table 2.2 JMP and MATLAB Equivalent Data Types for `MATLAB Send( )`

MATLAB Data Type	JMP Data Type
Double	Numeric
String	String
Double Matrix	Matrix
Structure	Data Table

Example

```
MATLAB Init( );
X = 1;
MATLAB Send( X );
S = "Report Title";
MATLAB Send( S );
M = [1 2 3, 4 5 6, 7 8 9];
MATLAB Send( M );
MATLAB Submit( "
X
S
M
" );
MATLAB Term( );
```

MATLAB Send File(filename, <MATLAB Name(name)>)

Description

Sends a data file to MATLAB.

Arguments

`filename` Specifies a quoted string that identifies the pathname to the file to be sent to MATLAB.

`MATLAB Name` Enables you to change the name of the file sent to MATLAB.

MATLAB Submit File('pathname', <named arguments>)

Description

Submits statements to MATLAB using a file pointed to by pathname.

Returns

0 if successful, otherwise nonzero.

Arguments

`Pathname` Positional, quoted string. Pathname to file containing MATLAB source lines to be executed.

Named Arguments

`Expand(Boolean)` Perform an Eval Insert on the MATLAB code prior to submission.

`Echo(Boolean)` Echo MATLAB source lines to the JMP log. Default is true.

MATLAB Submit(mCode, <named arguments>)

Description

Submits the MATLAB code to the active global MATLAB connection.

Returns

0 if successful, otherwise nonzero.

Arguments

`mCode` Positional, quoted string. The MATLAB code to submit.

Named Arguments

`Expand(Boolean)` Perform an Eval Insert on the MATLAB code prior to submission.

`Echo(Boolean)` Echo MATLAB source lines to the JMP log. Default is true.

Example

The following example creates two vectors of random points and plots them as x and y variables.

```
MATLAB Init();
mc = MATLAB Submit("/[
   x = rand(5);
   fprintf('%f/n', x);
```

```
    y = rand(5);
    fprintf('%f/n', x);
    z = plot(x, y);
]/" );
```

MATLAB Term();

Description

Terminates the currently active MATLAB integration interface.

Returns

1 if an active MATLAB connection exists, otherwise returns 0.

Arguments

None.

Matrix Functions

All(A, ...)

Returns

1 if all matrix arguments are nonzero; 0 otherwise.

Any(A, ...)

Returns

1 if one or more elements of one or more matrices are nonzero; 0 otherwise.

B Spline Coef(x, Internal Knot Grid, <degree=3>, <Knot End Points=min(x)||max(x)>)

Description

Finds the matrix of B-spline coefficients for the data in the *x* argument.

Returns

The matrix of B-spline basis coefficients. This matrix can be used as a design matrix in a linear model. The first column of the matrix contains an intercept term.

Arguments

x A row or column vector that contains the data.

Internal Knot Grid Either a single number that designates the number of desired knot points based on percentiles of *x* or a vector of values that designate the internal knot points. The number of internal knots must be greater than zero and less than or equal to the number of unique elements in *x* minus two.

`degree` A number that indicates the degree of the B-splines. Defaults to 3.

`Knot End Points` A 2x1 matrix that designates the lower and upper knot points. If this argument is not specified, the default lower and upper knot points are the minimum and maximum values of *x*, respectively.

Notes

This function is used in column formulas created by the Functional Data Explorer platform.

`CDF(Y)`

Description

Returns values of the empirical cumulative probability distribution function for *Y*, which can be a vector or a list. Cumulative probability is the proportion of data values less than or equal to the value of *QuantVec*.

Syntax

```
{QuantVec, CumProbVec} = CDF(YVec)
```

`Chol Update(L, V, C)`

Description

If L is the Cholesky root of an *n*x*n* matrix A, then after calling `cholUpdate` L is replaced with the Cholesky root of A+V*C*V' where C is an *m*x*m* symmetric matrix and V is an n*m matrix.

`Cholesky(A)`

Description

Finds the lower Cholesky root (L) of a positive semi-definitive matrix, L*L' = A.

Returns

L (the Cholesky root).

Arguments

`A` a symmetric matrix.

`Correlation(matrix, < <<"Pairwise">, < <<"Shrink">, < <<Freq(vector)>, < <<Weight(vector)>)`

Description

Calculates the correlation matrix of the data in the *matrix* argument.

Returns

The correlation matrix for the specified matrix.

Argument

`matrix` A matrix that contains the data. If the data has *m* rows and *n* columns, the result is an m-by-m matrix.

`"Pairwise"` Uses the pairwise method for missing values rather than the row-wise method.

`"Shrink"` Performs the Schafer-Strimmer shrinkage estimate.

`<<Freq(vector)` A vector that specifies frequencies for the rows of the matrix argument.

`<<Weight(vector)` A vector that specifies weights for the rows of the matrix argument.

Notes

By default, rows are discarded if they contain missing values. If the "Pairwise" option is specified, all pairs of nonmissing values are used in the correlation matrix calculation.

This function uses multithreading if available, so it is recommended for large problems with many rows.

When a column is constant, the correlations for it are 0, and the diagonal element is also 0.

`Covariance(matrix, < <<"Pairwise">, < <<"Shrink">, < <<Freq(vector)>, < <<Weight(vector)>)`

Description

Calculates the covariance matrix of the data in the *matrix* argument.

Returns

The covariance matrix for the specified matrix.

Argument

`matrix` A matrix that contains the data. If the data has *m* rows and *n* columns, the result is an m-by-n matrix.

`"Pairwise"` Uses the pairwise method for missing values rather than the row-wise method.

`"Shrink"` Performs the Schafer-Strimmer shrinkage estimate.

`<<Freq(vector)` A vector that specifies frequencies for the rows of the matrix argument.

`<<Weight(vector)` A vector that specifies weights for the rows of the matrix argument.

Notes

By default, rows are discarded if they contain missing values. If the "Pairwise" option is specified, all pairs of nonmissing values are used in the covariance matrix calculation.

This function uses multithreading if available, so it is recommended for large problems with many rows.

`Design(vector, < levelsList | <<levels, <<ElseMissing >)`

Description

Creates a design matrix that contains a column of 1s and 0s for each unique value of a *vector* of values.

Returns

A design matrix with a column of 1s and 0s for each unique value of the argument or a list that contains the design matrix and a list of levels.

Argument

`vector` A vector.

`levelsList` An optional list or matrix argument that specifies the levels in the returned matrix.

`<<levels` An optional argument that changes the return value to a list that contains the design matrix and a list of the levels.

`<<ElseMissing` An optional argument that changes the handling of values in the *vector* argument that do not appear in the *levelsList* argument. If this argument is specified, missing values are placed in the design matrix. Otherwise, 0s are placed in the design matrix.

Notes

Missing values in the *levelsList* argument are not ignored. For example:

```
Show( Design ( ., [. 0 1] ),
Design( 0, [. 0 1] ),
Design( 1, [. 0 1] ),
Design( [0 0 1 . 1], [. 0 1] ),
Design( {0, 0, 1, ., 1}, [. 0 1] ) );
   Design(., [. 0 1]) = [1 0 0];
   Design(0, [. 0 1]) = [0 1 0];
   Design(1, [. 0 1]) = [0 0 1];
   Design([0 0 1 . 1], [. 0 1]) =
   [  0 1 0,
      0 1 0,
      0 0 1,
      1 0 0,
      0 0 1];
   Design({0, 0, 1, ., 1}, [. 0 1]) =
   [  0 1 0,
      0 1 0,
      0 0 1,
      1 0 0,
      0 0 1];
```

Design Last(vector, < levelsList, <<ElseMissing >)

Description

Creates a design matrix that contains a column of 1s and 0s for all but the last of the unique values of the argument. The last level is coded as a row of 0s.

Returns

A full-rank design matrix or a list that contains the design matrix and a list of levels.

Arguments

`vector` A vector.

`levelsList` An optional list or matrix argument that specifies the levels in the returned matrix. If this argument is specified, the last level in this list or matrix is treated as the last level in the design matrix. Otherwise, the last level is defined as the largest value in the *`vector`* argument.

`<<ElseMissing` An optional argument that changes the handling of values in the *`vector`* argument that do not appear in the *`levelsList`* argument. If this argument is specified, missing values are placed in the design matrix. Otherwise, 0s are placed in the design matrix.

Design Nom(vector, < levelsList | <<levels, <<ElseMissing >)

DesignF(vector, < levelsList | <<levels, <<ElseMissing >)

Description

Creates a design matrix that contains a column of 1s and 0s for all but the last of the unique values of the argument. The last level is coded as a row of -1s.

Returns

A full-rank design matrix or a list that contains the design matrix and a list of levels.

Argument

`vector` A vector.

`levelsList` An optional list or matrix argument that specifies the levels in the returned matrix. If this argument is specified, the last level in this list or matrix is treated as the last level in the design matrix. Otherwise, the last level is defined as the largest value in the *`vector`* argument.

`<<levels` An optional argument that changes the return value to a list that contains the design matrix and a list of levels.

`<<ElseMissing` An optional argument that changes the handling of values in the *`vector`* argument that do not appear in the *`levelsList`* argument. If this argument is specified, missing values are placed in the design matrix. Otherwise, 0s are placed in the design matrix.

Notes

Missing values in the *`levelsList`* argument are not ignored. For example:

```
Show( Design Nom( ., [. 0 1] ),
Design Nom( 0, [. 0 1] ),
Design Nom( 1, [. 0 1] ),
Design Nom( [0 0 1 . 1], [. 0 1] ),
Design Nom( {0, 0, 1, ., 1}, [. 0 1] ) );
  Design Nom(., [. 0 1]) = [1 0];
  Design Nom(0, [. 0 1]) = [0 1];
  Design Nom(1, [. 0 1]) = [-1 -1];
  Design Nom([0 0 1 . 1], [. 0 1]) = [0 1, 0 1, -1 -1, 1 0, -1 -1];
  Design Nom({0, 0, 1, ., 1}, [. 0 1]) = [0 1, 0 1, -1 -1, 1 0, -1 -1];
```

Design Ord(vector, < levelsList | <<levels, <<ElseMissing >)

Description

Creates a design matrix that contains a column for all but the last of the unique values of the argument. The first level is coded as a row of 0s. Each subsequent (nth) level in the *levelsList* argument is coded as a row of (n-1) 1s and the rest 0s.

Returns

A full-rank design matrix or a list that contains the design matrix and a list of levels.

Argument

`vector` A vector.

`levelsList` An optional list or matrix argument that specifies the levels in the returned matrix.

`<<levels` An optional argument that changes the return value to a list that contains the design matrix and a list of levels.

`<<ElseMissing` An optional argument that changes the handling of values in the *vector* argument that do not appear in the *levelsList* argument. If this argument is specified, missing values are placed in the design matrix. Otherwise, 0s are placed in the design matrix.

Notes

Missing values in the *levelsList* argument are not ignored. For example:

```
Show( Design Ord( ., [. 0 1] ),
Design Ord( 0, [. 0 1] ),
Design Ord( 1, [. 0 1] ),
Design Ord( [0 0 1 . 1], [. 0 1] ),
Design Ord( {0, 0, 1, ., 1}, [. 0 1] ) );
  Design Ord(., [. 0 1]) = [0 0];
  Design Ord(0, [. 0 1]) = [1 0];
  Design Ord(1, [. 0 1]) = [1 1];
  Design Ord([0 0 1 . 1], [. 0 1]) = [1 0, 1 0, 1 1, 0 0, 1 1];
  Design Ord({0, 0, 1, ., 1}, [. 0 1]) = [1 0, 1 0, 1 1, 0 0, 1 1];
```

Det(A)

Description

Determinant of a square matrix.

Returns

The determinant.

Argument

`A` A square matrix.

Diag(A, <B>)

Description

Creates a diagonal matrix from a square matrix or a vector. If two matrices are provided, concatenates the matrices diagonally.

Returns

The matrix.

Argument

`A` a matrix or a vector.

Direct Product(A, B)

Description

Direct (Kronecker) product of square matrices or scalars A[i,j]*B.

Returns

The product.

Arguments

`A, B` Square matrices or scalars.

Distance(x1, x2, <scales>, <powers>)

Description

Produces a matrix of distances between rows of x1 and rows of x2.

Returns

A matrix.

Arguments

`x1, x2` Two matrices.

`scales` Optional argument to customize the scaling of the matrix.

`powers` Optional argument to customize the powers of the matrix.

`E Div(A, B)`

`A:/B`

Returns

A matrix that contains the element-by-element division of two matrices.

Arguments

`A, B` Two matrices. Both matrix elements must have the same dimensions.

`E Max(A, B)`

Returns

A matrix that contains the element-by-element maximum values of two or more matrices or scalar arguments.

Arguments

`A, B` Two or more matrices or scalars. All matrix elements must have the same dimensions.

`E Min(A, B)`

Returns

A matrix that contains the element-by-element minimum values of two or more matrices or scalar arguments.

Arguments

`A, B` Two or more matrices or scalars. All matrix elements must have the same dimensions.

`E Mult(A, B)`

`A:*B`

Description

Element-by-element multiplication of two matrices.

Returns

A matrix that contains the element-by-element multiplication of two or more matrices or scalar arguments.

Arguments

`A, B` Two or more matrices or scalars. All matrix elements must have the same dimensions.

Eigen(A)

Description

Eigenvalue decomposition.

Returns

A list {M, E} such that E * Diag(M) * E = A'.

Argument

A A symmetric matrix.

Estimate Bartlett Factor Score(dataRow, ManMeans, LatMeans, S, A)

Description

Estimates factor scores from a structural equation model (SEM) using Bartlett's method.

Returns

A row vector of estimated factor scores based on the structural equation model.

Arguments

dataRow A row vector of data values.

ManMeans A vector of model-implied manifest variable means.

LatMeans A vector of model-implied latent variable means.

S The symmetric S matrix from a structural equation model.

A The rectangular A matrix from a structural equation model.

Estimate Factor Score(dataRow, Covariance, ManMeans, LatMeans)

Description

Estimates factor scores from a structural equation model (SEM).

Returns

A row vector of estimated factor scores based on the structural equation model.

Arguments

dataRow A row vector of data values.

Covariance A model-implied variance-covariance matrix.

ManMeans A vector of model-implied manifest variable means.

LatMeans A vector of model-implied latent variable means.

Note

This function is used in the Save Factor Scores option in an Structural Equation Model report.

Fourier Basis Coef(x, Number Pairs, <Period=max(x)-min(x)+1>)

Description

Finds the matrix of Fourier basis coefficients for the data in the *x* argument.

Returns

The matrix of Fourier basis coefficients. This can be used as a design matrix in a linear model. The first column of the matrix contains an intercept term. The remaining columns contain pairs of basis coefficients, where pair *i* is defined as the `sin()` and `cos()` of *i* * (2 * π / Period) * x.

Arguments

`x` A row or column vector that contains the data.

`Number Pairs` The number of `sin()` and `cos()` pairs for the Fourier basis.

`Period` The period for trigonometric functions that make up the Fourier basis.

Notes

This function is used in column formulas created by the Functional Data Explorer platform.

G Inverse(A)

Description

Generalized (Moore-Penrose) matrix inverse.

H Direct Product(A, B)

Description

Horizontal direct product of two square matrices of the same dimension or scalars.

Hough Line Transform(matrix, <NAngle(number)>, <NRadius(number)>)

Description

Takes a matrix of intensities and transforms it in a way that is useful for finding streaks in the matrix. Produces a matrix containing the Hough Line Transform with angles as columns and radiuses as rows.

Argument

`matrix` A matrix that can be derived from the intensities of an image, but is more likely from a semiconductor wafer that may have defects across in a streak due to planarization machines.

`NAngle(number)` Enter the number of the angle to obtain a different sized transform. The default is 180 degrees.

`NRadius(number)` Enter the number of the radius to obtain a different sized transform. The default is sqrt(NRow*nRow+nCol*nCol).

Identity(n)

Description

Creates an *n*-by-*n* identity matrix with ones on the diagonal and zeros elsewhere.

Returns

The matrix.

Argument

`n` An integer.

Index(i, j, <increment>)

i::j

Description

Creates a column matrix whose values range from *i* to *j*.

Returns

The matrix.

Arguments

`i, j` Integers that define the range: *i* is the beginning of the range, *j* is the end.

`increment` Optional argument to change the default increment, which is +1.

Inv()

See "Inverse(A)".

Inv Update(A, X, 1|-1)

Description

Efficiently update an **X**′**X** matrix.

Arguments

`A` The matrix to be updated.

`X` One or more rows to be added to or deleted from the matrix *A*.

`1|-1` The third argument controls whether the row or rows defined in the second argument, *X*, are added to or deleted from the matrix *A*. `1` means to add the row or rows and `-1` means to delete the row or rows.

Inverse(A)

Inv(A)

Description

Returns the matrix inverse. The matrix must be square non-singular.

Is Matrix(x)

Description

Returns 1 if the evaluated argument is a matrix, or 0 otherwise.

J(nrows, <ncols>, <value>)

Description

Creates a matrix of identical values.

Returns

The matrix.

Arguments

`nrows` Number of rows in matrix. If `ncols` is not specified, `nrows` is also used as `ncols`.

`ncols` Number of columns in matrix.

`value` The value used to populate the matrix. If `value` is not specified, 1 is used.

KDTable(matrix)

Description

Returns a table to efficiently look up near neighbors.

Returns

A KDTable object.

Argument

`matrix` A matrix of k-dimensional points. The number of dimensions or points is not limited. Each column in the matrix represents a dimension to the data, and each row represents a data point.

Messages

`<<Distance between rows(row1, row2)` Returns the distance between two the two specified rows in the KDTable. The distance applies to removed and inserted rows as well.

`<<K nearest rows(stop, <position>)` Returns a matrix. Position is a point that is described as a row vector for the coordinate of a row, or as the number of a row. If *`position`* is not specified, returns the *n* nearest rows and distances to all rows. If *`position`* is specified, returns the *n* nearest rows and distances to either a point or a row. *`Stop`* is either *n* or {*n*, *limit*}. The limit parameter limits the number of rows that will be found. It can be specified one of two ways: a number, like 5, means return the 5 nearest rows. A list, like {5,10}, means return up to 5 nearest rows, stopping when the distance of 10 is exceeded. In the second case, the last row may have a distance greater than 10. Since the command continues until it finds the closest row beyond the stop radius, this point is also returned. This can be especially useful if there are no rows within the radius.

<<Remove rows(number | vector) Remove either the row specified by *number* or the rows specified by *vector*. Returns the number of rows that were removed. Rows that were already removed are ignored.

<<Insert rows(number | vector) Re-insert either the row specified by *number* or the rows specified by *vector*. Returns the number of rows that were inserted. Rows that were already inserted are ignored.

Notes

When rows are removed or inserted, the row indices do not change. You can remove and re-insert only rows that are in the KDTable object. If you need different rows, construct a new KDTable.

`Least Squares Solve(y, X, < <<noIntercept, <<weights(OptionalWeightVector), <<method("Sweep"|"GInv")>)`

Description

Computes least squares regression estimates for the assumed model y = X * beta + error.

Returns

A list that contains the matrix `Beta=Inverse(X'X)X'y` and the estimated variance matrix of Beta.

Optional Named Arguments

<<noIntercept Specifies a no-intercept model.

<<weights(optional weight vector) Specifies a vector of weights to perform weighted least squares.

<<method("Sweep" | "GInv") Specifies the method for solving the normal equations. The default Sweep method is more computationally efficient, but you can also specify the generalized inverse ("GInv") method, which is more numerically stable.

`Linear Regression(y, X, < <<noIntercept, <<printToLog, <<weight(OptionalWeightVector), <<freq(OptionalFreqVector)>)`

Description

Fits a linear regression for the assumed model y = X * beta + error.

Returns

A list that contains a vector of the estimates, a vector of the standard error, and a list of diagnostics. The list of diagnostics contains vectors of the *t* statistics and *p*-values for the estimates, as well as the R-Square and adjusted R-Square values for the regression fit.

Optional Named Arguments

<<noIntercept Excludes the intercept.

<<printToLog Prints a summary of the fit to the log.

<<weight(vector) Specifies a vector of weights to perform weighted least squares.

<<freq(vector) Specifies a vector of frequencies for each row of y and X.

Example

```
n = 10;
x = J( n, 1, Random Normal() );
y = 1 + x * 3 + J( n, 1, Random Normal() );
{Estimates, Std_Error, Diagnostics} = Linear Regression( y, x, <<printToLog );
As Table( y || x );
Bivariate( Y( :Col1 ), X( :Col2 ), Fit Line( 1 ) );
```

`Loc(A)`

`Loc(A, item)`

Description

Returns a matrix of subscript positions where *A* is nonzero and nonmissing. For the two-argument function, Loc returns a matrix of positions where *item* is found within *A*. If the first argument is a list, the second argument is required.

Argument

`A` a matrix or a list

`item` the item to be found within the matrix or list *A*

`Loc Max(A)`

Description

Returns the position of the maximum element in a matrix.

Returns

An integer that is the specified position.

Argument

`A` a matrix

`Loc Min(A)`

Description

Returns the position of the minimum element in a matrix.

Returns

An integer that is the specified position.

Argument

`A` a matrix

`Loc NonMissing(matrix, ..., {list}, ...)`

Description

Returns indices of nonmissing rows in matrices or lists. In lists, the function can also return indices of nonempty characters.

Returns

The new matrix or list.

`Loc Sorted(A, B)`

Description

Returns a column vector of subscript positions where the values of *A* have values less than or equal to the values in *B* based on a binary search. *A* must be a matrix sorted in ascending order without missing values.

Returns

The new matrix, which has the same dimensions as *B*. If a value in *B* is less than the first value in *A*, the returned subscript position for that value is 1.

Argument

`A, B` matrices

`Matrix({{x11, ..., x1m}, {x21, ..., 2m}, {...}, {xn1, ..., xnm}})`

`Matrix({x1, ..., xn})`

`Matrix(n, m)`

Description

Constructs an *n*-by-*m* matrix. The following specification methods are available:

- If you specify a list of *n* lists that each contain *m* row values, the matrix is formed by vertically concatenating the evaluated lists. The list items must evaluate to numeric values or row vectors, and the dimensions of the items must be conformable.
- If you specify a single list of *n* items, the return value is an *n*-by-1 column vector. The items of the evaluated list must evaluate to numeric values.
- If you specify two integer arguments, the return value is a matrix of zeros that contains *n* rows and *m* columns.

Examples

```
Matrix({{1, 2, 3}, {4, [5 6]}, {7, 8, 9}});
  [1 2 3, 4 5 6, 7 8 9]
Matrix({{[1 2 3], 4, 5, 6, 7, 8, 9}});
  [1 2 3 4 5 6 7 8 9]
Matrix({2,3+7});
  [2, 10]
Matrix(2,3);
  [0 0 0, 0 0 0]
```

Matrix Mult(A, B)

C=A*B, ...

Description

Matrix multiplication.

Arguments

A, B, ... Two or more matrices, which must be conformable (all matrices after the first one listed must have the same number of rows as the number of columns in the first matrix).

Notes

Matrix Mult() allows only two arguments, while using the * operator enables you to multiply several matrices.

Matrix Rank(A)

Description

Returns the rank of the matrix A.

Mode({list} or matrix)

Description

Selects the most frequent item from a numeric or character list or a numeric matrix. In the event of a tie, the lower value is selected. If multiple arguments are specified, a combination of numeric values and quoted character strings is acceptable.

Arguments

Specify either a list or a matrix.

Multivariate Normal Impute(yVec, meanYvec, symCovMat, colMin, colMax)

Description

Imputes missing values in yVec based on the mean and covariance.

Arguments

yVec The vector of responses.

meanYvec The vector of response means.

symCovMat A symmetric matrix containing the response covariances. If the covariance matrix is not specified, then JMP imputes with means.

colMin A vector of column minimums. Provides lower bounds for the imputations.

colMax A vector of column maximums. Provides upper bounds for the imputations.

`NChooseK Matrix(n, k)`

Description

Returns a matrix of *n* things taken *k* at a time (*n* select *k*).

`N Col(x)`

`N Cols(x)`

Description

Returns the number of columns in either a data table or a matrix.

Argument

x Can be a data table or a matrix.

`Ortho(A, <Centered(0)>, <Scaled(1)>)`

Description

Orthonormalizes the columns of matrix *A* using the Gram Schmidt method. `Centered(0)` makes the columns to sum to zero. `Scaled(1)` makes them unit length.

`Ortho Poly(vector, order)`

Description

Returns orthogonal polynomials for a *vector* of indices representing spacings up to the *order* given.

`P Spline Coef(x, Internal Knot Grid, <degree=3>)`

Description

Finds the matrix of penalized basis spline (P-spline) coefficients for the data in the *x* argument.

Returns

The matrix of P-spline basis coefficients, which is the truncated power basis of the specified *degree*. The truncated power basis of degree p with knots k_1 through k_K is defined as follows:

$$1, x, x^2, \ldots, x^p, (x - k_1)_+^p, \ldots, (x - k_K)_+^p$$

where $(x - k_1)_+$ is the positive part of $(x - k_1)$ and is set to zero for negative values of $(x - k_1)$.

Arguments

x A row or column vector that contains the data.

`Internal Knot Grid` Either a single number that designates the number of desired knot points based on percentiles of *x* or a vector of values that designate the internal knot points.

`degree` A number that indicates the degree of the P-splines. Defaults to 3.

Notes

This function is used in column formulas created by the Functional Data Explorer platform.

`Parallel Assign({thread_local_var = global_var, ...}, matrix[a, b] = expression using a and b)`

Description

Uses multiple threads to assign values to the matrix. Enables you to take advantage of multiple cores on a computer. The function has two arguments.

- The first argument is a list of assignment statements that copies global variables into each thread's local variable list.
- The second argument is an assignment expression with a left-hand-side that is a matrix with one or two prototype indexes and a right hand side that can be any JSL expression using those indexes and the local variables from the list (and in JMP global: variables).

Example

The following example provides read access to the global namespace.

```
a = 42;
x = [1 2 3, 4 5 6, 7 8 9];
Show( Parallel Assign( {}, x[i, j] = global:a ) );
Show( x );
   Parallel Assign({}, x[i,j] = global:a) = 1;
   x =
   [  42 42 42,
      42 42 42,
      42 42 42];
```

`Print Matrix(M, <named arguments>)`

Description

Returns a quoted string that contains a well-formatted matrix. You can use the function, for example, to print the matrix to the log.

Argument

`M` A matrix.

Optional Named Arguments

`<<ignore locale(Boolean)` Set to false (0) to use the decimal separator for your locale. Set to true (1) to always use a period (`.`) as a separator. The default value is false (0).

`<<decimal digits(n)` An integer that specifies the number of digits after the decimal separator to print.

`<<style("style name")` Use one of three available styles: Parseable is a reformatted JSL matrix expression. Latex is formatted for LaTex. If you specify Other, you must define the following three arguments.

`<<separate("character")` Define the separator for concatenated entries.

`<<line begin("character")` Define the beginning line character.

`<<line end("character")` Define the ending line character.

`QR(A)`

Description

Returns the QR decomposition of *A*. Typical usage is `{Q, R} = QR(A)`.

`Rank Index(vector)`

`Rank(vector)`

Description

Returns a vector of indices that, used as a subscript to the original *vector*, sorts the vector by rank. Excludes missing values. Lists of numbers or quoted strings are supported in addition to matrices.

`Ranking(vector)`

Description

Returns a vector of ranks of the values of *vector*, low to high as 1 to *n*, ties arbitrary. Lists of numbers or quoted strings are supported in addition to matrices.

`Ranking Tie(vector)`

Description

Returns a vector of ranks of the values of *vector*, but ranks for ties are averaged. Lists of numbers or quoted strings are supported in addition to matrices.

`Scoring Impute(rowWithMissing, VMat, colMeanVec, colStdDevVec)`

Description

Provides streaming functionality for the Automated Data Imputation (ADI) algorithm.

Returns

Returns the row vector with the missing values imputed using the standard least squares estimation.

Arguments

`rowwithMissing` A row vector that contains missing values.

`VMat` A loading matrix that is produced by the ADI algorithm.

`colMeanVec` A vector of the column means ignoring missing cells.

`colStdDevVec` a vector of the column standard deviations ignoring missing cells.

`Shape(A, nrow, <ncol>, < <<bycol>)`

Description

Reshapes the matrix *A* across rows to the specified dimensions. Each value from the matrix *A* is placed into the reshaped matrix. By default, the values are placed row-by-row.

Returns

The reshaped matrix.

Arguments

`A` A matrix.

`nrow` The number of rows that the new matrix should have.

`ncol` (Optional) The number of columns the new matrix should have.

`<<bycol` (Optional) Specifies that the values be placed into the reshaped matrix column-by-column, instead of row-by-row.

Notes

If *ncol* is not specified, the number of columns is whatever is necessary to fit all of the original values of the matrix into the reshaped matrix.

If a missing value is specified for *nrow*, the number of rows is whatever is necessary to fit all of the original values of the matrix into the reshaped matrix.

If the new matrix is smaller than the original matrix, the extra values are discarded.

If the new matrix is larger than the original matrix, the values are repeated to fill the new matrix.

Examples

```
a = Matrix({ {1, 2, 3}, {4, 5, 6}, {7, 8, 9} });
   [  1 2 3,
      4 5 6,
      7 8 9]

Shape(a, 2);
   [  1 2 3 4 5,
      6 7 8 9 1]

Shape(a, 2, 2);
   [  1 2,
      3 4]

Shape(a, 4, 4);
```

```
[  1 2 3 4,
   5 6 7 8,
   9 1 2 3,
   4 5 6 7]

Shape(a, 4, 4, <<bycol);
[  1 5 9 4,
   2 6 1 5,
   3 7 2 6,
   4 8 3 7]
```

Solve(A, b)

Description

Solves a linear system. In other words, `x=inverse(A)*b`.

Sort Ascending(source)

Description

Returns a copy of a list or matrix *source* with the items in ascending order.

Sort Descending(source)

Description

Returns a copy of a list or matrix *source* with the items in descending order.

Sparse SVD(X, <nSingularValues=min(nRow, nCol)>,<tolerance=1e-10>)

Description

Computes the singular value decomposition of matrix X using the implicitly restarted, partially reorthogonalized Lanczos method for sparse matrices.

Returns

Returns a list (U, M, V) such that U*diag(M)*V is equal to x.

Spline Coef(x, y, lambda)

Description

Returns a five column matrix of the form `knots||a||b||c||d` where `knots` is the unique values in *x*.

`x` is a vector of regressor variables, `y` is the vector of response variables, and `lambda` is the smoothing argument. Larger values for `lambda` result in smoother splines.

Spline Eval(x, coef)

Description

Evaluates the spline predictions using the `coef` matrix in the same form as returned by `SplineCoef()`, in other words, `knots||a||b||c||d`. The `x` argument can be a scalar or a matrix of values to predict. The number of columns of `coef` can be any number greater than 1 and each is used for the next higher power. The powers of `x` are centered at the knot values. For example, the calculation for `coef` of `knots||a||b||c||d` is `j` is such that `knots[j]` is the largest knot smaller than `x`.

`xx = x-knots[j]` is the centered *x* value:

```
result = a[j] + xx * (b[j] + xx * (c[j] + xx * d[j]))
```

The following line is equivalent:

```
result = a[j] + b[j] * xx + c[j] * xx ^ 2 + d[j] * xx ^ 3
```

Spline Smooth(x, y, lambda)

Description

Returns the smoothed predicted values from a spline fit.

`x` is a vector of regressor variables, `y` is the vector of response variables, and `lambda` is the smoothing argument. Larger values for `lambda` result in smoother splines.

SVD(A)

Description

Singular value decomposition.

Sweep(A, <indices>)

Description

Sweeps, or inverts a matrix a partition at a time.

Trace(A)

Description

The trace, or the sum of the diagonal elements of a square matrix.

Transpose(A)

Description

Transposes the rows and columns of the matrix *A*.

Returns

The transposed matrix.

Arguments

A A matrix.

Equivalent Expression

A‘

`V Concat(A, B, ...)`

Description

Vertical concatenation of two or more matrices.

Returns

The new matrix.

Arguments

Two or more matrices.

`V Concat To(A, B, ...)`

Description

Vertical concatenation in place. This is an assignment operator.

Returns

The new matrix.

Arguments

Two or more matrices.

`V Max(matrix)`

Description

Returns a row vector containing the maximum of each column of *matrix*.

`V Mean(matrix)`

Description

Returns a row vector containing the mean of each column of *matrix*.

`V Median(matrix)`

Description

Returns a row vector containing the median of each column of *matrix*.

`V Min(matrix)`

Description

Returns a row vector containing the minimum of each column of *matrix*.

V Quantile(matrix, p)

Description

Returns a row vector containing the p^{th} quantile of each column of *matrix*.

V Standardize(matrix)

Description

Returns a matrix column-standardized to mean = 0 and standard deviation = 1.

V Std(matrix)

Description

Returns a row vector containing the standard deviations of each column of *matrix*.

V Sum(matrix)

Description

Returns a row vector containing the sum of each column of *matrix*.

Varimax(matrix, <norm=1>)

Description

Performs a varimax rotation.

Returns

A list that contains the rotated matrix and the orthogonal rotation matrix.

Arguments

matrix A matrix to be rotated.

norm Specify 1 to perform a normalized rotation, and specify 0 to perform a non-normalized rotation. The default value is 1.

Vec Diag(A)

Description

Creates a vector from the diagonals of a square matrix *A*.

Returns

The new matrix.

Arguments

A A square matrix.

Notes

Using a matrix that is not square results in an error.

`Vec Quadratic(symmetric matrix, rectangular matrix)`

Description

Constructs an *n*-by-*m* matrix. Used in calculation of hat values.

Returns

The new matrix.

Arguments

Two matrices. The first must be symmetric.

Equivalent Expression

`Vec Diag(X*Sym*X')`

`VPTree(matrix)`

Description

Returns a table that is used for efficiently looking up nearest neighbors. The vantage-point tree algorithm is particularly useful for wide data problems.

Returns

A VPTree object.

Argument

`matrix` A matrix of *k*-dimensional points. The number of dimensions or points is not limited. Each column in the matrix represents a dimension to the data, and each row represents a data point.

Numeric Functions

`Abs(n)`

Description

Calculates the absolute value of *n*.

Returns

Returns a positive number of the same magnitude as the value of *n*.

Argument

`n` Any number.

`Ceiling(n)`

Description

If *n* is not an integer, rounds *n* to the next highest integer.

Returns

Returns the smallest integer greater than or equal to *n*.

Argument

`n` Any number.

`Derivative(expr, {name, ...}, ...)`

Description

Calculates the derivative of the *expr* expression with respect to *name*.

Returns

Returns the derivative.

Arguments

`expr` Any expression. Indirect arguments (for example, `Name Expr`, `Expr`, `Eval`) are supported.

`name` Can be a single variable or a list of variables.

Notes

Adding an additional variable (`Derivative(expr, name, name2)`) takes the second derivative.

`Floor(n)`

Description

If *n* is not an integer, rounds *n* to the next lowest integer.

Returns

Returns the largest integer less than or equal to *n*.

Argument

`n` Any number.

Examples

```
Floor( 2.7 );
  2
Floor( -.5 );
  -1
```

`Integrate(expr, varname, lowLimit, upLimit, <<Tolerance(1e-10), <<StoreInfo({list}), <<StartingValue(val))`

Description

Integrates an expression with respect to a scalar value, using the adaptive quadrature method from Gander and Gautschi (2000).

Arguments

`expr` an expression that defines the integrand.

`varname` the name of the variable of integration. If this variable contains a value, that value specifies a starting value that is used as a typical value to improve the accuracy of the integral.

`lowLimit` specifies the lower limit of integration. To specify negative infinity as the lower limit of integration, set this to missing.

`upLimit` specifies the upper limit of integration. To specify positive infinity as the upper limit of integration, set this to missing.

`StoreInfo` saves diagnostics of the numerical integration routine to the argument of StoreInfo().

`StartingValue` specifies a starting value that is used as a typical value to improve the accuracy of the integral.

`Invert Expr(expr, name)`

Description

Attempts to unfold *`expr`* around *`name`*.

`Mod()`

See "Modulo(number, divisor)"

`Modulo(number, divisor)`

`Mod(number, divisor)`

Description

Returns the remainder when *`number`* is divided by *`divisor`*.

Examples

```
Modulo( 6, 5 );
  1
```

`Normal Integrate(muVector, sigmaMatrix, expr, x, nStrata, nSim)`

Description

Returns the result of radial-spherical integration for smooth functions of multivariate, normally distributed variables.

Arguments

`muVector` A vector.

`sigmaMatrix` A matrix.

`expr` An expression in terms of the variable *`x`*.

`x` The variable used in the expression *`expr`*.

`nStrata` Number of strata.

nSim Number of simulations.

Num Deriv(f(x,...), <parnum=1>)

Description

Returns the numerical derivative of the *f(x,...)* function with respect to one of its arguments. You can specify that argument as the second argument in the *Num Deriv* function. If no second argument is specified, the derivative is taken with respect to the function's first argument. The derivative is evaluated using numeric values specified in the *f(x,...)* function expression.

Notes

The Num Deriv() function might appear not to produce the correct results as seen here:

```
x = 3;
n = Num Deriv( 3 * x ^ 2 );
    // 9.00000000001455
```

The preceding usage is not correct. The function was designed to be used in the Nonlinear platform to differentiate functions for which it does not know the analytic derivatives. The proper usage takes the following form:

```
x = 3;
f = Function( {x}, 3 * x ^ 2 );
n = Num Deriv( f( x ), 1 );
   // 18.000029999854
```

Num Deriv2(f(x,...))

Description

Returns the numerical second derivative of the *f(x,...)* function with respect to *x*. The derivative is evaluated using numeric values specified in the *f(x,...)* function expression.

Round(n, places)

Description

Rounds *n* to number of decimal *places* given.

Simplify Expr(expr(expression))

Simplify Expr(nameExpr(global))

Description

Algebraically simplifies an expression

Optimization Functions

Constrained Maximize(expr, {x1(low1, up1), x2(low2, up2), ...}, messages)

Description

Finds the values for the *x* arguments, specified as a list, that maximize the *expr* expression with optional linear constraints. You must either specify lower and upper bounds in parentheses for each argument or with the optional `Set Variable Limit()` message. The *x* arguments can be scalar values or vectors.

In the following messages, `A` is a matrix of coefficients. `x = [x1, x2, ...]` is the vector of arguments. `b` is a vector that forms the right side of the expression.

Messages

`<<Less than EQ({A, b})` Sets the constraint to less than or equal to the specified values `(A*x <= b)`.

`<<Greater Than EQ({A, b})` Sets the constraint to greater than or equal to the specified values `(A*x >= b)`.

`<<Equal To({A, b})` Sets the constraint as equal to the specified values `(A*x = b)`.

`<<Starting Values([x1Start, x2Start, ...])` Specifies a starting point.

`<<Max Iter(int)` An integer that specifies the maximum number of iterations to be performed.

`<<Tolerance(p)` *p* sets the tolerance for the convergence criterion. The default tolerance is 10^{-5}.

`<<Show Details("true")` Returns a list with the final values for (objective value, number of iterations, gradient, and Hessian). Shows the step-by-step results of the optimizer in the log.

`<<SetVariableLimit({low,high})` Specifies vectors for the lower and upper limits for the optimization variables.

Constrained Minimize(expr, {x1(low1, up1), x2(low2, up2), ...}, messages)

Description

Finds the values for the *x* arguments, specified as a list, that minimize the *expr* expression with optional linear constraints. You must either specify lower and upper bounds in parentheses for each argument or with the optional `Set Variable Limit()` message. The *x* arguments can be scalar values or vectors.

In the following messages, `A` is a matrix of coefficients. `x = [x1, x2, ...]` is the vector of arguments. `b` is a vector that forms the right side of the expression.

Messages

`<<Less than EQ({A, b})` Sets the constraint to less than or equal to the specified values `(A*x <= b)`.

<<Greater Than EQ({A, b}) Sets the constraint to greater than or equal to the specified values (A*x >= b).

<<Equal To({A, b}) Sets the constraint as equal to the specified values (A*x = b).

<<Starting Values([x1Start, x2Start, ...]) Specifies a starting point.

<<Max Iter(int) An integer that specifies the maximum number of iterations to be performed.

<<Tolerance(p) *p* sets the tolerance for the convergence criterion. The default tolerance is 10^{-5}.

<<Show Details("true") Returns a list with the final values for (objective value, number of iterations, gradient, and Hessian). Shows the step-by-step results of the optimizer in the log.

<<SetVariableLimit({low,high}) Specifies vectors for the lower and upper limits for the optimization variables.

Desirability(yVector, desireVector, y)

Description

Fits a function to go through the three points, suitable for defining the desirability of a set of response variables (*y*s). *yVector* and *desireVector* are matrices with three values, corresponding to the three points defining the desirability function. The actual function depends on whether the desire values are in the shape of a larger-is-better, smaller-is-better, target, or antitarget.

Returns

The desirability function.

Arguments

yVector Three input values.

desireVector the corresponding three desirability values.

y the value of which to calculate the desirability.

LPSolve(A, b, c, L, U, neq, nle, nge, <slackVars(Boolean)>)

Description

Returns a list containing the decision variables (and slack variables if applicable) in the first list item and the optimal objective function value (if one exists) in the second list item.

Arguments

A A matrix of constraint coefficients.

b A matrix that is a column of right hand side values of the constraints.

c A vector of cost coefficients of the objective function.

L, U Matrices of lower and upper bounds for the variables.

neq The number of equality constraints.

`nle` The number of less than or equal inequalities.

`nge` The number of greater than or equal inequalities.

`slackVars(Boolean)` (Optional) Determines whether the slack variables are returned in addition to the decision variables. The default value is `0`.

Notes

The constraints must be listed as equalities first, less than or equal inequalities next, and greater than or equal inequalities last.

Maximize(expr, {x1(low1, up1), x2(low2, up2), ...}, messages)

Description

Finds the values for the *x* arguments, specified as a list, that maximize the expression *expr*. You can specify lower and upper bounds in parentheses for each argument. Additional arguments for the function enable you to set the maximum number of iterations, tolerance for convergence, and view more details about the optimization. The Newton-Raphson method is used when an analytical derivative is found for the Hessian. Otherwise, the Symmetric-Rank One method (SR1), a quasi-Newton method, is used.

Messages

`<<Max Iter(int)` An integer that specifies the maximum number of iterations to be performed. The default maximum number of iterations is 250.

`<<Tolerance(p)` *p* sets the tolerance for the convergence criterion. The default tolerance is 10^{-8}.

`<<Details("both" | "displaySteps" | "returnDetails")` Specifies what output is returned. If "displaySteps" is specified, step-by-step results of the optimization appear in the Log window. If "returnDetails" is specified, the function returns a list that contains the final values for the objective value, number of iterations, gradient, and Hessian. Specify "both" to get the return value and the results in the Log.

`<<Gradient(exprList)` Specifies a list of expressions that define the analytical gradient that is used for the optimization. Each expression in the list represents a derivative of the expression *expr*.

`<<Hessian(exprList)` Specifies a list of expressions that define the analytical Hessian that is used for the optimization. Each expression in the list represents the upper triangular portion of the Hessian matrix in row-major order.

`<<Method(NR | SR1)` Specifies either the Newton-Raphson (NR) method or the Symmetric-Rank One (SR1) method for the optimization method.

`<<UseNumericDeriv("true")` Specifies that the optimization use a numeric approximation.

`Minimize(expr, {x1(low1, up1), x2(low2, up2), ...}, messages)`

Description

Finds the values for the *x* arguments, specified as a list, that minimize the expression *expr*. You can specify lower and upper bounds in parentheses for each argument. Additional arguments for the function enable you to set the maximum number of iterations, tolerance for convergence, and view more details about the optimization. The Newton-Raphson method is used when an analytical derivative is found for the Hessian. Otherwise, the Symmetric-Rank One method (SR1), a quasi-Newton method, is used.

Messages

`<<Max Iter(int)` An integer that specifies the maximum number of iterations to be performed. The default maximum number of iterations is 250.

`<<Tolerance(p)` *p* sets the tolerance for the convergence criterion. The default tolerance is 10^{-8}.

`<<Details("both" | "displaySteps" | "returnDetails")` Specifies what output is returned. If "displaySteps" is specified, step-by-step results of the optimization appear in the Log window. If "returnDetails" is specified, the function returns a list that contains the final values for the objective value, number of iterations, gradient, and Hessian. Specify "both" to get the return value and the results in the Log.

`<<Gradient(exprList)` Specifies a list of expressions that define the analytical gradient that is used for the optimization. Each expression in the list represents a derivative of the expression *expr*.

`<<Hessian(exprList)` Specifies a list of expressions that define the analytical Hessian that is used for the optimization. Each expression in the list represents the upper triangular portion of the Hessian matrix in row-major order.

`<<Method(NR | SR1)` Specifies either the Newton-Raphson (NR) method or the Symmetric-Rank One (SR1) method for the optimization method.

`<<UseNumericDeriv("true")` Specifies that the optimization use a numeric approximation.

Probability Functions

`Beta Density(x, alpha, beta, <theta=0>, <sigma=1>)`

Description

Returns the probability density function (pdf) evaluated at *x* of the beta distribution. The pdf is parameterized as follows:

$$f(x) = \frac{1}{B(\alpha, \beta)\sigma^{\alpha+\beta-1}}(x-\theta)^{\alpha-1}(\theta+\sigma-x)^{\beta-1}$$

where B(·) is the Beta function.

Arguments

x A quantile at which the pdf is evaluated. *x* must be between *theta* and *theta* + *sigma*.

alpha, beta Shape parameters α and β, which must both be greater than 0.

theta Optional threshold parameter θ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Notes

The beta distribution is useful for modeling the probabilistic behavior of random variables that are constrained to fall in the interval [0, 1], such as proportions.

`Beta Distribution(x, alpha, beta, <theta=0>, <sigma=1>)`

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the beta distribution. The cdf uses the same parameterization as the `Beta Density()` function.

Arguments

x A quantile at which the cdf is evaluated. *x* must be between *theta* and *theta* + *sigma*.

alpha, beta Shape parameters α and β, which must both be greater than 0.

theta Optional threshold parameter θ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

`Beta Quantile(p, alpha, beta, <theta=0>, <sigma=1>)`

Description

Returns the p^{th} quantile from a beta distribution with shape arguments *alpha* and *beta*. The quantile function does not have a closed form equation.

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

alpha, beta Shape parameters α and β, which must both be greater than 0.

theta Optional threshold parameter θ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

`Cauchy Density(q, <center=0>, <scale=1>)`

Description

Returns the probability density function (pdf) evaluated at *q* of a Cauchy distribution. The pdf is parameterized as follows:

$$f(q) = \frac{1}{\pi\sigma}\frac{1}{1+\left(\frac{q-\mu}{\sigma}\right)^2}$$

Arguments

`q` A quantile at which the pdf is evaluated.

`center` Optional location parameter μ. The default is 0.

`scale` Optional scale parameter, σ, which must be greater than 0. The default is 1.

Cauchy Distribution(q, <center=0>, <scale=1>)

Description

Returns the cumulative distribution function (cdf) probability that a Cauchy distributed random variable is less than *q*. The cdf is parameterized as follows:

$$F(q) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(\frac{x-\mu}{\sigma}\right)$$

Arguments

`q` A quantile at which the cdf is evaluated.

`center` Optional location parameter μ. The default is 0.

`scale` Optional scale parameter, σ, which must be greater than 0. The default is 1.

Cauchy Quantile(p, <center=0>, <scale=1>)

Description

Returns the p^{th} quantile from a Cauchy distribution. The p^{th} quantile is the value for which the probability is *p* that a random value would be less than or equal to *p*. The quantile function is parameterized as follows:

$$F^{-1}(p) = \sigma\tan\left[\pi\left(p + \frac{1}{2}\right)\right] + \mu$$

Arguments

`p` The probability of the quantile desired. *p* must be between 0 and 1.

`center` Optional location parameter μ. The default is 0.

`scale` Optional scale parameter σ, which must be greater than 0. The default is 1.

ChiSquare Density(q, df, <nc=0>)

Description

Returns the probability density function (pdf) evaluated at *q* of the chi-square distribution. The pdf is parameterized as follows:

$$f(q) = \exp(-\lambda/2)\sum_{r=0}^{\infty}\frac{(\lambda/2)^r}{r!}f_{n+2r}(q)$$

where $f_{n+2r}(q)$ is the density of a central chi-square distribution with $n+2r$ degrees of freedom.

Arguments

q A quantile at which the pdf is evaluated. *q* must be greater than or equal to 0.

df The degrees of freedom *n,* which must be greater than 0.

nc Optional noncentrality parameter λ, which must be nonnegative. The default is 0.

ChiSquare Distribution(q, df, <nc=0>)

Description

Returns cumulative distribution function at quantile *x* for chi-square with *df* degrees of freedom centered at *nc*. The cdf is parameterized as follows:

$$F(q) = \exp(-\lambda/2) \sum_{r=0}^{\infty} \frac{(\lambda/2)^r}{r!} F_{n+2r}(q),$$

where $F_{n+2r}(q)$ is the cumulative distribution of a central chi-square distribution with $n+2r$ degrees of freedom.

Arguments

q A quantile at which the cdf is evaluated. *q* must be greater than or equal to 0.

df The degrees of freedom *n,* must be greater than 0.

nc The optional noncentrality parameter λ, must be nonnegative. The default is 0.

ChiSquare Log CDistribution(x, df, <nc=0>)

Description

Returns the log of (1 - value), where value is the cumulative distribution function evaluated at *x* of the chi-square distribution with *df* degrees of freedom and noncentrality parameter *nc*.

ChiSquare Log Density(x, df, <nc=0>)

Description

Returns the log of the value of the probability density function evaluated at *x* of the chi-square distribution with *df* degrees of freedom and noncentrality parameter *nc*.

ChiSquare Log Distribution(x, df, <nc=0>)

Description

Returns the log of the value of the cumulative distribution function evaluated at quantile *x* of the chi-square distribution with *df* degrees of freedom and noncentrality parameter *nc*.

ChiSquare Noncentrality(x, df, prob)

Description

Returns the chi-square distribution noncentrality parameter *nc* that satisfies the following:

prob = ChiSquare Distribution(*x, df, nc*)

Arguments

x A quantile at which the cdf is evaluated.

df The degrees of freedom *n*, which must be greater than 0.

prob The probability of the quantile desired; *prob* must be between 0 and 1.

ChiSquare Quantile(p, df, <nc=0>)

Description

Returns the p^{th} quantile from a chi-square distribution with *df* degrees of freedom, centered at *nc*. The quantile function does not have a closed form equation.

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

df The degrees of freedom *n*, which must be greater than 0.

nc Optional noncentrality parameter λ, which must be nonnegative. The default is 0.

Dunnett P Value(q, nTrt, dfe, <lambdaVec=.>)

Description

Returns the *p*-value from Dunnett's multiple comparisons test.

Arguments

q A number that is the test statistic.

nTrt The number of treatments being compared to the control treatment.

dfe The error degrees of freedom.

lambdaVec A vector of parameters. If *lambdaVec* is missing (.), each of the parameters is set to 1/Sqrt(2).

Dunnett Quantile(1-alpha, nTrt, dfe, <lambdaVec=.>)

Description

Returns the quantile used in Dunnett's multiple comparisons test.

Arguments

1-alpha A number that is the confidence level.

nTrt The number of treatments being compared to the control treatment.

dfe The error degrees of freedom.

lambdaVec A vector of parameters. If *lambdaVec* is missing (.), each of the parameters is set to 1/Sqrt(2).

Exp Density(x, <theta=1>)

Description

Returns the probability density function (pdf) evaluated at *x* of the exponential distribution. The pdf is parameterized as follows:

$$f(x) = \frac{1}{\theta}\exp(-x/\theta)$$

Arguments

x A quantile at which the pdf is evaluated. *x* must be greater than or equal to 0.

theta Optional scale parameter θ, which must be greater than 0. The default is 1.

Exp Distribution(x, <theta=1>)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the exponential distribution. The cdf is parameterized as follows:

$$F(x) = 1 - \exp(-x/\theta)$$

Arguments

x A quantile at which the cdf is evaluated. *x* must be greater than or equal to 0.

theta Optional scale parameter θ, which must be greater than 0. The default is 1.

Exp Quantile(p, <theta=1>)

Description

Returns the p^{th} quantile from an exponential distribution with scale parameter *theta*. The quantile function is parameterized as follows:

$$F^{-1}(p) = -\theta\log(1-p)$$

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

theta Optional scale parameter θ, which must be greater than 0. The default is 1.

F Density(x, dfnum, dfden, <nc=0>)

Description

Returns the probability density function (pdf) evaluated at *x* for the F distribution with numerator and denominator degrees of freedom *dfnum* and *dfden*, with optional noncentrality parameter *nc*.

$$f(x) = \exp(-\lambda/2) \sum_{r=0}^{\infty} \frac{(\lambda/2)^r}{\mathrm{B}\left(\frac{v_2}{2}, \frac{v_1}{2}+r\right) r!} \left(\frac{v_1}{v_2}\right)^{\frac{v_1}{2}+r} \left(1+\frac{v_1}{v_2}x\right)^{-\left(\frac{v_1+v_2}{2}+r\right)} x^{\frac{v_1}{2}-1+r}$$

where B(·) is the Beta function.

Arguments

x A quantile at which the pdf is evaluated. *x* must be greater than 0.

dfnum The degrees of freedom, v_1, of the chi-square distribution in the numerator of the *F*-distribution. *dfnum* must be greater than 0.

dfden The degrees of freedom, v_2, of the chi-square distribution in the denominator of the *F*-distribution. *dfden* must be greater than 0.

nc Optional noncentrality parameter λ, which must be nonnegative. The default is 0.

F Distribution(x, dfnum, dfden, <nc=0>)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* for the F distribution with numerator and denominator degrees of freedom *dfnum* and *dfden* and noncentrality parameter *nc*.

F Log CDistribution(x, dfnum, dfden, <nc=0>)

Description

Returns the log of (1 - value), where value is the cumulative distribution function evaluated at *x* of the F distribution with numerator and denominator degrees of freedom *dfnum* and *dfden*, with optional noncentrality parameter *nc*.

F Log Density(x, dfnum, dfden, <nc=0>)

Description

Returns the log of the value of the probability density function (pdf) evaluated at *x* for the F distribution with numerator and denominator degrees of freedom *dfnum* and *dfden*, with optional noncentrality parameter *nc*.

F Log Distribution(x, dfnum, dfden, <nc=0>)

Description

Returns the log of the value of the cumulative distribution function (cdf) evaluated at *x* for the F distribution with numerator and denominator degrees of freedom *dfnum* and *dfden* and noncentrality parameter *nc*.

F Noncentrality(x, dfnum, dfden, prob)

Description

Returns the F distribution noncentrality parameter *nc* that satisfies the following:

prob = F Distribution(*x, dfnum, dfden, nc*)

See Also

"F Distribution(x, dfnum, dfden, <nc=0>)"

F Power(alpha, dfh, dfm, d, n)

Description

Returns the power from a given situation involving an *F* test or a *t* test.

Arguments

alpha The significance level of the test. *alpha* must be between 0 and 1.

dfh The hypothesis degrees of freedom. *dfh* must be greater than 0.

dfm The degrees of freedom in the whole model. *dfm* must be greater than 0.

d The squared effect size, defined as Δ^2/σ^2. In this equation, σ^2 is the error variance and Δ^2 is defined as follows:

$$\Delta^2 = (\bar{x} - \mu)^2 \quad \text{for a one-sample } t \text{ test}$$

$$\Delta^2 = \frac{(\bar{x}_1 - \bar{x}_2)^2}{4} \quad \text{for a two-sample } t \text{ test}$$

$$\Delta^2 = \sqrt{\sum_{i=1}^{k} \frac{(\bar{x}_i - \bar{x})^2}{k}} \quad \text{for a } k\text{-sample } F \text{ test}$$

n The total number of observations. *n* must be greater than *dfm*.

F Quantile(x, dfnum, dfden, <nc=0>)

Description

Returns the p^{th} quantile from the F distribution with numerator and denominator degrees of freedom *dfnum* and *dfden* and noncentrality parameter *nc*.

F Sample Size(alpha, dfh, dfm, d, power)

Description

Returns the sample size from a given situation involving an *F* test or a *t* test.

Arguments

`alpha` The significance level of the test. *`alpha`* must be between 0 and 1.

`dfh` The hypothesis degrees of freedom. *`dfh`* must be greater than 0.

`dfm` The degrees of freedom in the whole model. *`dfm`* must be greater than 0.

`d` The squared effect size, defined as Δ^2/σ^2. In this equation, σ^2 is the error variance and Δ^2 is defined as follows:

$$\Delta^2 = (\bar{x} - \mu)^2 \quad \text{for a one-sample } t \text{ test}$$

$$\Delta^2 = \frac{(\bar{x}_1 - \bar{x}_2)^2}{4} \quad \text{for a two-sample } t \text{ test}$$

$$\Delta^2 = \sqrt{\sum_{i=1}^{k} \frac{(\bar{x}_i - \bar{x})^2}{k}} \quad \text{for a } k\text{-sample } F \text{ test}$$

`power` The desired power for the test.

Frechet Density(x, mu, sigma)

Description

Returns the probability density function (pdf) evaluated at *`x`* of the Fréchet distribution. The pdf is parameterized as follows:

$$f(x) = \exp\left[-\exp\left(-\frac{\log(x) - \mu}{\sigma}\right)\right]\exp\left(-\frac{\log(x) - \mu}{\sigma}\right)\frac{1}{x\sigma}$$

Arguments

`x` A quantile at which the pdf is evaluated. *`x`* must be greater than 0.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

Frechet Distribution(x, mu, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *`x`* of the Fréchet distribution. The cdf is parameterized as follows:

$$F(x) = \exp\left[-\exp\left(-\frac{\log(x)-\mu}{\sigma}\right)\right]$$

Arguments

x A quantile at which the cdf is evaluated. *x* must be greater than 0.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

`Frechet Quantile(p, mu, sigma)`

Description

Returns the p^{th} quantile from a Fréchet distribution with location *mu* and scale *sigma*. The quantile function is parameterized as follows:

$$F^{-1}(p) = \exp[-\sigma\log\{-\log(p)\} + \mu]$$

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

`Gamma Density(x, <alpha=1>, <scale=1>, <threshold=0>)`

Description

Returns the probability density function (pdf) evaluated at *x* of the Gamma distribution. The pdf is parameterized as follows:

$$f(x) = \frac{1}{\Gamma(\alpha)\beta^{\alpha}}(x-\theta)^{\alpha-1}\exp(-(x-\theta)/\beta)$$

Arguments

x A quantile at which the pdf is evaluated. *x* must be greater than θ.

alpha Optional shape parameter α, which must be greater than 0. The default is 1.

scale Optional scale parameter β, which must be greater than 0. The default is 1.

threshold Optional threshold parameter θ. The default is 0.

`Gamma Distribution(x, <alpha=1>, <scale=1>, <threshold=0>)`

`IGamma(x, <alpha=1, scale=1, threshold=0>)`

Description

Returns the cumulative distribution function (cdf) evaluated at quantile *x* for the gamma distribution with parameters *alpha*, *scale*, and *threshold*.

Gamma Log CDistribution(x, <alpha=1>, <scale=1>, <threshold=0>)

Description

Same as `Log(1 – Gamma Distribution(x, alpha))` except that it has a much greater range.

Gamma Log Density(x, <alpha=1>, <scale=1>, <threshold=0>)

Description

Same as `Log(Gamma Density(x, alpha))` except that it has a much greater range.

Gamma Log Distribution(x, <alpha=1>, <scale=1>, <threshold=0>)

Description

Same as `Log(Gamma Distribution(x, alpha))` except that it has a much greater range.

Gamma Quantile(p, <alpha=1>, <scale=1>, threshold>)

Description

Returns the *p*th quantile from the gamma distribution with the *alpha*, *scale*, and *threshold* parameters given.

GenGamma Density(x, mu, sigma, lambda)

Description

Returns the probability density function (pdf) evaluated at *x* of an extended generalized gamma probability distribution. The pdf is parameterized as follows:

$$f(x) = \begin{cases} \frac{|\lambda|}{x\sigma}\phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{if } \lambda \neq 0 \\ \frac{1}{x\sigma}\phi_{nor}(\omega) & \text{if } \lambda = 0 \end{cases}$$

where $\omega = [\log(x) - \mu]/\sigma$. Note that the following is the pdf for the standardized log-gamma variable with shape parameter $\kappa > 0$:

$$\phi_{lg}(z;\kappa) = \frac{1}{\Gamma(\kappa)}\exp[\kappa z - \exp(z)]$$

Note that $\phi_{nor}(\cdot)$ is the standard normal pdf.

Arguments

`x` A quantile at which the pdf is evaluated. *x* must be greater than 0.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ.

GenGamma Distribution(x, mu, sigma, lambda)

Description

Returns the cumulative distribution function (cdf) of the extended generalized gamma distribution. The cdf is parameterized as follows:

$$F(x) = \begin{cases} \Phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{if } \lambda > 0 \\ \Phi_{nor}(\omega) & \text{if } \lambda = 0 \\ 1 - \Phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{if } \lambda < 0 \end{cases}$$

where $\omega = [\log(x) - \mu]/\sigma$. Note that the following is the cdf for the standardized log-gamma variable with shape parameter $\kappa > 0$:

$$\Phi_{lg}(z;\kappa) = \Gamma_I[\exp(z);\kappa]$$

where $\Gamma_I[\cdot]$ denotes the incomplete gamma function. Note that $\Phi_{nor}(\cdot)$ is the standard normal cdf.

Arguments

`x` A quantile at which the cdf is evaluated. *x* must be greater than 0.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ.

GenGamma Quantile(p, mu, sigma, lambda)

Description

Returns the p^{th} quantile from an extended generalized gamma distribution with parameters *mu*, *sigma*, and *lambda*. The quantile function does not have a closed form equation.

Arguments

`p` The probability of the quantile desired. *p* must be between 0 and 1.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ.

GLog Density(x, mu, sigma, lambda)

Description

Returns the probability density function (pdf) evaluated at *x* of a generalized logarithmic distribution. The pdf is parameterized as follows:

$$f(x) = \phi\left\{\frac{1}{\sigma}\left[\log\left(\frac{x+\sqrt{x^2+\lambda^2}}{2}\right)-\mu\right]\right\}\frac{x+\sqrt{x^2+\lambda^2}}{\sigma(x^2+\lambda^2+x\sqrt{x^2+\lambda^2})}$$

where $\phi(\cdot)$ is the standard normal pdf.

Arguments

`x` A quantile at which the pdf is evaluated.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ, which must be greater than 0.

Notes

When the shape parameter is equal to zero, the distribution reduces to a Lognormal(μ, σ).

GLog Distribution(x, mu, sigma, lambda)

Description

Returns the probability that a generalized logarithmically distribution random variable is less than *x*. The cdf is parameterized as follows:

$$F(x) = \Phi\left\{\frac{1}{\sigma}\left[\log\left(\frac{x+\sqrt{x^2+\lambda^2}}{2}\right)-\mu\right]\right\}$$

where $\Phi(\cdot)$ is the standard normal cdf.

Arguments

`x` A quantile at which the cdf is evaluated.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ, which must be greater than 0.

GLog Quantile(p, mu, sigma, lambda)

Description

Returns the p^{th} quantile from a generalized logarithmic distribution.

IGamma()

See "Gamma Distribution(x, <alpha=1>, <scale=1>, <threshold=0>)".

Johnson Sb Density(q, gamma, delta, theta, sigma)

Description

Returns the probability density function (pdf) evaluated at *q* of a Johnson Sb distribution. The pdf is parameterized as follows:

$$f(q) = \phi\left[\gamma + \delta \ln\left(\frac{q-\theta}{\sigma-(q-\theta)}\right)\right]\left(\frac{\delta\sigma}{(q-\theta)(\sigma-(q-\theta))}\right)$$

where $\phi(\cdot)$ is the standard normal pdf.

Arguments

q A quantile at which the pdf is evaluated. *q* must be in the interval *theta* to *theta* + *sigma*.

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Scale parameter σ, which must be greater than 0.

Johnson Sb Distribution(q, gamma, delta, theta, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *q* of a Johnson Sb distribution. The pdf is parameterized as follows:

$$F(q) = \Phi\left[\gamma + \delta \ln\left(\frac{q-\theta}{\sigma-(q-\theta)}\right)\right]$$

where $\Phi(\cdot)$ is the standard normal cdf.

Arguments

q A quantile at which the cdf is evaluated. *q* must be in the interval *theta* to *theta* + *sigma*.

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Scale parameter σ, which must be greater than 0.

Johnson Sb Quantile(p, gamma, delta, theta, sigma)

Description

Returns the p^{th} quantile from a Johnson Sb distribution.

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Scale parameter σ, which must be greater than 0.

Johnson Sl Density(x, gamma, delta, theta, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of a Johnson Sl distribution. The pdf is parameterized as follows:

$$f(x) = \frac{\delta}{|x-\theta|}\phi\left[\gamma + \delta \ln\left(\frac{x-\theta}{\sigma}\right)\right]$$

where $\phi(\cdot)$ is the standard normal pdf.

Arguments

x A quantile at which the pdf is evaluated. *x* must be greater than *theta* if *sigma* is 1 and less than *theta* if *sigma* is -1.

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Parameter σ that indicates if the distribution is skewed positively or negatively. sigma must be equal to either +1 (skewed positively) or -1 (skewed negatively).

Johnson Sl Distribution(q, gamma, delta, theta, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *q* of a Johnson Sl distribution.

$$F(x) = \begin{cases} \Phi\left[\gamma + \delta \ln\left(\frac{x-\theta}{\sigma}\right)\right], & \sigma = 1 \\ 1 - \Phi\left[\gamma + \delta \ln\left(\frac{x-\theta}{\sigma}\right)\right], & \sigma = -1 \end{cases}$$

where $\Phi(\cdot)$ is the standard normal cdf.

Arguments

`q` A quantile at which the cdf is evaluated. *`q`* must be greater than *`theta`* if *`sigma`* is 1 and less than *`theta`* if *`sigma`* is -1.

`gamma` Shape parameter γ.

`delta` Shape parameter δ, which must be greater than 0.

`theta` Location parameter θ.

`sigma` Parameter σ that defines if the distribution is skewed positively or negatively. `Sigma` must be equal to either +1 (skewed positively) or -1 (skewed negatively).

`Johnson Sl Quantile(p, gamma, delta, theta, sigma)`

Description

Returns the p^{th} quantile from a Johnson Sl distribution.

Arguments

`p` The probability of the quantile desired. *`p`* must be between 0 and 1.

`gamma` Shape parameter γ.

`delta` Shape parameter δ, which must be greater than 0.

`theta` Location parameter θ.

`sigma` Parameter σ that defines if the distribution is skewed positively or negatively. `Sigma` must be equal to either +1 (skewed positively) or -1 (skewed negatively).

`Johnson Su Density(x, gamma, delta, theta, sigma)`

Description

Returns the probability density function (pdf) evaluated at *`x`* of a Johnson Su distribution. The pdf is parameterized as follows:

$$f(x) = \frac{\delta}{\sigma}\left[1 + \left(\frac{x-\theta}{\sigma}\right)^2\right]^{-1/2}\phi\left[\gamma + \delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)\right]$$

where $\phi(\cdot)$ is the standard normal pdf.

Arguments

`x` A quantile at which the pdf is evaluated.

`gamma` Shape parameter γ.

`delta` Shape parameter δ, which must be greater than 0.

`theta` Location parameter θ.

`sigma` Scale parameter σ, which must be greater than 0.

Johnson Su Distribution(q, gamma, delta, theta, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *q* of a Johnson Su distribution. The cdf is parameterized as follows:

$$F(x) = \Phi\left[\gamma + \delta \sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)\right]$$

where $\Phi(\cdot)$ is the standard normal cdf.

Arguments

q A quantile at which the cdf is evaluated.

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Scale parameter σ, which must be greater than 0.

Johnson Su Quantile(p, gamma, delta, theta, sigma)

Description

Returns the p^{th} quantile from a Johnson Su distribution.

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Scale parameter σ, which must be greater than 0.

LEV Density(x, mu, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of the largest extreme value distribution with location *mu* and scale *sigma*. The pdf is parameterized as follows:

$$f(x) = \frac{1}{\sigma}\exp\left[-\frac{x-\mu}{\sigma} - \exp\left(-\frac{x-\mu}{\sigma}\right)\right]$$

Arguments

x A quantile at which the pdf is evaluated.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

LEV Distribution(x, mu, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the largest extreme value distribution with location *mu* and scale *sigma*. The cdf is parameterized as follows:

$$F(x) = \exp\left[-\exp\left(-\frac{x-\mu}{\sigma}\right)\right]$$

Arguments

x A quantile at which the cdf is evaluated. *x* must be greater than *sigma*.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

LEV Quantile(p, mu, sigma)

Description

Returns the p^{th} quantile from a largest extreme value distribution with location *mu* and scale *sigma*. The quantile function is parameterized as follows:

$$F^{-1}(p) = -\sigma\log(-\log(p)) + \mu$$

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

LogGenGamma Density(x, mu, sigma, lambda)

Description

Returns the probability density function (pdf) evaluated at *x* of a log generalized gamma probability distribution with parameters *mu*, *sigma*, and *lambda*. The pdf is parameterized as follows:

$$f(x) = \begin{cases} \frac{|\lambda|}{\sigma}\phi_{lg}[\lambda\omega + \log(\lambda^{-2});\lambda^{-2}] & \text{if } \lambda \neq 0 \\ \frac{1}{\sigma}\phi_{nor}(\omega) & \text{if } \lambda = 0 \end{cases}$$

where $\omega = [x - \mu]/\sigma$. Note that the following is the pdf for the log-gamma variable with shape parameter $\kappa > 0$:

$$\phi_{lg}(z;\kappa) = \frac{1}{\Gamma(\kappa)}\exp[\kappa z - \exp(z)]$$

Note that $\phi_{nor}(\cdot)$ is the standard normal pdf.

Arguments

`x` A quantile at which the pdf is evaluated.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ.

LogGenGamma Distribution(x, mu, sigma, lambda)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the log generalized gamma distributed random variable (with parameters *mu*, *sigma*, and *lambda*) The cdf is parameterized as follows:

$$F(x) = \begin{cases} \Phi_{lg}[\lambda\omega + \log(\lambda^{-2});\lambda^{-2}] & \text{if } \lambda > 0 \\ \Phi_{nor}(\omega) & \text{if } \lambda = 0 \\ 1 - \Phi_{lg}[\lambda\omega + \log(\lambda^{-2}); \quad \lambda^{-2}] & \text{if } \lambda < 0 \end{cases}$$

where $\omega = [x - \mu]/\sigma$. Note that the following is the cdf for the log-gamma variable with shape parameter $\kappa > 0$:

$$\Phi_{lg}(z;\kappa) = \Gamma_I[\exp(z);\kappa]$$

where $\Gamma_I[\cdot]$ denotes the incomplete gamma function. Note that $\Phi_{nor}(\cdot)$ is the standard normal cdf.

Arguments

`x` A quantile at which the cdf is evaluated.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ.

LogGenGamma Quantile(p, mu, sigma, lambda)

Description

Returns the p^{th} quantile from a log generalized gamma distribution.

Arguments

`p` The probability of the quantile desired. *p* must be between 0 and 1.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ.

Logistic Density(x, mu, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of a logistic distribution with location *mu* and scale *sigma*. The pdf is parameterized as follows:

$$f(x) = \frac{1}{\sigma}\frac{\exp\left(-\frac{x-\mu}{\sigma}\right)}{\left[1+\exp\left(-\frac{x-\mu}{\sigma}\right)\right]^2}$$

Arguments

`x` A quantile at which the pdf is evaluated.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

Logistic Distribution(x, mu, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the logistic distribution with location *mu* and scale *sigma*. The cdf is parameterized as follows:

$$F(x) = \frac{1}{\left[1+\exp\left(-\frac{x-\mu}{\sigma}\right)\right]}$$

Arguments

`x` A quantile at which the cdf is evaluated. *x* must be greater than σ.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

Logistic Quantile(p, mu, sigma)

Description

Returns the p^{th} quantile from a logistic distribution with location *mu* and scale *sigma*. The quantile function is parameterized as follows:

$$F^{-1}(p) = -\sigma\log\left(\frac{1}{p}-1\right)+\mu$$

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

Loglogistic Density(x, mu, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of a loglogistic distribution with location *mu* and scale *sigma*. The pdf is parameterized as follows:

$$f(x) = \frac{1}{x\sigma}\frac{\exp\left(\frac{\log(x)-\mu}{\sigma}\right)}{\left[1+\exp\left(\frac{\log(x)-\mu}{\sigma}\right)\right]^2}$$

Arguments

x A quantile at which the pdf is evaluated.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

Loglogistic Distribution(x, mu, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the loglogistic distribution with location *mu* and scale *sigma*. The cdf is parameterized as follows:

$$F(x) = \frac{1}{1+\exp\left(-\frac{\log(x)-\mu}{\sigma}\right)}$$

Arguments

x A quantile at which the cdf is evaluated.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

Loglogistic Quantile(p, mu, sigma)

Description

Returns the p^{th} quantile from a loglogistic distribution with location *mu* and scale *sigma*. The quantile function is parameterized as follows:

$$F^{-1}(p) = \exp\left[-\sigma\log\left(\frac{1}{p}-1\right)+\mu\right]$$

Arguments

p The probability of the quantile desired. *p* must be between 0 and 1.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

Lognormal Density(x, mu, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of a lognormal distribution with location *mu* and scale *sigma*. The pdf is parameterized as follows:

$$f(x) = \frac{1}{x}\phi\left[\frac{\log(x)-\mu}{\sigma}\right]$$

where $\phi(\cdot)$ is the standard normal pdf.

Arguments

x A quantile at which the pdf is evaluated. *x* must be greater than or equal to 0.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

Lognormal Distribution(x, mu, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of a lognormal distribution with location *mu* and scale *sigma*. The cdf is parameterized as follows:

$$F(x) = \Phi\left[\frac{\log(x)-\mu}{\sigma}\right]$$

where $\Phi(\cdot)$ is the standard normal cdf.

Arguments

x A quantile at which the pdf is evaluated. *x* must be greater than or equal to 0.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

Lognormal Quantile(x, mu, sigma)

Description

Returns the p^{th} quantile of a lognormal distribution with location *mu* and scale *sigma*.

Normal Biv Distribution(x, y, r, <mu1>, <s1>, <mu2>, <s2>)

Description

Computes the probability that an observation (X, Y) is less than or equal to (*x, y*) with correlation coefficient *r* where X is individually normally distributed with mean *mu1* and standard deviation *s1* and Y is individually normally distributed with mean *mu2* and standard deviation *s2*. If *mu1*, *s1*, *mu2*, and *s2* are not given, the function assumes the standard normal bivariate distribution with *mu1*=0, *s1*=1, *mu2*=0, and *s2*=1.

Normal Density(x, <mean=0>, <stddev=1>)

Description

Returns the probability density function (pdf) evaluated at *x* for the normal distribution with *mean* and *stddev*. The pdf is parameterized as follows:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

Arguments

`x` A quantile at which the pdf is evaluated.

`mu` Optional location parameter μ. The default is 0.

`sigma` Optional scale parameter σ, which must be greater than 0. The default is 1.

Notes

The normal distribution is bell shaped and symmetrical.

Normal Distribution(x, <mean=0>, <stddev=1>)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* for the normal distribution with *mean* and *stddev*. The cdf is parameterized as follows:

$$F(x) = \Phi\left(\frac{x-\mu}{\sigma}\right)$$

Note that $\Phi(\cdot)$ is the standard normal cdf, defined as follows:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_0^x \exp\left(-\frac{t^2}{2}\right) dt$$

Arguments

`x` A quantile at which the pdf is evaluated.

`mu` Optional location parameter μ. The default is 0.

`sigma` Optional scale parameter σ, which must be greater than 0. The default is 1.

`Normal Log CDistribution(x, <mean=0>, <std dev=1>)`

Description

Returns 1 - log (value) of the distribution function at quantile *x* for the normal distribution.

`Normal Log Density(x, <mean=0>, <stddev=1>)`

Description

Returns the log of the value of the density function at quantile *x* for the normal distribution with *mean* and *stddev*. The default *mean* is 0. The default *stddev* is 1.

`Normal Log Distribution(x, <mean=0>, <std dev=1>)`

Description

Returns the log of the value of the distribution function at quantile *x* for the normal distribution.

`Normal Mixture Density(q, mean, stdev, probability)`

Description

Returns the density at *q* of a normal mixture distribution with group means *mean,* group standard deviations *stdev,* and group probabilities *probability*. The *mean, stdev,* and *probability* arguments are all vectors of the same size.

`Normal Mixture Distribution(q, mean, stdev, probability)`

Description

Returns the probability that a normal mixture distributed variable with group means *mean,* group standard deviations *stdev,* and group probabilities *probability* is less than *q*. The *mean, stdev,* and *probability* arguments are all vectors of the same size.

`Normal Mixture Quantile(p, mean, stdev, probability)`

Description

Returns the p^{th} quantile, the values for which the probability is *p* that a random value would be lower. The *mean, stdev,* and *probability* arguments are all vectors of the same size.

Normal Quantile(p, <mean=0>, <stddev=1>)

Probit(p, <mean=0>, <stddev=1>)

Description

Returns the p^{th} quantile from the normal distribution with *mean* and *stddev*. The default *mean* is 0. the default *stddev* is 1.

Probit()

See "Normal Quantile(p, <mean=0>, <stddev=1>)".

SEV Density(x, mu, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of the smallest extreme distribution with location *mu* and scale *sigma*. The pdf is parameterized as follows:

$$f(x) = \frac{1}{\sigma}\exp\left[\frac{x-\mu}{\sigma} - \exp\left(\frac{x-\mu}{\sigma}\right)\right]$$

Arguments

x A quantile at which the pdf is evaluated.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

SEV Distribution(x, mu, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the smallest extreme distribution with location *mu* and scale *sigma*. The cdf is parameterized as follows:

$$F(x) = 1 - \exp\left[-\exp\left(\frac{x-\mu}{\sigma}\right)\right]$$

Arguments

x A quantile at which the cdf is evaluated. *x* must be greater than σ.

mu The location parameter μ.

sigma The scale parameter σ, which must be greater than 0.

SEV Quantile(p, mu, sigma)

Description

Returns the p^{th} quantile of the smallest extreme distribution with location *mu* and scale *sigma*. The quantile function is parameterized as follows:

$$F^{-1}(p) = \sigma\log[-\log(1-p)] + \mu$$

Arguments

`p` The probability of the quantile desired. *p* must be between 0 and 1.

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

SHASH Density(x, gamma, delta, theta, sigma)

Description

Returns the probability density function (pdf) evaluated at *x* of a sinh-arcsinh (SHASH) distribution. The pdf is parameterized as follows:

$$f(x) = \frac{\delta\cosh(w)}{\sqrt{\sigma^2 + (x-\theta)^2}}\phi[\sinh(w)]$$

where

$\phi(\cdot)$ is the standard normal pdf

$$w = \gamma + \delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)$$

Arguments

`x` A quantile at which the pdf is evaluated.

`gamma` The shape parameter γ.

`delta` The shape parameter δ, which must be greater than 0.

`theta` The location parameter θ.

`sigma` The scale parameter σ, which must be greater than 0.

SHASH Distribution(x, gamma, delta, theta, sigma)

Description

Returns the cumulative distribution function (cdf) evaluated at *x* of the sinh-arcsinh (SHASH) distribution. The cdf is parameterized as follows:

$$F(x) = \Phi\left[\sinh\left(\gamma + \delta\sinh^{-1}\left(\frac{x-\theta}{\sigma}\right)\right)\right]$$

where $\Phi(\cdot)$ is the standard normal cdf.

Arguments

`x` A quantile at which the cdf is evaluated.

`gamma` The shape parameter γ.

`delta` The shape parameter δ, which must be greater than 0.

`theta` The location parameter θ.

`sigma` The scale parameter σ, which must be greater than 0.

`SHASH Quantile(p, gamma, delta, theta, sigma)`

Description

Returns the p^{th} quantile from a sinh-arcsinh (SHASH) distribution (with parameters *gamma*, *delta*, *theta*, and *sigma*).

Arguments

`p` The probability of the quantile desired. *p* must be between 0 and 1.

`gamma` The shape parameter γ.

`delta` The shape parameter δ, which must be greater than 0.

`theta` The location parameter θ.

`sigma` The scale parameter σ, which must be greater than 0.

`Students t Density()`

See "t Density(x, df, <nc=0>)".

`Students t Distribution()`

See "t Distribution(q, df, <nc=0>)".

`Students t Quantile()`

See "t Quantile(p, df, <nc=0>)".

`t Density(x, df, <nc=0>)`

`Students t Density(x, df, <nc=0>)`

Description

Returns the probability density function (pdf) evaluated at *x* of the Student's *t* distribution with degrees of freedom *df*. The pdf is parameterized as follows:

$$f(x) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)} \frac{1}{\sqrt{\nu\pi}} \left[1 + \frac{x^2}{\nu}\right]^{-\left(\frac{\nu+1}{2}\right)}$$

Arguments

`x` A quantile at which the pdf is evaluated.

`df` The degrees of freedom ν, which must be greater than or equal to 1.

`nc` The optional noncentrality parameter λ, which must be nonnegative. The default is 0.

t Distribution(q, df, <nc=0>)

Students t Distribution(q, df, <nc=0>)

Description

Returns the probability that a Student's *t* distributed random variable is less than *q*. *nc* defaults to 0.

t Log CDistribution(x, df, <nc=0>)

Description

Returns 1 - log (value) of the normal distribution function at quantile *x* for the *t* distribution.

t Log Density(x, df, <nc=0>)

Description

Returns the log of the value of the density function at quantile *x* for the *t* distribution.

t Log Distribution(x, df, <nc=0>)

Description

Returns the log of the value of the distribution function at quantile *x* for the *t* distribution.

t Noncentrality(x, df, prob)

Description

Returns the *t* distribution noncentrality parameter *nc* that satisfies the following:

prob = T Distribution(*x, df, nc*)

t Quantile(p, df, <nc=0>)

Students t Quantile(p, df, <nc=0>)

Description

Returns the p^{th} quantile from the Student's *t* distribution with degrees of freedom *df*. *nc* defaults to 0.

Tukey HSD P Value(q, n, dfe)

Description

Returns the *p*-value from Tukey's HSD multiple comparisons test.

Arguments

`q` The test statistic. The test statistic that is specified is Tukey's adjusted critical value, which is the quantile of Tukey's studentized range distribution divided by the square root of 2.

`n` The number of groups in the study.

`dfe` The error degrees of freedom, based on the total study sample.

`Tukey HSD Quantile(1-alpha, n, dfe)`

Description

Returns the quantile used in Tukey's HSD multiple comparisons test. The quantile that is returned is Tukey's adjusted critical value, which is the quantile of Tukey's studentized range distribution divided by the square root of 2.

Arguments

`1-alpha` The confidence level.

`n` The number of groups in the study.

`dfe` The error degrees of freedom, based on the total study sample.

`Weibull Density(x, shape, <scale=1>, <threshold=0>)`

Description

Returns the probability density function (pdf) evaluated at *x* of the Weibull distribution. The pdf is parameterized as follows:

$$f(x) = \frac{\beta}{\alpha}\left(\frac{x-\theta}{\alpha}\right)^{\beta-1}\exp\left[-\left(\frac{x-\theta}{\alpha}\right)^{\beta}\right]$$

Arguments

`x` A quantile the pdf is evaluated at. *x* must be greater than *threshold*.

`shape` Shape parameter β, which must be greater than 0.

`scale` Optional scale parameter α, which must be greater than 0. The default is 1.

`threshold` Optional threshold parameter θ. The default is 0.

`Weibull Distribution(x, shape, <scale=1>, <threshold=0>)`

Description

Returns the cumulative distribution function (cdf) at *x* of the Weibull distribution. The cdf is parameterized as follows:

$$F(x) = 1 - \exp\left[-\left(\frac{x-\theta}{\alpha}\right)^{\beta}\right]$$

Arguments

`x` A quantile at which the pdf is evaluated. *x* must be greater than *threshold*.

`shape` Shape parameter β, which must be greater than 0.

`scale` Optional scale parameter α, which must be greater than 0. The default is 1.

`threshold` Optional threshold parameter θ. The default is 0.

`Weibull Quantile(p, shape, <scale=1>, <threshold=0>)`

Description

Returns the p^{th} quantile from the Weibull distribution with the parameters given. The quantile function is calculated as follows:

$$F^{-1}(p) = \alpha[\ln(1-p)]^{\frac{1}{\beta}} + \theta$$

Arguments

`p` The probability of the quantile desired. *p* must be between 0 and 1.

`shape` Shape parameter β, which must be greater than 0.

`scale` Optional scale parameter α, which must be greater than 0. The default is 1.

`threshold` Optional threshold parameter θ. The default is 0.

Programming Functions

`As Boolean(x)`

Description

Evaluates a JSL expression and returns a JSL Boolean value for use with JSON data.

Example

```
x = 45;
b = As Boolean( x > 2 );
Show( b );
```

```
b = true;
```

`As Column(name)`

`As Column(dt, name)`

`:name`

`dt:name`

Description

This scoping operator forces *name* to be evaluated as a data table column in the current data table (or the table given by the optional data table reference argument, *dt*) rather than as a global variable.

Arguments

`name` Variable name.

`dt` The data table reference

Notes

`:name` refers to a column name in the current data table. You can also specify which data table to refer to by use `dt:name`.

`As Constant(expr)`

Description

Evaluates an expression once to create a value that does not change after it is computed.

Returns

The result of the evaluation.

Argument

`expr` Any JSL expression.

Notes

A few platforms that can save prediction columns to a data table use `As Constant()`. The function is wrapped around the part of the formula that is constant across all rows. The argument is evaluated for the first row and then the result is used without re-evaluation for subsequent rows.

`As Global(name)`

`::name`

Description

This scoping operator forces *name* to be evaluated as a global variable rather than as a data table column.

Arguments

`name` Variable name.

`As List(matrix)`

See "As List(matrix)".

`As Name(string)`

Description

Evaluates argument as a quoted string and changes it into a name.

Returns

A name.

`As Namespace(name)`

Description

Accesses the specified namespace. An error is thrown if no such namespace exists.

Returns

The namespace.

Arguments

`name` Unquoted name of a defined namespace.

`As Scoped(namespace, variable)`

`namespace:variable`

Description

Accesses the specified *`variable`* within the specified *`namespace`*.

Returns

The value of the variable, or an error the scoped variable is not found.

Arguments

`namespace` The name of a defined namespace.

`variable` A variable defined within *`namespace`*.

`Associative Array({key, value}, ...)`

`Associative Array(keys, values)`

Description

Creates an associative array (also known as a dictionary or hash map).

Returns

An associative array object.

Arguments

Either list of key-value pairs; or a list, matrix, or data table column that contains keys followed by a list, matrix, or data table column, respectively, that contains the corresponding values.

Class Exists(class)

Description

Returns a value indicating whether a class definition represented by the class reference is a defined class.

Returns

0 or 1.

Argument

`class` A quoted string representation of the name of a defined class or reference to an instantiated class object.

Clear Globals(<name>, <name>, ...)

Description

Clears the values for all global symbols. Symbols in any scope other than global are not affected. If one or more names are specified, only those global symbols are cleared.

Returns

Null.

Optional Arguments

`name` Any global variable name(s).

See Also

"Clear Symbols(<name>, <name>, ...)"

Clear Log()

Description

Empties the log.

Clear Symbols(<name>, <name>, ...)

Description

Clear the values for all symbols in any and all scopes. If one or more names are specified, only those symbols are cleared.

Returns

Null.

Optional Arguments

`name` Any global variable name(s).

See Also

"Clear Globals(<name>, <name>, ...)"

`Close Log()`

Description

Closes the log.

```
Define Class("class name", <Base Class( "base class name", <"base class
name", ...> ),> <Show( All(Boolean) ) | Show( <Members(Boolean),>
<Methods(Boolean),> <Functions(Boolean)> ),> <Assignment Statements>)
```

Description

Defines a new class object.

Example

```
Define Class(
   "aa",
   _init_ = Method( {} ); x = 1; m1 = Method( {a, b}, a * b )
);
```

`Delete Classes(<Force(Boolean)>, < <class>, ...>)`

Description

Deletes all currently defined classes.

Optional Arguments

`Force(Boolean)` Deletes the class or classes even if they are in use.

`class` Specifies the classes to delete. You can specify more than one class. This argument can be a quoted string representation of the name of a defined class or a reference to an instantiated class object.

`Delete Globals(<name>, <name>, ...)`

Description

Deletes all global symbols, except global symbols that are locked. Symbols in any scope other than global are not affected. If one or more names are specified, only those global symbols are cleared.

Optional Arguments

`name` Any global variable name(s).

See Also

"Delete Symbols(<name>, <name>, ...)"

`Delete Namespaces(<Force(Boolean expression)>, < <namespace reference>, ...>)`

`Delete(<Force(Boolean expression)>, < <namespace reference>, ...>)`

Description

Deletes all currently defined namespaces or one or more specific namespaces.

Optional Arguments

`Force(Boolean expression)` Deletes the namespace even if it's in use.

`namespace reference` Specifies the namespaces to delete. You can specify more than one namespace reference.

Notes

- When you delete a namespace that contains locked namespaces, an error appears in the log. Use the `Force()` argument to delete the locked namespaces.
- With no arguments, `Delete Namespaces()` ignores locked namespaces.

`Delete Symbols(<name>, <name>, ...)`

Description

Deletes all symbols in any and all scopes. If one or more names are specified, only those symbols are deleted.

Optional Arguments

`name` Any global variable name(s).

See Also

"Delete Globals(<name>, <name>, ...)"

`Eval(expr)`

Description

Evaluates `expr`, and then evaluates the result of `expr` (unquoting).

Returns

The result of the evaluation.

Argument

`expr` Any JSL expression.

`Eval Insert(`*`string`*`, <startDel>, <endDel>, < <<Use Locale(1) >)`

Description

Allows for multiple substitutions.

Returns

The result.

Arguments

`string` A quoted string with embedded expressions.

`startDel` Optional starting delimiter. The default value is ^.

`endDel` optional ending delimited. The default value is the starting delimiter.

`Use Locale(1)` Optional argument that preserves locale-specific numeric formatting.

`Eval Insert Into(`*`string`*`, <startDel>, <endDel>)`

Description

Allows for multiple substitutions in place. The same operation as in `Eval Insert` is performed, and the result is placed into the quoted *`string`*.

Returns

The result.

Arguments

`string` A quoted string variable that contains a string with embedded expressions.

`startDel` Optional starting delimiter. The default value is ^.

`endDel` optional ending delimited. The default value is the starting delimiter.

`Eval List`

See "Eval List({`list`})".

`Exit(<NoSave>)`

`Quit(<NoSave>)`

Description

Exits JMP.

Returns

Void.

Arguments

`NoSave` Optional, named command; exits JMP without prompting to save any open files. This command is *not* case-sensitive, and spaces are optional.

`First(expr, <expr>, ...)`

Description

Evaluates all expressions provided as arguments.

Returns

Only the result of the first evaluated expression.

Arguments

`expr` Any valid JSL expression.

`Function({arguments}, <{local variables}>, <Return(<expr>)>, script)`

Description

Stores the body *`script`* with *`arguments`* as local variables.

Returns

The function as defined. If the `Return()` argument is specified, the expression is returned.

When called later, it returns the result of the *`script`* given the specified *`arguments`*.

Arguments

`{arguments}` A list of arguments to pass into the function. You can specify some arguments as optional or required.

`{local variables}` A list of variables that are local to the function. You can declare local variables in three ways:

```
{var1, var2}
{var1=0, var1="a string"}
{Default Local}
```

The last option declares that all unscoped variables used in the function are local to the function.

`Return(expr)` This optional argument returns an expression from an user defined function. If a null expression is used, a period, ".", is returned.

`script` Any valid JSL script.

`Get Class Names(< <class>, ...>)`

Description

Gets a set of names to all classes or the set of specific class references.

Arguments

`class` A quoted string representation of the name of a defined class or a reference to an instantiated class object.

Returns

A list of class names as determined by the arguments to the function.

`Get Classes(< <class>, ...>)`

Description

Gets a set of references to all classes or the set of specific class references.

Arguments

`class` A quoted string representation of the name of a defined class or a reference to an instantiated class object.

Returns

A list of class references as determined by the arguments to the function.

Get Environment Variable("variable")

Description

Retrieves the value of an operating system environment variable.

Returns

A quoted string that contains the value of the specified environment variable. If the specified variable is not found, an empty string is returned.

Arguments

"variable" A quoted string that contains the name of an environment variable.

Notes

On macOS, environment variable names are case-sensitive. On Windows, the names are case-insensitive.

Get Locale Setting()

Gets a local setting such as a decimal setting.

Get Log(<n>)

Description

Returns a list of lines from the log.

Returns

A list of quoted strings. Each string contains one line from the log.

Argument

n Optional, integer. If no argument is specified, all the lines are returned. If a positive number is specified, the first *n* lines are returned. If a negative number is specified, the last *n* lines are returned. If *n*=0, no lines are returned (an empty list). If the log is empty, an empty list is returned.

Get Namespace Names(< <namespace reference>,...>)

Description

Returns a list of the names of all currently defined namespaces.

Example

```
nsaa = New Namespace(
   "aa",
   {
      x = 1
   }
```

```
);
nsbb = New Namespace(
   "bb",
   {
      y = 1
   }
);
lns = Get Namespace Names();
Show( lns );
nsaa << Delete;
nsbb << Delete;
```

Get Namespaces(< <namespace reference>,...>)

Description

Returns a list of currently defined namespaces.

Example

```
nsaa = New Namespace(
   "aa",
   {
      x = 1
   }
);
nsbb = New Namespace(
   "bb",
   {
      y = 1
   }
);
lns = Get Namespaces();
```

Include("pathname", <named arguments>)

Description

Opens the script file identified by the quoted string *pathname*, parses the script in it, and executes it.

Returns

Whatever the included script returns. If you use the `<<Parse Only` option, `Include` returns the contents of the script.

Named Arguments

`<<Parse Only` Parses the script but does not execute the script.

`<<New Context` Causes the included script to be run its own unique namespace. When the parent and included scripts use the global namespace, include `<<Names Default to Here` along with `<<New Context`.

`<<Allow Include File Recursion` Lets the included script include itself.

Notes

If a trailing space is included in the path name, the space is ignored on Windows. On macOS, the script fails.

`Include File List()`

Description

Returns a list of files that are included at the point of execution.

`Is Class(class)`

Description

Returns a value that indicates whether the class reference is a class object.

Argument

A class reference to an instantiated class object.

Returns

Returns a zero or a 1.

`Is Log Open()`

Description

Returns result if log window is open.

`Length`

See "Length(`string`)".

`List`

See "List(a, b, c, ...)".

`Local({name=value, ...}, script)`

Description

Resolves *`names`* to local expressions.

`Local Here(expression)`

Description

Creates a local Here namespace block. Use this function to prevent name collisions when multiple scripts are executed from the same root namespace (for example, when a script

executes two button scripts that have the same variables). The argument can be any valid JSL expression.

Lock Namespaces(<*string*>,|< {*string*}, ...>)

Description

Locks all variables or specified named variables in this namespace and prevents variables from being added, changed, or removed.

Example

```
ns = New Namespace(
   "aaa"
);
ns << Lock Namespaces;
Try( ns << Delete Namespaces, Show( exception_msg ) );
Delete Namespaces();
Try( Delete Namespaces( "aaa" ), Show( exception_msg ) );
```

Lock Globals(name1, name2, ...)

Description

Locks one or more global variables to prevent it or them from being changed.

Lock Symbols(<name>, <name>, ...)

Description

Locks the specified symbols, which prevents them from being modified or cleared. If no symbols are provided, all global symbols are locked. If no symbols are provided and the script has the *Names Default To Here* mode turned on, then all local symbols are locked.

Log Capture(expr)

Description

Evaluates the expr, captures the output that would normally be sent to the log, and instead returns it.

Returns

A quoted string that contains the log output.

Argument

Any valid JSL expression.

Notes

No output appears in the log.

Method({arg1 = val1, ...}, script)

Description

Creates a method within a class. Note that methods use local scoping for all variables that are not explicitly scoped, with the exception of class member variables.

Arguments

`{ arg1 = val1, … }` The set of expected arguments and optional initialization expressions to be passed to the method when called.

`script` Any valid JSL script.

N Items

See "N Items(source)".

Names Default To Here(Boolean)

Description

Determines where unresolved names are stored, either as a global or local (if *Boolean* is 0) or in the `Here` scope (if *Boolean* is 1).

Namespace(name)

Description

Returns a reference to the named namespace (*name*).

Argument

`Name` A namespace name quoted string or a reference to a namespace.

Namespace Exists(name)

Description

Returns 1 if a namespace with the specified *name* exists; otherwise, returns 0.

New Namespace(<"name">, <{expr, ...}>)

Description

Creates a new namespace with the specified name. If a name is not provided, an anonymous name is provided.

Returns

A reference to the namespace.

Arguments

`name` An optional, quoted string that contains the name of the new namespace.

`{list of expressions}` An optional list of expressions within the namespace.

`Open Log(<Boolean>)`

Description

Opens the log. Include the Boolean argument to make the window active, even if it is already open.

```
New Object("class name"(constructor arguments))
New Object(class name(constructor arguments))
New Object(class reference(constructor arguments))
```

Description

Creates an instance object of a class.

Arguments

`"class name"` Name of the class to be instantiated.

`class name` Unquoted name of the class to be instantiated.

`class reference` Reference to an existing class object that will be used to instantiate a new object of the same class.

`constructor arguments` Set of arguments to be passed to the _init_ constructor.

Example

```
Define Class(
   "complex",
   real = 0; imag = 0;
   _init_ = Method( {a, b}, real = a; imag = b; );
   Add = Method( {y}, complex( real + y:real, imag + y:imag ) );
   Sub = Method( {y}, complex( real - y:real, imag - y:imag ) );
   Mul = Method( {y},
      complex( real * y:real - imag * y:imag, imag * y:real + real * y:imag )
   );
   Div = Method( {y},
      t = complex( 0, 0 );
      mag2 = y:Magsq();
      t:real = real * y:real + imag * y:imag;
      t:imag = imag * y:real + real * y:imag;
      t:real = t:real / mag2;
      t:imag = t:imag / mag2;
      t;
   );
   Magsq = Method( {}, real * real + imag * imag );
   Mag = Method( {}, Sqrt( real * real + imag * imag ) );
   To String = Method( {}, Char( real ) || " + " || Char( imag ) || "i" )
);
cl = New Object( complex( 1, 2 ) );
```

`Parameter({name=value, ...}, model expression)`

Description

Defines formula parameters for models for the Nonlinear platform.

`Parse(string)`

Description

Converts a quoted character string into a JSL expression.

`Print(expr, expr, ...)`

Description

Prints the values of the specified *expressions* to the log.

`Quit()`

See "Exit(<NoSave>)".

`Recurse(function)`

Description

Makes a recursive call of the defining *function*.

`Save Log(pathname)`

Description

Writes the contents of the log to the specified file location.

`Send(obj, message)`

`obj << message`

Description

Sends a *message* to a platform *object*.

`Set Environment Variable( "variable", <"value">)`

Description

Sets the environment variable to the value specified. If the "value" argument is missing or is an empty quoted string, then the environment variable is deleted from the JMP process environment variable table.

Show(expr, expr, ...)

Description

Prints the name and value of each *expression* to the log.

Show Classes(< <class>,...>)

Description

Shows the contents of user-defined classes in the log. You can specify more than one class. If you do not specify an argument, all user-defined classes are shown in the log.

Example

```
Define Class(
   "aa",
   _init_ = Method( {} ); x = 1; m1 = Method( {a, b}, a * b )
);
Define Class(
   "bb",
   _init_ = Method( {} ); y = 1; m2 = Method( {a, b}, a / b )
);
Show Classes();
   // Class aa

   _init_ = Method( {} );
   m1 = Method( {a, b}, a * b );
   x = 1;

   // Class bb

   _init_ = Method( {} );
   m2 = Method( {a, b}, a / b );
   y = 1;
```

Show Globals()

Description

Shows the values for all global symbols. Symbols in any scope other than global are not shown.

See Also

"Show Symbols()"

Show Namespaces(< <namespace reference>,...>)

Description

Shows the contents of all user-defined namespaces, both named and anonymous. You can specify zero or more namespaces.

Show Symbols()

Description

Shows the values for all symbols in any and all scopes.

See Also

"Show Globals()"

Sort List

See "Sort List({list}|expr)".

Sort List Into

See "Sort List Into({list}|expr)".

Throw("text")

Description

Returns a Throw. If you include *text*, throwing stores *text* in a global *exception_msg*. If *text* begins with "!" and is inside a Try() expression, throwing creates an error message about where the exception was caught. "!" stops the script even if the Throw() is caught by the second argument of Try().

Try(expr1, expr2)

Description

Evaluates *expr1*. If the evaluation returns a Throw, execution stops, and nothing is returned. *expr2* is evaluated next to return the result.

Examples

```
Try( Sqrt( "s" ), "invalid" );
   "invalid"

Try( Sqrt( "s" ), exception_msg );
   {"Cannot convert argument to a number [or matrix]"(1, 2, "Sqrt",
      Sqrt/*###*/("s"))}
```

Notes

Expr2 can be a quoted character string or the global exception message (exception_msg) that contains more information about the error returned.

`Type(x)`

Description

Returns a quoted string that names the type of object *x* is. The list of possible types is: Unknown, List, DisplayBox, Picture, Column, TableVar, Table, Empty, Pattern, Date, Integer, Number, String, Name, Matrix, RowState, Expression, Associative Array, BLOB.

`Unlock Symbols(name1, name2, ...)`

`Unlock Globals(name1, name2, ...)`

Description

Unlocks the specified symbols that were locked with a `Lock Symbols()` or `Lock Globals()` command.

`Wait(n)`

Description

Pauses *n* seconds before continuing the script. The default setting is 3 seconds. Specifying `Wait(0)` enables one cycle of message processing. For example, you can use this function to allow a button press in the UI. The shortest duration that actually allows JMP to pause is `n = 0.01`. The longest duration you can specify without prompting a JMP dialog is `n = 60*60*4`.

Notes

You can use `Wait(n)` if you want something to stay on the screen long enough to see it, if you need a platform to finish launching before scripting it, or if you need to press buttons in the UI while the script runs.

`Watch(all | name1, ...)`

Description

Shows variables from global, here, and local namespaces and their values in a window. If "all" is provided as the argument, all globals are placed into the window.

Notes

- New globals are not added to the window list.
- Watching associative arrays that have been modified using messages is not supported.

`Wild()`

Description

Only used with `Extract Expr()` for expression matching to denote a wildcard position that matches any expression.

`Wild List()`

Description

Only used with `Extract Expr()` for expression matching to denote a series of wildcard arguments that match any expression.

`Write("text")`

Description

Prints *`text`* to the log without surrounding quotation marks.

Python Integration Functions

`Python Connect(<Echo(Boolean),> <Path(path),> <Use Python Version(`*`string`*`),> <Python System Path(list)>)`

Description

Initializes the Python integration interfaces and returns an active Python integration interface connection as a scriptable object.

Returns

A Python scriptable object.

Optional Named Arguments

`Echo(Boolean)` Global argument. Prints the Python source lines to the JMP log. The default value is true.

`Path` Specifies the path to the Python DLL or shared library.

`Use Python Version(`*`string`*`)` Specifies which version of Python should be used for JMP-to-Python processing.

`Python System Path` Specifies a JSL list of paths that define a Python sys path set on macOS.

`Python Control(<named arguments>)`

Description

Sends control operations to signal Python with external events, such as source line echoing.

Returns

Returns 0 if the call succeeded and 1 if an error occurred.

Optional Named Arguments

`Interactive(Boolean)` Enables interactive mode in the Python matplotlib package. Determines whether the graphics window is released or closed when graphics rendering is complete.

`Echo(Boolean)` Global argument. Prints the Python source lines to the JMP log. The default value is true.

`Python Disconnect`

Description

Terminates the Python interfaces.

`Python Execute({list of inputs}, {list of outputs}, Python_Code, named_arguments)`

Description

Submits Python code to the active global Python integration interface connection given a list of inputs. On completion, returns a list of outputs.

Returns

Returns 0 if successful and 1otherwise.

Positional Arguments

`{list of inputs}` A list of JMP variable names to be sent to Python as inputs.

`{list of outputs}` A list of JMP variable names to be retrieved from Python as outputs.

`Python_Code` The Python code to submit.

Named Arguments

See "Python Submit(Python_Code, <named_arguments>)".

Example

This example initiates the Python connection, sends a character variable, a numeric variable, and a set of matrices to Python. Python is then instructed to perform a set of matrix operations on the sent matrices. The `Python Execute()` function then get the set of matrices created by the matrix operations and gets the values of the character and numeric variables that was originally sent. Upon completion of the data retrieval, the Python connection is closed.

```
Python Init();
a = "abcdef";
d = 3.141;
v = [9 8 7, 6 5 4, 3 2 1];
m = [1 2 3, 4 5 6, 7 8 9];
ml = Python Execute(
  {v, m, a, d},
  {x1, x2, y1, y2, z1, z2, a, d},
  "\[
```

```
import numpy as np
x1 = np.multiply(v, m) # matrix product
print('x1=', x1)
x2 = np.divide(v, m) # matrix division
print('x2=', x2)
y1 = np.dot(v, m) # dot product of v and m
print('y1=', y1)
y2 = np.dot(m, v) # dot product of m and v
print('y2=', y2)
z1 = np.inner(v, m) # inner product of v and m
print('z1=', z1)
z2 = np.inner(m, v) # inner product of m and v
print('z2=', z2)
]\"
);
Show( v, m, ml, x1, x2, y1, y2, z1, z2, a, d );
Python Term();
```

```
x1= [[  9.  16.  21.]
 [ 24.  25.  24.]
 [ 21.  16.   9.]]
x2= [[ 9.          4.          2.33333333]
 [ 1.5         1.          0.66666667]
 [ 0.42857143  0.25        0.11111111]]
...
```

Python Get(name)

Description

Gets a named variable from Python to JMP.

Returns

Returns the value of the named variable.

Argument

name The name of the Python variable to be sent to JMP. The argument can represent any of the following Python data types: numeric, quoted string, matrix, list, or data frame.

Example

```
Python Init(); // initiate the Python connection

qbx = "The right stuff";

// send the qbx variable and sample data table "Animals.jmp" to Python
Python Send( qbx );

dt = Open( "$SAMPLE_DATA/Animals.jmp" );
Python Send( dt );
Close( dt, nosave );
```

```
// get the Python variable qbx and place it into a JMP variable qbx
qbx = Python Get( qbx );

/* get the Python variable dt and place it into a JMP data table
referenced by df */
df = Python Get( dt );

Python Term();

Show( qbx );
df << New Data View;
Wait( 10 );
Close( df, nosave );
Python Term();
   qbx = "The right stuff";
   0
```

Python Get Graphics(format)

Description

Gets the last graphics object written to the Python graph display window in the specified graphics format. The graphics object can be returned in several different graphic formats.

Returns

Returns a JMP picture object.

Argument

`format` The format that the Python graph display window contents are to be converted to. Valid formats are `png`, `bmp`, `jpeg`, `jpg`, `tiff`, `tif`, and `gif`.

Python Get Version

Description

Returns the version number of Python being used with the JMP Python interfaces.

Python Init(<Echo(Boolean),> <Path(path),> <Use Python Version(*string*),> <Python System Path({list})>

Description

Initializes the Python integration interfaces.

Returns

Returns 0 if operation is successful and 1 if not successful.

Optional Named Arguments

`Echo(Boolean)` Global argument. Prints the Python source lines to the JMP log. The default value is true.

`Path` Specifies the path to the Python DLL or shared library.

`Use Python Version(`*`string`*`)` Specifies which version of Python should be used for JMP-to-Python processing.

`Python System Path` Specifies a JSL list of paths that define a Python sys path set on macOS.

Python Is Connected

Description

Determines whether a Python integration interface connection is currently connected to Python.

Returns

Returns 1 if connected and 0 otherwise.

Python JMP Name to Python Name(name)

Description

Maps a JMP variable name to its corresponding Python variable name using Python variable name naming rules.

Returns

A quoted string, the mapped Python name.

Argument

`name` The name of the JMP variable to be sent to Python.

Python Send(name)

Description

Sends a named variable from JMP to Python.

Returns

Returns 0 if successful.

Argument

`name` The name of the JMP variable to be sent to Python.

Python Send File(filename, <, Python Name(name)>)

Description

Sends a data file to Python. The filename argument is a quoted string that specifies a pathname to the file to be sent to Python.

Python Submit(Python_Code, <named_arguments>)

Description

Submits Python code to the active global Python integration interface connection.

Returns

Returns 0 if successful and non-zero otherwise.

Named Arguments

`Python_Code` The Python code to submit. Statements can be a quoted string value or a list of string values.

`Expand(Boolean)` (Optional) Performs an `Eval Insert()` on the Python code before submission.

`Echo(Boolean)` (Optional) Prints the Python source lines to the JMP log.

Example

```
Python Init(); // initiate the Python connection
commands =
"
friends = ['john', 'pat', 'gary', 'michael']
print(friends)
for i, name in enumerate(friends):
    print( \!"iteration {iteration} is {name}\!".format(iteration=i,
  name=name))
";
Python Submit( commands );
Python Term();
```

```
['john', 'pat', 'gary', 'michael']
iteration 0 is john
iteration 1 is pat
iteration 2 is gary
iteration 3 is michael

0
```

Python Submit File(path)

Description

Submits statements to Python using the file specified in the path name.

Argument

`path` The path to the file that contains the Python source lines to be executed.

Python Term

Description

Terminates the currently active Python integration interface.

Returns

Returns 0 if successful and 1 otherwise.

R Integration Functions

`R Connect( <named_arguments> )`

Description

Returns the current R connection object. If there is no connection to R, it intializes the R integration interfaces and returns an active R integration interface connection as a scriptable object.

Returns

R scriptable object.

Arguments

`Echo(Boolean)` (Optional) Sends all source lines to the JMP log. This option is global. The default value is `true`.

`R Control(Interrupt|Async(Boolean)|Echo(Boolean))`

Description

Changes the control options for R.

`R Execute( { list of inputs }, { list of outputs }, "rCode", <named_arguments> )`

Description

Submit the specified R code to the active global R connection given a list of inputs. On completion, the outputs are returned into the specified list.

Returns

0 if successful; nonzero otherwise.

Arguments

`{ list of inputs }` A list of JMP variable names to be sent to R as inputs.

`{ list of outputs }` A list of JMP variable names to contain the outputs returned from R.

`rCode` A quoted string that contains the R code to submit.

`Expand(Boolean)` An optional, Boolean, named argument. Performs an `Eval Insert()` on the R code before submitting to R.

`Echo(Boolean)` An optional, Boolean, named argument. Sends all source lines to the JMP log. This option is global. The default value is `true`.

Example

Send the JMP variables x and y to R, execute the R statement z <- x * y, and then get the R variable z and return it to JMP.

```
x = [1 2 3];
y = [4 5 6];
rc = R Execute( {x, y}, {z}, "z <- x * y" );
```

R Get(variable_name)

Description

Gets the named variable from R to JMP.

Returns

The value of the named variable.

Argument

`name` Required. The name of an R variable whose value to return to JMP.

Example

Assume that a matrix named qbx and a data frame named df are present in your R connection.

```
// get the R variable qbx and placed it into a JMP variable qbx
qbx = R Get( qbx );

// get the R variable df and placed it into a JMP data table referenced by df
df = R Get( df );
```

R Get Graphics("format")

Description

Gets the last graphics object written to the R graph display window in the specified format.

Returns

A JMP picture object.

Argument

`format` Required. Specifies the graphics format to be used. Valid formats are `png`, `bmp`, `jpeg`, `jpg`, `tiff`, `tif`, and `gif`.

R Get Version

Description

Returns the version number of R being used with JMP R interfaces.

R Init(named_arguments)

Description

Initializes the R session.

Returns

0 if the initialization is successful; any nonzero value otherwise.

Argument

`Echo(Boolean)` (Optional) Sends all source lines to the JMP log. This option is global. The default value is `true`.

R Is Connected()

Description

Determines whether a connection to R exists.

Returns

1 if connected; 0 otherwise.

Arguments

None.

R JMP Name to R Name(name)

Description

Maps the specified JMP variable name to the corresponding R variable name using R naming rules. Requires an active connection to R.

Argument

`name` The name of a JMP variable to be sent to R.

Returns

A quoted string that contains the R name.

R Send(name, <R Name(name)>)

Description

Sends named variables from JMP to R.

Returns

0 if the send is successful; any nonzero value otherwise.

Arguments

`name` required. The name of a JMP variable to be sent to R.

`R Name(name)` (Optional) You can give the variable that you send to R a different name. For example

```
R Send(Here:x, R Name("localx"))
```

For data tables only:

`Selected(Boolean)` optional, named, Boolean. Send only selected rows from the referenced data table to R.

`Excluded(Boolean)` optional, named, Boolean. Send only excluded rows from the referenced data table to R.

`Labeled(Boolean)` optional, named, Boolean. Send only labeled rows from the referenced data table to R.

`Hidden(Boolean)` optional, named, Boolean. Send only hidden rows from the referenced data table to R.

`Colored(Boolean)` optional, named, Boolean. Send only colored rows from the referenced data table to R.

`Markered(Boolean)` optional, named, Boolean. Send only markered rows from the referenced data table to R.

`Row States(Boolean, <named arguments>)` optional, named. Includes a Boolean argument and optional named arguments. Send row state information from the referenced data table to R by adding an additional data column named "RowState". Multiple row states are created by adding together individual settings. Here are the individual values:

- Selected = 1
- Excluded = 2
- Hidden = 4
- Labeled = 8
- Colored = 16
- Markered = 32

Here are the named arguments for the `Row States()` argument:

`Colors(Boolean)` optional, named, Boolean. Sends row colors. Adds additional data column named "RowStateColor".

`Markers(Boolean)` optional, named, Boolean. Sends row markers. Adds additional data column named "RowStateMarker".

Examples

Create a matrix, assign it to X, and send the matrix to R:

```
X = [1 2 3];
rc = R Send( X );
```

Open a data table, assign a reference to it (*dt*), and send the data table, along with its current row states, to R:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
rc = R Send( dt, Row States(1) );
```

R Send File("pathname", <R Name("name")>)

Description

Sends the specified data file from JMP to R.

Returns

0 if the send is successful; any nonzero value otherwise.

Arguments

`pathname` required. A quoted string that contains a pathname for a file.

`R Name(name)` (Optional) You can give the data file that you send to R a different name.

R Submit("rCode", <named_arguments>)

Description

Submits the specified R code to the active global R connection.

Returns

0 if successful; nonzero otherwise.

Arguments

`rCode` A required, quoted string that contains the R code to submit.

`Expand(Boolean)` An optional, Boolean, named argument. Performs an `Eval Insert()` on the R code before submitting to R.

`Echo(Boolean)` An optional, Boolean, named argument. Sends all source lines to the JMP log. This option is global. The default value is `true`.

`Async(Boolean)` An optional, Boolean, named argument. If set to true (`1`), the submit can be canceled either by pressing the ESCAPE key, or by using this message to an R connection: `rconn<<Control( Interrupt( 1 ) )`. False (`0`) is the default value.

Example

```
rc = R Submit("\[
  x <- rnorm(5)
  print(x)
  y <- rnorm(5)
  print(y)
  z = plot(x, y)
]\" );
```

R Submit File("pathname")

Description

Submits statements to R using a file pointed in the specified *pathname*.

Returns

0 if successful; nonzero otherwise.

Argument

`Pathname` A quoted string that contains the pathname to the file that contains the R code to be executed.

`R Term()`

Description

Terminates the currently active R integration interface.

Returns

Returns 0 if the termination is successful and -1 otherwise.

Arguments

None

Random Functions

`Col Shuffle(<By var,...>)`

Description

Creates a random ordering of the row numbers of the current data table when used in a column formula.

Note: This function is generally used in a column formula.

Returns

A random integer between 1 and the number of rows in the current data table.

Argument

`By var` (Optional) A By variable enables you to randomly order the rows within the groups of the By variable values.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Shuffle", Numeric, Continuous, Set Formula( Col Shuffle() )
  );
```

This example creates a column formula that shuffles the order of the row numbers (1 to 40) each time the formula is evaluated. Each number appears only once.

`Make Validation Formula(rates, <<Stratification Columns(cols), <<Grouped Columns(cols), <<Cutpoint Column(col))`

Description

Generates a validation column. This function is primarily used by the Make Validation Column utility.

Arguments

`rates` Vector of three rates that specify the training, validation, and test rates, respectively.

`<<Stratification Columns` Assigns one or more stratification columns.

`<<Grouped Columns` Assigns one or more grouping columns.

`<<Cutpoint Column` Assigns a numeric cutpoint column.

Predictive and Specialized Modeling

`Random Beta(alpha, beta, <theta=0>, <sigma=1>)`

Description

Returns a random number from a beta distribution with two shape parameters, *alpha* and *beta*, and optional parameters *theta* and *sigma*.

Arguments

`alpha, beta` Shape parameters α and β, which must both be greater than 0.

`theta` Optional threshold parameter θ. The default is 0.

`sigma` Optional scale parameter σ, which must be greater than 0. The default is 1.

`Random Beta Binomial(n, p, <delta=0>)`

Description

Returns a random number from a beta binomial distribution for *n* trials with probability *p* and overdispersion parameter *delta*.

Arguments

`n` The number of trials, which must be greater than or equal to 2. If the specified *n* is not an integer, the non-integer part is truncated.

`p` The probability of success for each trial, which must be between 0 and 1.

`delta` The overdispersion parameter δ, which must be between Maximum[$-p/(n-p-1)$, $-(1-p)/(n-2+p)$] and 1. The default is 0.

`Random Binomial(n, p)`

Description

Returns a random number from a binomial distribution with *n* trials and probability *p* of the event of interest occurring.

Arguments

p The probability of success for each trial, which must be between 0 and 1.

n The number of trials.

`Random Category(probA, resultA, probB, resultB, <..., ...,> resultElse)`

Description

Returns one of the specified result expressions at random, chosen from pairs of probability and result expressions. A random uniform number is generated and compared to the *prob* arguments to determine which *result* argument is returned.

Arguments

probA Numeric value between 0 and 1 that represents the probability of the corresponding result expression being returned.

resultA Expression that corresponds to *probA*.

resultElse Expression that is returned if no previous result expression has been returned.

`Random Cauchy()`

Description

Returns a random number from a Cauchy distribution with a median of zero.

`Random ChiSquare(df, <nc=0>)`

Description

Returns a random number from a chi-square distribution with given *df* (degrees of freedom) and optional noncentrality parameter.

Arguments

df The degrees of freedom *n*, which must be greater than 0.

nc Optional noncentrality parameter λ, which must be nonnegative. The default is 0.

`Random Exp()`

Description

Returns a random number from an exponential distribution with scale parameter equal to 1. Equivalent to the negative log of `Random Uniform`.

`Random F(dfnum, dfden, <noncentral=0>)`

Description

Returns a random number from an F distribution with a given *dfnum*, *dfden*, and optional noncentrality parameter.

Arguments

dfnum The degrees of freedom, v_1, of the chi-square distribution in the numerator of the *F*-distribution. *dfnum* must be greater than 0.

dfden The degrees of freedom, v_2, of the chi-square distribution in the denominator of the *F*-distribution. *dfden* must be greater than 0.

noncentral Optional noncentrality parameter λ, which must be nonnegative. The default is 0.

Random Frechet(<mu=0>, <sigma=1>)

Description

Returns a random number from a Fréchet distribution with the location *mu* and scale *sigma*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Random Gamma(alpha, <scale=1>)

Description

Returns a random numbers from a gamma distribution for given *alpha* and optional *scale*.

Arguments

alpha The shape parameter α, which must be greater than 0.

scale Optional scale parameter β, which must be greater than 0. The default is 1.

Random Gamma Poisson(lambda, <sigma=1>)

Description

Returns a random number from a gamma Poisson distribution with parameters *lambda* and *sigma*.

Arguments

lambda The shape parameter λ, which much be greater than 0.

sigma Optional overdispersion parameter σ, which must be greater than or equal to 1. The default is 1. When the overdispersion parameter is 1, the distribution reduces to a Poisson(λ) distribution.

Random GenGamma(<mu=0>, <sigma=1>, <lambda=0>)

Description

Returns a random number from an extended generalized gamma distribution with parameters *mu*, *sigma*, and *lambda*.

Arguments

`mu` Optional location parameter μ. The default is 0.

`sigma` Optional scale parameter σ, which must be greater than 0. The default is 1.

`lambda` Optional shape parameter λ. The default is 0.

`Random Geometric(p)`

Description

Returns a random number from the geometric distribution with probability *p* that a specific event occurs at any one trial.

`Random GLog(mu, sigma, lambda)`

Description

Returns a random number from a generalized logarithmic distribution with parameters *mu*, *sigma*, and *lambda*.

Arguments

`mu` The location parameter μ.

`sigma` The scale parameter σ, which must be greater than 0.

`lambda` A shape parameter λ, which must be greater than 0.

`Random Index(n, k)`

Description

Returns a *k* by 1 matrix of random integers between 1 and *n* with no duplicates.

`Random Integer(n)`

`Random Integer(k, n)`

Description

Returns a random integer from 1 to *n* or from *k* to *n*.

`Random Johnson Sb(gamma, delta, theta, sigma)`

Description

Returns a random number from a Johnson Sb distribution with parameters *gamma*, *delta*, *theta*, and *sigma*.

Arguments

`gamma` Shape parameter γ.

`delta` Shape parameter δ, which must be greater than 0.

`theta` Location parameter θ.

`sigma` Scale parameter σ, which must be greater than 0.

Random Johnson Sl(gamma, delta, theta, <sigma=1>)

Description

Returns a random number from a Johnson Sl distribution with parameters *gamma*, *delta*, *theta*, and optional *sigma*.

Arguments

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Optional parameter σ that indicates if the distribution is skewed positively or negatively. sigma must be equal to either +1 (skewed positively) or -1 (skewed negatively). The default is +1.

Random Johnson Su(gamma, delta, theta, sigma)

Description

Returns a random number from a Johnson Su distribution with parameters *gamma*, *delta*, *theta*, and *sigma*.

Arguments

gamma Shape parameter γ.

delta Shape parameter δ, which must be greater than 0.

theta Location parameter θ.

sigma Scale parameter σ, which must be greater than 0.

Random LEV(<mu=0>, <sigma=1>)

Description

Returns a random number from an LEV distribution with the location *mu* and scale *sigma*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Random LogGenGamma(<mu=0>, <sigma=1>, <lambda=0>)

Description

Returns a random number from a log generalized gamma distribution with parameters *mu*, *sigma*, and *lambda*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

lambda Optional shape parameter λ. The default is 0.

Random Logistic(<mu=0>, <sigma=1>)

Description

Returns a random number from a logistic distribution with location *mu* and scale *sigma*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Random Loglogistic(<mu=0>, <sigma=1>)

Description

Returns a random number from a loglogistic distribution with location *mu* and scale *sigma*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Random Lognormal(<mu=0>, <sigma=1>)

Description

Returns a random number from a lognormal distribution with location *mu* and scale *sigma*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Random Multivariate Normal(mean, covar, <nrows=1>)

Description

Returns a random vector from a multivariate normal distribution with mean vector *mean* and covariance matrix *covar*. To generate multiple vectors, specify an integer greater than 1 for the *nrows* argument. When *nrows* is greater than 1, the return value is a matrix. The number of columns in the random vector or matrix is equal to the number of rows in the *covar* argument.

Arguments

mean Mean vector for the multivariate normal distribution.

covar Covariance matrix for the multivariate normal distribution. This matrix must be a symmetric square matrix that contains the same number of columns as the mean vector.

nrows Optional argument that specifies the number of random vectors returned. The default number of rows is 1.

Random Negative Binomial(n, p)

Description

Returns a random number from a negative binomial distribution for *n* successes with probability of success *p*.

Random Normal(<mu=0>, <sigma=1>)

Description

Returns a random number from a normal distribution with mean *mu* and standard deviation *sigma*.

Arguments

mu Optional location parameter μ. The default is 0.

sigma Optional scale parameter σ, which must be greater than 0. The default is 1.

Random Normal Mixture(meanvec, sdvec, probabvec)

Description

Returns a random number from a normal mixture distribution with the specified arguments.

Arguments

meanvec A vector that contains group means.

sdvec A vector that contains the group standard deviations.

probvec A vector that contains the group probabilities.

Random Poisson(lambda)

Description

Returns a random number from a Poisson distribution with shape parameter *lambda*.

Arguments

lambda The shape parameter λ, which much be greater than 0.

Random Reset(seed)

Description

Restarts the random number sequences with *seed*.

Notes

You can use the **Random Reset** function to set a random seed that enables you to reproduce a set of random numbers. The random number algorithms in JMP are sometimes refined, and therefore the results might not be consistent across different versions of JMP.

Random Seed State(<seed state>)

Description

Retrieves or restores the random seed state to or from a BLOB object.

Random SEV(<mu=0>, <sigma=1>)

Description

Returns a random number from an SEV distribution with the specified location *mu* and scale *sigma*.

Arguments

`mu` Optional location parameter μ. The default is 0.

`sigma` Optional scale parameter σ, which must be greater than 0. The default is 1.

Random SHASH(gamma, delta, theta, sigma)

Description

Returns a random number from a sinh-arcsinh (SHASH) distribution with parameters *gamma*, *delta*, *theta*, and *sigma*.

Arguments

`gamma` The shape parameter γ.

`delta` The shape parameter δ, which must be greater than 0.

`theta` The location parameter θ.

`sigma` The scale parameter σ, which must be greater than 0.

Random Shuffle(matrix)

Description

Returns the matrix with the elements shuffled into a random order.

Random t(df, <noncentral=0>)

Description

Returns a random number from a t distribution with the specified *df* (degrees of freedom). The noncentrality argument may be negative or positive. The default value of *noncentral* is 0.

`Random Triangular(min, mode, max)`

`Random Triangular(mode, max)`

`Random Triangular(mode)`

Description

Generates a random number from a triangular distribution between 0 and 1 with the *mode* that you specify. The triangular distribution is typically used for populations that have a small number of data.

Arguments

`min` Specifies the lower limit of the triangular distribution. The default value is 0.

`mode` Specifies the mode of the triangular distribution.

`max` Species the upper limit of the triangular distribution. The default value is 1.

Notes

If you specify only the mode, the minimum value is 0, and the maximum value is 1. If you specify the mode and maximum value, the minimum value is 0 by default.

`Random Uniform()`

`Random Uniform(x)`

`Random Uniform(min, max)`

Description

Generates a random number from a uniform distribution between 0 and 1. `Random Uniform(x)` generates a number between 0 and x. `Random Uniform (min, max)` generates a number between *min* and *max*. The result is an approximately even distribution.

`Random Weibull(shape, <scale=1>)`

Description

Returns a random number from a Weibull distribution with parameters *shape* and optional *scale*.

Arguments

`shape` Shape parameter β, which must be greater than 0.

`scale` Optional scale parameter α, which must be greater than 0. The default is 1.

`Resample Freq(<rate=1, <column>>)`

Description

Generates a frequency count for sampling with replacement. If no arguments are specified, the function generates a 100% resample.

Note: This function is generally used in a column formula.

Arguments

`rate` (Optional) Specifies the rate of resampling. The default value is 1. A negative value specifies that fractional frequencies are allowed.

`column` (Optional) If you specify *`column`*, you must also specify *`rate`*. The sample size is calculated by the rate multiplied by the sum of the specified column. If rate is negative, then the sample size is the negative of the rate multiplied by the sum of the specified column. If you do not specify a column, the generated frequencies sum to the number of rows.

Example

To ensure that the numbers in the frequency column match each time you run the script, use `As Constant()`. `As Constant()` evaluates an expression to create a constant value that does not change after it has been computed.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dc = dt << New Column( "column",
   Formula(
      As Constant(
         Random Reset( 123 );
         0;
      ) + Resample Freq()
   )
);
dc << Eval Formula;
```

Notes

– A typical use of this function generates a column with many 1s, some 0s, some 2s, and so forth, corresponding to which rows were randomly assigned any of *n* randomly selected rows.

– A typical use of this with an *existing* frequency column produces a new frequency column whose values are similar to the old frequency column (have the same expected value); however, the values vary somewhat due to random selection at the rates corresponding to the old frequency column.

Row Functions

As Table(matrix, <matrix 2, ...>, < <<invisible >, < <<private >, < <<Column Names({list}) >)

Description

Creates a new data tables from the *`matrix`*.

Returns

The new data table.

Argument

`matrix` Any matrix.

`<<invisible` Creates an invisible data table that hides the table from view but lists it in the JMP Home Window and Window menu.

`<<private` Hides the table completely. Creating a private data table speeds the process of getting to the data; it does not save the computer from allocating the memory necessary to hold the data table data.

`<<Column Names(list)` The list specified column names for the data. The argument is a list of quoted column names.

`Col Stored Value(<dt>, col, <row>)`

Description

Returns the data values stored in the column and disregards values assigned through column properties (such as Missing Value Codes).

Arguments

`dt` Optional reference to a data table. If this value is not supplied, the current data table is used.

`col` Name of the column.

`row` (Optional) Row name or number. If this value is not specified, the current row is used.

Example

Suppose that the Missing Value Codes column property is assigned to the `x1` column to treat "999" as a missing value. Another column includes a formula that calculates the mean. To use the value "999" instead of a missing value to calculate the mean, use `Col Stored Value()` in the formula:

```
Mean( Col Stored Value( :x1 ), :x2, :x3 )
```

`Column(<dt>, "name", "formatted")`

`Column(<dt>, n)`

Description

Gets a reference to the data table column.

Arguments

`dt` Optional reference to a data table. If this is not supplied, the current data table is used.

`name` A quoted string that is the name of the column.

`formatted` A quoted string that returns the formatted string of the cell value.

`n` The column number.

Column Name(n)

Description

Determines the name of the column specified by number.

Returns

The name of the n^{th} column as an expression (not a quoted string).

Argument

`n` The number of a column.

Count(from, to, step, times)

Description

Used for column formulas. Creates row by row the values beginning with the *from* value and ending with the *to* value. The number of *steps* specifies the number of values in the list between and including the *from* and *to* values. Each value determined by the first three arguments of the count function occurs consecutively the number of *times* that you specify. When the *to* value is reached, `Count` starts over at the *from* value. If the *from* and *to* arguments are data table column names, `Count` takes the values from the first row only. Values in subsequent rows are ignored.

Returns

The last value.

Arguments

`from` Number, column reference, or expression. `Count` starts counting with this value.

`to` Number, column reference, or expression. `Count` stops counting with this value.

`step` Number or expression. Specifies the number of steps to use to count between *from* and *to*, inclusive.

`times` Number or expression. Specifies the number of times each value is repeated before the next step.

Examples

/* the rows in the column named *colname* are filled with the series 0, 3, 6, 0,

```
... until all rows are filled */
For Each Row(:colname[row()]=count(0, 6, 3, 1))

/* the rows in the column named colname are filled with the series 0, 0, 3,
   3, 6, 6, 0, ... until all rows are filled */
For Each Row(:colname[row()]=count(0, 6, 3, 2))
```

Notes

`Count()` is dependent on `Row()`, and is therefore mainly useful in column formulas.

Current Data Table(<dt>)

Description

Without an argument, gets the current (topmost) data table. With an argument, sets the current data table.

Returns

Reference to the current data table.

Argument

`dt` Optional name of or reference to a data table.

Notes

Private tables cannot be made current with `Current Data Table()`.

Data Table(n)

Data Table("name")

Get Data Table(<project(title|index|box|window),> name|index)

Description

Gets reference to the *n*th open data table or the table with the given *name* in a global variable.

Returns

Reference to the specified data table.

Argument

`n` Number of a data table.

`name` Quoted string, name of a data table.

Dif(col, n)

Description

Calculates the difference of the value of the column *col* in the current row and the value *n* rows previous to the current row.

Returns

The difference.

Arguments

`col` A column name (for example, `:age`).

`n` A number.

Dim(<dt|matrix>)

Description

Returns a row vector with the dimensions of the current data table, a specified data table, or a matrix. The dimensions are the number of rows and the number of columns and are listed in that order.

Arguments

`dt` A data table.

`matrix` A matrix.

Notes

If no argument is specified, the dimensions of the current data table are returned.

Get Data Table List(<Project(title|index|box|window>)

Description

Returns a list of all open data tables.

Notes

Use `Project(0)` to specify no project when running the expression in a project.

Lag(col, n)

Description

Returns for each row the value of the column *n* rows previous.

N Row(dt); NRow(matrix)

N Rows(dt); NRows(matrix)

Description

Returns the number of rows in the data table given by *dt* or in the *matrix.*

N Table()

Description

Returns the number of open data tables. Private tables are not included.

New Column("name", <"data type">, <"modeling type">, <Width(n)>, Format("format", width, precision), <Formula()>, <Set Values>, <Like(column reference)>, <actions>)

Description

Adds a new column named *"name"* after the last column in *dt*. Unless otherwise specified, columns are numeric, continuous, and 12 characters wide.

Returns

A column reference.

Notes

Can also be used as a message: `dt<<New Column`.

The `Like()` argument copies the data type, modeling type, format, formula, and other properties from the reference column into the new column.

See Also

"dt<<New Column(`name`, `<data type>`, `<modeling type>`, <Format(`format`, `width`)>, <Formula()>, <Set Values(`{..., ..., }`)>, <Set Property`(properties)`>)"

`New Table("name", <visibility("invisible" | "private" | "visible")>, <actions>)`

Description

Creates a new data table with the specified *`name`*.

Arguments

`name` A quoted string that contains the name of the new table.

`visibility` Optional quoted keyword. `invisible` hides the data table from view but lists it in the JMP Home Window and Window menu. `private` hides the table completely. `visible` shows the data table. `"visible"` is the default value.

Note: Creating a private data table speeds the process of getting to the data; it does not save the computer from allocating the memory necessary to hold the data table data.

`actions` Optional argument that can define the new table.

`Row()`

`Row() = y`

Description

Returns or sets the current row number. No argument is expected.

`Sequence(from, to, <step size>, <repeat times>)`

Description

Produces an arithmetic sequence of numbers across the rows in a data table. The *`step size`* and *`repeat times`* arguments are optional, and the default value for both is 1.

Subscribe to Data Table List(<subscriber name|"">,
<OnOpen(function)|OnClose(function)|OnRename(function)>)

Description

Subscribes to the data table list. You will be notified when a new data table has been added, closed, or renamed.

Subscript(a, b, c)

list[i]

matrix[b, c]

Description

Subscripts for lists extract the *i*th item from the *list*, or the *b*th row and the *c*th column from a *matrix*.

Suppress Formula Eval(Boolean)

Description

Turns off automatic calculation of formulas for all data tables.

Unsubscribe to Data Table List(<subscriber name>,
<"OnOpen"|"OnClose"|"All">)

Description

Removes a subscription to the data table list that has been added through Subscribe to Data Table List().

Row State Functions

As Row State(i)

Description

Converts *i* into a row state value.

Returns

A row state from the *i* given.

Argument

i an integer

Color Of(rowstate)

Description

Returns or sets the color index.

Returns

The color index of *rowstate*.

Argument

`rowstate` a row state argument

Example

Set the color of the fifth row to red.

```
Color Of( Rowstate( 5) ) = 3
```

Color State(i)

Description

Returns a row state with the color index of *i*.

Returns

A row state.

Argument

`i` index for a JMP color

Combine States(rowstate, rowstate, ...)

Description

Generates a row state combination from two or more row state arguments.

Returns

A single numeric representation of the combined row states.

Arguments

`rowstate` Two or more row states.

Excluded(rowstate)

Description

Returns or sets an excluded index.

Returns

The excluded attribute, 0 or 1.

Argument

`rowstate` One or more row states.

`Excluded State(num)`

Description

Returns a row state for exclusion from the *num* given.

`Hidden(rowstate)`

Description

Returns or sets the hidden index.

`Hidden State(num)`

Description

Returns a row state for hiding from the *num* given.

`Hue State(num)`

Description

Returns a hue state from the *num* given.

`Labeled(rowstate)`

Description

Returns or sets the labeled index.

`Labeled State(num)`

Description

Returns a labeled state from the *num* given.

`Marker Of(rowstate)`

Description

Returns or sets the marker index of a row state.

`Marker State(num)`

Description

Returns a marker state from the *num* given.

`Row State(<dt,> <n>)`

Description

Returns the row state changed from the initial condition of the active row or the *n*th row.

Arguments

`dt` Optional positional argument: a reference to a data table. If this argument is not in the form of an assignment, then it is considered a data table expression.

`n` The row number.

Example

The following example creates the data table references and then returns the row state of row 1 in Big Class.jmp:

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt2 = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
Row State( dt1, 1 );
```

Selected(rowstate)

Description

Returns or sets the selected index.

Selected State(num)

Description

Returns a selected state from the *num* given.

Shade State(num)

Description

The Shade State function assigns 5 shade levels to a color or hue.

SAS Integration Functions

As C Expr(x)

Description

Returns a C programming language representation of the expression.

Returns

A quoted string.

As JavaScript Expr(x)

Description

Returns a JavaScript representation of the expression.

Returns

A quoted string.

As JSON Expr(x)

Description

Returns a JSON (JavaScript Object Notation) representation of the expression.

Returns

A quoted string.

As Python Expr(x)

Description

Returns a Python representation of the expression.

Returns

A quoted string.

As SAS Expr(x)

Description

Converts an expression to a version that is more suitable for SAS DATA step. The code must be wrapped in a PROC DS2 call. Use `Expr(...)` for literal expressions. Use `NameExpr(name)` for expressions stored in a variable. Otherwise, the expression returns the expression to convert.

Returns

A quoted string.

Current Metadata Connection()

Description

Returns the active SAS metadata server connection, if any, as a scriptable object.

Current SAS Connection()

Description

Gets the active global SAS server connection, if any, as a scriptable object.

Get SAS Version Preference()

Description

Returns the SAS version selected in the SAS Integration page of the Preferences as a quoted string.

`JMP6 SAS Compatibility Mode(Boolean)`

Description

Setting this to 1 (true) causes SAS operators to operate in a mode compatible with JMP 6 capabilities.

`Meta Connect(<"machine", port>, <"authDomain">, <"username">, <"password">, <named arguments>)`

`Meta Connect(<Profile("profile name")>, <Password("password")>, <named arguments>)`

`Meta Connect(<Environment("environment name")>, <named arguments>)`

Description

Connects to a SAS Metadata Server. If no arguments are specified, an empty connection window appears. If some arguments are specified, a window partially filled in with the argument values appears. If all arguments are specified, the connection is made and no window appears.

Returns

`1` if connection is successful, `0` if not.

Arguments

`machine` (Optional) A quoted string that contains the DNS name of the machine.

`port` Required if machine is specified. A quoted string or integer that contains the port on which the metadata server listens.

`authDomain` (Optional) A quoted string that contains the authentication domain for the credentials supplied. Not necessary unless *`username`* and *`password`* are included.

`username` (Optional) A quoted string that contains the user name for the connection.

`password` (Optional) A quoted string that contains the password for the connection.

Optional Named Arguments

`Profile("profile_name")` A quoted string that contains the name of the metadata server connection profile from which connection information should be retrieved.

`Environment("environment_name")` A quoted string that contains the name of the WIP environment from which connection information should be retrieved.

`Password("password")` A quoted string that contains the password for the specified profile name.

`CheckPreferenceOnly(0|1)` If specified, `Meta Connect` returns the status of the **I want to connect to a SAS Metadata Server** option in the SAS Integration page of JMP Preferences. If that box is checked, **Meta Connect** returns 1; if not, 0.

`Repository(`*`string`*`)` Takes a quoted string that contains the name of the repository to which to connect.

`ProfileLookup(0|1)` If *machine* and *port* are specified rather than a profile name, and `ProfileLookup` is specified, an attempt is made to find a metadata server connection profile with a machine name and port matching those provided. If one is found, other connection information (such as authentication domain, user name, and password) is obtained from that profile.

`Prompt(Always|Never|IfNeeded)` Takes one of the keywords `Always` (always prompt before attempting to connect), `Never` (never prompt, just fail), or `IfNeeded` (the default; prompt if connection with the given arguments fails).

`SASVersion("<version number>" <,Strict>)` Attempts to change the SAS version preference to the specified value before making the metadata server connection. If the SAS version is already locked to a different version than the one specified, the SASVersion argument will fail. By default, if the SAS version cannot be set, JMP will try the metadata server connection. However, if you include `Strict` as the second argument, the inability to change the SAS version will be treated as an error, and JSL processing will stop. If you do not include `Strict`, the SAS version argument is treated as a hint and will set the version preference if it can. JMP will still try to connect if the version cannot be set. The order you put these arguments in can make a difference. The attempt to change the SAS Version is made immediately when that argument is encountered. That can affect the validity of other arguments, particularly for `MetaConnect`. Valid values for `SASVersion` are "9.3" and "9.4". Note: Using the `SASVersion` argument has the same effect as changing the SAS Server Version on the SAS Integration Preferences page.

Notes

- If no arguments are included and if no profile is saved, the Connect to SAS Metadata Server window appears.
- If you connect to a physical workspace server, there is no metadata server involved, so metadata security is never applied. You must connect to a SAS Metadata Server and then connect to a logical workspace server. Then metadata security is enforced on the metadata-defined libraries you access.

`Meta Create Profile("profile", <named arguments>)`

Description

Creates a metadata server connection profile and adds it to the current user's set of saved metadata server connection profiles.

Returns

1 if *profile* was successfully created, otherwise 0.

Arguments

`profile` A quoted string that contains the name of the created profile. If a profile by the given name already exists, `MetaCreateProfile` fails unless *Replace* is specified.

Optional Named Arguments

`HostName("name")` A quoted string that contains the name of the host computer running the SAS Metadata Server that this profile will connect to.

`Port(n)` The port number (*n*) that the SAS Metadata Server is listening for connections on.

`AuthenticationDomain("domain") | AuthDomain("domain")` A quoted string that sets the authentication domain to use for the connection.

`Description("desc") | Desc("desc")` A quoted string that sets a description for this profile.

`Password("password")` A quoted string that contains the password to store in this profile.

`Replace(0|1)` If *name* matches a profile that already exists, `Replace` must be specified for the existing profile to be replaced by the one provided. The default value is False (0).

`UserName("username")` A quoted string that contains the user name that this profile uses to connect to the SAS Metadata Server.

`UseSingleSignOn(0|1)` If specified, this profile attempts to use Single Sign-On (currently also known as Integrated Windows Authentication) to connect to the SAS Metadata Server. This option is valid only for connecting to SAS 9.3 or higher Metadata Servers. If `UseSingleSignOn` is True(1), *UserName* and *Password* cannot be specified. The default value is False (0).

`Meta Delete Profile("name")`

Description

Deletes the named metadata server connection profile from the current user's set of saved metadata server connection profiles

Returns

1 if profile was successfully deleted, otherwise 0.

Argument

`name` A quoted string that contains the name of the profile to delete.

`Meta Disconnect()`

Description

Disconnect the current SAS Metadata Server connection, if any.

Returns

Void.

Meta Get Environments()

Description

Returns a list of the SAS Environments that are defined in the SAS Environments definition file, which is configured in the SAS Preferences.

Meta Get Repositories()

Description

Gets a list of the repositories available on the current SAS Metadata Server connection.

Returns

A list of repository names as quoted strings.

Meta Get Servers()

Description

Get a list of the SAS Servers that are registered in the SAS Metadata Repository to which the session is currently connected.

Returns

A list of server names as quoted strings.

Meta Get Stored Process("path")

Description

Get a stored process object from the currently connected SAS Metadata Repository.

Returns

Stored Process scriptable object.

Arguments

`path` Quoted string that is the path to the stored process in metadata, starting at the BIP Tree.

Meta Is Connected()

Description

Determines whether a current connection to a SAS Metadata Server exists.

Returns

`1` if a connection exists; `0` otherwise.

Arguments

None.

`Meta Set Repository("repositoryName")`

Description

Set the SAS Metadata Repository to use for metadata searches.

Returns

1 if setting the repository was successful, 0 otherwise.

Arguments

`repositoryName` A quoted string that contains the name of the repository to make current.

`SAS Assign Lib Refs("libref", "path", <"engine">, <"engine options">)`

Description

Assign a SAS libref on the active global SAS server connection.

Returns

1 if successful, 0 otherwise.

Arguments

`libref` A quoted string that contains a library reference (8-character maximum) to assign.

`path` A quoted string that contains the full path on the SAS server to the library being assigned.

`engine` Optional, quoted string that contains the engine for the SAS server to use when accessing members of this library.

`engine options` Optional, quoted string that contains the options needed for the engine being used.

`SAS Connect(<"machine_name">, <"port">, <named_arguments>)`

Description

Connect to a local, remote, or logical SAS server.

Returns

SAS Server scriptable object.

Arguments

`machine_name` (Optional) A quoted string that can contain a physical machine name or the name of a metadata-defined (logical) server. In the first case, the port must be provided. In the second case, a port must *not* be provided. If neither *`name`* nor *`port`* are included, and JMP is running on Windows, a connection to SAS on the local machine (via COM) is attempted, and all named arguments are ignored.

`port` (Optional) A quoted string or integer. If *`name`* is a physical machine name, this is the port on that machine to connect to. If *`name`* is a metadata-defined (logical) server, *`port`* must *not* be included.

Optional Named Arguments

`UserName("name")` A quoted string that contains the user name for the connection.

`Password("password")` A quoted string that contains the password for the connection.

`ReplaceGlobalConnection(0|1)` A Boolean. The default value is `True`. If `True`, and a successful SAS server connection is made, this connection replaces the active SAS connection that becomes the target of other global SAS JSL function calls. If `False`, the global SAS connection is not changed, and the returned SASServer scriptable object should be used to send messages to this server connection.

`ShowDialog(0|1)` A Boolean. The default value is `False`. If `True`, other arguments (except `ReplaceGlobalConnection`) are ignored and the SAS Server Connection window appears. This provides the JSL programmer a way to open the SAS Connect window.

`Prompt(Always|Never|IfNeeded)` A keyword. `Always` means always prompt before attempting to connect. `Never` means never prompt even if the connection attempt fails (just fail and send an error message to the log), and `IfNeeded` (the default value) means prompt if the attempt to connect with the given arguments fails (or is not possible with the information given).

`ConnectLibraries(0|1)` A Boolean. Defaults to the SAS Integration Preference setting governing whether to automatically connect metadata-defined libraries when connecting to a SAS server. If true, all metadata-defined libraries are connected at SAS server connection time, which can be time-consuming. If false, metadata-defined libraries are not connected. To connect specific libraries later, use the `SAS Connect Libref` global function or `Connect Libref` message to a SAS server object.

`SASVersion("<version number>" <,Strict>)` Attempts to change the SAS version preference to the specified value before making the metadata server connection. If the SAS version is already locked to a different version than the one specified, the SASVersion argument will fail. By default, if the SAS version cannot be set, the metadata server connection will still be tried. However, if you include `Strict` as the second argument, the inability to change the SAS version will be treated as an error and JSL processing will stop. If you do not include `Strict`, the SAS version argument is treated as a hint and will set the version preference if it can, but if it cannot it will still try to connect. The order you put these arguments in can make a difference. The attempt to

change the SAS Version is made immediately when that argument is encountered.

That can affect the validity of other arguments, particularly for MetaConnect. Valid values for `SASVersion` are “9.3” and “9.4”. Note: Using the `SASVersion` argument has the same effect as changing the SAS Server Version on the SAS Integration Preferences page.

Example

```
// prompt for login credentials
Meta Connect( "dev.company.com", 28561 );
```

```
sas = SAS Connect( "SASApp" );

// dump some libraries and data sets to the JMP log
Show( sas << Get Librefs() );
Show( sas << Get Data Sets( "Chocolate Enterprises 2017" ));

sas << Import Data( "Chocolate Enterprises 2017", "Products" );

/* The preceding lines produce the following text in the JMP log and import
   the data set: */
sas << Get Librefs:{"BOOKS", "CHOCOENT", "EGSAMP", "TEST", "TST92", "MAPS",
   "SASDATA", "SASHELP", "SASUSER", "SGFDATA", "TEMPDATA", "V6LIB", "WORK"}
sas << Get Data Sets("Chocolate Enterprises 2008"):{"CHOC_DATA",
   "CHOC_SURVEY", "CUSTOMERS", "ORDER_DETAIL", "PRODUCTS", "SALES_ANALYSIS",
   "SALES_SUMMARY"}
```

`Get Librefs` returns the short library names, not the longer logical names. However, you can use either one, and metadata security will still be applied for metadata-defined libraries.

SAS Connect Lib Refs(libref)

Description

Connects a SAS libref on the active SAS server connection.

Returns

1 if successful and 0 otherwise.

SAS Deassign Lib Refs("libref")

Description

De-assign a SAS libref on the active global SAS server connection.

Returns

1 if successful; 0 otherwise.

Arguments

`libref` A quoted string that contains the library reference to de-assign.

SAS Disconnect()

Description

Disconnect the active global SAS connection, if any.

Returns

1 if a SAS connection exists and was successfully disconnected, 0 otherwise.

Arguments

None.

`SAS Export Data(dt, "library", "dataset", <named_arguments>)`

Description

Exports a JMP data table to a SAS data set in a library on the active global SAS server connection.

Returns

`1` if the data table was exported successfully; `0` otherwise.

Arguments

`dt` data table or a reference to a data table.

`"library"` the library to which to export the data table.

`"dataset"` the name of the new SAS data set.

Optional Named Arguments

`Columns(list)|Columns(col1, col2, ...)` A list of columns or a comma-separated list of columns.

`Password("password")` A quoted string that contains the password to serve as the READ, WRITE, and ALTER password for the exported SAS data set. If the exported data set is replacing an existing data set with an ALTER password, this password is used as the ALTER password for overwriting the data set. If *`Password`* is specified, values for *`ReadPassword`*, *`WritePassword`*, and *`AlterPassword`* are ignored.

`ReadPassword("password")` A quoted string that contains the password to serve as the READ password for the exported SAS data set.

`WritePassword("password")` A quoted string that contains the password to serve as the WRITE password for the exported SAS data set.

`AlterPassword("password")` A quoted string that contains the password to serve as the ALTER password for the exported SAS data set. If the exported data set is replacing an existing data set with an ALTER password, this password is used as the ALTER password for overwriting the data set.

`PreserveSASColumnNames(0|1)` A Boolean. If true *and* the JMP data table originally came from SAS, the original SAS column names are used in the exported SAS data set. The default value is `False`.

`PreserveSASFormats(0|1)` A Boolean. If true *and* the JMP data table originally came from SAS, the original SAS formats and informats are applied to the columns in the exported SAS data set. The default value is `True`.

`ReplaceExisting(0|1)` A Boolean. If true, an existing SAS data set with the specified name in the specified library is replaced by the exported SAS data set. If false, a data set with the specified name already exists in the specified library; the export is stopped. The default value is false.

SaveJMPMetadata(0|1) Includes SAS 9.4 Extended Attributed to store JMP metadata (such as table script and column properties). Default is 0 (disabled).

HonorExcludedRows(0|1) A Boolean. If true, any rows in the JMP data table that are marked as excluded are not exported. The default value is false.

Notes

Information about the export is sent to the log.

SAS Get Data Sets("libref")

Description

Returns a list of the data sets defined in a SAS library.

Returns

A list of quoted strings.

Arguments

libref A quoted string that contains the SAS libref or friendly library name associated with the library for which the list of defined SAS data sets is returned.

SAS Get File("source", "dest", "encoding")

Description

Get a file from the active global SAS server connection. JMP creates a FILENAME statement (with an encoding, if specified) and uses it to read the file on the SAS server.

Returns

1 if successful, 0 otherwise.

Arguments

source A quoted string that contains the full path of file on the server to be downloaded to the client machine.

dest A quoted string that contains the full path on the client machine for where to put the copy of the file downloaded from the server.

encoding A quoted string that contains the encoding used in the file (for example, "utf-8"). The server must support the specified encoding.

SAS Get File Names("fileref")

Description

Get a list of filenames found in the given fileref on the active global SAS server connection.

Returns

A list of quoted strings.

Arguments

`fileref` A quoted string that contains the name of the fileref from which to retrieve filenames.

`SAS Get File Names In Path("path")`

Description

Get a list of filenames found in the given path on the active global SAS server connection.

Returns

A list of quoted strings.

Arguments

`path` A quoted string that contains the directory path on the server from which to retrieve filenames.

`SAS Get File Refs()`

Description

Get a list of the currently defined SAS filerefs on the active global SAS server connection.

Returns

List of two lists. The first list is a list of quoted strings of fileref names. The second is a corresponding list of strings of physical names.

`SAS Get Lib Refs(<named arguments>)`

Description

Get a list of the currently defined SAS librefs on the current global SAS server connection.

Returns

A list of quoted strings.

Named Arguments

`Friendly Names(0|1)` Optional, Boolean. If `True`, then for any libraries that have friendly names (metadata-defined libraries), the friendly name is returned rather than the 8-character libref.

`SAS Get Log()`

Description

Retrieve the SAS Log from the active global SAS server connection.

Returns

A quoted string.

SAS Get Output()

Description

Retrieve the listing output from the last submission of SAS code to the current global SAS server connection.

Returns

A quoted string.

SAS Get Results()

Description

Retrieve the results of the previous SAS Submit as a scriptable object, which allows significant flexibility in what to do with the results.

Returns

A SAS Results Scriptable object.

SAS Get Var Names(*string*, <"dataset">, <password("password")>)

Description

Retrieves the variable names contained in the specified data set on the current global SAS server connection.

Returns

A list of quoted strings.

Arguments

`string` A quoted string that contains one of the following:

- The name of the SAS Library containing the SAS data set to be imported. In that case, the *`dataset`* name argument is required.
- The full member name of the SAS data set to be imported, in the form "libname.membername".
- The SAS Folders tree path to a logical SAS data table to be imported. This option requires a connection to a SAS 9.3 or higher Metadata Server.

`dataset` (Optional) A quoted string that contains the name of the data set from which to retrieve variable names.

`password("password")` (Optional) A quoted string that contains the read password for the data set. If this is not provided and the data set has a read password, the user is prompted to enter it.

SAS Import Data(*string*, <"dataset">, <named arguments>)

Description

Import a SAS data set from the active global SAS server connection into a JMP table.

Returns

JMP Data Table object.

Arguments

`string` A quoted string that contains one of the following:

- The name of the SAS Library containing the SAS data set to be imported. In that case, the *"dataset"* name argument is required. The name can be a friendly metadata library name or a SAS 8-character library name.
- The full member name of the SAS data set to be imported, in the form "libname.membername".
- The SAS Folders tree path to a logical SAS data table to be imported. This option requires a connection to a SAS 9.3 or higher Metadata Server.

`dataset` (Optional) A quoted string that contains the name of the data set.

Optional Named Arguments

`Columns("list")|Columns(col1, col2, ...)` A quoted string list or multiple strings that contain the names of columns to include in the import.

`ConvertCustomFormats(0|1)` The default value is True (1). If True and custom formats are found in the SAS data set being imported, an attempt is made to convert the SAS custom formats to JMP value labels for those columns.

`Invisible(0|1)` The default value is False (0). If true, the JMP data table is hidden from view. The data table appears only in the JMP Home Window and the Window menu. Hidden data tables remain in memory until they are explicitly closed, reducing the amount of memory that is available to JMP. To explicitly close the hidden data table, call `Close(dt)`, where *dt* is the data table reference returned by SASImportData.

`Where("filter")` A quoted string that contains the filter to use when importing data, as in `Where("salary<50000")`.

`Password("password")` A quoted string that contains the read password for the data set. If this is not provided and the data set has a read password, the user is prompted to enter it.

`UseLabelsForVarNames(0|1)` If `True`, the labels from the SAS data set become the column names in the resulting JMP table. If `False`, the variable names from the SAS data set become the column names in the JMP table. The default value is `False`.

`RestoreJMPMetadata(0|1)` Includes SAS 9.4 Extended Attributed to store JMP metadata. Default is 0 (disabled).

`Sample(named arguments)` optional, named. Allows a random sample of the SAS data set to be imported into JMP. If both Where and Sample are specified, the WHERE clause is used to filter the SAS data set first, and then a random sample of the resulting rows is

taken based on the arguments supplied to Sample. Note that Sample uses PROC SURVEYSELECT on the SAS server, which is available only if the SAS/STAT package is licensed and installed on that server. The documentation for PROC SURVEYSELECT might be helpful in understanding how sampling is performed. By default (if no arguments are supplied), a 5% simple random sample is taken. Here are the available arguments (all optional):

- `Simple | Unrestricted`: If `Simple` is specified, sampling is performed without replacement. If `Unrestricted` is specified, sampling is performed with replacement. These two options are mutually exclusive and only one can be specified.
- `SampleSize(int) | N(int)`: Total number of rows for the sample, or number of rows per strata level for stratified sampling
- `SampleRate(number) | Rate(number) | Percent(number)`: Specifies the sampling rate. For stratified sampling, the rate is applied to each strata level. Note that the supplied value is assumed to be a percentage, so `SampleRate(3.5)` means a 3.5% sampling rate.
- `Strata({col1, col2, ...}) | Strata(col1, col2, ...)`: Perform stratified random sampling using the column names supplied as Strata variables.
- `NMin(int)`: Minimum number of rows (either overall or per strata level for stratified sampling) to return. Only applies to rate-based sampling.
- `NMax(int)`: Maximum number of rows (either overall or per strata level for stratified sampling) to return. Only applies to rate-based sampling.
- `Seed(int)`: Number to use as the seed for sampling. Useful for replicating the same sample. By default, the seed is a random number based on time of day. See PROC SURVEYSELECT documentation.
- `OutputHits(0|1)`: Boolean; the default value is false. When doing Unrestricted sampling, if the same row of the input data set is selected more than once, by default that row still appears only once in the resulting JMP data table, with the NumberHits column indicating the number of times that the row was selected. Setting OutputHits to true causes an input row that is selected multiple times to appear multiple times in the resulting JMP data table.
- `SelectAll(0|1)`: Boolean, the default value is true. If `SelectAll` is true, PROC SURVEYSELECT selects all stratum rows whenever the stratum sample size exceeds the total number of rows in the stratum. If `SelectAll` is false and PROC SURVEYSELECT finds a case where the stratum sample size exceeds the total number of rows in a given stratum, an error results and sampling fails. `SelectAll` only applies to Simple random sampling.

`SQLTableVariable(0|1)` If `True`, an SQL table variable is created in the resulting JMP table that shows the SQL that was submitted to SAS to obtain the data. If `False`, the SQL table variable is not created. The default value is `True`. If an SQL table variable is created and the data set required a read password, the password is masked.

SAS Import Query("sqlquery", <named arguments>)

Description

Execute the requested SQL query on the current global SAS server connection, importing the results into a JMP data table.

Returns

JMP Data Table object.

Arguments

`sqlquery` A quoted string that contains the SQL query to perform and from which to import the result.

Optional Named Arguments

`ConvertCustomFormats(0|1)` The default value is true. If true and custom formats are found in the SAS data set being imported, an attempt is made to convert the SAS custom formats to JMP value labels for those columns.

`Invisible(0|1)` The default value is false. If true, the JMP data table is hidden from view. The data table appears only in the JMP Home Window and the Window menu. Hidden data tables remain in memory until they are explicitly closed, reducing the amount of memory that is available to JMP. To explicitly close the hidden data table, call `Close(dt)`, where *`dt`* is the data table reference returned by `SAS Import Query`.

`UseLabelsForVarNames(0|1)` The default value is true. If `True`, the labels from the SAS data set become the column names in the resulting JMP table. If `False`, the variable names from the SAS data set become the column names in the JMP table.

`RestoreJMPMetadata(0|1)` Includes SAS 9.4 Extended Attributes to store JMP metadata. Default is 0 (disabled).

`SQLTableVariable(0|1)` The default value is true. If `True`, an SQL table variable is created in the resulting JMP table that shows the SQL that was submitted to SAS to obtain the data. If `False`, the SQL table variable is not created. If an SQL table variable is created and the data set required a read password, the password is masked.

SAS Is Connected()

Description

Discovers whether there is an active global SAS server connection.

Returns

`1` if an active global SAS connection exists, `0` otherwise.

SAS Is Local Server Available()

Description

Returns True if a local SAS Server is available; otherwise, returns False.

SAS Load Text File("path")

Description

Download the file specified in path from the active global SAS server connection and retrieve its contents as a quoted string.

Returns

A quoted string.

Arguments

"path" A quoted string that contains the full path on the server of the file to download and retrieve the contents as a string.

SAS Name("name")

SAS Name({list of names})

Description

Converts JMP variable names to SAS variable names by changing special characters and blanks to underscores and various other transformations to produce a valid SAS name.

Returns

A quoted string that contains one or more valid SAS names, separated by spaces.

Argument

"name" A quoted string that represents a JMP variable name; or a list of quoted JMP variable names.

SAS Open For Var Names("path")

Description

Opens a SAS data set only to obtain the names of its variables, returning those names as a list of quoted strings.

Returns

A list of variable names in the file.

Argument

path A quoted string that is a pathname of a SAS data set.

SAS Send File("source", "dest", "encoding")

Description

Send a file from the client machine to the active global SAS server connection. JMP creates a FILENAME statement (with an encoding, if specified) and uses it to save the file on the SAS server.

Returns

1 if successful, 0 otherwise.

Arguments

`source` A quoted string that contains the full path of the file on the client machine to be uploaded to the server.

`dest` A quoted string that contains the full path on the server that receives the file uploaded from the client machine.

`encoding` A quoted string that contains the encoding used in the file (for example, "utf-8"). The server must support the specified encoding.

`SAS Submit("sasCode", <named arguments>)`

Description

Submit some SAS code to the active global SAS server connection.

Returns

1 if successful, 0 otherwise.

Arguments

`sasCode` A quoted string that contains the SAS code to submit.

Optional Named Arguments

`Async(0|1)` A Boolean. If True (1), the submit occurs asynchronously (in the background). Use the `Get Submit Status()` message on the SAS Server Scriptable Object to determine the status of the submit. The default value is False (0).

`ConvertCustomFormats(0|1)` A Boolean. When SAS data sets generated by submitted SAS code are imported into JMP after the submit completes (see `Open Output Datasets`), the value of `ConvertCustomFormats` determines whether an attempt is made to convert any custom formats found on columns in the SAS data to JMP value labels. The default value is True (1).

`DeclareMacros(var1, var2, ...)` JSL variable names. Provides a simple way to pass the values of JSL variables to SAS as macro variables. Each JSL variable specified should evaluate to a quoted string or numeric value. Fully qualified JSL variables names, only the variable name is sent to SAS. For example, `namespace:variable_name` becomes `variable_name` in SAS.

`GetSASLog(<Boolean|OnError>, <JMPLog|Window>)` A Boolean. If no arguments are supplied, the SAS Log is retrieved and displayed in the location indicated in SAS Integration Preferences. The first argument to `GetSASLog` can be either a Boolean value or the keyword OnError. If a Boolean value is supplied, true means display the SAS Log, and false means not to display it. OnError instructs JMP to only show the SAS Log if an error occurred in the submit. The second argument to GetSASLog tells JMP where to display the SAS Log. If JMPLog is specified, the SAS Log is appended to the JMP Log. If Window is specified, the SAS Log is opened in a separate window.

`GraphicsDevice(string) or GDevice(string)` A quoted string that specifies a value for the GDEVICE SAS option to be used for graphics generated by the submitted SAS code. The value must be a valid SAS graphics device. The default value is determined in Preferences.

`Interactive(0|1)` JMP includes the `QUIT` statement in the generated wrapper code. Interactive PROCs work even if JMP is generating the ODS wrapper. On every SUBMIT, specify the argument that is part of an interactive sequence. Otherwise, QUIT will be generated in both the prologue-generated and epilogue-generated code.

`NoOutputWindow` A Boolean. If `True`, the SAS Output window containing the listing output from the submission does not appear. The default value is `False`.

`ODS(0|1)` If true, additional SAS code is submitted causing ODS results to be generated for the submitted SAS code. The default value is determined in Preferences.

`ODSFormat(string)` A quoted string that determines the format of generated ODS results. Valid values are "HTML", "RTF", and "PDF". The default value is determined in Preferences.

`ODSGraphics(0|1)` If true, ODS statistical graphics are generated for the submitted SAS code. Setting `ODSGraphics` to true causes ODS to also be set to true. The default value is determined in Preferences.

`ODSStyle(string)` A quoted string that specifies the ODS Style to use when generating ODS results. *`String`* must be a valid SAS Style. The default value is determined in Preferences.

`ODSStyleSheet(path)` A quoted string that specifies a local CSS style sheet to use when formatting generated ODS results. *`Path`* must be a path to a CSS file valid for the client machine (the machine running JMP). The default value is determined in Preferences.

`OnSubmitComplete(script)` A quoted string that specifies a JSL script that should be run when the submit completes. This is especially useful for asynchronous submits. If script is the name of a defined JSL function, that function is executed, with the SAS Server scriptable object passed as the first argument.

`OpenODSResults(0|1)` If true, ODS results that are generated by the submitted SAS code (due to ODS being true) are automatically opened after the submit completes. The default value is True (1).

`OpenOutputDatasets(<All|None|dataset1, dataset2, ...>)` JMP detects when submitted SAS code creates new SAS data sets. `OpenOutputDatasets` (which can be abbreviated `OutData`) determines what, if anything, is done with those data sets with the SAS Submit completes. If `All` is specified, all data sets generated by the SAS code are imported into JMP when the SAS Submit completes. If `None` is specified, none of the generated data sets are imported. If there are specific data sets known to be generated by the submitted SAS code that you want to be imported into JMP when the SAS submit completes, you can alternative provide their names, and only the requested data sets are imported. The default value is determined in Preferences.

Title(*string*) A quoted string that specifies the window title to use for the window that displays ODS output from the submit.

SAS Submit File("filename", <named arguments>)

Description

Submit a SAS code file to the active global SAS server connection.

Returns

1 if successful; 0 otherwise.

Arguments

filename A quoted string that contains the name of file containing SAS code to submit.

Named Arguments

Same as for SAS Submit.

SQL Functions

Note: Database table names that contain the characters $# -+/%()&|;? are not supported.

As SQL Expr(x, <style>)

Description

Converts an expression to code that you can use in an SQL Select statement. Use Expr(...) for literal expressions. Use NameExpr(name) for expressions stored in a variable. Otherwise, the expression returns the expression to convert.

Returns

A quoted string that contains the expression converted to valid SQL syntax for use in an SQL Select statement.

```
New SQL Query(Connection
("ODBC:connection_string")|("SAS:connection_string"),
Select(Column("column", "t1")), From(Table("table", <Schema("schema")>,
<Alias("t1")>)), <Options(JMP 12 Compatible(1)|JMP 13 Compatible(1)|Run on
Open(1)))>

New SQL
Query(Connection("ODBC:connection_string;")|("SAS:connection_string;"),
```

```
Custom("SELECT col1, col2, col3 FROM table;")), <Options(JMP 12
Compatible(1)|JMP 13 Compatible(1)|Run on Open(1))>
```

Creates an SQL Query object for the specified connection, columns, data table, or for the custom SQL query.

Returns

A data table that contains the queried data. The data table includes the quoted SQL query string and table scripts for modifying and updating the query.

Arguments

`Connection` The quoted string for an ODBC or SAS connection.

`Select` The column that you want to select and its alias.

`From` The table that is queried and the optional schema and column alias.

`Custom` An SQL statement that selects columns from the specified table.

`Version` The minimum JMP version required to open the query. If this condition is not met, a message regarding compatibility is written to the log, and the query does not open.

`Options` Boolean. `JMP 12 Compatible` is included in generated scripts when you select the Query Builder preference to create a JMP 12 compatible option or select the corresponding Query Builder red triangle menu option. The option enables JMP 12 users to run a JMP 13 query that might contain compatibility issues. Include `Run on Open(1)` to run the query when opened rather than opening the query in edit mode.

Example

```
New SQL Query(
   Connection(
      "ODBC Connection String..."
   ),
   QueryName( "g6_Movies" ),
   Select( Column( "ItemNo", "t1" ), Column( "LengthMins", "t1" ), Column(
   "Genre", "t1" ) ),
   From( Table( "g6_Movies", Schema( "SQBTest" ), Alias( "t1" ) ) )
) << Run Background( On Run Complete( dt = queryResult ) );

Show( dt );
```

Notes

- Query Builder creates a symbol called `queryResult` in the context of an `On Run Complete()` script. This is a reference to the data table imported by the query. `queryResult` enables you to assign a global variable to the table for later use.
- `New SQL Query()` always closes the ODBC connection after performing the query.
- After you run `New SQL Query()`, the database connection that you make appears in the Database > Open Table window as an available connection. But the connection you see in the Database > Open Table window is due to JMP keeping track of how to connect to that database; no ODBC connection is left open.

- There is not a command to close an ODBC connection established with `New SQL Query()` because `New SQL Query()` always closes the connection after performing the query.
- After you run `New SQL Query()`, the database connection appears in the Database > Open Table window as an available connection. JMP is keeping track of how to connect to that database; the ODBC connection was not left open.

```
Query(<<dt1|Table(dt1, alias1)>, ..., <dtN, aliasN)>>, <private |
invisible>, <scalar>, sqlStatement )
```

Description

Performs a SQL query on selected data tables.

Returns

The result of the query, either a data table or a single value.

Arguments

`dt1, dtN` (Optional) A variable that has been assigned to the data table.

`Table` (Optional) Passes a reference to the data table.

`alias1, aliasN` Specifies the alias of the database table.

`private` (Optional) Avoids showing the resulting data table. Using a private data table speeds the process of getting to the data; it does not save the computer from allocating the memory necessary to hold the data table data.

`invisible` (Optional) Hides the resulting data table from view. The data table appears only in the JMP Home Window and the Window menu. Hidden data tables remain in memory until they are explicitly closed, reducing the amount of memory that is available to JMP. To explicitly close the hidden data table, call `Close(dt)`, where *`dt`* is the data table reference.

`scalar` (Optional) Indicates that the query returns a single value.

`sqlStatement` Required. The SQL statement, most likely a SELECT statement. The statement must be the last argument.

Example

The following example selects all data for students who are older than 14 years of age.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
result = Query( Table( dt, "t1" ), "SELECT * FROM t1 WHERE age > 14;" );
```

See Also

Appendix A, “SQL Functions Available for JMP Queries”

Statistical Functions

Arc Finder(X(col), Y(col), Group(lot, wafer))

Description

Finds arcs in the point data and creates a new column that identifies the arcs.

Example

```
dt = Open( "$SAMPLE_DATA/Wafer Stacked.jmp" );
Arc Finder(
   Group( :Lot, :Wafer ),
   X( :X_Die ),
   Y( :Y_Die ),
   Min Distance( 12 ), // minimum distance among 3 points to seed an arc
   Min Radius( 15 ), // minimum radius of the acceptable arc
   Max Radius( 2000 ), // maximum radius of acceptable arc
   Max Radius Error( 2 ), // how close a point needs to be added
   Min Arc Points( 5 ), // how many points to define an arc
   Number of Searches( 500 ), // how many random probes of data
   Max Number Arcs( 3 ) // number of arcs searched for
);
dt << Color or Mark by Column( :Arc Number );
dt << Graph Builder(
   Size( 1539, 921 ),
   Variables( X( :X_Die ), Y( :Y_Die ), Wrap( :Lot_Wafer Label ), Color( :Arc
   Number ) ),
   Elements( Points( X, Y, Legend( 6 ) ) )
);
```

Notes

- The function is scaled for data that have a range of 30 to 50 units.
- The function is suitable only for data that are subset to the interesting defect points.
- It is not suitable when the density of points is high.

ARIMA Forecast(column, length, model, estimates, from, to)

Description

Determines the forecasted values for the specified rows of the specified column using the specified model and estimates.

Returns

A vector of forecasted values for *column* within the range defined by *from* and *to*.

Arguments

`column` A data table column.

`length` Number of rows within the column to use.

`model` Messages for Time Series model options.

`estimates` A list of named values that matches the messages sent to `ARIMA Forecast()`. If you perform an ARIMA Forecast and save the script, the estimates are part of the script.

`from, to` Define the range of values. Typically, *`from`* is between 1 and *`to`*, inclusive. If *`from`* is less than or equal to 0, and if *`from`* is less than or equal to *`to`*, the results include filtered predictions.

`Best Partition(xindices, yindices, <<Ordered, <<Continuous Y, <<Continuous X)`

Description

Experimental function to determine the optimal grouping.

Returns

A list.

Arguments

`xindices, yindices` Same-dimension matrices.

`Col Cumulative Sum(name, <By var, ...>)`

`Cumulative Sum(name)`

Description

Returns the cumulative sum for the current row. `Col Cumulative Sum` supports By columns, which do not need to be sorted.

Arguments

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

`Col Maximum(name, <By var, ...>)`

`Col Max(name)`

Description

Calculates the maximum value across all rows of the specified column. The result is internally cached to speed up multiple evaluations.

Returns

The maximum value that appears in the column.

Arguments

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

`Col Mean(name, <By var, ...>)`

Description

Calculates the mean across all rows of the specified column. The result is internally cached to speed up multiple evaluations.

Returns

The mean of the column.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

`Col Median(name, <By var, ...>)`

Description

Calculates the median across all rows of the specified column. The ordering is cached internally to speed up multiple evaluations.

Returns

The median of the column.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

`Col Minimum(name, <By var, ...>)`

`Col Min(name)`

Description

Calculates the minimum value across all rows of the specified column. The result is internally cached to speed up multiple evaluations.

Returns

The minimum value that appears in the column.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

`Col Moving Average(name, options, <By var, ...>)`

`Moving Average(name, options)`

Description

Returns the moving average over a given interval based at the current row. `Col Moving Average` supports By columns.

Arguments

`name` A column name.

`Weighting(1|0|n)` Required positional argument. Determines how the values are weighted. 1 indicates uniform weighting. 0 indicates incremental weighting (a ramp or triangle). Any other number is the parameter for an exponential moving average (EWMA or EMA).

`Before(1|0|n)` Positional argument. Controls the size of the range (or window) by including the specified number of items before the current item in the average (in addition to the current item). The default value, -1, means all of the preceding items.

`After(1|0|n)` Positional argument. Controls the size of the range (or window) by including the specified number of items after the current item in the average (in addition to the current item). The default value, 0, means no following items.

`Partial Window is Missing` Boolean positional argument. Controls how missing values are treated. By default, missing values are ignored. 0 computes the average of partial windows.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Examples

```
// equal weighting of a five-item lagging range
Col Moving Average( x, 1, 4 );

// ramp weighting of all preceding items
Col Moving Average( x, 0 );

// triangle weighting of a five-item centered range
Col Moving Average( x, 0, 2, 2 );

// exponential weighting of all preceding items
Col Moving Average( x, 0.25 );
```

Col N Missing(name, <By var, ...>)

Description

Calculates the number of missing values across all rows of the specified column. The result is internally cached to speed up multiple evaluations.

Returns

The number of missing values in the column.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

Col Number(name, <By var, ...>)

Description

Calculates the number of nonmissing values across all rows of the specified column. The result is internally cached to speed up multiple evaluations.

Returns

The number of nonmissing values in the column.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

Col Quantile(name, p, <ByVar>)

Description

Calculates the specified quantile *p* across all rows of the specified column. The result is internally cached to speed up multiple evaluations.

Returns

The value of the quantile.

Argument

`name` A column name.

`p` A specified quantile *p* between 0 and 1.

`ByVar` (Optional) A By group.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Col Quantile( :height, .5 );
   63
```

63 is the 50th percentile, or the median, of all rows in the height column.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

Col Rank(column, <ByVar, ...>, <<tie("average"|"arbitrary"|"row"|"minimum"))

Description

Ranks each row's value, from 1 for the lowest value to the number of columns for the highest value. Ties are broken arbitrarily.

Arguments

`column` The column to be ranked.

`ByVar` (Optional) A By variable to compute statistics across groups of rows.

`<<tie` Determines how the tie is broken. A tie occurs when the values being ranked are the same. For the data `[33 55 77 55]`, `33` has rank 1 and 77 has rank 4, and the question is how to assign ranking for the `55`s. `average` reports the average of the possible rankings, 2.5, for both 55s. `arbitrary` matches JMP 12 behavior by assigning the possible rankings in an unspecified order, which could be 2 and 3 or 3 and 2. `row` assigns the ranks in the order that they originally appear. (The first 55 would be 2 and the second 55 would be 3.)

`minimum` gives both values the lowest possible rank, 2.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

Col Simple Exponential Smoothing(column, alpha, <ByVar>)

Description

Returns the simple exponential smoothing prediction for the current row using smoothing weight *alpha*.

Arguments

`column` The column of time series observations.

`alpha` The smoothing weight.

`ByVar` (Optional) A By variable to compute predictions across groups of rows. By variables do not need to be presorted.

Notes

The predicted value for row *t* is given by the following:

```
Predicted[t] = alpha * Observed[t-1] + (1-alpha) * Predicted[t-1]
```

By definition, `Predicted[1] = Observed[1]`.

Col Standardize(name,<By var, ...>)

Description

Calculates the column mean divided by the standard deviation across all rows of the specified column.

Returns

The standardized mean.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. If a By variable is specified, the values are standardized against the mean and standard deviation of their corresponding By variable group.

Notes

Standardizing centers the variable by its sample standard deviation. Thus, the following commands are equivalent:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "stdht", Formula( Col Standardize( height ) ) );
dt << New Column( "stdht2",
  Formula( (height - Col Mean( height )) / Col Std Dev( height ) )
);
```

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

Col Std Dev(name,<By var, ...>)

Description

Calculates the standard deviation across rows in a column. The result is internally cached to speed up multiple evaluations.

Returns

The standard deviation.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

`Col Sum(name,<By var, ...>)`

Description

Calculates the sum across rows in a column. Calculating all missing values (`Col Sum(.,.)`) returns missing. The result is internally cached to speed up multiple evaluations.

Returns

The sum.

Argument

`name` A column name.

`By var` (Optional) A By variable to compute statistics across groups of rows. Use the By variable in a column formula or in a `For Each Row()` function.

Notes

If a data value is assigned by a column property (such as Missing Value Codes), use `Col Stored Value()` to base the calculation on the value stored in the column instead.

See Also

"Col Stored Value(<dt>, col, <row>)"

```
Fit Censored(Distribution("name"), YLow(vector) | Y(Vector),
<YHigh(vector)>, <Weight(vector)>, <X(matrix)>, <Z(matrix)>,
<HoldParm(vector)>, <Use random sample to compute initial values(percent)>,
<Use first N observations to compute initial values(nobs)>)
```

Description

Fits a distribution using censored data.

Returns

A list that contains parameter estimates, the covariance matrix, the log-likelihood, the AICc, the BIC, and a convergence message. See *Fitting Linear Models*.

Arguments

`Distribution("name")` The quoted name of the distribution to fit.

`YLow(vector) | Y(Vector)` If you do not have censoring, then use `Y` and an array of your data, and do not specify `YHigh`. If you do have censoring, then specify `YLow` and `YHigh` as the lower and upper censoring values, respectively.

Optional Arguments

`YHigh(vector)` A vector that contains the upper censoring values. Specify this only if you have censoring and also specify YLow.

`Weight(vector)` A vector that contains the weight values.

`X(matrix)` The regression design matrix for location.

`Z(matrix)` The regression design matrix for scale.

`HoldParm(vector)` An array of specified parameters. The parameters should be nonmissing where they are to be held fixed, and missing where the are to be estimated. This is primarily used to test hypotheses that certain parameters are zero or some other specific value.

`Use random sample to compute initial values(percent)` A percent of the observations to be used in the computation of the initial values. Specify this if the data vector is large.

`Use first N observations to compute initial values(nobs)` A number of observations at the start of the data vector to be used in the computation of the initial values. Specify this if the data vector is large.

Fit Circle(Xvec, Yvec)

Description

Fits a circle that best goes through three or more points using a least squares approach. If only three points are specified, a direct solution can be found, and the sum of squared errors is zero.

Returns

A list that contains the X and Y coordinates of the center point of the circle, the length of the radius, and the sum of squared errors.

Arguments

`Xvec` Vector of X coordinates of three or more points.

`Yvec` Vector of Y coordinates of three or more points.

Syntax

```
{Xcenter, yCenter, radius, SSE} = Fit Circle(Xvec, Yvec)
```

Hier Clust(x)

Description

Returns the clustering history for a hierarchical clustering using Ward's method (without standardizing data).

Argument

`x` A data matrix.

IRT Ability(Q1, <Q2, Q3, ... Qn,> parmMatrix)

Description

Returns scores for the latent variable in an item response theory model with n binary items and a matrix of known parameters. The parameter matrix should contain as many rows as there are parameters in the model and as many columns as there are items in the analysis.

Arguments

`Q1, Q2, ..., Qn` A set of n binary items.

`parmMatrix` A matrix of parameters from an item response theory model.

`KDE(vector, <named arguments>)`

Description

Returns a kernel density estimator with automatic bandwidth selection.

Argument

`vector` A vector.

Optional Named Arguments

`<<weights` Must be a vector of the same length as vector, and can contain any nonnegative real numbers. `Weights` represents frequencies, counts, or similar concepts.

`<<bandwidth(n)` A nonnegative real number. Enter a value of 0 to use the bandwidth selection argument.

`<<bandwidth scale(n)` A positive real number.

`<<bandwidth selection(n)` *n* must be 0, 1, 2, or 3, corresponding to Sheather and Jones, Normal Reference, Silverman rule of thumb, or Oversmoother, respectively.

`<<kernel(n)` *n* must be 0, 1, 2, 3, or 4, corresponding to Gaussian, Epanechnikov, Biweight, Triangular, or Rectangular, respectively.

`LenthPSE(x)`

Description

Returns Lenth's pseudo-standard error of the values within a vector.

Argument

`x` A vector.

`Max()`

See "Maximum(var1, var2, ...)".

`Maximum(var1, var2, ...)`

`Max(var1, var2, ...)`

Description

Returns the maximum value of the arguments or of the values within a single matrix or list argument. If multiple arguments are specified, they must be all numeric values or all quoted strings.

`Mean(var1, var2, ...)`

Description

Returns the arithmetic mean of the arguments or of the values within a single matrix or list argument.

`Median(var1, var2, ...)`

Description

Returns the median of the arguments or of the values within a single matrix or list argument.

`Min()`

See "Minimum(var1, var2, ...)".

`Minimum(var1, var2, ...)`

`Min(var1, var2, ...)`

Description

Returns the minimum value of the arguments or of the values within a single matrix argument. If multiple arguments are specified, they must be either all numeric values or all quoted strings.

`N Missing(expression)`

Description

Rowwise number of missing values in variables specified.

`Number(var1, var2, ...)`

Description

Rowwise number of nonmissing values in variables specified.

`Product(i=initialValue, limitValue, bodyExpr)`

Description

Multiplies the results of *bodyExpr* over all *i* until the *limitValue* and returns a single product.

`Quantile(p, arguments)`

Description

Returns the p^{th} quantile of the arguments. The first argument can be a scalar or a matrix of values between 0 and 1. The remaining arguments can also be specified as values within a single matrix or list argument.

`Range(var1, var2, ...)`

Description

Returns the minimum and maximum values of the arguments. The result is returned as a two-element row vector that contains the minimum and the maximum.

`Robust PCA(X, <Lambda(2/sqrt(max(nrow, ncol)))>, <tolerance=1e-10>, <maxit(75)>, <Center(1)>, <Scale(1)>)`

Description

Performs a sequence of singular value decompositions and thresholding steps to decompose the data matrix into a low-rank matrix and a sparse matrix of residuals.

Returns

`A` The low-rank matrix estimation.

`E` The sparse matrix of residuals.

`S` A vector of singular values.

Arguments

`X` A data matrix.

`Lambda` Specifies a value greater than 0 that determines the sparsity of the matrix of residuals. For larger values of Lambda, the matrix of residuals is more sparse.

`tolerance` The convergence criterion.

`maxit` The maximum number of SVD iterations.

`Center` Centers the data prior to performing the SVD iterations.

`Scale` Scales the data prior to performing the SVD iterations

`Std Dev(var1, var2, ...)`

Description

Rowwise standard deviation of the variables specified.

`Sum(var1, var2, ...)`

Description

Rowwise sum of the variables specified. Calculating all missing values (`Sum(.,.)`returns missing.

`SSQ(x1, ...)`

Description

Returns the sum of squares of all elements. Takes numbers, matrices, or lists as arguments and returns a scalar number. Skips missing values.

`Summarize(<dt>, <by>, <count>, <sum>, <mean>, <min>, <max>, <stddev>, <corr>, <quantile>, <first>)`

Description

Gathers summary statistics for a data table and stores them in global variables.

Returns

None.

Arguments

`dt` Optional positional argument: a reference to a data table. If this argument is not in the form of an assignment, then it is considered a data table expression.

All other arguments are optional and can be included in any order. Typically, each argument is assigned to a variable so you can display or manipulate the values further.

`name=By(col | list | Eval)` Using a BY variable changes the output from single values for each statistic to a list of values for each group in the BY variable.

`Summarize YByX(X(<x columns>, Y (<y columns>), Group(<grouping columns>), Freq(<freq column>), Weight(<weight column>))`

Description

Calculates all Fit Y by X combinations on large-scale data sets.

Returns

A data table of *p*-values and LogWorth values for each Y and X combination.

Arguments

`X(col)` The factor columns used in the fit model.

`Y(col)` The response columns used in the fit model.

`Group(gcol)` The group of columns used in the fit model.

`Freq(col)` The frequency (for each row) column used in the fit model.

`Weight(col)` The importance (or influence) column used in the fit model.

Notes

Performs the same function as the Response Screening platform.

`Summation(init, limitvalue, body)`

Description

Summation sums the results of the *body* statement(s) over all *i* to return a single value.

`Tolerance Limit(1-alpha, p, n)`

Description

Constructs a *1-alpha* confidence interval to contain proportion *p* of the means with sample size *n*.

Transcendental Functions

`Arrhenius(n)`

Description

Converts the temperature *n* to the value of explanatory variable in Arrhenius model.

Returns

11604.5181215503/(n+273.15)

Argument

`n` Temperature in Celsius.

Notes

This is frequently used as a transformation.

`Arrhenius Inv(n)`

Description

The inverse of the Arrhenius function. Converts the value *n* to the temperature in Celsius.

Returns

(11604.5181215503/n)-273.15

Argument

`n` The value of the converted explanatory variable in Arrhenius model.

Notes

This is frequently used as a transformation.

`Beta(a, b)`

Description

$$\frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$$

Returns

Returns the beta function.

Arguments

`a, b` numbers

`Cytometry Logicle(x, T, W, M, A)`

Description

Computes a cytometry logicle transformation. For more information about the logicle transformation, see Moore and Parks (2012).

`Cytometry Logicle Inverse(y, T, W, M, A)`

Description

Computes the inverse cytometry logicle transformation. For more information about the logicle transformation, see Moore and Parks (2012).

`Digamma(n)`

Description

The derivative of the log of the gamma function (`LGamma`).

Returns

The digamma function evaluated at *n*.

Argument

`n` A number

`Exp(a)`

Description

Raises e to the power *a*.

Returns

e^a.

Argument

`a` A number

Equivalent Expression

`e()^a`

`ExpM1(x)`

Description

Returns a more accurate calculation of *Exp(x)-1* when *x* is very small.

`Factorial(n)`

Description

Multiplies all numbers 1 through *n*, inclusive

Returns

The product.

Arguments

`n` Any integer

Notes

One and only one argument must be specified.

`FFT({list}, <named arguments>)`

Description

Conducts a Fast Fourier Transformation (FFT) on a list of matrices.

Returns

The function takes one matrix, or a list of matrices for complex numbers. The returned value is a list of two matrices with the same dimensions as the first argument.

Argument

`List` A list of one or two matrices. If one is provided, it is considered to be the real part. If two are provided, the first is the real part and the second is the imaginary part. Both matrices must have the same dimensions, and both must have more than one row.

Optional Named Arguments

`<<inverse(Boolean)` If true (1), an inverse FFT is conducted.

`<<multivariate(Boolean)` If true (1), a multivariate FFT is conducted. If false(0), a spatial FFT is conducted.

`<<scale(number)` Multiplies the return values by the specified *number*.

`Fit Transform To Normal(Distribution("name"), Y(vector), <Freq(vector))`

Description

Fits a transformation to normality for a vector of data. This includes the Johnson Sl, Johnson Sb, Johnson Su, and GLog distributions.

Returns

A list that contains parameter estimates, the covariance matrix, the log-likelihood, AICc, a convergence message, and the transformed values. See *Fitting Linear Models*.

`Gamma(t, <limit>)`

Description

The gamma function of *x*, or for each row if *x* is a column:

$$\Gamma(t) = \int_0^{\infty} x^{t-1} e^{-x} dx$$

Returns

The gamma.

Arguments

`t` a number or a column

`limit` optional limit. The default is infinity.

Notes

`Gamma(t, limit)` is the same integral as `Gamma(t)` but with the limit of integration that is defined instead of infinity.

`LGamma(t)`

Description

Returns the log gamma function for *t*, which is the natural log of gamma.

`Ln(n)`

Description

Returns the natural logarithm (base *e* logarithm) of *n*.

`Log(n, <base>)`

Description

Returns the natural logarithm (base *e* logarithm) of *n*. An optional second argument lets you specify a different base. For example, `Log(n,3)` for the base 3 logarithm of *n*. The Log argument can be any numeric expression. The expression `Log(e())` evaluates as 1, and `Log(32,2)` is 5.

`Log10(n)`

Description

Returns the common (base 10) logarithm of *n*.

`Log1P(n)`

Description

Same as Log(1 + *x*), except that it is more accurate when *x* is very small.

`Logist(x)`

Description

Returns 1/(1+Exp(-x)), which converts a number in the domain -∞...+∞ into range 0...1. The function is useful in logistic regression.

`Logist Percent(p)`

Description

Similar to the `Logist()` function but with the result scaled from 0 to 100.

`Logit(p)`

Description

Returns `log(p/(1-p))`.

`Logit Percent(p)`

Description

Similar to the `Logit()` function with the argument 0 to 100 rather than 0 to 1.

`N Choose K(n, k)`

Description

This function returns the number of n things taken k at a time ("n choose k") and is computed in the standard way using factorials, as $n!/(k!(n-k)!)$. For example, `NChooseK(5,2)` evaluates as 10.

Notes

This is implemented internally in JMP using lGamma functions. The result is not always an integer.

`Power(a, <b>)`

`a^b`

Description

Raises *a* to the power of *b*.

Returns

The product of *a* multiplied by itself *b* times.

Arguments

a Can be a variable, number, or matrix.

b (Optional) Can be a variable or a number.

Notes

For `Power()`, the second argument (*b*) is optional, and the default value is 2. `Power(a)` returns a^2.

`Root(n, <r>)`

Description

Returns the *r*th root of *n*, where *r* defaults to 2 for square root.

`SbInv(z, gamma, delta, theta, sigma)`

Description

Returns a transformation of a standard normal variable to a double bounded Johnson variable.

`SbTrans(x, gamma, delta, theta, sigma)`

Description

Returns a transformation of a double bounded Johnson variable to a standard normal variable.

`Scheffe Cubic(x1, x2)`

Description

Returns `x1*x2*(x1-x2)`. This function supports notation for cubic mixture models.

`SHASHInv(z, gamma, delta, theta, sigma)`

Description

Returns a transformation of a standard normal variable to a sinh-arcsinh (SHASH) variable. The transformation is calculated as σ*`sinh((arcsinh(z)-`γ`)/`δ`)+`θ.

`SHASHTrans(x, gamma, delta, theta, sigma)`

Description

Returns a transformation of a sinh-arcsinh (SHASH) variable to a standard normal variable. The transformation is calculated as `sinh(`γ+δ*`arcsinh((x-`θ`)/`σ`))`.

`SlInv(z, gamma, delta, theta, sigma)`

Description

Returns a transformation of a standard normal variable to a Johnson Sl variable.

`SlTrans(x, gamma, delta, theta, sigma)`

Description

Returns a transformation of a Johnson Sl variable to a standard normal variable.

`Sqrt(n)`

Description

Returns the square root of *n*.

`Squash(expr)`

Description

An efficient computation of the function 1/ [1 + exp(*expr*)].

`Squish(expr)`

Description

Equivalent to `Squash(-expr)`, or $1/(1+e^{-\text{expr}})$.

`SuInv(z, gamma, delta, theta, sigma)`

Description

Returns a transformation of a standard normal variable to an unbounded Johnson variable.

`SuTrans(x, gamma, delta, theta, sigma)`

Description

Returns a transformation of an unbounded Johnson variable to a standard normal variable.

`Trigamma()`

Description

Returns the trigamma function evaluated at *n*. The trigamma function is the derivative of the digamma function.

Trigonometric Functions

JMP's trigonometric functions expect all angle arguments in radians.

`ArcCosH(x)`

Description

Inverse hyperbolic cosine.

Returns

The inverse hyperbolic cosine of *x*.

Argument

x Any number, numeric variable, or numeric expression.

`ArcCosine(x)`

`ArCos(x)`

Description

Inverse cosine.

Returns

The inverse cosine of *x*, an angle in radians.

Argument

x Any number, numeric variable, or numeric expression.

`ArcSine(x)`

`ArSin(x)`

Description

Inverse sine.

Returns

The inverse sine of *x*, an angle in radians.

Argument

x Any number, numeric variable, or numeric expression.

`ArcSinH(x)`

Description

Inverse hyperbolic sine.

Returns

The inverse hyperbolic sine of *x*.

Argument

x Any number, numeric variable, or numeric expression.

`ArcTangent(x1, <x2=1>)`

`ArcTan(x1 <x2=1>)`

`ATan(x1 <x2=1>)`

Description

Inverse tangent.

Returns

The inverse trigonometric tangent of *x1/x2*, where the result is in the range -Pi()/2, Pi()/2.

Argument

`x1` Any number, numeric variable, or numeric expression.

`x2=1` Specifies *atan2*.

`ArcTanH(x)`

Description

Inverse hyperbolic tangent.

Returns

The inverse hyperbolic tangent of *x*.

Argument

`x` Any number, numeric variable, or numeric expression.

`CosH(x)`

Description

Hyperbolic cosine.

Returns

The hyperbolic cosine of *x*.

Argument

`x` Any number, numeric variable, or numeric expression.

`Cosine(x)`

`Cos(x)`

Description

Cosine.

Returns

The cosine of *x*.

Argument

`x` Any number, numeric variable, or numeric expression. The angle in radians.

`Sine(expr)`

`Sin(expr)`

Description

Returns the sine.

`SinH(expr)`

Description

Returns the hyperbolic sine.

`Tangent(expr)`

`Tan(expr)`

Description

Returns the tangent of an argument given in radians.

`TanH(expr)`

Description

Returns the hyperbolic tangent of its argument.

Utility Functions

`Add(a, b, ...)`

`a+b+...`

Description

Adds the values of the listed arguments. No arguments are changed.

Returns

The sum.

Arguments

For `Add()`, a comma-separated list of variables, numbers, or matrices.

For `a+b`, any number of variables, numbers, or matrices.

Notes

- Any number of arguments is permitted. If no argument is specified, `Add()` returns 0.
- `Add()` returns missing if any arguments are missing. To ignore missing values, use `Sum()`.

See Also

the *Scripting Guide*

See "Sum(var1, var2, ...)"

Beep()

Description

Produces an alert sound.

Returns

Null.

BLOB MD5(blob)

Description

Converts the *blob* argument into a 16-byte blob.

Notes

The 16-byte blob is the MD5 checksum, or the hash, of the source blob.

BLOB Peek(blob, offset, length)

Description

Creates a new blob from a subrange of bytes of the *blob* argument.

Returns

A blob object.

Arguments

`blob` a binary large object.

`offset` An integer that specifies how many bytes into the blob to begin construction. The first byte is at offset 0, the second byte at offset 1.

`length` An integer that specifies how many bytes to copy into the new blob, starting at the offset.

Build Information()

Description

Returns the build date and time, whether it's a release or debug build, and the product name in a quoted comma-delimited string.

Caption({*h*, *v*}, "text", <Delayed(seconds)>, <Font(font)>, <Font Size(size)>, <Text Color("color")>, <Back Color("color")>, <Spoken(Boolean)

Description

Displays a caption window at the location described by *{h, v}* that displays *text*. The caption can be delayed before being displayed by *seconds*, or can be spoken. You can also specify the font type, size, and color and background color.

Returns

Null.

Arguments

`{h, v}` a list with two values. *h* is the horizontal displacement from the top left corner of the monitor in pixels. *v* is the vertical displacement from the top left corner in pixels.

`text` A quoted string or a reference to a string that is to be displayed in the caption.

`Delayed(seconds)` *seconds* is optional delay before displaying the caption. Setting this option causes this caption and all subsequent captions to be delayed by the specified number of seconds.

`Font(font)` Specify the font type.

`Font Size(size)` Specify the font size.

`Text Color("color")` Specify the color of text.

`Back Color("color")` Specify the background color.

`Spoken(Boolean)` Causes *text* to be spoken as well as displayed. The current setting (on or off) remains in effect until switched by another `Caption` statement that includes a `Spoken` setting.

Datafeed()

See "Open Datafeed()".

Debug Break()

When the JSL Debugger is open, this function stops a JSL script from executing at that point in the script. This function is useful for tracking in the debugger under user-specified conditions. If the JSL Debugger is not running, this function does not execute.

Decode64 BLOB(*string*)

Description

Decodes a printable quoted string of base 64 text into a blob.

Returns

A blob.

Arguments

`string` A quoted base-64 encoded string.

Example

```
Decode64 BLOB( "dGhlIHF1aWNrIGJyb3duIGZveA==" );
   Char To BLOB( "the quick brown fox", "ascii~hex" )
```

Decode64 Double(*string*)

Description

Creates a floating point number from a quoted base-64 encoded string.

Returns

A floating point number.

Arguments

`string` A quoted base-64 encoded string.

Disable JMP Live URL(url)

Description

Specifies the URLs that users cannot publish to. This preference must be added to jmpStartAdmin.jsl.

See the Common Tasks chapter in the *Scripting Guide* for details about the location of jmpStartAdmin.jsl.

If a URL appears in both the `Disable JMP Server()` and `Enable JMP Server()` lists, publishing to the URL is disabled.

Example

```
Disable JMP Live URL( "https://public.jmp.com" )
```

Divide(a, b)

Divide(x)

a/b

Description

Divides *a* by *b*. If only one argument is given (`divide(x)`), divides 1 by *x*.

Returns

The quotient of *a*/*b*; or the reciprocal of *x* (1/*x*) if only one argument is provided.

Arguments

`a, b, x` Can be a variable, number, or matrix.

Notes

If both arguments are matrices, it does matrix division.

Empty()

Description

Used in the formula editor for making empty boxes.

Returns

Missing.

Arguments

None.

Enable JMP Live URL(url)

Description

Specifies the URLs that users can publish to. This preference must be added to jmpStartAdmin.jsl.

See the Common Tasks chapter in the *Scripting Guide* for details about the location of jmpStartAdmin.jsl.

If a URL appears in both the `Enable JMP Server()` and `Disable JMP Server()` lists, publishing to the URL is disabled.

Example

```
Enable JMP Live URL( "https://public.jmp.com" )
```

Encode64 BLOB(x)

Description

Encodes a blob into a printable quoted string of base 64 text.

Returns

A quoted base-64 encoded string.

Example

```
Encode64 BLOB( Char To BLOB( "the quick brown fox" ) );
   "dGhlIHF1aWNrIGJyb3duIGZveA=="
```

Encode64 Double(n)

Description

Creates a quoted base-64 encoded string from a floating point number.

Returns

A quoted base-64 encoded string.

Arguments

`n` A floating point number.

Faure Quasi Random Sequence(nDim, nRow)

Description

Generates a sequence of spacefilling quasi random numbers using the Faure sequence.

Get Addin("id")

Description

Retrieves a registered add-in by *id*.

Returns

A scriptable object for the add-in. Returns empty if no add-in with the specified ID was found.

Argument

"id" The ID of an installed add-in.

Get Addins()

Returns

A list of all registered add-ins.

Get Addr Info("address", <port>)

Description

Converts a name to its numeric address.

Returns

A list of quoted strings. The first element is the command (Get Addr Info). The second is the results (for example, "ok" if the command was successful). The third is a list of strings of information. Included in that information is the address that corresponds to the name that was supplied.

Arguments

address A quoted string that specifies the name (for example, "www.sas.com").

port The port of the address.

Get Clipboard()

Description

Returns text from the computer's clipboard. If the content is not text, the result is null.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Copy Table Script( "Distribution" );
s = Get Clipboard();
nw = New Window( "Script", Script Box( s ) );
```

`Get Name Info("address", <port>)`

Description

Converts a numeric address to its name.

Returns

A list of quoted strings. The first element is the command (`GetNameInfo`). The second is the results (for example, "ok" if the command was successful). The third is a list of strings of information. Included in that information is the port name that corresponds to the address that was supplied.

Arguments

`address` A quoted string that specifies the numeric address (for example, `"149.173.5.120"`).

`port` The port of the address.

`Get Platform Preferences(<platform <(option, ...)> ... >)`

`Get Platform Preference(<platform <(option, ...)> ... >)`

Description

Returns the preferences for the specified platforms.

Returns

A list of platform preferences.

Argument

`platform` (Optional) Specifies the platform name. If not specified, all platform preferences are returned. You can specify one or more preferences for a platform.

`option` (Optional) Specifies the preference value. If not specified, all platform preference values are returned.

Notes

Table 2.3 describes the syntax for getting platform preferences.

Table 2.3 Get Platform Preferences() Syntax

Syntax	Description
`Get Platform Preferences( )`	Returns the current option values for all platform preferences.
`Get Platform Preferences( Platform )`	Returns the current option values for the specified platform preferences.

Table 2.3 Get Platform Preferences() Syntax *(Continued)*

Syntax	Description
`Get Platform Preferences( Platform( Option ) )`	Returns the current option values for the specified platform.
`Get Platform Preferences( <<Changed )`	Returns the current option values that have changed for all platforms.
`Get Platform Preferences( Platform( <<Changed ) )`	Returns the current option values that have changed for all platform preferences.
`Get Platform Preferences( Platform( Option ( <<Changed ) ) )`	Returns the current option values that have changed for the specified platform.

Examples

Suppose that the user modified several platform preferences through the JMP Platforms window or a script.

```
Platform Preferences(
  Distribution( Set Bin Width( 2 ), Horizontal Layout( 1 ) ),
  Model Dialog( Keep Dialog Open( 1 ) ),
  Graph Builder( Legend Position( "Bottom" ) )
);
```

To return all of the modified platform preferences, use `Get Platform Preferences( <<Changed )`:

```
Get Platform Preferences( <<Changed );
  Platform Preferences(
    Distribution( Horizontal Layout( 1 ), Set Bin Width( 2, <<On ) ),
    Graph Builder( Legend Position( "Bottom", <<On ) ),
    Model Dialog( Keep dialog open( 1 ) )
  );
```

Get Preferences(<preference_name>)

Get Preference(<preference_name>)

Description

Returns the settings for the specified preferences.

Returns

A list of preference settings.

Argument

`preference_name` (Optional) If no preference is specified, all preferences are returned. Otherwise, the settings for the specified preference are returned.

Notes

The preferences for the following areas are not accessible in JSL: Text Data Files, Windows Specific, Mac OS Settings, Fonts, Communications, Script Editor, and JMP Updates. For more information about getting platform preferences, see "Get Platform Preferences(<platform <(option, ...)> ... >)".

`Glue(expr1, expr2, ...)`

`expr1; expr2`

Description

Evaluates each argument in turn.

Returns

The result of the last argument evaluated.

Arguments

One or more valid JSL expressions.

Notes

A semicolon placed between expressions is the more commonly used form for `Glue()`.

`Gzip Compress(blob)`

Description

Compresses a blog of data into a gzip blob.

Example

```
Gzip Compress(
   Char To BLOB(
      "random data does not usually compress well and may get larger"
   )
);
   Char To BLOB(

      "~1F~8B~08~00~00~00~00~00~00~0A~0D~CA~C1~0D~00~21~08~04~C0V~B6~B5~CDA~FC
      ~80~5C~00c~EC^~E7=~C9)~E1~106~21~A1~85~19~8DU~8Bf~07_~F8~9FZ~85~ADfx~13~
      CE~83~A1~0Dc~0E~CD~0B~94*~16~1E=~00~00~00",
      "ascii~hex"
```

`Gzip Uncompress(blob)`

Description

Uncompresses a blob of gzip data into a blob.

Example

```
Gzip Uncompress( /*typically this data might come from Gzip Compress() but
  might also come from a .gz file using Load Text File() with the blob
  option*/
  Char To BLOB(

  "~1F~8B~08~00~00~00~00~00~00~0A~0D~CA~C1~0D~00~21~08~04~C0V~B6~B5~CDA~FC~8
  0~5C~00c~EC^~E7=~C9)~E1~106~21~A1~85~19~8DU~8Bf~07_~F8~9FZ~85~ADfx~13~CE~8
  3~A1~0Dc~0E~CD~0B~94*~16~1E=~00~00~00",
     "ascii~hex"
  )
);
  Char To BLOB(
     "random data does not usually compress well and may get larger",
     "ascii~hex"
  )
```

Host Is("argument")

Description

Determines whether the host environment is the specified OS.

Returns

True (1) if the current host environment matches the argument, false (0) otherwise.

Argument

`Argument Windows"` or "`Mac`" tests for the specified operating system.

Is Alt Key()

Description

Returns 1 if the Alt key is being pressed, or 0 otherwise.

Notes

On a macOS, `Is Alt Key()` tests for the Option key.

Is Command Key()

Description

Returns 1 if the Command key is being pressed, or 0 otherwise.

Is Context Key()

Description

Returns 1 if the Context key is being pressed, or 0 otherwise.

Is Control Key()

Description

Returns 1 if the Control key is being pressed, or 0 otherwise.

Notes

On a macOS, `Is Control Key()` tests for the Command key.

Is Option Key()

Description

Returns 1 if the Option key is being pressed, or 0 otherwise.

Is Shift Key()

Description

Returns 1 if the Shift key is being pressed, or 0 otherwise.

JMP Product Name()

Description

Returns either `"Standard"`, `"Pro"`, or `"Student"`, depending on which version of JMP is licensed.

JMP Version()

Description

Returns the version number of JMP that you are running.

Returns

release.revision<.fix>

Arguments

none

Load DLL("path" <,AutoDeclare(Boolean | Quiet | Verbose)>)

Load DLL("path" <, Quiet | Verbose>)

Description

Loads the DLL in the specified *path*.

Arguments

`path` A pathname that specifies where to load the DLL.

`AutoDeclare(Boolean | Quiet | Verbose)` Optional argument. `AutoDeclare(1)` and `AutoDeclare(Verbose)` write verbose messages to the log. `AutoDeclare(Quiet)`

turns off log window messages. If you omit this option, verbose messages are written to the log.

`Quiet | Verbose` Optional argument. When you use `Declare Function()`, this option turns off log window messaging (`Quiet`) or turns on log window messaging (`Verbose`).

See Also

Once a DLL is loaded, you send the DLL object messages to interact with it. See "Dynamic Link Libraries (DLLs)" for more information about these messages.

`Mail("address"|"addresses", "subject", "message", <"attachment filepath" | {"attachment 1 filepath", "attachment 2 filepath", ...}>)`

Description

(Windows) Sends e-mail (using MAPI) to the *`address`* with the specified *`subject`* and *`message`* texts. Sends one or more attachments specified by the optional *`attachment`* argument. The attachment argument can evaluate to a quoted string or list of strings.

(macOS) Creates an e-mail in the user's Mail application. The user must click **Send** in the e-mail. In Microsoft Outlook, you must manually add attachments to the e-mail.

Examples

To send an email with multiple attachments on Windows:

```
Mail(
   "yourname@company.com",
   "New data and script",
   "Today's updated data table and script are attached.",
   {"$DOCUMENTS/wd.jsl", "$DOCUMENTS/survey.jmp"}
);
```

or:

```
list = {"$DOCUMENTS/wd.jsl", "$DOCUMENTS/survey.jmp"};
Mail(
   "yourname@company.com",
   "New data and script",
   "Today's updated data table and script are attached.",
   list
);
```

To send an email to multiple recipients:

```
Mail(
   {"hername@company.com", "hisname@company.com"},
   "Database updates",
   "Today's sales database contains the numbers from last month."
);
```

Notes

On macOS, `Mail()` works on Yosemite and later operating systems.

Main Menu(*string*, <*string*>)

Description

Execute the command found on JMP's menu named by the quoted *string*.

Arguments

`string` The internal path name as shown in the menu editor for items. For example, "NEW" is the internal name for the New subcommand in the File menu.

`string` (Optional) The name of the window to send the command to.

Examples

`Main Menu()` accepts either a full path or a partial path. If a partial name is used, and there are other menu items with the same name, the first menu item found is executed. JMP searches the top-level menu (File, Tables, DOE, and so on) first for the partial name and then searches inside each of those menus in order.

```
Main Menu( "File:New:Data Table" ); // full path
Main Menu( "Data Table" ); // partial path
```

Minus(a)

-a

Description

Reverses the sign of *a*.

Returns

-*a* if *a* is positive (`a=3; -a=-3; Minus(a)=-3`).

a if *a* is negative (`a=-3; -a=3; Minus(a)=3`).

0 if *a* is 0 (`a=0; -a=0; Minus(a)=0`).

Missing if *a* is missing.

Argument

`a` Can be variable or a number. A variable must contain a number or a matrix.

Multiple File Import(arguments)

Description

Imports one or more files into a data table. You can create this JSL by selecting Save Script to Script Window from the Multiple File Import window.

Returns

Creates a Multiple File Import Object. The object accepts messages to set a folder, filter files, and specify import options.

Arguments

`<<Set Folder` Specifies the folder that contains the files you want to import.

<<Set Name Filter (Optional) Specifies the file name or extension of the files. The name filter uses * to represent zero-or-more characters and ? to represent exactly one character. * and ? also match a period. The default setting is *.*, or all files. Because you can specify files using a semicolon-delimited list of filters, file names that include semicolons or | must be specified using ? or *.

<<Set Name Enable(Boolean) Enables the name filter. The setting is off by default.

<<Set Size Filter (Optional) Filters the file list by file size. Specify the sizes by kB (kilobytes, or 1000 bytes) in a list. The default values are based on the size range of the files in the file list.

<<Set Size Enable(Boolean) (Optional) Enables the size filter. The default setting is off.

<<Set Date Filter (Optional) Filters the file list by date and time. Specify the date and time in a list in seconds. The default values are based on the date and time range of the files in the file list.

<<Set Date Enable(Boolean) (Optional) Enables the date filter. The default setting is off.

<<Set Add File Name Column(Boolean) (Optional) Includes a column that contains the imported file name. The default setting is off.

<<Set Add File Size Column(Boolean) (Optional) Includes a column that contains the size of the imported file. The default setting is off.

<<Set Add File Date(Boolean) (Optional) Includes a column that contains the time and date stamp of the imported file. The default setting is off.

<<Set Import Mode(Row Per File|Row Per Line|CSV Data) (Optional) Specifies the format of the file that is imported: whole file on one row, one line on one row, and CSV. CSV Data is the default setting.

<<Set Charset(Best Guess|utf-8|utf-16|us~ascii|windows-1252|x-mac-roman|x-mac-japanese|shift-jis|euc-jp|utf-16be|gb2312) (Optional) The character set in the imported file. The character set specified in the General preferences (Open Text File Charset) is set by default.

<<Set Stack Mode(Stack Similar|TablePerFile) (Optional) Specifies how the files are combined. Stack Similar is the default setting. (When JMP detects that the files have the same columns, the files are concatenated into a single data table.)

<<Set CSV Has Headers(Boolean) (Optional) Specifies whether the CSV file contains a header row. The setting is on by default.

<<Set CSV Allow Numeric(Boolean) (Optional) Sets the data type to numeric. The setting is on by default.

<<Set CSV First Header Line(*n*) (Optional) Specifies the header row number. 1 is the default setting.

<<Set CSV Number of Header Lines(*n*) (Optional) Specifies the number of header rows. 1 is the default setting.

`<<Set CSV First Data Line(n)` (Optional) Specifies the first line that contains data. 2 is the default setting.

`<<Set CSV EOF Comma(Boolean)` (Optional) Specifies a comma delimiter. The setting is on by default.

`<<Set CSV EOF Tab(Boolean)` (Optional) Specifies a tab delimiter.

`<<Set CSV EOF Space(Boolean)` (Optional) Specifies a space delimiter.

`<<Set CSV EOF Spaces(Boolean)` (Optional) Specifies spaces as the delimiter.

`<<Set CSV EOF Other("")` (Optional) Specifies a custom delimiter.

`<<Set CSV EOF CRLF(Boolean)` (Optional) Specifies carriage return and line feed end-of-line characters. The setting is on by default.

`<<Set CSV EOF CR(Boolean)` (Optional) Specifies a carriage return end-of-line character. The setting is on by default.

`<<Set CSV EOF LF(Boolean)` (Optional) Specifies a line feed end-of-line character.

`<<Set CSV Semicolon(Boolean)` (Optional) Specifies a semicolon end-of-line character. The setting is off by default.

`<<Set CSV EOL Other("")` (Optional) Specifies a custom end-of-line character.

`<<Set CSV Quote("")` (Optional) Specifies the character used as a quote. The default setting is `\!"`, a double quotation mark.

`<<Set CSV Escape("")` (Optional) Specifies the escape sequence such as a backlash instead of doubling the quotation mark.

`<<Import Data` Imports the data.

Example

```
mfi = Multiple File Import(
   <<Set Folder( "$SAMPLE_IMPORT_DATA" ),
   <<Set Name Filter( "UN*.csv" ), // import files with this name
   <<Set Name Enable( 1 ) // display the file name in a column
)
<<Import Data();
```

`Multiply(a, b, ...)`

`a*b*...`

Description

Multiplies all values. No arguments are changed.

Returns

The product.

Arguments

Any number of variables, numbers, or matrices.

Notes

Any number of arguments is permitted. If no argument is specified, `Multiply()` returns 1.

Name(*string*)

Description

A name is something to call an item.

- If the name begins with an alphabetic character or underscore, and continues with alphanumeric characters, whitespace, Unicode mathematical symbols and certain punctuation (apostrophes, percentage signs, periods, backslashes, and underscores, then you can use the name directly in scripts.
- You can use names that do not follow these rules with the Name() keyword.

Examples

("name"n) is preferred. Name(*string*) is deprecated.

```
"my-var-name"n = "hello";
Length( "my-var-name"n )
   5
```

New OAuth2 Token(user(yourgoogleaccount@gmail.com), client ID(*string*), client secret(*string*), refresh token(*string*), token URL(*string*))

Description

Creates an OAuth2 token for securely accessing data across different web APIs.

Arguments (Required)

`user` The user name, email, or personal identifier for the account being accessed.

`client ID` The public identifier that acts as an API key.

`client secret` The private identifier that corresponds to Client ID.

`refresh token` A token used to get access tokens.

`token URL` The URL that access tokens are received from. Unique to every service and accessible through their API or OAuth page.

Arguments (Authorization Code Grant)

`redirect URL( "https://app.getpostman.com/oauth2/callback" )` The URL that an access code is sent back to. Unless your company or the service provides one, we recommended that you create a free Postman account and use this redirect.

`client secret( "1aB893cdDeFf2D" )` The private identifier that corresponds to Client ID.

`request auth( ... )` Extra parameters indicating that you'll use an Authorization Code flow. Requires `Auth URL()`. Some services require scope. You can add custom fields with Fields.

`scope( "spreadsheets email docs" )` A space-separated list of scopes. Unique to every service, and accessible through their API or OAuth page. Only usable in `Request Auth()`.

`auth URL( "https://www.example.com/oauth2/v1/authorize" )` The URL for requesting authorization. Unique to every service, and accessible through their API or OAuth page. Only usable in `Request Auth()`.

Arguments (Implicit Grant)

`redirect URL( "https://app.getpostman.com/oauth2/callback" )` The URL that an access code is sent back to. Unless your company or the service provides one, it's recommended to create a free Postman account and use this redirect.

Arguments (Resource Owner)

`password( "wordpass123" )` The password that corresponds to the username.

`client secret( "1aB893cdDeFf2D" )` The private identifier that corresponds to Client ID.

Arguments (Custom Data)

`fields( fields )` Custom fields that are equivalent to HTTP Request's Form(fields(fields)). Can be specified both in `New OAuth2 Token()` and in `Request Auth()`. Only necessary if the service requires information that is not defined in the OAuth 2.0 standard.

`headers( headers )` Custom headers that are equivalent to `HTTP Request's Headers( headers )`. Can only be specified in `New OAuth2 Token()`. Only necessary if the service requires information not defined in the OAuth 2.0 standard.

Example

```
token = New OAuth 2 Token (
  User( "yourgoogleaccount@gmail.com" ),
  Refresh Token( "1a2b3c4e5F" ),
  Token URL( "https://www.example.com/oauth2/token" ),
  Client ID( "12ab" ),
  Client Secret( "3456dEfG" )
);
```

See Also

See your API documentation for more information about how to get values such as the client secret and token URL.

Open Datafeed()

Datafeed()

Description

Creates a Datafeed object and window.

Returns

A reference to the Datafeed object.

Arguments

No arguments are required. You usually set up the basic operation of the data feed within the `Open Datafeed()` command, however.

`Open Help("Help"|"Statistics Index"|"Scripting Index", ...)`

Description

Opens the specified help window.

`Parse XML(`*`string`*`, On Element("tagname", Start Tag(expr), End Tag(expr), Text))`

Description

Parses an XML expression using the `On Element` expressions for the specified XML tags.

Example

```
XMLData = "
<Book name='Foods'>All you want to know
       <Chapter num='1'>Fruit
             <kind>Apple</kind>
             <kind>Cherry</kind>
             <ps>I love dessert!</ps>
       </Chapter>
       <Chapter num='2'>Bread
             <kind>Wheat</kind>
             <kind>Corn</kind>
             <ps>I love sandwiches!</ps>
       </Chapter>
       <Chapter num='3'>Veggy
             <kind>Squash</kind>
             <kind>Cabbage</kind>
             <ps>I love anything else!</ps>
       </Chapter>
        and more.
</Book> ";

// variables are initialized so text can be concatenated
title = "";
subtitle = "";
chap = "";
chapnum = "";
ps = "";

Parse XML( XMLData,
      On Element( "Book",
            // capture the name attribute during the start of the Book
            Start Tag( title = XML Attr( "name" ) ),
           /* this book has split the subtitle and needs to join the text;
              the joined text will be used by endTag.
              Text(...) supplies the JSL.*/
```

```
                Text( subtitle = subtitle || " -- " || Trim( XML Text() ) ),
                  /* after endTag processes the variables, set them back
                  to their initial state, just in case there is a second book
                  to process in the same XML. */
                endTag( Write( "\!n", title, " ", subtitle ); title = ""; subtitle
        = ""; )
            ),
            On Element( "Chapter",
                  // capture the chapter number during the start of the Chapter
                  Start Tag( chapnum = XML Attr( "num" ) ),
                /* the chapter text is joined together, newlines
                  and extra space is trimmed, and a single space is used to
                  separate the separated texts. The <kind> tag is ignored by
                  this ParseXML specification. The <kind> text is processed
                  by this Text(...) because it wasn't consumed by any other
                  On Element. */
                  Text( chap = chap || Trim( XML Text() ) || " " ),
                /* after endTag processes the variables, set them back to
                  their initial state, because there is another chapter
                  that needs to start with a clean slate.*/
                endTag( Write( "\!n", chapnum, " ", chap, " ps: ", ps ); chapnum =
        ""; chap = ""; ps = ""; )
            ),
            On Element( "ps", End Tag( ps = XML Text() ) )
    );
      1 Fruit Apple Cherry  ps: I love dessert!
      2 Bread Wheat Corn  ps: I love sandwiches!
      3 Veggy Squash Cabbage  ps: I love anything else!
      Foods  -- All you want to know -- and more.
```

`Platform Preferences(platform(option(value)), ...)`

`Platform Preference(platform(option(value)), ...)`

`Set Platform Preferences(platform(option(value)), ...)`

`Set Platform Preference(platform(option(value)), ...)`

Description

Sets and resets values for platform options and turns the options on and off.

Arguments

`platform` Specifies the platform of the preference.

`option` Specifies the preference name.

`value` Specifies the preference value.

Notes

Table 2.4 describes the syntax for setting platform preferences.

Table 2.4 Platform Preferences() Syntax

Syntax	Description

`Platform Preferences( <<Default )`
`Platform Preferences( <<Factory Default )`
`Platform Preferences( Default )`
	Resets all platform preferences to their default values.

`Platform Preferences( Platform( <<Default ) )`
`Platform Preferences( Platform( <<Factory Default ) )`
`Platform Preferences( Platform( Default ) )`
	Resets the specified platform preferences to their default values.

`Platform Preferences( Platform( option ( <<Default ) ) )` `Platform Preferences( Platform( option ( <<Factory Default ) ) )`	Resets the specified platform option to its default value.
`Platform Preferences( Platform( option( value, <<On ) ) )`	Sets the value of the specified platform option and turns it on.
`Platform Preferences( Platform( option( value, <<Off ) ) )`	Sets the value of the specified platform option and turns it off.

Example

The following expression selects (or turns on) Set Bin Width in the Distribution platform preferences and sets the value to 2:

```
Platform Preferences( Distribution( Set Bin Width( 2 ) ) );
```

The following expression changes the Set Bin Width value and turns the option off:

```
Platform Preferences( Distribution( Set Bin Width( 2, <<Off ) ) );
```

The following expression resets the default Set Bin Width value and deselects the preference:

```
Platform Preferences( Distribution( Set Bin Width( <<Default ) ) );
```

Polytope Uniform Random(samples, A, b, L, U, neq, nle, nge, <nwarm=200>, <nstride=25>)

Description

Generates random uniform points over a convex polytope.

Arguments

`Samples` The number of random points to be generated.

`A` The constraint coefficient matrix.

`B` The right hand side values of constraints.

`L, U` The lower and upper bounds for the variables.

`neq` The number of equality constraints.

`nle` The number of less than or equal inequalities.

`nge` The number of greater than or equal inequalities.

`nwarm` (Optional) The number of warm-up repetitions before points are written to the output matrix.

`nstride` (Optional) The number of repetitions between each point that is written to the output matrix.

Notes

The constraints must be listed as equalities first, less than or equal inequalities next, and greater than or equal inequalities last.

```
Preferences(pref1(value1), ...)
Preference(pref1(value1), ...)
Pref(pref1(value1), ...)
Prefs(pref1(value1), ...)
Set Preferences(pref1(value1), ...)
Set Preference(pref1(value1), ...)
```

Description

Sets preferences for JMP.

Arguments

`Add Files Opened by Scripts to the Recent Files List(Boolean)` Determines whether a file that is opened by a script is added to the Home Window's Recent Files list.

`Analysis Destination(window)` Specifies where to route new analyses.

`Annotation Font("font", size, "style")` Font choice for annotations in reports.

`Axis Font("font", size, "style")` Font choice for axis labels.

`Axis Title Font("font", size, "style")` Font choice for axis titles.

`Background Color( {R, G, B} | <color> )` Sets the background color for windows.

`Calculator Boxing(Boolean)` Turns on boxing to show hierarchy of expressions.

`Conditional Formatting Rules` Creates rules for conditionally formatting text in reports. See "Examples".

`Data Table Font("font", size, "style")` Font choice for data tables.

`Data Table Title on Output(Boolean)` Titles reports with name of data table.

`Date Title on Output(Boolean)` Titles reports with current date.

`Evaluate OnOpen Scripts("always"|"never"|"prompt")` Determines whether an `On Open` table script is run after the user opens the data table. By default, the user is prompted. Their choice is remembered each time they open the data table in the current JMP session. Scripts that execute other programs are never run.

`Excel Has Labels(Boolean)` When on, forces JMP to interpret the first row of data as column headings.

`Excel Selection(Boolean)` When on, the user is prompted for which non-blank Excel worksheets should be imported from an Excel workbook.

`File Location Settings(<Directory Type>("<path>"<,"initial directory">))` Valid directory types are:

`Data Files Directory` Sets the default location for data files.

`Help Files Directory` Sets the default location for help files.

`Installation Directory` By default, this location is set to the JMP installation folder on Windows:

"C:/Program Files/SAS/JMP/16" or "C:/Program Files/SAS/JMPPro/16"

`License File Path` Sets the default location for JMP license file.

`Preferences File Directory` Sets the default location for the preferences settings file.

`Save As Directory` Sets the default location for Save As file operations.

`Foreground Color(color)` Sets the foreground color for windows.

`Formula Font("font", size, "style")` Font choice for the formula editor.

`Graph Background Color(color)` Sets the color for the background area inside the graph frame.

`Graph Marker Size(size)` Default size for drawing markers.

`Heading Font("font", size, "style")` Font choice for table column headings in reports.

`Initial JMP Starter Window(Boolean)` Specifies whether the JMP Starter window is shown at launch.

`Initial Splash Window(Boolean)` Enables you to show or suppress the initial splash screen.

`Maximum JMP Call Depth(size)` Sets the default for the maximum call depth (or stack size) for JMP in which JSL built-in functions, user-defined functions, or `Recurse()` function calls can be made. By default, the maximum call depth is set to 256KB.

Each thread that JMP creates has a 2MB stack by default. Increasing the maximum call depth can cause a physical runtime stack overflow, so incrementally increase this preference in small amounts until you find the best value that works for your JSL script.

`Marker Font("font", size, "style")` Font choice for markers used in plots.

`Monospaced Font("font", size, "style")` Font choice for monospaced text.

`ODBC Suppress Internal Quoting(Boolean)` Prevents internal quoting in SQL statements that contain table and variable names with mixed case and spaces.

`Outline Connecting Lines(Boolean)` Draws lines between titles for same-level outline nodes.

`Print Settings(option(value), ... )` Changes print options on the Page Setup window:

`Margins( <n>, <n>, <n>, <n>)` sets the left, top, right, and bottom margins. Margins are in inches.

`Margins(<n>)` sets all margins to the same value in inches.

`Orientation("portrait" | "landscape")` changes the page's print orientation.

`Headers(<"char">, <"char">, <"char">)` specifies text that appears in the left, middle, and right header.

`Headers(<"char">)` specifies the only text in the header.

`Footers(<"char">, <"char">, <"char">)` specifies text that appears in the left, middle, and right footer.

`Footers(<"char">)` specifies the only text in the footer.

`Scale(<n>)` decreases or increases the percentage at which the content prints.

`Show Explanations(Boolean)` Some analyses have optional text that explains the output.

`Show Menu Tips(Boolean)` Turns menu tips on or off.

`Show Status Bar(Boolean)` Turns display of the status bar on or off.

`Small Font("font", size, "style")` Font choice for small text.

`Text Font("font", size, "style")` Font choice for general text in reports.

`Thin Postscript Lines(Boolean)` macOS only. Specifies that line widths drawn to a Postscript printer be narrower than otherwise.

`Title Font("font", size, "style")` Font choice for titles. Arguments are name of font (for example, `"Times"`), size in points, and style (`"bold"`, `"plain"`, `"underline"`, `"italic"`).

`Use Triple-S Labels as Headings(Boolean)` When on, this argument forces JMP to interpret label names as column headings. Example: `Pref(Name("Use Triple-S Labels as Headings")(0) )`; turns off the preference.

Examples

The following expressions reset all preferences to their default values.

```
Preferences( "Default" );
Preferences( "Factory Default" );
```

The following script creates conditions for formatting text in reports.

```
Preferences(
    Conditional Formatting Rules(
```

```
            RuleSet(
               RuleName( "Warning" ),

               // if the value is not equal to 0, format the text as 80% gray
               NotEqualTo( Value( 0 ), Format( TextAlpha( 0.8 ) ) )
             )
        )
    );
```

Notes

The preferences for the following areas are not accessible in JSL: Text Data Files, Windows Specific, Mac OS Settings, Fonts, Communications, Script Editor, and JMP Updates.

See Also

"Platform Preferences(platform(option(value)), ...)"`Register Addin("unique_id", "home_folder", <named_arguments>)`

Description

Register a JMP Add-In and load the add-in if it registers successfully.

Returns

If successful, returns a scriptable object representing the registered add-in. If unsuccessful, returns Empty.

Arguments

`unique_id` A quoted string that contains the unique identifier for the add-in. The string can contain up to 64 characters. The string must begin with a letter and contain only letters, numbers, periods, and underscores. Reverse-DNS names are recommended to increase the likelihood of uniqueness.

`home_folder` A quoted string that contains the filepath for the folder containing the add-in files. The filepath must conform to the valid pathname requirement for the host operating system.

`DisplayName( "name" )` An optional, quoted string that contains a name that can be displayed in the JMP user interface wherever add-in names are displayed, instead of the unique ID.

`JMPVersion("version")` An optional string that contains a specific version of JMP. The default value is "All", which enables the add-in to be loaded and run in any version of JMP that supports add-ins. "Current" restricts the use of the add-in to only the current version. Any quoted version number (for example, "7" or "9") restricts the add-in to a single specific version of JMP.

`LoadsAtStartup(Boolean)` An optional Boolean. The default value is True (1), which causes the add-in to be loaded when JMP is started. If the value is False (0), the add-in is not loaded automatically.

`LoadNow(Boolean)` Loads the add-in immediately.

Notes

If a file named addin.def is found in the specified home folder, values from that file are used for any optional arguments that are not included in the `Register Addin()` function.

Example

In the following example, the first argument is the unique identifier. The second argument identifies where the add-in is installed. The third argument is the name that appears where add-in names are displayed (for example, the **View > Add-Ins** menu on Windows).

```
Register Addin("com.company.lee.dan.MyAddIn","$DOCUMENTS/myaddin",
  displayname( "Calculator Addin" ));
```

The second argument becomes the `$ADDIN_HOME` path variable definition. When you refer to the add-in scripts, be sure to include a trailing slash after the path variable.

```
Include("$ADDIN_HOME(com.jmp.jperk.texttocols)/texttocols.jsl");
```

Revert Menu()

Description

Resets your JMP menus to factory defaults.

Run Program(Executable("path/filename.exe"), Options({"/a", "/b", "..."}), Read Function(expression), Write Function(expression), Parameter(expression))

Description

Runs the external program specified by the `Executable` argument, with the command line arguments specified by the `Options` argument.

Results

Returns either a quoted string, a blob, or a `Run Program` object as controlled by the `Read Function` argument.

Arguments

`Executable` The path to the executable. On macOS, type the full path to the executable.

`Options` Command line arguments for the executable.

`Read Function` If `Read Function( "text" )` is specified, a quoted string is returned. If `Read Function( "blob" )` is specified, a blob is returned. The script waits until the external program closes its stdout. `Run Program` then returns all data that the external program has written to its stdout as a quoted string or a blob.

If `Read Function` is not specified, a `Run Program` object is returned.

`Write Function` Optional argument that accepts a function as its value; it does not accept `"text"` or `"blob"`.

`Parameter` Optional argument to read and write the expression in `Read Function`.

Notes:

- Use global variables when `Run Program()` is inside a function.
- The `Run Program` object, which is returned if `Read Function` is not specified, accepts the following messages to read data from the external program's stdout:
  - `<<Read`: reads any available data as a quoted string. If no data is available, an empty string is returned.
  - `<<Can Read`: returns `true` if there is data available to be read.
  - `<<Is ReadEOF`: returns `true` when the external program has completed and all its data has been read.

  You can use these messages to poll for data and process the data as it is produced by the external program.
- A `Run Program` object accepts the following messages to write data to the external program's `stdin`:
  - `<<Write( "text" )`: sends data to the external program's stdin.
  - `<<Can Write`: returns `true` if the external program will accept data immediately; otherwise, calling `<<Write` causes your script to block.
  - `<<WriteEOF`: signals to the external program that you are done sending data to it.
- Instead of sending messages to the returned `Run Program` object, you can specify the `Read Function` argument as an inline function. `RP` is the `Run Program` object.

```
RP = Run Program(
   Executable( ... ),
   Read Function(
      Function( {RP},
          <your code here>
         RP << Read
      )
   )
);
```

The `Parameter(optParm)` argument is optional in `Read Function`. If specified, the functions defined for `Read Function` and `Write Function` can receive a second argument, which is the value of `optParm`.

Examples

The following script is an example of the `Write Function` argument. `RP` is the `Run Program` object. In this context, it accepts the `<<Write` and `<<WriteEOF` messages.

```
RP = Run Program(
   Executable( ... ),
   Write Function(
      Function( {RP},
         <your code here>
```

```
            RP << Write( "Program finished." )
        )
    )
);
```

The following script shows an example of `Parameter(optParm)` argument:

```
RP = Run Program(
   Executable( ... ),
   Parameter( x ),
   Read Function( Function( {RP, optParm},... ) )
);
```

Within the `Read Function`, `optParm` contains the value of *x*. Do not attempt to access the `optParm` argument in your function if you have not specified a `Parameter` argument.

Schedule(n, script)

Description

Queues an event to run the *script* after *n* seconds.

Set Clipboard(*string*)

Description

Evaluates the quoted *string* argument looking for a character result, and then places the string on the clipboard.

Example

```
Set Clipboard( "copy me" );
```

SetJVMOption(Version("<version number>"))

Description

Sets the Java Runtime Environment (JRE) version that you want JMP to use (rather than the version installed with JMP). This script must be run before JMP connects to the JRE.

Argument

`version` (Windows only) In the Windows registry, there are two requirements for the JavaSoft/Java Runtime Environment key: the key must include a quoted string called "RuntimeLib" that points to a valid jvm.dll. And the Java Runtime Environment key must include a key named after the quoted JVM version number.

Set Platform Preference()

Set Platform Preferences()

See "Platform Preferences(platform(option(value)), ...)".

`Set Preference()`

`Set Preferences()`

See "Preferences(pref1(value1), ...)".

`Set Toolbar Visibility( "toolbar name" | default | all, window type | all, "true" | "false" )`

Description

On Windows, shows or hides a toolbar based on the window type or for all windows.

Arguments

`toolbar name | default | all` The internal name of the toolbar (see the **View > Toolbars** list in JMP), the default toolbar for the specified window type, or all toolbars. Include quotes around `"toolbar name"`.

`window type | all` Data table, script, report, journal, or all windows.

`true | false` A quoted string that shows or hides the toolbar.

`Shortest Edit Script( A, B )`

`Shortest Edit Script( strings( A, B, <matrix( 0|1 )>, <limit( number )> ) )`

`Shortest Edit Script( lines( A, B, <matrix( 0|1 )>, <limit( number )>, <separators( characters )>, <ignore( characters )|ignore white space( )> ) )`

`Shortest Edit Script( sequences( nA, nB, Function( {iA, iB}, adata[iA] == bdata[ib] ) ) )`

Description

Compares two quoted strings, lines, or sequences.

Returns

Returns a list or a matrix of edit commands. The simplest form returns a list. quoted `strings` and `lines` return a matrix (if set to 1) or a list. `sequences` returns a matrix.

There are three possible commands: common data in both quoted strings, delete data from the first string, and keep data from the second string.

Optional Arguments

`matrix` Indicates whether the returned value is a matrix.

`limit` Stops the evaluation when the edit list exceeds the specified number of inserted or deleted items. Two quoted random strings have a lot of common characters in a lot of distinct sections. The function runs for a long time trying to find a best match. `limit` stops the function sooner.

Optional Lines Arguments

`matrix` Indicates whether the returned value is a matrix.

`limit` Stops the evaluation when the edit list exceeds the specified number of inserted or deleted items. Two quoted random strings have a lot of common characters in a lot of distinct sections. The function runs for a long time trying to find a best match. `limit` stops the function sooner.

`separators` A character that separates words.

`ignore` Ignores the specified spaces or characters in a line.

`ignore white space` Ignores white space in a line.

Optional Sequences Argument

`Function` A user-defined function.

Examples

The following example compares two quoted strings with three common sequences of characters between them.

```
Shortest Edit Script( "abcdef", "abdezgh" );
```
```
{{"Common", "ab"}, {"Remove", "c"}, {"Common", "de"}, {"Insert", "zgh"},
   {"Remove", "f"}}
```

The following example examines each line in quoted string aa and bb:

```
aa = "this is
a test of
shortest
edit script
lines with several words";

bb = "this is
a test 2 of
shortest
edit ?., script
lineswithseveral words";

Shortest Edit Script( lines( aa, bb, separators( "\!n" ),

  // quote and newline separators
  ignore( "?., " ) ) ); // ignore these characters and spaces
{{"Common", "this is // lines in aa and bb contain "this is"
"}, {"Remove", "a test of // only on line 2 of aa
"}, {"Insert", "a test 2 of // only on line 2 of bb
"},
{"Common", "shortest
edit script
lines with several words"}}
// lines in aa and bb contain "shortest", "edit script", and "lines with
   several words"
```

`Show Addin Builder Dialog()`

Description

Opens a window in which you can make custom add-ins.

`Show Addins Dialog()`

Description

Opens the Add-In Status window (**View > Add-Ins**).

Arguments

None.

`Show Commands()`

Description

Lists scriptable objects and operators. Arguments are All, DisplayBoxes, Scriptables, Scriptable Objects, StatTerms, Translations.

`Show Preferences(<"all">)`

Description

Shows current preferences. If no argument is specified, preferences that have been changed are shown. If `"all"` is given as the argument, all preferences are shown.

`Show Properties(object)`

Description

Shows the messages that the given *`object`* can interpret, along with some basic syntax information.

`Sobol Quasi Random Sequence(nDim, nRow)`

Description

Generates a sequence of space-filling quasi random numbers using the Sobol sequence in up to 4000 dimensions.

`Socket(<STREAM | DGRAM>)`

Description

Creates a socket.

Returns

The socket that was created.

Arguments

STREAM | DGRAM Optional argument to specify whether the socket is a stream or datagram socket. If no argument is supplied, a stream socket is created.

`Speak(text, <wait(Boolean)>)`

Description

Calls system's speech facilities to read aloud the text. If Wait is turned on, script execution pauses until speaking is done.

`Status Msg("message")`

Description

Writes the quoted message string to the status bar.

`Subtract(a, b)`

`a-b-...`

Description

Subtracts the values of the listed arguments, left to right. No arguments are changed.

Returns

The difference.

Arguments

Two or more variables, numbers, or matrices.

Notes

Two or more arguments are permitted.

`Unregister Addin("unique_id")`

Description

Unregisters (removes) a previously registered add-in.

Argument

`unique_id` A quoted string that contains the unique identifier for the add-in to unregister.

`Web(`*`string`*`, <JMP Window>)`

Description

Opens the URL stored in the quoted *`string`* in the default web browser.

The `http://` prefix in the URL is optional.

Examples

```
url = "www.jmp.com"; // open the URL in the default web browser
Web( url );

Web( "www.jmp.com" ); // open the URL in the default web browser

Web( "www.jmp.com", JMP Window ); // open the URL in the JMP browser window
```

XML Attr("attr name")

Description

Extracts the quoted string value of an XML argument in the context of evaluating a `Parse XML` command.

XML Decode("xml")

Description

Decodes symbols in XML to ordinary text. For example, `&` becomes &, and `<` becomes <.

Argument

`xml` A quoted string containing XML.

XML Encode("text")

Description

Prepares text for embedding in XML. For example, & becomes `&`, and < becomes `<`.

Argument

`xml` A quoted string containing plain text.

XML Text()

Description

Extracts the quoted string text of the body of an XML tag in the context of evaluating a `Parse XML` command.

Chapter 3

JSL Messages

Summary of Messages for Objects and Display Boxes

This topic provides abbreviated descriptions for many of JMP's general object messages. For complete information about object messages, see the JMP Scripting Index. In JMP, select **Help > Scripting Index**.

For information about platform messages, see *Scripting Guide*.

Contents

Alpha Shape . . . 380
Associative Arrays . . . 380
Classes . . . 381
Data Tables . . . 383
Columns . . . 412
Rows . . . 420
Data Filter . . . 421
Data Feed (Windows Only) . . . 424
Display Boxes . . . 427
All Display Boxes . . . 427
Axis Boxes . . . 437
Border Boxes . . . 442
Data Browser Boxes . . . 443
Data Filter Source Boxes . . . 443
Frame Boxes . . . 443
Display 3D Boxes . . . 445
Excerpt Boxes . . . 445
Filter Col Selector . . . 446
Global Boxes . . . 446
Hier Boxes . . . 446
Matrix Boxes . . . 446
Nom Axis Boxes . . . 447
Number Col Boxes . . . 447
Number Col Edit Boxes . . . 450
Number Edit Box . . . 451
Outline Boxes . . . 451
Panel Boxes . . . 452
Plot Col Boxes . . . 452
Slider Boxes and Range Slider Boxes . . . 453
String Col Boxes . . . 454
Tab Boxes . . . 455
Table Boxes . . . 456
Text Boxes . . . 459
Tree Node and Tree Box . . . 461
Triangulation . . . 463
Windows . . . 464
Dynamic Link Libraries (DLLs) . . . 467
HTML 5 . . . 469
Web Report . . . 469

Images 470
JMP Applications 473
JMP App 473
JMP App Module 475
JMP App Module Instance 475
MATLAB 476
Namespaces 479
Platforms 481
Bubble Plot 487
DOE 487
Partition 488
Response Screening 488
Tabulate 489
Python Integration Messages 490
R Integration Messages 493
SAS Integration Messages 496
Metadata Server Objects 496
SAS Server Objects 497
Stored Processes 506
SAS Results 512
Schedule 514
Segments 515
Sockets 515
SQL 519
Other Objects 523
Zip Archives 523
Journals 523

Alpha Shape

For the following messages, `ashape` stands for an alpha shape or a reference to one.

`ashape <<Get Alpha`

Returns the current alpha value.

`ashape <<Set Alpha(`*`alpha`*`)`

Sets the current *alpha* value and recomputes the triangulation.

`ashape <<Get Tri Alpha`

Returns the alpha values for each triangle.

Associative Arrays

For the following messages, `map` stands for an associative array or a reference to one.

`map<<First`

Returns the first key within *map*, or `Empty()` if *map* has no keys. Note that keys are returned in lexicographical order.

`map<<Get Contents`

Returns a list of all key-value pairs within *map*.

`map<<Get Keys`

Returns a list of all the keys within *map*.

`map<<Get Default Value()`

Returns the implicit value of all absent keys, or `Empty()` if none has been set.

`map<<Get Value(`*`key`*`)`

Returns the value for the *key* within *map*.

map<<Get Values(*<{keyList}>*)

If no argument is provided, a list of all values within *map* is returned.

If a list of keys is provided, a list of the values corresponding to only those keys is returned.

map<<Insert(*key*, *value*)

Inserts the *key* into *map* and assigns *value* to it. If *key* already exists in *map*, its value is replaced by the new *value* given. This message is equivalent to the function Insert Into.

map<<Next(*key*)

Returns the key following the given *key* within the *map*, or Empty() if *map* has no keys. Note that keys are returned in lexicographical order.

map<<Remove(*key*)

Removes the *key* and *value* from *map*. This message is equivalent to the function Remove From.

map<<Set Default Value(*v*)

Sets the implicit value of all absent keys. Any key added without a value is assigned this value by default.

Classes

obj<<Clone

Returns a reference to a new class object that is a copy of the *obj* class object.

obj<<Contains(*quoted string*)

Returns 1 if the *obj* class object contains the specified quoted string expression, and 0 otherwise.

obj<<Delete Class

Deletes the *obj* class object.

`obj<<Equal(classref)`

Returns 1 if the *classref* class object is equal to the *obj* class object, and 0 otherwise.

`obj<<First`

Returns the quoted string representation of the name of the first member (item) in the *obj* class object. The members (items) in the class object are sorted in alphabetical order.

`obj<<Get Contents`

Returns a list of members (items) in the *obj* class object. Each element in the list is a two-item list that contains a key and an associated value.

`obj<<Get Keys`

Returns a list of keys within the *obj* class. Each key is a quoted string representation of the name of a member (item) in the *obj* class object.

`obj<<Get Name`

Returns a quoted string representation of the name of the *obj* class object.

`obj<<Get Value(key quoted string)`

Returns the value of the specified member (item) within the *obj* class object. The quoted *key quoted string* argument specifies the key to the member (item).

`obj<<Get Values`

Returns a list of values of the members (items) in the *obj* class object. Each element in the list is the expression that represents the value of each member (item) in the class.

`obj<<Insert(quoted string, value)`

Inserts a member (item) into the *obj* class object. The quoted *string* argument is the name of the member (item), and the *value* argument is the expression value of the member (item).

`obj<<Lock Class(<quoted string|{quoted stringList}>)`

Locks the *obj* class object, or locks specific members (items) within the *obj* class object. When a class object is locked, members (items) cannot be added, changed, or removed. The quoted *string* or quoted *stringlist* arguments specify a member (item) to lock. You can also specify a list of quoted strings to lock multiple members (items).

obj<<N Items

Returns the number of members (items) in the *obj* class object.

obj<<Next(*quoted string*)

Returns the quoted *string* representation of the name of the member (item) in the *obj* class object that follows the member (item) specified by the quoted *string*. The members (items) in the class object are sorted in alphabetical order.

obj<<Remove(<*string*|*{stringList}*>)

Removes the member (item) specified by the quoted *string* or quoted *stringlist* from the *obj* class. You can remove multiple members (items) using a list of quoted strings.

obj<<Show Contents

Shows the contents of the *obj* class object in the log window.

obj<<Unlock Class(<*quoted string*|*{stringList}*>)

Unlocks the *obj* class object, or unlocks specific members (items) within the *obj* class object. When a class object is unlocked, members (items) can be added, changed, or removed. The quoted *string* or *quoted stringlist* specify a member (item) to unlock. You can also specify a list of quoted strings to unlock multiple members (items).

Data Tables

dt<<Add Column Properties(*property argument*, ...)

Adds the specified properties (such as Value Order and Missing Value Codes) to the selected column.

dt<<Add Multiple Columns(*column prefix*, *n*, <"Before First"|"After Last"|After(*column*)>, "Character"|"Numeric"|"Row State", <Field Width(*n*)>)

Description

Adds *n* columns to *dt* at the position indicated.

Required Arguments

column prefix The prefix to add to the new columns names.

n The number of columns to add.

Character A new character column.

`Numeric` A new numeric column.

`Row State` A new row state column.

Optional Arguments

`Before First` Adds the columns before the first column.

`After Last` Adds the columns after the last column.

`After(`*column*`)` Adds the columns after the specified column.

`Field Width(`*n*`)` Specifies the width of the columns.

Notes

If you omit arguments, or the arguments are incorrectly specified, the Add Multiple Columns window appears.

`dt<<Add Rows(<`*n*`>, <"At Start"|"At End"|After(`*row number*`)>|{`*column name=value pairs*`})`

Description

Adds rows at the start, at the end, or after a specified row in the data table. The message can also add rows based on the specified column name and value pairs. Those rows are added to the end of the data table.

Notes

If you omit arguments, or the arguments are incorrectly specified, the Add Rows window appears.

`dt<<Add Scripts to Table(`*script*`, ...)`

`dt<<Add Properties to Table(`*script*`, ...)`

Adds the specified scripts to the data table.

`dt<<Anonymize(<Columns(`*column list(s)*`)>, <Output Table Name(`*name*`)>);`

Removes unique identifiers from data, some column properties, and table scripts. Applies to a data table or the specified list of columns. The new data table has the name that is specified by the quoted *name* argument.

`dt<<Begin Data Update`

Holds off display updating to allow for quick updating of data table cells. Use `End Data Update` in conjunction with this command to turn display updating back on.

Notes

`Begin Data Update` does not affect the data refresh due to some other table manipulations. For example, when you delete or add columns, the data table is updated and then the data update begins.

dt<<Clear Column Selection

Deselects all selected columns.

dt<<Clear Edit Lock(<"Modify Cells">, <"Add Rows">, <"Add Columns">, <"Delete Rows">, <"Delete Columns">)

Description

Allows the specified data table operation again.

Notes

If no arguments are specified, all locks are cleared.

dt<<Clear Row States

Cancels any row states in effect.

dt<<Clear Select

Turns off the current selection.

dt<<Clone Formula Column(*column*, *n*, Substitute Column Reference(*column1*, *{list}*))

Creates *n* new formula columns, substituting references to *column1* with columns from the *list* into the formula from the original *column*.

dt<<Close Data Grid(*Boolean*)

If true, closes the data table grid.

dt<<Close Side Panels(*Boolean*)

If true, closes the side panel in a data table.

dt<<Color or Mark by Column(*column*, *<named arguments>*)

dt<<Color by Column(*column*, *<named arguments>*)

dt<<Marker By Column(*column*, *<named arguments>*);

Description

Assigns colors or markers according to the values of a data table column. If no optional arguments are provided, colors are assigned according to the default color theme.

Required Argument

`column` The column to color or mark.

Optional Named Arguments

`Color(n)` Uses the specified JMP color.

`Add Marker(Boolean)` Shows or hides the marker in the data table.

`Color Theme(color theme)` Uses the specified quoted *color theme*.

`Marker Theme(marker theme)` Uses the specified quoted *marker theme:* `"Standard"`, `"Hollow"`, `"Paired"`, `"Classic"`, or `"Alphanumeric"`.

`Continuous Scale|Continuous Scale(Boolean)` Assigns colors in a chromatic sequential fashion based on the values in the highlighted column.

`Reverse Scale|Reverse` Reverses the color scheme in use.

`Excluded Rows(Boolean)` If true, applies the row states to excluded columns.

`"Make Window with Legend"` Creates a separate window with a legend.

`dt<<Color Rows by Row State`

Colors the rows in the data table grid using the color assignments by row states. Send the message again to turn off the row colors.

`dt<<Combine Columns(Delimiter("delim"), Columns(column1, column2, etc.), Column Name(quoted string))`

Combines several columns into a single column. Each source columns' values are separated by the delimiter specified as the quoted *delim* argument.

Examples

```
dt = Open( "$SAMPLE_DATA/Consumer Preferences.jmp" );
dt << Combine Columns(
  Delimiter( "," ),
  Columns(
     :Brush After Waking Up,
     :Brush After Meal,
     :Brush Before Sleep,
     :Brush Another Time
  ),
  Column Name( "When to Brush" )
);
```

`dt<<Compress File When Saved(Boolean)`

Compresses the file when the data table is saved.

`dt<<Compress Selected Columns({column1, ...})`

Compresses the listed columns into the most compact form that is possible. Columns with character data are compressed to 1 byte if there are fewer than 255 levels. Columns with numeric data are compressed to 1 byte if the integers are between -127 and 127.

`dt<<Concatenate(dt2|Data Table(name)|Multiple Data Table(name) arguments, (<"Private"|"Invisible">), <Output Table Name(name)>|"Append to First Table">, <"Keep Formulas">, <"Create Source Column">)`

Description

Creates a new table (*name*) from the rows of *dt* and *dt2*. By default, Concatenate creates a new data table and appends the rows of each data table that is specified.

Returns

A reference to the concatenated data table.

Required Arguments

`dt2|Data Table(name)|Multiple Data Table(name)` A data table reference or the names of the data table or data tables that you would like to combine.

Optional Arguments

`"Private"` A quoted keyword that opens the data table without displaying it in a data table window.

`"Invisible"` A quoted keyword that hides the data table. Use this argument to keep the data table hidden but use it in a subsequent expression. The data table is displayed in the Home Window's Window List and the Window > Unhide list.

`Output Table name(name)` The name of the final data table. If you do not enter a name, JMP names the data table Untitled # (for example, Untitled 1).

`"Append to First Table"` Appends rows to the first data table reference or data table name in the first argument. This option is an alternative to creating a new data table.

`"Keep Formulas"` Includes formulas in the final data table.

`"Create Source Column"` Adds a column called Source Table to the new data table.

Notes

`"Private"` and `"Invisible"` only apply if *not* using `"Append to First Table"`.

`dt<<Copy Column Properties`

Copies all of the column properties for the selected columns into a list of separate lists of properties. Optionally, you can specify a list of source columns instead of preselecting them in the data table.

Example

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
dt << Select Columns( :MODULUS, :ELONG );
dt << Copy Column Properties;
New Window( "Script", Script Box( "//Try paste here" ) );
```

or

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
dt << Copy Column Properties( {:MODULUS, :ELONG} );
```

```
New Window( "Script", Script Box( "//Try paste here" ) );
```

`dt<<Copy Selected Properties`

Description

Copies the selected table properties to the clipboard.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Properties( {"Distribution", "Oneway"} );
proplist = dt << Copy Selected Properties();
New Window( "Script", Script Box( "//Try pasting here" ) );
```

`dt<<Copy Table Script("No Data")`

Copies the script to recreate the data table onto the clipboard so that it can be pasted somewhere else. Add the `"No Data"` argument to omit the data.

`dt<<Copy Table Scripts`

`dt<<Copy Selected Properties`

Description

Copies the selected scripts to the clipboard.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Properties( {"Distribution", "Oneway"} );
proplist = dt << Copy Table Scripts();
New Window( "Script", Script Box( "//Try pasting here" ) );
```

`dt<<Data Filter(<Location(`*x*`, `*y*`), <"Close Outline">, <"Local">, <Inverse(`*Boolean*`)>, <Show Columns Selector(`*Boolean*`)>, <Title(`*quoted string*`)>, <Save and Restore Current Row States(`*Boolean*`)>, <Conditional(`*Boolean*`)>, <Auto Clear(`*Boolean*`)>, <Group By AND(`*Boolean*`)>, <Show Histograms and Bars(`*Boolean*`)>, <Count Excluded Rows(`*Boolean*`)>, <Mode(...)>, <Add Filter((`*cols*`(...), <Where(...)>, <Display(...)>, <Select Missing(`*cols*`)>, <Order By Count(`*cols*`)>)>, <Favorites(...)>, <Animation(...)>)`

Constructs a data filter. If no arguments are specified, the Add Filter Columns window appears.

Optional Arguments

`Location(`*x*`, `*y*`)` Moves the data filter window to the specified location. *x* and *y* are measured in pixels. 0,0 is the top left of the monitor.

`"Close Outline"` Closes the data filter outline.

"Local" Enables the filter to be embedded in reports to filter one or more platforms without affecting other reports.

Inverse(*Boolean*) Selects all but the specified rows for all filters.

Show Columns Selector(*Boolean*) If true, a column list is shown that adds a new column to the filter.

Title(*quoted string*) The title that is displayed on the outline.

Save and Restore Current Row States(*Boolean*) Restores your current row states when the Data Filter window is closed.

Conditional(*Boolean*) Limits the categories displayed for the selected filter column.

Auto Clear(*Boolean*) If you have more than one nominal or ordinal column selected in the Data Filter, this option clears any other selections before making a new selection.

Group By AND(*Boolean*) Enables you to create a filter group, specify OR, and add one or ore filters to create second filter group. If you specify Grouped By And, the behavior is reversed and grouped by AND instead.

Show Histograms and Bars(*Boolean*) Shows or hides the histogram and bars in the data filter.

Count Excluded Rows(*Boolean*) Shows or hides the number of excluded rows.

Mode The three modes of filtering: Select(*Boolean*) shows or hides the selected rows in the data table in a highlighted state; Show(*Boolean*) shows or includes the unselected rows and shows the Hide icon; Include(*Boolean*) shows or includes the unselected rows and shows the Exclude icon.

The global data filter default is Select(), Show(0), and Include(0), The local data filter default is Show(1), Include(1), Select() is not a valid option.

Add Filter Creates the data filter. Arguments include Columns(), Where(), Display(), Select Missing(*cols*), and Order By Count(*cols*). Columns() takes one or more column names separated by commas. You can add one or more Where clauses to define the filter.

Where Defines a Where clause by which the data is filtered.

Display(*column*, *size*, *display type*) Sets how the specified categorical column levels are displayed in the filter. The arguments are Blocks Display, List Display, Single Category Display, Check Box Display, Radio Box Display. In categorical columns, you can include the Find(Set Text(*quoted string*)) argument to include and initialize the search field. Display can also be included for a continuous column and can contain a size argument.

Select Missing Cols(*cols*) Selects missing values in continuous columns.

Order by Count(*cols*) For a categorical column, this option sorts the values in decreasing order by count.

Favorites Saves the current data filter criteria as a favorite.

Animation Cycles through the sorted values of the specified column, selecting and deselecting rows. Optional arguments include Animate Column(*col*), Animate

Rate(*number*), and "Forward"|"Backward"|"Bounce"highlights values from first to last. "Backward" highlights values from last to first. "Bounce" highlights forward and then backward repeatedly.

---

dt<<Data View(*<named arguments>*)

Description

Duplicates the data table in a new window. If you specify one of the following quoted arguments, the new data table includes only the corresponding rows.

Returns

A reference to the data view.

Optional Named Arguments

Excluded The new data table includes only the rows that are marked as excluded in the original data table.

Labeled|Labelled The new data table includes only the rows that are marked as labeled in the original data table.

Hidden The new data table includes only the rows that are marked as hidden in the original data table.

Selected The new data table includes only the rows that are selected in the original data table.

---

dt<<Delete Columns(*column1*, *column2*, ...)

dt<<Delete Column

Description

Deletes one or more columns from the data table *dt*. Specify which column or columns to delete. Without an argument, deletes the selected columns, if any.

---

dt<<Delete Rows(*<n>*)

dt<<Delete Rows(*{n, o, p, ...}*)

dt<<Delete Rows(*{n::q}*)

dt<<Delete Rows([n, o, p])

dt<<Delete Row(*preceding arguments*)

Description

Deletes the currently selected rows or rows specified. Returns the number of rows that were deleted.

dt<<Delete Scripts(*table script name*|*{table script1, table script2, ...}*)

Description

Deletes the specified data table script or scripts with the quoted name or names, or deletes a list of data table scripts.

Notes

In JMP versions prior to 14, use `Delete Table Property` to delete a table script.

dt<<Delete Table Property(*name*|*{property1, property2, ...}*)

Deletes a table property (for example, a script or variable) with the quoted *name*.

dt<<Delete Table Variable(*name*)

Deletes a table variable with the quoted *name*.

dt<<Disable Undo(*Boolean*)

If true, disables undo operations in the data table.

dt<<End Data Update

Resumes display updating after a `Begin Data Update` message. These commands are used for quick updates of the data table when many changes have to be made. Speed is gained by turning off display updating.

dt<<Exclude

dt<<Unexclude

Toggles selected rows in *dt* from excluded to unexcluded or *vice versa*.

dt<<Get All Columns As Matrix

Returns the values from all columns of *dt* in a matrix. Character columns are numbered according to the levels, starting at 1.

dt<<Get As Matrix(*<{list of columns by name}>*|*<{list of columns by number}>*, *<column range>*)

Returns values from the numeric columns of *dt* in a matrix. The default output is all numeric columns.

Examples

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
```

```
cols = dt1 << Get As Matrix(); // returns all numeric columns
Show( cols );
   cols =
   [ 12 59 95,
      12 61 123,
      12 55 74,...]
colnums = dt1 << Get As Matrix( {4, 5} ); // returns columns four and five
Show( colnums );
   colnums = [ 59 95, 61 123, 55 74, 66 145, 52 64, 60 84, 61 128, ...]

dt2 = Open( "$SAMPLE_DATA/Probe.jmp" );
colrange = dt2 << Get As Matrix( 10::22); // returns columns 10 through 22
Show( colrange );
   colrange =
   [ -0.08818069845438 0.711340010166168 1.85904002189636 0.396923005580902
      4.50656986236572 7.86504983901978 1.53891003131866 -2.76178002357483
      0.0711032971739769 5.75577020645142 -3.62023997306824 -0.971698999404907
      -0.0525696985423565, ...]
```

dt<<Get As Report

Returns the data table as a report. If rows and columns are selected in the data table, only those rows and columns are in the report.

Example

The following script returns `Big Class.jmp` as a report and displays it and a distribution in one window.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dtRpt = dt << Get As Report;
distRpt = V List Box(
   dt << Distribution(
      Continuous Distribution( Column( :weight ) ),
      Nominal Distribution( Column( :age ) )
   )
);
New Window( "Example", H List Box( dtRpt, distRpt ) );
```

dt<<Get Cell Height

Returns the data table cell height in pixels.

dt<<Get Column Names(*quoted string, <modeling type>, <data type>*)

Description

Returns a list of column names in a data table. The quoted string returns a list of quoted strings rather than a list of column references.

Required Argument

quoted string Returns a list of quoted strings rather than a list of column references.

Optional Arguments

modeling type The quoted modeling type. The options are "Continuous"- "Ordinal"- "Nominal"-"Multiple Response"- "Unstructured Text", "None", and "Vector".

data type The quoted data type. The options are "Numeric", "Character", "Row State", and "Expression".

Notes

The data types and the modeling types get only the specified types of columns. More than one of each type can be specified.

---

dt<<Get Column Reference(*{list of column names}|[matrix of column numbers]*)

dt<<Get Column References(*{list of column names}|[matrix of column numbers]*)

Returns the column references of the quoted strings in the list or matrix. If no list or matrix is used, JMP returns a list of references to all columns.

---

dt<<Get Display Width

Returns the column display width in pixels.

---

dt<<Get Edit Lock

Returns a list of the disallowed operations on the data table (if cells cannot be edited; rows cannot be added or deleted; and columns cannot be added or deleted).

---

dt<<Get Excluded Columns

Returns the currently excluded columns in the data table.

---

dt<<Get Excluded Rows

Returns the rows that are excluded in the data table.

---

dt<<Get Header Height

Returns the column header's display height (in pixels).

---

dt<<Get Hidden Columns

Returns the columns that are hidden in the data table.

dt<<Get Hidden Rows

Returns the currently hidden rows in the data table.

dt<<Get Journal

Returns a quoted string that contains journal source for the display box.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( :weight ), X( :height ) );
rbiv = biv << Report;
Print( rbiv << Get Journal );
```

dt<<Get Label Columns

dt<<Get Labeled Columns

dt<<Get Labelled Columns

Returns the currently labeled columns in the data table.

Example

In PopAgeGroup.jmp, the Country and Year columns are labeled. The following script returns a list of the labeled column names.

```
dt = Open( "$SAMPLE_DATA/PopAgeGroup.jmp" );
dt << Get Labeled Columns;
   {:Country, :Year}
```

dt<<Get Labeled Rows

dt<<Get Labelled Rows

Returns the currently labeled rows in the data table.

dt<<Get Name

Returns the name of the data table.

dt<<Get Path

Returns the absolute path for the JMP data table. Note that this function is not for imported data that is not saved yet.

dt<<Get Property(*name*)

Returns the script from the quoted property *name*.

dt<<Get Row Change Function

Returns the expression that is evaluated when a row is selected.

dt<<Get Row ID Width

Returns the row ID display width in pixels.

dt<<Get Row States

Returns a vector containing the row state for every row in the data table or data filter.

dt<<Get Rows Where(*where clause*)

Returns the rows in the data table that match the specified Where criteria. Here are some examples:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Get Rows Where( :sex == "M" );
dt << Get Rows Where( :sex == "M" & :age < 15 );
```

dt<<Get Script(<*script name*>)

Returns the script specified by the quoted *script name*. If the *script name* is omitted, `Get Script` returns a text representation of the data table and all scripts in the table.

dt<<Get Script Group(<*group name*>)

Description

Returns the list of table scripts in the quoted *group name*. If no group name is specified, a list of all table scripts in all groups is returned.

dt<<Get Script Group Names

Description

Returns the list of names of table script groups.

dt<<Get Scroll Locked Columns

Returns a list of columns that are locked from scrolling.

dt<<Get Selected Columns(<*quoted string*>)

Description

Returns a list of selected columns as column references. Include the quoted *string* argument to return the selected column names as a list of strings.

dt<<Get Selected Properties(*<{list of properties}>*)

Description

Returns the selected table properties in a list.

Optional Argument

list of properties Specifies the properties to get.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Properties( {2, 4} );
proplist = dt << Get Selected Properties();
// returns the second and fourth table scripts and highlights them
// in the data table
```

dt<<Get Selected Rows()

Returns the selected rows.

dt<<Get Table Script Names()

Returns a list of the names of all the scripts and properties in the data table.

dt<<Get Table Variable(*name*)

Returns the value of the quoted *name* variable.

dt<<Get Table Variable Names

Returns a list of the names of all the variables in the data table.

dt<<Go To Row(*n*)

Locates and selects row number *n* in *dt*.

dt<<Group Columns(*{column1, column2, ...}*)

dt<<Group Columns(*group name*, *column*, *n*)

dt<<Group Columns(*first column*, *n*)

Description

Groups the columns under the specified quoted *group name*. You can provide either a list of columns to group, or a column name and the number of columns to group. In the latter case, the number *n* specifies to group the column given with the *n*-1 columns that follow.

dt<<Group Scripts({*script1, script2, ...*})

Description

Groups a list of table scripts in the data table.

dt<<Hide

dt<<Unhide

Toggles selected rows in *dt* from hidden to unhidden or *vice versa*.

dt<<Hide and Exclude

Hides the selected rows from graphs and excludes them from contributing to calculations.

dt<<Invert Column Selection(<{*list of columns*}>)

Description

Selects any column that is currently deselected and deselects any column that is currently selected. If the *list of columns* is specified, the columns that are not in the list are selected.

dt<<Invert Row Selection

Selects any row that is currently deselected and deselects any row currently selected.

dt<<Is Dirty

Returns 1 if the table has been modified from its saved state. Otherwise, returns 0.

dt<<Join(With(*Data Table(name)*), (<"Private">|<"Invisible">), Select(*columns*), Select With(*columns*), (By Matching Columns(*column1=column2, ...*)|"Cartesian"|"By Row Number"), <"Merge Same Name Columns">, <"Match Flag">, <Copy Formula(*Boolean*)>, <Suppress Formula Evaluation(*Boolean*)>, <"Update">, <Drop Multiples(*Boolean*, *Boolean*)>, <Include Non Matches(*Boolean*, *Boolean*)>, <"Preserve Main Table Order">, <Output Table Name(*name*))>

Description

Combines data tables *dt* and *Data Table* side to side.

Returns

A data table.

Required Arguments

With(*Data Table(name)*) Specifies the data table to join with the active table.

"Private" Specifies a quoted keyword that opens the data table without displaying it in a data table window.

"Invisible" Specifies a quoted keyword that hides the data table. Use this argument to keep the data table hidden but use it in a subsequent expression. The data table is displayed in the Home Window's Window List and the Window > Unhide list.

Select(*columns*) Selects the data table to join with the active table.

Select With(*columns*)

By Matching Columns(*column1=column2*) Selects columns in both tables whose values and data types match.

"Cartesian" Joins two tables using a Cartesian fashion, where it forms a new data table consisting of all possible combinations of the rows from two original data tables. JMP crosses the data in the first table with the data in the second to display all combinations of the values in each set.

"By Row Number" Joins the two tables side by side.

"Merge Same Name Columns" Data from the second table replaces the data of the same name columns in the original table. Note that missing values in the first table are replaced by nonmissing values in the second.

"Match Flag" Determines whether the Match Flag column is created when you are matching by column.

Copy Formula(*Boolean*) Includes formulas from the main table and/or the second table in the output columns.

Suppress Formula Evaluation(*Boolean*) Prevents JMP from evaluating columns' formulas during the creation of the new table.

"Update" Column data from the second table change the data of the same name columns in the original table. The results are displayed in a new data table. Note the following: JMP does not replace data with missing values; the output table uses the same columns as the original table. Thus, when you use "Update", Select Columns, the "Update" option is available only when joining by row number or by matching columns.

Drop Multiples(*Boolean*, *Boolean*) Specifies that you want the new table to contain only one row for each name. Applies only when matching by columns.

Include Non Matches(*Boolean*, *Boolean*) Includes non-matching columns in the main table and new data table. Applies only when matching by columns.

"Preserve Main Table Order" Maintains the order of the original data table in the joined table, instead of sorting by the matching columns.

Output Table Name(*name*) Specifies the name of the joined table. If you do not specify a name, JMP names the data table Untitled # (for example, Untitled 1).

---

dt<<Journal

Makes a journal from the data table. Only the data grid is included, not notes, variables, or scripts.

Note

Journals that are created in JMP 14 or later might contain compressed matrix data for large matrices. If you have JSL scripts that open journals and extract data from them, you might need to use the `Get Journal` message (which does not compress the matrices) rather than saving the journal to disk with the `Journal` message.

See Also

See "dt<<Get Journal"

`dt<<Journal Link(<Save(<path>)|Embed()>, <Button Name(<name>)>))`

Adds a link to the data table in the current journal. If a journal does not exist, a new one is created.

Optional Arguments

`path` Specifies the quoted *`path`* where the table is saved. If omitted, the data table should already have a disk location (previously saved or loaded), otherwise the journal link is incomplete and will not reload the table.

`Embed` Embeds a JSL script to recreate the data table.

`Button Name(name)` Specifies the name that is displayed on the button. The *`name`* argument is quoted. If the button name is not specified, the button is named after the data table.

`dt<<Label`

`dt<<Unlabel`

Toggles selected rows in *`dt`* from labeled to unlabeled or *vice versa*.

`dt<<Last Modified`

Returns the date on which the data table was last saved.

`dt<<Layout`

`Layout` is deprecated and will be removed in a future release. Use `Journal` instead.

`dt<<Lock Data Table`

Description

Locks the data table so that data and column properties cannot be added or changed.

See Also

"dt <<Set Edit Lock(<"Modify Cells">, <"Add Rows">, <"Add Columns">, <"Delete Rows">, <"Delete Columns">)"

dt<<Make Indicator Columns(<Append Column Name(*Boolean*)>, <Include Missing(*Boolean*)>)

Creates indicator columns of 0 and 1 values for the specified categorical columns.

Example

The following example creates indicator columns for the sex column. Append Column Name creates columns named sex_F and sex_M. Otherwise, the columns are named after each level (F and M). Include Missing includes missing values.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Make Indicator Columns(
   Columns( :sex ),
   Append Column Name( 1 ),
   Include Missing( 1 )
);
```

dt<<Make RowState Handler

Creates a row state handler function. The argument of the function holds the rows whose row states get changed.

dt<<Make SAS Data Step

Returns the data table as a SAS Data Step.

dt<<Make SAS Data Step Window

Returns the data table as a SAS Data Step and places it in a SAS script window.

dt<<Make Validation Column

Creates a column that is used to divide the data into training and validation sets.

dt<<Marker by Column(*column*)

Description

Assigns markers according to the values of the specified data table *column*.

See Also

"dt<<Marker By Column(column, <named arguments>);"

dt<<Markers(*n*)

Assigns marker *n* to the selected rows.

`dt<<Maximize Display`

Deprecated. Use `Optimize Display` instead.

Forces the data table to remeasure all of its columns and zoom to the best-sized window.

`dt<<Move Script Group(`*`group name`*`, "To First"|"To Last"|After(table `*`script name`*`)|After(`*`group`*`))`

Description

Rearranges the table script groups that are specified by the quoted *group name*.

`dt<<Move Selected Column(`*`name(s)`*`, "To First"|"To Last"|After(`*`name`*`))`

`dt<<Move Selected Columns(`*`name(s)`*`, "To First"|"To Last"|After(`*`name`*`))`

Description

Moves the selected column or columns in the data table to the specified position. The *name* argument is quoted.

Example

The following example moves the age column to the last column in Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Go To( :age );
dt << Move Selected Columns( "To Last" );
```

You may also use a list to specify the column names.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
list = {"name", "sex"};
dt << Move Selected Columns( list, "To Last" );
```

`dt<<Move Rows("At Start"|"At End"|After(`*`n`*`))`

Moves the selected rows in the data table to the specified position. *n* represents a row number.

`dt<<New Column(`*`name`*`, <`*`data type`*`>, <`*`modeling type`*`>, <Format(`*`format`*`, `*`width`*`)>, <Formula()>, <Set Values(`*`{..., ..., }`*`)>, <Set Property(`*`properties`*`)>)`

Description

Adds a new column titled with the quoted name after the last column in *dt*. Unless otherwise specified, columns are numeric, continuous, and 12 characters wide.

Returns

A column reference.

Required Argument

`name` The name of the new column.

Optional Arguments

`data type` A quoted string that specifies the data type. The options are `"Numeric"`, `"Character"`, `"Row State"`, or `"Expression"`.

`modeling type` A quoted string that describes the modeling type (`"Continuous"`, `"Nominal"`, `"Ordinal"`, `"Multiple Response"`, `"Unstructured Text"`, `"None"`, or `"Vector"`).

`Format(`*`format`*`, `*`width`*`)>` Sets the format type and column width.

`Set Values({})` Specifies the data in the column.

`Formula` Specifies the column formula.

`Set Property(`*`properties`*`)` Specifies any messages that data table columns support. Action arguments are found in the Column Properties menu in the New Column window. `Axis` and `Link Reference` are action argument.

See Also

For examples of setting other numeric format properties, see "col<<Format(<width>, <decimal places>, <"Use Thousands Separator">)".

---

dt<<New Data Box()

Makes a data table view in a display box tree. Useful for displaying the data table and report in one window. A data browser box is created when you send the `New Data Box` message to the data table object.

Example

The following script creates a data table view and report in one window. The data table is placed in a data browser box. The width of that box is set to 800 pixels. Because auto stretch is turned off, the data table view remains 800 pixels wide even if you stretch the right border of the window.

```
dtA = Open( "$SAMPLE_DATA/Semiconductor Capability.jmp", invisible );
nw = New Window( "Example",
  H List Box(
     V List Box( dtbox = dtA << New Data Box() ),
     dtA << Distribution(
        Continuous Distribution( Column( :NPN1 ) ),
        Continuous Distribution( Column( :PNP1 ) )
     )
  )
);
dtbox << Set Stretch( "Off", "Off" ) << Set Width( 800 );
```

dt<<New Data View

Opens a duplicate of the data table. The second data table is identical to and linked to the original data table, so that any changes made in one are reflected in the other. Closing either data table also closes the other and all references to the data tables are deleted.

This can be useful to show an invisible data table.

dt<<New Script(*name*, *script*)

dt<<Set Property(*name*, *script*)

Creates a new table property (also called a *table script*) using the quoted *name* that stores the specified *script*.

Use New Script() or Set Property() rather than the deprecated New Property() and New Table Property().

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Set Property( "Bivariate Example", Bivariate( Y( :weight ), X( :height
  ), Fit Line ) );
```

dt<<New Table Variable(*name*, *number*)

dt<<Set Table Variable(*name*, *number*)

Creates a new table variable with the quoted *name* and the *number*.

dt<<Next Selected

Scrolls data table down to show the next selected row that is not already in view.

dt<<Optimize Display

Forces the data table to remeasure all of its columns and zoom to the best-sized window.

dt<<Original Order

Restores saved order of columns in *dt*.

dt<<Paste Column Properties

Pastes multiple lists of column properties to multiple columns. Optionally, you can specify a list of target columns instead of selecting them in the data table.

Example

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
dt << Copy Column Properties( {:MODULUS, :ELONG} );
dt2 = New Table( "test it",
```

```
    New Column( "T1", numeric, continuous ),
    New Column( "T2", numeric, continuous ),
    New Column( "T3", numeric, continuous ),
    Add Rows( 10 )
);
dt2 << Paste Column Properties( {:T1, :T3} );
// pastes the column properties from MODULUS and ELONG to T1 and T3
```

`dt<<Predictor Screening`

`dt<<Screen Predictors`

Description

Used to identify strong predictors.

Example

```
dt = Open( "$SAMPLE_DATA/Boston Housing.jmp" );
obj = dt << Predictor Screening(
  Y( :mvalue ),
  X( :crim, :indus, :chas, :nox, :rooms, :age, :distance, :radial )
);
```

`dt<<Previous Selected`

Scrolls data table up to show the previous selected row that is not already in view.

`dt<<Print Window(<"Show Dialog">)`

Prints the window. If the optional named argument `"Show Dialog"` is specified, the print window is displayed. Otherwise, the window is printed to the default printer using the current settings, and no print window is displayed.

`dt<<Rename Script Group(`*`old name`*`, `*`new name`*`)`

Description

Renames the table script group.

Example

```
dt << Rename Script Group( "Maps", "Street Maps" );
```

`dt<<Reorder By Data Type`

Reorders columns in *dt*, row state first, then character, then numeric.

`dt<<Reorder By Modeling Type`

Reorders columns in *dt* to continuous, then ordinal, then nominal.

`dt<<Reorder By Name`

Reorders columns in *dt* to alphanumeric order by name.

`dt<<Rerun Formulas`

Recalculates all formula-based data table variables. Recalculations are performed in the proper dependency order.

`dt<<Reverse Order`

Reverses columns in *dt* from current order.

`dt<<Revert`

Reverts to the most recently saved version of *dt*.

`dt<<Row Selection(Select Where(condition), <current selection("Extend"|"Restrict"|"Clear")> <Dialog("Keep Dialog Open")>)`

Description

Selects all rows that meet the specified condition.

Required Argument

`Select Where(condition)` Specifies the condition by which the rows are selected.

Optional Arguments

`current selection("Extend"|"Restrict"|"Clear")` Extends, restricts, or clears the existing selections. `Clear` is the default value.

`Dialog("Keep Dialog Open")` Shows the dialog so that the user can edit the options.

`dt<<Run Formulas`

Performs all pending formula evaluations, including evaluations that are pending as a result of evaluating other formulas.

`dt<<Run Script(`*name*`)`

Finds the table property with the quoted name and runs it as a JSL script.

`dt<<Save(`*path*`)`

`dt<<Save As(`*path*`)`

Description

Saves the table in the specified quoted *path*.

`dt<<Save Database(connection information, table name, <"Replace">)`

Saves the data table to the database named using the quoted connection information and quoted table name. The `"Replace"` option replaces the existing database with the current database.

`dt<<Save Script to Script Window`

Saves a script to reproduce the data table in a script editor window. Appends the script to any script that currently appears in the script editor.

`dt<<Select All Rows`

Selects all rows in the data table.

`dt<<Select Columns(<column1>, <column2>,...|"All")`

Selects the specified columns (or all columns) in the data table.

`dt<<Select Duplicate Rows`

Description

Selects the second and subsequent duplicate rows. If columns are selected, duplicate values are found in the rows of those columns. The duplicate values are case sensitive.

`dt<<Select Excluded`

Selects only those rows in the data table that are currently excluded.

`dt<<Select Hidden`

Selects only those rows in the data table that are currently hidden.

`dt<<Select Labeled`

Selects only those rows in the data table that are currently labeled.

`dt<<Select Randomly(n|p|Sample Size(n)|Sampling Rate(p))`

Randomly selects the given percentage *p* of the rows in the data table, or the number of rows *n*. If you specify an argument with a keyword, numbers between 0 and 1 represent a percentage, and numbers larger than 1 represent a number of rows.

`dt<<Select Rows([row1, row2, ...])`

Selects the rows given in the list of row numbers.

dt<<Select Script Group(*<group name|{group1, group2, ...}>*)

Selects the table script group specified as a quoted *group name* or a list of quoted strings. If no argument is provided, all groups are selected.

dt<<Select Where(*condition*, <Current Selection("Extend"|"Restrict"|"Clear")>)

Description

Selects the rows in *dt* where the condition evaluates as true.

dt<<Set Dirty(*Boolean*)

Marks the data table as changed, even if no changes have been made.

dt <<Set Edit Lock(<"Modify Cells">, <"Add Rows">, <"Add Columns">, <"Delete Rows">, <"Delete Columns">)

Description

Prevents cells from being modified; rows from being added or deleted; and columns from being added or deleted.

dt <<Set Cell Height(*n*)

Sets the cell height to the specified number of pixels.

dt<<Set Header Height(*n*)

Sets the column header's height to the specified number of pixels.

dt<<Set Label Columns(*column1, columns2, ...*)

Assigns the specified columns as label columns.

dt<<Set Matrix(*[matrix]*)

Inserts the specified matrix into a data table, adding new columns and rows as necessary.

dt<<Set Name(*name*)

Description

Specifies a name for the table. The *name* argument is quoted.

Returns

The data table name as a quoted string.

Notes

A change was made to the `Set Name` message so that now the new table name is returned as a quoted string. In previous releases, `Set Name` returned a scriptable data table object. As a result of this change, JMP scripts might need to be updated for the desired result to be returned. For example, rewrite the following script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" ) << Set Name( "Test" );
```

Separate the messages so that *dt* represents the data table instead of "Test":

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Set Name( "Test" );
```

The result is the same as in previous releases but will run successfully in both earlier and newer versions of JMP.

`dt<<Set Property(`*name*`, `*script*`)`

See "dt<<New Script(name, script)".

`dt<<Set Label Columns(`*column(s)*`, ...)`

`dt<<Set Label Columns`

Turns on the Label attribute for the specified columns. If no columns are listed, it turns the Label attribute off.

`dt<<Set Row ID Width(`*n*`)`

Sets the row ID display width to the specified number of pixels. If *n* is set to zero, the row ID display width automatically resizes.

`dt<<Set Row States(`*[matrix]*`)`

Sets the row states for all rows in the data table.

`dt<<Set Scroll Lock Columns(`*column name, ...*`)`

Locks scrolling for the columns specified as quotes quoted strings. If no columns are listed, unlocks scrolling.

`dt<<Set Table Variable(`*name*`, `*value*`)`

See "dt<<New Table Variable(name, number)".

```
dt<<Sort(<"Private">|<"Invisible">, <"Replace Table">, By(columns),
Order("Descending" | "Ascending"), <Output Table Name(name))>
```

Description

Creates a new table (named after the quoted name) by rearranging the rows of *dt* according to the values of one or more *columns*.

Returns

A reference to the sorted table.

```
dt<<Split(Split(columns), Split By(column), <Group(column)>,
<"Private">|<"Invisible">, <Remaining Columns("Keep All"|"Drop All"|
Keep(columns)|Drop(columns))>, <Copy Formula(Boolean)>, <Suppress
Formula Evaluation(Boolean)>, <Sort by Column Property>, <Output
Table (name)>)
```

Description

Unstacks multiple rows for each `Split` column into multiple columns as identified by the `Split by` column. The `Split` and `Split by` arguments are required.

Returns

A reference to the split data table.

Required Arguments

`Split(columns)` The column to split.

`Split By(column)` The column to split by.

Optional Arguments

`Group` Splits data within the specified groups.

`Remaining Columns("Keep All"|"Drop All"| Keep(columns)|Drop(columns))` Specifies what to do with the remaining columns in the resulting table. `Keep All` is the default setting.

Note: `Keep All` includes all columns in the output data table. However, the *values* of every column are not included. Because multiple rows are collapsed to a single row in the output data table, some values of the kept columns are dropped.

`Copy Formula(Boolean)` Includes column formulas from the source table in the resulting table.

`Suppress Formula Evaluation(Boolean)` Stops any copied formulas from being evaluated. True is the default setting.

`Sort by Column Property` Sorts the order of the output columns by the sort column property that is defined for the Split by column.

`Output Table(name)` Generates the output to the specified table name.

dt<<Stack(<"Private">|<"Invisible">, Columns(*columns*), <Source Label Column(*quoted string*)>, <Stacked Data Column(*quoted string*)>, <Copy Formula(*Boolean*)>, <Suppress Formula Evaluation(*Boolean*)>, <Drop All Other Columns(*Boolean*)|Name(*non-stacked columns*)(Keep(*column1, column2, ...*))|Name(*non-stacked columns*) (Drop(*column1, column2, ...*))>, <Output Table(*name*)>), <Number of Series(*n*)>, <"Contiguous">

Description

Creates a new table by combining the values from several columns in *dt* into one column.

Returns

A reference to the stacked data table.

dt<<Subscribe("keyname"(<"*client*">), On Delete Columns(<*function*>|<*script*>)|On Add Columns(<*function*>|<*script*>)|On Add Rows(<*function*>|<*script*>)|On Delete Rows(<*function*>|<*script*>)|On Rename Column(<*function*>|<*script*>)|On Close(<*function*>|<*script*>)|On Save(<*function*>|<*script*>)|On Rename(<*function*>|<*script*>))

Description

Subscribes to a data table to get messages regarding changes in the data table.

Returns

The keyname.

Arguments

"keyname"(<*client*>) Specifies the subscription name so that it can be referenced. The quoted client triggers a close confirmation when a close is attempted on the data table, warning that other open windows depend on the data table.

On Delete Columns(<*function*>|<*script*>) Returns the keyname when columns are deleted.

On Add Columns(<*function*>|<*script*>) Returns the keyname when columns are added.

On Add Rows(<*function*>|<*script*>) Returns the keyname when rows are added.

On Delete Rows(<*function*>|<*script*>) Returns the keyname when rows are deleted.

On Rename Column(<*function*>|<*script*>) Returns the keyname when columns are renamed.

On Close(<*function*>|<*script*>) Returns the keyname when the data table is closed. Takes one argument, a function. The function requires only one argument, the data table name.

On Save(<*function*>|<*script*>) Returns the keyname when the data table is saved.

On Rename(<*function*>|<*script*>) Returns the keyname when a rename is attempted on the data table. The function can be either the name of a previously defined function or the function itself.

Notes

Each subscription option remains in effect until you unsubscribe.

---

dt<<Subset(<"Private">|<"Invisible">, <"Selected Columns">, <Columns(*column list*)>, <"Selected Rows">, <Rows(*[number, number, ...]*)>, <By(*column list*)>, <Sampling Rate(*fraction*)>, <Sample Size(*integer*)>, <Stratify(*column list*)>, <Link to Original Data Table(*Boolean*)>, <Copy Formula(*Boolean*)>, <Suppress Formula Evaluation(*Boolean*)>, <"Keep by Columns">)

Description

Creates a new table from the rows and columns that you specify in *dt*.

Returns

A reference to the subset data table.

---

dt<<Summary(<"Private">|<"Invisible">, <Group(*column*)>, <Subgroup(*column*)>, <N(*column*)>, <Mean(*column*)>, <Std Dev(*column*)>, <Min(*column*)>, <Max(*column*)>, <Range(*column*)>, <Sum(*column*)>, <CV(*column*)>, <Freq(*column*)>, <Weight(*column*)>, "Include Marginal Statistics", <Link to Original Data Table(*Boolean*)>, <Statistics Column Name Format(*Stat(column)|Column|Stat of Column|Column Stat*)>)

Description

Creates a new table of summary statistics for the column that you specify, according to groups and subgroups. `Statistics Column Name Format` values are quoted.

Returns

A reference to the summary data table.

---

dt<<Suppress Formula Eval(*Boolean*)

Turns off automatic calculation of formulas for data table *dt*.

---

dt<<Text to Columns(Delimiters(<*separator*>, <"Tab">, <"Newline">), Columns(*column1, column2...*))

Makes a set of text columns or indicator columns from a delimited text column. "newline" includes the three forms: \r, \n, and \r\n. The separator is quoted.

Example

```
dt = Open( "$SAMPLE_DATA/Consumer Preferences.jmp" );
dt << Text To Columns(
```

```
    delimiter( "," ),
    columns( :Brush Delimited )
);
```

`dt<<Transpose(Columns(`*columns*`), Rows([`*matrix*`]), Output Table Name(`*name*`))`

Description

Creates a new table (named after the quoted *name*) from the rows and columns that you specify.

Returns

A reference to the transposed data table.

`dt<<Ungroup Columns({`*column1, column2, ...*`})`

Ungroups the columns defined in the list argument.

`dt<<Ungroup Scripts(`*Name of Script Group*`|{`*script1, script2,...*`})`

Description

Removes the specified table scripts or group from the group. The *Name of Script Group* argument is quoted.

`dt<<Unsubscribe(`*keyname*`, "On Delete Columns"|"On Add Columns"|"On Add Rows"|"On Delete Rows"|"On Close"|"On Col Rename"|"All")`

Releases any previous subscriptions to the data table *dt*. The *keyname* argument is quoted.

`dt<<Update from Database`

Updates the data in the table *dt* with data reimported from the database.

Columns

`col<<Add Column Properties(`*name*`, `*expression*`)`

Adds the quoted column property *name* with the *expression* given. You can add any standard column property by name or a user-specified property.

`col<<Add From Row States`

Updates a row state column with any currently used row state changes that are not the default state.

---

col<<Add To Row States

Copies all row state values in a column that are not the default state to the currently used row state in the data table.

---

col<<Color Cells(*color*)

Description

Colors the cells of the column within the data table grid. Use any quoted named color or 0 to clear the color.

---

col<<Color Cell by Value(*Boolean*)

Description

Colors the cells of the column in the data table grid using the value color property.

---

col<<Copy Column Properties

Copies the column properties into the buffer.

---

col<<Copy From Row States

Copies all row state values currently used in the data table to a column.

---

col<<Copy to Row States

Copies all row state values in the column to the currently used row state in the data table.

---

col<<Data Type(*type*, <Format(*format quoted string*)>, <Input Format(*format quoted string*)>, <*width*>)

col<<Set Data Type(*type*, <Format(*format quoted string*)>, <Input Format(*format quoted string*)>, <*width*>)

Description

Sets the *data type* to *col*.

Required Argument

type Specifies the "Numeric", "Character", "Row State", or "Expression" data type.

Optional Arguments

Format(*format quoted string*) Specifies the way the data are displayed, such as h:m for hours and minutes. The *format quoted string* argument is quoted.

Input Format(*format quoted string*) Specifies the way the data are input. The *format quoted string* argument is quoted.

width (Optional for numeric data) Specifies 1, 2, or 4 (the number of bytes in the column).

`col<<Delete Formula`

Deletes the formula from a column.

`col<<Delete Property(`*name*`)`

`col<<Delete Column Property(`*name*`)`

Deletes the quoted property name from a column.

`col<<Eval Formula`

Forces the formula to evaluate (perhaps again). If formula suppression is enabled, the evaluation is not performed.

`col<<Exclude(`*Boolean*`)`

Turns the excluded or unexcluded state on, depending on the Boolean argument.

`col<<Format(<`*width*`>, <`*decimal places*`>, <"Use Thousands Separator">)`

`col<<Format("Best", <`*width*`>, <"Use Thousands Separator">)`

`col<<Format(("Fixed Dec"|"Percent"), <`*width*`>, <`*decimal places*`>, <"Use Thousands Separator">)`

`col<<Format("Pvalue", <width>)`

`col<<Format(("Scientific"|"Engineering"|"Engineering SI"), <`*width*`>, <`*decimal places*`>)`

`col<<Format("Precision", <`*width*`>, <`*decimal places*`>, <"Use Thousands Separator">, <"Keep Trailing Zeroes">, <"Keep All Whole Digits">)`

`col<<Format("Currency", <"Currency Code">, <`*width*`>, <`*decimal places*`>, <"Use Thousands Separator">)`

`col<<Format("Datetime", <`*width*`>, <`*input format*`>)`

`col<<Format(("Latitude DDD"|"Latitude DDM"|"Latitude DMS"|"Longitude DDD"|"Longitude DDM"|"Longitude DDM"), <`*width*`>, <`*decimal places*`>, ("PUN"|"DIR"|"PUNDIR"))`

`col<<Format("Custom", Formula(...), <`*width*`>, <`*input format*`>)`

Description

Sets the numeric display specified format.

Arguments

See *Using JMP* for more information about the arguments.

Examples

```
col<<Format( 10, 2, "Use thousands separator");
col<<Format( "Currency", "EUR", 20 );
col<<Format( "m/d/y", 10 );
col<<Format( "Precision", 10, 2, "Keep trailing zeroes", "Keep all whole
  digits" );
col<<Format( "Latitude DDD", "PUNDIR"); // "PUN" for punctuation, "DIR" for
  direction, PUNDIR for both
col<<Format( "Custom", Formula( Abs( value ) ), 15 );
```

`col<<Formula(`*expression*`)`

`col<<Set Formula(`*expression*`)`

Sets the formula for the variable and evaluates it.

`col<<Get Column Field Width`

Returns the field width used for displaying data in the column.

`col<<Get Data Type`

Returns the data type of *col*.

`col<<Get Data Type Length`

Returns the data type and length of the data column. Only the data type is returned if the data length is not fixed, as with character columns.

`col<<Get Format`

Returns the format of the column.

`col<<Get Formula`

Returns the formula.

`col<<Get Hidden`

Returns 1 if the column is hidden.

`col<<Get Input Format`

Returns the format used for input and storing of data for the column.

`col<<Get Labeled`

Returns 1 if the column is labeled.

`col<<Get List Check`

Returns the list check definition. If list check is not defined for the column, a message is sent to the log stating so.

`col<<Get Lock`

Returns the current Lock setting.

`col<<Get Modeling Type`

Returns the modeling type of the column.

`col<<Get Name`

Returns the name of the column.

`col<<Get Property("property name")`

Returns the specified property definition. If the specified property is not defined for the column, a message is sent to the log stating so.

`col<<Get Range Check`

Returns the range check definition. If range check is not defined for the column, a message is sent to the log stating so.

`col<<Get Role`

Returns the preselected role of *col*.

`col<<Get Script`

Returns the script to reproduce the column.

`col<<Get Scroll Locked`

Returns 1 if the column is scroll locked.

`col<<Get Selected`

Returns 1 if the column is selected, or 0 otherwise.

`col<<Get Stored Values`

Returns the values in the columns without considering the Missing Value Codes column property.

`col<<Get Value Labels`

Returns the value labels definition. If value labels is not defined for the column, a message is sent to the log stating so.

`col<<Get Use Value Labels`

Returns 1 if the value labels are set to be used for the column, or 0 otherwise.

`col<<Get Values`

Returns the values in the column.

`col<<Hide(Boolean)`

Turns the Hide attribute on or off according to the Boolean argument given.

`col<<Ignore Errors`

Ignores formula evaluation errors in a column, and sets the cell value to missing when a formula error occurs.

`col<<Input Format(`*`format`*`)`

Sets the quoted *`format`* used for input and storage for the column. The argument is the name of any JMP format (for example, `"ddmmyyyy"` for a date column).

`date_col<<Is Transformed On SAS Export`

Returns true if the data in the resulting SAS data set for the date column will be changed when it is exported to SAS.

`col<<Label(`*`Boolean`*`)`

Turns the Label attribute on or off according to the Boolean argument given.

`col<<Lock(`*`Boolean`*`)`

`col<<Set Lock(`*`Boolean`*`)`

Turns the Lock attribute on or off according to the Boolean argument given.

`col<<Preselect Role(role)`

Preselects the specified *role* for the column. Choices are `"Y"`, `"X"`, `"Weight"`, `"Freq"`, and `"None"`, or `"No Role"`.

`col<<Reset Transform`

Removes the cached data for the transform column. Accessing column data rebuilds the caches. Use this option to reduce memory or to allow recalculation if the formula depends on external information.

`col<<Set Display Width(n)`

Sets the column display width to the *n* in pixels. If *n* is set to zero, the column display width automatically resizes.

`col<<Set Each Value(n)`

Sets all the values in the column to *n*.

`col<<Set Excluded`

Excludes the column.

`col<<Set Field Width(n)`

Sets the field width for the column to *n*.

`col<<Set Hidden`

Hides the column.

`col<<Set Labeled`

`col<<Set Labelled`

Uses the column's data values for labels.

`col<<Set Modeling Type(type)`

Sets the modeling type for the variable. Choices are `"Continuous"`, `"Ordinal"`, `"Nominal"`, `"None"`, `"Row State"`, `"Unstructured"`, `"Multiple Response"`, or `"Vector"`.

`col<<Set Name(name)`

Sets the name for the column. The *name* argument is quoted.

`col<<Set Property(name, expression)`

Sets the quoted property name to the *expression* given. You can set any standard column property by name or a user-specified property.

Examples

The following example adds the Value Colors column property to the sex column, with pink for females and blue for males.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column( "sex" ) << Set Property( "Value Colors", {"F" = 78, "M" = 69} );
```

The following example adds a custom column property named *Date recorded* to the height column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column( "height" ) << Set Property( "Date recorded", 05Jan1990 );
```

See Also

Scripting Guide

Using JMP

`col<<Set Scroll Locked(Boolean)`

Turns the Scroll Lock attribute on or off according to the Boolean argument given.

`col<<Set Selected(Boolean)`

Sets the column to be selected or not selected.

`col<<Set Use for Marker`

`col<<Use for Marker`

Uses the values in the column as markers in graphs. Designed to use with expression columns and character columns that have IDs. In the Big Class Families.jmp sample data table, the picture column is specified to use as markers in graphs. Not supported in Bubble Plot.

`col<<Set Values([matrix] or {list})`

`col<<Values([matrix] or {list})`

Sets values for the matrix (for numeric variables) or list (for character variables).

`col<<Suppress Eval(Boolean)`

Turns off automatic calculation of formulas for the column.

col<<Use For Marker(*Boolean*)

Uses the values in the column as markers in graphs or turns off the option. Designed to use with expression columns and character columns that have IDs. In the Big Class Families.jmp sample data table, the picture column is specified to use as markers in graphs.

Rows

row<<Colors(*n*)

Assigns the color *n* to the selected rows.

row<<Exclude(*Boolean*)

row<<Unexclude(*Boolean*)

Turns the excluded or unexcluded state on for the selected rows according to the Boolean argument given. Omit the argument to toggle the row state.

row<<Hide(*Boolean*)

row<<Unhide(*Boolean*)

Turns the Hide attribute on or off according to the Boolean argument given. Omit the argument to toggle the row state.

row<<Hide and Exclude

Shows or hides the selected rows from appearing on graphs, and excludes or unexcludes them from contributing to calculations.

row<<Label(*Boolean*)

row<<Unlabel(*Boolean*)

Turns the Label attribute on or off according to the Boolean argument given. Omit the argument to toggle the row state.

row<<Markers(*marker*)

Assigns the quoted "*marker*" to the selected rows.

row<<Next Selected

Causes the next selected row in the data table to blink.

`row<<Previous Selected`

Causes the previous selected row in the data table to blink.

`row<<Row Editor`

Opens the Row Editor window for the selected rows.

Data Filter

`dtf<<Add Favorites(`*name*`)`

Description

Associates the current filter selection with the quoted *"name"* and saves it in the Favorites list.

Returns

The favorite as a quoted string.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
df = dt << Data Filter(
   Add Filter(
      Columns( :age, :sex, :height, :weight ),
      Where( :sex == "F" ),
      Where( :height >= 55 & :height <= 65 )
   ),
   Mode( Select (1) )
);
Wait( 1 ); // for demonstration purposes
fav1 = df << Add Favorites( "Female Average Ht" );
```

`dtf<<Add Filter(Columns(`*column1*`, <`*column2*`>), <Where(`*clause*`)>)`

Add one or more filter columns in a new OR group.

`dtf<<Auto Clear(`*Boolean*`)`

Clears all currently selected rows before setting a new selection.

`dtf<<Clear`

Clears the currently selected rows.

`dtf<<Close`

Closes the data filter window.

`dtf<<Columns(column1, column2, ...)`

Sets the columns to use in the data filter.

`dtf<<Data Table Window`

Shows the data table that the data filter window is using.

`dtf<<Delete All`

Removes all filters that are set.

`dtf<<Delete(column1, column2, ...)`

Removes the specified columns from the data filter.

`dtf<<Display(column, <Size(x, y)>, "Blocks Display"|"List Display"|"Single Category Display"|"Checkbox Display")`

Sets how the specified categorical column levels are displayed in the filter.

`dtf<<Get Script`

Returns the data filter script as text in the log.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
df = dt << Data Filter(
     Add Filter( Columns( :age, :sex ), Where( :age == 12 ) )
);
txt = df << Get Script;
Show( txt );
```

`dtf<<Local Data Filter`

Embeds the data filter in the specified window. See *Scripting Guide* for more information about local data filters.

`dtf<<Location(x, y)`

Moves the data filter window to the specified location. *x* and *y* are measured in pixels. 0,0 is the top left of the monitor.

dtf<<Make Filter Change Handler(*function*)

Creates a data filter handler to handle notification that the filter has changed. The number of rows filtered is returned in the argument to the function.

Example

```
dt = Open( "$SAMPLE_DATA/PopAgeGroupSubset.jmp" );
dist = Distribution( Automatic Recalc( 1 ), Continuous Distribution( Column(
   :POP ) ) );
filter = dist << Local Data Filter( Add Filter( Columns( :Region ) ) );
f = Function( {a}, Print( a ) );
rs = filter << Make Filter Change Handler( f );
```

dtf<<Make Subset

Creates a new subset data table that contains the rows that are selected in the data filter.

dtf<<Match(Filter Columns(*column1*, column2, ...), Where(*clause*)

Sets the filter conditions for each column. The *Where clause* is used for all the columns listed. To use different Where clauses for different columns, send the Match message separately for each column.

dtf<<Mode(Select(*Boolean*)|Show(*Boolean*)|Include(*Boolean*))

Sets the action, or mode, that is used when rows are selected using the data filter.

dtf<<Save and Restore Current Row States

Saves the current row states for the data table, and then restores those states when the data filter is closed.

dtf<<Show Columns Selector(*Boolean*)

Displays or hides the column selector after completing a filter.

dtf<<To Clipboard

Creates a Where clause from the current state of the data filter and places it on the clipboard, where it can be pasted elsewhere.

dtf<<To Data Table

Creates a Where clause from the current state of the data filter and saves it as a property to the data table.

`dtf<<To Journal`

Creates a Where clause from the current state of the data filter and appends it to the current journal. If there is no current journal, a new journal is opened and the Where clause is added to it.

`dtf<<To Row State Column`

Creates a row state column whose formula is the Where clause.

`dtf<<To Script Window`

Creates a Where clause from the current state of the data filter and appends it to the current script window. If there is no current script window, a new script window is opened and the Where clause is added to it.

`dtf<<Use Floating Window(`*`Boolean`*`)`

Sets whether the data filter window floats on top of its associated data table or behaves as a normal window.

`dtf<<Where(`*`clause`*`)`

Sets a condition for selecting rows.

Data Feed (Windows Only)

`feed<<Close`

Closes the data feed object and its window.

`feed<<Connect(`*`port settings`*`)`

Sets up port settings for the connection to the device.

`feed<<Disconnect`

Disconnects the device from the data feed queue but leaves the data feed object active.

`feed<<EOL("CR", "LF", "CRLF")`

Sets the line ending value used as a separator when parsing incoming lines of data. The value is also used as the terminator in outgoing lines of data.

– `"CR"`: ASCII character 13 (carriage return)

– "LF": ASCII character 10 (line feed)
– "CRLF": Uses both CR and LF in sequence.

`feed<<Get Line`

Returns and removes one line from the data feed queue.

`feed<<Get Lines`

Returns as a list and removes all lines from the data feed queue.

`feed<<Print Queue`

Prints the internal queue of messages to the log window.

`feed<<Queue Line(`*`quoted string`*`)`

Sends one quoted *`string`* (or *line*) to the end of the data feed queue. `Queue Line` is primarily useful for testing your script without requiring it to be attached to a device. You can essentially simulate the data coming from the device to make sure the rest of your code handles the values properly when it's really attached to a working device.

`feed<<Restart`

Restarts processing queued lines.

`feed<<Set Script(`*`script`*`)`

Assigns the *`script`* that is run each time a line of data is received.

`feed<<Stop`

Stops processing queued lines.

`feed<<Write(`*`quoted string`*`)`

Description

Sends a quoted *`string`* to the data feed device.

Example

```
exfeed = Open Datafeed(
   Connect( Port( "com1" ), Baud rate( 4800 ), Parity( "even" ), DataBits( 8 )
   ),
   Set Script(
      ex = exfeed << Get Line;
      Show( ex );
```

```
    )
);
exfeed << Write( "Ready" );
/* Example - send a message to external device over the serial port to trigger
   data messages. This can be used to send control messages to a sensor or
   other attached device.*/
```

feed<<Write Line(*quoted string*)

Description

Sends a quoted *string* to the data feed device. If EOL has been set for the data feed, the quoted strings are terminated by the specified EOL value. If EOL has not been set, the line is terminated with CRLF.

Example

```
exfeed = Open Datafeed(
   Connect( Port( "com1" ), Baud rate( 4800 ), Parity( "even" ), DataBits( 8 )
   ),
   Set Script(
      ex = exfeed << Get Line;
      Show( ex );
   )
);
exfeed << Write Line( "Ready" );
/* Example - send a message to external device over the serial port to trigger
   data messages. This can be used to send control messages to a sensor or
   other attached device.*/
```

feed<<Write Lines({*quoted string1, quoted string2, quoted string3*})

Description

Sends a list of quoted *strings* to the data feed device. If EOL has been set for the data feed, the quoted strings are terminated by the specified EOL value. If EOL has not been set, the line is terminated with CRLF.

Example

```
exfeed = Open Datafeed(
   Connect( Port( "com1" ), Baud rate( 4800 ), Parity( "even" ), DataBits( 8 )
   ),
   Set Script(
      ex = exfeed << Get Line;
      Show( ex );
   )
);
exfeed << Write Lines( {"Ready", "Set", "Go"} );
```

```
/* Example - send a message to external device over the serial port to trigger
   data messages. This can be used to send control messages to a sensor or
   other attached device.*/
```

Display Boxes

For additional examples, see the JMP Scripting Index.

All Display Boxes

---

db<<Add Text Annotation(Text(*quoted string*), Text Box(<*x1, y1, x2, y2*>))

Draws a text annotation box at the specified pixel location that contains the quoted *string*. The Text Box argument controls where the text annotation box is drawn in the window, from the upper left corner to the lower right corner.

Note that *x1*, *y1*, *x2*, and *y2* are not graph axis values but the specific pixel locations in the window. Exactly where the text box appears depends on the user's window size, display resolution, and so on.

---

db<<Append(*db2*)

Adds *db2* as the last child of the *db*.

---

db<<Child

Returns the child of the box.

---

db<<Class Name

Returns the name of the display class for the box.

---

db<<Clone Box

Makes a new copy of the display box.

---

db<<Close Window

Closes the containing window.

---

db<<Copy Picture

Puts a picture of the box on the clipboard.

db<<Delete

Deletes the display box.

db<<Enable(*Boolean*)

Controls the ability to interact with the display box. 0 disables the display box. 1 enables the display box.

db<<Get HTML

Returns a quoted string containing HTML source for the box.

db<<Get Journal

Returns a quoted string containing journal source for the box.

db<<Get Menu Item State(*index*)

Returns the popup menu item state of the index menu item. The state can be normal (0), checked (1), or disabled (-1).

db<<Get Menu Items

Description

Returns the menu items used for popup menu when the button is clicked. Menu items are returned in a list.

See Also

For submenus see "db<<Get Submenu(index)".

db<<Get Menu Script

Returns the menu script attached to the calling object.

db<<Get Page Setup()

Returns the page setup settings.

Example

The example below creates a new window and returns the page setup configuration.

```
w = New Window( "Window",
   Text Box( "Page Setup Test" )
);
w << Get Page Setup();
```

The results of the message:

```
{Margins( {0.75, 0.75, 0.75, 0.75} ), Scale( 1 ), Portrait( 1 ),
Paper Size( "Letter" )}
```

`db<<Get Picture( <Scale(n)> )`

Captures *db* as a picture object. The `Scale(n)` argument is a factor of the original picture size. For example, `Scale(2)` makes the picture object twice as large.

`db<<Get RTF`

Returns a quoted string containing RTF source for the box.

`db<<Get Script`

Returns the script for recreating the display box.

`db<<Get Size`

Returns either { *x, y* } or { *h, v* } in pixels:

```
xy = DisplayBox << Get Size;
```

Returns x and y in pixels:

```
{ x, y } = DisplayBox << Get Size;
```

`db<<Get Submenu(index)`

Returns the number of submenu items under the given menu item.

Example

The example below creates a menu containing *"A"*, *"B"*, and *"C"* with *"A"* having a submenu *"A1"* and *"A2"* and *"B"* having a submenu *"B1"*, *"B2"*, and *"B3"*. `<<Get Submenu(inc)` returns the number of submenu items under each indexed menu item.

```
New Window( "Title",
obj = Outline Box( "title" ) );
submenus = { };
obj << Set Menu Script(
   {"A", "", "A1", Print( "A1" ), "A2", Print( "A2" ),
   "B", "", "B1", Print( "B1" ), "B2", Print( "B2" ), "B3", Print( "B3" ),
   "C", Print( "C" )}
);
obj << Set Submenu( 1, 2 ); // menu A with 2 items in submenu A1 and A2
obj << Set Submenu( 4, 3 ); // menu B with 3 items in submenu B1, B2, and B3
For( inc = 1, inc <= N Items( Words( obj << Get Menu Script, "," ) )/2, inc++,
   Insert Into( submenus, obj << Get Submenu( inc ) );
);
submenus;
   {2, 0, 0, 3, 0, 0, 0, 0}
```

The log output indicates that index(1) contains two submenu items and index(3) contains three submenu items.

`db<<Get Text`

Returns a quoted string containing the text of the box.

`db<<Horizontal Alignment(`*`position`*`)`

Aligns a child display box inside the display parent box according to the specified *`position`*. The default value is `"Left"`, or you can specify `"Center"`, or `"Right"`.

Example

```
New Window( "Example",
   Outline Box( "Parent display box",
      Button Box( "OK", <<Horizontal Alignment( "Center" ) )
   )
);
```

`db<<Inval`

Invalidates the display box area in the window. The window is updated the next time the operating system has an opportunity to update windows (for example, when the user resizes the display box).

Notes

Consider including the message `<<Update Window` rather than including `Wait(0)`. The problem with using `Wait(`*`n`*`)` is knowing how large *`n`* should be.

Many display box messages, such as `<<Set Text`, automatically mark the box as invalid, so the `<<Inval` message is usually unnecessary. Some interactive scripts that use sliders with JSL callbacks might need `<<Update Window` to keep various parts of the display synchronized with the slider.

`db<<Is Enabled`

Returns the enabled state of the control. The message is supported in `Busy Light Box()`, `Button Box()`, `Calendar Box()`, `Check Box()`, `Col List Box()`, `Combo Box()`, `Completion Box()`, `Filter Col Selector()`, `gtext()`, `List Box()`, `Number Edit Box()`, `Popup Box()`, `Radio Box()`, `Range Slider Box()`, `Slider Box()`, `Spin Box()`, `Text Edit Box()`, `Tree Box()`, `Tree Map Box()`, and `Tree Map Seg()`.

`db<<Journal`

Appends the box to the journal.

`db<<Journal Window`

Appends the containing window of the display box to the journal; compare with `Journal`.

`db<<Move Window(x, y)`

Moves the window to the (*x, y*) location on your screen.

`db<<Page Break`

Inserts a page break before the box.

`db<<Parent`

Returns the parent of this display box.

`db<<Prepend(db2)`

Add *db2* to the display tree before *db*.

`db<<Prev Sib`

Returns the previous sibling of the display box.

`db<<Reshow`

Invalidates the display box's area in the window and immediate removes invalid areas from the window.

`db<<Save Capture(<path>, <format>, <Add Sibling(n)>)`

Saves the display box as a graphic to the specified quoted *path* in the specified quoted *format*. The optional `Add Sibling` argument adds the number of sibling display boxes to include in the capture. The default value is 1, which captures only the specified display box. Note that the specified portion of the report is not guaranteed to be scrolled into view or unobstructed by other windows. If the display box is not visible, the saved graphic will not contain the contents that you expect.

If you omit the path, you are prompted to name and save the file when running the path.

`db<<Save HTML(<path>, <format>)`

Saves the HTML source and folder of graphics to the quoted *path* and in the quoted *format*. If you omit the *path* argument, you are prompted to name and save the file when running the script.

db<<Save Interactive HTML(*<path>*, "Is Static")

Saves the display box as a web page (that includes interactive HTML features) in the quoted *path*. Non-JMP users can then explore the data. Note that the data is embedded in the web page.

Arguments

path A optional quoted *path* that specifies the location where the web page will be saved.

"Is Static" Omits the data from the web page and saves a static version of the web page.

Examples

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( y( weight ), x( height ) );
rbiv = (biv << Report);
rbiv << Save Interactive HTML( "$DOCUMENTS/MyInteractiveHTML.htm" );
```

db<<Save Journal(*<path>*)

Saves the journal source for the box in the quoted *path*. If you omit the argument, you are prompted to name and specify the graphic type.

db<<Save MSWord(*<path>*)

(Windows Only) Saves the display box as a Microsoft Word document in the quoted *path*. If you omit the *path* argument, you are prompted to name and save the file when running the script.

db<<Save PDF(*<path>*, <Show Page Setup(*Boolean*)>, <Portrait(*Boolean*)>)

Description

Saves a PDF of the display box in the quoted *path*.

Optional Arguments

path Saves the file in the quoted *path*. If you omit the argument, you are prompted to name and save the file when running the script.

Show Page Setup(*Boolean*) (Windows only) Displays the Page Setup window, where you can specify page orientation, headers and footers, margins, page scale, and paper size.

Portrait(*Boolean*) Displays the content in portrait or landscape orientation.

Notes

The PDF file contains headers and footers. Use Save Picture to omit these components.

`db<<Save Picture(<path>, <format>)`

Description

Saves a picture of the display box in the quoted *path* and with the specified quoted format.

Notes

- If you omit the quoted *path* argument, you are prompted to name and save the file when running the script.
- Valid file formats include `"PDF"`, `"PNG"`, `"GIF"`, `"JPG"` or `"JPEG"`, `"EPS"`, `"SVG"`, and `"EMF"`.
- On Windows, the Windows Specific preferences determine the resolution (or DPI), or you can run the following script:

  `Pref( Save Image DPI( number ) );`
- On macOS, the operating system determines the DPI.
- Use `Save Picture` to export a report as a PDF file with no headers or footers. Use `Save PDF` to include these components.

`db<<Save Presentation(<path>, <Template(path)>, <Insert("Begin"|"End"|n)|Replace("Begin"|"End"|n)|Append>, <Outline Titles(title location)>, <format>)`

Saves display boxes in a Microsoft PowerPoint presentation. You can open the file in any presentation software program.

Optional Arguments

`path` Saves the file in the quoted *path*. You must include the .pptx extension in the filename. If you omit the *path* argument, you are prompted to name and save the file when running the script.

`Template(path)` Specifies the quoted *path* of a custom PowerPoint template. Without this argument, JMP uses the default template located in the `pptx` folder of the installation directory.

Include a simple table in your template, or a default table format is applied to report tables. For an example on Windows, see `/pptx/JMPExportTemplate.pptx` in the JMP installation folder.

`Insert` Determines where the slides are inserted in an existing presentation.

– *n* inserts the slides as the *n*th slide number.

– `"Begin"` inserts the slides at the beginning of the presentation.

– `"End"` inserts the slides at the end of the presentation.

`Replace` Determines which slides are replaced in an existing presentation. The arguments are *n*, `"Begin"`, and `"End"` as described for `Insert`.

`Append` The slides are inserted at the end of an existing presentation.

`Outline Titles` The location of the outline title and any parent outline titles on the slide. By default, the immediate parent outline title appears as a slide title above the slide content, with any parent outline titles positioned in the bottom left corner of the slide.

- `"None"` omits the slide title above the graphic and the outline titles.
- `"Hide"` omits the outline titles.
- `"TopLeft"`, `"TopRight"`, `"BottomLeft"`, `"BottomRight"` determine the position of any of the parent outline titles on the slide.

`format` The format of the embedded graphics. Options are `"Native"`, `"EMF"`, `"PNG"`, `"JPG"`, `"BMP"`, `"GIF"`, `"TIF"`. On Windows, the native format is EMF. On macOS, the native format is PDF. See "Notes" for compatibility issues. Without this argument, JMP applies the "Image Format for PowerPoint" General preference.

Notes

Windows does not support the native PDF graphics produced on macOS. macOS does not support the native EMF graphics produced on Windows. For cross-platform compatibility, specify `"PNG"`, `"JPG"`, `"GIF"`, or `"TIF"`.

If no arguments are provided, the user is prompted to name and save the file.

`db<<Save RTF(<path>, format)`

Saves the file in the specified `quoted path` and with the quoted `format`. If you omit the `path` argument, you are prompted to name and save the file when running the script.

`db<<Save Text(<path>, format)`

Saves a file containing the text of the box in the quoted `path` and with the specified quoted `format`. If you omit the `path` argument, you are prompted to name and save the file when running the script.

`db<<Scroll Window(Display Box|relative-vertical-pixels|relative-horizontal-pixels, relative-vertical-pixels|{absolute-vertical-pixels, absolute-horizontal-pixels})`

Scrolls the containing window.

`db<<Select`

`db<<Deselect`

Selects (highlights) or deselects the box.

db<<Set Menu Item State(*index*, 0|1|-1)

Sets the popup menu item at *index* to be normal (0), selected (1), or disabled (-1).

db<<Set Page Setup<Margins(*left*, *right*, *top*, *bottom*)>, <Scale(*s*)>,<Portrait(*Boolean*)>, <Paper Size(*paper size*)>)

db<<Set Page Setup<Margins({*left*, *right*, *top*, *bottom*})>, <Scale(*s*)>,<Portrait(*Boolean*)>, <Paper Size(*paper size*)>)

Sets the page settings. Margins are set in inches. Scale variable *s* is a number in the range of 10 (for 1000%) to 0.2 (for 20%) with the default as 1 (for 100%). If `Portrait` is True the page is oriented for portrait, otherwise the page is landscape. Paper Size is a quoted string specifying the paper size, for example, `"Letter"` or `"Legal"`.

Example

The example below creates a new window and configures the page setup.

```
w = New Window( "Window",
   Text Box( "Page Setup Test" )
);
w << Set page setup(
   margins( 1, 1, 1, 1 ),
   scale( 1 ),
   portrait( 1 ),
   paper size( "Letter" )
);
```

db<<Set Print Headers(*left header*, *center header*, *right header*)

Description

Sets the left, center, and right header for print output.

Example

```
w = New Window( "Window", Text Box( "Header Example" ) );
w << Set Print Headers(
   "Today is: &d;", // left
   "&wt;", // center
   "Page &pn; of &pc;" // right
);
w << Print Window;
```

db<<Set Print Footers(*left footer*, *center footer*, *right footer*)

Description

Sets the left, center, and right footer for print output.

Example

```
w = New Window( "Window", Text Box( "Footer Example" ) );
```

```
w << Set Print Footers(
   "Today is: &d;", // left
   "&wt;", // center
   "Page &pn; of &pc;" // right
);
w << Print Window;
```

db<<Set Stretch(x,y)

Description

Sets the horizontal and vertical stretching of display boxes.

Arguments

"Window" The box stretches to fill the space provided by the window, The sizing depends on the minimum and maximum properties for the box.

"Fill" The box stretches to fill the available space in its container.

"Off" The box does not stretch.

"Neutral" Many boxes start in the neutral state. A leaf box (a box that has no children) that is neutral resolves to "Off". Most container boxes aggregate the stretching behavior (and minimum and maximum) of their child boxes.

db<<Set Submenu (*index*, *submenu count*)

Description

Sets the submenu items for the item (specified by index number) by specifying the number of items in the submenu.

Example

The example below creates a menu containing “A”, “B”, and “C” with “A” having a submenu “A1” and “A2” and “B” having a submenu “B1”, “B2”, and “B3”.

```
New Window( "title", ob = Outline Box( "title" ) );
ob << Set Menu Script(
   {"A", "", "A1", Print( "A1" ), "A2", Print( "A2" ),
   "B", "", "B1", Print( "B1" ), "B2", Print( "B2" ), "B3", Print( "B3" ),
   "C", Print( "C" )}
);
ob << Set Submenu(1, 2); // menu A with 2 items in submenu A1 and A2
ob << Set Submenu(4, 3); // menu B with 3 items in submenu B1, B2, and B3
```

db<<Set Report Title(*title*)

Sets a new *title*. The *title* is quoted.

Show Properties(*db*)

Shows the messages a given display box can interpret.

db<<Sib

Returns the sibling of the display box.

db<<Sib Append(*db2*)

Appends a display as a sibling to this one. The argument must evaluate to a display box owner or reference.

db<<Size Window(*x*, *y*)

Resizes the containing window.

db<<Update Window

Updates the window that holds the display box (and possibly other windows as well, depending on the operating system) if there are invalidated regions. Previously invalidated box areas are redrawn with their new content.

Notes

In some interactive JSL scripts that combine sliders with JSL callbacks, you might need to use <<Update Window to keep parts of the display synchronized with the slider.

db<<Zoom Window

Resizes the window to be large enough to show all of its contents.

Axis Boxes

Axis Box<<Axis Settings(*<named arguments>*)

Opens the Axis Specification window or specifies axis settings, such as tick marks and axis labels.

If no arguments are included, the axis specification window appears.

Otherwise, specify named arguments for each axis.

- Specify the Y axis as Axis Box(1).
- Specify the X axis as Axis Box(2).

Optional Named Arguments

All Axes

```
Scale("Linear"|"Log"|"Power"|"Geodesic"|"Geodesic US"|"Custom
  Scale"|"Normal Probability|Weibull Probability|Frechet
  Probability|Logistic Probability|Exponential Probability|Gamma
```

Probability|Beta Probability|Mixture of 2 Normals Probabilities|Mixture of 3 Normals Probabilities) Specifies the scale of the axis. If the type is Custom Scale, this message expects two additional named arguments: Scale to Internal(*expr*) and Scale to External(*expr*).

Min(*n*) Changes the minimum value on the axis.

Max(*n*) Changes the maximum value on the axis.

Reverse Order(*Boolean*) Reverses the axes by reversing the minimum and maximum values.

Inc(*n*) Shows the numbers at the specified increments.

Set Font(*font*) Specifies the quoted *font* that is applied to the numbers. The JMP Font preferences determine the default font.

Set Font Size(*points*) Specifies the size of the font that is applied to the numbers. The JMP Font preferences determine the default font.

Set Font Style("Strikeout"|"Underline") Specifies the quoted style that is applied to the numbers.

Automatic Font Size(*Boolean*) JMP attempts to decrease the font size (down to a certain minimum) if all of the labels cannot fit at the default size. If 0, the font size is not decreased.

Automatic Tick Marks(*Boolean*) Turns on tick marks only if one or more labels are hidden (due to insufficient space).

Label Orientation("Automatic"|"Horizontal"|"Vertical"|"Perpendicular"|"Parallel"|"Angled") Rotates the axis label. The default value is "Automatic", which is based on the width of the labels.

Lower Frame(*Boolean*) Shows a frame below the labels. The default value is off.

Value Labels Displays the label that you specify instead of the data value.

Inside Ticks(*Boolean*) Shows tick marks inside or outside of the axis.

Add Ref Line({Label Row Nesting(*n*), *begin range*, <*end range*>, <"Solid"|"Dotted"|"Dashed"|"DashDot"|"DashDotDot">, <*color*>, <*label*>, <width*(n)*>, <*opacity*(%)>}) Defines the reference line range, line pattern, color, label, width, and opacity. A solid, black, 1-pixel line is the default setting. Label Row Nesting(*n*) specifies the number of nested rows on the axis. The *color* and *label* arguments are quoted.

Categorical Axes

Wrap Lines(*n*) Wraps long labels across multiple lines (*n*).

Numeric Axes

Format(*arguments*) Specifies the format of the numeric axis data. See the Format list in a numeric column's column properties for arguments. If you specify the a datetime

format, also include the `Interval` argument: `"Numeric"`, `"Year"`, `"Quarter"`, `"Month"`, `"Week"`, `"Day"`, `"Hour"`, `"Minute"`, or `"Second"`.

`Minor Ticks(`*`number`*`)` Specifies the *number* of minor tick marks between major tick marks.

`Tick Offset(`*`number`*`)` Specifies the starting point of the tick marks.

`Major Ticks(`*`Boolean`*`)` Shows or hides a major tick mark between each number.

`Minor Ticks(`*`Boolean`*`)` Shows or hides a minor tick mark between each number.

`Show Major Grid(`*`Boolean`*`)` Shows or hides a grid line at each major tick mark.

`Show Minor Grid(`*`Boolean`*`)` Shows or hides a grid line at each minor tick mark.

`Major Grid Line Color(`*`color`*`)` Sets the color for the major grid (if enabled) using the quoted *color*.

`Minor Grid Line Color(`*`color`*`)` Specifies the color of the grid line at each minor tick mark.

Example

The following example creates a bivariate plot and defines basic settings for the X and Y axes.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( X( :height ), Y( :weight ), FitLine );
rbiv = biv << Report;
xaxis = rbiv[Axis Box( 2 )];
yaxis = rbiv[Axis Box( 1 )];
xaxis << Axis Settings( Show Major Grid( 1 ) );
yaxis << Axis Settings( Decimal( 10, 3 ) );
```

`Axis Box<<Add Axis Label(`*`quoted string`*`)`

Adds an axis label with the specified quoted string.

`Axis Box<<Add Ref Line(`*`number`*`, `*`linestyle`*`, <`*`color`*`>, <`*`label`*`>, <`*`width`*`>)`

Adds a reference line at *number* in the specified *linestyle* (`"Solid"|"Dashed"|"Double"`), *color* (using the quoted *color*), quoted *label*, and *width* (in pixels).

`Axis Box<<Decimal(`*`width`*`, `*`decimal places`*`)`

Changes the numeric format for axis values.

`Axis Box<<Format(`*`name`*`)`

Changes to the numeric format given by the quoted *name*.

`Axis Box<<Get Inc(`*n*`)`

Gets the increment value of the axis.

`Axis Box<<Inc(`*n*`)`

Sets the increment between ticks.

`Axis Box<<Interval(`*format*`)`

Specifies the units used for `Inc()` with date/time formats: `"Numeric"`, `"Year"`, `"Quarter"`, `"Month"`, `"Week"`, `"Day"`, `"Hour"`, `"Minute"`, or `"Second"`.

`Axis Box<<Label Orientation(`*format*`)`

Rotates the axis label to one of the following formats: `"Automatic"` (the default setting based on the width of the label), `"Horizonal"`, `"Vertical"`, `"Perpendicular"`, `"Parallel"`, and `"Angled"`.

`Axis Box<<Major Grid Line Color(`*color*`)`

Sets the color for the major grid (if enabled) using the quoted *color*.

`Axis Box<<Max(`*maximum*`)`

Changes the maximum value on the axis.

`Axis Box<<Minor Grid Line Color(`*color*`)`

Sets the color for the minor grid (if enabled) using the quoted *color*.

`Axis Box<<Min(`*minimum*`)`

Changes the minimum value on the axis.

`Axis Box<<Minor Ticks(`*number*`)`

Specifies the *number* of minor tick marks between major tick marks.

`Axis Box<<Remove Axis Label`

Removes any label added with `Add Axis Label`.

Axis Box<<Reverse Scale(*Boolean*)

Reverses the normal scale direction so that the highest value is on the left or bottom (that is, closest to the origin).

Axis Box<<Revert Axis

Restores the axis' original settings (from time of creation).

Axis Box<<Scale(*type*)

Changes the scale of the axis to *type* ("Linear"|"Log"|"Exp Prob"|"Weibull Prob"|"Logistic Prob"|"Frechet Prob"|"Normal"|"Cube Root"|"Johnson Su Scale"|"Geodesic"|"Geodesic US"|"Custom Scale"|"Power"|"Gamma Prob"|"Beta Prob"|"Mixture of 2 Normals Prob"|"Mixture of 3 Normals Prob").

If the type is Custom Scale, this message expects two additional named arguments: Scale to Internal(expr) and Scale to External(expr).

Axis Box<<Tick Font(*name*, *<size>*, *<style|style style...>*, *<angle>*)

Sets the font name (quoted), size, and quoted properties for tick marks. To specify more than one style, include a space between each style and place them in quotes.

Axis Box<<Show Labels(*Boolean*)

Shows or hides labels for the axis values.

Axis Box<<Show Major Grid(*Boolean*)

Adds or removes grid lines at the major tick values.

Axis Box<<Show Major Ticks(*Boolean*)

Shows or hides major tick marks.

Axis Box<<Show Minor Grid(*Boolean*)

Adds or removes grid lines at the minor tick values.

Axis Box<<Show Minor Ticks(*Boolean*)

Shows or hides minor tick marks.

Axis Box<<Tick Label List(*<i>*, *{text1,text2, ...}*,*<{n1, n2, ...}>*)

Sets the values and positions of the axis tick labels.

Note: Major tick increments are automatically set to 1.0 if the tick labels are not specified.

Required Arguments

`{`*`text1`*`,`*`text2`*`, ...}` Specifies the quoted string titles for your labels.

Optional Arguments

`i` Specifies the label row index. Leaving it out clears any existing label rows and creates one new one as specified. Including it allows you to override any particular label row; using an index higher than the current number of label rows adds a new label row on to the end.

`{`*`n1`*`, `*`n2`*`, ...}` Specifies the values corresponding to each label. If the value list is omitted, the labels will be on integer increments starting with 1.

Border Boxes

Note: Border boxes support only one display box argument.

`Border Box<<Set Background Color(`*`{r, g, b}|<color>`*`)`

Sets the background color for a border box. Specify an optional quoted list of RGB values or *`color`*. For example:

```
border box<<Set Background Color("red");
```

or

```
border box<<Set Background Color( {255, 192, 3} );
```

`Border Box<<Set Color(`*`{r, g, b}|<color>`*`)`

Sets the border color for a border box. Specify a list of RGB values or a quoted `color`. For example:

```
border box<<Set Color("red");
```

`Border Box<<Get Color`

Gets the border color for a border box.

`Border Box<<Set Style(`*`style`*`)`

Sets the border style for a border box. Specify the *`style`* as one of the following numbers or keywords: `0` (`"Solid"`), `1` (`"Dotted"`), `2` (`"Dashed"`), `3` (`"DashDot"`), or `4` (`"DashDotDot"`). For example:

```
border box<<Set Style("Dotted");
```

Border Box<<Get Style

Gets the border style for a border box.

Data Browser Boxes

dbb<<Set Data Table(*<data table>*)

Sets the data table for the data browser box.

Data Filter Source Boxes

dfsb<<Set Row States(*dt*, *rs*)

Sets the row states for the given data table within the filter. Selections made in this row state will not be linked with the data table, but will be included in the reports linked to the selection filter.

Frame Boxes

Frame Box<<Add Graphics Script(*<order>*,*<description>*, *<script>*)

Description

Adds a script to draw graphics in the frame box.

Optional Arguments

order Specifies the order in which the graphics elements are drawn. The value can be the keyword "Back" or "Forward" or an integer that specifies the drawing order for a number of graphics element. 1 means the object is drawn first.

description A quoted string that appears in the Customize Graph window next to the graphics script. The *description* argument is quoted.

script A JSL script.

Example

In the following example, the graphics script draws the line first and then draws the other graphics elements: the grid lines, references lines, and markers that create the bivariate plot. Without the 1 order argument, the line is drawn last and covers up the markers.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
obj = dt << Bivariate( Y( :weight ), X( :height ) );
Report( obj )[FrameBox( 1 )] <<
Add Graphics Script(
  1, // draws the line first
  Description( "Pen Script" ),
```

```
    Pen Color( "red" );
    Pen Size( 5 );
    Y Function( 60 + 120 / 2 * (1 + Sine( (2 * Pi() * (x - 50)) / 22.5 )), x );
);
```

`Frame Box<<Append Seg`

Adds a display seg to the specified Frame Box.

`Frame Box<<Background Color({`*`RGB values`*`}|<`*`color`*`>)`

Changes the background color. Specify a list of RGB values or a quoted *color*.

`Frame Box<<Child Seg`

Returns the display seg child of the Frame Box.

`Frame Box<<Edit Graphics Script`

Brings up a dialog box to view, edit, or delete the current graphics scripts.

`Frame Box<<Find Seg`

Returns a display seg with the specified argument (for example, the name of a seg).

`Frame Box<<Frame Size(`*`x, y`*`)`

Resets the size of the frame, in pixel units.

`Frame Box<<Make Table of Graphs Like This`

Creates a data table of graphs.

`Frame Box<<Marker Size(`*`size`*`)`

Changes the marker size. The values are 0 (dot), 1 (small), 2 (medium), and so on.

`Frame Box<<Row Colors(`*`color`*`)`

`Frame Box<<Row Markers(`*`marker`*`)`

`Frame Box<<Row Exclude(`*`Boolean`*`)`

`Frame Box<<Row Hide(`*`Boolean`*`)`

`Frame Box<<Row Label(`*`Boolean`*`)`

Forwards commands to the data table associated with the report, so that the row states of selected rows can be manipulated. For `Row Exclude`, `Row Hide`, and `Row Label`, omitting

the argument toggles the option. If the option is off, the message turns it on. If the option is on, the message turns it off.

`frame box<<Set Background Fill(`*`Boolean`*`)`

Enables or disables filling the background with the background color. Use this option when you want to paste a graph and make the background transparent.

Example

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = Bivariate( y( weight ), x( height ) );
rbiv = biv << Report;
framebox = rbiv[Frame Box( 1 )];
// set background color
framebox << Background Color( "red" );
// for demonstration purposes: wait to see the color change
Wait( 1 );
// turn off background fill color
framebox << Set Background Fill( 0 );
```

`frame box<<X Axis(<Min(`*`minimum`*`)>, <Max(`*`maximum`*`)>, <Inc(`*`n`*`)>, <`*`named arguments`*`>)`

Scales the X coordinate system.

`frame box<<Y Axis(<`*`Min(min)`*`>, <`*`Max(max)`*`>, <Inc(`*`n`*`)>, <`*`named arguments`*`>)`

Scales the Y coordinate system.

Display 3D Boxes

`Graph 3D Box()`

Sends display commands to the 3D plot.

Excerpt Boxes

`Excerpt Box(`*`rptnum`*`, `*`lstSubscripts`*`)`

Returns a display box containing the excerpt designated by the report held at number `rptnum` and the list of display subscripts `lstSubscripts`. The subscripts reflect the current state of the report after previous excerpts have been removed.

Filter Col Selector

`Filter Col Selector(<Data Table(`*`name`*`)>, <width(`*`pixels`*`)>, <nLines(`*`n`*`)>, <`*`script`*`>, <onChange(`*`expr`*`)>)`

Returns a display box that contains a list of items. The control supports column filtering.

Global Boxes

`Global Box(`*`value`*`)`

Creates a display box that shows the value of a global variable.

Hier Boxes

`Hier Box(`*`title`*`, Hier Box(...), Hier Box(...), ...)`

Returns a display box with the *`title`* (quoted) that contains a hierarchy of quoted strings.

Matrix Boxes

`Matrix Box<<Get`

Returns the matrix contents.

`Matrix Box<<Make Into Data Table(<Invisible(`*`Boolean`*`)|Private(`*`Boolean`*`)>)`

Description

Turns the matrix into a new data table. `Invisible(1)` hides the data table from view. An invisible data table can be open from the JMP Home Window or the Window menu. `Private(1)` opens the data table without displaying it in a data table window. A private data table is generally for scripts that want better control of the data table by not exposing it to general use.

Returns

A reference to the new data table.

Matrix Box<<Set Format(*<width>*, *<decimal places>*, <"Use Thousands Separator">)

Description

Sets the numeric format for matrix elements.

Arguments

A number of other formats can be set on matrix boxes. See "Number Col Box<<Set Format(<width>|<width, decimal places>, <"Use Thousands Separator">)" for more information about the syntax.

Matrix Box<<Sort(*column number*, *ascending*)

Sorts the rows of the matrix based on the column number specified by column_num. The default sort order is ascending.

If column number is 0, the sort is removed.

ascending is a Boolean value. If ascending is "True", the sort is performed in ascending order. If ascending is "False", the sort is in descending order.

Nom Axis Boxes

Nom Axis Box<<Divider Lines(*Boolean*)

Adds or removes divider lines between labels in the axis box.

Nom Axis Box<<Lower Frame(*Boolean*)

Adds or removes a lower frame around the axis.

Nom Axis Box<<Rotated Tick Labels(*Boolean*)

Rotates or unrotates the labels at each tick value.

Number Col Boxes

Number Col Box<<Add Element(*item*)

Adds the *item* to the Number Col Box. *item* can be a single number, a list of numbers, or a matrix.

`Number Col Box<<Bootstrap(`*`nsample`*`, Random Seed(`*`number`*`), Fractional Weights(`*`Boolean`*`), Split Selected Column(`*`Boolean`*`), Discard Stacked Table if Split Works(`*`Boolean`*`)`

Description

Bootstraps the analysis, repeating it many times with different resampling weights and collecting tables as selected.

Arguments

`nsample` Sets the number of times that you want to resample the data and compute the statistics. A larger number results in more precise estimates of the statistics' properties. By default, the number of bootstrap samples is set to 2,500.

`Random Seed(`*`number`*`)` Sets a random seed that you can re-enter in subsequent runs of the bootstrap analysis to duplicate your current results. By default, no seed is set.

`Fractional Weights(`*`Boolean`*`)` Performs a Bayesian bootstrap analysis. In each bootstrap iteration, each observation is assigned a weight that is calculated as described in *Basic Analysis*. The weighted observations are used in computing the statistics of interest. By default, the fractional weights option is not selected and a simple bootstrap analysis is conducted.

`Split Selected Column(`*`Boolean`*`)` Places bootstrap results for each statistic in the column that you selected for bootstrapping into a separate column in the Bootstrap Results table. Each row of the Bootstrap Results table (other than the first) corresponds to a single bootstrap sample.

If you exclude this option, a Stacked Bootstrap Results table appears. For each bootstrap iteration, this table contains results for the entire report table that contains the column that you selected for bootstrapping. Results for each row of the report table appear as rows in the Stacked Bootstrap Results table. Each column in the report table defines a column in the Stacked Bootstrap Results table.

`Discard Stacked Table if Split Works(`*`Boolean`*`)` (Applicable only if the Split Selected Column option is included.) Determines the number of results tables produced by Bootstrap. If the `Discard Stacked Table if Split Works` option is not selected, then two Bootstrap tables are shown. The Stacked Bootstrap Results table, which contains bootstrap results for each row of the table containing the column that you selected for bootstrapping, gives bootstrap results for every statistic in the report, where each column is defined by a statistic. The unstacked Bootstrap Results table, which is obtained by splitting the stacked table, provides results only for the column that is selected in the original report.

`Number Col Box<<Get`

`Number Col Box<<Get(`*`i`*`)`

Gets the values in a list, or the *i*th value.

Number Col Box<<Get As Matrix

Gets the values in a matrix, specifically a column vector.

Number Col Box<<Get Format

Returns the current format.

Number Col Box<<Get Heading

Returns the column heading text.

Number Col Box<<Remove Element(*row number*)

Removes an element from the column at the specified position.

Number Col Box<<Set Format(*<width>*|*<width, decimal places>*, *<*"Use Thousands Separator">)

Number Col Box<<Set Format("Best", *<width>*, <"Use Thousands Separator">)

Number Col Box<<Set Format(("Fixed Dec"|"Percent"), *<width>*|*<width, decimal places>*, <"Use Thousands Separator">)

Number Col Box<<Set Format("Pvalue", *<width>*)

Number Col Box<<Set Format(("Scientific"|"Engineering"|"Engineering SI"), *<width>*|*<width, decimal places>*)

Number Col Box<<Set Format("Precision", *<width>*|*<width, decimal places>*, <"Use Thousands Separator">, <"Keep Trailing Zeroes">, <"Keep All Whole Digits">)

Number Col Box<<Set Format("Currency", *<currency code>*, *<width>*|*<width, decimal places>*, <"Use Thousands Separator">)

Number Col Box<<Set Format(*datetime*, *<width>*, *<input format>*)

Number Col Box<<Set Format(("Latitude DDD"|"Latitude DDM"|"Latitude DMS"|"Longitude DDD"|"Longitude DDM"|"Longitude DDM"), *<width>*|*<width, decimal places>*, ("PUN"|"DIR"|"PUNDIR"))

Number Col Box<<Set Format("Custom", Formula(...), *<width>*, *<input format>*)

Description

Sets the column format.

Arguments

Using JMP describes the arguments. Note that `Matrix Box()`, `Number Col Box()`, `Number Col Edit Box()`, `Number Edit Box()` have the same `Set Format` syntax.

Examples

```
<<Set Format( 10, 2, "Use thousands separator");
<<Set Format( "Currency", "EUR", 20, );
<<Set Format( "m/d/y", 10 );
<<Set Format( "Precision", 10, 2, "Keep trailing zeroes", "Keep all whole
  digits" );
<<Set Format( "Latitude DDD", "PUNDIR"); // "PUN" for punctuation, "DIR" for
  direction, PUNDIR for both
<<Set Format( "Custom", Formula( Abs( value ) ), 15 );
```

Notes

- For a list of currency codes, see *Scripting Guide*. The currency code is based on the locale if the code is omitted.
- If you don't specify the format, set the decimal places to greater than 100 for datetime values and to 97 for *p*-values.
- You must always precede the number of decimal places with the width.
- Options can be defined in a list or a variable, or they can be in a `Function()` that is evaluated.

```
ncbFunc = Function({}, {"Fixed", 12, 5});
```

`number col box<<Set Heading(`*`quoted string`*`)`

Changes the column heading text.

Number Col Edit Boxes

`Number Col Edit Box<<Set Format(`*`<width>`*`, `*`<decimal places>`*`, <"Use Thousands Separator">|`*`<other options>`*`)`

Description

Sets the column format.

Arguments

A number of other formats can be set on number col edit boxes.

See Also

"Number Col Box<<Set Format(<width>|<width, decimal places>, <"Use Thousands Separator">)"

Number Col Edit Box<<Remove Element(*x position*, *y position*, *i*)

Removes an element from the column at the specified position.

Number Edit Box

Number Edit Box<<Set Format(*<width>*, *<decimal places>*, <"Use Thousands Separator">|*<other options>*)

Description

Sets the column format.

Arguments

A number of other formats can be set on number edit boxes.

See Also

"Number Col Box<<Set Format(<width>|<width, decimal places>, <"Use Thousands Separator">)"

Outline Boxes

Outline Box<<Close(*Boolean*)

Closes the outline box.

Outline Box<<Close All Below

Closes all the node's child nodes.

Outline Box<<Close All Like This

Closes all nodes similar to this outline box.

Outline Box<<Close Where No Outlines

Closes all nodes that do not have children.

Outline Box<<Get Title

Gets the title of the outline box.

Outline Box<<Horizontal(*Boolean*)

Horizontally aligns the node's children.

`Outline Box<<Open All Below`

Opens all the node's child nodes.

`Outline Box<<Open All Like This`

Opens all nodes similar to this outline box.

`Outline Box<<Set Menu Script(`*{quoted string1, script1, quoted string2, script2, ...}*`)`

Adds an entry to the menu when the red triangle on an outline box is selected.

`Outline Box<<Set Title(`*title*`)`

Specifies the quoted *title* of the outline box.

Panel Boxes

`Panel Box<<Get Title`

Gets the title of the panel box.

`Panel Box<<Set Title(`*title*`)`

Specifies the *title* (quoted) of the panel box.

Plot Col Boxes

`Plot Col Box<<Get As Matrix`

Gets the values in a matrix, specifically a column vector.

`Plot Col Box<<Get Labels`

Gets the labels for each row.

`Plot Col Box<<Remove Element(`*row number*`)`

Removes an element from the column at the specified position.

`Plot Col Box<<Set Labels(`*{list}*`)`

Sets the labels for each row.

`Plot Col Box<<Set Scale(`*minimum*`, `*maximum*`, <"format">, <`*width*`>, <`*decimal places*`>, <"Use Thousands Separator">)`

Description

Specifies the minimum and maximum values of the horizontal axis.

Arguments

The format arguments specify a format that is used when the first two arguments are 0 and 1. A number of formats can be set on plot col boxes. See "Number Col Box<<Set Format(<width>|<width, decimal places>, <"Use Thousands Separator">)" for more information about the syntax.

`Plot Col Box<<Set Values(`*[matrix]* or *{list}*`)`

Sets values for the matrix (for numeric variables) or list (for character variables).

Slider Boxes and Range Slider Boxes

`Slider Box<<Get(`*<index>*`)`

`Range Slider Box<<Get Lower(`*<index>*`)`

`Range Slider Box<<Get Upper(`*<index>*`)`

Returns the current value of the slider.

`Slider Box<<Get Max()`

Returns the maximum value possible for the range slider and slider.

`Slider Box<<Get Min()`

Returns the minimum value possible for the range slider and slider.

`Slider Box<<Get Var`

`Range Slider Box<<Get Lower Var`

`Range Slider Box<<Get Upper Var`

Returns the variable name associated with the slider.

Slider Box<<Set(*float*, <*index*>, <Run Script(*Boolean)*>)

Range Slider Box<<Set Lower(*float*, <*index*>, <Run Script(*Boolean*)>)

Range Slider Box<<Set Upper(*float*, <*index*>, <Run Script(*Boolean*)>)

Sets the value of the slider. Run Script(*Boolean*) controls whether an on-change script runs after the Set, Set Lower, or Set Upper message.

Slider Box<<Set Max(*float*, <*index*>)

Sets the maximum value possible for the range slider and slider.

Slider Box<<Set Min(*float*, <*index*>)

Sets the minimum value possible for the range slider and slider.

Slider Box<<Set Script(<*script*>)

Sets a script to be run when the range sliders and slider is updated.

Slider Box<<Set Var(*slider variable*)

Range Slider Box<<Set Lower Var(*slider variable*)

Range Slider Box<<Set Upper Var(*slider variable*)

Sets the variable name associated with the slider.

String Col Boxes

String Col Box<<Add Element(*item*)

Adds the item to the quoted string Col Box. Item can be a single quoted string or a list of quoted strings.

String Col Box<<Get quoted

String Col Box<<Get(*i*)

Gets the values in a list or the *i*th value.

String Col Box<<Get Heading

Returns the column heading text.

String Col Box<<Remove Element(*row number*)

Removes an element from the column at the specified position.

String Col Box<<Set Allow Text Search(*Boolean*)

Description

In table boxes with selectable rows, allows a quoted string column that has focus to respond to keyboard input to change the selected row.

Example

```
// Run the example.
// Select K2.
// Type the letter g. Notice the last row is selected.
// Type the letters ki. Notice the third row is selected.
New Window( "Mountains",
   tb = Table Box(
      sb =
      String Col Box( "Mountain",
         {"K2", "Delphi", "Kilimanjaro",
         "Grand Teton"}
      ),
      Number Col Box( "Elevation (meters)",
         {8611, 681, 5895, 4199}
      ),
      Plot Col Box( "", {8611, 681, 5895, 4199} )
   )
);
tb << Set Selectable Rows( 1 );
sb << Set Allow Text Search( 1 );
```

String Col Box<<Set Heading(*title*)

Changes the column heading specified in the quoted *title*.

String Col Box<<Set Justify(*justification*)

Specifies the alignment of the contents in the quoted string col box to "Right", "Left", or "Center".

Tab Boxes

Tab Box<<Get Tab Margin()

Returns a list of the current margins in pixels for the tab box in this order: Left, Top, Right, and Bottom.

`Tab Box<<Set Style("Tab"|"Combo" |"Outline"|"Vertical Spread"|"Horizontal Spread"|"Minimize Size")`

Changes the appearance of the tab box from a tab to a combo box or outline node.

`"Vertical Spread"` and `"Horizontal Spread"` change the orientation of the tab title.

`"Minimize Size"` bases the tab style on the width of the tab title.

`Tab Box<<Set Tab Margin(`*n*`|{...})`

Sets the tab margin for the tab box. If a single number is specified, all four margins are set to that number of pixels. If a list of two numbers is specified, the left and right margins are set to the first number, and the top and bottom margins are set to the second number. If a list of four numbers is specified, the margins are set in this order: `{left, top, right, bottom}`.

`Tab Box<<Show Tabs(`*Boolean*`)`

Shows or hides the tabs for tab boxes. If you hide the tabs, you need to provide another way to select and show tabs. For example, a list box that contains a list of references to the tabs. The default value is 1.

Table Boxes

`Table Box<<Bootstrap(`*nsample*`, Random Seed(`*number*`), Fractional Weights(`*Boolean*`), Split Selected Column(`*Boolean*`), Discard Stacked Table if Split Works(`*Boolean*`)`

Description

Bootstraps the analysis, repeating it many times with different resampling weights and collecting tables as selected.

See Also

"Number Col Boxes"

`Table Box<<Get`

Gets the entries of the table in list form.

`Table Box<<Get As Matrix(<"Visible">)`

Gets the numeric entries of the table in matrix form. `"Visible"` means that only visible columns will be included.

Table Box<<Get Click Sort

Returns 1 if the table can be sorted by clicking a column header and 0 otherwise.

Table Box<<Get Locked Columns

Returns the number of columns that cannot be dragged with the cursor or have any columns dropped before them.

Table Box<<Get Row Change Function

Returns the expression that is evaluated when a row is selected.

Table Box<<Get Selectable Rows

Returns True if the table box currently allows row selection.

Table Box<<Get Selected Row Color

Returns the index number of the background color of the selected rows in the table box.

Table Box<<Make Combined Data Table

Returns a reference to the data table. Same as `Make Data Table`, but also searches the report for report tables with the same columns and combines all of these into the new data table.

Table Box<<Make Data Table(*name*)

Returns a reference to the data table. Turns the table entries into a new data table with the quoted *name* argument.

Table Box<<Reorder Columns(*from column index*, *to column index*)

Puts the column specified with *from column index* in the place of the column specified with *to column index*. The indexes are 0-based. For example, indicate the first column with "0", and indicate the second column with "1".

Table Box<<Set Cell Changed Function(Function({*this*, *col box*, *row*},<*script*>))

Description

Sets a function that is called whenever the user edits a cell in a column in a table.

Example

This example prints the new values for the changed cell to the log.

```
New Window( "Mountains",
   tb = Table Box(
      quoted string Col Edit Box(
         "Mountain",
         {"K2", "Delphi", "Kilimanjaro",
         "Grand Teton"}
      ),
      Number Col Edit Box(
         "Elevation (meters)",
         {8611, 681, 5895, 4199}
      ),
      Plot Col Box( "", {8611, 681, 5895, 4199} )
   )
);
tb <<
Set Cell Changed Function(
   Function( {this, col, row},
      Print(
         (col << Get Heading) || ": row:" ||
         Char( 3 ) || " is now " ||
         Char( col << Get( row ) )
      )
   )
);
```

Table Box<<Set Click Sort(*Boolean*)

Specifies if the table can be sorted by clicking a column header.

Table Box<<Set Column Borders(*Boolean*)

Draws a line on each side of the column.

Table Box<<Set Heading Column Borders(*Boolean*)

Draws a line on each side of the column headings.

Table Box<<Set Locked Columns(*n*)

Locks the first *n* columns. You cannot drag the locked columns or drag columns before them.

Table Box<<Set Row Borders(*Boolean*)

Draws a line above and below each row.

Table Box<<Set Row Change Function(*function*)

Sets the expression that is evaluated when a row is selected.

Table Box<<Set Selectable Rows(*Boolean*)

Makes the rows of the table box selectable or not.

Table Box<<Set Selected Row Color(*color*)

If the rows of the table box are selectable (`Set Selectable Rows(True)`), sets the background color (specified in the quoted *color* argument) for the selected rows.

Table Box<<Set Shade Cells(*Boolean*)

Shades the background of every cell in the table.

Table Box<<Set Shade Alternate Rows(*Boolean*)

Shades the background of every other row in the table.

Table Box<<Set Shade Heading(*Boolean*)

Shades the background in column headings.

Table Box<<Set Underline Headings(*Boolean*)

Draws a line underneath the column headings.

Table Box<<Sort By Column(<*column number*|*column title*>, <Ascending(*Boolean*)>

Sorts all rows based on the values in the specific column number or quoted column title. The default order sorting is descending.

Text Boxes

Text Box<<Font Color(*n*)

Sets the color for `Text` quoted strings.

Text Box<<Get Hidden State

Returns the current state of a text box.

Text Box<<Get Text

Returns the quoted string content of the box.

Text Box<<Get Tip

Returns the tooltip for the text box (or a text edit box).

Text Box<<Markup

Returns text formatted with the specified HTML tags. The HTML must be well-formed; make sure you close nested tags correctly.

The following example returns text formatted in bold, italic, and underlined.

```
w = New Window( "Formatted Text",
   Text Box( "This is <b>bold</b> text. This is <b><i>bold italic</i></b>
   text. This is <u>underlined</u> text.",
   <<Markup) );
```

Text Box<<Rotate Text(*direction*)

Rotates the text 90 degrees "Left" or "Right", or returns it to horizontal.

Text Box<<Set Font(*name*, *<size>*, *<style|style style...>*, *<angle>*)

Sets the font specified in the quoted *name* argument and the properties for text quoted strings. To specify more than one style, include a space between each *style* and place them in quotes.

Text Box<<Set Font Size(*n*)

Sets the font size in points for text quoted strings.

Text Box<<Set Script(*script*)

Associate a script with a text box. The script executes when the user presses Enter (or the text edit box otherwise loses focus).

Text Box<<Set Text(*quoted string*)

Sets the text in the box as specified in the quoted *string* argument.

Text Box<<Set Tip(*quoted string*)

Sets the tooltip for the text box (or a text edit box) as specified in the quoted *string* argument.

`Text Box<<Set Wrap(n)`

Set the wrap point, in pixels, in pixels (*n*).

Tree Node and Tree Box

For the following messages, `node` stands for a tree node or a reference to one and `root` stands for a tree box or a reference to one.

Caution: If you send a root node that contains one or more nodes with the `Set Node Select Script` defining a collapse message, then macOS runs the script twice. Windows doesn't run the script. This behavior on macOS doesn't just affect increments. Any script runs twice. It will print to the log twice, create a column twice, try to delete something twice, and so on.

`node<<Append(<node>)`

Inserts a referenced tree node after this node's children.

`node<<Collapse`

Closes the node. The behavior is not guaranteed if the node has a collapsed parent.

`node<<Expand`

Opens the node. The behavior is not guaranteed if the node has a collapsed parent.

`node<<Get Dimmed(<node>)`

Gets the option to dim text (decrease the opacity) for the node.

`node<<Get Font Style(<node>)`

Gets the font style for the node.

`node<<Get Tip`

Returns the tooltip for the node.

`node<<Prepend(<node>)`

Inserts a tree node before this node's children.

`node<<Remove`

Removes the given tree node and all its children from the tree display box.

`node<<Set Dimmed(Boolean)`

Sets the option to dim text (decrease the opacity)) for the node.

`node<<Set Font Style("Plain"|"Bold")`

Specifies the font style for the node.

`node<<Set Selected(<node>)`

Selects the node. The behavior is not guaranteed if the node has a collapsed parent.

`node<<Set Tip(tooltip)`

Sets a tooltip for the node. The *tooltip* argument is quoted.

`root<<Collapse(<node>)`

Collapses the given tree node.

`root<<Expand(<node>)`

Expands the given tree node.

`root<<Get Selected(<node>)`

Gets the currently selected tree node.

- In a single-item tree, the currently selected tree node or `Empty` is returned.
- Table 3.1 shows the results for a `Tree Box()` that contains the `MultiSelect` argument.

Table 3.1 Multi-Select Tree Results

Items Selected in Tree	Result
no items selected	empty
single item selected	list of one tree node
multiple items selected	list of selected tree nodes

`root<<Is Multiselect`

Returns 1 for a `MultiSelect` tree and 0 for a single-select tree.

`root<<Set Selected(node|{nodes}, <Boolean>)`

Selects the given tree node in the tree display box. In list of tree nodes, all nodes in the list are selected for `MultiSelect` trees. Otherwise, the first node in the list is selected. Specify the Boolean argument to indicate whether the node or nodes should be selected or unselected. The default value is 1, which selects the nodes.

Notes:

- On Windows, the `Set Selected` message expands all nodes between the selected node or nodes and the root of the tree; items that are selected deep within the tree are shown. The expansion state does not change for nodes that were previously selected.
- On macOS, the `Set Selected` message does not change the tree expansion state.

Triangulation

For the following messages, `tri` stands for a triangulation or a reference to one.

`tri<<Get N Points`

Returns the number of unique points in the triangulation.

`tri<<Get Points`

Returns the coordinates of the unique points in the triangulation.

`tri<<Get Y`

Returns the Y values of the unique points in the triangulation.

`tri<<Get N Hull Points`

Returns the number of points on the boundary of the triangulation.

`tri<<Get Hull Points`

Returns the indices of the points on the boundary of the triangulation.

`tri<<Get N Hull Edges`

Returns the number of edges on the boundary of the triangulation.

`tri<<Get Hull Edges`

Returns the indices of the edges on the boundary of the triangulation.

`tri<<Get Hull Path`

Returns the boundary of the triangulation as a path.

`tri<<Get N Triangles`

Returns the number of triangles.

`tri<<Get Triangles`

Returns the indices of the triangles in the form of an Nx3 matrix.

`tri<<Get N Edges`

Returns the number of edges in the triangulation.

`tri<<Get Edges`

Returns the indices of the edges in the form of an *N*x2 matrix.

`tri<<Subset(`*{indices}*`)`

Returns a triangulation resulting from the given subset of points.

`tri<<Peel`

Peel the boundary layer of a triangulation, returning a new triangulation.

Windows

`window<<Bring Window to Front`

Brings the window to the front.

`window<<Close Window(`*<nosave>*`)`

Closes the window. If the optional argument nosave is specified, the window (journal, report, and so forth) is closed without saving or prompting.

`window<<Get Content Size`

Returns the size of the window's contents.

`window<<Get Window Icon`

Returns the name of the window's icon.

`window<<Get Window Position`

Returns the position of the window.

`window<<Get Window Size`

Returns the size of the window.

`window<<Get Window Title`

Returns the title of the window.

`window<<Inval`

Invalidate the display box. The window updates either when the `<<Update Window` message is sent or when the operating system has time for the update. See "window<<Reshow" for another method.

`window<<Maximize Display`

Maximizes the window. Deprecated.

`window<<Maximize Window(`*Boolean*`)`

Maximizes the window. Deprecated.

`window<<Minimize Window(`*Boolean*`)`

Minimizes the window.

`window<<Move Window(`*x*`, `*y*`)`

Moves the window to the specified position.

`window<<On Close(`*script*`)`

Runs the script when the window is closed.

`window<<Pad Window(`*Boolean*`)`

Turns padding around a window's contents on (`1`) or off (`0`). The default value is off.

`window<<Print Window`

Prints the window to the default printer. Note that the Print window is not opened and user input is not required.

`window<<Reshow`

Invalidates the display box and updates the window with the new content. See `<<Inval` and `<<Update Window` messages if more control over timing of the update is required.

`window<<Save Window to Report(`*`pathname`*`, <Embed Data(`*`Boolean`*`)>)`

Saves the current report window to a JMP report file (.jrp).

`window<<Set Main Window`

Sets the specified window as the default window that appears when JMP is run.

`window<<Set Window Icon(`*`icon name`*`)`

Sets the window's icon as specified in the quoted *`icon name`* argument.

`window<<Set Window Size(`*`x`*`, `*`y`*`)`

Resizes the window.

`window<<Show Window(`*`Boolean`*`)`

1 shows the window (only if the window is not currently open). 0 hides the window. If the window is also minimized (on Windows) or docked (on macOS), showing the window restores it to the normal state and brings it to the front.

`window<<Size Window(`*`x`*`, `*`y`*`)`

Resizes the window.

`window<<Update Window`

Updates or refreshes the window holding the display box if there are invalidated regions. See "window<<Inval" and "window<<Reshow" for additional methods.

`window<<Window Class Name`

Returns the name of the window class for the display box. Valid responses include: DataTable, FormulaEditor, Starter, Journal, Launcher, Report, Dialog, DialogWithMenu, ModalDialog, FindReplace, User, Generic, ToolWindow, FindReplace, AppBuilder, and Debugger.

`window<<Zoom Window`

Resizes the window to be large enough to show all of its contents.

Dynamic Link Libraries (DLLs)

`dll object<<Call DLL(`*`function name`*`, `*`signature`*`, `*`arguments`*`)`

Calls the specified function in the DLL with the specified signature and arguments.

`dll object<<Declare Function(`*`name`*`, `*`<named arguments>`*`)`

Description

Declares the return type and argument types for the specified function so that it can be successfully invoked. You can use one of the named arguments for `Convention`: `"STDCALL"` or `"PASCAL"`, or `"CDECL"`. The *`type`* argument for `Returns` takes the same named arguments as `Arg`. The *`name`* argument is quoted.

Optional Named Arguments

`Alias(`*`name`*`)` Specifies a quoted *`name`* that you can include if you don't like the name encoded in the DLL.

`Arg(`*`type`*`, `*`<description>`*`, `*`<access mode>`*`, `*`<array>`*`)` `Arg` can appear multiple times, once for each argument to be sent to the function.

`type` is one of these keywords that specifies the argument type: `"Int8"`, `"UInt8"`, `"Int16"`, `"UInt16"`, `"Int32"`, `"UInt32"`, `"Int64"`, `"UInt64"`, `"Float"`, `"Double"`, `"AnsiString"`, `"UnicodeString"`, `"Struct"`, `"IntPtr"`, `"UIntPtr"`, or `"ObjPtr"`.

`description` is a quoted string that describes the argument for reference.

`access mode` is an optional keyword that specifies how the argument is passed. `"input"` specifies that the argument is passed by value. `"output"` specifies that the argument is passed by address with the initial value undefined. `"update"` specifies that the argument is passed by reference and the value of the JSL variable is set as the initial value. The default value is `"input"`.

`array` is an optional keyword. It is valid only if the *`type`* is specified as `"Double"` and the `access mode` is specified as either `"input"` or `"update"`. Specifies that the exported function expects an array of doubles.

`Convention(`*`calling convention`*`)` Specifies the calling convention: `"STDCALL"` or `"PASCAL"`, or `"CDECL"`. The default value is `"STDCALL"`. `STDCALL` and `PASCAL` are equivalent.

`MaxArgs(`*`n`*`)` Specifies the maximum number of arguments that can be supplied.

`MinArgs(`*`n`*`)` Specifies the minimum number of arguments that can be supplied.

`Returns(`*`type`*`)` Specifies the data type that the function returns: `"Int8"`, `"UInt8"`, `"Int16"`, `"UInt16"`, `"Int32"`, `"UInt32"`, `"Int64"`, `"UInt64"`, `"Float"`, `"Double"`, `"AnsiString"`, `"UnicodeString"`, `"Struct"`, `"IntPtr"`, `"UIntPtr"`, or `"ObjPtr"`.

`StackOrder(`*`order`*`)` Specifies the order in which arguments are placed on the stack when calling the function. Valid values are `"L2R"` (left-to-right) and `"R2L"` (right-to-left). The default value is `"R2L"`.

`StackPop(`*`pop`*`)` Specifies how the exported function expects the stack to be cleared after the function returns. Valid values are `"CALLER"` and `"CALLEE"`. The default value is `"CALLEE"`.

`StructArg(`*`Arg(...)`*`, `*`<Arg(...)>`*`, ..., `*`<access mode>`*`, `*`<pack mode>`*`, `*`<description>`*`)` Can appear multiple times. If an exported DLL function requires that a structure argument be passed in as an argument, use `StructArg` to declare the structure members. The `Arg` arguments use the same syntax as for `Arg` arguments to `Declare Function` (one for each structure member), an `access mode` indicator and a `pack mode` indicator.

`access mode` is an optional keyword that indicates whether the struct argument should be passed by value (`input`) or by reference (`update`).

`pack mode` is an optional integer that determines how the structure is packed. Valid values are 1, 2, 4, 8, and 16. The default value is 8.

`description` is an optional, quoted string that contains a description of the structure for reference.

`dll object<<Get Declaration JSL`

Sends the declaration JSL from the DLL object to log.

`dll object<<Load DLL(`*`path`*`, <AutoDeclare(`*`Boolean`*`|"Quiet"|"Verbose")>)`

`dll object<<Load DLL(`*`path`*`, <"Quiet"|"Verbose">)`

Description

Loads the DLL from the specified path.

Required Argument

`path` A quoted path that specifies where to load the DLL.

Optional Named Arguments

`AutoDeclare(`*`Boolean`*`|"Quiet"|"Verbose")` `AutoDeclare(1)` and `AutoDeclare("Verbose")` write verbose messages to the log. `AutoDeclare("Quiet")` turns off log window messages. If you omit this option, verbose messages are written to the log.

`Quiet|Verbose` When you use `Declare Function`, this option turns off log window messaging (`"Quiet"`) or turns on log window messaging (`"Verbose"`).

`dll object<<Show Functions`

Sends the declared functions for the DLL object to the log.

dll object<<Unload DLL

Unloads the DLL.

HTML 5

Web Report

webreport<<Publish(<Add Image(...)>, <Add Report(...)>, <Add Reports(...)>, <Public(*Boolean*)>, <Index(...), <User Name(...)>, <Password(...)|*password function)*>, <Prompt("IfNeeded"|"Always"|"Never")>, <URL(...)>, <Publish Data(*Boolean*)>, Replace(<*id*>, Prompt("IfNeeded"|"Always"|"Never"))>)

Description

Publishes the web report to the JMP server.

Returns

On success, the URL of the published report is returned.

Optional Arguments

Add Image Inserts an image at the top of the index page. Valid formats are "png", "bmp", "jpeg", "jpg", "tiff", and "tif". Title and Description are optional. Title appears above the image. Description appears below the image. Use File(*filepath*) or just a quoted string. Here is an example:

```
webrpt << Add Image( File( "C:\Users\Public\JMP\Projects\WebJMP\atlas.jpg" ),
Title( "Atlas" ), Description( "Holding up the world as always." ) );
```

Add Report Adds a report to publish within the web report.

Public(*Boolean*) Specifies whether the public has access to the report. By default, the report is private.

Index The name of the index page for multiple reports. You can also specify the description.

User Name Specifies the user name registered on the JMP server.

Password Specifies the user's password. You can also define a password function.

Prompt Displays a window in which the user types the server URL, user name, and password.

URL The location that you are publishing to.

Publish Data(*Boolean*) Includes the data in the HTML. Reports contains static rather than interactive images. In a public report, you might not want to share the data.

Replace Replaces the report. Get the URL from the address field in the browser where the page is displayed.

Images

The Scripting Index provides examples for processing images. In JMP, select **Help > Scripting Index** to view this interactive resource.

Additional resources are available from the JMP File Exchange at https://community.jmp.com/community/file-exchange.

`img<<Crop(Left(`*pix*`), Right(`*pix*`), Top(`*pix*`), Bottom(`*pix*`))`

Creates a new image from an existing image to the specified dimensions (in pixels).

`img<<Filter(`*name*`, <`*n*`>)`

Filters the image based on the specified algorithm. Filtering is useful for cleaning up noise in the image.

Note: All of the JMP image filters are supported at the operating system level. Images that are processed on Windows might differ from images processed on macOS.

Argument

`name` Specifies the quoted name of a JMP image filter. The following filters are available:

- `"Despeckle"` removes defects (that is, speckles) from a scanned or captured image (for example, scratches, dust, etc.).
- `"Edge"` identifies pixels in an image where the brightness changes sharply and darkens pixels with no sharp change. Edge detection is used to detect changes in surface, depth, material, and lighting.
- `"Enhance"` reduces the contrast between pixels in a noisy image.
- `"Median"` reduces noise (that is, the random variation) and smooths an image by comparing each pixel's brightness with its neighbors' and, if the value is very different, replaces it with the average of the neighbors' values.
- `"Negate"` creates the negative of the color or gray-scale image by changing each pixel color to its complementary color.
- `"Normalize"` changes a color image's pixels to use the full range of the file format's number system. Normalization will make the image's colors more intense.
- `"Sharpen"` reduces blur by sharpening edges of an image.
- `"Contrast"`, *n* brightens or darkens an image. A higher number (>0.0) brightens an image; a lower number (<0.0) darkens an image.

- "Gamma", *n* corrects the image visual display (brightness and intensity) to account for differences in monitor hardware. A higher number (> 1.0) lightens the image; a lower number (< 1.0) darkens the image.
- "Reduce Noise", *n* reduces the random variation (or noise) that occurs with higher ISO sensitivity or longer exposure times.
- "Gaussian Blur", radius, sigma reduces image noise and detail creating a smoother image. Radius is equal to the blur radius around each pixel and sigma is the standard deviation of the Gaussian distribution. Gaussian blur is commonly used when resizing or performing edge detection.

`img<<Flip Both`

Flips the image from left to right and top to bottom.

`img<<Flip Horizontal`

Flips the image from left to right.

`img<<Flip Vertical`

Flips the image from top to bottom.

`img<<Get EXIF`

Returns EXIF data from the image (such as the shutter speed and aperture value) in an associative array.

`img<<Get N Frames`

Description

Returns the number of frames in a multi-frame TIF or animated GIF file, where the number of frames begins with frame 0.

Example

The following example places a four-frame TIF file in a new window and shows the image that is in the first frame.

```
img = New Image( "$DOWNLOADS/Multiframe.tif" );
nframes = img << Get N Frames(); // return 4
img << Set Current Frame( 1 ); // show image 1
win = New Window( "Multi-Frame TIFF", img );
```

img<<Get Size

img<<Size

Returns a list containing the width and height (in pixels) of the image.

img<<Rotate(*degrees*)

Rotates the image by the specified number of degrees.

img<<Save Image(*path*)

Saves the image to the quoted *path*.

img<<Scale(*scale*|*xscale*, *yscale*)

Resizes the image by the specified dimensions. Provide one argument to resize both the width and height. Provide two arguments to resize the width and height separately.

Examples

```
img = New Image( "$SAMPLE_IMAGES/tile.jpg" );
xs = 2;
img << Scale( xs );
New Window( "Tilex 2", img );

img = New Image( "$SAMPLE_IMAGES/tile.jpg" );
img << Scale( 2, 0.5 ); // scale image width by 2 and height by 1/2
New Window( "Tile squished", img );
```

Notes

Using Scale is an alternative to getting the size of the image, multiplying by the scale factor, and then setting the size.

img<<Set Current Frame

Description

Sets the frame that shows in a multi-frame TIFF or animated GIF file. Specify 0 through the number of frames minus 1. For example, with four frames, you can specify frame 0 through frame 3.

See Also

"img<<Get N Frames"

img<<Set Size(*width*, *height*)

Resizes the image to the specified dimensions (in pixels). To scale the image proportionally, specify a width and height that correspond to the aspect ratio in the original image.

`img<<Transparency(`*fraction*`)`

Sets the transparency for the image where the fraction is between 0.0 (full transparency) to 1.0 (no transparency).

JMP Applications

The JMP Application Builder and JMP Dashboard Builder use the same infrastructure to design and execute applications and dashboards. Because a dashboard is a special form of an application, this section uses the term *application* to describe how both dashboard and application objects use scripting.

See the Scripting Index in the JMP Help menu for examples.

JMP App

The `JMP App` object is the main controller for JMP applications built by Application Builder or Dashboard Builder. Scripts both inside and outside of a JMP application can use a `JMP App` object.

A JMP application can have one of three states: initial (with no editor, and the application is not running), running, or editing. A `JMP App` object only exists in one state at a time; if you are editing a JMP Application and choose to run it, a copy of the JMP application is created before it is run.

`app<<Combine Windows({`*reports or data tables*`})`

Description

Combines the given list of platform reports or data tables into a new module. The application should be in the initial state when this message is sent.

`app<<Debug`

Invokes the JSL Debugger on a JMP application. The application script will run first. The Debugger then breaks as each module is created, invoking the module scripts. In the Debugger, set breakpoints to debug the scripts that are associated with the application or modules.

`app<<Edit`

Starts Application Builder on a JMP application that is in the initial state. There is no editor, and the application is not running.

`app<<Get Modules`

Gets the list of modules associated with an application. In Application Builder, each module corresponds to a tab in the workspace, which describes the layout and behavior for one type of window in the application.

`app<<Get Namespace`

The `JMP App()` object automatically creates an anonymous namespace for the variables created within the application script. Use this message to get a handle to this namespace to inspect or modify variables. There is a default symbol in this namespace called `thisApplication`, which holds a reference to the application itself.

`app<<Get Windows`

Gets a list of all windows created as instances of JMP app modules. Some modules might create more than one instance. All windows might not exist at the same time, so the number of windows might vary and might differ from the number of modules.

`app<<Open File(`*`path`*`)`

Resets the state of an existing application from a the .jmpapp or .jmappsource file. *`path`* is quoted.

`app<<Relaunch Analysis`

Creates a new copy of a running application and runs the new instance.

`app<<Run`

Runs the application. The application script runs first, and depending on settings, one or more JMP app module instance objects might be created automatically.

`app<<Save Script to Add-In`

`app<<Save Script to Data Table`

`app<<Save Script to Journal`

`app<<Save Script to Script Window`

Saves the script for the application to the given destination. An application script consists of a `JMP App()` object that contains the definition for the application. Scripts saved to an add-in, data table, or journal include a `Run` message to run the application. A script saved to the script window includes an `Edit` message to open Application Builder.

JMP App Module

A JMP application module is a definition of the display box layout and behavior for a single component in a JMP application or dashboard. Depending on the module type, the component might represent a window in the application or just part of a window.

module<<Create Instance

Use `Create Instance` within a `JMP App()` or `JMP App Module()` script to create an instance of a JMP app module. By default, one instance of each JMP application module is created when an application is run. For more complex applications with multiple windows, such as a launcher and report combination, it might be necessary to change the default settings and control how the module instance is created.

module<<Get Instance

Returns a handle to the application that owns a module.

JMP App Module Instance

The JMP application module instance is a running realization of a JMP app module, a window on the screen, or a collection of display box elements that can be inserted into another window.

inst<<Create Objects

Appears in the default template for a JMP app module script. This message controls the point at which the display and window for a module instance are created. The message appears in the script so that the script writer can choose to do certain setup before the objects are created. One example of this setup is for a parameterized application.

inst<<Get Box

Returns a handle to the top-most display box associated with a module instance. This might be useful to issue display or window commands, such as the `Save to PDF` or `Close Window` messages.

inst<<Get Namespace

Like `JMP App()`, each JMP application module instance also creates an anonymous namespace for all variables created in the module script. The namespace also includes all the variables that represent the display boxes in the module. This namespace contains a default symbol named `thisModuleInstance` that refers to itself.

inst<<Get User Data

inst<<Set User Data

Stores and retrieves a JSL value in the JMP application module instance. The value could be a number, quoted string, list, associative array, or other JSL type that is returned with the Type() function.

MATLAB

The MATLAB interfaces are scriptable using a MATLAB connection object. Use the MATLAB Connect() JSL function to obtain a scriptable MATLAB connection object.

mlconn<<Control(<Echo(*Boolean*)>, <Visible(*Boolean*)>)

Controls the execution of MATLAB.

Returns

None.

Optional Global Arguments

Echo(*Boolean*) Echo MATLAB source lines to the JMP Log window.

Visible(*Boolean*) Determine whether to show or hide the active MATLAB workspace.

mlconn<<Disconnect()

Description

Disconnects this MATLAB integration interface connection.

mlconn<<Execute({*list of inputs*}, {*list of outputs*}, *mCode*, <Expand(*Boolean*)>, <Echo(*Boolean*)>)

Submits MATLAB code to the active global MATLAB integration interface connection given a list of inputs and upon completion a list of outputs are retrieved.

Returns

0 if successful, otherwise nonzero.

Required Arguments

{*list of inputs*} Positional, name list. List of JMP variable names to be sent to MATLAB as inputs.

{*list of outputs*} Positional, name list. List of JMP variable names to be retrieved from MATLAB as outputs.

mCode Positional, quoted string. The MATLAB code to submit.

Optional Named Arguments

Expand(*Boolean*) Performs an Eval Insert on the MATLAB code prior to submission.

Echo(*Boolean*) Echos MATLAB source lines to the JMP Log window. Default is true.

---

mlconn<<Get Graphics(*format*)

Gets the last graphic object written to the MATLAB graph display window. The graphic object can be returned in several graphic formats.

Returns

JMP Picture object.

Optional Argument

format Positional. The quoted format the MATLAB graph display window contents are to be converted to. Valid formats are "png", "bmp", "jpeg", "jpg", "tiff", and "tif".

---

mlconn<<Get Version()

Gets the current version of the installed MATLAB.

Returns

Matrix, returns a vector of length 3 containing the MATLAB version number.

---

mlconn<<Get(*name*)

Description

Gets a named variable from MATLAB to JMP.

Returns

Value of named variable.

Required Argument

name The name of a JMP variable to be retrieved from MATLAB.

---

mlconn<<Is Connected()

Description

Determines whether connection is active.

Returns

1 if connected, otherwise 0.

---

mlconn<<JMP Name To MATLAB Name(*jmp name*)

Description

Maps a JMP variable name to its corresponding MATLAB variable name using MATLAB variable name naming rules.

Returns

A quoted string, a mapped MATLAB name.

Required Argument

`jmp name` Positional. The name of a JMP variable to be sent to MATLAB.

`mlconn<<Send(name, <named arguments>)`

Description

Sends the named variable from JMP to MATLAB.

Returns

0 if successful, otherwise nonzero.

Required Argument

`name` Positional. The name of a JMP variable to be sent to MATLAB.

Named Arguments

The following arguments are for data tables only:

`Selected(Boolean)` Send selected rows from the referenced data table to MATLAB.

`Excluded(Boolean)` Send only excluded rows from the referenced data table to MATLAB.

`Labeled(Boolean)` Send only labeled rows from the referenced data table to MATLAB.

`Hidden(Boolean)` Send only hidden rows from the referenced data table to MATLAB.

`Colored(Boolean)` Send only colored rows from the referenced data table to MATLAB.

`Markered(Boolean)` Send only markered rows from the referenced data table to MATLAB.

`Row States (Boolean, <named arguments>)` Send row states from referenced data table to MATLAB by adding an additional data column named "RowState". Create multiple selections by adding together individual settings. The row state consists of individual settings with the following values:

- Selected = 1
- Excluded = 2
- Labeled = 4
- Hidden = 8
- Colored = 16
- Markered = 32

Row State Optional Named Arguments

The following optional, named Row States arguments are supported:

`Colors(Boolean)` Send row colors. Adds additional data column named "RowStateColor".

Markers(*Boolean*) Send row markers. Adds additional data column named "RowStateMarker".

---

mlconn<<Submit(*mCode*, *<named arguments>*)

Description

Submits MATLAB code to the active global MATLAB integration interface connection.

Returns

0 if successful, otherwise nonzero.

Required Argument

mCode Positional quoted string. The MATLAB code to submit.

Named Arguments

Expand(*Boolean*) Perform an Eval Insert on the MATLAB code prior to submission.

Echo(*Boolean*) Echo MATLAB source lines to the JMP log. The default is true.

---

mlconn<<Submit File(*path*)

Description

Submits statements to MATLAB using a quoted *path*.

Returns

0 if successful, otherwise nonzero.

Arguments

path Positional quoted string. The path to a file containing the MATLAB source lines to be executed.

Namespaces

---

ns<<Contains(*string*)

Returns 1 or 0, depending on whether the specified quoted string exists within the namespace.

---

ns<<Delete Namespace

Removes this namespace from the internal global list.

To delete variables in the namespace, use the Remove(*variable name*) message.

---

ns<<First

Returns a quoted string that contains the first variable name used within the namespace.

`ns<<Get Contents`

Returns a list of key-value pairs, which are each enclosed in a list. Each key is a quoted string that contains a variable name, and each value is the unevaluated expression that the variable contains.

`ns<<Get Keys`

Returns a list of variable names.

`ns<<Get Name`

Returns the name of this namespace.

`ns<<Get Value(variable name);`

Returns the unevaluated expression that the quoted *variable name* contains in this namespace.

`ns<<Get Values`

Returns a list of unevaluated expressions that each variable in the namespace contains.

`ns<<Get Values({variable name1, variable name2, ... });`

Returns a list of unevaluated expressions that each quoted variable in the namespace specified in the list argument contains. If a requested variable name is not found, an error is returned.

`ns<<Insert(variable name, expr);`

Inserts into this namespace a quoted variable named *variable name* that holds the expression *expr*.

`ns<<Lock Namespace(<variable name, ...>)`

Locks all specified variables in the namespace and prevents quotes variables from being added or removed. If no variables are specified, all variables in the namespace are locked.

`ns<<N Items`

Returns the number of variables contained in the namespace.

`ns<<New Namespace(name, <{list of expressions}>)`

Creates a namespace where all functions and variables created are defined only within the optional quoted *name* argument.

`ns<<Next(variable name)l`

Returns the name of the variable that follows the specified quoted variable.

`ns<<Remove(variable name, ...)`

Removes the specified quoted variable or list of variables.

`ns<<Show Contents`

Shows the contents of a namespace in the log.

`ns<<Unlock Namespace(variable name, ...);`

Unlocks the specified quoted variables in the namespace. If no variables are specified, all variables are unlocked.

Platforms

`obj<<Action`

Evaluates expressions. Useful for quoted stringing together multiple platforms interrupted by user input.

`obj<<Automatic Recalc`

Redoes the analysis automatically for exclude and data changes. If automatic recalc is on, you should use `Wait(0)` commands to let the triggers take effect and do the recalculation.

Note: Not supported on all platforms.

`obj<<Bring Window To Front`

Brings the current window to the front.

`obj<<Close Window`

Closes window identified by *obj*, typically a platform surface.

`obj<<Column Switcher(`*`default column, {column1, column2, ...}`*`)`

Adds a control panel to a platform for switching variables.

`obj<<Copy ByGroup Script`

Creates a script to produce this analysis containing By variables and place it on the clipboard.

`obj<<Copy Script`

Creates a script to produce this analysis and place it on the clipboard.

`obj<<Data Table Window`

Makes the associated data table window active (front-most).

`obj<<Get Data Table`

Returns a reference to the data table.

`obj<<Get Script`

Returns script to reproduce the analysis as an expression in the log.

`obj<<Get Script With Data Table`

Creates a script to reproduce the analysis, specifically referencing the source data table, and returns it as an expression in the log.

`obj<<Get Timing`

Times the launch of the platform and returns it in the log.

`obj<<Get Web Support`

Returns the score for the display tree that is about to be saved as interactive HTML. Possible values are -1 (unsupported), 0 (supported), and 1 (supported). If the score does not equal -1, interactive HTML is supported and the `Save Interactive HTML` message can be used.

`obj<<Get Window Position`

Gets the position of the window. Returns an ordered pair.

obj<<Get Window Size

Gets the window size, in pixels. Returns an ordered pair.

obj<<Ignore Platform Preferences(*Boolean*)

Ignores the current settings of the platform's preferences. The message is ignored when sent to the platform after creation.

obj<<Journal Window

Appends the contents of the window to the journal.

obj<<Local Data Filter

Filters data to specific groups or ranges, but stays local to the platform.

obj<<Maximize Window

Maximizes the window. Equivalent to pushing the maximize button in the corner of the window. This message takes an optional Boolean argument:

```
// maximize the window:
obj<<Maximize Window(1)
// restore the window:
obj<<Maximize Window(0)
```

obj<<Minimize Window

Minimizes the window. Equivalent to pushing the minimize button in the corner of the window. This message takes an optional Boolean argument:

```
// minimize the window
obj<<Minimize Window( 1 )
// restore the window
obj<<Minimize Window( 0 )
```

obj<<Move Window(*x*, *y*)

Moves the window to the (*x*, *y*) location on your screen.

obj<<Print Window

Sends the selected window to the printer.

obj<<Redo Analysis

Reruns the analysis with the same options.

`obj<<Redo ByGroup Analysis`

Reruns the same analysis involving By groups.

`obj<<Relaunch Analysis`

Returns to the launch window for this analysis.

`obj<<Relaunch ByGroup`

Returns to the launch window for this analysis involving By groups.

`obj<<Remove Column Switcher`

Removes all Column Switchers that were added to the platform.

`obj<<Remove Local Data Filter`

Removes all Local Data Filters that were added to the platform.

`obj<<Report`

`Report(`*`obj`*`)`

Returns a display box reference for the report in the platform window.

`obj<<Report View`

Determines the level of detail visible in a platform report. Full shows all detail and Summary shows only select content, dependent upon the specific platform. For customized behavior, use the `Set Summary Behavior` message with display boxes.

`obj<<Save ByGroup Script to Data Table(<`*`name`*`>, <Append Suffix(`*`Boolean`*`)>, <Prompt(`*`Boolean`*`)>, <Replace(`*`Boolean`*`)>)`

Description

Creates a table script to produce the analysis involving By variables and saves it as a table script in the data table.

Optional Arguments

`name` The name of the script. *`name`* is quoted. If omitted, the platform names the script. For example, in Tabulate, the script is named "Tabulate". In Bivariate, the script might be named "Bivariate of height by weight" to reflect the platform and column names.

`Append Suffix(`*`Boolean`*`)` If true, appends a numerical suffix to the script name. This suffix differentiates the script from an existing script with the same name.

`Prompt(`*`Boolean`*`)` If true, prompts the user to specify a script name.

Replace(*Boolean*) If true, replaces an existing script with the same name.

---

obj<<Save ByGroup Script to Journal

Creates a table script to produce the analysis involving By variables and adds a button to the journal containing this script.

---

obj<<Save ByGroup Script to Script Window

Creates a script to produce the analysis involving By variables and appends it to the current Script window.

---

obj<<Save Script for All Objects

Saves script to reproduce all analyses found within the object's window in the Script Journal window.

---

obj<<Save Script for All Objects to Data Table

Saves a script for all report objects to the current data table. The script is named after the platform unless you specify the script name in quotes.

```
obj << Save Script for All Objects To Data Table("My Script")
```

---

obj<<Save Script to Data Table

Saves script to reproduce analysis as a property in the associated data table.

---

obj<<Save Script to Journal

Creates a script to produce the analysis and adds a button to the journal containing this script.

---

obj<<Save Script to Report

Saves script to reproduce analysis as a text box at the top of the report.

---

obj<<Save Script to Script Window

Saves a script to reproduce analysis in the Script Journal.

---

obj<<Scroll Window(*x*, *y*)

obj<<Scroll Window({*x*, *y*})

Scrolls the window *x* pixels to the right and *y* pixels down from the current position. Negative coordinates go left and up. If the coordinates are a list in braces { }, they are

absolute coordinates. The window scrolls to the point *x* pixels from the left and *y* pixels from the top.

`obj<<SendToReport`

Used with the Dispatch function to customize the appearance of a report.

`obj<<SendToByGroup`

Sends messages to open platforms or turn on platform features to each level of a by-group.

`obj<<Show Window(`*Boolean*`)`

1 shows the window (brings it to the front). 0 hides the window. If the window is also minimized (on Windows) or docked (on macOS), showing the window restores it to the normal state and brings it to the front.

`obj<<Size Window(`*x*`, `*y*`)`

Resizes the window to *x* pixels wide by *y* pixels high.

`obj<<Title(`*new title*`)`

Sets the quoted title of the platform.

`obj<<Top Report`

Returns a reference to the top display box in the report. Useful for By groups or other cases when several platform reports are in one window.

`obj<<View Web XML`

Returns the XML used to create the interactive HTML report. The XML code appears in the log.

`obj<<Zoom Window`

Resizes the window to be large enough to show all of its contents.

Bubble Plot

```
bp<<Set
Shape("Circle"|"Triangle"|"Square"|"Diamond"|"Arrow"|"Custom")
```

Description

Sets the shape for the bubble. When specifying a custom shape, use the `Set Custom Path` message to specify the path for the bubbles.

See Also

"bp<<Set Custom Path(path matrix|path text)".

```
bp<<Set Custom Path(path matrix|path text)
```

Sets a path for custom bubbles.

Arguments

path matrix An *N*x3 matrix.

path text A quoted string that contains SVG code.

See Also

"bp<<Set Shape("Circle"|"Triangle"|"Square"|"Diamond"|"Arrow"|"Custom")".

DOE

```
obj<<Get Prediction Variances
```

Description

Returns a vector of the MC variances used for the FDS plot.

Example

```
dt = Open( "$SAMPLE_DATA/Design Experiment/Bounce Data.jmp" );
d = DOE( Evaluate Design, X( :Silica, :Sulfur, :Silane ), Y( :Stretch ) );
d << Get Prediction Variances;
```

```
obj<<Set Number of FDS Points()
```

Description

Enables you to specify the number of runs used to generate an FDS plot.

Example

```
dt = Open( "$SAMPLE_DATA/Design Experiment/Bounce Data.jmp" );
d = DOE( Evaluate Design, X( :Silica, :Sulfur, :Silane ), Y( :Stretch ) );
d << Set Number of FDS Points( 20000 );
```

Partition

`obj<<Initial Splits(`*`condition`*`, `*`{left}`*`, `*`{right}`*`)`

Description

Describes the splits that are performed.

Example

```
dt = Open( "$SAMPLE_DATA/Car Poll.jmp" );
obj = Partition(
  Y( :country ),
  X( :sex, :marital status, :age, :type, :size ),
  Method( "Decision Tree" ),
  Initial Splits( :size == {"Large"}, {}, {:size == {"Medium"}} )
);
```

Notes

The condition is for the left side and is either `[name compareoperator value]` or `[name == list of values]`. The left is an empty list if the right has splits. Omit the right side if there are no splits. The left and right continue recursively in this format.

Response Screening

`obj<<Get PValues`

Returns a reference to a PValues table.

`obj<<Save PValues`

Stores the *p*-values in an output data table.

`obj<<Save Compare Means`

Stores the means comparisons in an output data table.

`obj<<Save Mean`

Stores the means in an output data table.

`obj<<Save Outlier Indicator`

Saves Outlier Indicator for each fit.

`obj<<Save Std Residuals`

Saves the residual formula for each fit.

obj<<Select Columns

Select columns in the original table corresponding to selected rows in this table.

Tabulate

obj<<Display Column Width(*<Data Column(Column Table(n),<column name path>), Row Label(Row Table(n), <column name path>)>, <width>*)

Returns or sets the display pixel width of a column in a Tabulate table.

Required Argument

Row Label Use Row Table and heading for columns in the row labels area.

Optional Arguments

Data Column Use Column Table and column references to define columns in the main body of the table.

column name path Specifies the Column Table or Row Table (both quoted), and the series of column headings that traces the path of the column. **Note:** Column Table or Row Table can be omitted if the table referenced is the first table.

width Specifies the pixel width of the column.

Examples

```
dt = Open( "$SAMPLE_DATA/Car Poll.jmp" );
obj = dt << Tabulate(
  Add Table(
     Column Table(
        Grouping Columns( :sex, :marital status ),
        Analysis Columns( :age ),
        Statistics( Sum, "% of Total" )
     ),
     Row Table( Grouping Columns( :type ) ),
     Row Table( Grouping Columns( :country, :size ) )
  )
);
Wait( 3 ); // for demonstration purposes
obj << Display Column Width( Row Label( Row Table( 2 ), "country" ), 150 );
Wait( 3 ); // for demonstration purposes
obj << Display Column Width(
  Data Column(
     Column Table( 1 ),
     "sex",
     "Female",
     "marital status",
     "Married",
     "age",
```

```
        "Sum"
    ),
    150
);
```

Python Integration Messages

The Python interfaces are also scriptable using a Python connection object. A scriptable Python connection object can be obtained using the `Python Connect()` function. See "Python Connect(<Echo(Boolean),> <Path(path),> <Use Python Version(string),> <Python System Path(list)>)".

`pythconn<<Control(<Interactive(`*`Boolean`*`)>|<Echo(`*`Boolean`*`)>)`

Description

Controls the execution of Python.

Optional Named Arguments

`Interactive(`*`Boolean`*`)` Enables interactive mode in the Python matplotlib package. Determines whether the graphics window is released or closed when graphics rendering is complete.

`Echo(`*`Boolean`*`)` Global argument. Prints the Python source lines to the JMP log. The default value is true.

`pythconn<<Disconnect`

Disconnects the Python integration interface connection.

`pythconn<<Execute(`*`{list of inputs}, {list of outputs}, Python code, <named arguments>`*`)`

Description

Submits Python code to the active global Python integration interface connection given a list of inputs. On completion, returns a list of outputs.

Returns

Returns 0 if successful and 1 otherwise. The results are returned using the list of outputs. Given each element of the JMP output list, the corresponding Python variable value is returned.

Positional Arguments

`{list of inputs}` A list of JMP variable names to be sent to Python as inputs.

`{list of outputs}` A list of JMP variable names to be retrieved from Python as outputs.

Python code The quoted Python code to submit.

Optional Named Arguments

See "pythconn<<Submit(Python code, <Expand(Boolean)>, <Echo(Boolean)>)".

---

pythconn<<Get(*name*)

Description

Gets a named variable from Python.

Returns

Returns the value of the named variable.

Argument

name The name of the JMP variable to be received from Python. The argument can represent any of the following Python data types: numeric, quoted string, matrix, list, or data frame.

---

pythconn<<Get Graphics(*format*)

Description

Gets the last graphics object written to the Python graph display window in the specified graphics format. The graphics object can be returned in several different graphic formats.

Returns

Returns a JMP picture object.

Argument

format The format that the Python graph display window contents are to be converted to. Valid formats are PNG, BMP, JPEG, JPG, TIFF, and TIF.

---

pythconn<<Get Version

Description

Gets the current version of the Python installation.

Returns

Returns a list of length 5 that contains the five components of the version number: major, minor, micro, releaselevel, and serial. The releaselevel value is a quoted string.

---

pythconn<<Is Connected

Description

Determines if the connection is active.

Returns

Returns 1 if connected and 0 otherwise.

`pythconn<<JMP Name to Python Name(`*name*`)`

Description

Maps a JMP variable name to its corresponding Python variable name using Python variable name naming rules.

Argument

`name` The name of the JMP variable to be sent to Python. Some variable names allowed by JMP are not allowed by Python. When you send using these variables from JMP to Python (the `Send` message), their names get changed. Use `JMP Name to Python Name` to determine what the variable name was changed to.

`pythconn<<Send(`*name*`)`

Description

Sends a named variable from JMP to Python.

Returns

Returns 0 if successful and non-zero otherwise.

Argument

`name` The name of the JMP variable to be sent to Python.

`pythconn<<Submit(`*Python code*`, <Expand(`*Boolean*`)>, <Echo(`*Boolean*`)>)`

Description

Submits Python code to the active global Python integration interface connection.

Returns

Returns 0 if successful and 1 otherwise.

Required Arguments

Python code The quoted Python code to submit.

Optional Arguments

`Expand(`*Boolean*`)` Performs an `Eval Insert()` on the Python code before submission.

`Echo(`*Boolean*`)` Prints the Python source lines to the JMP log. The default value is true.

`pythconn<<Submit File(`*path*`)`

Description

Submits statements to Python using the quoted path name.

Argument

`path` The quoted path to the file that contains the Python source lines to be executed.

R Integration Messages

The R interfaces are also scriptable using an R connection object. A scriptable R connection object can be obtained using the `R Connect()` function.

`rconn<<Control(Interrupt|Async(`*`Boolean`*`)|Echo(`*`Boolean`*`))`

Changes the control options for R. If `Async` is set to true (`1`) for `R Submit()`, this message immediately stops the execution of the R code that was submitted.

`rconn<<Disconnect()`

Disconnects this R connection.

`rconn<<Is Connected()`

Returns 1if the R connection is active, 0 otherwise.

`rconn<<Send File(`*`name`*`, `*`<named arguments>`*`)`

Send the specified JMP variable to R.

Returns

0 if successful, nonzero otherwise.

Required Argument

`name` A quoted string contains the name of a JMP variable to send to R.

Optional Named Arguments for Data Tables

`Selected(`*`Boolean`*`)` If true, sends only the selected rows from the referenced data table to R.

`Excluded(`*`Boolean`*`)` If true, sends only the excluded rows from the referenced data table to R.

`Labeled(`*`Boolean`*`)` If true, sends only labeled rows from the referenced data table to R.

`Hidden(`*`Boolean`*`)` If true, sends only hidden rows from the referenced data table to R.

`Colored(`*`Boolean`*`)` If true, sends only colored rows from the referenced data table to R.

`Markered(`*`Boolean`*`)` If true, sends only markered rows from the referenced data table to R.

`Row States(`*`Boolean`*`, `*`<named arguments>`*`)` Includes a Boolean argument and optional named arguments. Sends row state information from the referenced data table to R by adding an additional data column named “RowState”. The row state value consists of individual settings with the values shown in Table 3.2.

Table 3.2 Row States

Multiple row states are created by adding together individual settings.	Selected = 1 Excluded = 2 Labeled = 4 Hidden = 8 Colored = 16 Markered = 32
Arguments	`Colors(`*`Boolean`*`)` (Optional) If true, sends row colors and adds an additional data column named "RowStateColor". `Markers(`*`Boolean`*`)` (Optional) If true, sends row markers and adds an additional data column named "RowStateMarker".

`rconn<<Send(`*`name, <R Name(name)>`*`)`

Sends the quoted JMP data file to R. The *`name`* argument can represent any of the following data types: numeric, quoted string, matrix, list, or data table.

`rconn<<Get(`*`name`*`)`

Returns data from R. The *`name`* argument can represent any of the following data types: numeric, quoted string, matrix, list, or data table.

Returns

The value of the specified variable.

Arguments

`name` Specifies the quoted name of a JMP variable to be retrieved from R.

`rconn<<Get Graphics(`*`type`*`)`

Gets the last graphics object written to the R graph display window. The graphics object can be returned in different graphic formats.

Returns

A JMP picture object.

Required Argument

`type` The format the R graph display window contents are converted to. Valid formats are `"png"`, `"bmp"`, `"jpeg"`, `"jpg"`, `"tiff"`, and `"tif"`.

`rconn<<Submit(R code, Expand(Boolean), Echo(Boolean))`

Submits the quoted R code.

Returns

0 if successful, nonzero otherwise.

Required Argument

`code` Specifies the quoted R code to submit.

Optional Named Arguments

`Expand(Boolean)` Performs an `Eval Insert()` on the R code before submitting the code.

`Echo(Boolean)` Echoes the R source lines to the JMP log. The default value is true.

`Rconn<<Submit File(path)`

Submits statements to R using the file in the quoted *path*.

Arguments

`path` Specifies the quoted path to the file that contains R code to be executed.

`rconn<<Execute({list of inputs}, {list of outputs}, R code, <named arguments>)`

Submits the quoted R code to the R connection using the list of inputs. Upon completion, a list of outputs is returned.

Returns

0 if successful, nonzero otherwise.

Required Arguments

R code Specifies the quoted R code to submit.

`{list of inputs}` List of JMP variable names to be sent to R as inputs.

`{list of outputs}` List of JMP variable names to be retrieved from R as outputs.

Optional Named Arguments

See "rconn<<Submit(R code, Expand(Boolean), Echo(Boolean))".

`rconn<<Control(<Echo(Boolean)>)`

Controls the execution of R.

Returns

Void.

Optional Named Argument

`Echo(Boolean)` Echoes the R source lines to the JMP log.

rconn<<Get Version()

Gets the current version of R that is installed.

Returns

A vector of length 3 containing the R version number.

rconn<<JMP Name To R Name(*name*)

Maps a quoted JMP Name to its corresponding R Name using R variable name naming rules.

Returns

A quoted string that contains the quoted R name.

Arguments

name A quoted string that specifies the name of a JMP variable to be sent to R.

SAS Integration Messages

Metadata Server Objects

metaserver<<Disconnect()

Description

Disconnects the metadata server.

Returns

Void.

metaserver<<Get Display Name()

Description

Gets the display name of the metadata server.

Returns

A quoted string.

metaserver<<Get Host Name()

Description

Gets the host (machine) name of the metadata server.

Returns

A quoted string.

`metaserver<<Get Port()`

Description

Gets the port used for the metadata server connection.

Returns

An integer.

`metaserver<<Get User Identity()`

Description

Gets the identify of the connected user as defined in metadata.

Returns

A quoted string.

`metaserver<<Get User Name()`

Description

Gets the user name (login ID) that was used for the metadata server connection.

Returns

A quoted string.

SAS Server Objects

`sasconn<<Assign Libref(`*`libref`*`, `*`path`*`, `*`engine`*`, `*`engine options`*`)`

Description

Assign a SAS libref on this SAS server connection.

Returns

Void.

Arguments

See “SAS Assign Lib Refs("libref", "path", <"engine">, <"engine options">)”.

`sasconn<<Cancel Submit()`

Description

Cancels the currently running SAS Submit for this server that is presumably running asynchronously.

Returns

1 if a running submit was found and canceled; 0 otherwise.

sasconn<<Clear Log History()

Description

Clears the SAS Log history for this server.

Returns

Void.

sasconn<<Clear Output History()

Clears the SAS Output history for this server.

sasconn<<Connect(<User Name(*name*)>, <Password(*password*)>, <Prompt("Always"|"Never"|"IfNeeded")>)

Description

Attempt to reconnect a SAS server connection object that has become disconnected.

Returns

1 if the connection was successful, 0 otherwise.

Optional Named Arguments

User Name(*name*) Specifies the quoted user name for the connection.

Password(*password*) Specifies the quoted password for the connection.

Prompt A quoted keyword. "Always" means always prompt before attempt to connect. "Never" means never prompt even if the connection attempt fails (just fail with an error message going to the log), and "IfNeeded" (the default) means prompt if the attempt to connect with the given arguments fails (or is not possible with the information given).

sasconn<<Deassign Libref(*libref*)

Description

De-assign the quoted SAS libref on this SAS server connection.

Returns

Void.

Arguments

libref Specifies the quoted library reference.

sasconn<<Disconnect()

Description

Disconnect this SAS server connection.

Returns

Void.

`sasconn<Does Module Exist(module name)`

Description

Determines whether the specified SAS module exists in the SAS installation represented by the SAS connection. This can be helpful in determining whether certain SAS products are installed. The SAS DATA Step function MODEXIST is used to determine module existence.

Returns

1 if the specified module is found to exist, 0 if it does not exist.

Argument

`module name` Specifies the quoted SAS module, the existence of which should be checked. Do not include any extension.

`sasconn<<Export Data(dt, library, dataset, <named arguments>)`

Description

Exports a JMP data table to the specified SAS data set in the specified library on the active SAS server connection.

Returns

1 if the data table was exported successfully; 0 otherwise.

Optional Named Arguments

See "SAS Export Data(dt, "library", "dataset", <named_arguments>)".

`sasconn<<Get Data Sets(libref)`

Description

Returns a list of the data sets defined in a SAS library on this SAS server connection.

Returns

A list of quoted strings.

Arguments

libref Specifies the quoted SAS libref or friendly library name associated with the library for which the list of defined SAS data sets will be returned.

`sasconn<<Get Error Count()`

Description

Gets the count of the number of errors encountered in the previous SAS Submit.

Returns

An integer.

`sasconn<<Get File(source, dest)`

Description

Downloads a file from this SAS server connection.

Returns

Void.

Arguments

See "SAS Get File("source", "dest", "encoding")".

`sasconn<<Get File Names(fileref)`

Description

Gets a list of filenames found in the quoted *fileref* on this SAS server connection.

Returns

A list of quoted strings.

Arguments

fileref A quoted string that contains the name of *fileref* from which to retrieve filenames.

`sasconn<<Get File Names In Path(path)`

Description

Gets a list of filenames found in the quoted *path* on the current SAS server connection.

Returns

A list of quoted strings.

Arguments

path The quoted directory path on the server from which to retrieve filenames.

`sasconn<<Get File Refs()`

Description

Gets a list of the currently defined SAS filerefs on this SAS server connection.

Returns

A list of quoted strings.

`sasconn<<Get Librefs(<named arguments>)`

Description

Gets a list of the currently defined SAS librefs on this SAS server connection.

Returns

A list of quoted strings.

Optional Named Arguments

See "SAS Get Lib Refs(<named arguments>)".

`sasconn<<Get Log()`

Description

Retrieves the SAS Log from the last SAS Submit from this SAS server connection.

Returns

A quoted string.

`sasconn<<Get Option Name()`

Description

Queries SAS for the value of a SAS option variable.

Returns

A quoted string.

Example

The following script iterates through the define variables and prints out the values:

```
option_names = sasconn << Get Option Names();
For(i=1, i <= N Items(option_names), i++,
   option_value = sasconn << Get Option Value (option_names[i]);
   output = option_names[i] || "=" || char(option_value) || "/!n";
   Write(output);
);
```

`sasconn<<Get Output()`

Description

Retrieves the listing output from the last submission of SAS code to this SASServer object.

Returns

A quoted string.

`sasconn<<Get Results()`

Description

Retrieves the results of the previous SAS Submit as a scriptable object, which allows significant flexibility in what to do with the results.

Returns

A SAS Results Scriptable Object.

`sasconn<<Get Submit Status()`

Description

Gets the current status of a SAS Submit for this server that is presumably running asynchronously.

Returns

1 if the submit has not started; 2 if the submit is running; 3 if the submit has been canceled; 10 if the submit has completed successfully; 11 if the submit has completed with errors.

`sasconn<<Get Var Info(`*`libref`*`, `*`dataset`*`, <Password(`*`password`*`)>)`

Description

Returns information about the variables the specified SAS data set.

Required Arguments

`libref` The library name.

`dataset` The quoted name of the data set from which to retrieve variable names.

Optional Argument

`Password(`*`password`*`)` The quoted *`password`* for the connection.

`sasconn<<Get Var Names(`*`libref`*`, `*`dataset`*`, <`*`named arguments`*`>)`

Description

Retrieves the variable names contained in the specified data set on this SAS server connection.

Returns

A list of quoted strings.

Arguments

See “SAS Get Var Names(`string`, <"dataset">, <password("password")>)”.

`sasconn<<Get Version(<"Long">)`

Description

Returns the SAS version as a quoted string such as “9.3” or “9.4”.

Returns

A quoted string that contains the SAS version.

Optional Argument

`Long` A quoted keyword that specifies to return the long SAS version, which corresponds to the SYSVLONG SAS macro (for example, "9.02.02M0P01152009").

sasconn<<Get Work Folder()

Description

Returns the full path of the folder corresponding to the WORK library for this server.

Returns

A quoted string that specifies the work folder path.

sasconn<<Import Data(*library*, *dataset*, *<named arguments>*)

Description

Imports a SAS data set from this SAS server connection into a JMP table.

Returns

A JMP Data Table object.

Arguments

See "SAS Import Data(string, <"dataset">, <named arguments>)".

sasconn<<Import Query(*sqlquery*, *<named arguments>*)

Description

Executes the requested SQL query on this SAS server connection, importing the results into a JMP data table.

Returns

A JMP data table object.

Arguments

See "SAS Import Query("sqlquery", <named arguments>)".

sasconn<<Is Connected()

Description

Determines whether this SAS Server object is currently connected to SAS.

Returns

1 if *sasconn* is connect, 0 otherwise.

sasconn<<Is Product Available(*product name*)

Description

Determines whether the quoted SAS product is both licensed and installed in the session represented by the SAS connection. The SAS DATA Step functions SYSPROD and MODEXIST are used to determine the licensed and installed status of the product.

Returns

1 if the specified product is licensed, 0 if the product is not licensed, or -1 if the specified product is not recognized by SAS. This function throws an exception if the requested product is not one for which JMP knows how to check the installed status.

Required Argument

`product name` The quoted SAS product for which licensing should be checked. The product name can be specified with or without the "SAS/" prefix.

`sasconn<<Is Product Licensed(`*`product name`*`)`

Description

Determines whether the quoted SAS product is licensed in the session represented by the SAS connection. The SAS DATA Step function SYSPROD is used to determine the licensing status of the product.

Returns

1 if the specified product is licensed, 0 if the product is not licensed, or -1 if the specified product is not recognized by SAS.

Required Argument

`product name` The quoted SAS product for which licensing should be checked. The product name can be specified with or without the "SAS/" prefix.

`sasconn<<Kill Session(<`*`n`*`>)`

Description

If no argument is provided, the SAS connection is immediately terminated.

Returns

Void.

Arguments

`n` An optional number. The system waits n seconds for a normal shut-down before immediately terminating the SAS connection.

`sasconn<<Load Text File(`*`path`*`, <named arguments>)`

Description

Downloads the file specified in the quoted *`path`* from the active SAS server connection and retrieve its contents as a quoted string.

Returns

A quoted string.

Arguments

See "SAS Load Text File("path")".

`sasconn<<Open Log Window()`

Description

Opens (or brings to the front) the SAS Log window for this server.

Returns

Void.

`sasconn<<Open Output Window()`

Description

Opens (or brings to the front) the SAS Output window for this server.

Returns

Void.

`sasconn<<Open SAS Results()`

Description

Opens the results from the previous SAS Submit. Intended to be used with asynchronous SAS submits or the use of the OnSubmitComplete option to SAS Submit to give the JSL author a way to conditionally open the results of a submit.

Returns

Void.

`sasconn<<Open Submit Results()`

Description

Opens all the results from the last SAS Submit command.

Returns

Void.

`sasconn<<Send File(`*`source`*`, `*`dest`*`)`

Description

Uploads a file to this SAS server connection.

Returns

Void.

Arguments

See “SAS Send File("source", "dest", "encoding")”.

`sasconn<<Submit(sas code, <named arguments>)`

Description

Submits quoted SAS code to this SAS server connection.

Returns

Void.

Arguments

See "SAS Submit("sasCode", <named arguments>)".

`sasconn<<Submit File(filename, <named arguments>)`

Description

Submits a SAS code file to this SAS server connection.

Returns

Void.

Arguments

See "SAS Submit File("filename", <named arguments>)".

Stored Processes

`stp<<Begin Run(<named arguments>)`

Description

Starts this stored process executing in the background. This message is paired with `End Run`, which should also be called at some point after `Begin Run` to wait for the stored process to complete.

Returns

`-1` = execution failed.

`1` = not started.

`2` = running.

`3` = canceled.

`10` = completed successfully.

`11` = completed with errors.

Optional Named Arguments

Same as `Run`, except `AutoOpenResults` and `NoAlerts` are not supported. They are available on `EndRun`.

`AutoResume(<filename>)` If specified with no argument, it specifies that the stored process results should be auto-opened when the stored process completes. If a quoted

filename is specified, filename is opened rather than all results of the stored process being auto-opened.

`AutoResumeScript(`*`script`*`)` Specifies that after stored process execution completes, the quoted script should be evaluated. If the script is a function taking at least one argument, the function is evaluated with the scriptable stored process object passed as the first (and only) argument. `AutoResume` and `AutoResumeScript` are mutually exclusive.

`stp<<Delete Results(`*`<named arguments>`*`)`

Description

Deletes all results from the execution of this stored process.

Returns

`1` if deletion is successful, `0` otherwise (error message to JMP log).

Optional Named Arguments

`NoAlerts(`*`Boolean`*`)` If `True`, the user is not prompted for confirmation before the attempt is made to delete results.

`DeleteDirectory(`*`Boolean`*`)` If true, deletes the directory containing the stored process results along with the result files themselves. The default value is true.

`stp<<Edit Param Values()`

Description

Opens the stored process window for interactively setting parameter values.

Returns

1 if the user clicks OK to dismiss the window, 0 if the user clicks Cancel.

`stp<<End Run(`*`<named arguments>`*`)`

Description

Waits a specified amount of time (or forever) for a stored process started with `Begin Run` to complete. If the stored process is complete, retrieves the results, and opens them.

Returns

`-1` = execution failed.

`1` = not started.

`2` = running.

`3` = canceled.

`10` = completed successfully.

`11` = completed with errors.

Optional Named Arguments

AutoOpenResults(*Boolean*) Optional, Boolean. If True, results are automatically opened if the stored process completes in the time specified by *MaxWait*. If False, results are not automatically opened, and can be manually opened via the object returned by the Get Results message. Default is True.

MaxWait(*milliseconds*) An integer that specifies the maximum amount of time in milliseconds to wait for the stored process to complete. If MaxWait is not specified, End Run waits forever for the stored process to complete.

NoAlerts(*Boolean*) If True, error messages are sent to the JMP log rather than message boxes. The default value is False.

---

stp<<Get Metadata Id()

Description

Returns the metadata ID of the stored process.

Returns

A quoted string.

---

stp<<Get Metadata Path()

Description

Returns the full metadata path of the stored process.

Returns

A quoted string.

Stp<<Get Name()

Description

Returns the name of the stored process.

Returns

A quoted string.

---

stp<<Get Param Enum Labels(*name*)

Description

Gets the enumeration labels specified by the quoted *name* for a parameter.

Returns

A list of quoted strings.

Arguments

name Specifies the quoted name of the parameter whose enumeration labels to retrieve.

`stp<<Get Param Enum Values(`*name*`)`

Description

Gets the possible enumerated values for a parameter.

Returns

A list of quoted strings.

Arguments

`name` Specifies the quoted name of the parameter whose possible enumerated values to retrieve.

`stp<<Get Param Names(<`*named arguments*`>)`

Description

Gets a list of parameter names for this stored process of specific types.

Returns

A list of quoted strings.

Optional Named Arguments

`Visible(`*Boolean*`)` If true, gets only visible parameters. If `False`, gets only non-visible parameters. If not specified, gets both visible and non-visible parameters.

`Modifiable(`*Boolean*`)` If true, gets only modifiable parameters. If `False`, gets only non-modifiable parameters. If not specified, gets both modifiable and non-modifiable parameters.

`Required(`*Boolean*`)` If true, gets only required parameters. If `False`, gets only non-required parameters. If not specified, gets both required and non-required parameters.

`Expert(`*Boolean*`)` If true, gets only expert parameters. If `False`, gets only non-expert parameters. If not specified, gets both expert and non-expert parameters.

`stp<<Get Param Value(`*name*`)`

Description

Gets the current value of the specified parameter.

Returns

A quoted string.

Required Argument

name Specifies the name of the parameter whose value to retrieve.

`stp<<Get Results()`

Description

Gets the results generated by the execution of this stored process as a scriptable object.

Returns

A SAS Results scriptable object.

`stp<<Get Status()`

Description

Gets the execution status of the stored process.

Returns

-1 = execution failed.

1 = not started.

2 = running.

3 = canceled.

10 = completed successfully.

11 = completed with errors.

`stp<<Get Status Message()`

Description

Gets the message associated with the failure of the stored process, if any.

Returns

A quoted string.

`stp<<Reset Param Values()`

Description

Resets all parameter values to their metadata-defined default values.

Returns

Void.

`stp<<Run(<named arguments>)`

Description

Executes this stored process object in the foreground.

Returns

-1 = execution failed.

1 = not started.

2 = running.

3 = canceled.

10 = completed successfully.

11 = completed with errors.

Optional Named Arguments

AutoOpenResults(*Boolean*) If True, results are automatically opened when the stored process completes. If False, results are not auto-opened, and can be manually opened via the object returned by the GetResults message. The default value is True.

UserName(*username*) Specifies the quoted *user name* under which to run the stored process.

Password(*password*) Specifies the quoted *password* for *UserName*.

AuthDomain(*authDomain*) Specifies the quoted authentication domain of the credentials (*username*, *password*) given.

ODSDest(*dest*) Specifies the quoted ODS destination ("HTML", "PDF", "tagsets.SASReport12") for any ODS-generated results from the stored process. This requires the stored process SAS code to call %STPBEGIN. The default value is "HTML".

GraphicsDevice(*device*) Specifies the quoted SAS graphics device to use when generating graphics in ODS results. This requires the stored process SAS code to call %STPBEGIN. The default value is "GIF".

ODSStyle(*style name*) Specifies the quoted ODS style to apply to the results. This requires the stored process SAS code to call %STPBEGIN. There is no default value.

ODSStyleSheet(*path*) Specifies the quoted *path* to a CSS file on the client machine that is to be applied to generated ODS results. This requires the stored process SAS code to call %STPBEGIN. There is no default value.

NoAlerts(*Boolean*) If True, error messages are sent to the JMP log rather than message boxes. The default value is False.

---

stp<<Set Param Value(*name*, *value*)

Description

Sets the value of the specified stored process parameter to the specified value.

Returns

1 if successful, 0 otherwise (value can violate the parameter's constraints).

Arguments

name Specifies the quoted name of the parameter whose value to set.

value Specifies the quoted string that you want to set the parameter to.

---

stp<<Set Results Directory(*path*)

Description

Sets the quoted *path* on the client machine where stored process results are placed.

Returns

A quoted string.

Arguments

`path` Specifies the full quoted path of the directory where results of the stored process execution should be placed. The directory must exist or be creatable. If the results directory is not set, a temporary location appropriate for the operating system will be used, and that directory can be retrieved from the stored process Results scriptable object after the stored process executes.

SAS Results

`results<<Delete All Result Files()`

Description

Deletes all files created by the SAS Submit or Stored Process execution. Note that any result files that are still in use are not deleted.

Returns

1 if the deletion was successful; 0 if some of the files could not be deleted.

`results<<Get Directory()`

Description

Gets the directory where the results generated by the stored process or SAS submit are located.

Returns

A quoted string.

`results<<Get Log()`

Description

Get the SAS Log from the execution of the stored process or SAS submit.

Returns

A quoted string.

`results<<Get Main Result File Name(<Fullpath(Boolean)>)`

Description

Gets the full path of the main result file generated by the stored process or SAS submit.

Returns

A quoted string.

Optional Named Argument

`Fullpath(Boolean)` If true, the main result filename is returned as a full path. The default value is false.

`results<<Get Output()`

Description

Gets the SAS Listing output from the execution of the stored process or SAS submit.

Returns

A quoted string.

`results<<Get Output Datasets()`

Description

Get a list of output data set generated by the SAS Submit that created this SAS Results object.

Returns

A list of data set names in the form “libname.membername”.

`results<<Get Result File Info(<Mimetype(`*`mime-type`*`)>, <Fullpath(`*`Boolean`*`)>)`

Description

Get information about result files that were generated by the execution of the stored process or SAS submit.

Returns

List of two lists of quoted strings. The first list is filenames, and the second list is the MIME-type of the corresponding file from the first list.

Optional Arguments

`Mimetype(`*`mime_type`*`)` Restricts the set of files for which information is returned to only those files with the specified quoted MIME-type. If not specified, information about all generated files is returned.

`Fullpath(Boolean)` If true, the filename returned for each result file is returned as a full path; if false, only the name of the file is returned. The default value is false.

`results<<Make JMP Report()`

Description

Parses the ODS XML results and creates a JMP report.

Returns

The display box for the report.

results<<Open All Results()

Description

Opens all results generated by the execution of the stored process or SAS submit.

Returns

Void.

results<<Open Result File(*filename*, <Run Script(*Boolean*)>)

Description

Attempts to open the result file with the given name.

Returns

JMP Data Table if one was opened.

Required Argument

filename Specifies the quoted name of the file from the generated results to open. filename should just be the name of the file, not the full path. If *filename* is a filename with no extension, both JMP data tables and JSL scripts in the results are searched for a match, and if both exist, both are opened.

Optional Argument

Run Script(*Boolean*) If true, and if *filename* is a JSL script, the script is executed. If false, *filename* is just opened, even if it is a JSL script.

results<<Run Script(*filename*)

Description

Looks for the JSL file in the results with the given *filename* and runs it if it finds it.

Returns

Void.

Argument

"filename" Specifies the quoted name of the JSL file from the generated results to open. The *filename* argument should just be the name of the file, not the full path, and it does not need to include the .jsl extension.

Schedule

See also "Schedule(n, script)".

sch<<Clear Schedule()

Cancels all scheduled events.

sch<<Close()

Closes the scheduler.

sch<<Restart()

Restarts the scheduler after it was stopped from running all scheduled events.

sch<<Show Schedule()

Shows a list of all scheduled events.

sch<<Stop()

Stops the scheduler from running all scheduled events.

Segments

Pie Seg(<*style*>, {*x*, *y*}, *radius*, [*values*])

Description

Creates a pie seg at the specified origin, with the specified radius, based on given values.

Required Arguments

{*x*, *y*} Specifies the x and y coordinates at which the pie seg is displayed.

radius Specifies the radius.

values Specifies the values specified in matrix format.

Optional Argument

style A quoted string that specifies the style: "Pie" (traditional pie chart with each slice sized by the Summary Statistic), "Ring" (each variable or level of a stratifying variable is represented by a concentric ring), or "Coxcomb" (the central angles for all slices are equal).

Sockets

skt<<Accept(<*callback*, *timeout*>)

Description

Tells the server socket to accept a connection and return a new connected socket.

Returns

A list of up to four items. The first is a quoted string that echoes the command (`"accept"`). The second is a quoted string, either `"ok"` or an error. The third is a quoted string that specifies the name of the machine that just connected. The fourth is a reference to the socket that you can send more messages.

Optional Arguments

`callback` Specifies the name of a function to receive the data.

`timeout` If you use a *`callback`*, *`timeout`* specifies how long the function should wait for an answer. For a server socket, 0 is an acceptable value because a server should not shut down because no one has connected to it recently.

`skt<<bind(`*`localhost`*`, `*`port`*`)`

Description

Associates a port on the local machine with the socket.

Returns

A list of two quoted strings. The first string is the command name (“bind”) and the second is “ok” if successful or an error.

Required Arguments

`localhost` Specifies the quoted local machine. You cannot bind to another machine.

`port` Specifies the port that should be used.

`skt<<Close()`

Description

Closes a socket.

Returns

A list of two quoted strings. The first string is the command name (“close”) and the second is “ok” if successful.

`skt<<Connect(`*`socketname`*`, `*`port`*`)`

Description

Connects to a listening socket.

Returns

A list of two quoted strings. The first string is the command name (“connect”) and the second is “ok” for a successful connection or an error sent back by the other socket.

Arguments

`socketname` Specifies the name of the other socket. If you are connecting to a web server, this is the web address (the name is preferred to the IP address).

port Specifies the port of the other socket to connect through.

`skt<<GetPeerName()`

Description

Retrieves the address and port of the socket at the other end of the connection.

Returns

A list of four quoted strings. The first echoes the command (`"getpeername"`). The second is either "ok" or an error. The third and fourth are the address and the port.

`skt<<Get Sock Name()`

Description

Retrieves the address and port of the socket at this end of the connection.

Returns

A list of four quoted strings. The first echoes the command (`"getsockname"`). The second is either "ok" or an error. The third and fourth are the address and the port.

`skt<<ioctl(FIONBIO, `*`Boolean`*`)`

Description

Controls the socket's blocking behavior.

Returns

A list of two quoted strings. The first string is the command name ("ioctl") and the second is "ok" if successful or an error.

Arguments

`FIONBIO, 1 FIONBIO` means Non-Blocking I/O. If true, turns on the behavior and the argument.

`skt<<Listen()`

Description

Tells the server socket to listen for connections.

Returns

A list of two quoted strings. The first echoes the command (`"listen"`) and the second is `"ok"` or an error message.

`skt<<recv(n, <callback, timeout>)`

`skt<<recvfrom(n, <callback, timeout>)`

Description

Receives either a stream message (`recv`) or a datagram message (`recvfrom`) from the other socket. If the two optional arguments are used, the data is not received immediately. Instead, the data is received when the function *callback* is called.

Returns

A list of three quoted strings. The first string is the command name ("recv" or "recvto"). The second is "ok" if successful or an error message if not. The third string is the data that was received. If a callback function is used, a fourth element is the socket that was used in the original `recv` or `recvfrom` message.

Required Argument

n Specifies the number of bytes to receive from the other socket.

Optional Arguments

callback Specifies the name of a function to receive the data.

timeout If you use a *callback*, *timeout* specifies how long the function should wait for an answer.

`skt<<Send(stream)`

`skt<<SendTo(dgram)`

Description

Sends the data in the argument to the other socket. `Send` sends a stream and `sendto` sends a datagram.

Returns

A list of three quoted strings. The first string is the command name ("send" or "sendto"). The second is "ok" if successful or an error message if not. The third string is any portion of the stream that could not be sent, or empty if all the data was sent correctly.

Arguments

stream Specifies the command to send to the other socket.

dgram Specifies the command to send to the other socket.

Notes

Either argument might need to contain binary data. JMP represents non-printable ASCII characters with a tilde (~) followed by the hexadecimal number. For example,

```
skt<<send(("GET / HTTP/1.0~0d~0a~0d~0a");
```

sends a "get request" to an HTTP server.

SQL

obj<<Custom SQL(*sql*)

Description

Changes the query to a custom SQL query and sets the SQL.

Required Argument

`sql` The quoted SQL query.

obj<<Generate SQL

Returns the SQL that the query generates when you run it.

obj<<Modify

Opens the query in Query Builder.

obj<<PostQueryScript(*script as text*)

Sets a JSL script that runs after the query finishes executing. *script as text* is quoted JSL code.

obj<<Query Name(<*new name*>)

Gets (without the *new name* argument) or sets (with the *new name* argument) the name of the query. The name of the query is used as the name of the data table that results from running the query.

obj<<Run(<"Private"|"Invisible">, <Update Table(*table*)>, <OnRunComplete(*script*)>, <OnRunCanceled(*script*)>, <OnError(*script*)>)

Description

Runs the SQL query in the background or foreground depending on the Query Builder preference "Run queries in the background when possible".

Returns

Null (if the query runs in the background) or a data table (if the query runs in the foreground).

Optional Named Arguments

`"Private"` A quoted keyword that opens the data table that the query produces without displaying it in a data table window. `"Private"` is available only if `OnRunComplete` is included in the script.

"Invisible" A quoted keyword that hides the data table that the query produces. Use this argument to keep the query result hidden but use it in a subsequent query. The data table is displayed in the Home Window's Window List and the Window > Unhide list.

Update Table Updates the specified data table. Runs the query in the foreground.

OnRunComplete Specifies a script to run after the query is complete. To get the resulting data table, include OnRunComplete. The OnRunComplete script needs to be defined in the global namespace, as indicated by the double colons in this example:

```
Names Default To Here( 1 );
::onComplete = Function( {dt},
  {default local},
  Write(
     "\!NQuery is complete!  Result name: \!"",
     dt << Get Name,
     "\!", Number of rows: ",
     N Rows( dt )
  )
);

query = Include( "rentals_fam_romcom.jmpquery" );
query << Run Background( On Run Complete( ::onComplete ) );
```

OnRunCanceled Specifies a script to run after the user cancels the query.

OnError Specifies a script to run if an error occurs.

Notes

If you want the data table that results from the background query, use the OnRunComplete optional argument. You can include a script that runs when the query completes and then assigns a data table reference to the resulting data table. Or you might pass the name of a function that accepts a data table as its first argument. That function is called when the query completes.

Examples

The following example opens a query that you previously saved from Query Builder. The query opens privately, that is, without opening Query Builder. The query runs, and the resulting data table opens.

```
query = Open( "c:/My Data/Movies.jmpquery", "Private");
dt = query << Run();
```

You can include a .jmpquery file in a script and run the query in the background using the <<Run Background message.

```
query = Include( "C:/Queries/movies.jmpquery");
query <<Run Background();
```

The following example queries the database, opens the resulting data table, and prints the number of data table rows to the log.

```
confirmation = Function( {dtResult},
  Write( "\!NNumber of rows in query result: ", N Rows( dtResult ) )
);
query = New SQL Query(
  Connection(
     "ODBC:DSN=SQL
  Databases;APP=MYAPP;TrustedConnection=yes;WSID=D79255;DATABASE=SQB;"
  ),
  QueryName( "movies_to_update" ),
  Select( Column( "YearMade", "t1" ), Column( "Rating", "t1" ) ),
  From( Table( "g6_Movies", Schema( "SQB" ), Alias( "t1" ) ) ),

);
query << Run( OnRunComplete( confirmation ) );
```

`Run Background(<OnRunComplete(`*script*`), <"Private"|"Invisible">>, <OnRunCanceled(`*script*`)>, <OnError(`*script*`)>)`

Description

Runs the SQL query in the background. The running query is not displayed.

Returns

Null (or the data table object, if `OnRunComplete` is included).

Optional Named Arguments

`OnRunComplete(`*script*`)` Specifies a script to run after the query is complete. To get the resulting data table, include `OnRunComplete`. The `OnRunComplete` script needs to be defined in the global namespace, as indicated by the double colons in this example:

```
Names Default To Here( 1 );
::onComplete = Function( {dt},
  {default local},
  Write(
     "\!NQuery is complete!  Result name: \!"",
     dt << Get Name,
     "\!", Number of rows: ",
     N Rows( dt )
  )
);

query = Include( "rentals_fam_romcom.jmpquery" );
query << Run Background( On Run Complete( ::onComplete ) );
```

`"Private"` Does not open the resulting data table. Specify only with `OnRunComplete`. If you include `private` in a background query, JMP opens the data table as invisible instead.

`"Invisible"` Hides the data table. Use this argument to keep the query result hidden but use it in a subsequent query. The data table is displayed in the Home Window's Window List and the Window > Unhide list.

`OnRunCanceled` Specifies a script to run after the user cancels the query.

`OnError` Specifies a script to run if an error occurs.

Notes

All queries except for SAS queries run in the background based on the Query Builder preference "Run the queries in the background when possible", which is selected by default. For SAS queries, `Run Background()` is ignored.

You can include a .jmpquery file in a script and run the query in the background using the `Run Background` message.

```
query = Include( "C:/Queries/movies.jmpquery");
query <<Run Background();
```

`Run Foreground(<OnRunComplete(`*`script`*`), <"Private"|"Invisible">>, <OnRunCanceled(`*`script`*`)>, <OnError(`*`script`*`)>)`

Description

Runs the SQL query in the foreground.

Returns

A data table that opens when the query is finished.

See Also

"Run Background(<OnRunComplete(`script`), <"Private"|"Invisible">>, <OnRunCanceled(`script`)>, <OnError(`script`)>)"

`obj<<Save`

Saves the query to its associated file. The save fails if the query does not yet have an associated file.

`obj<<Save As(`*`path`*`, <Replace Existing(`*`Boolean`*`))`

Saves the query to the specified file. If the file already exists, the save fails unless `Replace Existing` is true.

Other Objects

Zip Archives

`list = za<<Dir`

Returns a list of member names

`data = za<<Read(`*`member name`*`, <Format("blob")>)`

Returns a quoted string that contains the entire quoted *`member name`*. A zip file consists of filenames, also called "member names".

Notes

For remote files, JMP copies the URL data to the local disk. When the zip archive is no longer accessible, the local data file is deleted.

`actualname = za<<Write(`*`member name`*`, `*`member data`*`, <"replace">)`

Writes a quoted string or quoted blob to a zip archive member file. If the quoted *`member name`* isn't in the current zip file, the returned `actualname` is the same as *`member name`*. This member name will be changed to prevent overwriting an existing member; the name actually used is returned. The quoted *`member data`* argument is the data to write into the zip file's member of that name. *`replace`* creates the file with a temporary name, deletes the old file, and renames the temporary file to the existing name.

Journals

`jnl<<Save HTML(<`*`path`*`>, <`*`format`*`>)`

Saves the journal as HTML.

Optional Arguments

`"path"` Specifies the quoted path for the saved HTML file (for example, `"c:/myFile.html"`).

`format` Specifies the quoted graphic file format. JPG, PNG, and TIFF formats are supported. The graphics are saved in a subdirectory named `gfx`.

`jnl<<Save RTF(<path>, <format>)`

Saves the journal as an RTF file.

Optional Arguments

`"path"` Specifies the quoted path for the saved RTF file (for example, `"c:/myFile.rtf"`).

"format" Specifies the quoted file format for the embedded graphics. "JPG", "PNG", and "EMF" formats are supported on Windows. All journals are saved as PDF files by default on macOS.

Notes

If no *path* or *format* are provided, you are prompted to name the file and specify the format on Windows. On macOS, you are prompted to name the file. The file is saved as a PDF file by default.

---

jnl<<Save PDF(<*path*>, <Show Page Setup(*Boolean*)>, <Portrait(*Boolean*)>)

Saves the journal as a PDF file.

Optional Arguments

"path" The quoted *path* for the saved PDF file (for example, "c:/myFile.pdf").

Show Page Setup If set to true, opens the Page Setup window to let the user change the margin, magnification level, and other page layout options.

Portrait Determines whether the page orientation is portrait or landscape. Overrides the user's selection in the Show Page Setup window.

Appendix A

SQL Functions Available for JMP Queries

The `Query()` JSL function performs a SQL query on selected tables and exports the data to a data table. The following example first assigns the t1 alias to Big Class.jmp. name, age greater than 13, and height are then selected from the t1 table.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Query( Table( dt, "t1" ),
       "SELECT t1.name, t1.age, t1.height FROM t1
       WHERE t1.age > 13" );
```

You can use SQL functions in a query. For example, `SELECT CURRENT_TIMESTAMP` returns the current UTC/GMT time stamp as a SQLite time string:

```
Query( Scalar, "SELECT CURRENT_TIMESTAMP;" );
```

This appendix lists the numeric, date-time, string, system SQL, and aggregate functions that you can use in SQL queries. “Yes” in the SQLite column indicates native SQLite functions. See the Online SQLite documentation at https://www.sqlite.org/lang.html.

Contents

Numeric SQL Functions 531
Date-Time SQL Functions 532
String SQL Functions 535
System SQL Functions 536
Aggregate SQL Functions 536

Numeric SQL Functions

The numeric SQL functions are described here.

Numeric Function	Native SQLite	Description
ABS(*number*)		Returns the absolute value of the specified number.
ACOS(*cosine*)		Returns the angle in radians for the specified cosine.
ASIN(*sin*)		Returns the angle in radians for the specified sine.
ATAN(*tangent*)		Returns the angle in radians for the specified tangent.
ATAN2(*x*, *y*)		Two-argument arctangent function.
CEILING(*number*) CEIL(*number*)		Returns the smallest integer larger than the specified number.
COS(*radians*)		Returns the cosine of the specified angle in radians.
COT(*radians*)		Returns the cotangent of the specified angle in radians.
DEGREES(*radians*)		Converts an angle in radians to an angle in degrees.
EXP(*number*)		Returns the constant *e* raised to the specified power.
FLOOR(*number*)		Returns the largest integer smaller than the specified number.
LN(*number*) LOG(*number*)		Returns the natural logarithm of the specified number.
LOG10(*number*)		Returns the common logarithm of the specified number.
MAX(*n1*, *n2*, ...)	Yes	Returns the largest of the specified numbers. A minimum of two numbers must be specified.
MIN(*n1*, *n2*, ...)	Yes	Returns the smallest of the specified numbers. A minimum of two numbers must be specified.
MOD(*dividend*, *divisor*)		Returns the remainder when *dividend* is divided by *divisor*. Floating-point values are truncated to integers before the modulus operation is performed.
PI()		Returns the value of the constant pi (π).
POWER(*number*, *power*) POW(*number*, *power*)		Raises *number* to the specified *power*.

Numeric Function	Native SQLite	Description
RADIANS(*degrees*)		Converts an angle in degrees to an angle in radians.
RANDOM() RANDOM(*max*) RANDOM(*min*, *max*)		Returns a random number. RANDOM() returns a number between 0 and 1. RANDOM(max) returns a number between 0 and *max*. RANDOM(min, max) returns a number between *min* and *max*. This function is equivalent to the Random Uniform() JSL function, and its seed can be controlled using the Random Reset() JSL function. RANDOM can be shortened to RAND.
RANDOMBLOB(*length*)	Yes	Returns an N-byte blob that contains pseudo-random bytes. See the SQLite Online documentation at https://www.sqlite.org/lang.html.
ROUND(*number*, *<precision>*)		Rounds *number* to the number of decimal places given by *precision*. The default value of *precision* is 0, and *precision* can be negative.
SIGN(*number*)		Returns 1 if *number* is positive, -1 if *number* is negative, or 0 if *number* is zero.
SIN(*radians*)		Returns the sin of the specified angle in radians.
SQRT(*number*)		Returns the square root of *number*.
TAN(*radians*)		Returns the tangent of the specified angle in radians.
TRUNCATE(*number*, *<precision>*)		Truncates *number* at the number of decimal places given by *precision*. The default value of *precision* is 0, and *precision* can be negative. TRUNCATE() can be shortened to TRUNC().

Date-Time SQL Functions

Using date-time functions in JMP queries is complicated by the fact that the SQL engine that handles JMP queries (SQLite) uses different formats for storing dates than JMP does. SQLite stores date-times as strings. However, JMP stores date-times as the number of seconds since January 1, 1904. When you have columns in your table that contain date-times, the conversions are handled automatically. However, when you use functions that return date-times, you might need to let JMP know when a conversion is required.

Consider the CURRENT_TIMESTAMP function. CURRENT_TIMESTAMP is a built-in SQLite function that returns the current UTC/GMT time stamp as a SQLite time string:

```
Query( Scalar, "SELECT CURRENT_TIMESTAMP;" );
```

returns:

```
"2016-02-16 15:44:42"
```

The string could perhaps be parsed as a date to return it as a JMP date. To prevent the need to do so, wrap the CURRENT_TIMESTAMP function in the JMPDATE() function:

```
Query( Scalar, "SELECT JMPDATE( CURRENT_TIMESTAMP );" );
```

returns:

```
3538482531
```

The string is an unformatted JMP date. However, if you pass a SQLite time string to another SQL date-time function, you do not need to use JMPDate(); the value will be converted to a JMP date automatically. Here is an example:

```
Query( Scalar, "SELECT EXTRACT('YEAR', CURRENT_TIMESTAMP);" );
```

Using native SQLite date-time functions (date(), time(), datetime(), julianday(), strftime()) in JMP queries is not recommended because JMP date-time values are not compatible with those functions.

Date-Time Function	Naive SQLite	Description
CURRENT_DATE	Yes	Returns the current date (UTC/GMT) as a SQLite time string.
CURRENT_TIME	Yes	Returns the current time (UTC/GMT) as a SQLite time string.
CURRENT_TIMESTAMP	Yes	Returns the current date and time (UTC/GMT) as a SQLite time string.
DATEDIFF(*date1*, *date2*, *interval*, <*alignment = "Start"*>)		Computes the difference between two dates in units specified by *interval*, based on *alignment*. This function works the same as the Date Difference() JSL function. Valid values for *interval* are: "Year", "Quarter", "Month", "Week", "Day", "Hour", "Minute" and "Second". Valid values for *alignment* are "Start", "Actual" and "Fractional". If *alignment* is not specified, "Start" is used.

Date-Time Function	Naive SQLite	Description	
EXTRACT(*datepart*, *datetime*, *<use_locale = 1>*)		Extracts a specific part of a date or date-time value. *Datetime* is a JMP date-time value or a SQLite time string. *Use_locale* is optional and applies only to date name parts such as "MonthName" and "DayName" and determines whether values from the current language or English are returned. The following values of *datepart* are supported:	
		"Year"	Returns the year as a number.
		"Month"	Returns the numeric month (1-12).
		"MonthName"	Returns the full name of the month in the current language (*use_locale = 1*) or English (*use_locale = 0*).
		"Mon", "MMM"	Returns the abbreviated name of the month.
		"Day"	Returns the day of the month (1-31).
		"DayName"	Returns the name of the day of the week.
		"DayOfWeek"	Returns the numeric day of the week (1-7).
		"DayOfYear"	Returns the numeric day of the year (1-366).
		"Quarter"	Returns the numeric quarter (1-4).
		"Hour"	Returns the hour (0-23).
		"Minute"	Returns the minute (0-59).
		"Second"	Returns the seconds, including any fractional part.
		"Date"	Returns just the date portion of a date-time value as a JMP date-time value.
		"Time"	Returns just the time portion of a date-time value as a JMP date-time value.
JMPDATE(*SQLite time string*)		Converts a SQLite time string to the equivalent JMP date-time value.	
NOW()		A synonym for TODAY().	
TODAY()		Returns the JMP date-time value of the current moment in *local* time, which matches the JMP Today() function.	

String SQL Functions

The string SQL functions are described here
.

Function	Native SQLite	Description
HEX(*binary*)	Yes	SQLite built-in function that converts a BLOB to a string of hexadecimal characters. Useful when paired with the RANDOMBLOB() function.
JLEFT(*string*, *len*, *<pad>*)		Like the JSL Left() function. Returns *len* characters from the beginning of *string*. If *pad* is specified and fewer than *len* characters are present in *string*, the result is padded with *pad* out to length *len*.
JRIGHT(*string*, *len*, *<pad>*)		Like the JSL Right() function. Returns *len* characters from the end of *string*. If *pad* is specified and fewer than *len* characters are present in *string*, the result is padded with *pad* at the front out to length *len*.
LENGTH(*string*)	Yes	SQLite equivalent of the ANSI standard CHAR_LENGTH() function. Returns the length of its string argument in characters.
LOCATE(*string1*, *string2*) POSITION(*string1*, *string2*)		Returns the (1-based) starting position of *string1* within *string2*, returning 0 if *string1* is not found within *string2*.
LOWER(*string*)		Returns a copy of *string* with all uppercase characters converted to lowercase.
LTRIM(*string*, *<trimchars>*)	Yes	Trims any characters contained in *trimchars* from the beginning of *string* and returns the result. If *trimchars* is omitted, spaces are trimmed.
PRINTF(*format*, *<arg1, ..., argN>*)	Yes	Allows constructing strings using placeholders and arguments. See the SQLite Online documentation at https://www.sqlite.org/lang.html.
REPLACE(*string*, *find*, *replace*)	Yes	Replaces all instances of *find* in *string* with *replace* and returns the result. If *replace* is numeric, it is converted to a string.
REVERSE(*string*)		Returns a copy of *string* with the order of the characters reversed.

Function	Native SQLite	Description
RTRIM(*string*, *<trimchars>*)	Yes	Trims any characters contained in *trimchars* from the end of *string* and returns the result. If *trimchars* is omitted, spaces are trimmed.
SPACE(*length*)		Returns a string consisting of *length* space characters.
SUBSTR(*string*, *start*, *<length>*)	Yes	Returns the substring of *string* starting at *start* (1-based) that is *length* characters long. If *length* is omitted, the substring starting at *start* and continuing to the end of *string* is returned.
TRIM(*string*, *<trimchars>*)		Trims any characters contained in *trimchars* from the end of *string* and returns the result. If *trimchars* is omitted, spaces are trimmed.
UPPER(*string*)		Returns a copy of *string* with all lowercase characters converted to uppercase.

System SQL Functions

The system SQL functions are described here.

Function	SQLite	Description
COALESCE(*arg1*,..., *argN*)	Yes	Returns the first argument passed to it that is non-NULL. Returns NULL if all arguments are NULL. Requires at least two arguments.
IFNULL(*arg1*, *arg2*)	Yes	Returns *arg1* if not NULL, otherwise *arg2*. Basically, IFNULL is a two-argument version of COALESCE().
NULLIF(*arg1*, *arg2*)	Yes	Returns *arg1* if *arg1* and *arg2* are different and returns NULL if the arguments are equal. Used when you have non-NULL values in your database that you want to treat as NULL.

Aggregate SQL Functions

When passing a single argument to an aggregate function, that argument can be preceded by the keyword DISTINCT, which filters out duplicate values.

For all aggregations other than COUNT(*), NULL and missing values are ignored.

Function	SQLite	Description
AVG(*num_expr*)		Computes the average of *num_expr* for the rows in the group. *Num_expr* must be numeric.
COUNT(*expr*) COUNT(*)		Counts the number of times *expr* is not NULL in the group. COUNT(*) returns the total number of rows in the group.
GROUP_CONCAT(*expr*, *<separator = ','>*)	Yes	Concatenates all non-NULL values of *expr* and returns them as a string. Numeric values of *expr* are converted to character. If *separator* is present, it is placed between the values. The default separator is a comma. DISTINCT can be used only with GROUP_CONCAT() if *separator* is not specified.
MAX(*expr*)		Returns the maximum value of *expr* in the group. *Expr* can be character or numeric.
MIN(*expr*)		Returns the minimum value of *expr* in the group. *Expr* can be character or numeric.
STDDEV_POP(*num_expr*)		Computes the population standard deviation of *num_expr* for the group.
STDDEV_SAMP(*num_expr*)		Computes the sample standard deviation of all *num_expr* for the group.
SUM(*num_expr*)		Returns the sum of *num_expr* for the group. If no non-NULL values are found, SUM() returns NULL.
TOTAL(*num_expr*)	Yes	Same as SUM(*num_expr*), except TOTAL() returns 0.0 if no non-NULL values are found.
VAR_POP(*num_expr*)		Computes the population variance of *num_expr* for the group.
VAR_SAMP(*num_expr*)		Computes the sample variance of *num_expr* for the group.

Appendix B

References

Gander, W., and Gautschi, W. (2000). "Adaptive Quadrature–Revisited." *BIT Numerical Mathematics* 40:84–101.

Moore, W. A., and Parks, D. R. (2012). "Update for the logicle data scale including operational code implementations." *Cytometry Part A* 81A:273–277. Accessed January 26, 2021. https://onlinelibrary.wiley.com/doi/full/10.1002/cyto.a.22030.

Appendix C

Technology License Notices

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved. Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth. Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr). All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org). All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents. See www.mathworks.com/patents. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.
  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard. All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California. All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group. NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois. All rights reserved. DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS "AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated. All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives. The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2021, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file. All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Version 16

Scripting Guide

"The real voyage of discovery consists not in seeking new landscapes, but in having new eyes."

Marcel Proust

JMP, A Business Unit of SAS
SAS Campus Drive
Cary, NC 27513

16.1

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2020–2021. *JMP® 16 Scripting Guide*. Cary, NC: SAS Institute Inc.

JMP® 16 Scripting Guide

Copyright © 2020–2021, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

March 2021

August 2021

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement. For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

Get the Most from JMP

Whether you are a first-time or a long-time user, there is always something to learn about JMP.

Visit JMP.com to find the following:

- live and recorded webcasts about how to get started with JMP
- video demos and webcasts of new features and advanced techniques
- details on registering for JMP training
- schedules for seminars being held in your area
- success stories showing how others use JMP
- the JMP user community, resources for users including examples of add-ins and scripts, a forum, blogs, conference information, and so on

https://www.jmp.com/getstarted

Contents

Scripting Guide

1 Learn about JMP 23

Documentation and Additional Resources

Formatting Conventions in JMP Documentation 25
JMP Help 26
JMP Documentation Library 26
Additional Resources for Learning JMP 32
JMP Tutorials 32
Sample Data Tables 32
Learn about Statistical and JSL Terms 33
Learn JMP Tips and Tricks 33
JMP Tooltips 33
JMP User Community 34
Free Online Statistical Thinking Course 34
JMP New User Welcome Kit 34
Statistics Knowledge Portal 34
JMP Training 34
JMP Books by Users 35
The JMP Starter Window 35
JMP Technical Support 35

2 Introduction 37

Welcome to the JMP Scripting Language

What JSL Can Do for You 39
Help with Learning JSL 39
The Scripting Guide 39
The Scripting Index 40
Let JMP Teach You JSL 42
Terminology 43
Basic JSL Syntax 46
Book Conventions 47

3 Getting Started 49
Let JMP Write Your Scripts

Capturing a Script for an Analysis Report 51
Capturing a Script for a Data Table 52
Capturing a Script to Import a File 53
Gluing Scripts Together 54

4 Scripting Tools 59
Using the Script Editor, Log Window, Debugger and Profiler

Work with the Script Editor 61
Introduction to the Script Editor 61
Run a Script 62
Stop a Script 63
Edit a Script 63
Color Coding and Themes 63
Autocomplete Functions 64
Tooltips 64
Split a Window 65
Match Parentheses, Brackets, and Braces 66
Select a Rectangular Block of Text 67
Select Fragmented Text 68
Drag and Drop Text 68
Find and Replace 68
Automatic Formatting 69
Errors Are Marked by Special Characters 69
Typing in Multiple Rows at Once 69
Add Code Folding Markers 70
Advanced Options 71
Set Preferences for the Script Editor 72
Work with the Log 72
Enhanced Log in JMP 73
Text Log in JMP 78
Debug or Profile Scripts 79
Debugger and Profiler Window 80
Work with Breakpoints 84
View Variables 87
Work with Watches 87
Modify Preferences in Debugger 88
Persistent Debugger Sessions 89
Examples of Debugging and Profiling Scripts 89

5 JSL Building Blocks 97

Learning the Basics of JSL

JSL Syntax Rules 100
- Value Separators 100
- Numbers 103
- Names 103
- Comments 104

Operators 105
Boolean Values 109
Global and Local Variables 110
- Local Namespaces 110
- Named Namespaces 111
- Show Symbols, Clear Symbols, and Delete Symbols 111
- Lock and Unlock Symbols 112
- Hide a Global Variable 113

Rules for Name Resolution 113
- Variable Names Used as Arguments 114
- Rules for Resolving Names 114
- Scoping Operators 116
- Troubleshoot Variables and Column Names 119
- Troubleshoot Variables and Keywords 120
- Frequently Asked Questions about Name Resolution 120

Alternatives for Gluing Expressions Together 121
Iterate 122
- For 122
- While 123
- Summation 125
- Product 125
- Break and Continue 126

Conditional Functions 128
- If 128
- Match 130
- Choose 132
- Interpolate 133
- Step 135

Compare Incomplete or Mismatched Data 135
- Missing Values 137

Inquiry Functions 138
- General Element Types 139
- Specific Element Types 139

Object Attributes ... 141
Host Information ... 141
Version Information ... 141

6 Types of Data ... 143
Working with Numbers, Strings, Dates, Currency, and More

Numbers and Strings ... 145
Formatting Numbers ... 146
Unicode Characters ... 147
Path Variables ... 147
Create and Customize Path Variables ... 150
Relative Paths ... 151
File Path Separators ... 151
Date-Time Functions and Formats ... 152
Date-Time Values ... 152
Program with Date-Time Functions ... 153
Date-Time Values in Data Tables ... 160
Custom Date-Time Formatting with Format Patterns ... 164
Currency ... 166
Hexadecimal and BLOB Functions ... 167
Work with Character Functions ... 170
Concat ... 170
Munger ... 171
Repeat ... 172
Substitute and Substitute Into ... 173
Regular Expressions ... 174
Regex ... 174
Regex Match ... 176
Special Characters in Regular Expressions ... 178
Escaped Characters in Regular Expressions ... 179
Greedy and Reluctant Regular Expressions ... 181
Backreferences and Capturing Groups ... 183
Lookaround Assertions ... 184
Pattern Matching ... 185
Parsing Strings in Fixed Fields ... 186
Patterns and Case ... 188
Troubleshooting Patterns ... 188

7 Data Structures ... 191
Working with Collections of Data

Lists ... 193

Evaluate Lists ... 193
Assignments with Lists ... 195
Perform Operations in Lists ... 195
Find the Number of Items in a List ... 195
Subscripts ... 195
Locate Items in a List ... 197
List Operators and Functions ... 198
Iterate through a List ... 201
Concatenate Lists ... 202
Insert a Nested List in an Existing List ... 203
Index into a List with Another List ... 203
Send Messages to Lists ... 203
Matrices ... 204
Construct Matrices ... 204
Subscripts ... 206
Inquiry Functions ... 210
Comparisons, Range Checks, and Logical Operators ... 210
Numeric Operations ... 211
Concatenation ... 214
Transpose ... 214
Indexing into a Matrix or List with Another Matrix or List ... 215
Matrices and Data Tables ... 216
Matrices and Reports ... 218
Loc Functions ... 219
Ranking and Sorting ... 221
Special Matrices ... 221
Inverse Matrices and Linear Systems ... 226
Decompositions and Normalizations ... 230
Build Your Own Matrix Functions ... 235
Statistical Examples ... 236
Associative Arrays ... 240
Create Associative Arrays ... 240
Work with Associative Arrays ... 243
Applications for Associative Arrays ... 246
Associative Arrays in Graph Theory ... 248
Associative Arrays in Set Operations ... 251

8 Programming Methods ... 253
Complex Scripting Techniques and Additional Functions

Lists and Expressions ... 255

Expressions 255
Macros 267
Manipulating Lists 267
Manipulating Expressions 270
Advanced Scoping and Namespaces 274
Names Default To Here 274
Scoped Names 278
Namespaces 282
Referencing Namespaces and Scopes 286
Resolving Named Variable References 289
Best Practices for Advanced Scripting 290
Advanced Programming Concepts 291
Throw and Catch Exceptions 291
Functions 293
Recursion 296
Includes 296
Load and Save Text Files 297
Run External Programs 297
Classes 299
File and Directory Operations 302
Select a Directory or File 302
Get a List of Filenames 303
Connect to a Network Directory 304
Scripting By Groups 305
Encrypt and Decrypt Scripts 306
Encryption and Global Variables 308
Encrypting Scripts in Data Tables 308
Working with Expressions in an Encrypted Script 309
Additional Numeric Functions 310
Derivatives 310
Algebraic Manipulations 312
Maximize and Minimize 314
Identifying Differences Between Strings, Lines, or Sequences 315
Scheduling Actions 317
Functions that Communicate with Users 318
Writing to the Log 319
Send Information to the User 320

9 Data Tables ... 323

Working with Data Table Objects

Get Started with Data Tables ... 327
Basic Data Table Scripting ... 330
- Open a Data Table ... 330
- Create a New Data Table ... 332
- Import Data ... 333
- Set the Current Data Table ... 357
- Name a Data Table ... 358
- Save a Data Table ... 358
- Hide a Data Table ... 359
- Print a Data Table ... 360
- Resize a Data Table ... 360
- Close a Data Table ... 360
- Set and Get a Data Table ... 361
- Perform Actions on All Open Data Tables ... 361
- Create Journals ... 362
- Create Excel Workbooks ... 363
- Prevent Changes to a Data Table ... 364

Advanced Data Table Scripting ... 365
- Select Columns in a Data Table ... 365
- Store Summary Statistics in Global Variables ... 366
- Create a Table of Summary Statistics ... 369
- Subset a Data Table ... 371
- Add a Data Filter ... 372
- Sort a Data Table ... 379
- Stack Values in a Data Table ... 379
- Split Values in a Stacked Data Table ... 380
- Transpose a Data Table ... 382
- Vertically Concatenate Data Tables ... 383
- Horizontally Concatenate Data Tables ... 383
- Virtually Join Data Tables ... 384
- Replace Data in Data Tables ... 387
- Create a Table Using Tabulate ... 389
- Copy Data Tables ... 390
- Find Missing Data Patterns ... 390
- Compare Data Tables ... 391
- Create a Summary Table ... 391
- Subscribe to a Data Table ... 392
- Move Data Between Matrices and Data Tables ... 394

Add Hyperlinks to Text Columns . . . 394
Columns . . . 396
Send Messages to Data Column Objects . . . 397
Create Columns . . . 398
Create a New Column by Text Matching . . . 400
Add Several Columns at Once . . . 401
Group Columns . . . 402
Use Set Selected to Select Columns . . . 404
Rearrange and Move Columns . . . 405
Recode Columns . . . 406
Add a Column Switcher . . . 407
Compress Selected Columns . . . 408
Create Custom Functions, Transforms, and Formats . . . 408
Custom Transform Example . . . 411
Delete Columns . . . 415
Get Column Names . . . 415
Column Attributes . . . 416
Column Properties . . . 423
Specify Engineering SI Units . . . 430
Rows . . . 431
Add Rows . . . 431
Delete Rows . . . 433
Select Rows . . . 433
Find Rows . . . 437
Move Rows . . . 437
Assign Colors and Markers to Rows . . . 438
Color Cells . . . 438
Hide, Exclude, and Label Rows . . . 439
Iterate on Rows in a Table . . . 440
Prevent Editing Rows . . . 443
Row States . . . 444
Accessing Data Values . . . 460
Set or Get Values by Column Name . . . 461
Additional Ways to Access Data Values . . . 462
Add Metadata to a Data Table . . . 463
Data Table Variables . . . 463
Table Scripts . . . 463
Formulas . . . 468
Delete Metadata . . . 469
Calculations . . . 469

Pre-Evaluated Statistics . . . 469
Calculator Formulas . . . 471

10 Scripting Platforms . . . 473
Create, Repeat, and Modify Analyses

Example of Scripting a Platform . . . 477
Send Messages to a Platform . . . 480
Conventions for Messages and Arguments . . . 480
Send Multiple Messages . . . 481
Find Messages for Objects . . . 482
Interpret the Show Properties List . . . 483
Send Messages to an Object . . . 484
Specify Which Columns to Analyze . . . 485
Create Column References . . . 485
Specify Multiple Column Names at Once . . . 486
Enable Users to Specify Columns . . . 486
Specify a By Variable . . . 486
Filter by Value using a Where Statement . . . 490
Enable User Input . . . 490
Execute Embedded Red Triangle Options . . . 491
Make Platforms Invisible . . . 491
Specify Report Titles . . . 493
Common Messages for Platform Windows . . . 493
Scripting Considerations by Platform . . . 494
Categorical . . . 494
Control Charts . . . 495
Distribution . . . 498
DOE . . . 499
Fit Model . . . 501
Formula Depot . . . 505
Graph Builder . . . 506
Neural and Neural Net . . . 508
Partial Least Squares and PLS . . . 508
Process Capability . . . 508
Scatterplot 3D . . . 509
Text Explorer . . . 510
Explore Outliers . . . 510
Scripting-Only Messages and Arguments by Platform . . . 511
Bivariate Script-Only Messages . . . 513
Boosted Tree Script-Only Messages . . . 514

Bootstrap Forest Script-Only Messages . . . 515
Bubble Plot Script-Only Messages . . . 516
Choice Script-Only Messages . . . 516
Clustering Script-Only Messages . . . 517
Contingency Script-Only Messages . . . 517
Contour Plot Script-Only Messages . . . 517
Contour Profiler Script-Only Messages . . . 517
Control Chart Script-Only Messages . . . 518
Control Chart Builder Script-Only Messages . . . 518
Cumulative Damage Script-Only Messages . . . 518
Custom Profiler Script-Only Messages . . . 519
Degradation Script-Only Messages . . . 519
Destructive Degradation Script-Only Messages . . . 519
Discriminant Script-Only Messages . . . 519
Distribution Script-Only Messages . . . 520
EMP Measurement Systems Analysis Script-Only Messages . . . 520
Explore Missing Values Script-Only Messages . . . 521
Explore Outliers Script-Only Messages . . . 521
Factor Analysis Script-Only Messages . . . 521
Fit Life by X Script-Only Messages . . . 521
Fit Model Script-Only Messages . . . 522
Fit Parametric Survival Script-Only Messages . . . 524
Functional Data Explorer Script-Only Messages . . . 525
Gaussian Process Script-Only Messages . . . 525
K Means Cluster Script-Only Messages . . . 525
K Nearest Neighbors Script-Only Messages . . . 525
Latent Class Analysis Script-Only Messages . . . 525
Life Distribution Script-Only Messages . . . 526
Make Validation Column Script-Only Messages . . . 527
MaxDiff Script-Only Messages . . . 527
Mixture Profile Script-Only Messages . . . 527
Model Screening Script-Only Messages . . . 528
Multivariate Script-Only Messages . . . 528
Naive Bayes Script-Only Messages . . . 528
Neural Script-Only Messages . . . 528
Nonlinear Script-Only Messages . . . 529
Normal Mixtures Script-Only Messages . . . 530
Oneway Script-Only Messages . . . 530
Parallel Plot Script-Only Messages . . . 530
Pareto Plot Script-Only Messages . . . 531

Partial Least Squares Script-Only Messages 531
Partition Script-Only Messages 531
Principal Components Script-Only Messages 532
Process Capability Script-Only Messages 532
Process Screening Script-Only Messages 532
Profiler Script-Only Messages 533
Reliability Forecast Script-Only Messages 533
Reliability Growth Script-Only Messages 533
Response Screening Script-Only Messages 534
Scatterplot 3D Script-Only Messages 534
Structural Equation Models Script-Only Messages 535
Support Vector Machines Script-Only Messages 536
Surface Plot Script-Only Messages 537
Tabulate Script-Only Messages 540
Ternary Plot Script-Only Messages 540
Text Explorer Script-Only Messages 540
Time Series Script-Only Messages 541
Uplift Script-Only Messages 541
Variability Chart Script-Only Messages 542

11 Display Trees 543
Create and Interact with Windows

Navigate JMP Reports 545
Examples of Common Display Boxes 546
View the Display Tree 547
Show Display Box Properties 549
Display Box Object References 550
Send Messages to Display Boxes 555
Example of Creating a Report 562
Construct Custom Windows 565
Example of Creating a Graph Box 565
Extract Values from a Window 566
Constructors for New Windows 568
Interact with a Closing Window 608
Update an Existing Display 610
Set Function and Set Script 614
Get and Set Selected Values of Display Elements That Return Lists 616
Send Messages to Constructed Displays 618
Construct Display Boxes That Contain Platforms 619
Counting the Number of Child Boxes 621

Examples of Creating a Dashboard from Two Reports . . . 622
Example of Creating a Cluster Platform Launch Window . . . 628
Example of Constructing a Custom Platform . . . 631
Example of Specifying Text and Background Colors . . . 634
Modal Windows . . . 635
Construct a Modal Window . . . 635
Construct a Column Dialog . . . 637
Comparison of Column Dialog and New Window . . . 639
Constructors for Column Dialogs . . . 640
Convert Deprecated Dialog to New Window . . . 643
New Window Example . . . 643
Example of Deprecated Dialog . . . 644
Differences Between New Window and the Deprecated Dialog . . . 646
Optional Scripts in New Window . . . 653
Scripting the Script Editor . . . 654
Technical Details . . . 655
Write Tab Box and Tab Page Box Scripts . . . 655

12 Scripting Graphs . . . 657
Editing and Creating Two-Dimensional Graphs

Add Scripts to Graphs . . . 659
Specify the Order of Graphical Elements . . . 660
Create New Graphs from Scratch and Customize Them . . . 666
Customize Graphs . . . 667
Create Shapes for a Bubble Plot . . . 669
How to Find Shapes . . . 672
Add Lines, Arrows, Points, Shapes, and Text . . . 673
Draw Lines . . . 673
Draw Arrows . . . 675
Draw Markers . . . 677
Draw Pies and Arcs . . . 678
Draw Regular Shapes: Circles, Rectangles, and Ovals . . . 680
Draw Irregular Shapes: Polygons and Contours . . . 682
Add Text . . . 685
Specify Colors . . . 686
Specify Transparency . . . 690
Add Fill Patterns . . . 691
Specify Line Types . . . 694
Draw with Pixels . . . 695
Interactive Graphs . . . 696

Handle() ... 696
Mousetrap() ... 701
Drag Functions ... 703
Troubleshoot Interactive Graphs ... 706
Create a Color Picker ... 706
Clip Display Segs in Graphics ... 707
Create Background Maps ... 709
Graph Elements ... 710
Plot Functions ... 711
Get the Properties of a Graphics Frame ... 717
Add a Legend ... 717
Hover Labels ... 718
Add Rich Text to Hover Labels Using Textlets ... 719
Add Graphs or Images to Hover Labels Using Graphlets ... 722
Update Names and Values in Hover Labels Using Gridlets ... 732
Create Drill-Down Graphs ... 737
Work with the Hover Label Execution Context ... 740

13 Three-Dimensional Scenes ... 749
Scripting in Three Dimensions

About JSL 3-D Scenes ... 751
JSL 3-D Scene Boxes ... 751
Setting the Viewing Space ... 754
Setting Up a Perspective Scene ... 755
Setting up an Orthographic Scene ... 756
Changing the View ... 757
The Translate Command ... 757
The Rotate Command ... 757
The Look At Command ... 759
The ArcBall ... 760
Graphics Primitives ... 761
Primitives Example ... 764
Controlling the Appearance of Primitives ... 765
Other Uses of Begin and End ... 771
Drawing Spheres, Cylinders, and Disks ... 771
Construction ... 771
Lighting ... 772
Drawing Text ... 773
Using Text with Rotate and Translate ... 773
Using the Matrix Stack ... 774

Lighting and Normals ... 778
Creating Light Sources ... 778
Lighting Models ... 780
Normal Vectors ... 781
Shading Model ... 781
Material Properties ... 782
Alpha Blending ... 783
Fog ... 783
Example ... 783
Bézier Curves ... 784
One-Dimensional Evaluators ... 785
Two-Dimensional Evaluators ... 787
Using the Mouse ... 788
Pick Commands ... 789
Three-Dimensional Arguments ... 790
Improve the Display of Graphics ... 790

14 Extending JMP ... 793
External Data Sources, Analytical Tools, and Automation

Real-Time Data Capture on Windows ... 797
Create a Data Feed Object ... 797
Read in Real-Time Data ... 799
Manage a Data Feed with Messages ... 800
Examples of Data Feed ... 802
Dynamic Link Libraries (DLLs) ... 804
Use Sockets in JSL ... 806
Socket-Related Commands ... 807
Messages for Sockets ... 808
Database Access ... 809
Run Query Builder Queries ... 809
Open Database Function ... 810
Creating a Database Connection and Executing SQL ... 812
Write a SQL Query ... 813
Work with SAS ... 814
Make a SAS DATA Step ... 814
Create SAS DATA Step Code for Formula Columns ... 814
SAS Variable Names ... 815
Connect to a SAS Metadata Server ... 815
Get the Values of SAS Macro Variables ... 818
Preferences ... 819

Sample Scripts . . . 819
Work with MATLAB . . . 821
Install MATLAB . . . 821
Work with R . . . 822
Installing R . . . 822
JMP to R Interfaces . . . 824
R JSL Scriptable Object Interfaces . . . 825
Conversion Between JMP Data Types and R Data Types . . . 825
Troubleshooting . . . 828
Examples . . . 829
Work with Python . . . 831
Install Python . . . 831
Troubleshooting the JMP Python Integration . . . 832
JMP-to-Python Interfaces . . . 836
Equivalent Data Types for Python Send() . . . 836
Equivalent Data Types for Python Get() . . . 837
JMP Scoping Operators and Python . . . 838
Python Name() and Python Send() Examples . . . 839
Additional Python Integration Examples . . . 840
Work with Microsoft Excel . . . 842
Parse XML . . . 843
OLE Automation . . . 845
Share Reports on JMP Live . . . 845
JMP Live Prerequisites . . . 845
Get an API Key . . . 845
Publish a Single Report to JMP Live . . . 846
Publish Several Reports to a JMP Live Folder . . . 847
Delete Reports in a JMP Live Folder . . . 847
Replace the Contents of a JMP Live Report . . . 848
Search for a JMP Live Report or Folder ID . . . 848
Update Data in JMP Live Reports . . . 849
Work with Other Content in JMP Live Reports . . . 850
HTTP Requests in JMP Live . . . 850
Communicate with REST Web Service . . . 850
Examples of HTTP Requests . . . 850
OAuth 2.0 for Web APIs . . . 853
Security . . . 853
Using Postman for Redirect URLs . . . 853
Authorization Code Grant . . . 854
Implicit Grant . . . 855

Resource Owner Grant 855
Using a Token 856
OAuth 2.0 Resources 856
Communicate with SAS Cloud Analytic Services 856
About CAS Actions 857
Connect to the CAS Server 858
Create and Submit a CAS Action 859
Create and Submit a CAS DATA Step Action 859
Create a New CAS Server 859
Import Data from the CAS Server 859
View a CAS Dataset 860
Get a List of Available CAS Libraries 860
Export Data from a Data Table to a CAS Server 860
Convert a CAS Table to a Data Table 860
Delete a CAS Table 863
Remove a CAS Table 863
Verify the CAS Server Connection 863
Get Details about the Current Connection 863
Disconnect from the CAS Server 863
Terminate All CAS Sessions 863

15 Creating Projects 865
Organizing Project Files in JMP

Create a Simple Project 867
Create a Project with a Bookmarks Pane and Log 868
Running Scripts in a Project 871
Create a Bookmark Group 872
Close Tabs in Projects 873
Get a Project 873
Reset the Window Layout 874
Run Startup Scripts to Control Workspaces 875

16 Creating Applications 877
Application Builder

Create Applications with Application Builder 879
Example of Creating an Application 879
Application Builder Terminology 881
Design an Application 882
Application Builder Window 883
Application Builder Red Triangle Options 884
Create an Application 886

Edit or Run an Application ... 897
Options for Saving Applications ... 898
Additional Examples of Creating Applications ... 899
Compile Add-Ins with the JMP Add-In Builder ... 915
Compile Scripts into an Add-In Using Add-In Builder ... 915
Edit an Add-In ... 920
Share an Add-In ... 920
Installing Multiple Add-Ins ... 920
Register an Add-In Using JSL ... 921
Create an Add-In Manually ... 921
Manage JMP Add-Ins ... 923

17 Common Tasks ... 925
Getting Started with Sample Scripts

Run a Script at Start Up ... 927
Convert Character Dates to Numeric Dates ... 928
Format Date-Time Values and Subset Data ... 929
Create a Formula Column ... 930
Extract Values from an Analysis into a Report ... 932
Create an Interactive Program ... 936

A Compatibility Notes ... 939
Changes in Scripting in JMP 16

Compatibility Issues in JMP 16.1 ... 939
Compatibility Issues in JMP 16.0 ... 939

B Efficient Scripts ... 941
Tips for Writing Scripts That Run Faster

Using Matrices ... 943
Design of Experiments ... 944
Scripting Data Tables ... 944
Scripting Graphs and Display Boxes ... 945
Scripting Numbers, Strings, Arrays, and Lists ... 946
Programming Tips ... 946
Troubleshooting ... 946

C References ... 947

D Glossary ... 949
Terms, Concepts, and Placeholders

E Technology License Notices ... 953

Chapter 1

Learn about JMP

Documentation and Additional Resources

Learn about JMP documentation, such as book conventions, descriptions of each JMP document, the Help system, and where to find additional support.

Contents

Formatting Conventions in JMP Documentation . . . 25
JMP Help . . . 26
JMP Documentation Library . . . 26
Additional Resources for Learning JMP . . . 32
JMP Tutorials . . . 32
Sample Data Tables . . . 32
Learn about Statistical and JSL Terms . . . 33
Learn JMP Tips and Tricks . . . 33
JMP Tooltips . . . 33
JMP User Community . . . 34
Free Online Statistical Thinking Course . . . 34
JMP New User Welcome Kit . . . 34
Statistics Knowledge Portal . . . 34
JMP Training . . . 34
JMP Books by Users . . . 35
The JMP Starter Window . . . 35
JMP Technical Support . . . 35

Formatting Conventions in JMP Documentation

These conventions help you relate written material to information that you see on your screen:

- Sample data table names, column names, pathnames, filenames, file extensions, and folders appear in Helvetica (or sans-serif online) font.
- Code appears in `Lucida Sans Typewriter` (or monospace online) font.
- Code output appears in *`Lucida Sans Typewriter`* italic (or monospace italic online) font and is indented farther than the preceding code.
- **Helvetica bold** formatting (or bold sans-serif online) indicates items that you select to complete a task:
  - buttons
  - check boxes
  - commands
  - list names that are selectable
  - menus
  - options
  - tab names
  - text boxes
- The following items appear in italics:
  - words or phrases that are important or have definitions specific to JMP
  - book titles
  - variables
- Features that are for JMP Pro only are noted with the JMP Pro icon JMP PRO. For an overview of JMP Pro features, visit https://www.jmp.com/software/pro.

Note: Special information and limitations appear within a Note.

Tip: Helpful information appears within a Tip.

JMP Help

JMP Help in the Help menu enables you to search for information about JMP features, statistical methods, and the JMP Scripting Language (or *JSL*). You can open JMP Help in several ways:

- Search and view JMP Help on Windows by selecting **Help > JMP Help**.
- On Windows, press the F1 key to open the Help system in the default browser.
- Get help on a specific part of a data table or report window. Select the Help tool from the **Tools** menu and then click anywhere in a data table or report window to see the Help for that area.
- Within a JMP window, click the **Help** button.

Note: The JMP Help is available for users with Internet connections. Users without an Internet connection can search all books in a PDF file by selecting **Help > JMP Documentation Library**. See "JMP Documentation Library" for more information.

JMP Documentation Library

The Help system content is also available in one PDF file called *JMP Documentation Library*. Select **Help > JMP Documentation Library** to open the file. You can also download the Documentation PDF Files add-in if you prefer searching individual PDF files of each document in the JMP library. Download the available add-ins from https://community.jmp.com.

The following table describes the purpose and content of each document in the JMP library.

Document Title	Document Purpose	Document Content
Discovering JMP	If you are not familiar with JMP, start here.	Introduces you to JMP and gets you started creating and analyzing data. Also learn how to share your results.
Using JMP	Learn about JMP data tables and how to perform basic operations.	Covers general JMP concepts and features that span across all of JMP, including importing data, modifying columns properties, sorting data, and connecting to SAS.

Document Title	Document Purpose	Document Content
Basic Analysis	Perform basic analysis using this document.	Describes these Analyze menu platforms: • Distribution • Fit Y by X • Tabulate • Text Explorer Covers how to perform bivariate, one-way ANOVA, and contingency analyses through Analyze > Fit Y by X. How to approximate sampling distributions using bootstrapping and how to perform parametric resampling with the Simulate platform are also included.
Essential Graphing	Find the ideal graph for your data.	Describes these Graph menu platforms: • Graph Builder • Scatterplot 3D • Contour Plot • Bubble Plot • Parallel Plot • Cell Plot • Scatterplot Matrix • Ternary Plot • Treemap • Chart • Overlay Plot The book also covers how to create background and custom maps.
Profilers	Learn how to use interactive profiling tools, which enable you to view cross-sections of any response surface.	Covers all profilers listed in the Graph menu. Analyzing noise factors is included along with running simulations using random inputs.

Document Title	Document Purpose	Document Content
Design of Experiments Guide	Learn how to design experiments and determine appropriate sample sizes.	Covers all topics in the DOE menu.
Fitting Linear Models	Learn about Fit Model platform and many of its personalities.	Describes these personalities, all available within the Analyze menu Fit Model platform: • Standard Least Squares • Stepwise • Generalized Regression • Mixed Model • MANOVA • Loglinear Variance • Nominal Logistic • Ordinal Logistic • Generalized Linear Model

Document Title	Document Purpose	Document Content
Predictive and Specialized Modeling	Learn about additional modeling techniques.	Describes these Analyze > Predictive Modeling menu platforms: • Neural • Partition • Bootstrap Forest • Boosted Tree • K Nearest Neighbors • Naive Bayes • Support Vector Machines • Model Comparison • Model Screening • Make Validation Column • Formula Depot Describes these Analyze > Specialized Modeling menu platforms: • Fit Curve • Nonlinear • Functional Data Explorer • Gaussian Process • Time Series • Matched Pairs Describes these Analyze > Screening menu platforms: • Modeling Utilities • Response Screening • Process Screening • Predictor Screening • Association Analysis • Process History Explorer

Document Title	Document Purpose	Document Content
Multivariate Methods	Read about techniques for analyzing several variables simultaneously.	Describes these Analyze > Multivariate Methods menu platforms: • Multivariate • Principal Components • Discriminant • Partial Least Squares • Multiple Correspondence Analysis • Structural Equation Models • Factor Analysis • Multidimensional Scaling • Item Analysis Describes these Analyze > Clustering menu platforms: • Hierarchical Cluster • K Means Cluster • Normal Mixtures • Latent Class Analysis • Cluster Variables
Quality and Process Methods	Read about tools for evaluating and improving processes.	Describes these Analyze > Quality and Process menu platforms: • Control Chart Builder and individual control charts • Measurement Systems Analysis • Variability / Attribute Gauge Charts • Process Capability • Model Driven Multivariate Control Chart • Legacy Control Charts • Pareto Plot • Diagram • Manage Spec Limits • OC Curves

Document Title	Document Purpose	Document Content
Reliability and Survival Methods	Learn to evaluate and improve reliability in a product or system and analyze survival data for people and products.	Describes these Analyze > Reliability and Survival menu platforms: • Life Distribution • Fit Life by X • Cumulative Damage • Recurrence Analysis • Degradation • Destructive Degradation • Reliability Forecast • Reliability Growth • Reliability Block Diagram • Repairable Systems Simulation • Survival • Fit Parametric Survival • Fit Proportional Hazards
Consumer Research	Learn about methods for studying consumer preferences and using that insight to create better products and services.	Describes these Analyze > Consumer Research menu platforms: • Categorical • Choice • MaxDiff • Uplift • Multiple Factor Analysis
Scripting Guide	Learn about taking advantage of the powerful JMP Scripting Language (JSL).	Covers a variety of topics, such as writing and debugging scripts, manipulating data tables, constructing display boxes, and creating JMP applications.
JSL Syntax Reference	Read about many JSL functions on functions and their arguments, and messages that you send to objects and display boxes.	Includes syntax, examples, and notes for JSL commands.

Additional Resources for Learning JMP

In addition to reading JMP help, you can also learn about JMP using the following resources:

- "JMP Tutorials"
- "Sample Data Tables"
- "Learn about Statistical and JSL Terms"
- "Learn JMP Tips and Tricks"
- "JMP Tooltips"
- "JMP User Community"
- "Free Online Statistical Thinking Course"
- "JMP New User Welcome Kit"
- "Statistics Knowledge Portal"
- "JMP Training"
- "JMP Books by Users"
- "The JMP Starter Window"

JMP Tutorials

You can access JMP tutorials by selecting **Help > Tutorials**. The first item on the **Tutorials** menu is **Tutorials Directory**. This opens a new window with all the tutorials grouped by category.

If you are not familiar with JMP, start with the **Beginners Tutorial**. It steps you through the JMP interface and explains the basics of using JMP.

The rest of the tutorials help you with specific aspects of JMP, such as designing an experiment and comparing a sample mean to a constant.

Sample Data Tables

All of the examples in the JMP documentation suite use sample data. Select **Help > Sample Data Library** to open the sample data directory.

To view an alphabetized list of sample data tables or view sample data within categories, select **Help > Sample Data**.

Sample data tables are installed in the following directory:

On Windows: C:\Program Files\SAS\JMP\16\Samples\Data

On macOS: \Library\Application Support\JMP\16\Samples\Data

In JMP Pro, sample data is installed in the JMPPRO (rather than JMP) directory.

To view examples using sample data, select **Help > Sample Data** and navigate to the Teaching Resources section. To learn more about the teaching resources, visit https://jmp.com/tools.

Learn about Statistical and JSL Terms

For help with statistical terms, select Help > Statistics Index. For help with JSL scripting and examples, select **Help > Scripting Index**.

Statistics Index Provides definitions of statistical terms.

Scripting Index Lets you search for information about JSL functions, objects, and display boxes. You can also edit and run sample scripts from the Scripting Index and get help on the commands.

Learn JMP Tips and Tricks

When you first start JMP, you see the Tip of the Day window. This window provides tips for using JMP.

To turn off the Tip of the Day, clear the **Show tips at startup** check box. To view it again, select **Help > Tip of the Day**. Or, you can turn it off using the Preferences window.

JMP Tooltips

JMP provides descriptive tooltips (or *hover labels*) when you hover over items, such as the following:

- Menu or toolbar options
- Labels in graphs
- Text results in the report window (move your cursor in a circle to reveal)
- Files or windows in the Home Window
- Code in the Script Editor

Tip: On Windows, you can hide tooltips in the JMP Preferences. Select **File > Preferences > General** and then deselect **Show menu tips**. This option is not available on macOS.

JMP User Community

The JMP User Community provides a range of options to help you learn more about JMP and connect with other JMP users. The learning library of one-page guides, tutorials, and demos is a good place to start. And you can continue your education by registering for a variety of JMP training courses.

Other resources include a discussion forum, sample data and script file exchange, webcasts, and social networking groups.

To access JMP resources on the website, select **Help > JMP User Community** or visit https://community.jmp.com.

Free Online Statistical Thinking Course

Learn practical statistical skills in this free online course on topics such as exploratory data analysis, quality methods, and correlation and regression. The course consists of short videos, demonstrations, exercises, and more. Visit https://www.jmp.com/statisticalthinking.

JMP New User Welcome Kit

The JMP New User Welcome Kit is designed to help you quickly get comfortable with the basics of JMP. You'll complete its thirty short demo videos and activities, build your confidence in using the software, and connect with the largest online community of JMP users in the world. Visit https://www.jmp.com/welcome.

Statistics Knowledge Portal

The Statistics Knowledge Portal combines concise statistical explanations with illuminating examples and graphics to help visitors establish a firm foundation upon which to build statistical skills. Visit https://www.jmp.com/skp.

JMP Training

SAS offers training on a variety of topics led by a seasoned team of JMP experts. Public courses, live web courses, and on-site courses are available. You might also choose the online e-learning subscription to learn at your convenience. Visit https://www.jmp.com/training.

JMP Books by Users

Additional books about using JMP that are written by JMP users are available on the JMP website. Visit https://www.jmp.com/books.

The JMP Starter Window

The JMP Starter window is a good place to begin if you are not familiar with JMP or data analysis. Options are categorized and described, and you launch them by clicking a button. The JMP Starter window covers many of the options found in the Analyze, Graph, Tables, and File menus. The window also lists JMP Pro features and platforms.

- To open the JMP Starter window, select **View** (**Window** on macOS) **> JMP Starter**.
- To display the JMP Starter automatically when you open JMP on Windows, select **File > Preferences > General**, and then select **JMP Starter** from the Initial JMP Window list. On macOS, select **JMP > Preferences > Initial JMP Starter Window**.

JMP Technical Support

JMP technical support is provided by statisticians and engineers educated in SAS and JMP, many of whom have graduate degrees in statistics or other technical disciplines.

Many technical support options are provided at https://www.jmp.com/support, including the technical support phone number.

Chapter 2

Introduction

Welcome to the JMP Scripting Language

The JMP Scripting Language, or *JSL*, lets you write scripts to re-create results in JMP. Power users often develop scripts to extend JMP's functionality and automate a regularly scheduled analysis in production settings. If you do not want to learn JSL, JMP can write the scripts for you.

JSL is used to perform many actions:

- implements column formulas
- launches platforms
- interactively modifies platforms
- creates graphics

Contents

What JSL Can Do for You 39
Help with Learning JSL 39
The Scripting Guide 39
The Scripting Index 40
Let JMP Teach You JSL 42
Terminology 43
Basic JSL Syntax 46
Book Conventions 47

What JSL Can Do for You

JMP can automatically save scripts to reproduce any data table or analysis in its current state. You can pause anytime in your analysis to save a script to a script window (or *script editor*), in a data table, or in an analysis report. You can then modify the script as needed for future projects. When you are finished with your work, you can then save a script to reproduce your final results.

Here are some examples where JSL scripts can be helpful:

- Suppose you need to describe an analysis process in detail, from beginning to end. An example is to create an audit trail for a governing agency, or for peers reviewing your journal article.
- Suppose you have a set of analysis steps that should be followed routinely by your lab technicians.
- Suppose you fit the same model to new data every day, and the steps are always the same.

You can use JMP interactively as usual, save scripts to reproduce your work, and in the future run those scripts to reproduce your results.

There are a few things that JSL is not designed to do:

- JMP cannot record scripts while you are working. Though script-recording is a useful feature in some other scripting languages, it is less important for software like JMP, where the results are what matter. You cannot use script-recording to observe how a sequence of interactive steps is performed.
- JSL is not an alternative command-line interface for using the program.

Help with Learning JSL

There are several places within JMP to get help with writing or understanding a JSL script.

The Scripting Guide

The *Scripting Guide* begins with basic information (such as terminology and syntax) for JMP users who are not familiar with the scripting language. The *Scripting Guide* then progresses to more advanced information.

Chapters 2 through 4	Includes information about learning JSL, producing basic scripts, and introduces you to the JSL scripting environment.
Chapters 5 through 8	Introduces the building blocks of the language; working with basic data types, such as numbers and strings; writings lists, matrices, and associate arrays; namespaces; and the fundamentals of programming in JSL.
Chapters 9 through 13	Covers using JSL with objects in JMP, such as data tables, platforms, windows, and graphics.
Chapter 14	Describes how to write scripts that work with external programs, such as SAS, R, and Microsoft Excel.
Chapter 15	Describes how to organize files that you use in an analysis, perform analyses, and run scripts from one workspace.
Chapter 16	Introduces creating JMP applications in Application Builder, a drag-and-drop environment for visually designing windows with buttons, lists, graphs, and other objects. The chapter also describes how to use Add-In Builder to compile scripts into one easily shared file.
Chapter 17	Contains a collection of recipes, or script examples, that you can copy and modify for your own use.
Appendices A, B, C, and D	Provides information about compatibility issues with the previous version of JMP, provides tips for writing more efficient scripts, gives information about references, and defines JSL concepts and terminology.

The Scripting Index

The Scripting Index on the **Help** menu provides a brief description and the syntax for JSL functions, objects, and display boxes. Each entry includes an example that you can run and modify to test your own code. And an embedded log window lets you see messages as examples are run.

Notes:

- To display the Scripting Index entry for a function in a script, run the script, press Alt, and then double-click the function. You can also run the script, right-click the function, and select **Help Scripting Index.**
- To copy the syntax or description from the Scripting Index window, right-click the area that you want to copy and select **Copy Text**.

The Scripting Index window includes the following buttons:

Click the **Clear** button to clear the search text box to begin a new search.

Click the arrow to set search filter options and parameters.

After you edit the sample script in the Scripting Index, click this button to revert to the original script.

Note: When you edit an example script, the changes persist when you view other entries in the Scripting Index. To revert the script, click **Reset**, which appears above the script after you edit it.

Search Filter Options

Click the down arrow button next to the search box to refine your search.

Contains Terms Returns items that contain a part of the search criteria. A search for “ease oom” returns messages such as “Release Zoom”.

Contains Phrase Returns items that contain the exact search criteria. A search for “text box” returns entries that contain “text” followed directly by “box” (for example, “Context Box” and “Text Box”).

Starts With Phrase Returns items that start with the search criteria.

Ends With Phrase Returns items that end with the search criteria.

Whole Phrase Returns items that consist of the entire string. A search for “text box” returns entries that contain only “text box”.

Regular Expression Enables you to use the wildcard (*) and period (.) in the search box. Searching for “get.*name” looks for items that contain “get” followed by one or more words. It returns “Get Color Theme Names”, “Get Name Info”, and “Get Effect Names”, and so on.

Invert Result Returns items that do not match the search criteria.

Match All Terms Returns items that contain both strings. A search for “t test” returns elements that contain either or both of the search strings: “Pat Test”, “Shortest Edit Script” and “Paired t test”.

Ignore Case Ignores the case in the search criteria.

Match Whole Words Returns items that contain each word in the string based on the Match All Terms setting. If you search for “data filter”, and Match All Terms is selected, entries that contain both “data” and “filter” are returned.

Search Category and Item Names Returns category names (in the left column) and item names (in the middle column) that match the search criteria.

Search Prototypes and Descriptions Returns prototypes (the JSL syntax) and descriptions that match the search criteria.

Search Examples Returns examples that match the search criteria.

Let JMP Teach You JSL

The best JSL writer is JMP. You can work in JMP interactively and then save the results as a script to reuse later. With simple modifications, your script can serve as a template for speeding up routine tasks.

Because JSL is a very flexible language, you can reach your goals in many different ways. Here is an example. Typically, the script that JMP saves for you specifies every detail of your analysis, even if most of the details happen automatically by default. Does that mean that the scripts that you write have to be just as complete and detailed? Not at all. You usually need to specify only those details that you would select in the graphical user interface (GUI). For example, if you open Big Class.jmp from the sample data folder and want to launch Distribution for height, weight, and sex, the following script is all that is necessary:

```
Distribution( Y( :height, :weight, :sex ) );
```

Suppose you run the Distribution platform in the GUI and then select **Save Script > To Script Window** from the red triangle menu for the report. The following script appears:

```
Distribution(
   Continuous Distribution( Column( :height ) ),
   Continuous Distribution( Column( :weight ) ),
   Nominal Distribution( Column( :sex ) ),
);
```

Both scripts give the same result.

Feel free to experiment with JSL. If you think something ought to be possible, it probably is. Give it a try, and see what happens.

Terminology

Before you begin creating scripts, you should become familiar with basic JSL terms used throughout the *Scripting Guide*.

Operators and Functions

An *operator* is one- or two-character symbol (such as + or =) for features such as common arithmetic actions, scoping names, regular expressions, concatenating, and subscripting.

A *function* is a command that might contain additional information for the function to use.

Certain JSL functions work the same as operators but provide access to more complex actions. For example, the following two lines are equivalent:

```
2 + 3; // returns 5
Add( 2, 3 ); // returns 5
```

The first line uses the + operator. The second line uses the `Add()` function equivalent.

Although all JSL operators have function equivalents, not all functions have operator equivalents. For example, `Sqrt(a)` can be represented only by the `Sqrt()` function.

Note: In previous versions of JMP and its documentation, the terms *operators* and *functions* were used interchangeably. Now each term has a specific meaning.

Objects and Messages

An *object* is a dynamic entity in JMP, such as a data table, a data column, a platform results window, a graph, and so on. Most objects can receive messages that instruct the object to perform some action on itself.

A *message* is a JSL expression that is directed to an object. That object knows how to evaluate the message. In the following example, `dt` is the data table object. << indicates that a message follows. In the following example, the message tells JMP to create a summary table with the specified variables.

```
dt << Summary( Group( :age ), Mean( :height ) )
```

In this expression, `dt` is the name of a variable that contains a reference to a data table. You could use any name for this variable. The *Scripting Guide* commonly uses `dt` to represent data table references. Here are some of the more common names used to represent references to certain objects:

Abbreviation	Object
`dt`	data table
`col`	column in a data table
`colname`	the name of a column in a data table
`obj`	an object
`db`	display box

These variables are not pre-assigned references. Each one must be assigned prior to its use. In the following example, the global variable named `A` is assigned the value “Hello, World”. When the `Show( A )` command is processed, the result is the value of `A`.

```
A = "Hello, World";
Show( A );
   A = "Hello, World";
```

Arguments and Parameters

An *argument* is additional information that you can provide to a function or message. For example, in `Root(25)`, `25` is an argument to the `Root()` function. `Root()` acts on the argument that you provide and returns the result: 5.

Programming and scripting books commonly talk about parameters as well. A *parameter* is a description of the argument that a function accepts. For example, the general specification for `Root()` might be `Root( number )`, where *`number`* is the parameter.

Parameter and *argument* express two perspectives of the same concept: information that a function needs.

For simplicity in the *Scripting Guide*, we use the word *argument* in both cases.

A *named argument* is an optional argument that you select from a predetermined set and explicitly define. For example, `title("My Line Graph")` in the `Graph Box()` function is a named argument because the title is explicitly defined as such.

```
Graph Box( title( "My Line Graph" ),
      Frame Size( 300, 500 ),
      Marker( Marker State( 3 ), [11 44 77], [75 25 50] );
      Pen Color( "Blue" );
      Line( [10 30 70], [88 22 44] ));
```

Note that the `Frame Size()` arguments `300` and `500` are not named. The position of these arguments implies meaning; the first argument is always the width, the second argument is always the height.

Optional Arguments

Functions and messages require certain arguments, and other arguments are optional. You can include them, but you do not have to. In specifications, optional arguments are enclosed in angle brackets. For example:

```
Root( x, <n> )
```

The *x* argument is required. The *n* argument is optional.

Optional arguments often have a default value. For example, for `Root()`, the default value of *n* is 2:

Code	Output	Explanation
`Root( 25 )`	`5`	Returns the square root of 25.
`Root( 25, 2 )`	`5`	Returns the square root of 25.
`Root( 25, 3 )`	`2.92401773821287`	Returns the cube root of 25.

Or and the Vertical Bar Symbol

A single vertical bar (`|`) represents a logical `OR`. For brevity, `|` represents the word *or* when referring to alternative values.

For example, a pathname can be either absolute or relative. When you see an argument such as `absolute|relative`, this means that you enter *either* one of the following two options:

- `absolute` indicates an absolute pathname.
- `relative` indicates a relative pathname.

More than two options can also be strung together with a vertical bar in this way.

Script Formatting

Whitespace characters (such as spaces, tabs, and newlines) and capitalization are ignored in JSL. This means that the following two expressions are equivalent:

Expression 1:

```
sum = 0;
For( i = 1, i <= 10, i++,
   sum += i;
   Show( i, sum );
);
```

Expression 2:

```
Sum = 0;
For( i = 1, i <= 10, i++,
   Sum += i;
   Show( i, Sum );
);
```

You can format your script in any way that you like. However, the script editor can also format your script for you. The *Scripting Guide* uses the script editor's default formatting for capitalization, spaces, returns, tabs, and so on. See "Work with the Script Editor" for more information about using the script editor.

Note: The only white space exception is two-character operators (such as <= or ++). The operators cannot be separated by a space.

Basic JSL Syntax

A JSL script is a series of expressions. Each expression is a section of JSL code that accomplishes a task. JSL expressions hold data, manipulate data, and send commands to objects.

Many expressions are nested message names. Message contents are enclosed in parentheses:

```
Message Name ( argument 1, argument 2, ... );
```

The meaning of JSL names depends on the context. The same name might mean one thing in a data table context and something entirely different in a function context. See "Rules for Name Resolution".

Almost anything that follows certain punctuation rules, such as matching parentheses, is a valid JSL expression. For example:

```
win = New Window( "Window Example",
   <<Modal,
   Text Box( "Hello, World" ),
   Text Box( "-----" ),
   Button Box( "OK" )
);
```

Notes:

- Names can have embedded spaces. See "Names".
- Message contents are enclosed in parentheses, which must be balanced. See "Parentheses".
- Items are separated by commas. See "Commas".
- JSL is not case sensitive; you can type "`text box();`" or "`Text Box()`".

- Messages are commonly nested inside other messages.

Book Conventions

In the *Scripting Guide*, the function names that are case sensitive are capitalized as you must use them. Arguments that are placeholders for actual choices are in `lowercase`. For example, `Connect Color` is a function that you need to type as is, and `color` stands for some color choice that you make yourself.

```
Connect Color(color);
```

In this case, the argument in parentheses must be some color value (for example, a JMP color number, or a supported color name like `"red"`, `"blue"`, and so on, or an RGB value given as a list, such as `{.75, .50, .50}`). Sometimes alternatives like these are shown with the vertical bar (`|`) character for "or," like this:

```
Connect Color( number | "color name" | {r,g,b} );
```

Syntax coloring is applied to scripts that you can paste into a script editor and then run.

Chapter 3

Getting Started

Let JMP Write Your Scripts

You often have to produce the same reports for the same data on a regular basis. This chapter shows you how to let JMP write scripts for common tasks like importing text data, opening Microsoft Excel files, and producing reports. A final tutorial shows you how to put it all together into a single script to open an Microsoft Excel file and produce three reports automatically.

The *Scripting Guide* is written for users who are familiar with JMP but might not be familiar with JSL. For more information about performing common tasks, see *Using JMP*. *Discovering JMP* is also a good resource for learning basic concepts and understanding the JMP workflow.

Contents

Capturing a Script for an Analysis Report 51
Capturing a Script for a Data Table 52
Capturing a Script to Import a File 53
Gluing Scripts Together 54

Capturing a Script for an Analysis Report

Here are the basic steps for capturing a script to reproduce an analysis:

1. Launch a platform, such as Distribution.
2. Make any changes or additions that you need. For example, add tests and other graphs.
3. Capture the script to recreate your results.

You can save the script in the data table, so that if you send the data table to others, they can run your script and duplicate your reports.

Example

Follow these steps to produce a distribution report, capture the script to reproduce it, and save it to the data table.

Note: The data tables that you use in examples are located in JMP's Samples/Data folder.

1. Select **Help > Sample Data Library** and open Companies.jmp.
2. Select **Analyze > Distribution** to open the Distribution launch window.
3. Select Profits ($M) in the Select Columns box and click the **Y, Columns** button.
4. Click **OK**.

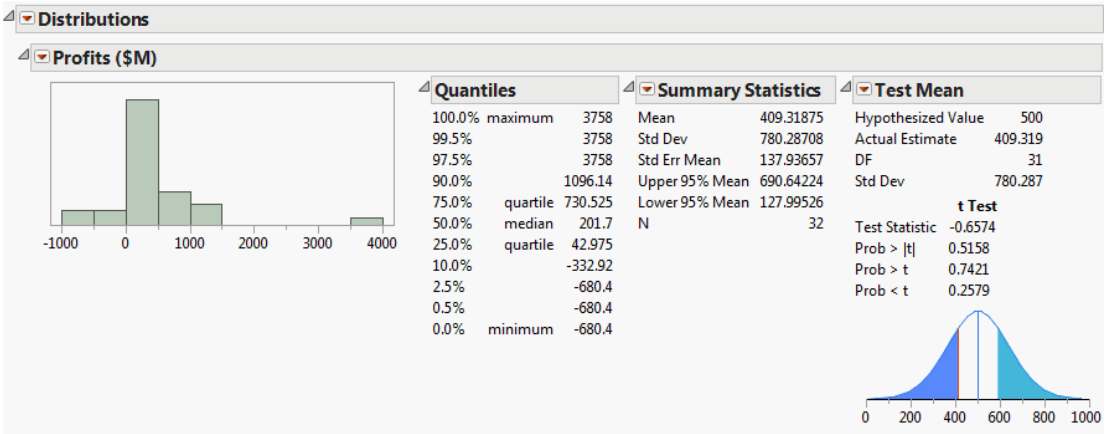
   The Distribution report window appears.
5. Click the Distributions red triangle and select **Stack** to make your report horizontal.
6. Click the red triangle next to Profits ($M) and deselect **Outlier Box Plot** to turn the option off.
7. Click the red triangle next to Profits ($M) and select **Test Mean**.

   The Test Mean window appears.
8. Type 500 in the **Specify Hypothesized Mean** box.
9. Click **OK**.

   The test for the mean is added to the report window.

   Now you have your customized report.

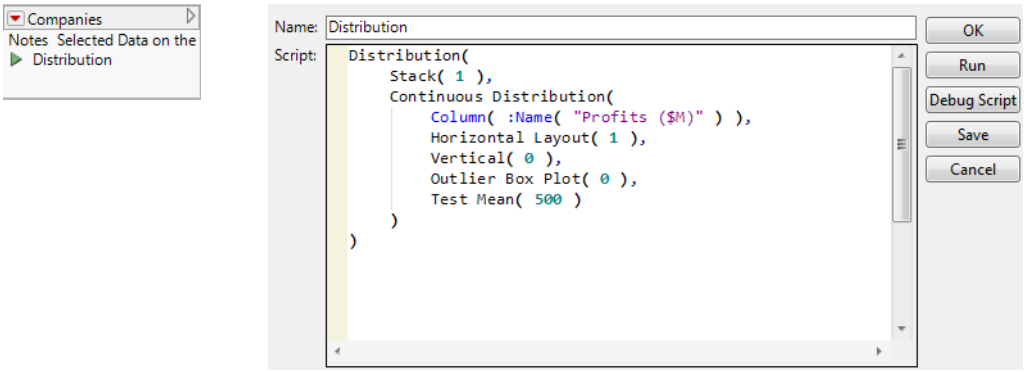
Figure 3.1 Customized Distribution Report



10. Click the Distributions red triangle and select **Save Script > To Data Table**.

    Your data table now has a script named Distribution saved to it. Right-click the Distribution script and select **Edit** to see the script.

Figure 3.2 Distribution Script Saved to the Data Table



11. To run the script and reproduce your final report exactly, click the green triangle next to the script.

Capturing a Script for a Data Table

Here are the basic steps for capturing a script to reproduce a data table:

1. Open the data table.

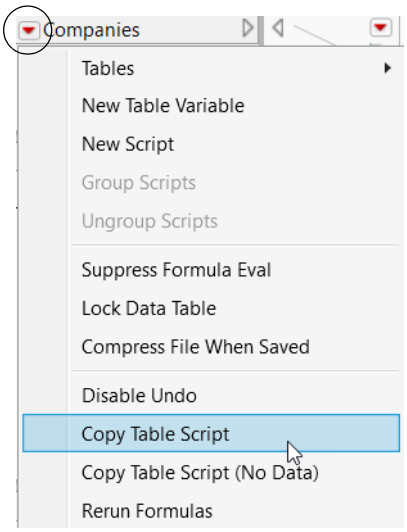
2. Make any changes that you need. For example, add a script, correct values, add new columns.
3. Capture the script to recreate your data table.

Example

Use the data table from the previous example, where you saved a script to it.

1. In the data table, select the red triangle next to the data table's name.
2. Select **Copy Table Script**.

Figure 3.3 Copy the Table Script



3. Open a script window by selecting **File > New > Script**.
4. Select **Edit > Paste**.

You now have a script that duplicates your data table. You can save this script and run it at any time to recreate your data table, with all its scripts attached.

Capturing a Script to Import a File

To capture a script that imports a file, you open the file in JMP. JMP automatically records the steps that occurred when you opened the file.

Import a Text File

1. Select **File > Open**.

The Open Data File window appears.

2. Select **Text Files** from the list next to **File name**.
3. In the Open as section, select **Data (Best Guess).**

   JMP formats the data based on tabs, commas, white space, and other characters in the text file.
4. Browse to select the file, and then select **Open**.

   The file is opened as a data table. The data table includes a script named Source. This JSL script imports your text file with the text import rules that you used.
5. Right-click the Source script and select **Edit**.

   You can copy this script, paste it into a new script window, and save it. Then you can run this script later to reimport the text file.

Tip: The import script is an `Open()` expression that specifies the text file and the import options to correctly import the file into JMP. The first part of this expression is the pathname to the specific file that you imported. If you save this script and want to run it a different place, you might need to edit the pathname so that it points to the text file. Pathnames are discussed in greater detail in "Path Variables".

Gluing Scripts Together

Suppose new data is saved out to an Microsoft Excel file once a week, and you need to produce the same reports every week. You could open the file and perform the same steps every week. However, creating a script that imports the new Microsoft Excel file into JMP and runs all analyses automatically is more efficient. The following example shows you how to set up your script and run it each week.

Import the Microsoft Excel File

1. Open a new script window (**File > New > Script**).
2. In your script window, enter the `Open()` expression to open the Solubil.xlsx sample import data file. The file is located in JMP's Samples/Import Data folder.

   ```
   dt = Open( "$SAMPLE_IMPORT_DATA/Solubil.xlsx" );
   ```

   Be sure to put the semicolon at the end of this expression, because you will add more expressions. The semicolon glues expressions together.
3. Run your script to import the Microsoft Excel file by selecting **Edit > Run Script**.

   The Microsoft Excel file opens as a data table.

Notes:

- You can also include the `Excel Wizard` argument in the `Open()` expression to preview the worksheet before importing it. See "Import Data from a Microsoft Excel File".
- You can specify an absolute or relative path to the file rather than using a path variable. For relative links, the script and file being opened must be in the same relative location each time you run the script. With absolute links, make sure that other users running the script have access to the file's location. See "Path Variables" for more information about using pathnames.

Run Your Reports and Capture Their Scripts

You have three reports to produce: a distribution report, a 3D scatterplot, and a multivariate report. Perform each one using the GUI, and add its script to the script window.

1. With your new data table open, select **Analyze > Distribution**.
2. Select all the columns except Labels and click **Y, Columns**.
3. Click **OK**.
4. Press Ctrl, click the eth red triangle, and then select **Histogram Options > Show Counts**.

   Bar counts are added to all six histograms.
5. In the Distribution window, click the Distributions red triangle and select **Save Script > To Clipboard**.
6. Click a line or two after your `Open()` expression and select **Edit > Paste**.
7. Type a semicolon after the last close parenthesis.
8. Select **Graph > Scatterplot 3D**.
9. Select all the columns except Labels and click **Y, Columns**.
10. Click **OK**.
11. Copy and paste the script for Scatterplot 3D into the script window just like you did for your Distribution report. Be sure to add the semicolon at the end.
12. Select **Analyze > Multivariate Methods > Multivariate**.
13. Select all the columns except Labels and click **Y, Columns**.
14. Click **OK**.
15. Copy and paste the script for Multivariate into the script window just like you did for Distributions and Scatterplot 3D.

    You should see the following script:

```
dt = Open( "$SAMPLE_IMPORT_DATA/Solubil.xlsx" );
Distribution(
  Continuous Distribution( Column( :eth ), Show Counts( 1 ) ),
  Continuous Distribution( Column( :oct ), Show Counts( 1 ) ),
```

```
    Continuous Distribution( Column( :cc14 ), Show Counts( 1 ) ),
    Continuous Distribution( Column( :c6c6 ), Show Counts( 1 ) ),
    Continuous Distribution( Column( :hex ), Show Counts( 1 ) ),
    Continuous Distribution( Column( :chc13 ), Show Counts( 1 ) ),
);
Scatterplot 3D(
     Y( :eth, :oct, :cc14, :c6c6, :hex, :chc13 ),
     Frame3D( Set Grab Handles( 0 ), Set Rotation( -54, 0, 38) )
);
Multivariate(
   Y( :eth, :oct, :cc14, :c6c6, :hex, :chc13 ),
   Estimation Method( "Row-wise" ),
   Scatterplot Matrix( Density Ellipses( 0 ), Shaded Ellipses( 0 ) )
);
```

Save the Script

You now have a script that reproduces all of the steps that you performed manually. Save the script, and close your data table and all its report windows.

1. In the script window that contains your script, select **File > Save** or **File > Save As**.
2. Specify a filename (for example, Weekly Report).
3. Click **Save**.

Run the Script

As long as your weekly updated Microsoft Excel file is saved in the same place and contains the same columns, you can run your script and automatically produce all your reports.

1. Open the script that you saved.
2. Select **Edit > Run Script**.

   Your Microsoft Excel file is opened in JMP, and all three of your reports appear.

You can send this script to others. As long as they have access to the same Microsoft Excel file in the same location, they can also run the script in JMP and see your reports.

Advanced Note: Auto-Submit

If you want a particular script to always be executed instead of opened into the script window, put the following command on the first line of the script:

```
//!
```

If this is not the very first line, with nothing else on the same line, this command does nothing.

You can override this command when opening the file.

1. Select **File > Open**.

2. Press Ctrl, select the JSL file, and then click **Open**.

   The script opens into a script window instead of being executed.

The command is also ignored when you right-click the file in the Home Window and select **Edit Script**.

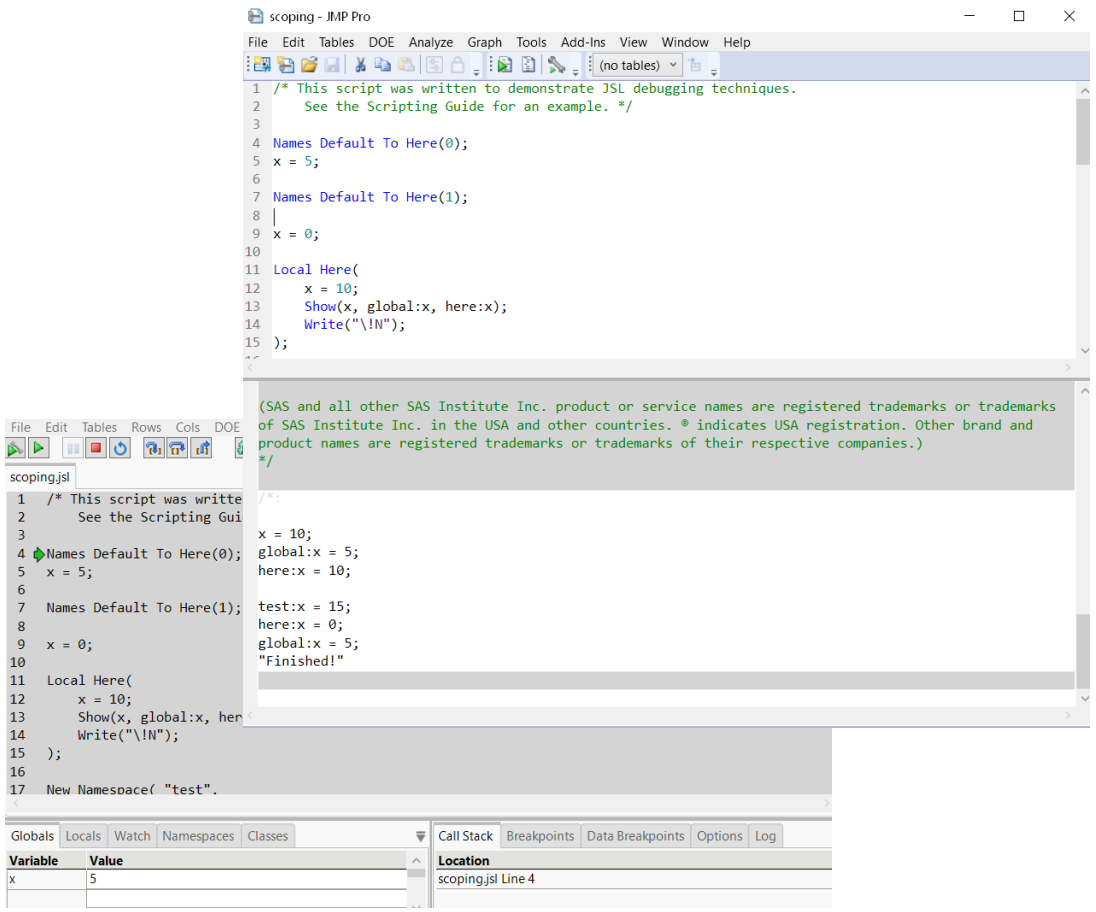
Chapter 4

Scripting Tools

Using the Script Editor, Log Window, Debugger and Profiler

JMP provides several programming tools for script writers. The script editor supports syntax coloring, autocompletes functions as you type, highlights matching braces, allows for code folding, and has additional features to help you develop scripts more quickly. Error messages and output are shown in the log window, which can be displayed inside the script editor. The JMP Scripting Language (JSL) Debugger and Profiler can help you troubleshoot your scripts.

Figure 4.1 Script Editor with Embedded Log and the Debugger



Contents

Work with the Script Editor . . . 61
Introduction to the Script Editor . . . 61
Run a Script . . . 62
Stop a Script . . . 63
Edit a Script . . . 63
Color Coding and Themes . . . 63
Autocomplete Functions . . . 64
Tooltips . . . 64
Split a Window . . . 65
Match Parentheses, Brackets, and Braces . . . 66
Select a Rectangular Block of Text . . . 67
Select Fragmented Text . . . 68
Drag and Drop Text . . . 68
Find and Replace . . . 68
Automatic Formatting . . . 69
Errors Are Marked by Special Characters . . . 69
Typing in Multiple Rows at Once . . . 69
Add Code Folding Markers . . . 70
Advanced Options . . . 71
Set Preferences for the Script Editor . . . 72
Work with the Log . . . 72
Enhanced Log in JMP . . . 73
Text Log in JMP . . . 78
Debug or Profile Scripts . . . 79
Debugger and Profiler Window . . . 80
Work with Breakpoints . . . 84
View Variables . . . 87
Work with Watches . . . 87
Modify Preferences in Debugger . . . 88
Persistent Debugger Sessions . . . 89
Examples of Debugging and Profiling Scripts . . . 89

Work with the Script Editor

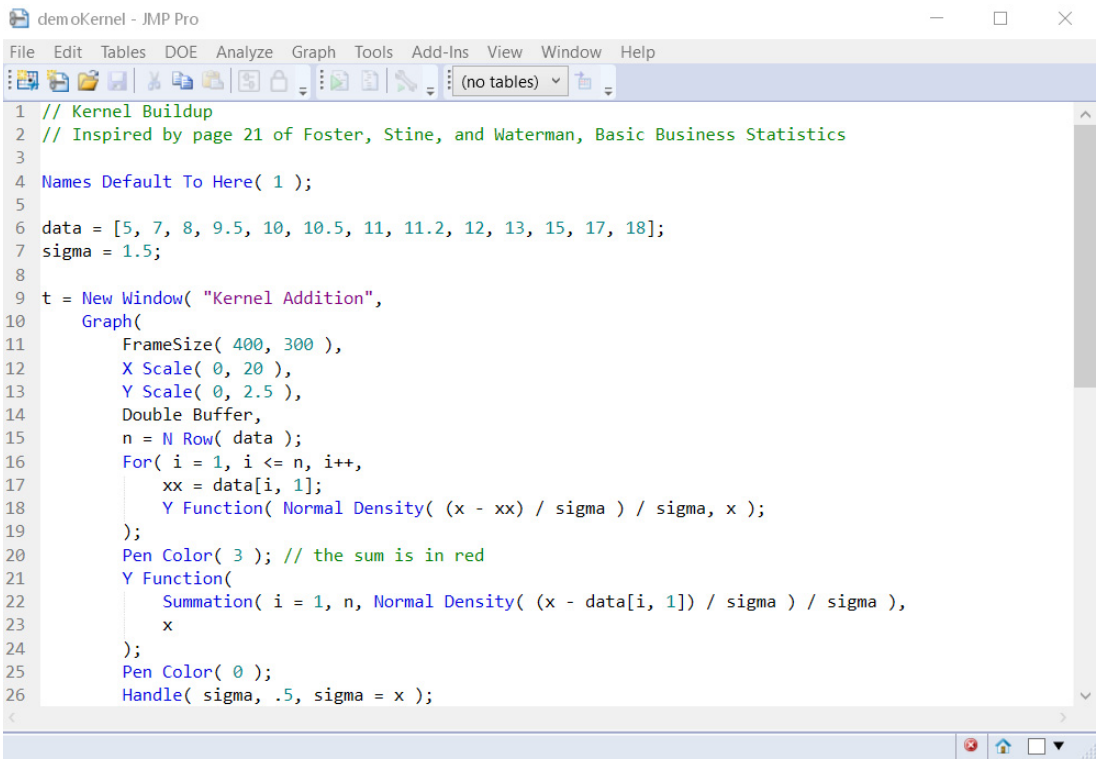
- "Introduction to the Script Editor"
- "Run a Script"
- "Stop a Script"
- "Edit a Script"
- "Color Coding and Themes"
- "Autocomplete Functions"
- "Tooltips"
- "Split a Window"
- "Match Parentheses, Brackets, and Braces"
- "Select a Rectangular Block of Text"
- "Select Fragmented Text"
- "Drag and Drop Text"
- "Find and Replace"
- "Automatic Formatting"
- "Errors Are Marked by Special Characters"
- "Typing in Multiple Rows at Once"
- "Add Code Folding Markers"
- "Advanced Options"
- "Set Preferences for the Script Editor"

Introduction to the Script Editor

The script editor provides a friendly environment for writing and reading JSL scripts. Figure 4.2 shows basic features such as syntax coloring, inline commenting, and automatic formatting. Other common programming options are described later in this section.

Script editor features are also available in the log window and anywhere else that you can edit or write a script (for example, in the Scripting Index or Application Builder).

Figure 4.2 The Script Editor



Tip: Consider setting the *Autosave timeout* value in the General preferences to automatically save open scripts at the specified number of minutes. This autosave value also applies to data tables, journals, scripts, projects, and reports.

Run a Script

To run an entire script, select **Edit > Run Script.**

To run specific lines in a script, select those lines and then select **Edit > Run Script**.

To run specific lines that are not adjacent, press Ctrl, select the lines, and then select **Edit > Run Script**.

On Windows, you can also click in a line or select several lines and press Enter on your numeric keypad.

Run the script automatically when you open it by using one of the following methods:

- Type `//!` on the first line.
- Include `Run JSL( 1 )` in the `Open()` statement:

```
Open( "$SAMPLE_SCRIPTS/scoping.jsl", Run JSL( 1 ) );
```

Stop a Script

To stop the script, press Esc on Windows (or Command-period on macOS). You can also select **Edit > Stop Script**. On macOS, **Edit > Stop Script** is available only when the script is running.

Edit a Script

To edit a script on Windows, press Insert. You can then type directly over (overwrite) any existing JSL code. Note that this feature is not available on macOS.

Color Coding and Themes

On Windows, the default font in the script editor is 10-point Consolas. On macOS, the default script editor font is 13-point Menlo.

JMP applies the following colors in the script window:

- black for text, identifiers (JSL functions), braces, and user macros
- white for the script editor background
- gray for the disabled background and guides
- green for comments
- purple for strings
- maroon for platform names

  Note: If a platform name is used in a message, the JSL messages color is used.

- teal blue for numbers
- dark blue for operator symbols, the first keyword, and JSL messages
- medium blue for function names, second and third keywords, and macros
- red for unknown objects

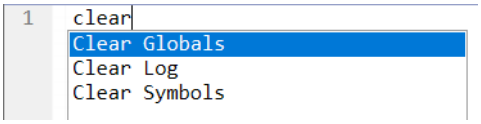
You can customize colors and fonts in the Customize Styles section of the Script Editor preferences. Syntax coloring for JSL, the log, SAS code, JavaScript, Text, JMP Text Analytics, R, Matlab, Python, C, JSON, and SQL are supported. In addition, options for dark and light themes are provided.

Autocomplete Functions

If you do not remember the exact name of a function, use autocompletion to see a list of functions that match what you have typed so far. Type part of the name, and then press Ctrl+Space on Windows (Option+Esc on macOS).

Suppose that you want to clear your JSL variables, but do not remember the command. You can type `clear` and then press Ctrl+Space, to see a list of possible clear commands. Select the command that you want to insert.

Figure 4.3 Autocomplete Example

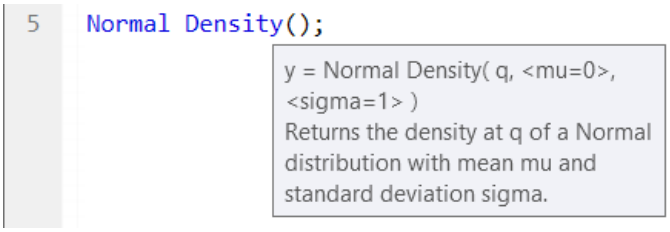


Tooltips

If you are using a function and do not remember the syntax or need more information about it, place the cursor over it to see a brief explanation. This works only with JSL function names, not platform commands, messages, or user-created functions. JSL function names are colored blue in the script editor.

The tooltip shows the syntax, arguments, and a brief explanation of the function (Figure 4.4). The tip also appears in the script editor window status bar.

Figure 4.4 Tooltip for a JSL Function



After running a script, you can also place the cursor over variable names to see their current value. To turn off variable tooltips, deselect **Preferences > Script Editor > Show Variable Value Tips**.

To turn off function tooltips, deselect **Preferences > Script Editor > Show Operator Tips**.

Tip: In addition to showing the tooltip for a function, you can display its Scripting Index entry. Run the script, press Alt, and then double-click the function. A description, sample script, and information about syntax is provided. You can also read help for the topic by running the script, right-clicking the function, and selecting **Help Browser**.

Example of a Tooltip for a JSL Variable

1. Enter and run the following line in a script window:

   ```
   my_variable = 8;
   ```

2. Hover over the variable name after you run the line.

   A tooltip shows the name of the variable and its value: 8.

3. Enter and run the following line:

   ```
   my_variable = "eight";
   ```

4. Hover over the variable name after you run the line.

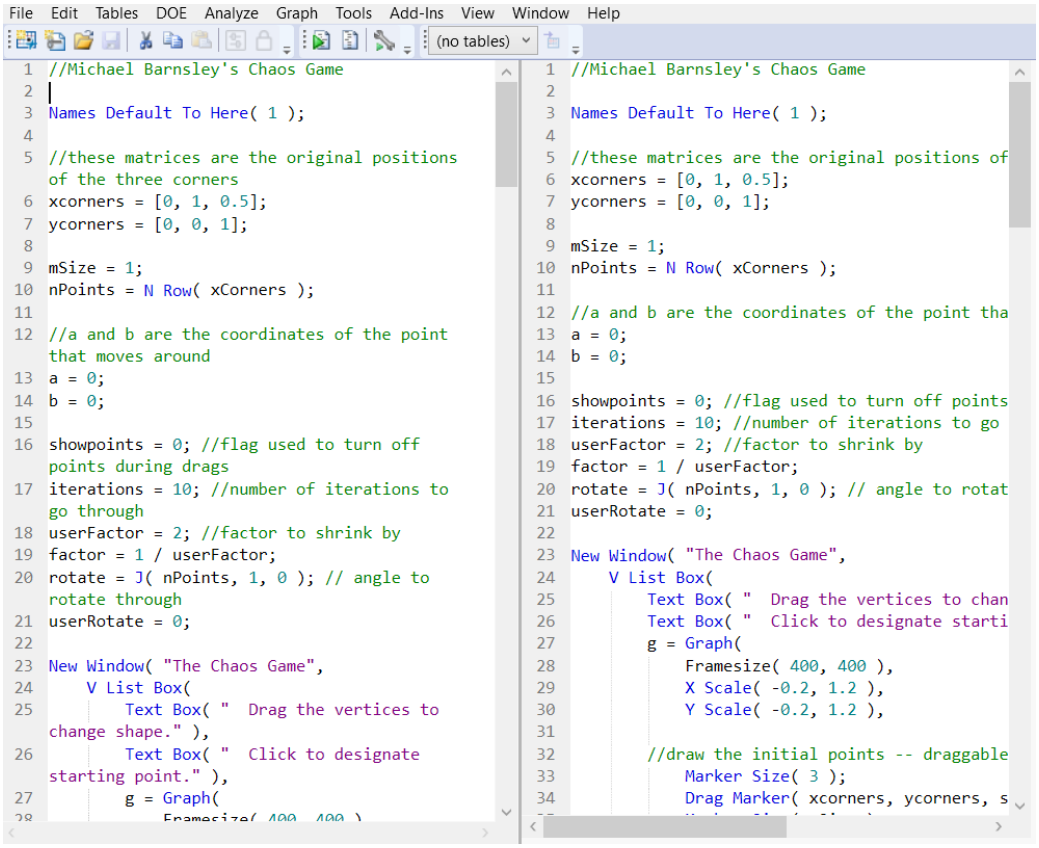
   A tooltip shows the name of the variable and its value: “eight”.

Split a Window

You can split the Script Editor window into two vertical or horizontal windows. This feature enables you to independently scroll through your code in two different places and edit the contents in both. When you make a change in one window, the change is immediately reflected in the other window.

- To split an open Script Editor window, right-click in the window and select **Split > Horizontal** or **Vertical**.
- To revert to a single window, right-click and select **Remove Split**.

Figure 4.5 Example of Splitting a Window Horizontally

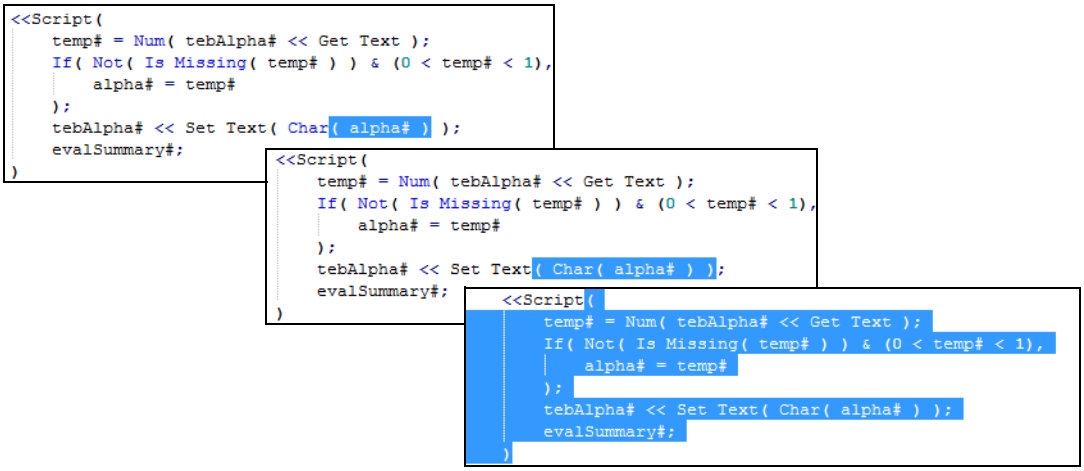


Match Parentheses, Brackets, and Braces

The script editor helps you match *fences* (or parentheses, square brackets, and braces) in the following ways:

- The matching closing fence is added when you type an opening fence.
- When you click next to either an opening or closing fence, the fence and its match are highlighted in blue. If the fence does not have a match, it is highlighted in red.
- If you double-click a fence, everything between the matching fence is selected (including the fences).
- If you put your cursor within an expression and press Ctrl+] on Windows (Command+B on macOS), the entire expression is selected. Fences that enclose the expression are included. Repeat this process to highlight the next-higher expression. Figure 4.6 shows an example.

Figure 4.6 Each Step in Matching Fences



When you type an opening brace, JMP adds the closing brace. Enter code between the braces, type the closing brace, and then your cursor automatically moves after the closing brace that JMP added. This prevents you from accidentally adding an unnecessary closing brace.

You can turn on and off the autocompletion of braces in the JMP preferences. See "Set Preferences for the Script Editor".

Select a Rectangular Block of Text

To select a rectangular block of text, press Alt and drag your cursor from the starting point to the end of the block. You can either copy or cut the text enclosed in the block.

Suppose that you want to the select all of the following code except for the comment marks.

```
// Y( :Y ),
// X( :X ),
```

Select a rectangular portion beginning with Y. When you paste, you get the following code:

```
Y( :Y ),
X( :X ),
```

The rectangular selection inserts returns where needed to maintain the structure of the text. Select `Get Menu Item State` on both lines in the following example.

```
bb << Get Menu Item State(1),
bb << Get Menu Item State(2),
```

When you paste, a return is inserted at the end of each line.

```
Get Menu Item State
Get Menu Item State
```

Select Fragmented Text

To select text that is not contiguous, press Ctrl on Windows or Command on macOS and drag your cursor over the text. Continue this action for any other text that you want to select. You can then copy and paste your selection into a new script or you can run the selected text. Text will be pasted or run in the order it was selected.

Drag and Drop Text

You can drag and drop text within a script editor window or between windows or from a data table into a script editor window. On Windows, pressing Ctrl before dragging and dropping copies the text. On macOS, the text is copied by default.

Drag and drop text in any of these ways:

- Select a row or column, pause, and then drop it into the script editor window.
- Double-click text in a text field and then drop it into the script editor window. Examples are text in a data table cell and any other selectable text.

On Windows, you can also drag and drop text into a minimized window.

1. Drag the text over the Home Window button in the lower right corner of the window.
   The Home Window appears.
2. In the Home Window list, drag the text over the destination window.
   That window appears.
3. Drop the text where you want it.

Find and Replace

Many find and replace options are available in the script editor, including the support of regular expressions. For example, searching with the following regular expression:

```
get.*name
```

returns messages such as “Get Button Name”, and “GetFontName”

Basic regular expressions such as `^` and `$` (which match the start of line and end of line) and `\n` (which matches a carriage return) are also supported.

See *Using JMP* for more information about the Search options.

Automatic Formatting

The script editor can format a script for easier reading. Any generated script (for example, by saving a platform script) is automatically formatted with tabs and returns in appropriate places.

You can also reformat individual scripts that are difficult to read (for example, scripts in which all commands are strung together with no whitespace characters). From the **Edit** menu, select **Reformat Script**.

Tip: This command alerts you if your script is badly formed (for example, if your parentheses are not matched).

Errors Are Marked by Special Characters

After you run a script, the /*###*/ characters in the log indicate the location of an error in the script execution. The following example shows a script in which the height column name is misspelled.

```
Names Default To Here( 1 );
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt:heihgt[3]; // misspelled column name
```

After you run the script, the log reads:

```
Scoped data table access requires a data table column or variable in access or
   evaluation of 'dt:heihgt' , dt:heihgt/*###*/
In the following script, error marked by /*###*/
Names Default To Here( 1 );
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt:heihgt/*###*/[3];
```

/*###*/ directly follows the misspelled height argument.

Typing in Multiple Rows at Once

When you edit a script, you can type in multiple rows at once instead of manually updating the same code on multiple lines. For example, in the following script, you want to change the s variable to aString. You can easily change the first instance of s to aString. To change the other instances at once, hold down Alt and select the column of s variables. Type aString once.

```
s = "";
For( i = 1, i <= n, i++,
   s += "Hello";
   s += "Number";
```

```
    s += Char( i );
);
```

The script then looks like this:

```
aString = "";
For( i = 1, i <= n, i++,
   aString += "Hello";
   aString += "Number";
   aString += Char( i );
);
```

Note: You can type in multiple rows at once, but you can type only in the same column. In the preceding example, the `s` variables are in the same column.

Add Code Folding Markers

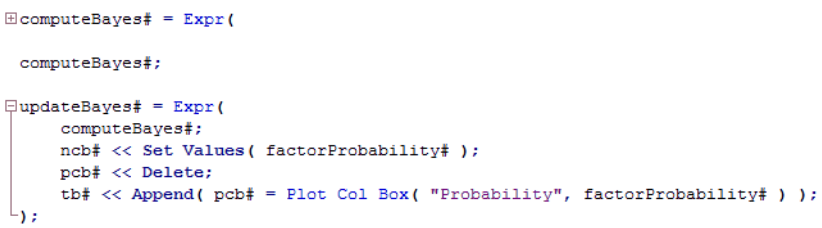
You can add code folding markers that show the beginning and the end of the code block, enabling you to collapse and expand code inside stand-alone functions.

To turn on this feature, select **Code folding** in the Script Editor preferences. Then you can expand and collapse blocks of code by right-clicking on a script and selecting **Advanced > Expand All** or **Collapse All**.

Tip: To fold or expand all code in the entire script, press Shift+Ctrl and then click a code marker.

After you select this preference, `Function` and `Expr` expressions are foldable. See "Add More Folding Keywords" for more information about adding folding markers to other expressions.

Figure 4.7 Code Folding Markers Shown in a Script



By default, code does not remain collapsed after you save the script and restart JMP. To save the state of the folded code, select **Save and restore document state information** in the Script Editor preferences.

Add More Folding Keywords

Custom code folding is supported for other stand-alone functions as shown in the following example:

```
{"If", "For", "For Each Row", "While", "Try", "New Window", "V List Box",
  "H List Box"}
```

JMP supports multiple keyword lists. A system administrator can define a set of keywords in jmpKeywords.jsl and save the script in C:/ProgramData/SAS/JMP/ or designated directory listed below. You save your version of jmpKeywords.jsl in your C:/Users/<user>/Documents/ folder. JMP merges all keyword lists from the designated directories.

On Windows, the following directories are examined in the order listed:

- C:/ProgramData/SAS/JMP/16/
- C:/ProgramData/SAS/JMP/
- C:/Users/<user>/AppData/Roaming/SAS/JMP/16/
- C:/Users/<user>/AppData/Roaming/SAS/JMP/
- C:/Users/<user>/Documents/

On macOS, the following directories are examined in the order listed:

- /Library/Application Support/JMP/16/
- /Library/Application Support/JMP/
- ~/Library/Application Support/JMP/16/
- ~/Library/Application Support/JMP/
- ~/Documents/

Note that jmpKeywords.jsl is stored in the designated JMP directory, even if you are using JMP Pro.

Notes:

- The list in jmpKeywords.jsl is case insensitive.
- Code folding is not supported for messages, platforms, user-defined functions, and comments.
- After you edit and save the list in jmpKeywords.jsl, turn the **Allow additional code folding keywords** preference off and then back on for the changes to take effect. Messages in the log indicate that the keywords were loaded.

Advanced Options

Right-clicking on selected text in the Script Editor provides the following Advanced options:

Expand All (Appears only if JSL code folding is on.) Expands all blocks of code.

Collapse All (Appears only if JSL code folding is on.) Collapses all blocks of code.

Comment Block Makes the selected text comments.

Uncomment Block Uncomments the selected comments.

Make Uppercase Changes all selected text to uppercase.

Make Lowercase Changes all selected text to lowercase.

Set Preferences for the Script Editor

In the JMP preferences, customize the script editor settings such as the font, colors, and spacing options.

Setting the Fonts

1. Select **File > Preferences**.
2. Select the **Fonts** group.
3. Click **Mono** to set the font for the script editor.

For more information about font preferences, see *Using JMP*.

Set Script Editor Preferences

Select **File > Preferences > Script Editor** to further customize the editor. For more information about script editor preferences, see *Using JMP*.

Work with the Log

The Log in JMP has two modes:

- "Enhanced Log in JMP"
- "Text Log in JMP"

You can switch between the two modes by selecting **File > Preferences > Log**.

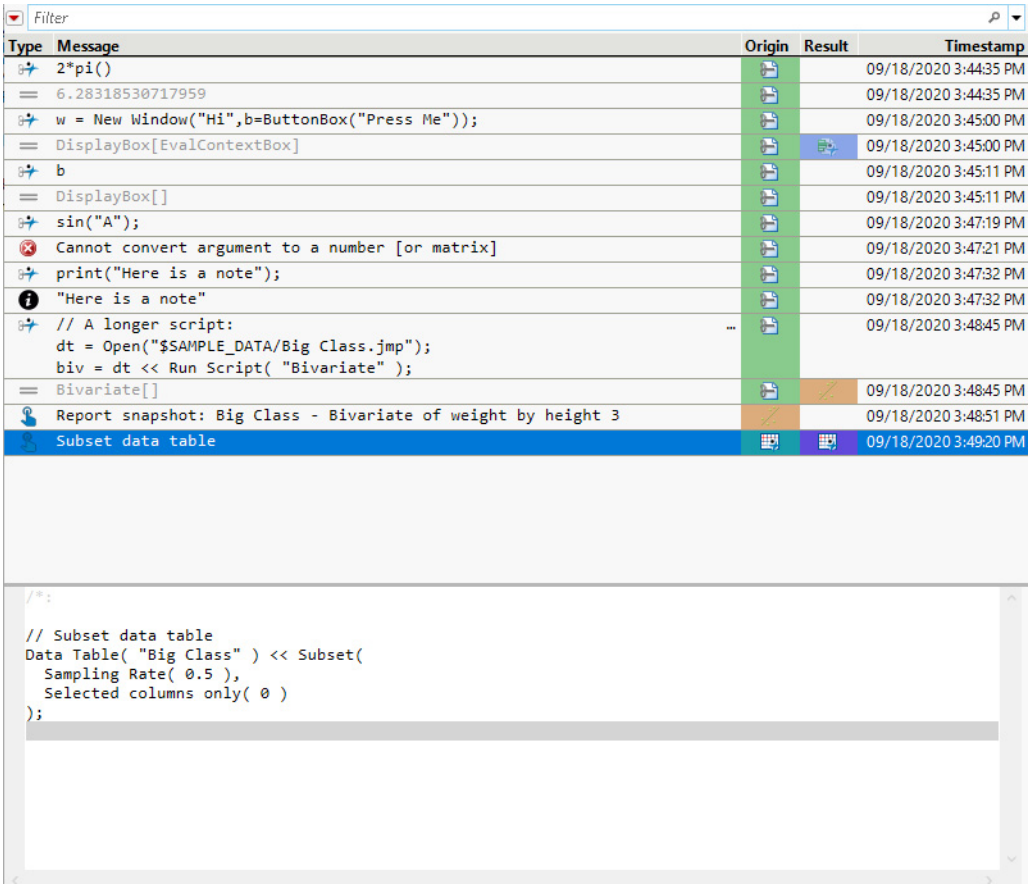
Note: The log shows the steps you take during data cleanup. For example, if you add a column property or import the data into Excel, then the JSL expressions for those actions are captured in the log. This enables you to clean up the data more efficiently when you receive a new data set.

Enhanced Log in JMP

The Enhanced Log is a tabular view of the events in a JMP session. To view the JMP Log, select **View > Log** (**Window > Log** on macOS). By default, the log in JMP appears in Enhanced mode. The log window contains events, or actions, in the rows of the table and the corresponding JSL code and output in a separate text pane. This tabular view of the events is easy to navigate, displays errors and warnings, and can help you create a JSL script of the actions in a JMP session. For an example of using the log to generate a JSL script for recorded actions, see "Example of Recording Actions in the Enhanced Log".

Note: When you used the embedded log in a script window, the embedded log does not use the Enhanced Log mode. Further, any script output from a script window with an embedded log does not appear in the Enhanced Log.

Figure 4.8 Enhanced Log



Each row of the tabular view contains information in the following columns:

Type Contains an icon that represents message type. For more information about the types of messages, see "Filter".

> **Tip:** You can hover over an icon in the Type column to see the type of message that a particular icon represents.

Message Contains a description of the event or result. Among other things, this could be a recorded action, JSL that was submitted, the result of JSL that was submitted, or an error message. Each description of a logged item or result is limited to three lines of text; longer items are denoted with an Ellipsis icon ··· in the upper right corner of the Message section of the row. The full text for longer items appears in the text pane at the bottom of the log window.

Origin Contains an icon that shows the type of JMP window from which the logged item or result was generated. If the Color by Window option is selected in the red triangle menu next to Filter, logged items generated by the same JMP window are specified by a common color. You can click on the icon in the Origin column to bring the corresponding window to the front.

> **Note:** Not all rows in the log table have an icon in the Origin column.

Result Contains an icon that shows the type of JMP window that contains the result. If the Color by Window option is selected, logged items generated by the same JMP window are specified by a common color. You can click on the icon in the Result column to bring the corresponding window to the front.

> **Note:** Not all rows in the log table have an icon in the Result column.

Timestamp Contains a date-time value that indicates when the logged item was generated.

Enhanced Log Options

The red triangle menu in the Enhanced Log contains the following options:

Go To (Available only when one or more rows of the log table are selected.) Brings the window corresponding to the selected row in the log table to the front. If an Origin window and a Result window both appear in the selected row, both windows are brought to the front with the Result window on top of the Origin window.

Run Scripts (Available only when one or more rows of the log table that contain a scriptable action are selected.) Runs the JSL script that corresponds to the selected rows in the log table.

Color by Window Specifies whether the rows in the log table are colored by the JMP window that created the row of the log output.

Clear (Available only when one or more rows of the log table are selected.) Removes the selected row or rows from the log table.

Caution: This action cannot be undone.

Clear Log Removes all rows from the log table.

Caution: This action cannot be undone.

Filter Contains the following options to filter what types of logged events appear in the log table.

Result Shows or hides rows that correspond to the results of running JSL.

Script Shows or hides rows that correspond to submitted JSL.

Log Shows or hides rows that correspond to informational log messages.

Warning Shows or hides rows that correspond to warning messages in the log.

Error Shows or hides rows that correspond to error messages in the log.

Action Shows or hides rows in the log table that correspond to interactive JMP actions. These actions include data table operations, platform launches, and closing report windows. You can customize which actions are shown in the log table by selecting specific options in the Action Recording submenu.

Action Recording Contains the following options to filter what types of interactive JMP actions appear in the log table.

Data Table Operations Shows or hides rows that correspond to manipulations of a data table. These actions can include row and column operations, as well as actions in the Tables, Rows, and Cols menus.

Platform Launch Shows or hides rows that correspond to launching a JMP platform.

Report Snapshot on Close Shows or hides rows that correspond to a JSL script that is saved when a report window is closed.

Save Script Contains the following options that save a script that reproduces all selected rows in the log table. If no rows are selected, these options save a script to reproduce all events recorded in the log table.

To Script Window Saves the script to a script window.

To Clipboard Saves the script to the clipboard.

Make Into Data Table Creates a data table that contains all rows from the log table. If rows are selected in the log table, they are also selected in the data table that is created.

Filter the Results in the Enhanced Log

To the right of the red triangle menu, use the search field to filter the items in the log table view.

Click the down arrow button next to the search box to refine your search.

Contains Terms Returns items that contain a part of the search criteria. A search for "ease oom" returns messages such as "Release Zoom".

Contains Phrase Returns items that contain the exact search criteria. A search for "text box" returns entries that contain "text" followed directly by "box" (for example, "Context Box" and "Text Box").

Starts With Phrase Returns items that start with the search criteria.

Ends With Phrase Returns items that end with the search criteria.

Whole Phrase Returns items that consist of the entire string. A search for "text box" returns entries that contain only "text box".

Regular Expression Enables you to use the wildcard (*) and period (.) in the search box. Searching for "get.*name" looks for items that contain "get" followed by one or more words. It returns "Get Color Theme Names", "Get Name Info", and "Get Effect Names", and so on.

Invert Result Returns items that do not match the search criteria.

Match All Terms Returns items that contain both strings. A search for "t test" returns elements that contain either or both of the search strings: "Pat Test", "Shortest Edit Script" and "Paired t test".

Ignore Case Ignores the case in the search criteria.

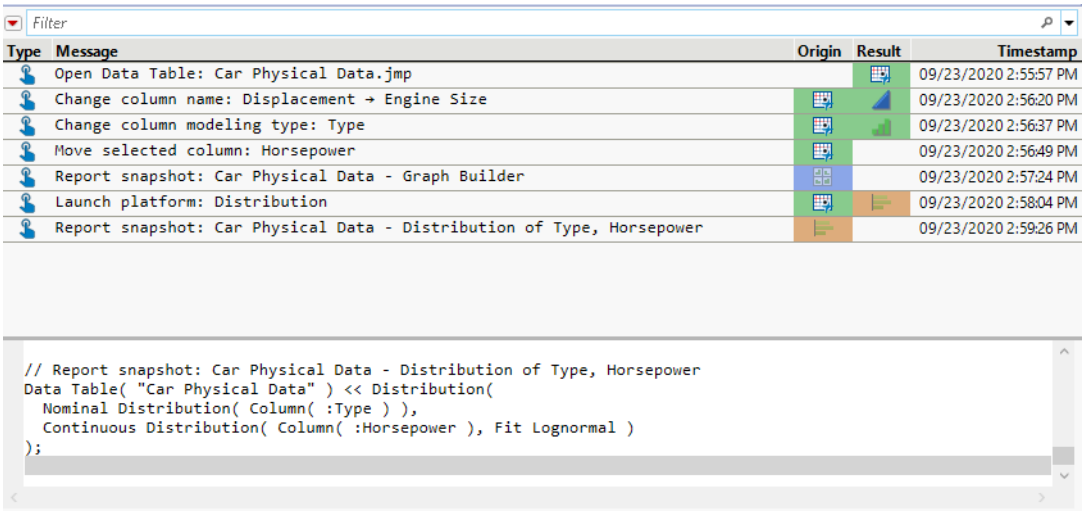
Match Whole Words Returns items that contain each word in the string based on the Match All Terms setting. If you search for "data filter", and Match All Terms is selected, entries that contain both "data" and "filter" are returned.

Example of Recording Actions in the Enhanced Log

You can use the Enhanced Log to record many of the interactive actions that you perform for data cleanup and analysis. In this example, the Enhanced Log is used to generate a JSL script for some data manipulation and analysis actions.

1. Select **View > Log**.
2. Click the red triangle next to the Filter box and select **Clear Log**.
3. Select **Help > Sample Data Library** and open Car Physical Data.jmp.
4. Right-click Displacement in the Columns panel in the data table, and select **Column Info**.
5. Type Engine Size for the Column Name. Click **OK**.
6. Right-click Type in the Columns panel in the data table, and select **Column Info**.
7. Select Ordinal for the Modeling Type. Click **OK**.
8. Click Horsepower in the Columns panel in the data table and drag it above Weight.
9. Select **Graph > Graph Builder**.
10. Drag Horsepower to the **Y** drop zone.
11. Drag Engine Size to the **X** drop zone.
12. Close the Graph Builder window.
13. Select **Analyze > Distribution**.
14. Select Type and Horsepower and click **Y, Columns**.
15. Click **OK**.
16. Click the red triangle next to Horsepower and select **Continuous Fit > Fit Lognormal**.
17. Close the Distribution window.
18. Select **View > Log**.

Figure 4.9 Enhanced Log



19. Press Ctrl and select the first, second, third, and last rows of the log table.
20. Click the red triangle next to the Filter box and select **Save Script > To Script Window**.

The script to reproduce the selected actions appears in a new script window. You can save this script to reuse later or use it as a starting point for a more complex analysis.

Text Log in JMP

The Text Log in JMP is a text-based log of JSL scripts that have been run during a JMP session and the results of those scripts. When the log is open, output appears in the log after you run a script. The script that was run is shaded in the log, and the corresponding output appears beneath it. To view the log in Text mode, select Text for the Mode preference after selecting **File > Preferences > Log**.

Syntax and compatibility errors are reported in the log, including the line number, script filename, and code that JMP could not process. Many interactive JMP actions are also recorded in the log.

Tips:

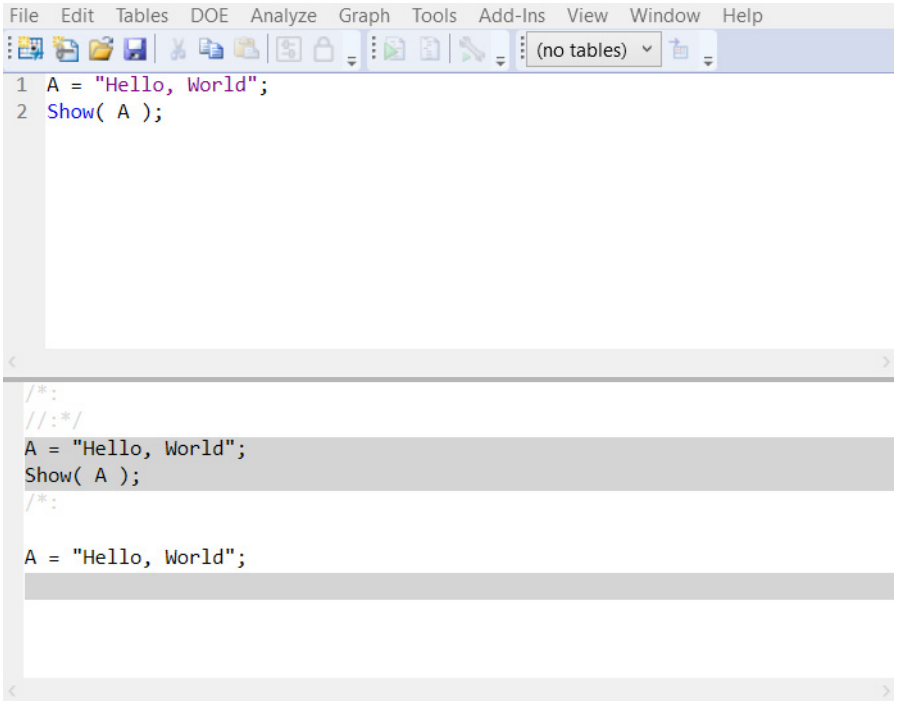
- To omit compatibility warnings from the log, deselect **Show log warnings for JSL compatibility changes in 12** in the JMP General preferences.
- To choose which actions are recorded in the log, you can use the Action Recording preferences after selecting **File > Preferences > Log**.

Show the Text Log in the Script Window

You can view the log inside the script window by right-clicking and selecting **Show Embedded Log**. To view the embedded log by default, select **Show embedded log on script window open** in the Script Editor preferences. This option makes it easy to edit and run a script, quickly see the results of your changes, and then continue to develop the script. The embedded log uses the Text log mode. See "Text Log in JMP".

The embedded log always appears in the Scripting Index script window, but it is not available in Application Builder and the Debugger.

Figure 4.10 The Script Window with an Embedded Text Log



Debug or Profile Scripts

In an open script, click the **Debug Script** button (or select **Edit > Debug Script**) to show the script in the JSL Debugger window. You can also use a keyboard shortcut:

- Press Ctrl+Shift+R (Windows).
- Press Shift+Command+R (macOS).

The JSL Debugger helps identify the point at which a script causes an error or fails. Rather than commenting out portions of the script or adding `Print()` expressions, you can use the Debugger to find the problem.

Once the JSL Debugger appears, you can continue in this mode, or you can click the **Profile JSL Script** button to move into the JSL Profiler mode. The JSL Profiler helps with optimization. You can profile your scripts during execution to see how much time is spent executing particular lines or how many times a particular line is executed.

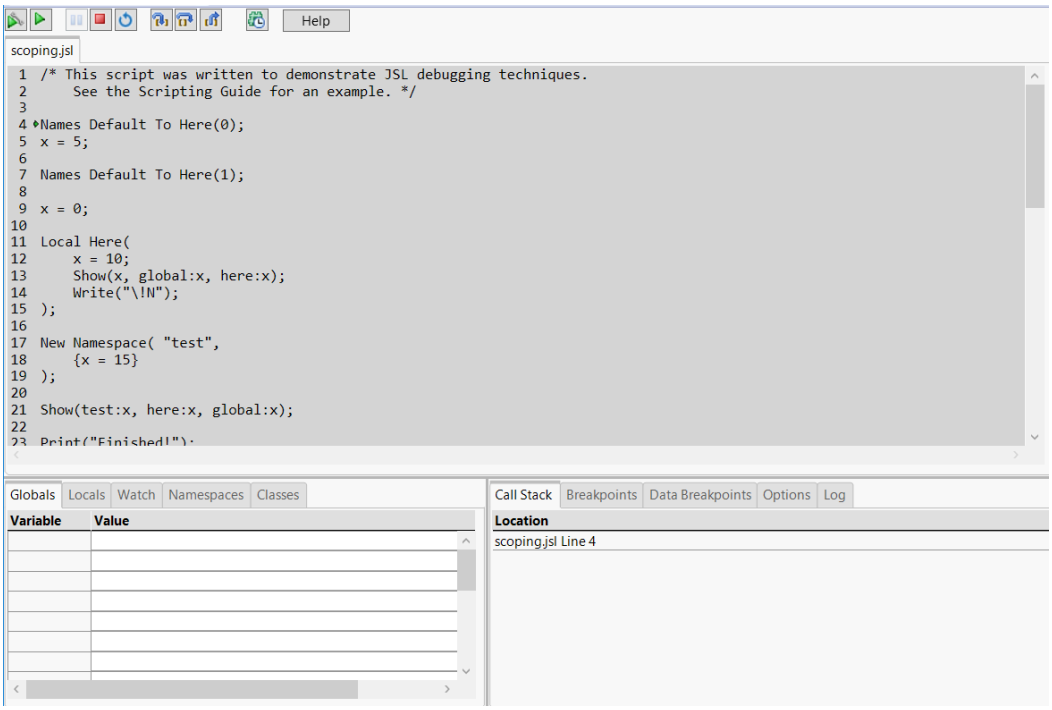
Tip: To debug a script automatically when you open it, include `Debug JSL( 1 )` in the `Open()` statement:

```
Open( "$SAMPLE_SCRIPTS/scoping.jsl", Debug JSL( 1 ) );
```

Debugger and Profiler Window

The Debugger opens in a new instance of JMP (Figure 4.11). The original instance is inoperable until the script produces something that requires interaction. At that point, the Debugger window becomes inoperable until you perform whatever action is required. Then control is returned to the Debugger. Close the Debugger to work again in the original instance of JMP.

Figure 4.11 The Debugger Window



Use the buttons at the top to control the Debugger or the JSL Profiler. One or more scripts that you are debugging or profiling are shown in tabs. If your script includes other scripts, each one opens in a new tab.

Tabs in the bottom portion of the Debugger provide options to view variables, namespaces, the log, and the current execution point; work with breakpoints; and set options.

Tip: To view or edit long values on a tab in the Debugger, right-click the value and select **Edit**. The code opens in a script editor.

Using the Execution Buttons

Use the buttons at the top to control execution of the script within the Debugger or JSL Profiler.

Table 4.1 Description of the Debugger Buttons

Button	Button Name	Action
	Run	Runs the script in the Debugger until it reaches either a breakpoint or the end of the script. **Note:** If there is an error in the script, an abbreviated error message is displayed at the top of the window. Click **More** to see the complete error message.
	Run without breakpoints Run profiler	Runs the script through the end without stopping.
	Break All	If the script is busy, click Break All to stop all action in the script and return to the Debugger or JSL Profiler (for example, if you are in a very long loop). The Debugger or JSL Profiler might not be able to break execution if the executing script is waiting on some interactive user action, such as completing a window or interacting with an opened window.
	Stop	Stops debugging the script and exits the Debugger or the JSL Profiler.
	Restart	Closes the current Debugger session and opens a new session.

Table 4.1 Description of the Debugger Buttons *(Continued)*

Button	Button Name	Action
	Step Into	Lets you step into a function or an included file. Otherwise, it behaves the same as Step Over.
	Step Over	Runs all expressions on a single line, or a complex expression that spans multiple lines, without stepping into a called expression, function, or `Include()` file.
	Step Out	Runs the current script or function to a breakpoint or the end and returns to the calling point. If you are in the main script and the Debugger reaches the end, a message appears: Program execution terminated. The Debugger remains open in order for you to inspect the final program conditions.
	Profile JSL Script	Opens the JSL Profiler. (Press the Run profiler button to start the JSL Profiler.) Use the JSL Profiler to see how much time is spent executing particular lines or how many times a particular line is executed. **Notes:** • You can switch back and forth between the Debugger and JSL Profiler modes only prior to the start of the program. • Some of the debugger buttons are disabled when profiling. • All breakpoints are disabled when running in the JSL Profiler mode.
	Show Profile by Line Count	Shows the number of times each line is executed.
	Show Profile by Time	Shows how much time is spent executing a line.
	Show Profile by Count	For line counts, shows the number of times the line is executed. For time, shows the number of microseconds (or milliseconds or seconds) the line takes to complete.
	Show Profile by Percent	For line counts, shows the individual line count divided by the total line count. For time, shows the percentage of time spent on an individual line (line time/total time*100).
	Time Units	Sets the time unit to microseconds, milliseconds, or seconds. Available in the JSL Profiler after you click the Run profiler button .

Table 4.1 Description of the Debugger Buttons *(Continued)*

Button	Button Name	Action
	Color Theme	Sets the color of the shading for the JSL Profiler. Available in the JSL Profiler after you click the Run profiler button .

Variable Lists

The tabs on the bottom left of the Debugger let you examine global variables, local variables, watch variables, and variables within namespaces.

Globals The Globals tab lists all global variables and updates their values as you step through the script. Each variable is added as it is initialized. If there are already global variables defined from running earlier scripts, they will be listed with their current values when you start the Debugger. See "View Variables".

Locals The Locals tab lists all variables by scope and updates their values as you step through the script. Select a scope in the menu. See "View Variables".

Watch If there is a particular variable or value of an expression whose values you want to watch as you step through the code, you can add them here. This is particularly useful if your script uses many variables that might be difficult to watch in the Globals or Locals lists. See "Work with Watches".

Namespaces As namespaces are defined, they are added to the menu. Select a namespace to view any variables and their values used within the namespace. See "View Variables".

Classes The Classes tab lists classes that are defined in the script. See "Classes".

Debugger Options

The tabs on the bottom right of the Debugger let you view the call stack, work with breakpoints, set options, and view the log.

Call Stack The call stack lists the current execution point in scripts and functions. The main script is always the first script listed. If you call a function, the function is added on top of the calling script. Likewise, any included files are added to the top of the list as you step through them. When you exit any function or script, it is removed from the list and you return to the next one in the list. The current line numbers are updated as you step through.

Double-click a row in the call stack to move the cursor to the specified line.

Breakpoints Add, edit, delete, and disable or enable breakpoints on lines. See "Work with Breakpoints". You can also double-click a row on the Breakpoints tab to move the cursor to the specified line.

Data Breakpoints Add, edit, delete, and disable or enable breakpoints in variables. Data breakpoints watch for changes in variables, unlike breakpoints, which watch for changes in lines.

Options Set the Debugger preferences interactively on this tab. See "Modify Preferences in Debugger".

Log The log from the script that you are debugging is shown on this tab.

Work with Breakpoints

A *breakpoint* interrupts the execution of a script. Although you can step through a script line by line, this can be tedious and lengthy for a long or complex script. You can set breakpoints at places of interest and simply run the script in the Debugger. The script is run normally until a breakpoint is reached. At the breakpoint, the Debugger stops executing the script so that you can look at the values of variables or start stepping line by line.

Breakpoints that appear on the Breakpoints tab in the lower right corner watch for changes in a line. Breakpoints that appear on the Data Breakpoints tab watch for changes in variables.

JMP preserves breakpoints across sessions. So when you close and reopen JMP, the breakpoints still appear.

Tip: Turn on line numbers by right-clicking in the script and selecting **Show Line Numbers**. You can also show line numbers by default in all scripts by modifying the Script Editor preferences.

Create a Breakpoint

When creating a breakpoint, you can specify settings such as conditions and break behavior. To do so, click on the Breakpoints tab or Data Breakpoints tab, and then enter the breakpoint information.

Otherwise, create a quick breakpoint by doing one of the following:

- In the Debugger margin, click the appropriate line (to the right of the line number if displayed).
- In the Debugger margin, right-click in the margin where you want the breakpoint and select **Set Breakpoint**.

The red breakpoint icon appears where you inserted the breakpoint and on the Breakpoints tab.

Delete Breakpoints

Do one of the following:

- In the Debugger margin, click the breakpoint icon.
- In the Debugger margin, right-click the breakpoint icon and select **Clear Breakpoint**.
- On the Breakpoints tab, select the breakpoint and then click .
- On the Breakpoints tab, click to delete all breakpoints (not just the selected breakpoints).

The red breakpoint icon is removed, and the breakpoint no longer appears on the Breakpoints tab.

Disable and Enable Breakpoints

Disabling a breakpoint is helpful when you potentially fix a problem and then want to see whether the script will run correctly past that breakpoint. You can then enable the breakpoint when necessary rather than re-creating it.

Do one of the following:

- In the Debugger margin, right-click the breakpoint icon and select **Enable Breakpoint** or **Disable Breakpoint**.
- On the Breakpoints tab, select or deselect the breakpoint's check box.
- On the Breakpoints tab, click to disable or enable all breakpoints.

A disabled breakpoint turns white; enabled breakpoints are shaded red.

Specify and Clear Conditional Expressions on Breakpoints

Setting a condition on a breakpoint is an alternative to single-stepping through code. Rather than single-step and view the variables for each expression, you specify that the script break only when a condition is met. Then you can step through the code and figure out where the problem arises.

Suppose that a calculation in your script is incorrect, and you suspect the problem occurs when `i==19`. Set a conditional breakpoint for `i==18`. The Debugger will run until that condition is met, then you can step through the code to identify the problem.

Specify a Breakpoint Condition

1. Right-click the breakpoint icon and select **Edit Breakpoint**.
2. On the Condition tab, select **Condition** and enter the conditional expression.
3. Specify whether to break when the expression **Is true** or **Has Changed**.
4. Click **OK**.

Disable or Enable a Condition

1. Right-click the breakpoint icon and select **Edit Breakpoint**.
2. On the Condition tab, deselect or select **Condition**.

Delete a Condition

On the Breakpoints tab, click in the breakpoint's Condition field and press Delete.

Specify Break Options

Right-clicking the breakpoint and selecting **Edit Breakpoint** provides a quick way to manage breakpoint behavior. Alternatively, select the breakpoint on the Breakpoints tab and click . Both methods display the Breakpoint Information window, where you customize settings on the Hit Count and Action tabs.

Change the Hit Count

You can control the number of times a breakpoint must be hit and when the break occurs. For example, to break when the condition is met twice, select **break when the hit count is equal to** and type 2 on the Hit Count tab.

Define an Action

You also have the option of defining a JSL expression or script that the Debugger executes when a breakpoint is hit and execution has stopped. This script is called an *action*. On the Action tab, enter the JSL expression to be executed.

Run the Script to the Cursor

When you right-click and select **Run To Cursor**, all expressions before the location of the cursor are executed. Select this option when you want to see values only up to the current line. To see values when each expression is executed, use the stepping options.

Tips for Setting Breakpoints

- If you do not want to watch for errors in a specific loop, set a breakpoint after the loop ends. The Debugger will hit the next breakpoint rather than stepping through each line of the loop.
- Avoid inserting a breakpoint in lines that do not trigger an action (such as comments, blank lines, and end parentheses). Debugger will not break on these lines.
- When you insert breakpoints, close Debugger, and edit the script, the breakpoints remain on the original line numbers. You might need to delete and then reinsert the breakpoints.

- Breakpoints are remembered across Debugger sessions. This means that your breakpoints list includes breakpoints that have been set in all scripts, not just the script that you are currently debugging.
- Breakpoints are remembered by the Debugger session, not by each script. This means that breakpoints are listed even for scripts that have been moved or deleted.
- On the Breakpoints tab, click to remove all breakpoints in scripts whether they are currently open or not, or for scripts that no longer exist.

View Variables

The variables lists are populated as they appear in the script. Their values are updated every time the script changes them. If you are uncertain why a variable has a particular value when you run your script, you can watch its value at every step to see what happens.

You can also assign the variables whatever value you want. For example, if you are stepping through a `For()` loop but are interested only in what happens starting with a particular iteration, you can assign your iterating variable that value. Step through the first part of the loop that initializes the iteration variable and then assign it the value that you want in the variable list at the bottom. Then when you step through, the loop begins executing at that point.

Tips for Managing Variables

- If you have run several scripts using the global scope, you might want to clear or delete global variables. This makes the list of variables in the Debugger shorter and relevant. Use the `Delete Symbols()` function to do so. You can also close JMP and restart to clear the space.
- If your script uses so many variables that they are difficult to find and watch in the variable lists, add watches for the specific variables in which you are interested.
- To view or edit long values on a tab in the Debugger, right-click the value and select **Edit**. The code opens in a script editor.

Work with Watches

JMP preserves the Watch variables across sessions. So when you close and reopen JMP, the Watch variables are still listed on the Watch tab.

Create a Watch

- On the Watch tab, click and enter the value in the window.

- In the Debugger, right-click the line that you want to watch, select **Add Watch**, and then enter the variable name in the window.
- In the Debugger, place the cursor in or next to a variable name (or select the variable name), right-click, and select **Add Watch**.
- On the Watch tab, enter the variable in an empty **Variable** field.

Modify a Watch

Do one of the following on the Watch tab to enter a new value:

- Select the watch and click .
- Click in the Variable field and enter a new value.

Delete Watches

Do one of the following on the Watch tab to delete watches:

- Select the watch and then click .
- Click to remove all watches.

Modify Preferences in Debugger

The Debugger lets you change preferences as you work in the Debugger. Select the Options tab to find the following settings:

Show Line Numbers Shows or hides the line numbers for the script in the Debugger.

Break on Multiple Statements Per Line Stops executing the script between each expression in a single line.

Break On Throw Breaks when the script executes the `Throw()` function. For example, `Throw()` might be enclosed in a `Try()` expression. The Debugger breaks on `Throw()` instead of continuing through the rest of the expression. This lets you identify where the problem occurred in the script and then return to debugging.

Break On Execution Error Stops executing the script when the error occurs rather than closing the Debugger.

Warn On Assignment In Condition Shows a warning when you enter a breakpoint condition that contains the assignment. For example, if you have a breakpoint on `x = 1` and add the condition `x = 1` to the breakpoint, you are prompted to verify the assignment of `x`.

Enter Debugger Upon Termination Keeps the Debugger open after a JSL program terminates execution. On by default, this option lets you examine attributes of the executed program.

Break For Compatibility Warnings Breaks when a JSL compatibility issue is found.

Persistent Debugger Sessions

JMP saves all breakpoints and watches until you delete them. Other user-specific settings, such as column widths on the tabs and the Debugger window size, persist between sessions of JMP.

These settings are stored in a file named JMP.jdeb, the location of which is defined in the USER_APPDATA variable:

- Windows 7 and higher: "/C:/Users/<username>/AppData/Roaming/SAS/JMP/16/"
- macOS: "/Users/<username>/Library/Application Support/JMP/16/"

As usual, the values of local variables, global variables, and namespaces clear when you close and reopen JMP.

Note: On Windows, the paths differ based on the JMP edition. In JMP Pro, the path refers to "JMPPro".

Examples of Debugging and Profiling Scripts

This section includes examples of setting breakpoints to watch variables; stepping into, over, and out of expressions; watching variables in different scopes and namespaces; debugging interactive scripts; and profiling scripts with the JSL Profiler.

Example scripts are located in the JMP Samples/Scripts folder.

Tip: Make sure that **Show Line Numbers** is selected on the Debugger Options tab before proceeding.

Using Breakpoints and Watching Global Variables

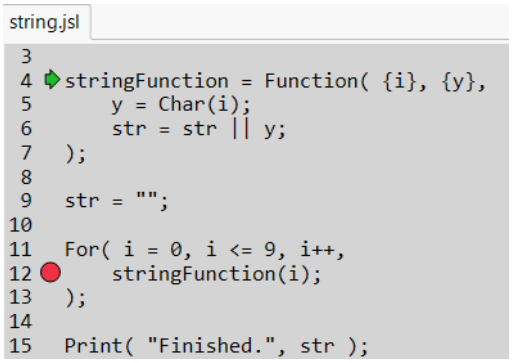
The following example shows how to set a breakpoint in a loop and watch variables change through each iteration of the loop.

1. Open the string.jsl sample script and click the **Debug Script** button.
2. Click in the margin for line 12 to add a breakpoint.

   You should have a breakpoint for the following expression inside the For() loop:

   ```
   stringFunction(i);
   ```

Figure 4.12 Set the Breakpoint



3. Click **Run**.

   The first two expressions are evaluated:

   – `stringFunction` is defined as a function.

   – `str` is defined as an empty string.

   Both variables and their types and values have been added to the Globals list. In addition, the `For()` loop has been evaluated up to the line with the breakpoint, shown in Figure 4.12.

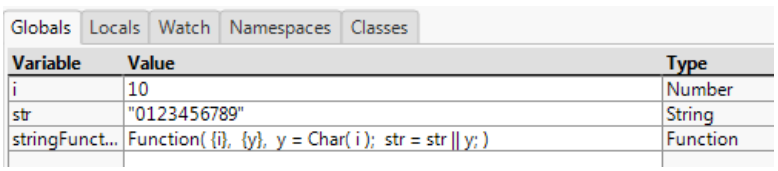
   – `i` has been assigned to 0.

   – `i` and its value and type have been added to the Globals list.

   – `i` has been determined to be less than or equal to 9.

   – `stringFunction()` has not yet been called.

Figure 4.13 View the Initial Global Variables



Tip: The initial value of `i` is 10 because the script was run before entering the Debugger; it's showing the value of `i` after running the script (10). Start a new instance of JMP, open the script without running it, and enter the Debugger immediately.

4. Click **Run** again.

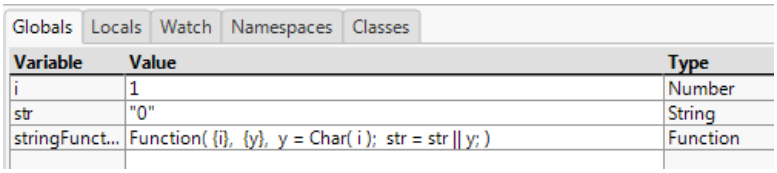
   The script runs until it hits the breakpoint. The results are shown in Figure 4.14.

   – `stringFunction()` is called, evaluated, and returns to the loop.

   – `i` is incremented and determined to be less than or equal to 9.

– In the Globals list, `i` is now 1 and `str` is now "0".

Figure 4.14 Global Variables at First Breakpoint



5. Click **Run** again.

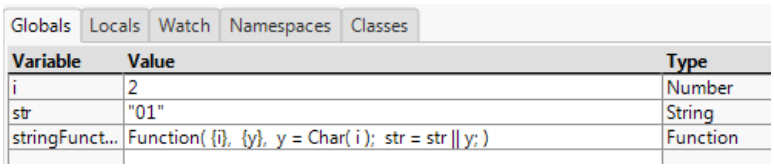
   The script runs until it hits a breakpoint. The results are shown in Figure 4.15.

   – `stringFunction()` is called, evaluated, and returns to the loop.

   – `i` is incremented and determined to be less than or equal to 9.

   In the Globals list, `i` is now 2, and `str` is now "01".

Figure 4.15 Global Variables at Second Breakpoint



You can continue to click **Run** and watch `i` and `str` change with each iteration of the loop. Or, click **Run without breakpoints** to complete running the script and exit the Debugger.

Stepping Into, Over, and Out

Step Into, **Step Over**, and **Step Out** offer flexibility when your script contains expressions, functions, or includes other JSL files.

1. Open the scriptDriver.jsl sample script and click the **Debug Script** button.
2. This script writes information to the log, so select the Log tab at the bottom of the Debugger to view the messages.
3. Click **Step Over**.

   The first line in the script is evaluated.

4. Click **Step Over** again.

   The current expression is evaluated, and the Debugger moves to the following line. In this case, the expression is a few lines long, and it assigns an expression to a variable.

5. Click **Step Over** again.

   This expression is several lines long, and assigns a function to a variable.

Line 30 calls the expression that was created earlier.

6. Click **Step Over**.

   The Debugger steps into the expression, running it line by line.

7. Continue clicking **Step Over** until the expression ends.

   The Debugger returns to the line following the expression call.

   Line 31 calls the function defined earlier.

8. Click **Step Over** to run the function without stepping into it. The Debugger runs the entire function, and returns to the line following the function call.

   Line 33 includes another script.

9. Click **Step Into**.

   The Debugger opens the script in another tab and waits.

10. Click **Step Over**.

    The next line in the included script is run.

11. Click **Step Out**.

    The Debugger runs the rest of the included script and returns to the line following the `Include()` function.

Watching Variables in Different Scopes and Namespaces

Tabs at the bottom of the Debugger window let you watch variables as they are created and changed. This example shows variables in several scopes and a namespace.

1. Open the scoping.jsl sample script and click the **Debug Script** button.
2. Click **Step Over**.

   The fourth line turns off `Names Default To Here`. If you run this script again in the same JMP session, this line resets the scoping so that the first variable that is created is in the global scope.

3. Click **Step Over**.

   A global variable named *x* is created. On the Globals tab, *x* has been added to the list, showing its value as 5 and its type as number.

4. Select the **Locals** tab, and then select **Global** from the list of scopes.

   The global variable *x* is also shown here.

5. Click **Step Over** twice.

   `Names Default To Here` is turned on, which places the rest of the script into a Here scope. Then a new variable *x* is created in that scope.

   Notice that the value of the global variable *x* has not changed.

6. Select **Here** from the list on the Locals tab.

   The local *x* is listed under **Here**, with its value and type.

7. Click **Step Over**.

   A `Local Here` scope is created. A second Here scope is shown in the Locals list.

8. Click **Step Over**.

   A new *x* variable is created in this Here scope. On the Locals tab, select each of the three scopes from the list (**Here**, **Here**, and **Global**) to see three different *x* variables.

9. Click **Step Over**.

   Look in the Debugger's log to see the output. Notice that *here:x* scopes to the local here, not the script window's here.

10. Click **Step Over**.

    After writing an empty line to the log, the script exits the Local Here scope. The second Here, along with its' *x* variable, has disappeared from the Locals list.

11. Click **Step Over**.

    A namespace called "test" is created, with another variable named *x*. Select the **Namespaces** tab to see it.

12. Click **Step Over** and look at the log.

13. Click **Step Over** to exit the Debugger.

Using the Debugger with Interactive Scripts

When your script creates interactive elements, the Debugger hands control back to the main instance of JMP so that you can interact with it. When you are finished, control returns to the Debugger.

1. Open the interactive.jsl sample script and click the **Debug Script** button.
2. Click **Step Over** twice.

   The `New Window` expression is evaluated, and a modal window waiting for input is created. You might need to move the Debugger window to see the new modal window.

3. Enter two numbers in the Assign X and Y window and click **OK**.

   Control is given back to the Debugger.

4. Click **Step Over** three times and look at the log in the Debugger.

   The log shows the two new numbers that you entered in the window.

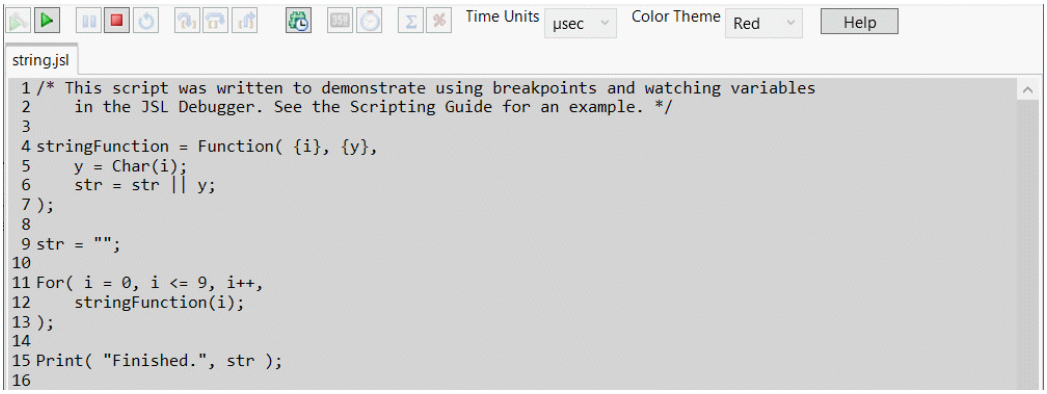
5. Click **Step Into** to exit the Debugger.

Using the JSL Profiler

Use the JSL Profiler to see how much time is spent executing particular lines or how many times a particular line is executed.

1. Open the string.jsl sample script.
2. Click the **Debug Script** button .
3. Click the **Profile JSL Script** button .

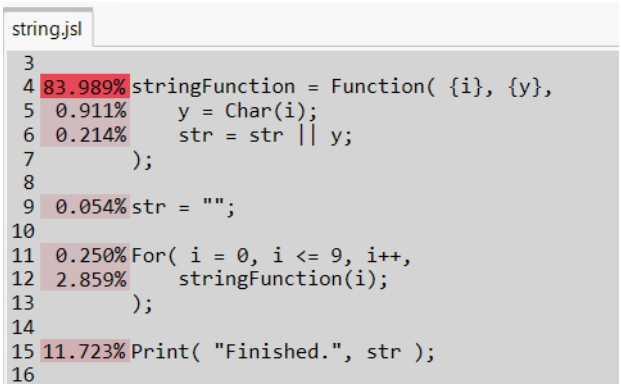
Figure 4.16 Initial JSL Profiler Window



4. Click the **Run** button to start profiling.

   The profiler collects information about the number of times a statement is executed and the time it takes to execute it. Time is cumulative and collected each time a JSL statement is executed.

Figure 4.17 Profiled Script Window



In the left margin, the selected statistics are displayed. Percent of time is displayed by default. Click the **Show Profile by Count** button to switch to percent of statement counts instead. The left margin is color-coded to allow for quick identification of problematic performance areas.

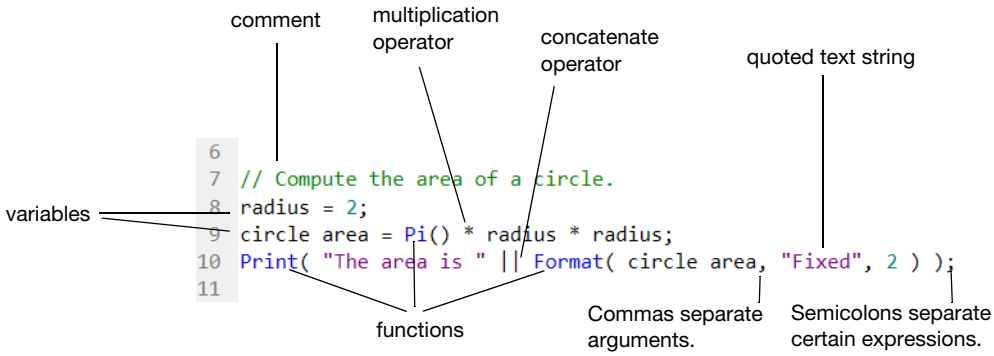
Chapter 5

JSL Building Blocks

Learning the Basics of JSL

Studying the syntax and basics of JSL is crucial, whether you are a beginning or an advanced user. Some concepts (such as loops and variables) are common among many scripting languages, but punctuation rules differ in JSL.

Figure 5.1 Example of a JSL Script



This chapter introduces you to the basic concepts of JSL, from syntax rules and file path conventions to conditions and namespaces.

Contents

JSL Syntax Rules . . . 100
Value Separators . . . 100
Numbers . . . 103
Names . . . 103
Comments . . . 104
Operators . . . 105
Boolean Values . . . 109
Global and Local Variables . . . 110
Local Namespaces . . . 110
Named Namespaces . . . 111
Show Symbols, Clear Symbols, and Delete Symbols . . . 111
Lock and Unlock Symbols . . . 112
Hide a Global Variable . . . 113
Rules for Name Resolution . . . 113
Variable Names Used as Arguments . . . 114
Rules for Resolving Names . . . 114
Scoping Operators . . . 116
Troubleshoot Variables and Column Names . . . 119
Troubleshoot Variables and Keywords . . . 120
Frequently Asked Questions about Name Resolution . . . 120
Alternatives for Gluing Expressions Together . . . 121
Iterate . . . 122
For . . . 122
While . . . 123
Summation . . . 125
Product . . . 125
Break and Continue . . . 126
Conditional Functions . . . 128
If . . . 128
Match . . . 130
Choose . . . 132
Interpolate . . . 133
Step . . . 135
Compare Incomplete or Mismatched Data . . . 135
Missing Values . . . 137
Inquiry Functions . . . 138
General Element Types . . . 139
Specific Element Types . . . 139
Object Attributes . . . 141

Host Information 141
Version Information 141

JSL Syntax Rules

All scripting and programming languages have their own syntax rules. JSL looks familiar if you have programmed in languages such as C and Java. However, rules for punctuation and spaces differ in JSL.

The following sections describe JSL syntax rules for the following basic components of the language.

Value Separators

Words in JSL are separated by parentheses, commas, semicolons, spaces, and various operators (such as +, –, and so on). This section describes the rules for using these separators and delimiters in JSL.

Commas

A comma separates items, such as items in a list, rows in a matrix, or arguments to a function.

```
my list = {1, 2, 3};
your list = List( 4, 5, 6) ;
my matrix = [3 2 1, 0 -1 -2];
If( Y < 20, X = Y );
Table Box( String Col Box( "Age", a ) );
```

Note: To glue a sequence of commands into a single argument inside a function, separate each sequence with a semicolon. See "Semicolons".

Parentheses

Parentheses have several purposes in JSL:

- Parentheses group operations in an expression. The following parentheses group the operations in the `If()` expression.

  ```
  y = 10;
  If(
     Y < 20, X = Y,
     X = 20
  );
  ```

- Parentheses delimit arguments to a function. In the following example, parentheses enclose the argument to the `Open()` function.

  ```
  Open( "$SAMPLE_DATA/Big Class.jmp" );
  ```

- Parentheses also mark the end of a function name, even when arguments are not needed. For example, the `Pi` function has no arguments. However, the parentheses are required so that JMP can identify *Pi* as a function.

  ```
  Pi();
  ```

Note: Be careful that parentheses match. Every `(` needs a `)`, or errors result.

The script editor can match fences (parentheses, brackets, and braces). Press Ctrl+] (Command+B on macOS) with your cursor in any part of a script. The editor searches for fences, highlighting the text between the first set of opening and closing fences that it finds. Repeat this process to highlight the next-higher fence. See "Match Parentheses, Brackets, and Braces" for an example.

Semicolons

Expressions separated by a semicolon are evaluated in succession, returning the result of the last expression. In the following code, 0 is assigned to the variable `i` and then 2 is assigned the variable `j`.

```
i = 0;
j = 2;
```

You can also use semicolons to join arguments that are separated by commas as shown in the following `If()` expression.

```
If( x < 5, y = 3; z++; );
```

The semicolon in other languages is used as an expression terminator character. In JSL, the semicolon is a signal to continue because more commands might follow. For more information about separating expressions with semicolons, see "Alternatives for Gluing Expressions Together".

Semicolons at the end of a script or at the end of a line of arguments are harmless, so you can also think of semicolons as terminating characters. Trailing semicolons are allowed at the end of a script stream and after a closing parenthesis or closing curly brace. In fact, terminating each complete JSL expression with a semicolon helps avoid errors when you copy and paste small scripts into a larger one.

The semicolon is equivalent to the `Glue()` function. See "Operators" for more information about semicolons and `Glue()`.

Double Quotes

Double quotes enclose text strings. Anything inside double quotes is taken literally, including spaces and upper- or lower-case; nothing is evaluated. If you have `"Pi() ^ 2"` (inside double quotes), it is just a sequence of characters, not a value close to ten.

You do have to be a careful with text strings. Extra spaces and punctuation do affect the output, because text strings inside double quotes are used literally, exactly as you enter them.

To have double quotes inside a quoted string, precede each quotation mark with the escape sequence \! (backslash-bang). For example, run the following script and look at the title of the window:

```
New Window( "\!"Hello\!" is a quoted string.",
  Text Box( Char( Repeat( "*", 70 ) ) )
);
```

Table 5.1 Escape Sequences for Quoted Strings

\!b	blank
\!t	tab
\!r	carriage return only
\!n	linefeed (newline) only
\!N	inserts line breaking characters appropriate for the host environment[a]
\!f	formfeed (page break)
\!0	null character **Note:** The null character is dangerous to use, because it is typically treated as the end of the string. Be sure to type the number zero, not the letter O.
\!\	backslash
\!"	double quotation mark

a. On macOS, this escape sequence is CR (carriage return character, hexadecimal '0D'). On Windows, this sequence is CR LF (carriage return followed by a linefeed, hexadecimal '0D0A').

Sometimes, long passages require a lot of escaped characters. In these cases, use the notation \[...]\ and everything between the brackets does not need or support escape sequences. Here is an example where \[...]\ is used inside a double-quoted string.

```
jslPhrase = "The JSL to do this is :\[
  a = "hello";
  b = a|| " world.";
  Show(b);
  ]\ and you use the Submit command to run it.";
```

Spaces

JSL allows whitespace characters inside names; spaces, tabs, returns, and blank lines inside or between JSL words are ignored. This is because most JSL words come from the user interface, and most of those commands have spaces in them. For example, the JSL expression for the Fit Model platform is `Fit Model()` or `FitModel()`.

Spaces inside an operator or between digits in a single number are not allowed. In these cases, the script generates errors. For example, you cannot put spaces between the two plus signs in `i++` (`i+ +`) or in numbers (`4 3` is not the same as `43`).

Note: Why does JSL allow whitespace characters inside names? For one reason, the names of commands and options match the equivalent commands in JMP menus and windows. Another reason is that data table column names often include spaces.

Numbers

Numbers can be written as integers, decimal numbers, dates, times, or date-time values. They can also be included in scientific notations with an `E` preceding the power of ten. For example, these are all numbers:

```
.   1   12   1.234  3E3  0.314159265E+1 1E-20 01JAN98
```

Note: A single period by itself is considered a missing numeric value (sometimes called *NaN* for "not a number").

For more information about dates, times, and date-time values, see "Date-Time Functions and Formats". See "Currency" for more information about combining numbers with currency symbols.

Names

A name is simply something to call an item. When you assign the numeric value 3 to a variable in the expression `a = 3`, `a` is a name.

Commands and functions have names, too. In the expression `Log( 4 )`, `Log` is the name of the logarithmic function.

Names have a few rules:

- Names *must* start with an alphabetic character or underscore and can continue with the following:
  - alphabetic characters (a-z A-Z)
  - numeric digits (0-9)

  - whitespace characters (spaces, tabs, line endings, and page endings)
  - mathematical symbols in Unicode (such as the Greek small alpha letter α)
  - a few punctuation marks or special characters (apostrophes (‘), percent signs (%), periods (.), backslashes (\), and underscores (_))
- When comparing names, JMP ignores whitespace characters (such as spaces, tabs, and line endings). Upper case and lower case characters are not distinguished. For example, the names `Forage` and `for age` are equivalent, despite the differences in white space and case.

You can still have a name that is any other sequence of characters. If the name does not follow the rules above, it needs to be quoted and `followed by "n"`. For example, to use a global variable with the name `taxable income(2011)`, you must use `"taxable income(2011)"n` every time the variable appears in a script:

```
"taxable income( 2011 )"n = 456000;
tax = .25;
Print( tax * "taxable income( 2011 )"n) ;
   114000
```

Note: Previously, the `Name()` parser directive wrapped the column name that contained illegal characters. `Name()` is deprecated.

For more information about how JMP interprets names, see “Rules for Name Resolution”.

Comments

Comments are notes in the code that are ignored by the JSL processor (or *parser*). You include comments to describe sections of the script. Comments are also convenient for removing portions of a script temporarily. For example, you can insert comment symbols around code that might be causing an error and then rerun the script.

Type the comment symbols around code that you want to comment. The following example shows code commented with `/* */` in the middle of a line. When the script is run, JMP considers both expressions to be identical.

```
tax /*percentage*/ = .25;
tax = .25;
```

Table 5.2 describes the comment symbols.

Table 5.2 Comment Symbols

Symbol	Syntax	Explanation
`//`	`// comment`	Begins a comment. The comment does not have to be at the beginning of a line, but everything up to the end of the line is a comment.
`/* */`	`/* comment */`	Begins and ends a comment. This comment can appear in the middle of a line. Script text before and after the comment is parsed normally.
`//!`	`//!`	Add `//!` to the first line of the script, and the script runs automatically when opened in JMP. (In other words, the script editor does not open.)

Operators

An operator is a one- or two-character symbol used to perform common arithmetic actions, compare values, construct lists, subscript into a data element, send messages, concatenate lists, scope names, glue expressions, and end expressions.

Arithmetic operators come in several varieties:

- *infix* (with arguments on either side, such as + in `3 + 4`, or = in `a = 7`)
- *prefix* (with one argument on its right side, such as `!a` for logical negation)
- or *postfix* (with one argument on its left side, such as `a++` for incrementing `a`

To make writing expressions easier, JSL uses certain special character operators that are alternatives to using functions. These operators have the same meaning as if the phrase had been written as a function. For example, the following two expressions are equivalent.

```
Net Income After Taxes = Net Income - Taxes;
Assign( Net Income After Taxes, Subtract( Net Income, Taxes ) );
```

The assignment operation can be written either as the `Assign()` function or as an infix operator =. Similarly, subtraction can be done with the `Subtract()` function or the infix minus sign; they are equivalent inside JMP.

Note: Usually white space is ignored in JSL, so that "netincomeaftertaxes" and "net income after taxes" are the same thing. There is one exception: the two-character operators must *not* have a space between characters. Always enter the following operators without spaces in between:

```
||, |/, <=, >=, !=, ==, +=, -=, *=, /=,  ++, --, <<, ::, :*, :/, /*, */
```

Another common operator is the semicolon (`;`). You use the semicolon to:

- Separate yet join one expression to another in a programming sequence. The semicolon returns the result of its last argument, so `a;b` is the same as `Glue(a,b)`.
- End an expression. Though the semicolon is permitted at the end of an expression, it is not the expression terminator used in other languages.

An expression can contain more than one operator. In these instances, the operators are grouped in order of precedence with decreasing priority. For example, * takes precedence over +:

```
a + b * c
```

So `b * c` is evaluated first, and then the result is added to `a`.

+ takes precedence over -:

```
a + b * c - d
```

So `b * c` is evaluated, and then the result is added to `a`. `d` is then subtracted from the result of `a + b * c`.

Table 5.3 shows operators shaded in order of precedence and each operator's function equivalent.

Table 5.3 Operators and Their Function Equivalents in Order of Precedence

Operator		Function Syntax	Explanation
`{ }`	`List`	`{a,b}` `List(a,b)`	Construct a list.
`[ ]`	`Subscript`	`a[b,c]` `Subscript(a,b,c)`	Subscripts identify specific elements within a data element *a*, where *a* could be a list, a matrix, a data column, a platform object, a display box, and so on.

Table 5.3 Operators and Their Function Equivalents in Order of Precedence *(Continued)*

Operator		Function Syntax	Explanation
++	Post Increment	*a*++ Post Increment(*a*)	Adds one (1) to *a*, in place. JMP does not have a pre-increment operator. Instead, use the Add To() operator, which is (+=).
--	Post Decrement	*a*-- Post Decrement(*a*)	Subtracts one (1) from *a*, in place. JMP does not have a pre-decrement operator. Instead, use the Subtract To() operator, which is (-=).
^	Power	*a*^*b* Power(*a*,*b*) Power(*x*)	Raise *a* to exponent power *b*. With only one argument, 2 is assumed as the power, so Power(*x*) computes x^2.
-	Minus	-*a* Minus(*a*)	Reverses sign of *a*.
!	Not	!*a* Not(*a*)	Logical Not. Maps nonzero (or true) values to 0 (which means false). Maps 0 (or false) values to 1 (which means true).
*	Multiply	*a**b* Multiply(*a*,*b*)	Multiplies *a* by *b*.
:*	EMult	*a*:**b* EMult(*a*,*b*)	Elementwise multiplication for matrices *a* and *b*. (Each element in matrix *a* is multiplied by each element in matrix *b*.)
/	Divide	*a*/*b* Divide(*a*,*b*) Divide(*x*)	Divide(*a*, *b*) divides *a* by *b*. Divide(*x*) interprets the argument as a denominator and implies 1 as the numerator, yielding the reciprocal 1/*x*.
:/	EDiv	*a*:/*b* EDiv(*a*,*b*)	Elementwise division for matrices *a* and *b*. (Each element in matrix *a* is divided by each element in matrix *b*.)

Table 5.3 Operators and Their Function Equivalents in Order of Precedence *(Continued)*

Operator		Function Syntax	Explanation
+	Add	*a*+*b* Add(*a*,*b*)	Adds *a* and *b*.
–	Subtract	*a*–*b* Subtract(*a*,*b*)	Subtracts *b* from *a*.
||	Concat	*a*||*b* Concat(*a*,*b*)	Joins two or more strings; two or more lists; and horizontally concatenates matrices. See "Concatenate Lists" or the *JSL Syntax Reference*.
|/	VConcat	*matrix1*|/*matrix2* VConcat(*matrix1*, *matrix2*)	Vertically concatenate matrices. (Use || or Concat() to horizontally concatenate matrices.)
::	Index	*a*::*b* Index(*a*,*b*)	For matrices, generates the integers from *a* to *b*. (Colons are also used as prefix operators for scoping, where :a means data table column a, and ::a means JSL global variable *a*. See "Scoping Operators".)
<<	Send	object << message Send(*object*, *message*)	Send *message* to *object*.
==	Equal	*a*==*b* Equal(*a*,*b*)...	Boolean values for comparisons. They all return 1 if true, 0 if false. Missing values in either *a* or *b* causes a return value of missing, which evaluates as neither true nor false. See "Missing Values", for treatment of missing values.
!=	Not Equal	*a*!=*b* Not Equal(*a*,*b*)...	
<	Less	*a*<*b* Less(*a*,*b*)...	
<=	Less or Equal	*a*<=*b* Less or Equal(*a*,*b*)	
>	Greater	*a*>*b* Greater(*a*,*b*)	
>=	Greater or Equal	*a*>=*b* Greater or Equal(*a*,*b*)	

Table 5.3 Operators and Their Function Equivalents in Order of Precedence *(Continued)*

Operator		Function Syntax	Explanation
<=, <	Less Equal Less	*a*<=*b*<*c* Less Equal Less(*a*,*b*,*c*)	Range check. Return 1 if true, 0 if false. Missing values in either *a* or *b* propagate missing values.
<, <=	Less Less Equal	*a*<*b*<=*c* Less Less Equal(*a*,*b*,*c*)	
&	And	*a*&*b* And(*a*,*b*)	Logical And. Returns true if both are true. If the value on the left is false, the value on the right is not evaluated. See "Missing Values", for treatment of missing values.
|	Or	*a*|*b* Or(*a*,*b*)	Logical Or. Returns true if either or both are true. See "Missing Values", for treatment of missing values.
=	Assign	*a*=*b* Assign(*a*,*b*)	Put the value of *b* into *a*. Replaces the current value of *a*.
+=	Add To	*a*+=*b* AddTo(*a*,*b*)	Add the value of *b* into *a*.
-=	Subtract To	*a*-=*b* SubtractTo(*a*,*b*)	Subtract *b* from *a*, and put back into *a*.
*=	Multiply To	*a**=*b* MultiplyTo(*a*,*b*)	Multiply *b* with *a*, and put back into *a*.
/=	Divide To	*a*/=*b* DivideTo(*a*,*b*)	Divide *b* into *a*, and put back into *a*.
;	Glue	*a*;*b* Glue(*expr*, *expr*, ...)	First do *a*, and then do *b*.

Boolean Values

A Boolean value enables or disables an option. To enable the option, use `1`, `true`, or `yes` as the argument.

```
obj << Unequal Variances( 1 );
obj << Unequal Variances( "true" );
obj << Unequal Variances( "yes" );
```

Note: Quoting the value prevents it from being evaluated as a variable.

Omitting the argument also enables the option. This is the default for all but row state messages.

```
obj << Unequal Variances;
obj << Unequal Variances();
```

To disable the option, use `0`, `false`, or `no` as the argument.

To toggle an option (turn off an enabled option or turn on a disabled option), use `"toggle".`

Notes:

- The quoted `"present"`, `"absent"`, `"on"`, `"off"`, `"switch"`, and `"flip"` are supported for backwards compatibility.
- For row state messages, an empty argument always toggles the option.

Global and Local Variables

Variables are names that hold values, which you reference later in scripts. There are two types of variables:

- *Global variables* are shared among all scripts that you run in a JMP session.
- *Local variables* apply only to the script context in which you define them. They can also be local to only a piece of a script, as with variables local to a particular function.

To limit the scope of variables, you can define them in a *namespace*, which is a collection of variables, functions, and other unique names. JMP has a single global variable namespace that all scripts use by default. When you use a name plainly, without a qualifying syntax, the name is an *unscoped* variable and therefore in the global namespace.

```
x = 1;
```

Local Namespaces

Putting variables in the global namespace can cause conflicts. When two scripts have variables with the same names, the value of the variable in the last script that you run last overwrites the variable's value in the first script.

To prevent this problem, we recommend that you begin each script with the following line:

```
Names Default To Here( 1 );
```

The `Names Default To Here( 1 );` function makes all unscoped variables in the script local to that script and does not affect the global variable namespace. See "Advanced Scoping and Namespaces".

Notes:

- The `Names Default to Here` setting is true by default for custom menus and toolbar buttons. A script that runs when you select a custom menu item or click a custom toolbar button does not affect global variables.
- The `Names Default to Here` setting is false by default for table scripts. If a table script specifies `Names Default to Here(1)`, this setting lasts only as long as the table script runs. After the script finishes running, the setting is reset to whatever it was before the script ran.

Named Namespaces

You can also create a variable in a specific namespace. In the following example, the `x` variable is created in the `aa` namespace:

```
aa:x = 1;
```

Preceding local variables with the `Local()` function is another option. Both `a` and `b` are local variables in the following expression:

```
Local( {a = 1, b}, ... );
```

Scoping operators also distinguish a global variable from a local variable. See "Rules for Name Resolution".

The following sections describe functions that help you manage variables.

Show Symbols, Clear Symbols, and Delete Symbols

The `Show Symbols()` function lists all variables and namespaces that are defined both globally and in the local script, along with their current values. Here is an example of `Show Symbols()` messages that are shown in the log:

```
Show Symbols();
   // Here
   a = 5;
   b = 6;
   // 2 Here

   // Global
   c = 10;
   // 1 Global
```

Tip: The JSL debugger also shows you the values of variables and namespaces. See "Debug or Profile Scripts".

The `Clear Symbols()` function erases the values set for variables that are defined both globally and in the local script. For example, after you clear and then show symbols, the variables are empty.

```
Clear Symbols();
Show Symbols();
   // Here
   a = Empty;
   b = Empty;
   // 2 Here

   // Global
   c = Empty;
   // 1 Global
```

Note: The older `Show Globals()` and `Clear Globals()` functions are aliases of the newer `Show Symbols()` and `Clear Symbols()` functions.

To remove *all* global variables and namespaces, use the function `Delete Symbols()`. After the last `Show Symbols()` in the following script is run, nothing shows up in the log. All variables have been completely removed from memory.

```
Delete Symbols();
Show Symbols();
```

To list variables in all namespaces, use `Show Namespaces()`. To delete only a specific namespace, use `ns << Delete. Clear Symbols()` and `Delete Symbols()` do not clear or delete variables in each namespace, although they do clear and delete variables that contain references to namespaces. See "Rules for Name Resolution" for more information about unscoped variables.

Note: `Clear Symbols()` and `Delete Symbols()` break all scripts that are currently in use. These functions can be very useful in a programming and debugging environment, but do not include them in any script that you plan to distribute. If you include `Names Default To Here(1)` in your scripts, clearing and deleting global symbols is unnecessary.

Lock and Unlock Symbols

If you want to lock a variable to prevent it from being changed, use the `Lock Symbols()` function. (`Lock Globals()` is an alias.)

```
Lock Symbols( name1, name2, ... );
```

To release the lock and enable the global to be changed, use the `Unlock Symbols()` function. (`Unlock Globals()` is an alias.)

```
Unlock Symbols( name1, name2, ... );
```

The primary use of these two commands is to prevent inadvertent changes to variables. For example, locking a variable prevents `Clear Symbols`() from clearing a variable that is being used by another script.

Note: You cannot use `Lock Symbols()` to lock a namespace. Instead, use `ns <<Lock Namespace.`

Hide a Global Variable

To hide (or protect) a global variable, put two underscores (__) before the name. Protecting a global variable means that it will be hidden and can not be examined or shown. However, the behavior is different depending upon the context.

This example will not print the variable because it is protected:

```
Show(::__xyz);
```

This example returns a NULL value for the protected variable:

```
::a = Name Expr( ::__xyz );
```

This example does return a value for the protected variable:

```
Show( Eval( ::__xyz ) );
```

Rules for Name Resolution

In JMP, you identify the following types of objects by a name:

- Columns and table variables in a data table
- Global variables, which hold values for the remainder of a JMP session
- Scriptable object types
- Arguments and local variables inside formulas

Most of the time, you can just use an object's name directly to refer to the object. Consider the following example:

```
ratio = height / weight;
```

Depending on the complexity of your script, it might be obvious that `ratio` is a variable and `height` and `weight` are data table column names. But what if the meanings are ambiguous? A script might use `ratio` as a global variable and as column names.

Variable Names Used as Arguments

Message arguments are evaluated as a rule. In previous versions of JMP, some messages interpreted a variable name as a string value. For example, the following script results in a linear scale in the current version of JMP and a log scale in JMP 11.

```
Log = "Linear";
axis box << Scale( Log );
```

The syntax `Scale( "Log" )` is preferred for setting a string literal. To catch ambiguous instances, set the "Allow Unquoted Strings in JSL" General preference to "Yes (with a warning)" or "No".

Rules for Resolving Names

JMP interprets object names using *name resolution*. The following rules are applied sequentially to unscoped names:

1. If the variable is followed by a pair of parentheses `( )`, look it up as a function.
2. If the variable is prefixed by `:` scope operator or an explicit data table reference, look it up as a data table column or table variable.
3. If the variable is prefixed by `::` scope operator, look it up as a global variable.
4. If the variable is an explicit scope reference (such as `group:vowel`), look it up in the user-defined `group` namespace.
5. If the variable is in a `Local` or `Parameter` function, look it up as a local variable. If it is nested, repeat until a function call boundary is found.
6. If the variable is in a user-defined function or method, look it up as a function or method argument or local variable.
7. Look the variable up in the current scope and its parent scope. Repeat until the `Here` scope is encountered.
8. Look the variable up as a variable in the `Here` scope.
9. Look the variable up as a global variable.
10. If `Names Default to Here(1)` is at the top of the script, stop looking. The scope is local.
11. Look the variable up as a data table column or table variable. Columns are preferred over table variables.

12. Look the variable up as an operator or a platform launch name (for example, Distribution, Bivariate, Chart, and so on).
13. When the name cannot be found:
    - If the name is used in a reference, print and error to the log.
    - If the name is used as the target of an assignment (as an L-value), test the following:

      If the variable is preceded by `::` scope operator, create and use a global variable.

      If the variable is an explicit scope reference, create and use the variable in the specified namespace or scope.

      If `Names Default to Here(0)` is at the top of the script, create a global variable.

      If `Names Default to Here(1)` is at the top of the script, create a `Here` namespace variable.

Exceptions

- Some names are variables that refer to an object such as a data table, data column, or platform; they are not used for getting or setting a value. These names are passed through (or interpreted literally) rather than resolved.
- For function definitions, column formulas, and Nonlinear platform formulas, the scope is the same for each row in a column.
- If a name is a direct reference to a column in a data table that has been closed, the name is resolved again to that column when the table is reopened.

  ```
  dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
  Show( :weight << Get As Matrix ); // weight  resolves to a column name
  Close( dt, NoSave );
  Show( :weight << Get As Matrix ); // weight  cannot be resolved
  dt = Open( "$SAMPLE_DATA/Big Class.jmp" ); // reopen data table
  Show( :weight << Get As Matrix ); // weight  resolves to a column name
  ```

  However, the following example does not resolve the variable to the second instance of the data table:

  ```
  dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
  col = Column( dt, 5); // col is Column( "weight" );
  Close( dt, NoSave );
  dt = Open( "$SAMPLE_DATA/Big Class.jmp" ); // reopen data table
  Show( col << Get As Matrix ); // the reference to the first data table no
    longer exists
  ```

The following sections describe how JMP resolves the names of data table columns. For more information about name resolution, see “Advanced Scoping and Namespaces”.

Scoping Operators

Using *scoping operators* is an easy way to help JMP resolve ambiguous names (for example, when a name refers to both a variable and a column name).

In the following example, the prefix double-colon operator (`::`) identifies `z` as a global variable. The single-colon prefix operator (`:`) identifies `x` and `y` as column names.

```
::z = :x + :y;
```

Tip: The `Names Default to Here(1)` function also affects name resolution. See "Names Default To Here and Global Variables".

Two JSL functions are interchangeable with scoping operators. Table 5.4 describes the functions and syntax.

Table 5.4 Scoping Operators

Operator and Equivalent Function	Function Syntax	Explanation

`:` `As Column`	`:name`
	`dt:name`
	`As Column(`*`dt`*`,`
	`name``)`
		Forces *`name`* to be evaluated as a data table column. The optional data table reference argument, *`dt`*, sets the current data table. See "Scoped Column Names" for examples.

`::` `As Global`	`::name` `As Global(name)`	Forces *`name`* to be evaluated as a global variable. **Note:** The double-colon is also used as an infix operator to represent ranges.

Scoped Column Names

Scoping column names is the simplest way to prevent conflicts with variable names. Use scoping operators to force names in a script to refer to columns.

1. The prefix colon (`:`) means that the name refers to a table column or table variable only, never a global variable. The prefix colon refers to the current data table context.

   ```
   :age;
   ```

2. The infix colon (`:`) operator extends this notion by using a data table reference to specify which data table has the column. This is particularly important when multiple data tables are referenced in a script.

   In the following example, the `dt` variable sets the data table reference to `Big Class.jmp`. The infix colon separates the data table reference and `age` column.

```
dt = Data Table( "Big Class.jmp" );
dt:age // The colon is an infix operator.
```

`As Column()` achieves the same results:

```
dt = Data Table( "Big Class.jmp" );
As Column( dt, age );
```

Therefore, the following expressions are equivalent when only Big Class.jmp is open:

```
:age;
As Column( dt, age );
dt:age;
```

The `Column` function can also identify a column. For Big Class.jmp, the following expressions all refer to age, the second column in the table:

```
Column( "age" );
Column( 2 );
Column( dt, 2 );
Column( dt, "age" );
```

Prevent Column Name and Variable Name Conflicts

To avoid conflicts, use unique column and variable names or scope the names.

- When a global variable and a column have the same name, the global variable name takes precedence. In this situation, you must scope the column name.

  ```
  ::age = [];
  age = :age << Get As Matrix;
  ```

- To avoid ambiguity between the global variable and column name, scope both variables.

  ```
  ::age = :age << Get As Matrix;
  ```

- If more than one data table might be open, assign data table references to variables. Scope your columns to the appropriate table.

  ```
  dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
  dt2 = Open( "$SAMPLE_DATA/Students.jmp" );
  ::age = dt1:age << Get As Matrix;
  ::height = dt2:height << Get As Matrix;
  ```

Note that JMP evaluates column formulas through each consecutive cell of the column, so scoping the column name is usually unnecessary. However, if a variable assigned in a formula has the same name as a column, you must scope the column name. See "Scoped Names".

Tip: The `Names Default to Here(1)` function also affects name resolution. See "Names Default To Here and Global Variables".

Unscoped Column Names

Sometimes an unscoped name gets or sets a value. JMP resolves it as a column in a data table (rather than a global variable) under these circumstances:

- if no global variable, local variable, or an argument using that name already exists,
- and the data table in context has a column of that name,
- and
  - either the current row is set to a positive value
  - or the name is subscripted (for example, the subscript `[1]` in `weight[1]` selects the first value in the weight column).

Exception

In column formulas and Nonlinear formulas, column names take precedence over global variables.

Set the Current Data Table Row

By default, the current row is 0, an illegal row number. So the following expression assigns a missing value to the *ratio* global variable:

```
ratio = height / weight;
```

Specify the row number with the `Row()` function. In the following example, the row is set to 3. The height in that row is divided by the weight, and the result is assigned to the *ratio* global variable.

```
Row() = 3;
ratio = height / weight;
```

Another possibility is to use subscripts to specify the row number. The following expression divides the height in row 3 by the weight in row 4.

```
ratio = height[3] / weight[4];
```

Specifying the row number is unnecessary when the script iterates a row at a time down the whole column. The following example creates the ratio column. For each row, the height is divided by the weight.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "ratio" );
For Each Row( :ratio = height / weight );
```

JMP evaluates formulas and calculates pre-evaluated statistics iteratively down a column. In these instances, identifying the row number is also unnecessary. (Pre-evaluated statistics are single numbers computed from the data table, as discussed in "Pre-Evaluated Statistics".)

Name Resolution in Automatically Run Scripts

In an `Open()` function, you can specify that the script run automatically when it's opened.

```
Open( "$MY_SCRIPTS/A.jsl", Run JSL( 1 ) );
Open( "$MY_SCRIPTS/B.jsl", Run JSL( 1 ) );
```

You can also type `//!` on the first line of the script to run the script automatically.

In these instances, the only shared names between `A.jsl` and `B.jsl` are globals and namespaces (not here and not local).

To apply the names in `A.jsl` to `B.jsl`, include `B.jsl` in `A.jsl` instead of opening and running both scripts.

```
Include( "$MY_SCRIPTS/B.jsl" );
```

Troubleshoot Variables and Column Names

When you reference a column name using `As Name()`, and `Names Default To Here( 1 )` is set, JMP returns a variable reference. That reference is then processed using the standard reference rules.

In the following example, there is no height variable in the `Here:` scope, so JMP returns an error.

```
Names Default To Here( 1 );
Open( "$SAMPLE_DATA/Big Class.jmp" );
As Name( "height" )[3]; // converts height to a variable reference
  As Name( "height" )[/*###*/3];
```

The `/*###*/` characters in the log indicate the location of an error in the script execution.

To prevent this problem, use one of the following methods:

- Use `As Column()` instead of `As Name()`:

  ```
  Names Default To Here( 1 );
  Open( "$SAMPLE_DATA/Big Class.jmp" );
  As Column( "height" )[3]; // converts height to a data column reference
  ```

- Explicitly scope height with `As Name():`

  ```
  Names Default To Here( 1 );
  dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
  dt:( As Name( "height" ) )[3]; // scopes height as a data column reference
  ```

These scripts return 55, the value of height in the third row of Big Class.jmp.

Troubleshoot Variables and Keywords

Name resolution errors can also occur when a variable and unquoted keyword have the same name. For example, one argument for `<<Preselect Role()` is "Y". Quote this argument if your script also uses Y as a variable.

Frequently Asked Questions about Name Resolution

Should you always scope?

Yes. When in doubt, scope. Scoping is especially important in scripts that might be used by many people on a variety of data tables; you will not necessarily know whether a name is used in two contexts (such as for both a global variable and column name).

If you are writing such scripts, consider using explicit scoping and namespaces. See "Advanced Scoping and Namespaces".

Prefix scope operators do not take run-time overhead after the first resolution. Infix scope operators, which follow data table references, always take run-time overhead.

What is the difference between a column reference and a column referred to by name? If I have a column reference in a global variable, how do I assign a value to a cell in the column?

With a column reference, you can send messages to change specific characteristics of the column or to access its values (for example, coloring cells or setting a formula).

When a column has been assigned to a global variable, assign a value to a cell in the column using a subscript. Suppose that the name of the column height has been assigned to the x variable:

```
x = Column( "height" );
```

Assign a value to the third row in the height column:

```
x[3] = 64 // sets the third row of height to 64
```

Note: The current row in a JSL script is *not* determined by selecting rows or positioning your cursor in the row. The current row is defined to be zero (no row) by default. You can set a current row with `Row()` (for example, `Row() = 3`). Please note that such a setting only lasts for the duration of that script. Then, `Row()` reverts to its default value, zero. This behavior means that submitting a script all at once can produce different results than submitting a script a few lines at a time.

Another way to establish a current row for a script is to enclose it in `For Each Row()`. This method executes the script once for each row of the current data table. For an example, see "If". See "Data Tables" for more information about working with data tables.

Which Has Precedence When Scoping, ":" or "[ ]"?

Scoping occurs before subscripting. This means that these two lines of code are equal:

```
dataTable:colName[i]
(dataTable:colName)[i]
```

Alternatives for Gluing Expressions Together

You can separate expressions with a semicolon, either on the same line or on different lines. JMP then evaluates each expressions in succession, returning the result of the last one. Here is an expression that first sets `a` to 2 and then sets `b` to 3:

```
a = 2;
b = 3;
```

The semicolon joins the two expressions and returns the value of the last one. So if `x = (a = 2; b = 3)`, the value of `x` is 3.

The `Glue()` function returns the result of the last expression. This function is equivalent to using semicolons. The following expressions both return 3:

```
Glue( a = 2, b = 3 );
a = 2; b = 3;
```

The `First()` function also evaluates each argument sequentially but returns the result of the first expression. The following expression returns 2:

```
First( a = 2, b = 3);
```

Example

What does `First()` do in the following script?

```
x = 1000;
```

```
First( x, x = 2000 );
```

The `First()` function returns the value of `x` (1000). 2000 is then assigned to `x`.

Iterate

JSL provides the `For()`, `While()`, `Summation()`, and `Product()` functions to repeat (or *iterate*) actions according to the conditions that you specify.

Note: A similar function called `For Each Row()` is for iterating actions over rows of a data table. See "If" for an example. "Additional Ways to Access Data Values" also describes iterating through table rows.

- "For"
- "While"
- "Summation"
- "Product"
- "Break and Continue"

For

The `For()` function expects four arguments separated by commas. The first three arguments are rules for how many times to repeat the loop, and the fourth is what to do each time the loop is executed.

Here is the basic syntax for `For()`:

```
For( initialization, while, iteration, body );
```

For example, the following script sums the numbers from 0 to 20:

```
s = 0;
For( i = 0, i <= 20, i++, s += i );
```

The script works like this:

`s = 0;`	Sets the `s` variable to 0. This variable holds the sum.
`For(`	Begins the `For()` loop.
`i = 0,`	Sets the initialization variable (`i`) to 0. This expression is performed only once.

`i <= 20,`	Each time the loop begins, compares `i` to 20. As long as `i` is less than or equal to 20, continue evaluating the loop. If `i` is greater than 20, immediately break out of the loop.
`i++,`	At the end of the loop, increments `i` by 1. Note that this step is done after the body of the loop (next line) is evaluated.
`s += i`	Evaluates the body of the loop. Adds the value of `i` to `s`. After the body is finished, *`i`* is incremented (previous line).
`);`	Ends the loop.

Infinite Loops

`For` loops that always evaluate as true create an infinite loop, which never stops. To stop the script, press Esc on Windows (or Command+period on macOS). You can also select **Edit > Stop Script**. On macOS, **Edit > Stop Script** is available only when the script is running.

Comparing For Loops in JSL to C and C++

The JSL `For()` loop works just like it does in the C (and C++) programming language, although the punctuation is different.

Tip: If you know C, watch out for the difference between JSL and C in the use of semicolons and commas. In JSL, `For()` is a function where commas separate arguments and semicolons join expressions. In C, `for` is a special clause where semicolons separate arguments and commas join them.

While

A related function is `While()`, which repeatedly tests the condition and evaluates its body script as long as the condition is true. The syntax is:

```
While( condition, body );
```

For example, here are two different programs that use a `While()` loop to find the least power of 2 that is greater than or equal to `x` (287). The result of both programs is 512.

```
x = 287;

// loop 1:
y = 1;
While( y < x, y *= 2 );
Show( y );
```

```
// loop 2:
k = 0;
While( 2 ^ k < x, k++ );
Show( 2 ^ k );
```

The scripts work like this:

`x = 287;`	Sets `x` to 287.
`// loop 1`	
`y = 1;`	Sets `y` to 1.
`While(`	Begins the `While()` loop.
`y < x,`	As long as `y` is less than `x`, continues evaluating the loop.
`y *= 2`	Multiplies 1 by 2 and then assigns the result to `y`. The loop then repeats while `y` is less than 287.
`);`	Ends the loop.
`Show(y);`	Shows the value of `y` (512).
`// loop 2`	
`k = 0;`	Sets `k` to 0.
`While(`	Begins the `While()` loop.
`2 ^ k < x,`	Raises 2 to the exponent power of `k` and continues evaluating the loop as long as the result is less than 287.
`k++`	Increments `k` to 1. The loop then repeats while `2 ^ k` is less than 287.
`);`	Ends the loop.
`Show(2 ^ k);`	Shows the value of `2 ^ k` (512).

As with `For()` loops, `While()` loops that always evaluate as true create an infinite loop, which never stops. To stop the script, press Esc on Windows (or Command-period on macOS). You can also select **Edit > Stop Script**. On macOS, **Edit > Stop Script** is available only when the script is running.

Summation

The Summation() function adds the body results over all i values. The syntax is:

```
Summation( initialization, limitvalue, body );
```

For example:

```
s = Summation( i = 1, 10, i );
```

returns 55, the result of 1+2+3+4+5+6+7+8+9+10.

The script works like this:

s =	Sets the s variable to the value of the function.
Summation(	Begins the Summation() loop.
i = 1,	Sets i to 1.
10,	Sets the limit of i to 10.
i	All values of i from 1 to 10 are added together, resulting in 55.
);	Ends the loop.

This behavior is similar to Σ in the Formula Editor. The following expression:

```
Summation( i = 1, N Row(), x ^ 2 );
```

is equivalent to the following formula in the Formula Editor:

$$\sum_{i=1}^{\text{NRow}} x^2$$

Product

The Product() function is similar to Summation() except that it multiplies the body results rather than adding them. The syntax is the same as for Summation(). For example:

```
p = Product( i = 1, 5 , i );
```

returns 120, the result of 1*2*3*4*5.

In this example, the initial value of i is 1, the upper limit is 5, then all integer values of i up to 5 are multiplied.

Here is the equivalent in the Formula Editor:

$$\prod_{i=1}^{5} i$$

Break and Continue

The `Break()` and `Continue()` functions give you more control over looping. `Break()` immediately stops the loop and proceeds to the next expression that follows the loop. `Continue()` is a gentler form of `Break()`. It immediately stops the current iteration of the loop and continues with the next iteration of the loop.

Break

`Break()` is typically used inside a conditional expression. For example:

```
For( i = 1, i <= 5, i++,
   If( i == 3, Break() );
   Print( "i=" || Char( i ) );
);
```

results in:

```
"i=1"
"i=2"
```

The script works like this:

`For(`	Begins the `For()` loop.
`i = 1,`	Sets `i` to 1.
`i <= 5,`	As long as `i` is less than or equal to 5, continues evaluating the loop.
`i++,`	Increments `i` by 1. Note that this step is done after the `If` loop is evaluated.
`If(`	Begins the `If()` loop.
`i == 3, Break()`	If `i` is equal to 3, breaks the loop.
`);`	Ends the loop.
`Print(`	When `i` is not equal to 3, opens the `Print()` loop.
`"i="`	Prints the string `"i="` to the log.
`||`	Places `"i="` on the same line as the value that follows.

`Char(i));`	Prints the value of `i` to the log. The `For()` loop then repeats until the value of i is less than or equal to 5, breaking when i equals 3, and breaking and printing only when i is less than 3.
`);`	Ends the loop.

Note that when the `If()` and `Break()` expressions follow `Print()`, the script prints the values of `i` from 1 to 3, because the loop breaks after "i=3" is printed.

```
"i=1"
"i=2"
"i=3"
```

Continue

As with `Break()`, `Continue()` is typically used inside a conditional expression. For example:

```
For( i = 1, i <= 5, i++,
   If( i < 3, Continue() );
   Print( "i=" || Char( i ) );
);
```

results in:

```
"i=3"
"i=4"
"i=5"
```

The script works like this:

`For(`	Begins the `For()` loop.
`i = 1,`	Sets `i` to 1.
`i <= 5,`	Evaluates 1 as less than or equal to 5.
`i++,`	Increments `i` by 1. Note that this step is done after the `If` loop is evaluated.
`If(`	Begins the `If()` loop.
`i < 3, Continue()`	Evaluates `i` as 1 and continues as long as `i` is less than 3.
`);`	Ends the `If()` loop.

`Print(`	When `i` is no longer less than 3, opens the `Print()` loop.
`"i="`	Prints the string `"i="` to the log.
`||`	Places `"i="` on the same line as the value that follows.
`Char(i));`	Prints the value of `i` to the log. The `For()` loop then repeats until the value of `i` is less than or equal to 5, continuing and printing only when i is equal to or greater than 3.
`);`	Ends the loop.

Conditional Functions

JSL provides five functions to evaluate an expression conditionally:

- "If"
- "Match"
- "Choose"
- "Interpolate"
- "Step"

If

The `If()` function evaluates the first result expression when its condition evaluates as true (a nonzero or nonmissing value). Otherwise, it evaluates the second result expression.

The syntax is:

```
If ( condition, result1, result2 );
```

For example, the following script returns "Young" when the age is less than 12. Otherwise, the script returns "Young at Heart".

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "age group", "Character" );
For Each Row(
   :age group = If( :age <= 12,
      "Young",
      "Young at Heart"
   )
);
```

You can also string together multiple conditions and results. The syntax is:

```
If( condition1, result1,
  condition2, result2,
  ...,
  resultElse );
```

In the preceding example, if condition1 is not true, the function continues evaluating until it finds a true condition. Then that condition's result is returned.

The last result is returned when all conditions are false. And when a value is missing, the missing value is returned. For these reasons, it's very important to include a default result at the end of the expression. Consider the following example, which recodes gender abbreviations in Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( sex =
  If(
     sex == "F", "Female",
     sex == "M", "Male",
     "Unknown" );
);
```

The script works like this:

`For Each Row( sex =`	For each row in the table, sex is the column that is recoded.
`If(`	Begins the `If()` loop.
`sex == "F", "Female",`	If the value of sex is F, replaces the value with Female.
`sex == "M", "Male",`	If the value of sex is M, replaces the value with Male.
`"Unknown" );`	If neither of the above conditions are true, replaces the value with Unknown. If this result were omitted and the value of sex were missing, the script would return a missing value.
`);`	Ends the loop.

You can also put actions and assignments in the result expression. The following example assigns 20 to `x`, because the first condition (`y < 20`) is false:

```
y = 25;
z = If( y < 20, x = y, x = 20 );
```

Note: Be careful to use two equal signs (==) for equality tests, not one equal sign (=). An `If` with an argument such as *`name=value`* assigns rather than tests the value.

Avoiding Memory Issues with Multiple If Statements

Running a script that contains dozens of If() Statements can cause memory problems. We recommend reducing the If() nesting. Suppose that your script contains 100 `If()` statements in this format:

```
If( val, If( val, If( val, If( val, If( val,
If( val, If( val, If( val, If( val, If( val,
...
) ) ) ) ) ) ) ) ) );
```

Rewrite the script:

```
If( val, val1, If(val, val2, If( val, val3,
If( val, val4, If( val, val5, If( val, val6, ...) ) ) ) ) );
```

You can also try setting the maximum call depth. The `Maximum Call Depth` preference sets the default for the maximum call depth (or stack size) in which JSL built-in functions, user-defined functions, or `Recurse()` function calls can be made. By default, the maximum call depth is set to 256.

In JMP, your JSL script has a limited amount of physical runtime stack memory in which to perform JSL function calls. By default, this size is set to 2MB. Increasing the maximum call depth can cause a physical runtime stack overflow, so incrementally increase this preference in small amounts until you find the best value that works for your JSL script.

```
Preferences[1] << Set( Maximum JMP call depth( 64 ) );
```

Match

You can use the `Match()` function to make several equality comparisons without needing to rewrite the value to be compared. The syntax is:

```
Match( x, value1, result1, value2, result2, ..., resultElse );
```

For example, the following script recodes gender abbreviations in Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( sex =
  Match(
     sex,
     "F", "Female",
     "M", "Male",
     "Unknown");
  );
```

The script works like this:

`For Each Row( sex =`	For each row in the table, `sex` is the column that is recoded.
`Match(`	Begins the `Match()` loop.
`sex,`	Specifies `sex` as the match argument.
`"F", "Female",`	If the value matches "F" replace it with “Female”.
`"M", "Male",`	If the value matches "M", replace it with "Male".
`"Unknown" );`	If F or M are not matched, replaces the value with "Unknown".
`);`	Ends the loop.

This `Match()` example is a simplified version of the example in “If”. The advantage of `Match()` is that you define the comparison value once rather than repeat it in each condition. The disadvantage is that you cannot use expressions with operators as you can with `If`; the argument `sex == "F"` returns an error in a `Match()` expression.

With more groups of conditions and results, the value of `Match()` becomes more apparent. The following script would require many additional lines of code with `If()`.

```
dt = Open( "$SAMPLE_DATA/Travel Costs.jmp" );
For Each Row(
   Booking Day of Week = Match( Booking Day of Week,
      "Sunday", "SUN",
      "Monday", "MON",
      "Tuesday", "TUE",
      "Wednesday", "WED",
      "Thursday", "THU",
      "Friday", "FRI",
      "Saturday", "SAT",
      "Not Specified"
   )
);
```

Be careful with the data type of the condition and result. In the preceding example, the conditions and results are both character data. If the data types do not match, JMP automatically changes the column’s data type. The results might not be what you want.

Note: The `Match()` function explicitly checks to see if the compare expression *x* is missing and if the value of *value1* is missing, then it returns the value of `result1`; otherwise it continues to compare the expression *x* to each *valueN* value in each *valueN/resultN* pair, ignoring any missing values. If the expression *x* is equal to any of the *valueN* value, then the corresponding *resultN* value is returned. If no matching *valueN* value is found, then the *resultElse* value is returned.

Choose

The `Choose()` function shortens scripts even more than `Match()`, provided the arguments are tested against integers. The syntax is:

```
Choose( expr, result1, result2, result3, ..., resultElse );
```

Suppose you have a data table with a group column of numeric values from 1 through 7. If the first cell contains the number 1, the following script returns x = "Low".

```
x = ( Choose( group[1], "Low", "Medium", "High", "Unknown" ); );
Show( x );
```

The script works like this:

`x =`	Creates the x variable.
`Choose(`	Begins the `Choose()` loop.
`group[1],`	Evaluates the value of group in the first row.
`"Low",`	If the value of group is 1, return "Low".
`"Medium",`	If the value of group is 2, return "Medium".
`"High",`	If the value of group is 3, return "High".
`"Unknown"`	Otherwise, return "Unknown".
`)`	Ends the loop.
`);`	Closes the x variable.
`Show( x );`	Returns the value of x.

If the expression evaluates to an out-of-range integer (such as 7 when only 4 replacement values are listed), the last result is returned. In the preceding example, "Unknown" is returned.

Notice that `If()` and `Match()` require more code to achieve the same results as the `Choose` function:

```
If(
  group[1] == 1, "Low",
  group[1] == 2, "Medium",
  group[1] == 3, "High",
  "Unknown"
);
Match( group, 1, "Low", 2, "Medium", 3, "High", "Unknown" );
```

Note: If the data types in the expression do not match, JMP automatically changes the column's data type.

Interpolate

The `Interpolate()` function performs linear interpolation for continuous data or bilinear interpolation for two-dimensional continuous data.

In the simplest case, the function finds the `y` value that corresponds to a given `x` value between two sets of points. You might use `Interpolate()` to calculate missing values between data points.

The data points can be specified as alternating ordered arguments:

```
Interpolate( x, x1, y1, x2, y2, ... );
```

or as matrices containing the `x` and `y` values:

```
Interpolate( x, xmatrix, ymatrix );
```

Suppose that your data set includes the height of individuals from age 20 through 25. However, there is no data point for age 23. To estimate the height for 23-year-olds, use interpolation. The following example shows the value that you want to evaluate (age 23), followed by matrices for ages (20 through 25) and heights (59 through 75).

```
Interpolate( 23, [20 21 22 24 25], [59 62 56 69 75] );
```

returns:

```
62.5
```

The value 62.5 is halfway between the `y` values 56 and 69, just as 23 is halfway between the `x` values 22 and 24.

You can also interpolate multiple points by specifying a matrix or list of values in the first argument.

```
Interpolate( [23, 24.5], [20 21 22 24 25], [59 62 56 69 75] );
```

returns:

```
[62.5, 72]
```

Bilinear Interpolation

You can also interpolate in two-dimensions:

```
Interpolate( {x, y}, xvector, yvector, zmatrix );
```

Here, the first argument is a list of two points, the second and third arguments are vectors that define the grid of `x` and `y` values, and the fourth argument is a matrix of data points. The function then finds the interpolated `z` value within the appropriate quadrant of the `zmatrix`. The appropriate quadrant is found by comparing the `x` and `y` values to the `xvector` and `yvector` arguments.

Suppose you have a 2x3 matrix and you want to interpolate a point that is halfway between the first and second rows of points and halfway between the second and third columns of points.

```
Interpolate( {0.5, 0.75}, [0 1], [0 0.5 1], [10 15 20, 12 16 18] );
```

returns:

```
17.25
```

Note that 0.5 is halfway between 0 and 1 for the x values (corresponding to the rows of the matrix) and 0.75 is halfway between 0.5 and 1 for the y values (corresponding to the columns of the matrix). You can find the bilinear interpolation in two steps:

1. Find the halfway points between the second and third columns for the each row of the matrix:
   - Row 1: 17.5 is halfway between 15 and 20.
   - Row 2: 17 is halfway between 16 and 18.
2. Find the halfway point between the two points in the previous step: 17.25.

Notes:

- The `x` values must be in ascending order. For example, `Interpolate(2,1,1,3,3)` returns 2. However, `Interpolate(2,3,3,1,1)` returns a missing value (.).
- `Interpolate()` is best used for continuous data, but `Step()` is designed for discrete data. See "Step".

Step

The `Step()` is like `Interpolate()` except that it finds the corresponding `y` for a given `x` from a step-function fit rather than a linear fit. Use `Step()` with discrete `y` values (that is, when the `y` value's corresponding `x` value can be only y_1 or y_2). However, when the `y` value's corresponding `x` value can fall between y_1 and y_2, use `Interpolate()`.

As with `Interpolate`, the data points can be specified as a list:

```
Step( x, x1, y1, x2, y2, ... );
```

or as matrices containing the `x` and `y` values:

```
Step( x, xmatrix, ymatrix );
```

Suppose that your data table shows the discount percentage for purchases of $25, $50, $75, and $100. You want to create a graph that shows the discount for a $35 purchase, which the data table does not specify. The following example shows the value that you want to evaluate, 35, followed by matrices for purchases from $25 to $100.

```
Step( 35, [25 50 75 100], [5 10 15 25] );
```

returns:

```
5
```

If the discounts were on a sliding scale (in this example, between 5 and 10, you would use `Interpolate()`:

```
Interpolate( 35, [25 50 75 100], [5 10 15 25] );
```

returns:

```
7
```

As with `Interpolate()`, the data points must create a positive slope.

Compare Incomplete or Mismatched Data

Comparing data that contains missing values can return misleading results unless you specify a condition that is always true or use functions such as `Is Missing()` or `Zero Or Missing()`. Comparisons of data with mismatched types (numeric versus character) or data in matrices can also be confusing.

Table 5.5 shows examples of such comparisons and matrices and explanations of the results. For a review of operators used in comparisons, see "Operators". For an overview of comparing data, see the *JSL Syntax Reference*.

Note: Matrices must include the same number of columns or rows.

Table 5.5 Some Special-Case Comparison Tests

Test	Result	Explanation
`m=.; m==1`	.	An equality test with a missing value returns missing.
`m=.; m!=1`	.	An inequality test with a missing value returns missing.
`m=.; m<1; m>1; and so on`	.	A comparison with a missing value returns missing (unless it could not possibly be true, see next).
`m=.; 1<m<0`	0	A comparison involving a missing value that could not possibly be true returns false; false takes precedence over missing for comparisons with more than two arguments (as with logical operators).
`{a, b}==List(a, b)`	1	An equality test of list arguments returns a single result.
`{a, b}<{a, c}`	.	A comparison test of list arguments is not allowed.
`1=="abc"`	0	An equality test with mixed data types returns false.
`1<="abc"`	.	A comparison with mixed data types returns missing.
`[1 2 3]==[2 2 5]`	`[0 1 0]`	An equality test of matrices returns a matrix of elementwise results. When a matrix is compared to a matrix, comparison is done element-by-element and returns a matrix of 1s and 0s.
`[1 2 3]==2`	`[0 1 0]`	An equality test of a matrix and a matrix filled with 2s. If a matrix is compared to a number, the number is treated as a matrix filled with that number.
`[1 2 3] < [2 2 5]`	`[1 0 1]`	A comparison of matrices returns a matrix of elementwise results.
`[1 2 3] < 2`	`[1 0 0]`	A comparison of a matrix and a matrix filled with 2s.
`Is Missing(m)`	1	Returns 1 for a missing value and returns 0 otherwise. For missing character values, you can also use empty quotes for the comparison, as in m == "".
`Zero Or Missing(m)`	1	Returns 1 when the value is 0 or missing. The argument must be numeric or a matrix and not a string.
`All([2 2]==[1 2])`	0	Summarizes elementwise comparisons; returns 1 only if *all* comparisons are true and returns 0 otherwise.

Table 5.5 Some Special-Case Comparison Tests *(Continued)*

Test	Result	Explanation
`Any([2 2]==[1 2])`	1	Summarizes elementwise comparisons; returns 1 if *any* comparison is true and returns 0 otherwise.

Missing Values

In a comparison, missing values typically return missing, not true or false. For this reason, it is very important to include a result that is always true. Suppose that a data table column contains the values 1, 2, 3, and a missing value in column A. A formula in column B sets up the comparison.

```
New Table( "Testing Comparisons",
   Add Rows( 4 ),
   New Column( "A",
      Numeric,
      "Continuous",
      Format( "Best", 10 ),
      Set Values( [1, 2, 3, .] )
   ),
   New Column( "B", Character, "Nominal", Formula( If( :A, "true", 1, "false"
   ) ) )
);
```

The following values are stored in column B:

```
"true"
"true"
"true"
"false"
```

The script works like this:

`If(`	Begins the comparison.
`:A, "true",`	If the value of A is nonmissing and nonzero, the result is "true". This comparison is true for the first three rows.
`1, "false"`	The value of 1 is always true, so the missing value returns "false".
`);`	Closes the comparison.

Notes:

- When a cell contains a non-empty string, the comparison fails with the first comparison. The following script returns a missing value if a is a string. It works for numeric values.

```
a = "a";
If( a, "true", "false" );
```

- When one value is false and another one is missing, `Or()` returns missing. Use `OrMZ()` to return false instead. See the *JSL Syntax Reference*.
- When one value is true and another one is missing, `And()` returns missing. Use `AndMZ()` to return false instead. See the *JSL Syntax Reference*.
- `MatchMZ()` is similar to `Match()` except that missing values are treated as 0.
- `IfMZ()` is similar to `If()` except that missing values are treated as 0.

Is Missing

If you know that some values are missing, you can also compare with `Is Missing()`. The comparison in the preceding example can be rewritten to return "missing" for missing values:

```
If( :A, "true", Is Missing( :A ), "missing", "false" );
```

The preceding expression returns "true" when A is nonmissing and nonzero, "missing" when A is missing, and "false" otherwise.

Zero Or Missing

If the missing value could be 0, use the `Zero Or Missing()` function instead:

```
Zero Or Missing( A );
```

This expression returns 1 when A is 0 or missing.

Tip: You cannot compare a known value with an explicitly defined missing value, only with variables, matrices, or other things that could *contain* missing values.

Inquiry Functions

Inquiry functions identify the type of an element, such as a string, list, or matrix. You can then write a script specific to that element type.

JMP also uses inquiry functions to determine the writability of a directory or file and to identify a computer's operating system and the JMP version.

General Element Types

The Type() function returns a string naming the type of the resulting value. For example:

```
Show( Type( 1 ), Type( "hi" ), Type( {"a", 2} ), Type( [10 24 325] ) );
```

results in:

```
Type(1) = "Integer"
Type("hi") = "String"
Type({"a", 2}) = "List"
Type([10 24 325]) = "Matrix";
```

Specific Element Types

Other inquiry functions (such as Is Matrix(), Is List(), Is Scriptable(), and so on) let you test for specific types of objects. In the following example, Is Matrix() evaluates as true, then the specified calculations are run:

```
a = [2 3];
b = [1, 1];
c = a * b;
If( Is Matrix( c ),
   (c ^ a) / (a * b),
   Print( "c is not a matrix." )
);
   [5 25]
```

Is Scriptable() returns 1 when the object is scriptable. Four variables in the following example refer to a data table, column, platform, and report. All four objects are scriptable, so Is Scriptable() returns 1 for each example.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = Column( "weight" );
plat = Bivariate( Y( :weight ), X( :height ) );
rep = Report( plat );
Show( Is Scriptable( dt ) );
   Is Scriptable(dt) = 1;
Show( Is Scriptable( col ) );
   Is Scriptable(col) = 1;
Show( Is Scriptable( plat ) );
   Is Scriptable(plat) = 1;
Show( Is Scriptable( rep ) );
   Is Scriptable(rep) = 1;
```

Is Empty() tests to see whether a variable has a value, a function, an expression, or a reference to an object. It returns 1 if the variable is undefined or holds the Empty() value.Otherwise, you get errors when referring to something that has not been created or assigned a value yet. Programmers call this an *uninitialized variable*.

Here is an example of a test to see whether a data table is opened and therefore assigned to the `dt` variable. If a data table is not opened, the `Open()` function prompts the user to open the table.

```
If( Is Empty( dt = Current Data Table() ),
   dt = Open()
);
```

You can use `Is Empty()` for any variable (such as global variable, local variable, and columns).

Table 5.6 shows functions that identify object types.

Table 5.6 Inquiry Functions That Identify Object Types

Syntax	Explanation
`Is Associative Array(`*x*`)`	Returns 1 if the evaluated argument is an associative array or 0 otherwise.
`Is Directory(x)`	Returns 1 if the x argument is a directory and 0 otherwise.

`Is Empty(`*global*`)`
`Is Empty(`*dt*`)`
`Is Empty(`*col*`)`
	Returns 1 if the global variable, data table, or data column does not have a value (is uninitialized) or 0 otherwise.

`Is Expr(`*x*`)`	Returns 1 if the evaluated argument is an expression or 0 otherwise.
`Is File(x)`	Returns 1 if the x argument is a file and 0 otherwise.
`Is List(`*x*`)`	Returns 1 if the evaluated argument is a list or 0 otherwise.
`Is Matrix(`*x*`)`	Returns 1 if the evaluated argument is a matrix or 0 otherwise.
`Is Name(`*x*`)`	Returns 1 if the evaluated argument is a name or 0 otherwise. See "Retrieve a Stored Expression Rather than its Result".
`Is Namespace(`*x*`)`	Returns 1 if the evaluated argument is a namespace or 0 otherwise.
`Is Number(`*x*`)`	Returns 1 if the evaluated argument is a number or missing numeric value or 0 otherwise.
`Is Scriptable(`*x*`)`	Returns 1 if the evaluated argument is a scriptable object or 0 otherwise.
`Is String(`*x*`)`	Returns 1 if the evaluated argument is a string or 0 otherwise.
`Type(`*x*`)`	Returns a string naming the type of x.

Object Attributes

JMP provides the following functions to determine whether a file or directory is writable before attempting to write to them. Use these functions in combination with `Is Directory(path)` and `Is File(path)` to verify a script destination and attribute. See the *JSL Syntax Reference* for additional information.

The `Is Directory Writable(path)` function returns 1 if the directory specified in the path argument is writable and 0 otherwise.

The `Is File Writable(path)` function returns 1 if the file specified in the path argument is writable and 0 otherwise.

For example, the following code verifies the path refers to a directory and then checks to ensure the directory is writable:

```
If( Is Directory( "$SAMPLE_DATA/Loss Function Templates" ),
  If( Is Directory Writable( "$SAMPLE_DATA/Loss Function Templates" ),
     "Directory is writable.",
     "Directory is read only!"
  ),
  "Is a read only directory."
);
```

Host Information

The `Host Is()` inquiry function identifies the current operating system. Then actions specific to that operating system can be performed.

The following example specifies text sizes in reports for different operating systems. If you commonly write your scripts on Windows and share them with macOS users, the results can look different from what you intended. For example, the following expression sets the text to a larger size (12) on macOS and a smaller size (10) on Windows:

```
textsize = If( Host is( "Mac" ), 12, 10 );
```

Version Information

The `JMP Version()` inquiry function returns the JMP version as a string. You might use this function to determine the JMP version and then run a script compatible with that version.

```
JMP Version(); // returns "16.0.0" in JMP 16
JMP Version(); // returns " 9.0.0" in JMP 9
```

Notice that a leading blank is inserted before versions less than 10.0.0. This blank helps when comparing version numbers. Without the leading blank, 9.0.0 is interpreted as greater than 10.0.0.

Chapter 6

Types of Data

Working with Numbers, Strings, Dates, Currency, and More

This chapter discusses basic data types:

- numbers and strings
- paths, which are a special type of string
- dates and times, which can be either special numbers or special strings
- currency
- hexadecimal values and blobs

At the end of the chapter are two sections that show more advanced methods of interacting with strings and pattern matching with regular expressions.

Contents

Numbers and Strings 145
Formatting Numbers 146
Unicode Characters 147
Path Variables 147
Create and Customize Path Variables 150
Relative Paths 151
File Path Separators 151
Date-Time Functions and Formats 152
Date-Time Values 152
Program with Date-Time Functions 153
Date-Time Values in Data Tables 160
Custom Date-Time Formatting with Format Patterns 164
Currency 166
Hexadecimal and BLOB Functions 167
Work with Character Functions 170
Concat 170
Munger 171
Repeat 172
Substitute and Substitute Into 173
Regular Expressions 174
Regex 174
Regex Match 176
Special Characters in Regular Expressions 178
Escaped Characters in Regular Expressions 179
Greedy and Reluctant Regular Expressions 181
Backreferences and Capturing Groups 183
Lookaround Assertions 184
Pattern Matching 185
Parsing Strings in Fixed Fields 186
Patterns and Case 188
Troubleshooting Patterns 188

Numbers and Strings

Numbers can be written as integers, decimal numbers, in scientific notation with an E preceding the power of ten, and as date-time values. A single period by itself is the missing numeric value.

For example, these are all numbers:

```
.   1   12   1.234  3E3  0.314159265E+1 1E-20
```

One or more characters placed within double quotation marks constitute a *string*. For example, these are all strings:

```
"Green" "Hello,\NWorld!" "54"
```

Notice that if a number is in quotation marks, it is a string, not a number. There are two functions you can use to change a number into a string or a string into a number.

- Use `Num()` to convert a string into a number. For example:

```
Num( "54" );
   54
```

Note: `Num()` cannot convert non-numeric characters, so it produces a missing value.

```
Num( "Hello" );
   .
```

- Use `Char()` to convert a number into a string. For example:

```
Char( 54 );
   "54"
Char( 3E3 )
   "3000"
```

To preserve locale-specific numeric formatting in `Num()` or `Char()` output, include the `<<Use Locale(1)` option as shown in the following example:

```
Char( 42, 5, 2, << Use Locale( 1 ) );
   // results in the character value "42,00" in the French locale
```

To look at each character in a string, use the `Substr()` function. This example looks for the letter "a" in the string and prints a message to the log:

```
ch = Substr( "alphabetic", 1, 1 ); // start and end with the first character
If( ch == "a",
   Print( "First letter is a." )
);
   "First letter is a."
```

Formatting Numbers

The `Format()` function provides options for formatting numbers with thousands separators, decimal points, geographic symbols, *p*-values, and so on. The following example shows a basic format for a fixed decimal column.

```
Format( x, "Fixed Dec", 10, 2, "Use thousands separator");
```

The format starts with the number to be formatted. Then you can specify the width of the number, the number of decimal places, and whether to use a thousands separator.

Here are other examples:

```
Format( x, "Currency", 20, <<Use Locale(0) );
  // ignores computer locale and uses a period for the decimal separator
Format( x, "Currency", 20, "Use thousands separator" );
Format( x, "m/d/y", 10 );
Format( x, "Precision", 10, 2, "Keep trailing zeroes", "Keep all whole digits"
  );
Format( x, "Latitude DDD", "PUNDIR"); // "PUN" for punctuation, "DIR" for
  direction, PUNDIR for both
Format( x, "Custom", Formula( Abs( value ) ), 15 );
```

The syntax for `Function()` arguments is shown in the *JSL Syntax Reference*. You can also find the argument names and options in the data table Column Info window.

See "Date-Time Functions and Formats" and "Currency" for more information about formatting date-time values and currency.

Note: If the date format is unknown, an error is written to the log.

Matrix Box, Number Col Box, Number Col Edit Box, Number Edit Box

To format numbers in `Matrix Box()`, `Number Col Box()`, `Number Col Edit Box()`, and `Number Edit Box()`, send the `Set Format` message to the display box. You can then specify the same arguments as for the `Format()` function, such as the following:

```
<<Set Format( 10, 2, "Use thousands separator");
<<Set Format( "Currency", "EUR", 20 );
```

The syntax for `Set Format` arguments is shown in the *JSL Syntax Reference*. You can also find the argument names and options in the data table Column Info window.

Note: If you specify the number of decimal places, that number must be preceded with the width.

Unicode Characters

JMP supports both Unicode UTF-8 and UTF-16 standards for encoding and representing text for most of the world languages. See the The Unicode Consortium for code charts and more information on the Unicode standard.

To display Unicode characters in JMP, precede the Unicode code for the character with '\!'. For example:

- Greek letter sigma (σ) in Unicode = U+03C3; in JMP, use \!U03C3
- Greek letter mu (μ) in Unicode = U+03BC; in JMP, use \!U03BC

To use Unicode to express superscripts and subscripts:

- subscript 1 ($_1$) in Unicode = U+2081; in JMP, use \!U2081
- superscript 2 (^{2}) in Unicode = U+00B2; in JMP, use \!U00B2

To express x^2 in Unicode, in JMP, use \!U0078\!U00B2.

To use Unicode surrogate pairs, specify the Unicode escape sequence twice. If you specify an emoji Unicode character, you might need to change the font for it to appear correctly. The following example shows how to include an emoji in a text box.

```
New Window( "Emoji Example",
   V List Box(
      Text Box( "Surrogate Pair with space: \!UD83D \!UDE00", Set Font( "Segoe
   UI Emoji" ) ),
      Text Box( "Surrogate Pair without space: \!UD83D\!UDE00", Set Font(
   "Segoe UI Emoji" ) )
   )
);
```

Path Variables

Path variables are shortcuts to directories or files. Rather than enter the entire path to the directory or file, you use the path variable in a script. A path variable is a special type of string and is always contained within double quotation marks.

One common predefined path variable in JMP is $SAMPLE_DATA. This variable points to the sample data folder in your JMP or JMP Pro installation folder. The following example opens the Big Class.jmp sample data table.

```
Open( "$SAMPLE_DATA/Big Class.jmp" );
```

Several path variables are predefined in JMP. The following table shows the definitions for the current JMP version. Variables in previous versions of JMP might differ.

Table 6.1 Predefined Path Variable Definitions

Variable	Path
ADDIN_HOME(com.your.addin.id) Defines the folder where the specified add-in is stored.	Add-In Builder stores the specified add-in in the following folders, based on your operating system: • Windows: "C:/Users/<username>/AppData/Roaming/SAS/JMP/Addins/" • macOS: "/Users/<username>/Library/Application Support/JMP/Addins/" For example, `Convert File Path( "$ADDIN_HOME(com.your.addin.id)` evaluates to the location of the `com.your.addin.id` add-in. To install your add-ins in a shared network location, use the `Register Addin()` function. See "Register an Add-In Using JSL".
ALL_HOME Defines a folder that can be accessed by all users on a machine.	• Windows (JMP): "/C:/ProgramData/SAS/JMP/16/" • Windows (JMP Pro): "/C:/ProgramData/SAS/JMPPro/16/" • macOS: "/Library/Application Support/JMP/16/" To see if the directory exists, run the script `Is Directory("$ALL_HOME");`. If the folder exists, 1 is returned.
BUILTIN_SCRIPTS	• Windows (JMP): "/C:/Program Files/SAS/JMP/16/Resources/Builtins/" • Windows (JMP Pro): "/C:/Program Files/SAS/JMPPRO/16/Resources/Builtins/" • macOS: "/Applications/JMP 16.app/Contents/Resources/Builtins/"
DESKTOP	• Windows: "/C:/Users/<username>/Desktop/" • macOS "/Users/<username>/Desktop/"
DOCUMENTS	• Windows: "/C:/Users/<username>/Documents/" • macOS: "/Users/<username>/Documents/"
DOWNLOADS	• Windows: "/C:/Users/<username>/Downloads/" • macOS: "/Users/<username>/Downloads/"
GENOMICS_HOME	"/<JMP Genomics installation directory>/"

Table 6.1 Predefined Path Variable Definitions *(Continued)*

Variable	Path
HOME	• Windows (JMP): "/C:/Users/<username>/AppData/Roaming/SAS/JMP/16/" • Windows (JMP Pro): "/C:/Users/<username>/AppData/Roaming/SAS/JMPPro/16/" • macOS: "/Users/<username>/"
JMP_HOME	• Windows: /C:/<JMP installation directory> **Note:** The variable is not available on macOS.
SAMPLE_APPS	• Windows: /C:/<JMP installation directory>/Samples/Apps/" • macOS: "/Library/Application Support/JMP/16/Samples/Apps/"
SAMPLE_DASHBOARDS	• Windows: /C:/<JMP installation directory>/Samples/Dashboards/" • macOS: "/Library/Application Support/JMP/16/Samples/Dashboards/"
SAMPLE_DATA	• Windows: "/C:/<JMP installation directory>/Samples/Data/" • macOS: "/Library/Application Support/JMP/16/Samples/Data/"
SAMPLE_IMAGES	• Windows: "/C:/<JMP installation directory>/Samples/Images/" • macOS "/Library/Application Support/JMP/16/Samples/Images/"
SAMPLE_IMPORT_DATA	• Windows: "/C:/<JMP installation directory>/Samples/Import Data/" • macOS: "/Library/Application Support/JMP/16/Samples/Import Data/"
SAMPLE_PROJECTS	• Windows: "/C:/<JMP installation directory>/Samples/Projects/" • macOS: "/Library/Application Support/JMP/16/Samples/Projects/"
SAMPLE_SCRIPTS	• Windows: "/C:/<JMP installation directory>/Samples/Scripts/" • macOS: "/Library/Application Support/JMP/16/Samples/Scripts/"
TEMP	• Windows: "/C:/Users/<username>/AppData/Local/Temp/" • macOS: "var/folders/..."

Table 6.1 Predefined Path Variable Definitions *(Continued)*

Variable	Path
USER_APPDATA Changes to JMP preferences, menus, and the Home Window are stored here, along with Debugger session settings.	• Windows (JMP): "/C:/Users/<username>/AppData/Roaming/SAS/JMP/16/" • Windows (JMP Pro): "/C:/Users/<username>/AppData/Roaming/SAS/JMPPro/16/" • macOS: "/Users/<username>/Library/Application Support/JMP/16/"

Path variable definitions are updated automatically based on the version of JMP you are using. For example, when you run a JMP 14 script in JMP 16, the JMP 16 path variable definitions are used.

To see the definition of any path variable, use the function `Get Path Variable`:

```
Get Path Variable( "HOME" );
   "/C:/Users/<username>/AppData/Roaming/SAS/JMP/16/"
```

Note that you don't include a dollar sign for `Set Path Variable()` or `Get Path Variable()`. But you must include the dollar sign when using the variable in a script.

Trailing Slashes

Make sure to include a trailing slash after the path variable. In the following example, the root name `"Big Class"` is assigned to the dtName variable. The `Open()` expression evaluates `$SAMPLE_DATA` and the trailing slash and then appends the `dtName` value along with the file extension `.jmp`.

```
dtName = "Big Class";
dt = Open( "$SAMPLE_DATA/" || dtName || ".jmp" );
```

The path is interpreted as:

```
C:/Program Files/SAS/JMP/16/Samples/Data/Big Class.jmp
```

Without the slash that follows `$SAMPLE_DATA`, the path is interpreted as:

```
C:/Program Files/SAS/JMP/16/Samples/DataBig Class.jmp
```

Create and Customize Path Variables

You can create your own path variables or override some of the built-in variables with the `Set Path Variable()`. In the following example, the path variable is called *root*. The variable points to the `c:/` directory.

```
Set Path Variable( "root", "c:/" );
```

To get the value of the new variable, use Get Path Variable().

```
Get Path Variable( "root" );
  "c:/"
```

Use your path variable as you would other variables. The following expression opens the myimportdata.txt file in the c:/ directory.

```
Open( "$root/myimportdata.txt" );
```

As with getting path variables, omit the dollar sign when setting path variables.

Relative Paths

If you plan to use relative paths in variables, you must set the default directory. Then any path not preceded by a drive letter is relative to the default directory. Here is an example:

```
Set Default Directory( "c:/users/smith/data" );
```

To return the value of the default directory, use Get Default Directory().

```
Get Default Directory(); // returns "c:/users/smith/data"
```

So the following expression resolves as C:/users/smith/data/cleansers.jmp:

```
Open("cleansers.jmp");
```

File Path Separators

In JMP, the preferred file path format is the Portable Operating System Interface (POSIX), or UNIX, format with forward slashes (/) as separators. This means that you do not have to identify the current operating system in scripts run on both Windows and macOS. However, each host still accepts its native format for compatibility.

You can a convert file path format from Windows to POSIX (and vice versa) using Convert File Path(). Converting from a POSIX to a Windows path might be useful when you need to output a path to a file or to another application. The syntax is:

```
Convert File Path (path, <absolute|relative>, <POSIX|windows>, <base(path)>);
```

For example, the following script converts a POSIX path to a Windows path:

```
Convert File Path( "c:/users/smith", windows);
  c:\users\smith
```

You can substitute a path variable (such as $HOME) for the path inside quotes.

Date-Time Functions and Formats

A date-time value consists of any portion of a date or time. The value can be seconds (3388594698), a complete date (such as "Wednesday, May 18, 2011"), the date and time ("05/18/2011 8:18:18 PM"), the week number (3), and so on.

JMP lets you convert date-time values to common formats, perform arithmetic on the values, and manipulate the data in a number of ways.

Tip: For descriptions of all date-time functions and their arguments, see the *JSL Syntax Reference*.

Date-Time Values

Date-time values are stored and calculated as the number of seconds since midnight, January 1, 1904. For example:

```
Today(); // returns 3388649872 on May 19, 2011 at 12:00:00 AM
```

As with `Today()`, the `Date DMY()` and `Date MDY()` functions also return month, day, and year arguments as seconds. For example, if it were 12:00:00 a.m. on May 19, 2011, all of the following statements would return the same value:

```
Date DMY( 19, 5, 2011 );
Date MDY( 5, 19, 2011 );
Today();
   3388608000
```

The `As Date()` function takes the number of seconds and displays it as a date or duration.

- Values that represent one year or more are returned as dates:

```
As Date( 3388608000 );
   19May2011
```

- Values that represent less than a year are returned as durations.

```
As Date( 50000 );
   :0:13:53:20
```

You can use date-time values in two ways:

- a literal value, for example `19May2011:10:10`
- a string, for example `"Thursday, May 19, 2011"`

You can perform arithmetic with date-time literals, which use the number of seconds as the base number.

```
As Date( 19May2011 + 1 );
```

19May2011:00:00:01

Program with Date-Time Functions

Table 6.2 shows functions that convert seconds into date-time values and date-time values into seconds.

Table 6.2 Date-Time Functions

Function	Explanation
Abbrev Date(*date*)	Returns a string representation for the *date* supplied. The format is based on your computer's regional setting. So for the English (United States) locale, the date is formatted like "02/29/2004". Even if you are running JMP in English with a different locale, the locale format is applied.

As Date(*expression*)	Formats a number or expression so that it shows as a date or duration in a text window. For example, values that represent one year or more are returned as dates.
	`x = As Date( 8Dec2000 + In Days( 2 ) );`
	shows as:
	10Dec2000
	Values that represent less than a year are returned as durations.
	`As Date( 50000 );`
	shows as:
	:0:13:53:20

Date DMY(*day*, *month*, *year*)	Returns the specified date expressed as the number of seconds since midnight, 1 January 1904. For example, the second Leap Day of the third millennium is DateDMY(29,2,2004), which returns 3160857600.
Date MDY(*month*, *day*, *year*)	Returns the specified date expressed as the number of seconds since midnight, 1 January 1904. For example, the second Leap Day of the third millennium is DateMDY(2,29,2004), which returns 3160857600.
Day Of Week(*date*)	Returns an integer representation for the day of the week of the *date* supplied. Weeks are Sunday–Saturday.

Table 6.2 Date-Time Functions *(Continued)*

Function	Explanation
`Day Of Year(date)`	Returns an integer representation for the day of the year of the *date* supplied.
`Day(date)`	Returns an integer representation for the day of the month of the *date* supplied.
`Format(date, "format")`	Returns the *value* in the *format* specified in the second argument. Most typically used for formatting datetime values from a number of seconds to a formatted date. Format choices are those shown in the **Column Info** dialog box. Also see Table 6.3.
`Hour(datetime)`	Returns an integer representation for the hour part of the *date-time* value supplied.
`In Days(n)`	These functions return the number of seconds per *n* minutes, hours, days, weeks, or years. Divide by these functions to express an interval in seconds as an interval in other units.
`In Hours(n)`	
`In Minutes(n)`	
`In Weeks(n)`	
`In Years(n)`	
`Long Date(date)`	Returns a string representation for the specified *date*. The format is based on your computer's regional setting. So for the English (United States) locale, the date is formatted like `"Sunday, February 29, 2004"`. Even if you are running JMP in English with a different locale, the locale format is applied.
`MDYHMS(date)`	Returns a string representation for the *date* supplied, formatted like `"2/29/2004 00:02:20 AM"`.
`Minute(date-time)`	Returns an integer representation for the minute part of the *date-time* value supplied.
`Month(date)`	Returns an integer representation for the month of the *date* supplied.
`Num(date-time)`	Returns an integer representation for the *date-time* value supplied.
`InFormat(string, "format")` `Parse Date(string, "format")`	Parses a *string* of a given *format* and returns datetime value expressed as if surrounded by `As Date()`, returning the date in `dd`Monyyyy format.

Table 6.2 Date-Time Functions *(Continued)*

Function	Explanation
`Second(`*`date-time`*`)`	Returns an integer representation for the second part of the *`date-time`* value supplied.
`Short Date(`*`date`*`)`	Returns a string representation for the *`date`* supplied, in the format MM/DD/YYYY, regardless of locale (for example, "02/29/2004").
`Time Of Day(`*`date`*`)`	Returns an integer representation for the time of day of the *`date-time`* supplied.
`Today()`	Returns the current date and time expressed as the number of seconds since midnight, 1 January 1904. No arguments are accepted, but the parentheses are still needed.
`Week Of Year(`*`date`*`, <rule_n>)`	Returns the week of the year as a date-time value. Three rules determine when the first week of the year begins. • With rule 1 (the default), weeks start on Sunday, with the first Sunday of the year being week 2. Week 1 is a partial week or empty. • With rule 2, the first Sunday begins with week 1, with previous days being week 0. • With rule 3, the ISO-8601 week number is returned. Weeks start on Monday. Week 1 is the first week of the year with four days in that year. It is possible for the first or last three days of the year to belong to the neighboring year's week number.
`Year(`*`date`*`)`	Returns an integer representation for the year of the specified *`date`*.

Examples of Common Date-Time Functions

You can use any function that returns seconds within a function that returns a date-time.

For example, if today is May 19, 2011 and the time is 11:37:52 AM, `Today()` returns the number of seconds, and the functions that follow show that number of seconds since the base time in different date-time formats:

```
Today()
   3666955847
Short Date( Today() );
   "03/13/2020"
```

```
Long Date( Today() );
   "Friday, March 13, 2020"
Abbrev Date( Today() );
   "3/13/2020"
MDYHMS( Today() );
   "03/13/2020 2:50:17 PM"
```

The date argument in parentheses can be seconds (or any function that returns seconds), or any date-time literal value. For example, both of the following expressions return the same value:

```
Long Date( 3388649872 );
Long Date( 19May2011 );
   "Friday, March 13, 2020"
```

Note: `Long Date()` and `Abbrev Date()` values are formatted according to your computer's regional settings.

Extract Parts of Dates

You can extract parts of date values using the functions `Month()`, `Day()`, `Year()`, `Day Of Week()`, `Day Of Year()`, `Week Of Year()`, `Time Of Day`, `Hour()`, `Minute()`, and `Second()`, which all return integers. If today is May 24th, 2011, each of the following examples returns the 144th day of the year:

```
Day of Year( Today() );
Day of Year( 24May2011 );
Day of Year( Date MDY( 5, 24, 2011 ) );
   144
```

Example

A data table column named Date contains date-time values that are formatted as `"m/d/y"`. You want to create a column that shows only the time. In the following script, the second column's formula extracts the time of day from the Date value in the first column.

```
New Table( "Assembly Tests",
   Add Rows( 1 ),
   New Column( "Date",
      Numeric, Continuous,
      Format( "m/d/y" ),
      Set Values( [3389083557] )
   ),
   New Column( "Time",
      Numeric, Continuous,
      Formula( Format( Time Of Day( :Date ), "h:m:s") )
   )
);
```

Figure 6.1 shows the result. Note that the time of day does not appear in the Date column, because the `Format` function applies the "m/d/y" format.

Figure 6.1 Example of Extracting the Time

	Date	Time
1	05/24/2011	12:05:57 PM

Rules for Determining the Week of the Year

`Week of Year()` returns the week of the year as a date-time value. Three rules determine when the first week of the year begins.

- With rule 1 (the default), weeks start on Sunday, with the first Sunday of the year being week 2. Week 1 is a partial week or empty.

```
Week Of Year( Date DMY( 19, 6, 2013 ), 1 );
  25
```

- With rule 2, the first Sunday begins with week 1, with previous days being week 0.

```
Week Of Year( Date DMY( 19, 6, 2013), 2 ) ;
  24
```

- With rule 3, the ISO-8601 week number is returned. Weeks start on Monday. Week 1 is the first week of the year with four days in that year. It is possible for the first or last three days of the year to belong to the neighboring year's week number.

```
Week Of Year( Date DMY( 19, 6, 2013 ), 3 );
  25
```

Arithmetic on Dates

You can perform the usual arithmetic operations with date-time data as with any other numeric data. One option is simple arithmetic, such as subtracting a number from a date-time value.

Another option is writing a formula to perform the arithmetic.

Example

The Date column in your data table shows when a customer uses his credit card to buy gas. You want to know how many days elapse between purchases. The following script creates a Days elapsed column. The formula in that column subtracts the Date value in the current row from that of the previous row.

```
New Table( "Gas Purchases",
  Add Rows( 3 ),
  New Column( "Date",
```

```
        Numeric,
        "Continuous",
        Format( "m/d/y" ),
        Set Values( [3392323200 3393532800 3394828800] )
    ),
    New Column( "Days elapsed",
        Formula(
        // returns a missing value for the first row
            If( Row() == 1,
                .,
                (:Date[Row()] - :Date[Row() - 1]) / In Days()
            )
        )
    )
);
```

Figure 6.2 shows the result.

Figure 6.2 Example of Calculating Date-Time Values

	Date	Days elapsed
1	07/01/2011	•
2	07/15/2011	14
3	07/30/2011	15

Time Intervals

The `In Minutes`, `In Hours`, `In Days`, `In Weeks`, and `In Years` functions are used to express time intervals in units rather than seconds. Each of these functions returns the number of seconds associated with a particular period of time. For example, the following expression returns the number of weeks between now and July 4, 2012.

```
(Date DMY( 04, 07, 2012 ) - Today()) / In Weeks();
  -208.225444775132
```

When the argument for the interval function is empty, JMP counts by 1. You can enter another number to change the count. For example, `In Years(10)` converts the interval to decades. The following expression returns the number of decades between now and December 31, 2037.

```
(Date DMY( 31, 12, 2037) - Today() ) / In Years( 10 );
  2.18927583529799
```

Two-Digit and Four-Digit Years

JMP applies its own algorithms for interpreting and displaying datetime strings rather than supporting operating system-specific datetime formats. However, JMP uses the date-time separators selected in the Region and Language control panel (Windows) or the Date & Time preferences (macOS) to interpret and display dates.

Two-digit years are interpreted according to the current system clock year and JMP rules. For example, when the year in a script is 11, and you run the script after 1990, the year shows as 2011.

```
Long Date( 25May11 );
  "Wednesday, May 25, 2011"
```

To avoid ambiguity, enter four-digit years. The following expression returns 1911 (rather than 2011) as indicated:

```
Long Date( 25May1911 );
  "Thursday, May 25, 1911"
```

Table 6.3 explains how JMP interprets two-digit years.

Table 6.3 How JMP Interprets Two-Digit Years

Two-Digit Year Value	When it is Evaluated	Result	Examples	Result
00–10	before 1990 (on Windows)	19__	enter 5 in year 1979	1905
	before or during 1990 (on macOS)			
	during or after 1990 (on Windows)	20__	enter 5 in year 1991	2005
	after 1990 (on macOS)			
11–89 (on Windows)	any time	current century	enter 13 in year 1988	1913
11–90 (on macOS)			enter 13 in year 2024	2013
90–99 (on Windows)	before 2011	19__	enter 99 in year 1999	1999
91–99 (on macOS)	during or after 2011	20__	enter 99 in year 2015	2099

Note: JMP always displays four-digit years regardless of the regional settings. If you need to show two-digit years, use character string functions. See "Types of Data".

Date-Time Values in Data Tables

Change Date-Time Input and Display Formats

In data tables, JMP can accept the input of date-time values in one format (the *input format*), store them internally as the number of seconds since the base date, and display them in a different date-time format. The `Informat()` and `Format()` functions give you this control.

- `Informat()` takes a string date-time value, defines the date format used in that string, and returns the date in `ddMonyyyy` format.

```
Informat( "19May2011 11:37:52 AM", "ddMonyyyy h:m:s" );
   19May2011:11:37:52
```

- `Format()` takes the number of seconds since the base date (or a date-time function that returns that number) and returns the date in the specified format.

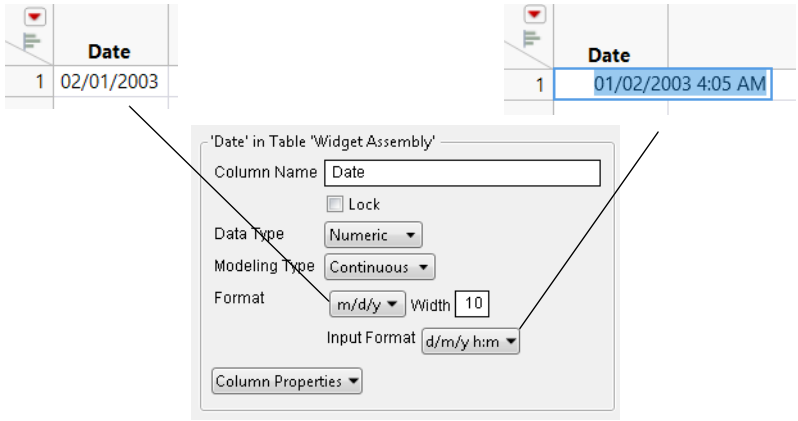
```
Format( 3388649872, "ddMonyyyy h:m:s" );
   "19May2011 11:37:52 AM"
Format( Today(), "ddMonyyyy h:m:s" );
   "19May2011 11:37:52 AM"
```

Suppose that you are entering dates into a column using the `d/m/y h:m` format, but you want to see the dates in the `m/d/y` format. `Input Format` defines the input format, and `Format` defines the display format. For example,

```
New Table( "Widget Assembly",
   Add Rows( 1 ),
   New Column( "Date",
      Numeric,
      "Continuous",
      Format( "m/d/y" ),
      Input Format( "d/m/y h:m" ),
      Set Values( [3126917100] )
   )
);
```

The `Format` and `Input Format` values are shown in the data table's column properties (Figure 6.3). Note that when you click in the cell to edit it, the date-time value appears in the input format. When you edit the value, or add a new value, the format specified in the data table column Format list is used to display the value.

Figure 6.3 Example of Date-Time Display and Input Values



Notes:

- In a script that converts a column from character to numeric, specify `Format()` and `Informat()` to prevent missing values. See "Convert Character Dates to Numeric Dates".
- The date-separator character on your computer might differ from the forward slash (/) character shown in the *Scripting Guide*.
- You can enter time values in 24-hour format (military time) or with AM or PM designators.

Table 6.4 describes the formats used as arguments in date-time functions or as data table formats. You can also use the formats for the *`format`* argument to a `Format` message to a data column. See "Set or Get Formats".

For descriptions of specific date-time functions, see the *JSL Syntax Reference*.

Notes:

- Date-time formats in which a colon or T separate the date and time always use a 24-hour format. Other formats use your computer's regional settings.
- If the format is unknown, an error is written to the log.

Table 6.4 Date-Time Formats

Type	Format argument	Example
Date only	`"m/d/y"`	`"01/02/1999"`
	`"mmddyyyy"`	`"01021999"`
	`"m/y"`	`"01/1999"`
	`"d/m/y"`	`"02/01/1999"`

Table 6.4 Date-Time Formats *(Continued)*

Type	Format argument	Example
	"ddmmyyyy"	"02011999"
	"ddMonyyyy"	"02Jan1999"
	"Monddyyyy"	"Jan021999"
	"y/m/d"	"1999/01/02"
	"yyyymmdd"	"19990102"
	"yyyy-mm-dd"	“1999-01-02“
	"yyyyQq"	1999Q1
Date and time	"m/d/y h:m"	"01/02/1999 13:01" "01/02/1999 1:01 PM"
	"m/d/y h:m:s"	"01/02/1999 13:01:55" "01/02/1999 1:01:55 PM"
	"d/m/y h:m"	"02/01/1999 13:01" "02/01/1999 1:01 PM"
	"d/m/y h:m:s"	"02/01/1999 13:01:55" "02/01/1999 1:01:55 PM"
	"y/m/d h:m"	‘1999/01/02 13:01’ ‘1999/01/02 1:01 PM’
	"y/m/d h:m:s"	‘1999/01/02 13:01:02 ‘1999/01/02 1:01:02 PM’
	"ddMonyyyy h:m"	"02Jan1999 13:01" "02Jan1999 1:01 PM"
	"ddMonyyyy h:m:s"	"02Jan1999 13:01:02" "02Jan1999 1:01:02 PM"
	"ddMonyyyy:h:m"	"02Jan1999:13:01" "02Jan1999:1:01 PM"
	"ddMonyyyy:h:m:s"	"02Jan1999:13:01:02" "02Jan1999:1:01:02 PM"
	"Monddyyyy h:m"	"Jan021999 13:01" "Jan021999 1:01 PM"
	"Monddyyyy h:m:s"	"Jan021999 13:01:02" "Jan021999 1:01:02 PM"

Table 6.4 Date-Time Formats *(Continued)*

Type	Format argument	Example
Day number and time	":day:hr:m"	"34700:13:01" ":33:001:01 PM"
	":day:hr:m:s"	"34700:13:01:02" ":33:001:01:02 PM"
	"h:m:s"	"13:01:02" "01:01:02 PM"
	"h:m"	"13:01" "01:02 PM"
	"yyyy-mm-ddThh:mm"	1999-01-02T13:01
	"yyyy-mm-ddThh:mm:ss"	1999-01-02T13:01:02
Duration	":day:hr:m"	"52:03:01" reads fifty-two days, three hours, and one minute
	":day:hr:m:s"	"52:03:01:30" reads fifty-two days, three hours, one minute, and thirty seconds
	"hr:m"	"17:37" reads seventeen hours and thirty-seven minutes
	"hr:m:s"	"17:37:04" reads seventeen hours, thirty-seven minutes, and 4 seconds
	"min:s"	"37:04" reads thirty-seven minutes and 4 seconds
Note: The following formats display the date-time according to your computer's regional settings. They are available only for the display of dates, not for date input in a data table. Examples are shown for the United States locale.		
Abbreviated date	"Date Abbrev"	(Display only) "01/02/1999"
Long date	"Date Long"	(Display only) "Saturday, January 02, 1999"
Locale date	"Locale Date"	(Display only) "01/02/1999"

Table 6.4 Date-Time Formats *(Continued)*

Type	Format argument	Example
Locale date and time	"Locale Date Time h:m"	(Display only) "01/02/1999 13:01" or "01/02/1999 01:01 PM"
	"Locale Date Time h:m:s"	(Display only) "01/02/1999 13:01:02" or "01/02/1999 01:01:02 PM"

Custom Date-Time Formatting with Format Patterns

Format patterns are strings that define a date-time format. Here's one example:

```
<YYYY></><MM></><DD> <hh><:><mm><:><ss><ampm>
```

The parts of the pattern in angle brackets are called *field descriptors*. They represent a value (such as <YYYY>, which is a four-digit year) or other date-time text (such as </>, which is a locale-specific date separator).

The conversion of a date-time value to a string and a string converted to a date-time value work as follows:

- When a date-time value is converted to a string, the field descriptors are replaced by the appropriate value or symbols.
- When a string is converted to a date-time value, a value is or matching symbols are expected to be found where each field descriptor is. Anything outside a format descriptor is treated as literal text. The literal text is replaced as-is when it is converted *to* a string, and is expected to be found where specified when converting *from* a string.

Table 6.5 Output for Custom Date-Time Formats

Format	Output
<YYYY></><MM></><DD> <hh><:><mm><:><ss><ampm>	2020/02/10 11:54:33 AM
<DayOfWeek>, <Month> <D>, <YYYY>	Wednesday, February 5, 2020
<Day>d <hh24>h <mm>m <ss>s	123d 02h 34m 56s

Note: <Day> is the day count, used as the most significant field of a duration. <Hour> and <Minute> field descriptors are also available.

Example of Outputting the Date-Time Value to the Log

The following script uses `Informat()` to read a string date-time value, defines the format pattern used in that string, and returns the date in `ddMonyyyy` format. The script then outputs the date-time value to the log.

```
dateVal = Informat( "June 16, 2016", "Format Pattern", "<Month> <D>, <YYYY>"
  );
  16Jun2016
```

Here's another way to specify the format pattern:

```
date2 = 16Jun2021;

  dayofwk2 = Format( date2, "Format Pattern", "<DayOfWeek>" );
  monthnew2 = Format( date2, "Format Pattern", "<Month>" );

  YYYYMMDD = Format( date2, "Format Pattern", "<YYYY></><MM></><DD>" );
  YYMMMDD = Format( date2, "Format Pattern", "<YY></><MMM></><DD>" );
  "21/Jun/16"
```

Example of Changing the Input Format of a Column

The following script shows how to change the input format of a column. The Input Format appears when you edit the value in a data table.

```
New Table( "temp_date",
  Add Rows( 2 ),
  New Column( "date1",
     Numeric,
     "Continuous",
     // specify how the value appears in the data table
     Format( "Format Pattern", "<Month> <D>, <YYYY>", 35 ),
     // specify how the value appears when you edit the value
     Input Format( "Format Pattern", "<YYYY>*<MM>*<DD>" ),
     Set Values( {"2018*10*31", "2018*09*09"} )
  ),
  New Column( "date2",
     Numeric,
     "Continuous",
     Format( "Format Pattern", "<YYYY>*<MM>*<DD>", 10 ),
     Set Values( {"2018*02*05", "2018*07*03"} )
  )
);
```

Note: See *Using JMP* for details about creating a format pattern in a data table rather than in a script.

Currency

JMP displays numbers as currency using the `Format()` function, which uses the following syntax:

```
Format(x,"Currency", <"currency code">, <decimal>, < << Use Locale(Boolean));
```

Where:

- *x* is a column or a number
- *"currency code"* is an International Standards Organization (ISO) 4217 code
- *decimal* is the number of decimal places

To illustrate the `Format` function:

```
Format( 12345.6, "Currency", "GBP", 3 );
  "£12,345.600"
```

By default, the operating system locale is used for the currency if the currency code symbol is omitted. For example, running the following script in a Japanese operating system formats the number with the yen symbol.

```
Format( 12345.6, "Currency", 3);
  "¥12,345.600"
```

To ignore the computer's locale and use a period for the decimal separator, include the `Use Locale(0)` argument:

```
Format( 12345.6, "Currency", 3, <<Use Locale( 0 ) );
```

Tip: If you change the region when JMP is open, restart JMP for the `Use Locale` setting to work properly.

If the currency code is not supported by JMP, the currency code string appears before the number.

```
Format( 12345.6, "Currency", "BBD", 3 );
  "BBD 12,345.600"
```

Table 6.6 lists the currencies supported in JMP.

Table 6.6 Currencies Supported in JMP

Code	Currency	Code	Currency	Code	Currency
AUD	Australian dollar	ILS	Israeli new shekel	RUB	Russian ruble
BRL	Brazilian real	INR	Indian rupee	SEK	Swedish krone

Table 6.6 Currencies Supported in JMP *(Continued)*

Code	Currency	Code	Currency	Code	Currency
CAD	Canadian dollar	JPY	Japanese yen	SGD	Singapore dollar
CHF	Swiss franc	KRW	South Korean won	THB	Thai baht
CNY	Chinese yuan	MXN	Mexican peso	TRY	New Turkish lira
COP	Colombian peso	MYR	Malaysian ringgit	TWD	New Taiwan dollar
DKK	Danish krone	NOK	Norwegian krone	USD	US dollar
EUR	Euro	NZD	New Zealand dollar	ZAR	South African rand
GBP	British pound	PHP	Philippine peso		
HKD	Hong Kong dollar	PLN	Polish zloty		

For more information about how to format currency, see "Formatting Numbers".

Hexadecimal and BLOB Functions

JMP can also handle binary (large) objects, commonly called BLOBs. The functions below convert between hexadecimal values, numbers, characters, and BLOBs. Some of the functions are covered in more detail following Table 6.7.

See the *JSL Syntax Reference* for more information.

Table 6.7 Hexadecimal and BLOB Functions

Syntax	Explanation

`Hex("text")`
`Hex("num")`
`Hex("blob")`
	Returns the hexadecimal codes for the characters in *text*, *number*, or *blob*.
	`Char To Hex` is an alias.

`Hex To BLOB("hexstring")`	Returns a BLOB representation of the hexadecimal code supplied as a quoted string.

Table 6.7 Hexadecimal and BLOB Functions *(Continued)*

Syntax	Explanation
`Hex To Char("hexstring", "encoding")`	Returns a character string that corresponds to the hexadecimal code supplied as a quoted string. The default encoding for the character string is `utf-8`. `utf-16le`, `utf-16be`, `us-ascii`, `iso-8859-1`, `ascii~hex`, `shift-jis`, and `euc-jp` are also supported.
`Hex to Number`	Returns the number that corresponds to the hexadecimal code suppled as a quoted string.
`Char To BLOB("string")` `Char To BLOB("string", "encoding")`	Converts a character string into a binary (BLOB). The default encoding for the blob is `utf-8`. `utf-16le`, `utf-16be`, `us-ascii`, `iso-8859-1`, `ascii~hex`, `shift-jis`, and `euc-jp` are also supported.
`BLOB To Char("blob")` `BLOB To Char("blob", "encoding")`	Converts binary data to a character string. The default encoding for the character string is `utf-8`. `utf-16le`, `utf-16be`, `us-ascii`, `iso-8859-1`, `ascii~hex`, `shift-jis`, and `euc-jp` are also supported.
`BLOB Peek("blob", offset, length)`	Returns a new BLOB that is a subset of the given BLOB that is *length* bytes long and begins at the *offset*. Note that the offset is 0-based.
`Matrix to BLOB(matrix, type, bytesEach, endian)`	Makes a BLOB from a matrix by converting the matrix elements to 1-byte, 2-byte, or 4-byte signed or unsigned integers or 4-byte or 8-byte floating point numbers.
`BLOB to Matrix(blob, "type", bytes, "endian", <nCols>)`	Creates a matrix by converting each byte in the `blob` to numbers.

`Hex(string)` returns the hexadecimal codes for each character in the argument. For example,

```
Hex( "Abc" );
```

returns

```
"416263"
```

since 41, 62, and 63 are the hexadecimal codes (in ASCII) for “A”, “b”, and “c”.

`Hex to Char(string)` converts hexadecimal to characters. The resulting character string might not be valid display characters. All the characters must be in pairs, in the ranges 0-9, A-Z, and a-z. Blanks and commas are allowed, and skipped. For example,

```
Hex To Char( "4142" );
```

returns

```
"AB"
```

since 41 and 42 are the hexadecimal equivalents of "A" and "B".

`Hex` and `Hex To Char` are inverses of each other, so

```
Hex To Char( Hex( "Abc" ) );
```

returns

```
"Abc"
```

`Hex To BLOB(string)` takes a string of hexadecimal codes and converts it to a binary object.

```
a = Hex To BLOB( "6A6B6C" );
Show( a );
   a = Char To BLOB("jkl", "ascii~hex")
```

`BLOB Peek(blob,offset,length)` extracts bytes as defined by the arguments from a blob.

```
b = BLOB Peek( a, 1, 2 );
Show( b );
   b = Char To BLOB("kl", "ascii~hex")
b = BLOB Peek( a, 0, 2 );
Show( b );
   b = Char To BLOB("jk", "ascii~hex")
b = BLOB Peek( a, 2 );
Show( b );
   b = Char To BLOB("l", "ascii~hex")
```

`Hex(blob)` converts a blob into hexadecimal.

```
c = Hex( a );
Show( c );
   c = "6A6B6C"
d = Hex To Char( c );
Show( d );
   d = "jkl"
```

`Concat(blob1,blob2)` or `blob1 || blob2` concatenates two blobs.

```
e = Hex To BLOB( "6D6E6F" );
Show( e );
f = a||e;
Show( f );
   e = Char To BLOB("mno", "ascii~hex")
   f = Char To BLOB("jklmno", "ascii~hex")
```

`Length(blob)` returns the number of bytes in a blob.

```
g = Length( f );
```

```
Show( g );
   g = 6
```

Note: When blobs are listed in the log, they are shown with the constructor function `Char To BLOB("...")`.

Any hex code outside the ASCII range (`space` to }, or hex 20 - 7D) is encoded as the three-character sequence [~][hexdigit][hexdigit]. For example,

```
h = Hex To BLOB( "19207D7E" );
Show( h );
i = Hex( h );
Show( i );
   h = Char To BLOB("~19 }~7E", "ascii~hex")
   i = "19207D7E"
```

`Char To BLOB(string)` creates a blob from a string, converting ~hex codes.
`BLOB To Char(blob)` creates a string with ~hex codes to indicate non-visible and non-ASCII codes.

Work with Character Functions

This section shows how to use some of the more complex character functions that are described in the *JSL Syntax Reference*.

Concat

In the `Concat` function, expressions yielding names are treated like character strings, but globals that have the name values are evaluated. The following example demonstrates that if you have a stored name value, you need to either use `Char` before storing it in a global, or `Name Expr` on the global name.

```
n = {abc};
c = n[1] || "def";
Show( c );
   "abcdef"

m = Expr( mno );
c = m || "xyz";
Show( c );
   Name Unresolved: mno in access or evaluation of 'mno' , mno/*###*/
// The /*###*/ characters indicate the location of an error in the script
   execution.

m = Expr( mno );
```

```
c = Name Expr( m ) || "xyz";
Show( c );
   "mnoxyz"

m = Char( Expr( mno ) );
c = m || "xyz";
Show( c );
   "mnoxyz"
```

`Concat Items()` converts a list of string expressions into a single string, with each item separated by a delimiter. If unspecified, the delimiter is a blank. Its syntax is

```
resultString = Concat Items ({list of strings}, <"delimiter string">);
```

For example,

```
a = {"ABC", "DEF", "HIJ"};
result = Concat Items(a, "/");
```

returns

```
"ABC/DEF/HIJ"
```

Alternatively,

```
result = Concat Items( a );
```

returns

```
"ABC DEF HIJ"
```

Munger

`Munger` works many different ways, depending on what you specify for its arguments:

```
Munger(string, offset, find | length, <replace>);
```

Table 6.8 Munger behaviors for various types of arguments

***Find*, *length*, and *replace* arguments**	**Example**
If you specify a string as the *find* and specify no *replace* string, `Munger` returns the position (after *offset*) of the first occurrence *find* string.	`Munger( "the quick brown fox", 1, "quick" );` `5`
If you specify a positive integer as the *length* and specify no *replace* string, `Munger` returns the characters from *offset* to *offset* + *length*.	`Munger( "the quick brown fox", 1, 5 );` `"the q"`

Table 6.8 Munger behaviors for various types of arguments *(Continued)*

***Find*, *length*, and *replace* arguments**	**Example**
If you specify a string as the *find* and specify a *replace* string, Munger replaces the first occurrence after *offset* of *text* with *replace*.	`Munger( "the quick brown fox", 1, "quick", "fast" );` `"the fast brown fox"`
If you specify a positive integer as the *length* and specify a *replace* string, Munger replaces the characters from *offset* to *offset* + *length* with *replace*.	`Munger( "the quick brown fox", 1, 5, "fast");` `"fastuick brown fox"`

If you specify a positive integer as the *length*, and *offset* + *length* exceeds the length of *text*, Munger either returns *text* from *offset* to the end or replaces that portion of *text* with the *replace* string, if it exists.	`Munger( "the quick brown fox", 5, 25);`
	`"quick brown fox"`
	`Munger( "the quick brown fox", 5, 25, "fast" );`
	`"the fast"`

If you specify zero as the *length* and specify no *replace* string, Munger returns a blank string.	`Munger( "the quick brown fox", 1, 0 );` `""`
If you specify zero as the *length* and specify a *replace* string, the string is inserted before the *offset* position.	`Munger( "the quick brown fox", 1, 0, "see " );` `"see the quick brown fox"`
If you specify a negative integer as the *length* value and specify no *replace* string, Munger returns all characters from the offset to the end of the string.	`Munger( "the quick brown fox", 5, -5 );` `"quick brown fox"`
If you specify a negative integer for *length* and specify a *replace* string, Munger replaces all characters from the offset to the end with the *replace* string.	`Munger( "the quick brown fox", 5, -5, "fast" );` `"the fast"`

Repeat

The `Repeat()` function makes copies of its first argument into a result. The second (and sometimes a third) argument is the number of repeats, where 1 means a single copy.

If the first argument evaluates to a character value or list, the result is that many copies.

```
Repeat( "abc", 2 );
```

```
   "abcabc"
Repeat( {"A"}, 2 );
   {"A","A"}
Repeat( {1, 2, 3}, 2 );
   {1,2,3,1,2,3}
```

If the first argument evaluates to a number or matrix, the result is a matrix. The second argument is the number of row repeats, and a third argument can specify the number of column repeats. If only two arguments are specified, the number of column repeats is 1.

```
Repeat( [1 2, 3 4], 2, 3 );
   [  1 2 1 2 1 2,
      3 4 3 4 3 4,
      1 2 1 2 1 2,
      3 4 3 4 3 4]
Repeat( 9, 2, 3 );
   [  9 9 9,
      9 9 9]
```

The `repeat` function is compatible with the function of the same name in the SAS/IML language, but is incompatible with the SAS character DATA step function, which repeats one more time than this function.

Substitute and Substitute Into

The `Substitute()` function returns a copy of a string with a replacement expression. The `Substitute Into()` function changes the original string with a replacement expression.

For example, consider the following script:

```
str1 = str2 = "All things considered";

// str3 holds the result of the Substitute, and str1 is not changed
str3 = Substitute( str1, "All", "Some" );

/* Substitute Into returns nothing, so str4 is missing,
and str2 is changed and now holds the result of Substitute Into */
str4 = Substitute Into( str2, "All", "Some" );
Show( str1, str2, str3, str4 );
```

This script returns the following output:

```
str1 = "All things considered";
str2 = "Some things considered";
str3 = "Some things considered";
str4 = .;
```

Regular Expressions

A regular expression is a specification of a pattern frequently used to clean up or extract pieces of data. You can search for a pattern and replace it with a different string or extract specific parts of the string. Define the pattern in the `Regex()` or `Regex Match()` function.

- "Regex"
- "Regex Match"
- "Special Characters in Regular Expressions"
- "Escaped Characters in Regular Expressions"
- "Greedy and Reluctant Regular Expressions"
- "Backreferences and Capturing Groups"
- "Lookaround Assertions"

Regex

`Regex()` searches for a *`pattern`* within a *`source`* string and returns a string. It simply identifies a pattern in a string or transforms a string into another string.

```
Regex(source, pattern, (<replacement string>, <GLOBALREPLACE>), <format>,
  <IGNORECASE>);
```

`IGNORECASE` disregards case. `GLOBALREPLACE` repeats the match until the entire string is processed. `format` is a backreference to the matched group. `Regex()` returns missing if the match fails.

Example of Matching a String

`bus|car` is the regular expression (in quotation marks because it is a string). The expression means match "bus" or "car".

```
sentence = "I took the bus to work.";
vehicle = Regex( sentence, "bus|car" );
  "bus"
```

Examples of Replacing a String

The third optional argument in `Regex()` is a specification of the result string. The default value, `\0`, is a backreference to everything that was matched by the regular expression. In the preceding example, the word "bus" is matched in `sentence`. The default third argument, `\0`, replaces the entire sentence with "bus".

A more interesting variation uses parentheses to create additional backreferences.

```
sentence = "I took the bus to work.";
Regex( sentence, "(.*) bus (.*)", "\1 car \2" );
   "I took the car to work."
```

The `(.*)` before and after `bus` are part of the regular expression. The parentheses create a capturing group. The `.` matches any character. The * matches zero or more of the previous expression. As a result, the first parenthesis pair matches everything before `bus`, and the second parenthesis pair matches everything after `bus`. The third argument, `\1 car \2`, reassembles the text; it leaves out `bus` and substitutes `car`.

See "Backreferences and Capturing Groups".

Example of Global Replacement

`GLOBALREPLACE` changes the behavior of `Regex()`. If the match succeeds, the entire source string is returned with substitutions made for each place where the pattern matches. If there are no matches, an unchanged source string is returned.

```
sentence = "I took the red bus followed by the blue bus to get to work
   today.";
Regex( sentence, "bus", "car", GLOBALREPLACE);
   "I took the red car followed by the blue car to get to work today."
```

You can also use backreferences. This example starts with a different sentence.

```
sentence = "I took the red bus followed by the blue car to get to work
   today.";
Regex(
   sentence,
   "(\w*) (bus|car)",
   "bicycle (not \2) that was \1",
   GLOBALREPLACE
);
   "I took the bicycle (not bus) that was red followed by the bicycle (not car)
      that was blue to get to work today."
```

The `\w*` matches zero or more word characters and becomes backreference 1 because of the parentheses. `bus|car` becomes backreference 2 because of the parentheses. The third argument, `bicycle (not \2) that was \1`, describes how to build the substitution text for the part of the source text that was matched.

Notice how the backreferences can be used to swap data positions. This might be useful for swapping the position of first names and last names.

Regex Match

`Regex Match()` returns an empty list with zero elements if the match fails. If the match succeeds, the first list is the text of the entire match (backreference `0`). The second list is the text that matches backreference 1, and so on.

```
Regex Match(source, pattern, <replacement>|<MATCHCASE>, <NULL>)
```

Unlike `Regex()`, `Regex Match()` is case insensitive. Include `MATCHCASE` for a case-sensitive match. Include `NULL` if you want to match case but there is no replacement text.

Example of Parsing Name-Value Pairs

The following example parses pairs of names and values.

```
Regex Match(
   "person=Fred id=77 friend= favorite=tea",
   "(\w+)=(\S*) (\w+)=(\S*) (\w+)=(\S*) (\w+)=(\S*)"
);
   {"person=Fred id=77 friend= favorite=tea", "person", "Fred", "id", "77",
      "friend", "", "favorite", "tea"}
```

The `\w+` matches one or more word characters. The `\S*` matches zero or more characters that are not spaces. In the resulting JSL list, the field names (`person, id, friend, favorite`) and their corresponding values (`Fred, 77, "", tea`) are separate strings.

If the first argument to `Regex Match()` is a variable and a third argument specifies the replacement value, the matched text is replaced in the variable.

Comparing Regex and Regex Match

`Regex()` and `Regex Match()` match a pattern in a given string but return different results. To transforms your string into another string, use `Regex()`. To identify the substrings that match specific parts of the pattern, use `Regex Match()`.

This example shows the efficiency of `Regex Match()` compared to `Regex()`. The `source` is a list of six strings. The goal is to extract portions of those six strings into the subject, verb, and object columns of a data table.

Figure 6.4 Final Data Table

	subject	verb	object
1	cat	ate	chicken
2	dog	chased	cat
3	ralph	like	mary
4	girl	pets	dog
5	cat	chased	dog

```
source = {"the cat ate the chicken", "the dog chased the cat", "did ralph like
   mary", "the girl pets the dog",
"these words are strange", "the cat was chased by the dog"};

dt = New Table( "English 101", // create the data table
   New Column( "subject", character ),
   New Column( "verb", character ),
   New Column( "object", character )
);
// iterate through the strings in the list
For( i = 1, i <= N Items( source ), i++,

   // assign the result of each match to matchList
   matchList = Regex Match(
      source[i],

   /* scan each string, match zero or more characters
   and one item in each group */

   ".*?(cat|dog|ralph|girl).*?(ate|chased|like|pets).*?(chicken|cat|mary|dog)
   "
   );

/* If matchList has zero items (string 5), don't add a row
to the table. Put each matched string in separate
data table cells. */
   If( N Items( matchList ) > 0,
      dt << Add Rows( 1 );
      dt:subject = matchList[2]; // match the first open parenthesis
      dt:verb = matchList[3]; // match the second open parenthesis
      dt:object = matchList[4]; // match the third open parenthesis
   );
);
```

`Regex Match()` returns `{"the cat was chased by the dog", "cat", "chased", "dog"}` in a single try with each answer in a separate string. Compare this example to a similar one using `Regex()`, which returns one answer at a time and builds the final string using backreferences.

```
For( i = 1, i <= N Items( source ), i++,
   s = Regex( source[i],
   ".*?(cat|dog|ralph|girl).*?(ate|chased|like|pets).*?(chicken|cat|mary|dog)
   ", "\1" ); // match an item in the first group
   v = Regex( source[i],
   ".*?(cat|dog|ralph|girl).*?(ate|chased|like|pets).*?(chicken|cat|mary|dog)
   ", "\2" ); // match an item in the second group
```

```
      o = Regex( source[i],
      ".*?(cat|dog|ralph|girl).*?(ate|chased|like|pets).*?(chicken|cat|mary|dog)
      ", "\3" ); // match an item in the third group
      If( !Is Missing( s ) & !Is Missing( v ) & !Is Missing( o ),
         dt << Add Rows( 1 );
         dt:subject = s; // return the match for \1
         dt:verb = v; // return the match for \2
         dt:object = o; // return the match for \3
      );
   );
```

Backreferences are discussed in "Backreferences and Capturing Groups".

Special Characters in Regular Expressions

Special characters are commonly used in regular expressions. The period is a special character that matches one instance of the specified character. It must be escaped with a backslash to be interpreted as a period. In the following expression, the period is replaced with an exclamation point.

```
Regex( "Bicycling makes traveling to work fun.", "\.", "!", GLOBALREPLACE );
  "Bicycling makes traveling to work fun!"
```

Table 6.9 describes the special characters and provides examples.

Table 6.9 Special Characters in Regular Expressions

\	• Precedes a literal character. `<\/a>` interprets the forward slash literally in the end HTML anchor tag. • Precedes an escape sequence. `\n` matches a newline character.
^	Matches the beginning of a string, not including the newline character. `^apple` matches "apple" at the beginning of a string.
$	Matches the end of a string, not including the newline character. `apple$` matches "apple" at the end of a string.
.	Matches any single character including a newline character. `.apple` matches any single character and then "apple".
|	Represents a logical OR to separate alternative values. `(apple|orange|banana)` matches "apple", "orange", or "banana".

Table 6.9 Special Characters in Regular Expressions *(Continued)*

?	Matches zero or one instance. `apple (pie)?` matches one or more instances of "pie".
*	Matches zero or more instances.
+	Matches one or more instances.
( )	Encloses a sub-expression. `(apple|orange|banana)` matches "apple", "orange", or "banana". `^(\w+)` matches the beginning of a line and then one or more word characters.
[ ]	Encloses an expression that matches set of characters. `[\s]` matches a whitespace character or a digit. `[a-z0-9]` matches "a" through "z" and numbers "0" through "9".
{ }	Encloses an expression that represents repetition. `apple{3}` repeats three times. `apple{3,}` repeats at least three times as many times as possible. `apple{3, 10}` repeats three times but no more than 10 times. Append a question mark to indicate repeating as few times as possible. For example, `apple{3,}?` repeats at least three times as few times as possible.

Escaped Characters in Regular Expressions

The backslash in a regular expression precedes a literal character. You also escape certain letters that represent common character classes, such as `\w` for a word character or `\s` for a space. The following example matches word characters (alphanumeric and underscores) and spaces.

```
Regex(
   "Are you there, Alice?, asked Jerry.", // source
   "(here|there).+(\w+).+(said|asked)(\s)(\w+)\." ); // regular expression
   "there, Alice?, asked Jerry."
```

`(here|there).+`	Matches "there", a comma, and a space.
`(\w+)`	Matches "Alice".

`.+`	Matches "?, ".
`(said|asked)(\s)`	Matches "asked" followed by a space. Without the space, the match would end here; "asked" is followed by a space in the source string.
`(\w+)\.`	Matches "Jerry" and a period.

Table 6.10 describes the escaped characters supported in JMP. `\C`, `\G`, `\X`, and `\z` are not supported.

Table 6.10 Escaped Characters

`\\`	single backslash
`\A`	start of a string
`\b`	word boundary. The zero-length string between `\w` and `\W` or `\W` and `\w`.
`\B`	not at a word boundary
`\c`*X*	ASCII control character
`\d`	single digit `[0-9]`
`\D`	single character that is NOT a digit `[^0-9]`
`\E`	stop processing escaped characters
`\l`	match a single lowercase letter `[a-z]`
`\L`	single character that is not lowercase `[^a-z]`
`\Q`	ignore escaped characters until `\E` is found
`\r`	carriage return
`\s`	single whitespace character
`\S`	single character that is NOT white space
`\u`	single uppercase character `[A-Z]`
`\U`	single character that is not uppercase `[^A-Z]`
`\w`	word character `[a-zA-Z0-9_]`
`\W`	single character that is NOT a word character `[^a-zA-Z0-9_]`

Table 6.10 Escaped Characters *(Continued)*

\x*00*-\x*FF*	hexadecimal character
\x{*0000*}-\x{*FFFF*}	Unicode code point
\Z	end of a string before the line break

Greedy and Reluctant Regular Expressions

The ?, *, and + operators are greedy by default. They match as many of the preceding character as possible. The ? operator makes them reluctant; ?? matches 0, then 1 if needed; +? matches 1 and then additional characters; *? matches 0 and then additional characters.

The following example starts at the letter n and compares it to the first \d (digits) in the pattern. No digit matches. Because the pattern does not begin with ^ (start of line), the matcher advances to u. The process repeats until the 3 matches the first \d and the 2 matches the second \d.

```
Regex( "number=32.5", "\d\d" );
   "32"
```

Change the pattern to use the greedy + (match one or more).

```
Regex( "number=324.5", "\d+" );
   "324"
```

The preceding example begins much the same, but as soon as the 3 is found and the \d matches, the + greedily matches the 2 and the 4.

Usually, the greedy behavior makes pattern matching faster because the string is consumed sooner. Sometimes a reluctant behavior is better. Adding the ? after the * or + changes them from greedy to reluctant.

```
Regex( "number=324.5", "\d+?" );
   "3"
```

Here, + requires at least one match of a digit character, but ? changes it from "as many as possible" to "as few as possible". It stopped after the 3 because the pattern was satisfied.

Compare the following results.

Greedy:

```
Regex( "number=324.5", "(\d+)(\d+)\.", "first=\1 second=\2" );
   "first=32 second=4"
```

Reluctant:

```
Regex( "number=324.5", "(\d+?)(\d+?)\.", "first=\1 second=\2" );
   "first=3 second=24"
```

In the greedy example above, the matcher greedily matched 3, 2, and 4 for the first \d+. The matcher then had to give back the 4 so that the second \d+ could match something. The reluctant example followed a different path to get a different answer. Initially, the second value was 2, but the pattern could not match the period to the 4, so the second \d+? reluctantly matched the 4 as well.

Use the Reluctant Match for Speed

The greedy and reluctant matches usually produce the same result but not always. See the previous section. One reason you might need the reluctant match is for speed. Suppose that you have a million-character string that begins "The quick fox…" and you want to find the word before "fox". You might write the following expression and expect \1 to contain "quick".

```
The (.+) fox
```

\1 might contain "quick" eventually, after the .+ grabs the million characters to the end of the string and then gives them up, one at a time, until "fox" is found. If there is more than one "fox", it will be the last fox, not this one. To speed it up and make sure we get the first fox, add the ? operator.

```
The (.+?) fox
```

The ? advances one character at a time to get past "quick" and find the first "fox". This method is much faster than going too far.

Typically, the + or * operator is applied to a more restrictive expression such as \d* to match a run of digits, and greedy is faster than reluctant.

Aside from the multiple fox possibility, greedy and reluctant eventually get the same answer. Using the right operator speeds up the match. The right one might be greedy, or it might be reluctant. It depends on what is being matched.

The greedy .* finds the last fox after backing up.

```
Regex(
   "The quick fox saw another fox eating grapes",
   "The (.*) fox",
   "\1"
);
   "quick fox saw another"
```

The reluctant .*? stops on the first fox.

```
Regex(
   "The quick fox saw another fox eating grapes",
   "The (.*?) fox",
   "\1"
);
   "quick"
```

The greedy `.*` has to back up a lot. There is no second fox.

```
Regex(
  "The quick fox saw another animal eating grapes",
  "The (.*) fox",
  "\1"
);
  "quick"
```

The greedy word character match is an even better choice for this problem.

```
Regex(
  "The quick fox saw another fox eating grapes",
  "The (\w*) fox",
  "\1"
);
  "quick"
```

Backreferences and Capturing Groups

A regular expression can consist of patterns grouped in parentheses, also known as capturing groups. In `([a-zA-Z])\s([0-9])`, `([a-zA-Z])` is the first capturing group; `([0-9])` is the second capturing group.

Use a backreference to replace the pattern matched by a capturing group. In Perl, these groups are represented by the special variables `$1`, `$2`, `$3`, and so on. (`$1` indicates text matched by the first parenthetical group.) In JMP, use a backslash followed by the group number (`\1`, `\2`, `\3`).

The following example includes a third argument that specifies the replacement text and backreferences.

```
Regex(
  "   Are you there, Alice?, asked Jerry.", // source
  " (here|there).+ (\w+).+(said|asked) (\w+)\.", // regular expression
"  I am \1, \4, replied \2." ); // optional format argument
  "  I am there, Jerry, replied Alice."
```

`" I am \1,`	Creates the text “I am”, a space, and then the first matched pattern, “there”.
`\4,`	Creates the text “Jerry” with the fourth matched pattern `(\w+)`.
`replied \2."`	Creates the text “replied” and a space. Matches “Alice.” with the second matched pattern `(\w+)`.

Lookaround Assertions

Lookaround assertions check for a pattern but do not return that pattern in the results. *Lookaheads* look forward for a pattern. *Lookbehinds* look back for a pattern.

Negative Lookahead Example

Negative lookaheads check for the absence of a pattern before a specific pattern. `?!` indicates a negative lookahead. The following expression matches a comma *not* followed by a number or space and replaces the pattern with a comma and space:

```
Regex( "one,two 1,234 cat,dog,duck fish, and chips, to go",
   ",(?!\d|\s)", ", ", GLOBALREPLACE );
   "one, two 1,234 cat, dog, duck fish, and chips, to go"
```

Positive Lookahead Example

Positive lookaheads check for the presence of a pattern before a specific pattern. `?=` indicates a positive lookahead. The following expression has the same result as the preceding negative lookahead but matches a comma followed by any lowercase character:

```
Regex( "one,two 1,234 cat,dog,duck fish, and chips, to go",
   ",(?=[a-z])", ", ",GLOBALREPLACE );
   "one, two 1,234 cat, dog, duck fish, and chips, to go"
```

Positive Lookbehind Example

In this example, the positive lookbehind regular expression matches the “ssn=” or “salary=” keywords without including the keyword in the matched text. The matched text is the string of characters that consists of zero or more dollar signs, digits, and hyphens.

```
data = "name=bill salary=$5 ssn=123-45-6789 age=13,name=mary salary=$6
   ssn=987-65-4321 age=14";
redacted = Regex(data, "(?<=(ssn=)|(salary=))[$\d-]*", "###", GLOBALREPLACE);
   "name=bill salary=### ssn=### age=13,name=mary salary=### ssn=### age=14"
```

Here is another way to get the same result using a backreference substitution. `((ssn=)|(salary=))` is the capturing group. `"\1"` is the backreference to that group.

```
data = "name=bill salary=$5 ssn=123-45-6789 age=13,name=mary salary=$6
   ssn=987-65-4321 age=14";
redacted = Regex(data, "((ssn=)|(salary=))[$\d-]*", "\1###", GLOBALREPLACE);
   "name=bill salary=### ssn=### age=13,name=mary salary=### ssn=### age=14"
```

Backreferences are discussed in “Backreferences and Capturing Groups”.

Pattern Matching

Pattern matching in JSL is a flexible method for searching and manipulating strings.

You define and use pattern variables just like any JMP variable:

```
i = 3; // a numeric variable
a = "Ralph"; // a character variable
t = textbox("Madge"); // a display box variable
p = ( "this" | "that" ) + patSpan(" ")
     + ( "car" | "bus" ); // a pattern variable
```

When the above statement executes, `p` is assigned a pattern value. The pattern value can be used either to construct another pattern or to perform a pattern match. The `patSpan` function returns a pattern that matches a span of characters specified in the argument; `patSpan("0123456789")` matches runs of digits.

```
p2 = "Take " + p + "."; // using p to build another pattern
If( Pat Match( "Take this bus.", p2 ), // performing a match
   Print( "matches" ),
   Print( "no match" )
);
```

Sometime all you need to know is that the pattern matched the source text, as above. Other times, you might want to know what matched; for example, was it a bus or a car?

```
p = ("this" | "that") + Pat Span( " " ) + ("car" | "bus") >?
vehicleType; // conditional assignment ONLY if pattern matches
If( Pat Match( "Take this bus.", p ),
   // do not use vehicleType in the ELSE because it is not set
   Show( vehicleType ),
   Print( "no match" )
);
```

You could pre-load `vehicleType` with a default value if you do not want to check the outcome of the match with an `if`. The >? conditional assignment operator has two arguments, the first being a pattern and the second a JSL variable. >? constructs a pattern that matches the pattern (first argument) and stores the result of the match in the JSL variable (second argument) after the pattern succeeds. Similarly, >> does not wait for the pattern to succeed. As soon (and as often) as the >> pattern matches, the assignment is performed.

```
findDelimString = Pat Len( 3 ) >> beginDelim + Pat Arb() >? middlePart
+Expr( beginDelim );
testString = "SomeoneSawTheQuickBrownFoxJumpOverTheLazyDog'sBack";
rc = Pat Match( testString, findDelimString, "<<<" || middlePart || ">>>" );
Show( rc, beginDelim, middlePart, testString );
```

The above example shows a third argument in the `patMatch` function: the replacement string. In this case, the replacement is formed from a concatenation (`||` operator) of three strings. One of the three strings, `middlePart`, was extracted from the `testString` by `>?` because the replacement cannot occur unless the pattern match succeeds (`rc == 1`).

Look at the pattern assigned to `findDelimString`. It is a concatenation of 3 patterns. The first is a `>>` operator that matches 3 characters and assigns them to `beginDelim`. The second is a `>?` operator that matches an arbitrary number of characters and, when the entire match succeeds, assigns them to `middlePart`. The last is an unevaluated expression, consisting of whatever string is in `beginDelim` at the time the pattern is executing, *not* at the time the pattern is built. Just like `expr()`, the evaluation of its argument is postponed. That makes the pattern hunt for two identical three letter delimiters of the middle part.

Other pattern functions might be faster and represent the problem that you are trying to solve better than writing a lot of alternatives; for example, `"a"|"b"|"c"` is the same as `Pat Any("abc")`. The equivalent example for `Pat Not Any("abc")` is much harder. Similar to `Pat Span` (above), `Pat Break("0123456789")` matches up to, but not including, the first number.

Here is a pattern that matches numbers with decimals and exponents and signs. It also matches some degenerate cases with no digits; look at the pattern assigned to digits.

```
digits = Pat Span( "0123456789" ) | "";

number = (Pat Any( "+-" ) | "") >? signPart + (digits) >? wholePart + ("."
+digits | "") >? fractionPart + (Pat Any( "eEdD" ) + (Pat Any( "+-" ) | "")
 + digits | "") >? exponentPart;

If( Pat Match( "-123.456e-78", number ),
   Show( signPart, wholePart, fractionPart, exponentPart )
);
```

Parsing Strings in Fixed Fields

Sometimes data is in fixed fields. The `Pat Tab()`, `Pat R Tab()`, `Pat Len()`, `Pat Pos()`, and `Pat R Pos()` functions make it easy to split out the fields in a fixed field string. `Pat Tab()` and `Pat R Tab()` work from the left and right end of the string and take a number as their argument. They succeed by matching forward to the specified tab position. For example:

```
p = Pat Pos(10) + Pat Tab(15);
```

`Pat Pos(10)` matches the null string if it is in position 10. So at match time, the matcher works its way forward to position 10, then `Pat Tab(15)` matches text from the current position (10) forward to position 15. This pattern is equivalent to `Pat Pos(10)+patLen(5)`. Another example:

```
p = Pat Pos(0) + Pat R Tab(0);
```

This example matches the entire string, from 0 characters from the start to 0 characters from the end. the `Pat Rem()` function takes no argument and is shorthand for `Pat R Tab(0)`; it means the remainder of the string. Pattern matching can also be anchored to the beginning of the string like this:

```
Pat Match( "now is the time", Pat Len(15) + Pat R Pos(0), NULL, ANCHOR );
```

The above pattern uses `NULL` rather than a replacement value, and `ANCHOR` as an option. Both are uppercase, as shown. `NULL` means that no replacement is done. `ANCHOR` means that the match is anchored to the beginning of the string. The default value is `UNANCHORED`.

Patterns can be built up like this, but this is *not* recursive:

```
p = "a" | "b"; // matches one character
p = p + p; // two characters
p = p + p; // four characters
Pat Match( "babb", Pat Pos(0) + p + Pat R Pos(0) );
```

A recursive pattern refers to its current definition using `Expr()`:

```
p = "<" + Expr(p) + "*" + Expr(p) + ">" | "x";
Pat Match( "<<x*<x*x>>*x>", Pat Pos(0) + p + Pat R Pos(0) );
```

Remember, `expr()` is the procrastination function; when the pattern is assigned to the variable `p`, `expr()` delays evaluating its argument (`p`) until later. In the next statement, `patMatch` performs the pattern match operation, and each time it encounters `expr()`, it looks for the current value of the argument. In this example, the value does not change during the match). So, if `p` is defined in terms of itself, how can this possibly work?

`p` consists of two alternatives. The right hand choice is easy: a single letter x. The left side is harder: <p*p> . Each `p` could be a single letter x, since that is one of the choices `p` could match, or it could be <p*p>. The last few example have used `patPos(0) + ... + patRPos(0)` to make sure the pattern matches the entire source text. Sometimes this is what you want, and sometimes you would rather the pattern match a subtext. If you are experimenting with these examples by changing the source text, you probably want to match the entire string to easily tell what was matched. The result from `Pat Match` is 0 or 1.

This example uses “Left” recursion:

```
x =  Expr(x) + "a" | "b"; // + binds tighter than |
```

If the pattern is used in `FULLSCAN` mode, it eventually uses up all memory as it expands. By default, the `patMatch` function does not use `FULLSCAN`, and makes some assumptions that allow the recursion to stop and the match to succeed. The pattern matches either a “b”, or anything the pattern matches followed by an “a”.

```
rc = Pat Match( "baaaaa", x );
```

Patterns and Case

Unlike regular expressions, pattern matching is case insensitive. To force case sensitivity, you can add the named argument MATCHCASE to either `Pat Match()` or `Regex Match()`. For example:

```
string = "abcABC";

result = Regex Match( string, Pat Regex( "[aBc]+" ) );
Show( string, result );
      / *returns string = "abcABC"
   result = {"abcABC"}*/

result = Regex Match( string, Pat Regex( "[cba]+" ), NULL, MATCHCASE );
Show( string, result );
      /* returns string = "abcABC"
   result = {"abc"}*/
```

Troubleshooting Patterns

When a pattern isn't working the way you expect, use `Log()` to send pieces of the match to the JMP log. The following example shows how to debug pattern matching and rewrite a script to account for the white space between words in the phrase "the quick brown fox".

```
// greedy match for one or more characters except for white space
word = Pat Repeat( Pat Not Any( " " ) );

Pat Match(
   "the quick brown fox",
   word >> a +
   word >> b +
   word >> c +
   word >> d
);

Show( a, b, c, d );
   a = "qu";
   b = "i";
   c = "c";
   d = "k";
```

The output is probably not what you want, given the name of the pattern *word*. None of the words in the pattern were matched.

The `Show( a, b, c, d )` expression is a good first step. If it doesn't tell you how JMP got that answer, try inserting `Log()`:

```
word = Pat Repeat( Pat Not Any( " " ) );
```

```
Pat Match(
   "the quick brown fox",
   (word >> a) >> Log( "a" ) +
   (word >> b) >> Log( "b" ) +
   (word >> c) >> Log( "c" ) +
   (word >> d) >> Log( "d" )
);

Show( a, b, c, d );
   0(a) the (the number 0 is the position of the match)

   0(a) th (the "a" is an aribtrary identifier you supplied to the log)

   2(b) e

   ...
   4(a) quick
   4(a) quic
   ...
```

With this extra information, you can now see into the workings of the match, how the word pattern does, in fact, match a word, at least initially. But then it backs up and tries a smaller amount of the word so that the next word pattern can match some non-whitespace characters. However, there's no provision in this pattern to match the white space between words. And that's exactly what it did (find four consecutive words of one or more non-whitespace characters). Maybe the following script is what you need:

```
word = Pat Repeat( Pat Not Any( " " ) );
gap = Pat Repeat( Pat Any( " " ), 1, 999, GREEDY );

Pat Match(
   "the quick brown fox",
   word >> a + gap +
   word >> b + gap +
   word >> c + gap +
   word >> d
);

Show( a, b, c, d );
   a = "the";
   b = "quick";
   c = "brown";
   d = "fox";
```

Chapter 7

Data Structures

Working with Collections of Data

JSL provides these basic data structures that can hold a variety of data in a single variable:

- A list holds a number of other values, including nested lists and expressions.
- A matrix is a row-by-column table of numbers.
- An associative array maps keys to values, which can be almost any other type of data.

Contents

Lists . . . 193
Evaluate Lists . . . 193
Assignments with Lists . . . 195
Perform Operations in Lists . . . 195
Find the Number of Items in a List . . . 195
Subscripts . . . 195
Locate Items in a List . . . 197
List Operators and Functions . . . 198
Iterate through a List . . . 201
Concatenate Lists . . . 202
Insert a Nested List in an Existing List . . . 203
Index into a List with Another List . . . 203
Send Messages to Lists . . . 203
Matrices . . . 204
Construct Matrices . . . 204
Subscripts . . . 206
Inquiry Functions . . . 210
Comparisons, Range Checks, and Logical Operators . . . 210
Numeric Operations . . . 211
Concatenation . . . 214
Transpose . . . 214
Indexing into a Matrix or List with Another Matrix or List . . . 215
Matrices and Data Tables . . . 216
Matrices and Reports . . . 218
Loc Functions . . . 219
Ranking and Sorting . . . 221
Special Matrices . . . 221
Inverse Matrices and Linear Systems . . . 226
Decompositions and Normalizations . . . 230
Build Your Own Matrix Functions . . . 235
Statistical Examples . . . 236
Associative Arrays . . . 240
Create Associative Arrays . . . 240
Work with Associative Arrays . . . 243
Applications for Associative Arrays . . . 246
Associative Arrays in Graph Theory . . . 248
Associative Arrays in Set Operations . . . 251

Lists

Lists are containers to store items, such as the following:

- numbers
- variables
- character strings
- expressions (for example, assignments or function calls)
- matrices
- nested lists

Create a list in one of the following ways:

- use the `List` function
- use `{ }` curly braces

Examples

Use the `List()` function or curly braces to create a list that includes numbers and variables:

```
x = List(1, 2, b);
x = {1, 2, b};
```

A list can contain text strings, nested lists, and function calls:

```
{"Red", "Green", "Blue", {1, "true"}, sqrt( 2 )};
```

You can place a variable into a list and assign it a value at the same time:

```
x = {a = 1, b = 2};
```

Evaluate Lists

When you run a script that contains a list, a copy of the list is returned. The items inside the list are not evaluated.

```
b = 7;
x = {1, 2, b, Sqrt( 3 )};
Show( x );
   x = {1, 2, b, Sqrt(3)};
```

To evaluate items in a list, use the `Eval List()` function.

```
b = 7;
x = {1, 2, b, Sqrt( 3 )};
c = Eval List( x );
   {1, 2, 7, 1.73205080756888}
```

When using a list of variables that reference lists, you will need to use an Eval() function.

Consider the following example, where you have a variable called fullMonth that is a list of 12 items.

```
::fullMonth = {January, February, March, April, May, June, July, August,
  September, October, November, December};
::abbrevMonth = {Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec};
::dow = {Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday};
::levels = {Very Low, Low, Medium Low, Medium, Medium High, High, Very High};
::feel = {Strongly Disagree, Disagree, Neutral, Indifferent, Agree, Strongly
  Agree};
::rating = {Failing, Unacceptable, Very Poor, Poor, Bad, Acceptable, Average,
  Good, Better, Very Good, Excellent, Best};
::mlist = {::fullMonth, ::abbrevMonth, ::dow, ::levels, ::feel, ::rating};
N Items(::fullMonth); // returns 12 because fullMonth has 12 items.
N Items(::mlist[1]); /* returns this message: "N Items() argument must be a
  list" because ::mlist[1] is a variable (fullMonth) and fullMonth is a
  list.*/
```

When you add the Eval() function:

```
N Items( Eval( ::mlist[1] ) );
  12
/* returns 12 because::mlist[1] holds ::fullMonth, and evaluating it
returns ::fullMonth's list.*/
```

In a loop, you need the Eval() function in the nested loop to evaluate the variable's contents. For example:

```
For( g = 1, g <= N Items( ::mlist ), g++,
    For( i = 1, i <= N Items(::mlist[g]), i++,
        Show( ::mlist[g][i] )
    )
);
// returns the message "N Items() argument must be a list."
```

To fix the problem, add the Eval() function:

```
::mlist = {::fullMonth, ::abbrevMonth, ::dow, ::levels, ::feel, ::rating};
For( g = 1, g <= N Items( ::mlist ), g++,
    For( i = 1, i <= N Items( Eval( ::mlist[g] ) ), i++,
        Show( ::mlist[g][i] )
    )
);
```

Assignments with Lists

Create a list to assign values to variables.

Examples

```
{a, b, c} = {1, 2, 3}; // assign 1 to a, 2 to b, and 3 to c
{a, b, c}--; // decrement a, b, and c
{{a}, {b, c}}++; // increment a, b, and c by 1
mylist = {1, log( 2 ), e()^pi(), height[40]}; // store the expressions
```

Perform Operations in Lists

In lists, you can perform operations.

```
a = {{1, 2}, 3, {4, 5}};
b = {{10, 20}, 30, {40, 50}};
c = a + b;
   c = {{11, 22}, 33, {44, 55}}
```

Find the Number of Items in a List

To determine the number of items in a list, use the `N Items()` function.

```
x = {1, 2, y, Sqrt( 3 ), {a, b, 3}};
N = N Items( x );
Show( n );
   n = 5;
```

Subscripts

Subscripts extract specified items from a list. Use a list as a subscript to return multiple items from a list.

Note: JSL starts counting from 1, so the first element in a list is [1], not [0] as in some other languages.

Examples

List a contains four items.

```
a = {"bob", 4, [1, 2, 3], {x, y, z}};
Show( a[1] );
   a[1] = "bob";
Show( a[{1, 3}] );
```

```
   a[{1, 3}] = {"bob", [1, 2, 3]};
a[2] = 5; // assign 5 to the second list item
```

You can also use subscripts to select or change items in a list:

```
Show( a );
   a = {"bob", 5, [1, 2, 3], {x, y, z}};
c = {1, 2, 3};
c[{1, 2}] = {4, 4};
// c[{1, 2}] = 4 produces the same result
Show( c );
   c = {4, 4, 3};
```

When you have assignments or functions in a list, you can use a quoted name for the subscript to extract the value.

```
x={sqrt( 4 ), log( 3 )};
xx= {a = 1, b = 3, c = 5};
x["sqrt"];
   4
xx["b"];
   3
```

The name must be in quotation marks, or else JMP tries to evaluate it and use its value. The following example shows the values of the second item in the list, rather than the value of `a` in the list.

```
a = 2;
Show( xx[a] );
   xx[a] = b = 3;
```

Notes:

- Multiple left-side subscripts (for example, `a[i][j] = value` where `a` contains a list of things that are subscriptable) are allowed in the following circumstances:
  - Each level except the outermost level must be a list. So, in the example above, `a` must be a list but `a[i]` can be anything subscriptable.
  - Each subscript except the last must be a number. So, in the example above, `i` must be a number, but `j` could be a matrix or list of indices.
- Subscripting can be done to any level of nesting, such as the following:

  ```
  a[i][j][k][l][m][n] = 99;
  ```

Note: To get a value in the matrix of an expression column, use double subscripts, as in `:Discrim Data Matrix[row()][i]`.

Locate Items in a List

Use the `Loc()` function or the `Contains()` function to find values in a list:

```
Loc( list, value );
Contains( list, value );
```

`Loc()` and `Contains()` return the positions of the values. `Loc()` returns the results in a matrix, and `Contains()` returns the results as a number.

Notes:

- The `Loc` function returns each occurrence of a repeated value. `Contains()` returns only the first occurrence of a repeated value.
- If the value is not found, the `Loc` function returns an empty matrix and `Contains()` returns a zero.
- To assess whether an item is in a list, use `Loc()` and `Contains()` with `>0`. A returned value of zero means that the item is not in the list. A returned value of 1 means that the item is in the list at least once.

Note: For more information about matrix manipulation and a description of the equivalent `Loc()` command for matrices, see "Matrices".

Examples

```
nameList = {"Katie", "Louise", "Jane", "Jane"};
numList = {2, 4, 6, 8, 8};
```

Search for the value `"Katie"` in the `nameList`:

```
Loc( nameList, "Katie" );
   [1]
Contains( nameList, "Katie" );
   1
```

Search for the value `"Erin"` in the `nameList`:

```
Loc( nameList, "Erin" );
   []
Contains( nameList, "Erin" );
   0
```

Search for the number `8` in the `numList`:

```
Loc( numList, 8 );
   [4, 5]
Contains( numList, 8 );
   4
```

Find out if the number `5` exists in the `numList`:

```
NRow( Loc( numList, 5 ) ) > 0;
   0
Contains( numList, 5 ) > 0;
   0
```

List Operators and Functions

Table 7.1 describes the list operators and their syntax.

Table 7.1 List Operators

Operator and Function		Syntax	Explanation
	As List()	As List(*matrix*)	Returns the matrix as a list. A matrix with multiple columns is returned as a list of lists, one list per row.
= += -= *= /= ++ --	Assign() Add To() SubtractTo() MultiplyTo() DivideTo() Post Increment() Post Decrement()	{*list*} = {*list*} {*list*} += *value* {*list*} -= {*list*} ...	If the target of an assignment operator is a list and the value to be assigned is a list, then it assigns item by item. The ultimate values in the left list must be L-values (in other words, names capable of being assigned values). **Notes:** • If you want to test equality of lists, use ==, not =. • JMP does not have pre-increment or pre-decrement operators. Instead, use the Add To() operator, which is (+=), or the Subtract To() operator, which is (-=).
||=	Concat To()	Concat To(*list1*, *list2*, ...)	Inserts the second and subsequent lists at the end of the first list.
||	Concat()	Concat(*list1*, *list2*, ...);	Returns a copy of the first list with any additional lists inserted at after it.

Table 7.1 List Operators *(Continued)*

Operator and Function	Syntax	Explanation
Eval List()	Eval List(*list*)	Returns a list of the evaluated expressions inside list. See "Evaluate Lists".
Insert Into()	Insert Into(*list*, *x*, <*i*>)	Inserts a new item (*x*) into the *list* at the given position (*i*). If *i* is not given, the item is added to the end of the list. This function does change the original list.
Insert()	*list* = Insert(*list*, *x*, <*i*>)	Returns a copy of the *list* with a new item (*x*) inserted into the *list* at the given position (*i*). If *i* is not given, the item is added to the end of the list. This function does not change the original list.
Is List()	Is List(*arg*)	Returns true (1) if *arg* is a classical list (in other words, one that would result from the construction by List(*items*) or {*items*}) and returns false (0) otherwise. An empty list is still a list, so IsList({ }) returns true. If miss=., then IsList(miss) returns false, not missing.
{ } List	List(*a*, *b*, *c*) {*a*, *b*, *c*}	Constructs a list from a set of items. An item can be any expression, including other lists. Items must be separated by commas. Text should either be enclosed in double quotation marks (" ") or stored in a variable and called as that variable.
N Items	N Items(*list*)	Returns the number of elements in the *list* specified. Can be assigned to a variable.

Table 7.1 List Operators *(Continued)*

Operator and Function	Syntax	Explanation
`Remove From()`	`Remove From(list, <i>, <n>)`	Deletes *n* items from the *list*, starting from the indicated position (*i*). If *n* is omitted, the item at *i* is deleted. If *n* and *i* are omitted, the item at the end is removed. This function does change the original list.
`Remove()`	`Remove(list, <i>, <n>)`	Returns a copy of the *list* with the *n* items deleted, starting from the indicated position (*i*). If *n* is omitted, the item at *i* is deleted. If *n* and *i* are omitted, the item at the end is removed. This function does not change the original list.
`Reverse Into()`	`Reverse Into(list)`	Reverses the order of the items in the *list*. This function does change the original list.
`Reverse()`	`Reverse(list)`	Returns a copy of the *list* with the items in reverse order. This function does not change the original list.
`Shift Into()`	`Shift Into(list, <n>)`	Shifts *n* items from the front of the *list* to the end of the *list*. If *n* is omitted, the first item is moved to the end of the list. This function does change the original list.
`Shift()`	`Shift(list, <n>)`	Returns a copy of the *list* with *n* items shifted from the front of the list to the end of the *list*. If *n* is omitted, the first item is moved to the end of the list. This function does not change the original list.
`Sort Ascending()`	`Sort Ascending(list)`	Returns a copy of the *list* sorted in ascending order. This function does not change the original list.

Table 7.1 List Operators *(Continued)*

Operator and Function		Syntax	Explanation
	`Sort Descending()`	`Sort Descending(`*`list`*`)`	Returns a copy of the *`list`* sorted in descending order. This function does not change the original list.
	`Sort List Into()`	`Sort List Into(`*`list`*`)`	Sorts the *`list`* in ascending order. This function does change the original list.
	`Sort List()`	`Sort List(`*`list`*`)`	Returns a copy of the *`list`* sorted in ascending order. This function does not change the original list.
`[ ]`	`Subscript()`	*`list`*`[`*`i`*`]` *`x`* `=` *`list`*`[`*`i`*`]` *`list`*`[`*`i`*`]` `=` *`value`* *`a`*`[`*`b`*`,` *`c`*`]` `Subscript(`*`a`*`,` *`b`*`,` *`c`*`)`	Subscripts for lists extract the i^{th} item from the *`list`*. Subscripts can in turn be lists or matrices.
	`Substitute()`	`Substitute(`*`list`*`,` *`pattExpr1`*`,` *`replExpr1`*`, ...)`	Returns a copy of a string, list, or expression, replacing instances of each pattern expression with the corresponding replacement expression. See "Substitute and Substitute Into".
	`Substitute Into()`	`Substitute Into(`*`list`*`,` *`pattExpr1`*`,` *`replExpr1`*`, ...)`	Changes a string, list, or expression, replacing instances of each pattern expression with the corresponding replacement expression. **Note:** The *`list`* or expression must be a variable. See "Substitute and Substitute Into".

Iterate through a List

Iterate through a list to do something with each value or look for a particular value. The following script looks at each item in the list. If the item in the list is less than or equal to 10, it is replaced with its square.

```
x = {2, 12, 8, 5, 18, 25};
n = N Items( x );
For( i = 1, i <= n, i++,
```

```
    If( x[i] <= 10,
       x[i] = x[i] ^ 2
    )
);
Show( x );
   x = {4, 12, 64, 25, 18, 25};
```

You can use `Loc()` to locate the items in the new list that are equal to 25:

```
Loc( x, 25 );
   [4, 6] // The fourth and sixth items in the list are equal to 25.
```

Concatenate Lists

Join two or more lists into one list using `Concat()` or the `||` operator. The original lists are not changed.

The following example uses `Concat()` to join lists *a* and *b*:

```
a = {1, 2};
b = {7, 8, 9};
Concat( a, b );
   {1, 2, 7, 8, 9}
```

The following example joins the same lists using the `||` operator:

```
{1, 2} || {7, 8, 9}
   {1, 2, 7, 8, 9}
```

Lists of different types can be concatenated (for example, lists that contain character strings and numbers).

```
d = {"apples", "bananas"};
e = {"oranges", "grapes"};
f = {1, 2, 3};
Concat( d, e, f);
   {"apples", "bananas", "oranges", "grapes", 1, 2, 3}
```

Join two or more lists and replace the first list with the combined list using `Concat to()` or the `||=` operator. The following example performs concatenation in place using `Concat to()`:

```
d = {"apples", "bananas"};
e = {"peaches", "pears"};
Concat to(d,e);
Show( d );
   d = {"apples", "bananas", "peaches", "pears"}
```

The following example joins the same lists using the `||=` operator:

```
d = {"apples", "bananas"};
e = {"peaches", "pears"};
d||=e;
```

```
Show( d );
  d = {"apples", "bananas", "peaches", "pears"}
```

Insert a Nested List in an Existing List

To insert a nested list in an existing list, specify the placement of the list:

```
list2 = {{"a", "b"}, {"c", "d"}};
list2[3] = {"apple", "banana"}; // specify a list as the third item
Show( list2 );
  list2 = {{"a", "b"}, {"c", "d"}, {"apple", "banana"}};
```

If you do not know the number of items in a list, use `N Items()`. In this example, an empty list is inserted at the end of the list.

```
list1 = {"a", "b"};
list1[N Items( list1 ) + 1] = {};
Show( list1 );
  list1 = {"a", "b", {}};
```

Index into a List with Another List

Indexing into a a list with another list is a powerful technique for making scripts fast and small. See “Indexing into a Matrix or List with Another Matrix or List”.

Send Messages to Lists

You can send a single message to a list instead of having to send individual messages. Suppose that you want to create a Bivariate report with a By variable. The script returns a list of the bivariate objects, one for each level of the By variable. Send a single `Fit Line` message to the list to fit a line to all of the Bivariate reports.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
/* return a list of bivariate objects:
{Bivariate[], Bivariate[], Bivariate[], Bivariate[], Bivariate[],
Bivariate[]}; */
biv = dt << Bivariate( X( dt:weight ), Y( dt:height ), By( dt:age ) );

biv << Fit Line; /* send the Fit Line message to
each Bivariate object in the list */
```

Troubleshooting Messages Sent to a List

Any message that is sent to a list is sent to each object in the list. If the message saves a file, it's performing the save for each object in the list to the *same file*. This means that only the last object is actually saved in a file.

Suppose that you have a list of three control charts and you send a message (such as `Save Interactive HTML`) to the list. The message is sent to each member of the list in turn. First, it saves an HTML page with the first control chart; then it saves a second HTML page with the same name that contains the second control chart; and then it saves a third HTML page with the same name that contains only the third control chart. You end up with a page that contains only the third control chart.

The solution is to send the message to a single container higher up the tree that contains all control charts. To do this, rewrite the following line of code:

```
obj << Save Interactive HTML( "Control Charts.html" );
```

`Top Parent` treats the first object as the single container:

```
tp = obj[1] << Top Parent();
tp << Save Interactive HTML( "Control Charts.html" );
```

Matrices

A matrix is a rectangular array of numbers that are arranged in rows and columns. Use matrices to store numbers and perform calculations on those numbers using matrix algebra.

Note the following for this section:

- Matrices are represented with an uppercase bold variable (for example, **A**).
- A matrix with one row or one column is a *vector* (or more specifically, a *row vector* or a *column vector* respectively).
- For clarity, we represent matrices that are vectors with lowercase bold letters (such as **x**).
- A *scalar* is a numeric value that is not in a matrix.

Construct Matrices

Note the following when creating matrices:

- Place matrix literals in square brackets. `[...]`
- Matrix values can contain decimal points, can be positive or negative, and can be in scientific notation.

- Separate items across a column with blank spaces. You can use any number of blank spaces.
- Separate rows with a comma.

For constructing more advanced matrices, see "Special Matrices".

Examples

Create matrix **A** with 3 rows and 2 columns:

```
A = [1 2, 3 4, 5 6];
```

R is a row vector and **C** is a column vector:

```
R = [10 12 14];
C = [11, 13, 15];
```

B is a 1-by-1 matrix, or a matrix with one row and one column:

```
B = [20];
```

E is an empty matrix:

```
E = [];
```

The following examples show two ways of creating empty matrices:

```
J( 0,4 );
  [](0,4)

[]( 5,0 )
  [](5,0)
```

Specifying the number of rows and columns in an empty matrix is optional. JMP creates the matrix as necessary. For example, the following script defines an empty matrix, iterates over each row in the data table, and returns the heights and weights in a matrix.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
data = []( 0, 2 );
For Each Row( data |/= Matrix( {{dt:height, dt:weight}} ) );
Show( data );
  data = [59 95, 61 123, 55 74,...]
```

A script can return an empty matrix. In Big Class.jmp, the following expression looks for rows in which age equals 8, finds none, and returns an empty matrix:

```
a = dt << Get Rows Where( age == 8 );
Show( a );
  a = [](0,1);
```

Construct Matrices from Lists

If you want to convert lists into a matrix, use the `Matrix()` function. A single list is converted into a column vector. Two or more lists are converted into rows.

Create a column vector from a single list:

```
A = Matrix( {1, 2, 3} );
   [1,2,3]
```

Create a matrix from a list of lists. Each list is a row in the matrix.

```
A = Matrix( {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}} );
   [1 2 3,
   4 5 6,
   7 8 9]
```

Construct Matrices from Expressions

To construct matrices from expressions, use `Matrix()`. Elements must be expressions that resolve to numbers.

```
A = Matrix( {4 * 5, 8 ^ 2, Sqrt( 9 )} );
   [20, 64, 3]
```

Subscripts

Use the subscript operator (`[ ]`) to pick out elements or submatrices from matrices. The `Subscript()` function is usually written as a bracket notation after the matrix to be subscripted, with arguments for rows and columns.

Single Element

The expression `A[i,j]` extracts the element in row `i`, column `j`, returning a scalar number. The equivalent functional form is `Subscript(A,i,j)`.

```
P = [1 2 3, 4 5 6, 7 8 9];
P[2, 3]; // returns 6
Subscript( P, 2, 3 ); // returns 6
```

Assign the value that is in the third row and the first column in the matrix **A** (which is 5) to the variable `test`.

```
A = [1 2, 3 4, 5 6];
test = A[3, 1];
Show( test );
   test = 5;
```

Matrix or List Subscripts

To extract a sub-matrix, use matrix or list subscripts. The result is a matrix of the selected rows and columns. The following expressions select the second and third rows with the first and third columns.

```
P = [1 2 3, 4 5 6, 7 8 9];
P[[2 3],[1 3]]; // matrix subscripts
P[{2, 3},{1, 3}]; // list subscripts
```

Both of these methods provide the following output:

```
[4 6,
7 9]
```

Single Subscripts

A single subscript addresses matrices as if all the rows were connected end-to-end in a single row. This makes the double subscript `A[i,j]` the same as the single subscript `A[(i-1)*ncol(A)+j]`.

Examples

```
Q = [2 4 6, 8 10 12, 14 16 18];
Q[8]; // same as Q[3,2]
  16
```

The following examples all return the column vector [10, 14, 18]:

```
Q = [2 4 6, 8 10 12, 14 16 18];
Q[{5, 7, 9}];
Q[[5 7 9]];
ii = [5 7 9];
Q[ii];
ii = {5, 7, 9};
Q[ii];
Subscript( Q, ii );
```

This script returns the values 1 through 9 from the matrix **P** in order:

```
P = [1 2 3, 4 5 6, 7 8 9];
For( i = 1, i <= 3, i++,
  For( j = 1, j <= 3, j++,
    Show( P[i, j] )
  )
);
```

Delete Rows and Columns

Deleting rows and columns is accomplished by assigning an empty matrix to that row or column.

```
A[k, 0] = [];  // deletes the kth row
A[0, k] = [];  // deletes the kth column
```

Select Whole Rows or Columns

A subscript argument of zero selects all rows or columns.

```
P = [1 2 3, 4 5 6, 7 8 9];
```

Select column 2:

```
P[0, 2];
   [2, 5, 8]
```

Select columns 3 and 2:

```
P[0, [3, 2]];
   [3 2, 6 5, 9 8]
```

Select row 3:

```
P[3, 0];
   [7 8 9]
```

Select rows 2 and 3:

```
P[[2, 3], 0];
   [4 5 6, 7 8 9]
```

Select all columns and rows (the whole matrix):

```
P[0, 0];
   [1 2 3, 4 5 6, 7 8 9]
```

Assignment through Subscripts

You can change values in matrices using subscripts. The subscripts can be single indices, matrices or lists of indices, or the zero index representing all rows or columns. The number of selected rows and columns for the insertion must either match the dimension of the inserted argument, or the argument can be inserted repeatedly into the indexed positions.

Examples

Change the value in row 2, column 3 to 99:

```
P = [1 2 3, 4 5 6, 7 8 9];
P[2, 3] = 99;
Show( P );
```

```
P=[1 2 3, 4 5 99, 7 8 9]
```

Change the values in four locations:

```
P = [1 2 3, 4 5 6, 7 8 9];
P[[1 2], [2 3]] = [66 77, 88 99];
Show( P );
   P=[1 66 77, 4 88 99, 7 8 9]
```

Change three values in one column:

```
P = [1 2 3, 4 5 6, 7 8 9];
P[0, 2] = [11, 22, 33];
Show( P );
   P=[1 11 3, 4 22 6, 7 33 9]
```

Change three values in one row:

```
P = [1 2 3, 4 5 6, 7 8 9];
P[3, 0] = [100 102 104];
Show( P );
   P=[1 2 3, 4 5 6, 100 102 104]
```

Change all the values in one row to the same value:

```
P = [1 2 3, 4 5 6, 7 8 9];
P[2, 0] = 99;
Show( P );
   P=[1 2 3, 99 99 99, 7 8 9]
```

Operator Assignment

You can use operator assignments (such as +=) on matrices or subscripts of matrices. For example, the following statement adds 1 to the *i* - *j*th element of the matrix:

```
P = [1 2 3, 4 5 6, 7 8 9];
P[1, 1] += 1;
Show( P );
   P=[2 2 3,
      4 5 6,
      7 8 9];

P[1, 1] += 1;
Show( P );
   P=[3 2 3,
   4 5 6,7 8 9];
```

Ranges of Rows or Columns

If you are working with a range of subscripts, use the `Index()` function or `::` to create matrices of ranges.

```
T1 = 1 :: 3; // create the vector [1 2 3]
T2 = 4 :: 6; // create the vector [4 5 6]
T3 = 7 :: 9; // create the vector [7 8 9]
T = T1 |/ T2 |/ T3; // concatenate the vectors into a single matrix
T[1 :: 3, 2 :: 3]; // refer to rows 1 through 3, columns 2 and 3
   [2 3, 5 6, 8 9]
T[Index( 1, 3 ), Index( 2, 3 )]; // equivalent Index function
   [2 3, 5 6, 8 9]
```

Inquiry Functions

The `NCol()` and `NRow()` functions return the number of columns and rows in a matrix (or data table), respectively:

```
NCol( [1 2 3, 4 5 6] ); // returns 3 (for 3 columns)
NRow( [1 2 3, 4 5 6] ); // returns 2 (for 2 rows)
```

To determine whether a value is a matrix, use the `Is Matrix()` function, which returns a 1 if the argument evaluates to a matrix.

```
A = [20, 64, 3];
B = {20, 64, 3};
Is Matrix( A ); // returns 1 for yes
Is Matrix( B ); // returns 0 for no
```

Comparisons, Range Checks, and Logical Operators

JMP's comparison, range check, and logical operators work with matrices and produce matrices of elementwise Boolean results. You can compare conformable matrices.

```
A < B; // less than
A <= B; // less or equal
A > B; // greater than
A >= B; // greater or equal
A == B; // equal to
A != B; // not equal to
A < B < C; // continued comparison (range check)
A | B; // logical OR
A & B; // logical AND
( A & !B ) | ( !A & B ); // exlusive OR
!( A & B ) & ( A | B ); // exlusive OR
```

You can use the `Any()` or `All()` functions to summarize matrix comparison results. `Any()` returns a 1 if any element is nonzero. `All()` returns a 1 if all elements are nonzero.

```
[2 2] == [1 2]; // returns [0 1], therefore:
All( [2 2] == [1 2] ); // returns 0
```

```
Any( [2 2] == [1 2] ); // returns 1
```

`Min()` or `Max()` return the minimum or maximum element from the matrix or matrices given as arguments.

```
A = [1 2 3, 4 5 6, 7 8 9, 10 11 12];
B = [0 1 2, 2 1 0, 0 1 1, 2 0 0];
Min( A ); // returns 1
Max( A ); // returns 12
Min( A, B ); // returns 0
```

Numeric Operations

You can perform numeric operations (such as subtraction, addition, and multiplication) on matrices. Most statistical methods are expressed in compact matrix notation and can be implemented in JSL.

For example, the following expression uses matrix multiplication and inversion to illustrate least squares regression:

$$b = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

Implement this equation through the following JSL expression:

```
b = Inv( X'*X )*X'*y;
```

Basic Arithmetic

You can perform the following basic arithmetic on matrices:

- addition
- subtraction
- multiplication
- division (multiplying by the inverse)

Note: The standard multiply operator is a matrix multiplier, not an elementwise multiplier.

To perform matrix multiplication, use one of the following methods:

- `*` operator
- `Multiply()` function
- `Matrix Mult()` function

To perform matrix division, use one of the following methods:

- `/` operator

- `Divide()` function

Note the following about matrix multiplication and division:

- Remember that while multiplication or division of scalars is commutative (ab = ba), multiplication or division of matrices is *not* commutative.
- When one of the two elements is a scalar, elementwise multiplication or division is performed.
- To use elementwise multiplication, use `:*` or the `EMult()` function.
- To use elementwise division, use `:/`, or the equivalent `EDiv()` function.

Examples

```
A = [1 2 3, 4 5 6, 7 8 9, 10 11 12];
B = [0 1 2, 2 1 0, 0 1 1, 2 0 0];
C = [1 2 3 4, 4 3 2 1, 0 1 0 1];
D = [0 1 2, 2 1 0, 1 2 0];
```

Matrix addition:

```
R = A + B;
   [1 3 5,
   6 6 6,
   7 9 10,
   12 11 12]
```

Matrix subtraction:

```
R = A - B;
   [1 1 1,
   2 4 6,
   7 7 8,
   8 11 12]
```

Matrix multiplication (inner product of rows of **A** with columns of **C**):

```
R = A * C;
   [9 11 7 9,
   24 29 22 27,
   39 47 37 45,
   54 65 52 63]
```

Matrix division (equivalent to `A*Inverse(D)`):

```
R = A / D;
   [1.5 0.5 0,
   3 2 0,
   4.5 3.5 0,
   6 5 0]
```

Matrix elementwise multiplication:

```
R = A :* B;
```

```
[0 2 6,
8 5 0,
0 8 9,
20 0 0]
```

Matrix scalar multiplication:

```
R = C * 2;
   [2 4 6 8,
   8 6 4 2,
   0 2 0 2]
```

Matrix scalar division:

```
R = C / 2;
   [0.5 1 1.5 2,
   2 1.5 1 0.5,
   0 0.5 0 0.5]
```

Matrix elementwise division (division by zero results in missing values):

```
R = A :/ B;
   [. 2 1.5,
   2 5 .,
   . 8 9,
   5 . .]
```

Numeric (Scalar) Functions on Matrices

Numeric functions work elementwise on matrices. Most of the pure numeric functions can be applied to matrices, resulting in a matrix of results. You can mix conformable matrix and scalar arguments.

Examples of numeric functions include the following:

- `Sqrt()`, `Root()`, `Log()`, `Exp()`, `^ Power()`, `Log10()`
- `Abs()`, `Mod()`, `Floor()`, `Ceiling()`, `Round()`, `Modulo()`
- `Sine,()`, `Cosine()`, `Tangent()`, `ArcSine()`, and other trigonometry functions.
- `Normal Distribution()`, and other probability functions.

Example

```
A = [1 2 3, 4 5 6, 7 8 9, 10 11 12];
B = Sqrt( A ); // elementwise square root
   [1 1.414213562373095 1.732050807568877,
      2 2.23606797749979 2.449489742783178,
      2.645751311064591 2.82842712474619 3,
      3.16227766016838 3.3166247903554 3.464101615137754]
```

Concatenation

The `Concat()` function combines two matrices side by side to form a larger matrix. The number of rows must agree. A double vertical bar (`||`) is the infix operator, equivalent for horizontal concatenation.

```
Identity( 2 ) || J( 2, 3, 4 );
   [1 0 4 4 4, 0 1 4 4 4]

B = [1, 1];
B || Concat( Identity( 2 ), J( 2, 3, 4 ) );
   [1 1 0 4 4 4, 1 0 1 4 4 4]
```

The `VConcat()` function stacks two matrices on top of each other to form a larger matrix. The number of columns must agree. A vertical-bar-slash (`|/`) is the infix operator, equivalent for vertical concatenation.

```
Identity( 2 ) |/ J( 3, 2, 1 );
// or VConcat( Identity( 2 ),J( 3, 2, 1 ) );
   [1 0, 0 1, 1 1, 1 1, 1 1]
```

Both `Concat()` and `VConcat()` support concatenating to empty matrices, scalars, and lists.

```
a=[];
a || [1]; // returns [1]
a || {2}; // returns [2]
a || [3 4 5]; // returns [3 4 5]
```

There are two in place concatenation operators: `||=` and `|/=` . They are equivalent to the `Concat To()` and `V Concat To()` functions, respectively.

- `a || = b` is equivalent to `a = a || b`
- `a |/ = b` is equivalent to `a = a | /b`

Transpose

The `Transpose()` function transposes the rows and columns of a matrix. A back-quote (`` ` ``) is the postfix operator, equivalent to `Transpose()`. In matrix notation, `Transpose()` is expressed as the common prime or superscript-T notation (A′ or A^T).

```
A = [1 2 3, 4 5 6, 7 8 9, 10 11 12];
A`
   [1 4 7 10,
   2 5 8 11,
   3 6 9 12]
Transpose( [1 2, 3 4] );
   [1 3, 2 4]
```

Indexing into a Matrix or List with Another Matrix or List

Indexing into a matrix (or a list) with another matrix (or list) is a powerful technique for making scripts fast and small. JMP 13 and later handles one edge case differently.

Consider the following example of using a matrix as an index into a data column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
tall rows = Loc( :height << Get Values() >= 67 ); // [25, 27, 30, 37, 39, 40]
:height[tall rows]; // [68, 69, 67, 68, 68, 70]
N Rows( :height[tall rows] ); // 6
```

The second line is the key. `:height << Get Values()` returns the values of the height column as a matrix. The comparison `>= 67` returns a matrix of 0s and 1s corresponding to where the expression is false (0) or true (1). The `Loc()` function returns the locations of the 1s within the previous matrix.

In this case, you get [25, 27, 30, 37, 39, 40] as the row numbers where the `height` column is greater than or equal to 67. That's pretty handy by itself, but the real power comes from being able to use that matrix as the index into a column or other matrix as in the third and fourth lines.

The alternative to this technique is to write a `For()` loop to iterate through every row explicitly. JMP still loops through the rows in either case, but the looping is much faster if done internally rather than in JSL. And the JSL is more compact without the explicit loop.

You can do the same thing with a matrix variable instead of a data column reference:

```
m = :height << Get Values();
tall rows = Loc( m >= 67 ); // [25, 27, 30, 37, 39, 40]
m[tall rows]; // [68, 69, 67, 68, 68, 70]
N Rows( m[tall rows] ); // 6
```

Consider the same examples with a different height cutoff (70 instead of 67):

```
tall rows = Loc( :height << Get Values() >= 70 ); // [40]
:height[tall rows]; // [70]
N Rows( :height[tall rows] ); // 1

m = :height << Get Values();
tall rows = Loc( m >= 70 ); // [40]
m[tall rows]; // [70]
N Rows( m[tall rows] ); /* error in JMP 12 and prior versions
otherwise, returns 1 */
```

Notice that `m[tall rows]` returned 70 instead of [70]. In many cases, a 1x1 matrix and a number are treated the same way, but not always. The `N Rows()` function is one example where it makes a difference. The indexing is too aggressively simplifying the 1x1 matrix into a number.

One work-around is to call `Is Matrix()` before using `N Rows()`. An alternative is to concatenate an empty matrix to the result, which creates a matrix whether the source is a number or a matrix:

```
m = m |/ []
```

In JMP 13 and later, the 1x1 matrix is maintained so that the result of the index operation is a matrix whenever the index itself is a matrix. The same principle applies to list indexing. Data table column indexing has always followed this principle.

Matrices and Data Tables

You can move information between a matrix and a JMP data table. You can use matrix algebra to perform calculations on numbers that are stored in JMP data tables, and you can save the results back to JMP data tables.

Move Data into a Matrix from a Data Table

These sections describe how to move data from a data table into a matrix.

Move All Numeric Values

The `Get As Matrix()` function generates a matrix containing all of the numeric values in a data table or column:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
A = dt << Get As Matrix;
   [12 59 95,
   12 61 123,
   12 55 74, ...]
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = Column( "Height" );
A = col << Get As Matrix;
   [59, 61, 55, 66, 52, ...]
```

Move All Numeric Values and Character Columns

The `Get All Columns As Matrix()` function returns the values from all columns of the data table in a matrix, including character columns. Character columns are numbered according to the alphanumeric order, starting at 1.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
A = dt << Get All Columns As Matrix;
   [21 12 1 59 95,
   28 12 1 61 123, ...]
```

Move Only Certain Columns

To get certain columns of a data table, use column list arguments (names or characters).

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
x = dt << Get As Matrix( {"height", "weight"} );
```

or

```
x = dt << Get As Matrix( {height, weight} );
   [59 95, 61 123, 55 74, ...]
```

Currently Selected Rows

To get a matrix of the currently selected row numbers in the data table:

```
dt << Get Selected Rows;
```

Note: If no rows are selected, the output is an empty matrix.

Find Rows Where

To see a matrix of row numbers where the expression is true:

```
dt << Get Rows Where( expression );
```

For example, the following script returns the row numbers where the sex is male (M):

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
A = dt << Get Rows Where( Sex == "M" );
   [6, 7, 8, 12, 13, 14, ...]
```

Move Data into a Data Table from a Matrix

This section describes how to move data from a matrix into a data table.

Move a Column Vector

The `Set Values()` function copies values from a column vector into an existing data table column:

```
col << Set Values( x );
```

`col` is a reference to the data table column, and **x** is a column vector.

For example, the following script creates a new column called test and copies the values of vector **x** into the test column:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "test" );
col = Column( "test" );
x = 1::40;
```

```
col << Set Values( x );
```

Move All Matrix Values

The `Set Matrix()` function copies values from a matrix into an existing data table, making new rows and new columns as needed to store the values of the matrix. The new columns are named Col1, Col2, and so on.

```
dt = New Table( "B" );
dt << Set Matrix([1 2 3 4 5, 6 7 8 9 10]);
```

This script creates a new data table called B containing two rows and five columns.

To create a new data table from a matrix argument, use the `As Table(matrix)` command. The columns are named Col1, Col2, and so on. For example, the following script creates a new data table containing the values of **A**:

```
A = [1 2 3, 4 5 6, 7 8 9, 10 11 12];
dt = As Table( A );
```

Matrices and Reports

You can extract matrices of values from reports. First, you need to locate the items that you want to extract. This information is in the tree structure of the report.

Run the following script to create a table of parameter estimates in a Bivariate report:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" ) ;
biv = dt << Bivariate( X( height ), Y( weight ), Fit Line );
```

Now, open the tree structure to identify which item contains the parameter estimates:

- Right-click a gray disclosure icon and select **Edit > Show Tree Structure**.

  The parameter estimates are contained in `NumberColBox(13)`. Continue with the script:

  ```
  colBox = Report( biv )[Number Col Box( 13 )];
  beta = colBox << Get As Matrix;
    [-127.145248610915, 3.711354893859555]
  ```

Notes:

- When a variable contains a reference to a table box, `Get As Matrix()` creates a matrix **A** that contains the values from all numeric columns in the table:

  ```
  A = tableBox << Get As Matrix;
  ```

- When a variable contains a reference to a numeric column in a report table, `Get As Matrix()` creates a matrix **A** as a column vector of values from that column.

  ```
  A = colBox << Get As Matrix;
  ```

Loc Functions

The Loc(), Loc Nonmissing(), Loc Min(), Loc Max(), and Loc Sorted() functions all return matrices that show positions of certain values in a matrix.

Loc

The Loc() function creates a matrix of positions that locate where **A** is true (nonzero and nonmissing).

```
A = [0 1 . 3 4 0];
B = [2 0 0 2 5 6];
```

The following example returns the indices for the values of **A** that are nonmissing and nonzero.

```
I = Loc( A );
   [2, 4, 5]
```

The following example returns the indices for the values of **A** that are less than the corresponding values of **B**. Note that the two matrices must have the same number of rows and columns.

```
I = Loc( A < B );
   [1, 5, 6]
```

The following script replaces all values less than 4 in **A** with 0.

```
A = [0 1 0 3 4 0];
A[Loc( A < 4 )] = 0;
Show( A );
   A = [0 0 0 0 4 0];
```

The following script returns an empty matrix:

```
Loc( [2 3 4 5 6] > 7 );
   [](0, 1)
```

Loc Nonmissing

The Loc Nonmissing() function returns a vector of row numbers in a matrix that do not contain any missing values. For example,

```
A = [1 2 3, 4 . 6, 7 8 ., 8 7 6];
Loc Nonmissing( A );
   [1, 4]
```

Loc Min and Loc Max

The `Loc Min()` and `Loc Max()` functions return the position of the first occurrence of the minimum and maximum elements of a matrix. Elements of a matrix are numbered consecutively, starting in the first row, first column, and proceeding left to right.

```
A = [1 2 2, 2 4 4, 1 1 1];
B = [6, 12, 9];
Show( Loc Max( A ) );
Show( Loc Min( B ) );
   Loc Max(A) = 5;
   Loc Min(B) = 1;
```

Loc Sorted

The `Loc Sorted()` function is mainly used to identify the interval that a number lies within. The function returns the position of the highest value in **A** that is less than or equal to the value in **B**. The resulting vector contains an item for each element in **B**.

```
A = [10 20 30 40];
B = [35];
Loc Sorted( A, B );
   [3]
A = [10 20 40];
B = [35 5 45 20];
Loc Sorted( A, B );
   [2, 1, 3, 2]
```

Here's an example that defines a list of strings. `Loc Sorted()` identifies the string at the seventh interval and puts it in a list.

```
{"No Fat", "Some Fat", "Low Fat", "Normal Fat", "Medium Fat", "High Fat"}
[Loc Sorted( [0, 1, 5, 10, 20, 25], 7)]
   {"Low Fat"}
```

To convert the returned value to a string, include the subscript [1] at the end of the expression:

```
{"No Fat", "Some Fat", "Low Fat", "Normal Fat", "Medium Fat", "High Fat"}
[Loc Sorted( [0, 1, 5, 10, 20, 25], 7)][1]
   "Low Fat"
```

Notes:

- **A** must be sorted in ascending order.
- The returned values are always 1 or greater. If the element in **B** is smaller than all of the elements in **A**, then the function returns a value of 1. If the element in **B** is greater than all of the elements in **A**, then the function returns *n*, where *n* is the number of elements in **A**.

Ranking and Sorting

The `Rank()` function returns the positions of the numbers in a vector or list, as if the numbers were sorted from lowest to highest.

```
E = [1 -2 3 -4 0 5 1 8 -7];
R = Rank( E );
   [9, 4, 2, 5, 7, 1, 3, 6, 8]
```

If **E** were sorted from lowest to highest, the first number would be -7. The position of -7 in **E** is 9.

The original matrix **E** can then be sorted using the matrix **R** as subscripts to **E**.

```
sortedE = E[R];
   [-7,- 4, -2, 0, 1, 1, 3, 5, 8]
```

The `Ranking Tie()` function returns ranks for the values in a vector or list, with ranks for ties averaged. Similarly, `Ranking()` returns ranks for the values in a vector or list, but the ties are ranked arbitrarily.

```
E = [1 -2 3 -4 0 5 1 8 -7];
Ranking Tie( E );
   [5.5, 3, 7, 2, 4, 8, 5.5, 9, 1]

E = [1 -2 3 -4 0 5 1 8 -7];
Ranking( E );
   [5, 3, 7, 2, 4, 8, 6, 9, 1]
```

The `Sort Ascending()` and `Sort Descending()` functions sort vectors.

```
E = [1 -2 3 -4 0 5 1 8 -7];
Sort Ascending( E );
   [-7 -4 -2 0 1 1 3 5 8]

E = [1 -2 3 -4 0 5 1 8 -7];
Sort Descending( E );
   [8 5 3 1 1 0 -2 -4 -7]
```

If the argument is not a vector or list, an error message is generated.

Special Matrices

Construct an Identity Matrix

The `Identity()` function constructs an identity matrix of the dimension that you specify. An identity matrix is a square matrix of zeros except for a diagonal of ones. The only argument specifies the dimension.

```
Identity( 3 );
```

```
[1 0 0,
0 1 0,
0 0 1]
```

Construct a Matrix with Specific Values

The `J()` function constructs a matrix with the number of rows and columns that you specify as the first two arguments, whose elements are all the third argument, for example:

```
J( 3, 4, 5 );
   [5 5 5 5,
   5 5 5 5,
   5 5 5 5]
J( 3, 4, Random Normal() ); // your results will differ
   [0.407709113182904 1.67359154091978 1.00412665221308 0.240885679837327,
   -0.557848036549455 -0.620833861982722 0.877166783247633 1.50413740148892,
   -2.09920574748608 -0.154797501010655 0.0463943433032137 0.0640418263933316]
```

Create a Diagonal Matrix

The `Diag()` function creates a diagonal matrix from a square matrix (having an equal number of rows and columns) or a vector. A diagonal matrix is a square matrix whose nondiagonal elements are zero.

```
D = [1 -1 1];
Diag( D );
   [1 0 0,
   0 -1 0,
   0 0 1]
Diag([1, 2, 3, 4]);
   [1 0 0 0,
   0 2 0 0,
   0 0 3 0,
   0 0 0 4]

A = [1 2,3 4];
f = [5];
D = Diag( A, f );
   [1 2 0
   3 4 0
   0 0 5]
```

In the third example, at first glance, not all of the nondiagonal elements are zero. Using matrix notation, the matrix can be expressed as follows:

```
[A 0,
0` f]
```

Where `A` and `f` are the matrices from the example, and **`0`** is a column vector of zeros.

Create a Column Vector from Diagonal Elements

The `VecDiag()` function creates a column vector from the diagonal elements of a matrix.

```
v = Vec Diag(
[1 0 0 1, 5 3 7 1, 9 9 8 8, 1 5 4 3]);
   [1, 3, 8, 3]
```

Calculate Diagonal Quadratic Forms

The `Vec Quadratic()` function calculates the hats in regression that go into the standard errors of prediction or the Mahalanobis or T2 statistics for outlier distances. `Vec Quadratic(Sym, X)` is equivalent to calculating `Vec Diag(X*Sym*X`)`.The first argument is a symmetric matrix, usually an inverse covariance matrix. The second argument is a rectangular matrix with the same number of columns as the symmetric matrix argument.

Return the Sum of Diagonal Elements

The `Trace()` function returns the sum of the diagonal elements for a square matrix.

```
D = [0 1 2, 2 1 0, 1 2 0];
Trace( D ); // returns 1
```

Generate a Row Vector of Integers

The `Index()` function generates a row vector of integers from the first argument to the last argument. A double colon `::` is the equivalent infix operator.

```
6::10;
   [6 7 8 9 10]
Index( 1, 5 );
   [1 2 3 4 5]
```

The optional *`increment`* argument changes the default increment of +1.

```
Index( 0.1, 0.4, 0.1 );
   [0.1, 0.2, 0.3, 0.4]
```

The increment can also be negative.

```
Index( 6, 0, -2 );
   [6, 4, 2, 0]
```

The default value of the increment is 1, or -1 if the first argument is higher than the second.

Reshape a Matrix

The `Shape()` function reshapes an existing matrix across rows to be the specified dimensions. The following example changes the 3x4 matrix `a` into a 12x1 matrix:

```
a = [1 1 1, 2 2 2, 3 3 3, 4 4 4];
Shape( a, 12, 1 );
   [1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4]
```

Summarize Columns of a Matrix

There are several functions that return a row vector based on summary values for each column in a matrix.

```
mymatrix = [11 22, 33 44, 55 66];
V Max( mymatrix ); // returns the maximum of each column
   [55 66]
V Min( mymatrix ); // returns the minimum of each column
   [11 22]
V Mean( mymatrix ); // returns the mean of each column
   [33 44]
V Median( mymatrix ); // returns the median of each column
   [33 44]
V Quantile( mymatrix, 0.75 ); // returns the 0.75 quantile of each column
   [55 66]
V Sum( mymatrix ); // returns the sum of each column
   [99 132]
V Std( mymatrix ); // returns the standard deviations of each column
   [22 22]
```

Create Design Matrices

The `Design()` function creates a matrix of design columns for a vector or list. There is one column for each unique value in the vector or list. The design columns have the elements 0 and 1. For example, **x** below has values 1, 2, and 3, then the design matrix has a column for 1s, a column for 2s, and a column for 3s. Each row of the matrix has a 1 in the column representing that row's value. So, the first row (1) has a 1 in the 1s column (the first column) and 0s elsewhere; the second row (2) has a 1 in the 2's column and 0s elsewhere; and so on.

```
x = [1, 2, 3, 2, 1];
Design( x );
   [1 0 0,
   0 1 0,
   0 0 1,
   0 1 0,
   1 0 0]
```

A variation is the `Design Nom()` or `Design F()` function, which removes the last column and subtracts it from the others. Therefore, the elements of `Design Nom()` or `Design F()` matrices are 0, 1, and –1. And the `Design Nom()` or `Design F()` matrix has one less column than the vector or list has unique values. This function makes full-rank versions of design matrices for effects.

```
x = [1, 2, 3, 2, 1];
Design Nom( x );
   [1 0, 0 1, -1 -1, 0 1, 1 0]
```

`Design Nom()` is further demonstrated in the "ANOVA Example".

To facilitate ordinal factor coding, use the `Design Ord()` function. This function produces a full-rank coding with one less column than the number of unique values in the vector or list. The row for the lowest value in the vector or list is all zeros. Each succeeding value adds an additional 1 to the row of the design matrix.

```
x = [1, 2, 3, 4, 5, 6];
Design Ord( x );
   [0 0 0 0 0,
   1 0 0 0 0,
   1 1 0 0 0,
   1 1 1 0 0,
   1 1 1 1 0,
   1 1 1 1 1]
```

`Design()`, `Design Nom()`, and `Design Ord()` support a second argument that specifies the levels to be looked up and their order. This feature allows design matrices to be created one row at a time.

- `Design( values, levels )` creates a design matrix of indicator columns.
- `DesignNom( values, levels )` creates a full-rank design matrix of indicator columns.

Notes:

- The *values* argument can be a single element or a matrix or list of elements.
- The *levels* argument can be a list or matrix of levels to be looked up.
- The result has the same number of rows as there are elements in the *values* argument.
- The result always has the same number of columns as there are items in the *levels* argument. In the case of `Design Nom()` and `Design Ord()`, there is one less column than the number of items in the *levels* argument.
- If a value is not found, the whole row is zero.

Examples

```
Design( 20, [10 20 30] );
   [0 1 0]
Design( 30, [10 20 30] );
```

```
   [0 0 1]
Design Nom( 20, [10 20 30] );
   [0 1]
Design Nom( 30, [10 20 30] );
   [-1 -1]
Design Ord( 20, [10 20 30] );
   [1 0]
Design( [20, 10, 30, 20], [10 20 30] );
   [0 1 0,
   1 0 0,
   0 0 1,
   0 1 0]
Design({"b", "a", "c", "b"}, {"a", "b", "c"});
   [0 1 0,
   1 0 0,
   0 0 1,
   0 1 0]
```

Find the Direct Product

The `Direct Product()` function finds the direct product (or Kronecker product) of two square matrices. The direct product of a *m* x *m* matrix and a *n* x *n* matrix is the *mn* x *mn* matrix whose elements are the products of numbers, one from **A** and one from **B**.

```
G = [1 2, 3 5];
H = [2 3, 5 7];
Direct Product( G, H );
   [2 3 4 6,
   5 7 10 14,
   6 9 10 15,
   15 21 25 35]
```

The `H Direct Product()` function finds the row-by-row direct product of two matrices with the same number of rows.

```
H Direct Product( G, H );
   [2 3 4 6, 15 21 25 35]
```

`HDirect Product()` is useful for constructing the design matrix columns for interactions.

```
XA = Design Nom( A );
XB = Design Nom( B );
XAB = HDirect Product( XA, XB );
X = J( NRow( A ), 1 ) || XA || XB || XAB;
```

Inverse Matrices and Linear Systems

JMP has the following functions for computing inverse matrices: `Inverse()`, `GInverse()`, and `Sweep()`. The `Solve()` function is used for solving linear systems of equations.

Inverse or Inv

The `Inverse()` function returns the matrix inverse for the square, nonsingular matrix argument. `Inverse()` can be abbreviated `Inv`. For a matrix **A**, the matrix product of **A** and `Inverse`(**A**) (often denoted $\mathbf{A}(\mathbf{A}^{-1})$) returns the identity matrix.

```
A = [5 6, 7 8];
AInv = Inv( A );
A*A Inv;
   [1 0,0 1]

A = [1 2, 3 4];
AInv = Inverse( A );
A*AInv;
   [1 1.110223025e-16,0 1]
```

Note: There can be small discrepancies in the results due to floating point limitations, as illustrated in the second example.

GInverse

The (Moore-Penrose) generalized inverse of a matrix **A** is any matrix **G** that satisfies the following conditions:

AGA = **A**

GAG = **G**

(**AG**)′ = **AG**

(**GA**)′ = **GA**

The `GInverse()` function accepts any matrix, including non-square ones, and uses singular-value decomposition to calculate the Moore-Penrose generalized inverse. The generalized inverse can be useful when inverting a matrix that is not full rank. Consider the following system of equations:

$$\begin{aligned} x + 2y + 2z &= 6 \\ 2x + 4y + 4z &= 12 \\ x + y + z &= 1 \end{aligned}$$

Find the solution to this system using the following script:

```
A = [1 2 2, 2 4 4, 1 1 1];
B = [6, 12, 1];
Show( GInverse( A ) * B );
   GInverse(A) * B = [-4, 2.5, 2.5];
```

Solve

The `Solve()` function solves a system of linear equations. `Solve()` finds the vector **x** so that $x = A^{-1}b$ where *A* equals a square, nonsingular matrix and *b* is a vector. The matrix **A** and the vector **b** must have the same number of rows. `Solve(A,b)` is the same as `Inverse(A)*b`.

```
A = [1 -4 2, 3 3 2, 0 4 -1];
b = [1, 2, 1];
x = Solve( A, b );
   [-16.9999999999999, 4.99999999999998, 18.9999999999999]
A*x;
   [1, 2, 0.999999999999997]
```

Note: There can be small discrepancies in the results due to floating point limitations, as illustrated in the example.

Sweep

The `Sweep()` function inverts parts (or *pivots*) of a square matrix. If you sequence through all of the pivots, you are left with the matrix inverse. Normally the matrix must be positive definite (or negative definite) so that the diagonal pivots never go to zero. `Sweep()` does not check whether the matrix is positive definite. If the matrix is *not* positive definite, then it still works, as long as no zero pivot diagonals are encountered. If zero (or near-zero) diagonal pivots are encountered on a full sweep, then the result is a g2 generalized inverse if the zero pivot row and column are zeroed.

About the Sweep Function

Suppose matrix **E** consists of smaller matrix partitions, **A**, **B**, **C**, and **D**:

$$E = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

The syntax for `Sweep()` appears as follows:

```
Sweep( E, [...] );
```

[...] indicates partition A.

This produces the matrix result equivalent to the following:

$$\begin{bmatrix} A^{-1} & A^{-1}B \\ -CA^{-1} & D - CA^{-1}B \end{bmatrix}$$

Notes:

- The submatrix in the **A** position becomes the inverse.
- The submatrix in the **B** position becomes the solution to **A**x = **B**.
- The submatrix in the **C** position becomes the solution to x**A** = **C**.

Use of the Sweep Function

`Sweep()` is sequential and reversible:

- `A = Sweep( A, {i,j} )` is the same as `A = Sweep( Sweep( A, i ), j )`. It is sequential.
- `A = Sweep( Sweep( A, i ), i )` restores **A** to its original values. It is reversible.

If you have a cross-product matrix partitioned as follows:

$$C = \begin{bmatrix} X'X & X'y \\ y'X & y'y \end{bmatrix}$$

Then after sweeping through the indices of **X′X**, the result appears as follows:

$$\begin{bmatrix} (X'X)^{-1} & (X'X)^{-1}X'y \\ -yX(X'X)^{-1} & y'y - y'X(X'X)^{-1}X'y \end{bmatrix}$$

The partitions are recognizable to statisticians as follows:

- the least squares estimates for the model Y = Xb + e in the upper right
- the sum of squared errors in the lower right
- a matrix proportional to the covariance of the estimates in the upper left

The `Sweep` function is useful in computing the partial solutions needed for stepwise regression.

The *index* argument is a vector that lists the rows (or equivalently the columns) on which you want to sweep the matrix. For example, if **E** is a 4x4 matrix, to sweep on all four rows to get $\mathbf{E}^{-1}$ requires these commands:

```
E = [ 5 4 1 1, 4 5 1 1, 1 1 4 2, 1 1 2 4];
Sweep( E, [1, 2, 3, 4] );
   [0.56 -0.44 -0.02 -0.02,
   -0.44 0.56 -0.02 -0.02,
   -0.02 -0.02 0.34 -0.16,
   -0.02 -0.02 -0.16 0.34]
Inverse( E ); // notice that these results are the same
   [0.56 -0.44 -0.02 -0.02,
   -0.44 0.56 -0.02 -0.02,
```

```
   -0.02 -0.02 0.34 -0.16,
   -0.02 -0.02 -0.16 0.34]
```

Note: For a tutorial on `Sweep()`, and its relation to the Gauss-Jordan method of matrix inversion, see Goodnight, J.H. (1979) "A tutorial on the SWEEP operator." *The American Statistician*, August 1979, Vol. 33, No. 3. pp. 149–58.

`Sweep()` is further demonstrated in the "ANOVA Example".

Determinant

The `Det()` function returns the determinant of a square matrix. The determinant of a 2 × 2 matrix is the difference of the diagonal products, as demonstrated below. Determinants for *n* × *n* matrices are defined recursively as a weighted sum of determinants for (*n* - 1) × (*n* - 1) matrices. For a matrix to have an inverse, the determinant must be nonzero.

$$\begin{vmatrix} 1 & 2 \\ 3 & 5 \end{vmatrix} = (1 \cdot 5) - (3 \cdot 2) = -1$$

```
F = [1 2, 3 5];
Det( F );
   -1
```

Decompositions and Normalizations

This section contains functions that calculate eigenvalues and eigenvectors and functions that decompose matrices.

Eigenvalues

The `Eigen()` function performs eigenvalue decomposition of a symmetric matrix. Eigenvalue decompositions are used in many statistical techniques, most notably in principal components and canonical correlation, where the transformation associated with the largest eigenvalues are transformations that maximize variances.

`Eigen()` returns a list of matrices. The first matrix in the returned list is a column vector of eigenvalues; the second matrix contains eigenvectors as the columns.

```
A = [ 5 4 1 1, 4 5 1 1, 1 1 4 2, 1 1 2 4];
Eigen( A );
   {[10, 5, 2, 1][0.632455532033676 - 0.316227766016838 - 2.77555756156289e-16
      -0.707106781186547, 0.632455532033676 - 0.316227766016837 -
      1.66533453693773e-16 0.707106781186547, 0.316227766016838
      0.632455532033676 0.707106781186548 0, 0.316227766016837
      0.632455532033676 - 0.707106781186547 0]}
```

Since the function returns a list of matrices, you might want to assign it to a list of two global variables. That way, the column vector of eigenvalues is assigned to one variable, and the matrix of eigenvectors is assigned to another variable:

```
{evals, evecs} = Eigen( A );
```

For some $n \times n$ matrix **A**, eigenvalue decomposition finds all λ (lambda) and vectors **x**, so that the equation Ax = λx has a nonzero solution **x**. The λ's are called eigenvalues, and the corresponding **x** vectors are called eigenvectors. This is equivalent to solving (A - λI)**x** = 0. You can reconstruct **A** from eigenvalues and eigenvectors by a statement like the following:

```
newA = evecs * Diag( evals ) * evecs`;
  [5.000000000000001 4 1 1,
  4 5 1 1,
  1 1 4 2,
  1 1 2 4]
```

Note the following about eigenvalues and eigenvectors:

- The eigenvector matrices are orthonormal, such that the inverse is the transpose: $E'E = EE' = I$.
- Eigenvectors are uniquely determined only if the eigenvalues are unique.
- Zero eigenvalues correspond to singular matrices.
- Inverses can be obtained by inverting the eigenvalues and reconstituting with the eigenvectors. Moore-Penrose generalized inverses can be formed by inverting the nonzero eigenvalues and reconstituting. (See "GInverse".)

  Note: You must decide whether a very small eigenvalue is effectively zero.

- The eigenvalue decomposition enables you to view any square-matrix multiplication as the following:
  - a rotation (multiplication by an orthonormal matrix)
  - a scaling (multiplication by a diagonal matrix)
  - a reverse rotation (multiplication by the orthonormal inverse, which is the transpose), or in notation:

  ```
  A * x = E` * Diag( M ) * E * x;
  ```

  `E` rotates, `Diag( M )` scales, and `E`` reverse-rotates.

Cholesky Decomposition

The `Cholesky()` function performs Cholesky decomposition. A positive semi-definite matrix **A** is re-expressed as the product of a nonsingular, lower-triangular matrix **L** and its transpose: **L***L' = **A**.

```
E = [ 5 4 1 1, 4 5 1 1, 1 1 4 2, 1 1 2 4];
```

```
L = Cholesky( E );
   [2.23606797749979 0 0 0,
   1.788854381999832 1.341640786499874 0 0,
   0.447213595499958 0.149071198499986 1.9436506316151 0,
   0.447213595499958 0.149071198499986 0.914659120760047 1.71498585142509]
```

To verify the results, enter the following:

```
L*L`;
   [5 4 1 1,
   4 5 1 1,
   1 1 4 2,
   1 1 2 4]
```

About Cholesky Decomposition

`Cholesky()` is useful for reconstituting expressions into a more manageable form. For example, eigenvalues are available only for symmetric matrices in JMP, so the eigenvalues of the product **AB** could not be obtained directly. (Although **A** and **B** can be symmetric, the product is not symmetric.) However, you can usually rephrase the problem in terms of eigenvalues of **L′BL** where **L** is the Cholesky root of **A**, which has the same eigenvalues.

Another use of `Cholesky()` is in reordering matrices within `Trace()` expressions. Expressions such as `Trace(A*B*A`)` might involve huge operations counts if **A** has many rows. However, if **B** is small and can be factored into LL′ by Cholesky, then it can be reformulated to `Trace(A*L*L`*A`)`. The resulting matrix is equal to `Trace(L`A`*AL)`. This expression involves a much smaller number of operations, because it consists of only the sum of squares of the **AL** elements.

Use the function `Chol Update()` to update a Cholesky decomposition. If **L** is the Cholesky root of a $n \times n$ matrix **A**, then after using `cholUpdate(L, C, V)`, **L** will be replaced with the Cholesky root of `A+V*C*V'`. **C** is an $m \times m$ symmetric matrix and **V** is an $n \times m$ matrix.

Examples

Manually update the Cholesky decomposition, as follows:

```
exS = [16 1 0 11 -1 12, 1 11 -1 1 -1 1, 0 -1 12 -1 1 0, 11 1 -1 11 -1 9, -1 -1
  1 -1 9 -1, 12 1 0 9 -1 12];
exAchol = Cholesky( exS ); // conducts the Cholesky decomposition

// add two column vectors to the design matrix
exV = [1 1, 0 0, 0 1, 0 0, 0 0, 0 1];

/* The first column vector is added to one of the rows in the design matrix.
The second column vector is subtracted from one of the rows in the design
  matrix. */
exC = [1 0, 0 -1];
exAnew = exS + exV * exC * exV`;
```

```
// update the Cholesky decomposition manually
exAcholnew = Cholesky( exAnew );
```

Instead of manual updating, use `Chol Update()` to update the Cholesky decomposition, as follows:

```
// update the Cholesky decomposition more efficiently
exAcholnew_test = Chol Update( exAchol, exV, exC );
Show( exAcholnew_test );
   exAcholnew_test =
   [ 4 0 0 0 0 0,
   0.25 3.30718913883074 0 0 0 0, ...]

// results are the same as the manual process
Show( exAcholnew );
   exAcholnew=
   [ 4 0 0 0 0 0,
   0.25 3.30718913883074 0 0 0 0, ...]
```

Singular Value Decomposition

The `SVD()` function finds the singular value decomposition of a matrix. That is, for a matrix **A**, `SVD()` returns a list of three matrices **U**, **M**, and **V**, so that `U*diag(M)*V`=A`.

Notes:

- When **A** is taller than it is wide, **M** is more compact, without extra zero diagonals.
- Singular value decomposition re-expresses **A** in the form USV′, where:
  - **U** and **V** are matrices that contain orthogonal column vectors (perpendicular, statistically independent vectors)
  - **S** is a $n \times n$ diagonal matrix containing the nonnegative square roots of the eigenvalues of **A′A**, the singular values of **A**.
- Singular value decomposition is the basis of correspondence analysis.

Example

```
A = [1 2 1 0, 2 3 0 1, 1 0 1 5, 0 1 5 1];
{U, M, V} = SVD( A );
newA = U * Diag( M ) * V`;
   [1 2 0.999999999999997 -2.99456640040496e-15,
   2 3 -1.17505831453979e-15 1,
   0.999999999999997 -2.16851276518826e-15 0.999999999999999 5,
   2.22586706011274e-15 1 5 0.999999999999997]
```

Orthonormalization

The `Ortho()` function orthogonalizes the columns and then divides the vectors by their magnitudes to normalize them. This function uses the Gram-Schmidt method. The column vectors of orthogonal matrices are unit-length and are mutually perpendicular (their dot products are zero).

```
B = Ortho( [1 -1, 1 0, 0 1] );
   [0.408248290463863 -0.707106781186548, 0.408248290463863 0.707106781186548,
      -0.816496580927726 3.14018491736755e-16]
```

To verify that these vectors are orthogonal, multiply **B** by its transpose, which should yield the identity matrix.

```
C = B` * B;
   [1 -3.119061760824e-16, -3.119061760824e-16 1]
```

By default, vectors are normalized, meaning that they are divided by their magnitudes, which scales them to have length 1. Include the option `Scaled(0)` to turn scaling off:

```
Ortho( [1 -1, 1 0, 0 1], Scaled( 0 ) );
   [0.408248290463863 -0.353553390593274, 0.408248290463863 0.353553390593274,
      -0.816496580927726 1.57009245868377e-16]
```

To create vectors whose elements sum to zero, include the `Centered(1)` option. This option is useful when constructing a matrix of contrasts.

```
result = Ortho( [1 -1, 1 0, 0 1], Centered( 1 ) );
   [0.408248290463863 -0.707106781186548, 0.408248290463863 0.707106781186548,
      -0.816496580927726 3.14018491736755e-16]
```

To verify that the elements of each column sum to zero, premultiply by a vector of ones to sum the columns.

```
J(1, 3) * result;
   [1.11022302462516e-16 2.02996189274239e-16]
```

Orthogonal Polynomials

The `Ortho Poly()` function returns orthogonal polynomials for a vector of indices. The polynomial order is specified as a function argument. Orthogonal polynomials can be useful for fitting polynomial models where some regression coefficients might be highly correlated.

```
Ortho Poly( [1 2 3], 2 );
   [-0.707106781186548 0.408248290463862, 0 -0.816496580927726,
      0.707106781186548 0.408248290463864]
```

The polynomial order must be less than the dimension of the vector. Use the `Scaled(1)` option to produce vectors of unit length, as described in "Orthonormalization".

QR Decomposition

`QR()` factorization is useful for numerically stable matrix work. `QR()` returns a list of two matrices. Here is an example:

```
{Q, R} = QR( [11 22, 33 44] );
```

`Q` and `R` hold the results. For a *m* × *n* matrix, `QR()` creates an orthogonal *m* × *m* matrix `Q` and an upper triangular *m* × *n* matrix `R`, so that `X = Q*R`.

Update Inverse Matrices

To add or drop one or more rows in an inverse of an **M`M** matrix, use the `Inv Update(S, X, w)` function. Updating inverse matrices is helpful in drop-1 influence diagnostics and also in candidate design evaluation.

Notes:

- The first argument, **S**, is the matrix to be updated.
- The second argument, **X**, is the matrix of rows to be added or dropped.
- The third argument, `w`, is either `1` to add or `-1` to delete the row or rows.
- Multiple rows can be added or deleted.

Using the `Inv Update(S, X, w)` function is equivalent to calculating the following:

```
S - w * S * X` * Inv( I + w * X * S * X` ) * X * S
```

Where **I** is an identity matrix and `Inv(A)` is an inverse matrix of **A**.

Build Your Own Matrix Functions

You can store your own operations in macros. See "Maximize and Minimize". Similarly, you can create custom matrix functions. For example, you can make your own matrix operation called `Mag()` to find the magnitude of a vector:

```
mag = Function( {x},
   Sqrt( x` * x )
);
```

Similarly, you could create a function called `Normalize` to divide a vector by its magnitude:

```
normalize = Function( {x},
   x / Sqrt( x` * x )
);
```

Statistical Examples

This section contains statistical examples of using matrices.

Regression Example

Suppose that you want to implement your own regression calculation, rather than use the facilities built into JMP. Because of the compact matrix notation, it might require only a few lines of code:

```
Y = [98, 112.5, 84, 102.5, 102.5, 50.5, 90, 77, 112, 150, 128, 133, 85, 112];
X = [65.3, 69, 56.5, 62.8, 63.5, 51.3, 64.3, 56.3, 66.5, 72, 64.8, 67, 57.5,
  66.5];
X = J( N Row( X ), 1 ) || X; // put in an intercept column of 1s
beta = Inv( X` * X ) * X` * Y; // the least square estimates
resid = Y - X * beta; // the residuals, Y - predicted
sse = resid` * resid; // sum of squared errors
Show( beta, sse );
```

This could be expanded into a script that gets its data from a data table, calculates additional results, and shows the results in a report window:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

// get data into matrices
x = (Column( "Age" ) << Get Values ) || (Column( "Height" ) << Get Values);
x = J(N Row( x ), 1, 1) || x;
y = Column( "Weight" ) << Get Values;

// regression calculations
xpxi = Inv( x` * x );
beta = xpxi * x` * y; // parameter estimates
resid = y - x * beta; // residuals
sse = resid` * resid; // sum of squared errors
dfe = N Row( x ) - N Col( x );  // degrees of freedom
mse = sse / dfe; // mean square error, error variance estimate

// additional calculations on estimates
stdb = Sqrt( Vec Diag( xpxi ) * mse ); // standard errors of estimates
alpha = .05;
qt = Students t Quantile( 1 - alpha / 2, dfe );
betau95 = beta + qt * stdb; // upper 95% confidence limits
betal95 = beta - qt * stdb; // lower 95% confidence limits
tratio = beta :/ stdb; // Student's T ratios
probt = ( 1 - TDistribution( Abs( tratio ), dfe ) ) * 2; // p-values
```

```
// present results
New Window( "Big Class Regression",
   Table Box(
      String Col Box( "Term",{"Intercept", "Age", "Height"} ),
      Number Col Box( "Estimate", beta ),
      Number Col Box( "Std Error", stdb ),
      Number Col Box( "TRatio", tratio ),
      Number Col Box( "Prob>|t|", probt ),
      Number Col Box( "Lower95%", betal95 ),
      Number Col Box( "Upper95%", betau95 ) ) );
```

ANOVA Example

You can implement your own one-way ANOVA. This example presents a problem involving a three-level factor indicating Low, Medium, and High doses and a response measurement. Therefore, this example solves the general linear model, as follows:

$$Y = a + bX + e$$

Where:

- **Y** is a vector of responses
- *a* is the intercept term
- **b** is a vector of coefficients
- **X** is a design matrix for the factor
- *e* is an error term

```
factor = [1, 2, 3, 1, 2, 3, 1, 2, 3];
y = [1, 2, 3, 4, 3, 2, 5, 4, 3];
```

First, build a design matrix for the factor:

```
Design Nom( factor );
   [1 0, 0 1, -1 -1, 1 0, 0 1, -1 -1, 1 0, 0 1, -1 -1]
```

Next, add a column of 1s to the design matrix for the intercept term. You can do this by concatenating `J` and `Design Nom()`:

```
x = J( 9, 1, 1) || Design Nom( factor );
   [1 1 0,
   1 0 1,
   1 -1 -1,
   1 1 0,
   1 0 1,
   1 -1 -1,
   1 1 0,
   1 0 1,
   1 -1 -1]
```

Now, to solve the normal equation, you need to construct a matrix **M** with partitions:

$$\begin{bmatrix} X'X & X'y \\ y'X & y'y \end{bmatrix}$$

You can construct matrix **M** in one step by concatenating the pieces:

```
M = ( x` * x || x` * y )
        |/
  ( y` * x || y` * y );
   [9 0 0 27,
      0 6 3 2,
      0 3 6 1,
      27 2 1 93]
```

Now, sweep **M** over all the columns in **X′X** for the full fit model, and over the first column only for the intercept-only model:

```
FullFit = Sweep( M, [1, 2, 3] ); // full fit model
InterceptOnly = Sweep( M, [1] ); // model with intercept only
```

Recall that some of the standard ANOVA results are calculated by comparing the results of these two models. This example focuses on the full fit model, which produces this swept matrix:

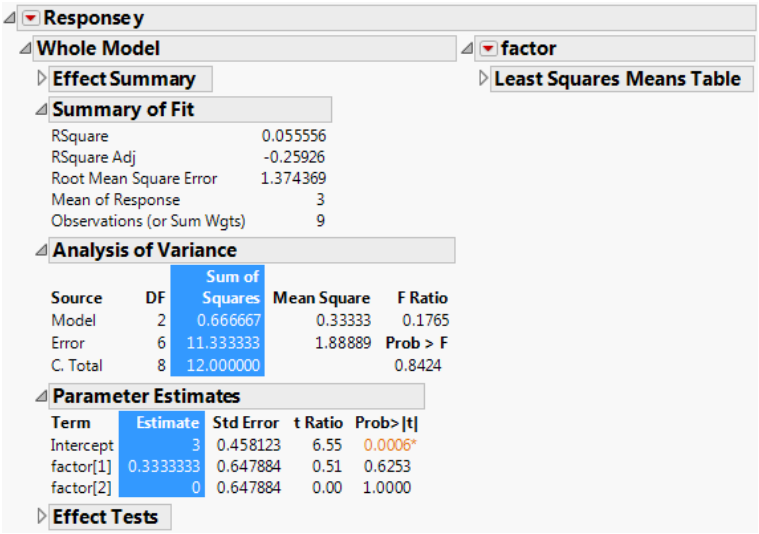
```
[0.11111111111111 0 0 3,
0 0.222222222222222 -0.11111111111111 0.333333333333333,
0 -0.11111111111111 0.222222222222222 0,
-3 -0.33333333333333 0 11.33333333333333]
```

Examine the model coefficients from the upper right partition of the matrix. The lower left partition is the same, except that the signs are reversed: 3, 0.333, 0. The results can be interpreted as follows:

- The coefficient for the intercept term is 3.
- The coefficient for the first level of the factor is 0.333.
- The coefficient for the second level is 0.
- Because of the use of `Design Nom()`, the coefficient for the third level is the difference, –0.333.
- The lower right partition of the matrix holds the sum of squares, 11.333.

You could modify this into a generalized ANOVA script by replacing some of the explicit values in the script with arguments. These results match those from the Fit Model platform.

Figure 7.1 ANOVA Report Within Fit Model



Here is how to construct the report in Figure 7.1:

1. Build a data table (described in "Data Tables"):

```
dt = New Table( "Data" );
dt << New Column( "y", Set Values( [1, 2, 3, 4, 3, 2, 5, 4, 3] ) );
dt << New Column( "factor", "Nominal", Values( [1, 2, 3, 1, 2, 3, 1, 2, 3] )
  );
```

2. Run a model (described in "Launch the Analysis and Save the Script to a Window"):

```
obj = Fit Model(
   Y( :y ),
   Effects( :factor ),
   Personality( "Standard Least Squares" ),
   Run(
      y << {Plot Actual by Predicted( 0 ), Plot Residual by Predicted( 0 ),
      Plot Effect Leverage( 0 )}
   )
);
```

3. Use JSL techniques for navigating displays (described in "Display Box Object References"):

```
ranova = obj << Report;
ranova[Outline Box( 6 )] << Close( 0 );
ranova[Outline Box( 7 )] << Close( 1 );
ranova[Outline Box( 9 )] << Close( 1 );
ranova[Outline Box( 5 ), Number Col Box( 2 )] << Select;
ranova[Outline Box( 6 ), Number Col Box( 1 )] << Select;
```

The result is shown in Figure 7.1.

Associative Arrays

An associative array maps unique keys to values that can be non-unique. An associative array is also called a dictionary, a map, a hash map, or a hash table.

Note the following about keys:

- Keys are placed in quotes.
- The value associated with a key can be a number, date, matrix, list, and so on.
- A key cannot be a floating point number. Only integers are supported.
- You can use matrices as both keys and values, or lists as both keys and values, but you cannot mix the two. In other words, you cannot use a matrix as a key and a list as a value, or the other way around.
- Though associative arrays are not usually ordered, in JMP, keys are returned in alphanumeric order for the purpose of iteration and serialization.

For very large lists, using an associative array is more efficient and faster.

Create Associative Arrays

To create an empty associative array, use the `Associative Array()` function or `[=>]`.

```
cary = Associative Array();
cary = [=> ];
   Associative array( 0 elements ) assigned.
```

Keys and values can be any JSL objects. Items can be added and changed with subscripting:

```
cary = Associative Array();
cary["state"] = "NC";
cary["population"] = 116234;
cary["weather"] = "cloudy";
cary["population"] += 10;
cary["weather"] = "sunny";
cary["high schools"] = {"Cary", "Green Hope", "Panther Creek"};
```

Default Values

A default value determines the value of a key that does not exist in an associative array. If you try to access a key that does not exist in an associative array, an error results. If you define a default value for your associative array, accessing a key that does not exist results in the following:

- adds the key to the associative array
- assigns the default value to the new key
- returns the new key's (default) value instead of an error

If you construct an associative array from a list of strings without assigning values to the keys, then the keys are assigned values of 1. The default value for the associative array is set to 0.

To set the default value:

```
cary = Associative Array();
cary << Set Default Value( "Cary, NC" );
```

To determine whether there is a default value set for an associative array, use the `<<Get Default Value` message.

```
cary << Get Default Value;
   "Cary, NC"
```

If there is no default value, `Empty()` is returned.

Besides the `Set Default Value` message, a default value can be set in the literal constructor using `=>value` without a key.

```
counts = ["a" => 10,
"b" => 3,
=> 0]; // default value of 0
counts["c"] += 1;
Show( counts );
   counts = ["a" => 10, "b" => 3, "c" => 1, => 0];
```

In the first line, the default value is set to 0. In the second line, the key `"c"` does not exist in `counts`. In the output, the key `"c"` is created with the default value of 0 and then incremented by 1.

Note: If a key's value is the default value, then the key is dropped because any key will return the default value.

Associative Array Constructors

Create an empty associative array:

```
map = [=>];
map = Associative Array();
```

Create an empty associative array with a default value:

```
map = [=>0];
map = Associative Array( 0 );
```

Create an associative array with specific values:

```
map = ["yes" => 0, "no" => 1];
```

Create an associative array with specific values with a default value:

```
map = ["yes" => 0, "no" => 1, => 2];
```

Create an associative array with specific values that contains an associative array:

```
map = ["yes" => 0, "no" => 1, "alt" => ["a" => "1", "b" => "2"]];
```

Create an associative array from a list that contains two lists of a key-value pair:

```
map = Associative Array( {{"yes", 0}, {"no", 1}} );
```

Create an associative array from a list that contains two lists of a key-value pair with a default value:

```
map = Associative Array( {{"yes", 0}, {"no", 1}}, 2 );
```

Create an associative array from a list of keys and a list of values:

```
map = Associative Array( {"yes", "no"}, {0, 1} );
```

Create an associative array from a list of keys and a list of values with a default value:

```
map = Associative Array( {"yes", "no"}, {0, 1}, 2 );
```

Create an associative array from two column references. The first column contains the keys and the second contains the values.

```
map = Associative Array( :name, :height );
```

Create an associative array from two column references with a default value:

```
map = Associative Array(:name, :height, .);
```

Create an associative array from a single list of keys or a single column reference of keys with a default value of 0:

```
set = Associative Array( {"yes", "no"} );
set = Associative Array( :name );
```

Work with Associative Arrays

Find the Number of Keys

To determine the number of keys that an associative array contains, use the `N Items()` function.

```
cary = Associative Array();
cary["state"] = "NC";
cary["population"] = 116234;
cary["weather"] = "cloudy";
cary["population"] += 10;
cary["weather"] = "sunny";
cary["high schools"] = {"Cary", "Green Hope", "Panther Creek"};
N Items( cary );
   4
```

Add and Delete Keys and Values

To add or delete key-value pairs from an associative array, use the following functions:

- `Insert()`
- `Insert Into()`
- `Remove()`
- `Remove From()`

Notes:

- `Insert()` and `Remove()` return a named copy of the given associative array with the key-value pair added or removed.
- `Insert Into()` and `Remove From()` add or remove the key-value pairs directly from the given associative array.
- `Insert()` and `Insert Into()` take three arguments: the associative array, a key, and a value.
- `Remove()` and `Remove From()` take two arguments: the associative array and a key.
- If you insert a key with no value provided, the key is assigned a value of 1.

Examples

The following examples illustrate `Insert()` and `Insert Into()`:

```
cary = Associative Array();
cary["state"] = "NC";
cary["population"] = 116234;
cary["weather"] = "cloudy";
cary["population"] += 10;
```

```
cary["weather"] = "sunny";
cary["high schools"] = {"Cary", "Green Hope", "Panther Creek"};
newcary = Insert( cary, "time zone", "Eastern" );
Show( cary, newcary );
   cary = ["high schools" => {"Cary", "Green Hope", "Panther Creek"},
      "population" => 116244, "state" => "NC", "weather" => "sunny"];
   newcary = ["high schools" => {"Cary", "Green Hope", "Panther Creek"},
      "population" => 116244, "state" => "NC", "time zone" => "Eastern",
      "weather" => "sunny"];

cary = Associative Array();
cary["state"] = "NC";
cary["population"] = 116234;
cary["weather"] = "cloudy";
cary["population"] += 10;
cary["weather"] = "sunny";
cary["high schools"] = {"Cary", "Green Hope", "Panther Creek"};
Insert Into(cary, "county", "Wake");
Show( cary );
   cary = ["county" => "Wake", "high schools" => {"Cary", "Green Hope", "Panther
      Creek"}, "population" => 116244, "state" => "NC", "weather" => "sunny"];
```

Note that `cary <<Insert` is a message sent to an associative array that does the same thing as the function `Insert Into()`. For example, these two statements are equivalent:

```
cary << Insert( "county", "Wake" );
Insert Into( cary, "county", "Wake" );
```

The following examples illustrate `Remove()` and `Remove From()`:

```
cary = Associative Array();
cary["state"] = "NC";
cary["population"] = 116234;
cary["weather"] = "cloudy";
cary["population"] += 10;
cary["weather"] = "sunny";
cary["high schools"] = {"Cary", "Green Hope", "Panther Creek"};
newcary = Remove( cary, "high schools" );
Show( cary, newcary );
   cary = ["high schools" => {"Cary", "Green Hope", "Panther Creek"},
      "population" => 116244, "state" => "NC", "weather" => "sunny"];
   newcary = ["population" => 116244, "state" => "NC", "weather" => "sunny"];
Remove From( cary, "weather" );
Show( cary );
   cary = ["high schools" => {"Cary", "Green Hope", "Panther Creek"},
      "population" => 116244, "state" => "NC", "weather" => "sunny"];
   newcary = ["population" => 116244, "state" => "NC", "weather" => "sunny"];
   cary = ["high schools" => {"Cary", "Green Hope", "Panther Creek"},
      "population" => 116244, "state" => "NC"];
```

Note that `aa << Remove` is a message sent to an associative array that does the same thing as the function `Remove From()`. For example, these two statements are equivalent:

```
cary << Remove( "weather" );
Remove From( cary, "weather" );
```

Find Keys or Values in an Associative Array

To determine whether a certain key is contained within an associative array, use the `Contains()` function.

```
cary = Associative Array();
cary["state"] = "NC";
cary["population"] = 116234;
cary["weather"] = "cloudy";
cary["population"] += 10;
cary["weather"] = "sunny";
cary["high schools"] = {"Cary", "Green Hope", "Panther Creek"};
Contains(cary, "high schools");
  1
Contains(cary, "lakes");
  0
```

To obtain a list of all keys contained in an associative array, use the `<< Get Keys` message.

```
cary << Get Keys;
  {"high schools", "population", "state", "weather"}
```

To obtain a list of all values contained in an associative array, use the `<<Get Values` message.

```
cary << Get Values;
  {{"Cary", "Green Hope", "Panther Creek"}, 116244, "NC", "sunny"}
```

If you want to see only the values for certain keys, you can specify them as arguments. The keys must be given as a list.

```
cary << Get Values({"state", "population"});
  {"NC", 116244}
```

To see a value for a single key, use the `<<Get Value` message. Specify only one key and do not place it in a list.

```
cary << Get Value("weather");
  "sunny"
```

To obtain a list of all key-value pairs in an associative array, use the `<<Get Contents` message.

```
cary << Get Contents;
  {{"high schools", {"Cary", "Green Hope", "Panther Creek"}},
  {"population", 116244},
  {"state", "NC"},
  {"weather", "sunny"}}
```

Note: Using the `<<Get Contents` message, the returned list does not include the default value. Keys are listed alphabetically.

Iterate through an Associative Array

To iterate through as associative array, use the `<<First` and `<<Next` messages. `<<First` returns the first key in the associative array. `<<Next(key)` returns the key that follows the *key* that is given as the argument.

The following example removes all key-value pairs from the associative array `cary`, leaving an empty associative array:

```
currentkey = cary << First;
total = N Items( cary );
For( i = 1, i <= total, i++,
   nextkey = cary << Next( currentkey );
   Remove From( cary, currentkey );
   currentkey = nextkey;
);
Show( cary );
   cary = [=>];
```

Applications for Associative Arrays

You can use associative arrays to quickly and efficiently perform other tasks.

Get the Unique Values from a Data Table Column

Note: The strategy described here is not useful for columns with floating point numbers. Use Summarize instead. See "Store Summary Statistics in Global Variables".

A key can exist only once in an associative array, so putting a column's values into one automatically results in the unique values. For example, the Big Class.jmp sample data table contains 40 rows. To see how many unique values are in the column height, run this script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
unique heights = Associative Array( dt:height );
nitems( unique heights );
   17
```

There are only 17 unique values for height. You can use those unique values by getting the keys:

```
unique heights << Get Keys;
   {51, 52, 55, 56, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70}
```

Note: This is possible because you can use any JMP data type as keys in an associative array, not only strings.

Using an associative array to discover unique values in a column is efficient and fast. The following script takes some time to create a data table with 100,000 rows. Finding the 39 unique values takes very little time.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
nms = dt:name << Get Values;
dtbig = New Table( "Really Big Class",
   New Column( "name",
      Character,
      Set Values( nms[J( 100000, 1, Random Integer( N Items( nms ) ) )] )
   )
);
Wait( 0 );
t1 = Tick Seconds();
Write(
   "\!N# names from Really Big Class = ",
   N Items( Associative Array( dtbig:name ) ),
   ", elapsed time=",
   Tick Seconds() - t1
);
   # names from Really Big Class = 39, elapsed time=0.116666666639503
```

Sort a Column's Values in Lexicographic Order

Because keys are ordered lexicographically, putting the values into an associative array also sorts them. For example, the `<<Get Keys` message returns the keys (unique values of the names column) in ascending order:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
unique names = Associative Array( dt:name );
unique names << Get Keys;
   {"ALFRED", "ALICE", "AMY", "BARBARA", "CAROL", "CHRIS", "CLAY", "DANNY",
      "DAVID", "EDWARD", "ELIZABETH", "FREDERICK", "HENRY", "JACLYN", "JAMES",
      "JANE", "JEFFREY", "JOE", "JOHN", "JUDY", "KATIE", "KIRK", "LAWRENCE",
      "LESLIE", "LEWIS", "LILLIE", "LINDA", "LOUISE", "MARION", "MARK",
      "MARTHA", "MARY", "MICHAEL", "PATTY", "PHILLIP", "ROBERT", "SUSAN",
      "TIM", "WILLIAM"}
```

Compare Columns in Two Different Data Tables

Using associative arrays, determining which values in one column are not in another column (or determining which values are in both columns) is fast. For example, given two data tables with information about countries, which countries are in both data tables?

Place the columns of each data table that contain country names into associative arrays:

```
dt1 = Open( "$SAMPLE_DATA/BirthDeathYear.jmp" );
dt2 = Open( "$SAMPLE_DATA/World Demographics.jmp" );
aa1 = Associative Array( dt1:Country );
aa2 = Associative Array( dt2:Territory );
```

Use `N Items()` to see how many countries appear in each data table:

```
N Items(aa1);
   23
N Items(aa2);
   239
```

Use the `<<Intersect` message to find the common values:

```
aa1 = Associative Array( dt1:Country );
aa1 << Intersect( aa2 );
```

Look at the results:

```
Show(N Items(aa1), aa1 << Get Keys);
   N Items(aa1) = 21;
   aa1 << get keys = {"Australia", "Austria", "Belgium", "France", "Greece",
      "Ireland", "Israel", "Italy", "Japan", "Mauritius", "Netherlands", "New
      Zealand", "Norway", "Panama", "Poland", "Portugal", "Romania",
      "Switzerland", "Tunisia", "United Kingdom", "United States"};
```

This example uses a set operation called intersection. For more examples of using set operations with associative arrays to compare values, see "Associative Arrays in Set Operations".

Associative Arrays in Graph Theory

You can use associative arrays for graph theory data structures, such as the following directed graph example:

```
g = Associative Array();
g[1] = Associative Array({1, 2, 4});
g[2] = Associative Array({1, 3});
g[3] = Associative Array({4, 5});
g[4] = Associative Array({4, 5});
g[5] = Associative Array({1, 2});
```

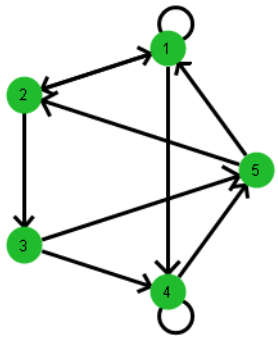
This is a two-level associative array. The associative array `g` contains five associative arrays (`1`, `2`, `3`, `4`, and `5`). In the containing array `g`, both the keys (1-5) and the values (the arrays that define the map) are important. In the inner associative arrays, the values do not matter. Only the keys are important.

The associative array represents the graph shown in Figure 7.2:

- node 1 is incident on nodes 1, 2, and 4
- node 2 is incident on nodes 1 and 3

- node 3 is incident on nodes 4 and 5
- node 4 is incident on nodes 4 and 5
- node 5 is incident on nodes 1 and 2

Figure 7.2 Example of a Directed Graph



The following depth-first search function can be used to traverse this graph, or any other directed graph represented as an associative array:

```
dfs = Function( {ref, node, visited},
    {chnode, tmp},
    Write( "\!NNode: ", node, ", ", ref[node] << Get Keys );
    visited[node] = 1;
    tmp = ref[node];
    chnode = tmp << first;
    While( !Is Missing( chnode ),
        If( !visited[chnode],
            visited = Recurse( ref, chnode, visited )
        );
        chnode = tmp << Next( chnode );
    );
    visited;
);
```

Notes:

- The first argument is the associative array that contains the map structure.
- The second argument is the node that you want to use as the starting point.
- The third argument is a vector that the function uses to keep track of nodes visited.

To see how the preceding example works, add the following expression to the end of the script:

```
dfs( g, 2, J( N Items( g << Get Keys ), 1, 0 ) );
```

Here is the output:

```
Node 2: {1, 3}
Node 1: {1, 2, 4}
Node 4: {4, 5}
Node 5: {1, 2}
Node 3: {4, 5}
[1, 1, 1, 1, 1]
```

The first five output lines show that starting from node 2, you can reach all the other nodes in the order in which they are listed. Each node also lists the nodes it is incident on (the keys). The value for each key is 1. The final line is a matrix that shows that you can reach each node from 2. If there were nodes that could not be reached from node 2, their values in the matrix would be 0.

Here is how to read the traversal of the nodes:

1. Start at node 2 and go to node 1.
2. From node 1, go to node 4.
3. From node 4, go to node 5.
4. From node 5, go back to node 2, and then to node 3.

Map Script

Here is the script that produced the map shown in Figure 7.2.

```
New Window( "Directed Graph",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( -1.5, 1.5 ),
      Y Scale( -1.5, 1.5 ),
      Local( {n = N Items( g ), k = 2 * Pi() / n, r, i, pt, from, to,
         edge, v, d},
         Fill Color( "green" );
         Pen Size( 3 );
         r = 1 / (n + 2);
         For( i = 1, i <= n, i++,
            pt = Eval List( {Cos( k * i ), Sin( k * i )} );
            edges = g[i];
            For( edge = edges << First, !Is Empty( edge ),
               edge = edges << Next( edge ),
               to = Eval List( {Cos( k * edge ), Sin( k * edge )} );
               If( i == edge,
                  Circle( Eval List( 1.2 * pt ), 0.9 * r ), // else
                  v = pt - to;
                  d = Sqrt( Sum( v * v ) );
                  {from, to} = Eval List(
                     {pt * (d - r) / d + to * r / d, pt * r / d + to *
```

```
                        (d - r) / d}
                    );
                    Arrow( from, to );
                );
            );
            Circle( pt, r, "fill" );
            Text( Center Justified, pt - {0, 0.05}, Char( i ) );
        );
    )
  )
);
```

Associative Arrays in Set Operations

You can also use associative arrays to perform set operations. The following examples show how to take a union of two sets, a difference of two sets, and an intersection of two sets.

First, create three sets and view them in the log:

```
set_y = Associative Array( {"chair", "person", "relay", "snake", "tripod"} );
set_z = Associative Array( {"person", "snake"} );
set_w = Associative Array( {"apple", "orange"} );

// write the sets to the log
Write(
  "\!NExample:\!N\!tset_y = ",
  set_y << Get Keys,
  "\!N\!tset_z = ",
  set_z << Get Keys,
  "\!N\!tset_w = ",
  set_w << Get Keys
);
  Example:
    set_y = {"chair", "person", "relay", "snake", "tripod"}
    set_z = {"person", "snake"}
    set_w = {"apple", "orange"}
```

Union Operation

To find the union of two sets, insert one set into the other:

```
set_z << Insert( set_w );
Write( "\!N\!NUnion operation (set_w, set_z):,\!N\!tset_z = ", set_z << Get
  Keys );
  Union operation (set_w, set_z):,
    set_z = {"apple", "orange", "person", "snake"}
```

Difference Operation

To find the difference of two sets, remove one set from the other:

```
set_y << Remove( set_z );
Write( "\!N\!NDifference operation (set_z from set_y):\!N\!tset_y = ", set_y
   << Get Keys );
   Difference operation (set_z from set_y):
      set_y = {"chair", "relay", "tripod"}
```

Intersect Operation

To find the intersection of two sets, use the `aa << Intersect` message.

```
set_w << Intersect( set_z );
Write( "\!N\!NIntersect operation (set_w, set_z):\!N\!tset_w = ", set_w << Get
   Keys );
   Intersect operation (set_w, set_z):
      set_w = {"apple", "orange"}
```

Example of Using Set Operations

Given a list of names, which of them are not contained in Big Class.jmp? You can find the answer by taking the difference of the two sets of names.

1. Get the list of names and open the data table:

   ```
   names list = {"ROBERT", "JEFF", "CHRIS", "HARRY"};
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   ```

2. Put the list of names into an associative array:

   ```
   names = Associative Array( names list );
   ```

3. Perform a difference operation by removing the column values from your list:

   ```
   names << Remove( Associative Array( dt:name ) );
   ```

4. Look at the result:

   ```
   Write( "\!NWhich of {ROBERT, JEFF, CHRIS, HARRY}, is not in Big Class = ",
      names << Get Keys );
      Which of {ROBERT, JEFF, CHRIS, HARRY}, is not in Big Class = {"HARRY", "JEFF"}
   ```

Chapter 8

Programming Methods

Complex Scripting Techniques and Additional Functions

This chapter includes advanced techniques, such as throwing and catching exceptions, encrypting scripts, and using complex expressions.

Contents

Lists and Expressions 255
Expressions 255
Macros 267
Manipulating Lists 267
Manipulating Expressions 270
Advanced Scoping and Namespaces 274
Names Default To Here 274
Scoped Names 278
Namespaces 282
Referencing Namespaces and Scopes 286
Resolving Named Variable References 289
Best Practices for Advanced Scripting 290
Advanced Programming Concepts 291
Throw and Catch Exceptions 291
Functions 293
Recursion 296
Includes 296
Load and Save Text Files 297
Run External Programs 297
Classes 299
File and Directory Operations 302
Select a Directory or File 302
Get a List of Filenames 303
Connect to a Network Directory 304
Scripting By Groups 305
Encrypt and Decrypt Scripts 306
Encryption and Global Variables 308
Encrypting Scripts in Data Tables 308
Working with Expressions in an Encrypted Script 309
Additional Numeric Functions 310
Derivatives 310
Algebraic Manipulations 312
Maximize and Minimize 314
Identifying Differences Between Strings, Lines, or Sequences 315
Scheduling Actions 317
Functions that Communicate with Users 318
Writing to the Log 319
Send Information to the User 320

Lists and Expressions

Expressions

This section delves into JSL expressions and introduces several JSL functions that can be used to manipulate JSL expressions. These functions provide JSL programmers with a rich, powerful, meta programming toolkit. One of the benefits of such a toolkit is that it provides JSL programmers with a powerful macro system capability that is similar in functionality to macro systems available in other modern programming languages.

In order to understand JSL expressions, it is useful to keep in mind that everything that happens in JSL is a result of a function call. Function calls come in three varieties:

prefix The function name comes before its arguments (for example, `Foo( m, n )`). Most function calls in JSL are of this form.

infix The function name is between its arguments (for example, `m + n`). This form is common for mathematical operators.

special Functions like `[ ]` (the subscript operator) fall into this category. Since there are only a handful of special functions, they are not discussed further in this section.

Although function calls come in three forms, all function calls can be written in prefix form. It turns out that JMP actually uses the prefix form internally and so, in general, it is very useful to think of JSL expressions in this way.

To illustrate this idea, begin by examining the syntactic structure of JSL expressions a bit more closely. All JSL expressions must conform to a very simple syntactic structure. Each expression consists of a *head* expression and a sequence of zero or more argument expressions. Consider the following JSL expressions, written first in infix form and then in prefix form, to see how this syntactic structure applies:

```
x = 100.1
m & n
1 <= z <= 20
```

Using the prefix form, these expressions are written equivalently:

```
Assign( x, 100.1 )
And( m, n )
Less or Equal( 1, z, 20 )
```

The head of the first example is `Assign` and its arguments are `x` and 100.1. For the second example, the head is `And` and its arguments are `m` and `n`, whereas for the third example, the head is `Less or Equal` and its arguments are 1, `z`, and 20. For these examples, the "=", "&", and "<=" operators are just human friendly ways of representing `Assign`, `And`, and `Less or Equal`.

Examine a few additional simple examples.

```
Assign( 100.1, x )
And( )
Less or Equal( z )
```

These examples also conform to the required syntactic structure, but they differ from the previous examples in that they all produce an error message when they are evaluated. This point leads to an important distinction that you should make when thinking about JSL expressions. That is, you should separate the idea of the syntactic structure of a JSL expression from what happens when the expression is evaluated.

Consider the following examples that present both the human friendly form and the prefix form in each case:

```
x = 100.1 + Log( z )                    // human friendly form
Assign( x, Add( 100.1, Log( z ) ) )     // prefix form
```

This example illustrates that the arguments of a head expression can also be expressions.

```
// human friendly form
x = 0.8;
x ||= Random Uniform( 0.2, 0.6 );

// prefix form
Glue(
   Assign( x, 0.8 ),
   Concat To( x, Random Uniform( 0.2, 0.6 ) )
)
```

This example would typically be referred to as a JSL script, but it is just another case of a JSL expression. Here, the *head* expression is `Glue`, which has two arguments: the expressions `Assign( x, 0.8 )` and `Concat To( x, Random Uniform( 0.2, 0.6 ) )`. This example illustrates two important points:

1. The ";" symbol is another instance of a human friendly operator. In this case, it is for the `Glue` function.
2. There is no distinction between what are typically referred to as a JSL script and a JSL expression.

The remainder of this section covers how to assign an expression to a variable and then how to manipulate, or execute, that expression when needed. For information about how JMP resolves names when evaluating JSL expressions, see the section "Rules for Name Resolution".

Quoting and Unquoting Expressions

The operators to control when expressions are evaluated are `Expr` and `Eval`, which you should think of as the procrastination and eager operators. `Expr` just copies its argument as an expression, rather than evaluating it. `Eval` does the opposite: it evaluates its argument, and then takes that result and evaluates it again.

`Expr` and `Eval` can be thought of as quoting and unquoting operators, telling JMP when you mean the expression itself, and when you mean the result of evaluating the expression.

The following examples all assume these two assignments:

```
x = 1; y = 20;
```

If you assign the expression `x+y` to *a*, quoting it as an expression with `Expr`, then whenever *a* is evaluated, it evaluates the expression using the current values of *x* and *y* and returns the result. (Exceptions are the utilities `Show`, `Write`, and `Print`, which do not evaluate expressions for pure name arguments.)

```
x = 1; y = 20;
a = Expr( x + y );
a;
   21
```

If you want the expression that is stored in the name, rather than the result of evaluating the expression, use the `NameExpr` function. See "Retrieve a Stored Expression Rather than its Result".

```
x = 1; y = 20;
Show( Name Expr( a ) );
   NameExpr(a) = x + y
```

If you assign an extra level of expression-quoting, then when *a* is evaluated, it unpacks one layer and the result is the expression `x+y`.

```
x = 1; y = 20;
a = Expr( Expr( x + y ) );
Show( a );
   a = Expr(x + y)
```

If you want the value of the expression, then use `Eval` to unpack all layers:

```
x = 1; y = 20;
Show( Eval( a ) );
   Eval(a) = 21
```

You can do this to any level, for example:

```
x = 1; y = 20;
a = Expr( Expr( Expr( Expr( x + y ) ) ) );
b = a;
   Expr( Expr( x + y ) )
c = Eval( a );
   Expr( x + y )
d = Eval( Eval( a ) );
   x+y
e = Eval( Eval( Eval( a ) ) );
   21
```

For more information about expression handling, see Morgan (2010).

Quote an expression as a string

The `JSL Quote()` function returns the contents of an expression as a quoted string. Comments and white space in the string are preserved.

The following expression is an example:

```
x = JSL Quote( /* Begin quote. */
For (i = 1, i <= 5, i++,
// Print the value of i.
Print( i );
);
// End expression.
);
Show( x );
```

In the output, the contents of the `JSL Quote()` function are enclosed in quotes.

```
x = " /* Begin quote. */
For (i = 1, i <= 5, i++,
   // Print the value of i.
Print(i);
);
   // End expression.
";
```

Quoting Expressions

A common use of the `Expr()` function is to store an expression in a JSL variable. This could be considered as a macro. Consider the following script:

```
dist = Expr( Distribution( Column( height ) ) );
```

To execute the above expression, just use the following:

```
dist;
```

You can use expressions in a loop to execute it:

```
For( i = 0, i < 3, i = i + 1, dist );
```

This loop results in three Distribution platform reports for height.

Use `Eval()` to evaluate an expression explicitly:

```
Eval( dist );
```

Note, however, that in column formulas, `Eval()` only works if it is outermost in the formula. So, for example,

```
Formula( Log( Eval( Column Name( i ) ) ) );
```

would generate an error. Instead, use:

```
Formula( Eval( Substitute( Expr( Log( xxx ) ), Expr( xxx ), Column Name( i ) )
  ) );
```

As another example,

```
Formula( Eval( Column Name( i ) ) + 10 );
```

generates an error, since `Eval()` is actually under the `Add` function. Instead, use:

```
Formula(Eval(Substitute(Expr(xxx+10), Expr(xxx), column name(i))))
```

Retrieve a Stored Expression Rather than its Result

What if you wanted the symbolic value of a variable (such as the expression `Distribution(Column(height))` stored in `dist` above), rather than the evaluation of it (the actual launched platform)? The `Name Expr` function does this. `Name Expr` retrieves its argument as an expression without evaluating it.

`Expr` returns its argument exactly, whereas `Name Expr` looks up the expression stored in its argument. `Name Expr` "unpacks" just one layer to get the expression, but does not keep unpacking to get the result.

For example, you would need to use this if you had an expression stored in a name and you wanted to edit the expression:

```
popVar = Expr( Summation( i = 1, N Row(), (y[i] - Col Mean( y )) ^ 2 / N Row()
  ) );
  Summation( i = 1, N Row(), (y[i] - Col Mean( y )) ^ 2 / N Row() )

unbiasedPopVar = Substitute( Name Expr( popVar ), Expr( Wild()/N Row() ),
  Expr( (y[i] - Col Mean( y )) ^ 2 / ( N Row() - 1 ) ) );
  Summation( i = 1, N Row(), (y[i] - Col Mean( y )) ^ 2 / (N Row() - 1) )
```

Compare `x`, `Expr(x)`, `NameExpr(x)`, and `Eval(x)` after submitting this script:

```
a = 1; b = 2; c = 3;
x = Expr( a + b + c );
```

Table 8.1 Compare Eval, Name Expr, and Expr

Command and result	Explanation
`x;` *6*	Evaluates *x* to the expression `a+b+c`, and then evaluates the expression, returning the result, 6 (unpacks *all* layers).
`Eval( x );` *6*	Equivalent to simply calling *x*. Evaluates *x* to the expression `a+b+c`, and then evaluates the expression, returning the result, 6 (unpacks *all* layers).
`NameExpr( x );` *a+b+c*	Returns the expression that was stored in *x*, which is `a+b+c` (unpacks the *outside* layer).
`Expr( x );` *x*	Returns the expression `x` (packs *one* layer).

JSL also supports functions to access and traverse expressions, all of them either a name or a literal expression as an argument. In the following, *expressionArg* is either a single name or a compound expression to be taken literally.

`NArg(`*expressionArg*`)` finds the number of arguments in *expressionArg*.

The *expressionArg* can be a name holding an expression, an expression evaluated to an expression, or a literal expression quoted by `Expr()`.

`NArg(`*name*`)` obtains the expression held in `name` (it is not evaluated) and returns the number of arguments

`NArg(`*expression*`)` evaluates `expression` and returns the number of arguments

`NArg(Expr(`*expression*`))` returns the number of arguments to literal expression.

For example, if `aExpr = {a+b,c,d,e+f+g};`

- `NArg(aExpr)` results in 4.
- `NArg(Arg(aExpr,4))` results in 3.
- `NArg(Expr({1,2,3,4}))` results in 4.

`Head(`*expressionArg*`)` returns the head of the expression without any arguments. If the expression is an infix, prefix, or postfix special character operator, then it is returned as the functional equivalent.

The *expressionArg* can be a name holding an expression, an expression evaluated to an expression, or a literal expression quoted by `Expr()`.

For example, if `aExpr = expr(a+b);`

- `r = Head(aExpr)` results in `Add()`.
- `r = Head (Expr(sqrt(r)))` results in `Sqrt()`.
- `r = Head({1,2,3})` results in `{}` .

`Arg(`*`expressionArg`*`,`*`indexArg`*`)` extracts the specified argument of the symbolic expression, resulting in an expression.

For example, `Arg(`*`expressionArg`*`,i)` extracts the i^{th} argument of *`expressionArg`*.

The *`expressionArg`* can be a name holding an expression, an expression evaluated to an expression, or a literal expression quoted by `Expr()`.

- `Arg(`*`name`*`,i)` obtains the expression held in *`name`* (it is not evaluated) and finds the `i`th argument.
- `Arg(`*`expression`*`,i)` evaluates *`expression`* and finds the `i`th argument.
- `Arg(Expr(`*`expression`*`),i)` finds the i^{th} argument of *`expression`*.

As another example, if `aExpr = Expr(12+13*sqrt(14-15));`

- `Arg(aExpr,1)` yields `12`.
- `Arg(aExpr,2)` yields `13*sqrt(14-15)`.
- `Arg(Expr(12+13*sqrt(14-15)),2)` yields `13*sqrt(14-15)`.

To extract an argument of an argument inside an expression, you can nest Arg commands:

- `Arg(Arg(aExpr,2),1)` yields the first argument within the second argument of `aExpr`, or 13.
- `Arg(Arg(aExpr,2),2)` yields `Sqrt(14-15)`.
- `Arg(Arg(Arg(aExpr,2),2),1)` yields `14-15`.
- `Arg(Arg(Arg(aExpr,2),2),3)` yields `Empty()`

Here is a description of how the last example line unwraps itself:

1. The inner `Arg` statement is evaluated.

   ```
   Arg(aExpr,2)
     13 * Sqrt( 14 - 15 )
   ```

2. Then the next one is evaluated.

   ```
   Arg(Arg(aExpr,2),2)
   // this is equivalent to Arg(Expr (13 * Sqrt( 14 - 15 ) ), 2)
     Sqrt( 14 - 15 )
   ```

3. Finally, the outer Arg is evaluated.

   ```
   Arg(Arg(Arg(aExpr,2),2),3)
   // this is equivalent to Arg (Expr (Sqrt( 14 - 15 ) ), 3)
   ```

```
Empty()
```

There is only one element to the `Sqrt` expression, so a request for the third argument yields `Empty()`. To access the two arguments inside the `Sqrt` expression, try this:

```
Arg(Arg(Arg(Arg(aExpr,2),2),1),2);
  15
```

`HeadName(`*`expressionArg`*`)` returns the name of the head of the expression as a string. If the expression is an infix, prefix, postfix, or other special character operator, then it is returned as the functional equivalent.

The *`expressionArg`* can be a name holding an expression, an expression evaluated to an expression, or a literal expression quoted by `Expr()`.

For example, if `aExpr = expr(a+b);`

- `r = HeadName (aExpr)` results in `"Add"`.
- `r = HeadName (Expr(sqrt(r)))` results in `"Sqrt"`.
- `r = HeadName ({1,2,3})` results in `"List"`.

In previous versions of JMP, other versions of `Arg`, `Narg`, `Head`, and `HeadName` were implemented, called `ArgExpr`, `NArgExpr`, `HeadExpr`, and `HeadNameExpr`, respectively. These did the same thing, but did not evaluate their argument. These forms are now deprecated and will not be documented in future versions.

Making Substitutions

`Eval Insert` is for the situation where you want to make substitutions, by evaluating expressions inside a character string.

With `Eval Insert`, you specify characters that delimit the start and end of an expression, and everything in between is evaluated and expanded.

There are two functions, one to return the result, the other to do it in-place.

```
resultString = EvalInsert( string with embedded
  expressions,startDelimiter,endDelimiter )
EvalInsertInto( string l-value with embedded
  expressions,startDelimiter,endDelimiter )
```

The delimiter is optional. The default start delimiter is `"^"`. The default end delimiter is the start delimiter.

```
xstring = "def";
r = Eval Insert( "abc^xstring^ghi" );  // returns "abcdefghi";

r = "abc^xstring^ghi"; // in-place evaluation
Eval Insert Into( r ); // r now has "abcdefghi";
```

```
// with specified delimiter
r = Eval Insert( "abc%xstring%ghi","%" );  // returns "abcdefghi";

// with different start and end delimiters
r = Eval Insert( "abc[xstring]ghi","[","]" );  // returns "abcdefghi";
```

When a numeric value contains locale-specific formatting, include the <<Use Locale(1) option. The following example substitutes a comma for the decimal point based on the computer's locale setting.

```
Eval Insert( "^1.2^", <<Use Locale( 1 ) );
  1,2
```

Evaluate expressions inside lists

`Eval List` evaluates expressions inside a list and returns a list with the results:

```
x = { 1 + 2, 3 + 4 };
y = Eval List( x );   // returns y is {3,7}
```

`Eval List` is useful for loading the list of user choices returned by `Column Dialog` or `New Window` with the `Modal` argument.

Evaluate expressions inside expressions

`Eval Expr()` evaluates only the inner expressions and returns an expression that contains the results of the evaluation. By comparison, `Eval` evaluates the inner expressions, takes the results, and then evaluates it again.

Suppose that a data table contains a column named X3. Here is an example of using `Eval Expr()` to evaluate the inner expression first:

```
x = Expr( Distribution( Column( Expr("X"||Char( i ) ) ) ) );
i = 3;
y = Eval Expr( x );   // returns Distribution( Column( "X3" ) )
```

To evaluate further, you need to either call the result in a subsequent step, or else put `Eval()` around the `Eval Expr()`. The following examples create a Distribution report.

```
// two-step method
x = Expr( Distribution( Column( Expr( "X" || Char( i ) ) ) ) );
i = 3;
y = Eval Expr( x );
y;

// one-step method
x = Expr( Distribution( Column( Expr( "X" || Char( i ) ) ) ) );
i = 3;
Eval( Eval Expr( x ) );
```

See Table 8.3 to learn what would happen if you tried to use Eval directly on *x* without first doing Eval Expr.

Parsing Strings into Expressions, and Vice Versa

Parsing is the syntactic scanning of character strings into language expressions. Suppose that you have read in a valid JSL expression into a character string, and now want to evaluate it. The Parse function returns the expression. To evaluate it, use the Eval function.

```
x = Parse( "a=1" ) ;  // x now has the expression a=1
Eval( Parse( "a=1" ) ); // a now has the value 1
```

To go in the reverse, use the Char function, which converts an expression into a character string. Usually the argument to a Char function is an Expr function (or a NameExpr of a variable), since Char evaluates its argument before deparsing it.

```
y = Char( Expr( a = 1 ) ); // results in y having the character value "a=1"
z = Char( 42 ); // returns "42"
```

The Char function allows arguments for field width and decimal places if the argument is a number. The default is 18 for width and 99 for decimal (Best format).

```
Char( 42, 5, 2 ); // returns the character value "42.00"
```

To preserve locale-specific formatting in the numeric value, include the <<Use Locale(1) option as shown in the following example:

```
Char( 42, 5, 2, <<Use Locale(1) ); // returns the character value "42,00" in
  the French locale
```

The reverse of Char is not quite as simple. To convert a character string into an expression, you use Parse, but to convert a character string into a number value, you use Num.

```
Parse( y );
Num( z );
```

Table 8.2 Functions to Store or Evaluate Expressions

Function	Syntax	Explanation
Char	Char(Expr(*expression*)) Char(*name*)	Converts an *expression* into a character string. The expression must be quoted with Expr; otherwise its evaluation is converted to a string.
	string = char(*number*, *width*, *decimal*)	Converts a *number* into its character representation. *Width* and *decimal* are optional arguments to specify formatting; the default is 18 for width and 99 for decimal.

Table 8.2 Functions to Store or Evaluate Expressions *(Continued)*

Function	Syntax	Explanation
Eval	Eval(*x*)	Evaluates *x*, and then evaluates the result of *x* (unquoting).
Eval Expr	Eval Expr(*x*)	Returns an expression with all the expressions inside *x* evaluated.
Eval List	Eval List(*list*)	Returns a list of the evaluated expressions inside *list*.
Expr	Expr(*x*)	Returns the argument unevaluated (expression-quoting).
NameExpr	NameExpr(*x*)	Returns the unevaluated expression of *x* rather than the evaluation of *x*. NameExpr is like Expr except that if *x* is a name, NameExpr returns the unevaluated expression stored in the name rather than the unevaluated name *x*.
Num	Num("*string*")	Converts a character string into a number.
Parse	Parse("*string*")	Converts a character string into a JSL expression.

Summary

Table 8.3 compares various ways that you can use the evaluation-control functions with *x*. Assume that a data table contains a column named X3, and *x* and *i* have been assigned:

```
x = Expr( Distribution(Column( Expr("X"||Char( i ) ) ) ) );
i = 3;
```

Table 8.3 Functions for Controlling Evaluation

Commands and results	Explanation

x; // or
Eval(x);
Not Found in access or evaluation of 'distribution' , Bad Argument({"X" || Char(i)})
	Eval(x) and simply calling x are equivalent.
	Evaluates the expression distribution(column(expr("X" || Char(i)))). This results in errors. The column name is recognized as "X"||Char(i) because it is packed by the Expr() function.

Expr(x); *x*	Returns the expression x (packs *an additional* layer).

Table 8.3 Functions for Controlling Evaluation *(Continued)*

Commands and results	Explanation
`Name Expr(x);` *`Distribution(Column(Expr("X" || Char(i))))`*	Returns the expression stored in *x* exactly as is: `Distribution(Column(Expr("X" || Char(i))))`.
`y=Eval Expr(x);` *`Distribution(Column("X3"))`*	Evaluates the inner expression but leaves the outer expression unevaluated, so that *y* is `Distribution(Column("X3"))`.

`y; //or`
`Eval(Eval Expr(x));`
`Distribution[]`
	`Eval(eval expr(x))` and simply calling `y` are equivalent.
	Evaluates `Distribution(Column("X3"))` to launch the platform.

`z = Char(nameexpr(x));` *`"Distribution(Column(Expr (\!"X\!" || Char(i))))"`*	Quotes the entire expression as a text string, adding `\!"` escape characters as needed. Note that `Char(x)` would first attempt to evaluate *x*, producing an error and ultimately returning a quoted missing value: `"."`
`Parse(z);` *`Distribution(Column(Expr("X" || Char(i))))`*	Unquotes the text string and returns an expression.

`a = Parse(Char( NameExpr(x)));`
`Eval(EvalExpr(a));`
`Distribution[]`
	Evaluation control taken to its logical extreme.
	Note that you must break this into at least two steps as shown. Combining it into one giant step produces different results because the `Eval Expr` layer causes the `Parse` layer to be copied literally, not executed.
	`Eval(`
	`EvalExpr(`
	`Parse(`
	`Char(`
	`NameExpr(x)))));`
	`Distribution(Column(Expr("X" || Char(i))))`

Macros

Stored expressions can serve as a macro feature. You can store a generalized action as an expression in a JSL variable, and then call that variable wherever you need that action to be performed. This example has four macros as the arguments to `If`:

```
lastStdzdThickness=expr(
  (thickness[nrow()]-col mean(thickness)) / col std dev(thickness));
continue=expr(...<script to read in more data>...);
log=expr(print("In control at "||char( long date(today()))));
break=expr(...<script to shut down process>...); limitvalue=1;

if(lastStdzdThickness<limitvalue,log;continue,break);
```

Storing the expression with `Expr` delays its evaluation until the variable is evaluated. Any variables, data points, or expressions included in that expression are evaluated on the fly when the expression is evaluated. See "Expressions" for detailed rules for storing expressions and later evaluating them.

Manipulating Lists

The following functions manipulate lists. They can also be used to manipulate expressions, as shown in the next section, "Manipulating Expressions". A summary of commands with explanations is in Table 8.4.

Most of the functions have two variants, one that produces a new value, and one that works in-place directly on its arguments. Here are some examples:

```
// In-place example
A = {2, 3, 4, 2, 1, 2, 1};
A = Remove( A, 3 ); // delete the third item in the list A, storing result in A
Remove From( A, 3 ); // delete the third item in the list A, in place
Show( A ); // A = {2, 3, 2, 1, 2, 1};

// New value example
B = {2, 3, 4, 2, 1, 2, 1};
onetwo = Insert( {1}, 2 );  // {1, 2}
Insert Into( B, {1, 2}, 4 ); // put 1, 2 before the current 4th item
Show( B ); // B = {2, 3, 4, 1, 2, 2, 1, 2, 1};
```

Note: If the position is omitted in the `Insert Into` command, items are placed at the end of the list.

```
a = Shift( {1,2,3,4}, 1 ); // store the list {2,3,4,1} in a
Shift Into( a, -1 );
Show( a ); // a = {1, 2, 3, 4};
```

```
b = Reverse( a ); // b is now {4,3,2,1}
Reverse Into( a ); // a is now {4,3,2,1}
Show( a ); // a = {4, 3, 2, 1};

s = Sort List( {1,4,2,5, -7.2, Pi(), -11, cat, apple, cake} );
Show( s ); // s = {-11, -7.2, 1, 2, 4, 5, apple, cake, cat, Pi()};

c = {5, pie, 2, Pi(), -2};
Sort List Into( c );
Show( c ); // c = {-2, 2, 5, Pi(), pie};
```

In-place functions

In-place functions are those that operate on lists or expressions directly. They have `From` or `Into` in their names (for example, `Remove From` and `Insert Into`). They do not return a result; you have to show the list to see the result. The first argument for an in-place function must be an L-value. An *L-value* is an entity such as a variable whose value can be set.

```
myList = {a, b, c, d};
Insert Into( myList, 2, 3 );
Show( myList );
   myList = {a, b, 2, c, d}
```

These examples show how to use `Insert Into` and `Remove From` with nested lists:

```
a = {{1, 2, 3}, {"A", "B", "C"}};
Show( a );
   a = {{1, 2, 3}, {"A", "B", "C"}}

Insert Into( a[1], 99, 1 );
Show( a );
   a = {{99, 1, 2, 3}, {"A", "B", "C"}}

Remove From( a[1], 1 );
Show( a );
   a = {{1, 2, 3}, {"A", "B", "C"}}
```

Not in-place functions

For the *not-in-place functions,* you must either state the list directly or else quote a name that evaluates to a list. Such functions do *not* have `From` or `Into` in their names. They *return* manipulated lists or expressions without changing the original list or expression given in the first argument.

```
myNewList = Insert( {a, b, c, d}, 2, 3 );
   {a, b, 2, c, d}

oldList = {a, b, c, d};
newList = Insert( oldList, 2, 3 );
```

```
{a, b, 2, c, d}
```

Substituting

`Substitute()` and `Substitute Into()` merit further discussion. Both functions find all matches to a pattern in a list (or expression) and replace them with another expression. Each pattern must be a name. The arguments are evaluated before they are applied, so most of the time you must quote them with an `Expr()` function.

```
Substitute( {a,b,c}, Expr( a ), 23); // returns {23,b,c}
Substitute( Expr( Sine( x ) ), Expr( x ), Expr( y ) ); // returns Sine(y)
```

To delay evaluating an argument, use `NameExpr` instead of `Expr`:

```
a = {quick, brown, fox, jumped, over, lazy, dogs};
b = Substitute( a, Expr( dogs ), Expr( cat ) );
canine = Expr( dogs );
equine = Expr( horse );
c = Substitute( a, Name Expr( canine ), Name Expr( equine ) );
Show( a, b, c );
   a = {quick,brown,fox,jumped,over,lazy,dogs}
   b = {quick,brown,fox,jumped,over,lazy,cat}
   c = {quick,brown,fox,jumped,over,lazy,horse}
```

`Substitute Into` does the same work, in place:

```
Substitute Into( a, Expr( dogs ), Expr( horse ) );
```

You can list multiple pattern and replacement arguments to do more than one replacement in a single step:

```
d = Substitute( a,
   Name Expr( quick ), Name Expr( fast ),
   Name Expr( brown ), Name Expr( black ),
   Name Expr( fox ), Name Expr( wolf )
);
   {fast,black,wolf,jumped,over,lazy,dogs}
```

Note that substitutions are done repeatedly over multiple instances of the expression pattern. For example:

```
Substitute( Expr( a + a ), Expr( a ), Expr( aaa ) );
```

results in:

```
aaa + aaa
```

Manipulating Expressions

The functions for manipulating lists can also operate on most expressions. Be sure to quote the expression with `Expr()`. For example:

```
Remove( Expr( A + B + C + D ), 2 ); // results in the expression A + C + D
b = Substitute( Expr( Log( 2 ) ^ 2 / 2 ), 2, 3 ); /* returns the expression
  Log( 3 ) ^ 3 / 3*/
```

As with lists, remember that the first argument for in-place functions must be an L-value. An L-value is an entity such as a variable whose value can be set. In-place functions are those that operate on lists or expressions directly. They have `From` or `Into` in their names (for example, `Remove From` and `Insert Into`). They do not return a result; you have to show the expression to see the result.

```
polynomial = Expr( a * x ^ 2 + b * x + c );
Insert Into( polynomial, Expr( d * x ^ 3 ), 1 );
Show( polynomial );
  polynomial = d * x ^ 3 + a * x ^ 2 + b * x + c
```

For the not-in-place functions, you must either state the expression directly or else quote a name that evaluates to an expression using `NameExpr`. Such functions do *not* have `From` or `Into` in their names. They *return* manipulated lists or expressions without changing the original list or expression given in the first argument.

```
cubic = Insert( Expr( a * x ^ 2 + b * x + c ), Expr( d * x ^ 3 ), 1 );
  d * x ^ 3 + a * x ^ 2 + b * x + c

quadratic = Expr( a * x ^ 2 + b * x + c );
cubic = Insert( Name Expr( quadratic ), Expr( d * x ^ 3 ), 1 );
  d * x ^ 3 + a * x ^ 2 + b * x + c
```

Substituting

Substituting is extremely powerful; please review the earlier discussion "Substituting". Here are a few notes regarding substituting for expressions.

`Substitute(`*`pattern`*`,`*`name`*`,`*`replacement`*`)` substitutes for names in expressions

`NameExpr()` looks through the name but copies instead of evaluates:

```
a = Expr(
  Distribution( Column( x ), Normal Quantile Plot )
);
Show( Name Expr( a ) );
  Name Expr(a) = Distribution(Column(x), Normal Quantile Plot);
```

`Substitute()` evaluates all its arguments, so they must be quoted correctly:

```
b = Substitute( Name Expr( a ), Expr( x ), Expr( :weight ) );
```

```
Show( Name Expr( b ) );
  Name Expr(b) = Distribution(Column(:weight), Normal Quantile Plot);
```

SubstituteInto() needs an L-value, so the first argument is not quoted:

```
Substitute Into( a, Expr( x ), Expr( :weight ) );
Show( Name Expr( a ) );
  Name Expr(a) = Distribution(Column(:weight), Normal Quantile Plot);
```

Substitute() is useful for changing parts of expressions, such as in the following example that tests the Is functions:

```
data = {1, {1, 2, 3}, [1 2 3], "abc", x, x( y )};
ops = {is number, is list, is matrix, is string, is name, is expr};
m = J( N Items( data ), N Items( ops ), 0 );
test = Expr(
  m[r, c] = _op( data[r] )
);
For( r = 1, r <= N Items( data ), r++,
  For( c = 1, c <= N Items( ops ), c++,
    Eval( Substitute( Name Expr( test ), Expr( _op ), ops[c] ) )
  )
);
Show( m );
  m =
  [1 0 0 0 0 0,
  0 1 0 0 0 1,
  0 0 1 0 0 0,
  0 0 0 1 0 0,
  0 0 0 0 1 1,
  0 0 0 0 0 1];
```

You can use SubstituteInto() to have JMP solve quadratic equations. The following example solves $4x^2 - 9 = 0$:

```
/* FIND THE ROOTS FOR THE EQUATION:
       a*x^2 + b*x + c = 0
The quadratic formula is x=(-b + - sqrt(b^2 - 4ac))/2a.
Use a list to store both the + and - results of the +- operation */
x = {Expr(
  (-b + Sqrt( b ^ 2 - 4 * a * c )) / (2 * a)
), Expr(
  (-b - Sqrt( b ^ 2 - 4 * a * c )) / (2 * a)
)};
Substitute Into( x, Expr( a ), 4, Expr( b ), 0, Expr( c ), -9 );

// plug in the coefficients
Show( x ); // see the result of substitution
Show( Eval Expr( x ) ); // see the solution
  x = {Expr(((-0) + Sqrt(0 ^ 2 - 4 * 4 * -9)) / (2 * 4)), Expr(((-0) - Sqrt(0
    ^ 2 - 4 * 4 * -9)) / (2 * 4))};
```

```
Eval Expr(x) = {1.5, -1.5};
```

The functions for manipulating lists and expressions are discussed in the previous section, "Manipulating Lists", and summarized in Table 8.4.

Table 8.4 Functions for Manipulating Lists or Expressions

Function	Syntax	Explanation

`Remove`	`x = Remove(list|expr)`
	`x = Remove(list|expr, position)`
	`x = Remove(list|expr, {positions})`
	`x = Remove(list|expr, position, n)`
		Copies the *list* or *expression*, deleting the item(s) at the indicated *position*. If *position* is omitted, items are deleted from the end. *Position* can be a list of positions. An extra argument, *n*, deletes *n* items instead of just 1.

`Remove From`	`Remove From(list|expr, position)`
	`Remove From(list|expr)`
	`Remove From(list|expr, position, n)`
		Remove items in place. The function returns the removed item(s), but you do not have to assign them to anything. The first argument must be an L-value.

`Insert`	`x = Insert(list|expr, item, position)` `x = Insert(list|expr, item)`	Inserts a new item into the *list* or *expression* at the given position. If position is not given, it is inserted at the end.
`Insert Into`	`Insert Into(list|expr, item, position)` `Insert Into(list|expr, item)`	Same as `Insert`, but does it in place. *List* or *expression* must be an L-value.
`Shift`	`x = Shift(list|expr)` `x = Shift(list|expr, n)`	Shift an item or *n* items from the front to the back of the *list* or *expression*. Shift items from back to front if *n* is negative.

Table 8.4 Functions for Manipulating Lists or Expressions *(Continued)*

Function	Syntax	Explanation
Shift Into	Shift Into(*list*|*expr*) Shift Into(*list*|*expr*, *n*)	Shift items in place.
Reverse	*x*=Reverse(*list*|*expr*)	Reverse the order of elements of a *list* or terms of an *expression*.
Reverse Into	Reverse Into(*list*|*expr*)	Reverse the order of elements of a *list* or terms of an *expression* in place.
Sort List	*x*=Sort List(*list*|*expr*)	Sort the elements of a *list* or the terms of an *expression*. Numbers sort low, followed by the name value of names, strings, or operators. For example 1+2 is lower than 1-2 because the name value Add sorts lower than the name value Subtract. {1,2} sorts lower than {1,3}, which sorts lower than {1,3,0}. {1000} sorts lower than {“a”}, but {a} and {“a”} sort as equal.
Sort List Into	Sort List Into(*list*|*expr*)	Sort the elements of a *list* or terms of an *expression* in place.
Sort Ascending	Sort Ascending(*list*|*matrix*)	Returns a copy of a *list* or *matrix* with the items in ascending order.
Sort Descending	Sort Descending(*list*|*matrix*)	Returns a copy of a *list* or *matrix* with the items in descending order.
Loc Sorted	Loc Sorted(*A*, *B*)	Creates a matrix of subscript positions where the values in matrix **A** match the values in matrix **B**. **A** must be a matrix sorted in ascending order.
Substitute	R = Substitute(*list*|*expr*, Expr(*pattern*), Expr(*replacement*), ...)	Finds all matches to the pattern in the *list* or *expression*, and replaces them with the replacement expression. Each pattern must be a name. The second and third arguments are evaluated before they are applied, so most of the time you must quote them with an Expr function. To delay evaluating an argument, use Name Expr instead of Expr. You can list multiple pattern-replacement pairs for multiple substitutions in one statement.

Table 8.4 Functions for Manipulating Lists or Expressions *(Continued)*

Function	Syntax	Explanation
Substitute Into	Substitute Into(*list|expr*, Expr(*pattern*), Expr(*replacement*), ...)	Substitute in place.

Advanced Scoping and Namespaces

Scripts that are used in production environments need to use more advanced scoping techniques to avoid collisions between scripts. JMP provides three progressively more advanced techniques:

- The `Names Default To Here()` function. If you have simple scripting needs, this single command might be sufficient. See "Names Default To Here".
- Scopes that are pre-defined by JMP. See "Scoped Names".
- Namespaces that you can create for your scripts. See "Namespaces".

Names Default To Here

If you write production scripts, you need to insulate the script from the current user environment. Otherwise, the variables that you use might interact with variables used by the user and by other scripts. The way to do this is to keep your names in a local environment, which you can do by setting an execution mode with the statement:

```
Names Default To Here( 1 );
```

Unqualified names in a script with the *Names Default To Here* mode turned on are private to that script. However, the names persist as long as the script persists, or as long as objects created by or holding the script are still active. It is recommend that all production scripts start with `Names Default To Here(1)` unless there is a specific reason not to do so. When the script uses an unqualified name in this mode, that name is resolved in the local namespace.

To refer to global variables, scope the name specifically as a global variable (for example, `::global_name`). To refer to columns in a data table, scope with name specifically as a data table column (for example, `:column_name`).

Note: `Names Default To Here( 1 )` defines a mode for a particular script. It is not a global function. One script can have this mode turned on, while another script can have it turned off. The default setting is off.

In JMP 8 and earlier, the only method to the insulate scripts was to use lengthy names that were less likely to collide with names in other scripts. Using `Names Default To Here(1)` makes this technique unnecessary.

If you have simple scripting needs, `Names Default To Here(1)` might be sufficient.

Local()

`Local()` creates local scopes only in specific contexts within a script and cannot enclose a longer script with interacting functions, while `Names Default To Here(1)` creates a local scope for an entire script. In `Local()`, you list the variables that are local to the local block. Anything not explicitly listed is not local to that block.

Local Here()

`Local Here()` provides a namespace block with `Names Default to Here(1)`. Use `Local Here()` to prevent name collisions when multiple scripts are executed from the same root namespace (for example, when a script executes two button scripts that have the same variables or an included script is run from the main script).

`Local()` does not always work due to the lifetime of the local block, but `Local Here()` is persistent across the call.

Here's an example that runs an included script from a main script. The scripts define x and y variables as different values. Variables in the included script are in the Local Here namespace.

Main.jsl:

```
Names Default To Here( 1 );

x = 5;
y = 0;

Print( "Main, Before: x=" || Char( x ) || ", y=" || Char( y ) );
Include( "Inc1.jsl" );
Print( "Main, After: x=" || Char( x ) || ", y=" || Char( y ) );
```

Inc1.jsl:

```
Names Default To Here( 1 );

Local Here( // variables are local to this script
  x = 100;
  y = 100;

Print( "Inc1: x=" || Char( x ) || ", y=" || Char( y ) );
);
Print( "Inc1 outside: x=" || Char( x ) || ", y=" || Char( y ) );
```

Return:

```
"Main, Before: x=5, y=0" // defined in Main.jsl, before running Inc1.jsl
"Inc1: x=100, y=100" // defined in Inc1.jsl
"Inc1 outside: x=5, y=0" // from Inc1.jsl, referring to Main.jsl
"Main, After: x=5, y=0" // from Main.jsl, after running Inc1.jsl
```

Notice that the x and y variables in inc1.jsl don't change because they're in the Local Here namespace.

Handling Unqualified Named Variable References

The `Names Default To Here()` function determines how *unqualified* named variable references are resolved. Explicitly scoping a variable using `here:var_name` always works, whether `Names Default To Here()` is on or off. See "Scoped Names" for more information about `here` and other scopes.

Enabling the *Names Default To Here* mode associates a scope called `Here` with an executing script. The `Here` scope contains all of the unqualified named variables that are created when they are the target of an assignment (as an L-value). In JMP 8 and earlier, these variables normally would have been placed in the `Global` scope. Using a `Here` scope keeps variables in multiple executing scripts separate from each other, avoiding name collisions and simplifying the scripting and management of variable name collisions. You can still share information using the `Global` scope.

Names Default To Here and Global Variables

Run this example script one line at a time to see how the `Names Default To Here()` function changes the resolution of variable names.

Example Script

```
a = 1;
Names Default To Here( 1 );
a = 5;
Show( global:a, a, here:a );
   global:a = 1;
   a = 5;
   here:a = 5;
```

1. Run the first line to create a global variable named *a* that holds the value 1.
2. Run the second line to turn on the *Names Default To Here* mode.
3. Run the third line to create a new variable named *a* in the local space that holds the value 5. This line does *not* change the value assigned to the global variable *a*.
4. Run the fourth line to see how scoped and unscoped variables are resolved.

The unqualified *a* is resolved to `here:a`. If `Names Default To Here()` were not on, *a* would be resolved to the global variable named *a*.

Note that if you use `::a` instead of `global:a` in the `Show()` function, your output is a little different:

```
Show(::a, a, here:a);
   a = 1;
   a = 5;
   here:a = 5;
```

Example of Using the Names Default To Here() Function

You have two scripts with the following definitions, and`Names Default To Here()` is turned *off* (the default condition) in both scripts.

Note: Both scripts must be in separate script windows for this example.

```
a = 1; // script 1
Show( a );

a = 3; // script 2
Show( a );
```

1. Run Script 1. Here is the result:

   ```
   a = 1
   ```

2. Run Script 2. Here is the result:

   ```
   a = 3
   ```

3. Run only the `show(a);` line in Script 1. Here is the result:

   ```
   a = 3
   ```

The log shows `a = 3` because variable *a* is global, and was last modified by Script 2. This is the default behavior in JMP 9 and later, and it is the only possible behavior in JMP 8 and earlier.

4. Now turn on `Names Default To Here()` in *both* scripts.

   ```
   Names Default To Here(1);
   ```

   Note: `Names Default To Here()` is local to a particular script. It is *not* a global setting.

5. Run Script 1. Here is the result:

   ```
   a = 1
   ```

6. Run Script 2. Here is the result:

   ```
   a = 3
   ```

7. Run only the `Show( a );` line in Script 1. Here is the result:

   ```
   a = 1
   ```

The log shows `a = 1`, because a copy of variable *a* is maintained for each script.

Note: Problems using this function are generally due to the mixing of unqualified and qualified references to global variables. Always explicitly scoping a name prevents accessing an unintended variable.

Scoped Names

Specify where a name is to be resolved by using a scope in the form `scope:name` where `scope` indicates how to resolve the name. For example, `here:name` indicates that the name should be resolved locally. Using the *Names Default To Here* mode, `here:name` is equivalent to `name`. The scope instructs how to look up the name.

The syntax is to use the colon scope operator:

```
scope:name
```

There are several types of scopes:

- Scope can be a resolution rule. For example, `here:x` means that `x` should be resolved to a name that is local to the script. `Global:x` means that `x` should be resolved to a global name.
- Scope can be a namespace reference variable. For example. `ref:a` means that `a` should be resolved within the namespace that `ref` refers to.
- Scope can be a data table reference to look up names as column names. For example, `dt:height` means that `height` should be resolved as a column in the data table that `dt` references.
- Scope can be the name of a namespace that you created. For example, `myNamespace:b` where `myNamespace` is a namespace that you created. `"myNamespace":b` is equivalent. See "Namespaces".

Note: If you create a function, make sure you scope it to avoid conflicts with built-in functions.

Examples of Scoping Column Formulas

The following examples demonstrate how to scope columns that contain formulas. In both scripts, *x* is a global variable, local variable, and column name.

In the first script, the column name x is unscoped. the formula in the second column multiplies the value in column `x` by 100. In this case, The result is a column with the values 100, 200, and 300.

```
::x = 5;
```

```
New Table( "Test",
   New Column( "x", Values( [1, 2, 3] ) ),
   New Column( "y", Formula( 100 * x ) ),
);
```

In the following script, the formula in column y assigns 500 to x and then adds 50 to x. Each cell in the column contains the value 550.

```
::x = 5;
New Table( "Test",
   New Column( "x", Values( [1, 2, 3] ) ),
   New Column( "y", Formula( Local( {x = 500}, x + 50 ) ) ),
);
```

Predefined Scopes

JMP provides predefined that cannot be removed or replaced. Each of these scopes has specific roles, depending on its associated object.

Table 8.5 Predefined Scopes

Scope	Description
`Global`	Global names are shared throughout the JMP environment.
`Here`	Scope of the executing script.
`Builtin`	JMP built-in functions. For example, `Builtin:Sqrt()`. These names are shared throughout the JMP environment. If you over-ride a JSL function with a custom function, you can still access the built-in JSL function by using this scope.
`Local`	Nearest local scope. Can be nested within the user-defined functions, `Local` and `Parameter`.
`Local Here`	Provides a namespace block inside `Names Default to Here(1)`. Use `Local Here()` to prevent name collisions when multiple scripts are executed from the same root namespace (for example, when a script executes two button scripts that have the same variables). The argument can be any valid JSL expression. `Local( {Default Local}, )` does not always work due to the lifetime of the local block, but `Local Here()` is persistent across the call.
`Window`	Scope of the containing user-defined window. (Rare.)
`Platform`	Scope of the current platform. (Rare.)

Table 8.5 Predefined Scopes *(Continued)*

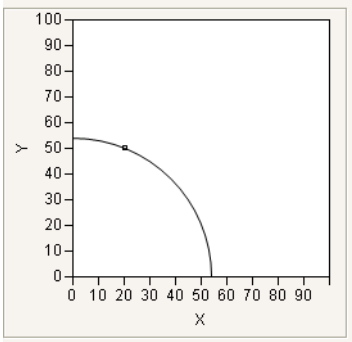
Scope	Description
Box	Scope of the containing context box. A context box is nested within a user-defined window. (Rare.)

Example of Using the Window Scope

This example uses the `Window` scope to pass information during execution. Explicitly scoping the variables `x` and `y` to this window ensures that JMP does not try to scope `x` and `y` in other contexts, such as a data table. The variables `x` and `y` are created and used solely inside the `Window` environment. The `Window` scope is similar to using `Local()`, but more useful because `Local()` is limited in the places that it can be used.

```
New Window( "Example",
   window:gx = 20;
   window:gy = 50;
   Graph Box(
      Frame Size( 200, 200 ),
      Handle(
         window:gx,
         window:gy,
         Function( {x, y},
            window:gx = x;
            window:gy = y;
         )
      );
      Circle( {0, 0}, Sqrt( window:gx * window:gx + window:gy * window:gy ) );
   );
);
```

Figure 8.1 Example of Current Window Namespace



Example of Using the Here Scope

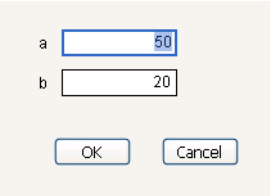
This example uses the `Here` scope to pass information between windows that are created by the same script. Scoping a variable using `Here:` is not dependent on turning `Names Default To Here()` on. The `Here:` scope is always available.

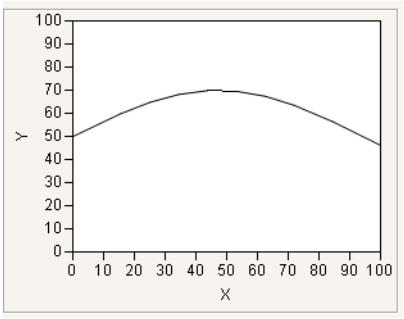
This script produces two windows and uses two different scopes.

The Launcher window asks the user for two values. Those two values are passed to the Output window, which uses them to graph a function. The Launcher window scopes `aBox` and `bBox` to the window: essentially, those two variables (pointers to Number Edit Boxes) exist only in the Launcher window and are not available to the Output window. The values from those two boxes are then copied into variables that are scoped to `Here`, and so are available to both windows that are produced by this script.

```
launchWin = New Window( "Launcher",
   <<Modal,
   V List Box(
      Lineup Box( N Col( 2 ), Spacing( 10 ),
         Text Box( "a" ),
         window:aBox = Number Edit Box( 50 ),
         Text Box( "b" ),
         window:bBox = Number Edit Box( 20 ),

      ),
      Lineup Box( N Col( 2 ), Spacing( 20 ),
         Button Box( "OK",
            // copy values before window goes away
            here:a = window:aBox << Get;
            here:b = window:bBox << Get;
         ),
         Button Box( "Cancel", Throw( 1 ) )
      ),

   )
);

New Window( "Output",
   Graph Box( Y Function( here:a + here:b * Sin( x / 30 ), x ) )
);
```

Figure 8.2 Launcher and Output





Namespaces

A *namespace* is a collection of unique names and corresponding values. You can store references to namespaces in variables. Namespace names are global, because JMP has only one namespace map. Namespace references are variables like any other variable that references an object, so they must be unique within their scope or namespace. The members of a namespace are referenced with the `:` scoping operator, such as `my_namespace:x` to refer to the object that is named `x` within the namespace called `my_namespace`. See "User-Defined Namespace Functions" for more information about creating and managing your own namespaces. Namespaces are especially useful for avoiding name collisions between different scripts.

User-Defined Namespace Functions

Create your own namespaces to hold related sets of variables and function definitions. There are several functions that you can use to manage namespaces.

New Namespace

```
nsref = New Namespace( <"nsname">, <{ name = expr, ... }> );
```

Creates a new namespace called `nsname` (a string expression) and returns a reference to the namespace. All arguments are optional.

`Nsname` is the name of the namespace in the internal global list of namespace names. `Nsname` can be used as the prefix in a scoped variable. The function returns a reference to the namespace, and can also be used as the prefix in a scoped variable reference. If `nsname` is absent, the namespace is anonymous and is given a unique name created by JMP. `Show Namespace()` shows all namespaces and their names, whether assigned or anonymous.

Important: If you already have a namespace named `nsname`, it is replaced. This behavior means that while you are developing your script, you can make your changes and re-run the script without having to clear or delete your namespace. To avoid unintentional replacement, you can either use anonymous namespaces, or test to see whether a particular namespace already exists:

```
If( !Namespace Exists( "nsname" ), New Namespace( "nsname" ) );
```

A list of named expressions is optional. The names are JMP variables that exist only within the namespace.

Note: The named expressions must be a comma-separated list. Separating the expressions with semi-colons causes the list to be ignored.

These namespaces must be uniquely named to prevent collisions in situations where multiple user-defined namespaces are being used. Using anonymous namespace names prevents collisions.

Namespace

```
nsref = Namespace( "nsname" | nsref);
```

Returns a namespace reference. The argument might be either of the following:

- a quoted string that contains a namespace
- a reference to a namespace

Note: `Namespace()` returns a reference to a namespace that already exists. It does not create a new namespace.

Is Namespace

```
b = Is Namespace( nsref );
```

Returns `1` (`true`) if *`nsref`* is a namespace or `0` (`false`) otherwise.

As Scoped

```
b = As Scoped( "nsname", var_name);
nsname:var_name;
```

`As Scoped()` is the function form of a scoped reference. The function returns a reference to the specified variable in the specified scope.

Namespace Exists

```
b = Namespace Exists( "nsname" );
```

Returns `1` (`true`) if *`nsname`* exists in the list of global namespaces, or `0` (`false`) otherwise.

Show Namespaces

```
Show Namespaces();
```

Shows the contents of all namespaces contained in the list of global namespaces. Namespaces are not visible unless a reference is made to one, using either the `New Namespace` or `Namespace` functions.

Namespace Messages

In addition to the namespace management functions, a namespace also supports a set of messages to access and manipulate its contents.

Note that these messages, as with all message, must be sent to a scriptable object. A namespace name is not a defined scriptable object and cannot be used in a Send operation. However, you can use the name of a namespace in variable references. For example, `nsmane::var` is equivalent to `nsref::var`.

See the *JSL Syntax Reference* for descriptions of the messages that are supported by user-defined namespace references.

Using Namespace References

The following are all equivalent references to a variable that is named `b` in the namespace that is named `nsname` that has a reference `nsref`:

```
nsref:b
nsname:b
"nsname":b
nsref["b"]
nsref<<Get Value("b") // used as an r-value
```

Namespaces and Included Scripts

An included script runs in the namespace of the parent script. If the included script has its own namespace definitions, you need to do one of the following:

- manage the namespace names to avoid name collisions
- use anonymous names created by the `New Namespace` function

In either case, you still need to manage variable references to namespaces.

There is also an option for the `Include` function (`New Context`) that creates a namespace that the included script runs in. This namespace is anonymous and it is independent from the parent script's namespace. For example:

```
Include("file.jsl", <<New Context);
```

This anonymous namespace can be referenced using `Here`.

If both the parent and included scripts use the global namespace, add `Names Default To Here` to the `New Context` option to avoid name collisions. For example:

```
Include("file.jsl", <<New Context, <<Names Default To Here);
```

See "Includes" for more information about the `Include` function.

Examples of User-Defined Namespaces

Creating and Using a Basic Namespace with Expressions

This example shows creating an anonymous namespace and using functions and variables within it.

```
new_emp = New Namespace(
   {name_string = "Hello, *NAME*!",

   print_greeting = Function( {a},
      Print( Substitute( new_emp:name_string, "*NAME*", Char( a ) ) )
   )}
);
```

Note that you must use the fully qualified name for variables defined within the namespace.

```
new_emp:print_greeting( 6 );
   "Hello, 6!"
```

Complex Number Operations

This example creates a namespace that contains functions to support using two-element lists to represent complex numbers, and then locks the namespace.

```
If( !Namespace Exists( "complex" ),
   New Namespace(
      "complex"
   );
   complex:makec = Function( {a, b},
      Eval List( {a, b} )
   );
   complex:addc = Function( {a, b}, a + b );
   complex:subtractc = Function( {a, b}, a - b );
   complex:multiplyc = Function( {a, b},
```

```
      Eval List( {a[1] :* b[1] - a[2] :* b[2], a[1] :* b[2] + a[2] :* b[1]} )
   );
   complex:dividec = Function( {a, b},
      d = b[1] ^ 2 + b[2] ^ 2;
      Eval List(
         {a[1] :* b[1] - a[2] :* b[2] / d, a[2] :* b[1] - a[1] :* b[2] / d}
      );
   );
   complex:charc = Function( {a},
      Char( a[1] ) || "+" || Char( a[2] ) || "i"
   );
);
Namespace( "complex" ) << Lock;
```

Here are examples using functions that are within the above user-defined namespace.

```
c1 = complex:makec( 3, 4 );
c2 = complex:makec( 5, 6 );
cadd = complex:addc( c1, c2 ); // returns {8, 10}
csum = complex:subtractc( c1, c2 ); // returns {-2, -2}
cmul = complex:multiplyc( c1, c2 ); // returns {-9, 38}
cdiv = complex:dividec( c1, c2 ); // returns {14.6065573770492,
  19.7049180327869}
Show( complex:charc( c1 ) ); // returns complex:char(c1) = "3+4i";
cm1 = complex:makec( [1, 2, 3], [4, 5, 6] ); // returns {[1, 2, 3], [4, 5, 6]}
```

Referencing Namespaces and Scopes

There are a number of factors in resolving a named variable reference. Table 8.6 describes the named variable references that are resolved for specific situations.

Notes:

- In JMP 9 and later, `a`, `:a`, and `::a` have the same meaning with the *Names Default To Here* mode turned off.
- If the current point of execution is in a user-defined function, or a Local or Parameter JSL function body, then the Local namespace is used for creation when `Default Local` is specified. However, the reference rule is unchanged.

Table 8.6 Namespace References

Form	Reference Type	Reference Rule	Creation Rule
`a`	Unqualified	If the *Names Default To Here* mode is on, JMP looks for the variable in these locations: • `Local` namespace • `Here` namespace • current data table If the *Names Default To Here* mode is off, JMP looks for the variable in these locations: • `Local` namespace • `Here` namespace • `Global` namespace • current data table	• If the *Names Default To Here* mode is on, then JMP creates the variable in the `Local` namespace or in the `Here` namespace. • If the *Names Default To Here* mode is off, then JMP creates the variable in the `Local` namespace or in the `Global` namespace.
`:a`	Current data table	JMP looks for the variable in the current data table.	(Not applicable)
`::a` `Global:a`	Global	JMP looks for the variable in the `Global` namespace.	JMP creates the variable in the `Global` namespace.

`ns:a`
`dt:a`
`Here:a`
`"name":a`
`expr:a`
	Qualified	JMP looks for the variable in the specified namespace. If the variable is not found, an error results.	JMP creates the variable in the specified namespace. Any previous values are replaced.

`ns["a"]` `ns[expr]`	Subscript	JMP looks for the variable in the specified namespace. If the variable is not found, an error results.	JMP creates the variable in the specified namespace. Any previous values are replaced.
`Platform:a`	Qualified	JMP looks for the variable in the encapsulating platform.	JMP creates the variable in the encapsulating platform.

Table 8.6 Namespace References *(Continued)*

Form	Reference Type	Reference Rule	Creation Rule
`Local:a`	Qualified	JMP looks for the variable within any nested local function, up to and including a function call boundary. See "Example of Local:a".	JMP creates the variable in the innermost nested local function or function call boundary.
`Window:a`	Qualified	JMP looks for the variable in the encapsulating `New Window` window namespace.	JMP creates the variable in the encapsulating `New Window` window namespace.
`Box:a`	Qualified	JMP looks for the variable in the encapsulating `Context Box` namespace contained in a `New Window` window.	JMP creates the variable in the encapsulating `Context Box` namespace contained in a `New Window` window.

Example of Local:a

Sample Script	Log Output

`Delete Symbols();`
`Local( {d1l1 = 12},`
`  local:f1f1 = Function( {fa1, fa2},`
`  {fl1 = 99},`
`  local:fa12 = fa1 + fa2;`
`  Local( {d2l1 = 56},`
`  local:l2l2 = 78;`
`  Show( fa12 );`
`  Show( fl1 );`
`  Try( Show( d1l1 ), Write(`
`"\!n\!n***Error=" || Char( exception_msg`
`) || "\!n" ) );`
`  Show Symbols();`
`  );`
`  local:fa12;`
`  );`
`  f1f1( 2, 3 );`
`);`
	fa12 = 5;
	fl1 = 99;
	dil1 = 12;
	
	// Local
	
	d2l1 = 56;
	l2l2 = 78;
	
	// 2 Local
	
	// Local
	
	fa1 = 2;
	fa12 = 5;
	fa2 = 3;
	fl1 = 99;
	
	// 4 Local
	
	// Local
	
	d1l1 = 12;
	
	// 1 Local
	
	5

Resolving Named Variable References

When variables are referenced within a JMP script, JMP resolves the storage location of the variable using a specific set of rules. If the variable is referenced by a qualified name, then the resolution is based on the specific qualification specification. If the variable is referenced by an unqualified name, the situation is a bit more complex. JMP looks through a hierarchy of scopes representing the point of execution with the executing script. This section describes the rules that are used to resolve named variable references.

By default, variable name resolution in JMP 9 and later worked the same way as in JMP 8 and earlier, allowing your current JSL scripts to be executed unchanged. For JMP 9 and later, the difference between qualified and unqualified variable named references is important to understand.

Qualified Named References

A qualified named reference uses the `:` and `::` operators to provide specific information about where a referenced variable resides, or where it is created. Examples of qualified named references include the following:

```
:var
::globalvar
datatable:var
nsref:var
"nsname":var
```

Unqualified Named References

An unqualified named reference provides no explicit information to completely identify where a variable resides or where it is created. No scoping operator (`:` or `::`) is specified in the reference. To change the behavior of JMP when resolving unqualified named variable references, use the `Names Default To Here(1)` function. For more information about variable name resolution, see the "Rules for Name Resolution".

Rules for Resolving Variable References

JMP applies rules in sequence to resolve variable references. See "Rules for Resolving Names".

Best Practices for Advanced Scripting

Minimize Polluting the Global Namespace and Prevent Scripts from Interacting

Always start your script with this line:

```
Names Default To Here(1);
```

Share Variables Across Scripts

Use named namespaces. Namespace names are placed in the global scope.

Use Anonymous Namespaces

Using namespace references to anonymous namespaces avoids possible conflicts with other namespaces.

Advanced Programming Concepts

This section covers some more advanced programming techniques that can be useful for developing scripts.

- "Throw and Catch Exceptions"
- "Functions"
- "Recursion"
- "Includes"
- "Load and Save Text Files"
- "Run External Programs"
- "Classes"

Throw and Catch Exceptions

A script can stop itself by executing the `Throw()` function. If you want to escape from part of a script when it is in an error condition, you can enclose it in a `Try()` expression.

If you include a character-valued expression as a `Throw()` argument, one of the following occurs:

- Throwing stores the string in a global variable named `exception_msg`. This is illustrated in the first example below.
- Throwing stores the string in a local variable named `exception_msg` if you define `exception_msg` in the Here namespace using `Names Default to Here(1)`.

`Try()` takes two expression arguments. It starts by evaluating the first expression, and if or when the first expression throws an exception by evaluating `Throw`, it does the following:

1. Immediately stops evaluating that first expression.
2. Returns nothing
3. Evaluates the second expression.

Examples

You can use `Try()` and `Throw()` to catch an exception that JMP itself throws.

```
Try( Throw( "Hello." ), Show( exception_msg ) );
  exception_msg = "Hello.";
```

The following example prints a message to the log when the `Try()` expression cannot be executed:

```
Try(
```

```
    dt = Open( "Mydata.jmp" ); // a file that cannot be opened
    Summarize( a = by( age ), c = count, meanHt = Mean( Height ) );
    Show( a, c, meanHt );
,
    Print( "This script does not work without the data set." );
    Throw();
);
```

If the Throw() string begins with "!" and is inside a Try() expression, throwing creates an error message about where the exception was caught.

```
Try(
    Throw( "!This message has an exclamation point." );
,
    Write( "\!Nexception_msg: ", exception_msg );
    Print( "Hello from the catch block (WITH exclamation)." );
);
Print( "This AFTER message will NOT print due to error." );
```

When writing the Try() error handler code, consider wrapping the exception variable exception_msg in a Char() function. If your script calls Concat() without wrapping it in a Char() call, you might get an error depending on what was actually returned in this variable. Though some commands return a string in exception_msg, other commands (such as Open()) return a list or another data structure. This would cause an error in the error handler if it were passed directly as an argument to Concat().

```
Try(
    Open( "no file" ),
    If( Is Empty( exception_msg ),
        // use Write to print "no error"if exception_msg is empty
        Write( "\!Nexception_msg (0): no error" ),
        // use Print to print the exception message
        Print( "\!Nexception_msg (1): ", exception_msg );
        /* or first make sure that exception message is a string
        and then use Concat to build the message that is passed to Print(). */
        Print( "\!Nexception_msg (2): " || Char( exception_msg ) );
    )
);
    exception_msg (1): "
    {"Cannot open table"(1, 2, "Open", Open /*###*/("no file"))}
    "
    exception_msg (2): {\!"Cannot open table\!"(1, 2, \!"Open\!", Open
        /*###*/(\!"no file\!"))}"
```

You can also use Try() and Throw() to escape from deep inside For loops.

```
a = [1 2 3, 4 5 ., 7 8 9];
b = a;
nr = N Row( a );
```

```
nc = N Col( a );
// a[2, 3] = 2; // uncomment this line to see the "Missing b" outcome

Try(
   sum = 0;
   For( i = 1, i <= nr, i++,
      For( j = 1, j <= nc, j++,
         za = a[i, j];
         If( Is Missing( za ),
            Throw( "Missing a" )
         );
         zb = b[j, i];
         If( Is Missing( zb ),
            Throw( "Missing b" )
         );
         sum += za * zb;
      )
   );
,
   Show( i, j, exception_msg );
   Throw();
);
   i = 2;
   j = 3;
   exception_msg = "Missing a";
```

You do not have to use `Try()` to make use of `Throw()`. In this example, `Throw()` is not caught by `Try()` but still stops a script that cannot proceed:

```
dt = New Table(); // to get an empty data table
If( N Row( dt ) == 0,
   Throw( "!Empty Data Table" )
);
```

Functions

JSL also has a function called `Function` to extend the macro concept with a local context arguments. Suppose that you want to create a function that takes the square root but tolerates negative arguments, returning zero rather than errors. You first specify the local arguments in a list with braces { } and then state the expression directly. You do not need to enclose the expression in `Expr` because `Function` stores it as an expression implicitly.

```
myRoot = Function( {x},
   If( x > 0, Sqrt( x ), 0 )
);
a = myRoot( 4 );  // returns 2
b = myRoot( -1 ); // returns 0
```

Functions are stored in variables, the same as values. This means that you cannot have both a root function and a root value. It also means that you can redefine a function anytime except when you are inside the function itself.

When a function is called, its arguments are evaluated and given to the local variables specified in the list forming the first argument. Then the body of the function, the second argument, is evaluated.

The values of the arguments are for the temporary use of the function. When the function is exited, the values are discarded. The only value returned is the return value. If you want to return several values, then return a list instead of a single value.

In defined functions, the stored function is not accessible directly, even by the `Name Expr` command. If you need to access the function expression in your script, you have to create the function within an `expr()` clause. For example,

```
makeFunction = Expr(
   myRoot = Function( {x},
      If( x > 0, Sqrt( x ), 0 )
   )
);
d = Substitute( Name Expr( MakeFunction ), Expr( x ), Expr( y ) );
Show( d );
makeFunction;
```

Optional Arguments

You can create optional arguments. For example:

```
f1 = Function( {x, y, z}); // all arguments are required
f2 = Function( {x, y=2, z=4}); // x is required, y and z are optional
```

Notes:

- To make an argument optional, give it a default value in the function's definition (x=1 rather than just x).
- Optional arguments always must follow a required argument. For example, this is not allowed:

  ```
  Function( {x = 1, y}, ... )
  ```

  It will work when defined, but you will always have to pass in a value for x and y, because the default value will never be used.
- When you call the function, you cannot skip optional arguments. In this example, you have to supply a value for y to supply a value for z when you call the function:

  ```
  ex = Function( {x, y = 2, z = 3},
    Return( x + y + z )
  );
  ex( 1, 4 ); // passes in 1 for x and 4 for y; z will be 3. returns 8.
  ```

Local Symbols

You can declare variables as local to a function so that they do not affect the global symbol space. This is particularly useful for recursive functions, which need to keep separate the values of the local variables at each level of function call evaluation.

As shown above, a function definition looks like this:

```
functionName=Function({arg1, ...}, body);
```

You can also have the function definition default all the unscoped names to be local.

```
functionName=Function({arg1, ...}, {Default Local}, body);
```

For the purposes of creation, the use of `Default Local` localizes all the names that meet the following qualifications:

- Are not scoped as globals (for example, `::name`)
- Are not scoped as data table column names (for example, `:name`)
- Occur without parentheses after them (for example, are not of the form `name(...)`)

For example, the following function sums three numbers.

```
add3 = Function( {a, b, c},
   {temp},
   temp = a + b;
   temp + c;
);
X = add3( 1, 5, 9 );
   15
```

The following function does the same thing, automatically finding locals.

```
add3 = Function( {a, b, c},
   {Default Local},
   temp = a + b;
   temp + c;
);
X = add3( 1, 5, 9 );
   15
```

In both cases, the variable `temp` is not a global, or, if it is already a global, remains untouched by evaluating the functions.

Using `Default Local` in user-defined functions can cause some confusion because it is context-sensitive. That is, the same function may behave differently in different contexts (depending on whether same-named outer variables are in scope) because the reference rules for unqualified names are unchanged when using `Default Local`. To make a function less context-sensitive, enumerate each and every variable that you want to be local. This reduces the confusion and the potential incorrect use of outer scope variable values.

Recursion

The `Recurse()` function makes a recursive call of the defining function. For example, you can make a function to calculate factorials. A factorial is the product of a number, the number minus 1, the number minus 2, and so on, down to 1.

```
myfactorial = Function( {a},
   If( a == 1,
      1,
      a * Recurse( a - 1 )
   )
);
myfactorial( 5 );
   120
```

You can define recursive calculations without using `Recurse()`. For example, you could replace `Recurse()` by `myfactorial`, and the script would still work. However, `Recurse()` offers these advantages:

- It avoids name conflicts when a local variable has the same name as the function.
- You can recurse even if the function itself has not been named (for example, assigned to a global variable, such as *myfactorial* above).

Includes

The `Include()` function opens a script file, parses the script in it, and executes the JSL in the specified file.

```
Include( "pathname" );
```

For example,

```
Include( "$SAMPLE_SCRIPTS/myStartupScript.jsl" );
```

There is an option to obtain the parsed expression from the file, rather than evaluating the file.

```
Include( "pathname", <<Parse Only );
```

Another named option creates a namespace that the included script runs in. This namespace is an anonymous namespace and it is independent from the parent script's namespace.

```
Include( "file.jsl", <<New Context );
```

See "Advanced Scoping and Namespaces" for information about using namespaces with your scripts.

Note the following about included files:

- JMP files aside from JSL cannot be used.

- Other recognized file types, such as image files, SAS data sets, and Microsoft Excel files cannot be used.
- Unrecognized file types are treated as a JSL file.
- Files with the .txt extension are treated as a JSL file. A text file that contains data can be included, however an error will appear since this is not valid JSL.

Load and Save Text Files

The `Load Text File()` and `Save Text File()` commands allow manipulation of text files from JSL. Note that the paths in the following code are strings.

```
text = Load Text File( "path" );
Save Text File( "path", text );
```

You can load a text file from a Web site:

```
Load Text File( "URL", <blob> );
```

The URL is a quoted string that contains the URL for the text file. The text file is returned as a string. If you add the optional named argument *blob*, a blob is returned instead.

Run External Programs

`Run Program()` enables you to control a program that uses standard input and standard output. Line-mode programs read commands from stdin and write answers to stdout. cmd.exe, the Windows command prompt, is one example. You can also use it to launch GUI programs, such as Windows Notepad.

```
Run Program(
   Executable( "cmd.exe" ),
   Options( {"/c", "TaskList\!n"} ),
   Read Function( "text" ) // returns all text
);
```

The preceding script writes the Windows task list and details to the log.

Figure 8.3 Log Output

```
"
Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
System Idle Process              0 Services                   0         24 K
System                           4 Services                   0     64,340 K
smss.exe                       516 Services                   0      1,568 K
```

You can also print the task list directly without using cmd.exe and then import the task list into a data table.

```
blob = Run Program(
   Executable( "tasklist.exe" ),
   Read Function( "blob" )
);

Open(
   blob,
   Columns(
      Column( "Image Name", Character, "Nominal" ),
      Column(
         "PID",
         Numeric,
         "Continuous",
         Format( "Best", 10 )
      ),
      Column( "Session Name", Character, "Nominal" ),
      Column(
         "Session#",
         Numeric,
         "Continuous",
         Format( "Best", 10 )
      ),
      Column( "Mem Usage", Character, "Nominal" ),
      Omitted Column( . )
   ),
   Import Settings(
      Fixed Column Widths( 26, 9, 17, 12, 12, 63 ),
      Labels( 1 ),
      Column Names Start( 2 ),
      Data Starts( 4 )
   )
);
```

Figure 8.4 Imported Data from Run Program()

	Image Name	PID	Session Name	Session#	Mem Usage
1	System Idle Proc...	0	Services	0	8 K
2	System	4	Services	0	42,560 K
3	smss.exe	456	Services	0	1,228 K

See the *JSL Syntax Reference*.

Note: If you use `Run Program()` to launch a GUI program, JMP is inoperable until you close the program.

Classes

Classes enable you to create objects based on templates. You can use JSL to create a definition of a class. A class definition is a template, which contains definitions for member variables, methods, and functions. After you have defined a class in JSL, you can instantiate an object that has all of the characteristics defined in the class definition.

Use the JSL function `Define Class()` to create a JSL class definition. Use the JSL function `New Object()` to instantiate an instance of a JSL class.

The JSL function `Define Class()` has a number of arguments that define the contents of the class:

```
Define Class( "class name",
   <Base Class( "base class name", <"base class name", ...> ),>
   <Show( All( Boolean ) ) | Show( <Members( Boolean ),> <Methods( Boolean ),>
   <Functions( Boolean )> ),>
   <Assignment Statements>
);
```

Where:

- `"class name"` is the name of the class being defined.
- `"base class name"` is one or more class names that make the set of base classes that are used as the foundation of the class being created.
- `Show` defines the value that appears when a class object is written out using the `Show()`, `Print()`, or `Write()` functions. Options include:
  - `All( Boolean )` shows all members, methods, and functions defined in a class object instance. The default value for All is true. If you specify the `All` option, that setting overrides any other arguments in the `Show` option.
  - `Members( Boolean )` shows all member variables contained in a class object instance.
  - `Methods( Boolean )` shows all methods contained in a class object instance.
  - `Functions( Boolean )` shows all functions contained in a class object instance.

  Note: You can specify either the `All` option or any number of the `Members`, `Methods`, and `Functions` options, but you cannot specify both options at the same time.

  Note: If a class contains a `_show_` method, that method overrides the `Show` option.

- `Assignment Statements` are zero or more of the following assignment statements, separated by semicolons:
  - `Member = JSL Expression`
  - `Method Name = Method( {<argument, ...>}, body )`

- `Function Name = Function( {<argument, ...}, {<local variable, ...>}, body )`

Note: The Assignment Statements can alternatively be specified as a list of JSL expressions.

There are three specialized methods that you can use in a class definition:

init Initializes the object at the time it is instantiated. This is considered the constructor for the class. You can have only one constructor per class.

show Defines the output of the `Show()`, `Print()`, and `Write()` JSL functions when an instance of the class is used as the argument.

to string Defines the output of the `Char()` JSL function when an instance of the class is used as the argument.

Notes:

- All methods, member variables, and functions are public.
- Functions defined within a class cannot access method variables or methods within the class.

Example of Defining a Class

The following example defines a simple class for a pet, containing the following information: name, breed, and color.

In this example, you define the constructor to require all information to create a new Pet object. Also, you define a set of accessor methods to obtain each of the attributes of a Pet object.

```
Define Class( "Pet",
   name = ""; breed = ""; color = "";
   _init_ = Method( { name, breed, color },
      this:name = name;
      this:color = color;
      this:breed = breed;
   );
   _show_ = Method( {},
      Return( name || " is a " || color || " " || breed || "." )
   );
   Get Pet Name = Method( {}, Return( name ) )
);
```

Instantiating a Class

To instantiate an instance of a JSL class object, use `New Object()` JSL function with any of the following functions and arguments:

```
New Object( "Class Name"( constructor arguments ) );
New Object( Class Name( constructor arguments ) );
New Object( Class Reference( constructor arguments ) );
New Object( "Class Name", { constructor arguments } );
New Object( Class Name, { constructor arguments } );
New Object( Class Reference, { constructor arguments } );
```

Examples of Instantiating and Using a Class

The following example shows how to instantiate a class to create and access JSL class objects.

```
class = "Pet";
xa = New Object( Pet( "Oliver", "Schnauzer", "Black" ) );
xb = New Object( class, { "Flynn", "Cockapoo", "Brown" } );
xc = New Object( class( "Mandi", "King Charles Cavalier", "Ruby" ) );
xd = New Object( "Pet"( "Charlie", "Asian", "White" ) );
xd1 = New Object( xd( "Fred", "Siamese", "Silver" ) );
xd2 = New Object( xd, { "Sam", "German Shepard", "Black and White" } );

Show( xa );
Show( xb );
Show( xc );
Show( xd );
Show( xd1 );
Show( xd2 );
xd2 = xd;                  // returns a second reference to xd
xdclone = xd << Clone;     // copy xd to a new instance, named xdclone
xd2:name = "Jerry";
Show( xd:Get Pet Name(), xd2:Get Pet Name(), xdclone:Get Pet Name() );
```

This example creates multiple instances of the Pet class. The output of the `Show()` JSL function is determined by the `_show_` method that is defined in the Pet class definition. This example also illustrates the difference between creating a second JSL reference to a class instance and cloning a class instance. When using the `Clone` message, you are creating a new class instance that is independent of the original. However, the new instance starts out as a copy of the original.

File and Directory Operations

`Pick Directory()` and `Pick File()` enable the user to select a file or directory when the script is run. To get a list of files in a directory, use `Files In Directory()`.

- "Select a Directory or File"
- "Get a List of Filenames"
- "Connect to a Network Directory"

Select a Directory or File

You can prompt the user to select a directory using the `Pick Directory()` function. The command displays a platform-specific window in which the user selects a folder. On Windows, the optional prompt string appears at the top of the Browse window (Windows) or below "Open" on the Finder window (macOS). You can also specify the path for an initial directory that appears, and specify whether files should appear in the Pick Directory window.

```
Pick Directory ( "Select a directory." );
```

To let the user select a file, use the `Pick File()` function:

```
Pick File(
  <"prompt message">, <"initial directory"> <{filter list}>,
  <first filter>, <save flag>, <"default file">,
  <multiple>)
```

The `"prompt message"` is used as the window title. The `"initial directory"` defines which folder initially appears. If a directory is defined as an empty string, the default directory is used.

You can also define the `{filter list}` used for the `Open()` window, forcing it to show only certain file types. This list must use the following syntax:

```
{"Label1|suffix1;suffix2;suffix3", "Label2|suffix4;suffix5"}
```

Each quoted string adds an entry to the **File name** list in the `Open()` window. `Label` defines the text that is displayed for each menu option. The following list of suffixes defines the file types that are displayed if its corresponding label is selected. Note the use of `"*"` to list all files in the window.

```
Pick File(
  "Select JMP File", // prompt message
  "$SAMPLE_DATA", // initial directory
  {"JMP Files|jmp;jsl;jrn", "All Files|*"}, // file filter list
  1, // intially selected item
  0, // doesn't prompt the user to save the file
```

```
    "Analgesics.jmp" // file that is selected by default
);
```

Tip: All arguments are optional, however, they are also positional. This means that you can leave out arguments only at the end of the script. Use empty strings for the arguments that you want to omit from the beginning of the script.

The script below does not set the default directory or the default file:

```
Pick File(
   "Select JMP File",
   "", // no default directory
   {"JMP Files|jmp;jsl;jrn", "All Files|*"},
   1,
   0,
   "" // no default file
);
```

The <first filter> argument sets the default selection where n is the index for the list item. In the script above, the <first filter> is the first item in the list: "JMP Files|jmp;jsl;jrn".

If <Save Flag> is false, the Multiple argument can be added to allow the user to select multiple files using the one window:

```
Pick File(
   "Select JMP File",
   "",
   {"JMP Files|jmp;jsl;jrn", "All Files|*"},
   1,
   0,
   "",
   multiple // save flag is 0, allows multiple file selection
);
```

Get a List of Filenames

To obtain a list of filenames in a specific directory, use the Files In Directory command.

```
Files In Directory( path, <recursive( 0 | 1 )> );
```

Both filenames and subdirectory names are returned as shown in the following example:

```
Files In Directory( "$SAMPLE_DATA" );
   { "2D Gaussian Process Example.jmp", "Abrasion.jmp", ... "Design
     Experiment", "Detergent.jmp", ... }
```

Notice that the files within the Design Experiment subdirectory are not included. And only files in the root $SAMPLE_DATA directory are listed.

To return a list of all file names, add the optional `recursive(Boolean)` argument to `Files In Directory`:

```
Files In Directory( "$SAMPLE_DATA", recursive( 1 ) );
   { "2D Gaussian Process Example.jmp", "Abrasion.jmp", ... "Design
     Experiment/2x3x4 Factorial.jmp", "Design Experiment/Algorithm Data.jmp",
     ... }
```

To get the full pathnames, recurse the directories and concatenate the file paths and file names. The following example loops through each file in the $SAMPLE_DATA directory and subdirectories. The file path is concatenated to each file name.

```
names = Files In Directory( "$SAMPLE_DATA", recursive( 1 ) );
For( i = 1, i <= N Items( names ), i++,
   names[i] = Convert File Path( "$SAMPLE_DATA" ) || names[i]
);
names;
   {"/C:/Program Files/SAS/JMP/16/Samples/Data/2D Gaussian Process
     Example.jmp",
   "/C:/Program Files/SAS/JMP/16/Samples/Data/Abrasion.jmp", ... }
```

The `Files in Directory` command accepts native and POSIX paths, as well as paths using path variables. See "Path Variables" for more information about working with paths.

Connect to a Network Directory

On Windows, you can connect to a network directory without connecting to the shared drive first.

```
Files in Directory( "\\myserver.company.com/source/Users/Smith" );
```

On macOS, you have to mount the volume first and then connect to the directory using a combination of JSL and AppleScript. Note that you can also modify this script to work on Windows. On macOS only, the user is prompted for a password.

```
mount = Function( {server},
   {Default Local},

   If( Host is( "Mac" ),

      /* Use osascript to mount the volume.
      The user will be prompted for credentials. */
      Run Program(
         Executable( "/usr/bin/osascript" ),
         Options( {"-e", Eval Insert( "mount volume \!"smb://^server^\!"" )} ),
         Read Function( "text" )
      );
      server = Concat Items( Remove( Words( server, "/" ), 1 ), "/" );
      Eval Insert( "/Volumes/^server^" );
```

```
    ,
       // Windows can use UNC to auto-mount the server.
       Eval Insert( "\!\\!\^server^" )
    )
);


users = Mount( "myserver.company.com/Users" );
Files In Directory( Eval Insert( "^users^/Smith" ) );
```

Scripting By Groups

By group arguments are supported for these functions: ColMean(), ColStdDev(), ColNumber(), ColNMissing(), ColMinimum(), ColMaximum().

Any number of By arguments can be specified, and you can use expressions for the By arguments. By arguments must be used in a column formula, or in the context of ForEachRow(). The first argument can also be a general numeric expression.

Here is an example:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Mean of height by sex", Numeric, Formula( Col Mean(
   :height, :sex ) ) );

dt << New Column( "Minimum of height by sex and age",
   Numeric,
   Formula( Col Minimum( :height, :sex, :age ) )
);

dt << Distribution( Continuous Distribution( Column( :height ) ), By( :sex )
   );

dt << Tabulate(
   Show Control Panel( 0 ),
   Add Table(
      Column Table(
         Analysis Columns( :height ),
         Statistics( Mean, N, Std Dev, Min, Max, N Missing )
      ),
      Row Table( Grouping Columns( :age, :sex ) )
   )
);
```

Encrypt and Decrypt Scripts

To add a basic level of protection to scripts, you can encrypt it so only someone who knows the password can view it; you can also require a password to run it. This is useful in situations when you want to implement controlled sharing of a script.

To encrypt a script:

1. Open the script that you want to encrypt.
2. Select **Edit > Encrypt Script**.
3. Enter a decrypt password so that the user needs a password to view the script.
4. (Optional) Enter a run password to require the user to enter a password before running the encrypted script.

   Note: The passwords must consist of single-byte characters; using a text Input Method Editor (IME) does not work.

5. Click **OK**.
6. If you entered only a decrypt password, click **Yes** to confirm that you do not want to assign a run password.

   The encrypted script opens in a new window. For example:

```
//-e6.0.2
S@FTQ;VGMUTF?J<;LS;B<=IRLXCU=BV;@NS<TW;LR<ZFOP=JJS>NNDA@T<V><DZA>SU@MG;LR<ZFO
   P=JJS>NNDA@T<V><DZA>SU@MG;LR<ZFOP=JJS>NNDA@T<V><DZA>SU@MG;LR<ZFOP=JJS>NNDA
   @T<HNIZ;WDN?RMJ;FR>KYAXTEPPF?;XFJJOP=RQGBIAGXOYNNZ>PLIF>SW>L>ACL<KGP;=QQTC
   EG??U<PUXLV?TRBO?J>QGWTJCFJA@BNHWLVORNNGQYPIKL<IM<>JX>@G?LJ>=;RBODH@PTKK@S
   IUE;IJOR<TUTRMTGSYRSVGOR<XK<F=IWQYE=LVZFP;AUHA?YJLL;EIT?ZJZC;*
```

7. Save the encrypted script.

To encrypt a column formula in a script:

Column formulas created by an encrypted script are visible by default. If you want a formula to be encrypted, follow these steps:

1. Paste just the formula into a new script window.
2. Encrypt the script by clicking **Edit > Encrypt Script**.
3. Provide a password only.
4. Copy the encrypted script.
5. Paste the encrypted script in a new script as a string argument for `JSL Encrypted()` as shown below.

```
dt = Open( "$SAMPLE_DATA\Big Class.jmp" );

dt << New Column( "Encrypted",
   Numeric,
   Continuous,
   Format( "Best", 12 ),
   Formula(
      JSL Encrypted(
         "//-e6.0.2
0,
136,4112eJwFwYkRhCAMAMB+roUQcITwDcoP4fovxN3RiOB+MFqvqXB8ZOhZdFC+dgqclJo0dwrlZ
   bFjgyPlZRuR2hp399ObhpxhQXP8
GPtftOMN6Rwz8pDnjmK1KOqq1bEG+HOxvyBb
,0,"
      )
   )
);
```

To decrypt a JSL script:

1. Open the encrypted script in JMP.
2. Select **Edit > Decrypt Script**.
3. Enter the decrypt password and click **OK**.

   The decrypted script opens in a new window.

To run an encrypted JSL script:

Note: You must know which data table the script runs on before running an encrypted script. If you do not know the name of the data table, you must decrypt the script before running it.

1. Open the encrypted script in JMP.
2. Select **Edit > Run Script**.
3. Enter the run password and click **OK**.

   The script runs:

   – If the script references a data table, you are prompted to open the data table, and then the script runs.

   – If the script requires an empty data table, you must create the table and then run the encrypted script.

Note that entering the run password runs the script, but does not show the script: you must supply the decrypt password to actually view the script.

Encryption and Global Variables

Encryption alone does not hide global variables and their values. A `Show Globals()` command displays them normally. If you want to hide global variables in an encrypted script, you can give them special names.

Any global variable whose name begins with two underscore characters (__) is hidden, and `Show Globals()` displays neither its name nor its value. For example:

```
myvar = 2;
__myvar = 5;
Show Symbols();
   // Globals
   myvar = 2;
   // 2 Global (1 Hidden)
```

This strategy works whether your script is encrypted or not.

Encrypting Scripts in Data Tables

You can also encrypt a script that is saved to a data table using the `JSL Encrypted()` or `Include( Char to Blob() )` functions.

- `JSL Encrypted()` is more straightforward, because it involves one function. You can include comments inside the encrypted script.
- `Include( Char to Blob() )` lets you include comments, but not inside the script.

Note: Column formulas in encrypted scripts are not encrypted. To encrypt them, include them in the `JSL Encrypted()` function.

Follow these steps to encrypt a data table script:

1. Place the script in a script window.

   You cannot directly encrypt a script that is already saved to a data table.
2. In the script window, select **Edit > Encrypt**.
3. Enter a decrypt password.
4. (Optional) Enter a run password to require the user to enter a password before running the script.
5. If you entered only a decrypt password, click **Yes** to confirm that you do not want to assign a run password.

   The encrypted script opens in a new script window.
6. Copy the entire encrypted script.
7. Create a new data table script or open an existing script.

8. In the script portion of the window, type one of the following functions:

   ```
   JSL Encrypted( "" );
   ```

   ```
   Include( Char to Blob( "" ) );
   ```

9. Paste the encrypted script inside the quotation marks in the function.
10. Click **OK**.

Figure 8.5 Example of an Encrypted Data Table Script

```
JSL Encrypted(
    "//-e6.0.2
WE@GSACGT<?CKEG=NG;B<=IRLXCU=BV;@NS<TW;LR<ZFOP=JJS>NNDA@T<V><DZA
>SU@MG;LR<ZFOP=JJS>NNDA@T<V><DZA>SU@MG;LR<ZFOP=JJS>NNDA@T<V><DZA
>SU@MG;LR<ZFOP=JJS>NNDA@T<FFHG=R;GYNBNME>ZBMIB?O;G<>?;UTG@=HVYCC
@MFYCMVQEOTNWASNOEV=Y<TCELCQHW>AFY>;PGSSF<IBIU?H<YLQABWEGSQBMLHW
BJE?HUC=BTP?;?;LVTT>ZQI=IXTJ<P=IHYJIPAC?PGZH;OCCMXPOJHLIRHERSW;V
;@GZDHDNF><YCN@WLBTOFZRAILT?L;NT<OOCLLLWYAIVCSPBMVL???KK=TBZJ@KU
BD>=@;EOUFMUUFLSP<HMZWL?VLHXKK?S@WAFXC>EZODQOHREMNYKRBKAWS=PSXSQ
CRYFOPVJRBNHIVLE?BNPVNM=LO>BAT<?CQF?JNNQAUEHMM;BTU><=WFVOO<;WVKB
;RSBEXTS;QGBEKNDCANPFP;DYDHTSMQNAGVESB;TCWJKGWNBG>JBESVNKOVOBPBE
;?ZPYS;X?=VE=>YFHD;;;*"
)
```

Working with Expressions in an Encrypted Script

An encrypted script might not run the same as the unencrypted script if the script attempts to convert parts of itself from expressions back to character strings. This is necessary to keep the script encrypted. You can avoid this problem by using the expression manipulation functions such as `Arg()`, `Head()`, `Substitute()`, and so on. You can also use `Parse()` to make expressions from character strings that are part of the encrypted script.

For example, here is a script that modifies an expression the wrong way for encryption:

```
ex = Expr(
   aaa = 20;
   Show( aaa );
);
// Wrong, don't do this.
temp = (Parse( Substitute( Char( Name Expr( ex ) ), "aaa", "bbb" ) ));
temp;
```

The script fails when encrypted because `Char()` is unable to break the encryption of the expression stored in `ex`. A second example follows:

```
ex = Expr(
   aaa = 20;
   Show( aaa );
);
// Best choice if it works for you.
temp = Substitute( Name Expr( ex ), Expr( aaa ), Expr( bbb ) );
temp;
```

A third example, though not optimal, follows:

```
ch = "aaa = 20; Show(aaa);";
/* Leaves the ch variable holding unencrypted text
and Temp holds an unencrypted expression.*/
temp = Parse( Substitute( ch, "aaa", "bbb" ) );
temp;
```

You might prefer the third choice over the second choice because it looks easier to understand; be aware that you might want to clear variables like `ch` and `Temp` when you are done with them. You probably encrypted the script to keep other folks from reading it, and they will definitely be able to see the content of variables when you are done.

Additional Numeric Functions

JSL also offers several categories of operations that do not make much sense in the context of the formula editor: matrix operations and numeric derivatives of functions. Algebraic derivatives are also available.

The basic arithmetic operators can also be used with matrix arguments for matrix-wise addition, subtraction, and so on. Matrices also have a few special operators for elementwise multiplication and division, concatenation, and indexing. See "Matrices".

Derivatives

JSL has three internal functions (not all available in the calculator) for taking derivatives.

`Derivative()` takes the first derivative of an expression with respect to names you specify in the second argument. A single name might be entered as this second argument; or multiple values can be specified in a list, in other words, surrounded by braces.

Note: `Derivative()` is also available as an editing command inside the formula editor (calculator), located on the drop-down list in the top center of the formula editor (above the keypad). To use it, highlight a single variable in the expression (to designate which variable the derivative should be taken with respect to), then select the Derivative command from the menu. The whole formula is replaced by its derivative with respect to the highlighted name.

In scripts, the easiest way to use the function is with a single name. In this example, the mathematical notation is shown first followed by the JSL equivalent.

For $f(x) = x^3$, the first derivative is $f'(x)$ or $\frac{d}{dx}x^3 = 3x^2$.

```
result = Derivative( x ^ 3, x );
```

```
Show( result );
  result = 3 * x ^ 2
```

If you want an efficient expression to take the derivative with respect to several variables, then the variables are specified in a list. The result is a list containing a *threaded* version of the original expression, followed by expressions for the derivatives. The expression is *threaded* by inserting assignments to temporary variables of expressions that are needed in several places for the derivatives.

Here is an example involving an expression involving three variables. Listing all three variables returns the first derivatives with respect to each. The result is a list with the original expression and then the derivatives in the order requested. However, note here that JMP creates a temporary variable `T#1` for storing the subexpression `x^2`, and then uses that subexpression subsequently to save calculations.

```
result2 = Derivative( 3 * y * x ^ 2 + z ^ 3, {x, y, z} );
Show( result2 );
  result2 = {3 * y * (T#1 = x ^ 2) + z ^ 3, 6 * x * y, 3 * T#1, 3 * z ^ 2}
```

To take second derivatives, specify the variable as a third argument. Both the second and third arguments must be lists. JMP returns a list with the original expression, the first derivative(s), and then the second derivative(s) in the order requested.

```
second = Derivative( 3 * y * x ^ 2, {x}, {x} );
Show( second );
  second = {3 * y * x ^ 2, 6 * x * y, 6 * y}

second = Derivative( 3 * y * x ^ 2, {y}, {y} );
Show( second );
  second = {3 * y * (T#1 = x ^ 2), 3 * T#1, 0}

second = Derivative( 3 * y * x ^ 2, {y}, {x} );
Show( second );
  second = {3 * y * (T#2 = x ^ 2), 3 * T#2, 6 * x}
```

`Num Deriv()` takes the first numeric derivative of a function with respect to the value of the first argument by calculating the function at that value and at that value plus a small delta (Δ) and dividing the difference by the delta. `Num Deriv2()` computes the second numeric derivative in a similar fashion. These are used internally for nonlinear modeling but are not frequently useful in JSL. Note that these functions do not differentiate using a variable, but only with respect to arguments to a function. In order to differentiate with respect to x, you have to make x one of the immediate arguments, not a symbol buried deep into the expression.

Suppose to differentiate $y = 3x^2$ at the value of $x = 3$. The *incorrect* way would be to submit the following script:

```
x = 3;
n = Num Deriv( 3 * x ^ 2 );
```

The *correct* way is to make x an argument in the function.

```
x = 3;
f = Function( {x}, 3 * x ^ 2 );
n = Num Deriv( f( x ), 1 );
   18.000029999854
```

Consider both the mathematical notation and the JSL equivalent for another example:

For $f(x) = x^2$, it calculates $\frac{d}{dx}x^2 = \frac{(x+\Delta)^2 - x^2}{\Delta}$. At $x_0 = 3$, $\frac{d}{dx}x^2 = 6.00001$.

```
x = 3;
y = Num Deriv( x ^ 2 ); // or equivalently: y = Num Deriv( 3 ^ 2 );
   6.0000099999513
```

And here are a few more examples:

```
x = Num Deriv( Sqrt( 7 ) ); // returns 0.188982168980445
y = Num Deriv( Normal Distribution( 1 ) ); // returns 0.241969514669371
z = Num Deriv2( Normal Distribution( 1 ) ); // returns -0.241969777547979
```

Table 8.7 Derivative functions

Function	Syntax	Explanation
Derivative	Derivative(*expr*, {*name*, ...})	Returns the derivative of the *expr* with respect to *name*. Note that the second argument can be specified in a list with braces { } or simply as a variable if there is only one. Give two lists of names to take second derivatives.
NumDeriv	NumDeriv(*expr*)	Returns the first numeric derivative of the *expr* with respect to the first argument in the expression.
NumDeriv2	NumDeriv2(*expr*)	Returns the second numeric derivative of the *expr* with respect to the first argument in the expression.

Algebraic Manipulations

JSL provides a way of algebraically unwinding an expression (essentially, solving for a variable). It is accomplished through the `Invert Expr()` function.

`Invert Expr(`*expression*`, `*name*`, `*y*`)`

where

- *expression* is the expression to be inverted, or the name of a global containing the expression
- *name* is the name inside expression to unwind the expression around

- *y* is what the expression was originally equal to

For example,

```
Invert Expr( Sqrt( log( x ) ), x, y );
```

is wound around the name x (which should appear in the expression only once), and results in

```
Exp( y ^ 2 )
```

It is performed exactly as you would when doing the algebra by hand.

$y = \text{Sqrt}(\log(x));$

$y^2 = \text{Log}(x);$

$\text{Exp}(y^2) = x;$

`Invert Expr` supports most basic operations that are invertible, and makes assumptions as necessary, such as assuming you are interested only in the positive roots, and that the trigonometric functions are in invertible areas so that the inverse functions are legal.

F, Beta, Chi-square, *t*, Gamma, and Weibull distributions are supported for the first arguments in their Distribution and Quantile functions. If it encounters an expression that it cannot convert, `Invert Expr()` returns `Empty()`.

JSL provides a `Simplify Expr` command that takes a messy, complex formula and tries to simplify it using various algebraic rules. To use it, submit

```
result = Simplify Expr(expr(expression));
```

or

```
result = Simplify Expr(nameExpr(global));
```

For example,

```
Simplify Expr( Expr( 2 * 3 * a + b * (a + 3 - c) - a * b ) );
```

results in

```
6*a + 3*b + -1*b*c
```

`Simplify Expr()` also unwinds nested `If` expressions. For example:

```
r = Simplify Expr( Expr( If( cond1, result1, If( cond2, result2, If( cond3,
  result3, resultElse ) ) ) ) );
```

results in

```
If(cond1, result1, cond2, result2, cond3, result3, resultElse);
```

Maximize and Minimize

The `Maximize()` and `Minimize()` functions find the factor values that optimize an expression. The expression is assumed to be a continuous function of the factor values.

Here is the form of the call:

```
result = Maximize(objectiveExpression,{list of factor names},
  <<option(value))
result = Minimize(objectiveExpression,{list of factor names},
  <<option(value))
```

objectiveExpression is the expression whose value is to be optimized, and can either be the expression itself, or the name of a global containing a stored expression.

{*list of factor names*} is an expression yielding a list of names involved in *objectiveExpression*.

The name can be followed by limits that bound the permitted values, for example name(`lowerBound`,`upperBound`).

If you want to limit the values on one side, make the other side a missing value, for example:

```
{beta} // unconstrained
{beta (0,1)} // constrained between 0 and 1
{beta (.,1)} // upper limit of 1
{beta (0,.)} or {beta (0)} // lower limit of 0
```

Factor values can be either numbers or matrices.

Options available, shown with their default value, include:

```
<< Tolerance(.00000001) // convergence criterion
<< Max Iter( 250) // maximum number of iterations
<< Limits() //
```

Initial values are assumed to be already supplied the factor values before calling the function.

These functions are not expected to find global optima for functions that have multiple local optima; they are useful only for taking an initial value and moving it to either a local or global optimum.

The return value is currently the value of the objective function, if the optimization was successful, or `Empty()` if not.

Least Squares Example

The following example uses `Minimize` to find the least squares estimates of this exponential model, with data taken from the Nonlinear Example/US Population.jmp sample data table.

```
xx = [1790, 1800, 1810, 1820, 1830, 1840, 1850, 1860, 1870, 1880, 1890, 1900,
  1910, 1920, 1930, 1940, 1950, 1960, 1970, 1980, 1990];
yy = [3.929, 5.308, 7.239, 9.638, 12.866, 17.069, 23.191, 31.443, 39.818,
  50.155, 62.947, 75.994, 91.972, 105.71, 122.775, 131.669, 151.325, 179.323,
  203.211, 226.5, 248.7];
b0 = 3.9;
b1 = .022;
sseExpr = Expr(
  Sum( (yy - (b0 * Exp( b1 * (xx - 1790) ))) ^ 2 )
);
sse = Minimize( sseExpr, {b0, b1}, <<Tolerance( .00001 ) );
Show( b0, b1, sse );
  b0 = 13.9991388055261;
  b1 = 0.0147104409355048;
  sse = 1862.14141218875;
```

Identifying Differences Between Strings, Lines, or Sequences

The `Shortest Edit Script()` function compares two strings, lines, or sequences and returns a list of changes or a matrix that describes the changes. You might use `Shortest Edit Script()` to identify differences between two columns, lists, or matrices. The function describes *a* (not *the*) shortest list of instructions to convert sequence A into sequence B. There might be more than one shortest script.

Example of Sequences

```
t1 = New Table( "t1",
  New Column( "Column 1",
     Numeric,
     Continuous,
     Format( "Best", 12 ),
     Set Values( [1, 2, 3, 4, 5, 6, 1, 2, 3, 4, 5, 6, 7] )
  )
);

t2 = New Table( "t2",
  New Column( "Column 1",
     Numeric,
```

```
      Continuous,
      Format( "Best", 12 ),
      Set Values( [2, 3, 4, 5, 2, 3, 4, 5, 2, 3, 4, 5, 6] )
   )
);
EditScript = Shortest Edit Script( // compares column 1 in each table
   sequences(
      N Rows( t1 ),
      N Rows( t2 ),
      Function( {a, b}, // subscripts the columns
         t1:column 1[a] == t2:column 1[b] // in data tables t1 and t2
      )
   )
);
   [-1 1 . 1, // delete 1 item at position 1 in string a
   0 2 1 4, // keep 4 items at position 2 in string a, position 1 in string b
   -1 6 . 2, // delete 2 items at position 6 in string a
   1 . 5 4, // add 4 items at position 5 in string b
   0 8 9 5, // keep 5 items at position 8 in string a, position 9 in string b
   -1 13 . 1] // delete 1 item at position 13 in string a
```

Here are the columns in the matrix:

- Column 1: delete the items (-1), keep the common items (0), or add the items (1)
- Column 2: the position in string a
- Column 3: the position in string b
- Column 4: the number of items the instruction uses

A missing value indicates that the position isn't used in the comparison.

Example of Separators

The following example considers "@" and "$" to be separators.

```
aa = "this is$a test of@shortest$edit script lines$with several words";
bb = "this is a$test of$shortest$edit script lines with several@words";

// @ and $ separators
Shortest Edit Script( lines( aa, bb, separators( "@$" ) ) );
   {{"Remove", "this is$a test of@"}, {"Insert", "this is a$test of$"},
      {"Common",
   "shortest$"}, {"Remove", "edit script lines$with several words"}, {"Insert",
   "edit script lines with several@words"}}
```

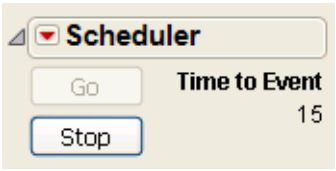
See the *JSL Syntax Reference*.

Scheduling Actions

A `Schedule()` function lets you set up a script to be executed some number of seconds later.

```
Schedule( 15, Print( "hello" ) );
```

Figure 8.6 JMP Scheduler



A Scheduler window shows the time until the next event and has buttons for restarting (Go) or stopping (Stop) the schedule. Its pop-up menu has a command Show Schedule, which echoes the current schedule to the log. For example, if you checked the schedule several times during the "hello" example, you would see something like this:

```
Scheduled at 11.55000000000018 :Print("hello")
Scheduled at 4.716666666666697 :Print("hello")
Scheduled at 3.083333333333485 :Print("hello")
```

The script might also be a name referring to a stored expression. For example, try submitting this script, which calls itself:

```
quickieScript = Expr(
   Show( "Hi there" );
   Schedule( 15, quickieScript );
);
quickieScript;
```

This script should show the string "Hi there" in a log window after 15 seconds, and reschedule itself for another 15 seconds, continuing until the Stop button is clicked.

More typically, in a production setting you might want to set up a schedule like this:

```
FifteenMinuteCheck = Expr(
   Show( "Checking data" );
   Open( "my file", options... );
   distribution( Column( column1 ), capability( spec limits ) );
   Schedule( 15 * 60, FifteenMinuteCheck );
);
FifteenMinuteCheck;
```

`Schedule()` initiates an event queue, but once it has the event queued, JMP proceeds with the next statement in the script. For example, the following has results that might surprise you:

```
Schedule( 3, Print( "one" ) );
```

```
Print( "two" );
   "two"
   "one"
```

If you want the script to wait until the scheduled events are finished before proceeding, one solution would be to use `Wait( )` with a suitable pause. Another is to embed the subsequent actions into the schedule queue. `Schedule()` accepts a series of arguments to queue many events in sequence. Each event is a separate call to the schedule. Each event time is an absolute time relative to "now" (or the instant that **Go** is clicked). Therefore, the following sequence finishes in five seconds, not in twelve:

```
Schedule( 3, Print( "hello" ) );
Schedule( 4, Print( ", world" ) );
Schedule( 5, Print( "--bye" ) );
```

To cancel all events in a schedule queue, use `Clear Schedule`.

```
scheduler[1] << Clear Schedule( );
```

Note: It is not possible to create multiple threads using `Schedule`.

Table 8.8 Schedule commands

Message	Syntax	Explanation
`Schedule`	*sc*=Schedule(*n*, *script*)	Queues an event to run the *script* after *n* seconds.
`Clear Schedule`	*sc*<<Clear Schedule()	Cancels all events in a schedule queue.

Functions that Communicate with Users

`Show()`, `Print()`, and `Write()` put messages in the log window. `Speak()`, `Caption()`, `Beep()`, and `StatusMsg()` provide ways to say something to a viewer. `Mail()` can send an e-mail alert to a process operator.

The JMP scripting language has methods for constructing dialog boxes to ask for data column choices and other types of information. See "Modal Windows".

Tip: To preserve locale-specific numeric formatting in `Show()`, `Print()`, or `Write()` output, include `<<Use Locale(1)`.

Writing to the Log

Show

`Show()` displays the items that you specify in the log. Notice that when you show variables, the resulting message is the variable name, an equal sign, and its current value.

```
X = 1;
A = "Hello, World";
Show( X, A, "foo" );
   x = 1
   a = "Hello, World"
   "foo"
```

Print

`Print()` sends the message that you specify to the log. `Print()` is the same as `Show()` except that it prints only the value of each variable without the variable name and an equal sign.

```
X = 1;
A = "Hello, World";
Print( X, A, "foo" );
   1
   "Hello, World"
   "foo"
```

Write

`Write()` sends the message that you specify to the log. `Write()` is the same as `Print()` except that it suppresses the quotation marks around the text string, and it does not start on a new line unless you include a return character yourself with the `\!N` escape sequence.

```
myText = "Here is a message.";
Write( "Here is a message." );
   Here is a message.

Write( myText || " Do not forget to buy milk." ); // use || to concatenate
Write( "\!NAnd bread." ); // use \!N for return
   Here is a message. Do not forget to buy milk.
   And bread.
```

The sequence `\!N` inserts the line breaking characters that are appropriate for the host environment. For an explanation of the three line breaking escape sequences, see "Double Quotes".

Send Information to the User

Beep

`Beep()` causes the user's computer to make an alert sound.

Speak

`Speak()` reads text aloud. On macOS, `Speak()` has one Boolean option, `Wait()`, to specify whether JMP should wait for speaking to finish before proceeding with the next step. The default is not to wait, and you need to issue `Wait(1)` each time. For example, here is a script certain to drive anybody crazy. With `Wait(1)`, you probably want to interrupt execution before too long. If you change it to `Wait(0)`, the iterations proceed faster than the speaking possibly can and the result sounds strange. On Windows, you can use a `Wait(n)` command to accomplish the same effect.

```
For( i = 99, i > 0, i--,
   Speak(
      Wait( 1 ),
      Char( i ) || " bottles of beer on the wall, " || Char( i ) || " bottles
   of beer; " ||
      "If one of those bottles should happen to fall, " || Char( i - 1 ) || "
   bottles of beer on the wall. "
   ) );
```

A more practical example has JMP announce the time every sixty seconds:

```
script = Expr(
   tod = Mod( Today(), In Days( 1 ) );
   hr = Floor( tod / In Hours( 1 ) );
   min = Floor( Mod( tod, In Hours( 1 ) ) / 60 );
   timeText = "time, " || Char( hr ) || " " || Char( min );
   text = Long Date( Today() ) || ", " || timeText;
   Speak( text );
   Show( text );
   Schedule( 60, script );    // seconds before next script
);
script;
```

You might use a similar technique to have JMP alert an operator that a process has gone out of control.

Caption

`Caption()` brings up a small window with a message to the viewer. Captions are a way to annotate demonstrations without adding superfluous objects to results windows. The first argument is an optional `{h,v}` screen location given in pixels from the upper left; the second argument is the text for the window. If the location argument is omitted, windows appear in the upper left corner.

You can include pauses in the playback by including the named argument `Delayed` and a time in seconds. Such a setting causes that caption *and all subsequent* caption windows to be delayed by that number of seconds, until a different `Delayed` setting is issued in a `Caption` statement. Use `Delayed(0)` to stop delaying altogether.

Specify the font type, font size, text color, or background color with the following arguments:

```
Font( font );
Font Size( size );
Text Color("color");
Back Color("color");
```

The `Spoken` option causes captions to be read aloud by the operating system's speech system (if available). `Spoken` takes a Boolean argument, and the current setting (on or off) remains in effect until switched by another `Caption` statement that includes a `Spoken` setting.

This script turns speaking on and leaves it on until the last caption. In the first caption, the font type, color, and background color is specified. Run the script and notice that the font and color settings apply only to the first caption.

```
Caption(
   {10, 30},
   "A Tour of the JMP Analyses",
   Font( "Arial Black" ),
   Font Size( 16 ),
   Text Color( "blue" ),
   Back Color( "yellow" ),
   Spoken( 1 ),
   Delayed( 5 )
);
Caption( "Open a data table." );
bigClass = Open( "$SAMPLE_DATA/Big Class.jmp" );
Caption( "A data table consists of rows and columns of data." );
Caption( "The rows are numbered and the columns are named." );
Caption( {250, 50}, "The data itself is in the grid on the right" );
Caption(
   {5, 30},
   Spoken( 0 ),
   "A panel along the left side shows columns and other attributes."
);
```

Each new `Caption` hides the previous one. In other words, there is only one caption window available at a time. To close a caption without displaying a new one, use the named argument `Remove`.

```
Caption( remove );
```

StatusMsg

This command sends a message to the status bar.

```
StatusMsg( "string" );
```

Mail

`Mail()` sends an e-mail message to a user. For example, a process control manager might include a test alert script in a control chart to trigger an e-mail warning to her pager:

```
Mail(
   "JaneDoe@company.com",
   "out of control",
   "Process 12A out of control at " || Format( Today(), "d/m/y h:m:s" )
);
```

`Mail()` can also send an attachment with the e-mail. An optional fourth argument specifies the attachment. The attachment is transferred in binary format after its existence on the disk is verified. For example, to attach the Big Class.jmp data table, submit

```
Mail(
   "JohnDoe@company.com",
   "Interesting Data Set",
   "Have a look at this class data.",
   "$SAMPLE_DATA\Big Class.jmp" );
```

Note: On macOS, `Mail()` works on Mountain Lion and later operating systems. On Mountain Lion, you must enter the e-mail address and subject in the e-mail due to operating system limitations. Click the **Send message** button to send the e-mail.

Chapter 9

Data Tables

Working with Data Table Objects

Before you can work with a data table or with objects in a data table, you must first open the data table and assign a *reference* to it. In the *Scripting Guide*, *dt* represents a reference to a data table object.

To manipulate an object in JSL, you send a message to the reference that represents the object, asking the object to perform one of its tasks. *Messages* are commands that can be understood only in context by a particular type of object. For example, messages for data table objects include Save, New Column, Sort, and so on. Most of these messages would not make sense for another object, such as a platform object.

In this chapter, you can learn the following:

- How to assign a reference to a data table
- How to send messages to the reference, resulting in specific actions
- About the different types of messages that data table objects can understand

Most of this chapter focuses on the different types of messages that you can send to data table objects, such as data tables, columns, rows, and values.

Tip: This chapter contains most, but not all of the JSL commands that you can use with data tables. For an exhaustive list, see the JMP Scripting Index. Select **Help > Scripting Index**. Select **Objects** from the menu, and then select **Data Table**.

Contents

Get Started with Data Tables . . . 327
Basic Data Table Scripting . . . 330
Open a Data Table . . . 330
Create a New Data Table . . . 332
Import Data . . . 333
Set the Current Data Table . . . 357
Name a Data Table . . . 358
Save a Data Table . . . 358
Hide a Data Table . . . 359
Print a Data Table . . . 360
Resize a Data Table . . . 360
Close a Data Table . . . 360
Set and Get a Data Table . . . 361
Perform Actions on All Open Data Tables . . . 361
Create Journals . . . 362
Create Excel Workbooks . . . 363
Prevent Changes to a Data Table . . . 364
Advanced Data Table Scripting . . . 365
Select Columns in a Data Table . . . 365
Store Summary Statistics in Global Variables . . . 366
Create a Table of Summary Statistics . . . 369
Subset a Data Table . . . 371
Add a Data Filter . . . 372
Sort a Data Table . . . 379
Stack Values in a Data Table . . . 379
Split Values in a Stacked Data Table . . . 380
Transpose a Data Table . . . 382
Vertically Concatenate Data Tables . . . 383
Horizontally Concatenate Data Tables . . . 383
Virtually Join Data Tables . . . 384
Replace Data in Data Tables . . . 387
Create a Table Using Tabulate . . . 389
Copy Data Tables . . . 390
Find Missing Data Patterns . . . 390
Compare Data Tables . . . 391
Create a Summary Table . . . 391
Subscribe to a Data Table . . . 392
Move Data Between Matrices and Data Tables . . . 394
Add Hyperlinks to Text Columns . . . 394

Columns 396
Send Messages to Data Column Objects 397
Create Columns 398
Create a New Column by Text Matching 400
Add Several Columns at Once 401
Group Columns 402
Use Set Selected to Select Columns 404
Rearrange and Move Columns 405
Recode Columns 406
Add a Column Switcher 407
Compress Selected Columns 408
Create Custom Functions, Transforms, and Formats 408
Custom Transform Example 411
Delete Columns 415
Get Column Names 415
Column Attributes 416
Column Properties 423
Specify Engineering SI Units 430
Rows 431
Add Rows 431
Delete Rows 432
Select Rows 433
Find Rows 436
Move Rows 437
Assign Colors and Markers to Rows 437
Color Cells 438
Hide, Exclude, and Label Rows 439
Iterate on Rows in a Table 440
Prevent Editing Rows 443
Row States 443
Accessing Data Values 460
Set or Get Values by Column Name 461
Additional Ways to Access Data Values 462
Add Metadata to a Data Table 463
Data Table Variables 463
Table Scripts 463
Formulas 468
Delete Metadata 469
Calculations 469
Pre-Evaluated Statistics 469

Calculator Formulas 471

Get Started with Data Tables

Tip: Keep the log window open to see the output of each script that you run. Select **View > Log** to open the Log window. See "Work with the Log".

The typical way to work with values in a data table is to follow these steps:

1. Set up the data table whose values you want to access as the current data table. Or, if you already have a data table reference, you can simply use that reference.
2. Specify the row or rows whose values you want to access and specify the column name that contains the values that you want to access.

The following example opens the Big Class.jmp sample data table (making it the current data table), and then specifies row 2 in the weight column. A value of 123 is returned in the log, which is the weight for Louise in row 2.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt:weight[2];
   123
```

If the data table that you want to work with is already open, proceed using one of the following examples:

```
dt = Data Table( "My Table" ); // the open table named My Table
dt = Current Data Table(); // the table in the active window
dt = Data Table( 3 ); // the third open table
```

Once you have an open data table with a reference, you can send it messages using either the << operator or the Send function. The following example illustrates both methods:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Save();
Send( dt, Save() );
```

Note the following about messages:

- You can send messages to any data table object (a data table, column, row, and so on).
- Messages sent to data table objects always have the same pattern:

  ```
  Reference << Message( Arguments );
  ```

- Usually, creating a reference and sending many messages to the reference is the easiest approach. However, you can also use the direct route instead of using a reference if you have only one message to send. The following example saves the current data table:

  ```
  Current Data Table() << Save();
  ```

- You can stack up a series of messages in one statement. Commands are evaluated from left to right, and each returns a reference to the affected object. The following example creates a new data table called My Table, adds two columns to it, and prompts you to save it:

  ```
  dt = New Table( "My Table" );
  dt << New Column( "Column 1" ) << New Column( "Column 2" ) << Save( "" );
  ```

  In this example, each message returns a reference to the data table (`dt`).
- If you specify too few arguments for a message, JMP presents a window to get the necessary information from you. JMP often presents windows when your script is incomplete, a behavior that you can use to your advantage when writing scripts that need to query users for their choices.
- Some messages come in pairs; one to "set" or assign each attribute, and one to "get" or query the current setting of each attribute.

Why are Some Commands Sent to Objects and Others Used Directly?

`New Table` and `Open` are commands to create objects that do not exist yet. Once created, you send them messages requesting changes. To close such objects, you must close the objects' *container*, because the objects cannot delete themselves.

The following example creates a table and assigns the data table reference to the *dt* variable. New Column messages are sent to the data table reference. To delete one of those columns, the Delete Columns message is sent to the data table reference, not to the column itself.

```
dt = New Table( "Airline Data" );
dt << New Column( "Date" );
dt << New Column( "Airline" );
dt << Delete Columns( "Date" );
```

How Can I See All of the Messages that Can be Sent to a Data Table Object?

To see all of the messages that can be sent to a data table object, see the Scripting Index:

1. Select **Help > Scripting Index**.
2. Select **All Categories** from the list.
3. Select **Data Table** in the list.

You can also use the `Show Properties()` command. This is a good approach if you want to print or copy the list of messages. The `Show Properties()` command lists the messages that you can send to a data table in the Log window. `Show Properties()` is a command that takes any scriptable object (such as a data table or column) as its argument. To show properties for a data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Show Properties( dt );
```

The resulting message list is hierarchical.

`[Subtable]` refers to a set of JMP menu commands. For example, the Tables subtable represents the JMP Tables menu, and the indented messages in this subtable correspond to commands in the Tables menu. Many messages include a short description.

Messages labeled as `[Action]` all result in some action being taken. Some messages are available for `[Scripting Only]`. Some messages take a Boolean argument, labeled as `[Boolean]`.

The platforms in the Analyze and Graph menus appear in the properties list because you can send a platform name as a message to a data table. This launches the platform for the data table through the usual launch window. For more information about writing platform-specific scripts, see "Scripting Platforms".

```
dt << Distribution( Y( height ) );
```

Note: In addition to data tables, `Show Properties` also works with platforms and display boxes.

The JMP Scripting Index provides more information about these properties. You can also run and modify sample scripts from the Scripting Index. Select **Help > Scripting Index** and search for the property in the Objects list.

Basic Data Table Scripting

Before you can work with data table objects, you must open or create a data table, and assign a reference to the data table.

- "Open a Data Table"
- "Create a New Data Table"
- "Import Data"
- "Set the Current Data Table"
- "Name a Data Table"
- "Save a Data Table"
- "Hide a Data Table"
- "Print a Data Table"
- "Resize a Data Table"
- "Close a Data Table"
- "Set and Get a Data Table"
- "Perform Actions on All Open Data Tables"
- "Create Journals"
- "Create Excel Workbooks"
- "Prevent Changes to a Data Table"

Open a Data Table

Use the `Open()` function to open a data table.

- To simply open a data table without returning a reference to it:

```
Open( "$SAMPLE_DATA/Big Class.jmp" ); // open the data table
```

- To open a data table and retain a reference to it:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" ); // open and store a reference
```

The path to the data table can be a quoted literal path (absolute or relative) or an unquoted expression that yields a pathname. Relative paths are interpreted relative to the location of the .jsl file (for a saved script). For unsaved scripts, the path is relative to your primary partition (Windows) or your <*username*>/Documents folder (macOS).

```
Open( "../My Data/Repairs.jmp" ); // relative path on Windows and macOS
Open( "::My Data:Repairs.jmp" ); // relative path on macOS
Open( "C:/My Data/Repairs.jmp" ); // absolute path
```

JMP provides shortcuts (*path variables*) to directories or files. Instead of entering the entire path to the directory or file, you include a path variable in the `Open()` expression. For example, JMP sample scripts typically use the $SAMPLE_DATA path variable to open files in the Samples/Data folder. For more information about path variables, see "Path Variables".

If you do not want to specify the entire path every time you open a data table, define a filepath string and concatenate the path with the filename:

```
myPath = "C:/My Data/Store25/Maintenance/Expenses/";
Open( myPath || "Repairs.jmp" );
```

Upon opening a data table, JMP stores the data table in memory. This means that if a script attempts to open a data table that is already open, the opened version appears. JMP does not read the version that is saved on your computer.

Test for an Open Data Table

In a script that depends on an opened data table, you can test to see whether the table is open using `Is Empty()` or `Is Scriptable()`. In the following example, the script performs a Bivariate analysis on Big Class.jmp and then closes the data table. Before proceeding to the Oneway analysis later in the script, `Is Scriptable()` tests for the open data table. If `1` (true) is returned, the table opens, and the script continues.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
obj = dt << Bivariate( Y( :height ), X( :age ), Fit Line );
Close( dt );
If( Not( Is Scriptable( dt ) ),
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" ),
);
obj = dt << Oneway( Y( :height ), X( :age ), Means( 1 ), Mean Diamonds( 1 ) );
```

You might want to see if the data table is open and then close it to discard any changes before running another script on the data table. To test for and close Big Class.jmp, run the following script:

```
Try( Close( Data Table( "Big Class" ) ) );
```

Prompt Users to Open a Data Table

You can use an `If()` expression to prompt a user to open a data table, if no open data table is found. And if they do not select a table, the script should end. The following script shows an example:

```
dt = Current Data Table();
If( Is Empty( dt ),
   Try( dt = Open(), Throw( "No data table found" ) )
);
```

The user is prompted to open a data table. If the user clicks Cancel instead of opening a data table, an error appears in the log.

Show Only Specific Columns

To open a data table and show only a specific set of columns, identify those columns in the `Open()` expression. This is particularly helpful with a large data table in which only a few columns are necessary.

The following example opens Big Class.jmp and includes only the age, height, and weight columns.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp", Select Columns( "age", "height",
  "weight" ) );
```

You can also specify the columns to leave out of the open data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp", Ignore Columns( "name", "sex" ) );
```

Create a New Data Table

You can start a new data table, or start a new data table and store its reference in a global variable. In either case, specify the table name as an argument.

```
New Table( "My Table" );
```

or

```
dt = New Table( "My Table" );
```

The following sections describe the optional arguments for the `New Table()` function.

Private, Invisible, and Visible

The optional `visibility` argument specifies whether the new data table is shown. `visibility( "invisible" )` hides the data table from view; it appears only in the JMP Home Window and the Window menu. `visibility( "private" )` avoids opening the data table. `visibility( "visible" )` shows the data table. `"visible"` is the default value.

```
New Table( "My Table", visibility( "private" ) );
```

A private data table enables you to avoid exposing the table to general use. Sometimes a script creates a private data table to store private data generated by a process. It is not for saving memory.

When you do not need a view of the data, or when you do not need to interact with the data, use an invisible data table.

Notes:

- The advantages of using private data tables is based on the size of the data in the table. If the data table has only five columns and 20 rows, the savings for a private data table is substantial. However, for a table with 100 columns and 1,000,000 rows, the overhead of the actual window is small in comparison to the actual data.
- Private tables are not counted when `N Table()` returns the number of open data tables.
- Private tables cannot be made current with `Current Data Table()`.

Actions

An optional argument that can define the new table. For example, the following script creates a new data table named *Little Class*. It adds three rows and two named columns as well as entering values for each cell in the table:

```
dt = New Table( "Little Class",
   Add Rows( 3 ),
   New Column( "name",
      Character,
      "Nominal",
      Set Values( {"KATIE", "LOUISE", "JANE"} )
   ),
   New Column( "height",
      "Continuous",
      Set Values( [59, 61, 55] )
   )
);
```

Import Data

Upon importing, JMP converts file types such as text files and Microsoft Excel files to the data table format. On Windows, JMP relies on filename extensions to identify the type of file and how to interpret its contents. Here are some examples of filename extensions identified by JMP:

```
Open( "$SAMPLE_IMPORT_DATA/Bigclass.xlsx" ); // Microsoft Excel file
Open( "$SAMPLE_IMPORT_DATA/Bigclass.txt" ); // text file
Open( "$SAMPLE_IMPORT_DATA/Carpoll.xpt" ); // SAS transport file
```

On macOS, JMP relies on the macOS Type and Creator codes (if present) and secondarily on three-letter filename extensions. Be sure to add the file extension before importing the file on macOS.

- Type and Creator codes are invisible data that enable the Finder to display a file with the correct icon corresponding to the application that created it.

- Files with generic icons should have the filename extensions (as with files created on other operating system).

```
Open( "$SAMPLE_IMPORT_DATA/Bigclass.txt", text );
```

Additional supported formats include .csv, .jsl, .dat, .tsv, and .jrn.

For more detailed information about import options, see the *JSL Syntax Reference*.

Import Multiple Files

When you save a script from the Multiple File Import (MFI) window into a script window, you get a compact form of the script that includes all of the parameters, default or otherwise, as messages. You could pare down the script as follows:

```
mfi = Multiple File Import(
   <<Set Folder( "$SAMPLE_IMPORT_DATA" ),
   <<Set Name Filter( "UN*.csv" ), // import files with this name
   <<Set Name Enable( 1 ),
   <<Set Add File Name Column( 1 ) // display the file names in a column
)
<<Import Data();
```

Note the slight distinction between the following expressions:

```
mfi = Multiple File Import();
```

`mfi` holds the Multiple File Import object.

```
dtlist = Multiple File Import(…) << Import Data;
```

`dtlist` holds the list of files produced by sending the `Import Data` message to the MFI object. The MFI object is gone because it was not stored in a variable or window. You can keep the MFI object by sending `Create Window` to it or storing the object in a variable, such as `mfi`, and by sending messages such as `Import Data` to that variable. Sending `Import Data` to an MFI object always returns a list, which could affect tasks such as creating reports.

`Create Window` can be useful for debugging. The window is visible *and* the MFI object is available. Run the following lines one at a time to see the results:

```
mfi = Multiple File Import();
mfi <<Create Window;
mfi <<Set Folder( "$DESKTOP" );
```

Here's an example of using a callback function to get a list of imported files after the user clicks Import on the Multiple File Import window.

```
Delete Directory( "$TEMP/test" );
Create Directory( "$TEMP/test" );
Save Text File( "$TEMP/test/argylesocks1.txt", "a1\!n1" ); // make some
Save Text File( "$TEMP/test/argylesocks2.txt", "a2\!n1" ); // data
```

```
global:importedFiles={};

mfi = Multiple File Import( // make an MFI object
     <<Set Folder( "$TEMP/test/" ),
     <<Set Name Filter( "argylesocks?.txt;" ),
     <<Set Name Enable( 1 ),
     <<Set Add File Name Column( 1 ),
     <<Set Import Callback(
           Function( {this, files}, // the callback function
                global:importedFiles = files;
/* optional: omit Close Window to keep the Multiple File Import window open so
   that the user can modify the settings. */
                win << Close Window;
           )
     )
);
win = mfi << Create Window; // open the window
While(
   Try(
      win << Parent;
      1;
   ,
      0
   ),
   Wait( 1 )
); // wait for the dialog to close
Show( global:importedFiles ); // proceed
```

Notes:

- See *Using JMP* for more information about MFI.
- See "Import Images from a Directory" for more information about using MFI to import images from a web page into a data table.

Import a PDF File

In a PDF file, data is laid out in tables. JMP auto-detects these tables to determine where the rows and columns begin and end.

You write scripts that specify the dimensions of each table on each page, the number of column headers, whether the tables should be concatenated or individual, and more.

Note: A PDF file consists of tables that contain data. In the PDF preview of the PDF Import Wizard, the tables are outlined and selected on each page.

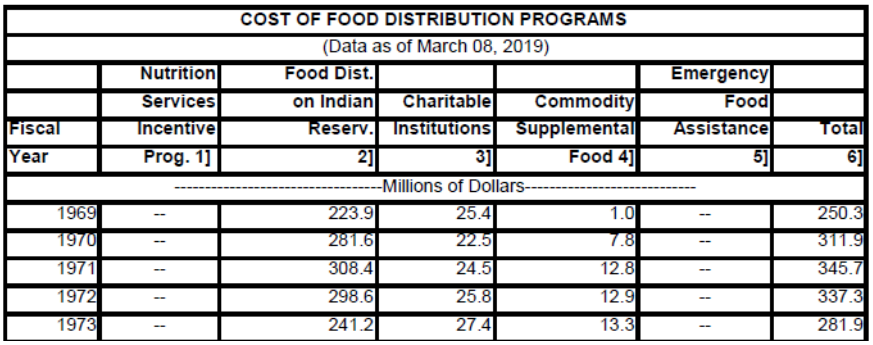
The following options are available for importing a PDF file:

```
Open( "PDF file path" ),
   Tables(
      Table( "data table name",
      Add Rows( Header Rows( number ),
      (Rect( left, right, top, bottom ),
      Page( number ), Row Borders(), Column Borders())) |
      Tables( Combine( All | Matching Headers | None),
         Minimum Rows( number ), Minimum Columns(#))))
   | PDF wizard );
```

Concatenate Tables Into One Data Table

This section shows how to import and concatenate the tables into one data table. The script that you create appears in a Source variable in the final data table, so you can run it again anytime.For this example, Figure 9.1 shows the sample import data file called Food Distribution.pdf, which contains one table per page.

Figure 9.1 Initial PDF File



COST OF FOOD DISTRIBUTION PROGRAMS

(Data as of March 08, 2019)

Fiscal Year	Nutrition Services Incentive Prog. 1]	Food Dist. on Indian Reserv. 2]	Charitable Institutions 3]	Commodity Supplemental Food 4]	Emergency Food Assistance 5]	Total 6]
----------Millions of Dollars----------						
1969	--	223.9	25.4	1.0	--	250.3
1970	--	281.6	22.5	7.8	--	311.9
1971	--	308.4	24.5	12.8	--	345.7
1972	--	298.6	25.8	12.9	--	337.3
1973	--	241.2	27.4	13.3	--	281.9

```
Open(
   "$SAMPLE_IMPORT_DATA/Food Distribution.pdf",
   PDF Tables(
      Table( // import tables into one table
         Table Name( "Food Distribution" ), // data table name
         Add Rows( // rows on page 1
            Header Rows( 7 ), // number of header rows
            Page( 1 ),
            Rect( 0.6922, 0.7155, 7.2842, 10.1124 ) // size of the table
         ), // rows and specifications on page 2
         Add Rows( Page( 2 ), Rect( 0.6922, 0.7156, 7.2842, 10.0708 ) )
      )
   )
);
```

Tip: Import a PDF file in the PDF Import Wizard and copy the script from the Source variable into a standalone script. This is a quick way to get the settings and avoid having to guess at the `Rect` settings.

Save as Individual Data Tables

The following syntax shows how to save tables in a PDF file into individual data tables. Each resulting data table contains a Source script like the one below.

```
Open(
   "PDF file path",
   PDF Tables(
      Table(
         Table Name( "data table name" ),
         Add Rows( Page( number ), Rect( left, right, top, bottom ) )
      )
   )
);
```

Concatenate Data Based on Column Headers

The following example shows how to concatenate data when column names on each page match.

Figure 9.2 Data with Matching Column Headers on each Page

Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa

```
pdftable = Open(
   "$SAMPLE_IMPORT_DATA/Iris.pdf",
   PDF All Tables( Combine( Matching Headers ) )
);
```

Figure 9.3 Final Data Table

Iris Page(1) T...
Source

Columns (7/0)
Page
Table
Sepal length
Sepal width
Petal length
Petal width
Species

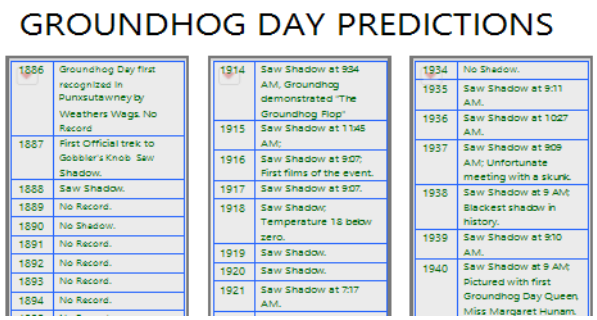
Rows
All rows 150
Selected 0
Excluded 0
Hidden 0
Labelled 0

	Page	Table	Sepal length	Sepal width	Petal length	Petal width	Species
1	1	1	5.1	3.5	1.4	0.2	setosa
2	1	1	4.9	3	1.4	0.2	setosa
3	1	1	4.7	3.2	1.3	0.2	setosa
4	1	1	4.6	3.1	1.5	0.2	setosa
5	1	1	5	3.6	1.4	0.2	setosa
6	1	1	5.4	3.9	1.7	0.4	setosa
7	1	1	4.6	3.4	1.4	0.3	setosa
8	1	1	5	3.4	1.5	0.2	setosa
9	1	1	4.4	2.9	1.4	0.2	setosa
10	1	1	4.9	3.1	1.5	0.1	setosa
11	1	1	5.4	3.7	1.5	0.2	setosa
12	1	1	4.8	3.4	1.6	0.2	setosa
13	1	1	4.8	3	1.4	0.1	setosa
14	1	1	4.3	3	1.1	0.1	setosa
15	1	1	5.8	4	1.2	0.2	setosa
16	1	1	5.7	4.4	1.5	0.4	setosa
17	1	1	5.4	3.9	1.3	0.4	setosa
18	1	1	5.1	3.5	1.4	0.3	setosa

Import PDF Pages that Contain Multiple Tables

A PDF file might contain data divided into multiple columns on each page. Figure 9.4 shows the sample import data PDF file called Groundhog Day Predictions.pdf, which contains multiple tables per page.

Figure 9.4 Multiple Columns on a Page


GROUNDHOG DAY PREDICTIONS

1886	Groundhog Day first recognized in Punxsutawney by Weathers Wags. No Record
1887	First Official trek to Gobbler's Knob. Saw Shadow.
1888	Saw Shadow.
1889	No Record.
1890	No Shadow.
1891	No Record.
1892	No Record.
1893	No Record.
1894	No Record.

1914	Saw Shadow at 9:34 AM. Groundhog demonstrated "The Groundhog Flop"
1915	Saw Shadow at 11:45 AM;
1916	Saw Shadow at 9:07; First films of the event.
1917	Saw Shadow at 9:07.
1918	Saw Shadow; Temperature 18 below zero.
1919	Saw Shadow.
1920	Saw Shadow.
1921	Saw Shadow at 7:17 AM.

1934	No Shadow.
1935	Saw Shadow at 9:11 A.M.
1936	Saw Shadow at 10:27 A.M.
1937	Saw Shadow at 9:09 AM; Unfortunate meeting with a skunk.
1938	Saw Shadow at 9 AM; Blackest shadow in history.
1939	Saw Shadow at 9:10 A.M.
1940	Saw Shadow at 9 AM; Pictured with first Groundhog Day Queen, Miss Margaret Hunam.

The following script imports all columns into a single data table.

```
Open(
  "$SAMPLE_IMPORT_DATA/Groundhog Day Predictions.pdf",
  PDF Tables(
    Table(
      Table Name( "Groundhog Day Predictions" ),
      // specifications for page 1
      Add Rows( Page( 1 ), Rect( 0.9726, 1.4325, 3.0351, 9.9974 ) ),
      Add Rows( Page( 1 ), Rect( 3.3063, 1.4325, 5.3688, 9.9608 ) ),
```

```
            Add Rows( Page( 1 ), Rect( 5.6399, 1.4325, 7.7025, 8.7925 ) ),
            // specifications for page 2
            Add Rows( Page( 2 ), Rect( 0.9726, 1.4325, 3.0351, 9.9441 ) ),
            Add Rows( Page( 2 ), Rect( 3.3063, 1.4325, 5.3688, 9.9274 ) ),
            Add Rows( Page( 2 ), Rect( 5.6399, 1.4325, 7.7025, 8.8225 ) ),
            // specifications for page 3
            Add Rows( Page( 3 ), Rect( 0.9726, 1.4325, 3.0351, 7.6404 ) ),
            Add Rows( Page( 3 ), Rect( 3.3063, 1.4325, 5.3688, 7.1254 ) ),
            Add Rows( Page( 3 ), Rect( 5.6399, 1.4325, 7.7025, 7.4721 ) )
        )
    )
);
```

Figure 9.5 Final Data Table

Groundhog Day P...
Source
Columns (4/0)
Page
Table
Column 3
Column 4
Rows
All rows 132

	Page	Table	Column 3	Column 4
1	1	1	1886	Groundhog Day first
2	1	1	1887	First Official trek to
3	1	1	1888	Saw Shadow.
4	1	1	1889	No Record.
5	1	1	1890	No Shadow.
6	1	1	1891	No Record.
7	1	1	1892	No Record.
8	1	1	1893	No Record.
9	1	1	1894	No Record.
10	1	1	1895	No Record.
11	1	1	1896	No Record.
12	1	1	1897	No Record.
13	1	1	1898	Saw Shadow.
14	1	1	1899	No Record. Groundh...
15	1	1	1900	Saw Shadow.

Export Data to a Text String

You can export data from a PDF file into a text string. You might want to do this to extract a pattern from the data.

Here is a basic example that imports all text from the Iris.pdf sample import data.

```
Open( "$SAMPLE_IMPORT_DATA/Iris.pdf", PDF text );
```

The following script finds the word “Figure” followed by a space, a digit, a period, and another digit and then prints the data to the log.

```
txt = Open( "$JMP_HOME/Documentation/JMP Documentation Library.pdf",
  PDFText() );
figures = {};
For( page = 1, page < N Items( txt ), page++,
  // search for the string
  next = Regex Match( txt[page], "Figure\s+\d+\.\d+", "" );
  While( N Items( next ) > 0,
```

```
        figures = Insert( figures, next );
        // extract the string
        next = Regex Match( txt[page], "Figure\s+\d+\.\d+", "" );
    );
);
Print( figures );
    {"Figure 2.1", "Figure 2.2", "Figure 2.3", "Figure 2.4", "Figure 2.4",
        "Figure 2.4", "Figure 2.4", "Figure 2.5", "Figure 2.6", "Figure 2.7",
        "Figure 2.8", "Figure 2.9", "Figure 2.10", "Figure 2.10", ...}
```

Import Data from a Text File

The Import Settings in the Text Data Files preferences determine how text files are imported. For example, column names begin on line one and data begin on line two by default. To use different settings, specify the import settings as `Open()` options in your script.

The default Import Settings and your custom import settings are saved in the data table Source script, so you can reimport the data using the same settings. However, the default Import Settings are optional in the script.

The following `Open()` options are available:

```
/* Charset options: "Best Guess", "utf-8", "utf-16", "us-ascii",
   "windows-1252", "x-max-roman", "x-mac-japanese", "shift-jis", "euc-jp",
   "utf-16be", "gb2312" */
CharSet("option")
Number of Columns(Number)

/* colType is Character|Numeric
colWidth is an integer specifying the width of the column */
Columns(colName=colType(colWidth),... )
Treat Empty Columns as Numeric(Boolean)
Scan Whole File(Boolean)
End Of Field(Tab|Space|Comma|Semicolon|Other|None)
EOF Other("char")
End Of Line(CRLF|CR|LF|Semicolon|Other)
EOL Other("Char")
Strip Quotes|Strip Enclosing Quotes(Boolean)
Labels | Table Contains Column Headers(Boolean)
Year Rule | Two digit year rule ("Decade Start")
Column Names Start | Column Names are on line(Number)
Data Starts | Data Starts on Line(Number)
Lines to Read(Number)
Use Apostrophe as Quotation Mark
CompressNumericColumns(Boolean)
CompressCharacterColumns(Boolean)
CompressAllowListCheck(Boolean)
```

The following script opens a text file of comma-delimited text, which includes no column names. The script defines the column names and the column widths.

```
Open(
   "$SAMPLE_IMPORT_DATA/EOF_comma.txt",
   End of Field( comma ),
   Labels( 0 ),
   Columns(
      name = Character( 12 ),
      age = Numeric( 5 ),
      sex = Character( 5 ),
      height = Numeric( 3 ),
      weight = Numeric( 3 )
   )
);
```

Here is an example of opening a text file in which the field separator is a space and the text file does not contain column headings:

```
Open(
   "$SAMPLE_IMPORT_DATA/EOF_space.txt",
   Labels( 0 ),
   End of Field( Space )
);
```

In the preceding example of a text file with space field separators, JMP can select the best file format so that you do not have to specify the labels or separator:

```
Open(
   "$SAMPLE_IMPORT_DATA/EOF_space.txt",
   Import Settings( Guess File Format() )
);
```

To set the import options interactively, include the `Text Wizard` argument. A preview of the text file opens in the text import window.

```
Open( "$SAMPLE_IMPORT_DATA/EOF_space.txt", "Text Wizard" );
```

The following sections describe each argument in more detail. For more information about import options, see the *JSL Syntax Reference*.

Number of Columns

Specifies the total number of columns in the source file. This option is important if data is not clearly delimited.

Columns

Identifies column names, column types, and column widths with a `Columns` argument as shown in the preceding examples.

If you specify settings for a column other than the first column in the file, you must also specify settings for all the columns that precede it. Suppose that you want to open a text file that has four columns (name, sex, and age, and ID, in that order). age is a numeric column, and the width should be 5. You must also set the name and sex column types and widths, and list them in the same order:

```
Columns(
   name = Character( 15 ),
   sex = Character( 5 ),
   age = Numeric( 5 )
);
```

You are not required to provide settings for any columns that follow the one that you want to set (in this example, ID).

After the data is imported, you use the modeling type for a column. See "Set or Get Data and Modeling Types".

Note: Most of the following arguments are defined in the JMP preferences. To override the preference, include the corresponding argument described below in your import scripts.

Treat Empty Columns as Numeric

Imports columns of missing data as numeric rather than character data. A period, Unicode dot, NaN, or a blank string are possible missing value indicators. This is a Boolean value. The default value is false.

Scan Whole File

Specifies how long JMP scans the file to determine data types for the columns. This is a Boolean value. The default value is true; the entire file is scanned until the data type is determined. To import large files, consider setting the value to false, which scans the file for five seconds.

Strip Quotes | Strip Enclosing Quotes

Specifies whether to include or remove the double quotation marks (") that surround string values. This is a Boolean value. The default value is `true`.

For example, suppose that the field delimiter is a space:

- *John Doe* is interpreted as two separate strings (*John* and *Doe)*.
- "*John Doe*" is interpreted as a single string. Most programs (including JMP) read a quotation mark and ignore other field delimiters until the second quotation occurs.
- If you include `Strip Quotes(1),` "*John Doe*" is interpreted as *John Doe* (one string without quotation marks).

Note that many word processors have a “smart quotation marks” feature that automatically converts double quotation marks (") into left and right curled quotation marks (“ ”). Smart quotation marks are interpreted literally as characters when the text file is imported, even when JMP strips double quotation marks.

End of Line(CR | LF | CRLF | Semicolon | Other)

Specifies the character or characters that separate rows:

- CR for carriage returns (typical for text files created on macOS up to version 9)
- LF for linefeeds (typical for UNIX and Mac OS X text files)
- CRLF for both a carriage return followed by a linefeed (typical for Windows text files).

All three characters are line delimiters by default.

Use the `Other` option to use an additional character for the row separator, which you must specify in the `EOLOther` argument. JMP interprets either this character or the default character as a row separator.

End of Field(Tab | Space | Spaces | Comma | Semicolon | Other | None)

Specifies the character or characters used to separate fields.

Notes:

- The default field delimiter is `Tab`.
- Use the `Other` option to use a different character for the field separator, which you must specify in the `EOFOther` argument.
- The `Space` option uses a single space as a delimiter.
- The `Spaces` option uses two or more spaces.

EOFOther, EOLOther

Specifies the character or characters used to separate fields or rows. For example, `EOLOther("*")` indicates that an asterisk separates rows in the text file.

Labels | Table Contains Column Headers

Indicates whether the first line of the text file contains column names. This is a Boolean value. The default value is true.

Year Rule | Two Digit Year Rule

Specifies how to import two-digit year values. If the earliest date is 1979, specify "1970". If the earliest date is 2001, specify "20xx".

Column Names Start | Column Names Are on Line

Specifies the starting line for column names. The following example specifies that the column names in the text file start on line three.

```
Open(
    "$SAMPLE_IMPORT_DATA/Animals_line3.txt",
    Columns(
        Column( "species", Character, "Nominal" ),
        Column( "subject", Numeric, "Continuous", Format( "Best", 10 ) ),
        Column( "miles", Numeric, "Continuous", Format( "Best", 10 ) ),
        Column( "season", Character, "Nominal" )
    ),
   Column Names Start( 3 )
);
```

Data Starts | Data Starts on Line

Specifies the starting line for data.

The following example specifies that the data in the text file start on line five.

```
Open(
   "$SAMPLE_IMPORT_DATA/Bigclass_L.txt",
   Columns(
      Column( "name", Character, "Nominal" ),
      Column( "age", Numeric, "Continuous", Format( "Best", 10 ) ),
      Column( "sex", Character, "Nominal" ),
      Column( "height", Numeric, "Continuous", Format( "Best", 10 ) ),
      Column( "weight", Numeric, "Continuous", Format( "Best", 10 ) )
   ),
   Data Starts( 5 )
);
```

Lines to Read

Specifies the number of lines to include in the data table. JMP starts counting after column names are read.

The following example includes only the first 10 lines in the data table.

```
Open(
    "$SAMPLE_IMPORT_DATA/Bigclass_L.txt",
    Columns(
        Column( "name", Character, "Nominal"),
        Column( "age", Numeric, "Continuous", Format( "Best", 10 ) ),
        Column( "sex", Character, "Nominal" ),
        Column( "height", Numeric, "Continuous", Format( "Best", 10 ) ),
        Column( "weight", Numeric, "Continuous", Format( "Best", 10 ) )
```

```
    ),
  Lines To Read( 10 )
);
```

Use Apostrophe as Quotation Mark

For data that are enclosed in apostrophes, this option treats apostrophes as quotation marks and omits them. For example, '2010' is imported as 2. This is a Boolean value. The default value is false.

This option is not recommended unless your data comes from a nonstandard source that places apostrophes around data fields rather than quotation marks.

Set Text Import Preferences

If you want to set preferences for importing text, it can be helpful to first see a list of all preferences. To do so, use the `Show Preferences (All)` function.

Then, copy and paste only the section you want into a Preferences function. For example, to specify import settings, write the following expression:

```
Preferences(
  Import Settings(
    End Of Field( Tab, Spaces, Comma )
  )
);
```

Import Data from a Microsoft Excel File

When you open a Microsoft Excel workbook in JMP, the file is converted to a data table. JMP supports .xls, .xlsm, and .xlsx formats. See *Using JMP* for more information about Microsoft Excel support.

Excel Preferences

In the JMP preferences, settings in the Third Party Data group can help determine how worksheets are imported:

Excel Open Method Specifies how a Microsoft Excel file should be opened by default, when using a non-specific open statement.

Use Excel Wizard Opens the Excel Import Wizard to import the file. This is the default setting.

Open All Sheets Opens all worksheets in the Microsoft Excel file.

Select Individual Worksheets Prompts users to select one or more worksheets when they open the file.

Use Excel Labels as Headings Determines whether text in the first row of the worksheet is converted to column headings in the data table.

By default, JMP takes the best guess. If names have been defined for all cells in the first row, the text in those cells is converted to column heading. Otherwise, columns are named *Column 1*, *Column 2*, and so on.

To override a preference, include the corresponding argument described later in this section.

Note: See "Create Excel Workbooks" for information about exporting data tables to an Excel workbook.

Open() Function

Using the `Open()` function without additional arguments to open an Excel file has different behaviors depending on the context:

- If the `Open()` function is a direct part of the script, the Excel files open into data tables using your Excel preferences. The following example opens both worksheets into data tables without using the wizard:

```
Path = Convert File Path( "$SAMPLE_IMPORT_DATA/Team Results.xlsx", absolute );
dt = Open( Path );
```

Note: To use the Excel Wizard, you must specify the `Excel Wizard` option in the argument as in `Open( "$SAMPLE_IMPORT_DATA/Team Results.xlsx", "Excel Wizard" );`.

- However, if the `Open()` function is part of a script that is run from clicking a button, the Preview window opens and requires the user to interact with it. Run the following example and click the button to see the Excel Import Wizard:

```
Path = Convert File Path( "$SAMPLE_IMPORT_DATA/Team Results.xlsx", absolute );
New Window( "button", Button Box( "Open", dt = Open( Path ) ) );
```

- To prevent the button script from opening the Preview window and importing the Excel file directly, provide additional arguments to the `Open()` function. Run the example and click the button. Both worksheets are opened into data tables without using the Excel Import Wizard.

```
Path = Convert File Path( "$SAMPLE_IMPORT_DATA/Team Results.xlsx", absolute );
New Window( "button", Button Box( "Open", dt = Open( Path, Use for all
   sheets(1) ) ) );
```

Alternatively, you can set the Excel Open Method preference to open all sheets:

```
Preference( Excel Open Method( "Open All Sheets" ) );
Path = Convert File Path( "$SAMPLE_IMPORT_DATA/Team Results.xlsx", absolute );
New Window( "button", Button Box( "Open", dt = Open( Path ) ) );
```

Open a Workbook in the Excel Import Wizard

The Excel Import Wizard shows a preview of the data and lets you modify the settings before importing the data. Specify `"Excel Wizard"` as the argument.

```
dt = Open( "$SAMPLE_IMPORT_DATA/Team Results.xlsx", Excel Wizard );
```

Customize Import Settings

Customize settings such as the number of rows in the column headers.

```
Open(
   "$SAMPLE_IMPORT_DATA/Team Results.xlsx",
   Worksheets( "Grouped Team Results" ),
   Use for all sheets( 1 ),
   Concatenate Worksheets( 0 ),
   Create Concatenation Column( 0 ),
   Worksheet Settings(
      1,
      Has Column Headers( 1 ),
      Number of Rows in Headers( 1 ),
      // customization based on Team Results structure
      Headers Start on Row( 3 )
   )
);
```

When you click a custom button to open a workbook, the Excel Open Method preference applies. To force the worksheet to open directly, specify the Excel Open Method in your script:

```
Preference( Excel Open Method( "Open All Sheets" ) );
```

Import Specific Worksheets

Suppose that you want to import data from specific worksheets in your workbook. Specify those worksheets using the `Worksheets` argument. In the following example, the worksheet named *small* is imported into JMP.

```
Open( "C:\My Data\cars.xlsx", Worksheets( "small" ) );
```

Or specify the number of the worksheet, the third worksheet in the following example:

```
Open( "C:\My Data\cars.xlsx", Worksheets( "3" ) );
```

Import multiple or all worksheets by including the worksheet names in a list:

```
Open( "C:\My Data\cars.xlsx", Worksheets( {"small", "medium", "large"} ) );
```

Import XML and JSON Files

The XML Import Wizard imports two types of nested text data: XML and JSON files. Both kinds of files contain structured text that can be nested to represent hierarchical relationships. This wizard helps you pick out the elements of the hierarchy that hold values and elements that determine rows to build a data table.

A useful way to imagine a nested text file: the document is a book of short stories. Some of the short stories might have chapters, others only have paragraphs. Chapters have paragraphs. Paragraphs have sentences. Here's an example of the full tree for such a book. This content is also available in the JMP Samples/Import Data directory as Book.xml.

```
<book>
  <story name="car poll">
    <wheels>4</wheels>
    <para><name>chev</name></para>
    <para><name>ford</name></para>
    <para><name>volk</name></para>
  </story>
  <story name="big class">
    <para><name>ralph</name><height>6</height></para>
    <para><name>billy</name><height>5</height></para>
  </story>
  <story name="cheese" location="NC">
    <chapter name="american">
      <para>1<price>2.00</price><quantity>2</quantity></para>
      <para>2<price>2.10</price><quantity>4</quantity></para>
      <para>3<price>2.20</price><quantity>3</quantity></para>
    </chapter>
    <chapter name="swiss">
      <para>1<price>3.00</price><quantity>3</quantity></para>
      <para>2<price>3.10</price><quantity>2</quantity></para>
    </chapter>
  </story>
</book>
```

This XML document (the book) contains three short stories; each short story is a data table, all different. This particular XML file uses paragraphs to represent rows in a table in all of the tables. That is not required, nor are any of the names special. If you wanted to import the cheese table, which is slightly more complicated than the other two, you'd probably want a variety column, containing "american" or "swiss" for each row. You would also want price and quantity columns. And you might want a column that doesn't have a clearly defined name for the 1, 2, 3, 1, 2 values of <para>. You'd expect five rows. Finally, you might want a table variable or a column for the location NC value.

The following example shows how to import the preceding data into a data table.

```
Open(
```

```
    "$SAMPLE_IMPORT_DATA/Book.xml",
    XML Settings(
       Row( "/book/story/chapter/para" ),
       // create a row for each value in <para>
       Col(
          "/book/story/@location",
          // create a column for the location attribute
          Column Name( "location" ),
          // rename the column
          Fill( "Use Forever" ),
          // fill an empty cell with the preceding value
          Type( "Character" ), // data type
          Format( "best" ) // column format
       ),
       Col(
          "/book/story/chapter/@name",
          Column Name( "variety" ),
          Fill( "Use Once" ),
          Type( "Character" ),
          Format( "best" )
       ),
       Col(
          "/book/story/chapter/para",
          Column Name( "interesting" ),
          Fill( "Use Once" ),
          Type( "Numeric" ),
          Format( "best" )
       ),
       Col(
          "/book/story/chapter/para/price",
          Column Name( "price" ),
          Fill( "Use Once" ),
          Type( "Character" ),
          Format( "best" )
       ),
       Col(
          "/book/story/chapter/para/quantity",
          Column Name( "quantity" ),
          Fill( "Use Once" ),
          Type( "Character" ),
          Format( "best" )
       )
    ),
    XML Wizard( 0 ) /* omit or change to 1 to preview the data
)                      in the XML Import Wizard */
```

Import from and Export to Google Sheets

A Google Sheet is a spreadsheet (which consists of sheets, like tabs) that is saved in your Google account. This enables you to continue to edit the spreadsheet and to share it with others.

To write a script to import data from a Google spreadsheet, log in to your Google account and get the URL for the spreadsheet. Also notice how the data is structured. For example, you must know which row the data begin on if there are column headers.

Give Google access to JMP when running the script for the first time.

To import data from a Google spreadsheet

```
dt = Google Sheet Import(
   Email( "yourgoogleaccount@gmail.com" ), // modify the email address
   // modify the sheet URL
   Spreadsheet(
   "https://docs.google.com/spreadsheets/d/1JO1M4ScGsAHb-bCUaOi575WWvQjXciQIV
   NOwVTrXDSe/edit#gid=2303744702" ),
   Sheets( "May","June" ), // specify the sheet name(s)
   Sheet Settings( // specify the structure of the data
      Has Column Headers( 1 ),
      Headers Start on Row ( 1 ),
      Data Starts on Row( 2 ),
      Cell Range( "" ),
      Import Cell Colors( 1 ),
      Suppress Empty Columns( 1 )
   )
);
```

Note: See *Using JMP* for more information about security, country restrictions, and more.

To export a JMP data table to a Google spreadsheet

Export a JMP data table to a new spreadsheet in Google Sheets by following this example:

```
email = "yourgoogleaccount@gmail.com"; // modify the Google account
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Google Sheet Export(
       dt,
       Email( email ),
      New Spreadsheet( "JSL Example" ), // specify the name of the spreadsheet
       Sheet Name( "Example 1" ) // specify the sheet name
);
```

“1” is returned in the log when the export is successful.

Export a JMP data table to an existing spreadsheet in Google Sheets by following this example. You'll need the URL of the existing spreadsheet.

```
id =
  "https://docs.google.com/spreadsheets/d/1J01M4ScGsAHb-bCUa0i575WWvQjXciQIV
  N0wVTrXDSe/edit#gid=2303744702"; // modify the URL
email = "yourgoogleaccount@gmail.com"; // modify the Google account
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Google Sheet Export(
       dt,
       Email( email ),
       Spreadsheet( id ),
       Sheet Name( "Example 1" )
);
```

"1" is returned in the log when the export is successful.

Note: See *Using JMP* for more information about security, country restrictions, and more.

Import SAS Data Sets

Open a SAS file as a data table without connecting to a SAS server.

```
sasxpt = Open( "$SAMPLE_IMPORT_DATA/carpoll.xpt" );
```

To convert the labels to column headings, include the `Use Labels for Var Names` argument.

```
sasdbf = Open( "$SAMPLE_IMPORT_DATA/Bigclass.sas7bdat", Use Labels for Var
  Names( 1 ));
```

.xpt and .stx file formats are also supported.

On Windows, you can also open SAS data sets from a SAS server. See "Connect to a SAS Metadata Server".

Import Web Pages and Remote Files

You can import data from websites or from other computers. The data might be a JMP data table, a table defined in a web page, or another file type that JMP supports.

Open or Import Data from a Website

In the `Open()` command, specify the quoted URL to open a file from a website. You can open JMP data tables or other supported file types this way.

```
Open( "http://company1.com/Repairs.jmp" );
Open( "http://company1.com/My Data.txt", text); // specify text on macOS
```

Import a Web Page

A web page can include data in tabular format. Import the table as a JMP data table:

```
Open( "http://company1.com", HTML Table( n ) );
```

n identifies which table you want to import. For example, to import the fourth table on the page, specify `HTML Table(4)`. If you omit the value, only the first table on the page is imported.

JMP attempts to preserve the table header defined in a `<th>` HTML tag. The table header is converted to column headings in the data table. If the `<th>` tag is wrong or missing, use `ColumnNames(n)` to specify the *n*th row. By default, `DataStarts(n)` will be the next row, or you can specify the `DataStarts` row.

Import Images from a Web Page

You can import images on the web into a data table, using the Expression data type. The following example creates a new data table and adds two images to it.

```
dt = New Table( "test", New Column( "Image", Expression ) );
dt << Add Rows( 1 );
dt:Image = New Image( Open(
  "https://www.jmp.com/support/help/en/16.0/jmp/images/iris_first.gif" )
  );
dt << Add Rows( 1 );
dt:Image = New Image( Open(
  "https://www.jmp.com/support/help/en/16.0/jmp/images/trimeth.png" ) );
```

Import Images from a Directory

You can use `Multiple File Import()` and an Event Handler script to create a data table of thumbnail images that link to a directory of full-size pictures. This example sets the cell height so that it fits the height of the tallest thumbnail.

Note: Typically, the Multiple File Import option is for importing multiple text files. However, this example shows how to use the option to import images.

```
path = "$SAMPLE_IMAGES/"; // images are in this directory

{dtx} = Multiple File Import(
  <<Set Folder( path ),
  <<Set Name Filter( "*.jpg; " ),
  <<Set Name Enable( 1 ), /* enable the ability to filter by name,
  in this example, files with the .jpg extension */
  <<Set Add File Name Column( 1 ), // must be included but can be hidden
  <<Set Add File Size Column( 1 ) // for Show Sizes at end
```

```
) << Import Data;

/* FileName is the column that contains the image file name.
The column is required for the image links to work, but you don't need
to see it. */
dtx:FileName << Hide;
maxheight = 0;

For Each Row( // recreate each image to minimize size of data table
   x = Eval( dtx:Picture );
   x << Scale( .1 ); // scale the image by 10%
   m = x << Get Pixels; // compress the image by recreating it
   dtx:Picture = New Image( m );
   maxheight = Max( maxheight, (dtx:Picture[] << Get Size)[2] );
   // width x height is returned, [2] gets the height
);

Eval( /* insert the value in the path variable into the Click handler. The
   path variable won't be available later, but the path value is needed to
   concatenate with the file name column's value to build a link to the image
   on the computer. */
   Eval Expr(
      dtx:picture << Set Property(
         "Event Handler",
         Event Handler( // Event Handler column property
            Click( JSL Quote( Function( {thisTable, thisColumn, iRow},
            // insert links to images
            Open( Expr( path )||Char( thisTable:FileName[ iRow ] ) ); ); ) ),

            Tip( JSL Quote( Function( {thisTable, thisColumn, iRow}, "Open " ||
   Char( thisTable:FileName[ iRow ] ) || " in your viewer."; ); ) ),

            // color of the link -- blue
            Color( JSL Quote( Function( {thisTable, thisColumn, iRow}, 5; );) )
         )          )
   )
);

// add two pixels to the maximum height
dtx << Set Cell Height( maxheight + 2 );

/* save the file to show that the data table is much smaller than the
original file */
dtx << Save( "$TEMP/Sample Images.jmp" );
Show( File Size( "$TEMP/Sample Images.jmp" ), Col Sum( dtx:FileSize ) );
```

Import a File from a Shared Computer

JMP can import files stored on a shared computer, such as another computer or a network drive. The following example shows how to open a file from Windows:

```
Open( "\\Data\Repairs.jmp" ); // UNC name
Open( "z:/Data/Repairs.jmp" ); // mapped letter drive
Open( "Repairs.jmp" ); // relative path in which the script is in the same
  directory as the file that you are importing
```

macOS doesn't have a concept like UNC naming or drive letters. Instead, you need to mount the volume through the user interface. It is possible to do it from a script, but you would need to have connected interactively at least once and stored the password in your keychain.

Here is an example that connects to a remote volume from either macOS or Windows:

```
mount = Function( {server},
  {Default Local},

  If( Host is( Mac ),
    Run Program(
      Executable( "/usr/bin/osascript" ),
      // replace ^server^ with your server name throughout the script
      Options( {"-e", Eval Insert( "mount volume \!"smb://^server^\!"" )} ),
      Read Function( "text" )
    );
    server = Concat Items( Remove( Words( server, "/" ), 1 ), "/" );
    Eval Insert( "/Volumes/^server^" );
  ,
    Eval Insert( "\!\\!\^server^" )
  )
);


users = mount( "data.company.com/Users" );
Files In Directory( Eval Insert( "^users^/Smith" ) );
```

Import JSON Data

JSON (JavaScript Object Notation) is a data interchange format that is based on JavaScript. It shares many of the same principles as languages based on C and is easy to read and write. Because of these characteristics, JSON is highly portable and can be imported into JMP.

Table 9.1 Representations of JSON in JSL

JSON	JSL
array	list (sequence of values)
object	associative array (set of named values)
number	number
string	string
true	1.0
false	0.0
null	. (missing value)

JSON consists of name-value pairs in an associative array. For example, here is JSON code that shows name-value pairs:

```
"[ { \!"key1\!": \!"a\!", \!"key2\!": \!"b\!"}, {\!"key1\!": \!"c\!",
  \!"key2\!": \!"d\!" }, {\!"key2\!": \!"f\!", \!"key1\!":\!"e\!",
  \!"key3\!": \!"g\!"}]";
```

Each list is imported as a column in the data table. “key1” is the name of the first column. The first value in the column is “a”. The second value is “c”, and the third value is “e”.

Use `JSON To Data Table()` to convert the JSON into a data table.

```
js = "[ { \!"key1\!": \!"a\!", \!"key2\!": \!"b\!"}, {\!"key1\!": \!"c\!",
  \!"key2\!": \!"d\!" }, {\!"key2\!": \!"f\!", \!"key1\!":\!"e\!",
  \!"key3\!": \!"g\!"}]";
d = JSON To Data Table( js ); // convert JSON to a data table
d << New Data View; // make a view of the data table
```

Figure 9.6 Imported JSON Data

	key1	key2	key3
1	a	b	
2	c	d	
3	e	f	g

You can also use `JSON To List()` to convert a string containing JSON into a list.

```
list = JSON To List(
```

```
    "[ { \!"name\!": \!"KATIE\!", \!"age\!": 12, \!"sex\!": \!"F\!",
    \!"height\!": 59, \!"weight\!": 95 }, { \!"name\!": \!"LOUISE\!",
    \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!": 61, \!"weight\!": 123 }, {
    \!"name\!": \!"JANE\!", \!"age\!": 12, \!"sex\!": \!"F\!", \!"height\!":
    55, \!"weight\!": 74 } ]"
);
Show( list );
   list = {{{"name", "KATIE"}, {"age", 12}, {"sex", "F"}, {"height", 59},
      {"weight", 95}}, ...}};
```

Exporting JSL to JSON

`As JSON Expr()` converts an associative array to a JSON string. You can then export the data into a JSON document.

```
list_of_associative_arrays = {["key1" => "a", "key2" => "b"], ["key1" => "c",
"key2" => "d"], ["key2" => "f", "key1" => "e", "key3" => "g"]};
jx = As JSON Expr( list_of_associative_arrays );
   "[{\!"key1\!":\!"a\!",\!"key2\!":\!"b\!"}..."[{\!"key1\!":\!"a\!",\!"key2\
      !":
   \!"b\!"}
```

Note: To import the data table as invisible, include the argument `Invisible(1)` in `JSON To Data Table()` and omit `New Data View`.

Import HDF5 Files

Hierarchical Data Format, Version 5 (HDF5) is a portable file format for storing data. An HDF5 file consists of groups and datasets. When you import the file, JMP opens a group to present the names of the inner datasets.

JMP handles only tables with numeric (integer, float, double) and string types, and compound files with three or fewer dimensions that contain only simple types.

You can import up to 1,000,000 columns and an unbounded number of rows.

Here is the syntax:

```
Open( "filename.h5", {"list_of", "dataset_names"});
```

Errors are written to the log when invalid data set names are passed.

Import ESRI Shapefiles

An ESRI shapefile is a geospatial vector data format used to create maps. JMP imports shapefiles as data tables. A .shp shapefile consists of coordinates for each shape. A .dbf shapefile includes values that refer to regions. To create maps in JMP, you modify the structure of the data and save the files with specific suffixes.

The following example imports a .shp file and saves it with the -XY suffix.

```
dt = Open( "$SAMPLE_IMPORT_DATA/Parishes.shp"),
:X << Format( "Longitude DDD", 14, 4 );
:Y << Format( "Latitude DDD", 14, 4 ) );
dt << Save( "c:/Parishes-XY.jmp" );
```

Save the .dbf file with the -Name suffix.

```
dt = Open( "$SAMPLE_IMPORT_DATA/Parishes.dbf" );
dt << Save( "c:/Parishes-Name.jmp" );
```

Restructuring the data requires several steps, including adding a Map Role column property to names in the -Name.jmp file. See *Essential Graphing*.

Import a Database

`Open Database()` opens a database using Open Database Connectivity (ODBC) and extracts data into a JMP data table. See the "Database Access".

JMP also converts DataBase Files (.dbf) files to data table format.

```
sasdbf = Open( "$SAMPLE_IMPORT_DATA/Bigclass.dbf",
  Use Labels for Var Names( 1 )
);
```

Set the Current Data Table

Tip: Be careful about assuming that the data table that you want is the current data table, even if you set it to be earlier in your script, because interim actions could change that.

A data table becomes the current data table in the following instances:

- When you open it like this:

  ```
  dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
  ```

- When you create a new table like this:

  ```
  dt2 = New Table( "Cities" );
  ```

To switch to another open data table, specify the name of the data table in `Current Data Table()`:

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt2 = New Table( "Cities" );
Current Data Table( dt1 ); // make Big Class.jmp the current data table
```

`Current Data Table()` can also take a scriptable object reference as an argument. The following expression would make the data table in use by the second Bivariate object the current data table:

```
Current Data Table( Bivariate[2] );
```

For more information about using references to analysis platform objects, see "Send Messages to a Platform".

Name a Data Table

Assign a name to a data table by sending it the `Set Name` message. The argument is a filename in quotation marks, or something that evaluates to a filename.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Set Name( "New Big Class.jmp" );

s = "New Big Class";
dt << Set Name( s );
```

To retrieve the name, send a `Get Name` message to the data table:

```
dt << Get Name();
   "New Big Class"
```

Save a Data Table

To save a data table, send a `Save` message to the data table. Here are some examples:

```
dt << Save(); // save using the current name
dt << Save( "Newest Big Class.jmp" ) // save as a new file
dt << Save( "c:/My Data/New Big Class.jmp" );  // save as a new file
dt << Save( "My Table", JMP( 5 ) ); // save as JMP 5 table
dt << Save("") // prompt to select the directory and save in the desired
   format
dt << Save( "Big Class.xlsx" ); // save as a Microsoft Excel file
Create Excel Workbook( "c:/MyData/data.xlsx", {"Abrasion", "Big Class"},
{"Abrasive", "Class"} ); // save several data table as a workbook
```

Note: If you specify the filename with no path and have not set the default directory, the file is saved on your primary partition (Windows) or in your *<username>*/Documents folder (macOS). For more information about setting the default directory, see "Relative Paths".

On Windows, saving with a `.txt` extension exports according to the Text Export preferences. On macOS, add `Text` as a second argument to the `Save` function:

```
dt << Save( "New Big Class.txt", Text );
```

If you plan to set the name of a data table and later send the `Save` message, you can just specify the name in a `Save` message.

```
dt << Set Name( "New Big Class.jmp" );
```

```
dt << Save();
```

works the same as

```
dt << Save( "New Big Class.jmp" );
```

Including Save and the pathname is also an alternative to using Save As along with the pathname.

Revert to a Saved Data Table

To return to the most recently saved data table, send a Revert message to the data table.

```
dt << Revert();
```

Hide a Data Table

There are two ways to hide a data table if the user does not need to see the table: by opening it as invisible or private.

Invisible Data Tables

An *invisible* data table is hidden from view but linked to analyses that you run on it. Open a data table as an invisible file:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp", "invisible" );
```

Here is an example of creating an invisible table, called Abc, that has ten rows. One column is named X.

```
dt = New Table( "Abc", "invisible", New Column( "X" ), Add Rows( 10 ) );
```

To find out if an open data table is invisible, pass the Boolean Has Data View message to the data table object. The following expression returns 0 (false) if the data table is invisible:

```
dt << Has Data View();
```

After you are finished with an invisible data table, do not forget to close it with the Close() function. Otherwise, the data table remains in memory until you quit JMP.

Showing an Invisible Data Table

The Show Window() function shows an invisible data table.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp", "invisible" );
dt << Show Window( 1 );
```

The user has the option of opening the table from the JMP interface. On Windows, an invisible data table appears in the Window List on the Home Window and the **Window > Unhide** menu. On macOS, the data table appears in the JMP Home window and the **Window > Hidden** menu.

Private Data Tables

Completely hide the data table from view by including the `private` argument:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp", "private" );
```

A private data table enables you to avoid exposing the table to general use. Sometimes a script creates a private data table to store private data generated by a process. It is not for saving memory.

See "Private, Invisible, and Visible".

Print a Data Table

Print a data table by sending it a `Print Window` message. JMP uses your computer's default printer settings.

```
dt << Print Window();
```

This message also applies to other display boxes.

Resize a Data Table

dt `<< Optimize Display` forces the data table to remeasure all of its columns and zoom to the best-sized window.

```
dt << Optimize Display;
```

Close a Data Table

To close a data table, use a `Close` command with the data table's reference as the argument. Data tables with unsaved changes are saved automatically, including any linked tables. Reports and graphs generated from the data tables are also closed.

To save and close the data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Close( dt );
```

To close the data table without saving changes:

```
Close( dt, No Save );
```

To save a copy with a new name and then close the original version:

```
Close( dt, Save("c:/My File.jmp" ) );
```

You can also close all data tables at once and either save or discard changes:

```
Close All( Data Tables );
```

```
Close All( Data Tables, No Save );
```

To close all private or invisible data tables, specify the appropriate argument:

```
Close All( Data Tables, private );
Close All( Data Tables, invisible );
```

Tip: `Close All()` also works with journals. Specify "Journals" in the argument.

Set and Get a Data Table

To specify the data table in a display box, for example, in `New Data Box()`, use `Set Data Table()` and `Get Data Table()` to set and return a data table.

The following example uses the Big Class.jmp and Cities.jmp sample data tables. The sample data tables are opened invisibly. In a new window, a new data box is created containing Big Class.jmp. After waiting 1 second, the data box contents change to display cities.jmp.

```
dtC = Open( "$SAMPLE_DATA/Cities.jmp", "invisible" );
dtA = Open( "$SAMPLE_DATA/Big Class.jmp", "invisible" );
nw = New Window( "My Table", H List Box( dtBox = dtA << New Data Box() ) );
Wait( 1.0 );
dtBox << Set Data Table( Data Table( "Cities.jmp" ) );
```

Perform Actions on All Open Data Tables

If you want to perform an action on all of the data tables that are currently open (except private tables), use `N Table()` to get a list of references to each one:

```
dt1 = Open( "$SAMPLE_DATA/Cities.jmp" );
dt2 = Open( "$SAMPLE_DATA/Big Class.jmp" );
openDTs = List();
For( i = 1, i <= N Table(), i++,
  Insert Into( openDTs, Data Table( i ) );
);
```

`openDTs` now is a list of references to all open data tables. You can send messages to any one by using `openDTs(n)`. You can use a for loop to send messages to all of the open data tables. This loop adds a new column named My Column to each open data table.

```
For( i = 1, i <= N Items( openDTs ), i++,
  openDTs[i] << New Column( "My Column" );
);
```

If you just want a list of table names and not references, use the `Get Name` message:

```
For( i = 1, i <= N Table(), i++,
  Insert Into( openDTs, Data Table( i ) << Get Name());
```

```
);
```

You can then use the list of table names to put the names of all of the open data tables in a list box (window) so that the user can choose and perform tables operations. Or, you can write the names out to a file.

Create Journals

Journals consist of JMP graphs and reports, graphics, text, and links to items such as web pages and files. They make it easy to reuse content in presentations and import into other documents.

Here is the basic syntax for journals:

```
dt << Journal;
```

You can also create a new journal window and immediately add to it within the `New Window()` command. The following examples create an empty untitled journal:

```
New Window( <<  Journal);
```

The following examples create an empty journal named “Sales”:

```
New Window( "Sales", << Journal );
```

Here is an example of creating a journal named “Test Buttons” that contains two buttons.

```
New Window( "Test Buttons", << Journal,
   Button Box( "Test One", New Window( "Hi there1", << Modal ) ),
   Button Box( "Test Two", New Window( "Hi there2", << Modal ) ) );
```

Capturing Journals

There is an optional argument in the journal and journal window commands. This argument captures (or freezes) the current display. That is, it converts the display tree with just a picture.

Here are the four options for arguments:

- Makes a bitmap “snapshot” of the display when it is sent to the journal, rather than sending a clone of the more editable display box structure to the journal. By freezing the display into a bitmap, references to variables from scripts in graphs are less problematic.

  ```
  << Journal("Freeze All");
  ```

- Similar to `Freeze All` but only within areas of a report called pictures.

  ```
  << Journal("Freeze Pictures");
  ```

- Similar to `Freeze Pictures` but only within frame boxes (which are within pictures).

  ```
  << Journal("Freeze Frames");
  ```

- Similar to `Freeze Frames` but only for the frame boxes that have scripts to draw something on them.

```
<< Journal("Freeze Frames with Scripts");
```

Current Journal

`Current Journal()` returns a reference to the display box at the top of the current journal display window. If no journal is open, one is created. There are no arguments.

You can add to the journal using the Append command. The following example adds a text box to the bottom of the current journal:

```
Current Journal() <<  Append( Text Box( "Hello World" ) );
```

You can also find items in an existing journal by enclosing the search specification in square brackets. Suppose that the current journal includes the string "Parameter Estimates". The following script appends a text box to the bottom of that journal:

```
Current Journal()["Parameter Estimates"] << Append( Text Box( "Asterisks show
  items significant at 0.05" ) );
```

Create Excel Workbooks

You can save open data tables in a new or existing Excel workbook by using the `Create Excel Workbook()` function. The first example adds all open data tables without changing the worksheet names:

```
dt1 = Open( "$SAMPLE_DATA/Big Class Families.jmp" );
dt2 = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
Create Excel Workbook( "c:/MyWorkbook1.xlsx" );
```

The following example creates MyWorkbook2.xlsx by combining the open Big Class Families.jmp and San Francisco Crime.jmp sample data tables.

```
dt1 = Open( "$SAMPLE_DATA/Big Class Families.jmp" );
dt2 = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );

// specify the data tables
tableList = {"Big Class Families", "San Francisco Crime"};

// specify the worksheets (optional)
sheetList = {"Families", "Crime"};
Create Excel Workbook( "c:/MyWorkbook2.xlsx", tableList, sheetList );
```

Use data table references to create the workbook.

```
dt1 = Open( "$SAMPLE_DATA/Big Class Families.jmp" );
dt2 = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
Create Excel Workbook(
  "c:/MyWorkbook3.xlsx",
  {dt1, dt2}, // data table references
```

```
    {"Families", "Crime"}
);
```

`Create Excel Workbook()` also accepts a mix of data table references and data table names in the list. For example, `{dt1, dt2}` above can also be expressed as `{dt1, "San Francisco Crime"}`.

Notes:

- The saved file is an .xlsx file; the .xls format is not supported.
- The JMP data tables must conform to the maximum Excel limit of 1 million rows and 16,535 columns, or the data will be truncated in Excel.
- Data table references or data table names must always be in a list, even if the list has only one item.

Prevent Changes to a Data Table

To prevent any changes to the data table (such as changing column names, changing the data type, deleting rows, and so on), use `Lock Data(1)`.

```
dt << Lock Data Table( 1 );
```

To prevent modifying adding or deleting columns and rows and modifying cells, use `Set Edit Lock`:

```
dt << Set Edit Lock( "Add Columns" );
dt << Set Edit Lock( "Delete Columns" );
dt << Set Edit Lock( "Add Rows" );
dt << Set Edit Lock( "Delete Rows" );
dt << Set Edit Lock( "Modify Cells" );
```

To get a list of data table operations that are locked, use `Get Edit Lock`:

```
dt << Get Edit Lock();
   {"Modify Cells", "Add Rows", "Delete Rows", "Add Columns", "Delete Columns"}
```

To allow a specific data table operation again, use `Clear Edit Lock`:

```
dt << Clear Edit Lock( "Modify Cells" );
```

Advanced Data Table Scripting

This section covers more advanced actions that you can perform on a data table.

- "Select Columns in a Data Table"
- "Store Summary Statistics in Global Variables"
- "Create a Table of Summary Statistics"
- "Subset a Data Table"
- "Add a Data Filter"
- "Sort a Data Table"
- "Stack Values in a Data Table"
- "Split Values in a Stacked Data Table"
- "Transpose a Data Table"
- "Vertically Concatenate Data Tables"
- "Horizontally Concatenate Data Tables"
- "Virtually Join Data Tables"
- "Replace Data in Data Tables"
- "Create a Table Using Tabulate"
- "Copy Data Tables"
- "Find Missing Data Patterns"
- "Compare Data Tables"
- "Create a Summary Table"
- "Subscribe to a Data Table"
- "Move Data Between Matrices and Data Tables"
- "Add Hyperlinks to Text Columns"

Select Columns in a Data Table

To select one or more columns, send the `Select Columns` message to the data table.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
clist = {:Height, :Weight};
dt << Select Columns( clist );
```

To select all columns, send the Select Columns message with the All argument to the data table.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Columns( all );
```

Store Summary Statistics in Global Variables

The `Summarize` command collects summary statistics for a data table and stores them in global variables. The `Summarize` command is different from the `Summary` command, which also calculates summary statistics, but presents them in a new data table.

The first argument is an optional data table reference. Include it if more than one data table might be open.

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt2 = Open( "$SAMPLE_DATA/Animals.jmp" );
Summarize( dt1,
   exg = By( :sex ),
   exm = Mean( :height )
);
Show( exg );
Show( exm );
```

Named arguments include the following: `Count`, `Sum`, `Mean`, `Min`, `Max`, `StdDev`, `First`, `Corr`, and `Quantile`. These statistics can be calculated only for numeric columns. Each argument takes a data column argument.

Notes:

- If a *name*=`By(`*groupvar*`)` statement is included, then a list of subgroup statistics is assigned to each *name*.
- `Count` does not require a column argument, but it is often useful to specify a column to count the number of nonmissing values.
- `Quantile` also takes a second argument for specifying which quantile, such as 0.1 for the 10th percentile.
- When you summarize data that contain row states, the summary statistics are not calculated. An error appears in the log instead.

Note: Excluded rows are excluded from `Summarize` calculations. If all data are excluded, Summarize returns lists of missing values. If all data have been deleted (there are no rows), Summarize returns empty lists.

The following example uses the Big Class sample data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Summarize(
   a = By( :age ),
   c = Count,
   sumHt = Sum( :height ),
   meanHt = Mean( :height ),
   minHt = Min( :height ),
```

```
    maxHt = Max( :height ),
    sdHt = Std Dev( :height ),
    q10Ht = Quantile( :height, .10 )
);
Show( a, c, sumHt, meanHt, minHt, maxHt, sdHt, q10Ht );
```

Because the script included a By group, the results are a list and six matrices:

```
a = {"12", "13", "14", "15", "16", "17"}
c = [8, 7, 12, 7, 3, 3]
sumHt = [465, 422, 770, 452, 193, 200]
meanHt = [58.125, 60.28571428571428, 64.16666666666667, 64.57142857142857,
   64.33333333333333, 66.66666666666667]
minHt = [51, 56, 61, 62, 60, 62]
maxHt = [66, 65, 69, 67, 68, 70]
sdHt = [5.083235752381126, 3.039423504234876, 2.367712103711172,
   1.988059594776032, 4.041451884327343, 4.163331998932229]
q10Ht = [51, 56, 61.3, 62, 60, 62]
```

You can format the results using `Table Box`.

```
New Window( "Summary Results",
  Table Box(
     String Col Box( "Age", a ),
     Number Col Box( "Count", c ),
     Number Col Box( "Sum", sumHt ),
     Number Col Box( "Mean", meanHt ),
     Number Col Box( "Min", minHt ),
     Number Col Box( "Max", maxHt ),
     Number Col Box( "SD", sdHt ),
     Number Col Box( "Q10", q10Ht )
  )
);
```

Figure 9.7 Results from Summarize

Age	Count	Sum	Mean	Min	Max	SD	Q10
12	8	465	58.125	51	66	5.08324	51
13	7	422	60.2857	56	65	3.03942	56
14	12	770	64.1667	61	69	2.36771	61.3
15	7	452	64.5714	62	67	1.98806	62
16	3	193	64.3333	60	68	4.04145	60
17	3	200	66.6667	62	70	4.16333	62

You can add totals to the window:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

Summarize(
  a = By( :age ),
  c = Count,
  sumHt = Sum( :height ),
```

```
    meanHt = Mean( :height ),
    minHt = Min( :height ),
    maxHt = Max( :height ),
    sdHt = Std Dev( :height ),
    q10Ht = Quantile( :height, .10 )
);

Summarize(
    tc = Count,
    tsumHt = Sum( :height ),
    tmeanHt = Mean( :height ),
    tminHt = Min( :height ),
    tmaxHt = Max( :height ),
    tsdHt = Std Dev( :height ),
    tq10Ht = Quantile( :height, .10 )
);

Insert Into( a, "Total" );
c = c |/ tc;
sumHt = sumHt |/ tsumHt;
meanHt = meanHt |/ tmeanHt;
minHt = minHt |/ tminHt;
maxHt = maxHt |/ tmaxHt;
sdHt = sdHt |/ tsdHt;
q10Ht = q10Ht |/ tq10Ht;

New Window( "Summary Results",
    Table Box(
        String Col Box( "Age", a ),
        Number Col Box( "Count", c ),
        Number Col Box( "Sum", sumHt ),
        Number Col Box( "Mean", meanHt ),
        Number Col Box( "Min", minHt ),
        Number Col Box( "Max", maxHt ),
        Number Col Box( "SD", sdHt ),
        Number Col Box( "Q10", q10Ht )
    )
);
```

Figure 9.8 Summarize with Total

Age	Count	Sum	Mean	Min	Max	SD	Q10
12	8	465	58.125	51	66	5.08324	51
13	7	422	60.2857	56	65	3.03942	56
14	12	770	64.1667	61	69	2.36771	61.3
15	7	452	64.5714	62	67	1.98806	62
16	3	193	64.3333	60	68	4.04145	60
17	3	200	66.6667	62	70	4.16333	62
Total	40	2502	62.55	51	70	4.24234	56.2

If you do *not* specify a By group, the result in each name is a single value:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Summarize(
// a = By( :age ),
   c = Count,
   sumHt = Sum( :height ),
   meanHt = Mean( :height ),
   minHt = Min( :height ),
   maxHt = Max( :height ),
   sdHt = Std Dev( :height ),
   q10Ht = Quantile( :height, .10 )
);
Show( c, sumHt, meanHt, minHt, maxHt, sdHt, q10Ht );
   c = 40;
   sumHt = 2502;
   meanHt = 62.55;
   minHt = 51;
   maxHt = 70;
   sdHt = 4.24233849397192;
   q10Ht = 56.2;
```

Summarize supports multiple By groups. For example, in Big Class.jmp, proceed as follows:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Summarize( g = By( :age, :sex ), c = Count() );
Show( g, c );
   g = {"12", "12", "13", "13", "14", "14", "15", "15", "16", "16", "17", "17"},
      {"F", "M", "F", "M", "F", "M", "F", "M", "F", "M", "F", "M"}}
   c = [5,3,3,4,5,7,2,5,2,1,1,2]
```

If you specify a By group, the results are always matrices. Otherwise, the results are scalars.

Create a Table of Summary Statistics

The Summary command creates a new table of summary statistics according to the grouping columns that you specify. Do not confuse Summary with Summarize, which collects summary statistics for a data table and stores them in global variables. See "Store Summary Statistics in Global Variables".

```
summDt = dt << Summary(
   Group( groupingColumns ),
   Subgroup( subGroupColumn ),
   Statistic( columns ),// where statistic is Mean, Min, Max, Std Dev, and so
   on.
   Output Table Name( newName ) );
```

The following example creates a new table with columns for the mean of height and weight by age, and the maximum height and minimum weight by age:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
summDt = dt << Summary(
  Group( :age ),
  Mean( :height, :weight ), Max( :height ), Min( :weight ),
  Output Table Name( "Height-Weight Chart" ) );
```

Tip: `Output Table Name` can take a quoted string or a variable that is a string.

By default, a summary table is linked to the original data table. If you want to produce a summary that is not linked to the original data table, add this option to your `Summary` message:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
summDt = dt << Summary( Group( :age ), Mean( :height ),
  Link to Original Data Table( 0 )
);
```

You can choose to add marginal statistics for grouping variables to the output columns of your data table. JMP also adds a row to the end of the data table that summarizes each level of the grouping variables. To add marginal statistics, add this option to your `Summary` message:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
summDt = dt << Summary( Group( :age ), Mean( :height ),
  Include marginal statistics
);
```

You can specify the format of the statistics column name using `statistics column name format()`. Table 9.2 shows the option and examples:

Table 9.2 Statistics Column Name Format Options and Examples

Option	Example
stat (column)	Mean (Profits ($M))
column	Profits ($M)
stat of column	Mean of Profits ($M)
column stat	Profits ($M) Mean

For example, add an option like this to your `Summary` message:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
summDt = dt << Summary( Group( :age ), Mean( :height ),
  statistics column name format( "stat of column" )
);
```

The height column is named Mean of height.

Subset a Data Table

Subset() creates a new data table from rows that you specify. If you specify no rows, Subset uses the selected rows. If no rows are selected or specified, it uses all rows. If no columns are selected or specified, it uses all columns. And if Subset has no arguments, the Subset window appears.

```
dt << Subset((<"Private">|<"Invisible">), <Selected Columns>, <Columns(column
   list)>, <Selected Rows>, <Rows([number, number, ...])>, <By(column
   list)>, <Sampling Rate(fraction)>, <Sample Size(integer)>,
   <Stratify(column list)>, <Link to Original Data Table(Boolean)>, <Copy
   Formula(Boolean)>, <Suppress Formula Evaluation(Boolean)>, <Keep by
   Column>);
```

Note: For more arguments, see the Scripting Index in the Help menu.

For example, using Big Class.jmp, to select the columns for all rows in which the age is 12:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( Selected( Row State() ) = (:age == 12) );
subdt = dt << Subset( Output Table Name( "Subset" ) );
```

To select three columns and all rows, run this script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
subDt1 = dt << Subset(
   Columns( :name, :age, :height ),
   Output Table Name( "Big Class 2" )
);
```

To select specified rows of two columns and link the two data tables, run this script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
subDt2 = dt << Subset(
   Columns( :name, :weight ),
   Rows( [2, 4, 6, 8] ),
   Linked
);
```

To select the columns for all rows in which the age is 12, run this script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age == 12 );
dt << Subset( ( Selected Rows ), Output Table Name( "Subset" ) );
```

To verify that rows were selected and then conditionally proceed to subset the current data table or open another data table, run this script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age == 12 );

/* If the condition selected no rows, open Fitness.jmp.
   Else, generate the desired subset table of Big Class.jmp.*/
If( N Rows( dt << Get Selected Rows ) == 0,

newdt = Open( "$SAMPLE_DATA/Fitness.jmp" ),
   newdt = dt << Subset( All Columns( 1 ), Selected Rows( 1 ) )
);
```

Add a Data Filter

The Data Filter provides a variety of ways to identify subsets of data. Using Data Filter commands and options, you can select complex subsets of data, hide these subsets in plots, or exclude them from analyses.

Here is the basic syntax:

```
dt << Data Filter(<Columns>, <Mode(...)>, <Add Filter(...)>, <Show Columns(),
   <Selector>, <Animation>, <Animate>, <Clear>, <Delete All>, <Delete>,
   <Filter Columns>, <Filter Column>, <Filter Group>, <Add Filter>, <Add>,
   <Match>, <Mode>, <Display>, <Where>, <Make Subset>, <Display>, <Show
   Histograms and Bars>)
```

You can send an empty `Data Filter` message to a data table, and the initial Data Filter window appears, showing the Add Filter Columns panel that lists all the variables in the data table.

`Mode` takes three arguments, all of which are optional: `Select(`*`Boolean`*`) Show(`*`Boolean`*`), Include(`*`Boolean`*`)`. These arguments turn on or off the corresponding options. The default value for `Select` is true (1). The default value for `Show` and `Include` is false (0).

`Add Filter()` adds rows and builds the WHERE clauses that describe a subset of the data table. Here is the basic syntax:

```
Add Filter( Columns( col, ... ), Where( ... ), ... )
```

To add columns to the data filter, list the columns names separated by commas. Note that this is not a list data structure.

You can define one or more WHERE clauses to specify the filtered columns:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
df = dt << Data Filter(
   Mode( Show( 1 ) ),
```

```
    Add Filter(
       Columns( :age, :sex, :height ),
       Where( :age == {13, 14, 15} ),
       Where( :sex == "M" ),
       Where( :height >= 52 & :height <= 65 )
    )
);
```

This script selects 13, 14, and 15-year-old males with a height greater than or equal to 52 and less than or equal to 65. In this filtered data, you can instead select the excluded ages by adding the `Invert Selection` message to the preceding script.

```
df << ( Filter Column( :age ) << Invert Selection );
```

In this example, ages other than 13, 14, and 15 in the filtered data are selected.

You can also use `Add Filter()` to select matching strings from columns with the Multiple Response property or Multiple Response modeling type.

```
dt = Open( "$SAMPLE_DATA/Consumer Preferences.jmp" );
df = dt << Data Filter(
   Location( {437, 194} ),
   Add Filter(
      Columns( :Brush Delimited ),
      Match None( Where( :Brush Delimited == {"Before Sleep", "Wake"} ) ),

      /* Define the display options. "Brush Delimited" appears above
      the list of values. Size is the size of the filter list. The
      values appear next to check boxes. */
      Display( :Brush Delimited, Size( 121, 70 ), Check Box Display )
      )
);
```

This script selects rows with values in the Brush Delimited column that do not match either of the specified values ("Before Sleep", "Wake"). Other available scripting options include `Match Any`, `Match All`, `Match Exactly`, and `Match Only`. See *Using JMP* for more information about the Multiple Response property and the Multiple Response modeling type.

You can also send messages to an existing Data Filter object:

```
Clear(), Display( ... ), Animate(), Mode(), ...
```

`Clear` takes no arguments and clears the data filter.

To prevent the data filter from appearing when the script is run, set `Show Window` to 0:

```
obj  = ( dt << Data Filter( Show Window( 0 ) ) );
```

To specify which values to exclude, use the `!=` operator. The following example excludes ages 16 and 17 from the filtered values.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
```

```
df = dt << Data Filter(
   Add Filter( Columns( :age, :sex ) ),
   Match( Columns( :age, :sex ),
      Where( :sex = "M" ), Where( :age != {16, 17} )
   )
);
```

To omit the Data Filter outline and red triangle menu, specify `No Outline Box`.

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
obj = dt << Data Filter(
   Add Filter( Columns( :Region, :POP ), Where( :Region == {"C", "N"} ) ),
   No Outline Box( 1 )
);
```

Add a Local Data Filter

A local data filter enables you to filter data in an individual report window without affecting other reports. When the local data filter is added to the platform, the reference to the data filter is returned. This provides an easy way to get access to the filter.

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
biv = dt << Bivariate(
   Y( :x ),
   X( :y ),
   Automatic Recalc( 1 ),
   Fit Line( {Line Color( {213, 72, 87} )} )
);

dfObj = biv << Local Data Filter(
// Include Show Histograms and Bars(0) to hide them in the filter.
// Enable Include to include rows selected by the local data filter in
// calculations that are required to build graphs controlled by the local data
// filter.  Deselected rows are excluded from calculations.
   Location( {566, 47} ),
   Mode( Include( 1 ) )
);

dfObj << Add Filter(
   Columns( :Region, :POP ),
   Where( :Region == {"C", "N"} )
);
```

Track Data Filter Changes

Make Filter Change Handler enables you to track changes to the data filter.

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
```

```
dist = Distribution(
  Automatic Recalc( 1 ),
  Continuous Distribution( Column( :POP ) )
);
filter = dist << Local Data Filter(Add Filter( columns( :Region ) ));
f = Function( {a}, Print( a ) );
rs = filter << Make Filter Change Handler( f );
```

Define the Context of a Data Filter

A data filter lets you interactively select complex subsets of data, hide the subsets in plots, or exclude them from analyses. Your selections affect all analyses of the data table.

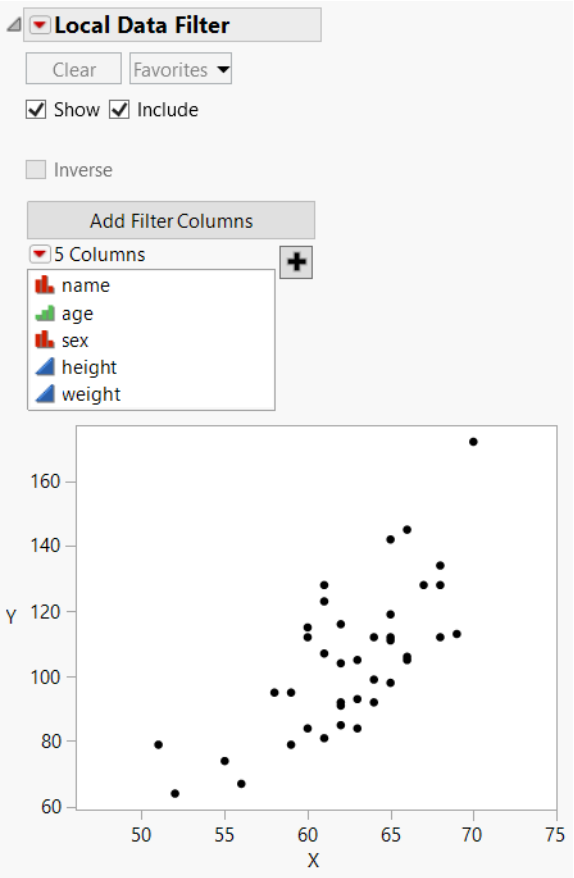
Another option is to filter data from specific platforms or display boxes. Create a local data filter inside the `Data Filter Context Box()` function. This defines the context as the current platform or display box rather than the data table.

The following example creates a local data filter for a Graph Box. See Figure 9.9 for the output.

```
New Window( "Marker Seg Example",
  Data Filter Context Box(
    V List Box(
      dt << Data Filter( Local ),
      g = Graph Box(
        Frame Size( 300, 240 ),
        X Scale( Min( xx ) - 5, Max( xx ) + 5 ),
        Y Scale( Min( yy ) - 5, Max( yy ) + 5 ),
        Marker Seg( xx, yy, Row States( dt, rows ) )
      )
    )
  )
);
```

Tip: The preceding script is part of a larger script that first builds the arrays required for a marker seg. To experiment with this script, open the Local Data Filter for Custom Graph.jsl sample script.

Figure 9.9 Local Data Filter and Graph



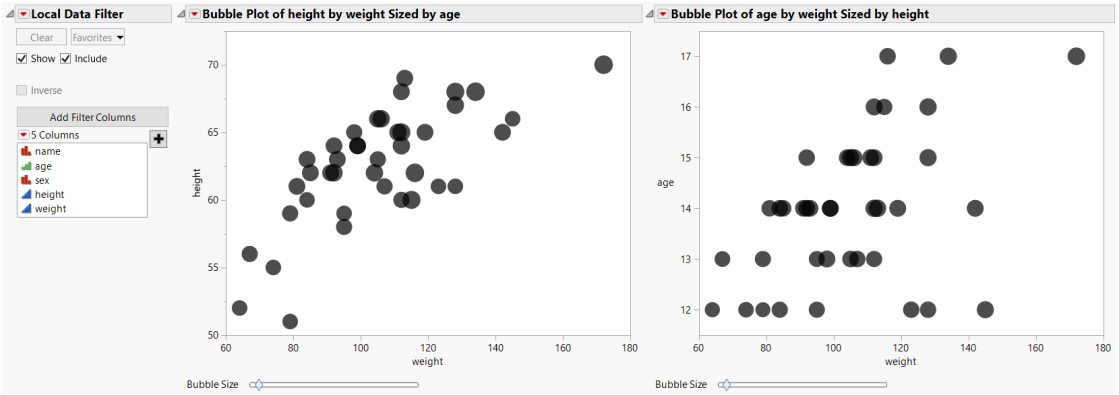
A context box can also contain several graphs with one local filter. The filtering then applies to both graphs. The following script creates two bubble plots and one data filter in one context box. See Figure 9.10 for the output.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Window( "Shared Local Filter",
   Data Filter Context Box(
      H List Box(
         dt << Data Filter( Local ),
         Platform(
            dt,
            Bubble Plot( X( :weight ), Y( :height ), Sizes( :age ) )
         ),
         Platform(
            dt,
            Bubble Plot( X( :weight ), Y( :age ), Sizes( :height ) )
         )
```

```
      )
   )
);
```

Tip: To experiment with this script, open the Local Data Filter Shared.jsl sample script.

Figure 9.10 Local Filter with Two Bubble Plots



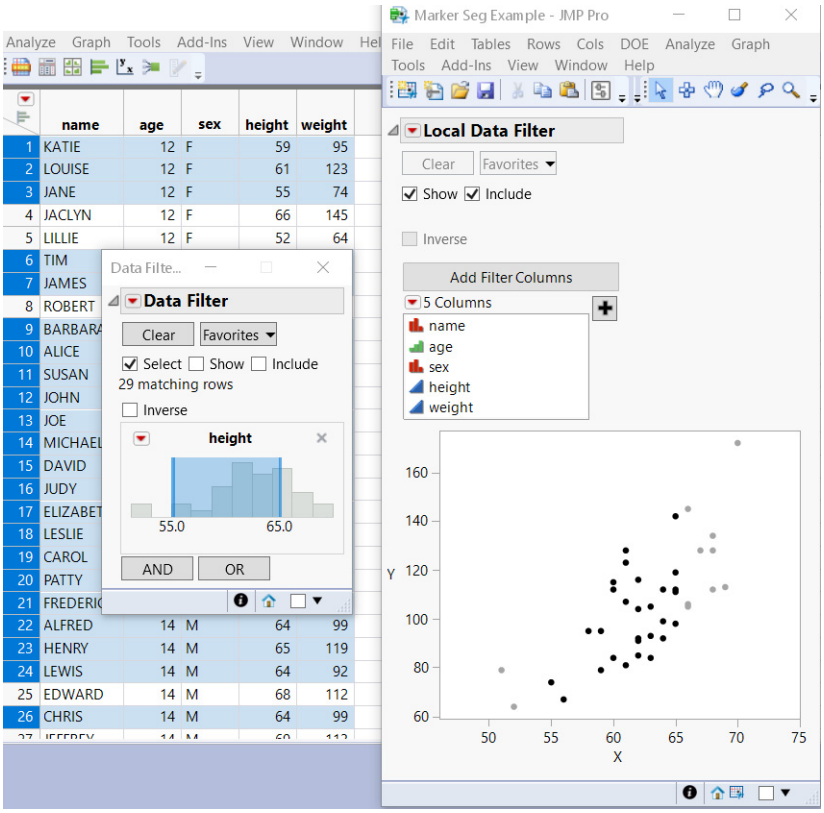
Data filters can be hierarchical. One script might generate a data filter and a local data filter. The outer data filter for the data table determines which data is available for the local data filter. To illustrate this point, the following script produces both types of data filters as shown in Figure 9.11.

Tip: The following script is part of a larger script that first builds the arrays required for a marker seg. See the Local Data Filter for Custom Graph.jsl sample script for code that builds those arrays.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
...
New Window( "Marker Seg Example",
   Data Filter Context Box(
      V List Box(
         dt << Data Filter( Local ),
         g = Graph Box(
            Frame Size( 300, 240 ),
            X Scale( Min( xx ) - 5, Max( xx ) + 5 ),
            Y Scale( Min( yy ) - 5, Max( yy ) + 5 ),
            Marker Seg( xx, yy, Row States( dt, rows ) )
         )
      )
   )
);
```

Heights greater than or equal to 55 and less than or equal to 65 are initially filtered. Then the user can work with the local data filter to filter columns on the graph.

Figure 9.11 Data Filter Hierarchy



Export a Report as Interactive HTML with a Live Data Filter

The preceding examples of adding a data filter enabled the `Include` mode. This mode includes rows selected by the local data filter in calculations that are required to build graphs controlled by the local data filter. Deselected rows are excluded from calculations.

For an Interactive HTML export, disable the `Include` mode, because Interactive HTML doesn't exclude rows from calculations.

```
dt = Open( "$SAMPLE_DATA/Quality Control/Coating.jmp" );
dt << Graph Builder(
   Size( 532, 455 ),
   Show Control Panel( 0 ),
   Variables( X( :Sample ), Y( :Weight ) ),
   Elements( Points( X, Y, Legend( 3 ) ), Line( X, Y, Legend( 5 ) ) ),
   Local Data Filter( Mode( Include( 0 ) ), Add Filter( columns( :Weight ) ) )
```

```
);
Current Report() << Save Interactive HTML( "$DESKTOP/example1.html" );
Web( "$DESKTOP/example1.html" );
```

Sort a Data Table

Sort() rearranges the rows of a table according to the values of one or more columns, either replacing the current table or creating a new table with the results. Specify ascending or descending sort for each By column.

```
dt << Sort(("<Private>"|"<Invisible>"), <Replace Table>, By(columns),
  Order(Descending | Ascending), <Output Table Name(name)>)>
```

The following example creates a new data table based on Big Class.jmp that sorts the data in descending order by age and by name:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
sortedDt = dt << Sort(
  By( :age, :name ),
  Order( Descending, Ascending ),
  Output Table Name( "Sorted age name" ) );
```

You can also use an associative array to sort values in a column. The associative array gives you a quick look at unique values in a column. See "Sort a Column's Values in Lexicographic Order".

Stack Values in a Data Table

Stack() combines values from several columns into one column.

```
dt << Stack(
  Columns ( columns ),  // the columns to stack together
  Source Label Column ( "name" ), // to identify source columns
  Stacked Data Column ( "name" ), // name for the new stacked column
  Number of Series( n ), // stacks selected columns into two or more columns
  Contiguous, // specifies that the series consists of adjacent columns
  Keep ( columns ), // the columns to keep in the data table
  Drop ( columns ), // the columns to drop in the data table
  Output Table ( "name" ), // the name for the new data table
  Columns( columns ) ), // specify which columns to include in the stacked
  table
  Copy Formula( Boolean ), // copy the formula into the stacked column
  Drop All Other Columns( Boolean ), // omit the non-stacked columns from the
  new table
  "non-stacked columns"n )( Keep( col1, ... ) // the non-stacked columns to
  keep
```

```
"non-stacked columns"n )( Drop( col1, ... ) // the non-stacked columns to
omit
Suppress Formula Evaluation( Boolean ) // do not evaluate the formula
```

For example, the following script stacks the weight and height columns in Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
StackedDt = dt << Stack(
    Columns( :weight, :height ),
    Source Label Column( "ID" ),
    Stacked Data Column( "Y" ),
   "Non-stacked columns"n )( Keep( :age, :sex ) ),
    Output Table( "Stacked Table" )
);
```

The `Columns(columns)` argument can take a list of columns, or an expression that evaluates to a list.

You can also stack multiple sets of columns into multiple columns.

```
dt = Open( "$SAMPLE_DATA/Blood Pressure.jmp" );
dt << Stack(
  Columns(
     :BP 8M,
     :BP 12M,
     :BP 6M,
     :BP 8W,
     :BP 12W,
     :BP 6W,
     :BP 8F,
     :BP 12F,
     :BP 6F
  ),
  Stacked Data Column( "BP" ),
  Source Label Column( "Day" ),
  Number of Series( 3 ), // number of sets in each series
  Contiguous, /* If you want to stack vertically. Otherwise, omit the
  argument. */

);
```

Split Values in a Stacked Data Table

`Split()` breaks a stacked column into several columns.

```
dt << Split(
  Split( columns ),
  // the column to split (required)
```

```
Split by( column ),
// the column to split by (required)

Group(column),
// split data within groups

<Private>|<Invisible>,
// resulting table is private or invisible

Remaining Columns( Keep All | Drop All | Select( columns ) ),
/* specify what to do with the remaining columns in the resulting table
(Keep All by default) */

<Copy formula( 0|1 )>,
/* include column formulas from the source table in the resulting table
(default is 1, true) */

<Suppress formula evaluation( 0|1 )>,
// stop any copied formulas from being evaluated (default is 1, true)

Sort by Column Property,
/* Sort the order of the output columns by the sort column property that is
defined for the Split by column. */

Output Table( "name" ));
// generate the output to the table name specified
```

The following example reverses the previous example for Stack(), returning essentially the original table, except that the height and weight columns now appear in alphabetic order:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Stack(
    Columns( :weight, :height ),
    Source Label Column( "ID" ),
    Stacked Data Column( "Y" ),
   "Non-stacked columns"n( Keep( :age, :sex ) ),
    Output Table( "Stacked Table" )
);
dt2 = Data Table( "Stacked Table" );
dt2 << Split(
/* Split adds the column or columns that you want to split into several
new columns. */
  Split( Y ),
/* Split By adds the column whose values you want to use as the new column
   names, and as the basis for splitting the column. */
  Split By( ID ),
```

```
  Output Table( "Split" ) );
```

When multiple rows are mapped to the same row, only one value is kept.

```
dtabc = New Table( "abc",
 New Column( "Column 1", Character, Set Values( {"a", "a", "b", "b", "c"} )),
 New Column( "Column 2", Character, Set Values( {"x", "y", "x", "y", ""} )),
 New Column( "Column 3", Set Values( [1, 2, 3, 4, 5] ))
);
/* Split adds Column 3. You want to split the values (1, 2, 3..) into several
   new cols.
Split By adds Column 2. You want to use the values (x, y) as the new column
   names. */
dtabc << Split( Split By( :Column 2 ), Split( :Column 3 ));
```

Transpose a Data Table

`Transpose` creates a new data table by flipping a data table on its side, interchanging rows for columns and columns for rows. If you specify no rows, `Transpose()` uses the selected rows. If no rows are selected, it uses all rows.

```
dt << Transpose(
   "private", "invisible",
   columns( columns ),
   Rows( row matrix ),
   By ( column ),
   Label column name( "name" ),
   Output Table( "name" )
);
dt << Transpose(
   Columns( columns ),
   Rows( row matrix ),
   Output Table Name( "name" )
);
```

The following example transposes the height and weight columns in the Big Class.jmp sample data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
tranDt = dt << Transpose( Columns( :height, :weight ),
   Output Table Name( "Transposed Columns" ) );
```

Note: The simple transpose command `dt << Transpose` brings up the Transpose window. If you do not want the window to appear, invoke the transpose as `dt << Transpose(no option)`.

Vertically Concatenate Data Tables

Concatenate, also known as a vertical join, combines rows of several data tables top to bottom.

```
dt << Concatenate( DataTableReferences,...,Keep Formulas,
  Output Table Name( "name" ) );
```

For example, if you have subsetted tables for males and females, you can put them back together using Concatenate:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :sex == "M" );
m = dt << Subset( Output Table Name( "M" ) );
dt << Invert Row Selection;
f = dt << Subset( Output Table Name( "F" ) );
both = m << Concatenate( f, Output Table Name( "Both" ) );
```

Or, instead of creating a new table containing all the concatenated data, you can append all the data to the current data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :sex == "M" );
m = dt << Subset( Output Table Name( "M" ) );
dt << Invert Row Selection;
f = dt << Subset( Output Table Name( "F" ) );
both = dt << Concatenate( m, f, "Append to First Table" );
```

Horizontally Concatenate Data Tables

Join, also known as horizontal join or concatenate, combines data tables side to side.

```
dt << Join( // message to the first table
  With( dataTable ), // the secondary data table
  Select( columns ), /* select columns from the main table
                      to add to the output table */
  Select With( columns ), /* select columns from the secondary table to
                           add to the output table */
  // join type; alternatives are
  Cartesian join, By Row Number, By matching columns(col1=col2, ...)
  Merge Same Name Columns, // merge columns with the same name
  Copy Formula( 0 ), // on by default; 0 turns it off
  Suppress Formula Evaluation( 0 ), // on by default; 0 turns it off
  Match Flag( 1 ), /* add the Match Flag column from the joined data table
  when you are matching by column */
  Update, /* replace the data in the main table with the corresponding data
  from the secondary table */
  // options for each table:
```

```
Drop Multiples( Boolean, Boolean ), // include all rows from the data table
Include Non Matches( Boolean, Boolean ), // include non-matching rows
Preserve Main Table Order(), // maintain the order of the original data
Output Table Name( "name" )); // the resulting table
```

To try this, first break Big Class.jmp into two parts:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
part1 = dt << Subset(
  Columns( :name, :age, :height ),
  Output Table Name( "Big Class 1" )
);
part2 = dt << Subset(
  Columns( :name, :sex, :weight ),
  Output Table Name( "Big Class 2" )
);
```

To make it a realistic experiment, rearrange the rows in part 2:

```
sortedPart2 = part2 << Sort( By( :name ), Output Table Name( "Sorted_Big
  Class" ) );
```

Now you have a data set in two separate chunks, and the rows are not in the same order, but you can join them together by matching on the column that the two chunks have in common.

```
part1 << Join(
  With( sortedPart2 ),
  By Matching Columns( :name == :name ),
  Preserve Main Table Order();
  Output Table Name( "Joined Parts" );
);
```

The resulting table has two copies of the name variable, one from each part, and you can inspect these to see how `Join` worked. Notice that you now have four Robert rows, because each part had two Robert rows (there were two Roberts in the original table) and `Join` formed all possible combinations.

Tip: To maintain the order of the original data table in the joined table (instead of sorting by the matching columns), include `Preserve Main Table Order()`. This function speeds up the joining process.

See the *JSL Syntax Reference* for more information about the Join arguments.

Virtually Join Data Tables

Virtual Join links a main data table to one or more auxiliary data tables. The feature enables the main data table to access data from the auxiliary data tables without physically joining the tables. See *Using JMP*.

The following example shows how to virtually join data tables and create distributions from the joined data:

```
dt1 = Open( "$SAMPLE_DATA/Pizza Profiles.jmp" );
dt1:ID << Set Property( "Link ID", 1 );// add Link ID and turn it on

dt2 = Open( "$SAMPLE_DATA/Pizza Responses.jmp" );

// add Link Reference to the Choice1, Choice2, and Choice columns
dt2:Choice1 << Set Property( "Link Reference", Reference Table(
  "$SAMPLE_DATA/Pizza Profiles.jmp" ) );
dt2:Choice2 << Set Property( "Link Reference", Reference Table(
  "$SAMPLE_DATA/Pizza Profiles.jmp" ) );
dt2:Choice << Set Property( "Link Reference", Reference Table(
  "$SAMPLE_DATA/Pizza Profiles.jmp" ) );

obj = dt2 << Distribution( // create a distribution of Crust[Choice1]
  Weight( :Subject ),
  Nominal Distribution(
    Column(
      Referenced Column(
        "Crust[Choice1]", // specify the joined column name
        Reference( Column( :Choice1 ), Reference( Column( :Crust ) ) )
      )
    )
  ),
  Nominal Distribution( Column( :Choice1 ) )
);
```

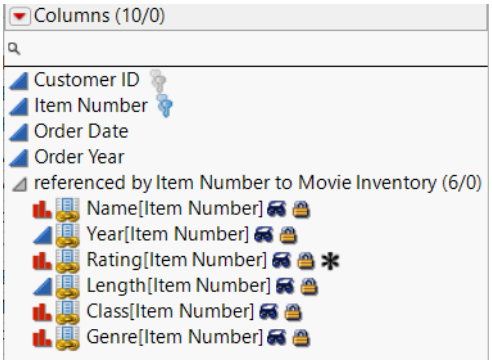
For another example of virtual join, search the Help > Scripting Index for “virtual join”.

Specifying Linked Column Names

The Columns panel in a data table shows you the names of the linked columns. For example, open the Movie Inventory.jmp and Movie Rentals.jmp sample data tables to see the linked columns shown in Figure 9.12.

The column name of the source column is followed by the name of the referencing column in brackets. For example, Figure 9.12 shows the Name source column followed by the referencing column [Item Number] in brackets.

Figure 9.12 Linked Columns in Movie Rentals.jmp



You change the name of the virtually linked column name with `Set Title`. For example, the following script changes the name of the virtually linked column Rating[Item Number] to Movie Rating in the Distribution report.

```
dt1 = Open( "$SAMPLE_DATA/Movie Inventory.jmp" );
dt2 = Open( "$SAMPLE_DATA/Movie Rentals.jmp" );
dt2 << Distribution(
   Nominal Distribution(
      Column(
         Referenced Column( // specify the joined column name
            "Rating[Item Number]",
            Reference( Column( :Item Number ), Reference( Column( :Rating ) ) )
         )
      )
   ),
   SendToReport( Dispatch( {}, "Rating[Item Number]", OutlineBox,
      {Set Title( "Movie Rating" )} ) ) /* change the name of the
      virtually linked column */
);
```

Note that running the script automatically opens the referenced table (the one with the Link ID property), just like running any other platform script, before the referenced table is opened.

You can also specify that the referenced column name be used in linked columns. This option lets you define a shorter column name (for example, Rating instead of Rating[Item Number]). Include the `Use Linked Column Name` option:

```
dt1 = Open( "$SAMPLE_DATA/Movie Inventory.jmp" );
dt2 = Open( "$SAMPLE_DATA/Movie Rentals.jmp" );

Column( dt2, "Item Number" ) << Set Property(
   "Link Reference",
   {Reference Table( "Movie Inventory.jmp" ), Options( "Use Linked Column
   Name"( 1 ) )}
```

```
);
dt2 << Distribution(
  Nominal Distribution(
     Column(
        Referenced Column( // specify the joined column name
           "Rating",
           Reference( Column( :Item Number ), Reference( Column( :Rating ) ) )
        )
     )
  )
);
```

Merging Referenced Data into One Data Table

To share the data from virtually joined data tables with another user, you might want to merge the data so that the data tables are permanently joined. This option is helpful because you provide one data table, not the main and auxiliary data tables.

When you merge referenced data, the data are replicated from the auxiliary data table to the main data table. When you save the main data table, the actual data are also saved.

The following example shows how to merge two referenced columns:

```
dt1 = Open( "$SAMPLE_DATA/Pizza Profiles.jmp" );
dt2 = Open( "$SAMPLE_DATA/Pizza Responses.jmp" );

dt1:ID << Set Property( "Link ID", 1 );

dt2:Choice << Set Property( "Link Reference", Reference Table( dt1 ) );
dt2:Choice1 << Set Property( "Link Reference", Reference Table( dt1 ) );
dt2:Choice2 << Set Property( "Link Reference", Reference Table( dt1 ) );

// select the columns to merge
dt2 << Select Columns( :Choice1, :Choice2 );

dt2 << Merge Referenced Data();
/* linked data from the referenced columns are now embedded in
Pizza Responses.jmp */
```

Note: If you don't select the columns to merge, all columns are merged.

Replace Data in Data Tables

Note: `Merge Update()` is an alias for `Update()`.

`Update()` replaces data in one table with data from a second table.

```
dt << Update( // message to first table
  With( dataTable ), // the other data table
  By Row Number, /* default join type; alternative is
                  By Matching Columns( col1 == col2 ) */
  Ignore Missing, // optional, does not replace values with missing values
  );
```

To try this, make a subset of Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
NewHt = dt << Subset( Columns( :name, :height), Output Table Name( "hts" ) );
```

Next, add 0-6 inches to each student's height:

```
diff = Random Uniform( 0, 6 );
For Each Row( NewHt, :height = :height + diff );
```

Finally, update the heights of students in Big Class.jmp with the new heights from the subset table:

```
dt << Update(
  With( NewHt ),
  By Matching Columns( :name == :name),
  );
```

Controlling the Columns Added to an Updated Table

Your updated table might contain more columns than your original table. You can select which columns are included in your updated table using the option `Add Column from Update Table()`.

To add no additional columns:

```
Data Table( "table" ) << Update(
  With( Data Table( "update data" ) ),
  Match Columns( :ID = :ID ),
  Add Columns from Update Table( None )
);
```

To add some columns:

```
Data Table( "table" ) << Update(
  With( Data Table( "update data" ) ),
  Match Columns( :ID = :ID ),
  Add Columns from Update table( :col1, :col2, :col3 )
);
```

Create a Table Using Tabulate

Tabulate constructs tables of descriptive statistics. The tables are built from grouping columns, analysis columns, and statistics keywords. The following example creates a table containing the standard deviation and mean for the height and weight of male and female students:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Tabulate( // message to data table
  Add Table( // start a new table
     Column Table( Grouping Columns( :sex ) ), // group using the column sex
     Row Table( // add rows to the table

        // use the height and weight columns for the analysis
        Analysis Columns( :height, :weight ),

        // show the standard deviation and mean
        Statistics( Std Dev, Mean )
     )
  )
);
```

You can apply a transformation to a column in the Tabulate table and set the format of the column at the same time. Use the `Transform Column()` function inside `Analysis Column()`. For example, the following script applies the Log transformation to height and sets the column format for the Mean and % of Total columns.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Tabulate(
  Set Format(
     Mean(
        :height( 10, 1 ),
        Analysis Column(
           Transform Column(
              "Log[height]",
              Formula( Log( :height ) )
           ),
           Format( 10, "Best" )
        )
     ),
     "% of Total"n(:height( 12, 2 ),
     Analysis Column(
        Transform Column(
           "Log[height]",
           Formula( Log( :height ) )
        ),
        Format( 12, 2 )
     ))
```

```
    ),
    Add Table(
       Column Table(
          Analysis Columns(
             :height,
             Transform Column(
                "Log[height]",
                Formula( Log( :height ) )
             )
          ),
          Statistics( Mean, "% of Total"n )
       ),
       Row Table( Grouping Columns( :sex ) )
    )
);
```

Note: You can also change the width of columns and perform additional operations within Tabulate. See the JMP Scripting Index in the Help menu.

Copy Data Tables

The most efficient way to copy a data table is with the `Subset()` function. The following example subsets all rows and columns into New Table.jmp.

```
sourceDT = Open( "$SAMPLE_DATA/Big Class.jmp" );
destDT = sourceDT << Subset( Columns(), Rows(), Output Table Name( "New
  Table.jmp" ) );
```

You can also use `Copy File()`.

```
Copy File( "$SAMPLE_DATA/Big Class.jmp", "c:/MyData/New Table.jmp" );
```

`Copy File()` is equivalent to copying the file on your computer. However, you have to consider whether the destination data table with the specified name already exists. And the source file has to be saved. If the source file is not saved, or the destination file already exists, the copy fails. It's safer to use the `Subset()` method shown above.

Find Missing Data Patterns

If your data table contains a lot of missing data in different columns, you might want to see whether there is a pattern in the distribution of the missing data relative to the different columns.

```
dt << Missing Data Pattern( // message to data table
  Columns( :miss ), // find missing data in this column
  Output Table( "Missing Data Pattern" ) // name the output table
```

```
);
```

Compare Data Tables

JMP can compare two open data tables and report the differences between data, scripts, table variables, column names, column properties, and column attributes. To compare data tables using JSL:

```
obj = dt << Compare Data Tables(
Compare With( Data Table ( "Data Table Name" ) );
```

For example, to compare the data tables Students1.jmp and Students2.jmp, proceed as follows:

```
dt = Open( "$SAMPLE_DATA/Students1.jmp" );
dt2 = Open( "$SAMPLE_DATA/Students2.jmp" );
obj = dt << Compare Data Tables( Compare With( Data Table( "Students2" ) ) );
```

To view the difference summary matrix for the comparison results, use this function:

```
mtx = ( obj << Get Difference Summary Matrix );
```

The resulting matrix appears in the log window:

```
[-1 1 2 2,
-1 2 4 3,
-1 1 7 4,
1 3 8 4,
1 1 10 9,
0 1 11 11,
-1 1 14 14,
-1 1 16 15,
1 1 18 16,
-1 1 19 18,
0 1 22 20,
0 1 26 24,
1 1 29 27,
1 1 34 33]
```

In the comparison summary matrix in the example, the first column represents the action performed on a column: -1 for a Delete, 0 for a Replace, and 1 for an Add. The second column represents the number of rows affected by the action. The third and fourth columns represent the row numbers affected in the two data tables in their respective order (Students1.jmp, Students2.jmp).

Create a Summary Table

A Summary table includes summary statistics for a data table, such as the mean, median, standard deviation, minimum value, and maximum value. By default, the summary table is linked to its source table. When you select rows in the summary table, the corresponding rows are highlighted in its source table.

```
dt << Summary( <private>, <invisible>, <Group( column )>, <Weight( column )>,
   <Freq( column )>, <N>, <Mean( column )>, <Std Dev( column )>, <Min( column
   )>, <Max( column )>...
```

Include columns for any statistic listed in the Summary launch window.

The following example creates a summary table from the Fitness.jmp sample data table. The N Rows column, which specifies the number of rows for each level, is included in the Summary table by default. This script creates a column of the mean for Runtime by Sex.

```
dt = Open( "$SAMPLE_DATA/Fitness.jmp" );
dt << Summary(
       Group( :Sex ),
       Mean( :Runtime ),
);
```

You can override the preselected Freq or Weight columns in the data table by specifying the Freq or Weight columns in the script. To omit the preselected Freq or Weight, use the keyword "none".

```
dt = Open( "$SAMPLE_DATA/Fitness.jmp" );
dt << Summary(
       Group( :Sex ),
       Mean( :Runtime ),
       Weight( "none" )
);
```

To unlink the Summary table, include `Link to Original Data Table( 0 )` in the script.

```
dt = Open( "$SAMPLE_DATA/Fitness.jmp" );
dt << Summary(
     Group( :Sex ),
     Mean( :Runtime ),
     Link to Original Data Table( 0 )
);
```

Subscribe to a Data Table

If you want to receive a message when a data table changes, use the `Subscribe` message. For example, you might want a message sent to the log when columns are added or deleted.

Here is the basic syntax:

```
dt << Subscribe( keyname(<client>), On Delete Columns|On Add Columns|On Add
   Rows|On Delete Rows|On Rename Column|On Close|On Save|On Rename(function)
   );
```

The first argument is a name for the subscription (or “client”), so that it can also be removed.

Note: The subscription is to the data table. The subscriber subscribes to the broadcast from the data table; it does not subscribe to a program or to script.

The application can also subscribe as a client to the data table (for example, most built-in platforms such as Distribution). If a data table has clients when the data table is closed, the user is warned that there are applications open that might need the data table.

Each subscription remains in effect until you unsubscribe:

```
dt << Unsubscribe("keyname", On Delete Columns | On Add Columns | On Add Rows
   | On Delete Rows | On Close | On Col Rename | All);
```

The following example sends messages to the log when a row is added or deleted:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
delRowsFn = Function( {a, b, rows},
   dtname = ( a << Get Name());
   Print( dtname );
   Print( b );
   Print Matrix( rows );
);
addRowsFn = Function( {a, b, insert},
   dtname = ( a << Get Name() );
   Print( dtname );
   Print( b );
   Print( insert );
);
dt << Subscribe( "Test Delete", onDeleteRows( delRowsFn, 3 ) );
dt << Subscribe( "Test Add", onAddRows( addRowsFn, 3 ) );
```

`Subscribe to Data Table List()` subscribes to a list of data tables to be notified when a new data table has been added, closed, or renamed. The following example shows how to subscribe and unsubscribe:

```
f2 = Function( {x}, Show( x ) );
Show( Subscribe to Data Table List( "My Data", OnClose( f2 ) ) );
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Wait( 2 );
Close( dt );
Show( Unsubscribe to Data Table List( "My Data" ) );
   Subscribe to Data Table List( "My Data", OnClose( f2 ) ) = "My Data";
   x = Data Table( "Big Class" );
   Unsubscribe to Data Table List( "My Data" ) = Empty();
```

Empty Application Names

If `Subscribe` is called with an empty application name, JMP generates a unique name that is returned to the caller. In the following example, `appname2` is subscribed to the data table as a client.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
appname1 = dt << Subscribe( "", On Close( Print( "Closing Data Table" ) ) );
appname2 = dt << Subscribe(
   ""( "client" ),
   On Close(
      Function( {dtab},
         dtname = ( dtab << Get Name() );
         Print( dtname );
      )
   )
);
dt << Unsubscribe( appname1, On Close );
dt << Run Script( "Distribution" )
```

Move Data Between Matrices and Data Tables

For information about moving information between a matrix and a data table, see "Matrices and Data Tables".

Add Hyperlinks to Text Columns

You can define a text column as a column of hyperlinks using the Event Handler column property. For example, you can create a column of geographical data as hyperlinks to a Google map, or you might want to include links to a Wikipedia web pages.

Note: To learn more about the Event Handler column property, add the column property to a column that contains a hyperlink and study the property in the JMP interface. Tips about the property are provided. To see the script that creates the property, select the data table red triangle menu; select **Copy Script (No Data)**; and copy the script to an empty script.

Use the Event Handler column property to build hyperlinks based on one or more columns in your data table. This example shows how the hyperlinks in the SAS Offices.jmp sample data table were created.

1. Select **Help > Sample Data Library** and open SAS Offices.jmp.
2. Right-click the City column and select **Column Info**.

   Notice that the Event Handler column property has been added.
3. The script on the Click tab inserts the code for the URL.

```
Function( {thisTable, thisColumn, iRow},
Web(
   "https://www.google.com/maps/@"||Char(thisTable:latitude[irow])||","||Char
   (thisTable:longitude[irow])||",12z" ) );
```

- `thisTable` is the data table object.
- `thisColumn` is the column object.
- `iRow` is the index of the row in `thisColumn`.
- `Web` opens the URL in the default browser.
- The `@` sign in "https://www.google.com/maps/@" is specific to the Google API for maps. Google expects coordinates to follow it.
- `||Char(thisTable:latitude[irow])` concatenates the latitude defined in the Latitude column.
- `||","||Char(thisTable:longitude[irow])||` concatenates the longitude defined in the Longitude column.
- The Google API uses `",12z"` to specify a zoom factor.

- The script on the Tip tab defines the tooltip that appears when you hover over the hyperlink. The tip accurately describes what happens after the link is clicked. Consider using this function to set the tooltip to the URL the Click script uses or a description of the platform.

  ```
  Function( {thisTable, thisColumn, iRow},
  "Open " ||
     "https://www.google.com/maps/@"||char(thisTable:latitude[irow])||","||char
     (thisTable:longitude[irow])||",12z" || " in your browser.";);
  ```

  - Returns the tool tip string.
- The script on the Color tab defines the color of the hyperlink (in this example, blue).

  ```
  Function( {thisTable, thisColumn, iRow}, 5;);
  ```

  See "JMP Colors" for a diagram of JMP colors.

You might also want to embed links to other files in a data table column. The following example shows how to use Event Handler scripts and set the values to the audio and video locations on your computer.

```
New Table( "Wildlife",
   New Column( "Column 1", Character, "Nominal",
      Set Property( "Event Handler",
         Event Handler(
            Click( JSL Quote(Function( {thisTable, thisColumn, iRow}, Open(
   Char(
thisTable:thisColumn[ iRow ] ) ); );) ),
            Tip( JSL Quote(Function( {thisTable, thisColumn, iRow}, "Open " ||
   Char(
```

```
thisTable:thisColumn[ iRow ] ) || " in a media player."; );) ),
           Color( JSL Quote(Function( {thisTable, thisColumn, iRow}, 5; );) )
        )
     ),
     Set Selected,
     Set Values( {
"C:\Users\Public\Videos\Sample Videos\Wildlife.wmv",
"C:\Users\Public\Music\Sample Music\Kalimba.mp3"
     } )
  )
)
```

The SAS Country Office and Photo columns in SAS Offices.jmp also contain examples of Event Handler scripts.

Columns

This section covers actions that you can perform on columns in a data table, such as creating, grouping, setting and getting attributes and properties, and so on.

- "Send Messages to Data Column Objects"
- "Create Columns"
- "Create a New Column by Text Matching"
- "Add Several Columns at Once"
- "Group Columns"
- "Use Set Selected to Select Columns"
- "Rearrange and Move Columns"
- "Recode Columns"
- "Add a Column Switcher"
- "Compress Selected Columns"
- "Create Custom Functions, Transforms, and Formats"
- "Custom Transform Example"
- "Delete Columns"
- "Get Column Names"
- "Column Attributes"
- "Column Properties"

Note: JMP can display math symbols and Greek letters (controlled in Preferences in the Font category). This means that if you save a column (such as T square limits), the column name could either be "T Square Limits" (no special characters) or "T^2 Limits" (with special characters). Any reference using the column name must match the name exactly, or it fails.

Send Messages to Data Column Objects

Just as you send data table messages to a data table reference, you can send column messages to a reference to a data column object. The `Column` function returns a data column reference. Its argument is either a name in quotation marks, something that evaluates to a name in quotation marks, or a number.

```
Column( "age" );       // a reference to the age column
col = Column( 2 );     // assign a reference to the second column
```

Note: You can also use the syntax `"Profits ($M)"n` to evaluate to a name that contains special characters.

The *Scripting Guide* uses *col* to represent data column references. To see the messages that you can send to data column objects, see the Scripting Index (**Help > Scripting Index > Objects > Data Tables > Column Scripting**). Alternatively, you can use the `Show Properties()` command:

```
Show Properties( col );
```

Note: With column references, you must include a subscript to refer to the individual data values. Without a subscript, the reference is to the column object as a whole.

Once you have stored a data column reference in a global variable, to modify columns, you send messages to the data column reference.

Sending messages to columns is comparable to sending messages to data tables. Either state the object, a double-angle operator <<, and then the message with its arguments in parentheses, or use the `Send()` function with the object and then the message. In some cases the messages themselves need no arguments, so the trailing parentheses are optional.

```
col << message( arg, arg2, ... );
Send( col, message(arg, arg2, ... ) );
```

As with data tables and other types of objects, you can stack or list messages:

```
col << message << message2 << ...
col << {message, message2, ...};
```

Tip: To delete a column, you must send the message to a data *table* reference, because objects cannot delete themselves, only their containers can delete them.

Access Cell Values through Column References

Always use a subscript on a column reference to access the values in the cells of columns. Without a subscript, the reference is to the column object as a whole.

```
x = col[irow]    // specific row
x = col[]        // current row
col[irow] = 2;   // as an l-value for assignment
dt << Select Where( col[] < 14 );  // in a WHERE clause
```

Access Cell Values that have Value Labels

If the column whose values you want to access has a Value Label property, you might want to access the value label (formatted value) instead of the actual data value. To do this, use the formatted option. Consider the following example:

```
dt = Open( "$SAMPLE_DATA\Big Class.jmp" );
dt << Run Script( "Set Age Value Labels" );
x = Column( dt, "age", formatted )[1];
     Show( x );
y = Column( dt, "age" )[1];
     Show( y );
```

The log shows the formatted value for *x*, which is `Twelve`, and the actual data value for *y*, which is `12`.

Create Columns

To add a new column to a data table, send a `New Column` message to a data table reference or use the `New Column()` function. The first argument, the column's name, is required. Either enclose the name in quotation marks or specify an expression evaluating to the name.

```
dt = New Table( "MyData1.jmp" );
dt << New Column( "wafer" );
```

or

```
dt = New Table( "MyData2.jmp" );
a = "wafer";
dt << New Column( a );
```

or

```
dt = New Table( "MyData3.jmp" );
col = New Column( "X", Formula( Random Uniform() ) );
```

If the table already includes a column by the same name, a sequential numeric value is appended to the new column name (wafer, wafer 2, wafer 3, and so on).

You can specify the following properties when creating a new column:

- Data type (numeric, character, row state, or expression)
- Modeling type (continuous, nominal, ordinal, multiple response, unstructured text, or vector)
- Column width (only for numeric columns)
- Numeric format

Unless otherwise specified, columns are numeric, continuous, and 12 characters wide.

The following example creates a new column whose data type is numeric, modeling type is continuous, width is set to 5, and numeric format is set to Best:

```
dt << New Column( "wafer", Numeric, "Continuous", Format( "Best", 5 ) );
```

The following example creates a character column and automatically assigns the nominal modeling type.

```
dt << New Column( "Last Name", Character );
```

You can also add formulas and other script messages appropriate to the column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Ratio", Numeric, "Continuous", Formula( :height/:weight )
  );
dt << New Column( "myMarkers",
   Row State,
   Set Formula( Marker State( age - 12 ) )
);
```

You can also create new columns that have the same properties as an existing column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Like Name", Like( :name ) );
```

If you plan to work with specific columns later (for example, grouping or changing data types), create a column reference:

```
myCol = dt << New Column( "Birth Date" );
```

Fill a Column with Data

To fill the column with data, use `Values` or its equivalent, `Set Values`. Include the value for each cell in a list.

The following example adds a new column called Last Name to a new data table. Three values are added: Smith, Jones, and Anderson.

```
dt = New Table( "My Data");
dt << New Column( "Last Name", Character, Values( {"Smith", "Jones",
   "Anderson"} ) );
```

The column can also be filled with numeric values. The following example adds a new column called Row Number to a new data table. The function N Row returns the number of rows in the table, and numeric values populate all of the rows in the column, beginning with 1.

```
dt = New Table( "My Data");
dt << New Column( "Last Name", Character, Values( {"Smith", "Jones",
   "Anderson"} ) );
dt << New Column( "Row Number",
   Numeric,
   Values( 1 :: N Row() ),
   Format( "Best", 5 )
);
```

To add a column of random numbers, insert a formula along with the random function.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Random", Numeric, Formula( Random Uniform() ) );
```

Another option is to add a constant numeric value to each cell. In the following example, the number 5 is added to each cell.

```
dt << New Column( "Number" );
:Number << Set Each Value( 5 );
```

New Column() can also be used as a built-in function, in other words, without the << (Send) command applied to the data table reference. When used in this way, the column is added to the current data table.

```
dt << New Column( "Address" ); // command is sent to the data table reference
New Column( "Address" ); // column is added to the current data table
```

Create a New Column by Text Matching

The New Column by Text Matching option enables you to use regular expressions (which identify patterns in text) to match text in a column and then create a new column from the matched text. This feature is an alternative to using the Text Explorer platform to write regular expressions and then save the results to a column.

The following example shows how to select *cat*, *cats*, *dog*, and *dogs* in a column and put the values in new columns.

```
dt = Open( "$SAMPLE_DATA/Pet Survey.jmp" );
dt << New Column by Text Matching(
```

```
   Column( :Survey Response ),
   Set Regex(
      Custom(
         Title( "Cat" ),
         /* finds "cat", optionally followed by "s",
            with either a space or period afterward. */
         Regex( "(cat(?:s?))([\s\.])" ),
         Result( "\[\1]\" ), // results in the value matched by (cat...)
      ),
   ),
   Output Column Name( "Instances of Cats" )
);
dt << New Column by Text Matching(
   Column( :Survey Response ),
   Set Regex(
      Custom(
         Title( "Dog" ),
         Regex( "(dog(?:s?))([\s\.])" ),
         Result( "\[\1]\" ),
      ),
   ),
   Output Column Name( "Instances of Dogs" )
);
```

For more information about using regular expressions in JSL, see "Regular Expressions". To learn more about regular expressions, visit Regular-Expressions.info.

Add Several Columns at Once

The `Add Multiple Columns` message creates several columns at once. Arguments include the column name prefix, the number of columns, where to insert them (`Before First`, `After Last`, or `After(col)`), and the data type (`Numeric`, `Row State`, or `Character(width)`). An additional argument, field width, is optional for a numeric column.

```
// add two row state columns named "beginning <n>" before the first column
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Add Multiple Columns( "beginning", 2, Before First, Row State );

// add three numeric columns named "middle <n>" after the height column
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Add Multiple Columns( "middle", 3, After( :height ), Numeric );

// add four character columns named "end <n>" after the last column
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Add Multiple Columns( "end", 4, After Last, Character( 4 ) );
```

The column name is a prefix. If multiple columns with the same name are added, a sequential number is appended to each column name (beginning 1, beginning 2, and so on).

Group Columns

You can group columns by sending the data table the `Group Columns` message, which takes a list of columns to group as an argument. For example, the following code opens the Big Class.jmp sample data table and groups the age and sex columns.

```
dt = Open( "$SAMPLE_DATA\Big Class.jmp" );
dt << Group Columns( {:age, :sex} );
   "age etc."
```

You can also send a column name followed by the number of columns to include in the group. The group includes the first column named and the *n*-1 columns that follow it. This line is equivalent to the line above, which groups age and sex:

```
dt << Group Columns( :age, 2 );
```

The group name is based on the first column specified in the argument. So in the preceding example, the group is automatically named age etc. To customize the name, include the group name as the first argument.

```
dt << Group Columns( "My group", :age, 2 );
```

To ungroup grouped columns, use the `Ungroup Columns` message, which takes a list of columns in a group as an argument. For example, the following line ungroups the two columns grouped in the previous example.

```
dt << Ungroup Columns( {:age, :sex} );
```

Note the following about grouping and ungrouping columns:

- Both messages take a single list as an argument. The list must be enclosed in braces.
- You cannot create more than one group in a single message (for example, by giving the `Group Columns` message two lists of columns). Instead, you must send the data table two separate `Group Columns` messages.
- The `Ungroup Columns` message takes a list of columns to ungroup, not the name of a group of columns. You can remove a partial list of columns from a group. For example, this line creates a group of four columns:

  ```
  dt << Group Columns( {:age, :sex, :height, :weight} );
  ```

  And this line removes two of the columns, while leaving the other two in the group:

  ```
  dt << Ungroup Columns( {:age, :sex} );
  ```

  Notice that the column group's name does not change automatically. You have to change the name.

Get Column Groups or Names

To get column groups or names, use `Get Column Group` or `Get Column Groups Names`. The first part of this example creates two sets of grouped columns as shown in the previous section. `Get Column Group()` returns the column names in the specific group. `Get Column Groups Names` returns the names of the column groups.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Group Columns( {:age, :sex} );
   "age etc."
dt << Group Columns( {:weight, :height} );
   "height etc."
dt << Get Column Group( "age etc." );
   {:age, :sex}
dt << Get Column Groups Names();
   {"age etc.", "height etc."}
```

Select a Column Group

To select or deselect a column group, use the following messages:

```
dt << Select Column Group( "name" ); // select the columns in the group
dt << Deselect Column Group( "name" ); // deselect the columns in the group
```

The following example groups columns X and Y and groups columns Ozone through Lead, and then selects those groups in the data table:

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Group Columns( "xy", {:X, :Y} );
dt << Group Columns(
   "pollutants",
   :Ozone :: :Lead
);
dt << Select Column Group( "xy", "pollutants" );
```

Rename a Column Group

To rename a column group, use the following message:

```
dt << Rename Column Group( "name", "toname" );
```

The following example renames the column group from xy to coordinates:

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Group Columns( "xy", {:X, :Y} );
dt << Group Columns(
   "pollutants",
   :Ozone :: :Lead
);
Wait( 3 );
```

```
dt << Rename Column Group( "xy", "coordinates" );
```

Move a Column Group

To move a column group, use the following message:

```
dt << Move Column Group (To First | To Last | After(column) | After(group));
```

The following example moves the columns in the pollutants group to the end:

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Group Columns( "xy", {:X, :Y} );
dt << Group Columns(
   "pollutants",
   :Ozone :: :Lead
);
dt << Move Column Group( To Last, "pollutants" );
```

Use Set Selected to Select Columns

Send the `Set Selected` message to select a column.

```
col << Set Selected( 1 );
```

To select all columns with a specific string in the name, run the following script:

```
dt = Open( "$SAMPLE_DATA/Semiconductor Capability.jmp" );

colList = dt << Get Column Names( String );

For( i = 1, i <= N Cols( dt ), i++,
   If( Contains( colList[i], "NPN" ), /* select only columns with NPN
   in the name */
      Column( colList[i] ) << Set Selected( 1 )
   )
);
```

If you want to select many columns, consider sending the `Select Columns` message to the data table. See the *Scripting Guide*.

Get Selected Columns

To get a list of currently selected columns, use the `Get Selected Columns` message.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
:age << Set Selected( 1 );
:sex << Set Selected( 1 );
dt << Get Selected Columns();
   {:age, :sex}
```

Return the list of selected columns as a string using the `string` argument:

```
dt << Get Selected Columns( string );
   {"age", "sex", "height"}
```

Once you know what columns are selected, you can then write a script that acts upon these columns. Or your script can iteratively select columns and act upon them one at a time.

To actually select the columns before getting the columns, send the `Set Selected` message to a column. See "Column Attributes".

Go To Column

To select and move to a specific column, use the `Go To` message.

```
dt << Go To( column name | column number );
```

For data tables with many columns, you can use this message to scroll the data table all the way to the left, so that the first column comes into view and is selected:

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
dt << Go To( 1 );
```

Invert Column Selections

To select any column that is currently deselected and deselect any column that is currently selected, use the `Invert Column Selection` message.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt:age << Set Selected( 1 );
dt:height << Set Selected( 1 );
Wait( 1 );
b = dt << Invert Column Selection;
```

If a list of columns is provided, the columns that are not in the list are selected.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
a = {:height, :weight};
b = dt << Invert Column Selection( a );
```

Rearrange and Move Columns

These messages enable you to rearrange columns in a data table:

```
dt << Reorder By Name;           // alphanumeric order
dt << Reverse Order;             // reverse current order
dt << Reorder By Data Type;      // row state, character, and then numeric
dt << Reorder By Modeling Type;  // continuous, ordinal, nominal
dt << Original Order;            // saved order
```

These commands move the currently selected column or columns to the indicated destination point.

```
dt << Move Selected Column( to first );
dt << Move Selected Column( to last );
dt << Move Selected Column( after( "name" ) );
```

Note that `Move Selected Column` is an alias for `Move Selected Columns`.

You can also move columns without first selecting them in the data table by using the following syntax.

```
dt << Move Selected Columns( {"name"}, to first );
dt << Move Selected Columns( {"name"}, to last );
dt << Move Selected Columns( {"name"}, after( "name" ) );
```

You may use a list to specify the column names.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
list = {"name", "sex"};
dt << Move Selected Columns( list, To Last );
```

Recode Columns

Use `Recode Column` to change all of the values in a column at once. Specify the original column, or reference column, a list of transformations, and a target column. The function applies the transformations to the reference column, updating the target column. If the `Update Properties` argument is turned on, the following column properties are updated when a column is recoded: Value Labels, Value Scores, Value Order, Value Colors, Supercategories, Coding, and Missing Value Codes.

```
obj << Recode Column(<column reference>, {<transform>, ...}, <Update
  Properties(0|1)>, Target Column(<column reference> | <column name>))
```

For example, in Consumer Preferences.jmp, change the coding of the Gender column to 0|1 instead of 1|2, but keep the value labels the same.

```
dt = Open( "$SAMPLE_DATA/Consumer Preferences.jmp" );
col = New Column( :Gender );
dt << Recode Column(
   :Gender,
   {if(
      _rcNow == 1, 0,
      1,
   )},
   Update Properties( 1 ),
   Target Column( col )
);
```

You can also make the transformations to the original column. Recode the age column to age groups in Big Class.jmp.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Recode Column(
   :age,
   {if(
      _rcNow >= 17, "Older",
      _rcNow >= 15, "Middle",
      "Younger"
   )},
   Target Column( :age)
);
```

Add a Column Switcher

Use JSL to add a Column Switcher panel to a report window. The Column Switcher lets you quickly analyze different variables without having to re-create your analysis.

```
// add a Column Switcher to the report
obj << Column Switcher(<default_col>(<col1>, <col2>, ...);

// remove the Column Switcher from a report
obj << Remove Column Switcher();
```

For example, add the Column Switcher to a Contingency report for the Car Poll.jmp data:

```
dt = Open( "$SAMPLE_DATA/Car Poll.jmp" );
obj = dt << Contingency(
   Y( :size ),
   X( :marital status )
);
ColumnSwitcherObject = obj << Column Switcher(
   :marital status,
   {:sex, :country, :marital status}
);
// number of pixels for the switcher width
ColumnSwitcherObject << Set Size( 200 );

// number of columns to display in the switcher
ColumnSwitcherObject << Set NLines( 6 );
```

You can also share a Column Switcher between two platform reports:

```
dt = Open( "$SAMPLE_DATA/Nicardipine.jmp" );
New Window( "Column Switcher Example",
   H List Box(
      sexCS = dt << Column Switcher( :Sex, {:Sex, :Causality, :Race} ),
```

```
        nDist = Distribution( Nominal Distribution( Column( :Sex ) ) ),
        bar = Graph Builder(
           Show Control Panel( 0 ),
           Variables( X( :Sex ), Y( :Age ) ),
           Elements( Box Plot( X, Y, Legend( 5 ) ) )
        )
     )
  );

  sexCS << Link Platform( nDist );
  sexCS << Link Platform( bar );
```

Note: If you are publishing reports that share a Column Switcher to JMP Public or JMP Live, add them to a dashboard instead of using JSL. See *Using JMP*.

For more information about the Column Switcher, see *Using JMP*.

Compress Selected Columns

To minimize the size of a large data table, use the `Compress Selected Columns` message. Each column is compressed into the most compact form.

```
dt << Compress Selected Columns( {column name, column name} );
```

For example, compress the age, sex, height, and weight columns in Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Compress Selected Columns(
{:age, :sex, :height, :weight}
);
```

For more information about this feature, see *Using JMP*.

Create Custom Functions, Transforms, and Formats

You can create user-defined custom functions and then use them in the Formula Editor's function list, in JSL scripting, as custom formats, and as custom transforms. Note that these custom functions can only be used in the context of the current JMP session. Restarting JMP removes previously defined custom functions. They must be defined and added for each session. An easy way to do this is to define your functions in an add-in or a startup script that runs each time you start JMP.

To define a custom function, use `New Custom Function()`. After a function is defined, you can add or activate it in the current running instance of JMP using `Add Custom Functions()`. Similarly, you can remove or deactivate a function using `Remove Custom Functions()`.

`New Custom Function()` requires three parameters: namespace, functionName, functionDefinition. Additional messages may be passed to define the following:

- Formula Editor values: Formula category, Result type, Parameters
- Custom transform values: Transform category
- Custom format values: Custom format category
- Scripting Index values: Description, Prototype, Scripting Index category, Examples

Tip: If you receive a data table that uses a custom transform, function, or format, you'll see "###" in the column. The data table author must give you a script to add that custom item to your computer. Run the script and then reopen the data table.

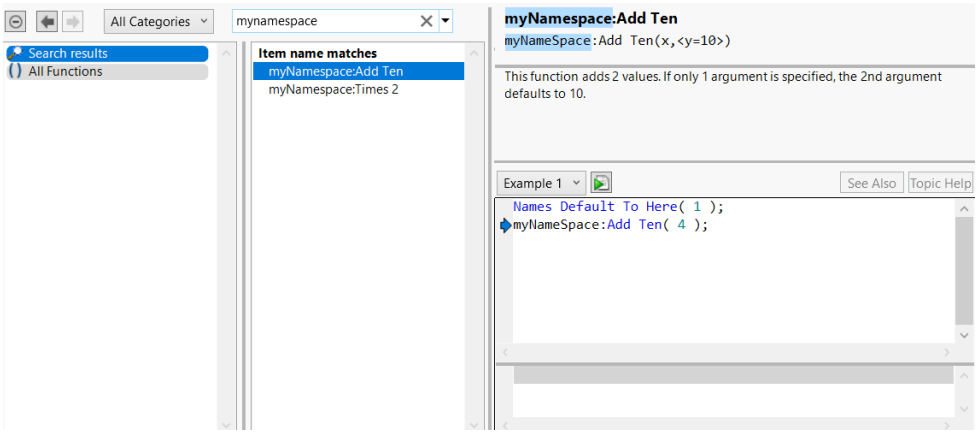
Custom Function Example

Let's create a completely specified function (including Scripting Index information and hint text). We will define the function first, then activate it.

```
funcDef = Function( {x, y = 10}, x + y );
description = "This function adds 2 values. If only 1 argument is specified,
   the 2nd argument defaults to 10.";

// examples can be multiple statements inside expr()
ex2 = Expr(
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   myNameSpace:Add Ten( :age[4], 20 );
);
parmHint1 = "x value";
parmHint2 = "<y=10>";

/* create function using variables
the commands can also be accepted as arguments in New Custom Function */
newAdd = New Custom Function( "myNamespace", "Add Ten", funcDef );
newAdd << Description( description );
newAdd << Prototype( "myNameSpace:Add Ten(x,<y=10>)" );
newAdd << Example( "myNameSpace:Add Ten(4)" );

// expr() Examples need to be passed inside NameExpr()
newAdd << Example( Name Expr( ex2 ) );
newAdd << Parameter( "Number", parmHint1 );
newAdd << Parameter( "Number", parmHint2 );
newAdd << Formula Category( "NumberStuff" );

// add myNamespace:Add Ten function to system
Add Custom Functions( newAdd );
```

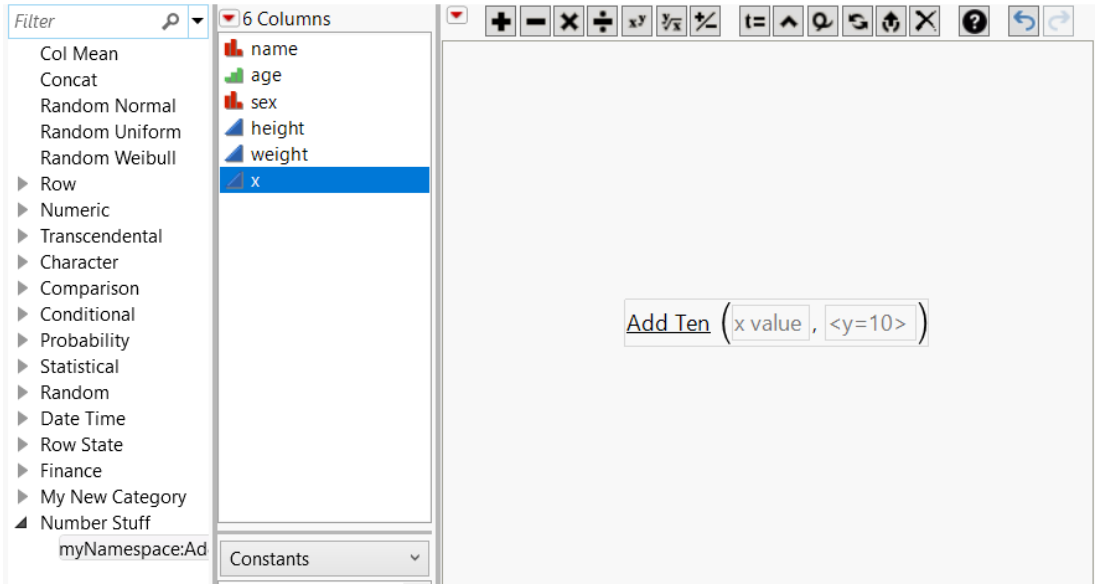
Running the above example will write “Deploying function: myNamespace:Add Ten” in the JMP log. Search for this function in the Scripting Index to see the results of using the Description, Prototype, and Example messages.

Figure 9.13 Custom Function in the Scripting Index



The `Formula Category` message results in a new category in the Formula Editor’s list of formula categories. Custom functions are shown with an underline in the Formula Editor. Also see the hint text from the `Parameter` messages.

Figure 9.14 Custom Function in the Formula Editor



Delete a Custom Function Example

To delete a custom function, use `Remove Custom Functions( {"name"} )`.

```
Remove Custom Functions( {"myNamespace:Add Ten"} );
```

Custom Transform Example

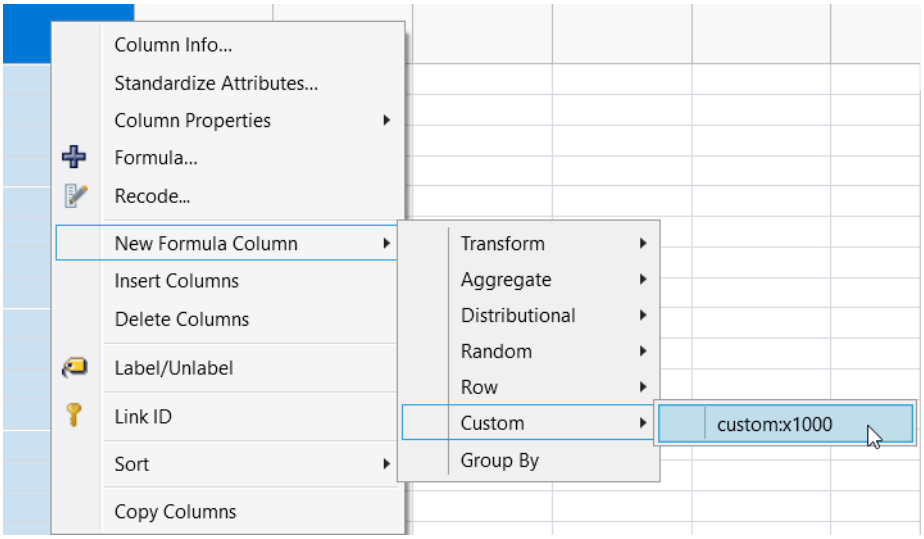
Let's create a simple Custom Transform to multiply a value by 1,000. We embed the function definition inside `Add Custom Functions()`.

```
Add Custom Functions(
  New Custom Function(
    "custom", // namespace in which new function resides
    "x1000", /* function name. The completely scoped name is
      "custom:x1000".*/

    Function( {x}, x * 1000 ),// function definition

    // optional message to enables this as a custom transform
    <<Transform Category( 1 )
    )
);
```

Running the above example writes "Deploying function: custom:x1000" in the JMP log. Search for this function in the Scripting Index to show that `custom:x1000` is an existing function.

When you create a new formula column in a data table, the custom transform appears as shown in Figure 9.15.

Figure 9.15 Custom Transform in a Data Table



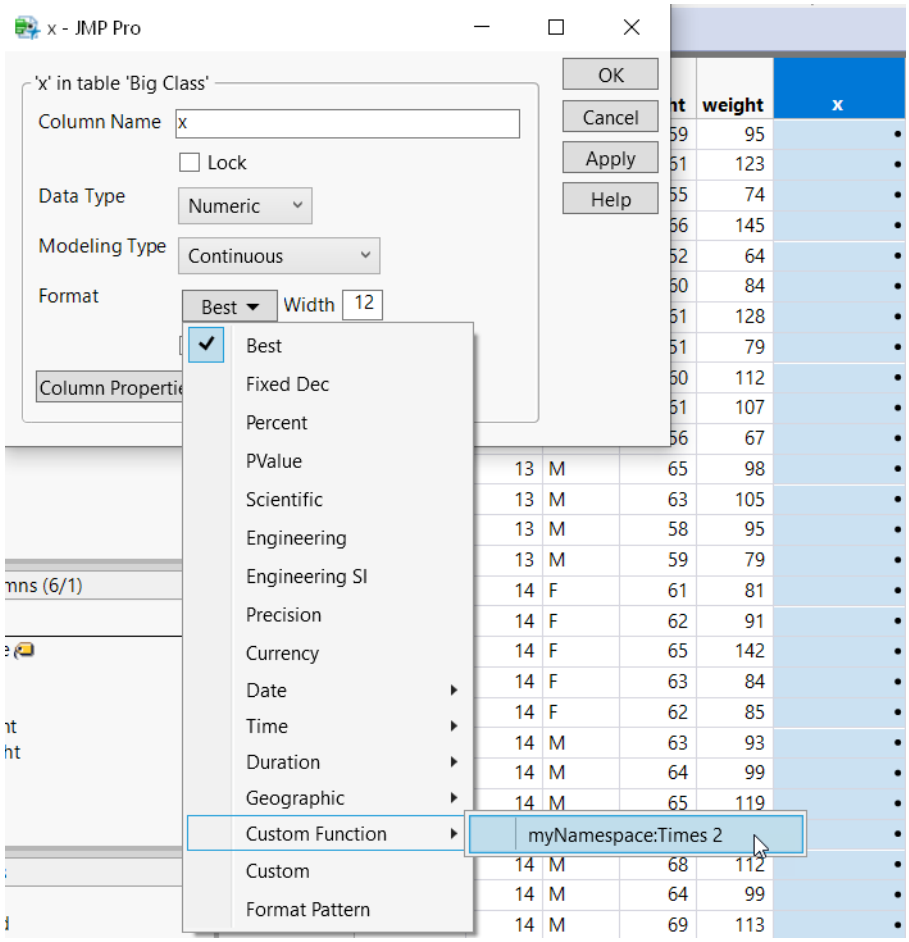
Custom Format Example: Multiplication

The last type of custom function is a custom format. You create custom formats using `New Custom Function()` and by specifying the `Custom Format Category(1)` message. Here's an example that multiplies an input variable by 2 and also displays `(x2)` after the value. Note that, in this example, this function persists as a custom format and in the formula category "My New Category". This allows you to use the format in a column formula or as a custom format.

```
/* add a New Custom Function, specifying it as a Custom Format
show "(x2)" after the value in the data table */
Add Custom Functions(
   New Custom Function(
      "myNamespace",
      "Times 2",
      Function( {inputVar},
         {Default Local},
         Char( inputVar * 2 ) || " (x2)"
      ),
      <<Description( "Multiply input by 2. Display (x2) after the new value."
   ),
      <<Formula Category( "My New Category" ),
      <<Custom Format Category( 1 ) // enable this as a custom format
   )
);
```

By specifying the `Custom Format Category(1)` message, this format now persists under the Custom Function category in the Format menu. Here's what this looks like when applying this new Times 2 custom format in the Column Info window:

Figure 9.16 Custom Format in the Col Info Window



Custom Format Example: Dates

Suppose that you want to display dates in a format that isn't provided in JMP. The following script formats a data to show the month, year, day, and day of week.

```
Add Custom Functions(
  New Custom Function(
    "myNamespace",
    "DateYMD_Day",
    Function( {inputVar},
      {Default Local},
```

```
            Char(
               Match( Month( inputVar ),
                  1, "January ",
                  2, "February ",
                  3, "March ",
                  4, "April ",
                  5, "May ",
                  6, "June ",
                  7, "July ",
                  8, "August ",
                  9, "September ",
                  10, "October ",
                  11, "November ",
                  12, "December "
               )
            ) || Char( Year( inputVar ) ) || ", Day#" || Char( Day( inputVar ) )
      || "-" ||
            Char(
               Match( Day Of Week( inputVar ),
                  1, "Sunday",
                  2, "Monday",
                  3, "Tuesday",
                  4, "Wednesday",
                  5, "Thursday",
                  6, "Friday",
                  7, "Saturday"
               )
            )

         ),
         <<Description(
            "Date format showing Month, Year, Day and Day of Week \!nfor example,
      April 1930, Day#29-Tuesday"
         ),
         <<Custom Format Category( 1 )
      )
   );
```

After you run the script, you can assign the format to a column of date values. In the Cols > Column Info window, the new date format is in the Format > Custom Function menu. You need to widen the data table column to be wide enough to display all of the characters.

Tip: See "Display Numbers as Four Digits" for another example of creating a custom format.

Delete Columns

To delete columns, send a `Delete Columns` message and specify which column or columns to delete. To delete more than one column, list the columns as multiple arguments or as a list.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Delete Columns( "weight" );
dt << Delete Columns( "weight", "age", "sex" );
dt << Delete Columns( {"weight", "age", "sex"} );
```

Without an argument, `Delete Columns` deletes columns that were previously selected. See "Column Attributes".

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column("age") << Set Selected;
dt << Delete Columns();
```

Get Column Names

`Column Name(`*n*`)` returns the name of the *n*th column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column Name( 2 );
   age
```

The returned value is a name value, not a quoted string. What this means is you can use it anywhere you would normally use the actual name in a script. For example, you could subscript it:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column Name( 2 )[1];
   12
```

If you want the name as a text string, quote it with `Char`:

```
Char( Column Name( 2 ) );
   "age"
```

To retrieve a list of the names of all columns in a data table, submit `Get Column Names`.

```
dt << Get Column Names( argument );
```

where the optional *argument* controls the output of the `Get Column Names` function, as follows:

- Specify `Numeric`, `Character`, `Row State`, or `Expression` to include only those column data types.
- Specify `Continuous`, `Ordinal`, `Nominal`, `Multiple Response`, `Unstructured Text`, `None`, and `Vector` to include only those modeling types.
- Specify `String` to return a list of strings rather than column names.

For example, if you want to get numeric and continuous columns in the Big Class.jmp sample data table, proceed as follows:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
names = dt << Get Column Names(Numeric, "Continuous")
   {height, weight}
```

Column Attributes

Use a collection of message pairs for data table columns to control all of the various attributes or characteristics of a column, including its name, data, states, and metadata. The messages come in pairs, one to "set" or assign each attribute and one to "get" or query the current setting of each attribute.

For example, you can hide, exclude, label, and turn on or off the scroll lock for a column through scripting. The value is Boolean; enter a one to turn the column attribute on, and a zero to turn it off.

In the following examples, the name column is unhidden, unexcluded, labeled, and locked from horizontal scrolling.

```
Column( "name" ) << Hide( 0 );
Column( "name" ) << Exclude( 0 );
Column( "name" ) << Label( 1 );
Column( "name" ) << Set Scroll Locked( 1 );
```

Note: All the messages to set various arguments (for example, `Set Name`, `Set Values`, `Set Formula`) start with `Set`. The word `Set` is optional for all messages except `Set Name` (recall that `Name` is already used for something else, the command that lets you use unusual characters in a name). Use whichever form you prefer or find easier to remember. The corresponding messages to retrieve the current value of an argument (for example, `Get Formula`) are the same, except that they start with `Get` instead of `Set`, and the word `Get` is *not* optional.

To deselect all selected columns, send a `Clear Column Selection` message to the data table object.

```
dt << Clear Column Selection;
```

Set or Get a Column Name

`Set Name` lets you name or rename a column, and `Get Name` returns the name for a column. The following example changes the name of the second column from age to ratio. It then returns the current column name to the log.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = Column( 2 );
```

```
col << Set Name( "ratio" );
col << Get Name;
  "ratio"
```

Set or Get Column Values

Similarly, `Set Values` sets values for a column. If the variable is character, the argument should be a list. If the variable is numeric, the argument should be a matrix (vector). If the number of values is greater than the current number of rows, the necessary rows are added to the data table. `Get Values` returns the values in list or matrix form. `Get As Matrix` is similar to `Get Values` but returns values in the numeric columns.

```
col << Set Values( myMatrix ); // for a numeric variable
col << Set Values( myList ); // for a character variable
col << Get Values; // returns a matrix, or list if character
col << Get Matrix(<list of column names>|<list of column numbers>|<column
  range>; // returns the specified columns as a matrix
```

The following example returns a list and a matrix of values:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column( "name" ) << Values( {Fred, Wilma, Fred, Ethel, Fred, Lamont} );
myList = :name << Get Values;
  {"Fred", "Wilma", "Fred", "Ethel", "Fred", "Lamont", "JAMES", "ROBERT",
    "BARBARA", ...}

Column( "age" ) << Values( [28, 27, 51, 48, 60, 30] );
myVector = :age << Get Values;
  [28, 27, 51, ... ]
myMatrix = :weight << Get as Matrix;
  [95, 123, 74, ... ]
```

Set or Get Value Labels

Note: For more information about value labels, see *Using JMP*.

Value labels provide a method of displaying a descriptive label for abbreviated data. For example, you might have a column of 0 and 1 values, where 0 represents a male and 1 represents a female. The value label “male” for 0 and “female” for 1 are more readable.

You can specify value labels in any one of the following three ways. Using the Big Class.jmp sample data table, assume that M maps to Male, and F maps to Female.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
:sex << Value Labels( {"F", "M"}, {"Female", "Male"} ); // use two lists
:sex << Value Labels( {"F", "Female", "M", "Male"} ); // use a list of pairs
:sex << Value Labels( {"F" = "Female", "M" = "Male"} ); // use a list of
  assignments
```

You can activate value labels by sending `Use Value Labels` as a message to the column.

```
:sex << Use Value Labels( 1 );
```

To revert back to showing the column's actual values:

```
:sex << Use Value Labels( 0 );
```

The same message can be used for the data table to turn value labels on and off for all columns.

```
dt << Use Value Labels( 1 );
```

Set or Get Data and Modeling Types

You can set or get the data type of a column using JSL. The choices are character, numeric, and row state. The following example adds a new column that has a character data type to the Big Class.jmp sample data table.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "New" );
Column( "New" ) << Data Type( Character );
Column( "New" ) << Get Data Type;
   "Character"
```

To set or get the modeling type of a column:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = New Column( "New" );
col << Modeling Type( "Continuous" );
col << Get Modeling Type;
   "Continuous"
```

You can specify the format of a column when changing its data type:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = New Column( "Date" );
Column( "Date" ) << Data Type( Numeric, Format( "ddMonYYYY" ) );
```

Set or Get Formats

The `Format` message controls numeric and date/time formatting. The first argument is a quoted string from the list of format choices shown in the Column Info window. Subsequent arguments depend on the format choice. You can also set the field width by itself.

```
col << Format( "Best", 5 ); // width is 5
col << Format( "Fixed Dec", 9, 3 ); // width is 9, with 3 decimal places
col << Format( "PValue", 6 );
col << Format( "d/m/y", 10 );
col << Set Field Width( 30 );
```

For date formats, the `Format` message sets how dates appear in a data table column. To set the format that you use for entering data, or for displaying the current cell when you have it selected for entry or editing, use the `Input Format` message.

```
col << Format( "d/m/y", 10 ); // display the date in day-month-year order
col << Input Format( "m/d/y" ); // enter the date in month-day-year order
```

For more information about date/time formatting choices, see "Date-Time Functions and Formats".

Note: Do not confuse the `Format` *message* for columns with the `Format` *function* for converting numeric values to strings according to the format specified (typically used for date/time notation as described in "Date-Time Functions and Formats"). Sending a message to an object has a very different effect from using a function that might happen to have the same name.

To get the current format of a column, submit a `Get Format` message:

```
col << Get Format;
```

Display Numbers as Four Digits

The `Format()` function and the `Format` message have a custom format capability that enables you to provide your own script to do the formatting. The name of the format is "Custom" (rather than "Best", "Fixed", and so on) and the argument is a script that works with the variable named "value" to return a string to represent that value. To make positive or negative integers always have at least four digits by adding zeros, you could do this:

```
New Table( "Untitled",
  New Column( "Column 1",
    Numeric,
    "Continuous",
    Format(
      "Custom", // make a custom format
      Formula( // using a JSL formula
        Local( {sign, length = 4, result}, // local variables
          sign = If( value < 0, -1, 1 ); // handle negative numbers
          value = Char( Abs( value ) );
          If( Length( value ) > length, // handle too big
            value = Repeat( "#", length )
          );
          result = Right( value, length, "0" ); // add zeros
          If( sign < 0,
            result = "-" || result // re-attach sign
          );
          result; // return the modified value
        )
```

```
        )
      ),
      // sample data to test positive, negative, zero
      Set Values( [0, 3, -5, 33, -55, 333, -555, 3333, -5555, 33333, -55555] )
    )
);
```

See "Create Custom Functions, Transforms, and Formats" for more information about creating custom formats.

Specify the Significant Digits in a Numeric Column

Significant digits are the first nonzero digit and all succeeding digits. For example, 3.14159 has six significant digits, and 0.00314 has three significant digits.

Set the number of significant digits with the Precision column format:

```
Format( "Precision", 12, 3 ); // 12 wide with 3 significant digits
```

Set, Get, or Evaluate a Formula

The following examples show how to set, get, and evaluate a formula:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = New Column( "Ratio" ); // create column and stores its reference
col << Set Formula( :height/:weight ); // set the formula
col << Eval Formula; // evaluate the formula
col << Get Formula; // return the expression :height/:weight
```

To use the values from columns in scripts, be sure to add commands to evaluate the formula. Formula evaluation timing can differ between different versions of JMP.

Notes:

- When formulas are added, they are scheduled to be evaluated in a background task. This can be a problem for scripts if they depend on the column having the values while the script is running.
- To force a single column to evaluate, you can send an `Eval Formula` command to the column. You can do this inside the command to create the column, right after the Formula clause:

  ```
  dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
  dt << New Column( "Ratio",
     Numeric,
     Formula( :height / :weight ),
     Eval Formula
  );
  ```

  where `Formula()` is an alias for `Set Formula()`.

However, it is best to wait until you are finished adding a set of formulas, and then use the command `Run Formulas` to evaluate all of the formulas in their proper order, as follows:

```
dt << Run Formulas;
```

- The `Run Formulas` command is preferable to the `Eval Formula` command, because while it is evaluating the formulas, `Eval Formula` does not suppress the background task from evaluating them again. The formula dependency system background task takes great care to evaluate the formulas in the right order, and `RunFormulas` simply calls this task until all the formulas are finished evaluating.
- If you use random numbers and use the `Random Reset(`*`seed`*`)` feature to make a reproducible sequence, then you have another reason to use `Run Formulas`, in order to avoid a second evaluation in the background.
- If formula suppression is enabled with `Supress Eval`, turn it off to enable `Eval Formula` to run.
- The `Add Custom Function()` function enables you to create your own functions and display them in the Formula Editor's Functions list. See "Create Custom Functions, Transforms, and Formats".

Set and Get Range and List Checks

You can manipulate list and range check properties using JSL. The following examples use the Big Class.jmp sample data table.

Set and clear the list check property in the sex column:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column( "sex" ) << List Check( {"M", "F"} ); // set the property
Column( "sex" ) << List Check(); // clear the property
```

Range checks require the specification of a range using the syntax in Table 9.3.

Table 9.3 Range Check Syntax

To specify this range	Use this function
$a \leq x \leq b$	`LELE(a, b)`
$a \leq x < b$	`LELT(a,b)`
$a < x \leq b$	`LTLE(a,b)`
$a < x < b$	`LTLT(a,b)`

The following example specifies that the values in the age column must be greater than zero and less than 120:

```
Column( "age" ) << Range Check( LTLT( 0, 120 ) );
```

All of the functions can be preceded by `Not` and one of them can be missing. The following example specifies that the values in the age column should be greater than or equal to 12:

```
Column( "age" ) << Range Check( not( LT( 12 ) ) );
```

To clear a range check state, submit an empty `Range Check()`:

```
Column( "age" ) << Range Check();
```

To retrieve the list or range check assigned to a column, send a `Get List Check` or `Get Range Check` message to the column:

```
Column( "sex" ) << Get List Check;
Column( "age" ) << Get Range Check;
```

Here is an example of `Get Range Check` for the age column in Big Class.jmp:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column( "age" ) << Range Check( LTLT( 0, 120 ) );
Column( "age" ) << Get Range Check;
   Range Check( LTLT( 0, 120 ) )
```

Note that you can also use `Set Property`, `Get Property`, and `Delete Property` to set, retrieve, and remove list checks and range checks. See "Column Properties".

Note: Operations sent through JSL that involve range check columns show any warnings in the log rather than in interactive windows.

Get a Column Script

`Get Script` returns a script to create the column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Column( "Ratio", Set Formula( :height/ :weight) );
Column( "Ratio" ) << Get Script;
   New Column( "Ratio",
      Numeric,
      "Continuous",
      Format( "Best", 10 ),
      Formula( :height / :weight )
   );
```

Preselect Roles

To preselect a role on a column, use the `Preselect Role` message. Choices include No Role, X, Y, Weight, Freq, and Validation. The `Get Role` message returns the current setting.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = New Column( "New" );
col << Preselect Role( X );
col << Get Role;
```

Lock a Column or Cells

To lock or unlock a column, use `Lock` or `Set Lock` with a Boolean argument. `Get Lock` returns the current setting.

```
col << Lock( 1 ); // lock
col << Set Lock( 0 ); // unlock
col << Get Lock; // show current state
```

Column Properties

Tip: JSL choices for properties are the same as those in the **Column Properties** menu in the Column Info window. The arguments for each property correspond to the settings in the Column Info window. An easy way to learn the syntax is to establish the property that you want in the Column Info window first, and then use `Get Property` to view the JSL. See *Using JMP* for more information about each property.

Data columns have numerous optional metadata attributes that can be set, queried, or cleared using the messages `Get Property`, `Set Property`, and `Delete Property`.

```
col << Set Property( "propertyName", {argument list} );
col << Get Property( "propertyName" );
col << Delete Property( "propertyName" );
```

The name of the property in question is always the first argument for `Set Property`, and what is expected for subsequent arguments depends on which property you set:

- `Get Property` and `Delete Property` always take a single argument, which is the name of the property.
- `Get Property` returns the property's settings. `Delete Property` completely removes the property from the column.
- To get a list of all column property names for a column, specify the column and then use `Get Properties List`.

If you want to set several properties, you need to send several separate `Set Property` messages. You can stack several messages in a single JSL statement if you want.

```
col << Set Property( "Axis",{Min(50), Max(180)} )<< Set Property( "Notes", "to
  get proportions" );
```

To get a property's value, send a `Get Property` message whose argument is the name of the property that you want:

```
Column("ratio") << Get Property( "axis" ); // return axis settings
```

To set columns as label columns:

```
dt << Set Label Columns( col1, col2, col3 );
```

To clear all label columns:

```
dt << Set Label Columns();
```

The same syntax works for `Set Scroll Lock Columns`, and `Scroll Lock`.

Set Column Properties with Variables

When writing JSL to script a column property, it is important to understand that the column is a repository for all column properties. The column stores the column property with the values supplied without evaluation or validation of the supplied arguments. In order to add column properties that contains variables using JSL code, the script must convert any variables to their actual values.

One way to do this is to use expressions. In the following example, the `Eval Expr()` function evaluates and replaces any variable wrapped in the `Expr()` function with its value. The outer `Eval()` function evaluates the entire statement after the replacement has occurred.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
// assign desired spec limits
lLimit = 55;
uLimit = 70;
tLimit = 62.5;

Eval(
   Eval Expr(

      // store the limits in the Spec Limits column property
      :height << Set Property("Spec Limits",
         {LSL( Expr( lLimit ) ), USL( Expr( uLimit ) ), Target( Expr( tLimit )
   ),
         Show Limits( 0 )}
      )
   )
);
```

Set Value Order Column Properties

The Value Order column property specifies the order in which you want the data to appear in reports. Data can be sorted by numerical order, custom order, or by the order in which the data appear in the data table.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

// numerical ordering sorts character column values by splitting
the data into string and numeric segments. The following example turns off
numerical order, which you might specify if the column contains character
data. Hex data ("A1FE2", "A13E4", "BEEEE") sort differently with numerical
ordering on or off).
:age << Set Property( "Value Order", {Numerical Order( 0 )} );

// specifies a custom order for the age column
:age << Set Property( "Value Order", {Custom Order( {15, 17, 12, 13, 14, 16}
  )} );

// sorts "ELIZABETH" first, the remaining items are sorted in alphabetical
  order
:name << Set Property( "Value Order", {Custom Order( {"ELIZABETH"} )} )

// specifies that if all values have a predefined sort order
// (for example, the days of the week), the remaining items are
// sorted on that basis. This option is available only for character data.
:age << Set Property( "Value Order", {Common Order( 1 )} )

// sorts the column values by their occurrence in the data table,
// rather than sorting by value
:name << Set Property( "Value Order", {Row Order Levels( 1 )} );
```

When you set the column property, you are setting the entire property. Suppose that you specify the row order levels. If the data happen to match a Common Order (for example, the days of the week), then Common Order takes precedence. In that case, you need to also include `Common Order(0)` in the property; you cannot send separate `Set Property` messages.

Examples of Setting and Getting Popular Column Properties

Table 9.4 shows examples for setting and getting popular column properties. See *Using JMP* for more information about each property.

Table 9.4 Common Properties for Data Table Columns

Property	Description	Example Using Arguments

Notes	Stores notes about a column. Is a quoted text string.	`col<<Set Property( "Notes", "Extracted from Fisher iris data" );`
		
		`col<<Get Property( "Notes" );`

List Check
Range Check
	Prescribes the possible values that can be entered in a column.	`col<<Set Property( List Check,{"F","M"} );`
		
		`col<<Set Property( "Range Check",LTLT(0, 120) );`
		
		`col<<Get Property( "List Check" );`
		
		`col<<Delete Property( "Range Check" );`

Missing Value Codes	Specifies column values that should be treated as missing.	`col<<Set Property( "Missing Value Codes", {0, 1} );`

Value Labels	Specifies labels to be displayed in place of the values.	`col<<Value Labels( {0 = "Male", 1 = "Female"} );`
		
		Use Boolean values to turn value labels on or off:
		
		`col<<Use Value Labels( 1 );.`

Value Order	Specifies the order in which you want the data to appear in reports.	`col<<Set Property( "Value Order", {"Spring", "Summer", "Fall", "Winter"} );`

Table 9.4 Common Properties for Data Table Columns *(Continued)*

Property	Description	Example Using Arguments

`Value Colors and Color Gradient`	Specifies colors for either categorical or continuous data, respectively.	Specify the color by assigning a JMP number to each value:
		`col<<Set Property( "Value Colors", {"Female" = 3, "Male" = 5} );`
		Specify a color theme:
		`col<<Set Property( "Value Colors", Color Theme( "White to Blue" ) );`
		If you omit the color theme, JMP uses the color theme preferences.
		Specify the gradient and the range and midpoint for Value Gradient:
		`col<<Set Property( "Color Gradient", {"White to Blue", Range( {18, 60, 25} )} );`

`Axis`	Most platforms use this (if it exists) when constructing axes. Mostly Boolean.	`col<<Set Property( "Axis",{Min(50), Max(180), Inc(0), Minor Ticks(10), Show Major Ticks(1), Show Minor Ticks(1), Show Major Grid(0), Show Labels(1), Scale(Linear)} );`
`Units`	Provided for custom uses. Specify the units of measure.	`col<<Set Property( "units", "grams" );` `col<<Get Property( "units" );`
`Coding`	Used for DOE and fitting. List with low and high values.	`col<<Set Property( "Coding", {59,172} );` `col<<Get Property( "Coding" );`
`Mixture`	Used for DOE, fitting, and profiling. Specifies the Mixture column properties in a list.	`col<<Set Property( "Mixture", {0.2, 0.8, 1, L PseudoComponent Coding} );` `col<<Get Property( "Mixture" );`
`Row Order Levels`	Specify to sort levels by their occurrence in the data instead of by value.	`col<<Set Property( "Row Order Levels", 1 );`
`Spec Limits`	Used for capability analysis and variability charts.	`col<<Set Property( "Spec Limits", {LSL(-1), USL(1), Target(0)} );` `col<<Get Property( "Spec Limits" );`

Table 9.4 Common Properties for Data Table Columns *(Continued)*

Property	Description	Example Using Arguments

`Control Limits`	Used for control charts.	`col<<Set Property( "Control Limits", {XBar(Avg(44), LCL(29), UCL(69))} );`
		
		`col<<Get Property( "Control Limits" );`

`Detection Limits`	Set in DOE and used in generalized regression.	`col<<Set Property( "Detection Limits", {10, 50} );`

`Response Limits`	Set in DOE and used in desirability profiling.	`col<<Set Property( "Response Limits", {Goal("Match Target"), Lower(1,1), Middle(2,2), Upper(3,3)} );`
		
		`col<<Get Property("Response Limits");`
		
		Choices for `Goal` are `Maximize`, `Match Target`, `Minimize`, `None`. Other arguments take numeric value and desirability arguments.

`Design Role`	Used for DOE. Specify a single role.	`col<<Set Property( "Design Role", "Covariate" );`
		
		`col<<Get Property( "Design Role" );`
		
		Choices for role are `Continuous`, `Discrete Numeric`, `Categorical`, `Blocking`, `Covariate`, `Mixture`, `Constant`, `Uncontrolled`, `Random Block`, `Signal`, and `Noise`.

`Factor Changes`	Sets the difficulty of changing a factor (Easy, Hard, or Very Hard).	`col<<Set Property( "Factor Changes", Hard) ;`
		
		`col<<Get Property( "Factor Changes" );`

`Sigma`	Used for control charts. Specify known sigma value.	`col<<Set Property( "Sigma",1.332 );`
		
		`col<<Get Property( "Sigma" );`
		
		Each type of chart uses a different sigma calculation.

`Distribution`	Set the distribution type to fit to the column.	`col<<Set Property( "Distribution", Distribution( GLog ) );`
		
		`col<<Get Property( "Distribution" );`

Table 9.4 Common Properties for Data Table Columns *(Continued)*

Property	Description	Example Using Arguments

`Time Frequency`	Set the type of time frequency.	`col<<Set Property( "Time Frequency", Time Frequency( "Annual" ) );`
		
		`col<<Get Property( "Time Frequency" );`

`Map Role`	Set how the column is used to connect map shape data with name data. Specify the role and other information as necessary.	`col<<Set Property( "Map Role", Map Role( Shape Name Use( "filepath to data table", "column name" ) ) );`
		
		`col<<Get Property( "Map Role" );`

`Supercategories`	Groups specific categories into one category. Supported only in the Categorical platform.	`col<<Set Property( "Supercategories", {Group( "Genders", {"F", "M"} )} );`
		
		`col<<Get Property( "Supercategories" );`

`Multiple Response`	Specifies the character that separates the responses within a cell.	`col<<Set Property( "Multiple Response", Multiple Response( Separator( "." ) ) );`
		
		`col<<Get Property( "Multiple Response" ) );`
		
		Note: We recommend assigning the Multiple Response modeling type if the delimiter is a comma. Then you don't have to specify the delimiter in the column property. See *Using JMP*.

Table 9.4 Common Properties for Data Table Columns *(Continued)*

Property	Description	Example Using Arguments

`Profit Matrix`	Assigns weights that define a decision model.
	
	Specify a list that contains the weights in a matrix and categories in another list. In the matrix, each row contains weights that reflect the profit for predicting the category indicated by the column of the matrix.
	
	Caution: The matrix argument is the transpose of the profit matrix as displayed in the Column Properties window.
		`col<<Set Property( "Profit Matrix", {[5 -1, -1 4, -2 -2], {"F", "M", "Undecided"}} );`
		
		`col<<Get Property( "Profit Matrix" );`

`Expression Role`	Applies to columns that contain expressions. Specifies whether the expression should be interpreted as a Picture, a Matrix, or an Expression.	`col<<Set Property( "Expression Role", Expression Role( "Picture", MaxSize( 640, 480 ), StretchToMaxSize( 1 ), PreserveAspectRatio( 1 ), Frame( 0 ) )`

`[Custom property]`	Provided for custom uses. Corresponds to **Column Properties > Other** in the Column Info window.
	
	The first argument is a name for the custom property, and the second argument is an expression.
		`col<<Set Property( "Date recorded",12Dec1999 );`
		
		`long date( col<<Get Property( "Date recorded") );`

Specify Engineering SI Units

The Engineering SI format displays the value with an appropriate SI suffix. For example, 3,100 is displayed as 3.1k. 0.0055 is displayed as 5.5m.

Magnitude	Suffix	Magnitude	Suffix	Magnitude	Suffix	Magnitude	Suffix
103	k	1015	P	10-3	m	10-15	f
106	M	1018	E	10-6	µ	10-18	a
109	G	1021	Z	10-9	n	10-21	z
1012	T	1024	Y	10-12	p	10-24	y

Rows

This section describes the messages for adding and manipulating the rows in a data table. Row messages are directed to a data table reference, and most act on the currently selected rows. A number of row messages might not be practical in scripting (for example, `Move Rows`).

- "Add Rows"
- "Delete Rows"
- "Select Rows"
- "Find Rows"
- "Move Rows"
- "Assign Colors and Markers to Rows"
- "Color Cells"
- "Hide, Exclude, and Label Rows"
- "Iterate on Rows in a Table"
- "Prevent Editing Rows"
- "Row States"

Add Rows

To add rows, send an `Add Rows` message and specify how many rows. You can also specify after which row to insert the new rows. The arguments can either be numbers or expressions that evaluate to numbers.

```
dt << Add Rows( 3 ); // add 3 rows to bottom of data table
dt << Add Rows( 3, 10 ); /* add 3 rows after the 10th row, moving the 11th and
   lower rows farther down */
```

A variation of `Add Rows` lets you specify an argument yielding a list of assignments. Assignments can be separated with commas or semicolons.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Add Rows(
   {:name = "Peter", :age = 14, :sex = "M", :height = 61, :weight = 124}
);
```

or

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Add Rows(
   {:name = "Peter"; :age = 14; :sex = "M"; :height = 61; :weight = 124}
);
```

You can send several arguments yielding lists, or even a list of lists. The following script creates a data table with `Add Rows` commands of each variety:

```
dt = New Table( "Cities" );
dt << New Column( "xx", Numeric );
dt << New Column( "cc", Character, width( 12 ) );

dt << Add Rows( {xx = 12, cc = "Chicago"} ); // single list
dt << Add Rows( // several lists
   {xx = 13, cc = "New York"}, {xx = 14, cc = "Newark"} );
dt << Add Rows( // list of lists
   {{xx = 15, cc = "San Francisco"}, {xx = Sqrt( 256 ), cc = "Oakland"}}
);
a = {xx = 20, cc = "Miami"};
dt << Add Rows( a ); // evaluate as single list

b={{xx = 17, cc = "San Antonio"},{xx = 18, cc = "Houston"}, {xx = 19, cc =
   "Dallas"}};
dt << Add Rows( b ); // evaluate as list of lists
```

Further information for rows can be specified with messages described under "Row States".

Delete Rows

To delete rows, send a `Delete Rows` message and specify which row or rows to delete. To delete more than one row, give a list or matrix as the *rownum* argument, or combine `Delete Rows` with other commands such as `For`. The *rownum* argument can be a number, list of numbers, range of numbers, matrix, or an expression that yields one of these. Without an argument, `Delete Rows` returns the number of selected rows and deletes those rows. When an argument is not specified or rows are not selected, `Delete Rows` only returns the number of selected rows: 0.

```
dt << Delete Rows( 10 ); // delete row 10
```

```
dt << Delete Rows( {11, 12, 13} ); // delete rows 11-13
myList = {11, 12, 13};
dt << Delete Rows( mylist ); // delete rows 11-13
dt << Delete Rows( 1 :: 20 ); // delete first 20 rows
dt << Delete Rows( [1 2 3] );  // delete first 3 rows
```

For example, the following script opens Big Class.jmp and deletes row 10:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Selected( Row State( 10 ) ) = 1; // select row 10
dt << Delete Rows;  // delete row 10
```

You can list duplicate rows, and you can list rows in any order with no consequence.

Here is a general way to remove the bottom *x* rows of a data table of any size. The following example removes five rows from the bottom of the data table:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
x = 5;
n = N Row( dt );
For( i = n, i > n - x, i--,
   dt << Delete Rows( i )
);
```

`NRow` counts the rows in the table. See "Iterate a Script on Each Row".

Select Rows

`Select All Rows` selects (or highlights) all of the rows in a data table.

```
dt << Select All Rows;
```

If all rows are selected, you can deselect them all by using `Invert Row Selection`. This command reverses the selection state for each row, so that any selected rows are deselected, and any deselected rows are selected.

```
dt << Invert Row Selection;
```

Note: With the exception of `Invert Row Selection`, whose result depends on the current selection, any new selection message starts over with a new selection. If you already have certain rows selected and you then send a new message to select rows, all rows are first deselected.

To select a specific row, use `Go To Row`:

```
dt << Go To Row( 9 );
```

To select specific rows in a data table based on their row number, use the `Select Rows` command. The argument to the command is a list of row numbers. For example, to select rows 1, 3, 5, and 7 of a data table:

```
dt << Select Rows( {1, 3, 5, 7} );
```

To select a range of rows, specify one of the following messages:

```
dt << Select Rows( Index( 7, 10 ) );
dt << Select Where( Any( Row() == Index( 7, 10 ) ) );
```

Both of these examples select rows seven through 10 in the current data table.

Select Where

To select rows according to data values, use `Select Where`, specifying a logical test inside the parentheses.

Tip: For a description of the functions and operators that you can use within a `Select Where` message, see "Operators".

For example, using the Big Class.jmp sample data table, select the rows where the students' age is greater than 13:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age > 13 );
```

Or, select the rows where the students' ages are less than 14:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
col = Column( dt, 2 );
dt << Select Where( col[] < 14 );
```

The following example selects the rows where the student's ages are less than 15 and the sex is "F":

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age < 15 & :sex == "F" );
```

To select a row without deselecting a previously selected row, combine `<< Select Where` with `<< Select Where` and the `Current Selection("extend")` argument. This is an alternative to using an `OR` statement.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age == 14 );
dt << Select Where( :sex == "F", Current Selection( "extend" ) );
```

To select rows that contain specific strings, follow this example:

```
dt = Open( "$SAMPLE_DATA/Hollywood Movies.jmp" );
my_genres = {"Romance", "Comedy"};
dt << Select Where( Contains( my_genres, :Genre ) );
```

You can also create more complex conditions by using regular expressions.

```
// select rows where :name begins with "j" or "J".
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
/* \0 is a group reference that causes the expression to return the entire
   matched text. !Is Missing is needed because the regex returns the matched
   text if it successfully matched or missing if it failed to match. So !Is
   Missing means the string successfully matched */
dt << Select Where( !Is Missing( Regex( :name, "^j", "\0", IGNORECASE ) ) );

// select rows where :name ends with "N" or "E".
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( !Is Missing( Regex( :name, "(N|E)$", "\0" ) ) );
```

Select Excluded, Hidden, or Labeled Rows

To select rows that are currently excluded, hidden, or labeled:

```
dt << Select Excluded;
dt << Select Hidden;
dt << Select Labeled;
```

To select rows that are *not* excluded, hidden, or labeled, stack a select message and an invert selection message together in the same statement, or send the two messages sequentially:

```
dt << Select Hidden << Invert Row Selection;
dt << Select Hidden;
dt << Invert Row Selection;
```

Select Duplicate Rows

To select duplicate rows in a data table, use the `Select Duplicate Rows` message:

```
dt = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
dt << Select Duplicate Rows(); // selects rows 301 and 8864
```

The second and subsequent duplicate rows are selected. Note that the duplicate values are case sensitive.

Note: By default, if no matching column is specified, all of the columns are matched.

In the preceding example, the duplicate rows might have been data input errors. Exclude the duplicate rows to remove them from the analysis:

```
dt = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
dt << Select Duplicate Rows();
dt << Exclude();
```

To select duplicate values in the rows of selected columns, specify the column names. The following example finds duplicate values in the rows of the Incident Number and Time columns.

```
dt = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
dt << Select Duplicate Rows( Match( Column( "Incident Number" ), :Time ) );
```

To select duplicate rows according to data values, select the duplicate rows, specify the data values, and then restrict the duplicate rows to the current selection.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Duplicate Rows( Match( :age ) );
dt << Select Where( :age > 15, Current Selection( "restrict" ) );
```

Refer to a Specific Cell in the Row

To refer to a specific cell, assign a subscript to the cell's row number. In the following example, the subscript `[1]` is used with the weight column. The formula then calculates the ratio between each height and the first value in the weight column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Column( "ratio", Formula( height / weight[1] ) );
```

Select Random Rows

To obtain a random selection, use this syntax:

```
dt << Select Randomly( number );
dt << Select Randomly( Sample Size( number ) );
dt << Select Randomly( probability );
dt << Select Randomly( Sampling Rate( probability ) );
```

These commands use a conditional probability to obtain the exact count requested.

Select Matching Cells

The row menu command `Select Matching Cells` is also implemented in JSL.

```
// select matching cells in the current data table
dt << Select Matching Cells;
// select matching cells in all open data tables
dt << Select All Matching Cells;
```

For more complicated selections, or to store selections permanently as row state data, see "Row States".

Find Rows

`Get Rows` returns a matrix of rows that match the specified condition. The following example select rows in which the age is greater than or equal to 16:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
```

```
dt << Get Rows Where( :age >= 16 );
   [35, 36, 37, 38, 39, 40]
```

`Get Selected Rows` returns a matrix of the currently selected rows. The following example selects rows 1, 3, 5, and 7 and then returns the row numbers in a matrix:

```
dt << Select Rows( {1, 3, 5, 7} );
dt << Get Selected Rows;
   [1, 3, 5, 7]
```

Find Selected Rows

`Next Selected` and `Previous Selected` scroll the data table window up or down so that the next selected row that is not already in view moves into view. The table wraps, so `Next Selected` jumps from the bottom-most selected row to the top-most, and *vice versa* for `Previous Selected`.

```
dt << Next Selected;
dt << Previous Selected;
```

Clear Selected Rows

To cancel a selection, leaving no rows selected, use `Clear Select`:

```
dt << Clear Select;
```

Move Rows

These commands move the currently selected rows to the indicated destination point.

```
dt << Move Rows( At Start );
dt << Move Rows( At End );
dt << Move Rows( After( rowNumber ) );
```

Assign Colors and Markers to Rows

You can use the `Colors` and `Markers` messages to assign (or change) colors and markers used for rows. These settings mostly affect graphs produced from the data table. Both messages expect numeric arguments to choose which color or marker to use. For more information about how numbers correspond to colors and markers, see "Colors and Markers".

```
dt << Colors( 3 );  // set selected rows to red
dt << Markers( 2 ); // pick the X marker for selected rows
```

As with other row messages, you can stack selection and other messages together:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age == 12 ) // select the youngest subjects
   << Colors( 8 ) << Markers( 8 ); // and use purple open circles for them
```

`Color by Column` sets colors according to the values of a column that you specify, and `Marker by Column` works similarly:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Color by Column( :age );
dt << Marker by Column( :age );
```

Here are additional, named arguments:

- `Continuous Scale` (`Color by Column` only) Assigns colors in a chromatic sequential fashion based on the values in the highlighted column.
- `Reverse Scale` Reverses the color scheme in use.
- `Make Window with Legend` Creates a separate window with a legend.
- `Excluded Rows` Applies the row states to excluded rows.
- `Marker Theme` Specifies the marker type.
- `Color Theme` Specifies the color theme.

Color Cells

You can color individual cells in the data grid. For example, this line uses the row state color to color the cells:

```
dt << Color Rows by Row State;
```

You can also specify a color theme for either categorical or continuous columns:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
:height << Set Property(
  "Color Gradient",
  {"White to Blue", Range( 40, 80 )}
);
:height << Color Cell By Value( 1 ); // turn on the cell coloring

:age << Set Property(
  "Value Colors", // assign the value colors
  {12 = Red, 13 = Yellow, 14 = Green, 15 = Blue, 16 = Magenta, 17 =
  Gray}
);

:age << Color Cell By Value( 0 ); // turn off the cell coloring
```

You can also color specific cells. The following example sets rows 1, 5, 8 of the “name” column to red:

```
:name << Color Cells( red, {1, 5, 8} );
```

To remove the color from specific cells, set the color to black. The following example removes the color for row 1 of the “name” column.

```
:name << Color Cells( black, {1} );
```

You can color cells based on specific values. The following example colors the cells in the height column. All cells containing a value greater than 60 are blue. All cells containing a value equal to or less than 60 are purple.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For( i = 1, i <= N Row( dt ), i++,
   If( Column( dt, "height" )[i] > 60,
      Column( dt, "height" ) << Color Cells( 5, {i} ),
      Column( dt, "height" ) << Color Cells( 8, {i} )
   )
);
```

Note: The first argument for `Color Cells` represents the color value. The second argument contains the row numbers.

Hide, Exclude, and Label Rows

Note: For more information about hiding, excluding, and labeling rows using row state functions, see “Row States”.

Use the `Hide`, `Exclude`, and `Label` messages to hide, exclude, and label rows. These messages are toggles, meaning that to turn the messages on you send them once, and to turn them off you send the message a second time. They also accept Boolean arguments to explicitly turn them off and on.

For example, to hide all rows in Big Class where age is greater than 13, you could do the following:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age > 13 );
dt << Hide( 1 );
```

Messages to the same object can be stacked together in a single statement, so you could simplify the preceding script.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age > 13 ) << Hide( 1 );
```

Iterate on Rows in a Table

In addition to built-in programming functions for iterating, JSL provides functions for iterating through data table rows, groups, or conditional selections of rows.

Generally, an expression is executed on the current row of the data table only. Some exceptions are the expressions inside formula columns, `Summarize` and the pre-evaluated statistics functions, and any use of data table columns by analysis platforms.

Set the Current Row

Note: The current row for scripting is *not* related to rows being selected (or highlighted) in the data table or to the current cursor position in the data table window. The current row for scripting is defined to be zero (no row) by default.

You can set the current row for a script using `For Each Row` or `Row() = X`.

```
Row() = 3; ...
For Each Row( ... );
```

`For Each Row` executes the script once for each row of the current data table. Note that `Row()=1` only lasts for the duration of the script, then `Row()` reverts to its default value, which is zero. This means that submitting a script all at once can produce different results than submitting a script a few lines at a time.

Throughout this chapter, examples without an explicit current row should be assumed to take place within a context that establishes a current row. See "What is the Current Row?".

What is the Current Row?

By default, the current row number is 0. The first row in a table is row 1, so row 0 is essentially not a row. In other words, *by default, an operation is done on no rows*. Unless you take action to set a current row or to specify some set of rows, you get missing values due to the lack of data. For example, a column name returns the value of that column on the current row. Scope the column name with the prefix `:` operator to avoid ambiguity (to force the name to be interpreted as a column name).

```
:sex; // returns ""
:age; // returns .
```

Scoping names prevents you from getting a result that might look reasonable for the whole data table but is actually based on only one row. It also protects you from accidentally overwriting data values when making assignments to ambiguous names under most circumstances. You can have even more complete protection by using the prefix or infix `:` operator to refer specifically to a data column and the prefix :: operator to refer specifically to a global script variable. See "Advanced Scoping and Namespaces".

You can use the Row() function to get or set the current row number. Row() is an example of an *L-value* expression in JSL: a function that returns its value unless you place it before an assignment operator (=, +=, and so on.) to set its value.

```
Row(); // returns the number of the current row (0 by default)
x = Row(); // store the current row number in x
Row() = 7; // make the 7th row current
Row() = 7; :age; // make the 7th row current and returns 12
```

Note that the current row setting only lasts for the portion of a script that you select and submit. After the script executes, the current row setting resets to the default (row 0, or no row). Therefore, a script submitted all at once can produce different results from the same script submitted a few lines at a time.

How Many Rows and Columns?

The N Rows() and N Cols() functions return the rows and columns in a data table.

```
N Rows( dt ); // number of rows
N Cols( dt ); // number of columns
```

N Rows and N Cols also count the number of rows in matrices. Note that NRow and NCol are synonyms. See the "Inquiry Functions".

Iterate a Script on Each Row

To iterate a script on each row of the current data table, put For Each Row around the script.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( If( :age > 15, Show( :age ) ) );
```

To specify the open data table, include a data table reference as the first argument.

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt2 = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
For Each Row( dt1, If( :age > 15, Show( :age ) ) );
```

You can use For Each Row to set row states instead of creating a new formula column in the data table. The scripts below are similar, except that the first one creates a row state column, and the For Each Row script simply sets the row state without creating a column.

```
New Column( "My Row State", Row State, Formula( Color State( :age - 9 ) ) );
For Each Row( Color of( Row State() ) = :age - 9 );
```

To iterate a script on each row that meets a specified condition, combine For Each Row and If:

```
For Each Row( Marker of( Row State() ) = If( :sex == "F", 2, 6 ) );
```

You can use Break and Continue to control the execution of a For Each Row loop. See "Break and Continue".

Return Row Values

`Dif()` and `Lag()` are special functions that can be useful for statistical computations, particularly when working with time series or cumulative data.

- `Lag()` returns the value of a column *n* rows before the current row.
- `Dif()` returns the difference between the value in the current row and the value *n* rows previous.

The following lines are equivalent:

```
dt  <<  New Column( "htDelta" );
For Each Row( :htDelta = :height - Lag( :height, 1 ) );
For Each Row( :htDelta = Dif( :height, 1 ) );
```

Add Sequence Data

`Sequence()` corresponds to the Sequence function in the Formula Editor and is used to fill the cells in a data table column. It takes four arguments and the last two are optional:

```
Sequence( from, to, stepsize, repeat );
```

`From` and `to` are not optional. They specify the range of values to place into the cells. If `from = 4` and `to = 8`, the cells are filled with the values 4, 5, 6, 7, 8, 4, ...

`Stepsize` is optional. If you do not specify a stepsize, the default value is 1. Stepsize increments the values in the range. If `stepsize = 2` with the above `from` and `to` values, the cells are filled with the values 4, 6, 8, 4, 6, ...

`Repeat` is optional. If you do not specify a Repeat, the default value is 1. Repeat specifies how many times each value is repeated before incrementing to the next value. If `repeat = 3` with the above `from`, `to`, and `stepsize` values, the cells are filled with the values 4, 4, 4, 6, 6, 6, 8, 8, 8, 4, ... If you specify a `Repeat` value, you must also specify a `Stepsize` value.

The sequence is always repeated until each cell in the column is filled.

```
dt = New Table( "Sequence Example" ); // create a new data table
dt << New Column( "Count to Five" ); // add two columns
dt << New Column( "Count to Seventeen by Fours" );
dt << Add Rows( 50 ); // add fifty rows

/* fill the first column with the data sequence 1, 2, 3, 4, 5, ...
fill the second column with the data sequence 1, 1, 5, 5, 9, 9, 13, 13, 17,
  17, ... */
For Each Row (
  Column( 1 )[ ] = Sequence( 1, 5 );
  Column( 2 )[ ] = Sequence( 1, 17, 4, 2 );
);
```

Because `Sequence()` is a formula function, you can also set a column's formula to use `Sequence()` to fill the column. The following example creates a new column named Formula Sequence and adds a formula to it. The formula is a sequence that fills the column with values between 25 and 29, incremented by 1, and repeated twice (25, 25, 26, 26, 27, 27, 28, 28, 29, 29, 25, ...).

```
dt = New Table( "Formula Sequence Example" );
dt << Add Rows( 20 );
dt << New Column( "Formula Sequence", Formula( Sequence( 25, 29, 1, 2 ) ) );
```

The following are more examples of `Sequence()` results:

- `Sequence(1,5)` produces 1,2,3,4,5,1,2,3,4,5,1, ...
- `Sequence(1,5,1,2)` produces 1,1,2,2,3,3,4,4,5,5,1,1, ...
- `Sequence(10,50,10)` produces 10,20,30,40,50,10, ...
- `10*Sequence(1,5,1)` also produces 10,20,30,40,50,10, ...
- `Sequence(1,6,2)` produces: 1,3,5,1,3,5, ... The limit is never reached exactly.

Note: If you want a matrix of values, then use the `Index` function, not `Sequence`.

Prevent Editing Rows

To prevent adding or deleting rows, use `Set Edit Lock`:

```
dt << Set Edit Lock( "add rows", "delete rows");
```

See "Prevent Changes to a Data Table" for more information about preventing other data table operations.

Row States

There is a special data element type called a row state that stores various attributes in the data table. Row states can indicate the following:

- Whether a row is selected, excluded, hidden, or labeled
- Which marker type, color, shade, and hue to use for graphs

In JSL, you can use row state functions to manipulate row states.

About Row States

Row states change how JMP works with your data. Table 9.5 explains each row state. Remember that you can use several row states at once to get the combination of effects that you want.

Table 9.5 Row States

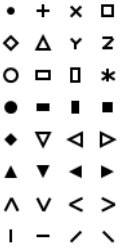
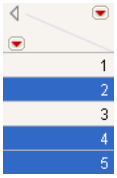
Row states	How they affect results
Excluded	If rows are excluded, JMP omits them from calculations for statistical analyses (text reports and charts). Results are the same as if the data was not entered. However, points are still included in *plots*. (To omit points from plots, use `Hide`. To omit points from all results, use both `Exclude` and `Hide`.)
Hidden	If rows are hidden, JMP does not show them in plots. However, the rows *are* still included in text reports and *charts*. (To omit points from reports and charts, use `Exclude`. To omit points from all results, use both `Exclude` and `Hide`.)
Labeled	If rows are labeled, JMP places row number labels, or the values from a designated Label column, on points in scatterplots.
Color	If rows have colors, JMP uses those colors to distinguish the points in scatterplots.
Marker 	If rows have markers, JMP uses those markers to distinguish the points in scatterplots.

Table 9.5 Row States *(Continued)*

Row states	How they affect results
Selected 	If rows are selected, JMP highlights the corresponding points and bars in plots and charts.

Note: For row state messages, omitting the argument *toggles* the option: if the option is off, the message turns it on, and if the option is on, the message turns it off. You can also use the `"toggle"`, `"switch"`, or `"flip"` argument, as in `r << Exclude( "toggle" )`.

About Row State Functions

When you give a row state function a number as its argument (or an expression that evaluates to a number), the function interprets that number as an index to its possible values.

Table 9.6 shows a comparison chart of the different row state functions, so that you can see which functions convert row states to numbers and numbers to row states. It also includes the numbers that you can use with each function.

Table 9.6 Row State Functions

Numbers	Convert from numbers to row states →	Row states	Convert from row states to numbers →	Numbers
1 or 0	`Excluded State(`*`n`*`)`	Excluded	`Excluded(`*`rowstate`*`)`	1 or 0
1 or 0	`Hidden State(`*`n`*`)`	Hidden	`Hidden(`*`rowstate`*`)`	1 or 0

Table 9.6 Row State Functions *(Continued)*

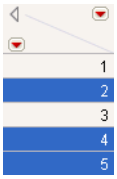
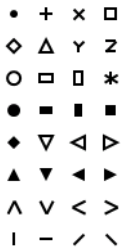
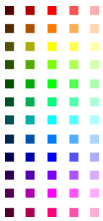
Numbers	Convert from numbers to row states →	Row states	Convert from row states to numbers →	Numbers
1 or 0	`Labeled State(n)`	Labeled	`Labeled(rowstate)`	1 or 0
1 or 0	`Selected State(n)`	Selected 	`Selected(rowstate)`	1 or 0
0 to 31	`Marker State(n)`	Marker 	`Marker Of(rowstate)`	0 to 31
0 to 84 (0–15 basics, 16–31 dark, 32–47 light, 48–63 very dark, 64–79 very light, 80–84 grays)	`Color State(n)`	Color	`Color Of(rowstate)`	0 to 84 (0–15 basics, 16–31 dark, 32–47 light, 48–63 very dark, 64–79 very light, 80–84 grays)
0–11 (rainbow order)	`Hue State(n)`	Hue		

Table 9.6 Row State Functions *(Continued)*

Numbers	Convert from numbers to row states →	Row states	Convert from row states to numbers →	Numbers
–2 to 2 (dark to light)	`Shade State(n)`	Shade 		

Assign Row States

To assign a row state using JSL, use `Select Where` to indicate which rows are affected, and then specify which row state to assign to the rows. In the following example, marker type 5 (a triangle) is assigned to rows in which sex is “F”.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :sex == "F" ) << Markers( 5 );
```

You can also use a formula to assign row states. Here is how you would do it in the data table:

1. Create a row state column.
2. Add a formula to the column that assigns a marker to each row in which sex is “F”.
3. In the Column panel, right-click the star next to the row state column and select **Copy to Row States**.

The JSL equivalent would be something like this:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Row State",
   Row State,
   Set Formula( If( :sex == "F", Marker State( 4 ) ) )
);
Column( "Row State" ) << Copy To Row States();
```

Store Row State Information

A row state column is a dedicated column that *stores* row state information, but does not put the information into effect. You can then use `For Each Row` to put the row state column into effect. For more information about row state columns, see *Using JMP*.

The following example creates a row state column then puts the row states into effect:

1. Submit this line to start a new data table:

```
dt = New Table( "Row State Testing" );
```

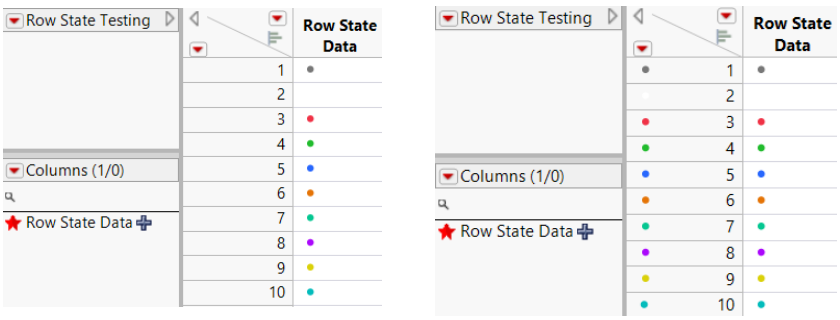
2. Submit these lines to add row states to a column and add ten rows:

```
dt << New Column( "Row State Data", Row State, Set Formula( Color State( Row()
   ) ) );
dt << Add Rows( 10 );
```

3. Submit this line to put the row states into effect:

```
For Each Row( Row State() = :Row State Data );
```

Figure 9.17 Table with No Row States (left) and Table with Row States (right)



This action replaces any row states in effect with the row state combination from the row state column. For more information about how to change selected attributes of a row state without changing or canceling others, see "Set One Characteristic and Cancel Others".

Set or Get Row States

From JSL, you can set or get row states directly using the `Row State` function. You can set or get the state for row *n* with `Row State(n)`. If you do not supply an argument, you set or get the current row. To work with all rows, either use a `For Each Row` loop or work with formula columns.

To set a row state, place the row state expression on the left of an assignment (L-value):

```
Row State( 1 ) = Color State( 3 ); // make row 1 red
Row() = 8; Row State() = Color State( 3 ); // make the 8th row red
For Each Row( Row State() = Color State( 3 ) ); // make each row red
```

To specify the data table, include a data table reference as the first argument:

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt2 = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
Row State( dt1, 1 ) = Color State( 3 ); // make row 1 red;
```

To get a row state, place the row state expression on the right side of an assignment:

```
x = Row State( 5 );  // put the row state of row 5 into x
x = Row State();   // row state of current row
```

Notes on Setting Row States

Be careful whether you set every aspect of `Row State()` or just one aspect of it, such as `Color Of( Row State() )`. To see how this works, first color and mark all the rows:

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row(
  Row State() = Combine States( Color State( Row() ), Marker State( Row() ) )
  );
```

And now observe the difference between setting one attribute of a row state:

```
Color Of( Row State( 1 ) ) = 3; // make row 1 red without changing marker
```

And setting every aspect of a row state to a single state:

```
Row State( 1 ) = Color State( 5 ); // make row 1 blue and remove its marker
```

To copy all the current row states into a row state column:

```
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row(
       dt1,
       Row State() = Combine States(
              Color State( Row() ),
              Marker State( Row() )
       )
);
dt1 << New Column( "rscol", "Row State", Set Formula( Row State() ) );
```

To copy several but not all of the current row states into a row state column, use a script like the following (commenting out or omitting any states that you do not want):

```
New Column( "rscol2",
  Set Formula(
     Combine States(
        Color State( Color Of() ),
        Excluded State( Excluded() ),
        Hidden State( Hidden() ),
        Labeled State( Labeled() ),
        Marker State( Marker Of() ),
```

```
            Selected State( Selected() )
        )
    )
);
```

To set a component of a row state:

```
Color Of( Row State( i ) ) = 3; // change color to red for row i
Selected( Row State( i ) ) = 1; // select the ith row state and set to 1
```

You can see from the example above that some of the functions convert numbers into states, and others convert states into numbers. Here are some helpful hints for remembering which are which:

Number-to-state functions have the word "State" Functions that take number arguments and either return states or accept state assignments all have the word "State" in their names: `Row State`, `As Row State`, `Color State`, `Combine States`, `Excluded State`, `Hidden State`, `Hue State`, `Labeled State`, `Marker State`, `Selected State`, `Shade State`.

State-to-number functions are one word or have the word "Of" Functions that take row state arguments (and assume that the argument `Row State()` if none is given) and functions that return or are set to numbers are either one word, or their second word is "Of": `Color Of`, `Excluded`, `Hidden`, `Labeled`, `Marker Of`, `Selected`.

Table 9.6 is a helpful comparison chart for these functions.

The following lines are equivalent to their interactive commands:

```
Copy From Row States
Add From Row States
Copy To Row States
Add To Row States
```

Examples of Getting Row States

For example, create the following table containing row states:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Row State Data", Row State, Set Formula( Color State( Row()
  ) ) );
dt << Add Rows( 10 );
For Each Row( Row State() = :Row State Data );
```

To get row states, submit the following lines:

```
Row State( 1 ); // returns row state for row 1
Row() = 8; Row State();  // returns row state for current (8th) row
For Each Row( Print( Row State()) ); // returns the row states for each row
```

To get a row state and store it in a global, place `Row State()` on the right side of an assignment. Add the following lines to the preceding script:

```
::x = Row State( 1 ); // put the row state of row 1 in x, a global
Row() = 8; ::x = Row State();  // put the row state of the 8th row in x
Show( x ); // returns x = Color State( 8 )
dt << New Column( "rscol", Row State) ;
// put row states in rscol (a row state column)
For Each Row( :rscol = Row State() );
```

To get a component of a row state, place the row state expression on the right side of an assignment and also use one of the L-value operators:

```
x = Selected( Row State() ); // selection index of current row selected
```

Get and Set Multiple Characteristics at Once

You can get or set many characteristics at once by combining state settings inside `Combine States()`. You can also get or set each characteristic one at a time, the ultimate row state being the accumulation of characteristics. The following example sets green Y markers for males, but hides them in plots for now. It also sets red X markers for females but shows them in plots.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row(
  If( sex == "M",
     /* then */ Row State() = Combine States(
        Color State( 4 ), Marker State( 6 ), Hidden State( 1 ) ),
     /* else */ Row State() = Combine States(
        Color State( 3 ), Marker State( 2 ), Hidden State( 0 ) ) ) );
```

Get the row state for one row, such as the 6th:

```
Row State( 6 );
  Combine States(Hidden State(1), Color State(4), Marker State(6))
```

Notice that JMP returns a `Combine State()` combination. This is because a row state datum is not just the state of one characteristic, such as color, but the cumulative state of *all* the characteristics that have been set: exclusion, hiding, labeling, selection, markers, colors, hues, and shades. A list of such characteristics is called a *row state combination*.

Just as there can be many row state characteristics in effect, a row state *column* can have multiple characteristic row states as its values.

Set One Characteristic and Cancel Others

In addition to the overall `Row State()` function for getting or setting all the characteristics of a row state, there are separate functions to get or set one characteristic at a time preemptively. That is, to give a row one characteristic, canceling any other characteristics that might be in effect. The functions that set one characteristic and cancel others are `Color State`, `Combine States`, `Excluded State`, `Hidden State`, `Hue State`, `Labeled State`, `Marker State`, `Selected State`, `Shade State`.

For example, to make row 4 be hidden only:

```
Row State( 4 ) = Hidden State( 1 );
```

Set or Get One Characteristic at a Time

A row state is not just one characteristic, but many. To work with just one characteristic at a time, use one of the L-value operators with `Row State` on either side of the equal sign. The side of the equal sign depends on whether you want to get or set a characteristic. There is an L-value operator for each of the following characteristics: `Color Of`, `Excluded`, `Hidden`, `Labeled`, `Marker Of`, `Selected`.

This example hides row 4 without affecting any other characteristics:

```
Hidden( Row State( 4 ) ) = 1
```

This example stores the color of row 3 without getting any other characteristics:

```
::color = Color Of( Row State( 3 ) );
```

Identify Row State Changes

The `MakeRowStateHandler` message (sent to a data table object) obtains a callback when the row states change. For example,

```
f = Function( {X}, Show( x ) );
obj = dt << Make Row State Handler( f );
```

Then when you select a group of rows, the row numbers of any row whose row state changed are sent to the log. For example:

```
x:[3, 4, 28, 40, 41]
```

When a group is highlighted, it might call the handler twice, once for rows whose selection is cleared, then again for the new selection.

Exclude, Hide, Label, and Select

This section discusses conditions that have Boolean states (either on or off). These conditions include: excluding, hiding, labeling, and selecting rows.

- `Excluded` gets or sets an excluded index. The index is 1 for true or 0 for false, indicating whether each row is excluded.
- `Hidden` gets or sets a hidden index, which is 1 for hidden or 0 for not hidden.
- `Labeled` gets or sets a labeled index, which is 1 for labeled or 0 for not labeled.
- `Selected` gets or sets a selected index, which is 1 for selected or 0 for not selected.

The following examples illustrate these conditions:

```
Excluded( Row State() ); // returns 1 if current row is excluded, 0 if not
Hidden();                // returns 1 if current row is hidden, 0 if not
Labeled( Row State() );  // returns 1 if current row is labeled, 0 if not
Selected();              // returns 1 if current row is selected, 0 if not

Excluded( Row State() ) = 1; // exclude current row
Hidden() = 0;                // unhide current row
Labeled( Row State() ) = 1;  // label current row
Selected() = 0;              // deselect current row
```

Remember that these functions assume the argument `Row State()` if none is given.

`Excluded State`, `Hidden State`, `Labeled State`, and `Selected State` do the reverse; they get or set a row state condition as true or false according to the argument. Nonzero values set the row state to true, and zero values set it to false. Missing values result in no change of state.

```
Row State() = Excluded State( 1 ); // change current row state to excluded
Row State() = Hidden State( 0 );   // change current row state to not hidden
Row State() = Labeled State( 1 );  // change current row state to labeled
Row State() = Selected State( 0 );
// change current row state to not selected
```

Notice that the first two expressions above replace the row state with just the exclusion or just the unhiding, so that any preexisting row state characteristics are lost. More commonly, you would issue the `state` commands for all of the characteristics that you want inside a `Combine States`. The following example changes the third row to hidden and a green square marker:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Row State( 3 ) = Combine States(
  Color State( 4 ),
  Marker State( 3 ),
  Hidden State( 1 ) );
```

Another common way to use a `-State` command would be in a row state data column whose values could be added to the row state (for cumulative characteristics). The following example excludes each odd numbered row:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "myExcl",
  Row State,
```

```
    Set Formula( Excluded State( Modulo( Row(), 2 ) ) )
);
For Each Row( Row State() = Combine States( :myExcl, Row State() ) );
```

Clear Row States

`Clear Row States()` removes row states from the data table. The following script assigns a row state to each row and then removes the row state.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( Row State() = Color State( Row() ) );
Wait( 3 ); // pause for demonstration purposes
dt << Clear Row States();
```

To clear row states for selected rows, select the rows first and then clear the selected row states.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
For Each Row( Row State() = Color State( Row() ) );
dt << Select Rows( [ 5 ] );
Wait( 3 ); // pause for demonstration purposes
dt << Clear Selected Row States();
```

Colors and Markers

This section discusses conditions that have many choices. These conditions include: coloring, adding markers, different hues, or different shades to rows.

`Color Of()` returns or sets the color index. The color index is a number from the JMP color map that corresponds to the *`row state`*, or a missing value if there is no assigned color.

```
Color Of( Row State() );   // returns color index for current row
Color Of() = 4;           // set current row to Color 4
```

Similarly, `Marker Of` returns or sets the marker index. The marker index is a number from the JMP marker map that corresponds to the active marker, or a missing value if there is no assigned marker.

```
Marker Of();                  // returns marker index for current row
Marker Of( Row State() ) = 4; // set current row to Marker 4
```

Both `Color Of()` and `Marker Of()` accept any row state expression or column or `Row State()` as arguments. They also assume the argument `Row State()` if none is given (some examples are shown with, and some without).

`Color State()` and `Marker State()` are similar to `Color Of()` and `Marker Of()`, except they work in the opposite direction. Where the `-Of` functions turn actual states into indices, the `-State` functions turn indices into states.

```
Row State() = Color State( 4 );  // change current row to green
```

```
Row State() = Marker State( 4 ); // change current row to the diamond marker
```

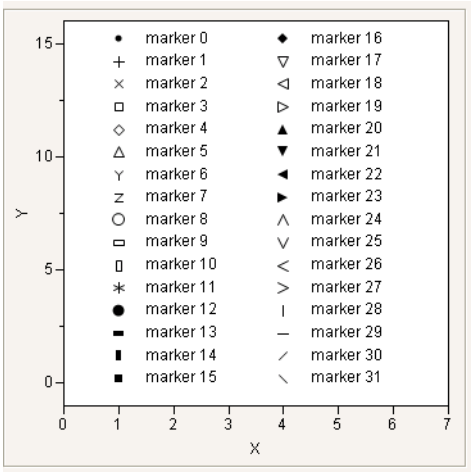
Notice that the last two commands replace the row state with just the color or just the marker, so that any preexisting row state characteristics are lost. More commonly you would issue the `-State` commands for all the characteristics that you want inside a `Combine States()`. The following example puts a green square marker in the third row and hides the row:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Row State( 3 ) = Combine States(
   Color State( 4 ),
   Marker State( 3 ),
   Hidden State( 1 ) );
```

The following script shows the standard JMP markers, which are numbered 0–31. Indices outside the range 0–31 have undefined behavior.

```
New Window( "Markers",
   Graph Box(
      FrameSize( 300, 300 ),
      Y Scale( -1, 16 ),
      X Scale( 0, 7 ),
      For(
         i = 0;
         jj = 15;,
         i < 16;
         jj >= 0;,
         i++;
         jj--;, // 16 rows, 2 columns
         Marker Size( 3 );
         Marker( i, {1, jj + .2} ); // markers 0-15
         Marker( i + 16, {4, jj + .2} ); // markers 16-31
         Text( {1.5, jj}, "marker ", i ); // marker labels 0-15
         Text( {4.5, jj}, "marker ", i + 16 ); // marker labels 16-31
      )
   )
);
```

Figure 9.18 JMP Markers



Tip: For more information about this script, see "Display Trees".

Note the following about colors:

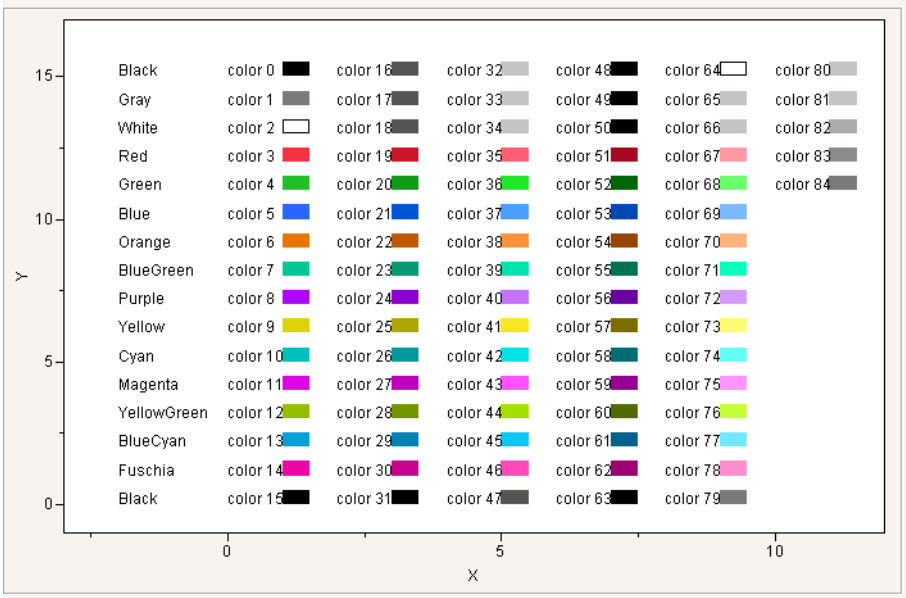
- JMP colors are numbered 0 through 84.
- The first 16 named colors are the basic colors. See the script below.
- Numbers higher than 16 are darker or lighter shades of the basic colors.
- Indices outside the range 0–84 have undefined behavior.
- For more information about using JMP colors, see "Specify Colors".

The following script illustrates the standard JMP colors:

```
Text Color( 0 );
New Window( "Colors",
   Graph Box(
      FrameSize( 640, 400 ),
      Y Scale( -1, 17 ),
      X Scale( -3, 12 ),
      k = 0;
      For( jj = 1, jj <= 12, jj += 2,
         l = 15;
         For( i = 0, i <= 15 & k < 85, i++,
            thiscolor = Color To RGB( k );
            Fill Color( k );
            thisfill = 1;
            If( thiscolor == {1, 1, 1},
               Pen Color( 0 );
               thisfill = 0;
```

```
            ,
               Pen Color( k )
            );
            Rect( jj, l + .5, jj + .5, l, thisfill );
            Text( {jj - 1, l}, "color ", k );
            k++;
            l--;
         );
      );
      jj = -2;
      color = {"Black", "Gray", "White", "Red", "Green", "Blue", "Orange",
"BlueGreen",
      "Purple", "Yellow", "Cyan", "Magenta", "YellowGreen", "BlueCyan",
"Fuschia", "Black"};
      For(
         i = 0;
         l = 15;, i <= 15 & l >= 0,
         i++;
         l--;,
         Text( {jj, l}, color[i + 1] )
      );
) );
```

Figure 9.19 JMP Colors



If you prefer to use RGB values, each color should be a list with percentages for each color in red, green, blue order. For example, the following percentages produce a teal color:

```
Pen Color( {.38,.84,.67} ); // a lovely teal
```

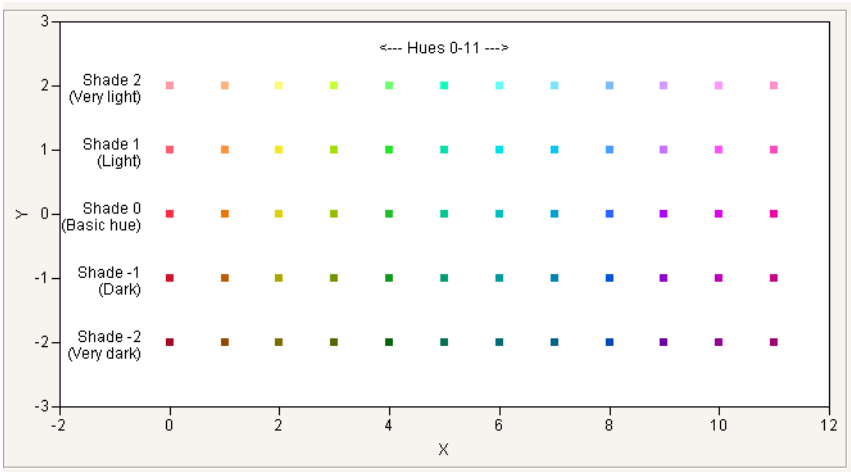
Hue and Shade Example

`Hue State` and `Shade State` together are an alternative to `Color State` for choosing colors. You cannot select black, white, or the shades of gray when you use `Hue State`. For these, you must use `Shade State` alone, or `Color State`.

The following script demonstrates how hue and shade values relate to colors:

```
New Window( "Hues and Shades",
   Graph Box(
      FrameSize( 600, 300 ),
      Y Scale( -3, 3 ),
      X Scale( -2, 12 ),
      k = 0;
      For( h = 0, h < 12, h++,
         For( s = -2, s < 3, s++,
            myMk = Combine States( Hue State( h ), Shade State( s ), Marker
   State( 15 ) );
            Marker Size( 3 );
            Marker( myMk, {h, s} );
         )
      );
      Text( Center Justified, {5, 2.5}, " <--- Hues 0-11 ---> " );
      Text( Right Justified,
         {-.5, -2}, "Shade -2", {-.5, -2.25}, "(Very dark)",
         {-.5, -1}, "Shade -1", {-.5, -1.25}, "(Dark)",
         {-.5, 0}, "Shade 0", {-.5, -.25}, "(Basic hue)",
         {-.5, 1}, "Shade 1", {-.5, .75}, "(Light)",
         {-.5, 2}, "Shade 2", {-.5, 1.75}, "(Very light)"
      );
   )
);
```

Figure 9.20 Hues and Shades



There are no `-Of` functions for `Hue` and `Shade`. `Color Of` returns the equivalent `Color State` index for a color row state that has been set with `Hue State` or `Shade State`. For example, the following example gives rows 4 and 5 the same dark red marker:

```
Row State( 4 ) = Combine States( Hue State( 0 ), Shade State( -1 ), Marker
   State( 12 ) );
Row State( 5 ) = Combine States(
   Color State( Color Of( Row State( 4 ) ) ),
   Marker State( Marker Of( Row State( 4 ) ) )
);
```

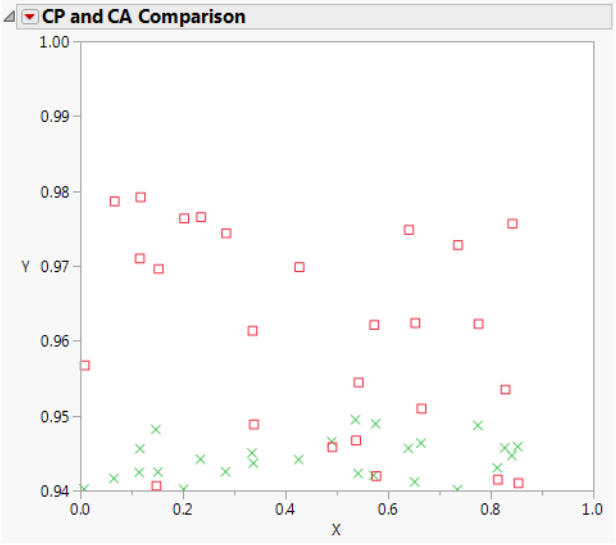
Row State and Matrices Example

In the following example, row state values are prepared ahead and passed to the `Marker` routine, along with matrices of coordinates.

```
dt = New Table( "Artificial CP and CA data",
   Add Rows( 26 ),
   New Column( "cover_cp",
      Numeric,
      "Continuous",
      Formula( Random Uniform() / 100 + 0.94 )
   ),
   New Column( "cover_ca",
      Numeric,
      "Continuous",
      Formula( Random Uniform() * 0.04 + 0.94 )
   ),
   New Column( "p", Numeric, "Continuous", Formula( Random Uniform() ) )
);
```

```
dt << Run Formulas;
greenMark = Combine States( Marker State( 2 ), Color State( 4 ) );
redDiamond = Combine States( Marker State( 3 ), Color State( 3 ) );
New Window( "CP and CA Comparisons",
   Graph Box(
      Title( "CP and CA Comparison" ),
      FrameSize( 400, 350 ),
      X Scale( 0, 1 ),
      Y Scale( 0.94, 1 ),
      For Each Row(
         Marker( greenMark, {p, cover_cp} );
         Marker( redDiamond, {p, cover_ca} );
      )
   )
);
```

Figure 9.21 Example of Row States Values and Matrices



Accessing Data Values

The typical way to work with values in a data table is to follow these steps:

1. Set up the data table whose values you want to access as the current data table. Or, if you already have a data table reference, you can simply use that reference. See "Set the Current Data Table".

2. Specify the row or rows whose values you want to access and specify the column name that contains the values that you want to access. See "Set or Get Values by Column Name".

The following example opens the Big Class.jmp sample data table (making it the current data table), and then specifies row 2 in the weight column. A value of 123 is returned in the log, which is the weight for Louise in row 2.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt:weight[2];
   123
```

Set or Get Values by Column Name

The easiest way to refer to a column is by its name. If you have a global variable and a column with the same name, to prevent ambiguity, scope the column name with the `:` prefix operator.

To set the value of a cell in the current row, provide the column name and the new value.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Row() = 5; // select row 5
dt:age = 19; // set the age in row 5 to 19
dt:name = "Sam"; // set the name in row 5 to Sam
```

To set the value of a cell in row 10, specify the row number in a subscript.

```
dt:age[10] = 20; // set the value of age in row 10 to 20
```

An empty subscript refers to the current row, so :age[ ] is the same as :age.

To get the value of a cell in row 16, specify the column name.

```
Row() = 16; // select row 16
myGlobal = :age; // store the age as a variable
:age; // returns 14, the age in row 16
Show( :age ); // returns age = 14;
```

To get the value of a cell in a specific row, include the column name and row number. Both of the following examples return 13, the value of age in row 12 of Big Class.jmp:

```
:age[12];
myGlobal = :age[12];
```

To get a value in a data column reference, use `Column()` and `As Column()` to get the value in a data column reference. See "Access Cell Values through Column References".

Notes:

- If you do not specify the row number, the current row is selected.
- An empty subscript, such as `:age[ ]`, refers to the current row.

- Be careful that you are subscripting to a table row that exists. The default row number is zero, so statements like `:name` that refer to row zero generate an `Invalid Row Number` error.

Additional Ways to Access Data Values

There are other ways to specify a data table, row, and column. You can specify all three items in one expression by using an infix operator and a subscript:

```
dt:age[2] = 12; // table, column, and row
```

If you want to target multiple rows, you can use subscripts with a list or matrix of row numbers.

```
age[i] = 3;
age[{3, 12, 32}] = 14;
list = age[{3, 12, 32}]; // results in a list
vector = age[1 :: 20]; // results in a matrix
dt[1,1] = dt[2,1]; // in Big Class.jmp, “KATIE” (row 1, column 1) is now
  “LOUISE” (from row 2, column 1)
dt[0,{height}] = dt[0,{weight}]; // everybody’s height is their weight
dt[1,{height,weight}] = dt[2,4::5]; // row 1 height and weight is from Louise
dt[[5 3 1], 0] = .; /* set rows 5, 3, and 1 to missing. Using out-of-order
  subscripts makes more sense if the right hand side is a matrix or a data
  table with subscripts */
```

Note about Changing Values

Whenever you change values in a data table, messages are sent to the displays to keep them up-to-date. However, if you have thousands of changes in a script, this increases the time it takes to complete the updates.

In order to speed up changes, use `Begin Data Update` before the changes to block these update messages. Use `End Data Update` after the changes have been completed to release the messages and update the displays.

```
dt << Begin Data Update;
...<many changes>...
dt << End Data Update;
```

Be sure to always send the `End Data Update` message, otherwise the display is not updated until forced to do so in some other way.

Note: `Begin Data Update` does not affect the data refresh due to some other table manipulations. For example, when you delete or add columns, the data table is updated and then the data update begins.

Add Metadata to a Data Table

Data tables store observation data, or measurements of various variables on a specific set of subjects. However, JMP data tables can also store *metadata*, or data about the data.

- "Data Table Variables"
- "Table Scripts"
- "Formulas"
- "Delete Metadata"

Data Table Variables

Table variables are for storing a single text string value, such as "Notes". To understand how variables work, first get its existing value by sending a `Get Table Variable` message:

```
dt = Open( "$SAMPLE_DATA/Solubility.jmp" );
dt << Get Table Variable( "Notes" );
   "Chemical compounds were measured for solubility in different solvents. This
      table shows the log of the partition coefficient (logP) of 72 organic
      solutes in 6 aqueous/nonpolar systems."
```

Now change the existing value of the string using `Set Table Variable` and then use `Get Table Variable` again to check that the string has been updated:

```
dt << Set Table Variable( "Notes", "Solubility of chemical compounds" );
dt << Get Table Variable( "Notes" );
   "Solubility of chemical compounds"
```

The following example adds two new table variables to a data table:

```
dt = Open( "$DOCUMENTS/Big Class.jmp" );
myvar = "This is my version of the JMP Big Class sample data.";
dt << Set Table Variable( "key1", myvar );
dt << Set Table Variable( "key2", myvar );
```

Notice that setting the value creates a new variable only if one by the given name does not already exist. If you add two table variables with the same name, only one variable is created.

Table Scripts

You can add a new script, run a script, or get a script using JSL.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Script( "Bivariate Example", Bivariate( Y( :weight), X( :height ) )
   );
dt << Get Table Variable( "Bivariate Example" );
```

```
    // returns Bivariate( Y( weight ), X( height ) )
dt << Run Script( "Bivariate Example" )
```

The following example replaces the contents of the table script that was created by the preceding script:

```
dt << Set Property( "Bivariate Example", Bivariate( Y( :weight ), X( :height
  ), Fit Line ) ); // include Fit Line in the Bivariate Example script
```

Suppose you want a text representation of a data table, perhaps to e-mail to a colleague or to use as part of a script. You can obtain a script that reconstructs the information in a data table with `Get Script`. The following example opens Big Class.jmp and prints the data, table variables, and column properties to the log. A portion of the output is shown here:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Get Script;
   New Table( "Big Class",
      Add Rows( 40 ),
      New Script(
         ["en" => "Distribution",...],
         Distribution(
            Continuous Distribution( Column( :weight ) ),
            Nominal Distribution( Column( :age ) )
         ) ),
```

`Delete Scripts` deletes one or more scripts.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Delete Scripts( "Distribution" );
```

In JMP versions prior to 14, use `Delete Property` to delete a script.

Note: The `Names Default to Here` setting is false by default for table scripts. If a table script specifies `Names Default to Here(1)`, this setting lasts only as long as the table script runs. After the script finishes running, the setting is reset to whatever it was before the script ran.

Options for Downloaded Data Tables that Contain Scripts

You are prompted to open or examine downloaded data tables that contain scripts. This help you prevent potentially damaging scripts from running on your computer. In the `Open()` command, you can control this behavior.

To open the data table and enable scripts to run:

```
Open( "$DOWNLOADS/file.jmp", Quarantine Action( "Allow Scripts" ) );
```

To open the data table and block scripts:

```
Open( "$DOWNLOADS/file.jmp", Quarantine Action( "Block Scripts" ) );
```

To display the default prompt window to open or examine the file:

```
Open( "$DOWNLOADS/file.jmp", Quarantine Action( "Show Dialog" ) )
```

To enable a `Try()` expression to capture the error:

```
Try( Open( "$DOWNLOADS/file.jmp", Quarantine Action( "Do Not Open" ) ),
Show( exception_msg ) );
```

If the user selects Examine, the scripts are disabled.

Notes:

- The Block Scripts and Show Dialog options don't work if the domain that you downloaded the data table from is on your Windows Internet Options Trusted sites list.
- After you select Open for a quarantined data table, the data table is trusted until the next time you open JMP.

Prompting to Run a Script upon Opening a Data Table

You can prompt to open a table script named `On Open` or `OnOpen` when the data table opens. The choice is remembered each time you open the data table in the current JMP session.

To create an `On Open` script, perform one of the following actions:

- Create the script using the **Save Script > To Data Table** option, double-click the script name, and change the name to `On Open`.
- Store the script using a `New Script` message.

In this example, you create an OnOpen script in Big Class.jmp.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Script(
   "OnOpen", // create the script
   sortedDt = Data Table( "Big Class" ) << Sort( By( :name ), Output Table
   Name( "Sorted Big Class" ) ) // sort the data and put it in a new data table
);
```

The JMP preference called Evaluate OnOpen Scripts determines when the script runs. By default, the user is prompted to run the script. You can set the preference to always run the `On Open` script or to prevent it from running:

```
Preference( Evaluate OnOpen Scripts( "always" ) );
Preference( Evaluate OnOpen Scripts( "never" ) );
Preference( Evaluate OnOpen Scripts( "prompt" ) ); // default setting
```

`On Open` scripts that execute other programs are never run. As a safety precaution, you might consider suppressing automatic execution when opening data tables that you receive from others.

Notes:

- If the OnOpen script is in a downloaded data table, you must specify `Quarantine Action( "Allow Scripts" )` in the `Open()` expression to run the OnOpen script automatically. See "Options for Downloaded Data Tables that Contain Scripts".
- When you create a new data table in a script and include the `On Open()` function, `On Open()` is called after the data table is created.
- Some operations in an OnOpen script are considered unsafe, such as saving the current data table. Run an external script to save that table instead of opening a table to run the OnOpen script.

Work with Groups of Table Scripts

Suppose that you want to put related table scripts in a group to save room in the list of data table scripts. Use `Group Scripts`, specify the group name, and then specify the script names to put in the group.

```
dt = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
dt << Group Scripts(
   "Maps",
   {"Bubble Plot Street Map", "Graph Builder Street Map: Traffic Incidents",
   "Graph Builder Street Map Zoomed"}
);
dt << Group Scripts(
   "Tabulate",
   {"Tabulate: Category and Description", "Tabulate: Category, Summary,
   Percentage",
   "Tabulate: Category and Resolution", "Tabulate: Resolution and
   Description"}
);
```

Move a Script Group

`Move Script Group` rearranges the table script groups using the `to first`, `to last`, `after(script)`, and `after(group)` arguments:

```
dt << Move Script Group( "Tabulate", after( "Maps" ) );
dt << Move Script Group( "Tabulate", to last);
```

Get Groups and Group Names

To get the list of table scripts in the group, use `Get Script Group`. Then you can perform an operation such as running a script in the group.

```
tb = dt << Get Script Group( "Tabulate" );
dt << Run Script( tb[2] ); // runs the second script in the Tabulate group
```

`Get Script Groups Names` returns the group names in a list:

```
gn = dt << Get Script Groups Names; // returns {"Maps", "Tabulate"}

// move the Maps group to the end of the table scripts
dt << Move Script Group(gn[1], to last);
```

Select a Script Group

`Select Script Group` selects the named table script group, a list of scripts in a group, or all scripts.

```
dt << Select Script Group; // select all scripts groups
dt << Select Script Group( "Tabulate" ); // select Tabulate group
dt << Select Script Group( {"Tabulate", "Maps"} );// select both groups
```

Ungroup Scripts

`Ungroup Scripts` removes specified table scripts or group from the group. You can send the message to the name of the script group or a list of scripts.

```
dt << Ungroup Scripts( "Tabulate" );
dt << Ungroup Scripts(
   {"Tabulate: Category and Description", "Tabulate: Category, Summary,
   Percentage",
   "Tabulate: Category and Resolution", "Tabulate: Resolution and
   Description"}
);
```

Rename a Script Group

You can rename a table script group using `Rename Script Group`, specifying the existing group name, and specifying the new group name.

```
dt << Rename Script Group( "Maps", "Street Maps" );
```

Delete a Script Groups

To delete a table script group, you get it first and then delete it.

```
a = dt << Get Script Group( "Street Maps" );
dt << Delete Scripts( a );
```

Formulas

The message `Suppress Formula Eval` takes a Boolean argument to specify whether formula evaluation should be suppressed or not. You might want to suppress evaluation if you plan to make numerous changes to the data table and do not want to wait for formula updates between steps.

```
dt << Suppress Formula Eval( 1 );
dt << Suppress Formula Eval( 0 ) ;
```

To accomplish the same effect for all data tables, use the `Suppress Formula Eval` command to turn off formulas globally. This is the same as the message above, except that you do not send it to a data table object.

```
Suppress Formula Eval( 1 ); // make formulas static globally
Suppress Formula Eval( 0 ); // make formulas dynamic globally
```

Note that formulas are not evaluated when they are installed in the column. Even when you force evaluation, they end up being evaluated again in the background. This can be a problem for scripts if they depend on the column having the values while the script is running. If you need a mechanism to control evaluation, use the `EvalFormula` command or the `Run Formulas` command.

To force a single column to evaluate, send an `Eval Formula` command to the column. You can even do this inside the command to create the column, after the formula clause:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Ratio",
   Numeric,
   Formula( :height / :weight ),
   Eval Formula
);
```

`dt << Run Formulas` performs all pending formula evaluations, including evaluations that are pending as a result of evaluating other formulas. This function is useful when you have a whole series of columns to run.

Tip: This method is preferred over `EvalFormula`. Although `EvalFormula` evaluates the formulas, it does not suppress the background task from evaluating them again. The background task takes great care to evaluate the formulas in the right order.

If you send the `Run Formulas` command to a data column, the evaluation is done at the time of the command, but it does not suppress the scheduled evaluations that are pending. Therefore, formulas might end up being evaluated twice if you also send the command to the data table and the data column. Being evaluated twice might be desirable for formulas that have random function in them, or it might be undesirable if they depend on randomization seeds being set. If you use random numbers and use the `Random Reset(seed)` feature to make a replicable sequence, then use the `Run Formulas` command, because it avoids a second evaluation.

Note: All platforms send a `Run Formulas` command to the data table to assure that all formulas have finished evaluating before analyses start.

Set Values without a Formula

`col << Set Each Value(expression)` evaluates the expression for each row of the data table and assigns the result to the column. It does not store the expression as a formula.

Delete Metadata

You can delete table variables, table properties (such as a script or a variable), and formulas from the data table, using the following commands:

```
dt << Delete Table Variable( name );
dt << Delete Table Property( name|{property1, property2, ...});
col << Delete Formula;
col << Delete Property( name|{property1, property2, ...} );
col << Delete Column Property( name );
```

Calculations

This section discusses functions for pre-evaluated columnwise and rowwise statistics and shows how JSL expressions work behind the scenes in the JMP formula calculator.

Pre-Evaluated Statistics

The following functions are special, pre-evaluated functions: `Col Maximum`, `Col Mean`, `Col Minimum`, `Col N Missing`, `Col Number`, `Col Quantile`, `CV` (Coefficient of Variation), `Col Standardize`, `Col Std Dev`, `Col Sum`, `Maximum`, `Mean`, `Minimum`, `NMissing`, `Number`, `Std Dev`, and `Sum`.

Note: Statistics are also computed with `Summarize` ("Store Summary Statistics in Global Variables"). Although the named arguments to `Summarize` have the same names as these pre-evaluated statistic functions, they are *not* calling the pre-evaluated statistic functions. The resemblance is purely coincidental.

All the statistics are *pre-evaluated*. That is, JMP calculates them once over the rows or columns specified and thereafter uses the results as constants. Because they are computed once and then used over and over again, they are more efficient to use in calculations than the equivalent formula-calculated results.

When JMP encounters a pre-evaluated function in a script, it immediately evaluates the function and then uses the result as a constant thereafter. Therefore, pre-evaluated functions enable you to use columnwise results for rowwise calculations. For example, if you use `Col Mean` inside a column formula, it first evaluates the mean for the column specified and then uses that result as a constant in evaluating the rest of the formula on each row. A formula might standardize a column using its pre-evaluated mean and standard deviation:

```
( Height-Col Mean( Height ) ) / Col Std Dev( Height )
```

For the Big Class.jmp data, `Col Mean(Height)` is 62.55 and `Col Std Dev(Height)` is 4.24. So for each row, the formula above would subtract 62.55 from that row's height value and then divide by 4.24.

Note: Pre-evaluated functions disregard the excluded row state, meaning that any excluded rows are included in calculations. For summary statistics that obey row exclusion, use the Distribution platform.

Columnwise Functions

The functions whose names begin with "`Col`" all work *columnwise*, or down the values in the specified column, and return a single number. For example, `Col Mean(height)` finds the mean of the values in all the rows of the column height and returns it as a scalar result. Some examples include the following:

```
Average Student Height = Col Mean( height );
Height Sigma = Col Std Dev( height );
```

With the `Col` functions, column properties such as Missing Value Codes assign data values that produce incorrect calculations. Suppose that the Missing Value Codes column property is assigned to the `x1` column to treat "999" as a missing value. Another column includes a formula that calculates the mean. To use the value "999" instead of a missing value to calculate the mean, refer to `Col Stored Value()` in the formula:

```
Mean( Col Stored Value( :x1 ), :x2, :x3 )
```

Rowwise Functions

The functions *without* "Col" listed below work *rowwise* across the values in the variables specified and return a column result. For example, `Mean(height, weight)` finds the mean of the height and weight for the current row of the data table. The rowwise statistics are valid only when used in an appropriate data table row context. The following are some possibilities:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

// scalar result for row 7 assigned to JSL global variable
Row() = 7;
::scalar = Mean( height, weight );

// formula column created in data table
dt << New Column( "Scaled Ht-Wt Ratio",
   Formula( Mean( height, weight ) / age )
);

vector = J( 1, 40 ); // create a 1x40 matrix to hold results
For Each Row( vector[Row()] = Mean( height, weight ) ); // fill the vector
```

Rowwise functions can also take vector (column matrix) or list arguments:

```
myMu = Mean( [1 2 3 4] );
mySigma = Std Dev( {1, 2, 3} );
```

Calculator Formulas

You can store formulas in columns that are automatically evaluated to create the values in the cells of the column. If you open the formula, you get a calculator interface to edit the formula structurally. However, the formula is implemented with JSL, and you can obtain the text JSL form of any expression in the calculator by double-clicking it. The text can be edited, and when it is de-focused, it is compiled back into the structural form.

Note: There is no difference between a formula column created through the calculator window and one created directly through JSL with commands such as `New Column(..., Formula(...))` or `Col << Formula(...)`.

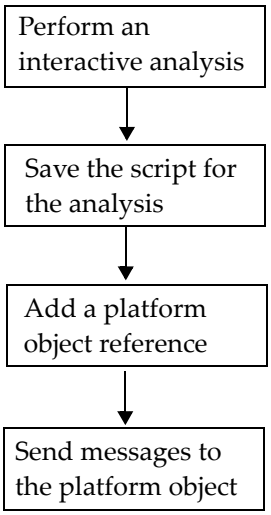
Chapter 10

Scripting Platforms

Create, Repeat, and Modify Analyses

If you run the same analysis frequently, you can script it to automate the process. Anyone can then run the script, ensuring the same results every time. To get started, perform your analysis interactively as you normally would with JMP, and then save a script that re-creates that analysis. You can modify the script to further customize it.

Figure 10.1 Typical Workflow for Scripting Platforms



This chapter is about scripting platforms, not reports. *Platform* object references and *report* object references receive different types of JSL messages. Platforms can run tests, draw plots, and so on. Reports can copy pictures, select display boxes, or close outline nodes. To learn about scripting reports, see the section "Navigate JMP Reports".

Tip: For additional scripting help, see the JMP Scripting Index (**Help > Scripting Index**) and the *JSL Syntax Reference*.

Contents

Example of Scripting a Platform . . . 477
Send Messages to a Platform . . . 480
Conventions for Messages and Arguments . . . 480
Send Multiple Messages . . . 481
Find Messages for Objects . . . 482
Interpret the Show Properties List . . . 483
Send Messages to an Object . . . 484
Specify Which Columns to Analyze . . . 485
Create Column References . . . 485
Specify Multiple Column Names at Once . . . 485
Enable Users to Specify Columns . . . 486
Specify a By Variable . . . 486
Filter by Value using a Where Statement . . . 489
Enable User Input . . . 490
Execute Embedded Red Triangle Options . . . 490
Make Platforms Invisible . . . 491
Specify Report Titles . . . 493
Common Messages for Platform Windows . . . 493
Scripting Considerations by Platform . . . 493
Categorical . . . 493
Control Charts . . . 494
Distribution . . . 498
DOE . . . 498
Fit Model . . . 500
Formula Depot . . . 505
Graph Builder . . . 506
Neural and Neural Net . . . 508
Partial Least Squares and PLS . . . 508
Process Capability . . . 508
Scatterplot 3D . . . 509
Text Explorer . . . 509
Explore Outliers . . . 510
Scripting-Only Messages and Arguments by Platform . . . 510
Bivariate Script-Only Messages . . . 512
Boosted Tree Script-Only Messages . . . 513
Bootstrap Forest Script-Only Messages . . . 514
Bubble Plot Script-Only Messages . . . 515
Choice Script-Only Messages . . . 515
Clustering Script-Only Messages . . . 516

Contingency Script-Only Messages 516
Contour Plot Script-Only Messages 516
Contour Profiler Script-Only Messages 516
Control Chart Script-Only Messages 517
Control Chart Builder Script-Only Messages 517
Cumulative Damage Script-Only Messages 517
Custom Profiler Script-Only Messages 518
Degradation Script-Only Messages 518
Destructive Degradation Script-Only Messages 518
Discriminant Script-Only Messages 518
Distribution Script-Only Messages 519
EMP Measurement Systems Analysis Script-Only Messages 519
Explore Missing Values Script-Only Messages 520
Explore Outliers Script-Only Messages 520
Factor Analysis Script-Only Messages 520
Fit Life by X Script-Only Messages 520
Fit Model Script-Only Messages 521
Fit Parametric Survival Script-Only Messages 523
Functional Data Explorer Script-Only Messages 524
Gaussian Process Script-Only Messages 524
K Means Cluster Script-Only Messages 524
K Nearest Neighbors Script-Only Messages 524
Latent Class Analysis Script-Only Messages 524
Life Distribution Script-Only Messages 525
Make Validation Column Script-Only Messages 526
MaxDiff Script-Only Messages 526
Mixture Profile Script-Only Messages 526
Model Screening Script-Only Messages 527
Multivariate Script-Only Messages 527
Naive Bayes Script-Only Messages 527
Neural Script-Only Messages 527
Nonlinear Script-Only Messages 528
Normal Mixtures Script-Only Messages 529
Oneway Script-Only Messages 529
Parallel Plot Script-Only Messages 529
Pareto Plot Script-Only Messages 530
Partial Least Squares Script-Only Messages 530
Partition Script-Only Messages 530
Principal Components Script-Only Messages 531
Process Capability Script-Only Messages 531

Process Screening Script-Only Messages 531
Profiler Script-Only Messages 532
Reliability Forecast Script-Only Messages 532
Reliability Growth Script-Only Messages 532
Response Screening Script-Only Messages 533
Scatterplot 3D Script-Only Messages 533
Structural Equation Models Script-Only Messages 534
Support Vector Machines Script-Only Messages 535
Surface Plot Script-Only Messages 536
Tabulate Script-Only Messages 539
Ternary Plot Script-Only Messages 539
Text Explorer Script-Only Messages 539
Time Series Script-Only Messages 540
Uplift Script-Only Messages 540
Variability Chart Script-Only Messages 541

Example of Scripting a Platform

If you are not familiar with JSL, it can be helpful to start by interactively performing your analysis in JMP, and then saving the analysis to a script. JMP creates the script for you, and you can then add to it or modify it as desired, by creating a platform object and sending it messages.

Overview

This example takes you through these steps:

1. "Launch the Analysis and Save the Script to a Window"
2. "Add a Platform Reference Object and Send Messages"
3. "Add a Data Table Reference and Save the Script"

Launch the Analysis and Save the Script to a Window

1. Select **Help > Sample Data Library** and open Big Class.jmp.

   This fictional data set contains the names, ages, sex, heights, and weights for 40 students.
2. Select **Analyze** > **Fit Y by X**.
3. Select height and click **Y, Response**.
4. Select age and click **X, Factor**.
5. Click **OK**.

   A Oneway plot of students' height by age appears.
6. Click the Oneway Analysis red triangle and select the following options:
   - **Means/Anova**
   - **Compare Means > Each Pair, Student's t**
7. Click the Oneway Analysis red triangle and select **Save Script > To Script Window**.

Here is the resulting script:

```
Oneway(
   Y( :height ), X( :age ),
   Each Pair( 1 ),
   Means( 1 ),
   Mean Diamonds( 1 )
);
```

Notice the following:

- The script begins with a call to the `Oneway()` function, which returns a Oneway object.

- The parentheses after the `Oneway()` function contain arguments that tell the Oneway function how to make the object.
  - The first two arguments, `Y` and `X`, are required at launch.
  - The next three arguments are optional: `Each Pair`, `Means`, and `Mean Diamonds`.
  - When you selected the Each Pair, Student's t option, the Each Pair feature was turned on.
  - When you selected the Means/Anova option, the Means and Mean Diamonds features were turned on.
- The scripting equivalent of turning an option on or off is the Boolean argument 0 (off) or 1 (on).

Add a Platform Reference Object and Send Messages

1. (Optional) Right-click in the script and select **Show Embedded Log**.

   Output that appears in the log is now easily visible below the script.
2. Choose a JSL variable to remember the Oneway platform object.

   In this example, use the name `oneObj`:

   ```
   oneObj = Oneway(
     Y( :height ),
     X( :age ),
     Each Pair( 1 ),
     Means( 1 ),
     Mean Diamonds( 1 )
   );
   ```

   This gives you a way to address the platform object and send messages to it.
3. Click **Run Script** .

   This creates the Oneway object and sets the variable `oneObj`. Now, send a message to tell the platform to turn on the Unequal Variances report.
4. At the end of the script, add the following line: `oneObj << Unequal Variances(1)`, as shown here:

   ```
   oneObj = Oneway(
     Y( :height ),
     X( :age ),
     Each Pair( 1 ),
     Means( 1 ),
     Mean Diamonds( 1 )
   );
   oneObj << Unequal Variances( 1 );
   ```
5. Highlight the Unequal Variances line of JSL and click **Run Script** .

Tip: You can also press Ctrl+R to run a highlighted line of JSL.

At the bottom of the report window, the Tests That the Variances Are Equal report now appears. Now, suppose that you want to see only that report (and the graph), and you want to close the other reports.

6. Add the following lines to the script:

```
rep = Report( oneObj );
rep["Oneway Anova"] << Close( 1 );
rep["Means Comparisons"] << Close( 1 );
```

The `Report()` function returns the report object for the Oneway platform, and stores a reference to the report in the JSL variable called `rep`. You can send messages or perform actions on the report object, such as closing specific report outlines.

7. Highlight these last 3 lines of the script and click **Run Script** .

In the report window, the Oneway Anova and the Means Comparisons reports are now closed. Only the initial graph and the Tests that the Variances are Equal report (which includes the Welch's Test report) appear.

If you close the Big Class.jmp sample data table at this point and then try to run the script, you are prompted to open a data table. A best practice is to precede the call to the Oneway platform with an `Open()` function to open the associated data table each time the script is run.

Add a Data Table Reference and Save the Script

1. Add the following line as the first line in the script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
```

If the data table is not open when the script is run, this line opens the associated data table.

Your finished script should look like this:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
oneObj = Oneway(
   Y( :height ),
   X( :age ),
   Each Pair( 1 ),
   Means( 1 ),
   Mean Diamonds( 1 )
);
oneObj << Unequal Variances( 1 );
rep = Report( oneObj );
rep["Oneway Anova"] << Close( 1 );
rep["Means Comparisons"] << Close( 1 );
```

Once you complete your script, save it for future access.

2. From the script window, select **File** > **Save As**. Enter a name for your script and save it to any directory.

Send Messages to a Platform

Before you can send messages to a platform, you need to create a platform object that you can send the messages to. Use the Send operator (<<).

The following line keeps a reference to a Oneway platform in the JSL variable called `oneObj`:

```
oneObj = Oneway( Y( height ), X( age ) );
```

Then, add the Send operator to send the platform object a message. In this example, the message is to turn on the Unequal Variances report:

```
oneObj << Unequal Variances( 1 );
```

To see which messages can be sent to a specific object, do one of the following:

- Use the autocomplete shortcut. On a new line, enter the object name, then the Send operator (<<), and press Ctrl+space (Windows) or Ctrl+Option+space (macOS). The following example shows the messages that you can send to the platform object `oneObj` on Windows:

  ```
  oneObj << (Ctrl+space)
  ```

  A list of messages appears. Click one to insert it.
- Use the `Show Properties ()` function. Specify the object in the parenthesis. The following example shows the messages that you can send to the platform object `oneObj`:

  ```
  Show Properties (oneObj)
  ```

  A list of messages appears in the log.

Conventions for Messages and Arguments

- For most messages, omitting the argument for a Boolean command enables the option. For example, all of these messages create a test for unequal variances:

  ```
  oneObj << Unequal Variances;
  oneObj << Unequal Variances( 1 );
  oneObj << Unequal Variances( "true" );
  oneObj << Unequal Variances( "yes" );
  oneObj << Unequal Variances( "present" );
  oneObj << Unequal Variances( "on" );
  ```

The first three approaches are recommended.

Notes:

- The quoted arguments "present", "absent", "on", "off", "switch", and "flip" are supported for backwards compatibility.
- When the name of the JMP option contains several options separated by a comma or slash (such as the **Means/Anova** option or the **T Square, T^2** option) you can use any *one* of the menu names. Or, you can use the full menu name, including the comma or slash, if the string is followed by n. For example, any of these messages create a Means/Anova test:

  ```
  oneObj << Means( 1 )
  oneObj << Anova( 1 )
  oneObj << "Means/Anova"n( 1 )
  ```

- When an option appears under a menu in JMP, the corresponding script message is the option itself without the parent menu. For example, in the Oneway red triangle menu, the Nonparametric menu has multiple options under it, including Wilcoxon Test. Simply use Wilcoxon Test as a function or message in a script:

  ```
  oneObj = Oneway( Y( height ), X( age ), Wilcoxon Test( 1 ));
  oneObj << Wilcoxon Test( 1 );
  ```

- When a menu in JMP contains values rather than options, in the script, specify the parent menu and the value as an argument. For example, in the Oneway red triangle menu, the Set Alpha Level menu has values, such as 0.10, 0.05, 0.01, and Other. To specify 0.01 in a script, add a line like the following:

  ```
  oneObj << SetAlphaLevel(0.01);
  ```

Send Multiple Messages

To send several messages, just add more << operators or more Send arguments:

```
dist << Quantiles( 1 ) << Moments( 1 ) << More Moments( 1 ) << Horizontal
  Layout( 1 );
Send( dist, Quantiles( 1 ), Moments( 1 ), More Moments( 1 ), Horizontal
  Layout(1));
```

Because << is an *eliding operator*, it combines arguments and works differently than if its arguments were grouped. You can stack multiple messages with extra << symbols to perform them all in order, from left to right.

Another way to stack messages is to send a list of messages:

```
dist << {Quantiles( 1 ), Moments( 1 ), More Moments( 1 ), Horizontal
  Layout( 1 )};
```

These approaches work well assuming that no value is returned as a result of one of your messages. However, if your goal is to send a message to the *result* of another message, add grouping parenthesis. The second line shows the wrong way to do it; the third line shows how to do it correctly:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
/* the following line is incorrect, and sends the Fit Mean message to the
   original data table object */
bv = dt << Run Script( "Bivariate" ) << Fit Mean( 1 );
/* the following line is correct, and sends the Fit Mean message to the
   resulting Bivariate object/report */
bv2 = ( dt << Run Script( "Bivariate" ) ) << Fit Mean( 1 );
```

Find Messages for Objects

Some messages apply to all platform objects, and others are specific to certain platforms. Once you have created a platform object, find out what messages you can send to the object, using any of the following approaches:

Use the Scripting Index in JMP

1. Select **Help** > **Scripting Index**.
2. Enter the object type that you are interested in.
3. Click the correct object in the resulting list. Messages for the object appear in the Items list.

Once you have created a platform object, use the Show Properties(obj) function

Replace `obj` with your named platform object. A list and brief description of all the messages the object can receive appears in the Log. To understand more about the Show Properties output, see "Interpret the Show Properties List".

Tip: The `Show Properties()` function also works with data tables and display boxes. See "How Can I See All of the Messages that Can be Sent to a Data Table Object?" and "The << Operator".

- Examine the platform in JMP launch windows and report windows. Most options that you find there have JSL equivalents with the same names and arguments.

Interpret the Show Properties List

Show Properties lists the messages that can be sent to a scriptable object, such as a data table, an analysis platform, or a display, producing a text report in the log. Each type of message is categorized and appears in brackets, and can be any of the following:

Message Type	Description	Example
`[Action]`	Correspond to options in the JMP interface.	`Redo Analysis [Action] (Rerun this same analysis in a new window. The analysis will be different if the data has changed.)`
`[Action Choice]` `[Enum]`	List specific choices for their arguments.	`Fit Polynomial [Action Choice] {2, quadratic, 3, cubic, 4, quartic, 5, 6}` `Draw [Enum] {Filled, Outlined, Filled and Outlined}`
`[Boolean]`	Turn options on or off. Arguments are usually 1 or 0. If a message is specified without an argument, sending the message changes it to the opposite state. `[Default On]` indicates that the option is on by default.	`Show Points [Boolean] [Default On]`

`[Subtable]`	List options within submenus. You use the option itself, not the parent item.	`Script [Subtable]`
		`Redo Analysis [Action] (Rerun this same analysis in a new window. The analysis will be different if the data has changed.)`
		`Save Script to Datatable [Action] (Return to the launcher for this analysis.)`

`[New Entity]`	Open a new window in the interface.	`Prediction Interval [New Entity] (Prediction Interval to contain a single future observation or the mean of m future observations.)`

For example, run the following JSL:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = Bivariate( Y( :height ), X( :age ) );
Show Properties( biv );
```

The object `biv` supports `Subtable`, `Boolean`, `Action`, and `Action Choice` type messages. `Enum` and `New Entity` messages are not supported.

Send Messages to an Object

Suppose that you want to send a message to an object in the output. For example, in Fit Model, you want to send the Response Limits message to the Profiler object, Use the subscript operator (`[ ]`) to refer to the Profiler object.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

fm = dt << Fit Model(
   Y( :height ),
   Effects( :age, :sex, :weight, :age * :weight, :sex * :weight ),
   Personality( Standard Least Squares ),
   Emphasis( Effect Screening ),
   Run
);

obj = fm << Profiler(
   1,
   Confidence Intervals( 1 ),
   Desirability Functions( 1 ),
   Term Value( age( 12 ), sex( "F" ), weight( 105 ) )
);

Show( Profiler[1] << Get Desirability ); // show the limits before

Profiler[1] << (height << Response Limits(
   {Lower( 50, 0.95 ), Middle( 58, 0.7 ), Upper( 70, 0.066 ),
   Goal( "Minimize" ), Importance( 1 )}
));
Wait( 0.1 ); // for demonstration purposes
Show( Profiler[1] << Get Desirability ); // show the limits after
```

Specify Which Columns to Analyze

If you created your script interactively, you already specified columns in the launch window, so there is no need to specify them again. However, when creating a script from scratch, you normally want to specify the columns to analyze. For example, the following line launches the Distribution platform and specifies the height and weight columns as the Y variables:

```
Distribution( Y( height,weight ));
```

Create Column References

Tip: To see all the messages you can send to a column, go to **Help** > **Scripting Index** > **Data Table** > **Column Scripting**.

Creating a column reference enables you to access and send messages to the column. If you store each column reference in a JSL variable, you can use the variables instead of column names in the Y and X roles and elsewhere in the script.

For example, using Big Class.jmp, the following lines create Ycol as a column reference for weight, and Xcol as a column reference for height. The last line sends a message to Xcol to get the data type.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Ycol = Column( "weight" );
Xcol = Column( "height" );
Xcol << Get Data Type;
```

In the log, “Numeric” is output for the data type of the Xcol (height column).

Referring to a column reference with the : scoping operator (rather than plain text) is convenient in scripts saved to a data table. If the columns are renamed in the data table, the column references in the script are updated.

Specify Multiple Column Names at Once

If you have a lot of column names, instead of listing each one individually, you can use lists as column arguments. The following example gets the Y column names from a list and creates a Oneway analysis on each column in the list.

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Fit Group( Oneway( Y( {:SO2, :POP} ), X( :State ) ) );
```

If you do not know the column names when you are writing the script, use column indices, as done in the following example.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Distribution( Y( 1::N Col( dt ) ) );
```

You can also use indices when specifying effects as shown here:

```
dt << Fit Model( Y( :y ), Effects( 1 :: 4 ) );
```

Here's another example of specifying multiple column names. `5::31` selects columns 5 through 31.

```
Names Default To Here( 1 );
dt = Open( "$SAMPLE_DATA/Baltic.jmp" );
dt << Partial Least Squares( Y( :ls, :ha, :dt ), X( 5 :: 31 ), Go );
```

Enable Users to Specify Columns

Instead of specifying analysis columns yourself, you can enable a user to specify them, but you can still control what happens after they do so. In this case, use an empty specification. The following example runs the Distribution launch window:

```
Distribution();
```

In this scenario, the Distribution launch window appears and the user can choose the columns to analyze. After the user selects the columns and clicks **OK**, the Distribution report is shown as specified.

Specify a By Variable

In many JMP platforms, you can specify columns as By variables. To do this in a script, include a `By` argument in the platform command, listing each column as an argument.

The following example uses the Big Class.jmp data table, which contains the names, ages, sex, heights, and weights for 40 students. Create a bivariate report of weight by height, using sex as a By variable, and adding a variety of fits.

Create the Bivariate Report Using a By Variable

1. Select **File** > **New** > **Script**.
2. Add the following lines:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( weight ), X( height ), By( sex ));
```

The first line opens the Big Class.jmp sample data table. The second line creates a platform object called biv that runs a bivariate report of weight and height by sex.

3. Click **Run Script** .

   The report window shows two graphs, one where sex is F and one where sex is M.

Send Messages to the Entire Platform or to a Single By Level

1. If you do not have the log showing, turn it on. Right-click in the script and select **Show Embedded Log**.
2. Add this line to the script:

   ```
   Show( biv );
   ```

3. Highlight the `Show( biv )` line and click **Run Script** .

   In the log, rather than returning a single reference to a platform, `Bivariate[]`, the platform object returns a list of references: `biv = {Bivariate[], Bivariate[]}`. The two references correspond to the two levels (F and M) of the By variable, sex.

   You can direct messages to each By level individually or to all By levels.

4. Send a message to all By levels to add a linear regression fit. Add this line to the script:

   ```
   biv << Fit Line;
   ```

5. Highlight the line you just created and click **Run Script** .

   In the report window, a linear regression line is added to both graphs, with the corresponding Linear Fit reports.

6. Send a message to only the F level, adding a cubic polynomial fit. Add this line to the script:

   ```
   biv[1] << Fit Polynomial( 3 );
   ```

   Tip: The number for each By level corresponds to their order in the report window, which is usually alphanumeric. If the By column contains a Value Order column property, that ordering is followed.

7. Highlight the line you just created and click **Run Script** .

   In the report window, a polynomial fit line is added to the F graph, with the corresponding Polynomial Fit reports.

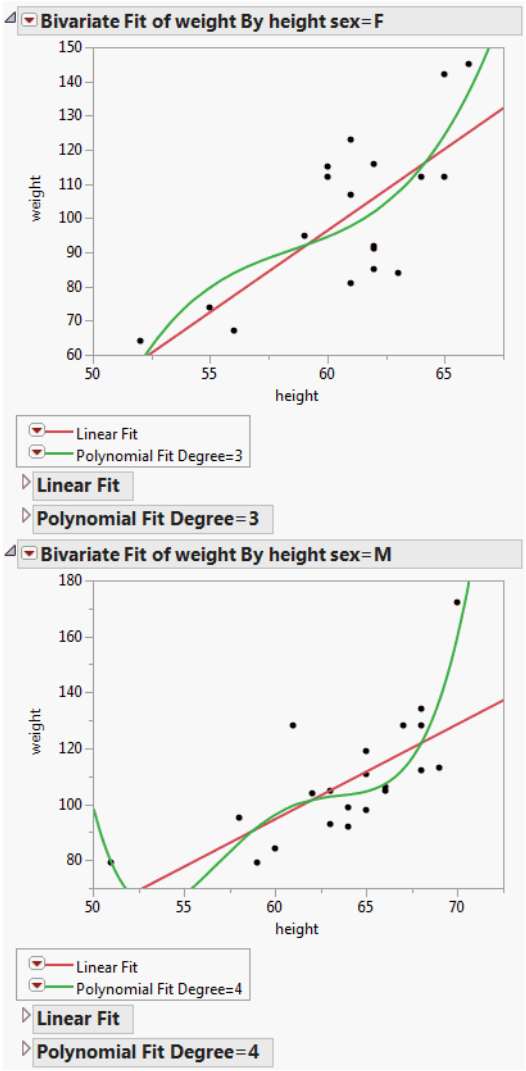
8. Send a message only to the M level, adding a quartic polynomial fit. Add this line to the script:

   ```
   biv[2] << Fit Polynomial( 4 );
   ```

9. Highlight the line you just created and click **Run Script** .

   In the report window, a quartic fit line is added to the M graph, with the corresponding Polynomial Fit reports.

Figure 10.2 By Group Reports



If you specify more than one column in `By` argument, graphs appear for each subgroup of each By variable. In this example, `By( sex, age )` would produce graphs for females age 12, females age 13, and so on, up to age 17. It would also produce graphs for males age 12 up to age 17.

Extract Results from a Report

The following example shows how to launch a platform with `By` groups and extract results (in this case, variances) from each group:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
```

```
// onew is the JSL variable holding a list of platforms
onew = dt << Oneway( x( :age ), y( :height ), by( :sex ), anova );

// r is a list of reports
r = onew << Report;

// nBy is the number of reports generated
nBy = N Items( r );

// vc is an array of as many rows as reports, one column, all set to zero
vc = J( nBy, 1, 0 );

// for each report, do the following:
For( i = 1, i <= nBy, i++,
  vc[i] = r[i] // vc[i] is the variance of the ith report
[Outline Box( "Analysis of Variance" ), // look for this outline

// then look for this column and get the second value
  Column Box( "Sum of Squares" )][2]
);
Show( vc ); // debugging, look in log to see this value

// byValues becomes a list of the values in the sex column
Summarize( byValues = By( :sex ) );

// make a new table with two rows (M.F) and two columns
New Table( "Variances" )
<< New Column( "Sex", // create a new column called Sex
  character,
  width( 8 ),
  values( byValues )

// create a new column called Variance
) << New Column( "Variance", Numeric, "Continuous", Values( vc ) );
```

Filter by Value using a Where Statement

You can use a `Where` statement to filter by a specific value. If you are adding a Where clause to a script that contains column references, you must first resolve the references. The following example limits the analysis to females and fits a line on the Bivariate report.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Ycol = Column( "weight" );
```

```
Xcol = Column( "height" );
biv2 = dt << Bivariate(
Y( Ycol ),
X( Xcol ),
Fit Line( ),
Where( :sex == "F" ) );
```

Enable User Input

When you send an `Action()` function to a platform, it evaluates an expression. For example, use `Action` messages when you want the launch window to appear. A user chooses columns in the launch window and the script continues running after the user runs the analysis.

The following example launches four platforms: Distribution, Bivariate, Oneway, and Contingency. The launch window for each platform appears, and the user chooses the analysis columns and clicks OK. The report window for each analysis appears, and then the next platform launch window comes up, until all platforms have been launched.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Distribution( Action( doit ) );
doit = Expr(
  New Window( "Bivariate", Bivariate( Action( doit2 ) ) )
);
doit2 = Expr(
  New Window( "Oneway", Oneway( Action( doit3 ) ) )
);
doit3 = Expr(
  New Window( "Contingency", Contingency( Action( doit4 ) ) )
);
doit4 = Expr();
```

Execute Embedded Red Triangle Options

In addition to the red triangle menu at the top of a report, some report windows contain embedded red triangle menus. These appear when you select an option that has sub-options, such as profilers. To execute options in embedded red triangle menus, use the `Get Scriptable Object` message. The following example creates a Life Distribution report then turns off the Confidence Intervals option in the Hazard Profiler red triangle menu:

```
dt = Open( "$SAMPLE_DATA/Reliability/Fan.jmp" );

// create a Life Distribution report with a lognormal distribution
obj = dt << Life Distribution(
```

```
        Y( :Time ),
        Censor( :Censor ),
        << Fit Lognormal
);

// get a reference to the Hazard Profiler red triangle menu
haz_prof = ( report( obj )[ "Hazard Profiler" ] << Get Scriptable Object);

// turn off confidence intervals in the Hazard Profiler
haz_prof << Confidence Intervals( 0 );
```

Make Platforms Invisible

In certain circumstances, you might want to make the platform analysis invisible. This means that no platform windows appear, and the platform runs in the background.

If you are doing a simulation or bootstrap analysis in your script, the script might call a platform hundreds or thousands of times. Perhaps you just want to pull one or two results out of each platform report. The following example performs a simulation and then creates a table of results and shows them in a Distribution report:

```
dt = As Table( (0 :: 10)`, << Column Names( {"X"} ) ); // create X variable

// set initial random seed to get reproducible results
Random Reset( 12345 );
dt << New Column( "Y",
   "Continuous",
   "Numeric",
   Set Formula( :X * 2 + 1 + Random Normal() )
);

// Y values have a random component
res = []; // define results matrix (empty to start)
For( i = 1, i <= 100, i++, // loop through simulation runs

   // for each iteration, fit a regression line and save parameter estimates
   bv = dt << Bivariate( Y( :Y ), X( :X ), Fit Line( 1 ), Invisible );
   res |/= (Report( bv )["Parameter Estimates"][

   // concatenate the results to the bottom of the results matrix
   Number Col Box( "Estimate" )] << Get as Matrix)`;

   // close the window (even though it's invisible)
   bv << Close Window;
      dt:Y << Eval Formula; // generate new Y values
```

```
);
// create table of results
dtres = As Table( res, << Column Names( {"Intercept", "Slope"} ) );

// show results in Distribution
dtres << Distribution( Y( :Intercept, :Slope ) );
```

If you want to run an analysis as an intermediate step in your script, and do not need the user to see the analysis, hide it with the `invisible` option. In this case, you might want to show only specific results from the report, which you can output to a journal or to the log.

- The following example runs the Distribution platform as invisible, and outputs only the CDF plot to a new journal:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dist = dt << Distribution( Y( :height ), CDF Plot( 1 ), invisible );
Report( dist )["CDF Plot"] << Journal;
dist << Close Window;
```

- The following example extracts the *F*-Ratio from an invisible Bivariate report, and outputs it to the log:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( x( :height ), y( :weight ), invisible );
biv << Fit Line;
r = biv << Report;
fratio = r[ColumnBox( "F Ratio" )][1];
r << Close Window;
Show( fratio );
  fratio = 1.15609545178219;
```

Tip: Invisible windows use resources that must be manually freed. Be sure to close the invisible window when the script is done with it.

You can also use the `invisible` option on the options in the Tables menu, such as Subset, Sort, Stack, and so on. The following example makes the Subset operation invisible:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Select Where( :age == 14 );
dt = dt << Subset( invisible );
subDt << Bivariate( x( :height ), y( :weight ), Fit Line );
```

Note that the preceding line could also be handled using a `WHERE` clause:

```
subDt << Bivariate( x( :height ),y( :weight ), Where( :age == 14 ), Fit Line
  );
```

Specify Report Titles

You can specify the title (shown in the title bar of a platform's report) by adding the `title` command to the launch request. The following example replaces the standard bivariate report's title with a user-specified one (My Title).

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Bivariate( x( :height ), y( :weight ), Title( "My Title" ) );
```

Common Messages for Platform Windows

For a list of common messages that you can send to most platforms, see the *JSL Syntax Reference*.

Scripting Considerations by Platform

This section explains specific considerations when scripting the following platforms:

- "Categorical"
- "Control Charts"
- "Distribution"
- "DOE"
- "Fit Model"
- "Formula Depot"
- "Graph Builder"
- "Neural and Neural Net"
- "Partial Least Squares and PLS"
- "Process Capability"
- "Scatterplot 3D"
- "Text Explorer"
- "Explore Outliers"

Categorical

The Free Text option in Categorical displays a Text Explorer report. Scripts that call the Free Text option in Categorical should be updated to use the Text Explorer platform.

Scripting for Supercategories

When ratings are involved in a data set (for example, a five point scale), you might want to know the percent of the responses in the top two or another subset of ratings. You can define a group of ratings in the data using the supercategories scripting option (instead of the column property).

The following example creates a new column called Floss Delimited 2, and adds a supercategory called Sleep Related, which contains two values: Before Sleep and Wake.

```
dt = Open( "$SAMPLE_DATA/Consumer Preferences.jmp" );
dt << New Column( "Floss Delimited 2",
   Character,
   Set Property(
      "Supercategories",
      {Group( "Sleep Related", {"Before Sleep,", "Wake,"} )}
   ), ...);
```

In the data table, under the Columns panel at left, scroll down to the new Floss Delimited 2 column. Click the column property indicator ✻ and select Supercategories. You can see the new Sleep Related supercategory with its two values.

Supercategories can also be specified in a Categorical launch command, where the properties are listed inside parentheses after the column name:

```
dt = Open( "$SAMPLE_DATA/Consumer Preferences.jmp" );
dt << Categorical(
   Supercategories(
      :Floss Delimited(
         {Group( "Sleep Related", {"Before Sleep,", "Wake,"} )}
      )
   ),
   Multiple Response( :Floss Delimited )
);
```

In the Categorical report, notice that the Sleep Related category has 203 cases. If you had run the Categorical platform without creating this supercategory, Before Sleep and Wake would appear as separate categories with 143 and 60 cases, respectively.

For more information about supercategories, see *Consumer Research*.

Control Charts

Using JSL, you can create control charts, customize tests, run alarm scripts, and more.

Reference Lines in Control Chart Builder

If you add a custom reference line to a Control Chart Builder chart through JSL, the reference line label must be something other than “LSL”, “USL”, or “AVG”. These names are keywords used by the Control Chart Builder platform.

The following example creates custom reference lines with unique labels.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
obj = dt << Control Chart Builder(
  Variables( Y( :height ) ),
  SendToReport(
    Dispatch(
      {},
      "height",
      ScaleBox,
      {Add Ref Line( 70, "Solid", "Black", "Upper Limit", 1 ),
      Add Ref Line( 55, "Solid", "Black", "Lower Limit", 1 )}
    )
  )
);
```

Customize Tests in Control Chart Builder

You can design custom tests and select or deselect multiple tests at once using the `Customize Tests` function. You specify the description, desired number, and label. This option is available only for Variables and Attribute chart types.

The following example creates a custom test called Test 1 and turns it on.

```
dt = Open( "$SAMPLE_DATA/Quality Control/Diameter.jmp" );
dt << Control Chart Builder(
  Variables( Subgroup( :DAY ), Y( :DIAMETER ) ),
  Customize Tests( Test 1( 2, "1" ) ), // test number, n, label
  Chart( Position( 1 ), Warnings( Test 1( 1 ) ) ), // turn Test 1 on
  Chart( Position( 2 ) )
);
```

Run Alarm Scripts

An alarm script alerts you when the data fail one or more tests. As an Alarm Script is invoked, the following variables are available, both in the issued script and in subsequent JSL scripts:

`qc_col` is the name of the column

`qc_test` is the test that failed

`qc_sample` is the sample number

`qc_phase` is the label of the phase during which the failure occurred

`qc_firstRow` is the first row in the sample

`qc_lastRow` is the last row in the sample

Example 1: Automatically Writing to a Log

One way to generate automatic alarms is to make a script and store it with the data table as a data table script or column property named QC Alarm Script.

The following example automatically writes a message to the log whenever a test fails:

```
dt = Open( "$SAMPLE_DATA/Quality Control/Coating.jmp" );
obj = dt << Control Chart(
   Sample Size( :Sample ),
   KSigma( 3 ),
   Chart Col( :Weight, XBar, R ),
   Alarm Script(
      Write(
         "Out of Control for test ",
         qc_test,
         " in column ",
         qc_col,
         " in sample ",
         qc_sample
      )
   )
);
obj << Test 1( 1 );
```

Example 2: Running a Chart with Spoken Tests

You can create a control chart with tests that are spoken aloud using the Speak function. Try the following example:

```
dt = Open( "$SAMPLE_DATA/Quality Control/Coating.jmp" );
dt << Control Chart(
   Alarm Script(
      Speak(
         Match( QC_Test,
            1, "One point beyond Zone A",
            QC_Test, 2,
            "Nine points in a row in zone C or beyond", QC_Test,
            5,
               "Two out of three points in a row in Zone A
               or beyond"
         )
      )
   ),
```

```
   Sample Size( :Sample ),
   Ksigma( 3 ),
   Chart Col( :Weight, Xbar( Test 1( 1 ), Test 2( 1 ), Test 5( 1 ) ), R )
);
```

You can have either of these scripts use any of the JSL alert commands, such as `Speak`, `Write`, or `Mail`.

Set Phase Limits

A *phase* is a group of consecutive observations in the data table. For example, phases might correspond to time periods during which a new process is brought into production and then put through successive changes. Phases generate, for each level of the specified Phase variable, a new sigma, set of limits, zones, and resulting tests.

The following example illustrates setting the limits for the different phases of Diameter.jmp:

```
dt = Open( "$SAMPLE_DATA/Quality Control/Diameter.jmp" );
dt << Control Chart(
   Phase( :Phase ),
   Sample Size( :DAY ),
   KSigma( 3 ),
   Chart Col(
      :DIAMETER,
      XBar(
         Phase Level(
            "1",
            Sigma( .29 ),
            Avg( 4.3 ),
            LCL( 3.99 ),
            UCL( 4.72 )
         ),
         Phase Level(
            "2",
            Sigma( .21 ),
            Avg( 4.29 ),
            LCL( 4 ),
            UCL( 4.5 )
         )
      ),
      R( Phase Level( "1" ), Phase Level( "2" ) )
   )
);
```

Distribution

You can set specification limits using the Spec Limits column property or using the `Spec Limits` argument in JSL:

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Distribution(
   Column( :OZONE ),
   Fit Distribution(
      Weibull(
         Spec Limits( LSL( 0.075 ), Target( 0.15 ), USL( 0.25 ) )
      )
   )
);
```

To set specification limits using K Sigma normal probabilities, use the `Set Spec Limits for KSigma` message. The first argument to the message is the value of K Sigma. By default, the limits are two-sided. To make them one-sided, use the optional `sided` argument. Specify `sided=1` for LSL only and `sided=2` for USL only.

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
obj = Distribution( Column( :CO ) );
obj << Fit Distribution(
   LogNormal( Set Spec Limits for KSigma( 3 ) )
);
```

DOE

Tip: For a complete list of DOE scripting options, select **Help > Scripting Index**, select **Objects** from the menu, and search for DOE.

To reproduce your DOE work, most DOE platforms provide these approaches:

- In the red triangle menu for the DOE window, the Save Script to Script Window option provides a script that reproduces the work that you have entered. This option is not available for the Nonlinear Design and Taguchi Array platforms.
- In the output design tables created by most DOE platforms, a script called DOE Dialog reproduces the work in the DOE window that created the design table. This script also contains the random seed (and number of starts or column starts, where appropriate) that enables you to reproduce your design exactly.

For many DOE platforms, the design table that is created also contains one or more scripts that you can run to conduct an appropriate analysis. These scripts are prefilled with the appropriate settings for your design.

The Random Seed

In constructing designs using the DOE platform, JMP uses a random seed to control certain actions that have a random component. A random seed can control the following:

- initializing search algorithms for design generation
- randomizing Run Order for design construction
- selecting a starting design for designs based on random starts

To reproduce a design using a script, you need to specify the random seed that generated them. For designs using random starts, set a random seed in the script before making the design.

The following example creates a Custom Design, sets the random seed, and makes the design:

```
DOE(
  Custom Design,
  Add Factor( Continuous, -1, 1, "X1", 0 ),
  Add Factor( Continuous, -1, 1, "X2", 0 ),
  Set Random Seed( 34067086 ),
  Make Design
);
```

Reproduce a Custom Design or a Main Effects Screening Design

Custom designs and main effects screening designs (constructed in the Screening Design platform) are generated by allocating a maximum number of seconds (Design Search Time) spent searching for a design. The default setting for Design Search Time is based on the complexity of the design. Even when two custom or main effects screening designs are constructed using the same random seed, the designs might not be identical because of different machine processing capabilities.

When you save the script for either of these designs using the Save Script to Script Window option, or when the DOE Dialog script is saved to the data table, the script specifies the Random Seed and the Number of Starts. The Number of Starts is the number of random starts used during the Design Search Time. The combination of the Set Random Seed and the Number of Starts options enables you to reproduce the design.

Note: If you write your own script, be aware that the order in which you specify options in your script can affect your results. For example, you should specify the design options before you make the design. In particular, if your design is constructed using a random seed, set it before you make the design.

Global Variables

Here are some global variables used to initiate or tune search algorithms.

Starting Design Enables you to specify a starting design. The following example replaces the random starting design with a specified matrix:

```
DOE Starting Design = matrix;
```

If a starting design is supplied, the Custom Design platform has only one start using this design.

K Exchange Value The coordinate exchange algorithm considers every row of factor settings for possible replacement in every iteration. You can limit the number of rows considered for exchange in an iteration so that only a small number of most likely rows is considered. The following example tells the algorithm to consider only the three most likely rows for exchange in each iteration:

```
DOE K Exchange Value = 3;
```

Bayes Diagonal Defines a vector that is added to the diagonal elements of the $X'X$ matrix. This new matrix is used in finding the D-optimal design. The following example adds the elements of `vector` to the diagonal elements of the $X'X$ matrix:

```
Bayes Diagonal = vector:
```

Fit Model

When scripting the Fit Model platform, here are a few general tips for controlling the behavior of the launch window:

- To keep the launch window open and fit the model at the same time, do one of the following (they are equivalent):
  - Include the `Run Model` message in your script.
  - Include both the `Run` message and the `Keep Dialog Open(1)` message.
- To run the model without showing the launch window, include the `Run` message in your script.

Model Scripts in the Data Table

When a data table contains a *Model* script and you launch Fit Model interactively, the Fit Model launch window is pre-populated according to the roles assigned by the table script.

Fit Group

If you want to fit different models and see a shared Profiler so that you can compare the models, use the `Fit Group` function. You can fit models such as least squares, nonlinear, neural, Gaussian processing, and mixed, in the same window with a shared Profiler. The following example creates both a Standard Least Squares model and a Gaussian Process model.

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
obj = dt << Fit Group(
  Fit Model(
     Y( :ABRASION ),
     Effects(
        :SILICA & RS,
        :SILANE & RS,
        :SULFUR & RS,
        :SILICA * :SILICA,
        :SILICA * :SILANE,
        :SILANE * :SILANE,
        :SILICA * :SULFUR,
        :SILANE * :SULFUR,
        :SULFUR * :SULFUR
     ),
     Personality( "Standard Least Squares" ),
     Emphasis( Minimal Report ),
     Run()
  ),
  Gaussian Process(
     Y( :HARDNESS ),
     X( :SILICA, :SILANE, :SULFUR ),
     Set Correlation Function( "Cubic" )
  )
);
```

In the report, click the Fit Group red triangle and select **Profiler**. Scroll down to the Prediction Profiler. You can see the ABRASION row for the SLS model, and the HARDNESS row for the Gaussian Process model appearing in the same profiler.

Effects

Effects in the model can be more than just a list of columns, and can have a specialized syntax:

```
Effects( list of effects, list of effect macros, or both lists );
```

An effect can be a column name, a crossing of several column names with asterisk (*) notation, or nested columns specified with subscript bracket ([ ]) notation. Additional effect options can appear after an ampersand (&) character. Here are some examples:

```
A,                  // a column name alone is a main effect
A*B,                // a crossed effect, interaction, or polynomial
A[B],               // nested
A*B[C D],           // crossed and nested
effect&Random,      // a random effect
effect&LogVariance, // a variance model term
effect&RS,          // a response surface term
effect&Mixture,     // for an effect participating in a mixture
effect&Excluded,    // for an effect that with no model arguments
effect&Knotted,     // for a knotted spline effect
```

Effect macros are:

```
Factorial( columns ),   // for a full factorial design
Factorial2( columns ),  // for up to 2nd-degree interactions only
Polynomial( columns ),  // for a 2nd-degree polynomial only
```

Responses and Effects for MANOVA

To address an individual response function analysis, use a subscripted `Response`:

```
manovaObj << ( Response[ 1 ] << {response options} );
manovaObj << ( Response[ "Contrast" ] << {response options} );
```

Each response function supports the `Custom Test` message:

```
Custom Test( matrix, <Power Analysis( ... )>, <Label( "..." )> )
```

where each row of the *matrix* specifies coefficients for all the arguments in the model.

To address an individual Effect test, use subscripted `Effect` with a name or number:

```
manovaObj << ( Response[1] << ( Effect["Whole Model"] << {effect options} ) );
manovaObj << ( Response[1] << ( Effect[i] << {effect options} ) );
```

The effects are numbered:

- 0 for the intercept
- 1, 2, and so on, for regular effects
- *n*+1 for the "Whole Model" test, where *n* is the number of effects *not* including the intercept

Each effect in each response function supports the following messages, where each row of the *matrix* has coefficients for all the levels in the effect:

```
Test Details( 1 ),
Centroid Plot( 1 ),
```

```
Save Canonical Scores,
Contrast( matrix, <Power Analysis(...)> );
```

For example, the following JSL script adds test details for an effect to the report window:

```
dt = Open( "$SAMPLE_DATA/Dogs.jmp" );
manObj = dt << Fit Model(
   Y( logHist0, logHist1, logHist3, logHist5 ),
   Effects( dep1, drug, drug * dep1 ),
   Personality( "MANOVA" ),
   Run Model
);
manObj << Response Function( "Contrast" );
manObj << (Response["Contrast"] << (Effect["Whole Model"] <<

/* send the Test Details message to the Whole Model outline
under Contrast in the report window */
Test Details( 1 )));

/* send the Test Details message to response 1 (Contrast)
and effect 3 (drug*dep1) */
manObj << (Response[1] << (Effect[3] << Test Details( 1 )));
```

Send Function

Tip: Recall that the << operator is equivalent to the `Send()` function.

When you have multiple responses, you can send messages to a specific response column's fit. Use the following JSL (where *responseName* is the specific response):

```
fitObj << ( responseName << {options, ...});
```

The second `Send()` function finds the named response and sends the list of messages to it.

Note: If you send the messages directly to the *fitObj* with a single `Send()` function, the messages are sent to all responses.

In the following script, the last line sends a message to the AICc report for the ABRASION response:

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
fitObj = dt << Fit Model(
   Y( :ABRASION, :MODULUS, :ELONG, :HARDNESS ),
   Effects(
      :SILICA & RS,
      :SILANE & RS,
      :SULFUR & RS,
```

```
        :SILICA * :SILICA,
        :SILANE * :SILICA,
        :SILANE * :SILANE,
        :SULFUR * :SILICA,
        :SULFUR * :SILANE,
        :SULFUR * :SULFUR
    ),
    Personality( "Standard Least Squares" ),
    Run
);
/* shows up under the Studentized Residuals plot*/
fitObj << (:ABRASION << {AICc( 1 )});
```

To send messages to an individual effect, nest even further:

```
fitObj << ( responseName << ((effectName) << effectOption ));
```

Standard Least Squares

For Standard Least Squares models that contain only fixed effects and more than one Y response, you can choose to fit the models for the Y responses together or separately. If some of the rows have missing values, the following apply:

- In a script, the Y responses are fit together by default. The model fits each Y using only those rows that are nonmissing for all of the Y variables. For example, a row that includes one or more missing Y values is *excluded* from the model. You can explicitly include `Run( "Fit Together" )` in a script to get the same result.
- If you include the `Run( "Fit Separately" )` option, the model fits each Y using all rows that are nonmissing for that particular Y. For example, a row that includes one or more missing Y values is *included* in the model.

The following script fits the Y responses together:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Fit Model(
    Y( :height, :weight ),
    Effects( :sex ),
    Personality( "Standard Least Squares" ),
    Emphasis( "Minimal Report" ),
    Run // or Run( "Fit Together" )
);
```

The following script fits the Y responses separately:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Fit Model(
    Y( :height, :weight ),
    Effects( :sex ),
```

```
    Personality( "Standard Least Squares" ),
    Emphasis( "Minimal Report" ),
    Run( "Fit Separately" )
);
```

When running the model in JMP, the user is prompted to select fitting the responses separately or together. The Fit Model launch window contains a Fit Separately option, which is deselected by default. For more information about how missing Y values are handled, see *Fitting Linear Models*.

Note: In models that contain a random effect, Y values are fit separately by default.

Formula Depot

Sending commands via JSL to a Formula Depot platform report requires that you first create an empty Formula Depot. Once you have a reference to the Formula Depot report, you can send commands to it. The following script opens a formula depot report and adds a model to the depot.

```
dt = Open( "$SAMPLE_DATA/Liver Cancer.jmp" );
fd = Formula Depot(); // open a new Formula Depot
obj1 = dt << Run Script( "Lasso Poisson, Validation Column" );
model = obj1 << (Fit[1] << Publish Prediction Formula );

/* Publish a model to the Formula Depot. The
following script opens a formula depot report, adds a model to
the depot, generates scoring code for the formula, saves the code to a file,
and closes all windows.*/
case_name = "fitLS_prediction";
dest_dir = "c:\"; //CHANGE ME to a better location
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Print( "Running " || case_name );
fm_ls = Fit Model(
   Y( :weight ),
   Effects( :age, :sex, :height ),
   Personality( "Standard Least Squares" ),
   Emphasis( "Minimal Report" ),
   Run
);
fd = fm_ls << Publish Prediction Formula;
Try(
   Print( "Generate Python Code" );
   code = fd << Generate Python Code( No Editor );
   Save Text File( dest_dir || case_name || ".py", code );
   Print( "Done." );
```

```
,
  Print( "Code generation failed for " || case_name || ", reason: " );
  Show( exception_msg );
);
fdw = Window( "Report: Formula Depot" );
fdw << Close Window( No Save );
Close( dt, No Save );
```

Graph Builder

The following is an example of the basic structure of a Graph Builder script:

```
Graph Builder(
Size( x, y ),
// Use any commands from the red triangle menu - Size is an example
Variables( role( column ), ...), //Required
Elements( element name( element options, ...), //Required
Data Filter( ... ),
// Use any display customizations here - Data Filter is an example
);
```

The following is an example of a simple Graph Builder script that creates a bar chart:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Graph Builder(Variables(
  X( :age ),
  Y( :weight ) ),
Elements( Bar( X, Y, Legend( 8 ) ) )
// Legend is not required, but provides an internal legend ID that you can use
  for later customization
);
```

The following is a more detailed Graph Builder script that creates a bar chart:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Graph Builder(
Size( 373, 332 ),
Show Control Panel( 0 ),
Variables(
  X( :age ),
  Y( :weight ),
  Y( Transform Column( "weight25",
       Formula( Col Quantile( :weight, 0.25, :age ) )
       ),
     Position( 1 )
/* By default each Y variable gets its own axis, but the Position(1)
option keeps each additional Y in the first slot and merges the axes. */
  ),
```

```
    Y( Transform Column( "weight75", Formula( Col Quantile( :weight, 0.75, :age
    ) ) ),
       Position( 1 )
    )
  ),
  Elements(
    Bar( X, Y( 2 ), Y( 3 ), Legend( 10 ), Bar Style( "Range" ) ),
  // This bar element uses only the second and third Y variables and
  // draws a Range bar between the values.*/
    Bar( X, Y( 1 ), Legend( 8 ),
       Bar Style( "Float" ),Summary Statistic( "Median" )
  // This bar element uses the first Y variable and draws its median with the
  // float style, which is a short horizontal line
    ),
    Points( X, Y( 1 ), Legend( 9 ) )
  // The Points element shows the raw data as markers.
  ),
  SendToReport(
  // Contains display customizations
    Dispatch( {}, "400", ScaleBox,
  // 400 is the fixed name
       {Legend Model( 10,
       Level Name( 0, "1st to 3rd
       quartiles",
       Item ID( "Mean(weight25)..Mean(weight75)", 1 )
  // Renames the legend, moves the legend name onto several lines, and changes
  // the color for legend 10 (the Range bar)
    ),
       Properties( 0, {Fill Color( 32 )},
       Item ID( "Mean(weight25)..Mean(weight75)", 1 )
  // Changes the color for legend 10 (the range bar)
    )
       ), Legend Model( 9,
       Properties( 0, {Marker Size( 4 )}, Item ID( "weight",1 ) )
  // Changes the marker size for legend 9 (the Points element)
    )}
  ),
  Dispatch( {}, "graph title", TextEditBox,
    {Set Text( "Weights with quartile ranges" )}
  // Changes the graph title
  ),
  Dispatch( {}, "Y title", TextEditBox, {Set Text( "weight" )} ),
  // Changes the Y axis title
  Dispatch( {}, "400", LegendBox,
    {Legend Position( {10, [0], 8, [1], 9, [-1]} )}
  // Hides the legend item for legend 9 by giving it a position of -1
```

```
    )
)
);
```

Neural and Neural Net

The Neural platform replaces the Neural Net platform. Scripts that use Neural Net are deprecated and might not work in the future. It is recommended to use the Neural platform for new scripts and update any old scripts using Neural Net to Neural.

Partial Least Squares and PLS

The Partial Least Squares (PLS) platform has replaced the PLS platform. Scripts that use PLS might not work in the future. It is recommended to use the Partial Least Squares platform for new scripts and update any old scripts using PLS to Partial Least Squares.

Process Capability

The Process Capability platform has replaced the Capability platform. Scripts that use Capability might not work in the future. It is recommended to use the Process Capability platform for new scripts and update any old scripts using Capability to Process Capability.

Specification Limits in JSL Scripts

Specification limits can be read from JSL scripts, column properties, or from a specification limits data table. As an example of reading in specification limits from JSL, consider the following example:

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Process Capability(
   Process Variables( :OZONE, :CO, :SO2, :NO ),
   Spec Limits(
      OZONE( LSL( 0.075 ), Target( 0.15 ), USL( 0.25 ) ),
      CO( LSL( 5 ), Target( 7 ), USL( 12 ) ),
      SO2( LSL( 0.01 ), Target( 0.04 ), USL( 0.09 ) ),
      NO( LSL( 0.01 ), Target( 0.025 ), USL( 0.04 ) )
   )
);
```

Using a Limits Data Table and JSL

There is no extra syntax needed to differentiate between the two table types (wide and tall) when they are read using JSL. This example is based on the CitySpecLimits.jmp sample data table. It places the specification limits data table inside an `Import Spec Limits()` expression.

```
dt = Open( "$SAMPLE_DATA/Cities.jmp" );
dt << Process Capability(
   Process Variables( :OZONE, :CO, :SO2, :NO ),
   Spec Limits(
      Import Spec Limits(
         "$SAMPLE_DATA/CitySpecLimits.jmp"
)));
```

Scatterplot 3D

If your machine supports acceleration, and you want to display the 3-D scatterplot faster, you can turn on the Use Hardware Acceleration option, or in JSL, specify the `Scene3DHardwareAcceleration` global variable. A value of 1 turns it on, and a value of zero turns it off.

```
dt = Open( "$SAMPLE_DATA/Solubility.jmp" );
Scene3DHardwareAcceleration = 1;
dt << Scatterplot 3D(
   Y( :"1-Octanol"n, :Ether, :Chloroform, :Benzene, :Carbon Tetrachloride,
   :Hexane )
);
```

Text Explorer

By default, text is converted to all lowercase in the Text Explorer platform. You can use the `"From Source"` keyword in the Save Regex Column message to save the results of a custom regex that preserve the original case of the text. The first argument is the name of the column that is created in the data table. The second argument is the string `"From Source"` or `"From Result"`. The default behavior is equivalent to specifying the `"From Result"` keyword.

```
dt = Open( "$SAMPLE_DATA/Aircraft Incidents.jmp" );
obj = dt << Text Explorer(
   Text Columns( :Airport Name),
   Tokenizing( "Regex" ),
   Save Regex Column( "Output Column", "From Source" )
);
obj << Close Window();
```

Explore Outliers

The Explore Outliers platform provides options to identify, explore, and manage outliers in your univariate or multivariate data.

In the Probe.jmp sample data table, several columns contain outlier values of 9999. Many industries use nines as a missing value code. The following example selects VDP_PCOLL, VDP_PINNBASE, and VDP_PINPBASE, the columns with the highest number of 9s, and adds the Missing Value Codes column property to the columns. The script then rescans the columns and removes these columns from the report.

```
dt = Open( "$SAMPLE_DATA/Probe.jmp" );
obj = Explore Outliers(
   Y(
      :VDP_M1,
      :VDP_M2,
      :VDP_NBASE,
      :VDP_NEMIT,
      :VDP_NENBNI,
      :VDP_NSINK,
      :VDP_PBASE,
      :VDP_PBL,
      :VDP_PCOLL,
      :VDP_PEMIT,
      :VDP_PINNBASE,
      :VDP_PINPBASE,
      :VDP_PSINK,
      :VDP_SICR
   ),
   Quantile Range Outliers( 1 ),
   Show only columns with outliers( 1 ),
);
obj << Add Highest Nines to Missing Value Codes( :VDP_PCOLL, :VDP_PINNBASE,
   :VDP_PINPBASE );
obj << Rescan;
```

Scripting-Only Messages and Arguments by Platform

Some messages and arguments that can be sent to a platform object are scripting-only. This means that these options do not have corresponding options in JMP launch windows or reports. However, note that scripting-only options can appear as Preferences.

This section describes scripting-only messages and arguments for specific platforms.

Tip: For more information about any of these messages or arguments, and for specific examples, go to the Scripting Index (**Help** > **Scripting Index**).

- "Bivariate Script-Only Messages"
- "Boosted Tree Script-Only Messages"
- "Bootstrap Forest Script-Only Messages"
- "Bubble Plot Script-Only Messages"
- "Choice Script-Only Messages"
- "Clustering Script-Only Messages"
- "Contingency Script-Only Messages"
- "Contour Plot Script-Only Messages"
- "Contour Profiler Script-Only Messages"
- "Control Chart Script-Only Messages"
- "Control Chart Builder Script-Only Messages"
- "Cumulative Damage Script-Only Messages"
- "Custom Profiler Script-Only Messages"
- "Degradation Script-Only Messages"
- "Destructive Degradation Script-Only Messages"
- "Discriminant Script-Only Messages"
- "Distribution Script-Only Messages"
- "EMP Measurement Systems Analysis Script-Only Messages"
- "Explore Missing Values Script-Only Messages"
- "Explore Outliers Script-Only Messages"
- "Factor Analysis Script-Only Messages"
- "Fit Life by X Script-Only Messages"
- "Fit Model Script-Only Messages"
- "Fit Parametric Survival Script-Only Messages"
- "Functional Data Explorer Script-Only Messages"
- "Gaussian Process Script-Only Messages"
- "K Means Cluster Script-Only Messages"
- "K Nearest Neighbors Script-Only Messages"
- "Latent Class Analysis Script-Only Messages"
- "Life Distribution Script-Only Messages"
- "Make Validation Column Script-Only Messages"
- "MaxDiff Script-Only Messages"

- "Mixture Profile Script-Only Messages"
- "Model Screening Script-Only Messages"
- "Multivariate Script-Only Messages"
- "Naive Bayes Script-Only Messages"
- "Neural Script-Only Messages"
- "Nonlinear Script-Only Messages"
- "Normal Mixtures Script-Only Messages"
- "Oneway Script-Only Messages"
- "Parallel Plot Script-Only Messages"
- "Pareto Plot Script-Only Messages"
- "Partial Least Squares Script-Only Messages"
- "Partition Script-Only Messages"
- "Principal Components Script-Only Messages"
- "Process Capability Script-Only Messages"
- "Process Screening Script-Only Messages"
- "Profiler Script-Only Messages"
- "Reliability Forecast Script-Only Messages"
- "Reliability Growth Script-Only Messages"
- "Response Screening Script-Only Messages"
- "Scatterplot 3D Script-Only Messages"
- "Structural Equation Models Script-Only Messages"
- "Support Vector Machines Script-Only Messages"
- "Surface Plot Script-Only Messages"
- "Tabulate Script-Only Messages"
- "Ternary Plot Script-Only Messages"
- "Text Explorer Script-Only Messages"
- "Time Series Script-Only Messages"
- "Uplift Script-Only Messages"
- "Variability Chart Script-Only Messages"

Bivariate Script-Only Messages

The Fit Where message for a Bivariate object fits a curve or line to a single level of a categorical variable.

```
biv_object << Fit Where( WHERE_clause, fit_command );
```

The Curve message for a Bivariate object enables you to send a message to a curve or line that has already been fit in the Bivariate report. The curves are numbered sequentially in the order they appear in the report.

```
biv_object << ( Curve[n] << curve_message );
```

The Set Kernel message for a Nonpar Density curve in a Bivariate object enables you to set the kernel standard deviations for the X and Y values:

```
biv_object << ( Curve[n] << Set Kernel( x, y ) );
```

Boosted Tree Script-Only Messages

When you launch the Boosted Tree platform via JSL, you can use the following arguments that are available only in JSL:

```
Maximum Depth( number )
Use Excluded Rows for Validation( 0|1 )
```

The following messages for a Boosted Tree object are available only in JSL:

```
boosted_tree_object << Get Average Absolute Error Test;
boosted_tree_object << Get Average Absolute Error Training;
boosted_tree_object << Get Average Absolute Error Validation;
boosted_tree_object << Get Average Log Error Test;
boosted_tree_object << Get Average Log Error Training;
boosted_tree_object << Get Average Log Error Validation;
boosted_tree_object << Get Confusion Matrix Test;
boosted_tree_object << Get Confusion Matrix Training;
boosted_tree_object << Get Confusion Matrix Validation;
boosted_tree_object << Get Confusion Rates Test;
boosted_tree_object << Get Confusion Rates Training;
boosted_tree_object << Get Confusion Rates Validation;
boosted_tree_object << Get Gen RSquare Test;
boosted_tree_object << Get Gen RSquare Training;
boosted_tree_object << Get Gen RSquare Validation;
boosted_tree_object << Get Measures;
boosted_tree_object << Get Microseconds;
boosted_tree_object << Get Misclassification Rate Test;
boosted_tree_object << Get Misclassification Rate Training;
boosted_tree_object << Get Misclassification Rate Validation;
boosted_tree_object << Get MM SAS DATA Step;
boosted_tree_object << Get MM Tolerant SAS DATA Step;
boosted_tree_object << Get Prediction Formula;
boosted_tree_object << Get RMS Error Test;
boosted_tree_object << Get RMS Error Training;
boosted_tree_object << Get RMS Error Validation;
boosted_tree_object << Get ROC Area Test;
```

```
boosted_tree_object << Get ROC Area Training;
boosted_tree_object << Get ROC Area Validation;
boosted_tree_object << Get RSquare Test;
boosted_tree_object << Get RSquare Training;
boosted_tree_object << Get RSquare Validation;
boosted_tree_object << Get SAS DATA Step;
boosted_tree_object << Get Seconds;
boosted_tree_object << Get Tolerant Prediction Formula;
boosted_tree_object << Get Tolerant SAS DATA Step;
```

Bootstrap Forest Script-Only Messages

When you launch the Boosted Tree platform via JSL, you can use the following arguments that are available only in JSL:

```
Portion Bootstrap( fraction )
Time Limit( number )
Use Excluded Rows for Validation( 0|1 )
```

The following messages for a Bootstrap Forest object are available only in JSL:

```
bootstrap_forest_object << Get Average Absolute Error Test;
bootstrap_forest_object << Get Average Absolute Error Training;
bootstrap_forest_object << Get Average Absolute Error Validation;
bootstrap_forest_object << Get Average Log Error Test;
bootstrap_forest_object << Get Average Log Error Training;
bootstrap_forest_object << Get Average Log Error Validation;
bootstrap_forest_object << Get Confusion Matrix Test;
bootstrap_forest_object << Get Confusion Matrix Training;
bootstrap_forest_object << Get Confusion Matrix Validation;
bootstrap_forest_object << Get Confusion Rates Test;
bootstrap_forest_object << Get Confusion Rates Training;
bootstrap_forest_object << Get Confusion Rates Validation;
bootstrap_forest_object << Get Gen RSquare Test;
bootstrap_forest_object << Get Gen RSquare Training;
bootstrap_forest_object << Get Gen RSquare Validation;
bootstrap_forest_object << Get Measures;
bootstrap_forest_object << Get Microseconds;
bootstrap_forest_object << Get Misclassification Rate Test;
bootstrap_forest_object << Get Misclassification Rate Training;
bootstrap_forest_object << Get Misclassification Rate Validation;
bootstrap_forest_object << Get MM SAS DATA Step;
bootstrap_forest_object << Get MM Tolerant SAS DATA Step;
bootstrap_forest_object << Get Prediction Formula;
bootstrap_forest_object << Get RMS Error Test;
bootstrap_forest_object << Get RMS Error Training;
bootstrap_forest_object << Get RMS Error Validation;
```

```
bootstrap_forest_object << Get ROC Area Test;
bootstrap_forest_object << Get ROC Area Training;
bootstrap_forest_object << Get ROC Area Validation;
bootstrap_forest_object << Get RSquare Test;
bootstrap_forest_object << Get RSquare Training;
bootstrap_forest_object << Get RSquare Validation;
bootstrap_forest_object << Get SAS DATA Step;
bootstrap_forest_object << Get Seconds;
bootstrap_forest_object << Get Tolerant Prediction Formula;
bootstrap_forest_object << Get Tolerant SAS DATA Step;
```

Bubble Plot Script-Only Messages

The following messages for a Bubble Plot object are available only in JSL:

```
bubble_plot_object << Auto Stretching( string );
bubble_plot_object << Bubble Size( number );
bubble_plot_object << Color Levels( vector );
bubble_plot_object << Get Custom Path;
bubble_plot_object << Get Draw;
bubble_plot_object << Get Label;
bubble_plot_object << Get Shape;
bubble_plot_object << Go;
bubble_plot_object << Label Offset( list );
bubble_plot_object << Prev;
bubble_plot_object << Set Custom Path( ... );
bubble_plot_object << Speed( number );
bubble_plot_object << Step;
bubble_plot_object << Stop;
bubble_plot_object << Time Index( number );
bubble_plot_object << Title Position( x, y );
bubble_plot_object << Toggle Animation;
```

Choice Script-Only Messages

When you launch the Choice platform via JSL, you can set the acceptable criterion for parameter convergence. In addition, when using Hierarchical Bayes, you can set the number of burn-in iterations.

```
Convergence Criterion( fraction )
Number of Burn In Iterations ( number)
```

Clustering Script-Only Messages

The following messages for a Hierarchical Cluster object are available only in JSL:

```
hierarchical_cluster_object << Get Column Names;
hierarchical_cluster_object << Get Clusters;
hierarchical_cluster_object << GetDisplayOrder;
hierarchical_cluster_object << GetColumnDisplayOrder;
hierarchical_cluster_object << Get Distance Matrix;
hierarchical_cluster_object << Standardize( 0|1 );
```

The following messages for a KDTable object are available only in JSL:

```
kdtable_object << Distance between rows( row1, row2 );
kdtable_object << Insert rows( number|[ vector ] );
kdtable_object << K nearest rows( stop, <position> );
kdtable_object << Remove rows( number|[ vector ] );
```

Contingency Script-Only Messages

The following message is available only in JSL. It creates a data table that contains values from the Correspondence Analysis plot in a Contingency report.

```
contingency_obj << Correspondence Analysis( Make Table );
```

Contour Plot Script-Only Messages

The following message for a Contour Plot object is available only in JSL:

```
contour_plot_object << Set Alpha( number );
```

Contour Profiler Script-Only Messages

The following messages for a Contour Profiler object are available only in JSL:

```
contour_profile_object << Contour Value( y1( ... ), y2( ... ), ... );
contour_profile_object << Get Constraints;
contour_profile_object << Get Factor Settings;
contour_profile_object << Get Factor Settings Script;
contour_profile_object << Get Simulator;
contour_profile_object << Horizontal Factor( column );
contour_profile_object << Number of Plots Across( number );
contour_profile_object << Reset;
contour_profile_object << Term Value( x1( number ), x2( number ), ... );
contour_profile_object << Vertical Factor( column );
contour_profile_object << Y Colors( color1, color2, ... );
```

Control Chart Script-Only Messages

When you launch the Control Chart platform via JSL, you can use the following arguments that are available only in JSL:

```
Chart Col( column, chart_type( Avg( number ), UCL( number ), LCL( number ) )
Chart Type( string )
Get Limits( string )
Get Targets( string ) //Multivariate Control Charts
Use Excluded Points on MR
```

The following messages for a Control Chart object are available only in JSL:

```
control_chart_object << Point Marker( string ); //CUSUM charts
control_chart_object << Save Limits( string );
```

Control Chart Builder Script-Only Messages

When you launch the Control Chart Builder platform via JSL, you can use the following arguments that are available only in JSL:

```
Chart( position( i ), Add Limits( ... ) )
Chart( position( i ), Add Spec Limits( ... ) )
Chart( position( i ), Limits( ... ) )
Chart( position( i ), Points( ... ) )
Chart( position( i ), Set Control Limits( ... ) )
Class( string )
Customize Tests( ... )
KSigma( number )
Show Lower Control Limit( 0|1 )
Show Two Shewhart Charts( 0|1 )
Variables( ... )
Variables( ..., nTrials( column|number ) )
```

The following messages for a Control Chart Builder object are available only in JSL:

```
control_chart_builder_object << Chart( position( i ), ... );
control_chart_builder_object << Rerun All Tests;
control_chart_builder_object << Save Limits( string );
control_chart_builder_object << Size( width, height );
```

Cumulative Damage Script-Only Messages

The following message for a Cumulative Damage object is available only in JSL. It returns a list of model fit results.

```
cumulative_damage_object << Get Results;
```

Custom Profiler Script-Only Messages

The following messages for a Custom Profiler object are available only in JSL:

```
custom_profiler_object << Get Constraints;
custom_profiler_object << Get Factor Settings;
custom_profiler_object << Get Factor Settings Script;
custom_profiler_object << Get Objective;
custom_profiler_object << Get Objective Formula;
custom_profiler_object << Get Simulator;
custom_profiler_object << Objective Formula;
custom_profiler_object << Reset;
custom_profiler_object << Term Value( x1( number ), x2( number ), ... );
```

Degradation Script-Only Messages

The following messages for a Degradation object are available only in JSL:

```
degradation_object << Get Inverse Prediction Results;
degradation_object << Get Prediction Results;
degradation_object << Get Residuals;
degradation_object << Get Results;
degradation_object << Residual Plot;
degradation_object << Set Baseline;
```

Destructive Degradation Script-Only Messages

The following messages for a Destructive Degradation object are available only in JSL:

```
destructive_degradation_object << Control( yTrans, xTrans, distribution, path
  );
destructive_degradation_object << Model( yTrans, xTrans, distribution, path
  );
```

Discriminant Script-Only Messages

The following messages for a Discriminant object are available only in JSL:

```
discriminant_object << Get Descrim Matrices;
discriminant_object << Get Measures;
discriminant_object << Score Data( 0|1 );
```

Distribution Script-Only Messages

The following message for a Distribution object is available only in JSL. The message creates an array of handles to the fitted distributions, so you can send commands to specific distributions:

```
dist_object << ( Fit Handle[n] << ...);
```

You can also get a reference to a fitted distribution directly if you send the Fit Distribution message to a Distribution object. This enables you to send commands to specific distributions:

```
fit_object = dist_object << Fit Distribution( distribution_name );
fit_object << ...;
```

The following message for a Distribution object is available only in JSL. The message specifies that PpK labeling be used in the Process Capability reports within the Distribution report object.

```
dist_object << PpK Capability Labeling( 1 );
```

The following message for a Distribution object is available only in JSL:

```
dist_object << Outlier Box Plot Row Cutoff( number );
```

The following messages for a Distribution object use the name of a fitted distribution, such as `Fit Lognormal`, and they are available only in JSL:

```
dist_object << Fit Distribution Name( Process Capability( Set Proabilities for
  Quantile Spec Limits( ... ) );
dist_object << Fit Distribution Name( Process Capability( Set Sigma Multiplier
  for Quantile Spec Limits( ... ) );
```

EMP Measurement Systems Analysis Script-Only Messages

The following messages for an EMP Measurement Systems Analysis object are available only in JSL:

```
emp_msa_object << Conv Limit( number );
emp_msa_object << Dispersion Chart( 0|1 );
emp_msa_object << Include Interactions in Reproducibility( 0|1 );
emp_msa_object << Max Iter( number );
emp_msa_object << Set Alpha Level( number );
emp_msa_object << Set Random Seed( number );
emp_msa_object << Show Monitor Classification Legend( 0|1 );
emp_msa_object << Show Part Legend( 0|1 );
emp_msa_object << Show Shift Detection Profiler Legend( 0|1 );
```

Explore Missing Values Script-Only Messages

When you launch Automated Data Imputation through the Explore Missing Values utility via JSL, you can use the following arguments that are available only in JSL:

```
Maximum Dimension( number )
Validation Portion( number )
```

The following messages for an Explore Missing Values report are available only in JSL:

```
explore_missing_values_object << Get U V Sigma ADI Matrices;
```

Explore Outliers Script-Only Messages

When you launch K Nearest Neighbors Outliers through the Explore Outliers utility via JSL, you can use the following arguments that are available only in JSL:

```
Impute Missing( 0|1 )
```

The following messages for a Quantile Range Outliers object in Explore Outliers object are available only in JSL:

```
explore_outliers_obj << Get Quantile Outliers;
explore_outliers_obj << Save Quantile Outlier Limits;
```

When you launch Robust PCA Outliers through the Explore Outliers utility via JSL, you can use the following arguments that are available only in JSL:

```
Tolerance( number )
```

Factor Analysis Script-Only Messages

The following messages for a Factor Analysis model fit are available only in JSL. The model fits are numbered sequentially in the order they appear in the report.

```
factor_analysis_object << ( Fit[n] << Save Rotated Components );
factor_analysis_object << ( Fit[n] << Save Rotated Components with Imputation
  );
```

Fit Life by X Script-Only Messages

The following messages for a Fit Life by X object are available only in JSL:

```
fit_life_object << Add Density Curve to Scatterplot( number );
fit_life_object << Add Quantile Line to Scatterplot( number );
fit_life_object << Density( ... );
fit_life_object << Get Results;
fit_life_object << Hazard( ... );
```

```
fit_life_object << Maximum Iterations( number );
fit_life_object << Probability( ... );
fit_life_object << Quantile( ... );
fit_life_object << Rejection Sample Maximum Trials( number );
fit_life_object << Set Level of Quantile Line CI Bands( number );
fit_life_object << Set Scale( distribution );
fit_life_object << Set Scriptables( ... );
fit_life_object << TAF( ... );
fit_life_object << Time Acceleration Baseline( number );
```

Fit Model Script-Only Messages

The following messages for a Model Dialog object are available only in JSL:

```
model_dialog_object << Cauchy Fit( 0|1 ); //Response Screening
model_dialog_object << Choose High Target( 0|1 ); //Nominal Logistic
model_dialog_object << Generalized Distribution( string ); //Generalized
  Regression
model_dialog_object << GLM Distribution( string ); //Generalized Linear Model
model_dialog_object << Max Iterations( number ); //Partial Least Squares
model_dialog_object << Power Link Parameter( number ); //Generalized Linear
  Model
model_dialog_object << Robust Fit( 0|1 ); //Response Screening
model_dialog_object << Run Model;
```

The following sections describe scripting-only messages for MANOVA, Generalized Linear Models, Generalized Regression, Nominal and Ordinal Logistic, Standard Least Squares, and Stepwise.

MANOVA

The following message for a Fit MANOVA object is available only in JSL:

```
fit_model_object << ( Response[1] << (Effect[1] << ... ) );
```

The third message can be one of the following:

```
Test Details // show or hide the test details for an individual effect
Centroid Plot // show or hide the centroid plot for an individual effect
Save Canonical Scores // save the canonical scores for an individual effect
Contrast /* run a customized F test contrasting different levels for
        an effect in the model*/
```

The following example saves canonical scores for the Response Function, which in this data, is Sum:

```
dt = Open( "$SAMPLE_DATA/Dogs.jmp" );
obj = dt << Fit Model(
  Y( :LogHist0, :LogHist1, :LogHist3, :LogHist5 ),
```

```
    Effects( :drug, :dep1, :drug * :dep1 ),
    Personality( "Manova" ),
    Run( Response Function( Sum ) )
);
obj << (Response[1] << (Effect[1] << Save Canonical Scores) );
```

Generalized Linear Models

The following messages for a Fit Generalized Linear Model object are available only in JSL:

```
fit_model_object << GLM Distribution( distribution );
fit_model_object << Parametric Formula;
fit_model_object << Power Link Parameter( number );
```

Generalized Regression

The following messages for a Fit Generalized Regression object are available only in JSL:

```
fit_model_object << Fit( ... );
fit_model_object << Get X Matrix;
```

The following messages for a Fit Generalized Regression object are available only in JSL and must be sent to an individual model fit. The model fits are numbered sequentially in the order they appear in the report.

```
fit_model_object << (Fit[n] << Get Prediction Formula);
fit_model_object << (Fit[n] << Set Solution ID( number ));
fit_model_object << (Fit[n] << Step Backward);
fit_model_object << (Fit[n] << Step Forward);
```

Nominal and Ordinal Logistic

The following messages for Fit Nominal Logistic and Fit Ordinal Logistic objects are available only in JSL:

```
fit_model_object << Get Confusion Matrix Test;
fit_model_object << Get Confusion Matrix Training;
fit_model_object << Get Confusion Matrix Validation;
fit_model_object << Get Confusion Rates Test;
fit_model_object << Get Confusion Rates Training;
fit_model_object << Get Confusion Rates Validation;
fit_model_object << Get MM SAS Data Step;
fit_model_object << Get SAS Data Step;
fit_model_object << Get Measures;
fit_model_object << Get Probability Formulas;
```

The following message for a Fit Nominal Logistic object is available only in JSL:

```
fit_model_object << Positive Level( level );
```

Standard Least Squares

The following messages for a Fit Least Squares object are available only in JSL:

```
fit_model_object << Get Conditional Formula;
fit_model_object << Get Effect Names;
fit_model_object << Get Effect PValues;
fit_model_object << Get Estimates;
fit_model_object << Get Indiv Confid Limit Formula;
fit_model_object << Get Mean Confid Limit Formula;
fit_model_object << Get Measures;
fit_model_object << Get MM SAS DATA Step;
fit_model_object << Get Parameter Names;
fit_model_object << Get Parameterized Formula;
fit_model_object << Get Prediction Formula;
fit_model_object << Get Random Effect Names;
fit_model_object << Get Std Errors;
fit_model_object << Get Standard Error Formula;
fit_model_object << Get Variance Components;
fit_model_object << Get X Matrix;
fit_model_object << Get XPX Inverse;
fit_model_object << Get Y Matrix;
fit_model_object << Profiler( Extrapolation Control Criterion( string ) );
fit_model_object << Show VIF;
```

Stepwise

The following messages for a Fit Stepwise object are available only in JSL:

```
fit_stepwise_object << Backward Step;
fit_stepwise_object << Enter( column );
fit_stepwise_object << Finish; // preferred in scripts over << Go
fit_stepwise_object << Forward Step;
fit_stepwise_object << Get Measures;
fit_stepwise_object << Lock( column );
fit_stepwise_object << Prob to Enter( number );
fit_stepwise_object << Prob to Leave( number );
fit_stepwise_object << Remove( column );
fit_stepwise_object << Unlock( column );
```

Fit Parametric Survival Script-Only Messages

The following messages for a Fit Parametric Survival object are available only in JSL:

```
fit_parametric_survival_object << Estimate Time Quantile( ... );
fit_parametric_survival_object << Get Parameter Names;
fit_parametric_survival_object << Get Estimates;
```

```
fit_parametric_survival_object << Get Std Errors;
fit_parametric_survival_object << Get Effect Names;
fit_parametric_survival_object << Get Effect PValues;
```

Functional Data Explorer Script-Only Messages

The following messages for a Functional Data Explorer FDOE object are available only in JSL:

```
fdoe_object << Generalized Regression FPC Model( FPC Number( n ), Commands );
```

Gaussian Process Script-Only Messages

When you launch the Gaussian Process platform via JSL, you can use the following arguments that are available only in JSL:

```
Set Nugget( number )
```

K Means Cluster Script-Only Messages

The followings messages for a K Means Cluster object are available only in JSL:

```
kmeans_cluster_object << Get Statistics;
```

K Nearest Neighbors Script-Only Messages

When you launch the K Nearest Neighbors platform via JSL, you can use the following arguments that are available only in JSL:

```
Use Excluded Rows for Validation( 0|1 )
```

The following messages for a K Nearest Neighbors object are available only in JSL:

```
knn_object << Get Measures;
knn_object << ( Response[number] << Get Prediction Formula );
knn_object << ( Response[number] << Get Best K );
knn_object << ( Response[number] << Set K( number ) );
```

Latent Class Analysis Script-Only Messages

The following message for a Latent Class Analysis model fit constructs a script to create probability formula columns and returns them to the log. This message is available only in JSL.

```
latent_class_analysis_object << ( Fit[number] << Get Probability Formulas );
```

The following example sets a random seed in order for the model to be reproducible.

```
dt = Open( "$SAMPLE_DATA/Car Poll.jmp" );
obj = dt << Latent Class Analysis(
   Y( :sex, :marital status, :country, :size, :type ),
   Set Random Seed( 12345 ),
   Number of Clusters( 3 ),
);
```

Life Distribution Script-Only Messages

The following messages for a Life Distribution object are available only in JSL:

```
life_dist_obj << Compute Remaining Life Distribution( time0, time1 );
life_dist_object << Confidence Invterval Method( string );
life_dist_object << Density( number );
life_dist_object << Fit DS Frechet;
life_dist_object << Fit DS Loglogistic;
life_dist_object << Fit DS Lognormal;
life_dist_object << Fit DS Weibull;
life_dist_object << Fit Exponential;
life_dist_object << Fit Frechet;
life_dist_object << Fit GenGamma;
life_dist_object << Fit LEV;
life_dist_object << Fit LogGenGamma;
life_dist_object << Fit Logistic;
life_dist_object << Fit Loglogistic;
life_dist_object << Fit Lognormal;
life_dist_object << Fit Model( ... );
life_dist_object << Fit Normal;
life_dist_object << Fit SEV;
life_dist_object << Fit TH Frechet;
life_dist_object << Fit TH Loglogistic;
life_dist_object << Fit TH Lognormal;
life_dist_object << Fit TH Weibull;
life_dist_object << Fit Weibull;
life_dist_object << Fit ZI Frechet;
life_dist_object << Fit ZI Loglogistic;
life_dist_object << Fit ZI Lognormal;
life_dist_object << Fit ZI Weibull;
life_dist_object << Get Causes;
life_dist_object << Get Estimates;
life_dist_object << Get Formula;
life_dist_object << Get Life Distribution( i );
life_dist_object << Get Model Specification;
life_dist_object << Get Results;
```

```
life_dist_object << Hazard( number );
life_dist_object << Mean Remaining Life( Compute( vector ) );
life_dist_object << Omit( i, 0|1 );
life_dist_object << Probability( number );
life_dist_object << Quantile( number );
life_dist_object << Rejection Sample Maximum Trials( number );
life_dist_object << Select Distribution( ... );
life_dist_object << Select Scale( distribution );
life_dist_object << Set Scale( distribution );
life_dist_object << Subdistribution( ... );
life_dist_object << Suppress Plot( plot_name );
```

Make Validation Column Script-Only Messages

When you launch the Make Validation Column platform via JSL, you can use the following arguments that are available only in JSL:

```
Test Set( number )
Training Set( number )
Validation Set( number )
```

MaxDiff Script-Only Messages

When you launch the MaxDiff platform via JSL, you can set the acceptable criterion for parameter convergence. In addition, when using Hierarchical Bayes, you can set the number of burn-in iterations:

```
Convergence Criterion( fraction );
Number of Burn In Iterations ( number);
```

Mixture Profile Script-Only Messages

The following messages for a Mixture Profiler object are available only in JSL:

```
mixture_profiler_object << Contour Value( y1( ... ), y2( ... ), ... );
mixture_profiler_object << Get Constraints;
mixture_profiler_object << Get Factor Settings;
mixture_profiler_object << Get Factor Settings Script;
mixture_profiler_object << Get Simulator;
mixture_profiler_object << Left Factor( column );
mixture_profiler_object << Reset;
mixture_profiler_object << Right Factor( column );
mixture_profiler_object << Term Value( x1( ... ), x2( ... ), ... );
mixture_profiler_object << Top Factor( column );
```

Model Screening Script-Only Messages

When you launch the Model Screening platform via JSL, you can use the following arguments that are available only in JSL:

```
Model NParm Limit( number )
Set Probability Threshold( number )
SVM NRow Limit( number )
```

The following message for a Model Screening object is available only in JSL:

```
model_screening_object << Select Fit( ... );
```

Multivariate Script-Only Messages

The following message for a Multivariate object is available only in JSL:

```
multivariate_object << Get Correlation Matrix;
multivariate_object << Get Inv Correlation Matrix;
```

Naive Bayes Script-Only Messages

When you launch the Naive Bayes platform via JSL, you can use the following arguments that are available only in JSL:

```
Prior Bias( fraction )
Use Excluded Rows for Validation( 0|1 )
```

The following messages for a Naive Bayes object are available only in JSL:

```
naive_bayes_object << Get Measures;
naive_bayes_object << Get Probability Formulas;
```

Neural Script-Only Messages

Note: The Neural platform replaces the Neural Net platform. Scripts that use Neural Net are deprecated and might not work in the future. It is recommended to use the Neural platform for new scripts and update any old scripts using Neural Net to Neural.

When you launch the Neural platform via JSL, you can use the following arguments that are available only in JSL:

```
Multithreading( 0|1 )
N Boost( number )
Validation Method( ... )
```

The following messages for a Neural Fit object are available only in JSL:

```
neural_object << Fit( ... );
neural_object << ( Fit[n] << Get Average Absolute Error Test );
neural_object << ( Fit[n] << Get Average Absolute Error Training );
neural_object << ( Fit[n] << Get Average Absolute Error Validation );
neural_object << ( Fit[n] << Get Average Log Error Test );
neural_object << ( Fit[n] << Get Average Log Error Training );
neural_object << ( Fit[n] << Get Average Log Error Validation );
neural_object << ( Fit[n] << Get Confusion Matrix Test);
neural_object << ( Fit[n] << Get Confusion Matrix Training);
neural_object << ( Fit[n] << Get Confusion Matrix Validation);
neural_object << ( Fit[n] << Get Confusion Rates Test);
neural_object << ( Fit[n] << Get Confusion Rates Training);
neural_object << ( Fit[n] << Get Confusion Rates Validation);
neural_object << ( Fit[n] << Get Gen RSquare Test );
neural_object << ( Fit[n] << Get Gen RSquare Training );
neural_object << ( Fit[n] << Get Gen RSquare Validation );
neural_object << ( Fit[n] << Get Measures );
neural_object << ( Fit[n] << Get Misclassification Rate Test );
neural_object << ( Fit[n] << Get Misclassification Rate Training );
neural_object << ( Fit[n] << Get Misclassification Rate Validation );
neural_object << ( Fit[n] << Get MM SAS DATA Step );
neural_object << ( Fit[n] << Get NBoost );
neural_object << ( Fit[n] << Get Prediction Formula );
neural_object << ( Fit[n] << Get RMS Error Test );
neural_object << ( Fit[n] << Get RMS Error Training );
neural_object << ( Fit[n] << Get RMS Error Validation );
neural_object << ( Fit[n] << Get ROC Area Test);
neural_object << ( Fit[n] << Get ROC Area Training );
neural_object << ( Fit[n] << Get ROC Area Validation);
neural_object << ( Fit[n] << Get RSquare Test );
neural_object << ( Fit[n] << Get RSquare Training );
neural_object << ( Fit[n] << Get RSquare Validation );
neural_object << ( Fit[n] << Get SAS DATA Step );
neural_object << ( Fit[n] << Get Seconds);
```

Nonlinear Script-Only Messages

When you launch the Nonlinear platform via JSL, you can use the following arguments that are available only in JSL:

```
Second Deriv Method( 0|1 )
```

The following messages for a Nonlinear object are available only in JSL:

```
nonlinear_object << CL Alpha( number );
```

```
nonlinear_object << CL Limit( number );
nonlinear_object << Delta( number );
nonlinear_object << Finish;
nonlinear_object << Get CI;
nonlinear_object << Get Corr;
nonlinear_object << Get Cov;
nonlinear_object << Get Estimates;
nonlinear_object << Get Parameter Names;
nonlinear_object << Get SSE;
nonlinear_object << Get Std Errors;
nonlinear_object << Gradient Limit( number );
nonlinear_object << Iteraction Limit( number );
nonlinear_object << Loss is Neg LogLikelihood( 0|1 );
nonlinear_object << Numeric Chain Deriv Delta;
nonlinear_object << Obj Change Limit( number );
nonlinear_object << Relative Gradient( number );
nonlinear_object << Set Parameter( Name = Expr, ... );
nonlinear_object << Unlock Parameter( Name, ... );
```

Normal Mixtures Script-Only Messages

When you launch the Normal Mixtures platform via JSL, you can use the following arguments that are available only in JSL:

```
Mixtures N Starts( number )
Mixtures Tolerance( number )
```

The following messages for a Normal Mixtures object are available only in JSL:

```
normal_mixtures_object << Get Statistics;
```

Oneway Script-Only Messages

The following message for a Oneway object is available only in JSL:

```
oneway_object << Select Group( Xvalue );
```

Parallel Plot Script-Only Messages

The following message for a Parallel Plot object is available only in JSL:

```
parallel_plot_object << Reversed( columns );
```

Pareto Plot Script-Only Messages

The following message for a Pareto Plot object is available only in JSL:

```
No Plot( 0|1 );
```

Partial Least Squares Script-Only Messages

The following messages for a Partial Least Squares object are available only in JSL:

```
pls_object << ( Fit[ number ] << Get Measures ) );
```

Partition Script-Only Messages

When you launch the Partition platform via JSL, you can use the following arguments that are available only in JSL:

```
Initial Splits( ... )
Missing Value Order( Low( list of numeric columns ), High( list of numeric
  columns ) )
Multithreading( 0|1 )
Set Random Seed( number )
```

The following messages for a Partition object that return the requested item from the fitted model are available only in JSL:

```
partition_object << Get Average Absolute Error Test;
partition_object << Get Average Absolute Error Training;
partition_object << Get Average Absolute Error Validation;
partition_object << Get Average Log Error Test;
partition_object << Get Average Log Error Training;
partition_object << Get Average Log Error Validation;
partition_object << Get Confusion Matrix Test;
partition_object << Get Confusion Matrix Training;
partition_object << Get Confusion Matrix Validation;
partition_object << Get Confusion Rates Test;
partition_object << Get Confusion Rates Training;
partition_object << Get Confusion Rates Validation;
partition_object << Get Difference Formula;
partition_object << Get Gen RSquare Test;
partition_object << Get Gen RSquare Training;
partition_object << Get Gen RSquare Validation;
partition_object << Get Measures;
partition_object << Get Microseconds;
partition_object << Get Misclassification Rate Test;
partition_object << Get Misclassification Rate Training;
partition_object << Get Misclassification Rate Validation;
```

```
partition_object << Get MM SAS Data Step;
partition_object << Get MM Tolerant SAS Data Step;
partition_object << Get Prediction Formula;
partition_object << Get RMS Error Test;
partition_object << Get RMS Error Training;
partition_object << Get RMS Error Validation;
partition_object << Get ROC Area Test;
partition_object << Get ROC Area Training;
partition_object << Get ROC Area Validation;
partition_object << Get RSquare Test;
partition_object << Get RSquare Training;
partition_object << Get RSquare Validation;
partition_object << Get SAS Data Step;
partition_object << Get Seconds;
partition_object << Get Tolerant Prediction Formula;
partition_object << Get Tolerant SAS Data Step;
```

Principal Components Script-Only Messages

The following message for a Principal Components object is available only in JSL:

```
prin_comp_object << Save Principal Component Script( number );
```

Process Capability Script-Only Messages

The following messages for a Process Capability object are available only in JSL:

```
process_capability_object << Goal Plot( 1, Capability Lines( number ) );
process_capability_object << Capability Index Plot( 1, Capability Lines(
  number ) );
process_capability_object << Process Performance Plot( 1, Capability
  Boundary( number ) );
process_capability_object << Process Performance Plot( 1, Stability Boundary(
  number ) );
```

Process Screening Script-Only Messages

When you launch the Process Screening platform via JSL, you can use the following arguments that are available only in JSL:

```
Drift Alpha( number )
Make Detailed Shift Data( 0|1 )
Ppk Capability Boundary( number )
Remove( column )
Stability Index Boundary( number )
```

```
Stability Ratio Boundary( number )
```

The following messages for a Process Screening object are available only in JSL:

```
process_screening_object << RowStates;
process_screening_object << Select All;
```

Profiler Script-Only Messages

The following messages for a Profiler object are available only in JSL:

```
profiler_object << Colorize;
profiler_object << Converge Limit( number );
profiler_object << General Extrapolation Control Multiplier( number );
profiler_object << Get Constraints;
profiler_object << Get Desirability;
profiler_object << Get Factor Settings;
profiler_object << Get Factor Settings Script;
profiler_object << Get Main Indices;
profiler_object << Get Simulator;
profiler_object << Get Total Indices;
profiler_object << Max Cycles( number );
profiler_object << MaxIter( number );
profiler_object << Reset;
profiler_object << Term Value( factor( ... ) );
profiler_object << Trips( number )
```

Reliability Forecast Script-Only Messages

The following messages for a Reliability Forecast object are available only in JSL:

```
reliability_forecast_object << Forecast Options( ... );
reliability_forecast_object << Get Results;
```

Reliability Growth Script-Only Messages

The following messages for a Reliability Growth object are available only in JSL:

```
reliability_growth_object << Get Results;
(reliability_growth_object << Mean Time Between Failures Plot) << Options(
  Interval Size( number ) );
(reliability_growth_object << Mean Time Between Failures Plot) << Options(
  Sample MTBF Type( string ) );
```

Response Screening Script-Only Messages

The following message for a Response Screening object is available only in JSL:

```
response_screening_object << Get PValues;
```

Scatterplot 3D Script-Only Messages

The following messages for a Scatterplot 3D object are available only in JSL:

```
scatterplot_3D_object << Frame3D( Add Ellipsoid( ... ) );
scatterplot_3D_object << Frame3D( Add Markers( ... ) );
scatterplot_3D_object << Frame3D( Add Vector( ... ) );
scatterplot_3D_object << Frame3D( Get Axes );
scatterplot_3D_object << Frame3D( Get Box );
scatterplot_3D_object << Frame3D( Get Grab Handles );
scatterplot_3D_object << Frame3D( Get Graph Size );
scatterplot_3D_object << Frame3D( Get Grids );
scatterplot_3D_object << Frame3D( Get Hide Lights Border );
scatterplot_3D_object << Frame3D( get light active( number ) );
scatterplot_3D_object << Frame3D( get light color( number ) );
scatterplot_3D_object << Frame3D( get light position( number ) );
scatterplot_3D_object << Frame3D( Get Line Scale );
scatterplot_3D_object << Frame3D( Get Marker Quality );
scatterplot_3D_object << Frame3D( Get Marker Scale );
scatterplot_3D_object << Frame3D( Get Marker Transparency );
scatterplot_3D_object << Frame3D( Get Rotation );
scatterplot_3D_object << Frame3D( Get Text Scale );
scatterplot_3D_object << Frame3D( Get View Ortho );
scatterplot_3D_object << Frame3D( Get View Perspective );
scatterplot_3D_object << Frame3D( Get View Zoom );
scatterplot_3D_object << Frame3D( Get Wall Color );
scatterplot_3D_object << Frame3D( Get Walls );
scatterplot_3D_object << Frame3D( Get X Axis Color );
scatterplot_3D_object << Frame3D( Get X Axis Label );
scatterplot_3D_object << Frame3D( Get Y Axis Color );
scatterplot_3D_object << Frame3D( Get Y Axis Label );
scatterplot_3D_object << Frame3D( Get Z Axis Color );
scatterplot_3D_object << Frame3D( Get Z Axis Label );
scatterplot_3D_object << Frame3D( Set Axes( 0|1 ) );
scatterplot_3D_object << Frame3D( Set Box( 0|1 ) );
scatterplot_3D_object << Frame3D( Set Graph Size( x, y ) );
scatterplot_3D_object << Frame3D( Set Grids( 0|1 ) );
scatterplot_3D_object << Frame3D( Set Hide Lights Border( 0|1 ) );
scatterplot_3D_object << Frame3D( set light active( ... ) );
scatterplot_3D_object << Frame3D( set light color( ... ) );
```

```
scatterplot_3D_object << Frame3D( set light position( ... ) );
scatterplot_3D_object << Frame3D( Set Line Scale( number ) );
scatterplot_3D_object << Frame3D( Set Marker Quality( number ) );
scatterplot_3D_object << Frame3D( Set Marker Scale( number ) );
scatterplot_3D_object << Frame3D( Set Marker Transparency( number ) );
scatterplot_3D_object << Frame3D( Set Oscillation( ... ) );
scatterplot_3D_object << Frame3D( Set Rotation( ... ) );
scatterplot_3D_object << Frame3D( Set Spin( ... ) );
scatterplot_3D_object << Frame3D( Set Text Scale( number ) );
scatterplot_3D_object << Frame3D( Set View Ortho( 0|1 ) );
scatterplot_3D_object << Frame3D( Set View Perspective( number ) );
scatterplot_3D_object << Frame3D( Set View Zoom( number ) );
scatterplot_3D_object << Frame3D( Set Wall Color( number ) );
scatterplot_3D_object << Frame3D( Set Walls( 0|1 ) );
scatterplot_3D_object << Frame3D( Set X Axis Color( color ) );
scatterplot_3D_object << Frame3D( Set X Axis Label( string ) );
scatterplot_3D_object << Frame3D( Set Y Axis Color( color ) );
scatterplot_3D_object << Frame3D( Set Y Axis Label( string ) );
scatterplot_3D_object << Frame3D( Set Z Axis Color( color ) );
scatterplot_3D_object << Frame3D( Set Z Axis Label( string ) );
scatterplot_3D_object << Frame3D( XAxis( ... ) );
scatterplot_3D_object << Frame3D( YAxis( ... ) );
scatterplot_3D_object << Frame3D( Z Axis( ... ) );
scatterplot_3D_object << Nonpar Density Contour Settings( ... );
```

Structural Equation Models Script-Only Messages

The following messages for a Structural Equation Models object enable you to specify aspects of the model specification, the fitted model, and the path diagram, respectively. These messages are available only in JSL.

```
structural_equation_models_object << Model Specification( ... );
structural_equation_models_object << Fit( ... );
structural_equation_models_object << Path Diagram Properties( ... );
```

The following messages are available within a Model Specification() or Fit() message sent to a Structural Equation Models object.

```
structural_equation_models_object << Fit( Covariances( ... ) );
structural_equation_models_object << Fit( Loadings( ... ) );
structural_equation_models_object << Fit( Means( ... ) );
structural_equation_models_object << Fit( Model Name( string ) );
structural_equation_models_object << Fit( New Latent( string ) );
structural_equation_models_object << Fit( Regressions( ... ) );
structural_equation_models_object << Fit( Variances( ... ) );
```

The following messages are available within a Path Diagram Properties message sent to a Structural Equation Models object.

```
SEM_object << Path Diagram Properties( Constant Border Color( string ) );
SEM_object << Path Diagram Properties( Constant Fill Color( string ) );
SEM_object << Path Diagram Properties( Constant Font( font ) );
SEM_object << Path Diagram Properties( Constant Height( number ) );
SEM_object << Path Diagram Properties( Constant Shape( ... ) );
SEM_object << Path Diagram Properties( Constant Size Option( string ) );
SEM_object << Path Diagram Properties( Constant Text Color( string ) );
SEM_object << Path Diagram Properties( Constant Width( number ) );
SEM_object << Path Diagram Properties( Diagram Size( {x, y} ) );
SEM_object << Path Diagram Properties( Latent Border Color( string ) );
SEM_object << Path Diagram Properties( Latent Fill Color( string ) );
SEM_object << Path Diagram Properties( Latent Font( font ) );
SEM_object << Path Diagram Properties( Latent Height( number ) );
SEM_object << Path Diagram Properties( Latent Shape( ... ) );
SEM_object << Path Diagram Properties( Latent Size Option( string ) );
SEM_object << Path Diagram Properties( Latent Text Color( string ) );
SEM_object << Path Diagram Properties( Latent Width( number ) );
SEM_object << Path Diagram Properties( Lock Diagram( 0|1 ) );
SEM_object << Path Diagram Properties( Manifest Border Color( string ) );
SEM_object << Path Diagram Properties( Manifest Fill Color( string ) );
SEM_object << Path Diagram Properties( Manifest Font( font ) );
SEM_object << Path Diagram Properties( Manifest Height( number ) );
SEM_object << Path Diagram Properties( Manifest Shape( ... ) );
SEM_object << Path Diagram Properties( Manifest Size Option( string ) );
SEM_object << Path Diagram Properties( Manifest Text Color( string ) );
SEM_object << Path Diagram Properties( Manifest Width( number ) );
SEM_object << Path Diagram Properties( Path Styles( ... ) );
SEM_object << Path Diagram Properties( Paths Alpha Level( number ) );
SEM_object << Path Diagram Properties( Paths Color( string ) );
SEM_object << Path Diagram Properties( Paths Font( font ) );
SEM_object << Path Diagram Properties( Paths Opacity( number ) );
SEM_object << Path Diagram Properties( Paths Thickness( number ) );
SEM_object << Path Diagram Properties( Place Nodes( ... ) );
SEM_object << Path Diagram Properties( Rotate Loops( ... ) );
```

Support Vector Machines Script-Only Messages

The following messages for a Support Vector Machines object are available only in JSL:

```
svm_object << Get Measures;
svm_object << Get Prediction Formula;
```

Surface Plot Script-Only Messages

The following messages for a Surface Plot object are available only in JSL:

```
surface_plot_object << Frame3D( Get Axes );
surface_plot_object << Frame3D( Get Box );
surface_plot_object << Frame3D( Get Grab Handles );
surface_plot_object << Frame3D( Get Graph Size );
surface_plot_object << Frame3D( Get Grids );
surface_plot_object << Frame3D( Get Hide Lights Border );
surface_plot_object << Frame3D( get light active );
surface_plot_object << Frame3D( get light color );
surface_plot_object << Frame3D( get light position );
surface_plot_object << Frame3D( Get Line Scale );
surface_plot_object << Frame3D( Get Marker Quality );
surface_plot_object << Frame3D( Get Marker Scale );
surface_plot_object << Frame3D( Get Marker Transparency );
surface_plot_object << Frame3D( Get Rotation );
surface_plot_object << Frame3D( Get Text Scale );
surface_plot_object << Frame3D( Get View Ortho );
surface_plot_object << Frame3D( Get View Persepctive );
surface_plot_object << Frame3D( Get View Zoom );
surface_plot_object << Frame3D( Get Wall Color );
surface_plot_object << Frame3D( Get Walls );
surface_plot_object << Frame3D( Get X Axis Color );
surface_plot_object << Frame3D( Get X Axis Label );
surface_plot_object << Frame3D( Get Y Axis Color );
surface_plot_object << Frame3D( Get Y Axis Label );
surface_plot_object << Frame3D( Get Z Axis Color );
surface_plot_object << Frame3D( Get Z Axis Label );
surface_plot_object << Frame3D( Set Axes( 0|1 ) );
surface_plot_object << Frame3D( Set Box( 0|1 ) );
surface_plot_object << Frame3D( Set Graph Size( x, y ) );
surface_plot_object << Frame3D( Set Grids( 0|1 ) );
surface_plot_object << Frame3D( Set Hide Lights Border( 0|1 ) );
surface_plot_object << Frame3D( set light active( light number, 0|1 ) );
surface_plot_object << Frame3D( set light color( light number, red value,
   green value, blue value ) );
surface_plot_object << Frame3D( set light position( light number, X, Y, Z ) );
surface_plot_object << Frame3D( Set Line Scale( number ) );
surface_plot_object << Frame3D( Set Marker Quality( number ) );
surface_plot_object << Frame3D( Set Marker Scale( number ) );
surface_plot_object << Frame3D( Set Marker Transparency( fraction ) );
surface_plot_object << Frame3D( Set Oscillation( X, Y, Z, duration ) );
surface_plot_object << Frame3D( Set Rotation( X, Y, Z ) );
surface_plot_object << Frame3D( Set Spin( dx, dy, sx, sy ) );
```

```
surface_plot_object << Frame3D( Set Text Scale( number ) );
surface_plot_object << Frame3D( Set View Ortho( 0|1 ) );
surface_plot_object << Frame3D( Set View Perspective( fraction ) );
surface_plot_object << Frame3D( Set View Zoom( number ) );
surface_plot_object << Frame3D( Set Wall Color( number ) );
surface_plot_object << Frame3D( Set Walls( 0|1 ) );
surface_plot_object << Frame3D( Set X Axis Color( color ) );
surface_plot_object << Frame3D( Set X Axis Label( string ) );
surface_plot_object << Frame3D( Set Y Axis Color( color ) );
surface_plot_object << Frame3D( Set Y Axis Label( string ) );
surface_plot_object << Frame3D( Set Z Axis Color( color ) );
surface_plot_object << Frame3D( Set Z Axis Label( string ) );
surface_plot_object << Frame3D( XAxis( Min( number ), Max( number ), Inc(
  number ), Format( ) );
surface_plot_object << Frame3D( YAxis( Min( number ), Max( number ), Inc(
  number ), Format( ) );
surface_plot_object << Frame3D( ZAxis( Min( number ), Max( number ), Inc(
  number ), Format( ) );
surface_plot_object << Clip Sheet( 0|1 );
surface_plot_object << Data points Color( color );
surface_plot_object << Datapoints Choice( Off | Points | Needles | Mesh |
  Surface );
surface_plot_object << Dependent Variables Points( 0|1 );
surface_plot_object << Dependent Variables Response Grid( 0|1 );
surface_plot_object << Equation( column, <column>, <column>, <column> );
surface_plot_object << Frame3D( Scatterplot 3D options );
surface_plot_object << Iso Value( id, value );
surface_plot_object << Mode( "Sheet, points" | "Isosurface" | "Density grid" )
surface_plot_object << Resolution( number );
surface_plot_object << Response( column, ... );
surface_plot_object << Response Column Color Theme( Green to Black to Red |
  Green to White to Red | White to Black | Blue to Gray to Red | Blue to Green
  to Red | Spectral | Jet | White to Blue | White to Green | White to Red );
surface_plot_object << Response Column Fill( Solid | Continuous | Gradients |
  Discrete Gradients );
surface_plot_object << Response Column Gradient Lines( 0|1 );
surface_plot_object << Response Column Gradients( number );
surface_plot_object << Set Z Variable( column );
surface_plot_object << SetVariableAxis( column, <Current Value( number )>,
  <Axis Data( axis options )> );
surface_plot_object << SetXVariable( column );
surface_plot_object << SetYVariable( column );
surface_plot_object << SetZAxis( column, <Current Value( number )>, <Axis
  Data( axis options )> );
surface_plot_object << Surface Alpha( number );
```

```
surface_plot_object << Surface Color Method( "Solid" | formula, <"Solid" |
  formula>, <"Solid" | formula>, <"Solid" | formula> );
surface_plot_object << Surface Gradient Type( Solid | Continuous Gradients |
  Discrete Gradients );
surface_plot_object << Surface Gradients( number );
surface_plot_object << Surface Selector( 0|1 );
surface_plot_object << X Grid( 0|1 );
surface_plot_object << X Resolution( number );
surface_plot_object << X Rotate( degrees );
surface_plot_object << Y Grid( 0|1 );
surface_plot_object << Y Resolution( number );
surface_plot_object << Y Rotate( degrees );
surface_plot_object << Z Grid( 0|1 );
surface_plot_object << Z Resolution( number );
surface_plot_object << Z Rotate( degrees );
```

The following messages for a Surface Plot object act on one of the four surfaces or dependent variables and therefore need a number, <n>, where n=1, 2, 3, or 4.

```
surface_plot_object << Clip Sheet<n>( 0|1 );
surface_plot_object << Contour Color<n>( color );
surface_plot_object << Data points Color<n>( color );
surface_plot_object << Datapoints Choice<n>( Off | Points | Needles | Mesh |
  Surface );
surface_plot_object << Mesh Color<n>( color );
surface_plot_object << Response Column Color Theme<n>( Green to Black to Red |
  Green to White to Red | White to Black | Blue to Gray to Red | Blue to Green
  to Red | Spectral | Jet | White to Blue | White to Green | White to Red );
surface_plot_object << Response Column Fill<n>( Solid | Continuous | Gradients
  | Discrete Gradients );
surface_plot_object << Response Column Gradient Lines<n>( 0|1 );
surface_plot_object << Response Column Gradients<n>( number );
surface_plot_object << Show Contour<n>( Off | Below | Above | On Surface );
surface_plot_object << Show Mesh<n>( Off | X | Y | X and Y );
surface_plot_object << Show Surface<n>( Off | Both Sides | Above | Below Only
  );
surface_plot_object << Surface Alpha<n>( number );
surface_plot_object << Surface Color Range<n>( Data | Axis );
surface_plot_object << Surface Color Theme<n>( Green to Black to Red | Green
  to White to Red | White to Black | Blue to Gray to Red | Blue to Green to
  Red | Spectral | Jet | White to Blue | White to Green | White to Red );
surface_plot_object << Surface Color<n>( color );
surface_plot_object << Surface Gradient Type<n>( Solid | Continuous Gradients
  | Discrete Gradients );
surface_plot_object << Surface Gradients<n>( number );
surface_plot_object << Surface Lighting<n>( None | Low Reflection | Normal );
```

Tabulate Script-Only Messages

The following messages for a Tabulate object are available only in JSL:

```
tabulate_object << Add Table( ... );
MC: I decided to ignore the following that must be specified within a Column
  Table message OR a Row Table message OR within Add Table or Modify Table:
  Aggregate Statistics, Analysis Columns, Columns by Categories, Groupign
  Columns, Statistics,
tabulate_object << Change Item Label( ... );
tabulate_object << Display Column Width( ... );
tabulate_object << Modify Table( ... );
tabulate_object << Order by Count( ... );
tabulate_object << Plot Scale( min, max );
tabulate_object << Remove Column Label( Grouping Columns( column ) );
tabulate_object << Retore Column Label( Grouping Columns( column ) );
tabulate_object << Set Format( ... );
tabulate_object << Test Build( Sample Size( number ) );
tabulate_object << Test Data View;
tabulate_object << Uniform plot scale( 0|1 );
```

Ternary Plot Script-Only Messages

The following message for a Ternary Plot object is available only in JSL:

```
ternary_plot_object << Contour Value( [c1, c2, c3, c4, c5] );
```

Text Explorer Script-Only Messages

The following messages for a Text Explorer object are available only in JSL:

```
text_explorer_object << Add Delimiters( string )
text_explorer_object << Add Phrases( list )
text_explorer_object << Add Phrase Exceptions( list )
text_explorer_object << Add Stem Exceptions( list )
text_explorer_object << Add Stem Overrides( list )
text_explorer_object << Add Stop Words( list )
text_explorer_object << Add Stop Word Exceptions( list )
text_explorer_object << Add Recodes( list )
text_explorer_object << Add Recode Exceptions( list )
text_explorer_object << Cloud Width( pixels )
text_explorer_object << Custom Stemmer( function( {string, dot}, ... ) )
text_explorer_object << Latent Class Analysis( Set Random Seed( number ) );
text_explorer_object << Minimum Frequency for Phrase( n )
text_explorer_object << Phrases Alphabetical( Boolean )
text_explorer_object << Save Regex Column( string, <"From Source"> )
```

```
text_explorer_object << Set Delimiters( string )
text_explorer_object << Set Regex( ... )
text_explorer_object << Terms Alphabetical( Boolean )
```

The following messages for a Sentiment Analysis object in the Text Explorer platform are available only in JSL:

```
sentiment_analysis_object << Add Feature Words
sentiment_analysis_object << Add Intensifier Exception Words
sentiment_analysis_object << Add Intensifier Words
sentiment_analysis_object << Add Negation Exception Words
sentiment_analysis_object << Add Negation Words
sentiment_analysis_object << Add Sentiment Exception Words
sentiment_analysis_object << Add Sentiment Words
```

Time Series Script-Only Messages

The following messages for a Time Series object are available only in JSL:

```
time_series_object << Fit Recommended ETS( ... );
time_series_object << Get Model Specs;
time_series_object << Get Models;
time_series_object << Input Series( ... );
time_series_object << Keep Best Models( "AIC" | "SBC" );
time_series_object << Simple Moving Average Centering Method( "No Centering" |
   "Centered" | "Centered and Double Smoothed for Even Number of Terms" );
```

The following messages for a Simple Moving Average object in the Time Series platform are available only in JSL:

```
sma_object << Get Results;
sma_object << Remove Model( ... );
```

Uplift Script-Only Messages

When you launch the Uplift platform via JSL, you can use the following arguments that are available only in JSL:

```
Multithreading( 0|1 )
Set Random Seed( number )
```

The following messages for an Uplift object are available only in JSL:

```
uplift_object << Get Average Absolute Error Test;
uplift_object << Get Average Absolute Error Training;
uplift_object << Get Average Absolute Error Validation;
uplift_object << Get Average Log Error Test;
uplift_object << Get Average Log Error Training;
```

```
uplift_object << Get Average Log Error Validation;
uplift_object << Get Confusion Matrix Test;
uplift_object << Get Confusion Matrix Training;
uplift_object << Get Confusion Matrix Validation;
uplift_object << Get Confusion Rates Test;
uplift_object << Get Confusion Rates Training;
uplift_object << Get Confusion Rates Validation;
uplift_object << Get Difference Formula;
uplift_object << Get Gen RSquare Test;
uplift_object << Get Gen RSquare Training;
uplift_object << Get Gen RSquare Validation;
uplift_object << Get Measures;
uplift_object << Get Microseconds;
uplift_object << Get Misclassification Rate Test;
uplift_object << Get Misclassification Rate Training;
uplift_object << Get Misclassification Rate Validation;
uplift_object << Get Prediction Formula;
uplift_object << Get RMS Error Test;
uplift_object << Get RMS Error Training;
uplift_object << Get RMS Error Validation;
uplift_object << Get ROC Area Test;
uplift_object << Get ROC Area Training;
uplift_object << Get ROC Area Validation;
uplift_object << Get RSquare Test;
uplift_object << Get RSquare Training;
uplift_object << Get RSquare Validation;
uplift_object << Get Seconds;
uplift_object << Get Tolerant Prediction Formula;
```

Variability Chart Script-Only Messages

The following messages for a Variability Chart object are available only in JSL:

```
variability_chart_object << Analysis Type( string );
variability_chart_object << Conv Limit( number );
variability_chart_object << Max Iter( number );
variability_chart_object << Model( string );
variability_chart_object << Number Function Evals( number );
variability_chart_object << "Reduced Gauge R&R Report"n( 0|1 );
variability_chart_object << Set Random Seed( number );
```

Chapter 11

Display Trees

Create and Interact with Windows

After you learn the basics of scripting, you're ready to manipulate JMP reports and create new JMP windows. This chapter focuses on the following topics:

- navigating JMP display trees to manipulate items in report windows
- building new windows with custom results
- building modal windows
- interacting with the script editor
- converting deprecated `Dialog()` scripts to `New Window()` scripts

Display Tree Compatibility Issues

JMP platforms are generally stable from release to release. However, sometimes the structure of reports changes in major releases. When this happens, old scripts can fail because of the new structure. To avoid the continual updating of old scripts, write scripts so that they avoid dependence on the report structure. Using a string to index a box, followed by a numbered subscript to pick a specific child box, is most reliable across releases. See "Create a Reference by Subscripting". See "Compatibility Notes" for compatibility issues in JMP 16.

Note: This chapter shows how to create and use the most common display boxes. All display boxes are documented in the JMP Scripting Index, which is available in the Help menu.

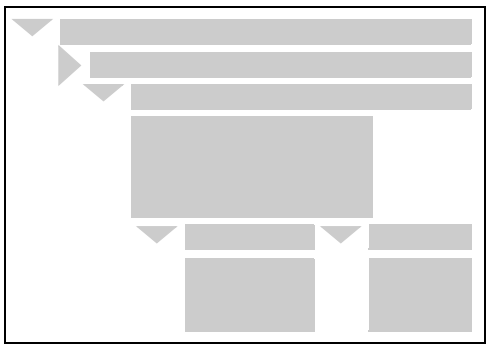
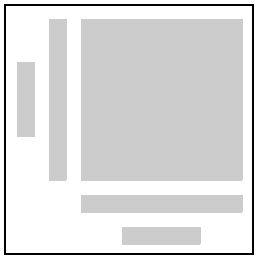
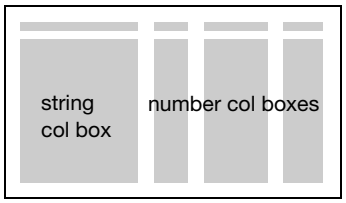
Contents

Navigate JMP Reports 545
Examples of Common Display Boxes 546
View the Display Tree 547
Show Display Box Properties 549
Display Box Object References 550
Send Messages to Display Boxes 555
Example of Creating a Report 562
Construct Custom Windows 565
Example of Creating a Graph Box 565
Extract Values from a Window 566
Constructors for New Windows 568
Interact with a Closing Window 608
Update an Existing Display 610
Set Function and Set Script 614
Get and Set Selected Values of Display Elements That Return Lists 616
Send Messages to Constructed Displays 618
Construct Display Boxes That Contain Platforms 619
Counting the Number of Child Boxes 621
Examples of Creating a Dashboard from Two Reports 622
Example of Creating a Cluster Platform Launch Window 628
Example of Constructing a Custom Platform 631
Example of Specifying Text and Background Colors 634
Modal Windows 635
Construct a Modal Window 635
Construct a Column Dialog 637
Comparison of Column Dialog and New Window 639
Constructors for Column Dialogs 640
Convert Deprecated Dialog to New Window 643
New Window Example 643
Example of Deprecated Dialog 644
Differences Between New Window and the Deprecated Dialog 646
Optional Scripts in New Window 653
Scripting the Script Editor 654
Technical Details 655
Write Tab Box and Tab Page Box Scripts 655

Navigate JMP Reports

As shown in Table 11.1, reports in JMP are built in a hierarchical manner by nesting and concatenating different types of rectangular boxes called *display boxes*. This section shows you how to navigate reports, extract data from them, and customize them.

Table 11.1 Common Display Boxes in JMP Reports

`Outline Box()` creates an outline hierarchy.	
`Picture Box()` combines axes, frames, and labels to make graphs.	
`Table Box()` is a special `H List Box()` whose elements are string and number columns.	

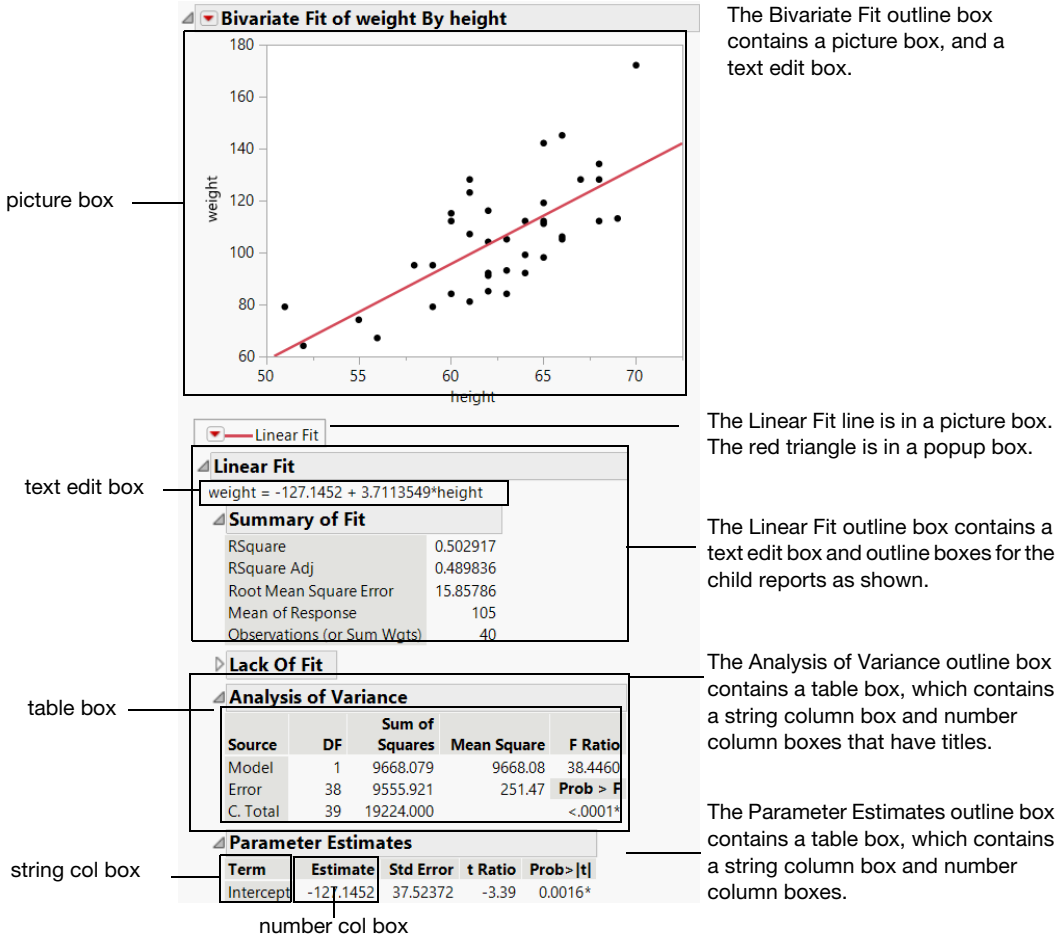
The hierarchical (or parent-child) relationships between display boxes help determine the sizing. The outermost display box is the parent; the inner display boxes are children. When a report is displayed, the parent display box detects the size of the last child box (the leaf) and then detects the size of the child box that precedes the last child box. The size is detected for each box on the way back up the tree. Then the positioning of the boxes takes places from the top down so that the boxes can be shown in an orderly arrangement.

The hierarchical (or parent-child) relationships between display boxes form a tree of display boxes, called the *display tree*. For (parent) boxes that contain other (child) boxes, the parent box's size is determined by the sizes of its child boxes rather than some fixed size, which keeps the whole display resizable. Similarly, positions of boxes are determined by positions of the boxes before them rather than using fixed positions. In tree terminology, the sizing is bottom-up and the positioning is top-down.

Examples of Common Display Boxes

Figure 11.1 shows a Bivariate report that consists of several common display boxes.

Figure 11.1 Display Boxes in a Report



Run the following script to create the report:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( :weight ), X( :height ), Fit Line() );
```

View the Display Tree

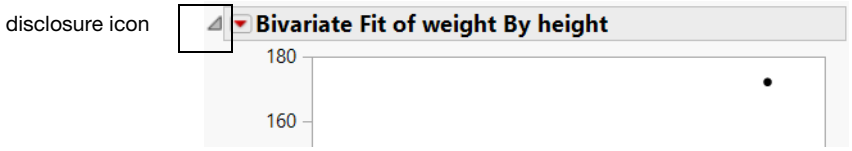
View the report structure to see how display boxes are arranged in the display tree.

1. Run the following script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate(
   Y( :weight ),
   X( :height ),
   Fit Mean( {Line Color( {57, 177, 67} )} )
);
```

2. Right-click the top disclosure icon and select **Edit > Show Tree Structure**.

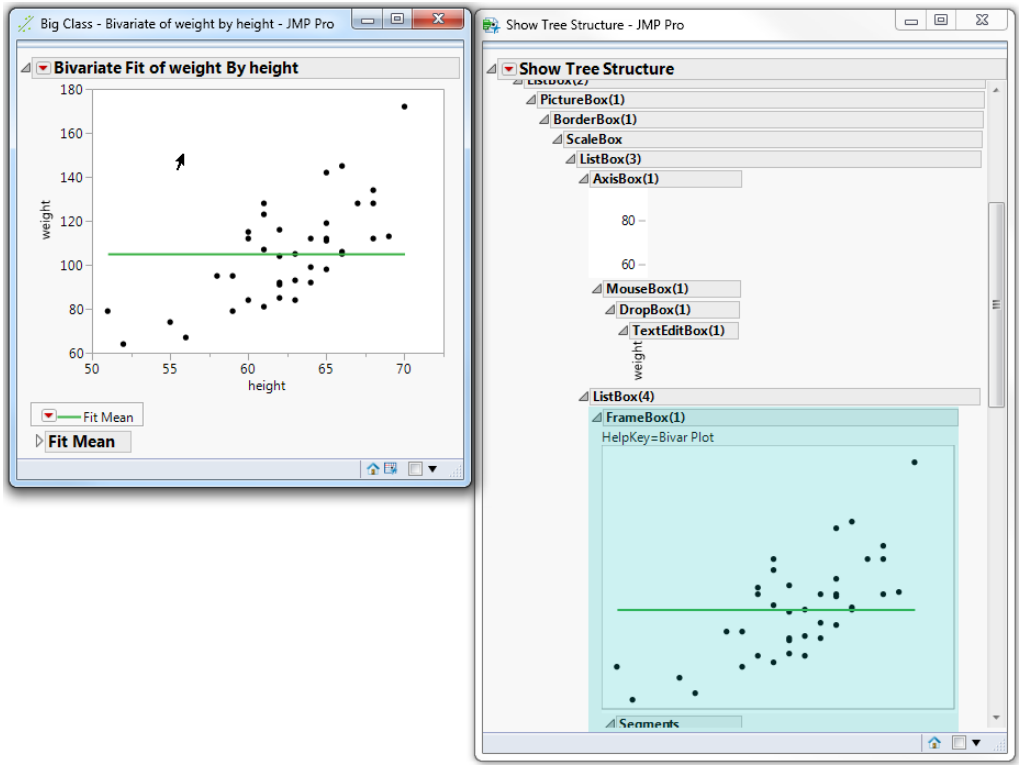
Figure 11.2 Top Disclosure Icon



In the Show Tree Structure window, you see the display box structure of the report as nested outline boxes. Each outline box corresponds to a display box in the original report.

3. Click the Bivariate graph. In Figure 11.3, the frame box that contains the graph is highlighted.

Figure 11.3 Show Tree Structure

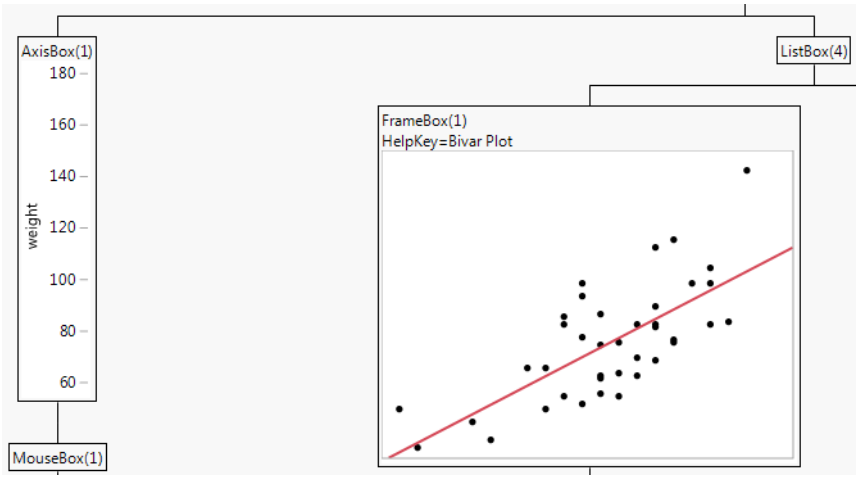


You can also obtain the tree structure through a script. Send the `Show Tree Structure()` message to any report. Or, send the message to a piece of the report (any display box object) to see the tree structure for just that part of the report.

The newer JMP tree structure provides a flat view of the display box tree rather than an outline view. To view the classic structure, press Shift before you right-click the disclosure icon and then select **Edit > Show Tree Structure**. Note the following differences between the newer and classic tree structure views:

- The classic tree structure is not linked to the original report; you cannot highlight specific portions of the tree.
- Siblings appear to the left or right of each other in the classic tree structure. In the newer tree structure, siblings are above or below each other.

Figure 11.4 Partial View of the Classic Tree Structure

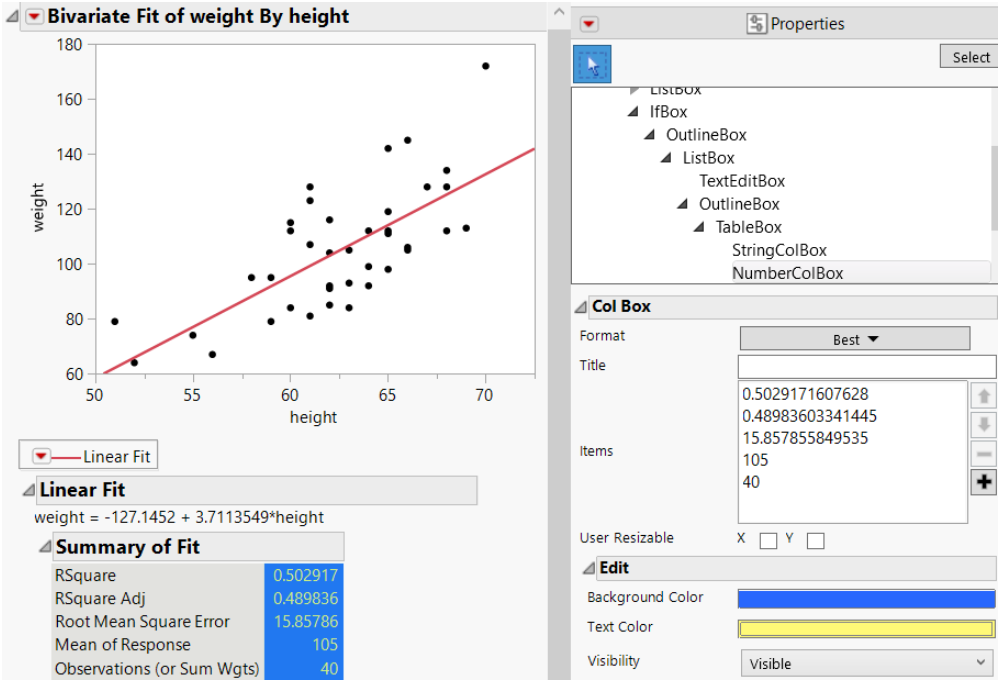


Show Display Box Properties

There is an easy way to format reports before you save them to a script. You can edit the display box properties such as color and position in the Show Tree Structure window. In the report window, select **Edit > Properties** (Windows) or **View > Show Properties** (macOS). Click the node that you want to edit.

In Figure 11.5, the Summary of Fit statistics were selected below the graph. The Background Color and Text Color were then changed in the Properties. These changes are preserved when you save the report as a script or journal the report.

Figure 11.5 Example of Showing Properties for a Plot



Display Box Object References

When you launch a platform, the results are in two places: a visible report, made of display boxes, and an invisible object, called a platform, that accepts commands (or messages).

One of the messages that you can send a platform is the `Report` message. The `Report` message retrieves the display box that holds the report.

The red triangle menu at the top of the report is connected to the platform object and is the way you interactively send messages to the platform.

To fit the mean in the Bivariate platform, assign a reference to the Bivariate platform.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( // biv references the analysis layer
   X( height ),
   Y( weight ),
   Fit Mean,
   Fit Polynomial( 4 )
);
```

To get a reference to the display box that represents the view of the platform, use the `Report` message. For example, add this expression to the preceding script:

```
rbiv = biv << Report; // rbiv references the report layer
```

Now you can manipulate the display boxes in the report layer as described in the following sections.

See What You Can Do with a Display Box

The following example creates a Bivariate report and stores its reference in the `rbiv` report object. It then puts the `rbiv` report object in a frame box and assigns the `fbx` variable.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( // biv references the analysis layer
   Y( :weight ),
   X( :height ),
   Fit Mean( {Line Color( {57, 177, 67} )} )
);
rbiv = biv << Report; // assign rbiv to the report layer
fbx = rbiv[Frame Box( 1 )];
```

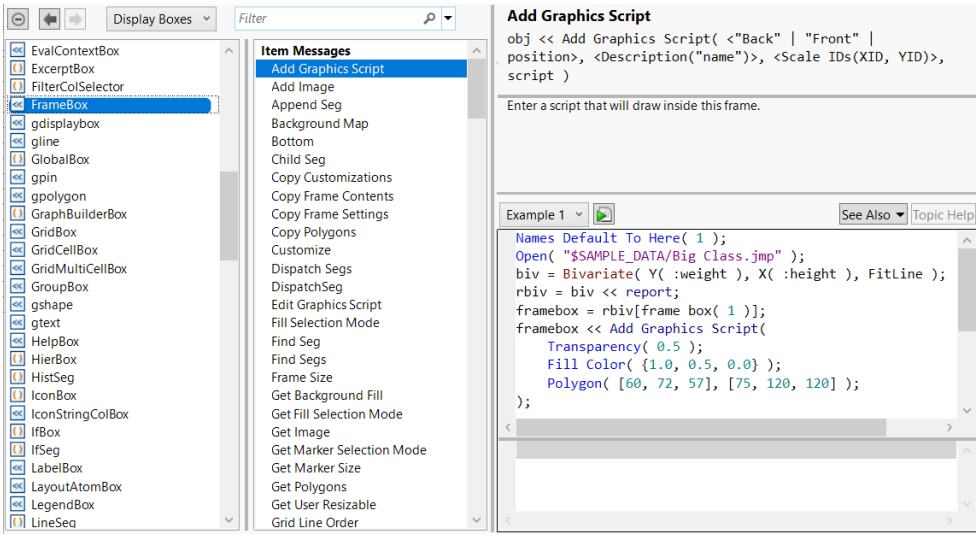
After fbx has been defined, you can send it a message.

```
fbx << Set Background Color( blue ); // colors the frame box in blue
```

The `Set Background Color` message works the same as right-clicking in the graph and selecting **Background Color > Blue**.

To find out which messages you can send to a frame box, select **Help > Scripting Index** and search for “frame box”.

Figure 11.6 Scripting Index Display Boxes List



Running the `Show Properties()` function is an alternative to using the Scripting Index. The messages for the specified display box are printed to the log.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( :weight ), X( :height ) );
Show Properties( biv );
   Show Points [Boolean] [Default On]
   Histogram Borders [Boolean]
   Fit Mean [Action](Fits a flat line at the mean.)
   ...
```

Many messages are the same as items in the pop-up menu for a given object. The next section, "Construct Custom Windows", discusses messages for common display boxes in more detail.

Create a Reference by Subscripting

Use the subscript operator to navigate to a specific display box in the report. The best practice is to refer to the display box with an index number. The following expression identifies the second outline box in the display tree:

```
var = Outline Box[2];
```

Another way to navigate a report is to search for a text string. The following expression identifies the report that has the specified text string:

```
var = report[text];
```

This expression finds the display box in *rpt* that has the title *text*. Note that *text* must be the complete title, not just a substring of the title. The "*text*" argument can also be any expression that evaluates to a text string. This approach is less reliable than referring to the display box with a subscript because the string in a report can easily change.

Typically, you want to assign the identified part of the report to a variable so that you can send messages to it easily. For example, to find the Analysis of Variance outline box in the Bivariate report, use this expression:

```
var = rbiv["Analysis of Variance"];
```

The sections below describe other methods for using a subscript operator to locate display boxes.

Wildcard String

You can use a wildcard character (such as "?") with a substring to match the rest of the outline title. The wildcard character "?" represents places in the search string where you want to match any sequence of any characters. The following script searches for a string that ends with "Mean", finds "Fit Mean", and then opens the Fit Mean outline:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate(
   Y( :weight ),
   X( :height ),
   Fit Mean( {Line Color( {57, 177, 67} )} ),
   Fit Polynomial( 4 )
);
rbiv = biv << Report; // return a reference to the Bivariate report
out1 = rbiv["? Mean"]; // find "Fit Mean" in the Bivariate report
out1 << Close( 0 ); // open the Fit Mean outline
```

Box Type and Wildcard String

Another method for locating display boxes is to include the display box and wildcard search string in the substring operator.

```
var = rbiv[Display Box( "?text" )];
```

This expression finds the outline box that contains the specified text string in the `rbiv` report. The `out1` variable assigns a reference to the outline box.

```
out1 = rbiv[Outline Box( "? Mean" )];
```

Box Type and Index Number

Another method for locating display boxes uses the *boxType* and an index number, *n*.

```
rpt[Display Box ( n )];
```

The expression above finds the *n*th of the specified display box of the type *boxType*. For example, this expression finds the first outline box in the `rbiv` report and assigns it to the `out1` variable:

```
out1 = rbiv[Outline Box( 1 )];
```

This method is less reliable across releases because the number of a display box might change.

Tip: To determine the index number of a display box, view the tree structure as described in "View the Display Tree".

Nested Display Boxes

To reference a display box that appears several layers down in a report, use this format:

```
var = rpt[arg1][arg2][arg3][...];
```

The expression above matches the last argument to a display box that is *contained by* the next to last argument's outline node, which is in turn contained by the argument before the next to last, and so on. In other words, it is a way of digging down into the hierarchy of an outline tree to identify a nested display box.

If you want to identify an item nested several layers down in an outline box, use a subscript with more than one argument of any of the above types. After locating the first item, JMP looks *inside* that item for the next, and so on.

```
rpt[arg1][arg2][arg3] // finds the first item then the next starting after
  that location, and so on
rpt[arg1, arg2, arg3] // different syntax, same result
```

Note that you can string together subscripts that have text or index numbers. The expression below finds the "Parameter Estimate" inside the outline node "Polynomial Fit Degree=4" and selects the third display box in the first table of the outline node.

```
out = rbiv["Polynomial Fit Degree=4"]["Parameter Estimates"][1][3] << Select;
```

"Example of Creating a Report" shows how to extract parameter estimates from the report.

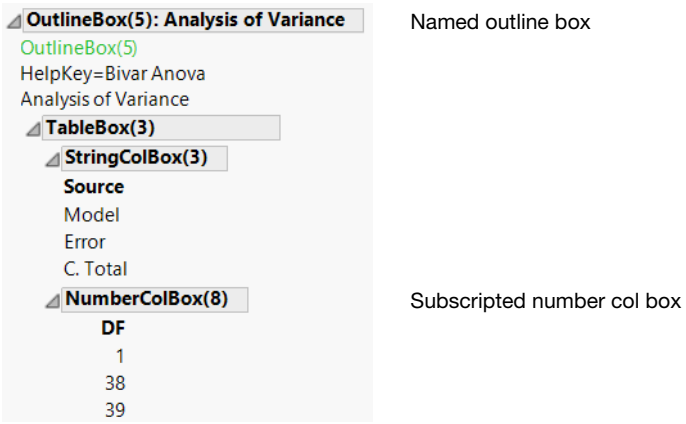
Subscripting Display Boxes for Compatibility

JMP might add or remove display boxes in a report from release to release, causing numbered subscripts to choose a different box. Using a string to index a box, followed by a numbered subscript to pick a specific child box, is more reliable across releases.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
rp = dt << Bivariate(
  Y( :weight ),
  X( :height ),
  Fit Line( {Line Color( {213, 72, 87} )} )
);
```

```
Print(
  "C.Total DF=",

  // string references the parent box
  Report( rp )[Outline Box( "Analysis of Variance" )][
  Number Col Box( 1 )][3] // subscript references the child box
);
```

Figure 11.7 Subscripted Display Box



The numbered subscripts (in the preceding example) are relative to the named box, which makes them less likely to change. However, this also means that the absolute numbers in Show Tree Structure can't be used directly.

Send Messages to Display Boxes

Use the display reference to send messages to the display elements using the `Send` or `<<` operator. For example, if `out2` is a reference to an outline node, you can ask the node to close itself if it is open:

```
out2 << Close(); // close the outline node
```

`Close()` toggles an outline node back and forth between closed and open states, just like selecting items in the red triangle menu. You can include a Boolean argument (1 or 0) for "close or leave closed" or "open or leave open."

```
rbiv["Fit Mean"] << Close;  // toggle open and closed
rbiv["Fit Mean"] << Close( 1 ); // close or leave closed
rbiv["Fit Mean"] << Close( 0 ); // open or leave open
```

Tip: The Report Invalid Display Box Messages option in the General preferences helps you identify invalid display box messages.

- The preference is off by default. When a display box receives an invalid message, JMP ignores those errors and continues executing the script.
- When this preference is on, an error appears in the log.

This option can be useful during script development but can cause unwanted log messages for existing scripts.

The << Operator

You can send messages to display boxes using the send (<<) operator. The operator makes it clear when the script is evaluating child arguments and optional arguments. It also helps make it clearer which argument is an option and which is a script to run inside the graph.

Here is an example of sending a Graph Box a message:

```
win = New Window( "Messages",
   gb = Graph Box(
      Frame Size( 400, 400 ),
      X Scale( 0, 25 ),
      Y Scale( 0, 25 )
   )
);
gb << Background Color( "red" );
```

Nesting and Precedence

When you stack (or *nest*) messages, each message is evaluated left to right.

- The following expression sends `message1` to `box`, then `message2` to `box`, then `message3` to `box`. Note that any of the messages can change `box` before the next message is sent.

  ```
  box << message1 << message2 << message3;
  ```

- The following expression sends `message1` to `box` and gets the result. `Message2` is sent to that result, and `message3` is sent to the result of `message2`.

  ```
  ( (box << message1) << message2) << message3
  ```

- The result of `message3` is assigned to the variable `x`.

  ```
  x = box << message1 << message2 << message3;
  ```

Here is an example of using nested messages:

```
win = New Window( "Messages", gb = Graph Box() );
```

```
// message 1 sets the background color
sz = gb << Background Color( "red" )

// message 2 saves the graph box as a PNG file
<<Save Picture( "$DOCUMENTS/red.png", "png" )

// message 3 sets the graph box background color to white
<<Background Color( "white" )

// message 4 returns the graph box size and prints it to the log
<<Get Size();
```

If you nest several messages, you might want to use parentheses to group the messages to be sure your script does what you want it to.

Customize Reports with the Send To Report and Dispatch Functions

The `Send To Report()` and `Dispatch()` functions provide a way for JMP to record display box modifications inside an analysis script in a self-contained way. For example, arguments within the two functions open and close outline nodes, resize graphics frames, or customize the colors in a graphics frame. You can also specify the arguments in separate JSL statements.

`Send To Report()` contains a list of options that change the display tree.

`Dispatch()` contains four arguments that change the display tree:

- The first argument is a list of outline nodes that need to be traversed to find the desired part of the display tree.
- The second and third arguments work together. The second argument is the name of a display element, and the third argument is the display element's type. These two arguments specify which particular part of the display tree is to be customized.
- The fourth argument is a list of options.

For example, open Big Class.jmp and run the attached Bivariate script. This generates a report with a fitted line. Open the Lack of Fit outline node, and close the Analysis of Variance outline node. Select **Save Script > To Script Window**. The following script appears in the script editor window:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate(
   Y( :weight ),
   X( :height ),
   Fit Line(),
   SendToReport(
      Dispatch(
         {"Linear Fit"},
         "Lack Of Fit",
```

```
            OutlineBox,
            {Close( 0 )} ),
         Dispatch(
            {"Linear Fit"},
            "Analysis of Variance",
            OutlineBox,
            {Close( 1 )}
         )
      )
   );
```

The `Send To Report()` function contains two `Dispatch()` functions that customize the default report.

- The first argument finds an outline node named “Linear Fit”.
- The second and third arguments finds an outline box named “Lack of Fit” below the “Linear Fit” outline.
- The fourth argument is the argument to send to this outline box. In this case, the message is `Close(0)`, in other words, open the node.

Note: If there are several outline nodes with identical names, subscripts are assigned to them. For example, if you have a Bivariate analysis with two quadratic fits (resulting in identical titles), when you dispatch a command to the second fit, the subscript [2] is added to the duplicated title.

The best way to deal with `Send to Report()` and `Dispatch()` is to first create a customized report interactively. Save the report as a script and then examine the script that JMP generates. Remember: the best JSL writer is JMP itself.

Create a Journal

Creating a journal enables you to capture the contents of a JMP window at a fixed moment in time. Suppose that a JMP window contains a graph and selected check boxes. The journal shows the graph and the check boxes that were selected when you created the journal.

The following examples show what you can do with a journal.

Suppose that you start with a Bivariate report.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( weight ), X( height ) );
rbiv = biv << Report;
```

Journal the results:

```
rbiv << Journal Window;
```

Save the journal to a file:

```
rbiv << Save Journal(
   "$DOCUMENTS/test.jrn"
);
```

Note that the above example uses macOS pathname conventions. On Windows, you can use forward slashes (which are appropriate for all platforms) or backslashes (which are appropriate only for Windows).

When a script uses a By variable, the result is a list of references to the analysis results for each By group. In order to journal all By member parts of the report, you need to use parent messages to get to the top of the report.

For example, the following script creates a Bivariate report with a By group and then journals the entire report:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate(
   Y( weight ),
   X( height ),
   By( sex )
);
( ( Report( biv[1] ) << Parent ) << Parent ) <<
Save Journal( "test.jrn" );
```

Journals with Grid Lines

When axes are collapsed, grid lines disappear when the report is journaled, pasted, or saved as a PDF file. Saving the report from JMP as a picture file or a Microsoft PowerPoint presentation retains the grid lines in the graph.

Using Common Messages

Update a Numeric Column

To update a numeric column in a display box table, use the `Set Values` message.

```
win = New Window( "Numeric Column",
   num = Number Col Box( "Values", [9, 10, 11] )
);
num << Set Values( [1, 2, 3] );
```

The matrix argument specifies the new numbers for the table.

Add the Thousands Separator in a Numeric Column

To add the thousands separator in a numeric column, use the `Set Format` message.

```
New Window( "Example",
   ncb = Number Col Box( "Random Numbers",
      {Random Uniform(), Random Uniform() * 10,
      Random Uniform() * 100, Random Uniform() * 1000,
      Random Uniform() * 10000}
   )
);
ncb << Set Format(10,3, "Fixed Dec", "Use thousands separator");
```

Update a String Column

To update a string column in a display box table, use the `Set Values` message.

```
win = New Window( "String Column",
   str = String Col Box( "Values", {"a", "b", "c"} )
);
str << Set Values( {"A", "B", "C"} );
```

The list argument specifies the new strings for the table.

Wrap Text

Generally, JMP automatically wraps text within `Text Box()`. However, the default wrap point can be overridden with a `Set Wrap` (*n*) message. The following script wraps the text at 200 pixels.

```
win = New Window( "Set Wrap",
   tb = Text Box(
      "Hover over a data point for more information."
   )
);
tb << Set Wrap( 200 );
```

Include Bullet Points in Text Boxes

You can add bullet points using the `Bullet Point( 1 )` message.

```
win = New Window( "Bullet List",
   text1 = Text Box( "Hover over a data point for more information." ),
   text2 = Text Box( "Draw a circle around a statistic for more information." )
);
text1 << Bullet Point( 1 );
text2 << Bullet Point( 1 );
```

Sending the `Bullet Point( 1 )` message to a text box places a bullet in front of the text and indents subsequent lines within that text box.

You can also send the `Bullet Point( 1 )` message inline as shown in the following example:

```
text1 = Text Box( "Hover over a data point for more information." ), << Bullet
  Point( 1 ) ),
```

Blink a Selected Display Box

You can use `Select`, `Reshow`, and `Deselect` to blink the selection highlight on a display box. As previously described, you can string together several << clauses to send an object several messages in a row. The results read left to right; for example, here `Select` is done first, then `Reshow`, then `Deselect`, then the other `Reshow`.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate(
  Y( :weight ),
  X( :height ),
  Fit Mean( )
);

// assign the bivariate analysis layer to rbiv
rbiv = biv << Report;

/* find outline box with the specified string in the
rbiv report and assigns it to the out2 variable */
out2 = rbiv[Outline Box( "? Mean" )];
out2 << Close( 0 ); // open the outline box

// find the first string col box in rbiv and
assign the scbx variable */
scbx = rbiv[String Col Box( 1 )];

Wait( .25 ); // wait .25 seconds
For( i = 0, i < 20, i++, // set i to 0, iterate through the loop 20 times

  /* select the string col box and reshow it,
  deselect it and reshow it */
  scbx << Select << Reshow << Deselect << Reshow
);
```

Add Shading or Borders to Table Headings

Headings are shaded and have column borders by default. To turn them off, use the `Set Shade Headings ( 0 )` message and the `Set Heading Column Borders ( 0 )` message. The following example shows the effect of turning shading and borders off:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Summarize(
  meanHt = Mean( Height ),
  minHt = Min( Height ),
```

```
   maxHt = Max( Height )
);
win = New Window( "Summary Results",
   tb = Table Box(
      Number Col Box( "Mean", meanHt ),
      Number Col Box( "Min", minHt ),
      Number Col Box( "Max", maxHt )
   )
);
Wait( 2 );
tb << Set Shade Headings( 0 );
Wait( 1 );
tb << Set Heading Column Borders( 0 );
Wait( 2 );
tb << Set Shade Headings( 1 );
Wait( 1 );
```

Example of Creating a Report

Here is a script to build a report from start to finish.

1. Open a data table.

   ```
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   ```

2. Launch a Bivariate platform and assign it to the platform reference `biv`.

   ```
   /* a reference to the platform */
   biv = dt << Bivariate( Y( weight ), X( height ) );
   ```

3. To find out what you can do with the Bivariate platform, search for Bivariate in the Objects list of the Help > Scripting Index. Here are a few options:

   ```
   biv << Fit Spline( 1000000 ) << Fit Mean;

   // degree 4, orange curve
   biv << Fit Polynomial( 4, {Line Color( RGBColor( 1, 0.5, 0 ) )});
   biv << ( Curve[1] << Line Color( red ) ); // first curve is red
   ```

4. Assign `rbiv` to the report layer:

   ```
   rbiv = biv << Report;
   ```

5. Get the frame box to a comfortable size and scroll up to the top to see the graph.

   ```
   rbiv[Frame Box( 1 )] << Size Window( 500, 700 );
   biv << Scroll Window( {0, 0} );
   ```

6. Get to work on the report. First, you need to create a reference, and then see what you can do with it.

   ```
   Show Properties( rbiv ); // show the properties of the report
   ```

Note: The log lists the messages to use with reports. Using `Show Properties()` is an alternative to the Scripting Index.

```
Close [Boolean]
GetOpen [Action] [Scripting Only]
SetOpen [Boolean] [Scripting Only]
...
```

7. Open the Fit Mean node of the outline:

```
rbiv["Fit Mean"] << Close( 0 );
```

`Close()` is a Boolean function (true or false), and passing 0 indicates that the outline is not closed.

8. Practice selecting some results (submit each line alone to see its result):

```
rbiv["Summary of Fit"] << Select;
rbiv["Parameter Estimates"] << Select;
rbiv["Analysis of Variance"] << Select;
rbiv["Polynomial Fit Degree=4", "Parameter ?", Column Box( "Estimate" )] <<
  Select; // dig down in the outline tree
rbiv << Deselect;
```

9. Change the format of the Estimate column in the 4-degree polynomial Parameter Estimates report.

```
pe = rbiv["Polynomial Fit Degree=4", "Parameter ?"];
ests = pe[Number Col Box( "Estimate" )];
ests << Set Format( 12, 6 );
```

The first argument to `Set Format` sets the column width by the number of characters to display. The second argument sets how many decimal places are shown in the table.

Figure 11.8 Applying Changes to a Report

before

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	-17.60008	84.82175	-0.21	0.8368
height	1.9521428	1.346006	1.45	0.1559
(height-62.55)^2	-0.220179	0.29996	-0.73	0.4678
(height-62.55)^3	0.0703948	0.035633	1.98	0.0561
(height-62.55)^4	0.0073899	0.004231	1.75	0.0895

after
Set Format(12,6)

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	-17.600085	84.82175	-0.21	0.8368
height	1.952143	1.346006	1.45	0.1559
(height-62.55)^2	-0.220179	0.29996	-0.73	0.4678
(height-62.55)^3	0.070395	0.035633	1.98	0.0561
(height-62.55)^4	0.007390	0.004231	1.75	0.0895

Note: You can even get a single number out of the table. For example, get the estimate for the cubic term by adding the following script to the preceding script:

```
terms = rbiv[Outline Box( "Parameter Estimates" )][String Col Box( "Term"
  )]; // find the Term column
ests = rbiv[Outline Box( "Parameter Estimates" )][Number Col Box(
  "Estimate" )]; // find the Estimate column
estimate = .; // in case there is no cubic term
Try(
    For( i = 1, i < 10, i++,

        // get(i) goes off the end of the table handled by Try
        If( Contains( terms << Get( i ), "^3" ),
         estimate = ests << Get( i );
            Break();
        )
    )
);
Show( estimate );
  0.070394822744608
```

10. Use a `For()` loop to count down to the row for the term that you want. The second argument to `For()` is a condition; as long as the condition tests true, looping continues. Here the test is "when the string in the `Terms` column is not `"^3"` and we have not reached the tenth row." As soon as the string *does* match, looping stops and `i`'s value is the number for the matching row. You then use `i` as a subscript to the `Get` message on the Estimates column.
11. You can also get values from boxes as a matrix, which you can then use for further computations or write to a data table. The following example also shows how to make a data table:

```
// get the values in the Sum of Squares column as a matrix
myVector = rbiv[Table Box( 5 )][Number Col Box( "Sum of Squares" )] << Get as
  Matrix;

/* create a new column in Big Class.jmp named "Sum of Squares"
insert the values of myVector in the column */
dt << New Column( "Sum of Squares", Values( myVector ) );

// put the values of the fifth table box into a new data table
rbiv[Table Box( 5 )] << Make Data Table( "ANOVA table" );
```

12. Now adjust the scales on the axes.

```
rbiv[Axis Box( 1 )] << Min( 70 ) << Max( 170 ); // set the Y axis
rbiv[Axis Box( 2 )] << Min( 50 ) << Max( 70 );  // set X axis
```

13. Copy the graph at the top of the report. Note that you need to select the picture box that contains the graph; selecting just the graph would leave its axes behind.

```
rbiv[Picture Box( 1 )] << Copy Picture;
```

Construct Custom Windows

You can use constructor functions to create objects and construct your own displays. For example, you might prompt the user for information, create custom reports, create custom launch windows, or add check box and radio button items to a window.

Start with a `New Window()`, including a title in quotes, and then construct display boxes inside the window. Make a reference to the new window by assigning it a variable so that you can send messages to it. When display objects are created or referred to by JSL, they are freely shared references until they are copied into another display box or until you close the window and they disappear. When you plug a display object into another display tree, JMP makes a copy of it that the new box owns.

The names of all display box constructors end in “Box”. There are similar objects that live inside graphics frames called display segs (segments such as markers and lines). The names of all display seg objects end in “Seg”.

Example of Creating a Graph Box

The following example creates a new window that contains a graph box:

```
win = New Window( "Simple Earth Map",
   gb = Graph Box(
      X Scale( -180, -60 ),
      Y Scale( 20, 80 ), // end of graph box parameters
      <<Background Map( Images( "Simple Earth" ) ) // beginning of script
   )
);
```

The parameters set the x-axis and y-axis scale and the frame size. The `Background Map` message then shows the map. Note that the parameters for `Graph Box()`, `X Scale` and `Y Scale`, come first. After the first comma, the script begins. If you put the script in the middle of the parameters in a `New Window()` function, the script might run but not produce the results you expect. Scripts are also easier for the users to read if the parameters are at the top of the new window.

Extract Values from a Window

After the user selects options in a window, you can extract those selections from the window using the `Return Result` message, `Get` message, or `Get Selected` message.

Extract Values: Method One

To have a `New Window()` script automatically return the results after the user clicks OK, include the `Return Result` message after the `Modal` message.

```
win = New Window( "Set a Value",
   <<Modal,
   <<Return Result,
   Text Box( "Set this value" ),
   variablebox = Number Edit Box( 42 ),
   Button Box( "OK" ),
   Button Box( "Cancel" )
);
Write( win["variablebox"] ); // create a subscript to the variablebox variable
   33 // the user typed "33" in the number edit box
```

Extract Values: Method Two

Include the `Get` message to return 1 for a selected check box and 0 for a deselected check box. To view the selection, add a `Show()` expression at the end of the script.

```
win = New Window( "V List Box",
   <<Modal,
   V List Box(
      kb1 = Check Box( "a" ),
      kb2 = Check Box( "b" ),
      kb3 = Check Box( "c" )
   ),
   Button Box( "OK",
      val1 = kb1 << Get; // get the value of the first check box
      val2 = kb2 << Get;
      val3 = kb3 << Get;
   )
);
Show( val1, val2, val3 ); // return variables after window closes
   val1 = 1; // first and second checkboxes are selected
   val2 = 1;
   val3 = 0; // third checkbox is not selected
```

Extract Values: Method Three

Include the `Get Selected` message to return the selected column names and then insert the columns in a Bivariate plot.

Note: Method two and method three of extracting values are alternate examples of the same method. The method, in both examples, uses a JSL callback from one of the controls in the window. The JSL callback captures a value from a display box in the window into a global variable that will be available after the window closes. In method two, the JSL callback is on a button box. In method three, the JSL callback is on a list box.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
xvar = .;
yvar = .;
win = New Window( "Return Values",
   <<Modal,
   // require the user to select two variables before clicking OK
   <<On Validate(
      Show( xvar, yvar );
         If( Is Missing( xvar ) | Is Missing( yvar ),
            0, // if xvar or yvar are missing, do nothing when OK is clicked
         1
      );
   ),
   Text Box( " Select two numeric columns. " ),
   H List Box(
      Text Box( " X, Factor " ),
      x = Col List Box(
         dt, // data table reference
         all, // display all columns from the data table
         xvar = (x << Get Selected)[1];
         // get the name of the selected column before the window closes
         Show( xvar );
      ),
      Text Box( "Y, Response" ),
      y = Col List Box(
         dt,
         all,
         yvar = (y << Get Selected)[1];
         Show( yvar );
      )
   )
);
xcol = Column( dt, xvar ); // get the columns
ycol = Column( dt, yvar );
```

```
dt << Bivariate( Y( ycol ), X( xcol ) ); // create a Bivariate plot
```

Constructors for New Windows

JSL supports both modal and non-modal windows.

- Modal windows must be closed before other windows can be used: clicking outside the modal window produces an error sound, and almost all script execution stops until the user closes the modal window.
- JMP reports and platform launch windows are examples of non-modal windows. Many non-modal windows can be open at once and you can switch between them freely. The Transparency dialog in a graph is an example of a modal window. After the Transparency dialog is opened, you must close it before continuing.

This section describes common display boxes that you can construct in a window. There are many other display boxes (such as axis boxes) in JMP that cannot be constructed. Display boxes that cannot be constructed are used by platforms in JMP. Even though they can't be constructed in JSL, they still have functions and messages that can be used from JSL.

Tip: In the Scripting Index (**Help** > **Scripting Index**), display boxes that cannot be constructed are marked by . Display boxes that you can construct are marked by .

Notes:

- The `Dialog()` function, which creates modal windows, is deprecated and will not work at all in future versions of JMP. Use either `New Window()` with the `Modal` message or `Column Dialog()` for column selection. See "Construct a Column Dialog" and "Convert Deprecated Dialog to New Window".
- In JMP, the following display boxes support only one direct child box: `Border Box()`, `Center Box()`, `If Box()`, `Mouse Box()`, `Scroll Box()`, and `Sheet Panel Box()`. To make a border box that appears to have multiple children, use a `V List Box()` or `H List Box()` as the only child and put multiple children in the `V List Box()` or `H List Box()`.

Container Display Boxes

Container display boxes enclose child display boxes.

Border Box

`Border Box()` adds space around the *displaybox* argument.

```
Border Box (Left( pix ), Right( pix ), Top( pix ), Bottom( pix ),
  Sides( int ), display box args);
```

`Left`, `Right`, `Top`, and `Bottom` add space around the *`displaybox`* argument. `Sides` draws a border around the box, as described in Table 11.2. Additional effects can also be applied to the borders using `Sides`, as described in Table 11.2. To add both an effect and a border, add the two numbers.

For example, the following example produces a text box with a border at the top and bottom (Draw Border value of 5) colored blue, the default color (Effect value of 16):

```
win = New Window( "Borders",
   bb = Border Box( Sides( 5 ), // top and bottom border
      Text Box( "Hello World!" )
   )
);
bb << Set Color( "red" ); // set the border color to red
```

The `bb` variable is assigned to the border box so that the color can be set on the last line.

Note: Border boxes support only one display box argument. To make a border box that appears to have multiple children, use a `V List Box()` or `H List Box()` as the only child and put multiple children in the `V List Box()` or `H List Box()`.

In the `Sides()` function, add the following values to get the borders that you want. For example, "5" draws a top and a bottom border.

1: top

2: left

4: bottom

8: right

`Border Box()` provides the additional effects described in Table 11.2.

Table 11.2 Border Box Sides() Additional Effects

Add to Number Above	Additional Effect
16	Make the border lines the same color as the operating system's highlight color. Not recommended because this color is different on different computers.
32	Fill the border box with white before drawing the border box's child.

Table 11.2 Border Box Sides() Additional Effects *(Continued)*

Add to Number Above	Additional Effect
64	Fill the border box with the background color before drawing the border box's child. This option makes a difference only if the border box's parent draws something behind the border box first.

Col List Box

`Col List Box()` returns a display box that shows a list of current data table columns.

```
Col List Box( <Data Table( <name> ), <all>|<character>|<numeric>,
   <Width( n )>, <Max Selected( n )>, <NLines( n )>, <Max Items( n )), <Min
   Items( n )>, <On Change( expression )>, <script> );
```

Note: To add a transform column to a custom display box, use `Filter Col Selector()` instead of `Col List Box()`. See "Filter Col Selector".

`All` specifies that all columns in the current data table should be included. `Width` is measured in pixels. `Max Selected` is the maximum number of items that might be selected in the list box. `NLines` is the number of lines displayed in the box. `Max Items` is the maximum number of items displayed in col list box. `Min Items` is the minimum number of items displayed in the col list box. `On Change` is the script that runs when the selection changes.

You can send a `Get Items` message to a col list box to retrieve a list of all columns selected. Here is an example script showing `Get Items` in use. See the Scripting Index in the Help menu for other messages and examples.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Get Items Demonstration",
   H List Box(
      chooseme =

      // Col List Box shows all columns, 100 pixels wide with 6 lines
      Col List Box( "All", width( 100 ), NLines( 6 ) ),

      /* line up box arranges button boxes in 1 column
      with 2 pixels of space around the boxes*/
      Line Up Box( N Col( 1 ), Spacing( 2 ),

         Button Box( "Add Column >>",
```

```
            /* append the selected column to previously
            selected columns */
            listcols << Append( chooseme << Get Selected );

            // get a list of selected columns from the col list box
               Chosen Columns = listcols << Get Items;
                  ),
         Button Box( "<< Remove Column",
            listcols << Remove Selected; // remove the selected column
            Chosen Columns = listcols << Get Items; // get the selected column
         ),
      ),
         // create another col list box with no columns inside
         listcols = Col List Box( width( 100 ), NLines( 6 ) ),
   ),
   Text Box( " " ),

   /* at the bottom of the h list box,
   show which columns Get Items returned */
   stuff = Global Box( Chosen Columns )

);
```

The `All` argument for `Col List Box()` includes all data table columns in the display box. Because all columns are required, you cannot remove selected rows using `Remove Selected` or `Remove All`. Instead, create an empty column list box and append all columns. Then the columns that you select can be removed.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
colNames = dt << Get Column Names; // get a list of column names

New Window( "Example",
// create Col List Box with all the column names from the data table
myBox = Col List Box( width( 200 ), nlines( 10 ) ) );
myBox << Append( colNames );

myBox << Set Selected( 3 ); // select one or all of the entries in myBox
myBox << Remove Selected(); // remove selected entries in myBox
```

Note: To show only the character or numeric columns from the data table in the col list box, send `Set Data Type(numeric|character)` to the col list box.

Col Span Box

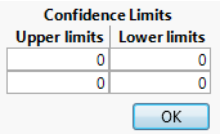
`Col Span Box()` creates spanned columns headers inside a table box. The top column header spans two child column headers.

```
Col Span Box( "title", display box args);
```

In the following script, the Confidence Limits header spans the Upper limits and Lower limits columns below it.

```
win = New Window( "Col Span Box",
   <<Modal,
   Table Box(
      Col Span Box(
         "Confidence Limits",
         neb = Number Col Edit Box( "Upper limits", [0, 0] ),
         Number Col Edit Box( "Lower limits", [0, 0] )
      )
   )
);
```

Figure 11.9 Confidence Limits



Filter Col Selector

`Filter Col Selector()` creates a display box that shows a list of items, for example, columns in a data table. The display box is similar to `Col List Box()` except that `Filter Col Selector()` enables the user to filter columns by name, modeling type, data type, and so on. And `Filter Col Selector()` supports transform columns; `Col List Box()` does not.

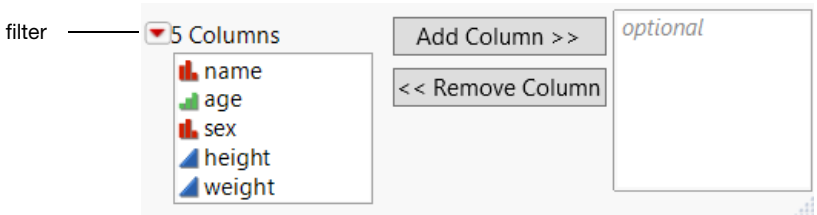
```
Filter Col Selector( <Data Table( name )>, <Width( pixels )>, <Nlines( n )>,
   <script>, <On Change( expr )>, display box args, ... );
```

Here's an example that shows how to use `Get Items` to display a list of selected columns from the filter col selector.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Filter Col Selector Example",
   H List Box(
      chooseme =

      // filter col selector shows all columns, 100 pixels wide with 6 lines
      Filter Col Selector( width( 100 ), NLines( 6 ) ),
```

```
        /* line up box arranges button boxes in 1 column
        with 2 pixels of space around the boxes */
        Line Up Box( N Col( 1 ), Spacing( 2 ),
                Button Box( "Add Column >>",

             /* append the selected column to previously
             selected columns */
             listcols << Append( chooseme << Get Selected );

                // get a list of selected columns from the filter col selector
                Chosen Columns = listcols << Get Items;
                     ),
           Button Box( "<< Remove Column",
              listcols << Remove Selected; // remove the selected column
              Chosen Columns = listcols << Get Items; // get the selected column
           ),
        ),
           // create another col list box with no columns inside
           listcols = Col List Box( width( 100 ), NLines( 6 ) ),
     )
  );
```

Figure 11.10 Filter Col Selector



H List Box and V List Box

`H List Box()` aligns boxes horizontally.

```
H List Box( <Align( center|bottom )>, display box args, ...);
```

`Align` specifies center or bottom alignment of the contents in the display box. The contents are top aligned by default.

The following script aligns the check boxes horizontally:

```
win = New Window( "H List Box",
   <<Modal,
   H List Box(
      kb1 = Check Box( "a" ),
      kb2 = Check Box( "b" ),
      kb3 = Check Box( "c" )
```

```
   ),
);
```

`V List Box()` aligns boxes vertically.

```
V List Box( <Align( center|right )>, display box args, ...);
```

`Align` specifies left, right, or center alignment of the contents in the display box. The contents are left aligned by default.

```
win = New Window( "V List Box",
   Text Box( "Select your favorite fruit." ),
   V List Box(
      Check Box( "apple" ),
      Check Box( "banana" ),
      Check Box( "orange" ),
   ),
);
```

Note: The default maximum width is 180 pixels. To set the width, set the width of the display box that encloses the V List Box.

Suppose that you want to enable the user to add a transform column to custom dialog box. Here is an example of how to create a horizontal list box with an Add and Remove button. The user right-clicks a column to create a transform column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Window( "Example",
     H List Box(
           ll1 = Filter Col Selector( ),
           Button Box( "Add", ll2 << Append( ll1 << Get Selected() ) ),
           ll2 = Col List Box( "Numeric", MaxItems( 1 ), nlines( 1 ) ),
           Button Box( "Remove", ll2 << Remove Selected )
     )
);
```

Note: Transform columns use formulas or calculations to define the column values. Closing the window deletes any transform columns. See *Using JMP*.

If Box

`If Box()` creates a display box whose contents are conditionally displayed.

```
If Box( 0|1, display box args );
```

The Boolean argument 1 displays the display boxes inside the if box. 0 does not display them.

The following script creates a splitter box with three scroll boxes. The second scroll box is enclosed in `If Box()`. Because `If Box()` is set to 1, the second scroll box appears in the window.

```
win = New Window( "If Box",
   H Splitter Box(
      Size( 500, 250 ),
      Scroll Box(),
      If Box( 1, Scroll Box() ), // if box sets the scroll box to show
      Scroll Box()
   )
);
```

When you set an if box to be hidden, then anything within the if box disappears from the report window. In the tree structure view, the hidden display box is no longer numbered. Hide the if box in this example, and the second scroll box is no longer numbered in the tree structure view. `Scroll Box(3)` becomes `Scroll Box(2)`. Change the argument for `If Box()` in this script from 1 to 0 and then send the `win << Show Tree Structure` message to see the structure view.

Note: If boxes support only one display box argument.

H Splitter Box and V Splitter Box

`H Splitter Box()` and `V Splitter Box()` create a display box that arranges the display boxes provided by the arguments in a horizontal or vertical layout (or panel). The splitter enables the user to interactively resize the panel.

```
H Splitter Box( <Size( h,v )>, display box args, <arguments> );
V Splitter Box( <Size( h,v )>, display box args, <arguments> );
```

`Size( h, v )` specifies the size of the splitter box in pixels. Inner display boxes are proportionately sized according to the width and height of the splitter box. For a list of arguments, see the Scripting Index in the JMP Help menu.

The following example puts a Bivariate and Oneway graph inside an H splitter box. Splitter boxes are especially useful when the content is set to auto-stretching as shown in the example.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "H Splitter Box",
   H Splitter Box(
      Size( 900, 600 ),
      biv = dt << Bivariate( Y( :weight ), X( :height ), Fit Line(), Line
   Style( Dotted ) ),
      one = Oneway( Y( :height ), X( :sex ), "Means/Anova"n( 1 ), Mean
   Diamonds( 1 ) )
   )
);
```

```
/* make a reference to the report layers
set the x and y axes on the frame boxes to auto stretch
when the user drags the splitter */
(biv << Report)[FrameBox( 1 )] << Set Stretch( "Window", "Window" );
(one << Report)[FrameBox( 1 )] << Set Stretch( "Window", "Window" );
```

Line Up Box

`Line Up Box()` shows the *display box* arguments in *n* columns. You can specify optional spacing, in pixels, for the space between columns.

```
Line Up Box( NCol( n ), <Spacing( pixels )>, display box args, ... );
```

`Spacing` specifies the number of pixels around the box.

In the following example, the Distribution reports are lined up in three columns:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Line Up Box",
   Line Up Box( NCol( 3 ), // line up the Distributions in 3 columns
      dist = Distribution(
         Continuous Distribution( Column( :height ) ),
         By( :age )
      )
   )
);
```

Outline Box

`Outline Box()` creates a display box to hold other display boxes and show them in an outline view.

```
Outline Box("title", display box args, ... );
```

In the following example, each outline contains a Distribution:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Outline Box",
   obj1 = Outline Box( "Age and Weight",
      Distribution( Column( :age, :weight ) ),

   ),
   obj2 = Outline Box( "Weight and Height",
      Distribution( Column( :weight, :height ) )
   )
);
```

To close an outline by default, send the `Close( 1 )` message to the outline object.

```
obj2 << Close( 1 );
```

The first argument for an outline box is the title. You can specify the title in the script, or the title can be based on another component, such as the value of a slider box. In the following example, as the user adjusts the slider, the title of the outline box is updated:

```
sliderValue = .6;
win = New Window( "Outline Box",
   Panel Box( "Slider Box",
      tb = Outline Box( "Value: " || Char( sliderValue ) ),
      sb = Slider Box(
         0,
         1,
         sliderValue,
         tb << Set Title( "Value: " || Char( sliderValue ) )
      )
   )
);
```

To make your own red triangle menu within an outline box, send the `Set Menu Script` message to the outline box reference. The following example also uses the `Set Submenu` message to create nested menu items.

```
win = New Window( "Red Triangle Menu Items",
ob = Outline Box( "Outline Box" ) );
ob << Set Menu Script(
   {"A", "", "A1", Print( "A1" ), "A2", Print( "A2" ), "B", "",
   "B1", Print( "B1" ), "B2", Print( "B2" ), "B3",
   Print( "B3" ), "C", Print( "C" ), "Selected", Print( "Selected" )}
);
ob << Set Submenu( 1, 2 ); // menu A with 2 items in submenu A1 and A2
ob << Set Submenu( 4, 3 ); // menu B with 3 items in submenu B1, B2, and B3
```

To indicate whether an item is selected, unavailable, or available, include `Set Menu Item State`. The following expression shows a check mark beside the ninth menu item (“Selected”) in the preceding script.

```
ob << Set Menu Item State( 9, 1 );
```

Specify the value -1 to disable the menu item or 0 to make it available.

Panel Box

`Panel Box()` encloses the *`displaybox`* argument in a labeled border.

```
Panel Box( "title", display box args );
```

In the following example, the panel box appears around the edge of the Distribution launch window:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Panel Box",
   H List Box( Panel Box( "My Distribution", Distribution() ) )
);
```

Display Element Boxes

Graph Box

`Graph Box()` constructs a display box that contains a frame box and axes.

```
Graph Box( "title", <X Scale( low, high ), <Y Scale( low, high )>, <Frame
   Size( h, v )>,< <XName( "name" )>, <YName( "name" )>, <SuppressAxes>,
   script );
```

`X Scale()` and `Y Scale()` specify the lower and upper limits of the x and y axes. `Frame Size()` specifies the size of the frame around the graph. `XName` and `YName` specify the name of the x and y axes.

The following example creates a simple graph:

```
win = New Window( "Graph Box",
   Graph Box(
      Frame Size( 300, 300 ),
      Marker( Marker State( 3 ), [11 44 77], [75 25 50] );
      Pen Color( "Blue" );
      Line( [10 30 70], [88 22 44] );
   )
);
```

To remove tick marks, add two arguments to the axes.

```
win = New Window( "Graph Box",
   Graph Box(
      Frame Size( 300, 300 ),
      xaxis( // remove tick marks from the x axis
         Show Major Ticks( false ),
         Show Minor Ticks( false ),
         Show Labels( false )
      ),
      yaxis( // remove tick marks from the y axis
         Show Major Ticks( false ),
         Show Minor Ticks( false ),
         Show Labels( false )
      ),
      Marker( Marker State( 3 ), [11 44 77], [75 25 50] );
```

```
        Pen Color( "Blue" );
        Line( [10 30 70], [88 22 44] );
    )
);
```

JMP provides a robust set of features to make your graphs interactive (such as dragging objects or capturing mouse clicks). "Scripting Graphs" describes graph box options in detail.

H Sheet Box and V Sheet Box

`Sheet Box()` lets you create a grid of plots. `H Sheet Box()` and `V Sheet Box()` contain display boxes and arrange them in columns and rows.

```
H Sheet Box( <<Hold( report ), display box args );
V Sheet Box( <<Hold( report ), display box args );
```

The general approach is to first consider which display boxes you want and in which arrangement. Create either an `H` or `V Sheet Box` and send it a `Hold` message for each graph to specify which element the sheet box should hold. Finally, create interior `H` or `V Sheet Box`es and tell each one which graph the sheet box should hold.

Here is an example of creating a sheet with four plots: a Bivariate plot, a Distribution, a Treemap, and a Bubble Plot.

First, open the data table and create a new window.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
nw = New Window( "Sheet Box Example",
```

Use a `V Sheet Box` to organize the window into two columns.

```
    V Sheet Box(
```

Send the `V Sheet Box()` four `Hold` messages, one for each graph. The order matters.

```
        <<Hold(
            dt << Bivariate( // graph 1
                Y( :weight ),
                X( :height ),
                Fit Line()
            )
        ),
        <<Hold(
            Distribution( // graph 2
                Continuous Distribution(
                    Column( :height ),
                    Horizontal Layout( 1 ),
                    Outlier Box Plot( 0 )
                )
            )
```

```
        ),
        <<Hold( Treemap( Categories( :age ) ) ), // graph 3
        <<Hold(
           Bubble Plot( // graph 4
              X( :height ),
              Y( :weight ),
              Sizes( :age ),
              Coloring( :sex ),
              Circle Size( 6.226 ),
              All Labels( 0 )
           )
        ),
```

Finally, add two `H Sheet Box`es to the `V Sheet Box` and tell each one which graph it should hold. Each H `Sheet Box` holds two side-by-side graphs. They are held by a `V Sheet Box`, so the `H Sheet Box`es are displayed vertically.

```
        H Sheet Box(
           Sheet Part(
              "",
```

`Sheet Part` displays a previously defined graph, held by `Excerpt Box`. The first argument is the number of the graph, determined by the order in which you defined the graphs. The first `H Sheet Box` contains the Bivariate graph on the left and the Distribution on the right. `{Picture Box( 1 )}` designates which picture box from the report to display. Typically, use `1`, the first picture box in the report.

```
              Excerpt Box( 1, {Picture Box( 1 )} ) // excerpt box 1
           ),
           Sheet Part(
              "Distribution of height",
              Excerpt Box( 2, {Picture Box( 1 )} ) // excerpt box 2
           )
        ),
        H Sheet Box(
           Sheet Part(
              "",
              Excerpt Box( 3, {Picture Box( 1 )} ) // excerpt box 3
           ),
           Sheet Part(
              "My Title Here",
              Excerpt Box( 4, {Picture Box( 1 )} ) // excerpt box 4
           )
        )
     )
  );
```

The `Sheet Part()` title is required. If you include an empty string as the title, the sheet part's title is the default report title, for example, "Bivariate Fit of weight By height".

Journal Box

The `Journal Box()` constructs a display box from instructions that are stored in a journal.

```
box = Journal Box("journal text");
```

where *"journal text"* is text that has been extracted from a journal file.

Since journal text has lots of rules about which boxes can be with other boxes, we recommend that you get journal text by highlighting an area of the report, selecting Edit > Journal, and save it as a journal. Open the journal in a text editor, and then paste it into your script as the `Journal Box()` argument.

We highly recommend that you use the `"\[ ... ]\"` quoting mechanism so that you do not have to escape double quotes within the journal text. This excerpt shows how to escape the journal text:

```
win = New Window( "Mosaic Plot",
   Journal Box(
      "\[ // beginning of quoting mechanism
   ...display box arguments...
   ]\"
   ) // end of quoting mechanism
);
```

Another way to get journal text is to send the `Get Journal` message to a display box. For example, run the following script and copy the journal text from the log into the `Journal Box()` function:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( y( :weight ), x( :height ) );
rbiv = biv << Report;
Print( rbiv << Get Journal ); // print the journal to the log
```

Picture Object

JSL has a Picture Object, which stores pictures of JMP output or formulas. You can take a picture of anything in a display box or create a picture of a text formula as it would look in the Formula Editor.

To create picture data, send a `Get Picture` message to a *display box*.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( y( :weight ), x( :height ) );
rbiv = biv << Report;
rbiv << Get Picture;
```

The `Expr As Picture()` function evaluates its argument and creates a picture of the expression, using the same formatting mechanism as the Formula Editor. If you have a literal expression as the argument, remember to enclose it in `Expr()` so that JMP just takes a picture of the result. The expression then is not evaluated.

```
win = New Window( "Formula",
   Expr As Picture( Expr( a + b * c + r / d + Exp( x ) ) )
);
```

After you have the picture, there are two ways of using it:

- Add the picture to a new display tree using a display box constructor.
- Write the picture to a file using `Save Picture()`.

  ```
  picture << Save Picture( "path", type );
  ```

  where *type* can be EMF (Windows), JPEG or JPG, GIF, PDF, or PNG.

On Windows, the Windows Specific preferences determine the resolution (or DPI), or you can run this expression:

```
Pref( Save Image DPI( <number> ) );
```

On macOS, the operating system determines the DPI.

Input Selector Display Boxes

Input selector display boxes enable users to enter information and interact with the display box. The display box script includes a script that runs when the user interacts with the display box. See "Set Function and Set Script" for more information about writing the scripts.

Button Box

`Button Box()` draws a button that contains text.

```
Button Box("text", <script>);
```

The following example creates a new window with an OK button and a Cancel button:

```
win = New Window( "Set the Value",
   <<Modal,
   Text Box( "Set this value" ),
   variablebox = Number Edit Box( 42 ),
   Button Box( "OK" ),
   Button Box( "Cancel" ) );
```

Note that a modal window must have at least one button. If you omit the `Button Box()` from your script, an OK button is automatically included. To omit the OK button, set the visibility to collapse.

```
win = New Window( "No OK Button",
```

```
    <<Modal,
    b = Button Box( "Close Window", b << Close Window ),
    bb = Button Box( "OK", <<Visibility( "collapse" ) )
);
```

You can add a tooltip for a button by sending a `Set Tip` message to it. The following script creates a button named "Submit". A tooltip that contains "Send my information." appears when you hover over the button.

```
Button Box( "Submit", << Set Tip( "Send my information." ) );
```

You can also send the message `Open Next Outline` as a script command, which causes the next outline box to open. If you are sending more than one message to the button box, this must be the first command listed. When the user clicks the next button in this example, the second outline box opens.

```
win = New Window( "Open Next Outline",
   Outline Box( "First",
      ex = Button Box( "Next", ex << Open Next Outline )
   ),
   Outline Box( "Second", Text Box( "Last" ), << Close )
);
```

Using the `Set Menu Items` message, you can create a button that contains a menu:

```
win = New Window( "Set and Get Menu Choice",
   bb = Button Box( "Select a dessert",

   // get menu choices when the button is clicked
      choice = bb << Get Menu Choice;
      Show( choice );
   )
);
bb << Set Menu Items( {"fruit", "no thank you", "-", "chocolate bar", "ice
   cream"} );
```

The `"-"` in the preceding script creates a menu separator. The separator counts as an item in the list. If you select `chocolate bar` from the menu, `4` is returned, not `3`.

To indicate whether an item is selected, unavailable, or available, include `Set Menu Item State`. The following example makes the fifth menu item ("ice cream") in the preceding script unavailable:

```
bb << Set Menu Item State( 5, -1 );
```

Specify 0 to make the menu item available or 1 to show a check mark before the menu item.

Note that, after you click the button, the button box doesn't know where the script is stored because the button is *disconnected* from the script. For example, relative paths don't work when executed from a button box because of this disconnection. An `Open()` script must evaluate the current directory and concatenate it to the relative file path to construct an absolute path.

```
nw = New Window( "Open File Example",
   Button Box( "Open",
      Open(
         Eval( Get Path Variable( "$SAMPLE_DATA" ) ) ||
         "Quality Control/Coating.jmp"
      )
   )
);
```

In the following example, the path is interpreted as an absolute path beginning with the root.

```
nw = New Window( "Example",
Button Box( "Open", Open( "/MyData/Big Class.jmp" ) ) );
```

Note: Line-break characters are ignored in button boxes.

Check Box

`Check Box()` constructs a display box to show one or more check boxes. Any number of the check box items can be selected simultaneously.

```
Check Box ({"item 1", "item 2", ...}, <script>);
```

The following example shows how to get the user's selection by sending the `Get Selected` message to the check box. Each selection is printed to the log.

```
win = New Window( "Check Box",
   cb = Check Box(
      {"apple", "banana", "orange"},
      scb = cb << Get Selected();
      Show( scb );
   )
);
   scb = {"banana", "orange"};
```

The following example shows one way to enable the user to click OK to close the window.

```
win = New Window( "Check Box",
   <<Modal,
   <<On Close(
      flagbox1 = ckbox <<Get( 1 );
      flagbox2 = ckbox <<Get( 2 );
      1; // On Close returns non-zero, so the window closes.
   ),
```

```
   ckbox = Check Box( {"Check Box 1", "Check Box 2"} ),
   ckbox << Set( 1, 1 ),
   ckbox << Set( 2, 0 )
);
```

Combo Box

`Combo Box()` creates a drop-down list.

```
Combo Box( {"item 1", "item 2", ...}, <script> );
```

"item1" and "item2" are strings in the drop-down list.

The `Editable` argument allows the user to enter text in the combo box. The following example creates a new window and an editable combo box with the list items "one", "two", and "three". The script prints the selected item name and index number to the log.

```
win = New Window( "Combo Box",
   cb = Combo Box(
      {"One", "Two", "Three"},
      Editable,
      selection = cb << GetSelected();
      Print( "Selected: " || selection );
      Print( "Index: " || Char( cb << Get() ) );
   )
);
```

Note that closing the window for an editable combo box can produce unexpected results. On Windows, the `Print()` script runs after you select an item from the list or enter the item name. On macOS, the `Print()` script runs after you select an item from the list or enter the item name and then press Return.

For cross-platform compatibility, include `!Is Empty` to test for an existing combo box before running the script.

```
win = New Window( "Combo Box",
   cb = Combo Box(
      {"One", "Two", "Three"},
      Editable,
      If( !Is Empty( cb ),
         selection = cb << GetSelected();
         Print( "Selected: " || selection );
         Print( "Index: " || Char( cb << Get() ) );
      )
   )
);
```

To make an item selectable or unselectable, send the `<<Enable Item` message to `Combo Box()`.

```
cb << Enable Item( 2, 0 ) // makes the second item unselectable
```

Note: If the editable combo box has no empty strings, nothing is selected by default. If the editable combo box has one or more empty strings, the first empty string is selected by default

Global Box

`Global Box` shows the name and current value of a JSL global variable.

```
Global Box( name );
```

The user can assign a new value to the global variable by editing the value directly in the window and pressing Enter or Return to commit the change. Global boxes automatically update the displayed value of a variable when the variable changes.

```
ex = Sqrt( 4 );
win = New Window( "Global", Global Box( ex ) );
```

The preceding script creates a new window named “Global” that displays the following result:

ex=2

The following example shows the value of the myGlobal variable on the graph:

```
myGlobal = 6;
   win = New Window( "Global Box",
      V List Box(
          gr = Graph Box(
              Frame Size( 300, 300 ),
              X Scale( 0, 10 ),
              Y Scale( 0, 10 ),
              Y Function( x, x ),
              Text( {5, 5}, "My global is ", myGlobal )
          ),
          Global Box( myGlobal ) // value of myGlobal
      )
   );
```

Note: When you are using `Global Box()`, remember that every time the variable is changed, JMP remeasures, reshows, and updates the window. Depending on the number of `Global Box()` objects, changes to the variable can affect the refresh rate of the window. JMP recommends avoiding numerous global boxes in deliverable scripts. Global boxes can be replaced with text boxes that the script can manually update when necessary.

List Box

`List Box()` creates a display box that shows a list box of items that can be selected.

```
List Box({"item 1", "item 2", ...}, <Width( n )>, <MaxSelected( n )>,
  <NLines( n )>, <script>);
```

Item names are case sensitive by default. `Width` is measured in pixels. `MaxSelected` is the maximum number of items that might be selected in the list box. `NLines` is the number of lines to display in the box, with a default value of 3.

Add items to a list box by either using `Append` or `Insert`:

```
win = New Window( "test", lb = List Box( {"a", "e"} ) );
lb << Append( {"f", "g"} ); // result is a, e, f, g
lb << Insert( {"b", "c", "a", "d"}, 1 ); // result is a, b, c, d, e, f, g
```

`Append` always adds the list to the end of the list box. `Insert` adds the list after the position specified.

You can have two mutually exclusive list boxes, so that the item you select in one box is deselected when you select an item in the other box. Send the `Clear Selection` message to deselect the item.

```
win = New Window( "Each Box Clears the Other Box",
   window:la = List Box(
      {"broccoli", "pepper", "spinach"},
      <<Set Script( window:lb << Clear Selection )
   ),
   window:lb = List Box(
      {"avocado", "pumpkin", "tomato"},
      <<Set Script( window:la << Clear Selection )
   )
);
```

A list box can also contain an image. In the following example, an image from the user's computer appears in the "first" list item. The JMP nominal icon appears in the "second" list item.

```
win = New Window( "List Box with Image",
    List Box(
        {{"first", "$SAMPLE_IMAGES/pi.gif"}, {"second", "nominal"}},
        width( 200 ),
    )
);
```

To set the width of the list box, use the `width` argument.

```
listA = {"a", "b", "c",
"This is a very, very long name like Mary Ann Johnston."};
New Window( "Example",
```

```
// set the width to the length of the string
  V List Box( listBoxA = List Box( listA, width( 0 ) ) ) );
```

To set the font of the list box, use the `Set Base Font` message.

```
New Window( "Example",
   fontobj = lb = List Box(
      {"First Item", "Second Item", "Third Item"},
      width( 200 ),
      max selected( 2 ),
      nlines( 6 )
   )
);
Wait( 2 );
fontobj << Set Base Font( "Title" );
```

`Set Base Font` also works with `Bar Seg`, `Col List Box()`, `Contour Seg`, `Filter Col Selector`, `Hier Box()`, `Number Edit Box()`, `Outline Box()`, `Poly Seg`, `Text Box()`, `Text Edit Box()`, and `Text Seg`.

Mouse Box

`Mouse Box()` creates a box that can make JSL callbacks for dragging and dropping, marking, or clicking and tracking mouse actions.

```
Mouse Box( display box args, messages );
```

The following example is from the DragDrop.jsl sample script in the Samples/Scripts folder. One list box contains “a”, “b”, and “c”. The other list box contains “d”, “e”, “f”. You can drag a letter from one list box to the other list box.

Note: Mouse boxes support only one display box argument.

```
win = New Window( "Drag and drop between list boxes",
   H List Box(
      MouseBox(
         one = List Box( {"a", "b", "c"} ),

         // if the mouse is over the preceding list box, drop on the other box
         <<setTrack(
            (two << parent) << setDropEnable( 1 );
            (one << parent) << setDropEnable( 0 );
         ),

         // watch the mouse movement (not clicks)
         <<setTrackEnable( 1 ),
```

```
            // when a drop happens...
            <<setDropCommit(
               Function( {this, clickPt, text},
               // find out what is selected in the other box
                  x = two << getSelected;
                  If( Words( text, "\!t" ) == x,
                     two << removeSelected; // and take it...
                     one << append( x ); // and add it to me
                  ,
                     Print( "Drag from the other box please." )
                  );
               )
            )
         ),
         Spacer Box( size( 20, 20 ) ),

         /* same comments as above with reversed logic
         for current box versus other box */
         MouseBox(
            two = List Box( {"d", "e", "f"} ),
            <<setTrack(
               (one << parent) << setDropEnable( 1 );
               (two << parent) << setDropEnable( 0 );
            ),
            <<setTrackEnable( 1 ),
            <<setDropCommit(
               Function( {this, clickPt, text},
                  x = one << getSelected;
                  If( Words( text, "\!t" ) == x,
                     one << removeSelected;
                     two << append( x );
                  ,
                     Print( "Drag from the other box please." )
                  );
               )
            )
         )
      )
   );
```

Number Col Edit Box

`Number Col Edit Box()` creates a display box that shows editable numbers in a column. The numeric entries can be in list or matrix form.

```
Number Col Edit Box("title", numbers);
```

The following script creates number edit boxes in which the user enters values. The `Return Result` message extracts the values.

```
x = y = z = 0;
win = New Window( "Number Col Edit Box",
   <<Modal,
   <<Return Result,
   Table Box(
      neb = Number Col Edit Box( "values", {x, y, z} )
   )
);
   {neb = {2, 4, 6}, Button( 1 )}
```

Number Edit Box

`Number Edit Box()` creates an editable number box that initially contains the value argument. You can also set the width of the box in characters.

```
Number Edit Box( initValue, <width>);
```

The following script shows an editable box that initially contains the number "42". The number that the user types is printed to the log.

```
win = New Window( "Set a Value in Number Edit Box",
   <<Modal,
   <<Return Result,
   Text Box( "Set this value" ),
   variablebox = Number Edit Box( 42 ),
   Button Box( "OK" ),
   Button Box( "Cancel" )
);
Write( win["variablebox"] ); // create a subscript to the variablebox variable
   33 // the user typed "33" in the number edit box
```

The following script shows how to make the initial value a date, and enables the user to choose a different date.

```
New Window( "Date/Time Selector",
      <<Modal,
      neb = Number Edit Box(
            Today(),
            23,
            <<Set Format( Format( "m/d/y h:m:s", 23, 0 ) )
      )
);
```

Popup Box

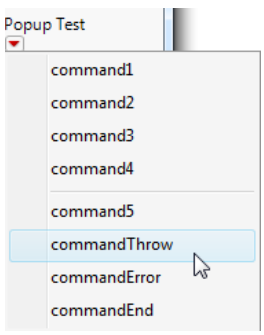
`Popup Box()` creates a red triangle menu that contains menu item commands.

```
Popup Box( {"command1", script1, "command2", script2, ...} );
```

The following example stores a command list in a variable and then uses `Popup Box()` to display the items:

```
List = {
"command1", Print( "command1" ),
"command2", Print( "command2" ),
"command3", Print( "command3" ),
"command4", Print( "command4" ),
"", Empty(), // make a separator line
"command5", Print( "command5" ),
"commandThrow", Throw( "commandThrow1" ),
"commandError", Sqrt( 1, 2, 3 ),
"commandEnd", Print( "commandEnd" )};

win = New Window( "Example",
  Text Box( "Popup Test" ),
  Popup Box( List );
);
```

Figure 11.11 Red Triangle Menu



Note that you can disable and re-enable the menu using the `Enable( Boolean )` message. An argument of 1 turns the menu on, and an argument of 0 turns the menu off. Using the previous example, assign the popup box to a variable and then send messages to it:

```
win = New Window( "Example",
  Text Box( "Popup Test" ),
  mymenu = Popup Box( List );
);

mymenu << Enable( 0 ); // disabled menu
```

You can also add a popup menu to an outline box. See "Outline Box" for an example.

Radio Box

`Radio Box()` creates a list of radio buttons.

```
Radio Box({"item 1", "item 2", ...}, <script>);
```

Only one item in the radio box can be selected at any time.

The following script creates "A", "B", and "C" radio buttons and selects the second box.

```
win = New Window( "Radio Box",
   V List Box(
      rb = Radio Box( {"A", "B", "C"}, <<Set( 2 ) ),
   )
);
```

"New Window Example" also provides an example of creating radio boxes.

Slider Box

`Slider Box()` creates a slider control for picking any value for the variable, within the minimum and maximum range that you specify.

```
Slider Box( min, max, global variable, script, <Set Width( n )>, <Rescale
   Slider( min, max )> );
```

`Set Width` specifies the width of the slider box. `Rescale Slider` specifies the minimum and maximum values.

When the slider is moved, the value given by the current position of the slider is assigned to the global variable. Thus, `Slider Box()` is another way to parameterize a graph.

```
ex = .5;
win = New Window( "Slider Box",
   tb = Text Box( "Value: " || Char( ex ) ),
   sb = Slider Box(
      0,
      1,
      ex,
      tb << Set Text( "Value: " || Char( ex ) )
   )
);
sb << Set Width( 100 ) << Rescale Slider( 0, .8 );
```

The following script creates multiple slider boxes without using a unique global variable to hold the current value for each one:

```
sb1 = Slider Box( 1, 10 );
```

```
// create a slider box with only the min and max value
sb2 = Slider Box( -10, 10 );
sb1 << Set Function( // set the script or the function
   Function( {this},
      {},
      Show( this << Get, sb2 << Get )
   )
);
sb2 << Set Script( Show( sb2 << Get, sb1 << Get ) );
New Window( "Values", sb1, sb2 ); // place the slider boxes in a window
```

As you move the slider, the value is printed to the log.

See "Example of Slider Boxes and Range Slider Boxes" for another example of slider boxes.

Spacer Box

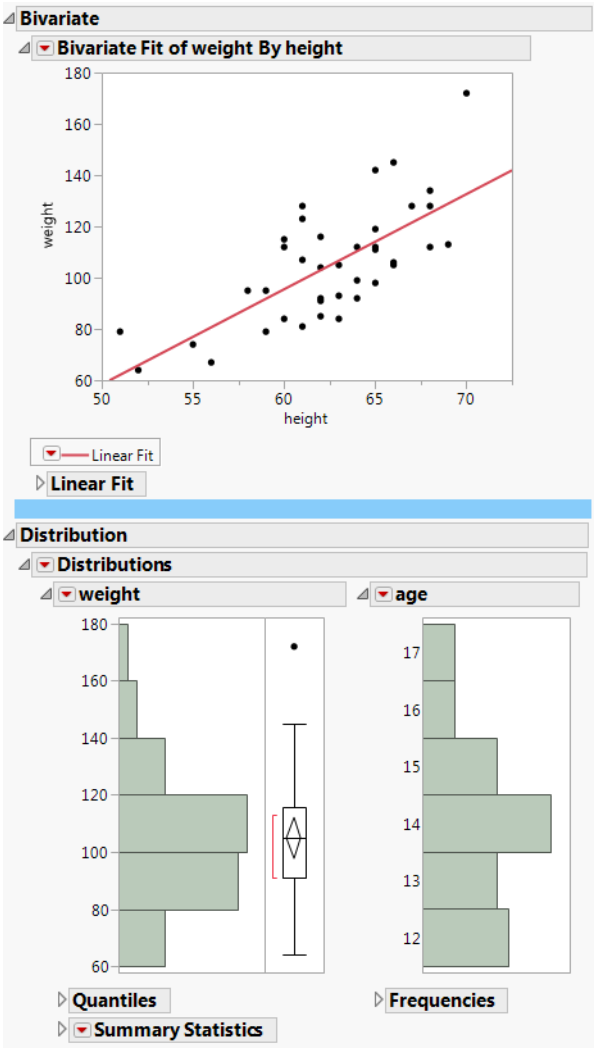
`Spacer Box()` creates space between display boxes.

```
Spacer Box(<size(horizontal_pixels,vertical_pixels)>, <color(color)>)
```

The following script places a 450 pixel wide and 15 pixel tall blue spacer box between the two outline boxes.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Spacer Box",
   V List Box(
      Outline Box( "Bivariate",
         biv = dt << Run Script( "Bivariate" ),
         // Color string consists of percentages of RGB values
         Spacer Box( Size( 450, 15 ), Color( {.53, .80, .98} ), )
      ),
      Outline Box( "Distribution",
         V List Box( dist = dt << Run Script( "Distribution" ), )
      )
   )
);
```

Figure 11.12 Spacer Box Example



String Col Edit Box

`String Col Edit Box()` creates a column named title in a table containing the string items listed. The string boxes are editable.

```
String Col Edit Box( "title", {strings} );
```

The following script creates a column of editable text boxes, preceded by the title “Names”:

```
a = b = c = "";
win = New Window( "String Col Edit Box",
   <<Modal,
   <<Return Result,
```

```
   Table Box(
      seb =
      String Col Edit Box(
         "Names",
         {a, b, c}
         )
      )
);
   {seb = {"apple", "orange", "banana"}, Button( 1 )}
```

Tab Box and Tab Page Box

`Tab Page Box()` organizes the title and contents of the page in one display box. When `Tab Page Box()` is inside `Tab Box()`, a tabbed window with multiple tabbed pages appears.

```
Tab Box( Tab Page Box( Title( "page title 1" ), <options>, contents of
   page 1), Tab Page Box( "page title 2", <options>, contents of page 2),
   ...);
```

The following script creates three tab boxes:

```
win = New Window( "Tab Box",
   Tab Box(
      Tab Page Box(
         Title( "First page" ),
         Text Box( "first line of first page" ),
         Text Box( "second line of first page" )
      ),
      Tab Page Box(
         Title( "Second page" ),
         Text Box( "first line of second page" ),
         Text Box( "second line of second page" )
      ),
      Tab Page Box(
         Title( "Third page" ),
         Text Box( "first line of third page" ),
         Text Box( "second line of third page" )
      )
   )
);
```

Figure 11.13 Tab Boxes

You can specify which tab should be selected by sending the `Set Selected(n)` message, where *n* is the tab number, to the tab box object.

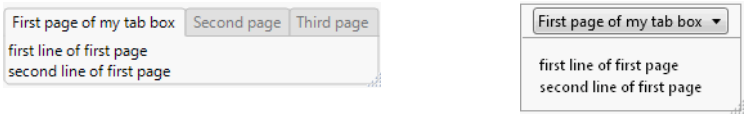
The `Set Style` message lets you select the visual appearance of the tab box. The default value is `tab`. Other options include the following:

- `combo` creates a combo box.
- `outline` creates an outline node.
- `vertical spread` displays the tab title vertically.
- `horizontal spread` displays the tab title horizontally.
- `minimize size` bases the tab style on the width of the tab title. In the following example, the first tab title is fairly long:

```
win = New Window( "Tab Box",
   tb = Tab Box(
      Tab Page Box(
         Title( "First page of my tab box" ),
         Text Box( "first line of first page" ),
         Text Box( "second line of first page" )
      ),
      Tab Page Box(
         Title( "Second page" ),
         Text Box( "first line of second page" ),
         Text Box( "second line of second page" )
      ),
      Tab Page Box(
         Title( "Third page" ),
         Text Box( "first line of third page" ),
         Text Box( "second line of third page" )
      )
   )
);
tb << Set Style( "minimize size" );
```

Minimizing the size of the tab box converts the tab box into a combo box. Figure 11.14 compares the default and minimized tab boxes.

Figure 11.14 Default Tab Box (Left) and Minimized Tab Box (Right)



For `Tab Page Box()` examples, see "Examples of Creating a Dashboard from Two Reports".

Tips:

- Unlike with other display boxes, you must define the title in the `Title()` function.
- The `Set Style` message works for both bare words and quoted words. However, you cannot assign the style to a variable and then pass the variable as the argument.
- You can combine `Tab Box()` with `Splitter Box()` to create docked tabs, which can be dragged to drop zones in a window to create a different layout. See "Examples of Creating a Dashboard from Two Reports".
- To delete a tab, you can either send `Delete(index)` to the tab box, or you can send `Delete Box()` to a tab page box. Note that the entire display box (all tabs) is deleted, not just a single tab.
- Use the `Visibility("Collapsed")` message to hide a tab.
- When tabs in a tab list are not wide enough for the titles, send the `Set Overflow Enabled(1)` message to `Tab Box()`. A triangle symbol is shown to the right of a tab list, and the tab titles are truncated using ellipses.
- When `Tab Page Box()` is created or dragged outside a `Tab Box()`, it is a stand-alone container. The title appears in a shaded box at the top of the page, similar to the title in `Sheet Part()`; the title does not appear in an interactive tab.
- See "Write Tab Box and Tab Page Box Scripts" for more information about the differences between `Tab Box()` and `Tab Page Box()`.
- In Bubble Plot and Graph Builder scripts, set the `Fit to Window` message to `"On"` to resize the window when you resize the graph in `Tab Box()`.
- To close a tab, first make a tab closeable by sending `Closeable(1)` to the tab page box and then send the `Close` message.
- Closeable tabs can be closed with the middle mouse button.

Text Box

`Text Box()` creates a non-editable text box. Text boxes are frequently used as labels of other controls.

```
Text Box( "text" );
```

You can format the text with HTML tags. For example, the following script formats text in bold:

```
win = New Window( "Formatted Text",
  Text Box( "This is <b>bold</b> text.",
  <<Markup) );
```

Make sure you close nested tags correctly and escape the quotes as shown here:

```
"This is <font color=\!"Blue\!" face=\!"Lucida Sans Unicode\!"
  size=\!"24\!">blue</font> text. This is a <background
```

```
color=\!"yellow\!">yellow</background> background. This is <font
color=\!"Blue\!"><background color=\!"yellow\!">blue text on a yellow
background</background></font>."
```

See "Example of Specifying Text and Background Colors" for another example.

Note: Rotated text boxes support `Markup` text except for larger text boxes with multiple lines, multiple formats or justifications applied.

Text Edit Box

`Text Edit Box()` creates an editable text box.

```
Text Edit Box ( "text" );
```

You can add a script to a text edit box by sending it a script message. This is usually most convenient at the time the box is created. Simply add the script message as the last argument.

The following example prints "Changed" in the log each time the text edit box is changed:

```
win = New Window( "Text Edit Box",
   Text Edit Box( "Change Me", <<Script( Print( "Changed" ) ) )
);
```

Assign a reference to the text edit box object to access its contents. The following example prints the value of the text edit box to the log each time the box changes:

```
win = New Window( "Text Edit Box",
   teb = Text Edit Box( "Change Me",
      <<Script( Print( teb << Get Text ) )
   )
);
```

Text edit boxes are empty. However, you can specify placeholder text as a visual cue for the user to enter a value in the box. The placeholder text is only a hint and does not affect the value of the text field.

The following example produces the text edit box shown in Figure 11.15. The `Hint` option returns `"mm/dd/yyyy"` and formats it as light gray text.

```
win = New Window( "Text Edit Box",
   Text Edit Box( "Current Date" ),
   Text Edit Box( "", Hint( "mm/dd/yyyy" ) )
);
```

Figure 11.15 Text Edit Box with Placeholder Text

Tip: To omit the border around the editable box and the white background color, send the `White Box Style( 0 )` message to the text edit box.

If you need to use a text edit box for your user to enter passwords, you can apply a style to the text edit box to replace all characters entered with an asterisk. For example, the following example creates a text edit box whose value is the string "a". When a user types a new string into the text edit box, each character is displayed as an asterisk, and the message "Changed!" is printed to the log.

```
win = New Window( "Text Edit Box",
   teb = Text Edit Box( "a",
      Password Style( 1 ),
      Set Script( Print( "Changed!" ) )
   )
);
```

You can also send a text edit box a message to either use password style or to stop using password style.

```
q << Password Style( 1 ); // set the text edit box to password style
q << Password Style( 0 ); // set the text edit box to standard style
```

Filtering Display Boxes

`Data Filter Source Box()` and `Data Filter Context Box()` work together to filter data in multiple graphs. `Data Filter Source Box()` defines which graph is the "source" of the selection filter. When data are selected in the source graph, graphs in the common `Data Filter Context Box()` are updated.

Data Filter Context Box

`Data Filter Context Box()` creates a display box that enables you to filter data in multiple graphs. The function determines the number of graphs that the filter applies to, for example, all graphs in a report.

```
Data Filter Context Box( display box args );
```

In the following example, the `H List Box()` that contains the graphs is wrapped in `Data Filter Context Box()`. After you run the script, select values in the filter to see both graphs update.

```
dt = Open( "$SAMPLE_DATA/Hollywood Movies.jmp" );
win = New Window( "Shared Local Filter",
   Data Filter Context Box(

   // enclose the h list box that contains distribution and graph builder
      H List Box(
```

```
            dt << Data Filter( Local ), // add the local data filter
            dt << Distribution(
               Weight( :Profitability ),
               Nominal Distribution( Column( :Lead Studio Name ) ),
               Nominal Distribution( Column( :Genre ) ),
               Histograms Only
            ),
            dt << Graph Builder(
               Variables(
                  X( :Genre ),
                  Y( :Domestic Gross ),
                  Y( :Foreign Gross, Position( 1 ) )
               ),
               Show Control Panel( 0 ),
               Elements(
                  Bar(
                     X,
                     Y( 1 ),
                     Y( 2 ),
                     Legend( 2 ),
                     Bar Style( "Side by side" ),
                     Summary Statistic( "Mean" )
                  ),
                  Frequencies( 0 ),
               )
            )
         )
      )
   );
```

Data Filter Source Box

`Data Filter Source Box()` and `Data Filter Context Box()` work together to filter data in multiple graphs. `Data Filter Source Box()` defines which graph is the "source" of the selection filter. When data are selected in the source graph, graphs in the common `Data Filter Context Box()` are updated.

`Data Filter Source Box(` *`display box args`* `);`

The following script lets you select data in Distribution graphs, and only that data is shown in the Graph Builder graph:

```
dt = Open( "$SAMPLE_DATA/Hollywood Movies.jmp" );
win = New Window( "Shared Local Filter",
   Data Filter Context Box(

   // enclose data filter source box and the graph builder platform
      H List Box(
```

```
            Data Filter Source Box(
               dt << Distribution( // use distribution as the source graph
                  Weight( :Profitability ),
                  Nominal Distribution( Column( :Lead Studio Name ) ),
                  Nominal Distribution( Column( :Genre ) ),
                  Histograms Only
               )
            ),
            dt << Graph Builder(
               Variables(
                  X( :Genre ),
                  Y( :Domestic Gross ),
                  Y( :Foreign Gross, Position( 1 ) )
               ),
               Show Control Panel( 0 ),
               Elements(
                  Bar(
                     X,
                     Y( 1 ),
                     Y( 2 ),
                     Legend( 2 ),
                     Bar Style( "Side by side" ),
                     Summary Statistic( "Mean" )
                  ),
                  Frequencies( 0 ),
               )
            )
         )
      )
   );
```

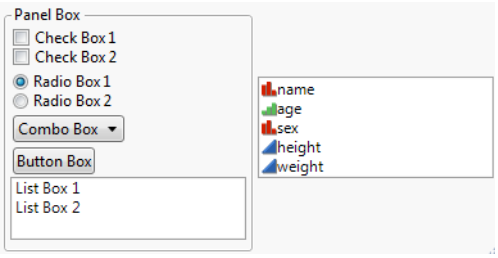
Examples of Combined Display Boxes

The following example generates a sample of many controls illustrated in the previous sections.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
win = New Window( "Window Controls",
   Line Up Box( N Col( 2 ), Spacing( 3 ),
      Panel Box( "Panel Box",
         Check Box( {"Check Box 1", "Check Box 2"} ),
         Radio Box( {"Radio Box 1", "Radio Box 2"} ),
         Combo Box( {"Combo Box"} ),
         Button Box( "Button Box" ),
         List Box( {"List Box 1", "List Box 2"} )
      ),
      Col List Box( "all" )
```

```
    )
);
```

Figure 11.16 Example of Many Interactive Display Elements



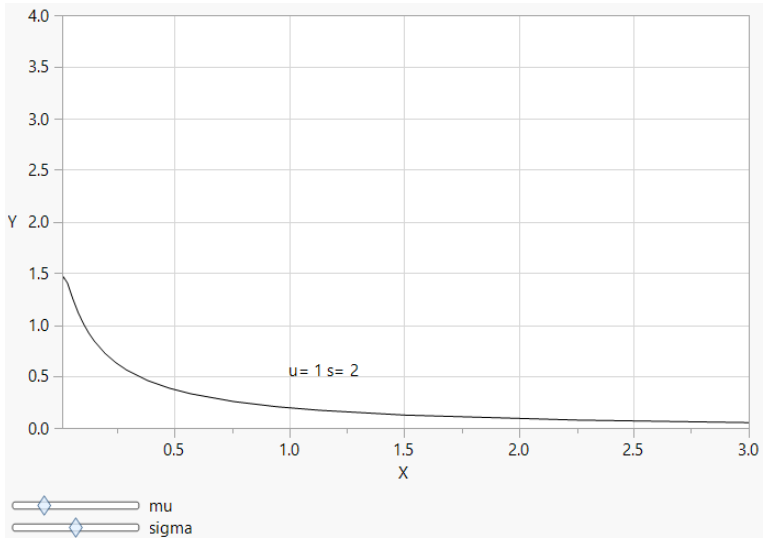
Example of Slider Boxes and Range Slider Boxes

The following example combines a graph with two sliders and a button by gluing the graph box, two horizontal boxes, and a button together in a vertical list box:

```
// Slider LogNormal
lU = 1;
lS = 2;
win = New Window( "LogNormal Density",
   V List Box(
      gr = Graph Box( // define the arguments and script for the graph box
         Frame Size( 500, 300 ),
         X Scale( 0.01, 3 ),
         Y Scale( 0, 4 ),
         XAxis( Show Major Grid ),
         YAxis( Show Major Grid ),
         Y Function(
            Exp( -(Log( x ) - Log( lU )) ^ 2 / (2 * lS ^ 2) ) / (lS * x *
            Sqrt( 2 * Pi() )),
            x
         );

         /* display the specified text at 1 on the x axis
         and .5 on the y axis */
         Text( {1, .5}, "u= ", lU, " s= ", lS );
               ),

      /* define the minimum and maximum range
      insert the value of lU and updates the display as the slider moves */
      H List Box( Slider Box( 0, 4, lU, gr << Reshow ), Text Box( "mu" ) ),
         H List Box(
         Slider Box( 0, 4, lS, gr << Reshow ),
         Text Box( "sigma" )
```

```
        ),
        Button Box( "Reshow", gr << Reshow )
    )
);
Show( gr );
gr << Reshow; // update the display of the Graph Box
```

Figure 11.17 Example of Sliders and Buttons in a Report Window

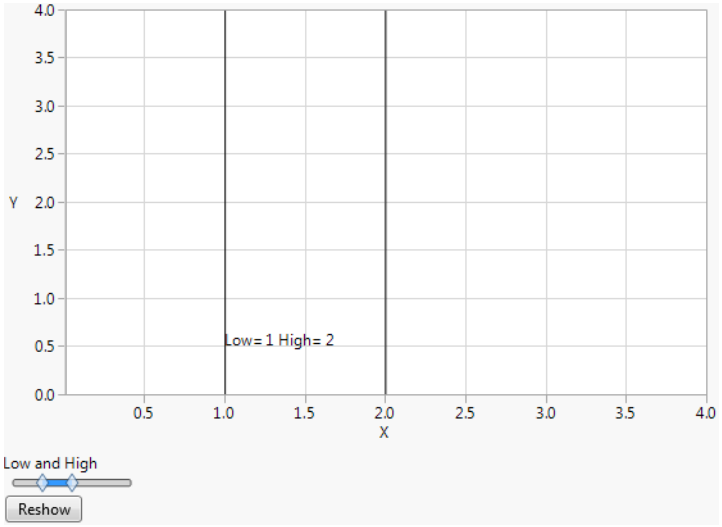


The following script uses a `Range Slider Box()` instead of a `Slider Box()` to perform the same function:

```
lLow = 1;
lHigh = 2;
win = New Window( "Range Slider",
   V List Box(
      gr = Graph Box(
         Frame Size( 500, 300 ),
         X Scale( 0.01, 4 ),
         Y Scale( 0, 4 ),
         XAxis( "Show Major Grid" ),
         YAxis( "Show Major Grid" ),
         X Function( lLow, x );
         X Function( lHigh, x );
         Text( {1, .5}, "Low= ", lLow, " High= ", lHigh );
      ),
      V List Box(
         Text Box( "Low and High" ),
         sb = Range Slider Box( 0, 4, lLow, lHigh, gr << Reshow )
```

```
      )
   ),
   Button Box( "Reshow",
      lLow = 1;
      lHigh = 2;
      gr << Reshow;
      sb << Reshow;
   )
);
Show( gr );
gr << Reshow;
```

Figure 11.18 Example of Using Range Slider Box

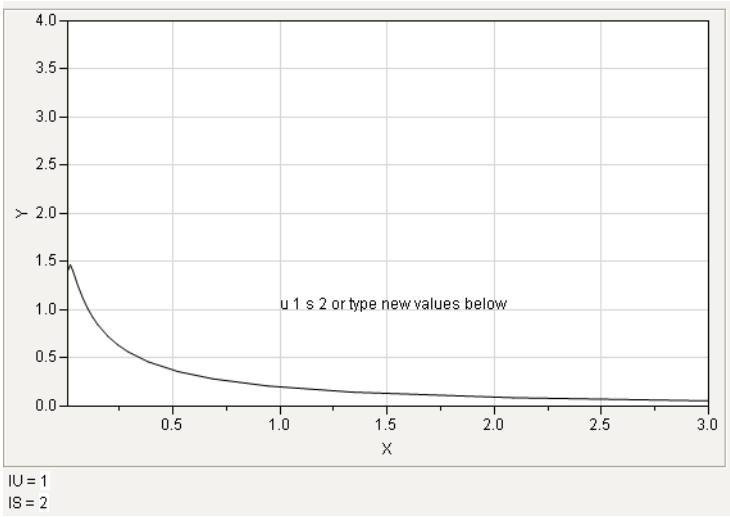


The script below is similar but uses `Global Box()` as an editable text box instead of two `Slider Box()` controls:

```
// Global LogNormal
lU = 1;
lS = 2;
win = New Window( "LogNormal Density",
   V List Box(
      gr = Graph Box( // define the arguments and script for the Graph Box
         Frame Size( 500, 300 ),
         X Scale( 0.01, 3 ),
         Y Scale( 0, 4 ),
         XAxis( "Show Major Grid" ),
         YAxis( "Show Major Grid" ),
         Y Function(
            Exp( -(Log( x ) - Log( lU )) ^ 2 / (2 * lS ^ 2) ) / (lS * x *
```

```
            Sqrt( 2 * Pi() )),
            x
         );

         // display the specified text at 1 on the x and y axes
         Text( {1, 1}, "u= ", lU, " s= ", lS, " or type new values below" );
      ),

      // display the lU variable value in the first H List Box
      H List Box( Global Box( lU ) ),

      // display the lS variable value in the second H List Box
      H List Box( Global Box( lS ) )
   )
);
```

Figure 11.19 Example of Using a Global Box Instead of Sliders



Example of a Calendar and Date Selector

You can create a calendar or date selector using `Calendar Box()` and `Number Edit Box()`.

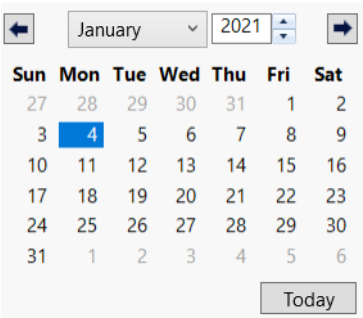
Calendar Box

With `Calendar Box()`, you can select dates from a calendar and input the year. The following example creates a calendar with October 5, 1989 initially selected. You can select any date 60 days before or 60 days after October 5, 1989.

```
win = New Window( "Calendar Box", cal = Calendar Box() );
date = Date MDY( 01, 04, 2021 );
```

```
cal << Date( date );
cal << Show Time( 0 ); // omit the time

/* earliest date that can be selected is 60 days before 01/04/2021
"start" truncates the value so the time is not included */
cal << Min Date( Date Increment(date, "Day", -60, "start" ) );

// latest date that can be selected is 60 days after 01/04/2021
cal << Max Date( Date Increment(date, "Day",  60, "start" ) );
// print the abbreviated date to the log
cal << Set Function( Function( {this}, Print( Abbrev Date(this << Get Date())
  ) ) );
```

Figure 11.20 Creating a Calendar



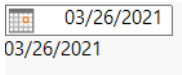
Number Edit Box

With `Number Edit Box()`, you can enter a date in the box or click the calendar button to select the date from a calendar.

The following example shows how to create an interactive calendar using `Number Edit Box()`. The user enters a date in the box or clicks the calendar to select a date from the calendar. The number appears in a text box below the number edit box. “01/04/2021” is initially selected.

```
f = Function( {this}, // callback
   textbox << Set Text( Format( this << Get, "m/d/y" ) ) // get the date
);
win = New Window( "Date Selector",
   numbox = Number Edit Box( 0, 20, << Set Function( f ) ),
   textbox = Text Box( "" ) // specified date
);
numbox << Set Format(
   Format( "m/d/y", 12 ) // date format and number of characters displayed
   );
numbox << Set( Date MDY( 01, 04, 2021 ) ); // default date
```

Figure 11.21 Creating a Date Selector



Note that a callback is assigned to the `f` variable. In the callback, `this` refers to the number edit box, the box that calls the function later. Referring to `this` prevents the need for using the `numbox` variable within the callback. The `Get` message is sent to the `this` argument to get the specified date.

Example of Constructing a Summary Results Report

The following example uses the `Summarize()` function to collect summary statistics on the Height column of Big Class.jmp and then uses display box constructors to show the results in a nicely formatted window.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Summarize(
   a = by( age ),
   c = count,
   sumHt = Sum( Height ),
   meanHt = Mean( Height ),
   minHt = Min( Height ),
   maxHt = Max( Height )
);
win = New Window( "Summary Results",
   Table Box(
      String Col Box( "Age", a ),
      Number Col Box( "Count", c ),
      Number Col Box( "Sum", sumHt ),
      Number Col Box( "Mean", meanHt ),
      Number Col Box( "Min", minHt ),
      Number Col Box( "Max", maxHt )
   )
);
```

This produces the summary results shown in Figure 11.22.

Figure 11.22 Producing a Customized Summary Report

Age	Count	Sum	Mean	Min	Max
12	8	465	58.125	51	66
13	7	422	60.2857	56	65
14	12	770	64.1667	61	69
15	7	452	64.5714	62	67
16	3	193	64.3333	60	68
17	3	200	66.6667	62	70

The following script saves the report as a Microsoft PowerPoint presentation.

```
Show Properties( win );
win << Save Presentation( "$DOCUMENTS/Summary.pptx" );
```

Interact with a Closing Window

A user can close a window by clicking the OK or Cancel button or by clicking the red Close button in the top corner of the window. A message can also be sent to close the window. `On Validate` and `On Close` expressions determine what happens you interact with the window.

- `On Validate` evaluates its expression *when the OK button is clicked*. The expression is not evaluated for any other button click or action. If the expression evaluates as true, the window closes. If the expression evaluates as false, the window is not closed. The script for an OK button runs when you click the button, no matter what the `On Validate` expression does.
- `On Close` evaluates its expression *right before the window closes*. The `On Close` return value is like the `On Validate` return value and can prevent the window from closing. However, the best practice for `On Close` is to always return 1 so that you do not accidentally create a window that cannot be closed.

  If the window closes, `On Close` is evaluated. `On Close` is not evaluated if the `On Validate` expression returns false, because the window is not closed.

Tips:

- `On Close` works for modal or non-modal windows. `On Validate` works only for modal windows.
- Do not use `On Close` for validation because it does not distinguish between OK and Cancel. Use `On Close` to clean up resources (such as invisible data tables) because it always runs when the window closes.
- Do not use `On Validate` to clean up resources. Use `On Close` instead or in addition to `On Validate`, because `On Validate` does not run for the Cancel button or red Close button.

On Close() Example

Suppose that a non-modal window is a report window, and the resource that needs cleaning up is an invisible data table. The `On Close` script closes the data table. This is good because the user might not know that the data table is open and has no easy way to close it.

```
dt = New Table( "Untitled", << Invisible,
    New Column( "x", Set Values( [1, 2, 3, 4] ) ),
    New Column( "y", Set Values( [4, 2, 3, 1] ) ) )
);

dt << Bivariate(
    y( dt:y ), x( dt:x ), <<On Close( Close( dt, "nosave" ) )
```

```
);
```

Before the script is run, no data tables are open. The Untitled invisible table is created and displayed in the Home Window and in the Data Tables list above the script. Then the graph is created. After you close the report window, the invisible table is closed by `On Close`.

On Validate() Example

Suppose that a modal window contains a question that the user must answer before closing the window. The following example shows how to validate the input when the user closes the window.

```
win = New Window( "Validate Example",
   <<Modal,
   <<Return Result,
   <<On Validate(
      If( (!Is Missing( variablebox << Get ) & 40 <= variablebox << Get <= 50),
         ( 1 ), // else the user did not enter the correct answer
         tb << Set Text(
            Match( Random Integer( 1, 3 ),
               1, "Are you sure?",
               2, "Guess again.",
               3, "Please read the question."
            )
         );
         ( 0 );
      )
   ),
   Text Box( "I'm thinking of a number between 40 and 50. What's your guess?"
   ),
   variablebox = Number Edit Box( . ),
   H List Box( Button Box( "OK" ), Button Box( "Cancel" ) ),
   tb = Text Box( "" )
);

New Window( "Modal Window",
   <<Modal,
   Text Box(
      If(

         // user clicked Cancel or the red Close button
         win["Button"] == -1, "Try again when you feel better.",
         win["variablebox"] == 42, "Good job.", // user entered 42
         "Take the test again." // user did not enter 42 and clicked OK
      )
   )
);
```

Update an Existing Display

Sometimes, you do not know how many display boxes will appear in a future report. For example, you might write a generic script that analyzes and reports on one or more variables. You do not know how many display boxes are needed, because the number of variables can change from one run of the script to the next.

The following sections describe how to add and delete display boxes from a report using `Append()`, `Prepend()`, `Delete()`, and `Sib Append()`.

Append

Use the `Append` message to add a display box to the bottom of an existing display. In the script, construct a single, empty box, then append boxes to it for each variable in the analysis. You can also use the `Append` message to modify an existing display with custom information or organization.

The following code example assumes that there is a list of effect names in the variable `effectsList`, and that each one corresponds to a column in a matrix `varprop`. In other words, `effectsList[1]` is the label for `varprop[0,1]`; `effectsList[2]` is the label for `varprop[0,2]`; and so on.

```
varprop = [0 1 2, 3 4 5, 6 7 8];
effectsList = {"one", "two", "three"};
```

First, an empty `Outline Box` containing an `H List Box` is created. The interior empty container is given the name `hb`:

```
win = New Window( "H List Box Example",
   Outline Box( "Variance Proportions", hb = H List Box() )
);
```

Then, a `for` loop steps through the `effectsList` and adds a `Number Col Box` for each element of `effectsList`:

```
For( i = 1, i <= N Items( effectsList ), i++,
   Eval(
      Substitute(
            Expr(
               hb << Append(
                  Number Col Box( effectslist[i], varprop[0, i] )
               )
            ),
         Expr( i ), i
      )
   )
);
```

`Sib Append()` is similar to `Append()` but makes the display box a sibling of the last child display box. For an example that compares `Append()` to `Sib Append()`, see Figure 11.24.

Prepend

The `Prepend` message works just like `Append` but adds the item at the beginning of the display box rather than at the end. If the display box is one of several that do not allow appending, then `Prepend` delegates the command to a child display box that can accept the command. It is fine to apply it to the top of the tree.

For example, the following example creates a Bivariate report with a button box at the top.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( height ), X( weight ), Fit Line );

/* select the first outline box in the biv report layer
send the Prepend message to the first outline box */
(biv << Report)(Outline Box( 1 )) << Prepend(

     // prepend a button box that fits a quadratic curve
     Button Box( "Click here for curve", biv << Fit Polynomial( 2 ) )
);
```

Click the **Click here for curve** button to add a quadratic curve to the graph.

You can accomplish the same thing by appending the button to the top of the tree:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( height ), X( weight ), Fit Line );
biv << Report << Prepend( // send the prepend message to the report

  // prepend the button box to the top of the report
  Button Box( "Click here for curve", biv << Fit Polynomial( 2 ) )
);
```

Delete

The `Delete` message removes the specified display box and all its children from the report. This is useful with the `Append` and `Prepend` messages for building completely dynamic displays. In the example below, a text box is replaced with another text box. In this case, the script could have used `Set Text`, but many display boxes cannot change their content.

```
win = New Window( "X",
   list = V List Box(
      t1 = Text Box( "t1" ),
      t2 = Text Box( "t2" )
   )
);
```

```
t1 << Delete;
list << Append( t1 = Text Box( "t1new" ) );
```

Note: Avoid deleting display boxes from a JMP platform. Send the `Hide( 1 )` message or the `Visibility( "Hidden" )` message to the display box that you want to hide.

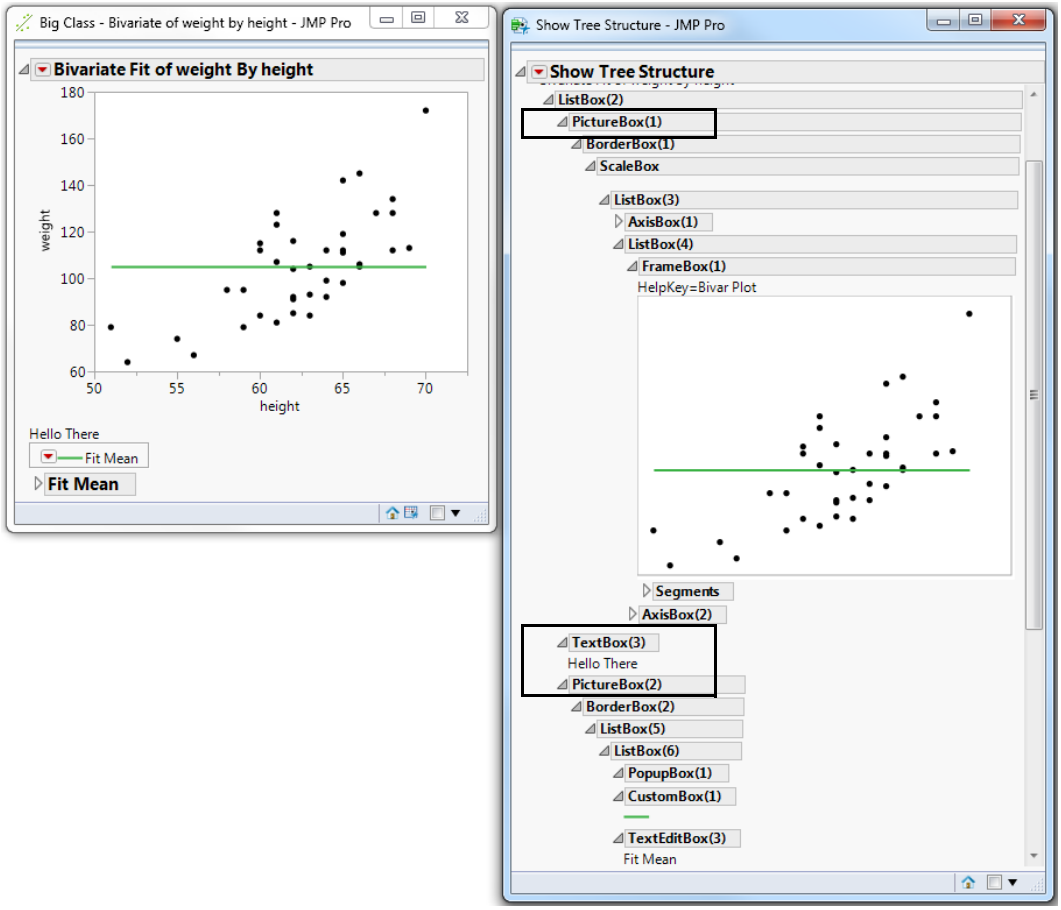
Sib Append

You can use the `Sib Append` message to add a display box immediately after an existing display box. Figure 11.23 shows two picture box trees. `Picture Box( 1 )` holds the Bivariate scatterplot. `Picture Box( 2 )` holds the Fit Mean menu, determined by seeing the green line and the Fit Mean text box.

Suppose you wanted to insert a text box in between these two boxes. You want to append a sibling to `Picture Box( 1 )`, so you send it the `Sib Append` message:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate(
   Y( :weight ),
   X( :height ),
   Fit Mean( {Line Color( {57, 177, 67} )} )
);

/* the report function takes the biv object and returns a report object.
 append the text box immediately after the first picture box */
Report( biv )[Picture Box( 1 )] << Sib Append( Text Box( "Hello There" ) );
```

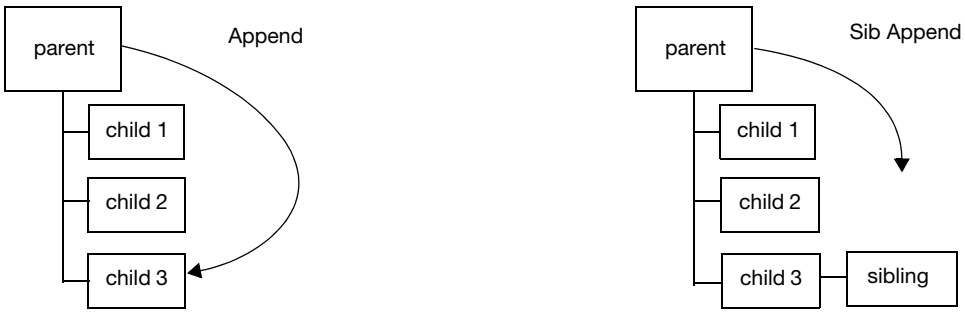
Figure 11.23 Appending a Sibling Text Box



Note: In Figure 11.23, the appended sibling text box appears below the first picture box in the Show Tree Structure View window. In the classic Show Tree Structure View window, siblings appeared next to each other in the display tree. See "View the Display Tree".

Figure 11.24 shows the difference between and `Append()` and `Sib Append()`. `Append()` makes the display box a *child* of the parent display box. `Sib Append()` makes the display box a *sibling* of the last display box.

Figure 11.24 Append (Left) and Sib Append (Right)



Note: If the parent display box is not a `V List Box()`, `Sib Append()` wraps that display box in `V List Box()`.

Set Function and Set Script

You can use the `Set Script` message to have a display box control (for example, a button box or combo box) run a script when it is clicked with the mouse.

```
win = New Window( "Set Script Example",
   ex = Button Box( "Press me" )
);
ex << Set Script( Print( "Pressed" ) );
```

The preceding script prints the following text to the log when the button box is clicked:

```
"Pressed"
```

Alternatively, you can use the `Set Function` message to have a display box control run a specific function where the first argument is the specific display box. `Set Function` enables you to build more object-oriented scripts and to create a larger system.

```
win = New Window( "Set Function Example",
   Button Box( "Press me",

      // functions specified with Set Function get 'this' display box
      << Set Function(
         Function(
            {this},
            this << Set Button Name( "Thanks" )
         )
      )
   )
);
```

The preceding script creates a button box with the name “Press me”. When the button is clicked, the function called by `Set Function` changes the name to “Thanks”.

If you have multiple buttons that can be serviced by the same script, `Set Function` is much simpler than `Set Script` because it tells you which button was pressed. In this example, both check boxes use the script that begins on the first line.

```
f = Function( {this, idx}, // idx just changed
   Write(

   /* <<Get Items returns a list of all items; indexing that list with idx
   is the name of the item that is changing */
      "\!n changing item=" || (this << Get Items())[idx] || " to " || Char(

   // <<Get returns the new value of that item
   this << Get( idx ) ) ||
      "\!n new selection=" || Char( this << Get Selected ) || "\!n"
   )// <<Get Selected returns a list of the currently checked (selected) items
);
New Window( "Example",
   H List Box( // 'f' is the 'named' function, used twice
      V List Box( Text Box( "Column 1" ), Check Box( {"a", "b"}, <<Set
   Function( f ) ) ),
      Spacer Box( size( 50, 50 ) ),
      V List Box( Text Box( "Column 2" ), Check Box( {"c", "d"}, <<Set
   Function( f ) ) )
   )
);
```

In the following example of `Set Function`, the same function is used to create the buttons.

```
New Window( "Cash Box",
   V List Box(
      H Center Box( TB = Text Box() ), // TB will show total of coins
      LB = Lineup Box( N Col( 3 ) ), // LB will hold buttons, added below
      H Center Box( // CLEAR button will reset the total
         Button Box( "CLEAR",
            total = 0;
            TB << Set Text( Char( total ) );
         )
      )
   )
);
coins = {1, 5, 10, 25, 50, 100}; // coins is a list of the button labels
total = 0;
```

```
/* loop creates the buttons and shows that the same function
is used for each */
For( iButton = 1, iButton <= N Items( coins ), iButton++,
   LB << Append( Button Box( Char( coins[iButton] ), Set Function(
   buttonFunction ) ) )
);
buttonFunction = Function( {this},
   total = total + Num( this << Get Button Name );
   TB << Set Text( Char( total ) );
);
```

The function called for any button, except CLEAR, uses the button's name to determine what to do. Rather than using the button's name, you might want to use a sibling display box by `this<<sib` or perhaps `(this<<parent)<<child` to find the first sibling of this button.

Notes:

- The `Set Script` and `Set Function` messages work for button boxes, calendar boxes, check boxes, combo boxes, list boxes, popup boxes, radio boxes, range slider boxes, slider boxes, and spin boxes.
- You cannot use both `Set Script` and `Set Function` at the same time. Use `Set Function` if you need to reference a specific display box object.

Get and Set Selected Values of Display Elements That Return Lists

You can use the `Set Selected(Item Number, <State>, <Run Script(0|1)>)` message to pre-select an item. You can use this message separately to a saved display box reference, or you can specify it inline as a list box << message.

To retrieve the selected value, use the `Get Selected` message, which returns the value of the selected item. The `Get Selected Indices` message returns the index number of the selected item.

```
antennaList = {"Dish","Helical","Polarizing","Radiant Array"};

win = New Window( "Test List", // method 1: display box reference
   listObj = List Box(
      antennaList,
      Print(
         "iList",
         listObj << Get Selected,
         listObj << Get Selected Indices
      )
   )
);
listObj << Set Selected( 2, 1 );
```

```
win = New Window( "Test List", // method 2: inline
   listObj = List Box(
      antennaList,
      <<Set Selected( 2, 1 ),
      Print(
         "iList",
         listObj << Get Selected,
         listObj << Get Selected Indices
      )
   )
);
```

Both of these scripts print the following text to the log:

```
"iList"
{"Helical"}
{2}
```

In the preceding examples, the `Print` expression is executed when the `Set Selected` message is completed. To prevent the script from running, include `Run Script(0)` as the last argument. `Run Script( 0|1 )` controls whether a display box on-change script runs after a `Set` or `Set Selected` message.

```
antennaList = {"Dish","Helical","Polarizing","Radiant Array"};
win = New Window( "Test List",
   listObj = List Box(
      antennaList,
      Print(
         "iList",
         listObj << Get Selected,
         listObj << Get Selected Indices
      )
   )
);
listObj << Set Selected( 2, 1, Run Script( 0 ) );
```

With `Run Script( 1 )`, the script is executed when the `Set` message is completed, even if the value is unchanged. (The script does not run again if the user selects the same value.) With `Run Script( 0 )`, the script does not run.

On most interactive display boxes, the script does not run on most interactive display boxes if you leave out `Run Script()`. However, on `List Box()`, the script runs by default, which is consistent with previous behavior.

Send Messages to Constructed Displays

When you assign a constructed display to a name, that name becomes a reference to the window, which in turn owns the display boxes inside it. Using subscripts, you can then send messages to the display boxes inside the window.

For example, the following script creates an interactive sine wave graph. The script automates the interaction by sending messages to the frame box inside the window. Note the frame box assignment to *tf* in the middle of the script).

```
amplitude = 1;
freq = 1;
phase = 0;
win = New Window( "Wiggle Wave",
   Graph Box(
      Frame Size( 500, 300 ),
      X Scale( -5, 5 ),
      Y Scale( -5, 5 ),
      Y Function( amplitude * Sine( x / freq + phase ), x );
      Handle(

         /* current values of phase and amplitude position
         the square handle.
         script begins after the last comma after amplitude */
         phase,
         amplitude,

         /* x and y have been set to the handle's position
         but the handle doesn't move unless the values specified
         in the first two arguments are updated */
         phase = x;
         amplitude = y;
      );

      // Handle works similarly except that the y value is .5
      Handle( freq, .5, freq = x );

      Text( // display a text string in the graph
         {3, 4},
         "amplitude: ",
         Round( amplitude, 4 ), // display the current value
         {3, 3.5},
         "frequency: ",
         Round( freq, 4 ), // display the current value
         {3, 3},
         "phase: ",
         Round( phase, 4 ) // display the current value
```

```
      );
   )
);
tf = win[Frame Box( 1 )]; // get the frame box (the graph)
For( amplitude = -4, amplitude < 4, amplitude += .1,// animate the amplitude
   tf << Reshow // force the graph to update
);
amplitude = 1; // use for loops for more complex movement
freq = 1;
phase = 0;
For( i = 0, i < 1000, i++,
   amplitude += (Random Uniform() - .5);
   amplitude = If(
      amplitude > 4, 4,
      amplitude < -4, -4,
      amplitude
   );
   freq += (Random Uniform() - .5) / 20;
   phase += (Random Uniform() - .5) / 10;
   tf << Reshow;
   Wait( .05 );
);
```

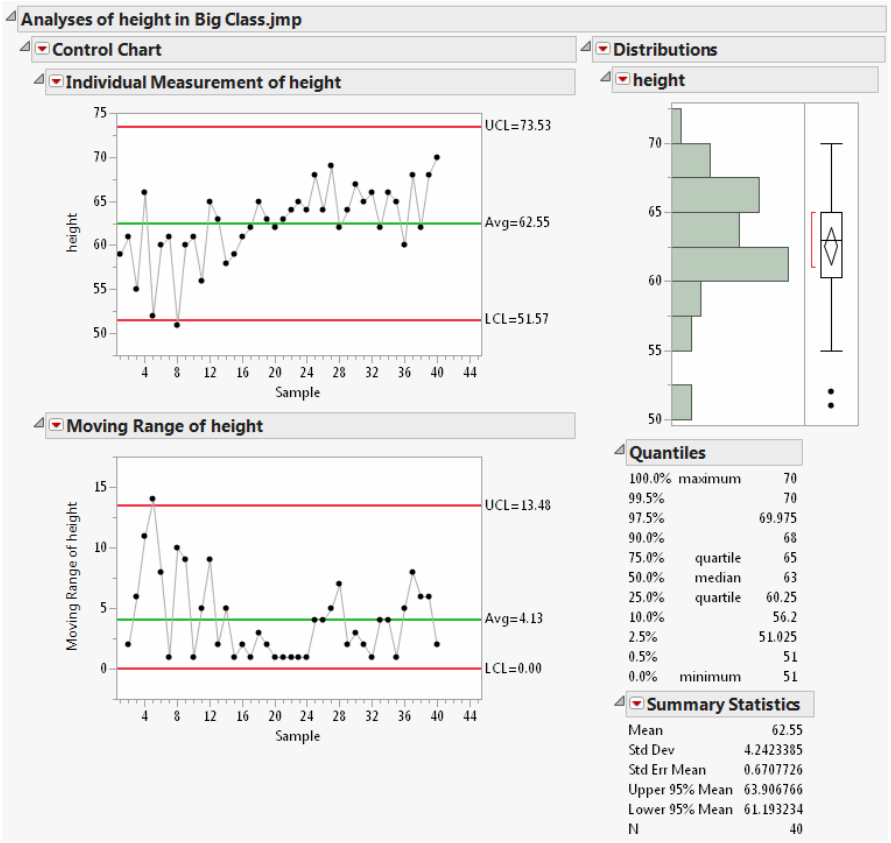
Construct Display Boxes That Contain Platforms

Another type of display that you might want to construct is your own combination of results from the analysis platforms in JMP. Write the platform script inside a display box and assemble the display boxes into a window. For ease in routing messages to it later, assign the whole script to a reference.

The following example creates several graphs and reports in one window:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
csp = New Window( "Platform Example",
   Outline Box( "Analyses of height in Big Class.jmp",
      H List Box(
         cc = Control Chart(
            Chart Col( Height, "Individual
            Measurement", "Moving Range" ),
            K Sigma( 3 )
         ),
         dist = Distribution( Columns( Height ) )
      )
   )
);
```

Figure 11.25 Example: Multiple Graphs in One Report Window



Now you can work with the window by sending messages to the reference *csp*. This is a display box reference, whose capabilities are similar to those of a `Report` for a platform. You can use multiple-argument subscripting to locate specific items within the outline tree:

```
csp["Control ?", "moving range ?"] << Close;
csp["Dist?", "quantiles"] << Close;
```

Notice that the preceding script not only assigned the whole window to a reference (`csp`) but also assigned the platform launch scripts to names (`cc` and `dist`) within their display boxes. This makes it easy to route messages to the platforms. You could, in turn, get the reports for these and have yet another way to manipulate display boxes. The following are equivalent messages that reopen the nodes:

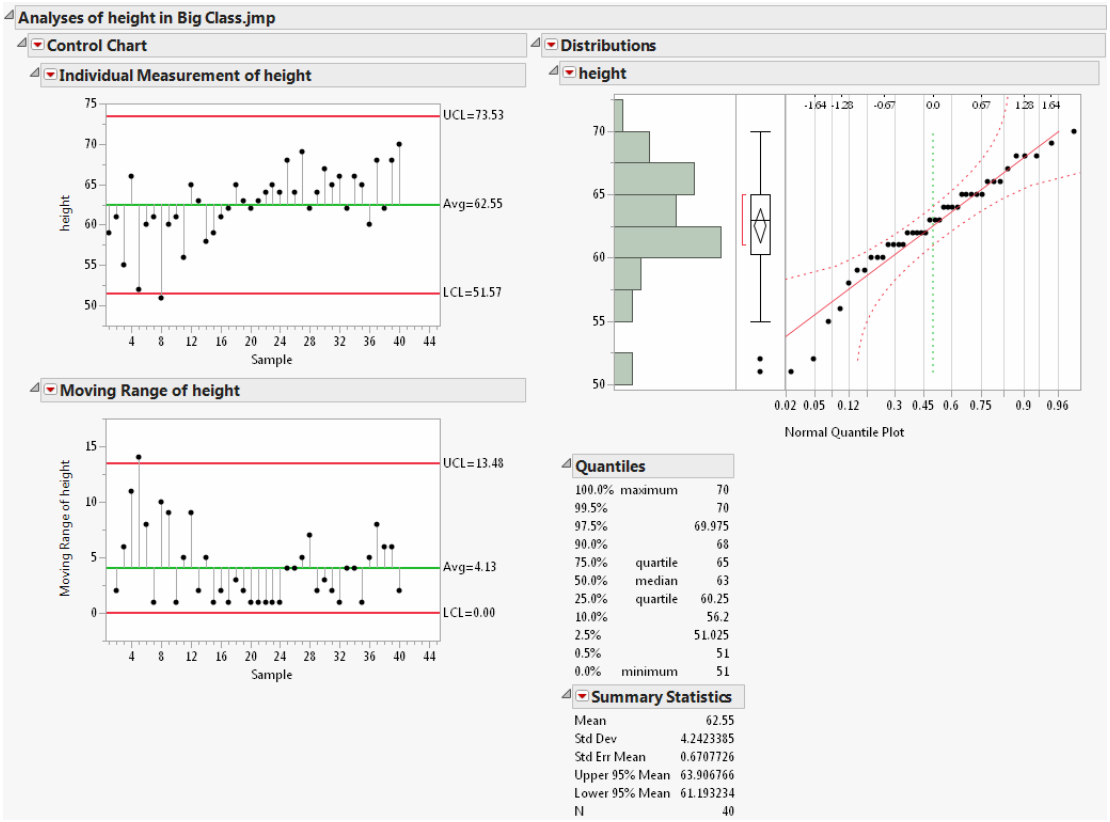
```
rcc = cc << Report;
rdist = dist << Report;
rcc["moving range ?"] << Close;
rdist["quantiles"] << Close;
```

You can send messages directly to the platform references themselves. Find out your options by searching for "Distribution" and "Control Chart" in the JMP Scripting Index. For example, Control Chart provides a message called `Needle` that creates a needle chart. Distribution provides a message called `Normal Quantile Plot`.

To execute these options from JSL, send them as messages to the platform references:

```
cc << Needle;
dist << Normal Quantile Plot;
```

Figure 11.26 Changing a Custom Report



Counting the Number of Child Boxes

Suppose that you want to count the number of child text boxes in a list box. The most reliable way is to use XPath to find a specific string in the text boxes and then use `N Items()` to count the number of text boxes.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = Bivariate(
```

```
    Y( :weight ),
    X( :height ),
    Group By( :age ),
    Fit Line( {Line Width( 3 )} )
);

/* start with the root list box, find a text edit box
that starts with “Linear Fit age==”, and count the number of children */
N Items( biv << XPath( "//ListBox[starts-with(TextEditBox,'Linear Fit
  age==')]" ) );
```

Examples of Creating a Dashboard from Two Reports

In JMP, you create dashboards in Dashboard Builder. See *Using JMP*. This section shows how to write scripts to create dashboards.

Dashboard with One Row of Reports

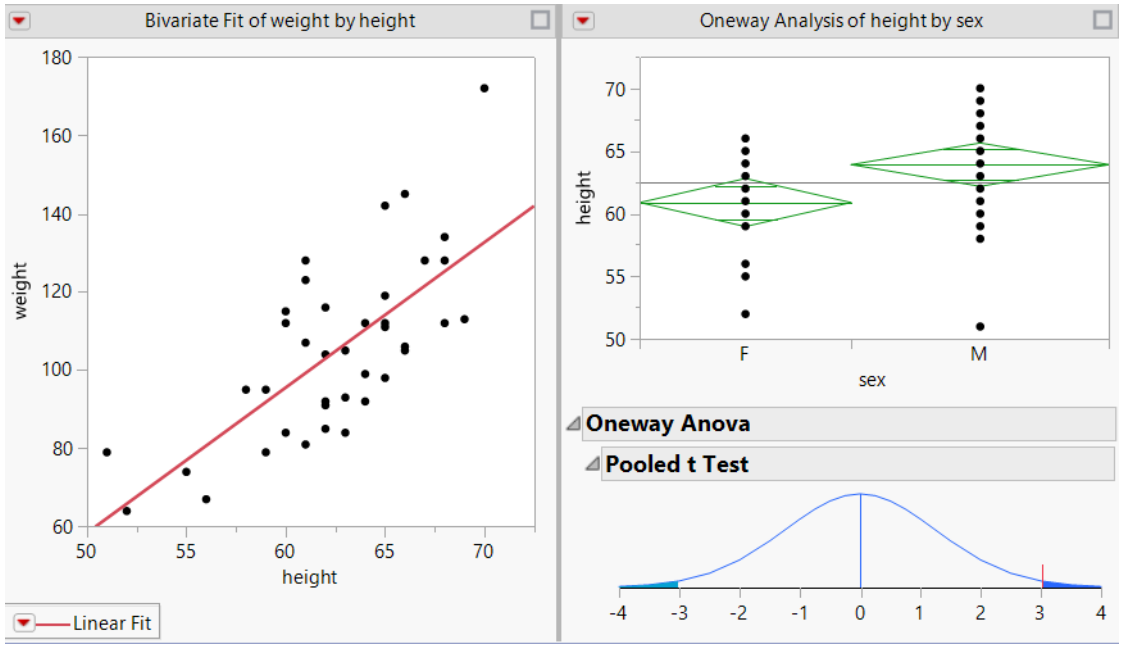
In the report window, reports appear in tab page boxes. You reposition reports in a dashboard by clicking the tab title and dragging the tab page to a highlighted drop zone. The highlighted drop zones indicate how to arrange reports horizontally, vertically, or in tabs. If a valid drop zone is not selected, the box returns to its original location.

This example shows how to write a script that creates a dashboard with two reports in a row:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Window( "My Dashboard",
  H Splitter Box( // place reports in a splitter box
  Size( 700, 400 ),
     Tab Page Box( // place report in a tab page box
        Title( "Bivariate Fit of weight by height" ),
        dt << Bivariate( // Bivariate report
           Y( :weight ),
           X( :height ),
           Fit Line,
           Report View( "Summary" ) // show only the graph
        ),
        <<Moveable( 1 ) // make the report moveable
     ),
     Tab Page Box(
        Title( "Oneway Analysis of height by sex" ),
        dt << Oneway( // Oneway report
           Y( :height ),
           X( :sex ),
           Means( 1 ),
           Mean Diamonds( 1 ),
```

```
            Report View( "Summary" )
         ),
         <<Moveable( 1 )
      ),
      /* each tab box can be dragged to a different location
      required for moveable display boxes */
      <<Dockable( 1 )
   )
);
```

Figure 11.27 Dashboard with Two Reports



Note: For dockable and moveable tabs that contain a single platform, the top-level outline box is subsumed into the tab title. This means that the red triangle options now appear in the title bar of the tabs.

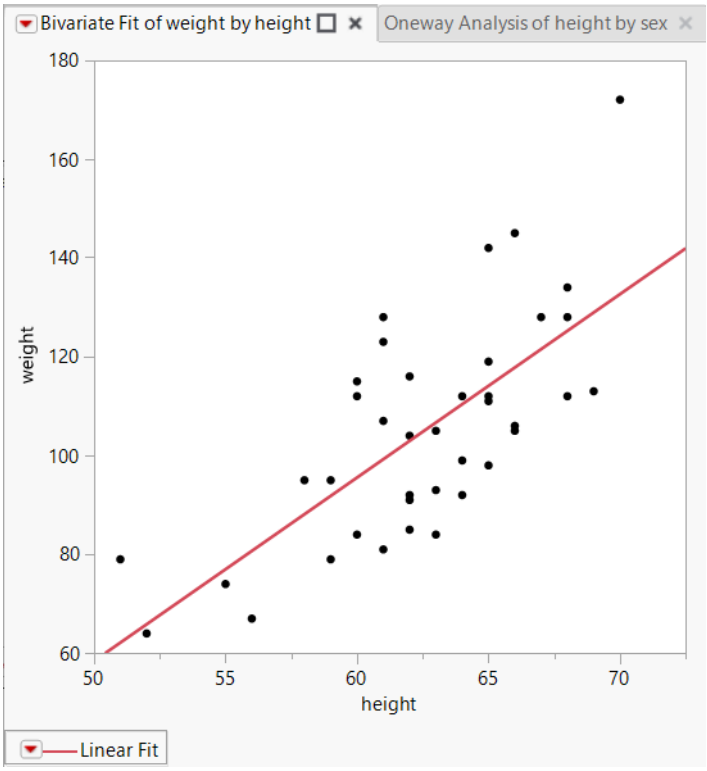
Dashboard with Tabs

The following script creates a dashboard that places each report on a tab:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
New Window( "My Dashboard",
   Tab Box( // place reports in a tab box
      Tab Page Box( // place report in a tab page box
         Title( "Bivariate Fit of weight by height" ),
```

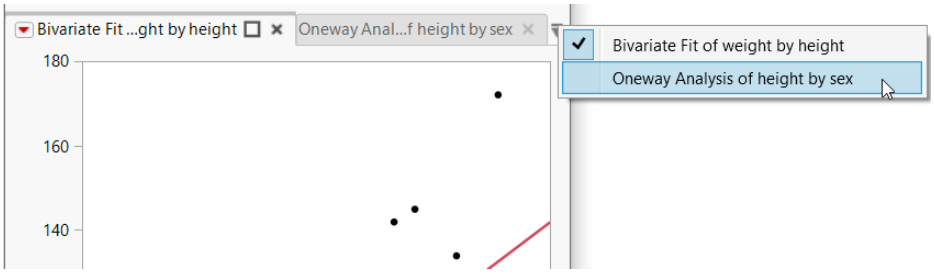
```
            dt << Bivariate( // Bivariate report
               Y( :weight ),
               X( :height ),
               Fit Line,
               Report View( "Summary" ) // show only the graph
            ),
            <<Moveable( 1 ), // make the report moveable
            <<Closeable( 1 ) // make the report closeable
         ),
         Tab Page Box(
            Title( "Oneway Analysis of height by sex" ),
            dt << Oneway( // Oneway report
               Y( :height ),
               X( :sex ),
               Means( 1 ),
               Mean Diamonds( 1 ),
               Report View( "Summary" )
            ),
            <<Moveable( 1 ),
            <<Closeable( 1 )
         ),
         /* each tab box can be dragged to a different location
         required for moveable display boxes */
         <<Dockable( 1 ),
         <<Set Overflow Enabled( 0 ) // both tab pages appear
      )
   );
```

Figure 11.28 Dashboard with Tabs



In a dashboard with several tabs, the window widens to show all tabbed reports. You might want to show only the first report and then let the user select the other tabbed reports from a list. Change `Set Overflow Enabled( 0 )` near the end of the script to `Set Overflow Enabled( 1 )`. Figure 11.29 shows the result.

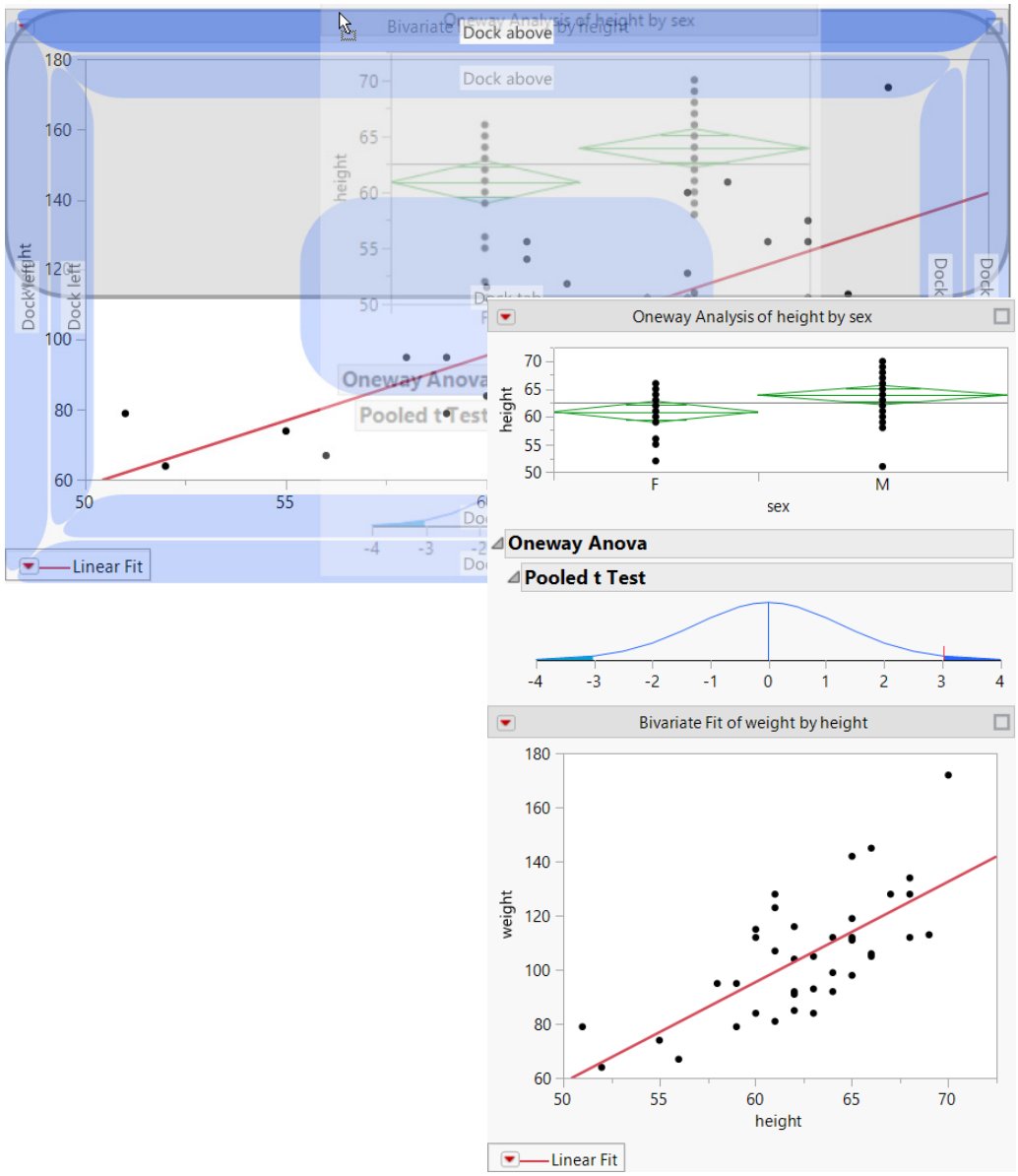
Figure 11.29 Dashboard with Overflow Tabs



How to Dock Tabs in Tab Boxes

A docking layout is a collection of `Tab Page Box()` objects and docking containers that enable reports to be rearranged. In the report window, you drag a docked tab to a highlighted drop zone to create a new layout. Figure 11.27 shows two docked reports. In Figure 11.30, the Oneway report was dragged to the highlighted drop zone to display the reports in one column.

Figure 11.30 Docking a Tab in a Dashboard



You combine `Tab Box()` with `Splitter Box()` to create docked tabs. Tab boxes and splitter boxes must directly contain other docking containers or the tab pages that define the content. Note the stacked boxes from a portion of the preceding example:

```
H Splitter Box( // place reports in a splitter box
  Size( 700, 400 ),
  Tab Page Box( // place report in a tab page box
```

The following expressions specify that the tab box and splitter box are docking containers, within which tab pages can be docked:

```
Tab Box << Set Dockable();
Splitter Box << Set Dockable();
```

In the report window, you reposition tab pages by clicking the tab title and dragging the tab page to a highlighted drop zone. The highlighted drop zones indicate how to arrange reports horizontally, vertically, or in tabs. If a valid drop zone is not selected, the box returns to its original location.

Closing Tabs

The following expression creates a close button to the right of the report title on a closeable tab page:

```
Tab Page Box << Set Closeable();
```

Clicking the close button deletes the tab and its contents. Parent containers might also be removed if they are no longer necessary (for example, if there are no more tabs in a tab box or only a single child in a splitter box).

Use the `Tab Page Box <<Set Close` message to run a JSL script when the close button is clicked. A return value of "1" indicates that the tab is closed. A return value of "0" indicates that closing the tab is not allowed at that time. Closing the tab might not be allowed as the result of a user prompt or based on the state of the program.

Controlling Wide Tabs

When a docking layout includes tab page boxes, you may place many pages in a single `Tab Box()` container. In this case, the titles may become wider than the actual content or wider than desired in the overall display. The `Tab Box <<Set Overflow Enabled( 0|1 )` message specifies that the tab titles should not cause the tab box to grow wider. If the titles do not all fit, a pop-up button appears in the upper right corner of the tab box to select tabs that are in the "overflow list". See Figure 11.29 for an example.

Example of Creating a Cluster Platform Launch Window

The following example creates a simplified replica of the Cluster platform launch window, which then launches the platform with the given arguments.

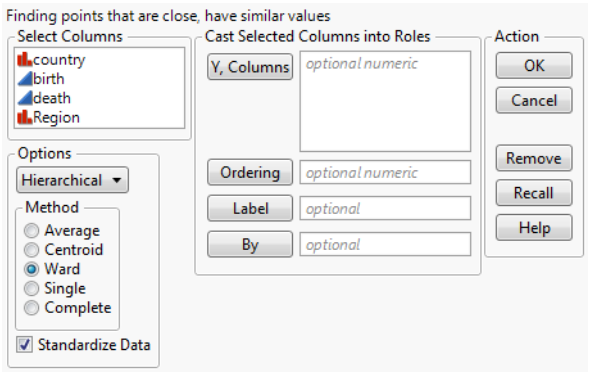
Note: Some functions (`Recall` and `Help`) are not implemented in the script, so an alert window is shown when the Recall and Help button are clicked. In addition, switching from Hierarchical to K-Means clustering does not change anything, unlike the real Cluster launch window.

```
dt = Open( "$SAMPLE_DATA/Birth Death.jmp" );
nc = N Col( dt ); // number of columns in the window
lbWidth = 130; // width of the window

// define the method list
methodList = {"Average", "Centroid", "Ward", "Single", "Complete"};
notImplemented = Expr(
   win = New Window( "Feature Not Implemented Yet", <<Modal, Button Box( "OK"
   ) )
);
clusterDlg = New Window( "Clustering", // create the window
   <<Modal,
   Border Box( Left( 3 ), Top( 2 ),
      V List Box(
         Text Box( "Finding points that are close, have similar values" ),
         H List Box(
            V List Box(
               Panel Box( "Select Columns",
                  colListData = Col List Box(
                     All,
                     width( lbWidth ),
                     NLines( Min( nc, 10 ) )
                  )
               ),
               Panel Box( "Options",
                  V List Box(
                     comboObj = Combo Box(
                        {"Hierarchical", "K-Means"},
                        <<Set( 1 )
                     ),
                     Panel Box( "Method",
                        methodObj = Radio Box( methodList, <<Set( 3 ) )
                     ),
                     checkObj = Check Box( {"Standardize Data"}, <<Set( 1, 1 ) )
                  )
               )
            ),
            Panel Box( "Cast Selected Columns into Roles",
               Line Up Box( N Col( 2 ), Spacing( 3 ),
                  Button Box( "Y, Columns",
                     colListY << Append( colListData << GetSelected )
                  ),
                  colListY = Col List Box(
                     width( lbWidth ),
                     NLines( 5 ),
                     "numeric"
```

```
            ),
            Button Box( "Ordering",
               colListO << Append( colListData << GetSelected )
            ),
            colListO = Col List Box(
               width( lbWidth ),
               NLines( 1 ),
               "numeric"
            ),
            Button Box( "Label",
               colListL << Append( colListData << GetSelected )
            ),
            colListL = Col List Box( width( lbWidth ), NLines( 1 ) ),
            Button Box( "By",
               colListB << Append( colListData << GetSelected )
            ),
            colListB = Col List Box( width( lbWidth ), NLines( 1 ) )
         )
      ),
      Panel Box( "Action",
         Line Up Box( N Col( 1 ),
            Button Box( "OK",
               If( (comboObj << Get) == 1,
                  Hierarchical Cluster(
                     Y( Eval( colListY << GetItems ) ),
                     Order( Eval( colListO << GetItems ) ),
                     Label( Eval( colListL << GetItems ) ),
                     By( Eval( colListB << GetItems ) ),
                     Method( methodList[methodObj << Get]
                     ),
                     Standardize( checkObj << Get( 1 ) )
                  ),
                  KMeansCluster( Y( colListY << GetItems ) )
               );
               clusterDlg << Close Window;
            ),
            Button Box( "Cancel", clusterDlg << Close Window ),
            Text Box( " " ),
            Button Box( "Remove",
               colListY << RemoveSelected;
               colListO << RemoveSelected;
               colListL << RemoveSelected;
               colListB << RemoveSelected;
            ),
            Button Box( "Recall", notImplemented ),
            Button Box( "Help", notImplemented )
```

```
            )
         )
      )
   )
)
);
```

Figure 11.31 The Cluster Launch Window



Example of Constructing a Custom Platform

An example in "Manipulating Expressions", showed how to use the `Substitute Into()` function to input coefficients for a quadratic polynomial into the quadratic formula and then use the formula to calculate the roots of the polynomial. That example required specifying the coefficients as arguments to `Substitute Into()`.

The section "Construct a Column Dialog" shows an example to collect coefficients from the user using a modal dialog box.

This section further develops the example into a complete customized platform that first displays a dialog box to ask for coefficients, finds the roots, and then displays the results along with a graph in a custom window.

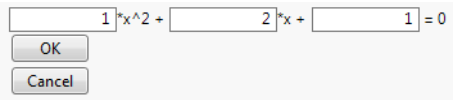
```
// create a window to collect coefficients from the user
myCoeffs = New Window( "Find the roots for the equation",
   <<Modal,
   H List Box(
      a = Number Edit Box( 1 ),
      Text Box( "*x^2 + " ),
      b = Number Edit Box( 2 ),
      Text Box( "*x + " ),
      c = Number Edit Box( 1 ),
      Text Box( " = 0" )
   ),
```

```
      Button Box( "OK",
         a = a << Get;
         b = b << Get;
         c = c << Get;
         Show( a, b, c );
      ),
      Button Box( "Cancel" )
   );

   /* calculate the results: The quadratic formula is
   x=(-b + - sqrt(b^2 - 4ac))/2a. Plug the coefficients into
   the quadratic formula */
   x = {Expr(
      (-b + Sqrt( b ^ 2 - 4 * a * c )) / (2 * a)
   ), Expr(
      (-b - Sqrt( b ^ 2 - 4 * a * c )) / (2 * a)
   )};
   xx = Eval Expr( x );


   results = Expr( // store the solution list
   /* test whether real roots were found and make an appropriate display
   if yes (for example, with the window's defaults), show roots and a graph */
      xmin = xx[1] - 5;
      xmax = xx[2] + 5;
      ymin = -20;
      ymax = 20;
      win = New Window( "The roots of a quadratic function",
         V List Box(
            Text Box( "The real roots for the equation " ),
            Text Box( "    " || Expr( po ) || " = 0" ),
            H List Box( Text Box( "are x=" ), Text Box( xxx ) ),
            Text Box( " " ), // to get a blank line
            Graph Box(
               Frame Size( 200, 200 ),
               X Scale( xmin, xmax ),
               Y Scale( ymin, ymax ),
               Line Style( 2 ),
               H Line( 0 ),
               Line Style( 0 ),
               Y Function( polynomial, x ),
               Line Style( 3 ),
               Pen Color( 3 ),
               V Line( xx[1] ),
               V Line( xx[2] ),
               Marker Size( 2 ),
```

```
            Marker( 0, {xx[1], 0}, {xx[2], 0} )
         )
      )
   );
);
error = Expr(

/* if no (for example, with a=3, b=4, c=5), put up an error
window with a helpful graph */
   win = New Window( "Error",
      V List Box(
         Text Box( " " ),
         Text Box( "  Polynomial " || po || " has no real roots.  " ),
         Text Box( " " ),
         Text Box( "Examine a graph of the function to get an idea why." ),
         Graph Box(
            Frame Size( 200, 200 ),
            X Scale( -20, 20 ),
            Y Scale( -20, 20 ),
            Line Style( 2 ),
            H Line( 0 ),
            Line Style( 0 ),
            Y Function( polynomial, x )
         )
      )
   )
);

/* either way, the script needs to have some strings ready
rewrite the polynomial with the coefficients specified */
polynomial = Expr( a * x ^ 2 + b * x + c );

// store this instance of the polynomial as a string
po = Char( Eval Expr( polynomial ) );

// store the solution list as a string
xxx = Char( Eval Expr( x ) );

If( Is Missing( xx[1] ) | Is Missing( xx[2] ), // now it's ready for a test
   error,
   results
);
```

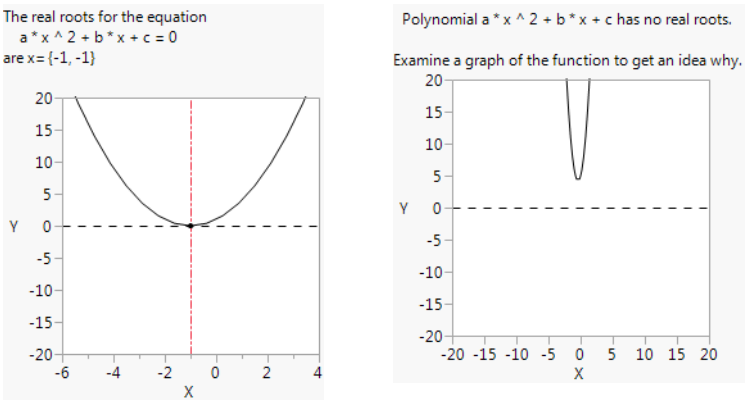
When you run the preceding script, you first see a window like this:

Figure 11.32 A Custom Platform Launch Window



Clicking **OK**, displays a results window (Figure 11.33, left), with either the roots or an error message. Rerun the script, input 5, 4, and 5 respectively and click **OK**. Note that JMP displays an error message (Figure 11.33, right).

Figure 11.33 The Custom Platform's Report

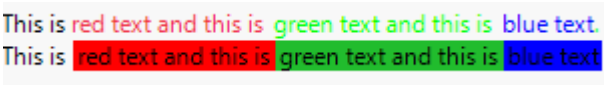


Example of Specifying Text and Background Colors

The `markup` argument in a display box enables you to write HTML code to specify text and background colors. The color can be a hex value (starting with "0x"or "#") or a known JMP color. Here is an example:

```
New Window( "Colors",
   Text Box(
      "This is <font color=\!"Red\!">red text and this is <font
   color=\!"0x00ff00\!">green text and this is <font color=\!"#0000FF\!">blue
   text</font>.",
      <<markup
   ),
   Text Box(
      "This is <background color=\!"#ff0000\!"> red text and this
   is<background color=\!"Green\!"> green text and this is<background
   color=\!"0x0000FF\!"> blue text</background>",
      <<markup
   )
);
```

Figure 11.34 HTML Text and Background Colors



Modal Windows

As described in "Constructors for New Windows", *modal* means that the user must interact with the window immediately. Clicking outside the window produces an error sound. Script execution stops until the user responds to the window.

JMP provides two types of modal window functions:

- `New Window()` with the `Modal` message. This creates a new window in which you arrange other display boxes.
- `Column Dialog()` constructs a window for column role assignments as found in many launch windows. The specialized constructors for `Column Dialog()` mean that you have less flexibility than with `New Window()`. However, `Column Dialog()` lets you easily create a launch window.

Construct a Modal Window

When you submit a script with a modal window, JMP draws the window, waits for the user to make choices and click **OK**, and then stores a list of the variables with their values. Any attempt to click outside the window produces an error sound, and script execution is suspended until the user clicks **OK** or **Cancel**.

The `Column Dialog()` function is specifically intended to prompt users to choose columns from the current (topmost) data table. Windows created by `Column Dialog()` are also modal windows.

Tips:

- Put all the modal windows near the beginning of the script, if possible. This way, all the user interaction can be accomplished at once, and then users can leave JMP to finish its work unattended.
- Make sure your modal windows give the user enough information. Do not just present a number field. Tell users how the number is used. If there are limits for valid responses, say so.

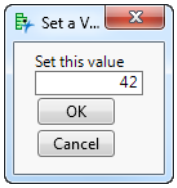
A very simple modal window might request a value for one variable:

```
win = New Window( "Set a Value",
   <<Modal,
```

```
    Text Box( "Set this value" ),
    variablebox = Number Edit Box( 42 ),
    Button Box( "OK" ),
    Button Box( "Cancel" )
);
```

Notice that the argument to `Number Edit Box()` is the default value, 42.

Figure 11.35 Sample Modal Display Box



If you click **OK**, the window closes and {`Button(1)`} is returned. To get the value entered in the box, use the `Return Result` message and a subscript into the window. Note that the deprecated `Dialog()` returns the list of variable assignments the same way.

```
win = New Window( "Set a Value",
    <<Modal,
    <<Return Result,
    Text Box( "Set this value" ),
    variablebox = Number Edit Box( 42 ),
    Button Box( "OK" ),
    Button Box( "Cancel" )
);

// create a subscript to the variablebox variable
Write( win["variablebox"] );
```

In this example, the user typed 33 in the number edit box. That value is stored in `win`.

If you click **Cancel**, the window closes, it returns {`Button(-1)`}, and the script continues.

To detect whether the user clicked Cancel, add the following expression:

```
If( win["Button"] == 1,
    Print( win["variablebox"] );
    ,
    Print("Canceled")
);
```

Note that a modal window must have at least one button. If you omit the `Button Box()` from your script, an OK button is automatically included. To omit the OK button, set the visibility to collapse.

```
win = New Window( "No OK Button",
```

```
    <<Modal,
    b = Button Box( "Close Window", b << Close Window ),
    bb = Button Box( "OK", <<Visibility( "collapse" ) )
);
```

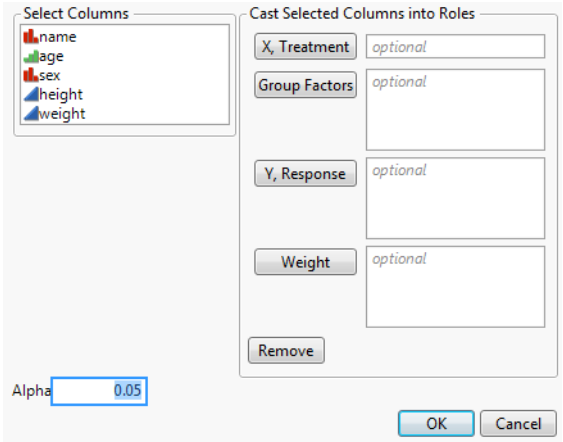
Note: Modal windows require at least one button for dismissing the window. You can use **OK**, **Yes**, **No**, or **Cancel** to label modal window buttons. If you do not include at least one of these buttons in a modal window, JMP adds an **OK** button.

Construct a Column Dialog

`Column Dialog()`, a variant of the deprecated `Dialog()` function, lets you prompt for column selections from the current data table or contextual data table. The `OK`, `Cancel`, and `Remove` buttons and the list of columns to choose from the data table are added automatically.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dlg = Column Dialog(
   Col ID = Col List( "X, Treatment", Max Col( 1 ) ),
   Group = Col List( "Group Factors" ),
   Split = Col List( "Y, Response" ),
   w = Col List( "Weight" ),
   H List( "Alpha", alpha = Edit Number( .05 ) )
);
```

Figure 11.36 Column Dialog



The following example returns a list similar to this one, depending on the user's choices:

```
{Col ID = {}, Group = {}, Split = {}, w = {}, alpha = 0.05, Button( -1 )}
```

For each destination list, a `Col List` clause must be a direct argument of `Column Dialog()` (not nested inside some other argument). An optional `MaxCol(`*`n`*`)` argument restricts the number of data columns that can be chosen to *n*. Lists are *always* returned, although they can sometimes be empty lists. You can include as many as twelve `Col List` clauses.

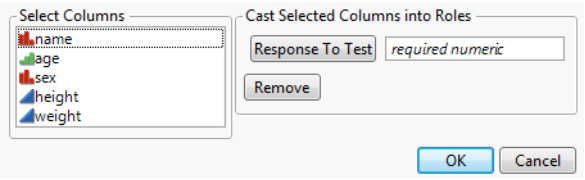
Other items permitted in the deprecated `Dialog()` are permitted in `Column Dialog()` and have the same functionality. You can specify the minimum and maximum number of columns that are allowed in a column dialog box with the `MinCol` and `MaxCol` arguments.

You can also specify the modeling type of the columns that are allowed to be selected (`Ordinal`, `Nominal` or `Continuous`). You can set the width of the list using `Select List Width(`*`pixels`*`)` argument. To set the width of the column list, use `Width(`*`pixels`*`)` inside the `Col List()` function.

The following example generates a column dialog box that only allows the selection of exactly one numeric column:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
rt_dlg = Column Dialog(
   cv = Col List( "Response To Test", MaxCol( 1 ), MinCol( 1 ), DataType(
   "Numeric" ) )
);
```

Figure 11.37 Restricting Selection of Columns



The `Data Type` choices are `Numeric`, `Character`, and `Any`.

In addition, use the `Columns` specification to pre-fill some column selections. For example, the following script assigns height to the X role and weight and age to the Y role:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dlg = Column Dialog(
   xCols = Col List( "X, Factors", Columns( :height ) ),
   yCols = Col List( "Y, Response", Columns( :weight, :age ) )
);
```

Comparison of Column Dialog and New Window

`New Window()` and `Column Dialog()` enable you to construct a launch window that has a column selector. Use `Column Dialog()` to construct a simple launch window. Several items are added automatically to column dialogs: the Remove and Cancel buttons, the list of columns, and an OK button (if no buttons are explicitly defined).

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dlg = Column Dialog(
   x = ColList( "X", Max Col( 1 ) ),
   y = ColList( "Y", Max Col( 1 ) )
);

// returns dlg = {x = {:weight}, y = {:height}, Button(1)};
Show( dlg );

If( dlg["Button"] == 1,
   xCol = dlg["x"];
   yCol = dlg["y"];
);

Show( xCol, yCol );
   xCol = {:height};
   yCol = {:weight};
```

`New Window()` is more flexible because of numerous optional arguments. In the following example, the user must select two columns before clicking the OK button. A Bivariate graph is then created. Validating the user's selections is not possible in `Column Dialog()`.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
xvar = .;
yvar = .;
win = New Window( "Example of Returning Values",
   <<Modal,

   // require the user to select two variables before clicking OK
   <<On Validate(
      Show( xvar, yvar );
         If( Is Missing( xvar ) | Is Missing( yvar ),

         // if xvar or yvar are missing do nothing when OK is clicked
         0,
         1
      );
   ),
   Text Box( " Select two numeric columns. " ),
   H List Box(
```

```
      Text Box( " X, Factor " ),
      x = Col List Box(
         dt, // data table reference
         all, // display all columns from the data table

         // get the name of the selected column before the window closes
         xvar = (x << Get Selected)[1];
         Show( xvar );
      ),
      Text Box( "Y, Response" ),
      y = Col List Box(
         dt,
         all,
         yvar = (y << Get Selected)[1];
         Show( yvar );
      )
   )
);
If (win["Button"] == 1, // if the user clicks OK...
xcol = Column( dt, xvar ); //  get the columns
ycol = Column( dt, yvar );
dt << Bivariate( Y( ycol ), X( xcol ) ); // create a Bivariate plot
);
```

Constructors for Column Dialogs

Constructors such as `Edit Number()` and `Check Box()` can be used in column dialogs, which construct launch windows with a column selector. The following example shows a typical column dialog with an `Edit Number()` box:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
Column Dialog(
   ex x = ColList( "X", Max Col( 1 ) ),
   HList( "Alpha",
   ex = Edit Number( .05 ) )
);
```

Table 11.3 describes the column dialog constructors.

Notes:

- A column dialog automatically includes the list of columns in the data table.
- A column dialog also automatically includes OK, Cancel, and Remove buttons.

Table 11.3 Column Dialog Constructors

Constructor	Syntax	Explanation
`Button`	`Button( "OK" ), Button( "Cancel" )`	Draws an OK or a Cancel button. If OK is clicked, `Button(1)` is returned. If Cancel is clicked, `Button(-1)` is returned.
`Check Box`	*var*`( CheckBox( "`*Text after box*`", <1|0> ) )`	When OK is clicked, a selected check box assigns 1 to the variable. A check box that is not selected assigns 0 to the variable. Add an optional 1 for the check box to be selected (on), or 0 for it to be not selected (off) when the window first appears. The default value is 0 (off).
`Col List`	*var*`=Col List( "`*role*`", <MaxCol( `*n*` )>, <Datatype(` *type* `)> )`	Creates a selection destination with a *role* button; the user's choices are returned in a list item of the form *var*={*choice 1*, *choice 2*, …, *choice n*}. You can specify the minimum and maximum number of columns using `MinCol(` *n* `)` or `MaxCol(` *n* `)`. You can specify the required data type of the column using `Datatype(`*type*`)`. The choices for *type* are `Numeric`, `Character`, or `Rowstate`.
`Combo Box`	*var*`( ComboBox( "`*choice1*`", "`*choice2*`", ...) )`	Produces a menu with the choices listed. The first choice is the default. Choices evaluate to quoted text strings. Choices can also be inside a list.
`Edit Number`	*var*`( Edit Number(` *number* `) )`	Produces an edit field for a number with *number* as the default value. When OK is clicked, the number entered in the field is assigned to the variable.

Table 11.3 Column Dialog Constructors *(Continued)*

Constructor	Syntax	Explanation
Edit Text	*var*(Edit Text("*string*", <width(*x*)>))	Produces an edit field for a string with *string* as the default value. You can also specify the minimum width of the box in pixels. The default width is 72 pixels. When OK is clicked, the text entered in the field is assigned to the variable.
HList	HList(*item*, *item*, ...)	Top aligns and spaces the *items* in a horizontal row. Placing a pair of VLists within an HList produces a top aligned, spaced pair of columns.
Line Up Box	Line Up Box(*n*, *item_11*, *item_12*, ..., *item_1n*, ..., *item_nn*)	Lines up the *items* listed in *n* columns, where *item_ij* is the *j*th item of the *i*th row.
List Box	var=List Box({"item", "item", ...}, width(50), max selected(2), NLines(6))	Creates a display box that contains a list of items.
Radio Buttons	*var*(RadioButtons("*choice1*", "*choice2*", ...))	Produces a vertical, left-justified list of radio buttons with the choices specified. The first choice is the default. When OK is clicked, the button that is selected is assigned to the variable. Choices must evaluate to quoted text strings.
string	"*string*"	Draws text in the window. For example, can include a labeling string before an Edit Number field. All strings must be quoted.
VList	VList(*item*, *item*, ...)	Left aligns and spaces the *items* in a vertical column. Placing a pair of HLists within a VList produces a left-aligned, spaced pair of rows.

Convert Deprecated Dialog to New Window

Because `New Window()` provides more display options and better control over the content and functions of the window, rewrite your `Dialog()` scripts.

Note: `Dialog()` scripts will not run in the future.

The following sections illustrate how to rewrite `Dialog()` scripts as `New Window()` scripts.

- "New Window Example"
- "Example of Deprecated Dialog"
- "Differences Between New Window and the Deprecated Dialog"
- "Optional Scripts in New Window"

New Window Example

The following example creates a window with multiple vertical and horizontal list boxes:

```
win = New Window( "New Window Example",
   <<Modal,
   <<ReturnResult, // ensure that you get the same result as with Dialog
   V List Box(
      V List Box(
         Text Box( "Radio Frequency Embolism Projection" ),
         Line Up Box(
            NCol( 2 ),
            Text Box( "Lower Spec Limit" ),
            lsl_box = Number Edit Box( 230 ),
            Text Box( "Upper Spec Limit" ),
            usl_box = Number Edit Box( 340 ),
            Text Box( "Threshold" ),
            threshold_box = Number Edit Box( 275 )
         ),
         H List Box(
            Panel Box( "Type of Radio",
                  rb_box1 = Radio Box( {"RCA", "Matsushita",
                  "Zenith", "Sony"} )),
            Panel Box( "Type of Antenna" ,
                  rb_box2 = Radio Box( {"Dish", "Helical", "Polarizing",
                  "Radiant Array"} ) )
         ),
         cb_box1 = Check Box( "Emission Synchronization" ),
         Text Box( "Title for plot" ),
         title_box = Text Edit Box( "My projection" ),
```

```
            H List Box( Text Box( "Quality" ),
            cb_box2 = Combo Box( {"Fealty", "Loyalty", "Piety", "Obsequiousness"}
      ) )
         ),
         H List Box(
            Align( Right ),
            Spacer Box(),
            Button Box( "OK",
               lsl = lsl_box << Get;
               usl = usl_box << Get;
               threshold = threshold_box << Get;
               radio_type = rb_box1 << Get;
               antenna = rb_box2 << Get;
               synch = cb_box1 << Get;
               title = title_box << Get Text;
               quality = cb_box2 << Get;
            ),
            Button Box( "Cancel" )
         )
      )
   );
   If(win["Button"] == 1,

   // if the user clicks OK, show the selections in the log
      Show( "OK", lsl, usl, threshold, radio_type, antenna, synch, title,
      quality);
      ,
      Show( "Canceled" ); // if the user clicked Cancel, print "Canceled"
   );
```

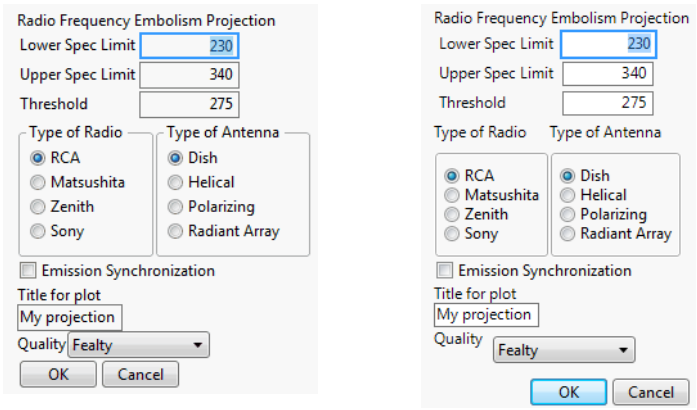
Example of Deprecated Dialog

Compare the previous `New Window()` example to this example of the deprecated `Dialog()`.

```
dlg = Dialog(
   Title( "Dialog Example" ),
   H List(
      V List(
         "Radio Frequency Embolism Projection",
         Lineup( 2,
            "Lower Spec Limit", lsl = Edit Number( 230 ),
            "Upper Spec Limit", usl = Edit Number( 340 ),
            "Threshold", threshold = Edit Number( 275 )
         ),
         H List(
            V List(
```

```
                "Type of Radio",
                type = Radio Buttons( "RCA", "Matsushita", "Zenith", "Sony" )
             ),
             V List(
                "Type of Antenna",
                antenna = Radio Buttons( "Dish", "Helical", "Polarizing",
                   "Radiant Array" )
             )
          ),
          synch = Check Box( "Emission Synchronization", 0 ),
          "Title for plot",
          title = Edit Text( "My projection" ),
          H List(
             "Quality",
             quality = Combo Box( "Fealty", "Loyalty", "Piety", "Obsequiousness"
             )
          )
       ),
       V List( Button( "OK" ), Button( "Cancel" ) )
    )
);
If( dlg["Button"] == 1,
   Show(
      "OK",
      dlg["lsl"],
      dlg["usl"],
      dlg["threshold"],
      dlg["type"],
      dlg["antenna"],
      dlg["synch"],
      dlg["title"],
      dlg["quality"]
   ),
   Show( "Canceled" )
);
```

Figure 11.38 Results from New Window (left) and the Deprecated Dialog (right)



Note: Use `Return Result` to get the same result in `New Window()` as with the deprecated `Dialog()`. See "Construct a Modal Window" for an example.

Differences Between New Window and the Deprecated Dialog

This section shows the differences between display boxes in `New Window()` and the deprecated `Dialog()` scripts. Table 11.4 summarizes those differences. More information can be found in the sections that follow the table.

Table 11.4 Display Boxes in New Window and Deprecated Dialog Scripts

Box Type	New Window	Deprecated Dialog

"Strings and Lists"	`win = New Window( "Combo Box",`
	`  <<Modal,`
	`  <<ReturnResult,`
	`  cb = Combo Box( {"True", "False"} ),`
	`);`
		`Dialog( Title( "Combo Box" ),`
		`  cb = Combo Box( "True", "False" ),`
		`);`

"List Boxes and Check Boxes"	`win = New Window( "H List Box",`
	`  <<Modal,`
	`  <<ReturnResult,`
	`  H List Box(`
	`  kb1 = Check Box( "a" ),`
	`  kb2 = Check Box( "b" ),`
	`  kb3 = Check Box( "c" )`
	`  ),`
	`);`
		`dlg = Dialog( Title( "H List" ),`
		`  H List(`
		`  kb1 = Check Box( "a", 0 ),`
		`  kb2 = Check Box( "b", 0 ),`
		`  kb3 = Check Box( "c", 0 )`
		`  ),`
		`);`

Table 11.4 Display Boxes in New Window and Deprecated Dialog Scripts *(Continued)*

Box Type	New Window	Deprecated Dialog
"Line Up Items"	win = New Window("Line Up Box", <<Modal, <<ReturnResult, V List Box(Line Up Box(NCol(2), Text Box("Set this value"), var1=Number Edit Box(42), Text Box("Set another value"), var2=Number Edit Box(86),),), H List Box(Button Box("OK"), Button Box("Cancel")); // returns: {Button(1)}	dlg = Dialog(V List(Line Up(2, "Set this value", variable=Edit Number(42), "Set another value", var2=Edit Number(86)), H List(Button("OK"), Button("Cancel")))); // returns: {variable = 42, var2 = 86, Button(1)}
"Radio Boxes"	win = New Window("Radio Box", <<Modal, <<ReturnResult, Panel Box("Select", rbox = Radio Box({"a", "b", "c"})));	dlg = Dialog(Title("Radio Buttons"), rb = Radio Buttons({"a", "b", "c"}),);
"Text Edit Boxes"	win = New Window("Text Edit Box", <<Modal, <<ReturnResult, V List Box(Text Box("Strings"), str1 = Text Edit Box("The"), str2 = Text Edit Box("quick"),));	dlg = Dialog(Title("Edit Text"), V List("Strings", str1 = Edit Text("The"), str2 = Edit Text("quick"),));

Table 11.4 Display Boxes in New Window and Deprecated Dialog Scripts *(Continued)*

Box Type	New Window	Deprecated Dialog

"Number Edit Boxes"	`win = New Window( "Number Edit Box",`
	`<<Modal,`
	`<<ReturnResult,`
	`V List Box(`
	`Text Box("Random Numbers"),`
	`num1 = Number Edit Box(`
	`Random Uniform()),`
	`num2 = Number Edit Box(`
	`Random Uniform() * 10),`
	`...);`
		`dlg = Dialog(`
		`Title( "Edit Number" ),`
		`V List(`
		`"Random Numbers",`
		`num1 = Edit Number(`
		`Random Uniform() ),`
		`num2 = Edit Number(`
		`Random Uniform() * 10 ),`
		`...);`

Strings and Lists

A major difference between the `New Window()` and the deprecated `Dialog()` functions is how you specify lists and strings as arguments. In `New Window()`, the items must be placed in a list. In the deprecated `Dialog()`, items are generally separated by commas. For example, compare the following two instances of a combo box.

```
win = New Window( "Combo Box", // modal New Window
   <<Modal,
   <<Return Result,
   cb = Combo Box( {"True", "False"} ), // items are in a bracketed list
);

Dialog( Title( "Combo Box" ),// modal Dialog
   cb = Combo Box( "True", "False" ), // items are separated by commas
   );
```

Note: The script does not explicitly define the OK button. For modal windows, JMP automatically adds the OK button. Explicitly define each additional button.

Additionally, the deprecated `Dialog()` could include `" "` to indicate empty text. `New Window()` requires that empty text appear in a `Text Box (" ")`.

List Boxes and Check Boxes

In `New Window()`, boxes are horizontally aligned with `H List Box()` and vertically aligned with `V List Box()`. In the deprecated `Dialog()`, you use `H List` and `V List` instead.

Note: Boxes are vertically aligned by default if `H List Box()` or `V List Box()` are omitted.

The following example creates a window with three horizontal check boxes and an OK button:

```
win = New Window( "H List Box",
   <<Modal,
   <<Return Result,
   H List Box(
      kb1 = Check Box( "a" ),
      kb2 = Check Box( "b" ),
      kb3 = Check Box( "c" )
   ),
);
```

Here is the same window created with the deprecated `Dialog()` and an `H List()`:

```
dlg = Dialog( Title( "H List" ),
   H List(
         kb1 = Check Box( "a", 0 ),
         kb2 = Check Box( "b", 0 ),
         kb3 = Check Box( "c", 0 )
     ),
);
```

Line Up Items

In `New Window()`, use `Line Up Box()` to arrange items in the number of columns that you specify. In the deprecated `Dialog()`, you used `Line Up`.

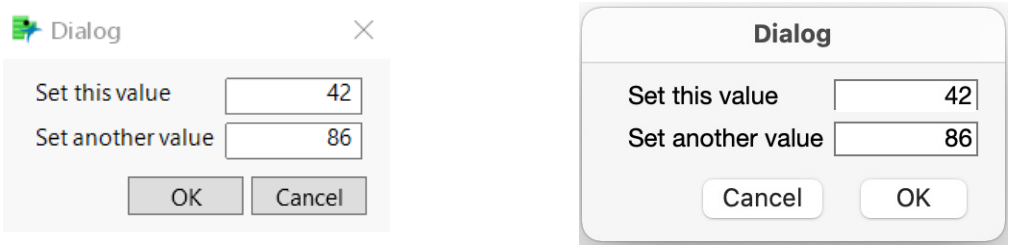
The following example arranges the text boxes in one column and the number edit boxes in another column:

```
win = New Window( "Line Up Box",
   <<Modal,
   <<Return Result,
   V List Box(
      Lineup Box( N Col( 2 ),
         Text Box( "Set this value" ),
         var1 = Number Edit Box( 42 ),
         Text Box( "Set another value" ),
         var2 = Number Edit Box( 86 ),

      ),
   ),
   H List Box( Button Box( "OK" ), Button Box( "Cancel" ) ) )
);
```

Clicking **OK** returns the following result in the log window:

```
List( 3 elements ) assigned.
```

Figure 11.39 Default Dialog Arrangements for Windows (left) and macOS (right)



Here's the same window created with the deprecated `Dialog()` and `Line Up`:

```
dlg = Dialog(
   V List(
      Line Up( 2,
         "Set this value", variable = Edit Number( 42 ),
         "Set another value", var2 = Edit Number( 86 )
      ),
      H List( Button( "OK" ), Button( "Cancel" ) )
   )
);
```

Clicking **OK** returns the following result in the log window:

```
{variable = 42, var2 = 86, Button(1)}
```

Note: JMP does exert some control over OK and Cancel button positions to ensure that dialog boxes are consistent with what the operating system expects. In certain cases, JMP needs to override your `H List Box()`, `V List Box()`, and `Line Up Box()` settings for `Button( "OK" )` and `Button( "Cancel" )`. Do not be alarmed if the result is slightly different from what you expect.

Radio Boxes

Another difference between `New Window()` and the deprecated `Dialog()` is the usage of `Radio Box`.

In `New Window()`, you must define the `Panel Box()` container for `Radio Box()` if you want the radio buttons in a box.

```
win = New Window( "Radio Box",
   <<Modal,
   <<Return Result,
   Panel Box( "Select",
      rbox = Radio Box( {"a", "b", "c"} )
   )
);
```

In the deprecated `Dialog()`, the `Radio Buttons()` automatically appear in a panel box.

```
dlg = Dialog( Title( "Radio Buttons" ),
   rb = Radio Buttons( {"a", "b", "c"} ),
);
```

Text Edit Boxes

In `New Window()`, use `Text Edit Box()` to create editable boxes that contain specified strings. In the deprecated `Dialog()`, you used `Edit Text()`.

```
win = New Window( "Text Edit Box",
   <<Modal,
   <<Return Result,
   V List Box(
      Text Box( "Strings" ),
      str1 = Text Edit Box( "The" ),
      str2 = Text Edit Box( "quick" ),
      str3 = Text Edit Box( "brown" ),
      str4 = Text Edit Box( "fox" ),
      str5 = Text Edit Box( "jumps" ),
      str6 = Text Edit Box( "over" ),
      str7 = Text Edit Box( "the" ),
      str8 = Text Edit Box( "lazy" ),
      str9 = Text Edit Box( "dog" ) ) );
```

Here is the same window created with the deprecated `Dialog()` and `Edit Text()`:

```
dlg = Dialog(
   Title( "Edit Text" ),
   V List(
      "Strings",
      str1 = Edit Text( "The" ),
      str2 = Edit Text( "quick" ),
      str3 = Edit Text( "brown" ),
      str4 = Edit Text( "fox" ),
      str5 = Edit Text( "jumps" ),
      str6 = Edit Text( "over" ),
      str7 = Edit Text( "the" ),
      str8 = Edit Text( "lazy" ),
      str9 = Edit Text( "dog" )
   )
);
```

Alternatively, use `String Col Edit Box()` to create editable boxes within in a table structure that contains specified strings. For example, the following script creates a window named "String Col Edit Box". The window contains a column (labeled "Strings") of editable boxes, each of which contains the specified string:

```
win = New Window( "String Col Edit Box",
   <<Modal,
   <<Return Result,
   steb = String Col Edit Box(
      "Strings",
      {"The", "quick", "brown", "fox", "jumps", "over", "the", "lazy", "dog"}
   )
);
```

Note: In the preceding example, `New Window()` assigns only two elements to the list (the specified list items and the button).

Number Edit Boxes

In `New Window()`, use `Number Edit Box()` to create editable boxes that contains the specified numbers. In the deprecated `Dialog()`, you used `Edit Number()`.

```
win = New Window( "Number Edit Box",
   <<Modal,
   <<ReturnResult,
   V List Box(
      Text Box( "Random Numbers" ),
      num1 = Number Edit Box( Random Uniform() ),
      num2 = Number Edit Box( Random Uniform() * 10 ),
      num3 = Number Edit Box( Random Uniform() * 100 ),
      num4 = Number Edit Box( Random Uniform() * 1000 )
   ),
);
```

Here is the same window created with the deprecated `Dialog()` and `Edit Number()`:

```
dlg = Dialog(
   Title( "Edit Number" ),
   V List(
      "Random Numbers",
      num1 = Edit Number( Random Uniform() ),
      num2 = Edit Number( Random Uniform() * 10 ),
      num3 = Edit Number( Random Uniform() * 100 ),
      num4 = Edit Number( Random Uniform() * 1000 )
   ),
);
```

Another method for creating the same window uses `Number Col Edit Box()` to create editable boxes within in a table structure that contains specified numbers. For example, the following script creates a window named "Number Col Edit Box". The window contains a column (labeled "Random Numbers") of editable boxes, each of which shows the specified type of random number:

```
win = New Window( "Number Col Edit Box",
   <<Modal,
   <<ReturnResult,
   nceb = Number Col Edit Box(
      "Random Numbers",
      {num1 = Random Uniform(), num2 = Random Uniform() * 10, num3 =
      Random Uniform() * 100, num4 = Random Uniform() * 1000}
   ),
);
```

Note: In the preceding example, the `Number Col Edit Box()` assigns only two elements to the list. The `New Window()` and deprecated `Dialog()` examples for `Number Col Edit Box()` assign five elements to the list.

Optional Scripts in New Window

One of the benefits of using `New Window()` is the ability to add optional scripts to display boxes. In the deprecated `Dialog()`, the following combo box could not include an optional script. If you wanted an action associated with any of the display box controls, you had to place a script as the control's last argument. For example:

```
win = New Window( "Combo Box",
<<Modal,
<<ReturnResult,
   comboObj = Combo Box(
      {"True", "False"},
      << Set( 1 ),
      Print( comboObj << Get )
   )
);
```

When the user selects a different value, the selected item number (1 or 2 in this case, because there are two items in the combo box) is printed to the log.

Scripting the Script Editor

Even the script editor window is a display tree in JMP, which means you can write a JSL script to write and save another JSL script.

There is no `New Script` command. Instead, to open a new script window, you use the `New Window()` function and then send it a message to tell it that it's a script window:

```
ww = New Window( "Window Title", <<Script, "Initial Contents" );
```

The last argument is optional. If you include a string, the new script window contains that string.

In the `New Window` example above, `ww` is a reference to the display box that is the entire window. To write to a script window, you first need to get a reference to the part of the display box that you can write to, which is called a script box:

```
ed = ww[Script Box(1)];
```

Using the reference `ed`, you can add text, remove text, and get the text that is already there.

```
ed << Get Text();
   "Initial Contents"
```

Use `Set Text` to set all the text in the script window. The following command clears all text in the Script Window and then adds `aaa=3;` followed by a return:

```
ed << Set Text( "aaa=3;\!N" );
```

Use `Append` to add additional text to the end of the script window.

```
ed << Append Text( "bbb=1/10;" );
ed << Append Text( "\!Ncc=4/100;" );
```

Use the `Get Line Text` command to get the text at the line of a specified line number. Use the `Set Line Text` command to replace a specified line of text with new text.

```
ed << Get Line Text( 2 );
ed << Set Line Text( 2, "bbb = 0.1;" );
```

Use the `Get Line Count` message to get the total number of lines in the script. The `Get Lines` message returns a list of each line in the script as a string.

```
ed << Get Line Count();
ed << Get Lines();
```

Use the `Reformat` message to automatically format a script for easier reading.

```
ed << Reformat();
```

To run an entire script in a script window, use it the `Run` message.

```
ed << Run();
```

To close the script window without saving the script, send the window the `Close Window` message, just like you can do with any JMP window.

```
ww << Close Window( nosave );
```

To save and view the script, use the `Save Text` and `Load Text File` messages:

```
ww = New Window( "Test", <<Script, "Open(\!"$SAMPLE_DATA/Big
  Class.jmp\!");");
ww << Save Text( "$TEMP/Test.jsl" );
Write( Load Text File( "$TEMP/Test.jsl" ) );
```

Note: When the script is loaded into the script window, the window is considered "dirty" or modified. Use `<<Set Dirty(0)` to remove this property. Otherwise, you are prompted to save the script when you close the window.

To save the script and close the window, follow this example:

```
ww = New Window( "Test", <<Script, "Open(\!"$SAMPLE_DATA/Big
  Class.jmp\!");");
ww << Save Text( "$TEMP/Test.jsl" );
ww << Close Window();
```

Technical Details

Write Tab Box and Tab Page Box Scripts

`Tab Page Box()` organizes the title and contents of the page in one display box. When `Tab Page Box()` is inside `Tab Box()`, a tabbed window with multiple tabbed pages appears. See "Tab Box and Tab Page Box".

In previous versions of JMP, display boxes such as `V List Box()` contained tab contents. Consider the following script:

```
win = New Window( "Tab Box",
   tb = Tab Box(
      "First page", // name of the tab
      V List Box( // tab contents
         Text Box( "first line of first page" ),
         Text Box( "second line of first page" )
      ),
      "Second page", // name of the tab
      V List Box( // tab contents
         Text Box( "first line of second page" ),
         Text Box( "second line of second page" )
      ),
```

```
    )
);
```

We recommend that `Tab Box()` contain a `Tab Page Box()` for each tab. The preceding example is rewritten as follows:

```
win = New Window( "Tab Box",
   tb = Tab Box(
      Tab Page Box( // tab contents
         Title( "First page of my tab box" ), // name of the tab
         Text Box( "first line of first page" ),
         Text Box( "second line of first page" )
      ),
      Tab Page Box( // tab contents
         Title( "Second page" ), // name of the tab
         Text Box( "first line of second page" ),
         Text Box( "second line of second page" )
      )
   )
);
```

Note: `Tab Page Box()` requires that the title be defined in a `Title()` function.

When a tab page box is created or dragged outside a tab box, the tab page box is a stand-alone container. The title appears in a shaded box at the top of the page, similar to the title in `Sheet Part()`; the title does not appear in an interactive tab. "Dashboard with One Row of Reports" provides an example.

Deprecated Tab Box and Tab Page Box Messages

Some messages previously used for `Tab Box()` are deprecated. Their counterparts are compatible with `Tab Page Box()`.

Table 11.5 Deprecated Tab Box() Messages

Deprecated Tab Box Message	Current Tab Page Box Message
`Set Title`	`Title`
`Set Tip`	`Tip`
`Set Icon`	`Icon`
`Set Closeable`	`Closeable`

See "Tab Box and Tab Page Box" for more information about the two display boxes.

Chapter 12

Scripting Graphs

Editing and Creating Two-Dimensional Graphs

The JSL script for a graph contains expressions that create shapes and lines, apply fill patterns, assign background colors, and so on. You can also right-click a graph and select Customize to add the same expressions.

This chapter also describes how to customize hover labels with rich text, images, and drill-down graphs.

For information about scripting three-dimensional graphs, see the chapter "Three-Dimensional Scenes".

Contents

Add Scripts to Graphs 659
Specify the Order of Graphical Elements 660
Create New Graphs from Scratch and Customize Them 666
Customize Graphs 667
Create Shapes for a Bubble Plot 669
How to Find Shapes 672
Add Lines, Arrows, Points, Shapes, and Text 673
Draw Lines 673
Draw Arrows 675
Draw Markers 677
Draw Pies and Arcs 678
Draw Regular Shapes: Circles, Rectangles, and Ovals 680
Draw Irregular Shapes: Polygons and Contours 682
Add Text 685
Specify Colors 686
Specify Transparency 690
Add Fill Patterns 691
Specify Line Types 694
Draw with Pixels 695
Interactive Graphs 696
Handle() 696
Mousetrap() 701
Drag Functions 703
Troubleshoot Interactive Graphs 706
Create a Color Picker 706
Clip Display Segs in Graphics 707
Create Background Maps 709
Graph Elements 710
Plot Functions 711
Get the Properties of a Graphics Frame 717
Add a Legend 717
Hover Labels 718
Add Rich Text to Hover Labels Using Textlets 719
Add Graphs or Images to Hover Labels Using Graphlets 722
Update Names and Values in Hover Labels Using Gridlets 732
Create Drill-Down Graphs 737
Work with the Hover Label Execution Context 740

Add Scripts to Graphs

When you right-click a graphics frame, you can enter or paste JSL commands as shown in the following example. The script usually contains drawing commands that run in the context of the graphics frame. The context of the graphics frame includes the data range of the axes and the order that the data and scripts are drawn in.

1. Select **Help > Sample Data Library** and open Big Class.jmp.
2. Select **Analyze** > **Fit Y by X**.
3. Choose weight for Y and height for X, and then click **OK**.
4. Right-click inside the graph and select **Customize**.

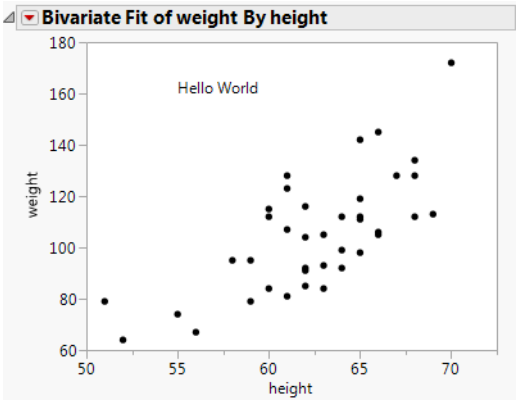
Figure 12.1 Customizing the Graph Options

5. Click the **Add** button ➕ to add a new graphics script.
6. Type the following text and click **OK**.

```
Text( {55, 160}, "Hello World" );
```

Now the graph has a text element at the graph's *x*-coordinate 55 and *y*-coordinate 160.

Figure 12.2 Adding a Script to a Graph Interactively



By default, the data points appear on top of graphics.

For example, add a new script in the Customize Graph window:

```
Fill Color( "Green" ); Rect( 57, 175, 65, 110, 1 );
```

A solid green rectangle appears. The list of scripts shows the order in which each script is drawn, so the first item on the list is drawn first.

- If you want the rectangle behind everything, move the rectangle script above the text script: select the rectangle script and click the up button .
- If you want the rectangle on top of everything, select the rectangle script and click the down button .

You can arrange all the scripts in a graph into the drawing order that you prefer. Any new script is initially added directly after the item in the list that is selected.

Tip: To use a script that references a column name, use `Column( colname )` or a colon (`:colname`) to scope it properly.

Hint: To see the JSL for the above actions, select **Save Script > To Script Window** from the red triangle menu.

Specify the Order of Graphical Elements

You can also add graphics elements using JSL instead of doing so interactively:

```
Frame Box <<Add Graphics Script(<order>, <"description">, script)
```

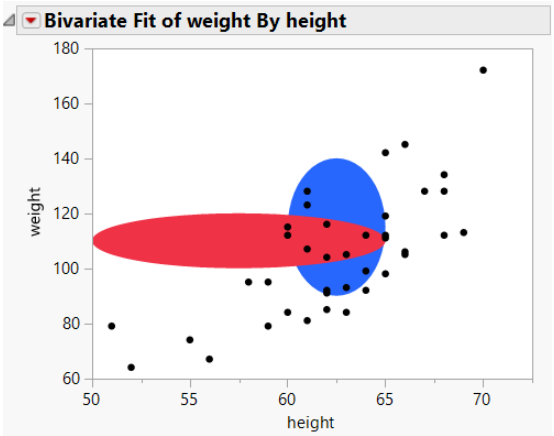
When you add graphical elements using JSL, they are drawn on top of whatever is already in the graph.

- The optional `order` argument specifies in what order to draw the graphics element. The order can be the keyword `Back` or `Front` *or* an integer that specifies the drawing order for a number of graphics elements. For example, if you add an oval to a scatterplot, the oval is drawn on top of the markers. The keyword `Back` or `1` cause the oval to be drawn last. `Front` or `2` means the object is drawn first.
- The optional `Description` argument is a string that appears in the Customize Graph window next to the graphics script.

To specify the drawing order for a number of graphics elements, use an integer for the `order` argument to determine where each is drawn in relation to the others. The following script first adds a blue oval and then adds a red oval over the blue oval. Both ovals are behind the points.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
op = dt << Bivariate(
  X( :height ),
  Y( :weight ),
);

Report( op )[Frame Box( 1 )] << Add Graphics Script(
  1, // draw the blue oval
  Description( "Blue Oval" ),
  Fill Color( "Blue" );
  Oval( 60, 140, 65, 90, 1 );
);

Report( op )[Frame Box( 1 )] << Add Graphics Script(
  2, // draw the red oval over the blue oval
  Description( "Red Oval" ),
  Fill Color( "Red" );
  Oval( 50, 120, 65, 100, 1 );
);
```

Figure 12.3 Specifying the Drawing Order



Copy and Paste Frame Contents or Settings

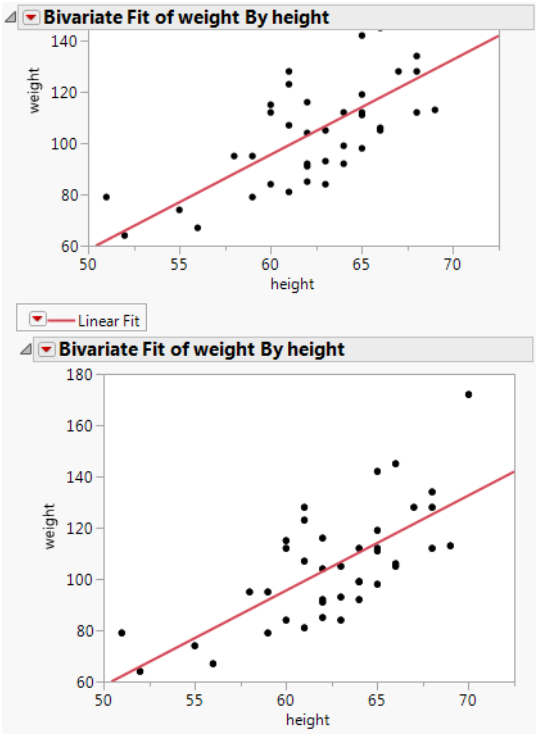
You can copy and paste the contents or the settings of a frame with these JSL commands:

```
obj << Copy Frame Contents // contents such as a graph or line of fit
obj << Paste Frame Contents
obj << Copy Frame Settings // settings such as background color
obj << Paste Frame Settings
```

The following example creates two Bivariate graphs, adds a line of fit to the first graph, and then copies and pastes the line to the second graph.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( y( :weight ), x( :height ) );
rbiv1 = biv << Report; // create a report object
rbiv2 = rbiv1 << Clone Box; // create a copy of the Bivariate graph

/* place the cloned box below the Bivariate graph
to create a second Bivariate graph */
rbiv1 << Append( rbiv2 );

framebox1 = rbiv1[Frame Box( 1 )]; /* assign references to
the reports' frame boxes*/
framebox2 = rbiv2[Frame Box( 1 )];
biv << Fit Line; // fit a line to the first Bivariate graph

// copy the frame contents from framebox1
framebox1 << Copy Frame Contents;

// paste the frame contents to framebox2
framebox2 << Paste Frame Contents;
```

Figure 12.4 Copying and Pasting Frame Contents

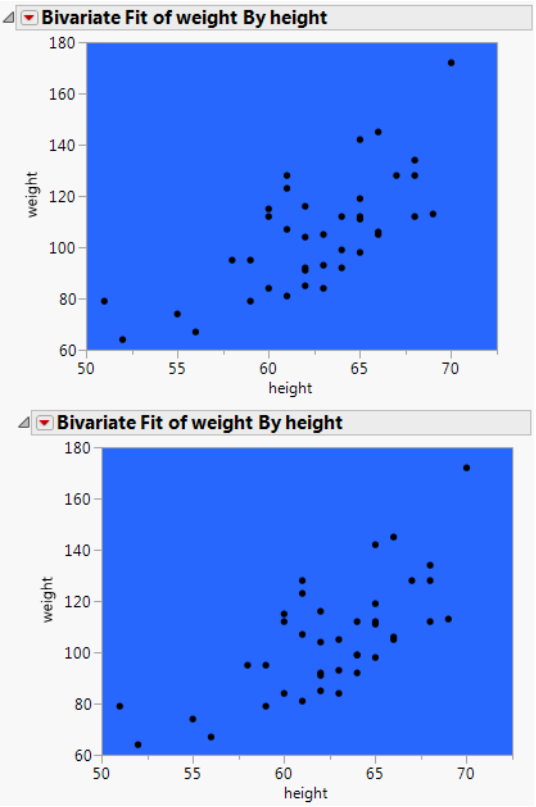


The following example creates two Bivariate graphs, sets the background of the first graph to blue, and then copies and pastes the blue background setting to the second graph:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

biv1 = dt << Bivariate( y( :weight ), x( :height ) );

// create the first Bivariate graph
rbiv1 = biv1 << Report; // create a report object
rbiv2 = rbiv1 << Clone Box; // create a copy of the Bivariate graph

/* place the cloned box below the Bivariate graph
to create a second Bivariate graph */
rbiv1 << Append( rbiv2 );

framebox1 = rbiv1[Frame Box( 1 )]; /* assign a reference to the
                                      reports' frame boxes */
framebox2 = rbiv2[Frame Box( 1 )];

// assign the background color to framebox1
framebox1 << Background Color( "Blue" );
```

```
// copy the frame settings from framebox1
framebox1 << Copy Frame Settings;

// paste the frame settings to framebox2
framebox2 << Paste Frame Settings;
```

Figure 12.5 Copying and Pasting the Background



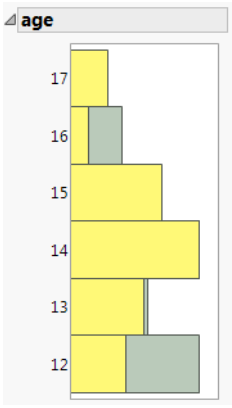
Copy and Paste Histograms

The following example shows how to create two histograms and copy the second histogram into the first histogram. A platform will not save a pasted-in segment, so you'll need to run a script like this to reproduce the graph.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dist = dt << Distribution(
   SendToByGroup( {:sex == "F"} ),
   Nominal Distribution( Column( :age ) ),
   Histograms Only,
   By( :sex ),
```

```
   SendToByGroup(
      {:sex == "M"},
      SendToReport(
         Dispatch(
            {"Distributions sex=M", "age"},
            "Distrib Nom Hist",
            FrameBox,
            {DispatchSeg(
               Hist Seg( 1 ),

               // set the histogram color for males
               Fill Color( "Light Yellow" ),
            )}
         )
      )
   )
);

For( i = 2, i <= N Items( dist ), i++,
   Report( dist[i] )[FrameBox( 1 )] << Copy Frame Contents;

   // copy the the second histogram into the first histogram
   Report( dist[1] )[FrameBox( 1 )] << Paste Frame Contents;
);

New Window( "Distribution", Outline Box( "age", Report( dist[1] )[Picture Box(
   1 )] ) );
```

Figure 12.6 Pasted Histogram



Interactively Copy and Paste Frame Contents and Settings

To interactively copy and paste the frame settings, follow these steps:

1. Right-click the graph and select **Edit > Copy Frame Settings**.
2. Right-click the frame to which you want to paste the settings and select **Edit > Paste Frame Settings**.

To interactively copy and paste the frame contents, follow these steps:

1. Right-click the graph and select **Edit > Copy Frame Contents**.
2. Right-click the frame to which you want to paste the contents and select **Edit > Paste Frame Contents**.

Create New Graphs from Scratch and Customize Them

Graphics scripts are set up inside the `Graph Box()` command within a `New Window()` command.

```
New Window("title", <arguments>, Graph Box( named arguments,..., script));
```

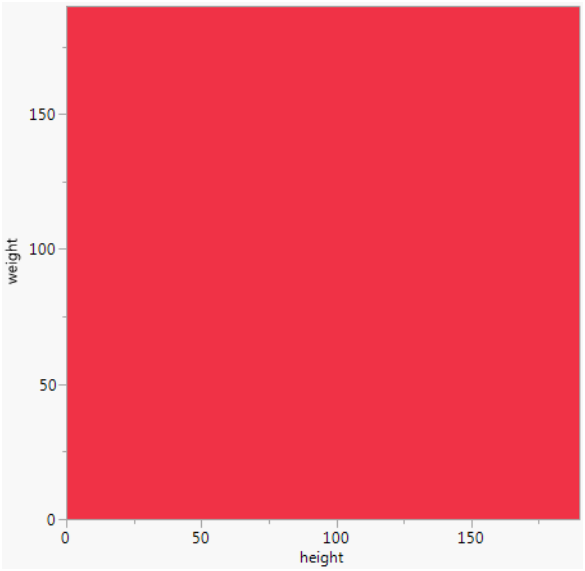
The following arguments are named arguments for `Graph Box()`:

```
Frame Size( horizontal, vertical ), // size in pixels
X Scale( xmin, xmax ), Y Scale( ymin, ymax ), // range of x and y axes
X Name( "x" ), Y Name( "y" ), // names for x and y axes
Suppress Axes // omit the axes
```

For example, `Graph Box()` accepts named arguments as shown in the following example, which creates a graph with a red background.

```
win = New Window( "Named arguments",
   Graph Box(
      Frame Size( 300, 300 ), // construct the graph box
      X Scale( 0, 190 ), // set the x-axis range
      Y Scale( 0, 190), // set the y-axis range
      X Name( "height" ),
      Y Name( "weight" ),
      <<Background Color( "Red" ) // set the background color
   )
);
```

Figure 12.7 Creating a Graph



Alternatively, you can use the send << operator to send commands to the `Graph Box` instead of using the named arguments. This example also creates a graph with a red background.

```
win = New Window( "Messages",
   Graph Box(
      <<Frame Size( 300, 300 ),
      <<XAxis( 0, 190 ),
      <<YAxis( 0, 190 ),
      <<Background Color( "Red" )
   )
);
```

Customize Graphs

You can also make changes to graphs in a script. For example, create a window with a graph, get a reference to the report, and then set the frame box size.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( :weight ), X( :height ), Fit Line );
rbiv = biv << Report;
rbiv[Frame Box( 1 )] << Frame Size( 400, 400 );
```

To see a list of possible messages for any given display box object, select **Help > Scripting Index** and select **Display Boxes** from the list. Running the `Show Properties()` command is an alternative to using the Scripting Index. For example, here is a partial list of messages that you can send to an axis:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( Y( :weight ), X( :height ), Fit Line );
rbiv = biv << Report;
Show Properties( rbiv[Axis Box( 1 )] );
   Axis Settings [Action](Provides a way to bring up the axis dialog or set
      configurable axis options for a given axis.)
   Revert Axis [Action](Restore the settings that this axis had originally.)
   Add Axis Label [Action]
   Remove Axis Label [Action]
   ...
```

The following expressions change the font of both axis labels (weight and height in the example above) to 12-point, italic Arial Black:

```
rbiv[Text Edit Box( 1 )] << Set Font( "Arial Black", 12, Italic );
rbiv[Text Edit Box( 2 )] << Set Font( "Arial Black", 12, Italic );
```

or

```
rbiv[Text Edit Box( 1 )] << Set Font( "Arial Black" );
rbiv[Text Edit Box( 1 )] << Set Font Style( "Italic" );
rbiv[Text Edit Box( 1 )] << Set Font Size( 12 );
rbiv[Text Edit Box( 1 )] << Set Font( "Arial Black" );
rbiv[Text Edit Box( 2 )] << Set Font Style( "Italic" );
rbiv[Text Edit Box( 2 )] << Set Font Size( 12 );
rbiv[Text Edit Box( 2 )] << Set Font Name( "Arial Black" ) << Set Font Size(
  12 ) << Set Font Style( Italic )
```

The default font specified in the Preferences is applied if the font is not installed on the computer.

Specifying a Missing Color Value

When you specify a color theme in a script, you can specify the color of a missing value. Use the `Missing` argument. In this example, the missing color is yellow. Refer to an online RGB value table for the color values.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Graph Builder(
   Size( 528, 450 ),
   Show Control Panel( 0 ),
   Variables(
      X( :height ),
      Y( :weight ),
      Color( Transform Column( "wc", Formula( Root( :weight - 80 ) ) ) ) )
```

```
    ),
    Elements( Points( X, Y, Legend( 8 ) ) ),
    SendToReport(
       Dispatch( {}, "400", ScaleBox,
          {Legend Model( 8,
             Properties( 0, {gradient(
                {Color Theme(
                   {"Blue to Gray to Red Copy", 4099, {{42, 63, 255}, {192,
                   192, 192}, {252, 11, 11}, Missing( {255, 211, 0} )}}
                )}
             )},
             Item ID( "wc", 1 )
             )
          )}
       )
    )
);
```

Create Shapes for a Bubble Plot

Bubble Plot supports predefined shapes and custom shapes. Shapes can give visual cues about the data or provide a more interesting shape than a circle. For example, you can show arrows (Figure 12.8) or a hurricane symbol on a map (Figure 12.9).

Specify a Predefined Shape

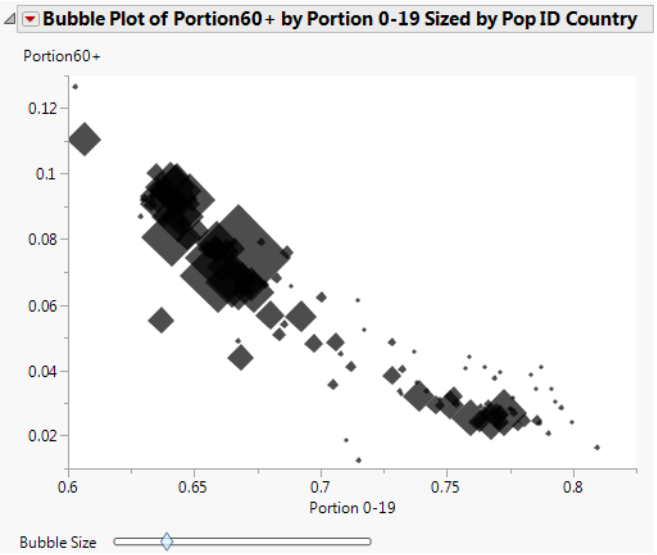
The `Set Shape` message specifies the predefined shape:

```
bp << Set Shape( "Circle"|"Triangle"|"Square"|"Diamond"|"Arrow" );
```

The following example creates the bubble plot and diamond shape shown in Figure 12.8:

```
dt = Open( "$SAMPLE_DATA/PopAgeGroup.jmp" );
bp = dt << Bubble Plot(
   X( :"Portion 0-19"n ),
   Y( :"Portion60+"n),
   Sizes( :Pop ),
   ID( :Country )
);
bp << Set Shape( "Diamond" );
```

Figure 12.8 Diamond Shapes



Create a Custom Shape

You can create custom shapes only in JSL through a combination of two messages:

```
bp << Set Custom Path( pathMatrix | pathText );
bp << Set Shape( "Custom" );
```

- In `Set Custom Path()`, the path can consist of an *N*x3 matrix or a string that contains SVG code. The default value is 0.

  ```
  bp << Set Custom Path(
     "M-1,-1 L-1,1 L0,0.5 L1,1 L1,-1 L0,-0.5 L-1,-1 Z, ..." );
  ```

- In `Set Shape()`, specify `"Custom"`.

  ```
  bp << Set Shape( "Custom" );
  ```

The following script creates the map and custom hurricane shape shown in Figure 12.9.

```
dt = Open( "$SAMPLE_DATA/Hurricanes.jmp" );
dt << Bubble Plot(
   X( :Longitude ),
   Y( :Latitude ),
   Sizes( "Wind (Knots)"n ),
   Time( :Date ),
   Coloring( :Landfall in USA ),
   ID( :Name and ID ),
   Speed( 220 ),
   Bubble Size( 51.43 ),
   Time Index( 729.9 ),
```

```
   Trail Bubbles( 1 ),
   All Labels( 0 ),
   No Labels( 0 ),
/* rotate shapes in the direction of the movement
   good for animated plots */
   Orient Shapes( 1 ),
   Set Custom Path( // specify the SVG string
      "M-0.99999898,0.00921400000000006 C-0.99999898,-0.51266322
   -0.55204076,-0.93622284 -0.000119060000000104,-0.93622284
   C0.1835362,-0.93622284 0.3556076,-0.88949112 0.5035232,-0.80766502
   C0.3236248,-1.36413554 0.0260398000000001,-1.815694896 -0.3555946,-2
   C0.4067376,-1.7567781 0.9372286,-0.91116206 1.0000004,0.00829720000000009
   C1.0000004,0.5301742 0.5518026,0.9548002 -0.00011905999999982,0.9547998
   C-0.18139054,0.9547998 -0.35150496,0.9090838 -0.49816916,0.8292164
   C-0.31759686,1.3771358 -0.0174124399999999,1.8176124 0.3602494,1.999999
   C-0.38991788,1.7606584 -0.92375834,0.908779 -0.99999898,0.00921400000000034
   z"
   ),
   Set Shape( "Custom" ), // set the custom shape
   Title Position( -88.679, 59.73 ),
   SendToReport(
      Dispatch(
         {},
         "Bubble Plot",
         FrameBox,
         {Frame Size( 743, 603 ), Background Map( Images( "Simple Earth" ) ),
         Grid Line Order( 3 ), Reference Line Order( 2 )}
      )
   )
);
```

Figure 12.9 Custom Hurricane Shape

How to Find Shapes

Search Wikimedia Commons or Google Images and include the keyword *SVG*. Look for SVG examples that are composed of stroke paths. SVG can be used to represent very complex drawings, with gradients and other elements. Only a path representation, which provides both outline and fill modes, is supported.

You might need to translate and scale the SVG that you find in order to get correct coordinates. Inkscape or similar tools can be useful for doing this manipulation. To copy the path from Inkscape, select **Edit > XML Editor**, find the `svg:path` element, and copy the path string from the `d Attribute` into `Set Custom Path()`.

Add Lines, Arrows, Points, Shapes, and Text

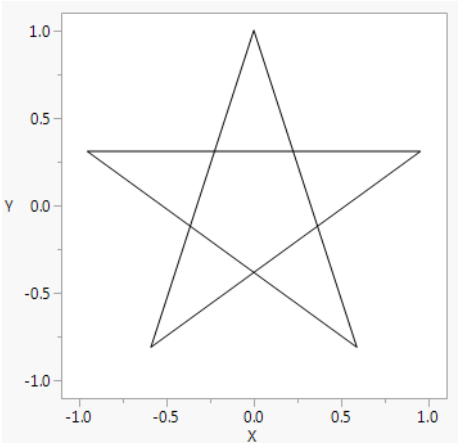
- "Draw Lines"
- "Draw Arrows"
- "Draw Markers"
- "Draw Pies and Arcs"
- "Draw Regular Shapes: Circles, Rectangles, and Ovals"
- "Draw Irregular Shapes: Polygons and Contours"
- "Add Text"
- "Specify Colors"
- "Specify Transparency"
- "Add Fill Patterns"
- "Specify Line Types"
- "Draw with Pixels"

Draw Lines

`Line()` draws lines between points.

```
win = New Window( "Five-Point Star",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( -1.1, 1.1 ),
      Y Scale( -1.1, 1.1 ),
      Line(
         {Cos( 1 * Pi() / 10 ), Sin( 1 * Pi() / 10 )},
         {Cos( 9 * Pi() / 10 ), Sin( 9 * Pi() / 10 )},
         {Cos( 17 * Pi() / 10 ), Sin( 17 * Pi() / 10 )},
         {Cos( 5 * Pi() / 10 ), Sin( 5 * Pi() / 10 )},
         {Cos( 13 * Pi() / 10 ), Sin( 13 * Pi() / 10 )},
         {Cos( 1 * Pi() / 10 ), Sin( 1 * Pi() / 10 )}
      )
   )
);
```

Figure 12.10 Using Lines to Draw a Star



You can either specify the points in two-item lists as demonstrated above or as matrices of *x* and then *y* coordinates. Matrices are flattened by rows. You can use either row or column vectors, as long as you have the same number of elements in each matrix. The following expressions have the same effect:

```
Line( {1,2}, {3,0}, {2,4} ); // several {x,y} lists
Line( [1 3 2],[2 0 4] ); // row vectors
Line( [1,3,2], [2,0,4] ); // column vectors
Line( [1 3 2], [2,0,4] ); // one of each
```

The star example could also be drawn this way. Note that the script must include the full `Matrix( {...} )` notation rather than `[ ]` shorthand because the entries are expressions. The following example uses the `Matrix()` function.

```
win = New Window( "Five-Point Star",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( -1.1, 1.1 ),
      Y Scale( -1.1, 1.1 ),
      Line(
         Matrix(
            { // the x coordinates
            Cos( 1 * Pi() / 10 ), Cos( 9 * Pi() / 10 ), Cos( 17 * Pi() / 10 ),
            Cos( 5 * Pi() / 10 ), Cos( 13 * Pi() / 10 ), Cos( 1 * Pi() / 10 )}
         ),
         Matrix(
            { // the y coordinates
            Sin( 1 * Pi() / 10 ), Sin( 9 * Pi() / 10 ), Sin( 17 * Pi() / 10 ),
            Sin( 5 * Pi() / 10 ), Sin( 13 * Pi() / 10 ), Sin( 1 * Pi() / 10 )}
         )
      ) ) );
```

`HLine()` draws a horizontal line across the graph at the *y*-value that you specify. Similarly, `VLine()` draws a vertical line down the graph at the *x*-value that you specify. Both functions support drawing multiple lines by using a matrix of values in the *y* argument. See "Mousetrap()".

Draw Arrows

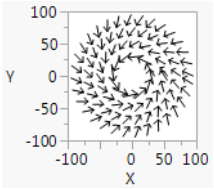
`Arrow()` draws an arrow from the first point to the second point. The default arrowhead is scaled to 1 plus the square root of the length of the arrow. To set the length of the arrowhead, add an optional first argument, specifying the length of the arrowhead in pixels. The following example draws simple arrows.

```
win = New Window( "Arrowheads",
   Graph Box(
      Pen Size( 4 );

      /* matrix 1 defines the x coordinates of an arrow
      matrix 2 defines the y coordinates of an arrow */
      Arrow( 20, [10 30 90], [88 22 44] );
   )
);
```

The following example draws arrows in a circular pattern.

```
win = New Window( "Hurricane",
   Graph Box(
      Frame Size( 100, 100 ),
      X Scale( -100, 100 ),
      Y Scale( -100, 100 ),
      For( r = 35, r < 100, r += 20,
         ainc = 2 * Pi() * 3 / r;
         For( a = 0, a < 2 * Pi(), a += ainc,
            x = r * Cosine( a );
            y = r * Sine( a );
            aa = a + ainc * 45 / r;
            rr = r - r / 6;
            x2 = rr * Cosine( aa );
            y2 = rr * Sine( aa );

            /* list 1 defines the start point for the arrow
            list 2 defines the end point for the arrow */
            Arrow( {x, y}, {x2, y2} );
         );
      )
   )
);
```

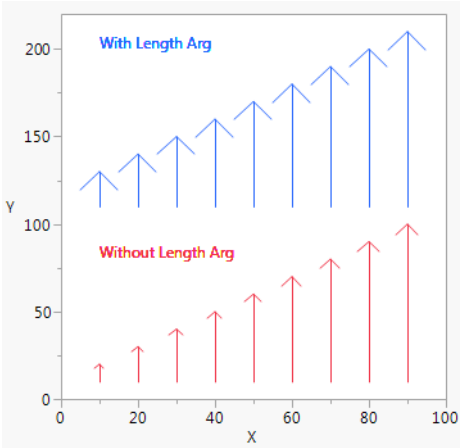
Figure 12.11 Drawing Arrows



The following example compares drawing with a specified length (19 pixels) and drawing with the default arrow head size.

```
win = New Window( "Arrowheads",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( 0, 100 ),
      Y Scale( 0, 220 ),
      x = 10;
      y1 = 10;
      y2 = y1 + 10;
      For( i = 1, i < 10, i++,
         Pen Color( "Red" );
         Arrow( {x, y1}, {x, y2} );

         y2 += 10;
         y1 += 100;
         y2 += 100;
         Pen Color( "Blue" );
         Arrow( 20, {x, y1}, {x, y2} );

         x += 10;
         y1 -= 100;
         y2 -= 100;

         Text Color( "Red" );
         Text( {10, 80}, "Without Length Arg" );

         Text Color( "Blue" );
         Text( {10, 200}, "With Length Arg" );
      );
   )
);
```

Figure 12.12 Arrowhead Sizes



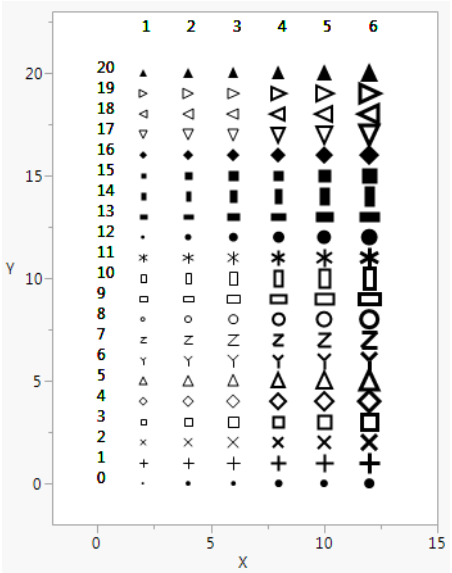
As with `Line()`, you can either specify the points in two-item lists as demonstrated above or as matrices of *x* and then *y* coordinates.

Draw Markers

`Marker()` draws a marker of the type that you specify (1–20) in the first argument at the point that you specify in the second argument. `Marker Size()` scales markers from 0 to 6 (dot to XXXL). To set markers to the size that is specified in the Preferences, use a value of `-1`.

```
ymax = 20;
win = New Window( "Markers",
   Graph Box(
      Frame Size( 300, 400 ),
      X Scale( -2, ymax - 5 ),
      Y Scale( -2, ymax + 3 ),
      For( j = 1, j < 7, j++,
         Marker Size( j );
         For( i = 0, i < (ymax + 1), i++,
            Marker( i, {j * 2, i} );
            Text( {0, i}, i );
            Text( {j * 2, ymax + 2}, j );
         );
      )
   )
);
```

Figure 12.13 Drawing Markers



You can also include a row state argument before, after, or instead of the marker ID argument. By using `Combine States()`, you can set multiple row states inside `Marker()`. Try substituting each of the following lines in the preceding script:

```
Marker( i, Color State( i ), {j * 2, i} );
Marker( Color State( i ), i, {j * 2, i} );
Marker(
   Combine States( Color State( i ), Marker State( i ), Hidden State( i ) ),
   {j * 2, i}
);
```

As with `Line()` and `Arrow()`, you can also specify points as matrices of *x* and then *y* coordinates.

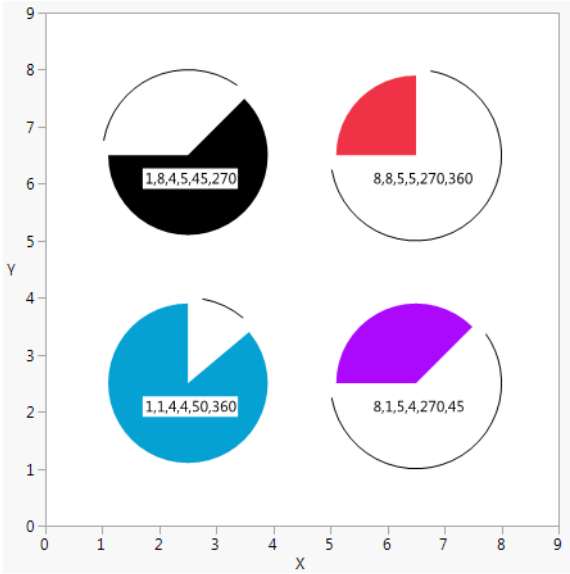
Draw Pies and Arcs

`Pie()` and `Arc()` draw wedges and arc segments. The first four arguments are *x1, y1, x2,* and *y2,* the coordinates of the rectangle to inscribe. The last two arguments are the starting and ending angle in degrees, where 0 degrees is 12 o'clock and the arc or slice is drawn clockwise from start to finish.

```
win = New Window( "Pies and Arcs",
   Graph Box(
      Frame Size( 400, 400 ),
      X Scale( 0, 9 ),
      Y Scale( 0, 9 ),
```

```
        Fill Color( "Black" ), // top left
        Pie( 1.1, 7.9, 3.9, 5.1, 45, 270 ),
        Text( Erased, {1.75, 6}, "1,8,4,5,45,270" ),
        Arc( 1, 8, 4, 5, 280, 35 ),
        Fill Color( "Red" ), // top right
        Pie( 7.9, 7.9, 5.1, 5.1, 270, 360 ),
        Text( Erased, {5.75, 6}, "8,8,5,5,270,360" ),
        Arc( 8, 8, 5, 5, 370, 260 ),
        Fill Color( "BlueCyan" ), // bottom left
        Pie( 1.1, 1.1, 3.9, 3.9, 50, 360 ),
        Text( Erased, {1.75, 2}, "1,1,4,4,50,360" ),
        Arc( 1, 1, 4, 4, 370, 40 ),
        Fill Color( "Purple" ), // bottom right
        Pie( 7.9, 1.1, 5.1, 3.9, 270, 45 ),
        Text( Erased, {5.75, 2}, "8,1,5,4,270,45" ),
        Arc( 8, 1, 5, 4, 55, 260 )
    )
);
```

Figure 12.14 Drawing Pies and Arcs



Draw Regular Shapes: Circles, Rectangles, and Ovals

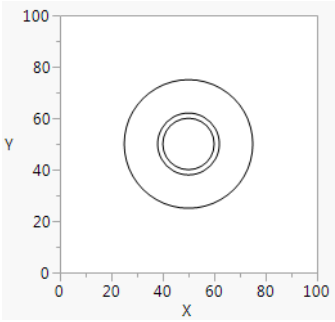
Circles

`Circle()` draws a circle with the center point and radius given. Subsequent arguments specify additional radii.

```
win = New Window( "Circles",
   Graph Box( Frame Size( 200, 200 ),
      Circle( {50, 50}, 10, 12, 25 )
   )
);
```

`Circle()` also has a final optional argument, `"fill"`. This string indicates that all circles defined in the function are filled with the current fill color. If `"fill"` is omitted, the circle is empty.

Figure 12.15 Drawing Circles



Note that a circle is always a circle, even if you resize the graph to a different aspect ratio. If you want your circle to change aspect ratios (in other words, cease being a circle) when the graph is resized, use an oval instead.

If you do not want your circle to resize if the graph is resized, specify the radius in pixels instead:

```
win = New Window( "Circles",
   Graph Box(
      Frame Size( 200, 200 ),
      Circle( {50, 50}, Pixel Radius( 10 ), Pixel Radius( 12 ), Pixel Radius(
   25 ) )
   )
);
```

Rectangles

`Rect()` draws a rectangle from the diagonal coordinates you specify. The coordinates can be specified either as four arguments in order (*left*, *top*, *right*, *bottom*), or as a pair of lists ({*left*, *top*}, {*right*, *bottom*}).

```
win = New Window( "Rectangles",
   Graph Box(
      Frame Size( 200, 200 ),
      Pen Color( 1 );
      Rect( 0, 40, 60, 0 );
      Pen Color( 3 );
      Rect( 10, 60, 70, 10 );
      Pen Color( 4 );
      Rect( 50, 90, 90, 50 );
      Pen Color( 5 );
      Rect( 0, 80, 70, 70 );
   )
);
```

`Rect()` has an optional fifth argument, *fill*. Specify 0 to get an unfilled rectangle and 1 to get a filled rectangle. The default value for *fill* is 0. The rectangle is filled with the current `fill color`.

Any negative fill argument produces an unfilled frame inset by one pixel:

```
win = New Window( "Framed Rectangle",
   Graph Box( Frame Size( 200, 200 ), Rect( 0, 40, 60, 0, -1 ) )
);
```

Ovals

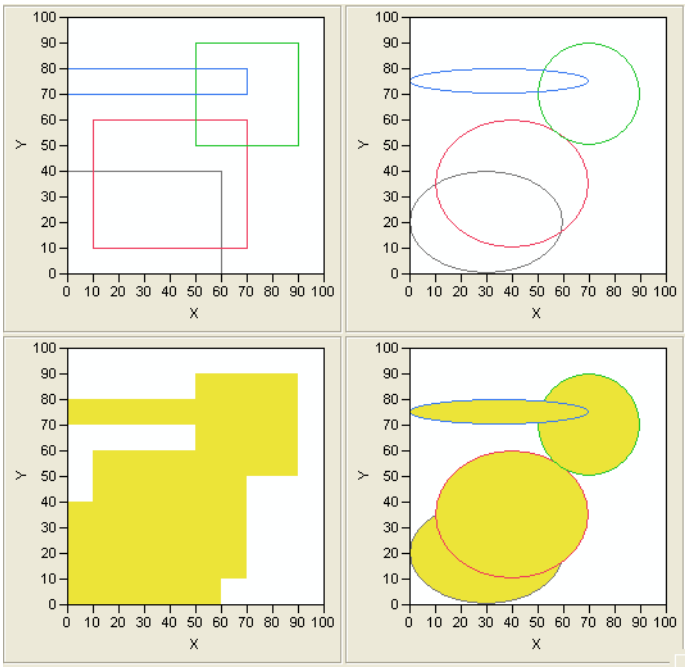
`Oval()` draws an oval inside the rectangle given by its *x1*, *y1*, *x2*, and *y2* arguments:

```
win = New Window( "Ovals",
   Graph Box(
      Frame Size( 200, 200 ),
      Pen Color( 1 );
      Oval( 0, 40, 60, 0 );
      Pen Color( 3 );
      Oval( 10, 60, 70, 10 );
      Pen Color( 4 );
      Oval( 50, 90, 90, 50 );
      Pen Color( 5 );
      Oval( 0, 80, 70, 70 );
   )
);
```

`Oval()` also has an optional fifth argument, *`fill`*. Specify 0 to get an unfilled rectangle and 1 to get a filled oval. The default value for *`fill`* is 0. The oval is filled with the current `fill color`.

Figure 12.16 shows rectangles and ovals, drawn both filled and unfilled. Notice that filled rectangles do not have outlines. Ovals do have outlines. If you want a filled rectangle with an outline, you must draw the filled rectangle and then draw an unfilled rectangle with the same coordinates.

Figure 12.16 Rectangles and Ovals, Unfilled and Filled



Draw Irregular Shapes: Polygons and Contours

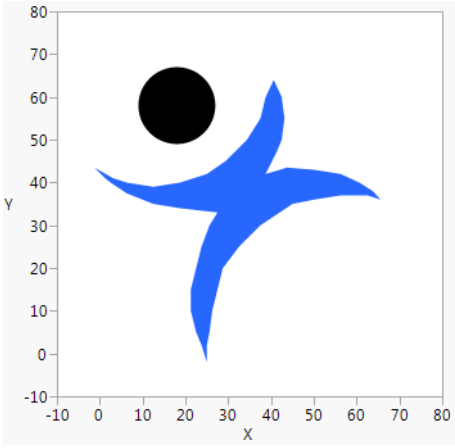
Polygons

`Polygon()` works similarly to `Line()`. They both connect points, but `Polygon()` returns to the first point to close the polygon and filling the resulting area. You can specify the points as individual points in two-item lists (as shown in "Draw Markers") or as matrices of *x* and then *y* coordinates. Matrices are flattened by rows, so you can use either row or column vectors, as long as you have the same number of elements in each matrix. First set up the matrices of points and then call them inside `Polygon()`.

```
gCoordX = [25, 23.75, 22.5, 21.25, 21.25, 22.5, 23.75, 25.625, 27.5, 18.75,
  12.5, 6.25, 2.5,
```

```
1.25, -1.25, 3.125, 6.25, 12.5, 18.75, 25, 29.375, 34.375, 37.5, 38.75,
  40.625, 42.5, 43.125,
42.5, 41.25, 38.75, 43.75, 50, 56.25, 60.625, 63.75, 65.625, 62.5, 56.25, 50,
  45, 37.5, 32.5,
28.75, 27.5, 26.25, 25.625, 25];
gCoordY = [-2, 2, 5, 10, 15, 20, 25, 30, 33, 34, 35, 37.5, 40, 41, 43.5, 41,
  40, 39, 40, 42, 45,
50, 55, 60, 64, 60, 55, 50, 47, 42, 43.5, 43, 42, 40, 38, 36, 37, 37, 36, 35,
  30, 25, 20, 15, 10, 5, 2];
win = New Window( "The JMP Man",
  Graph Box(
     Frame Size( 300, 300 ),
     X Scale( -10, 80 ),
     Y Scale( -10, 80 ),
     Pen Color( "Black" );
     Fill Color( "Blue" );
     Polygon( gCoordX, gCoordY );
     Fill Color( "Black" );
     Circle( {18, 58}, 9, "FILL" );
  )
);
```

Figure 12.17 Drawing a Polygon



A related command, `In Polygon()`, tells whether a given point falls inside the specified polygon. This code checks some points from the JMP man pictured in Figure 12.17:

```
In Polygon( 0,60, GcoordX,GCoordY );  // return 0
In Polygon( 30,38, GcoordX,GCoordY ); // return 1
```

You can add `In Polygon()` to the JMP man script. Run the following script, and then click various locations in the picture and watch the log window. "In" and "out" are printed to the log when you click the body.

```
gCoordX = [25, 23.75, 22.5, 21.25, 21.25, 22.5, 23.75, 25.625, 27.5, 18.75,
   12.5, 6.25, 2.5, 1.25, -1.25, 3.125, 6.25, 12.5, 18.75, 25, 29.375, 34.375,
   37.5, 38.75, 40.625, 42.5, 43.125, 42.5, 41.25, 38.75, 43.75, 50, 56.25,
   60.625, 63.75, 65.625, 62.5, 56.25, 50, 45, 37.5, 32.5, 28.75, 27.5, 26.25,
   25.625, 25];
gCoordY = [-2, 2, 5, 10, 15, 20, 25, 30, 33, 34, 35, 37.5, 40, 41, 43.5, 41,
   40, 39, 40, 42, 45, 50, 55, 60, 64, 60, 55, 50, 47, 42, 43.5, 43, 42, 40,
   38, 36, 37, 37, 36, 35, 30, 25, 20, 15, 10, 5, 2];
win = New Window( "The JMP Man",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( -10, 80 ),
      Y Scale( -10, 80 ),
      Pen Color( "Black" );
      Fill Color( "Black" );
      Polygon( gCoordX, gCoordY );
      Mousetrap(
         {},
         Print( If( In Polygon( x, y, gCoordX, gCoordY ), "in", "out" ) )
      );
   )
);
```

Contours

`Contour()` draws contour lines using a grid of coordinates.

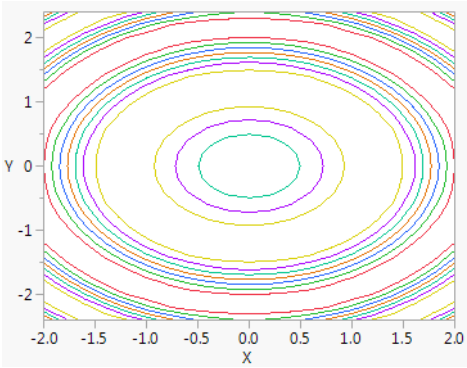
```
Contour( xVector, yVector, zGridMatrix, zContour, <zColors> );
```

Suppose that you have an **n** by **m** matrix *zGridMatrix* of values on some surface (defined across the *n* values of *xVector* by the **m** values of *yVector*). `Contour()` draws the contour lines that are defined by the values in *zContour* in the colors defined by *zColors*. Here is an example:

```
x = (-10 :: 10) / 5;
y = (-12 :: 12) / 5;
grid = J( 21, 25, 0 );
z = [-.75, -.5, -.25, 0, .25, .5, .75];
zcolor = [3, 4, 5, 6, 7, 8, 9];
For( i = 1, i <= 21, i++,
   For( j = 1, j <= 25, j++,
      grid[i, j] = Sin( (x[i]) ^ 2 + (y[j]) ^ 2 )
   )
);
```

```
Show( grid );
win = New Window( "Hat",
   Graph Box(
      X Scale( -2, 2 ),
      Y Scale( -2.4, 2.4 ),
      Contour( x, y, grid, z, zcolor )
   )
);
```

Figure 12.18 Drawing Contour Lines



Add Text

You can use `Text()` to draw text at a given location.

```
Text( <properties>, ( {x, y}|{left, bottom, right, top} ), "text" );
```

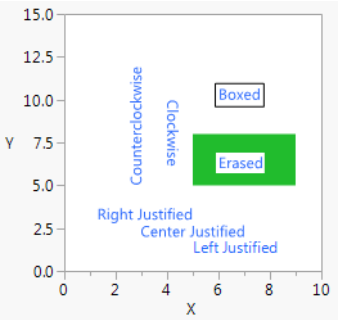
The point and text can be in any order and repeated. You can precede the point and text with an optional first argument: `Center Justified`, `Right Justified`, `Erased`, `Boxed`, `Counterclockwise`, or `Clockwise`. `Erased` is for removing whatever would otherwise obscure the text in a graph. It paints a background-colored rectangle behind the text.

In the following example, notice how the erased text appears inside a white box over the green `Rect()`.

```
mytext = New Window( "Adding Text",
   Graph Box(
      Frame Size( 200, 200 ),
      Y Scale( 0, 15 ),
      X Scale( 0, 10 ),
      Text Size( 9 );
      Text Color( "blue" );
      Text( {5, 1}, "Left Justified" );
      Text( Center Justified, {5, 2}, "Center Justified" );
      Text( Right Justified, {5, 3}, "Right Justified" );
```

```
        Fill Color( 4 );
        Rect( 5, 8, 9, 5, 1 );
        Text( Erased, {6, 6}, "Erased" );
        Text( Boxed, {6, 10}, "Boxed" );
        Text( Clockwise, {4, 10}, "Clockwise" );
        Text( Counterclockwise, {3, 5}, "Counterclockwise" );
    )
);
```

Figure 12.19 Drawing Text in a Graph Box



There is a variant of the `Text()` function that draws a string inside the rectangle that is specified by four coordinates. Here is the syntax:

```
Text( {left, top, right, bottom}, string);
```

Specify Colors

The following commands control colors:

- `Fill Color()` for solid areas
- `Level Color()` for categorical data
- `Pen Color()` for lines and points
- `Back Color()` for the background of text (similar to the box around the erased text in Figure 12.19)
- `Background Color()` for the graph's background color
- `Font Color()` for added text

Fill colors overwrite pen colors for drawn shapes. You do not get both, as in some drawing packages. To get both a fill and a pen line, draw two shapes, one with fill and one without. You can select a color with a single numeric argument, a color name in quotation marks, or an RGB value. The standard colors are specified with numbers 0–15 (both 0 and 15 are black) or by their names (Table 12.1).

Tip: For more shades, add the word Dark, Medium Dark, Medium Light, or Light before a color name in quotation marks.

Table 12.1 Standard JMP Colors

Number	Name
0	Black
1	Gray
2	White
3	Red
4	Green
5	Blue
6	Orange
7	BlueGreen
8	Purple
9	Yellow
10	Cyan
11	Magenta
12	YellowGreen
13	BlueCyan
14	Fuchsia

The following script creates a graph that shows all JMP colors:

```
Text Color( 0 );
New Window( "Colors",
  Graph Box(
     FrameSize( 640, 400 ),
     Y Scale( -1, 17 ),
     X Scale( -3, 12 ),
     k = 0;
     For( jj = 1, jj <= 12, jj += 2,
        l = 15;
```

```
            For( i = 0, i <= 15 & k < 85, i++,
               thiscolor = Color To RGB( k );
               Fill Color( k );
               thisfill = 1;
               If( thiscolor == {1, 1, 1},
                  Pen Color( 0 );
                  thisfill = 0;
               ,
                  Pen Color( k )
               );
               Rect( jj, l + .5, jj + .5, l, thisfill );
               Text( {jj - 1, l}, "color ", k );
               k++;
               l--;
            );
         );
         jj = -2;
         color = {"Black", "Gray", "White", "Red", "Green", "Blue", "Orange",
      "BlueGreen",
         "Purple", "Yellow", "Cyan", "Magenta", "YellowGreen", "BlueCyan",
      "Fuschia", "Black"};
         For(
            i = 0;
            l = 15;, i <= 15 & l >= 0,
            i++;
            l--;,
            Text( {jj, l}, color[i + 1] )
         );
      ) );
```

Larger numbers cycle through shading variations of the same color sequence. A script demonstrating this appears under "Colors and Markers". Values outside the range 0–84 are not accepted.

Table 12.2 How Numbers Map to Colors

Number	Result
16–31	Medium Dark
32–47	Medium Light
48–63	Dark
64–79	Light

Table 12.2 How Numbers Map to Colors *(Continued)*

Number	Result
80–84	Light, Medium Light, Gray, Medium Dark, Dark

If you prefer to use RGB values, create a list with the fraction for each color in red, green, and blue order.

```
Pen Color( {.38,.84,.67} ); // teal
```

`RGB Color()` and `Color to RGB()` convert color values between JMP color numbers and the red-green-blue system. The following example finds the RGB values for JMP color 3 (red):

```
Color to RGB( 3 );
  {0.941176470588235, 0.196078431372549, 0.274509803921569}
```

Likewise, `HLS Color()` and `Color to HLS()` convert color values between JMP color numbers and the hue-lightness-saturation system.

```
win = New Window( "Color Wheel",
   Graph(
      Frame Size( 200, 200 ),
      For( hue = 0, hue < 360, hue += 30,
         y = 50 - 40 * Cos( hue * 2 * Pi() / 360 );
         x = 50 + 40 * Sin( hue * 2 * Pi() / 360 );
         Fill Color( HLS Color( hue / 360, 0.5, 1 ) );
         Oval( x - 10, y - 10, x + 10, y + 10, 1 );
      )
   )
);
```

Finally, `Heat Color()` returns the JMP color that corresponds to a value in any color theme that is supported by Cell Plot, Treemap, and so on. The syntax is:

```
Heat Color( n, <<"theme" )
```

The theme message is optional, and the default value is `"Blue to Gray to Red"`. You can specify any color theme, including custom color themes. You can also create and use an anonymous color theme as shown in the following examples:

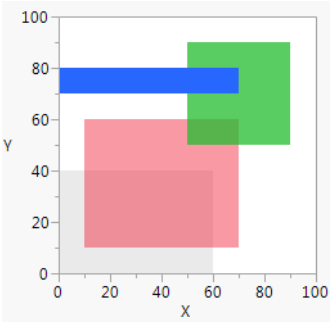
```
Heat Color( z, <<{"", {{1, 1, 0}, {0, 0, 1}} } );
Heat Color( z, <<{"", {blue, green, yellow} } );
```

Specify Transparency

In a graphics environment (like a Frame Box), use the Transparency function to set the level of transparency. The argument, alpha, can be any number between zero and one. The value 0 means clear and drawing has no effect. The value 1 means completely opaque and is the usual drawing mode. Intermediate values build semi-transparent color layers on top of what has already been drawn below it. The following example script illustrates transparency with rectangles.

```
win = New Window( "Transparency",
   Graph Box(
      Frame Size( 200, 200 ),
      Pen Color( "gray" );
      Fill Color( "gray" );
      Transparency( 0.25 );
      Rect( 0, 40, 60, 0, 1 );

      Pen Color( "red" );
      Fill Color( "red" );
      Transparency( 0.5 );
      Rect( 10, 60, 70, 10, 1 );

      Pen Color( "green" );
      Fill Color( "green" );
      Transparency( 0.75 );
      Rect( 50, 90, 90, 50, 1 );

      Pen Color( "blue" );
      Fill Color( "blue" );
      Transparency( 1 );
      Rect( 0, 80, 70, 70, 1 );
   )
);
```

Figure 12.20 Transparency and Rectangles



Add Fill Patterns

`Fill Pattern()` sets the pattern for filled areas.

Matrix Example

Specify a mask (a matrix of values between 0 and 1) or an image. Each value in the matrix creates a pixel.

```
win = New Window( "Example",
   Graph Box(
      Fill Pattern(
         [1 0.5 0 0, 0.5 0 0 1, 0 0 1 0.5, 0 1 0.5 0]
      );
      Polygon( [10 30 90], [88 22 44] );
   )
);
```

Named Pattern Example

Specify a name in quotation marks that corresponds to a pattern shown in Table 12.3.

```
win = New Window( "Named Pattern Example",
   Graph Box(
      Fill Pattern( "vertical light" );
      Polygon( [10 30 90], [88 22 44] );
   )
);
```

Graphic Example

Specify an image in quotation marks. If the image is not installed on the viewer's computer, question marks appear in the shape.

```
win = New Window( "Graphic Example",
   Graph Box(
      Fill Pattern( Open( "$SAMPLE_IMAGES/pi.gif", "gif" ) );
      Polygon( [10 30 90], [88 22 44] );
   )
);
```

Table 12.3 Fill Patterns

Pattern Name	Pattern
left slant light	

Table 12.3 Fill Patterns *(Continued)*

Pattern Name	Pattern
right slant light	
vertical light	
horizontal light	
grid light	
hatch light	
h wave light	
v wave light	
hollow circle	
left slant medium	
right slant medium	
vertical medium	
horizontal medium	
grid medium	
hatch medium	
h wave medium	
v wave medium	
filled circle	

Table 12.3 Fill Patterns *(Continued)*

Pattern Name	Pattern
left slant heavy	
right slant heavy	
vertical heavy	
horizontal heavy	
grid heavy	
hatch heavy	
h wave heavy	
v wave heavy	
diamond	
left slant heavy b	
right slant heavy b	
random	
square	
square offset	
wide	
tall	
checkerboard	

Table 12.3 Fill Patterns *(Continued)*

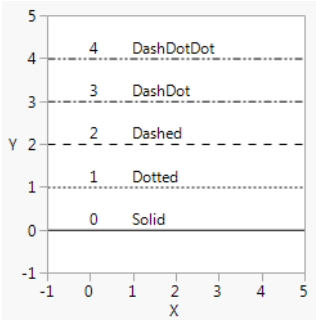
Pattern Name	Pattern
grid dots	
triangle up	
triangle down	
triangle left	
triangle right	
weave light	
weave heavy	
honeycomb	

Specify Line Types

You can also control `Line Style()` by number (0–4) or name (`Solid`, `Dotted`, `Dashed`, `DashDot`, `DashDotDot`). Figure 12.21 shows the numerical values for each line type.

```
linestyles = {"Solid", "Dotted", "Dashed", "DashDot", "DashDotDot"};
win = New Window( "Line Styles",
   Graph Box(
      Frame Size( 200, 200 ),
      X Scale( -1, 5 ),
      Y Scale( -1, 5 ),
      For( i = 0, i < 5, i++,
         Line Style( i );
         H Line( i );
         Text( {0, i + .1}, i );
         Text( {1, i + .1}, linestyles[i + 1] );
      )
   )
);
```

Figure 12.21 Line Styles



To control the thickness of lines, set a `Pen Size` and specify the line width in pixels. The default is 1 for single-pixel lines. For printing, think of `Pen Size` as a multiplier for the default line width, which varies according to your printing device.

```
win = New Window( "Pen Size",
   Graph Box(
      Pen Size( 2 ); // double-width lines
      Line( [10 30 90], [88 22 44] );
   )
);
```

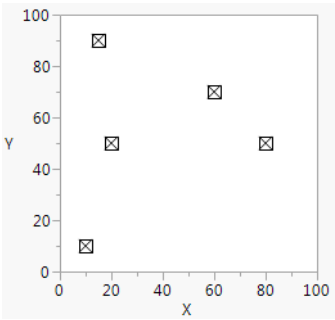
Draw with Pixels

You can also draw using pixel coordinates. First, set the `Pixel Origin()` in terms of graph coordinates, and then use `Pixel Move To()` or `Pixel Line To()` in pixel coordinates relative to that origin. The main use for `Pixel()` functions is for drawing custom markers that do not vary with the size or scale of the graph. You can store a marker in a script and then call it within any graph. This example uses `Function()` to store pixel commands in a script with its own arguments, *x* and *y*.

```
ballotBox = Function( {x, y},
   Pixel Origin( x, y );
   Pixel Move To( -5, -5 );
   Pixel Line To( -5, 5 );
   Pixel Line To( 5, -5 );
   Pixel Line To( -5, -5 );
   Pixel Line To( 5, 5 );
   Pixel Line To( -5, 5 );
   Pixel Move To( 5, 5 );
   Pixel Line To( 5, -5 );
);
win = New Window( "Custom Markers",
   Graph Box(
```

```
        Frame Size( 200, 200 ),
        ballotBox( 10, 10 );
        ballotBox( 15, 90 );
        ballotBox( 20, 50 );
        ballotBox( 80, 50 );
        ballotBox( 60, 70 );
    )
);
```

Figure 12.22 Drawing Custom Markers



Interactive Graphs

`Handle()` and `Mousetrap()` are functions for making interactive graphs that respond to clicking and dragging. `Handle()` lets you parametrize a graph by adding a handle-marker that can be dragged around with the mouse, executing the graph's script at each new location. `Mousetrap()` is similar, but it takes its arguments from the coordinates of a click without dragging a handle. The main difference is that `Handle()` only catches mousedowns at the handle-marker's location, but `Mousetrap()` catches mousedowns at any location.

Another approach is to place buttons or slider controls outside the graph with `Button Box()`, `Slider Box()`, or `Global Box()`.

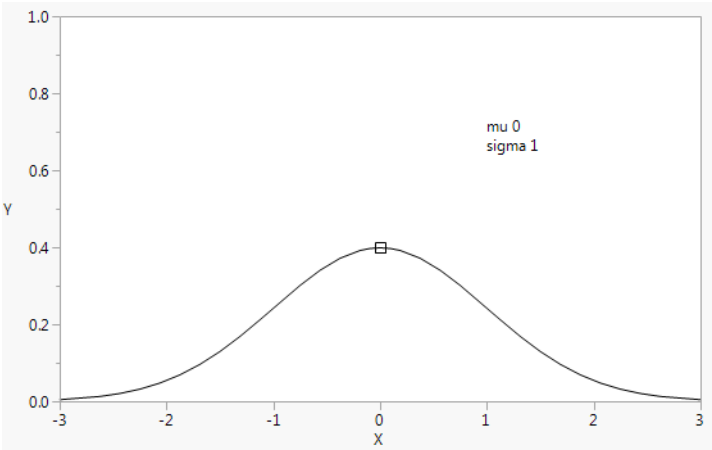
Handle()

`Handle()` places a marker at the coordinates given by the initial values of the first two arguments and draws the graph using the initial values of the arguments. You can then click and drag the marker to move the handle to a new location. The first script is executed at each mousedown to update the graph dynamically, according to the new coordinates of the handle. The second script (optional, and not used here) is executed at each mouseup, similarly; see the example for "Mousetrap()".

```
// Normal Density
mu = 0;
sigma = 1;
rsqrt2pi = 1 / Sqrt( 2 * Pi() );
win = New Window( "Normal Density",
  Graph Box(
     Frame Size( 500, 300 ),
     X Scale( -3, 3 ),
     Y Scale( 0, 1 ),
     Y Function( Normal Density( (x - mu) / sigma ) / sigma, x );
     Handle(
        mu,
        rsqrt2pi / sigma,
        mu = x;
        sigma = rsqrt2pi / y;
     );
     Text( {1, .7}, "mu ", mu, {1, .65}, "sigma ", sigma );
  )
);
```

Run demoPlotProb.jsl in the JMP Samples/Scripts folder to see Beta Density, Gamma Density, Weibull Density, and LogNormal Density graphs. The output for Normal Density is shown in Figure 12.23. Because we cannot show you the picture in motion, be sure to run this script yourself.

Figure 12.23 Normal Density Example for `Handle()`



To avoid errors, be sure to set the initial values of the handle's coordinates, as in the first line of this example.

If you want to use some function of a handle's coordinates, such as in the normal density example, you should adjust the arguments for `Handle()`. Otherwise, the handle marker would run away from the mouse.

```
Y Function( a * x ^ b );
Handle( a, b, a = 2 * x, b = y );
```

Suppose you drag the marker from its initial location to (3,4). The argument *a* is set to 6 and *b* to 4; the graph is redrawn as $Y = 6x^4$; and the handle is now drawn at (6,4), several units away from the mouse. To compensate, you would adjust the first argument to handle, for example.

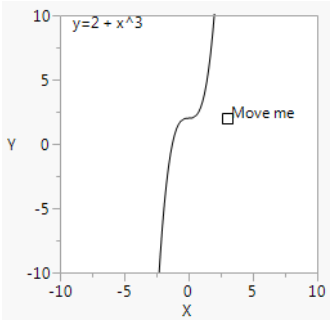
```
Handle( a / 2, b, a = 2 * x; b=y );
```

To generalize, suppose you define the `Handle()` arguments as functions of the handle's (x, y) coordinates. For example, `a=f(x)` and `b=g(y)`. If `f(x)=x` and `g(y)=y`, then you would specify simply `a,b` as the first two arguments. If not, you would solve $a = f(x)$ for x and solve $b = g(y)$ for y to get the appropriate arguments.

You can use other functions to constrain `Handle()`. For example, here is an interactive graph to demonstrate power functions that uses `Round()` to prevent bad exponents and to keep the intercepts simple.

```
a = 3;
b = 2;
win = New Window( "Intercepts and Powers",
   Graph Box(
      Frame Size( 200, 200 ),
      X Scale( -10, 10 ),
      Y Scale( -10, 10 ),
      Y Function( Round( b ) + x ^ (Round( a )), x );
      Handle(
         a,
         b,
         a = x;
         b = y;
      );
      Text( {a, b}, " Move me" );
      Text( {-9, 9}, "y=", Round( b ), " + x^", Round( a ) );
   )
);
```

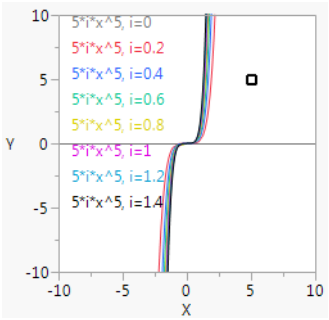
Figure 12.24 Intercepts and Powers for `Handle()`



`Handle()` and `For()` can be nested for complex graphs.

```
a = 5;
b = 5;
win = New Window( "Powers",
   Graph Box(
      Frame Size( 200, 200 ),
      X Scale( -10, 10 ),
      Y Scale( -10, 10 ),
      For( i = 0, i < 1.5, i += .2,
         Pen Color( 1 + 10 * i );
         Text Color( 1 + 10 * i );
         Y Function( i * x ^ Round( a ), x );
         Handle(
            a,
            b,
            a = x;
            b = y;
         );
         h = 9 - 10 * i;
         Text( {-9, h}, b, "*i*x^", Round( a ), ", i=", i );
      ) ) );
```

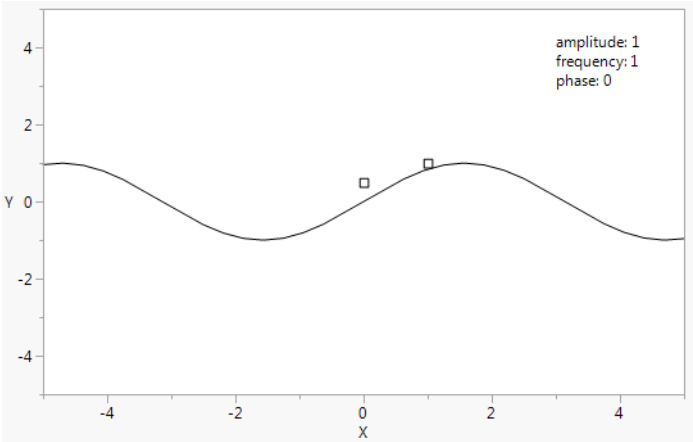
Figure 12.25 Nested Handle() and For()



You can use more than one handle in a graph:

```
amplitude = 1;
freq = 1;
phase = 0;
win = New Window( "Sine Wave",
   Graph Box(
      Frame Size( 500, 300 ),
      X Scale( -5, 5 ),
      Y Scale( -5, 5 ),
      Y Function( amplitude * Sine( x / freq + phase ), x );
      Handle( // first handle
         freq,
         amplitude,
         freq = x;
         amplitude = y;
      );
      Handle( phase, .5, phase = x ); // second handle
      Text(
         {3, 4},
         "amplitude: ",
         Round( amplitude, 4 ),
         {3, 3.5},
         "frequency: ",
         Round( freq, 4 ),
         {3, 3},
         "phase: ",
         Round( phase, 4 )
      );
   )
);
```

Figure 12.26 Two Handles



Mousetrap()

`Mousetrap()` takes arguments for a graph from the coordinates of a mouse click. The first script is executed after each mousedown and the second script after each mouseup to update the graph dynamically, according to the new coordinates of the handle. As with `Handle`, it is important to set the initial values for the `Mousetrap's` coordinates. If you include both `Mousetrap()` and `Handle()` in a graph, put the `Handle()` before the `Mousetrap()`. Then `Handle()` has a chance to catch clicks before a `Mousetrap()` does.

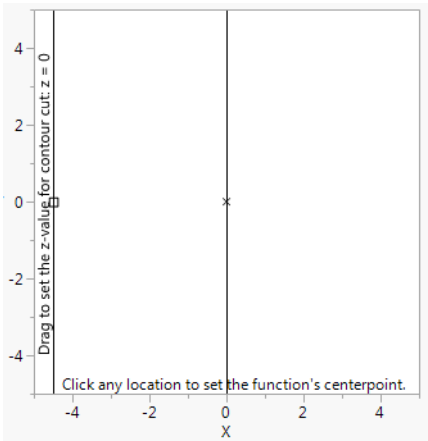
This example uses both `Mousetrap()` and `Handle()` to draw a three-dimensional function centered on the `Mousetrap()` coordinates, where the single contour line takes its value from a `Handle()`.

```
x0 = 0;
y0 = 0;
z0 = 0;
win = New Window( "Viewing a 3-D Function in Flatland",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( -5, 5 ),
      Y Scale( -5, 5 ),
      Contour Function(
         Exp( -(x - x0) ^ 2 ) * Exp( -(y - y0) ^ 2 ) * (x - x0),
         x,
         y,
         z0 / 10
      );
```

```
        // get the z-cut values from a handle
        Handle( -4.5, z0, z0 = Round( y * 10 ) / 10 );
              V Line( -4.5 );
        Text Size( 9 );
        Text(
           Counterclockwise,
           {-4.6, -4},
           "Drag to set the z-value for contour cut: z = " || Char( z0 / 10 )
        );
        Marker Size( 2 );
        Marker( 2, {x0, y0} );
        Mousetrap( // set the origin to the click-point
           x0 = x;
           y0 = y;
        );
        Text(
           {-4.25, -4.9},
           "Click any location to set the function's centerpoint."
        );
     )
  );
```

Note: Instead of setting `x0` to `x` and `y0` to `y` in the `Mousetrap()` expression, you might want to use `Function( {xx, yy}, Show( xx, yy ); x0 = xx; y0 = yy );`. `xx` and `yy` are formal parameters to the function. They come from arguments that the mousetrap passed when the user clicked on the graph. The `Function( {xx, yy}...)` alternative makes the `xx` and `yy` names explicit, and makes it clear the you can name them anything, not just `x`, `y`.

Figure 12.27 Mousetrap() and Handle()



You might use Mousetrap() to collect points in a data table, such as for visually interpolating points in a graph. The following example illustrates a script that could be adapted and added to a data plot (such as a scatterplot from Fit Y by X) for that purpose. After you run the script and click in the graph, the point at which you click is recorded in the data table.

```
dt = New Table( "dat1" );
New Column( "xx", Numeric );
New Column( "yy", Numeric );
x = 0;
y = 0;
Add Point = Expr(
   dt << Add Rows( 1 );
   Row() = N Row();
   :xx = x;
   :yy = y;
);
win = New Window( "Add Points",
   Graph Box(
      Frame Size( 500, 300 ),
      X Scale( -5, 5 ),
      Y Scale( -5, 5 ),
      For Each Row( Marker( {xx, yy} ) );
      Mousetrap( {}, Add Point );
   )
);
```

Notice that the first script argument is empty. At mousedown, nothing happens. The second script, add point, is executed at mouseup to add a data point. If you click, drag, and release, then the point that is added to your data set is the point where you let go of the mouse button; it is not the point where you pressed down the mouse button.

Drag Functions

There are five Drag() functions to perform similar functions to Handle() and Mousetrap() but with more than one point at a time. For *n* coordinates in matrices listed as the first two arguments:

- Drag Marker() draws n markers.
- Drag Line() draws a connected line with n vertices and $n-1$ segments.
- Drag Rect() draw a filled rectangle using the first two coordinates, ignoring any further coordinates.
- Drag Polygon() draws a filled polygon with n vertices.

- `Drag Text()` draws a text item at the coordinates. Or if there is a list of text items, it draws the *i*th list item at the *i*th (*x*,*y*) coordinate. If there are fewer list items than coordinate pairs, the last item is repeated for remaining points.

The syntax for these commands:

```
Drag Marker( xMatrix, yMatrix, dragScript, mouseupScript );
Drag Line( xMatrix, yMatrix, dragScript, mouseupScript );
Drag Rect( xMatrix, yMatrix, dragScript, mouseupScript );
Drag Polygon( xMatrix, yMatrix, dragScript, mouseupScript );
Drag Text( xMatrix, yMatrix, "text", dragScript, mouseupScript );
```

They all must have L-value arguments for the coordinates. The coordinates are literal matrices or names of matrix values that are modified if you click a vertex and drag it to a new position. The script arguments are optional and behave the same as with `Handle()`. However, there is no *x* nor *y* that is modified as in `Handle()`.

`Drag()` functions are ways to display data that the user can adjust and then capture the adjusted values. Consider the earlier script to draw the JMP man. `Drag Polygon()` makes it possible to draw an editable JMP man; using a matching `Drag Marker()` statement makes the vertices more visible. And, similar to the `Mousetrap()` example, you can save the new coordinates to a data table. Notice how `:` and `::` operators avoid ambiguity among matrices and data table columns with the same names.

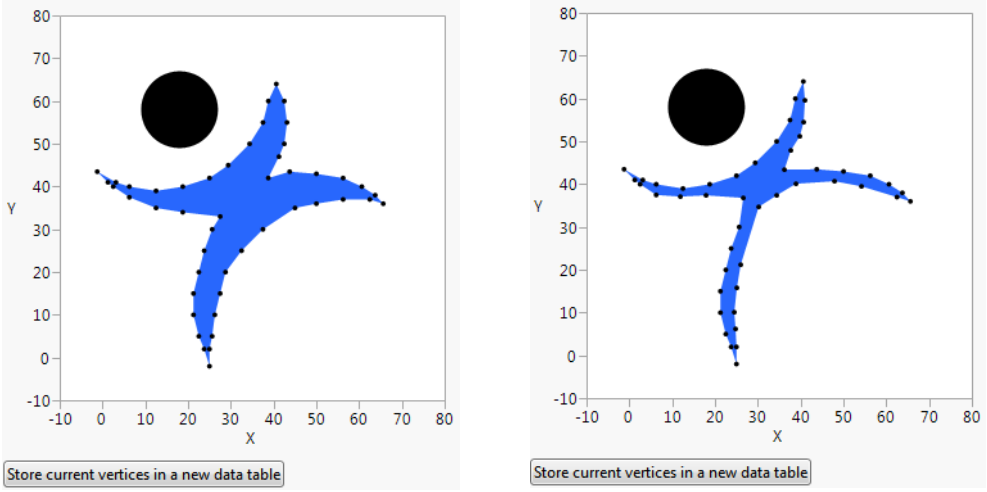
You could just as easily put *`storepoints`* in the fourth argument of `Drag Polygon()` or `Drag Marker()`, but that would create a data table after each drag. You probably just want a single data table when you are finished. Regardless, the values in *`gCoordX`* and *`gCoordY`* update with each drag.

```
::i = 1;
storepoints = Expr(
  mydt = New Table( "My coordinates" || Char( i ) );
  i++;
  New Column( "GCoordX", Numeric );
  New Column( "GCoordY", Numeric );
  mydt << Add Rows( N Row( GcoordX ) );
  :GCoordX << Values( ::GcoordX );
  :GCoordY << Values( ::GcoordY );
);

::GcoordX = [25, 23.75, 22.5, 21.25, 21.25, 22.5, 23.75, 25.625, 27.5, 18.75,
  12.5, 6.25, 2.5, 1.25, -1.25, 3.125, 6.25, 12.5, 18.75, 25, 29.375, 34.375,
  37.5, 38.75, 40.625, 42.5, 43.125, 42.5, 41.25, 38.75, 43.75, 50, 56.25,
  60.625, 63.75, 65.625, 62.5, 56.25, 50, 45, 37.5, 32.5, 28.75, 27.5, 26.25,
  25.625, 25];
::GcoordY = [-2, 2, 5, 10, 15, 20, 25, 30, 33, 34, 35, 37.5, 40, 41, 43.5, 41,
  40, 39, 40, 42, 45, 50, 55, 60, 64, 60, 55, 50, 47, 42, 43.5, 43, 42, 40,
  38, 36, 37, 37, 36, 35, 30, 25, 20, 15, 10, 5, 2];
```

```
win = New Window( "Redraw the JMP Man!",
   V List Box(
      Graph Box(
         Frame Size( 300, 300 ),
         X Scale( -10, 80 ),
         Y Scale( -10, 80 ),
         Fill Color( "blue" );
         Drag Polygon( GcoordX, GCoordY );
         Pen Color( "gray" );
         Drag Marker( GcoordX, GCoordY );
         Fill Color( {0, 0, 0} );
         Circle( {18, 58}, 9, "FILL" );
      ),
      Button Box( "Store current vertices in a new data table", storepoints )
   )
);
```

Suppose you want the body of the JMP man to be smaller. Here is how he looks before and after some judicious vertex-dragging. Clicking the button after re-shaping the JMP Man executes the *storepoints* script to save his new, slender figure in a data table of coordinates.

Figure 12.28 Redraw the JMP Man



This example uses two functions discussed under "Constructors for New Windows":

- `Button Box()`, which creates controls outside the graph.
- `V List Box()`, which combined the graph box and the button box in the same graph window.

Troubleshoot Interactive Graphs

If your interactive graphs do not work as expected, make sure that you supply initial values for the `Handle()` or `Mousetrap()` coordinates (and other globals as needed). Also make sure that the values make sense for the graph.

Create a Color Picker

Let users select a color to apply to graphs by creating a color picker. Users can select a predefined color or create their own color. You can also specify a default color in your script.

- Use the named or numbered colors shown in "Colors and Markers".

  or

- Specify values between 0 and 1 for each of the red, blue, and green components in `RGB Color()`.

Examples of Creating Color Pickers

The following example does not name the window, so the default title is used. Because the script does not specify a color, Black is selected by default. If the user selects a color in the color picker window, then the lines are drawn in the selected color.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Run Script( "Bivariate" );
Wait( 0 );
pickColor = Pick Color();
Report( biv )[AxisBox( 1 )] <<
Add Ref Line( 105, "Dashed", pickColor, "Mean" );
```

The following example creates a color picker with the window title "Select a Line Color". BlueGreen is preselected.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Run Script( "Bivariate" );
Wait( 0 );
pickColor = Pick Color( "Select a Line Color", "BlueGreen" );
Report( biv )[AxisBox( 1 )] <<
Add Ref Line( 105, "Dashed", pickColor, "Mean" );
```

The following example creates a color picker with the window title "Select a Background Color". BlueGreen is preselected.

```
picked color = Pick Color( "Select a Background Color", "BlueGreen" );
New Window( "Example",
        gb = Graph Box(
```

```
                Frame Size( 300, 300 ),
                Marker( Marker State( 3 ), [11 44 77], [75 25 50] );
                Line( [10 30 70], [88 22 44] );
        )
);
gb[FrameBox(1)] << Background Color ( picked color );
```

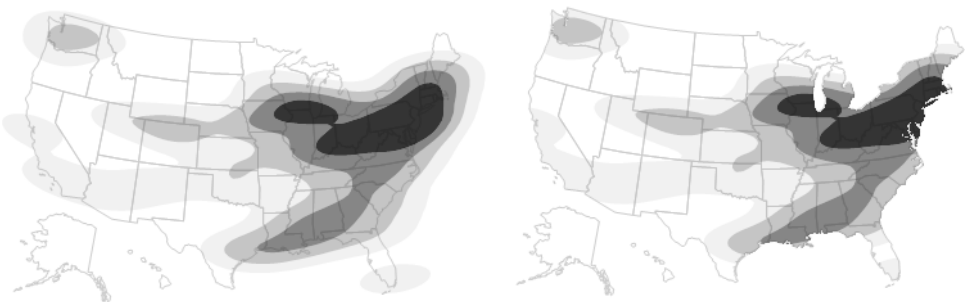
The following example specifies the RGB values for red, green, and blue to create pink. If you do not know the JMP color name or number, you might consider this option.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Run Script( "Bivariate" );
Wait( 0 );
pickColor = Pick Color( "RGB", RGBCOLOR( 1/*red*/, 0/*green*/, 1/*blue*/ ) );
Report( biv )[AxisBox( 1 )] <<
Add Ref Line( 105, "Dashed", pickColor, "Mean" );
```

Clip Display Segs in Graphics

Display segs (such as reference lines, grid lines, and contours) can be clipped to conform to the natural boundaries of the underlying data. This gives you a better view of the boundaries. For example, Figure 12.29 shows contours that are clipped to the boundaries of a map of the United States.

Figure 12.29 Unclipped (Left) and Clipped (Right) Boundaries



Send the `Clip Shape` message to the Contour Seg object to clip the display segs.

```
Open( "$SAMPLE_DATA/Cities.jmp" );
gb = Graph Builder(
   Size( 653, 396 ),
   Show Control Panel( 0 ),
   Variables( X( :Longitude ), Y( :Latitude ) ),
   Elements( Contour( X, Y, Legend( 2 ) ) ),
   SendToReport(
```

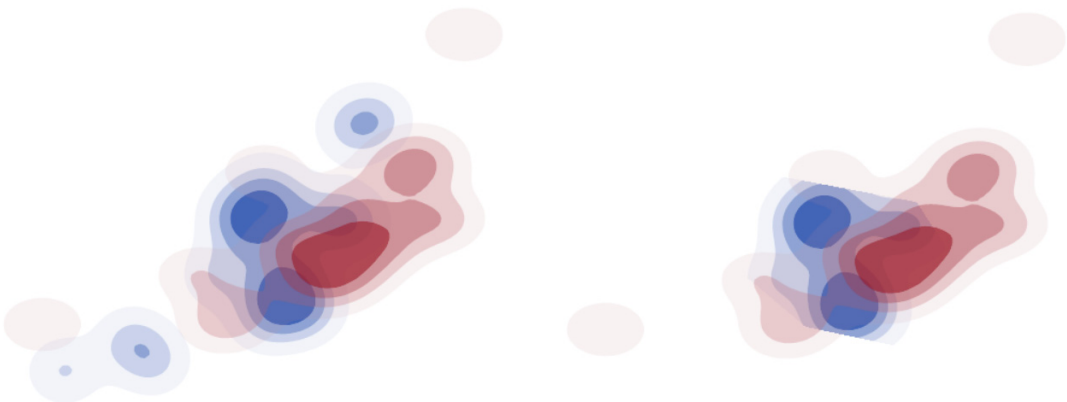
```
      Dispatch(
         {},
         "Graph Builder",
         FrameBox,
         {Background Map( Boundaries( "US States" ) ), Grid Line Order( 2 ),
   Reference Line Order( 3 )}
         )
      )
);
cs = (gb << Report)[FrameBox( 1 )] << Find Seg( Contour Seg( 1 ) );

Wait( 2 );

cs << Clip Shape( Boundaries( "US States" ) );
```

Tip: `Clip Shape(Empty())` removes the clipping path.

Here's an example of defining the clip shape path with a matrix:

```
Open( "$SAMPLE_DATA/Big Class.jmp" );
gb = Graph Builder(
   Show Control Panel( 0 ),
   Variables( X( :height ), Y( :weight ), Overlay( :sex ) ),
   Elements( Contour( X, Y, Legend( 3 ) ) )
);
r = ( gb << Report );
cs = r[FrameBox( 1 )] << Find Seg( Contour Seg( 1 ) );
cs << Clip Shape( Path( [60 80 1, 50 140 2, 65 120 2, 70 65 -2] ) );
```

Figure 12.30 Unclipped (Left) and Clipped (Right) Contours



The matrix [60 80 1, 50 140 2, 65 120 2, 70 65 -2] indicates the following:

1. Start at 60, 80.

2. Draw a line to 50, 140.
3. Draw a line to 65, 120.
4. Draw a line to 70, 65.
5. Connect back to the first point.

Create Background Maps

Background maps can be scripted in JSL. You can write a script that creates a graph and then turns on the background map in the script.

There are two types of background maps: `Images()` and `Boundaries()`. Each of these takes a parameter, which is the name of the map to use. The name is one of the maps listed in the window.

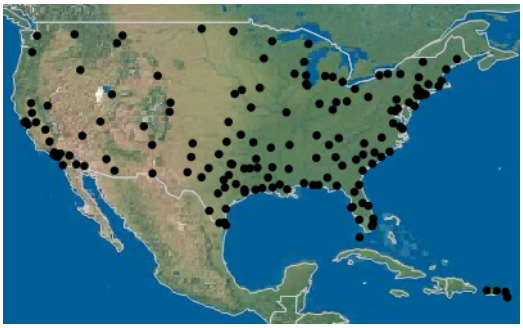
- For `Images()`, the choices are Simple Earth, Detailed Earth, NASA, Street Map Service, and Web Map Service. If you use Web Map Service, then there are two additional parameters: the WMS URL and the layer supported by the WMS server.
- For `Boundaries()`, the choices vary since boundaries can be user-defined. A typical choice is World.

The following example uses the Simple Earth as an image and World as a boundary:

```
dt = Open( "$SAMPLE_DATA/Air Traffic.jmp" );
dt << Graph Builder(
   Size( 1101, 603 ),
   Show Control Panel( 0 ),
   Variables( X( :Longitude ), Y( :Latitude ) ),
   Elements( Points( X, Y, Legend( 8 ) ) ),
   SendToReport(
      Dispatch(
         {},
         "Graph Builder",
         Frame Box,
         {Background Map( Images( "Simple Earth" ), Boundaries( "World" ) ),
         Grid Line Order( 3 ), Reference Line Order( 4 )}
      )
   )
);
```

Figure 12.31 shows an excerpt from the map:

Figure 12.31 JSL Scripting Example



To change the script to use a WMS server, the command would look like this:

```
Background Map ( Images ( "Web Map Service",
  "http://sedac.ciesin.columbia.edu/geoserver/wms",
  "gpw-v3:gpw-v3-population-density_2000" ), Boundaries ( "US States" ) )
```

To see which layers are available on a WMS server, install the WMS Explorer Add-In. Download the add-in from the JMP File Exchange at https://community.jmp.com/docs/DOC-6095. Note that some WMS servers are not reliable. If the server is down, or if the user does not have an Internet connection, the WMS map does not appear.

Tips:

- To see the syntax for the JSL script, add a background map through the user interface. Then from the red triangle menu, select **Save Script > To Script Window** to see the script that is generated.
- To specify image and boundary names in the JSL script, use the names shown in the Set Background Map window. (Right-click a map and select **Graph > Background Map**).
- Look at sample data to find examples of background map scripts. Select **Help > Sample Data Library** and open Napoleons March.jmp, Pollutants Map.jmp, or San Francisco Crime.jmp. Right-click the map table script and select **Edit** to see the `Background Map()` function.

Graph Elements

You can use the following commands inside `Graph Box()` statements. This chapter focuses on the JSL that is specific to graphing, but you can also use general script commands such as `For` and `While`.

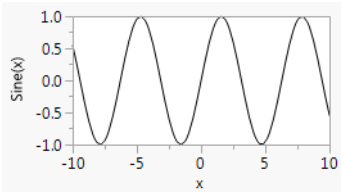
Note: `New Window()` expects window and display box commands, and platforms expect platform commands. Therefore, the general script commands such as `For` and `While` might not work inside platforms or `New Window()`, or you might get unexpected results.

Plot Functions

A `Y Function()` function is used to draw smooth functions. The first argument is the expression to be plotted. The second argument is the name of the X variable in the expression.

```
win = New Window( "Sine Function",
   Graph Box(
      Frame Size( 200, 100 ),
      X Scale( -10, 10 ),
      Y Scale( -1, 1 ),
      X Name( "x" ),
      Y Name( "Sine(x)" ),
      Y Function( Sine( x ), x ) ) );
```

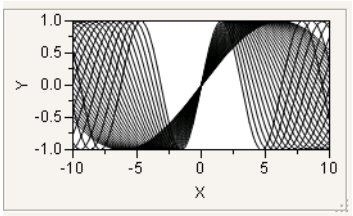
Figure 12.32 Sine Wave



You can use `For` to overlap several sine waves:

```
win = New Window( "Overlapping Sine Waves",
   Graph Box(
      Frame Size( 200, 100 ),
      X Scale( -10, 10 ),
      Y Scale( -1, 1 ),
      For( i = 1, i <= 4, i += .1,
         Y Function( Sine( x / i ), x )
      )
   )
);
```

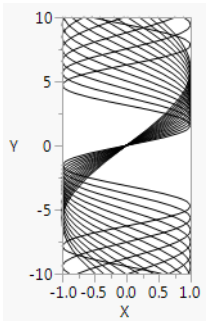
Figure 12.33 Overlapping Sine Waves



Similarly, `X Function()` is for drawing a graph where the symbol is varied on the Y variable.

```
win = New Window( "Overlapping Sine Waves",
   Graph Box(
      Frame Size( 100, 200 ),
      X Scale( -1, 1 ),
      Y Scale( -10, 10 ),
      For( i = 1, i <= 4, i += .2,
         X Function( Sine( y / i ), y )
      )
   )
);
```

Figure 12.34 Overlapping Sine Waves along the X-Axis



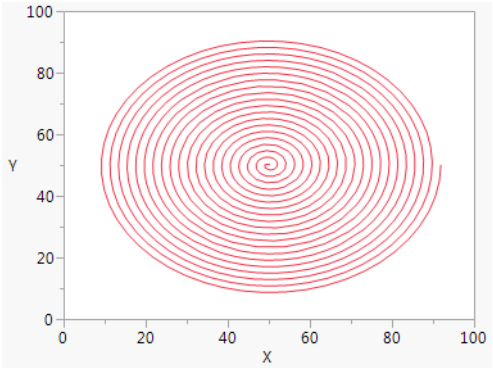
An `XY Function()` draws a smooth curve using a pair of formulas (parametric equations) that depend on a third variable. The third variable's value is incremented from a minimum value to a maximum value to generate the X-Y pairs.

```
win = New Window( "Spiral",
   Graph Box(
      Pen Color( "red" ); // red line color
      xCenter = 50; // location of the center of the curve on the x axis
      yCenter = 50; // location of the center of the curve on the y axis
      minAngle = 0;
      maxAngle = Pi() * 2 * 20;
      XY Function(
```

```
            xCenter + ((ta / 3) * Cos( ta )),
            yCenter + ((ta / 3) * Sin( ta )),
            ta,
            Min( minAngle ),
            Max( maxAngle ),
            Inc( Pi() / 100 )
         );
      )
   );
```

In this example, `Sin()` and `Cos()` use `ta` as an argument (rotates) *and* as a factor (expands). (`Sin()` and `Cos()` use radians, not degrees.)

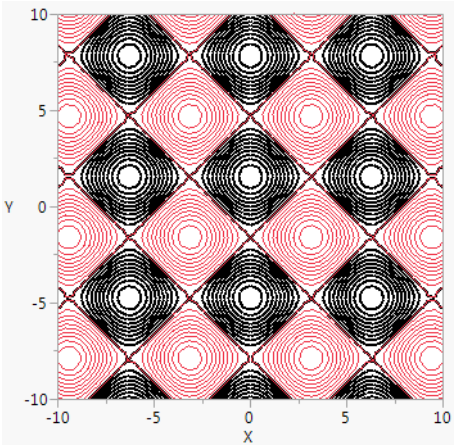
Figure 12.35 Spiral Parametric Plot



`Contour Function()` is an analogous way to represent a three-dimensional function in a two-dimensional space. The final argument specifies the value(s) for the contour line(s). The argument can be a value, an indexed range of values using `::`, or a matrix of values.

```
win = New Window( "Bird's eye view of the egg carton function",
   Graph Box(
      Frame Size( 300, 300 ),
      X Scale( -10, 10 ),
      Y Scale( -10, 10 ),
      Pen Color( "black" );
      Pen Size( 2 );
      Contour Function( Sine( y ) + Cosine( x ), x, y, (0 :: 20) / 5 );
      Pen Color( "red" );
      Pen Size( 1 );
      Contour Function( Sine( y ) + Cosine( x ), x, y, (-20 :: 0) / 5 );
   )
);
```

Figure 12.36 Egg Carton Function



`Normal Contour()` draws normal probability contours for *k* populations and two variables. The first argument is a scalar probability or a matrix of probability values for the contours. Subsequent arguments are matrices to specify means, standard deviations, and correlations. The mean and standard deviation matrices have dimension *k* × 2. The correlation matrix should be *k* × 1, where the first row pertains to the first contour, the second row to the second contour, and so on. The first column is for *x* and the second column for *y*. Consider the following example:

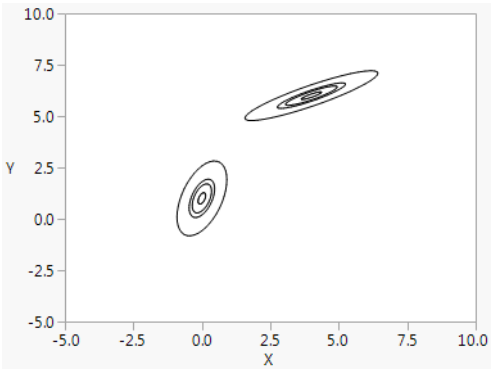
```
Normal Contour(
   [ prob1,
     prob2,
     prob3, ...],
   [ xmean1 ymean1,
     xmean2 ymean2,
     xmean3 ymean3, ...],
   [ xsd1 ysd1,
     xsd2 ysd2,
     xsd3 ysd3, ...],
   [ xycorr1,
     xycorr2,
    xycorr3, ...]);
```

The following script draws contours at probabilities 0.1, 0.5, 0.7, and 0.99 for two populations and two variables. The first population has *x* mean 0 and *y* mean 1, with standard deviation 0.3 along the *x* axis and 0.6 along the *y*-axis, and with correlation 0.5. The second has *x* mean 4 and *y* mean 6, with standard deviation 0.8 along the *x* axis and 0.4 along the *y*-axis, and with correlation 0.9.

```
win = New Window( "Normal Contours",
   Graph Box(
```

```
        X Scale( -5, 10 ),
        Y Scale( -5, 10 ),
        Normal Contour( [.1, .5, .7, .99], [0 1, 4 6], [.3 .6, .8 .4], [.5, .9] )
    )
);
```

Figure 12.37 Normal Contour Function



`Normal Contour()` is a general way to accomplish effects such as Bivariate's density ellipses. The Bivariate script in the Football.jmp sample data creates an example.

Gradient Function

`Gradient Function()` fills a set of rectangles on a grid according to a color determined by the expression value as it crosses a range corresponding to a range of colors.

`Gradient Function( `*`expression`*`, `*`xname`*`, `*`yname`*`, [`*`zlow`*`, `*`zhigh`*`], ZColor( [`*`colorLow`*`, `*`colorHigh`*`] ), <XGrid( `*`min`*`, `*`max`*`, `*`incr`*` )>, <YGrid( `*`min`*`, `*`max`*`, `*`incr`*` )>, <Transparency( t )>;`

Here are explanations of the elements in the `Gradient Function()` syntax:

```
GradientFunction(
    /* the expression to be contoured, which is a function in terms of
    the two variables that follow expression*/
    Expression,

    // the two variable names used in the expression
    xname, yname,

    // the low and high expression values the gradient is scaled between
    [zlow, zhigh],

    // the colors that correspond to the low value and the high value
    ZColor([colorLow, colorHigh])
```

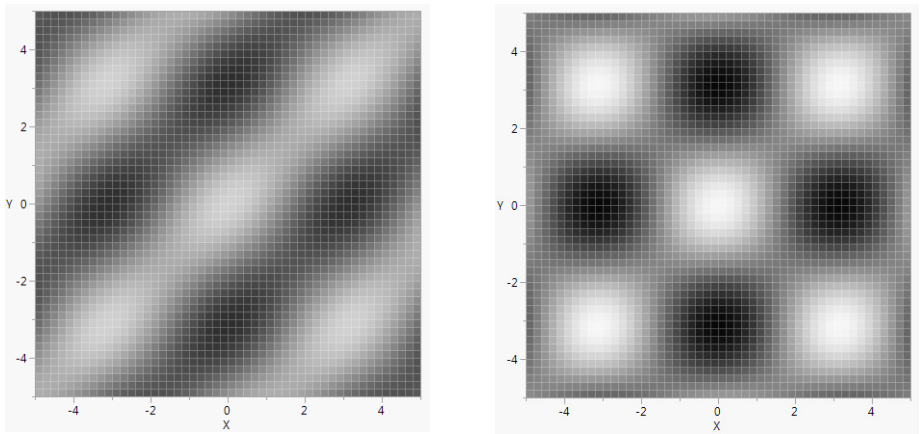
```
// optional specification for the grid of values
<XGrid(min, max, incr),>
<YGrid(min, max, incr)> );

/* the value 0 means clear. The value 1 means completely opaque and
is the usual drawing mode. */
<Transparency>
```

The `ZColor()` values must be numeric codes rather than names. You can use the color menu indices (such as 0=black, 1=grey, 2=white, 3=red, 4=green, and 5=blue) found in "Specify Colors".

The following example uses `Gradient Function()` to create an animated graph.

```
phase = 0.7;
win = New Window( "Gradient Function",
   a = Graph(
      Frame Size( 400, 400 ),
      X Scale( -5, 5 ),
      Y Scale( -5, 5 ),
      Gradient Function(
         phase * Sine( x ) * Sine( y ) + (1 - phase) * Cosine( x ) * Cosine( y
   ),
         x,
         y,
         [-1 1],
         zcolor( [0, 2] )
      )
   )
);
b = a[FrameBox( 1 )];
For( i = 1, i <= 5, i++,
   For( phase = 0, phase < 1, phase += 0.05,
      b << Reshow;
      Wait( 0.01 );
   );
   For( phase = 1, phase > 0, phase -= 0.05,
      b << Reshow;
      Wait( 0.01 );
   );
);
```

Figure 12.38 Gradient Function



Get the Properties of a Graphics Frame

Several functions are useful for getting properties of an existing graphics frame:

H Size Returns the horizontal size of the graphics frame in pixels.

V Size Returns the vertical size of the graphics frame in pixels.

X Origin Returns the *x*-value for the left edge of the graphics frame.

X Range Returns the distance from the left to right edges of a display box.

Y Origin Returns the *y*-value for the bottom edge of the graphics frame.

Y Range Returns the distance from the bottom to top edges of a display box.

In the following expressions, the first line calculates the right edge. The second line calculates the top edge.

```
Oval(
   ...,
   rightEdge = X Origin() + X Range();
   topEdge = Y Origin() + Y Range();
);
```

Add a Legend

To add a legend to a graph, send the `Row Legend` message to the frame box. In the `Row Legend` message, specify which column you want to base the legend on and whether the legend affects colors and markers.

For example, using Big Class.jmp, submit the following JSL to turn on a legend based on the age column. Set both colors and markers by values in the age column.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( y( :height ), x( :weight ) );
rbiv = biv << Report;
framebox = rbiv[Frame Box( 1 )];

/* make a legend with the "age" title
display colors and markers in the legend */
framebox << Row Legend( "age", Color( 1 ), Marker( 1 ) );
```

The `Color()` and `Marker()` arguments are optional. Colors are on and markers are off by default.

To use a continuous scale if your variable is nominal or ordinal, you can use a continuous color theme with the `Continuous Scale( 1 )`.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
biv = dt << Bivariate( y( :weight ), x( :height ) );
rbiv = biv << Report;
framebox = rbiv[Frame Box( 1 )];
framebox << Row Legend( "age", Color( 1 ), Continuous Scale( 1 ) );
```

To interactively add a legend, right-click the graph, select **Row Legend**, and change the column settings.

Hover Labels

When you hover over a visual element in a graph, usually you see a basic description of the data point on a hover label.

You can customize what appears on the hover label using the following methods:

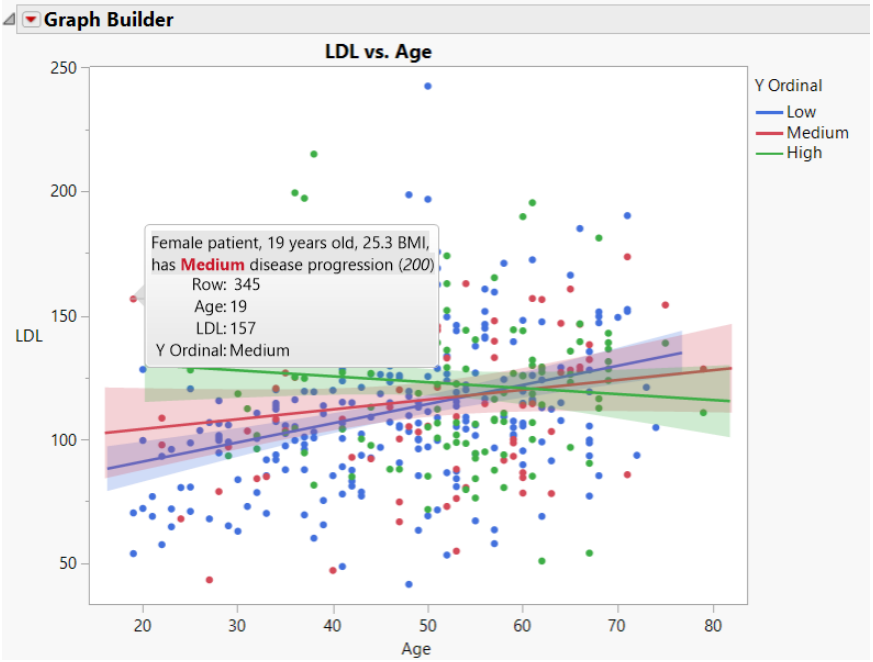
- "Add Rich Text to Hover Labels Using Textlets"
- "Add Graphs or Images to Hover Labels Using Graphlets"
- "Update Names and Values in Hover Labels Using Gridlets"
- "Create Drill-Down Graphs"
- "Work with the Hover Label Execution Context"

Add Rich Text to Hover Labels Using Textlets

A *textlet* adds richly formatted text to the hover label. You might use a textlet to describe data in an explanatory paragraph or to emphasize important information by making it bold or applying a color.

The textlet in Figure 12.39 shows text that is formatted in a gray box with red text.

Figure 12.39 Textlet Example



The following script creates the textlet shown in Figure 12.39.

```
Names Default To Here( 1 );
dt = Open( "$SAMPLE_DATA/Diabetes.jmp" );

// create the baseline visualization
gb = dt << Graph Builder(
   Size( 534, 456 ),
   Show Control Panel( 0 ),
   Variables( X( :Age ), Y( :LDL ), Overlay( :Y Ordinal ) ),
   Elements( Points( X, Y, Legend( 8 ) ), Line Of Fit( X, Y, Legend( 10 ) ) )
);

// get a handle to the graph frame box
rpt = gb << Report;
```

```
frame = rpt[Framebox( 1 )];

/* configure the textlet -- a hover label extension that adds
   a rich text paragraph to the hover label. */
frame << Set Textlet(
    /* "Setup" is a JSL fragment that uses Hover Label Execution Contect
     variables to define content that will be embedded in the
     rich text paragraph.*/
   Setup(
      // map numerical values to text
      local:gender = If( Column( local:_dataTable, "Gender" )[local:_firstRow]
         == "1",
         "Male",
         "Female"
      );
      // map a categorical value to text color
      local:color = Match( local:_Y Ordinal,
         "Low", "Medium Dark Blue",
         "Medium", "Medium Dark Red",
         "High", "Medium Dark Green"
      );
      // capture values from other columns for the current row
      local:progression = Column( local:_dataTable, "Y" )[local:_firstRow];
      local:BMI = Column( local:_dataTable, "BMI" )[local:_firstRow];
   ),
   /* markup defines the rich text template. It supports HTML-like markup such
   as bold, italic and other font tags. Variable references between braces
   ({}) are substituted with their actual value. */
   Markup(
      "{local:gender} patient, {local:_age} years old, {local:BMI} BMI,
      has <font color='{local:color}'><b>{local:_Y Ordinal}</b></font>
      disease progression (<i>{local:progression}</i>)"
   )
);
// pin a hover label to display a textlet-created rich text paragraph.
frame << Add Pin Annotation(
   Seg( Marker Seg( 2 ) ),
   Index( 64 ),
   Index Row( 344 ),
   UniqueID( 568154464 ),
   FoundPt( {96, 222} ),
   Origin( {18.68125, 156.93706122449} ),
   Tag Line( 1 )
);
```

Texlet Commands

See the example in "Add Rich Text to Hover Labels Using Textlets" for more information about writing texlet expressions.

Setup

`Setup()` defines variables used in the rich text that is displayed on the hover label.

```
...
frame << Set Textlet(
    /* "Setup" is a JSL fragment that uses Hover Label Execution Contect
     variables to define content that will be embedded in the
     rich text paragraph.*/
  Setup(
     // map numerical values to text
     local:gender = If( Column( local:_dataTable, "Gender" )[local:_firstRow]
        == "1",
        "Male",
        "Female"
     );
     // map a categorical value to text color
     local:color = Match( local:_Y Ordinal,
        "Low", "Medium Dark Blue",
        "Medium", "Medium Dark Red",
        "High", "Medium Dark Green"
     );
     // capture values from other columns for the current row
     local:progression = Column( local:_dataTable, "Y" )[local:_firstRow];
     local:BMI = Column( local:_dataTable, "BMI" )[local:_firstRow];
  )
...
```

In the interface, Setup is called "JSL Variables" on the Textlets Markup tab.

Markup

`Markup()` specifies a rich text paragraph template based on a subset of HTML tags. Can contain JSL variable references enclosed in delimiters. The default delimiters are braces ({}).

```
...
Markup(
     "{local:gender} patient, {local:_age} years old, {local:BMI} BMI,
     has <font color='{local:color}'><b>{local:_Y Ordinal}</b></font>
     disease progression (<i>{local:progression}</i>)"
  )
```

In the interface, Markup is called "HTML Markup" on the Textlets Markup tab.

See *Using JMP* for more information about the supported font formatting tags.

Delimiters

`Delimiters()` specifies the delimiters that indicate variables for replacement in the Markup area. For example, you might use square bracket delimiters for JMP variables if the script contains C++ code.

```
Delimiters( "[]" )
```

In the interface, Delimiters is on the Textlets Other tab.

Width

`Width()` enables you to specify a preferred width in pixels for the hover label.

```
Width( 300 )
```

In the interface, Width is on the Textlets Other tab.

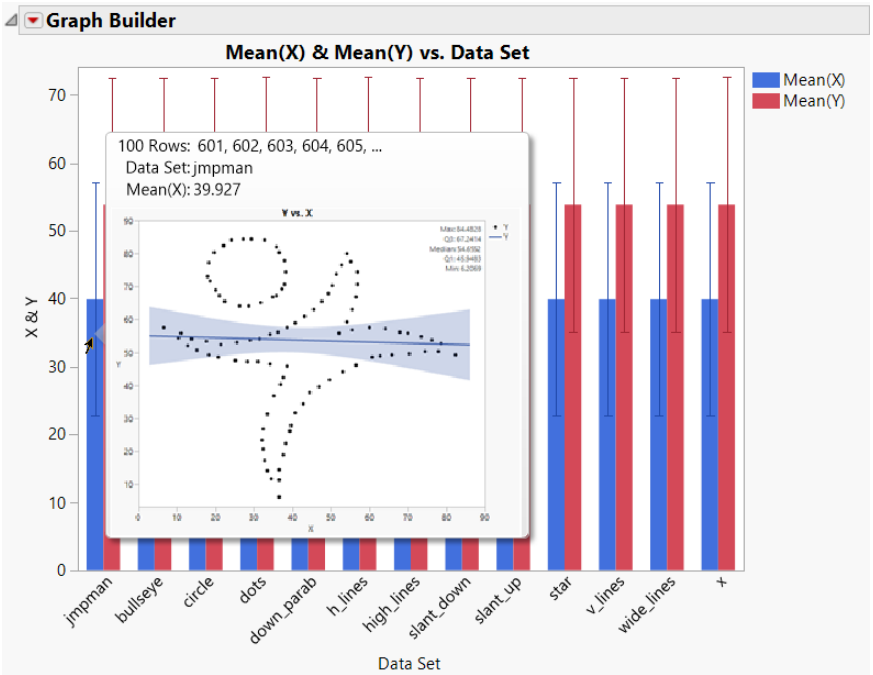
Add Graphs or Images to Hover Labels Using Graphlets

You can add a thumbnail of a related graph or an image to a hover label using a *graphlet*. These graphs and images can have actions associated with them that are triggered by a mouse click. For example, you can click a graph to launch it, or click an image to open a related link.

Example of a Graphlet in Graph Builder

Click the JMP man graph shown in Figure 12.40 to open it in Graph Builder and customize the graph. See *Using JMP* for details about opening a graph in a platform, customizing the graph, and then saving the updated graph in the original graph.

Figure 12.40 Graphlet Example



The following script creates the graph and hover label that are shown in Figure 12.40.

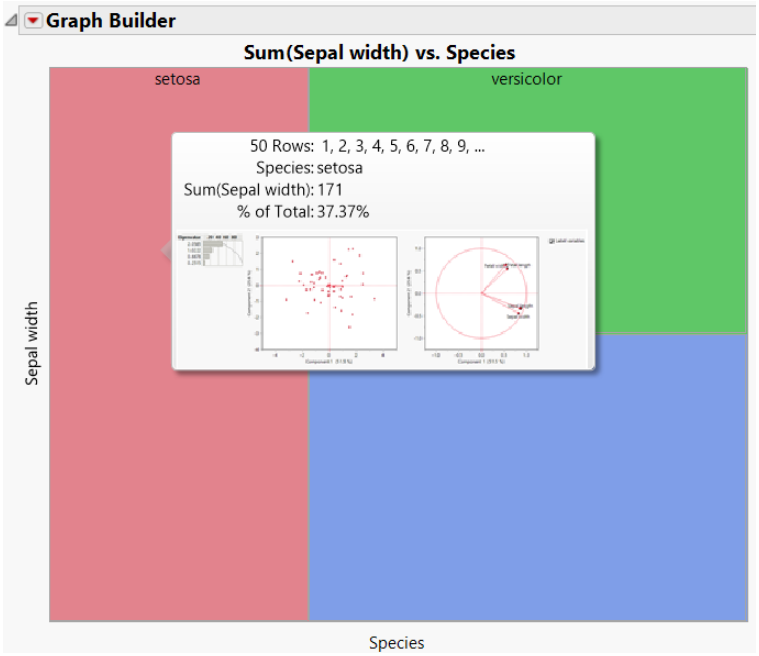
```
Names Default To Here( 1 );
// open the data table
dt = Open( "$SAMPLE_DATA/JMP Man Dozen.jmp" );

// launch the baseline visualization
gb = Graph Builder(
    Size( 529, 466 ),
    Show Control Panel( 0 ),
    Variables( X( :Data Set ), Y( :X ), Y( :Y, Position( 1 ) ) ),
    Elements(
        Bar(
            X,
            Y( 1 ),
            Y( 2 ),
            Legend( 12 ),
            Error Interval( "Standard Deviation" )
        )
    )
);

// get a reference to the visualization frame
```

```
frame = (gb << Report)[FrameBox(1)];

/* use the "Set Graphlet" command to define the visualization
that will be rendered inside the hover label using the
"Picture" attribute */
frame << Set Graphlet(
    /* The value of the "Picture" attribute is a JSL fragment
     that is evaluated to generate a thumbnail. The thumbnail is
     displayed inside the hover label. In this case, the hover
     label shows a scatterplot with a caption box. */
  Picture(
       Graph Builder(
           Show Control Panel( 0 ),
           Variables( X( :X ), Y( :Y ) ),
           Elements(
               Points( X, Y, Legend( 1 ) ),
               Line Of Fit( X, Y, Legend( 3 ) ),
               Caption Box( X, Y, Legend( 4 ),
                Summary Statistic( "5 Number Summary" ) )
           )
       )
    /* Because the optional "Click" attribute is not defined, the
        "Picture" attribute script is re-used to launch the associated
        platform when the user clicks the graphlet thumbnail. */
  )
);
```

Example of a Graphlet in Principal Components

Create a graphlet in the Principal Components platform. When the user clicks the hover label, the Principal Components report window appears. The user can then explore and filter the data.

Figure 12.41 Prinicipal Components Graphlet



The following script creates the graphlet shown in Figure 12.41:

```
dt = Open( "$SAMPLE_DATA/Iris.jmp" );
gb = dt << Graph Builder(
   Size( 531, 451 ),
   Show Control Panel( 0 ),
   Variables( X( :Species ), Y( :Sepal width ) ),
   Elements( Treemap( X, Y, Legend( 4 ) ) ),
   SendToReport(
      Dispatch(
         {},
         "Graph Builder",
         FrameBox,
         {Set Graphlet(
            Picture(
               Principal Components(
                  Y( :Sepal length, :Sepal width, :Petal length, :Petal width
   ),
                  Z( :Species ),
                  Estimation Method( "Default" ),
                  "on Correlations"
               )
            )
         )}
```

```
        )
    )
);
```

Graphlet Commands

Picture

A required script that adds a thumbnail of a graph or an image to the hover label.

`Picture()` is always executed in a Hover Label Execution Context derived from the current visual element. The first `Picture Box` (outside of a local data filter) found will be used to generate the displayed image.

If `Picture()` is a call to a platform, a local data filter is included.

```
...
Picture(
        Graph Builder(
            Show Control Panel( 0 ),
            Variables( X( :X ), Y( :Y ) ),
            Elements(
                Points( X, Y, Legend( 1 ) ),
                Line Of Fit( X, Y, Legend( 3 ) ),
                Caption Box( X, Y, Legend( 4 ),
                 Summary Statistic( "5 Number Summary" ) )
            )
        )...
```

See "Add Graphs or Images to Hover Labels Using Graphlets" for a complete example.

Click

An optional script for user-defined actions to be executed when the user clicks the associated image.

- Clicking runs the main graph's platform (for example, Graph Builder), which launches in its own window.
- If Click is not provided, the script provided in `Picture` is evaluated in a Hover Label Execution Context scope.

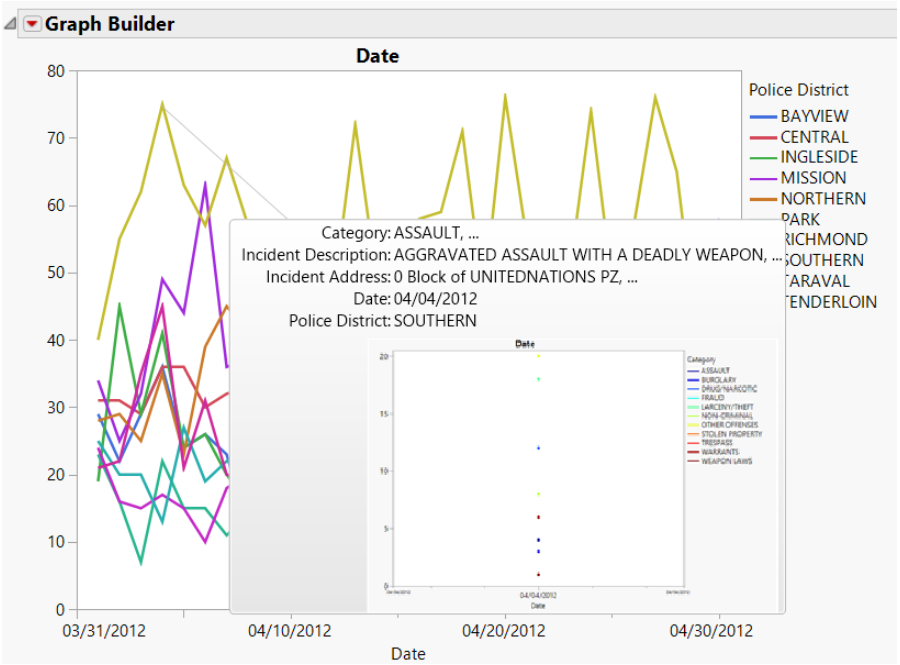
Skip Filters

An optional script. By default, any code executed in the context of a hover label customization (such as a graphlet) executes in a context defined in the hovered visual element (for example, a bar in a bar chart). This context includes a local data filter based on the current visual role assignment, and it is usually what you want. However, in some rare cases, you don't want the filter, or maybe part of the filter, so that the next graph generated by the graphlet can include more of the original data.

- If the Skip Filters field evaluates to a non-0 value (usually a positive integer such as 1) the selected Skip Filters Columns will be removed from the filter that will be applied to the new graphlet.
- Otherwise (for example, if it evaluates to 0), the selected columns are ignored and the filter is created as usual.
- If the Skip Filters Flag field is empty and no columns are selected in Skip Filters Columns (the default), then no columns are skipped when creating filters for the new window. Note that Skip Filters Flag is a JSL expression, which can be as complex as needed.
- If Skip Filters Flag field is not empty, but Skip Filter Columns is empty (no columns are selected), then all columns are filtered; the graphlet or new visualization includes all the data used by the original visualization. See "Example of Defining the Context of a Scatterplot Matrix and Ellipse".

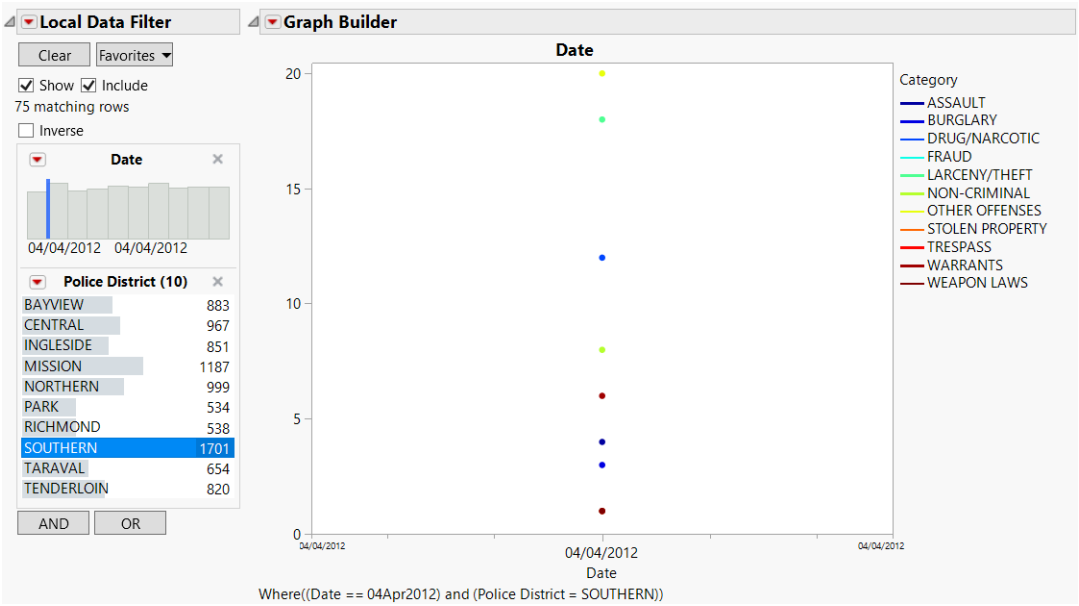
Suppose that you are looking at crime over time in the San Francisco Crime.jmp sample data table. You use a line chart where X = Date, an overlay of Police District, and no Y assignment. Lines represent counts over time per district. Now you want to use a graphlet to look at crime over time inside each district, broken down by category. Create this new graph and select **Save > to Clipboard** and then **Paste Graphlet** on the original graph to create the graphlet. However, the hovered graph shown in Figure 12.42 is not what you expect.

Figure 12.42 Initial Filtered Data



JMP created a filter based on both roles, X and the overlay. This is easier to see if you click on the graphlet to launch the new graph, and then expand the local data filter (on the left, collapsed by default). The filter on a single day causes the line chart to collapse into what looks like a marker plot as shown in Figure 12.43.

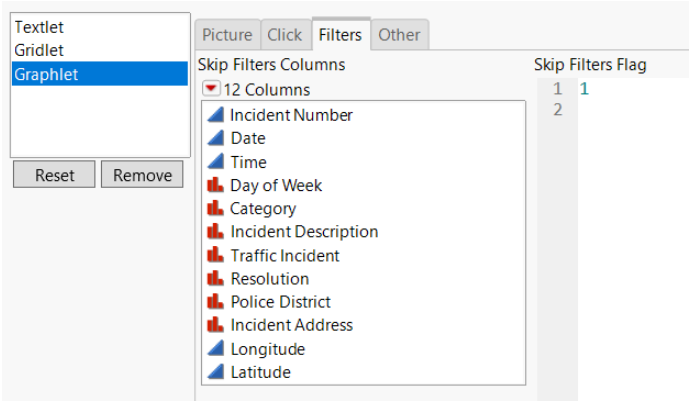
Figure 12.43 Local Data Filter with a Line Chart that Looks like a Marker Plot



You want the overlay value to restrict what you see in the new graph. However, you don't want the date value to do so. You still want to filter by the "SOUTHERN" Police District, but you do not want to filter on "04/14/2012" Date.

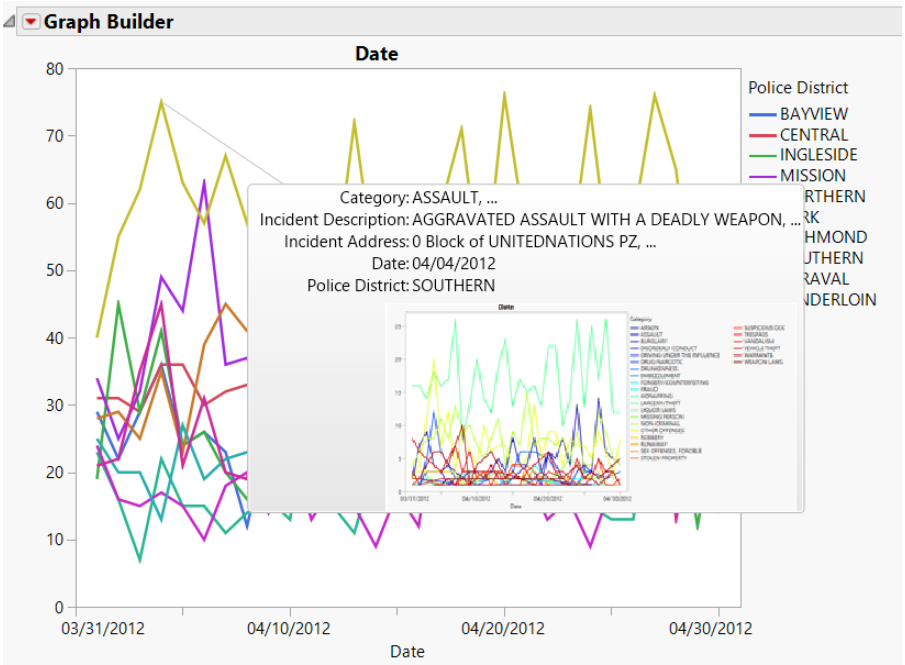
The solution is to use the Skip Filters Flag to remove the filter on Date from your original graph. On the Hover Label Editor, the script looks like this:

Figure 12.44 Example of Skip Filters Flag script in the Hover Label Editor



After the filter is removed, the end result is what you expect as shown in Figure 12.45.

Figure 12.45 Line Chart where Skip Filters Flag = 1



The following script creates the graph shown in Figure 12.45.

```
Names Default To Here( 1 );

dt = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );
gb = dt << Graph Builder(
   Size( 528, 456 ),
   Show Control Panel( 0 ),
   Variables( X( :Date ), Overlay( :Police District ) ),
   Elements( Line( X, Legend( 4 ) ) ),
   SendToReport(
      Dispatch(
         {},
         "Graph Builder",
         FrameBox,
         {Set Graphlet(
            Picture(
               Graph Builder(
                  Size( 528, 456 ),
                  Show Control Panel( 0 ),
                  Variables( X( :Date ), Overlay( :Category ) ),
                  Elements( Line( X, Legend( 1 ) ), Smoother( X, Legend( 2 ) ) )
               )
```

```
            ),
            Skip Filters( 1, {:Date} )
          )}
        )
      )
    );
```

The Skip Filters Flag field holds an expression that is evaluated in the hovered visual element context. If expression evaluates to a truthful value (usually a positive integer such as 1), the selected “Skip Filters Columns” will be removed from the filter that will be applied to the new graphlet. Otherwise (for example, if it evaluates to zero), the selected columns are ignored and the filter is created as usual.

Title

Title is an expression that evaluates to a string. The result is the title that appears on the title bar of the window that contains the graph. Variables defined in the Hover Label Evaluation Context can be used in the Title expression.

For example, the following expression indicates the baseline visualization type and how many drilling levels have been used so far:

```
Title("Graphlet for " || local:_displaySegName ||
       ", depth is " || Char( local:_drillDepth ))
```

If the Title expression is not specified, the new window is named “Graphlet”.

Note: In the JMP interface, Title is on the Hover Label Editor Other tab of a graphlet.

Reapply

Reapply is an expression that indicates whether the graphlet definition should be reapplied to new graphs or platforms created by clicking on a graphlet image. This is the core mechanism behind the ability to drill down using graphlets.

```
Reapply( local:_underlyingRows > 1 )
```

Note: In the JMP interface, Reapply is on the Hover Label Editor “Other” tab of a graphlet.

If Reapply is empty (the default setting), the graphlet definition is not reapplied. New windows launched by clicking on this graphlet doesn’t show graphlets on its hover labels.

Update Names and Values in Hover Labels Using Gridlets

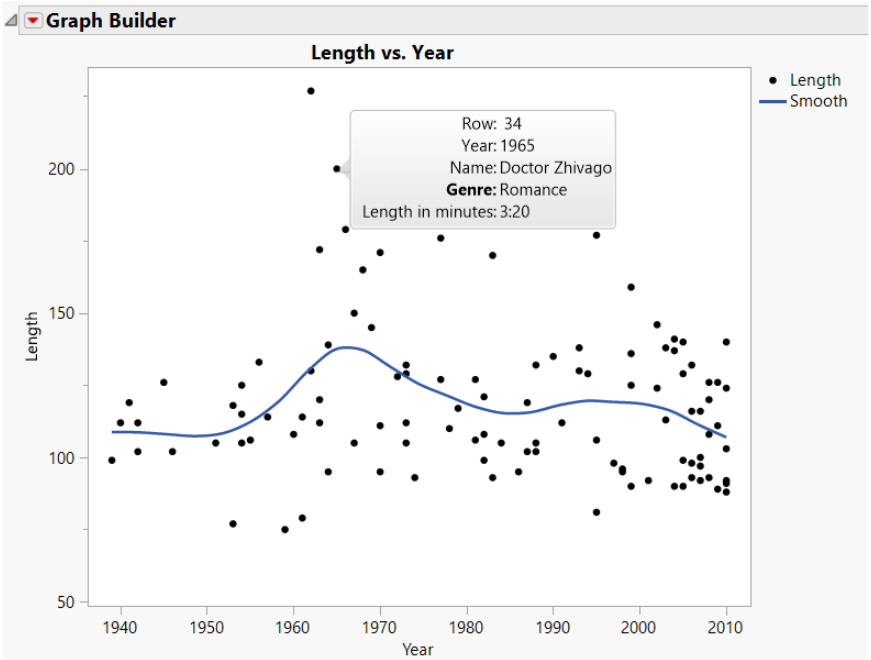
A *gridlet* supports the full customization of the hover label name and value pairs that reflect the underlying visual data. Data can be added, removed, or renamed. You can also change the style and numeric formats.

When you hover over a visualization in a graph, you see content in the hover label for that marker. A grid entry is text that appears on the hover label by default, either from the label columns or from the columns associated with the visualization. By default, the column names that are assigned to roles (and their corresponding values) are shown, along with columns that are labeled. You can customize these column names or values or add other content. Here are a few things you can do:

- add or remove a column name and its value
- rename a grid label
- add a new grid entry with a dynamic value and an optional click action. For example, you can insert an entry named “See also” with a clickable link to a Wikipedia page as the value.
- applying a style to text (for example, formatting text in bold)
- formatting numeric values (for example, setting the number of decimal places to 2)

Figure 12.46 shows adding Genre, Name, and Length in minutes, reformatting the minutes, and formatting “Genre” in bold.

Figure 12.46 Customized Gridlet



Example of Customizing a Gridlet

The following script creates the graph and hover label shown in Figure 12.46.

```
Names Default To Here( 1 );
dt = Open( "$SAMPLE_DATA/Movie Inventory.jmp" );

// launch the baseline visualization
dt << Graph Builder(
   Size( 534, 456 ),
   Show Control Panel( 0 ),
   Variables( X( :Year ), Y( :Length ) ),
   Elements( Points( X, Y, Legend( 3 ) ), Smoother( X, Y, Legend( 4 ) ) ),
   SendToReport(
      Dispatch(
         {},
         "Graph Builder",
         FrameBox,
         {Set Gridlet(
            // delete the Length column
            Expunge( {{Matcher( "Length" )}} ),
            // add the Name, Genre, and Length in minutes columns
            Annex(
               {{Matcher( "Name" ), value( :Name[local:_firstRow] )},
               {Matcher( "Genre" ), value( :Genre[local:_firstRow] )},
               {Matcher( "Length in minutes" ), value( local:_Length * 60 )}}
            ),
            Reformat(
            // reformat the Length in minutes format
               {{Matcher( "Length in minutes" ), Format( "hr:m", 80 ), 80}}
            ),
            Style(
            // format "Genre" in bold nine-point text
               {{Matcher( "Genre", Scope( Name ) ), Font("Segoe UI", 9,
                  "Bold")
               }}
            )
         ), Add Pin Annotation(
            Seg( Marker Seg( 1 ) ),
            Index( 33 ),
            Index Row( 33 ),
            UniqueID( 610643713 ),
            FoundPt( {508, 199} ),
            Origin( {1964.8951552795, 199.00456244898} ),
            Tag Line( 1 )
         )}
      )
```

```
    )
);
```

Gridlet Commands

The first argument in all `Set Gridlet` commands is a matcher, which identifies its scope of action (the grid entries it affects). A matcher can refer to the following:

- A single grid entry: by name, matching against a string. A grid entry is text that appears on the hover label by default, either from the label columns or from the columns associated with the visualization.
- Multiple grid entries: by column reference, regular expression or using the argument `All`.

For example, a grid entry for Mean(height) can be matched against the following:

- the name `"Mean(height)"n`
- the column reference `:height`
- the regular expression `PatReg("^.*\(height\)$")`
- the special argument `All`, which always matches all rows in the grid.

If there is a match, the command-associated action is carried against the matching entries using the arguments.

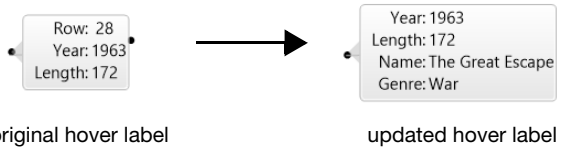
The exception is the `Annex` command, which adds a new entry to the grid if there are no matches with existing entries (to avoid unplanned naming collisions).

Example of Adding and Deleting Columns

This example describes how to use the `Expunge` command to remove the first entry added to the hover label when the underlying table has no Label columns, followed by two instances of the `Annex` command to add new rows. Note that you could have used a single `Annex` instance and passed to it a list of lists instead.

Figure 12.47 shows the original hover label and the updated hover label. Notice that Row was removed from the label. Name and Genre were added.

Figure 12.47 Example of the Expunge and Annex Commands



The following script creates the updated hover label shown in Figure 12.47.

```
Names Default To Here( 1 );
```

```
dt = Open( "$SAMPLE_DATA/Movie Inventory.jmp" );

// create the baseline visualization
gb = dt << Graph Builder(
        Size( 526, 454 ),
        Show Control Panel( 0 ),
        Variables( X( :Year ), Y( :Length ) ),
        Elements( Points( X, Y, Legend( 3 ) ), Smoother( X, Y, Legend( 4 ) ) )
    );

rpt = gb << Report();
box = rpt[Framebox( 1 )];
box << Set Gridlet(
     // delete Row
      Expunge( {{Matcher( 1 )}} ),
      Annex(
        // add the Name and Genre columns to the hover label
        {{Matcher( "Name" ), value( :Name[local:_firstRow] )},
         {Matcher( "Genre" ), value( :Genre[local:_firstRow] )}}
      )
);
```

Annex (Append)

`Annex({`*`matcher`*`, `*`value`*`(<script>), <Click(<script>)>})` adds a new row to the hover label grid. The `Annex` command corresponds to the Append tab in the Hover Label Editor Gridlet panel.

This command takes three arguments:

- `Matcher`: The new name. `Matcher` should not match existing names.
- `Value`: Required. The script is evaluated in the Hover Label Execution Context. The result is the value that appears for the new row. See "Work with the Hover Label Execution Context" for details.

  ```
  box << Set Gridlet(
         Annex( { "LDL, HDL", value(
         Char(:LDL[local:_firstRow]) || ", " || Char(:HDL[local:_firstRow])
         ) })
     );
  ```

- `Click`: Optional. `Click` is evaluated in the Hover Label Execution Context when the user clicks on annexed entry value. This option is equal to the `Click` attribute in graphlets. If present, the value portion of the new entry is rendered using a link style (blue text color with an underline) to indicate that it can be clicked.

Expunge (Delete)

`Expunge({Matcher})` deletes the matcher. The `Expunge` command corresponds to the Delete tab in the Hover Label Editor Gridlet panel.

All matching entries are removed from the grid.

```
box << Set Gridlet( Expunge( {"age"} ) );
```

Rename

`Rename({`*`Matcher`*`, Value(<string>)})` replaces the name portion of the matched entry or entries with the given value. The `Rename` command corresponds to the Rename tab in the Hover Label Editor Gridlet panel.

```
box << Set Gridlet( Rename( {Matcher( "name" ), Value( "students" )} ), );
```

Reformat

`Reformat({`*`Matcher`*`, <`*`format`*`>})` takes a matcher and an argument that specifies a format to be applied to the matched row (or rows) grid numerical value or values. Matching entries are changed to reflect the specified format.

```
box << Set Gridlet( Reformat( {:height, Format( "Scientific", 10, 4 )} ) );
```

Style

`Style({`*`Matcher`*`, <`*`style1`*`>, <`*`style2`*`>, ...})` takes a matcher and one or more style arguments. Here are the supported styles:

- Background Color
- Text Color
- Font (which further supports font family, size and style)
- Justification (left, center, or right)
- Matching entries are changed to reflect the specified styles.

```
box << Set Gridlet(
   Style(
      {{Matcher( "sex" ), Text Color( "Red" ), Font("Papyrus", 14,
      "Bold Underline")}}
   )
);
```

Create Drill-Down Graphs

A drill-down graph is created when you click on the thumbnail. Clicking on the hover graph opens it in a new window, enabling you to see the graph in more detail and also make modifications such as changing the data filter. For example, click a cell on a treemap to see details about the data in a new window (Figure 12.48). Press Ctrl and click to replace the contents of the last launched window instead of opening a new window.

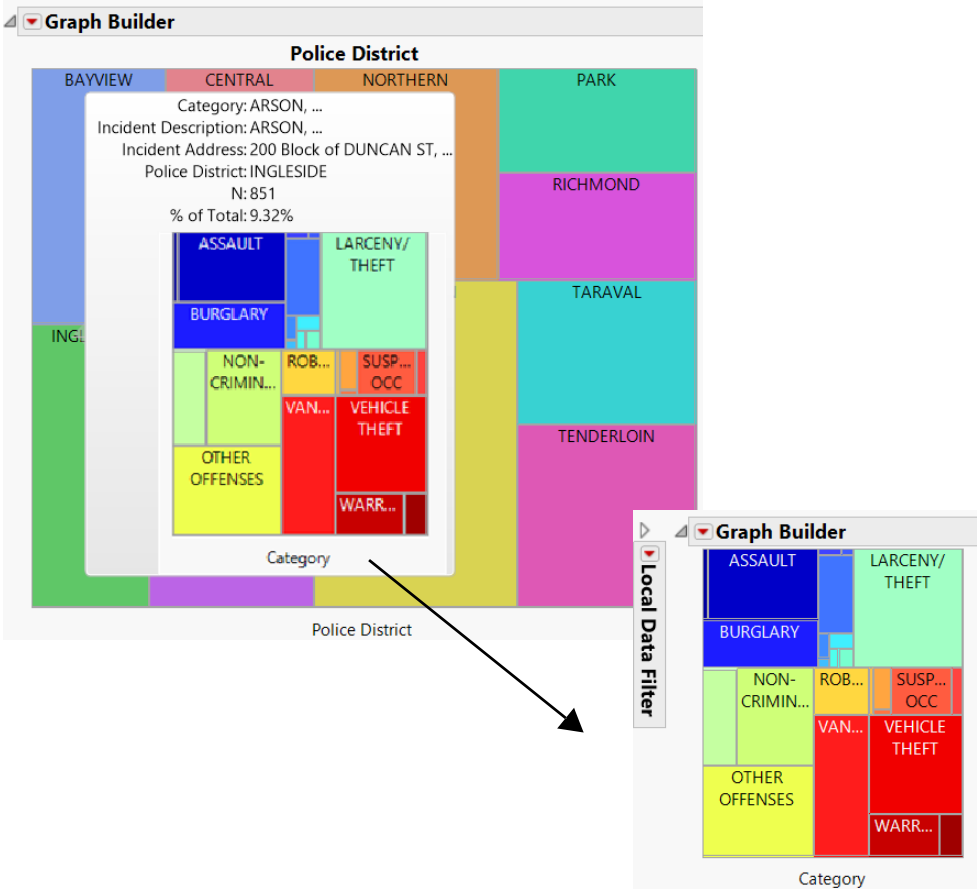
- The Next in Hierarchy column property enables you to define the behavior of categorical drill downs in preset graphs. This column property defines only one level of the drill down, from a category in a grouping role in the current visualization to the category in the grouping role in the next one, launched by the graphlet.
- If the Next in Hierarchy column property is not defined, the list of categorical or nominal columns from the data table that have not been added to a filter is used.

Notes:

- The Next in Hierarchy column property is not supported in graphlets defined using Paste Graphlet. See *Using JMP* for details about the Paste Graphlet command.
- If there is no measurement, JMP tries to get one from the data table and uses Count (or no measurement) if there are no numeric columns in the data table.

Figure 12.48 shows drilling-down on a cell in a treemap.

Figure 12.48 Example of a Drill-Down Graph on a Hover Label



The following script creates the simplified versions of the treemap presets shown in Figure 12.48.

```
Names Default To Here( 1 );

/* traverse the given list of column names, defining the Next in Hierarchy
   column property so that it creates a chain to enable the drill down */
setHierarchy = Function( {dt, cols},
   {Default Local},
   nCols = N Items( cols );
   If( nCols < 2,
      Write( "\!NsetHierarchy: Insufficient number of columns!" ),
      col = Column( dt, cols[1] );
      For( i = 2, i <= nCols, i++,
```

```
            nextCol = cols[i];
            Try(
               Column( dt, nextCol ),
               Write( "Invalid column name: " || nextCol );
               Throw();
            );
            // Eval() is needed for now
            col << Set Property( "Next in Hierarchy", Eval( nextCol ) );
            nih_name = col << Get Property( "Next in Hierarchy" );
            col = Column( dt, nextCol );
         );
      );
   );

   // open San Francisco Data.jmp
   dt = Open( "$SAMPLE_DATA/San Francisco Crime.jmp" );

   // add a Next in Hierarchy chain to the data table
   setHierarchy( dt, {"Police District", "Category", "Incident Description"} );

   /* Graph Builder treemap based on the first level of our hierarchy, "Police
      District" */
   gb = dt << Graph Builder(
      Size( 495, 440 ),
      Show Control Panel( 0 ),
      Variables( X( :Police District ) ),
      Elements( Treemap( X ) )
   );

   // get a handle to the graph framebox
   rpt = gb << Report;
   frame = rpt[Framebox( 1 )];

   /* configure the graphlet -- a hover label extension that adds a visualization
      thumbnail to the hover label. Clicking on the thumbnail launches the
      associated visualization in its own window (or replaces the graph in the
      current window on a Ctrl-click). */
   frame << Set Graphlet(
      // define the thumbnail and launch treemap
      Picture(
         // next-level grouping is retrieved from the Next in Hierarchy property
         local:propColName = local:_groupings[1] << GetProperty( "Next In
      Hierarchy" );
         local:nextGrouping = Column( local:_dataTable, local:propColName );

         // define the Graph Builder graph for the thumbnail
```

```
      Graph Builder(
         Size( 200, 250 ),
         Show Legend( 0 ),
         Show Title( 0 ),
         Show Footer( 0 ),
         Show Control Panel( 0 ),
         Variables( X( local:nextGrouping ) ),
         Elements( Treemap( X ) )
      );
   )
);

// add a pinned annotation to display the graphlet
frame << Add Pin Annotation(
   Seg( TreeMapSeg( 1 ) ),
   Index( 2 ),
   Index Row( 7033 ),
   UniqueID( 581248018 ),
   FoundPt( {52, 248} ),
   Origin( {27.6544421487603, 200.14987244898} ),
   Tag Line( 1 )
);
```

Work with the Hover Label Execution Context

Hover Label Execution Context variables define the environment in which your graphlet and textlet scripts are executed.

Local Data Filter

When defining a graphlet, you specify how to visualize the data that appears as a thumbnail. But how do you specify which rows should be displayed?

This information is specified indirectly you hover over different visual elements of the baseline graph (the graph to which a graphlet definition was added). Suppose that the baseline graph is a bar chart. Hovering over one of the bars indicates that all the rows behind that bar (and only those rows) should be included in the generation of the visualization. This visualization shows up as a thumbnail on the hover label. In other words, the hovered visual element provides the data context in which the visualization is generated.

This data context is expressed in a where clause that is computed by JMP based on:

- the type of graph used in the baseline visualization
- the columns assigned to its different roles
- the current values for the particular visual element being inspected

The resulting where clause is then used to create a local data filter that is assigned to the platform specified in the graphlet definition.

The following graph roles are supported as sources for the creation of local data filters:

- X, including nested columns
- Y, if not representing a summary statistic or other calculation
- Group X
- Group Y
- Wrap
- Page
- overlays

Numeric columns are used in comparison to a single value or checked for inclusion in a range, depending on the underlying visualization type. Visualizations that group numeric data in bins such as histograms and heat maps (and the box in box plots) result in the use of corresponding ranges in the where clause.

If present, the where clause of a Local Data Filter associated with the baseline visualization is captured as well.

Local Scope Context

The JSL expressions that configure different aspects of a graphlet and other hover label extensions (such as the textlet `Setup` and the gridlet `Append` and `Click` scripts) are executed inside a local namespace. This namespace is initialized with local variables that reveal information from the execution context: the hover label grid entries and references to the underlying visual element and associated data. These variables are used extensively by the graphlet presets.

Figure 12.4 describes variables that are provided in a graphlet local namespace. See "Complete Example of the Hover Label Execution Context Variables".

Note: To see the variables at work, right-click on a graph, select **Hover Label Editor**, select the Graphlet panel, and then select one of the presets. The code that appears in the Picture field illustrates how many of the Hover Label Excecution Context variables are used to control the graphlet's behavior.

Table 12.4 Hover Label Execution Context Variables

Variable	Description

`_mode`	A string indicating the current evaluation context. Values include `Picture`, `Click`, `Textlet`, `Graphlet`, and `Other`. You can use `_mode` to customize a Graph Builder expression to show more or less information.
	From the JMP log:
	`{"_mode", "Textlet"},`
	Example from the Bar preset:
	`If( local:_mode == "Picture",`
	`  local:sizeDef = "Size( 200, 250 ),";`
	`  local:l = 0;`
	`,`
	`  local:sizeDef = "";`
	`  local:l = 1;`
	`);`
	Note: `Other` means the other attributes that take expressions (for example, `SkipFiltersFlag`).

`_groupings`	A list of the current grouping (X or overlay roles) data column references. See "Example of Defining the Context of a Scatterplot Matrix and Ellipse". `{"_groupings", {:sex}}`
`_measurements`	A list of the current measurements (Y role) data column references. `{"_measurements", {:height}}`
`_filters`	A list of the current local data filter data column references. `{"_filters", {:sex}}`

Table 12.4 Hover Label Execution Context Variables *(Continued)*

Variable	Description

_where	A textual representation of the where clause associated with the hovered visual element. The expression is parsed to be used with JSL statements that take where clauses such as `Get Rows Where`. `_where` is useful to cache complex calculations that are based on context.
	`expr = Parse( local:_where );`
	`r = local:_dataTable << Get Rows Where( expr );`
	`ss = dt << Subset( Private, Invisible, Rows( r ) );`
	`Summarize( ss, exg = By( :sex ), exm = Mean( :height ) );`
	Another example:
	`{"_where", ":sex == \!"F\!""}`

_whereExpr	The where clause that is associated with the hovered visual element as a JSL expression.
	`local:_h = Round(Mean( :height[local:_dataTable << Get Rows Where( local:_whereExpr )] ), 2);`
	Another example:
	`{"_whereExpr", :sex == "F"}`

_displaySegName	The class name of the underlying seg. Useful when creating descriptive text or in the definition of graph-dependent drill-down logic.
	`Title("Graphlet for " || local:_displaySegName || ", depth is " || Char( local:_drillDepth ))`
	Another example:
	`{"_displaySegName", "BarSeg"}`

_dataTable	A reference to the underlying data table. Useful for creating subsets. See "Add Rich Text to Hover Labels Using Textlets" for an example. `{"_dataTable", Data Table( "Big Class" )}`

Table 12.4 Hover Label Execution Context Variables *(Continued)*

Variable	Description
`_summaryStatistic`	A string with the current summary statistic. Useful for passing to new graphs in a drill down. `{"_summaryStatistic", "Mean"}`
`_drillDepth`	An integer that indicates how many drill-down levels have been opened before. Useful to limit drill-down chains or to control `SkipFilters` logic. See "Example of Defining the Context of a Scatterplot Matrix and Ellipse" `{"_drillDepth", 1}`
`_firstRow`	The 1-based index of the first row associated with the hovered visual element. See "Example of Adding and Deleting Columns". `{"_firstRow", 1}`
`_underlyingRows`	The count of rows associated with the visual element that appears in the hover label. You can use this variable to decide the type of the visualization based on the number of rows to display, going from the low granularity of a heat map to a dense marker plot if the number of rows is small enough (and vice versa). See "Reapply" for an example. `{"_underlyingRows", 18}`

In addition to these variables, a variable is defined for each of the hover label grid entries. The name of this variable is taken for the grid entry name, with a '_' prefix.

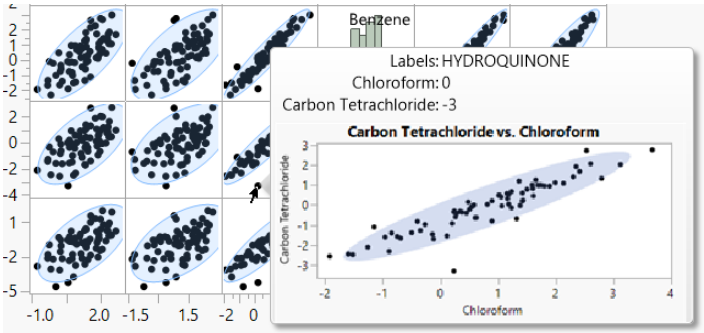
Note: See "Example of Defining the Context of a Scatterplot Matrix and Ellipse" for examples of these variables.

As an example, using Hover Label Execution Context variables, you can create a scatterplot matrix, hover over a point to show a close-up ellipse, and then click the ellipse to modify it in Graph Builder. The graphlet shows the same visualization as the one in the underlying scatterplot matrix cell (a scatterplot of the two columns associated with that cell).

Clicking on the thumbnail then has the effect of "extracting" the cell from the matrix and opening it up as an independent visualization.

The script shown in "Example of Defining the Context of a Scatterplot Matrix and Ellipse" enables the user to extract the matrix cells in the main scatterplot matrix.

Figure 12.49 Viewing an Ellipse in a Scatterplot Matrix



Example of Defining the Context of a Scatterplot Matrix and Ellipse

The following script shows an example of using the context in which the script runs. This script creates the scatterplot matrix shown in Figure 12.49.

```
Names Default To Here( 1 );

dt = Open( "$SAMPLE_DATA/Solubility.jmp" );
win = New Window( "Scatterplot Matrix Graphlet Example",
   plt = dt << Multivariate(
      Y(
         "1-Octanol"n,
         :Ether,
         :Chloroform,
         :Benzene,
         :Carbon Tetrachloride,
         :Hexane
      ),
      Estimation Method( "Row-wise" ),
      Scatterplot Matrix(
         Density Ellipses( 1 ),
         Shaded Ellipses( 1 ),
         Ellipse Color( 69 ),
         Horizontal( 1 )
      )
   )
);
frame = win << xpath( "//FrameBox" );
frame << Set Graphlet(
   Picture(
      Show( local:_displaySegName );
```

```
        If( local:_displaySegName == "MarkerSeg",
           // scatterplot cells
           Graph Builder(
              Size( 400, 200 ),
              Show Control Panel( 0 ),
              Show Legend( 0 ),
              Variables(
                 X( local:_groupings[1] ),
                 Y( local:_groupings[2] )
              ),
              Elements( Points( X, Y ), Ellipse( X, Y ) )
           ),
           // else histograms in matrix diagonal
           If( local:_N > 5, /* Note: only present if baseline graph
              is a histogram.*/
              Graph Builder(
                 Size( 400, 400 ),
                 Show Control Panel( 0 ),
                 Show Legend( 0 ),
                 Variables( X( local:_groupings[1] ) ),
                 Elements( Histogram( X ) )
              ),
              Graph Builder(
                 Size( 400, 400 ),
                 Show Control Panel( 0 ),
                 Show Legend( 0 ),
                 Variables( X( local:_groupings[1] ) ),
                 Elements( Points( X ) )
              )
           )
        );
     ),
     // skip filters if you are at the first drill level -- the original graph
     SkipFilters( local:_drillDepth == 1 ), // no list -- all filters
     Title( "Graphlet" ),
     Reapply( local:_displaySegName == "HistSeg" & local:_N > 5 )
  );
```

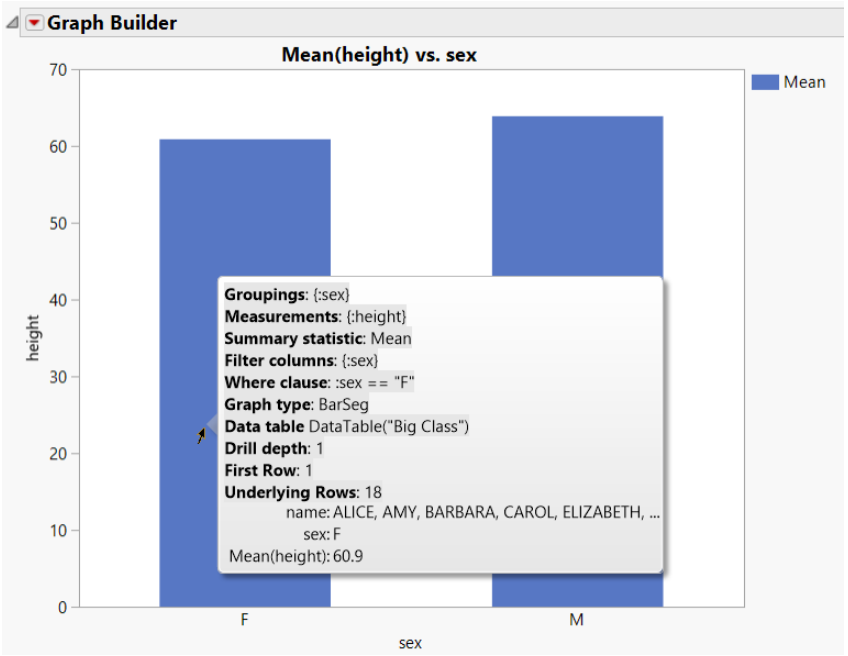
Complete Example of the Hover Label Execution Context Variables

The following script produces the graph shown in Figure 12.50. When you hover over a bar, the hover label appears. Details about the data and the Hover Label Context Execution variables appear in the log.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Graph Builder(
   Size( 528, 450 ),
```

```
   Show Control Panel( 0 ),
   Variables( X( :sex ), Y( :height ) ),
   Elements( Bar( X, Y, Legend( 3 ) ) ),
   SendToReport(
      Dispatch(
         {},
         "Graph Builder",
         FrameBox,
         {Set Textlet(
            Setup(
               local:encodedWhere = XML Encode( local:_where );
               /* Show() provides additional information about the
               HLEC variables in the log */
               Show( Namespace( "local" ) << Get Contents() );
            ),
            Markup(
               "
<b>Groupings</b>: {local:_groupings}
<b>Measurements</b>: {local:_measurements}
<b>Summary statistic</b>: {local:_summaryStatistic}
<b>Filter columns</b>: {local:_filters}
<b>Where clause</b>: {local:encodedWhere}
<b>Graph type</b>: {local:_displaySegName}
<b>Data table</b> {local:_dataTable}
<b>Drill depth</b>: {local:_drillDepth}
<b>First Row</b>: {local:_firstRow}
<b>Underlying Rows</b>: {local:_underlyingRows}
"
            )
         )}
      )
   )
);
```

Figure 12.50 Bar Chart with Textlet



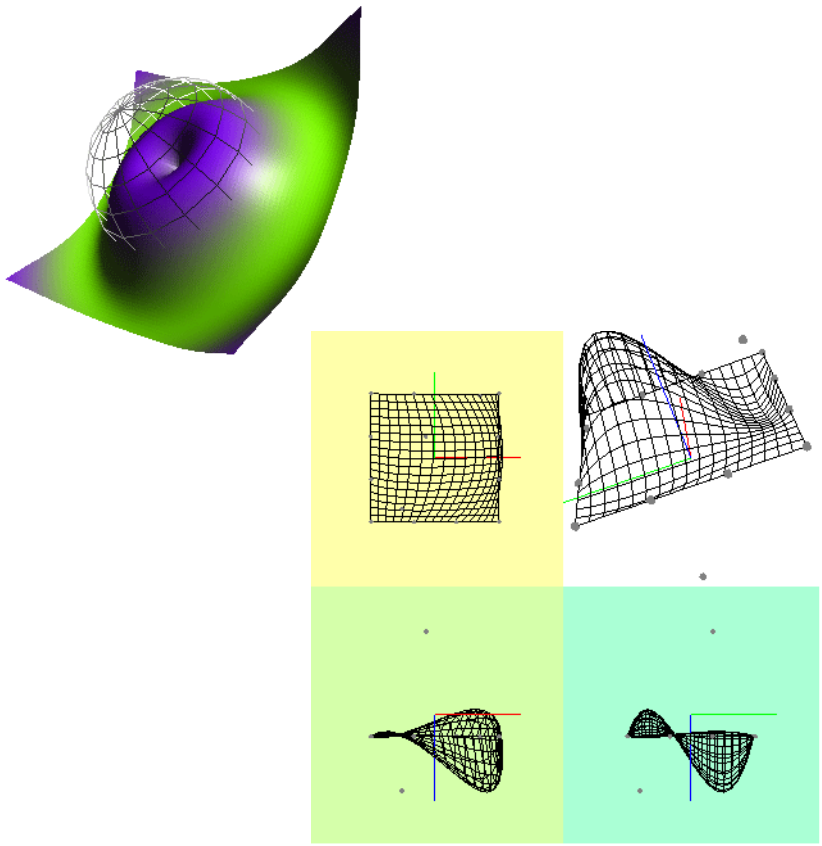
Chapter 13

Three-Dimensional Scenes

Scripting in Three Dimensions

JSL includes commands for scripting three-dimensional scenes derived from OpenGL. Although not a complete OpenGL® implementation, JSL's 3-D scene commands enable complex, interactive plots to be constructed and viewed. The Surface Plot platform in JMP is built using JSL scene commands.

Figure 13.1 Sample Three-Dimensional Shapes



Contents

About JSL 3-D Scenes . . . 751
JSL 3-D Scene Boxes . . . 751
Setting the Viewing Space . . . 754
Setting Up a Perspective Scene . . . 755
Setting up an Orthographic Scene . . . 756
Changing the View . . . 757
The Translate Command . . . 757
The Rotate Command . . . 757
The Look At Command . . . 759
The ArcBall . . . 760
Graphics Primitives . . . 761
Primitives Example . . . 764
Controlling the Appearance of Primitives . . . 765
Other Uses of Begin and End . . . 771
Drawing Spheres, Cylinders, and Disks . . . 771
Construction . . . 771
Lighting . . . 772
Drawing Text . . . 773
Using Text with Rotate and Translate . . . 773
Using the Matrix Stack . . . 774
Lighting and Normals . . . 778
Creating Light Sources . . . 778
Lighting Models . . . 780
Normal Vectors . . . 781
Shading Model . . . 781
Material Properties . . . 782
Alpha Blending . . . 783
Fog . . . 783
Example . . . 783
Bézier Curves . . . 784
One-Dimensional Evaluators . . . 785
Two-Dimensional Evaluators . . . 787
Using the Mouse . . . 788
Pick Commands . . . 789
Three-Dimensional Arguments . . . 790
Improve the Display of Graphics . . . 790

About JSL 3-D Scenes

JMP's 3-D scene language is built on top of the OpenGL API, extending, replacing, and leaving out various parts of the OpenGL API. As such, the 3-D scene language is not certified or licensed by Silicon Graphics, Inc. under the OpenGL API.

This chapter documents JMP's JSL commands for creating 3-D scenes but is not a tutorial on OpenGL programming. If you are not familiar with OpenGL programming, you might want supplemental material. If you are familiar with OpenGL programming, you still need to read this chapter because some items are nonstandard.

JMP is shipped with sample files in the Scene3D subfolder of Sample Scripts to get you started and give you some ideas. Some of the example scripts are similar to some of the examples in this chapter; some are almost complete applications.

The website https://opengl.org is a good source for information, as is your favorite search engine.

JMP's Scene 3-D language does some work for you that the OpenGL API requires you to do for yourself. JMP makes text easy, gives you a built-in arcball controller, and makes sure the matrix operations that belong on the model view stack and projection stack go on their respective stacks. JMP uses its own display list manager so that your scenes can be journaled and played back later, and provides a select mechanism that calls back to your JSL code to tell you what object in your scene is under the mouse, with almost no extra effort on your part. At this time, JMP does not provide access to some features, like texturing.

JSL 3-D Scene Boxes

These commands are necessary to set up and configure a 3-D scene.

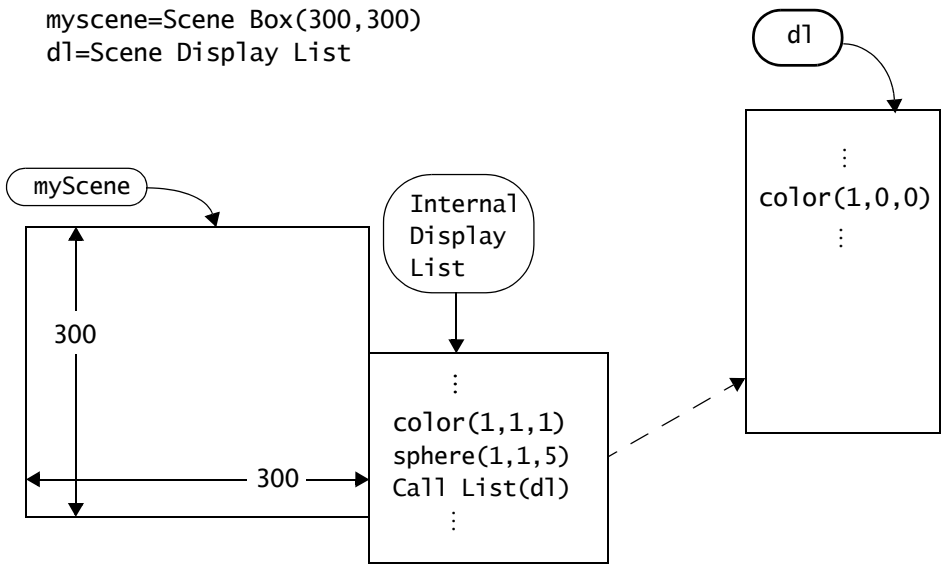
Like all displays in JMP (detailed in "Display Trees"), 3-D scenes must be placed in a display box (in this case, a Scene Box). This box is then placed in a window. Therefore, a simple 3-D scene script has the following form.

```
myScene = Scene Box(300, 300); // create a 300 by 300 pixel scene box
...(commands to set up the scene)...
New Window("3-D Scene", myScene); // draw the scene in a window
...(commands that manipulate the scene)
```

The scene can be sent messages that construct elements in the scene. Typical messages alter the viewer's vantage point, construct physical elements in the scene itself, or manipulate lights and textures. These messages are maintained in a display list and are manipulated in one of two ways:

- They are sent as messages to the scene, which immediately adds them to the scene's internal display list.
- They are sent as messages to a display list stored in a global variable, which is called by the scene's display list later.

Figure 13.2 Constructing Scenes



For example, of commands sent to the scene's display list directly, consider the following small script. Each of the commands is explained in detail later in the chapter.

```
scene = Scene Box( 400, 400 ); // make a scene box
New Window( "Example 1", scene ); // put the scene in a window
scene << Perspective( 45, 3, 7 ); // define the camera
scene << Translate( 0.0, 0.0, -4.5 ); // move to (0,0,-4.5) to draw
scene << Color( 1, 0, 0 ); // set the RGB color of the text
scene << Text( center, baseline, 0.2, "Hello, World." ); // add text
scene << Update; // update the scene
```

The first two lines create a scene and place it in a window. The `Perspective` command defines the viewing angle and field depth. By sending it as a message to the scene, it is immediately added to the scene's display list. Because the "Hello World" text is drawn at the origin (0, 0, 0), the `Translate` command is added to the display list. This command moves the camera back a bit so that the origin is in the field of vision. The color is set to red with the `Color` command, the text is drawn, and the `Update` command causes the scene to be rendered (in other words, causes the display list that contains the commands to be drawn.)

Figure 13.3 Hello World

Equivalently, the commands to construct the display can be accumulated in a display list stored in a global variable, which is then sent to the scene all at once. To define a global variable as a display list, assign it using the `Scene Display List` function. For example, to use the global `greeting` as a display list, issue the command

```
greeting=Scene Display List();
```

Display commands can then be sent as messages to `greeting`. An equivalent "Hello World" example using a display list follows.

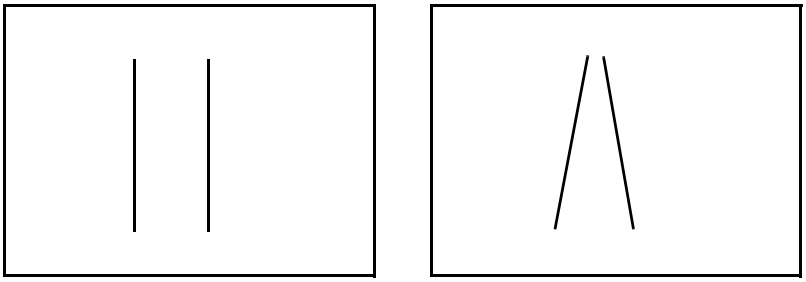
```
greeting = Scene Display List();
greeting << Color( 1, 0, 0 ); // set the RGB color of the text
greeting << Text( center, baseline, 0.2, "Hello, World." ); // add text

// draw the window and send it the stored display list
scene = Scene Box( 400, 400 ); // make a scene box
New Window( "Example 1", scene ); // put the scene in a window
scene << Perspective( 45, 3, 7 ); // define the camera
scene << Translate( 0.0, 0.0, -4.5 ); // move to (0,0,-4.5) to draw
scene << Call List( greeting ); // send the display list to the scene
scene << Update; // update the scene
```

Note which commands were separated into the display list, and which were applied to the scene directly. Those that manipulate the camera (`Translate` and `Rotate`) are applied to the scene. Those that define the object (`Color` and `Text`) were relegated to the display list. This is done so that the display list can be called many times to replicate the object at different positions.

Setting the Viewing Space

3-D scenes can be rendered in two ways. *Orthographic* projections place the elements in a box, where coordinates are not changed to accommodate the perspective of the viewer. *Perspective* projections modify the display to simulate the position of the elements in relation to the position of the viewer. For example, two parallel lines (like railroad tracks) stay parallel in orthographic projections, but seem to connect at a distance in perspective projections.

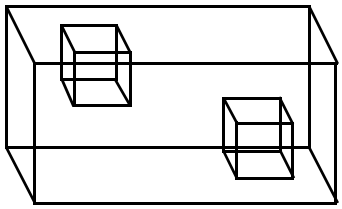
Figure 13.4 Parallel Lines in an Orthographic Projection (left) and a Perspective Projection (right)



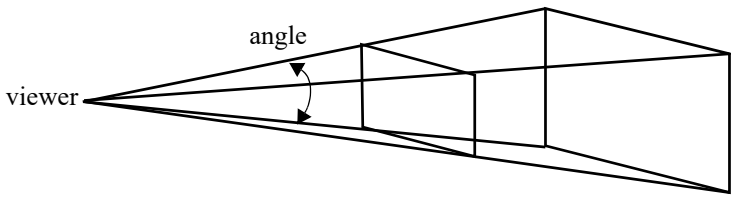
As another example, imagine looking at a tube edge-on (like a telescope). In an orthographic projection, the tube would appear as a thin circle. In a perspective projection, the circle would have a thickness; the hole at the far end of the tube would appear smaller than the close hole, and the interior of the tube is visible.

Therefore, the viewable space of an orthographic projection is a rectangular shape. That of a perspective projection is the frustum of a pyramid (that is, a pyramid whose top has been sliced off).

Figure 13.5 Comparing Projections



Orthographic Scene



Perspective Scene

In general, perspective projections give a more realistic view of the world, since it mimics the way an eye or a camera sees. Orthographic projections are important when it is essential to preserve dimensions, such as an architectural CAD program.

Setting Up a Perspective Scene

To set up a perspective scene in JSL, send the `Perspective` command to a display list.

```
Perspective (angle, near, far)
```

where *`angle`* is the viewing angle, *`near`* is the distance to the near plane, and *`far`* is the distance to the far plane, as illustrated in the drawing above. A couple of things need to be remembered when defining the viewing space.

- Items outside the viewing space (for example, closer than the *`near`* plane or farther than the *`far`* plane) are not drawn. They are clipped off.
- The ratio of *`far`* to *`near`* needs to be small. This enables the rendering engine to effectively determine which items should be drawn "on top of" other items, simulating closeness of items. The *near* argument must be greater than zero.

The "Hello World" example contains the line

```
scene << Perspective( 45, 3, 7 ); // define the camera
```

This defines a 45-degree viewing angle, with a near plane 3 units from the viewer and a far plane 7 units from the viewer.

The viewing angle functions in the same way as a wide angle or telephoto lens on a camera. Small viewing angles zoom into a drawing; wide angles zoom out. In other words, a small viewing angle maps the screen space onto a small portion of the scene, resulting in apparently larger scene elements. A large viewing angle maps the screen space onto a large portion of the scene, resulting in apparently small screen elements. The size of scene elements can therefore be manipulated using the *angle* argument of the `Perspective` function. The picture here shows the hello world script with perspective angles of 45 and 90 degrees.

Figure 13.6 Changing the Perspective

As an alternative to the `Perspective` command, you can define the actual viewing frustum with the `Frustum` command.

```
Frustum(left, right, bottom, top, near, far);
```

The frustum's viewing volume is defined by (*left, bottom, near*) and (*right, top, near*), which specify the (*x, y, z*) coordinates of the lower left and upper right corners of the near clipping plane; *near* and *far* give the distances from the viewpoint to the near and far clipping planes.

Setting up an Orthographic Scene

Orthographic scenes are specified in ways similar to perspective scenes. Issue the command

```
Ortho(left, right, bottom, top, near, far)
```

which specifies the four corners of the near plane, the distance to the near plane, and the distance to the far plane.

If you are dealing with a simple 2-D environment, you can set up a two-dimensional orthographic scene with the command

```
Ortho2D (left, right, bottom, top)
```

which specifies the corners of the two-dimensional view.

Changing the View

One of the advantages of creating a 3-D scene is the ease that they can be viewed from different angles and positions. The `Translate` and `Rotate` commands let you set the position from which you view the scene.

In addition, you can use the `ArcBall` command to enable the user to change the viewing angle interactively.

The Translate Command

You have actually seen the `Translate` command in earlier sample scripts. It sets the position from which the scene is viewed. The arguments give the amount to move from the current position in the *x*, *y*, and *z* direction.

```
Translate (x, y, z)
```

For example,

```
Translate( 0.0, 0.0, -2 );
```

moves the origin two units in the negative *z* direction.

Initially, the origin and camera were at the same place. Now, the camera can see the origin because the camera faces down the negative *z* axis.

The Rotate Command

The `Rotate` command is used to modify the viewing angle of a scene. It has the following format.

```
Rotate (degrees, xAxis, yAxis, zAxis)
```

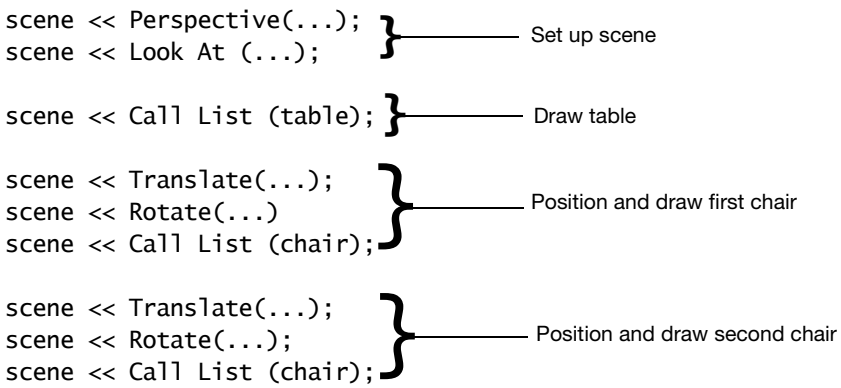
This rotates by degrees around the axis described by the vector (*xAxis*, *yAxis*, *zAxis*). For example, to rotate a model 90 degrees about the *x*-axis, use `Rotate( 90, 1, 0, 0 )`.

You can also specify the three axis values in a matrix. For example, `Rotate( 90, [1, 0, 0] )`.

Note: The `Rotate` command uses degrees, in contrast to JMP's trigonometric functions, which use radians.

`Translate` and `Rotate` are also used to position objects with respect to each other. The first `Translate` or `Rotate` can be thought of as positioning everything that follows with respect to the camera. Subsequent `Translate` and `Rotate` commands are used to position objects, such as spheres, cylinders, disks, and display lists in `Call List` and `ArcBall` commands. For example, suppose you have a display list named `table` and another named `chair`. Your scene might look like this:

Figure 13.7 Using Translate and Rotate

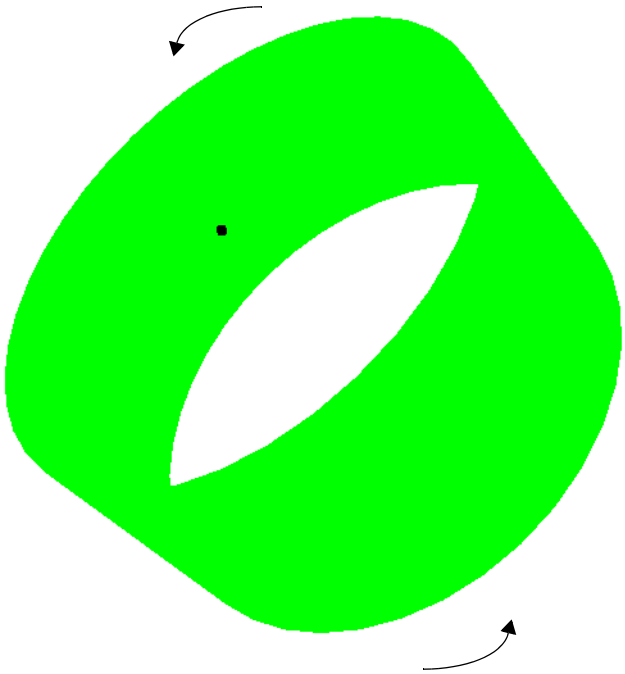


The following example uses the `Rotate` command inside a `For` loop to continuously change the viewing angle of a scene. It draws a cylinder that swings around a central point. This central point is shown by a small sphere.

```
scene = Scene Box( 600, 600 ); // make a scene box...holds an OpenGL scene.

New Window( "Example 1", scene ); // put the scene in a window

For( i = 1, i < 360, i++,
   scene << Clear;

   // the lens is 50 degrees, near is 1 units from the camera, far is 10.
   scene << Perspective( 50, 1, 10 );
   scene << Translate( 0.0, 0.0, -2 );

   scene << Rotate( i, 1, 0, 0 );
   scene << Rotate( i * 3, 0, 1, 0 );
   scene << Rotate( i * 3 / 2, 0, 0, 1 );
```

```
scene << Color( 0, 1, 0 ); // green for cylinder
scene << Cylinder( 0.5, 0.5, 0.5, 40, 10 );
scene << Color( 0, 0, 0 ); // black for sphere
scene << Sphere( 0.01, 10, 5 );
scene << Update;
Wait( 0.01 ); );
```

Figure 13.8 Rotating a Cylinder



Note the use of the `Update` command at the end of the scene messages. This command tells JMP to make the displayed screen agree with the current state of the display list. It is important to clear the list at the beginning (so that the list does not contain the old angles as well as the current) and update the scene after each change.

The Look At Command

The `Look At` command is an alternative way to set the camera view.

`Look At(` *eyeX*, *eyeY*, *eyeZ*, *centerX*, *centerY*, *centerZ*, *upX*, *upY*, *upZ* `)`

The Look At command puts the camera at the *eye* coordinates and points it toward the *center* coordinates. The *up* vector describes how the camera is rotated on its line of sight. The model is typically constructed at the origin. Therefore, a JMP scene should have either a `Look At` or a `Translate` command near its beginning to move the camera away from the origin.

First, clear the scene box of any commands from the previous frame.

```
scene << Clear;
```

Then use one of these projections:

```
scene << Perspective( 45, 2, 10 );
scene << Frustum( -.5, .5, -.5, .5, 1, 10 );
scene << Ortho( -2, 2, -2, 2, 1, 10 );
scene << Ortho2d( -2, 2, -2, 2 );
```

Note: If you use the `ortho2d` projection, you should not also set the camera position using either `Translate` or `Look At`.

Finally, use either `Translate` or `Look At` to set the camera position:

```
scene << Translate(0.0, 0.0, -4.5);
/* the camera faces down the negative Z axis.
   move it back so that 0,0,0 is in view. */
scene << Look At( /*eye*/ 3,3,3, /*center*/ 0,0,0, /*up*/ 1,0,0 );
/*this is much easier. */
```

Once the scene and camera position are set, add your model.

The ArcBall

Sometimes you want a scene to rotate based on the movements of the mouse. The Surface Plot platform in JMP is an example of a 3-D scene that rotates based on mouse movements.

An *ArcBall* creates a sphere around the 3-D scene and enables the user to click the sphere's surface and drag it around, thus causing the scene to rotate.

Use an `ArcBall` instead of a `Call List` command to place the scene in an ArcBall. Scenes that are attached to an ArcBall automatically respond to clicks and drags of the mouse. Custom programming is not needed. However, rotations made in the arcball are not saved. (Technically, the ArcBall is surrounded by an implicit `Push Matrix` and `Pop Matrix` block, so the movements are gone after it returns. See "Using the Matrix Stack" for more information about pushing and popping.)

For example, examine the script from "Primitives Example". Change the single line

```
scene << CallList(shape); // send the display list to the scene
```

so that it reads

```
scene << ArcBall(shape,2); // send the display list to an arcball
```

This displays the script with an associated arcball with diameter 2. When you run the script and the window appears, right-click and select **Show ArcBall > Always** from the menu that appears.

Note: ArcBall comes from an article by Shoemake (1994) found in *Graphics Gems IV*, published by Academic Press.

This sets the display so that the ArcBall is always showing. Click and drag on the ArcBall to rotate the scene. The right-click menu with Background Color, Use Hardware Acceleration, and Show ArcBall is always available, whether the scene is displayed through a platform, in a journal, or through JSL.

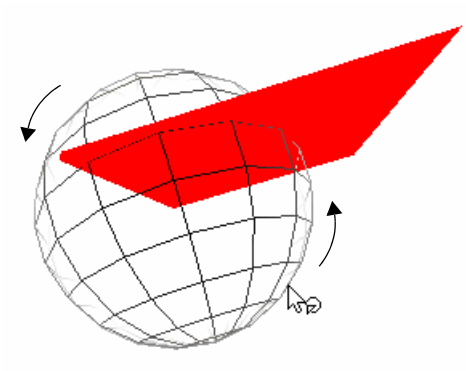
Note: The ArcBall does not have to be showing to react to mouse commands. It is shown here for display purposes only.

You can also set the display state of the ArcBall in JSL using the `Show ArcBall` command.

```
scene << Show Arcball(state)
```

where *state* is `During Drag`, `Always`, or `Never`.

Figure 13.9 Showing the Arc Ball



Graphics Primitives

All scenes in JSL are built with a small number of graphics primitives. These fundamental elements function as the building blocks for complicated scenes.

Every graphics primitive involves specifying vertices. In some cases, the vertices are simply drawn as points. In others, the vertices are connected to form polygons. To draw a primitive, you must specify the type of primitive and the coordinates and properties of the vertices involved. In JSL, this specification is accomplished through the `Begin` and `End` statements.

```
scene<<Begin(primitive type);
...(commands specifying vertices and their properties)...
scene<<End();
```

To specify the coordinates of the vertices, use the `vertex` command.

```
scene<<Begin(primitive type);
scene<<Vertex(x, y, z);
...
scene<<End();
```

The options for *`primitive type`* are the following. In these examples, assume that `v0`, `v1`, and so on, have been specified between a `Begin` and `End` pair, similar to the following.

```
scene<<Begin(primitive type);
scene<<Vertex(x0,y0,z0) // specify vertex v0
scene<<Vertex(x1,y1,z1) // specify vertex v1
...
scene<<Vertex(xn,yn,zn) // specify vertex vn
scene<<End();
```

Table 13.1 Primitive Types

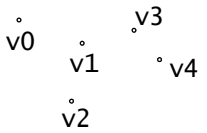
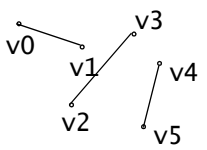
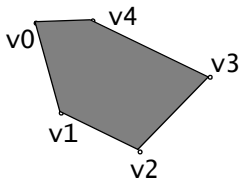
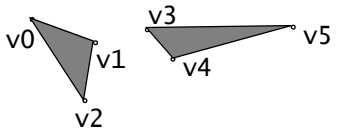
primitive type=POINTS	Draws a point at each of the vertices.	
primitive type=LINES	Draws a series of (unconnected) line segments. Segments are drawn between `v0` and `v1`, between `v2` and `v3`, and so on. If `n` is odd, the last vertex is ignored.	
primitive type=POLYGON	Draws a polygon using the points `v0`,..., `vn` as vertices. Three vertices must exist, or nothing is drawn. In addition, the polygon specified must not intersect itself and must be convex. If the vertices do not satisfy these conditions, the results are unpredictable.	
primitive type=TRIANGLES	Draws a series of (disconnected) triangles using vertices `v0`, `v1`, `v2`, then `v3`, `v4`, `v5`, and so on. If the number of vertices is not an exact multiple of 3, the final one or two vertices are ignored.	

Table 13.1 Primitive Types *(Continued)*

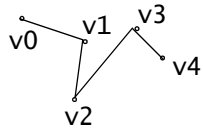
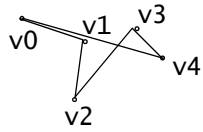
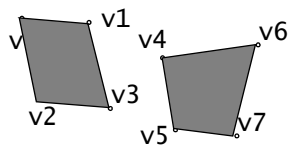
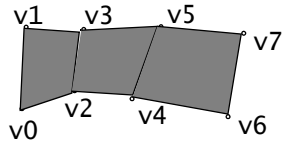
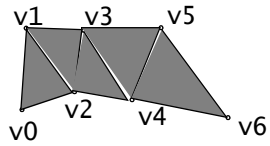
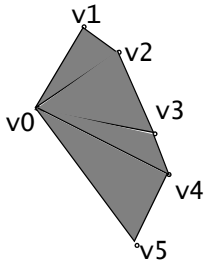
primitive type=LINE_STRIP	Draws a line segment from v0 to v1, then from v1 to v2, and so on. Therefore, n vertices specify n–1 line segments. Nothing is drawn unless there is more than one vertex. There are no restrictions on the vertices describing a line strip; the lines can intersect arbitrarily.	
primitive type=LINE_LOOP	Same as LINE_STRIP, except that a final line segment is drawn from the last vertex to the first, completing a loop.	
primitive type=QUADS	Draws a series of quadrilaterals (four-sided polygons) using vertices v0, v1, v2, v3, then v4, v5, v6, v7, and so on. If the number of vertices is not a multiple of 4, the final one, two, or three vertices are ignored.	
primitive type=QUAD_STRIP	Draws a series of quadrilaterals (four-sided polygons) beginning with v0, v1, v3, v2, then v2, v3, v5, v4, then v4, v5, v7, v6, and so on. The number of vertices must be at least 4 before anything is drawn, and if odd, the final vertex is ignored.	
primitive type=TRIANGLE_STRIP	Draws a series of triangles (three-sided polygons) using vertices v0, v1, v2, then v2, v1, v3 (note the order), then v2, v3, v4, and so on. The ordering is to ensure that the triangles are all drawn with the same orientation so that the strip can correctly form part of a surface. There must be at least three vertices for anything to be drawn.	

Table 13.1 Primitive Types *(Continued)*

primitive type=TRIANGLE_FAN	Same as TRIANGLE_STRIP, except that the vertices are v0, v1, v2, then v0, v2, v3, then v0, v3, v4, and so on.	

Primitives Example

The following short example illustrates the use of a graphics primitive.

```
shape = Scene Display List(); // create a display list and send it commands
shape << Color( 1, 0, 0 ); // set the RGB color of the text
shape << Begin( POLYGON );
shape << Vertex( 0, 0, 0 );
shape << Vertex( 0, 3, 0 );
shape << Vertex( 3, 3, 0 );
shape << Vertex( 5, 2, 0 );
shape << Vertex( 4, 0, 0 );
shape << Vertex( 2, -1, 0 );
shape << End();

// draw the window and send it the stored display list
scene = Scene Box( 400, 400 ); // make a scene box
New Window( "Primitive", scene ); // put the scene in a window
scene << Perspective( 90, 3, 7 ); // define the camera
scene << Translate( 0.0, 0.0, -5 ); // move to (0,0,-5) to draw
scene << Call List( shape ); // send the display list to the scene
scene << Update; // update the scene
```

The first section of the script creates a display list named `shape`. Inside this display list, a polygon is defined using six vertices.

The second section of the script creates a scene box and a new window. It then uses the `Call List` message to put the list in the display.

Note that all the *z*-coordinates are zero, which makes sure the polygon lies in a plane. Polygons that do not lie in a plane can cause unpredictable results.

Experiment with the line

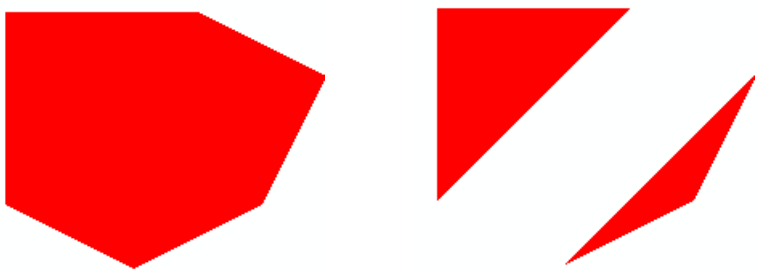
```
shape <<Begin(POLYGON);
```

by changing it to some of the other primitive types. For example, changing it to

```
shape <<Begin(TRIANGLES);
```

results in a different picture.

Figure 13.10 Polygon (left) and Triangles (right)



Controlling the Appearance of Primitives

JSL has several commands that let you tailor-make the appearance of primitive drawing objects. You can also specify the widths of lines and their stippling pattern (that is, whether they are dashed, dotted, and so on.)

Size and Width

To set the point size of rendered objects, use the `Point Size` command.

```
Point Size( n )
```

where *n* is the number of pixels. Note that this might not be the actual number of pixels rendered, depending on other settings such as anti-aliasing and your hardware configuration.

Set the line width using the `Line Width` command

```
Line Width( n)
```

where *n* is the number of pixels. The argument *n* must be larger than zero and is, by default, one.

Stippling Pattern

To make stippled lines, use the `Line Stipple` command.

```
Line Stipple( factor, pattern )
```

Factor is a stretching factor. *Pattern* is a 16-bit integer that turns pixels on or off. Use `Enable(LINE_STIPPLE)` to turn the effect on.

To construct a line stippling pattern, write a 16-bit binary number that represents the stippling pattern that you desire. Note that the pattern should read from right to left, so your representation might seem backward to the way it is rendered. Convert the binary number to an integer and use this as the *`pattern`* argument.

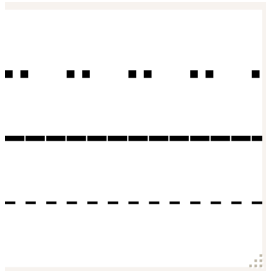
For example, imagine you want the dotted line pattern 0000000011111111. This is equal to 255 in decimal notation, so use the command `Line Stipple(1, 255)`.

The *`factor`* argument expands each binary digit to two digits. In the example above, `Line Stipple(2, 255)` would result in 0000000000000000111111111111111.

For example, the following script draws three lines, each of different widths (the `Line Width` commands) and stippling patterns.

```
scene = Scene Box( 200, 200 ); // make a scene box...holds an OpenGL scene

New Window( "Stipples", scene ); // put the scene in a window
scene << Ortho( -2, 2, -2, 2, -1, 1 );
scene << Color( 0, 0, 0 ); // set the RGB color of the text

scene << Enable( LINE_STIPPLE );
scene << Line Width( 2 );
scene << Line Stipple( 1, 255 );

scene << Begin( LINES );
scene << Vertex( -2, -1, 0 );
scene << Vertex( 2, -1, 0 );
scene << End();

scene << Line Width( 4 );
scene << Line Stipple( 1, 32767 );

scene << Begin( LINES );
scene << Vertex( -2, 0, 0 );
scene << Vertex( 2, 0, 0 );
scene << End();

scene << Line Width( 6 );
scene << Line Stipple( 3, 51 );

scene << Begin( LINES );
scene << Vertex( -2, 1, 0 );
scene << Vertex( 2, 1, 0 );
scene << End();
scene << Update;
```

Figure 13.11 Stipples



Note: Stipple patterns "crawl" on rotating models because they are in screen pixels, not model units. Lines in the model change length on the screen even though nothing changes in model units.

Fill Pattern

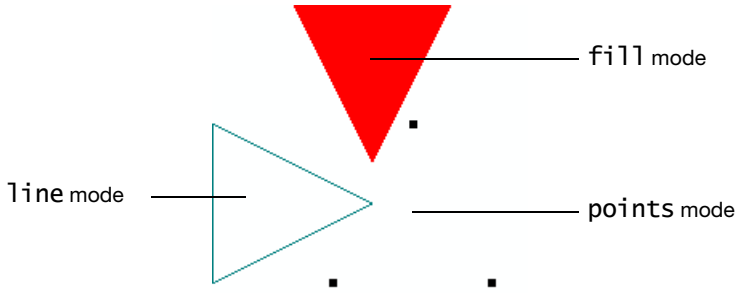
Polygons are rendered with both a front and a back, and the drawing mode of each side is customizable. This enables the user to see the difference between the back and front of the polygon.

To set the drawing mode of a polygon, use the `Polygon Mode` command.

```
Polygon Mode( face, mode )
```

where *`face`* can be `FRONT`, `BACK`, or `FRONT_AND_BACK`, and *`mode`* can be `POINT`, `LINE`, or `FILL`.

Figure 13.12 Points, Line, and Fill Modes



For example, the following script creates a display list that defines a triangle. This display list is used three times in conjunction with `Translate`, `Rotate`, and `Color` commands to draw triangles in three positions. In addition, the `Polygon Mode` command changes the drawing mode of each triangle. Note there is no explicit call to the `FILL` mode, since it is the default.

The following table dissects the script, showing how the `Translate` and `Rotate` commands accumulate to manipulate a single display list.

Table 13.2 Translate and Rotate Commands

Code from above script	comments
`shape = Scene Display List();`	Creates a display list

`shape << Begin( TRIANGLES );`
`shape << Vertex( 0, 0, 0 );`
`shape << Vertex( -1, 2, 0 );`
`shape << Vertex( 1, 2, 0 );`
`shape << End();`
	Creates a display list named shape that holds vertices for the triangles. All the z vertices are zero since this is a two dimensional scene

`scene = Scene Box( 200, 200 );`
`New Window( "Fill Modes", scene );`
`scene << Ortho2d( -2, 2, -2, 2 );`
	Put the scene in a display box, and create a new window.

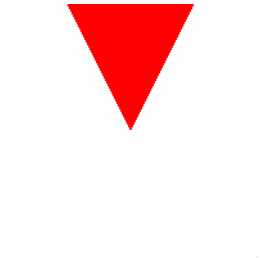
`scene << Color(1,0,0);`
`scene << Call List( shape );`
 	
`// update the scene to see the triangle`
`scene << Update;`
	Draw the first triangle in red.
	 

Table 13.2 Translate and Rotate Commands *(Continued)*

Code from above script	comments

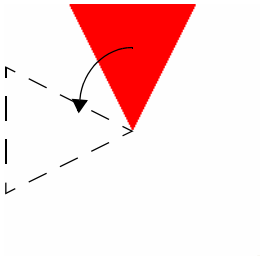
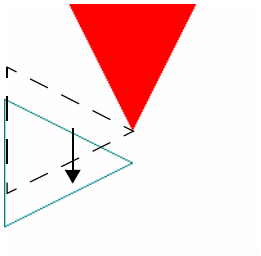
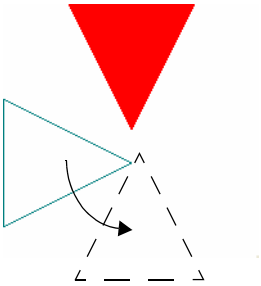
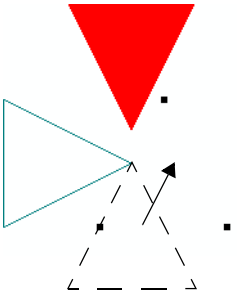
`scene << Rotate ( 90, 0, 0, 1 );`
`scene << Translate ( -0.5, 0, 0 );`
`scene << Color( 0, 0.5, 0.5 );`
`scene << Polygon Mode( FRONT_AND_BACK, LINE );`
`scene << Call List( shape );`
 	
`// update the scene to see the triangle`
`scene << Update;`
	Draw the second triangle in teal. Note that we first rotate the triangle.
	 
	And then translate it.
	 

Table 13.2 Translate and Rotate Commands *(Continued)*

Code from above script	comments

`scene << Rotate( 90, 0, 0, 1 );`
`scene << Translate( -0.5, -1, 0 );`
`scene << Color( 0, 0, 0 );`
 	
`// large points so that they are visible`
`scene << Point Size( 5 );`
`scene << Polygon Mode( FRONT_AND_BACK, POINT );`
`scene << Call List( shape );`
 	
`// update the scene to see the triangle`
`scene << Update;`
	Draw the third triangle as black points. First rotate.
	 
	And then translate to get the final picture.
	 

Some developers use the fill mode in concert with the line mode to draw a filled polygon with a differently colored border. However, due to the way the figures are rendered, they sometimes do not line up correctly. The `Polygon Offset` command is used to correct for this so-called "stitching" problem.

`Polygon Offset (`*`factor`*`, `*`units`*`)`

To enable offsetting, use `Enable(POLYGON_OFFSET_FILL)`, `Enable(POLYGON_OFFSET_LINE)`, or `Enable(POLYGON_OFFSET_POINT)`, depending on the desired mode. The actual offset values are calculated as m*(*`factor`*)+r*(*`units`*). m is the maximum depth slope of the polygon and r is the smallest value guaranteed to produce a resolvable difference in window coordinate depth values. Start with `Polygon Offset(1,1)` if you need this.

An example of `Polygon Offset` is in the Surface Plot platform, when a surface and a mesh are displayed on top of each other, or a surface and contours displayed on top of each other. In either case, the surface would interfere with the lines if the lines were not moved closer or the surface moved farther from the viewer.

Other Uses of Begin and End

Although vertices are typically specified between begin and end statements, there are other commands that are valid. These commands are discussed in other sections of this chapter.

- `Vertex` adds a vertex to the list
- `Color` changes the current color
- `Normal` sets the normal vector coordinates
- `Edge Flag` controls drawing of edges
- `Material` sets material properties
- `Eval Coord` and `Eval Point` generate coordinates
- `Call List` executes a display list.

Drawing Spheres, Cylinders, and Disks

There are several pre-defined commands that allow for quick rendering of spheres, cylinders, and disks. The advantage of these commands is not only their ease-of-use, but that they have special lighting properties (their "normals") built in.

Construction

The following commands are used to construct cylinders, disks, partial disks, and spheres.

Cylinders

```
Cylinder( baseRadius, topRadius, height, slices, stacks )
```

`baseRadius` is the radius of the cylinder's base. Similarly, *`topRadius`* is the radius of the top. *`height`* is the height of the cylinder.

`Slices` can be 10 for a reasonably accurate cylindrical shape. Using `QuadricNormals(Smooth)` helps the appearance.

`Stacks` sets the number of vertices available for lighting reflections. Use a larger value for *`Stacks`* for accurate "hot-spots".

Disks

The following command draws a paper-thin disk with an *`innerRadius`* hole in the middle.

```
Disk( innerRadius, outerRadius, slices, loops )
```

Like `Cylinder`, *`slices`* controls the accuracy of the curve and *`loops`* makes more vertices (for lighting accuracy).

```
Partial Disk( innerRadius, outerRadius, slices, loops, startAngle,
  sweepAngle )
```

The `Partial Disk` command works like `Disk`, but with a slice of the disk removed. Specify the part of the disk that is showing using *`startAngle`* and *`sweepAngle`*.

Spheres

The following command draws a sphere with the specified *`radius`*.

```
Sphere( radius, slices, stacks )
```

The *`slices`* can be thought of as longitudes and *`stacks`* as latitudes. About 10 of each make a nicely drawn sphere.

Lighting

It is not necessary to make specific calculations of normal vectors (as is the case for customized surfaces) for spheres, disks, and cylinders. However, you can use the following commands to customize the automatic lighting.

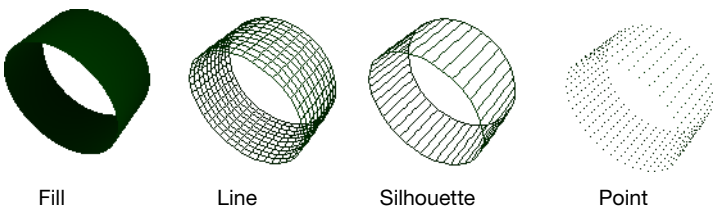
`Quadric Normals(`*`mode`*`)` tells what type of normal should be automatically generated. The argument *`mode`* can be `None`, `Flat`, or `Smooth`. `Flat` makes faceted surfaces. `Smooth` makes the normals at each vertex be the average of the adjacent polygons.

`Quadric Orientation(`*`mode`*`)` determines which way the normals point. The argument *`mode`* can be `Inside` or `Outside`.

`Quadric Draw Style(`*`mode`*`)` specifies the drawing mode. The argument *`mode`* can be `Fill`, `Line`, `Silhouette`, or `Point`.

JMP uses the values that you set for `Quadric Normals`, `Quadric Orientation`, and `Quadric Draw Style` for subsequently generated cylinders, disks, and spheres.

Figure 13.13 Draw Styles



Note: Other OpenGL documentation refers to quadric objects. JMP has only one and always uses it.

Drawing Text

As shown in the "Hello World" example above, text is added to a scene using the `Text` command.

`Text( horz, vert, size, string, <billboard>)`

- *horz* can be `Left`, `Center`, or `Right` justification.
- *vert* can be `Top`, `Middle`, `Baseline`, or `Bottom` justification.
- *size* represents the height of a capital letter M in model coordinates.
- *string* is the text to draw.
- *billboard* is an optional argument that causes the text to rotate with the model. Text with this option always faces the viewer.

The font is always the JMP Text font. You can change the text font from the preferences menu, but because of the way JMP caches fonts for scenes, changes might not take effect until JMP is restarted.

Note: Text is not part of the standard OpenGL definition.

Using Text with Rotate and Translate

The following example uses the text command in conjunction with the `Translate` and `Rotate` commands.

```
scene = Scene Box( 600, 600 ); // make a scene box...holds an OpenGL scene
New Window( "Example 2", scene ); // put the scene in a window

// the "lens" is 45 degrees, near is 3 units from the camera, far is 7
scene << Perspective( 45, 3, 7 );

// move the world so that 0,0,0 is visible in the camera
scene << Translate( 0.0, 0.0, -4.5 );

// rotate the first text about the Y (vertical on screen) axis
scene << Rotate( 30, 0, 1, 0 );
scene << Color( 1, 0, 2 ); // magenta
scene << Text( "center", "baseline", .2, "Top magenta string" );
```

```
// the next string is even farther away from the camera
scene << Translate( 0.0, 0.0, -2.0 );

// rotate the second text about the Y (vertical on screen) axis
scene << Rotate( 30, 0, 1, 0 );
scene << Color( 0, 1, 1); // aqua blue
scene << Text( "center", "baseline", .2, "Back aqua blue very long string" );

// update the display box in the window using the current display list
scene << Update;
```

Figure 13.14 Rotating and Translating Text Strings

Note that the aqua blue string extends backward beyond the far clipping plane. Change the 7 to 10 in the `Perspective` command to see the complete string.

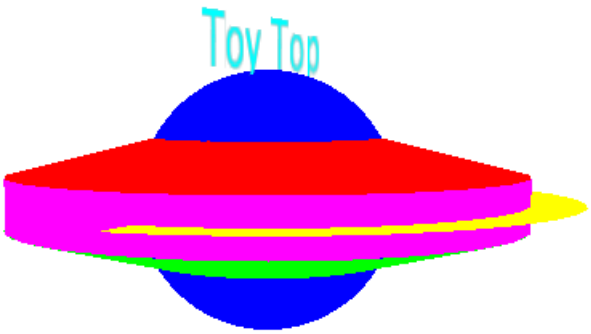
Using the Matrix Stack

JMP 3-D scenes use a matrix stack to keep track of the current transform. The stack is initialized to the identity matrix, and each time a translate, rotate, or scale command is given, the top matrix on the stack is changed.

Note: Unlike many OpenGL implementations, JMP does not use a transposed matrix.

The JSL example below uses `Push Matrix` and `Pop Matrix` to position pieces of the toy top and then return to the origin. This is faster than using the `Translate` command a second time in reverse.

Figure 13.15 Drawing With a Matrix Stack




```
toyTop = Scene Display List();
toyTop << Push Matrix;
toyTop << Translate( 0, 0, .1 );
toyTop << Color( 1, 0, 0 ); // red

// baseRadius, topRadius, height, slices, stacks
toyTop << Cylinder( 1, .2, .2, 25, 5 );
toyTop << Pop Matrix;
toyTop << Push Matrix;
toyTop << Translate( 0, 0, -.1 );
toyTop << Rotate( 180, 1, 0, 0 );
toyTop << Color( 0, 1, 0 ); // green
toyTop << Cylinder( 1, .2, .2, 25, 5 );
toyTop << Pop Matrix;

toyTop << Color( 0, 0, 1 ); // blue
toyTop << Sphere( .5, 30, 30 ); // radius, slices, stacks

toyTop << Color( 1, 1, 0 ); // yellow

// innerRadius, outerRadius, slices, rings, startAngle, sweepAngle
toyTop << Partial Disk( 1, 1.2, 25, 2, 0, 270 );

toyTop << Push Matrix;
toyTop << Translate( 0, 0, -.1 );
toyTop << Color( 1, 0, 1 ); // magenta
toyTop << Cylinder( 1, 1, .2, 25, 3 );

// baseRadius, topRadius, height, slices, stacks
toyTop << Pop Matrix;
```

```
toyTop << Push Matrix;
toyTop << Rotate( 90, 1, 0, 0 );
toyTop << Translate( 0, .5, 0 );
toyTop << Color( 0, 1, 1 ); // cyan
toyTop << Text( "center", "baseline", .2, "Toy Top" );
toyTop << Pop Matrix;

scene = Scene Box( 600, 600 ); // make a scene box...holds an OpenGL scene

New Window( "Example 3", scene ); // put the scene in a window
scene << Perspective( 45, 3, 7 );
scene << Translate( 0.0, 0.0, -4.5 );
scene << Rotate( -85, 1, 0, 0 );
scene << Rotate( 65, 0, 0, 1 );
scene << Call List( toyTop );

scene << Update; // update the display box
```

There are some cases where you want to replace the current matrix on the stack. For these cases, use the `Load Matrix` command.

```
Load Matrix(m)
```

where *m* is a 4x4 JMP matrix that is loaded onto the current matrix stack.

Similar is the `Mult Matrix` command

```
Mult Matrix(m)
```

When the `Mult Matrix` command is issued, the matrix on the top of the current matrix stack is multiplied by *m*.

The following matrices perform some simple commands.

Translation:

```
[1  0  0  x
0  1  0  y
0  0  1  z
0  0  0  1]
```

In the following rotation matrices, `c` = cos(angle) and `s`=sin(angle).

Rotation about X axis:

```
[1  0  0  0
0  c -s  0
0 -s  c  0
0  0  0  1]
```

Rotation about Y axis:

```
[c  0  s  0
 0  1  0  0
-s  0  c  0
 0  0  0  1]
```

Rotation about Z axis:

```
[c -s  0  0
 s  c  0  0
 0  0  1  0
 0  0  0  1]
```

For example, here are two equivalent (except for the translation being opposite) ways to translate and rotate a display list.

```
// first way uses matrix
gl << Push Matrix;
xt = Identity( 4 ); // translate this one left by .75
xt[1, 4] = -.75;
xr = Identity( 4 ); // rotate this one, cos needs radians, not degrees
xr[2, 2] = Cos( 3.14159 * a / 180 );
xr[2, 3] = -Sin( 3.14159 * a / 180 );
xr[3, 2] = Sin( 3.14159 * a / 180 );
xr[3, 3] = Cos( 3.14159 * a / 180 );
yr = Identity( 4 );
yr[1, 1] = Cos( 3.14159 * a / 180 );
yr[1, 3] = Sin( 3.14159 * a / 180 );
yr[3, 1] = -Sin( 3.14159 * a / 180 );
yr[3, 3] = Cos( 3.14159 * a / 180 );
zr = Identity( 4 );
zr[1, 1] = Cos( 3.14159 * a / 180 );
zr[1, 2] = -Sin( 3.14159 * a / 180 );
zr[2, 1] = Sin( 3.14159 * a / 180 );
zr[2, 2] = Cos( 3.14159 * a / 180 );

// order of multiplication matters with matrices
gl << Mult Matrix( xt * xr * yr * zr );
gl << Arcball( dl, 1 );
gl << Pop Matrix;

// second way uses functions
gl << Push Matrix;
gl << Translate( .75, 0, 0 ); // translate this one right by .75
gl << Rotate( a, 1, 0, 0 ); // rotate this one in degrees
gl << Rotate( a, 0, 1, 0 ); // order of operations also matters here
gl << Rotate( a, 0, 0, 1 );
```

```
gl << Arcball( dl, 1 );
gl << Pop Matrix;
```

It is not possible to read back the current transform matrix, because the matrix exists only while the display list is drawing, not while your JSL script is creating it. If you must know its content, create it in JSL and use `Load Matrix` to put it on the stack.

Lighting and Normals

The following methods enable you to add lighting, materials, and normal vectors to your shapes. Using these methods, models can appear shiny or light-absorbing.

- "Creating Light Sources"
- "Lighting Models"
- "Normal Vectors"
- "Shading Model"
- "Material Properties"
- "Alpha Blending"
- "Fog"
- "Example"

Creating Light Sources

Light sources are specifications of a color, position, and direction. JSL allows for up to eight lights (numbered 0 to 7) defined by the `Light` command, where *n* is the number of the light.

```
Light( n, argument, value, ... value )
```

Note: To turn each light on, issue an `Enable(`*`Lighting`*`)` and an `Enable(light`*`n`*`)` command, where *n* is the light number. Then, move the light to a position in the scene with a `Light(`*`n`*`, `*`POSITION`*`, `*`x`*`, `*`y`*`, `*`z`*`)` command.

The value of *argument* can be any one of those shown in Table 13.3. The table shows default values for each argument.

Table 13.3 Light Arguments and Default Values

Argument	Default Value	Meaning
AMBIENT	(0, 0, 0, 1)	Ambient RGBA intensity
DIFFUSE	(1, 1, 1, 1)	diffuse RGBA intensity
SPECULAR	(1, 1, 1, 1)	specular RGBA intensity
POSITION	(0, 0, 1, 0)	(x, y, z, w) position
SPOT_DIRECTION	(0, 0, -1)	(x, y, z) direction of spotlight
SPOT_EXPONENT	0	spotlight exponent
SPOT_CUTOFF	180	spotlight cutoff angle
CONSTANT_ATTENUATION	1	constant attenuation factor
LINEAR_ATTENUATION	0	linear attenuation factor
QUADRATIC_ATTENUATION	0	quadratic attenuation factor

Note: The default values for DIFFUSE and SPECULAR in this table apply only to Light 0. For other lights, the default value is (0, 0, 0, 1) for both arguments.

The first three arguments (AMBIENT, DIFFUSE, and SPECULAR) are used to color the light. DIFFUSE is the argument that is most closely associated with the physical color of the light. AMBIENT refers to the property of the light when it functions as a background light. SPECULAR alters the way a light is reflected off a surface.

Specify the position of the light using the POSITION argument. Nonzero values of the fourth (w) coordinate position the light in homogenous object coordinates.

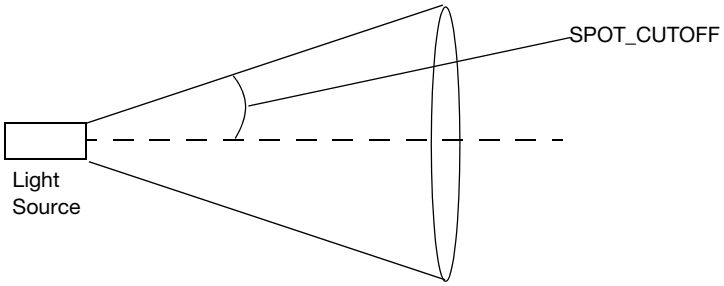
Light in the real-world decreases in intensity as distance from the light increases. Since a directional light is infinitely far away, it does not make sense to attenuate its intensity as a function of distance. However, JSL attenuates a light source by multiplying the contribution of the source by an attenuation factor

$$\text{attenuation factor} = \frac{1}{c + ld + qd^2}$$

where c = CONSTANT_ATTENUATION, l = LINEAR_ATTENUATION, and q = QUADRATIC_ATTENUATION.

To create a spotlight, limit the shape of the light to a cone. Use the `SPOT_CUTOFF` argument to define the side of the cone, as shown in the following illustration.

Figure 13.16 Spotlight



In addition to the cutoff angle, you can control the intensity and direction of the light distribution in the cone. `SPOT_DIRECTION` specifies the direction for the spotlight to point; `SPOT_EXPONENT` influences how concentrated the light is.

Lighting Models

Lighting models are specified with the `Light Model` command.

```
Light Model( argument, value,...,value )
```

Light models specify three attributes of lights.

- The global ambient light intensity
- Whether the viewpoint is local or is an infinite distance away
- Whether lighting calculations should be performed differently for the front and back faces of objects.

Table 13.3 shows the three valid arguments for the `Light Model` command.

Table 13.4 Light Model Arguments and Default Values

Argument	Default Value	Meaning
`LIGHT_MODEL_AMBIENT`	(0.2, 0.2, 0.2, 1)	Ambient RGBA intensity of the entire scene
`LIGHT_MODEL_LOCAL_VIEWER`	0 (false)	how specular reflection angles are computed
`LIGHT_MODEL_TWO_SIDE`	0 (false)	nonzero values imply two-sided lighting

Normal Vectors

Normal vectors point in a direction perpendicular to a surface. For a plane, all normals are the same. For a more complicated surface, normals are more complicated. JSL enables you to specify the normal vector for each vertex. These normals specify the orientation of the surface in space, necessary for lighting calculations. Accurate normals assure accurate lighting.

The normal vector is of length 1 and is perpendicular to the vertex. Typically, a vertex is shared between several polygons. A smooth shaded effect is desired, so the perpendicular at the vertex is calculated as a (possibly weighted) average of the polygon's normals. It is important to calculate the "outward" normal for polygons unless two-sided shading is enabled because only the outer face of the polygon is illuminated. With a scaled polygon, the normal's length is not 1 after scaling, and the lighting is wrong.

Normal vectors are set at the same time the surface is constructed, and are specified with the `Normal` command. Use the `Enable(NORMALIZE)` command to have the normals re-normalized to 1 each time the scene is drawn.

Shading Model

The shading model of a polygon is set using the `Shade Model` command.

```
Shade Model (mode)
```

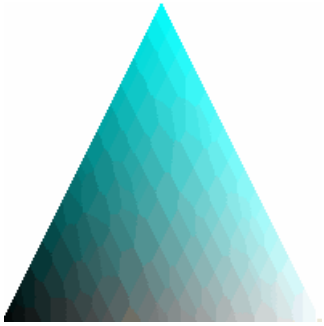
where *mode* can be `SMOOTH` (the default) or `FLAT`. `SMOOTH` shading interpolates the colors of the primitive from one vertex to the next. `FLAT` mode duplicates the color of one vertex across the entire primitive.

The following script changes the color at each of a triangle's vertices. The `FILL` shade model interpolates the color of the interior automatically.

```
scene = Scene Box( 200, 200 ); // make a scene box...holds an OpenGL scene.

New Window( "Shade Model", scene ); // put the scene in a window
scene << Clear;
scene << Ortho2D( -1, 1, -1, 1 );

scene << Shade Model( SMOOTH );
scene << Polygon Mode( FRONT_AND_BACK, FILL );
scene << Begin( TRIANGLES );
scene << Color( 0, 0, 0 ); // black
scene << Vertex( -1, -1, 0 );
scene << Color( 0, 1, 1 ); // cyan
scene << Vertex( 0, 1, 0 );
scene << Color( 1, 1, 1 ); // white
scene << Vertex( 1, -1, 0 );
scene << End();
```

```
scene << Update;
```

Figure 13.17 Shading



Material Properties

To set the material properties of a surface, use the `Material` command.

```
Material( face, argument, value,...value )
```

face can be `Front`, `Back`, or `Front_and_back`. (Note that the material properties can be set separately for the front and back faces of a polygon.)

Table 13.5 shows the arguments and default values for `Material` arguments.

Table 13.5 Material Arguments and Default Values

Argument	Default Value	Meaning
`AMBIENT`	(0.2, 0.2, 0.2, 1.0)	Ambient color of material
`DIFFUSE`	(0.8, 0.8, 0.8, 1.0)	Diffuse color of material
`AMBIENT_AND_DIFFUSE`		Both `AMBIENT` and `DIFFUSE`
`SPECULAR`	(0.0, 0.0, 0.0, 1.0)	Specular color of material
`SHININESS`	0	Specular exponent that can range from 0 to 128.
`EMISSION`	(0, 0, 0, 1)	Emissive color of material

Alpha Blending

The BlendFunc command allows for alpha blending. To use it, send a BlendFunc message to a scene, for example:

```
scene << BlendFunc(SRC_ALPHA, ONE_MINUS_SRC_ALPHA)
```

SRC_ALPHA and ONE_MINUS_SRC_ALPHA are OpenGL constants that tell BlendFunc to use alpha to blend against the existing display buffer. Disabling *z*-buffer testing or rendering primitives from back to front might be needed for some applications. By default, the *z*-buffer tests prevent anything from drawing behind a transparent polygon after it is drawn.

More information about all the constants available to BlendFunc (many of which are not useful to the JSL programmer) are available in the OpenGL documentation at opengl.org.

Fog

Fog enables figures to fade into the distance, making for more realistic models. All types of geometric figures can be fogged. To turn fog on, enable the FOG argument.

```
Enable (FOG)
```

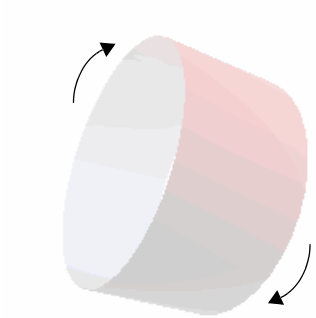
Example

The following example uses several of the concepts presented in this section, including lighting, fog, and normalization. It draws a spinning cylinder that is affected by two lights.

```
scene = Scene Box( 300, 300 ); // make a scene box
New Window( "Cylinder", scene ); // put the scene in a window

For( i = 1, i < 360, i++,
  scene << Clear;

  // the lens is 45 degrees, near is 3 units from the camera, far is 7.
  scene << Perspective( 50, 1, 10 );

  // move the world so that 0,0,0 is visible in the camera
  scene << Translate( 0.0, 0.0, -2 );
  scene << Enable( Lighting );
  scene << Enable( Light0 );
  scene << Enable( Light1 );
  scene << Light( Light0, POSITION, 1, 1, 1, 1 ); // near viewer
  scene << Light( Light0, DIFFUSE, 1, 0, 0, 1 ); // red light
  scene << Light( Light1, POSITION, -1, -1, -1, 1 ); // behind object
  scene << Light( Light1, DIFFUSE, .5, .5, 1, 1 ); // blue-gray light
  scene << Enable( Fog );
```

```
    scene << Enable( NORMALIZE );

    scene << Rotate( i, 1, 0, 0 );
    scene << Rotate( i * 3, 0, 1, 0 );
    scene << Rotate( i * 3 / 2, 0, 0, 1 );
    scene << Cylinder( 0.5, 0.5, 0.5, 40, 10 );
    scene << Update;
    Wait( 0.01 );
);
```

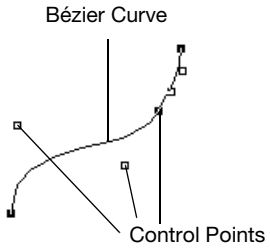
Figure 13.18 Fog



Bézier Curves

A complete discussion of Bézier curves is beyond the scope of the *Scripting Guide*. JSL has several commands for defining and drawing curves and their associated meshes.

Figure 13.19 Bézier Curve



One-Dimensional Evaluators

To define a one-dimensional map, use the `Map1` command.

```
Map1( target, u1, u2, stride, order, matrix )
```

The *`target`* argument defines what the control points represent. Values of the *`target`* argument are shown in Table 13.6. Note that you must use the `Enable` command to enable the argument.

Table 13.6 Map1 *`Target`* Arguments and Default Values

`target` Argument	Meaning
`MAP1_VERTEX_3`	(x, y, z) vertex coordinates
`MAP1_VERTEX_4`	(x, y, z, w) vertex coordinates
`MAP1_INDEX`	color index
`MAP1_COLOR_4`	R, G, B, A
`MAP1_NORMAL`	normal coordinates
`MAP1_TEXTURE_COORD_1`	s texture coordinates
`MAP1_TEXTURE_COORD_2`	s, t texture coordinates
`MAP1_TEXTURE_COORD_3`	s, t, r texture coordinates
`MAP1_TEXTURE_COORD_4`	s, t, r, q texture coordinates

The second two arguments (*`u1`* and *`u2`*) define the range for the map. The *`stride`* value is the number of values in each block of storage. In other words, the value is the offset between the beginning of one control point and the beginning of the next control point. The *`order`* should equal the degree of the curve plus one. The *`matrix`* holds the control points.

For example, `Map1( MAP1_VERTEX_3, 0, 1, 3, 4, <4x3 matrix> )` is typical for setting the two end points and two control points to define a Bézier line.

You use the `MapGrid1` and `EvalMesh1` commands to define and apply an evenly spaced mesh.

```
MapGrid1( un, u1, u2 )
```

sets up the mesh with *`un`* divisions spanning the range *`u1`* to *`u2`*. Code is simplified by using the range 0 to 1.

```
EvalMesh1( mode, i1, i2 )
```

actually generates the mesh from *i1* to *i2*. The *mode* can be either POINT or LINE. The EvalMesh1 command makes its own Begin and End clause.

The following example script demonstrates a one-dimensional outlier. A random set of control points draws a smooth curve. Only the first and last points are on the curve. Using NPOINTS=4 results in a cubic Bézier spline.

```
boxwide = 500;
boxhigh = 400;

gridsize = 100; // bigger for finer divisions

/* We suggest you use only values between 2 and 8 (inclusively). Numbers
   beyond these might be interpreted differently, depending on implementation.
   This value is the degree+1 of the fitted curve */
NPOINTS = 4;

point = J( NPOINTS, 3, 0 );

// create an array of x,y,z triples
For( x = 1, x <= NPOINTS, x++,
   point[x, 1] = (x - 1) / (NPOINTS - 1) - .5;
      // x from -.5 to +.5
   point[x, 2] = Random Uniform() - .5; // y is random in same range

   // z is always zero, which causes the curve to stay in a plane
   point[x, 3] = 0;
);

spline = Scene Box( boxwide, boxhigh );

// data from -.5 to .5 in x and y; this is a little larger
spline << Ortho( -.6, .6, -.6, .6, -2, 2 );

spline << Enable( MAP1_VERTEX_3 );
spline << MapGrid1( gridsize, 0, 1 );
spline << Color( .2, .2, 1 ); // blue curve

spline << Map1( MAP1_VERTEX_3, 0, 1, 3, NPOINTS, point );
spline << Line Width( 2 ); // not-so-skinny curve
spline << EvalMesh1( LINE, 0, gridsize ); // also try LINE, POINT

spline << Color( .2, 1, .2 );
spline << Point Size( 4 ); // big fat green points

// show the points and label them
For( i = 1, i <= NPOINTS, i++,
```

```
    spline << Begin( "POINTS" );
    spline << Vertex( point[i, 1], point[i, 2], point[i, 3] );
    spline << End;
    spline << Push Matrix;
    spline << Translate( point[i, 1], point[i, 2], point[i, 3] );
    spline << Text( center, bottom, .05, Char( i ) );
    spline << Pop Matrix;
);

New Window( "Spline", spline );
```

https://www.tinaja.com/glib/bezconn.pdf offers an explanation of connecting cubic segments so that both the slope and the rate of change match at the connection point. This example does not illustrate doing so; there is only one segment here.

Two-Dimensional Evaluators

Two-dimensional evaluators follow their one dimensional counterparts, and are used in a similar way.

```
Map2(target, u1, u2, ustride, uorder, v1, v2, vstride, vorder, matrix)
Eval Coord2(u, v)
```

Values for the *target* argument are the same as those shown in Table 13.6 with *Map1* replaced with *Map2* appropriately. The *u1, u2, v1,* and *v2* values specify the range of the two-dimensional mesh.

For example, `Map2(MAP2_VERTEX_3, 0, 1, 3, 4, 0, 1, 12, 4, <16x3 matrix>)` is typical for setting the 16 points that define a Bézier surface.

Use the `MapGrid2` and `EvalMesh2` commands to define and apply an evenly spaced mesh.

```
MapGrid2(un, u1, u2, vn, v1, v2)
```

sets up the mesh with *un* and *vn* divisions spanning the range *u1* to *u2* and *v1* to *v2*. Code is simplified by using ranges that span 0 to 1.

```
EvalMesh2(mode, i1, i2, j1, j2)
```

actually generates the mesh from *i1* to *i2* and *j1* to *j2*. The *mode* can be `POINT`, `LINE`, or `FILL`. The `EvalMesh2` command makes its own `Begin` and `End` clause.

Using the Mouse

Mouse activity is supported through two feedback functions. The Patch Editor.jsl sample script uses these functions to support the dragging and dropping of points. Part of that script, the call-back function for mouse activity, is explained below. To run the script, open PatchEditor.jsl in the Scene3D folder inside the JMP Sample Scripts folder.

```
topClick2d = Function( {x, y, m, k},
   dragfunc( x, boxhigh - y, m, 1, 2 );
   1;
);
frontClick2d = Function( {x, y, m, k},
   dragfunc( x, boxhigh - y, m, 1, 3 );
   1;
);
rightClick2d = Function( {x, y, m, k},
   dragfunc( x, boxhigh - y, m, 2, 3 );
   1;
);

Click3d = Function( {x, y, m, k, hitlist},
   If( m == 1,
      If( N Items( hitlist ) > 0,
         CurrentPoint = hitlist[1][3], /* first matrix in the list is the
   closest; 3rd element of matrix is ID*/
         CurrentPoint = 0
      );
      makePatch();
   );
   0; /* cares only about initial mouse down. return 1 if drag, release is
   needed, but then arcball does not happen. */
);

/* after one of the 3 Click2d functions figures out which axis of the model is
   represented by the screen X, Y, pass in to this common code */
dragfunc = Function( {x, y, m, ix, iy}, /* ix, iy are the index of the X, Y,
   or Z part of the coord in the points matrix */
   If( CurrentPoint > 0,
      points[CurrentPoint, ix] = (x / boxwide) * (orthoright - ortholeft) +
   ortholeft;
      points[CurrentPoint, iy] = (y / boxhigh) * (orthotop - orthobottom) +
   orthobottom;
      makepatch();
   )
);
```

When a 2-D function is called, the arguments are X, Y, M, K.

- X and Y are the coordinates of the mouse.
- M shows the state of the mouse and button. M=0 says that the mouse button is up. M=1 says that the button was just pressed. M=2 says that the button is down and the mouse is moving. M=3 says that the button was just released.
- K is related to the keys Shift, Alt, and Ctrl. K=1 for the Shift key, K=2 for the Ctrl (Command) key, and K=3 for the Alt (Option) key.

The 3-D function is called similarly. The arguments are X, Y, M, K, `hitlist`, where `hitlist` is a list of matrices

```
[znear, zfar, id1, id2, id3, ...]
```

`znear`, `zfar` is the Z distance from the camera of the near and far edge of the object. The matrices are sorted from near to far by the midpoint of `znear`, `zfar`. The `id`s in the list are the `pushname`, `loadname`, and `popname` values you just put in the display list.

The drag functions use a return value to tell if mouse processing should continue. That is the trailing "1" you see in the functions. Anything else stops the mouse tracking. This is needed because the 2-D and 3-D functions do not run in parallel. You might want the 2-D to return 0 and the 3-D to return 1 so the tracking would happen in 3-D rather than 2-D.

Pick Commands

This `SceneBox` callback gets the 2-D mouse coordinates, and then uses `Pick` to determine the "named" object under the mouse. For example, `hitlist` is a 5x5 pixel pick box around `x,y`; up to 1000 items returned, but just the leaf names. The format of the return is determined by the last argument (1 returns a simple array, 0 returns a sorted (by depth) list of arrays).

```
Track2d=function({x, y, m, k},
hitlist = theSceneBox<<pick( x, y, 5, 5, 1000, 1 );
  if ( nrow(hitlist) > 0, // something IS in the pick box
     ... hitlist[1..n] // are names that you put in the display list
  ) );
```

Contrast this with a call back `Track3d` function, where the pick rectangle is always 1x1 and picking only happens when the mouse moves. This is almost always what you want, but points are difficult to pick because the 1x1 pick area is the same small size as the point. This function lets you pick without a mouse move.

The `Track3d` function always provides a depth-sorted list of arrays; each array can describe multiple names in a hierarchy (pushname and popname construct a hierarchy of objects). The sorting can be very slow when thousands of objects are selected. The final argument (1, above) controls whether the pick function replaces the sorted list of arrays with a simple array. The simple array contains only the "leaf" names, not higher level names.

Three-Dimensional Arguments

Arguments enable you to specify special modes and settings. To enable an argument, use the `Enable(argument)` command. To disable an argument, use the `Disable(argument)` command. Available arguments are shown in Table 13.7.

Table 13.7 Arguments

ALPHA_TEST	LIGHT5	MAP2_TEXTURE_COORD_3
AUTO_NORMAL	LIGHT6	MAP2_TEXTURE_COORD_4
BLEND	LIGHT7	MAP2_VERTEX_3
CLIP_PLANE0	LIGHTING	MAP2_VERTEX_4
CLIP_PLANE1	LINE_SMOOTH	NORMALIZE
CLIP_PLANE2	LINE_STIPPLE	POINT_SMOOTH
CLIP_PLANE3	MAP1_COLOR_4	POLYGON_OFFSET_FILL
CLIP_PLANE4	MAP1_INDEX	POLYGON_OFFSET_LINE
CLIP_PLANE5	MAP1_NORMAL	POLYGON_OFFSET_POINT
COLOR_LOGIC_OP	MAP1_TEXTURE_COORD_1	POLYGON_SMOOTH
COLOR_MATERIAL	MAP1_TEXTURE_COORD_2	POLYGON_STIPPLE
CULL_FACE	MAP1_TEXTURE_COORD_3	SCISSOR_TEST
DEPTH_TEST	MAP1_TEXTURE_COORD_4	STENCIL_TEST
DITHER	MAP1_VERTEX_3	
FOG	MAP1_VERTEX_4	
LIGHT0	MAP2_COLOR_4	
LIGHT1	MAP2_INDEX	
LIGHT2	MAP2_NORMAL	
LIGHT3	MAP2_TEXTURE_COORD_1	
LIGHT4	MAP2_TEXTURE_COORD_2	

Improve the Display of Graphics

Hardware acceleration determines how fast and smoothly graphics are drawn. On Windows, hardware acceleration is turned off by default.

Note: JMP uses OpenGL 1.1 to perform 3-D rendering.

If you need OpenGL and have the hardware support, turn on hardware acceleration in one of the following ways:

- Right-click the graph and select **Use Hardware Acceleration**.
- Send the `Use Hardware Acceleration(1)` message to the scene box.

- In your script, set the global variable `Scene3DHardwareAcceleration = 1;` If you want hardware acceleration to be the default across runs of JMP, add it to your jmpStartAdmin.jsl file. Because it is a global variable, it will remain in effect until changed.

Chapter 14

Extending JMP

External Data Sources, Analytical Tools, and Automation

This chapter discusses scripting features that are particularly useful for production settings, such as the following:

- a data feed for capturing real-time data from a laboratory instrument
- using SAS, MATLAB, R, or Python through JMP Scripting Language (JSL)
- using JMP with Microsoft Excel
- connecting to databases
- controlling JMP externally by OLE automation
- parsing XML
- publishing to JMP Public or JMP Live
- communicating with REST web services
- communicating with web APIs

Some general JSL commands that might be of particular interest for use in a production setting include `Caption`, `Speak`, `Print`, `Write`, and `Mail`. These commands are described in "Functions that Communicate with Users".

Contents

Real-Time Data Capture on Windows . . . 797
Create a Data Feed Object . . . 797
Read in Real-Time Data . . . 799
Manage a Data Feed with Messages . . . 800
Examples of Data Feed . . . 802
Dynamic Link Libraries (DLLs) . . . 804
Use Sockets in JSL . . . 806
Socket-Related Commands . . . 807
Messages for Sockets . . . 808
Database Access . . . 809
Run Query Builder Queries . . . 809
Open Database Function . . . 810
Creating a Database Connection and Executing SQL . . . 812
Write a SQL Query . . . 813
Work with SAS . . . 814
Make a SAS DATA Step . . . 814
Create SAS DATA Step Code for Formula Columns . . . 814
SAS Variable Names . . . 815
Connect to a SAS Metadata Server . . . 815
Get the Values of SAS Macro Variables . . . 818
Preferences . . . 819
Sample Scripts . . . 819
Work with MATLAB . . . 821
Install MATLAB . . . 821
Work with R . . . 822
Installing R . . . 822
JMP to R Interfaces . . . 824
R JSL Scriptable Object Interfaces . . . 825
Conversion Between JMP Data Types and R Data Types . . . 825
Troubleshooting . . . 828
Examples . . . 829
Work with Python . . . 831
Install Python . . . 831
Troubleshooting the JMP Python Integration . . . 832
JMP-to-Python Interfaces . . . 836
Equivalent Data Types for Python Send() . . . 836
Equivalent Data Types for Python Get() . . . 837
JMP Scoping Operators and Python . . . 838
Python Name() and Python Send() Examples . . . 839

Additional Python Integration Examples . . . 840
Work with Microsoft Excel . . . 842
Parse XML . . . 842
OLE Automation . . . 844
Share Reports on JMP Live . . . 845
JMP Live Prerequisites . . . 845
Get an API Key . . . 845
Publish a Single Report to JMP Live . . . 846
Publish Several Reports to a JMP Live Folder . . . 847
Delete Reports in a JMP Live Folder . . . 847
Replace the Contents of a JMP Live Report . . . 848
Search for a JMP Live Report or Folder ID . . . 848
Update Data in JMP Live Reports . . . 849
Work with Other Content in JMP Live Reports . . . 850
HTTP Requests in JMP Live . . . 850
Communicate with REST Web Service . . . 850
Examples of HTTP Requests . . . 850
OAuth 2.0 for Web APIs . . . 853
Security . . . 853
Using Postman for Redirect URLs . . . 853
Authorization Code Grant . . . 854
Implicit Grant . . . 855
Resource Owner Grant . . . 855
Using a Token . . . 856
OAuth 2.0 Resources . . . 856
Communicate with SAS Cloud Analytic Services . . . 856
About CAS Actions . . . 857
Connect to the CAS Server . . . 858
Create and Submit a CAS Action . . . 859
Create and Submit a CAS DATA Step Action . . . 859
Create a New CAS Server . . . 859
Import Data from the CAS Server . . . 859
View a CAS Dataset . . . 860
Get a List of Available CAS Libraries . . . 860
Export Data from a Data Table to a CAS Server . . . 860
Convert a CAS Table to a Data Table . . . 860
Delete a CAS Table . . . 863
Remove a CAS Table . . . 863
Verify the CAS Server Connection . . . 863
Get Details about the Current Connection . . . 863

Disconnect from the CAS Server 863
Terminate All CAS Sessions 863

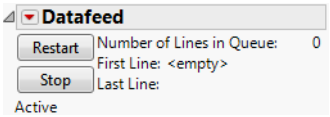
Real-Time Data Capture on Windows

A data feed is a real-time method to read data continuously, such as from a laboratory measurement device connected to a serial port. A data feed object sets up a concurrent thread with a queue for input data lines that arrive in real time and are processed with background events. You set up scripts to interpret the data lines and push the data to data tables, or do whatever else your process requires.

For example, submit this to get records from com1: and list them in the log.

```
feed = Open Datafeed(
  Connect( Port( "com1:" ), Baud( 9600 ), DataBits( 7 ) ),
  Set Script( Print( feed << GetLine ) )
);
```

Figure 14.1 A Datafeed Window Shows the Status and Offers Controls



Create a Data Feed Object

To create a data feed object, use the `Open Datafeed()` function with arguments specifying details about the connection:

```
feed = Open Datafeed( options );
```

No arguments are required. You can simply create the data feed object and then send messages to it. Messages might include connecting to a port or setting up a script to process the data coming in. However, you would typically set up the basic operation of the data feed in the `Open Datafeed()` function and subsequently send messages as needed to manage the data feed. Any of the options below work both as options inside `Open Datafeed()` or as messages sent to a data feed object.

It's a good idea to store a reference to the object in a global variable, such as *feed* above, so that you can easily send messages to the object. You can store a reference to an existing object by using a subscript; for example, to store a reference to the second data feed created:

```
feed2 = Datafeed[2];
```

Data Feed Options

To connect to a live data source, use `Connect( )` and specify details for the port. Each setting takes only one argument; in this syntax summary the symbol | between argument choices means "or." The `Port` specification is needed if you want to connect, otherwise the object still works, but is not connected to a data feed. The last three items, `DTR_DSR`, `RTS_CTS`, and `XON_XOFF`, are Boolean to specify which control characters are sent back and forth to indicate when the data feed is ready to get data. Typically, you would turn on at most one of them.

```
feed = Open Datafeed(
  Connect(
     Port( "com1:" | "com2:" | "lpt1:" | ... ),
     Baud( 9600 | 4800 | ... ),
     Data Bits( 8 | 7 ),
     Parity( None | Odd | Even ),
     Stop Bits( 1 | 0 | 2 ),
     DTR_DSR( 0 | 1 ), // Data Terminal Ready
     RTS_CTS( 0 | 1 ), // Request To Send | Clear To Send
     XON_XOFF( 1 | 0 ) // Transmitter On | Transmitter Off
  )
);
```

This command creates a scriptable data feed object and stores a reference to it in the global variable *feed*. The `Connect()` argument starts up a thread to watch a communications port and collect lines. The thread collects characters until it has a line, and then appends it to the line queue and schedules an event to call the script.

Note: A line of data from the feed might generate multiple rows in a data table or might be a fraction of one row. You provide JSL to parse the data lines and to add rows to a data table as needed.

`Set Script()` attaches a script to the data feed object. This script is triggered by the `Open Datafeed` message whenever a line of data arrives. The argument for `Set Script()` is simply the script to run, or a global containing a script.

```
feed = Open Datafeed( Set Script( myScript ) );
feed = Open Datafeed( Set Script( Print( feed << Get Line ) ) );
```

A data feed script typically uses `Get Line` to get a copy of one line and then does something with that line. Often it parses the line for data and adds it to some data table.

Read in Real-Time Data

The term *live data feed* describes the way an external data source sends information via a physical or a logical communication link to another device. You can connect JMP to a live data feed through the serial port of your Windows computer to read a stream of incoming data in real time. Remember the following:

- The data feed must come through a standard nine-pin serial port. Data cannot be read through a USB port unless there is a driver that can simulate a serial port.
- You need to know the exact baud rate, parity, stop bits, and data bits for the attached device.

Once you obtain the numbers for your device, enter them into the `Open Datafeed()` command in the script below. (The `4800`, `even`, `2`, and `7` in the script below are examples, so replace them with your information). Then connect the data feed to your computer and open and run the script:

```
streamScript = Expr(
   line = feed << Get Line;
   Show( line );
   len = Length( line );
   Show( len );
   If( Length( line ) >= 1,
      Show( "Hi" );
      Show( line );
      field = Substr( line, 5, 8 );
      Show( field );
      x = Num( field );
      Show( x );
      If( !Is Missing( x ),
         dt << Add Row( {:Column1 = x} );
         Show( x );
      );
   );
);
feed = Open Datafeed(
   Baud Rate( 4800 ),
   parity( even ),
   Stop bits( 2 ),
   Data bits( 7 )
);
feed << Set Script( streamScript );
feed << Connect;
```

To ensure harmony between the communications settings for JMP and the instrument reading data from an external source, select **File > Preferences > Communications**. Refer to the documentation for your instrument to find the appropriate settings.

Manage a Data Feed with Messages

A data feed object responds to several messages, including `Connect` and `Set Script`. These are detailed above as arguments for `Open Datafeed()`. They can also be sent as messages to a data feed object that already exists:

```
feed << Connect( port( "com1:" ), baud( 4800 ), databits( 7 ), parity( odd ),
  stopbits( 2 ) );
feed << Set Script( myScript );
```

The following messages could also be used as arguments to `On Datafeed()`. However, it would be more common to send them as messages to a data feed object that is already present.

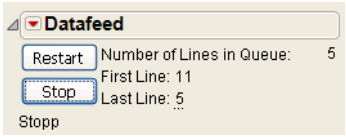
You can send lines to a data feed from a script. This is a quick way to test a data feed. If you do not have a device available, `Queue Line` does not send data to a device; it simulates getting data from a device. Include a text argument or a global that stores text:

```
feed << Queue Line( "14" );
feed << Queue Line( myValue );
```

Here is a test script to queue five lines of data:

```
feed << Queue Line( "11" );
feed << Queue Line( "22" );
feed << Queue Line( "33" );
feed << Queue Line( "44" );
feed << Queue Line( "55" );
```

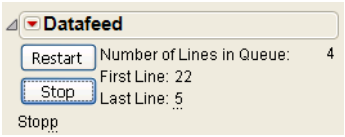
Figure 14.2 Datafeed: 5 Lines Queued



To get the first line currently waiting in the queue, use a `Get Line` (singular) message. When you get a line, it is removed from the queue. Five lines were queued with the test script above, and `Get Line` returns the first line and removes it from the queue:

```
feed << Get Line
   "11"
```

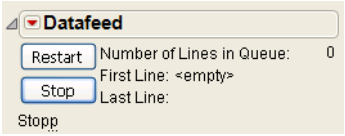
Figure 14.3 Datafeed: 4 Lines Queued



To empty all lines from the queue into a list, use `Get Lines` (plural). This returns the next four lines from the test script in list `{ }` format.

```
myList = feed << GetLines;
   { "22", "33", "44", "55" }
```

Figure 14.4 Datafeed: 0 Lines Queued



To stop and later restart the processing of queued lines, either click the **Stop** and **Restart** buttons in the Datafeed window, or send the equivalent messages:

```
feed << Stop;
feed << Restart;
```

To close the data feed and its window:

```
feed << Close;
```

To disconnect from the live data source:

```
feed << Disconnect
```

Table 14.1 Datafeed Messages

Message	Syntax	Explanation
`Open Datafeed` `Datafeed`	`feed = Open Datafeed( commands )`	Creates a data feed object. Any of the following can be used as commands inside `Open Datafeed` or sent as messages to an existing data feed object. `Datafeed` is a synonym.
`Set Script`	`feed << Set Script( script )`	Assigns the *`script`* that is run each time a line of data is received.
`Get Line`	`feed << Get Line`	Returns and removes one line from the data feed queue.

Table 14.1 Datafeed Messages *(Continued)*

Message	Syntax	Explanation
Get Lines	`feed << Get Lines`	Returns as a list and removes all lines from the data feed queue.
Queue Line	`feed << Queue Line( string )`	Sends one line to the end of the data feed queue.
Stop	`feed << Stop`	Stops processing queued lines.
Restart	`feed << Restart`	Restarts processing queued lines.
Close	`feed << Close`	Closes the data feed object and its window.
Connect	`feed << Connect( Port( "com1:" | "lpt1:" | ... ), Baud( 9600 | 4800 | ... ), Data Bits( 8 | 7 ), Parity( None | Odd | Even ), Stop Bits( 1 | 0 | 2 ), DTR_DSR( 0 | 1 ), RTS_CTS( 0 | 1 ), XON_XOFF( 1 | 0 ) )`	(Windows only) Sets up port settings for the connection to the device. The symbol | between arguments represents "or"; choose one argument for each setting.
Disconnect	`feed << Disconnect`	(Windows only) Disconnects the device from the data feed queue but leaves the data feed object active.

Examples of Data Feed

Reading Data

Here is a typical data feed script. It expects to find a string 3 characters long starting in column 11. If it does, it uses it as a number and then adds a row to the data table in the column "thickness."

```
feed = Open Datafeed( );
myScript = Expr(
   line = feed << Get Line;
   If( Length( line ) >= 14,
      x = Num( Substr( line, 11, 3 ) );
```

```
        If( !Is Missing( x ),
           dt << Add Row( {thickness = x} )
        );
    );
);
```

Assign the script to the data feed object by using `Set Script`:

```
feed << Set Script( myScript );
```

Set Up a Live Control Chart

Here is a sample script that sets up a new data table and starts a control chart based on the data feed.

```
dt = New Table( "Gap Width" ); // make a data table with one column
dc = dt << New Column( "gap", Numeric, Best );

// set up control chart properties
dc << Set Property(
  "Control Limits",
  {XBar( Avg( 20 ), LCL( 19.8 ), UCL( 20.2 ) )}
);
dc << Set Property( "Sigma", 0.1 );

// make the data feed
feed = Open Datafeed();
feedScript = Expr(
  line = feed << Get Line;
  z = Num( line );
  Show( line, z ); // if logging or debugging
  If( !Is Missing( z ),
     dt << Add Row( {:gap = z} )
  );
);
feed << Set Script( feedScript );

// start the control chart
Control Chart Builder(
       Show Capability( 0 ),
       Variables( Y( :gap ) ),
       Set Subgroup Size( 5 )
);
```

```
/* Either start the feed from the device or test-feed some data
to see it work (comment out one of the lines):
feed << Connect( Port( "com1:" ), Baud( 9600 ) );*/
For( i = 1, i <= 20, i++,
  feed << Queue Line( Char( 20 + Random Uniform() * .1 ) );
  Wait( .1 ); // simulate a delay in the feed
);
```

Store the Script in a Data Table

You can further automate the production setting by placing a data feed script such as the one above in an `On Open` data table property. A property with this name is run automatically each time the table is opened (unless you set a preference to suppress execution). If you save such a data table as a template, opening the template runs the data feed script and creates a new data table.

Dynamic Link Libraries (DLLs)

Note: 64-bit JMP cannot load 32-bit DLLs. You must recompile a 32-bit DLL for JMP to be able to load it.

You can extend JMP functionality by using JMP Scripting Language (JSL) to load a DLL and call functions exported by that DLL. There is one JSL command and six messages that implement this functionality.

```
dll_obj = Load DLL("path" <, AutoDeclare(Boolean | Quiet | Verbose) |Quiet |
  Verbose > )
```

`Load DLL()` loads the DLL in the specified *`path`*. Use the `AutoDeclare(Quiet)` argument to suppress log window messaging.

Use the `Declare Function` message to declare a function that is defined in the DLL.After you declare the function, you can call it.

```
dll_obj <<Declare Function("name", Convention(named_argument),
  Alias("string"), Arg(type, "string"), Returns(type), other_named_arguments)
```

The `Alias` defines an alternate name that you can use in JSL. For example, if you declared `Alias("MsgBox")` for a function that is named `"Message Box"` in the DLL, then you would call it as follows:

```
result = dll_obj <<MsgBox(...)
```

Here are the named arguments for `Convention`:

- STDCALL or PASCAL

- CDECL

The type argument for both `Arg` and `Returns` can be one of the following:

Table 14.2 Types for Arg and Returns

`Int8`	`UInt8`	`Int16`	`UInt16`
`Int32`	`UInt32`	`Int64`	`UInt64`
`Float`	`Double`	`AnsiString`	`UnicodeString`
`Struct`	`IntPtr`	`UIntPtr`	`ObjPtr`

See the *JSL Syntax Reference* for the `Declare Function` message arguments.

Finally, use the `UnLoadDLL` message to unload the DLL:

```
dll_obj << UnLoadDLL
```

Note: Refer to the documentation for that function provided by the DLL author when you declare a function. If the argument types and calling convention declared do not match the actual function in the DLL, calling the function could cause JMP to terminate.

Example

In the following example, the DLL contains a function named `netPresentValue`, which takes four parameters. `Declare Function()` defines these parameters and the data type that the function returns so that JMP can successfully call the function.

```
xftdll = Load Dll( "c:\ExtFunctionTests\ExtFunctionTests_x64.dll" ),

Show( xftdll );

xftdll << Declare Function(
  "netPresentValue",
 Convention( CDECL ),
  Alias( "npv" ), // use this name to call the function in JMP
  Arg( Double, "discount rate" ),
  Arg( Double, "cash flows per period" ),
  Arg( Int32, "number of cash flows" ),
  Arg( Double, Array, "array of cash flow values" ),
  Returns( Double )
);

result = xftdll << npv( 0.10, 1, 10, [ 100, 100, 100, 100, 100,
```

```
    100, 100, 100, 100, 100 ]
);
```

Other DLL Messages

The `Show Functions` message sends any functions that have been declared using `Declare Function()` to the log:

```
dll_obj << Show Functions;
```

If you are writing your own DLL, you can create functions in it using JSL. The `Get Declaration JSL` message sends any JSL functions in your DLL to the log:

```
dll_obj << Get Declaration JSL;
```

Use Sockets in JSL

Another tool that can be useful in establishing a live data feed is JSL Sockets. You can create two types of sockets using JSL:

Stream Stream sockets create a reliable connection between JMP and another computer. The other computer might be running JMP, or it might be a vending machine, data collector, printer, or other device that is capable of socket communication. Some devices implement their interface as an HTTP web server.

Datagram Datagram sockets create a less reliable connection between JMP and another computer. Datagrams are connectionless and information might arrive multiple times, not at all, and out of order. A datagram connection does not include all the overhead of a stream connection that does guarantee reliability. Because datagrams are connectionless, the destination address must be supplied each time (and for the same socket that can be different each time).

Once a socket is created, it can do two things: wait for a connection from another socket, or make a connection to another socket. Here is a simple example program that makes a connection to another computer's web server to get some data:

```
tCall = Socket( );
tCall << Connect( "www.jmp.com", "80" );
tCall << Send( Char To Blob( "GET / HTTP/1.0~0d~0a~0d~0a", "ASCII~HEX" ) );
tMessage = tCall << Recv( 1000 );
text = Blob To Char( tMessage[3] );
Show( text );
tCall << Close( );
```

The first line creates a socket and gives it a reference name (`tCall`). By default, a stream socket is created. You can designate the type of socket to create with an optional argument: `socket(STREAM)` or `socket(DGRAM)`.

The second line connects the `tCall` socket to port 80 (which is generally the HTTP port) of the JMP website.

The third line sends a `GET` request to the JMP web server; this message tells the JMP web server to send the JMP home page back. The / that follows the word GET should be the path to the page to be opened. A / opens the root page.

The fourth line receives up to 1000 bytes from the JMP web server and stores a list of information in `tMessage`. Each socket call returns a list. The first element of the list is the name of the call, and the second is a text message, which might be `ok` or a longer diagnostic message. Additional elements, if present, are specific to the call. In this case, the third element in the list is the data received.

The fifth line converts the binary information received into a character string. `tMessage[3]` is the third item in the list returned by `Recv`; it is the data from the JMP web server.

The sixth line displays the data in the log.

The last line closes the socket. The web server has already closed the far end, so this socket either needs reconnecting or proper disposal (`close`).

See the Scripting Index in the Help menu for more examples. The JMP Samples/Scripts folder also contains several examples of scripts using sockets.

Socket-Related Commands

Before creating and using a socket, you might need to retrieve information about the end that you want to connect to. `GetAddrInfo( )` and `GetNameInfo( )` each takes an address argument and an optional port argument and returns a list of information. For example:

```
Print( Get Addr Info( "www.sas.com" ) );
Print( Get Addr Info( "www.sas.com", "80" ) );
Print( Get Name Info( "149.173.5.120" ) );
   {"Get Addr Info", "ok", {"PF_INET", "SOCK_0", "IPPROTO_0", "149.173.5.120",
      "0"}}
   {"Get Addr Info", "ok", {"PF_INET", "SOCK_0", "IPPROTO_0", "149.173.5.120",
      "80"}}
   {"Get Name Info", "ok", {"PF_INET", "SOCK_0", "IPPROTO_0",
      "chess.exnet.sas.com", "0"}}
```

Sometimes there can be more than one answer. In that case, the sublist might be repeated one or more times. These functions can be quite slow; you probably should not try to build a data table of every website name with it. For IPV6 compatibility, you should generally use names like “www.sas.com” rather than the numerical form of an address.

Messages for Sockets

Once you have created a socket with `Socket( )`, there are many messages that you can send to it.

connect Connects to a listening socket. Returns a list: `{"connect", "ok"}` if the connection was successful; or an error if not (for example, `{"connect", "CONNREFUSED: The attempt to connect was forcefully rejected. "}`).

close Closes the connection when you are finished with it. Returns a list (for example, `{"Close", "ok"}`).

send Sends a STREAM message to the other end of the socket.

sendto Sends a DGRAM message to the other end of the socket.

recv Receives a STREAM message. The data comes back in a list, along with some other information. `Recv` takes a required numeric argument that specifies the number of bytes to accept.

recvfrom Receives a DGRAM message.

ioctl Controls the socket's blocking behavior. By default, JMP sockets block if no data is available; the socket does not return control to the JSL program until data *is* available. This makes scripts easy to write, but not particularly robust if the remote end of the connection fails to supply the data. A socket that is set for non-blocking behavior always returns immediately, either with an `ok` return code and some data, or with a "`WOULDBLOCK: ...`" return code, which means if it were a blocking socket, it would have to wait (block progress of the next JSL statement) until data became available.

Important: Background operations that use a JSL callback avoid this issue; a socket used in a background `recv`, `recvfrom`, or `accept` is set to non-blocking and is polled during `wait` statements and when JMP is otherwise idle.

`Ioctl` returns a list. For example, `{"ioctl", "ok"}`, or `{"ioctl", "NOTCONN: The socket is not connected. "}` if the socket has not been bound (see `bind`, below) or connected already.

bind Tells the server socket what address the client socket listens on. `Bind` associates a port on the local machine with the socket. This is required before a socket can `Listen`. (See below). `Bind` is not usually used on sockets that connect; the operating system selects an unused port for you. `Bind` is needed for a server because anyone that wants to connect to the server needs to know what port is being used. A common port is `80`, the HTTP port. `Bind` returns a list. For example, `{"bind", "ok"}`, or `{"bind", "ADDRNOTAVAIL: The specified address is not available from the local machine. "}` if you try binding to a name that is not on your machine. Another socket can connect to this socket if it knows your machine name and the number.

listen Tells the server socket to listen for connections. A listening socket is listening for connections from other sockets. You need only to put the socket into listen mode once. `Accept` (see below) is used to accept a connection from another socket. `Listen` returns a list. For example, `{"listen", "ok"}`, or `{"listen", "INVAL: The socket is (or is not, depending on context) already bound to an address. or, Listen was not invoked prior to accept. or, Invalid host address. or, The socket has not been bound with Bind. "}` if your bind call did not succeed.

accept Tells the server socket to accept a connection and return a new connected socket. `Accept` returns a list stating what happened, and if successful, a new socket that is connected to the socket at the other end. For example, the returned list might be `{"Accept", "ok", "localhost", socket( )}`. In this case `localhost` is the name of the machine that just connected, and the fourth argument is a socket that you can use to `send` or `recv` a message.

getpeername Asks about the other end of a connection. `GetPeerName` returns a list with information about the other end's socket: `{"getpeername", "ok", "127.0.0.1", "4087"}`. If this is a server socket, you can discover the address and port of the client that connected. If this is a client socket, you re-discover the name and port of the server that you used in the connect request.

getsockname Asks about this end of a connection. `GetSockName` returns a list with information about this socket: `{"getsockname", "ok", "localhost", "httpd"}`. If this is a client socket, you can discover the port the operating system assigned. If this is a server socket, you already know this information from a `bind` you already made.

Database Access

Run Query Builder Queries

Query Builder is the preferred method for selecting and importing data from a SQL database without writing SQL statements. You can preview the data before importing it into a data table. Share your queries so that other users can customize and run the queries.

After you create the query, you can run the query with a JSL script. Use `Include()` to include the query in your script without opening it.

```
query = Include( "$DOCUMENTS/Airline.jmpquery" );
query << Run Foreground( );
```

Instead of running the query in the foreground, you can use `Run Background()` to run the query in the background or `Run()` to use the Query Builder preference for running queries. By default, queries run in the background.

See the *JSL Syntax Reference* for more information about query messages.

Open Database Function

JMP supports ODBC access to SQL databases through JSL with the `Open Database` function.

```
dt = Open Database(
   "Connect Dialog" | "DSN=...", // data source
   "sqlStatement" | "dataTableName" | "SQLFILE=...", // SQL statement
   Invisible,  //Optional keyword to hide the table upon importing it
   "outputTableName" // new table name
);
```

Note: The characters `$# -+/%()&|;?` in a database table name must be quoted with square brackets [].

The first argument is a quoted connection string to specify the data source. It should be either of the following:

- `"Connect Dialog"` to display the Select Data Source window (Windows) or Choose DSN window (macOS).
- `"DSN="` and then the data source name and any other information needed to connect to the data source. On Windows, the data source name is shown in the name column of the User DSN or System DSN tab of the ODBC Data Source Administrator. On macOS, the DSN is shown in the ODBC Manager or iODBC Driver Manager. The rest of the string depends on the ODBC data source.

For example:

```
"DSN=dBASE Files;DBQ=C:/Program Files/SAS/JMP/16/Samples/Import Data;"
```

The second argument is a double-quoted string that can be one of the following:

1. An SQL statement to execute. For example, the second argument might be a `SELECT` statement in a quoted string like the following:

   ```
   "SELECT AGE, SEX, WEIGHT FROM BIGCLASS"
   ```

   The SQL must conform to the SQL that the data source supports. That is, a table named "Big Class" would need to be appropriately quoted because of the space between “Big” and “Class” (if whitespace is supported). The method for quoting depends on the data source and is usually done with ", ’, ‘, or [] characters.
2. The name of a data table. In this case, the effect is an SQL `"SELECT * FROM"` statement for the data table indicated. For example, `Open Database` would in effect execute the statement `"SELECT * FROM BIGCLASS"` if you specify this for the second argument:

   ```
   "BIGCLASS"
   ```

3. "SQLFILE=" and a path to a text file containing an SQL statement to be executed. For example, with the following argument, JMP would attempt to open the file mySQLFile.txt from the C:\ directory and then execute the SQL statement in the file.

   ```
   "SQLFILE=C:\mySQLFile.txt"
   ```

The optional `Invisible` argument creates a hidden data table. Hidden data tables remain in memory until they are explicitly closed, reducing the amount of memory that is available to JMP. To close the hidden data table, call `Close(dt)`, where `dt` is the data table reference.

The optional *outputTableName* argument is optional and specifies the name of the output table to be created, if any. Note that `Open Database` does not always return a data table. The return value might be null. Whether it returns a data table depends on the type of SQL statement executed. For example, a `SELECT` statement would return a data table, but a `DROP TABLE` statement would not.

To save a table back to a database through JSL, send the data table reference a `Save Database( )` message:

```
dt << Save Database( "connectInfo", "TableName" );
```

The first argument works the same way as it does in `Open Database`. Note that some databases do not allow you to save a table over one that already exists. If you want to replace a table in a database, use a DROP TABLE SQL statement in an Open Database command:

```
Open Database ( "connectinfo", "DROP TABLE TableName" );
```

Note: JMP 13 or later can save tables with spaces and mixed case in the names if the data source supports them. Most data sources do support spaces, except Apache Hive and Apache Hadoop. Mixed case is preserved, but for the most part, SQL is case insensitive.

The following script opens a database with an SQL query, saves it back to the database under a new name, and then deletes the new table.

```
dt = Open Database(
   "Connect Dialog",
   "SELECT age, sex, weight FROM \!"Bigclass$\!"",
   "My Big Class"
);
dt << Save Database( "Connect Dialog", "MY_BIG_CLASS" );
Open Database( "Connect Dialog", "DROP TABLE BIGCLASS.MY_BIG_CLASS" );
```

Note: When you import data from an ODBC database, a table variable is added that can contain user ID and password information. To prevent this from happening, set the following preference: `pref(ODBC Hide Connection String(1))`. You can also select the **File** menu (Windows) or **JMP** menu (macOS), **Preferences > Tables,** and then select **ODBC Hide Connection String**. See *Using JMP*.

Creating a Database Connection and Executing SQL

You can use the following functions to handle more complex database operations:

```
db = Create Database Connection( "Connection String With Password" );
Execute SQL( db, "SQL statement", <invisible>, <"New Table Title"> );
Close Database Connection( db );
```

Using these three functions, you can open a connection, call `Execute SQL` several times, and then close the connection. `Create Database Connection` returns a handle for use in `Execute SQL` and `Close Database Connection`.

Depending on the SQL submitted, a table might or might not be returned. A SELECT statement typically returns a JMP table. INSERT INTO would not return a table, because it is modifying one in the database.

Examples

Open a connection to your database:

```
dbc = Create Database Connection(
  "DSN=dBASE Files;DBQ=$SAMPLE_IMPORT_DATA/;"
);
```

Execute one or more SQL statements using this connection:

```
dt = Execute SQL( dbc,
  "SELECT HEIGHT, WEIGHT FROM Bigclass", "NewTable"
);
```

When you are finished, close your connection.

```
Close Database Connection( dbc );
```

You might want to create a database connection without selecting a table. The script would open the Database Open window with the connection already showing. We strongly recommend that you encrypt this script. Here's an example:

```
dbc = Create Database Connection(
  "DSN=dBASE Files;DBQ=$SAMPLE_IMPORT_DATA/;UID=MyDatabase;pwd=MyPassword");
Main Menu( "Open Table" );
```

Note: `Execute SQL()` creates a connection string that includes the cleartext password in the resulting data table. We encourage you to create the SQL query in Query Builder and then run the query in JSL. Then the password is not included in the data table.

Write a SQL Query

The `Query()` function enables you to manipulate JMP data tables using SQL statements. The SQL statement is the only required argument to `Query()`. However, most interesting SQL statements refer to one or more data tables. Any data table that the SQL statement refers to must be passed in using a *table reference* argument.

A table reference argument can be a data table reference (`dt` in the following example):

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
result = Query( dt, "SELECT * FROM 'Big Class' WHERE age > 14;" );
```

The SQL statement in the preceding example includes the name of the JMP data table. If the data table path is long, you might want to use an alias for the data table in the SQL statement. To use an alias, pass the table reference as a `Table()` argument:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
result = Query( Table( dt, "t1" ), "SELECT * FROM t1 WHERE age > 14;" );
```

Using `Table()` to alias your data tables also enables you to write SQL statements for multiple queries with tables that have different names without rewriting the statements.

Private and Hidden Data Tables

If you want the data table that is produced by `Query()` to be either private or hidden, you can pass either `Private` or `Invisible` as arguments to `Query()`.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
result = Query( dt, invisible, "SELECT * FROM 'Big Class' WHERE age > 14;" );
```

Returning a Single Value

The `Query()` function normally returns the result of the query as a JMP data table. However, sometimes you might write a SQL query that returns a single value; you would rather have `Query()` return that value instead of placing the value in a JMP data table. You can do that by passing the `Scalar` argument.

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
result = Query(
  Table( dt, "t1" ),
  Scalar,
  "SELECT AVG(height) FROM t1 WHERE age > 14;"
);
```

The preceding query returns 65, the average height of students older than 14.

Work with SAS

Make a SAS DATA Step

Sending `Make SAS Data Step` to a data table returns the text for a SAS DATA Step that can re-create the data table in SAS. For example,

```
dt << Make SAS Data Step
```

prints a DATA Step to the log that can be used in the SAS Program Editor.

Sending `Make SAS Data Step Window` produces this code in a window with a .SAS suffix, so that it can be easily sent to SAS.

Create SAS DATA Step Code for Formula Columns

Sending `Get SAS Data Step for Formula Columns` to a data table includes column formulas in the SAS data step code. Here is an example that outputs the formula for the Ratio column:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << New Column( "Ratio", Formula( :height / :weight ));
dt << Get SAS Data Step for Formula Columns;
```

This script output the following code to the log:

```
/*%PRODUCER: JMP - DataTable Formulas */
/*%TARGET: Ratio */
/*%INPUT: height */
/*%INPUT: weight */
/*%OUTPUT: Ratio */
/* Code to score Ratio */
Ratio =height/weight
drop ;
```

To get formulas for all columns, omit the column names as shown here:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Get SAS Data Step for Formula Columns;
```

You can also include column formulas in scoring code for SAS Model Manager. Send `Get MM SAS Data Step for Formula Columns` to the data table.

```
dt = Open( "$SAMPLE_DATA/Tiretread.jmp" );
dt << Get MM SAS Data Step for Formula Columns;
```

The results of this script are also shown in the log.

As with `Get SAS Data Step for Formula Columns`, specifying column names is optional.

SAS Variable Names

`SAS Open For Var Names()` opens a SAS data set only to obtain the names of its variables, returning those names as a list of strings.

The rules for SAS variable names are more strict than those of JMP. The `SAS Name` function converts JMP variable names to SAS variable names, changing special characters and blanks to underscores, and various other transformations to produce a valid SAS name.

```
result = SAS Name(name);
result = SAS Name({list of names});
```

If the argument is a list of names, the result is a blank-separated character string of names. For example,

```
SAS Name({“x 1”, “x 2”})
```

produces

```
“ x_1 x_2”
```

Connect to a SAS Metadata Server

You can connect to a SAS server and work directly with SAS data sets. Making connections and interacting with SAS data sets is scriptable through JSL. For information about connecting to a metadata server in the JMP interface, see *Using JMP*.

Note: If you connect to a physical workspace server, there is no metadata server involved, so metadata security is never applied. You must connect to a SAS Metadata Server and then connect to a logical workspace server. Then metadata security is enforced on the metadata-defined libraries you access.

Make the Connection

First, use the `Meta Connect()` function to connect to a SAS metadata server:

```
connected = Meta Connect( "MyMetadataServer", port );
```

To specify the SAS version for the metadata server, use the `SASVersion` named argument:

```
connected = Meta Connect( "MyMetadataServer", port, SASVersion("9.4") );
```

If you supply only the machine name (for example, myserver.mycompany.com) and the port, you are prompted to provide the authentication domain, your user name, and your password. You can also specify all that in JSL:

```
connected = Meta Connect( "MyMetadataServer", port, "authdomain", "user
  name", "password" );
```

When you are finished using the SAS metadata server, use `Meta Disconnect()` to disconnect the connection. No arguments are necessary; the command closes the current metadata server connection.

You can see the repositories that are available on a metadata server and set the one that you want to use:

```
Meta Get Repositories();
   {"Foundation"}
Meta Set Repository( "Foundation" );
```

Note that if there is only one repository available, it is selected automatically and you do not need to explicitly set it.

Once your repository is set, you can view the servers that are available:

```
mylist = Meta Get Servers();
   {"SASMain", "Schroedl", "SASMain_ja", "SASMain_zh", "SASMain_ko",
      "SASMain_fr", "SASMain_de", "SASMain_Unicode"}
```

Next, set your SAS connection. You can also use this command to connect directly to a local or remote server instead of using a metadata server.

```
conn = SAS Connect( "SASMain" );
```

Now you can send `Disconnect` and `Connect` messages to the `conn` object to close and open the SAS connection.

```
conn << Disconnect();
conn << Connect();
```

This is an example of using an object and messages with SAS server connections. You might have also connected to SAS servers using global functions. If so, the `Disconnect` and `Connect` messages do not affect those global connections. However, if there is no active global connection, that global connection is set to the connection opened by the object.

Automatically Connect SAS Libraries

When you connect to a SAS server, use `Connect Libraries` to automatically connect metadata-defined libraries.

```
conn = SAS Connect( "SASMain", Connect Libraries( 1 ) );
```

All metadata-defined libraries are connected, which can slow down your connection.

To connect specific libraries later, use the `SAS Connect Libref()` function or `Connect Libref` message to a SAS server object.

View SAS Libraries

After you connect to a SAS server, use the `Get Lib Ref()` function to view the libraries on that server:

```
conn << Get Lib Refs();
   {"BOOKS", "EGSAMP", "GENOMICS", "GISMAPS", "JMPSAMP", "JMPTEST",
   "MAILLIB", "MAPS", "OR_GEN", "ORION_RE", "ORSTAR", "SASHELP",
   "SASUSER", "TEMPDATA", "TSERIES", "V6LIB", "WORK", "WRSTEMP"}
```

If the library containing the data that you want is not assigned, assign it:

```
conn << Assign Lib Refs( "MyLib", "$DOCUMENTS/data" );
```

Open SAS Data Sets

First, assign a SAS library reference:

```
conn << Assign Lib Refs( "MyLib", "$DOCUMENTS" );
```

The first argument is any name that you want to use to refer to the library reference. The second is the path on the server where the data sets are located.

Next, get the list of data sets in the selected library:

```
datasets << Get Data Sets( "MyLib" );
   {"ANDORRA", "ANDORRA2", "ANYVARNAME", "BOOKS", "BOOKSCOPYNOT", "BOOKS_VIEW",
      "CATEGORIES", "DATETIMETESTS", "MOREUGLY", "NOTTOOUGLY", "PAYPERVIEW",
      "PUBLISHERS", "PURCHASES", "PURCHASES_FULL",
```

Now you can open a data set:

```
conn << Import Data( "MyLib", "PURCHASES" );
```

or

```
conn << Import Data( librefs[1], datasets[12] );
```

or

```
conn << Import Data( "MyLib.PURCHASES" );
```

Now you can get information about any SAS data set in that library. For example, you can get a list of variables:

```
conn << Get Var Names( "MyLib.PURCHASES" );
   {"purchaseyear", "purchasemonth", "purchaseday", "bookid", "catid",
   "pubid", "price", "cost"}
```

With that information, you can choose to import only part of the data set by specifying the variables to import.

```
conn << Import Data( librefs[1], datasets[12], columns( bookvars[1],
   bookvars[2], bookvars[4] ) );
```

Note: The name specified in `Import Data` can be a friendly metadata library name or a SAS 8-character library name.

Save SAS Data Sets

To save a JMP data table or an imported SAS Data Set, use the `SAS Export Data()` command:

```
conn << Export Data( dt, librefs[1], datasets[4], ReplaceExisting );
```

Run a Stored Process

To get a reference to a stored process:

```
stp = Meta Get Stored Process( "Samples/Stored Processes/Sample: Hello World"
  );
```

There is no way to acquire a list of stored processes through JSL; you must know the path to the stored process that you want to run.

To run it, send the stored process a message:

```
stp << Run();
```

Submit SAS Code from JMP

You can also directly submit SAS code and get back SAS results. For example:

```
conn << Submit( "proc print data=sashelp.class; run;" );
```

Two optional arguments control whether you see the output and the SAS log in JMP:

```
conn << Submit( "SAS Code" <,No Output Window(True|False)> <,Get
  Log(True|False )> );
```

You can also see the SAS Log at any time using the command

```
conn << Get Log();
```

`Get Log()` returns the contents, which can be placed in a JSL variable and used like any JSL string.

Get the Values of SAS Macro Variables

JMP provides several methods for querying SAS macro variables.

To show the `systime` value:

```
systime = sas << Get Macro Var("SYSTIME");
show(systime);
```

To show the defined SAS macro variables:

```
macro_names = sas << Get Macro Var Names();
show(macro_names);
```

To iterate through the SAS macro variables and print out the values:

```
macro_names = sas << Get Macro Var Names();
For( i = 1, i <= N Items( macro_names ), i++,
  macro_value = sas << Get Macro Var( macro_names[i] );
  output = macro_names[i] || " " || Char( macro_value );
  Show( output );
);
```

To submit SAS code that defines “test” as a SAS macro variable and then gets the value from SAS:

```
sas << Submit( "%let test = 1;" );
test = sas << Get Macro Var( "test" );
Show( test );
```

All macro variable values will evaluate to numbers, if possible, otherwise they will be characters.

Preferences

To get the current SAS version preference, use:

```
Get SAS Version Preference();
```

To set the current SAS version preference, use:

```
Preference( SAS Integration Settings( SASVersion( "9.4" ) ) );
```

Sample Scripts

The JMP Samples/Scripts/SAS Integration folder contains sample scripts. To run the stored process scripts successfully, the stored processes need to be placed on your SAS Metadata Server. The stored processes can be found in the sampleStoredProcesses.spk file, also in this folder.

To import sampleStoredProcesses.spk into your SAS Metadata Server

Caution: We recommend that you import these stored processes into a SAS Metadata Server that is used for testing rather than into a production system.

1. Run SAS Management Console.

2. Connect to your SAS Metadata Server using an account with administrative privileges.
3. Expand the BI Manager node in the left pane of the SAS Management Console.
4. Navigate to the folder in the tree under which you would like to create the imported sample stored processes.
5. Right-click that folder in either the left pane or the right pane of the SAS Management Console and select **Import**.

   The Import Wizard appears.
6. Enter the full path to sampleStoredProcesses.spk or use the **Browse** button to navigate to it.
7. Select **All Objects** in the Import Options section of the wizard.
8. Click **Next**.

   The next panel reports that during the import process, you must specify values for **Application servers** and **Source code repositories**.
9. Click **Next**.

   Select which of the application servers defined in your SAS Metadata Server that you would like to use to execute the imported stored processes.
10. Select an application server from the drop-down list under Target.
11. Click **Next**.

    Select the source code repository (directory) defined on your SAS Metadata Server where you would like the SAS code for the imported stored processes to be placed.
12. Select a source code repository from the drop-down list under Target Path.
13. Click **Next**.

    The next panel gives a summary of what occurs if you click **Import**.
14. Review the information about the panel, and if it looks correct, click **Import**.
15. During the Import process, you might be asked to provide login credentials for connecting to the metadata server to perform the import. Provide credentials with administrative privileges and click **OK**.

After the import completes, you will find a folder named BIP Tree under the folder that you imported the stored processes into. Under BIP Tree is a folder named JMP Samples, and in the JMP Samples folder are two sample stored processes: Shoe Chart and Diameter.

Please note that the paths to the sample stored processes needs to be adjusted in the sample scripts storedProcessHTML.jsl and storedProcessJSL.jsl to match the folder into which you imported the sample stored processes. Otherwise, these scripts will not work correctly.

Work with MATLAB

MATLAB, a product of MathWorks Inc., provides an interactive working environment for analyzing and visualizing computational models. MATLAB is available for Windows, Mac OS X, and Linux (64-bit). JMP supports MATLAB on both Windows and macOS platforms.

You can interact with MATLAB using JMP Scripting Language (JSL):

- Submit statements to MATLAB from within a JSL script.
- Exchange data between JMP and MATLAB.
- Display graphics produced by MATLAB.

See the *JSL Syntax Reference* for more information about using MATLAB functions in JMP.

Textual output and error messages from MATLAB appear in the log window.

Install MATLAB

MATLAB must be installed on the same computer as JMP. For the supported version of MATLAB, see the JMP website: https://www.jmp.com/system/.

How JMP Finds MATLAB

JMP delays loading MATLAB until a JSL-based script requires access to it. When you run a JSL script that calls MATLAB, JMP locates the software based on the operating system's `PATH` environment variable (for example, `C:\Program Files\MATLAB\R2012a\`).

Test Your Install

To verify that your computer is able to run JSL-based scripts using MATLAB:

1. Run the following JSL script:

```
MATLAB Init();
MATLAB Submit( "m = magic(3)" );
magicMat = MATLAB Get( m );
Show( magicMat );
MATLAB Term();
```

   The MATLAB function `M = magic(3)` returns a 3-by-3 matrix using integers in the range of 1 to 3^2 with equal row and column sums. This matrix is called a *magic square*.

2. Select **View > Log**.

   You should see the following response in the log window:

```
m =
```

```
      8     1     6
      3     5     7
      4     9     2


magicMat =
[ 8 1 6,
  3 5 7,
  4 9 2];
0
```

If you see the following message in the log window:

```
An installation of MATLAB cannot be found on this system.
```

1. Add a new environment variable with the `name` of `MATLABROOT` and a value of C:\Program Files\MATLAB\R2012a\ or C:\Program Files (x86)\MATLAB\R2012a\.

   Note: The path entered depends on the MATLAB installation path.

2. Verify that the `MATLAB` path is included in the `PATH` variable.
3. Run the script again to ensure JMP can access MATLAB.

Work with R

You can interact with R using JSL:

- Submit statements to R from within a JSL script.
- Exchange data between JMP and R.
- Display graphics produced by R.

Text output and error messages from R appear in the log window.

Installing R

Install R on the same computer as JMP. You can download R from the Comprehensive R Archive Network website:

https://cran.r-project.org

Note: For macOS Catalina and later versions, download the notarized and signed package.

Because JMP is a 64-bit application, you must install the corresponding 64-bit version of R. For the supported version of R, see the system requirements on the JMP website: https://www.jmp.com/system.

Override Default R Install Location on Windows

Normally JMP determines the R_HOME environment variable internally if it is not defined in the Windows system registry:

computer\HKEY_LOCAL_MACHINE\SOFTWARE\R-code\R\InstallPath

You might have installed more than one version of R on your computer. To override the default R installation location, define the R_HOME environment variable using either of the two following methods:

1. Create the variable in your system environment variables using the Control Panel, select **Start > Control Panel > System > Advanced system settings**.
2. Click **Environment Variables**.
3. In the System variables pane, click **New**.
4. Type R_HOME for the **Variable name**.
5. Type the path to the R .exe file (for example, C:\Program Files\R\R-2.15.3).
6. Click **OK** and click **OK** again to close the System Properties window.

or

Create the variable using the JSL `Set Environment Variable()` function:

```
Set Environment Variable( "R_HOME", "C:\Program Files\R\R-2.15.3" );
```

Override the Default R Install Location on macOS

You might have installed more than one version of R on your computer. To use an older version, change the R_HOME environment variable to point to the R version that you want to use.

To specify the R_HOME location, follow these steps:

1. Open a terminal window.
2. Type the following code, where *value* is the location of R.

   `export R_HOME=value`
3. Press Enter.

How JMP Finds R on Windows

JMP delays loading R until a JSL-based script requires access to it. When JMP needs to load R, it follows the standard steps for finding R on a Windows computer:

1. Look up the environment variable `R_HOME`.

   If the variable exists, load R from the specified directory.
2. If the environment variable `R_HOME` does not exist, look up the `InstallPath` value in the Windows registry under the following key:

   `HKEY_LOCAL_MACHINE\SOFTWARE\R-core\R`

   If the `InstallPath` value exists, load R from the specified directory.
3. If the `InstallPath` value does not exist, an error message states that R could not be found.

How JMP Finds R on macOS

JMP uses the value of the `R_HOME` environment variable if the variable exists. Otherwise, it looks in `/Library/Frameworks/R.framework/Versions/Current/Resources` for R.

Testing Your Setup

To test that your computer is able to run JSL-based scripts that use R, run the following JSL script:

```
R Init( );
R Submit( "
   x <- 1:5
   x
" );
R Term( );
```

You should see the following output in the log:

```
[1] 1 2 3 4 5
```

JMP to R Interfaces

The following JMP interfaces are provided to access R. The basic execution model is to first initialize the R connection, perform the required R operations, and then terminate the R connection. In most cases, these functions return 0 if the R operation was successful, or an error code if it was not. If the R operation is not successful, a message is written to the log. The single exception to this is `R Get()`, which returns a value.

R JSL Scriptable Object Interfaces

The R interfaces are also scriptable using an R connection object. A scriptable R connection object can be obtained using the `R Connect()` JSL function.

Conversion Between JMP Data Types and R Data Types

Table 14.3 shows what JMP data types can be exchanged with R using the `R Send( )` function. Sending lists to R recursively examines each element of the list and sends each base JMP data type. Nested lists are supported.

Table 14.3 Equivalent JMP and R Data Types for R Send( )

R Data Type	JMP Data Type
Double	Numeric
String	String
Double Matrix	Matrix
List	List
Data Frame	Data Table
Integer	Row State
Date and Time	Datetime
Time/Duration	Duration
Paired List	Associative Array

Example

```
R Init();
X = 1;
R Send( X );
S = "Report Title";
R Send( S );
M = [1 2 3, 4 5 6, 7 8 9];
R Send( M );
R Submit( "
X
S
M
" );
```

```
R Term();
```

Table 14.4 shows what JMP data types can be exchanged with R using the `R Get()` function. Getting lists from R recursively examines each element of the list and sends each base R data type. Nested lists are supported.

Table 14.4 Equivalent JMP and R Data Types for R Get( )

R Data Type	JMP Data Type
Double	Numeric
Logical (Boolean)	Numeric (0 | 1)
String	String
Integer	Numeric
Date and Time	Datetime
Time/Duration	Duration
Factor	Expanded to a list of Strings or a Numeric matrix
Data Frame	Data Table
List	List of converted R data types
Matrix	Numeric Matrix
Numeric Vector	Numeric Matrix
String Vector	List of Strings
Graph	Picture
Time Series	Matrix
Paired List	Associative Array

JMP Scoping Operators and R

A JMP object sent to R using `R Send()` uses the same JMP reference as the name of the R object that gets created. For example, sending the JMP variable `dt` to R creates an R object named `dt`. The colon and double colon scoping operators (`:` and `::`) are not valid in R object names, so these are converted as follows:

- A single colon scoping operator is replaced with a period (`.`).

  For example, sending `nsref:dt` to R creates a corresponding R object named `nsref.dt`.

- A double colon scoping operator (designating a global variable) is ignored.

  For example, sending `::dt` to R creates a corresponding R object named `dt`.

Using R Name() with R Send()

The `R Name()` option to `R Send()` has an argument that is a quoted string that contains a valid R object name. The JMP object sent to R becomes an R object with the name specified. For example:

```
R Send( jmp_var_name, R Name( "r_var_name") );
R Submit( "print(r_var_name)" )
```

Example

This example creates a variable `x` in the `Here` namespace, a variable `y` in the global namespace, and a variable `z` that is not explicitly referenced to any namespace. The variable `z` defaults to Global unless `Names Default To Here(1)` is on. These variables are then passed to R.

```
Here:x = 1;
::y = 2;
z = 3;

R Init(); // initiate the R connection

/* send the Here variable to R
Here:x creates the R object Here.x */
R Send( Here:x );

/* note that the JMP log labels the output with the original JMP variable
   reference Here:x */
R Submit( "print(Here.x)" );

R Send( ::y ); // ::y create the R object y
R Submit( "print(y)" );

// to use a different name for the R object, use the R Name() option
R Send( Here:x, R Name( "localx" ) );

/* The R Name option to the R Send() command creates the R object named
   "localx"" corresponding to the JMP variable "Here:x". Again the log shows
   the original corresponding JMP variable name. */
R Submit( "print(localx)" );
R Send( z ); // z creates the R object z
R Submit( "print(z)" );
```

Troubleshooting

Recording Output

On Windows, if you want to record output to the graphics window, send the following R code using `R Submit( )`.

```
windows.options( record = TRUE );
```

Character Vectors

A JMP list of strings is not the same as an R character vector. If you send a list of strings to R, it becomes a list of strings in R, not a character vector. You can use the R function Unlist to convert it:

```
R Init();
X = {"Character", "JMP", "List"};
R Send( X );

/* R output is:
[1] "list"
*/
R Submit( "class(X)" );

/* Object Y is now a character vector. The R output is:
[1] "character"
*/
R Submit( "Y<-unlist(X)
     class(Y)" );
R Term();
```

Element Names

A feature of an R list (called attributes) lets you associate a name with each element of the list. You can use the name to access that element instead of having to know the position of it in the list

In the following example, the list that is created in R has two elements named x and y that are created using the `List()` function of R. When your bring the R list into JMP and then send it back to R, the names are lost. Therefore in R, you cannot access the first matrix using `pts$x`. Instead, you must use the index using `pts[[1]]`.

```
R Init();
R Submit("
   pts <- list(x=cars[,1], y=cars[,2])
   summary(pts)
");
```

```
JMP_pts = R Get( pts );

R Send( JMP_pts );
R Submit("
   Summary( JMP_pts )
");
R Term();
```

Examples

Sending a Data Table to R

This example initiates an R connection, sends a data table to R, prints it to the log, and closes the R connection.

```
R Init();
dt = Open( "$SAMPLE_DATA/Big Class.jmp", invisible );
R Send( dt ); // send the opened data table represented by dt to R;
R Submit( "print( dt )" );
R Term();
```

Creating Objects in R

This example initiates an R connection, creates an R object, retrieves the object into JMP, and closes the R connection.

```
R Init();
R Submit(
   "
   L3 <- LETTERS[1:3]
   d <- data.frame(cbind(x=1, y=1:15), Group=sample(L3, 15, repl=TRUE))
"
);
R Get( d ) << NewDataView;
R Term();
```

Using R Functions and Graphics

This example initiates an R connection and plots the normal density function in R using the R graphics device,. Then the graph is retrieved from R and displayed in JMP. Finally, the R connection is closed.

```
R Init();
R Submit( "\[plot(function(x) dnorm(x), -5, 5, main = "Normal(0,1) Density")
   ]\" );
```

```
picture = R Get Graphics( "png" );
New Window( "Picture", picture );
Wait( 10 );
R Term();
```

Simple Matrix Addition in R

This example initiates an R connection, sends a matrix to R, creates a matrix in R, adds them together, returns the new matrix to JMP, and closes the R connection.

```
R Init();
X = J( 2, 2, 1 );
R Send( X );
R Submit(
  "
  X                                    #prints X to the log
  Y <- matrix(1:4, nrow=2, byrow=TRUE)  #makes a 2x2 matrix object Y
  Y                                    #prints Y to the log
  Z <- X + Y #matrix object Z is addition of X and Y
"
);
Z = R Get( Z );
R Term();
Show( Z );
```

A Bootstrap Sample

See the file JMPtoR_bootstrap.jsl in the sample scripts folder for an example script.

This script performs a bootstrap simulation by using the JMP to R Project integration.

The script produces a JMP window that asked the user to specify the variable to perform bootstrapping over. Then the user selects a statistic to compute for each bootstrap sample. Finally, the data is sent to R using the R interface in JSL.

The boot package in R is used to call the `boot()` function and the `boot.ci()` function to calculate the sample statistic for each bootstrap sample and the bootstrap confidence interval.

The results are brought back to JMP and displayed using the JMP Distribution platform.

Work with Python

The JMP-Python integration allows JMP users to interact with a large amount of scientific, statistical, and data analysis tools written in Python. JMP implements this capability by loading the Python shared library into the running JMP process. JSL gives Python access to JMP.

The JMP installer does not contain Python. Several Python implementations are available. Visit https://www.python.org/download/alternatives to view the options.

JMP has several methods for finding an instance of Python installed on the computer. You should not need to set any environment variables or additional arguments to the JSL `Python Init()` statement. See "Troubleshooting the JMP Python Integration" for more information.

The Python integration enables you to perform the following tasks:

- submit statements to Python from within a JSL script
- exchange data between JMP and Python
- display graphics produced by Python

Text output and error messages from Python appear in the log window.

Install Python

JMP 16 is a 64-bit program; only 64-bit Python is supported

- Python.org 3.6 through 3.9.1 distributions are supported on Windows and macOS. On Apple silicon, you will not have full functionality until third-party packages have been updated.
- Anaconda 3.6 distribution is supported on Windows only.
- Anaconda 3.7 distribution is not supported on Windows.
- The way Python is built from versions 3.6 to 3.7 changed. In Python 3.7, Python must be installed for all users. Otherwise, the libraries that the Python DLL depend on cannot be located at DLL load time. Other Python distributions might work fine installed for a single user or might also require a system-wide install.
- JMP does not currently support Python virtual environments.

Notes:

- Python is not distributed with JMP. Download Python from the Python web site at https://python.org.
- Several Python implementations are available. Visit https://www.python.org/download/alternatives to view the alternatives.

- On macOS, installing Python 3.6.x from https://python.org installs in /Library/Frameworks/Python.framework.

Required Python Packages

For the JMP-to-Python interfaces to work properly, you must install certain Python packages and their prerequisites. In many Python installations, such as Anaconda, these packages are already installed. If you install Python from Python.org, also install the following packages:

numpy https://www.numpy.org

pandas https://pandas.pydata.org

matplotlib https://matplotlib.org

scipy https://www.scipy.org

sqlite3 https://www.sqlite.org

PyQt5 https://pypi.org/project/PyQt5/

Use the standard Python update tools to install these packages for your Python configuration. You must use the correct pip-tool for your installation. The pip-tool name is dependent on the what product has installed Python. The installation instructions for Python have instructions on using pip tools.

Troubleshooting the JMP Python Integration

Sometimes JMP has trouble locating the correct version of Python. The first thing to try is running Python from the command line. If Python doesn't run from a terminal or console window, something is wrong with the Python installation.

Note: Before you proceed with troubleshooting, verify that you're running a supported version of Python. See "Install Python".

How JMP Finds Python on Windows

JMP delays loading Python until a JSL-based script requires access to it. When JMP loads Python, it needs to be able to find the Python shared library and any of the shared libraries that the Python DLL depends on. Once the Python interpreter is loaded, Python needs to be able to find the distribution's standard Python modules. To guide this process, JMP uses registry entries, environment variables, and user-specified paths.

Python also looks for certain environment variables. Avoid using any user-specified paths or environment variables, and let JMP find Python in the registry. The following methods help JMP locate Python if the automatic location doesn't work:

- JMP_PYTHON_MODULE_PATH is a JMP-specific version of PYTHONPATH. This variable is the replacement for Python Sys Path. In JMP_PYTHON_MODULE_PATH, specify standard OS paths. Separate the paths with a colon on macOS. Separate the paths with a semicolon on Windows.
- JMP_LIB_PYTHON_PATH is a JMP-specific version of PYTHONHOME. The variable contains the full path to the Python shared library, or the parent directory to the shared library.

Windows finds Python in the following order of precedence:

1. JMP looks at the environment variables Python Sys Path (deprecated), PYTHONINSTALLPATH (deprecated), and JMP_PYTHON_MODULE_PATH, in that order. JMP uses the first variable that is found.
2. JMP checks JMP_LIB_PYTHON_PATH for a path to the Python shared library.
3. JMP checks for options that are specified in the Python Init() or Python Connect() function calls. Path() corresponds to the location of the Python shared library. Use Python Version() allows you to specify the version of Python to look for.

   Python Sys Path provides the mechanism to set Python's sys.path variable. If present, script parameters take precedence over the environment variables. There is one caveat. After JMP loads the Python shared library, the library is unloaded only when JMP shuts down. As a result, the Path() portion is only effective on the first load of Python, and does not allow for running different versions of Python in different scripts without shutting down JMP and restarting.
4. JMP reads the Windows registry to determine if Python has been installed. It looks at appropriate locations based on whether the installation was for an individual user or for all users on the Windows machine. If you have specified a Python version, other versions of Python will not be considered.

If JMP cannot locate Python, try the following option in the Python Init() expression:

```
Python Init( Init Trace( "TRUE" ) );
```

JMP prints details to the log about where it detects Python, including paths and registry entries for Python.

Typically, JMP determines the equivalent to the PYTHONHOME environment variable automatically if it is defined in the Windows registry.

Computer\<root>\SOFTWARE\Python\PythonCore\<version>\InstallPath

<root> is either the root key HKEY_LOCAL_MACHINE or HKEY_CURRENT_USER depending on where and how you installed Python. *<version>* is the Python version number.

If multiple versions of Python are installed on your computer, you must specify which installation for JMP to use.

- The easiest method is to specify the version using the Use Python Version() argument in Python Init() or Python Connect() functions.

- Use the `Path("path to an installed Python DLL or shared library")` argument with the `Python Init()` or `Python Connect()` function.
- Define the `JMP_LIB_PYTHON_PATH` environment variable as described below.

Change the Default Python Install Location on Windows

Create the variable in your system environment variables using the Control Panel.

1. Select **Start > Control Panel > System > Advanced system settings**.
2. Click **Environment Variables**.
3. In the System variables pane, click **New**.
4. Type `JMP_LIB_PYTHON_PATH` for the Variable name.
5. Type the path to the Python<version>.dll file, for example:
   `C:/Program Files/Python/<version>/Python<version>.dll`
6. Click **OK** and click **OK** again to close the System Properties window.

macOS: Override How JMP Finds Python

Locating Python on the Mac works the same as on Windows except for the step of looking in the registry. JMP looks in a couple of typical installation directories so see if it can locate Python. Python typically installs in `/Library/Frameworks/Python.framework`. A compatible version of Python located in the standard system location should work without any changes to environment variables or parameters passed to JSL.

If Python is installed in an alternate location, set configuration variables in your shell environment as needed. JMP on macOS uses the same environment variables as described in "How JMP Finds Python on Windows".

`JMP_LIB_PYTHON_PATH` is the path to the Python shared library `libpython3.x.dylib`, where *x* is the minor version. If environment variables are needed, you typically only need to set `JMP_LIB_PYTHON_PATH`.

If JMP is unable to find install Python packages, you might also need to set the variable `JMP_PYTHON_MODULE_PATH`. The `JMP_PYTHON_MODULE_PATH` contains the same paths that the Python call print(sys.path) outputs. When you set an environment variable, use host OS path conventions. When the variable is passed in JSL, use a JSL list.

One way is to have Python tell you what it thinks its `sys.path` should be. In the example in this section, the same Python commands can be run on Windows to determine `sys.path`.

In the terminal or console window, enter the following boldface commands to get a string that specifies the location of the Python packages:

```
MDz320:~ shlori$ python --version
Python 3.6.4 :: Anaconda, Inc.
MDz320~ shlori$ python3
Python 3.6.4 |Anaconda, Inc.| (default, Jan 16 2018, 12:04:33) [GCC 4.2.1
  Compatible Clang 4.0.1 (tags/RELEASE_401/final)] on darwin Type "help",
  "copyright", "credits" or "license" for more information.
>>> import sys
>>> print(sys.path)
  ['', '/anaconda3/lib/python36.zip', '/anaconda3/lib/python3.6',
     '/anaconda3/lib/python3.6/lib-dynload',
     '/anaconda3/lib/python3.6/site-packages',
     '/anaconda3/lib/python3.6/site-packages/aeosa']
```

Test Your Setup

To test that your computer is able to run JSL scripts that use Python, run the following script:

```
Python Init();
Python Submit( "\[basket = ['apple', 'orange', 'pear']
print(basket)]\" );
Python Term();
```

You should see the following output in the log:

```
['apple', 'orange', 'pear']
0
```

JMP-to-Python Interfaces

JMP provides the interfaces to access Python that are described in the following sections. The basic execution model is to first initialize the Python connection, perform the required Python operations, and then terminate the Python connection. In most cases, these functions return 0 if the Python operation was successful or an error code if it was not. If the Python operation is not successful, a message is written to the log. The single exception to this is `Python Get()`, which returns a value.

Python JSL-Scriptable Object Interfaces

The Python interfaces are also scriptable using a Python connection object. A scriptable Python connection object can be obtained using the `Python Connect()` function.

```
// return a Python connection scriptable object
PythonConnection = Python Connect();

// return 1 if a Python connection is active. Otherwise, returns 0.
x = PythonConnection << Is Connected;
Show( x );
PythonConnection << Disconnect;
   x = 1;
   0
```

Equivalent Data Types for Python Send()

Table 14.5 shows what JMP data types can be exchanged with Python using the `Python Send()` function. Sending lists or associative arrays to Python recursively examines each element of the list or associative array and sends each base JMP data type. Nested lists and associative arrays are supported.

Table 14.5 Equivalent JMP and Python Data Types for Python Send()

JMP Data Type	Python Data Type
Boolean	Boolean
Data Table	Pandas.DataFrame
Associative Array	Dictionary

Table 14.5 Equivalent JMP and Python Data Types for Python Send() *(Continued)*

JMP Data Type	Python Data Type
Numeric	Float
Matrix	Float Matrix
List	List
String	Unicode String

Python Send() Example

```
Python Init();
X = 1;
Python Send( X );
S = "Report Title";
Python Send( S );
M = [1 2 3, 4 5 6, 7 8 9];
Python Send( M );
Python Submit( "\[
print(X)
print(S)
print(M)
]\" );
Python Term();
```

```
1.0
Report Title
[[ 1.  2.  3.]
 [ 4.  5.  6.]
 [ 7.  8.  9.]]
0
```

Equivalent Data Types for Python Get()

Table 14.6 shows which JMP data types can be exchanged with Python using the `Python Get()` function. Getting lists from Python recursively examines each element of the list and gets each base Python data type. Nested lists are supported.

Table 14.6 Equivalent Python and JMP Data Types for Python Get()

Python Data Type	JMP Data Type
Array	Numeric Matrix
Date and time	Numeric

Table 14.6 Equivalent Python and JMP Data Types for Python Get() *(Continued)*

Python Data Type	JMP Data Type
Pandas.DataFrame	Data Table
Dictionary	Associative Array
Graph	Picture Object
Integer	Integer
List	List of converted Python data types
Matrix	Numeric Matrix
Set	List
Unicode String	String

Python Get() Example

```
Python Init();
x1 = [1, 2, 3];
Python Send( x1 );
x2 = Python Get( x1 );
Show( x1, x2 );
dt1 = Open( "$SAMPLE_DATA/Big Class.jmp" );
Python Send( dt1 );
dt2 = Python Get( dt1 );
dt2 << New Data View;
Close( dt1 );
Python Term();
```

JMP Scoping Operators and Python

A JMP object sent to Python with `Python Send()` uses the same JMP reference as the name of the Python object that gets created. For example, sending the JMP variable `dt` to Python creates a Python object named `dt`.

The colon and double colon JSL scoping operators (`:` and `::`) are not valid in Python object names, so the colons are converted as follows:

- A single colon scoping operator is replaced with an underscore (_).

  For example, sending `nsref:dt` to Python creates a corresponding Python object named `nsref_dt`.
- A double colon scoping operator (designating a global variable) is ignored.

For example, sending `::dt` to Python creates a corresponding Python object named `dt`.

Note: The `Python Name` option for `Python Send()` has a quoted string argument that contains a valid Python object name. The JMP object sent to Python becomes a Python object with the specified name.

Python Name() and Python Send() Examples

The following example creates the `jmp var name` variable, assigns it to the Python name `python_var_name`, submits the `print` statement, and closes the connection.

```
Python Init();
jmp var name = 25;
Python Send( jmp var name, Python Name( "python_var_name" ) );
Python Submit( "print(python_var_name)" );
Python Term();
```

```
25.0

0
```

The following example creates a variable *x* in the Here namespace, a variable *y* in the global namespace, and a variable *z* that is not explicitly referenced to any namespace. The variable *z* defaults to Global unless `Names Default To Here(1)` is on. These variables are then passed to Python.

```
Here:x = 1;
::y = 2;
z = 3;

Python Init(); // initiate the Python connection

/* send the Here variable to Python
Here:x creates the Python object Here_x */
Python Send( Here:x );

Python Submit( "print(Here_x)" );

Python Send( ::y ); // ::y creates the Python object y
Python Submit( "print(y)" );

// to use a different name for the Python object, use the Python Name() option
Python Send( Here:x, Python Name( "localx" ) );

/* The Python Name option to the Python Send() command creates the Python
   object named "localx", which corresponds to the JMP variable "Here:x". */
Python Submit( "print(localx)" );
```

```
Python Send( z ); // z creates the Python object z
Python Submit( "print(z)" );
Python Term();
  1.0

  2.0

  1.0

  3.0

  0.0
```

Additional Python Integration Examples

Send a Data Table to Python

This example initiates a Python connection, sends a data table to Python, prints it to the log, and closes the Python connection.

```
Python Init();
dt = Open( "$SAMPLE_DATA/Big Class.jmp", invisible );
Python Send( dt ); // send the opened data table represented by dt to Python
Python Submit( "print( dt )" );
Python Term();
          name     age   sex   height   weight
  0       KATIE    12    F       59       95
  1      LOUISE    12    F       61      123
  2        JANE    12    F       55       74
  ...
  0
```

Create Objects in Python

This example initiates a Python connection, creates a Python object, retrieves the object into JMP, and closes the Python connection.

```
Python Init();
Python Submit("\[
import numpy as np
import pandas as pd
# Generate basic series
ss = pd.Series(np.random.randn(5), index=['a', 'b', 'c', 'd', 'e'])
print(ss)
sn = pd.Series(np.random.randn(5))
print(sn)

# Generate Data Frames from Series
```

```
s1 = pd.Series(np.random.randn(5), index=['a', 'b', 'c', 'd', 'e'])
print(s1)
dfs1 = pd.DataFrame(s1)
print(dfs1)

d1 = {'one' : pd.Series([1., 2., 3.], index=['a', 'b', 'c']),
      'two' : pd.Series([1., 2., 3., 4.], index=['a', 'b', 'c', 'd'])}
print(d1)
dfd1 = pd.DataFrame(d1)
print(dfd1)

d2 = {'one' : [1., 2., 3., 4.],
      'two' : [4., 3., 2., 1.]}
print(d2)
dfd2 = pd.DataFrame(d2)
print(dfd2)
]\"
);
ss = Python Get( ss );
Show( ss );
sn = Python Get( sn );
Show( sn );
dfs1 = Python Get( dfs1 );
dfs1 << New Data View;
dfd1 = Python Get( dfd1 );
dfd1 << New Data View;
dfd2 = Python Get( dfd2 );
dfd2 << New Data View;
Python Term();
   a   -0.384649
   b    0.455864
   c   -2.214405
   d    0.260955
   e    1.342286
   dtype: float64
   ...
```

Matrix Operations in Python

In JMP, a simple matrix looks like this:

```
xx = [1 2 3, 4 5 6, 7 8 9 ];
```

In numpy, the matrix is an array:

```
xx = numpy.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
```

In order to submit a matrix as a numpy array, you must translate the JMP matrix.

```
q = "new = numpy.array ([";
```

```
For( i = 1, i <= N Items( xx ), i++,
   zz = Matrix( xx[i] );
   q = q || Char( zz ) || ",";
);
q = q || "])";
Show( q );

Python Submit( "import numpy
" || q );
Python Submit( "print(new)" );
```

Work with Microsoft Excel

You can script the Profiler interface to Microsoft Excel, although not the Transfer to JMP interface. Here is the basic syntax:

```
excel_obj = Excel Profiler(
   Workbook( "excel_workbook_path" ),
   Model( "name_of_model" )
);
```

The `Model` argument is optional. If you supply only the workbook, you are prompted to select the model. You do not need to specify the worksheet, because the model is found no matter where it is located.

Once you have an Microsoft Excel Profiler object, you can send the following messages to run the JMP prediction profiler using the Microsoft Excel model. For example:

```
excel_obj <<Prediction Profiler( 1 );
```

Parse XML

JSL has several commands available to parse XML.

`Parse XML( `*`string`*`, On Element( "`*`tagname`*`", Start Tag( `*`expr`*` ), End Tag( `*`expr`*` ) ) );`

parses an XML expression using the `On Element()` expression for specified XML tags.

`value = XML Attr( "`*`attribute name`*`" );`

extracts the string value of an XML argument when evaluating a `Parse XML()` expression.

```
value = XML Text();
```

extracts the string text of the body of an XML tag when evaluating a `Parse XML()` expression.

Example of Parsing XML

Suppose that a Microsoft Excel file contains one row of data from Big Class.jmp. The file is saved as the valid XML document BigclassExcel.xml, shown here and also saved in the JMP Samples/Import Data folder.

```
<?xml version="1.0" encoding="UTF-8"?>
<Workbook xmlns="urn:schemas-microsoft-com:office:spreadsheet"
 xmlns:o="urn:schemas-microsoft-com:office:office"
 xmlns:x="urn:schemas-microsoft-com:office:excel"
 xmlns:ss="urn:schemas-microsoft-com:office:spreadsheet"
 xmlns:html="http://www.w3.org/TR/REC-html40">
 <Worksheet ss:Name="Bigclass">
  <Table ss:ExpandedColumnCount="5" ss:ExpandedRowCount="41"
   x:FullColumns="1"
   x:FullRows="1">
   <Row>
    <Cell><Data ss:Type="String">name</Data></Cell>
    <Cell><Data ss:Type="String">age</Data></Cell>
    <Cell><Data ss:Type="String">sex</Data></Cell>
    <Cell><Data ss:Type="String">height</Data></Cell>
    <Cell><Data ss:Type="String">weight</Data></Cell>
   </Row>
   <Row>
    <Cell><Data ss:Type="String">KATIE</Data></Cell>
    <Cell><Data ss:Type="Number">12</Data></Cell>
    <Cell><Data ss:Type="String">F</Data></Cell>
    <Cell><Data ss:Type="Number">59</Data></Cell>
    <Cell><Data ss:Type="Number">95</Data></Cell>
   </Row>
  </Table>
 </Worksheet>
</Workbook>
```

The following script reads BigclassExcel.xml and creates a JMP data table with the information in it. This script, named ParseXML.jsl, is in the JMP Samples/Scripts folder.

```
file contents = Load Text File( "$SAMPLE_IMPORT_DATA/BigclassExcel.xml" );
Parse XML( file contents,
   OnElement(
      "urn:schemas-microsoft-com:office:spreadsheet^Worksheet",
      StartTag(
         sheetname = XML Attr(
            "urn:schemas-microsoft-com:office:spreadsheet^Name",
            "Untitled"
         );
         dt = New Table( sheetname );
         row = 1;
```

```
            col = 1;
         )
      ),
      OnElement(
         "urn:schemas-microsoft-com:office:spreadsheet^Row",
         StartTag(
            If( row > 1, // first row has column names
               dt << Add Rows( 1 )
            )
         ),
         EndTag(
            row++;
            col = 1;
         )
      ),
      OnElement( "urn:schemas-microsoft-com:office:spreadsheet^Cell", EndTag(
      col++ ) ),
      OnElement(
         "urn:schemas-microsoft-com:office:spreadsheet^Data",
         EndTag(
            data = XML Text( collapse );
            If( row == 1,
               New Column( data, Character( 10 ) ), // first row has column names
               Column( col )[row - 1] = data // and other rows have data
            );
         )
      )

   );
```

OLE Automation

Most of JMP can be driven through OLE automation. Please see the Automation Reference.pdf at https://www.jmp.com/en_us/support/jmp-documentation.html for more information about automating JMP. This document introduces how to automate JMP through Visual Basic and using Visual C++ with MFC. It also contains more information about the methods and properties that JMP exposes to automation clients like Visual Basic and Visual C++.

The JMP Samples/Automation folder contains several sample Visual Basic .Net, and Visual C# .Net files.

Share Reports on JMP Live

Write JSL scripts to automate the generation of JMP analyses and publish them to JMP Live.

- "JMP Live Prerequisites"
- "Get an API Key"
- "Publish a Single Report to JMP Live"
- "Publish Several Reports to a JMP Live Folder"
- "Delete Reports in a JMP Live Folder"
- "Replace the Contents of a JMP Live Report"
- "Search for a JMP Live Report or Folder ID"
- "Update Data in JMP Live Reports"
- "Work with Other Content in JMP Live Reports"
- "HTTP Requests in JMP Live"

JMP Live Prerequisites

Before you can use JSL to publish reports to JMP Live, do the following:

1. Get an API key (if required by a JMP Live administrator). See "Get an API Key".
2. Establish a connection to a JMP Live server. See *Using JMP*. When you add a JMP Live server, you need to enter your API key (if required).

   When you establish a connection, a JMP Live Connection is created in JSL:

   ```
   liveconnection = New JMP Live( Connection( "My JMP Live" ), Prompt( If Needed
     ));
   ```

   where `"My JMP Live"` is the name of the JMP Live server connection you created. If authentication is required, an interactive prompt appears. You can change the `Prompt` to `No`, but if authentication is required, the connection fails.

Get an API Key

Before you can script reports and publish them to JMP Live, you might need an API key. JMP Live administrators determine whether an API key is required for scripting.

An API key is unique for a user and a JMP Live server. That means that if you obtain a new API key, it overrides the previous one, and any scripts using the previous API key will need to be updated to use the new API key.

To obtain an API key:

1. Log into JMP Live.
2. At the top, click *your user icon* and select **Settings**.
3. Click **Generate API Key**. Make a note of your key.

Publish a Single Report to JMP Live

Publish a single report to JMP Live using a JMP Live Report.

The following example creates a Bivariate report and an empty JMP Live Report called `webreport`, adds the Bivariate report to the `webreport`, and publishes the `webreport` to JMP Live:

```
dt = Open( "$SAMPLE_DATA\Big Class.jmp" );
// Create a Bivariate report.
biv = Bivariate( X( :Height ), Y( :Weight ));
// Create an empty JMP Live Report.
webreport = New Web Report();
/* Add the Bivariate report to the webreport. A title is required and a
   description is optional.*/
webreport << Add Report( biv, Title( "Bivariate Report" ), Description( "A
   test of the new scripting" ));
/* Publish the webreport to JMP Live as a private report. Use the JMP Live
   Connection you established and return a JMP Live object called liveresult.
   */
liveresult = liveconnection << Publish( webreport );
```

You can use JSL to make the report visible to everyone or share the report with groups in JMP Live. See the `Set Public` and `Share With Groups` messages in the JMP Scripting Index at Help > Scripting Index.

Work with a JMP Live Result

The `Publish` function returns a JMP Live Result (`liveresult`). Use JSL scripting functions on a JMP Live Result to get information, such as whether the publishing succeeded or failed, failure codes, and more:

```
// Return the outcome of the last action (1 for success, 0 for failure).
liveresult << Succeeded ();
// Return any error messages from the last action.
liveresult << Get Error Message();
```

The JMP Live Result can be made scriptable and turned into a JMP Live Report (for a single report) or a JMP Live Folder object (for multiple reports) so that you can send commands to it. The following example changes a JMP Live Result into a scriptable JMP Live Report and then changes the report access from private to public:

```
livereport = liveresult << As Scriptable();
livereport << Set Public( 1 );
```

See the JMP Live Report object in the JMP Scripting Index at Help > Scripting Index.

Publish Several Reports to a JMP Live Folder

Publish several reports to JMP Live by creating a JMP Live Folder and adding the JMP Live Report to the folder. A JMP Live Report object can hold a single report or multiple reports.

The following example creates a scriptable JMP Live Folder and a JMP Live Report called `webreport`, adds a Bivariate and a Oneway report to the `webreport`, and adds the reports to a folder:

```
/* Create a folder in JMP Live. A title is required and a description is
   optional. Use the JMP Live Connection you established and return a JMP Live
   object called liveresult. */
liveresult = liveconnection << Create Folder( Title( "Test Folder"
  ),Description( "Folder created for script testing" ));
// Make the liveresult scriptable.
folder = liveresult << As Scriptable();
// Create an empty JMP Live Report.
webreport = New Web Report();
// Add a Bivariate report to the webreport.
webreport << Add Report( biv, Title( "Bivariate Report" ));
// Add a Oneway report to the webreport.
webreport << Add Report( oneway, Title( "Oneway Report" ));
// Add the reports to the folder.
liveresult = folder << Add Reports To Folder( webreport );
```

You can use JSL to share the folder with JMP Live groups. See the `Share With Groups` object in the JMP Scripting Index at **Help** > **Scripting Index**.

Delete Reports in a JMP Live Folder

You can delete reports and folders using JSL. Folders must be empty before they can be deleted.

```
returncode = liveconnection << Delete Report( report << Get ID() );
returncode = liveconnection << Delete Folder( folder << Get ID() );
```

In both of these cases, a value of 1 (True) or 0 (False) will indicate if the operation was successful.

Replace the Contents of a JMP Live Report

You can replace the contents of an existing report rather than creating a new report with updated content. You need to know the ID of the report you want to replace. If you have the report object available, you can use the `Get ID( )` function to return the ID.

The following example creates a new JMP Live Report called `webreport2`, adds an updated Oneway report, and replaces an existing Oneway report using the report ID:

```
// Create an empty JMP Live Report.
webreport2 = New Web Report();
/* Add a Oneway report to webreport3. A title is required and a description is
   optional. */
webreport2 << Add Report( oneway, Title( "New Oneway Report" ), Description(
   "Replace a report" ));
/* Use the JMP Live Connection you established to replace the report with the
   specified ID with webreport2. */
liveconnection << Replace( webreport2, ID(report << Get ID() ));
```

Search for a JMP Live Report or Folder ID

For many JSL functions, you need to know the report or folder ID. Using JSL, you can search by keyword (in the title or description) or by the publisher of the report or folder. If your search may return many results, you can specify how many results to return at a time.

The following example searches all reports that contain the word "Production" in the title or description, and are published by a user with the display name "dianesmith". It also specifies that the search will return up to ten items at a time.

```
/* Search for reports with the keyword Production that are published by the
   user dianesmith. Return no more than 10 results at a time. */
liveresult = liveconnection << Find Reports( Search( "Production" ),
   Publisher( "dianesmith" ), ( PageSize( 10 ));
// Change the liveresult into JMP Live Reports and make them scriptable.
livereports = liveresult << As Scriptable();
// Show how many results are returned.
show( livereports << Get Number Of Items());
// Return the first report in list of results.
report1 = livereports[ 1 ];
// Show the title of the first report.
show( report1 << Get Title());
```

In the example above, if you might have more than ten results, you can page through them using the Next and Previous functions:

```
// Put the next 10 results into the next page.
nextpage = livereports << Next();
```

```
// Show the current page number.
show( livereports << Get Current Page Number());
// Get the first item in the list of items in the next page.
next1 = nextpage[ 1 ];
// Show the title of the first report.
show( next1 << Get Title());
// Show the number of items in the current page.
show( nextpage << Get Number Of Items());
```

If `Next()` finds no further pages, an error appears, which you should catch to proceed with further processing.

Tip: If you know the number of pages to expect, you can jump directly to a page. See the `Get Page` function in the JMP Scripting Index at **Help** > **Scripting Index**.

Update Data in JMP Live Reports

If the data behind a JMP Live report changes, you can update the data and regenerate the report. You can update the data for a standalone report (using the report ID) or for all the reports in a folder (using the folder ID).

The following example updates data for an individual report. Suppose that in the Big Class.jmp data table, a subject's age changed and you want to update an associated age report on JMP Live.

```
dt:Age[1] = 77;
/* Use the JMP Live Connection you established to update the data for the
   report with the specified ID based on the updated Big Class.jmp data table.
   */
updateresult = liveconnection << Update Data( report << Get ID( ), Data( dt,
   "Big Class.jmp" ));
// Show the outcome of the update.
show( updateresult << Succeeded );
```

Similarly, you can also specify the folder that the report resides in:

```
updateresult = liveconnection << Update Data( folder << Get ID( ), Data( dt,
   "Big Class.jmp" ));
```

Only reports in the folder using the specified data table update.

Work with Other Content in JMP Live Reports

Primarily in JSL, you work with JMP Live folders and reports. However, you can publish images to JMP Live reports. In JSL, you can use a generic object called a JMP Live Post to manage images or other content that does not fall under reports or folders. See the `JMP Live Post` function in the JMP Scripting Index at **Help** > **Scripting Index**.

HTTP Requests in JMP Live

To invoke HTTP methods in JMP Live, use `Get HTTP Request()`:

```
lc = New JMP Live(Connection("JMP Live Connection"), Prompt(If Needed));
hrequest = lc << Get HTTP Request();
hrequest << URL(url);
hrequest << Method("GET");
data = hrequest << Send();
json_data = Parse JSON(data);
show(json_data);
```

For more details, see "Communicate with REST Web Service".

Communicate with REST Web Service

Representational state transfer (REST) or RESTful web services is a way of providing interoperability between computer systems on the Internet. Use JSL to communicate with a RESTful application program interface (API) that uses HTTP requests to GET, POST, PUT, and DELETE data. For example, you can get data from a web service and return it as a data table. Some web services are Amazon Web Services (AWS), DropBox, and Text-Processing.com.

Notes:

- Strings that contain "&" are truncated. "+" is converted to a space.
- `New HTTP Request()` supports redirects, for example, from HTTP to HTTPS.

Examples of HTTP Requests

The following examples communicate with the National Aeronautics and Space Administration (NASA) API. You must request a key from https://api.nasa.gov/index.html#apply-for-an-api-key and assign it to the `key` variable to run the scripts successfully.

Write Data to the Log

```
url = "http://technology.nasa.gov/api/query/patent/all";
request = New HTTP Request( URL( url ), Method( "Get" ) );
data = request << Send();
Write( data );
   {"results":[["59fa04859600022b4d2e076f","LAR-TOPS-48","All-Organic
      Electroactive Device","NASA's Langley Research Center offers you an
      all-organic electroactive device system fabricated with single-wall
      carbon nanotube (SWCNT)....}
   // JSON output written to the log
```

Notes:

- The query string is at the end of the URL, following "?". For example, in https://www.google.com/search?q=jmp, "q=jmp" is the query string. `New HTTP Request()` automatically URI-encodes key/value pairs in a query string (and FORM).
- When you put an associative array in either a query string or a form, the key/value pairs get URI encoded. Spaces, ampersands, and other characters get encoded to ASCII, and the ampersand is escaped. The encoding is a "%" encoding.
- The web service defines the valid key/value pairs that it accepts. Refer to the API documentation for valid key/value pairs.
- For Kerberos authentication, specify the password as follows:

  `Password(":")`

Return Data as an Associative Array and Write it to the Log

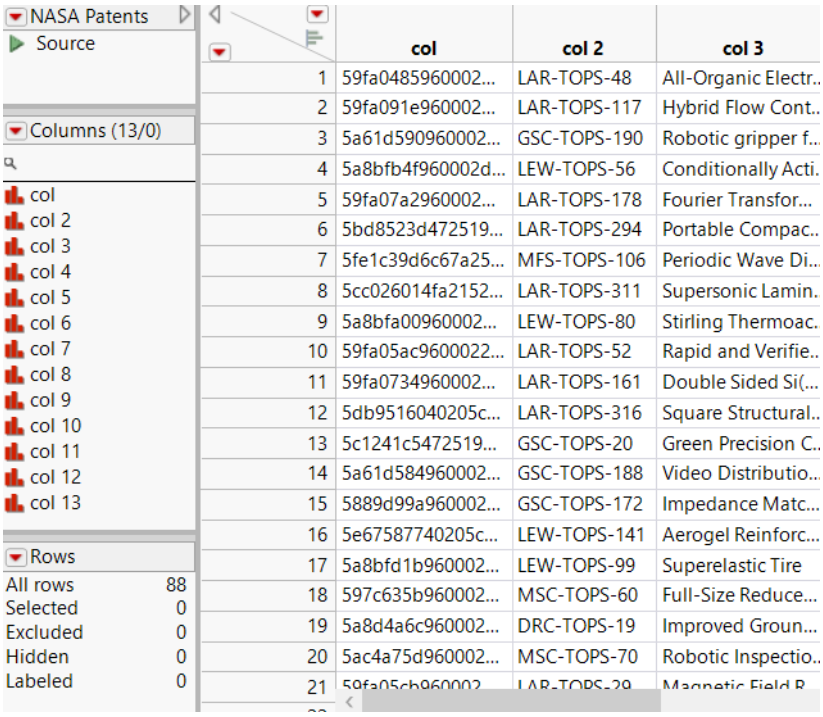
```
url = "http://technology.nasa.gov/api/query/patent/all";
request = New HTTP Request( URL( url ), Method( "Get" ) );
json = request << Send();
jsl_json = Parse JSON( json );
// turns JSON data into a JSL associative array
results = jsl_json["results"];
Write( results );
   {{"59fa04859600022b4d2e076f", "LAR-TOPS-48", "All-Organic Electroactive
      Device", "NASA's Langley Research Center offers you an all-organic
      electroactive device system fabricated with single-wall carbon nanotube
      (SWCNT)....}}
   // associative array
```

Return Data as a JMP Data Table

```
url = "http://technology.nasa.gov/api/query/patent/all";
request = New HTTP Request( URL( url ), Method( "Get" ) );
json = request << Send();
jsl_json = Parse JSON( json );
json = As JSON Expr( jsl_json["results"] );
dt = JSON To Data Table(
```

```
    json,
    JSON Settings(
       Stack( 0 ),
       Col(
          "/root",
          Column Name( "root" ),
          Fill( "Use Once" ),
          Type( "Pandas Values" ),
          Format( {"Best"} ),
          Modeling Type( "Continuous" )
       )
    )
  );
  dt << Set Name( "NASA Patents" );
```

Figure 14.5 Data Imported in a Data Table



OAuth 2.0 for Web APIs

OAuth 2.0 is an authorization standard for accessing personal user information through web APIs. An authorization standard is an agreed upon set of rules by many major web companies that are responsible for authorizing who can access information, and what scope of information they can access. This authorization is granted in the form of a limited time access *token* (a unique identifier specific to the individual and the scope of information they can access). Due to this two-pronged approach of limiting both users and scopes, OAuth is the current standard for securing personal information. Web sites such as Google, Salesforce, Reddit, and Dropbox use OAuth 2.0.

Since OAuth secures your personal information, it can appear to be fairly complex. OAuth is even more complicated due to the variety of ways it is implemented, which are called *grant types*. We'll walk through the different ways you can get an OAuth token in JSL. Please note that not all methods will work with all companies, so you'll have to do a little research to find out which grant type to use. Usually, the company will have documentation of this either with their APIs or possibly in a separate OAuth section.

We recommend that you familiarize yourself with the OAuth protocol at the official website (https://oauth.net/2/).

Note: Only OAuth 2.0 is supported in JSL. OAuth 1.0 and 1.0a are still used by a handful of companies (such as Twitter) but are largely regarded as inferior to 2.0.

Security

OAuth tokens can contain sensitive information. Depending on the grant type, you might need to provide your password or a client secret. JMP automatically handles this information for you.

Information that Google uses to authenticate your account is stored in C:/Users/<user name>/AppData/Roaming/SAS/JMP/OAuthTokens.jmp. (Your Google password is not stored.) After using an OAuth token on your personal account, you can delete all columns except for User and Client ID. Keep in mind that if you need to share your script, others won't have the User and Client ID columns saved on their machine. We recommend moving this information to an encrypted JSL script before sharing the OAuth script.

Using Postman for Redirect URLs

JMP does not provide a redirect URL for free. However, using Postman, it is easy to set one up yourself. Some examples that follow use Postman to redirect requests.

1. Go to https://www.getpostman.com.

2. Click the orange sign-in button in the top right corner.
3. Sign in with your Google account, or proceed to create your own account.
4. You can then use https://app.getpostman.com/oauth2/callback as a redirect URL.

Authorization Code Grant

The Authorization Code grant usually sends *refresh tokens*. Refresh tokens are unique in that they act as a sort of pseudo, permanent access token. Although you won't access sensitive information with them, a refresh token is all you need to get more access tokens. Treat it like you would your password.

The following code snippet creates an OAuth token using the Authorization Code grant. Notice this requires `Request Auth()`, `Client Secret()`, and `Redirect URL()`. Remember, after you make the token once, you can remove everything but the User and Client ID from OAuthTokens.jmp.

```
token = New OAuth2 Token(
  User( "Test User" ),
  Client ID( "12ab" ),
  Client Secret( "3456dEfG" ),
  Request Auth(
     Scope( "history" ),
     Auth URL( "https://example.com/services/oauth2/authorize" ),
  ),
  Redirect URL( "https://www.getpostman.com/oauth2/callback" ) ,
  Token URL( "https://example.com/services/oauth2/token" )
);
```

Note: See your API documentation for more information about how to get values such as the client secret and token URL.

After you run this code once, clean it up and make it secure.

```
token = New OAuth2 Token(
  User( "Test User" ),
  Client ID( “12ab” ),
);
```

Existing Refresh Tokens

You might already be familiar with OAuth and have saved a refresh token to your script. The following snippet creates an OAuth token with that existing token.

```
token = New OAuth 2 Token (
  User( "yourgoogleaccount@gmail.com" ),
```

```
  Refresh Token( "1a2b3c4e5F" ),
  Token URL( "https://www.example.com/oauth2/token" ),
  Client ID( "12ab" ),
  Client Secret( "3456dEfG" )
);
```

After you run this code once, clean it up and make it secure.

```
token = New OAuth2 Token(
  User( "yourgoogleaccount@gmail.com" ),
  Client ID( "12ab" )
);
```

Implicit Grant

The Implicit grant type doesn't require any sensitive information. If you're going to share a script with other people, it would be easiest to use this grant type. Note, however, that because sensitive information isn't required, services are less likely to implement this flow.

The following code snippet creates an OAuth token using the Implicit grant. This requires `Redirect URL()`.

```
token = New OAuth2 Token(
  User( "Implicit Flow"),
  Client ID( "123abc" ),
  Redirect URL( "https://www.getpostman.com/oauth2/callback") ,
  Token URL( "https://example.com/services/oauth2/token" )
);
```

Resource Owner Grant

The Resource Owner grant is the only type that doesn't require a redirect URL. If setting up Postman (the API development platform) is proving difficult, try this grant type.

The following snippet will create an OAuth token using the Resource Owner grant. This requires `Client Secret()` and `Password()`.

```
token = New OAuth2 Token(
  User( "yourgoogleaccount@gmail.com" ),
  Password( "wordP@ss0192" ),
  Client ID( "1234abcd" ),
  Client Secret( "5678EfGh" ),
  Token URL( "https://example.com/services/v1/token" )
);
```

After you run this code once, clean it up and make it secure.

```
token = New OAuth2 Token(
```

```
    User( "yourgoogleaccount@gmail.com" ),
    Client ID( "1234abcd" ),
);
```

Using a Token

In order to access a web API that requires OAuth, pass the token to `HTTP Request()` using the OAuth2 field.

```
request = HTTPRequest(
    URL( apiEndpoint ),
    Method( "POST" ),
    OAuth2( token ),
);
```

OAuth 2.0 Resources

The official website for OAuth 2.0 is https://oauth.net/2/.

https://auth0.com/docs/protocols/oauth2 provides more information about the OAuth 2 protocol.

The OAuth 2.0 standard, RFC6749, is at https://tools.ietf.org/html/rfc6749.

Google provides comprehensive documentation and examples of OAuth 2.0 at https://developers.google.com/identity/protocols/OAuth2.

Communicate with SAS Cloud Analytic Services

SAS Viya uses Cloud Analytic Services (CAS) code to perform calculations in the cloud. This feature provides a way for SAS Viya users and JMP users to collaborate. You can write CAS code in JMP to import data from and export data to the CAS server, convert CAS JSON text to a JMP data table, get a list of available CAS libraries, and more.

- See the SAS CAS documentation at https://developer.sas.com/apis/cas/rest/current/apidoc.html for more information.
- The SAS CAS documentation for CAS actions cover CASL, R, Python, Lua.

This section covers the following topics:

- "About CAS Actions"
- "Connect to the CAS Server"
- "Create and Submit a CAS Action"
- "Create and Submit a CAS DATA Step Action"
- "Create a New CAS Server"
- "Import Data from the CAS Server"
- "View a CAS Dataset"
- "Get a List of Available CAS Libraries"
- "Export Data from a Data Table to a CAS Server"
- "Convert a CAS Table to a Data Table"
- "Delete a CAS Table"
- "Remove a CAS Table"
- "Verify the CAS Server Connection"
- "Get Details about the Current Connection"
- "Disconnect from the CAS Server"
- "Terminate All CAS Sessions"

About CAS Actions

The smallest unit of work for the CAS server is a CAS action. CAS actions can load data, transform data, compute statistics, perform analytics, and create output.

You configure each action by specifying a set of input parameters. Running a CAS action on the CAS server processes the action's parameters and the data, and creates an action result. See *CAS Actions* at https://developer.sas.com/apis/cas/actions.html.

The JSON parameters are specified in the CAS documentation *CAS REST API*. See https://developer.sas.com/apis/cas/rest/current/apidoc.html. For JSON, you can look at the Lua format and make the appropriate adjustments.

For example, in Figure 14.6, notice the difference between the JSON syntax and the Lua syntax. JSON uses the `"name" : value` format. Lua uses the `name = value` syntax.

Figure 14.6 Differences Between JSON and LUA

JSON Syntax

```
{
    "attrTable": "string",
    "attrs": [ {
        column": "string",
        key": "string",
        value": "string" |
                number
}, {...} ],
...
}
```

LUA Syntax

```
{
    attrTable="string",
    attrs={ {
        column="string",
        key="string",
        value="string" |
              64-bit integer |
              integer |
              double |
             binary-large-object
      }, {...} },
,,,
}
```

Connect to the CAS Server

You can show the login window and prompt the user to provide the URL for the CAS server and their login information.

```
CAS Connect( Prompt( "Never"|"Always"|"IfNeeded" ) );
```

To add the login details to the script, follow this example:

```
CAS Connect( URL( URL ), Username( username ), Password( password ) );
```

Information about the CAS server appears in the log after the user logs in.

To specify a trusted certificate for https connections to the CAS server, use the optional Certificates argument.

```
CAS Connect( URL( URL ), Certificates( string ) );
```

To specify whether a certificate is verified, add either `Verify Certificates()` or `No Verify Certificates()` to the script:

```
Verify Certificates( 0|1 )
No Verify Certificates( 0|1 )
```

To specify a proxy server, add `Proxy Server()` to the script:

```
Proxy Server( "https://mycompany.com:80" )
Proxy User( username )
```

To bypass a proxy server, add `Bypass Proxy()` to the script:

```
Bypass Proxy( "https://localhost:80" )
```

Create and Submit a CAS Action

```
New CAS Action( Action( "builtins.echo" ), JSON( echo ) );
```

`Action` is the JSL variable name. builtins.echo is the name of the CAS action that SAS created. `echo` contains the JSL arguments.

Create and Submit a CAS DATA Step Action

```
New CAS DATA Step action( Code( code ) );
```

code defines the contents of the DATA step. Action is the JSL variable name.

```
cas = Current CAS Connection();
code =
"\[
   data temp;
   x = 9.1; y = 6; z = sqrt(x**2 + y**2);
   A = "SAS"; B = "Statistics";
   put _ALL_;              /* display all variables and values */
   run;
]\";
action = New CAS DATA Step action( Code( code ) );
cas << Submit( action );
```

Create a New CAS Server

`New CAS Server()` creates a new CAS server.

```
url = "URL";
cas = New CAS Server( Connect( URL( url ), Prompt(
   "Never"|"Always"|"IfNeeded" ) ) );
```

Import Data from the CAS Server

`CAS Import Data()` imports data from the CAS server:

```
CAS Import Data( libref.dataset, <Private|Invisible>, <"Use Labels for Var
   Names">);
```

- `invisible` hides the data table from view but shows it in the JMP Home Window. `private` hides the data table completely. Specify `private` if the user doesn't need to interact with the data table.
- `"Use Labels for Var Names"` indicates that the table name and the column names will be set to the SAS dataset label and the column labels. The appropriate column property, SAS Name SAS Label, is used.

View a CAS Dataset

CAS datasets are on the CAS file system. Datasets in memory are not listed.

To see the datasets on the CAS server that you want to import, use `CAS Get Data Sets()` after you connect to the server.

The following example shows how to list datasets that are only in casuser:

```
datasets = CAS Get Data Sets( casuser ); // quote the argument
Show( datasets );
```

The following example shows how to lists datasets that you have permissions to view:

```
datasets = CAS Get Data Sets();
Show( datasets );
```

Get a List of Available CAS Libraries

Libraries are namespaces, just like SAS libraries, librefs, or libnames. Use `CAS Get Libraries()` to get libraries from the CAS server.

```
libraries = CAS Get Libraries();
Show( libraries );
```

Export Data from a Data Table to a CAS Server

`CAS Export Data()` exports a data table to a CAS server to run an action on it.

```
CAS Export Data(
  Open( "$SAMPLE_DATA/Big Class.jmp" ),
  "casuser",
  "Big Class"
);
```

`CAS Export Data()` also has two arguments:

- `Save(0)` saves the data in memory.
- `Save(1)` saves the data to the file system.

Convert a CAS Table to a Data Table

`CAS Table to Data Table()` converts JSON data to a data table. A JSON element in the results of an action contains "`_ctb`". This is a marker indicating that it is formatted to represent a CAS table. You can send the JSON results to `CAS Table To Data Table()`. JMP will create a JMP data table from the JSON.

If there's more than one result, you might need to perform a little extra work as shown with `Get Results` at the end of the following example. This example shows how to export data, run an action, and save the data in several data tables.

```
cas = Current CAS Connection();
If( IsEmpty(cas),
   cas = CAS Connect( URL( url ), Username( username ), Prompt( "IfNeeded"
   ));
);
dt = Open( "$SAMPLE_DATA/Animals.jmp" );
cas << Export Data( dt, "CASUSER.Animals" );
json = "\[
{
  "table": {
   "name": "Animals",
   "caslib": "CASUSER"
  },
  "class": [
    "species",
    "subject",
    "season"
  ],
  "model": {
    "depVar": "miles",
    "effects": [
      {
        "vars": [
          "species",
          "season"
        ],
        "interaction": "BAR"
      }
    ],
    "printsol": true,
    "cl": false,
    "dfmethod": "RESIDUAL"
  },
  "random": [
    {
      "depVars": "miles",
      "effects": [
        {
          "vars": [
            "subject"
          ],
          "nest": [
            "species"
```

```
            ]
          }
        ]
      }
    ],
    "method": "REML"
}
]\";
action = New CAS Action(
   Action( "mixed.mixed" ),
   JSON( json )
);
rc = cas << Submit( action );
if(rc,
   results = action << Get Results;
   keys = results << Get Keys();
   For(i = 1, i <= N Items( keys ), i++,
      dt = CAS Table To Data Table( results[keys[i]] );
   );
);
cas << Disconnect();
```

`Get Results` enables you to examine the results of a CAS action; it's a JSL associative array. `Get JSON` gives you the raw JSON; it's a JSL string.

The JSON data is in the format discussed at https://developer.sas.com/apis/cas/rest/current/apidoc.html. Search for "ResultsTable".

In `CAS Table to Data Table()`, specify the JSON string, open the data table as invisible or private, and control the table name and column names.

```
json = "jsonstring";
CAS Table to Data Table( json, <Invisible(Boolean)|Private(Boolean)>, <"Use
   Labels For Var Names">);
```

- `Invisible` hides the data table from view but shows it in the JMP Home Window. `private` hides the data table completely. Specify `Private` if the user doesn't need to interact with the data table.
- `"Use Labels for Var Names"` indicates that the table name and the column names will be set to the SAS dataset label and the column labels. The appropriate column property, SAS Name SAS Label, is used.

Note: CAS ResultsTable attribute values (peers to "`_ctb`") found in the "`_ctb`" JSON are also set as table variables. The "CREATETIME" attribute creates a character table variable with a locale-formatted timestamp (for example, 04/03/2019 4:38:05 PM).

Delete a CAS Table

`CAS Delete Table()` deletes the underlying file from the CAS file system.

```
CAS Delete Table( table name, <remove> ); // quote the table name
```

The `CAS Delete Table()` function has a remove option, which removes an in-memory table from the CAS server. `CAS Remove Table()` serves the same purpose.

Remove a CAS Table

`Remove CAS Table()` removes an in-memory table from the CAS server.

```
CAS Remove Table( casuser, Big Class ) // quote both arguments
```

Verify the CAS Server Connection

`CAS Is Connected()` determines whether the CAS server is connected.

```
connected = CAS Is Connected();
Show( connected );
  connected = 1;
```

Get Details about the Current Connection

`Current CAS Connection()` obtains details about the current connection: the URL of the CAS server, your user name, and the session ID.

```
connection = Current CAS Connection();
Show( connection );
```

Disconnect from the CAS Server

`CAS Disconnect()` disconnects from the current CAS server and optionally terminates the session. By default, the session is terminated when disconnected.

```
CAS Disconnect(<"Terminate"|"NoTerminate">);
```

Terminate All CAS Sessions

`CAS Terminate Sessions()` ends all sessions owned by the current user except for the current session.

```
CAS Teminate Sessions();
```

You can also specify the session ID to terminate the specified session.

```
CAS Terminate Sessions(session ID);
```

Use `Get Session()` to get the session ID.

```
url = "http://myCASserver"; // quoted string
cas = CAS Connect(
   URL( url ),
   Username( "my_username" ), // quoted string
   Prompt( "IfNeeded" )
);
session_id = cas << Get Session;
```

To terminate the current session, use `CAS Disconnect()`.

Chapter 15

Creating Projects

Organizing Project Files in JMP

A JMP project provides a way to organize files that you use in an analysis. You can add JMP files (reports, data tables, scripts, journals, and so on) and non JMP files, such as Microsoft Word or Adobe PDF files. After you perform an analysis, the data table, report, and graph appear in the project on tabs. You can display multiple reports or graphs, display the project or script editor log, and run scripts from the project. Reports and graphs remain linked to the data table.

Projects help you avoid the clutter of several open JMP windows. You can maximize the project window to have a bigger view of the project. And when you save the project, the state of the project is saved (for example, the open reports and the layout of the window).

Contents

Create a Simple Project 867
Create a Project with a Bookmarks Pane and Log 868
Running Scripts in a Project 870
Create a Bookmark Group 872
Close Tabs in Projects 872
Get a Project 873
Reset the Window Layout 874
Run Startup Scripts to Control Workspaces 875

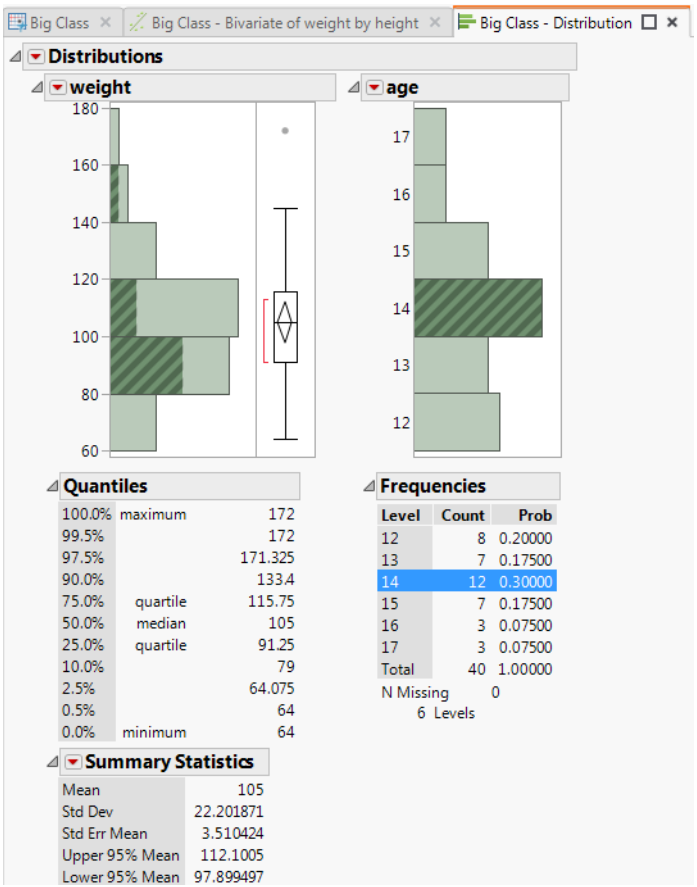
Create a Simple Project

A JMP project provides a way to organize files that you use in an analysis and display the output. A simple project might display the data table, graphs, and reports on separate tabs in the project window.

```
project = New Project();
project << Run Script(
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   dt << Run Script( "Bivariate" );
   dt << Run Script( "Distribution" );
);
```

Figure 15.1 shows the project with selected bars, showing that the graph is still linked to the data table.

Figure 15.1 Example of a Simple Project



Note: The best way to learn how to write a script for a project is to change the project extension to .zip, open the zip file, and then look for a file called project.jsl in the root directory.

Create a Project with a Bookmarks Pane and Log

A more complicated project might include a Bookmarks pane (to show file, folders, and groups that use frequently), and a project log. The data table and output are in separate tabs.

Begin by creating a new project and creating a bookmark.

```
project = New Project();
project << Add Bookmarks({
   File( "$SAMPLE_DATA/Big Class.jmp" )
});
```

`Run Script` opens the data table and creates output inside the project.

```
project << Run Script(
   Open(
      "$SAMPLE_DATA/Big Class.jmp",
      Set Window ID( "data" ) // unique window ID
   );
   Distribution(
      Set Window ID( "distrib" ),
      Continuous Distribution( Column( :weight ) ),
      Nominal Distribution( Column( :age ) )
   );
   Graph Builder(
      Set Window ID( "graphbuilder" ),
      Size( 443, 409 ),
      Show Control Panel( 0 ),
      Fit to Window( "Maintain Aspect Ratio" ),
      Variables( X( :age ), Y( :height ), Y( :weight ) ),
      Elements( Position( 1, 1 ), Line( X, Y, Legend( 1 ) ) ),
      Elements( Position( 1, 2 ), Bar( X, Y, Legend( 2 ) ) )
   );
);
```

`Set Layout` specifies the layout of the tabs that show the data table and output.

```
project << Set Layout(
   V Splitter Box(
      Size( 1215, 700 ),
      <<Set Sizes( {0.77, 0.23} ),
      <<Dockable(),
      V Splitter Box(
```

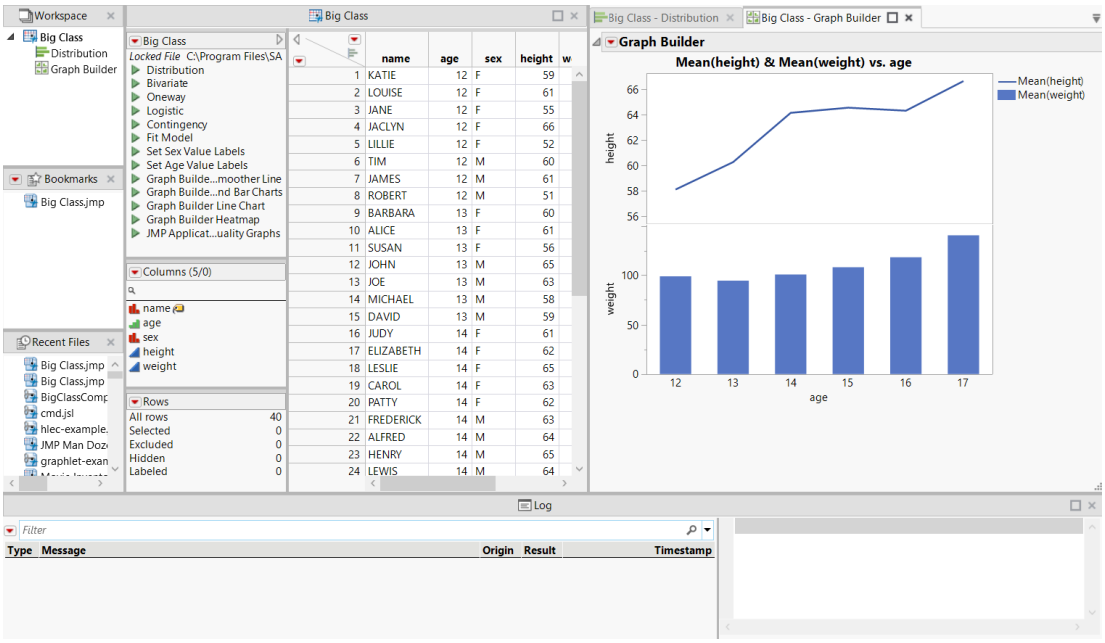
```
            Size( 1215, 540 ),
            <<Set Sizes( {1} ),
            <<Dockable(),
            H Splitter Box(
               Size( 1215, 540 ),
               <<Set Sizes( {0.11, 0.42, 0.47} ),
               <<Dockable(),
               V Splitter Box(
                  Size( 128, 540 ),
                  <<Set Sizes( {0.33, 0.33, 0.33} ),
                  <<Dockable(),
                  Tab Page Box(
                     Title( "Window List" ), // show the Window List
                     Window ID( "Window List" )
                  ),
                  Tab Page Box(
                     Title( "Bookmarks" ), // show the Bookmarks pane
                     Window ID( "Bookmarks" )
                  ),
                  Tab Page Box(
                     Title( "Recent Files" ), // show the Recent Files pane
                     Window ID( "Recent Files" )
                  )
               ),
               Tab Page Box( // show the data table
                  Title( "Big Class" ),
                  Window ID( "data" )
               ),
               Tab Box(
                  <<Dockable(),
                  Tab Page Box( // show the Distribution graph and reports
                     Title( "Big Class - Distribution" ),
                     Window ID( "distrib" )
                  ),
                  Tab Page Box( // show the Graph Builder graph
                     Title( "Big Class - Graph Builder" ),
                     Window ID( "graphbuilder" )
                  )
               )
            )
         ),
         Tab Page Box( Title( "Log" ), Window ID( "Log" ) )
      )
   );
```

Save saves the project.

```
// if the project has already been saved, the Save message
// with no argument saves the file to the existing location
project << Save( "$DOCUMENTS/Big Class.jmpprj" );
```

Note: If you're saving a project with unsaved documents in JSL, the project doesn't close and you get a log message that the project has unsaved documents. All files in a project must be saved or closed before the project can be saved. However, if you interactively save a project with unsaved documents, you are prompted to save the documents.

Figure 15.2 Example of a Complex Project



Open the project from another script:

```
project = Open( "Big Class.jmpprj" );
```

In this example, the script that you are running to open the project is in the same folder as the project.

Running Scripts in a Project

When you run a script outside of a project, the reports generated by the script appear in different windows. When you run a script inside a project, the reports appear in different tabs inside the project.

Running a JSL script outside of a project

Open a Script Editor window by selecting File > New > Script. Paste and run the following script:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Run Script( "Bivariate" );
dt << Run Script( "Distribution" );
Wait( 2 );
Get Window( "Big Class - Distribution" ) << Close Window();
```

Note that this script opens three JMP windows: one for the data table and one for each report. Close the reports and then the data table.

Running a JSL script inside of a project

Create a new project by selecting File > New > Project. From the project window, create a new script editor window by selecting File > New > Script. Note that the script editor window opens as a tab in the project. Paste the same JSL script you previously ran into the script editor and run it:

```
dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
dt << Run Script( "Bivariate" );
dt << Run Script( "Distribution" );
Wait( 2 );
Get Window( "Big Class - Distribution" ) << Close Window();
```

Note that this time, the data table and the reports all opened as tabs in the project, and that `Get Window()` correctly found the distribution tab. In general, all JSL functions that open, find, manipulate, and close JMP windows, tables, and reports work from within a project the same as they would outside projects.

Running a JSL script using a project's Run Script message

You can use the `Run Script` message to run JSL in the project's context. Open a Script Editor window, then paste and run the following code:

```
project = New Project();
project << Run Script(
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   dt << Run Script( "Bivariate" );
   dt << Run Script( "Distribution" );
   Wait( 2 );
   Get Window( "Big Class - Distribution" ) << Close Window();
);
```

Note that the script inside the `Run Script` message runs the same as if it had been in a script editor window inside the project.

Scoping Variables

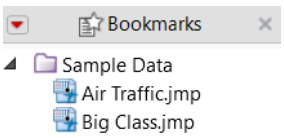
JMP can have several projects open at the same time, as well as items not open in any project. Each project has its own global scope, which prevents conflicts caused by identically named variables in different projects. In the rare case that you do want to share specific variables among all opened projects, use the `:::` root operator.

Create a Bookmark Group

You can create bookmark groups to organize the bookmarks in a project. A bookmark group can contain bookmarked files, folder, and additional bookmark groups. The bookmark group exists only in the project, not on your computer. Use `Group()` to create the bookmark group.

```
project = New Project();
project << Add Bookmarks(
        Group(
           "Sample Data",
           Expanded( 1 ), // open the group
           {File( "$SAMPLE_DATA/Air Traffic.jmp" ),
           File( "$SAMPLE_DATA/Big Class.jmp" )}
        )
);
project << Set Layout(
   H Splitter Box(
      <<Set Sizes( {0.15, 0.85} ),
      Tab Page Box( Window ID( "Bookmarks" ), )
   )
);
project << Save( "$DOCUMENTS/My Project.jmpprj" );
```

Figure 15.3 "Sample Data" Group in a Project



Close Tabs in Projects

Close a tab in a project by identifying the window with `Get Window()` and then closing it with `Close Window`.

```
project = New Project();
```

```
project << Run Script(
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   dt << Run Script( "Bivariate" );
   dt << Run Script( "Distribution" );
);

Wait( 2 ); // for demonstration purposes

// returns a reference to the open "Big Class - Distribution" window
dist = Get Window( Project( project ), "Big Class - Distribution" ) ;
dist << Close Window();
```

To close all tabs in a project, use `Get Window List << Close Window()`.

```
project = New Project();
project << Run Script(
   dt = Open( "$SAMPLE_DATA/Big Class.jmp" );
   dt << Run Script( "Bivariate" );
   dt << Run Script( "Distribution" );
);

Wait( 2 ); // for demonstration purposes

Get Window List( Project( project ) ) << Close Window();
```

Get a Project

When running a JSL script from within a project, you can use the `This Project()` function to obtain that project. The following example displays the name of the current project in a new window.

```
project = This Project();
If( Is Empty( project ),

   /* if the script is not run from a project, display this
   sentence in the new window */
   New Window( "My Project", Text Box( "Script isn't running in a project." )
   ),

   /* if the script is run from a project, display the window title
   in the new window */
   New Window( "My Project", Text Box( project << Get Window Title() ) )
);
```

`Get Project()` takes the same arguments as the `Window()` function. Omit the arguments to get a list of the currently open projects, include a string to find a specific project by title, or include a number to get the *i*th open project.

```
Open( "$SAMPLE_PROJECTS/Big Class.jmpprj" );
Open( "$SAMPLE_PROJECTS/Sports.jmpprj" );

// print the title of Sports.jmpprj
Print( Get Project( 2 ) << Get Window Title() );
```

`Get Project()` is also helpful when you want to save a project or set the layout from within the project.

```
Get Project() << Save(...);
Get Project() << Set Layout(...);
```

Reset the Window Layout

Suppose that you create a project and customize the layout of the windows. For example, you might hide the Window List. Use the `Reset Layout` message to return the windows to their original layout.

```
project = New Project();
project << Run Script( // open data table and create a graph
       Open(
              "$SAMPLE_DATA/Big Class.jmp",
             Set Window ID( "data" )
       );
       New Window( "Big Class - Bivariate of weight by height",
              Set Window ID( "bivariate" ),
              Bivariate(Y( :weight ), X( :height ),
              )
       ) << Set Window Icon( "Bivariate" );
);

Wait(3); // for demonstration purposes
project << Set Layout( // set a custom layout
       H Splitter Box(
              <<Set Sizes( {0.2, 0.8} ),
              Tab Page Box( Window ID( "Bookmarks" ) ),
                V Splitter Box(
                     <<Set Sizes( {0.53, 0.47} ),
                     Tab Page Box(
                         Window ID( "bivariate" ) ),
                     Tab Page Box(
                         Window ID( "data" ) )
```

```
                )
            )
);

Wait( 3 ); // for demonstration purposes
project << Reset Layout(); // return to the original layout
```

Run Startup Scripts to Control Workspaces

A workspace is either a project window or the windows that are not in a project (for example, the home window). A default workspace loads when JMP starts, and then each time you create or open a project, JMP creates another workspace.

Consider writing a workspace startup script to customize variables for a specific project. You might also construct a new customized window in a workspace startup script to put a copy of that window in each project. This is an alternative to having a single copy of the script that runs outside of projects.

The workspace startup script runs each time a new workspace loads. This means that the workspace startup script runs once when JMP starts (when the "default workspace" loads) and then runs again each time a new project is loaded. The workspace startups script runs in the context of the project, so any windows it opens or variables it sets are for that project only.

Name the script workspaceStart.jsl and place it in one of the following folders, as appropriate for your operating system. When JMP starts, JMP looks for the workspaceStart.jsl script in these folders in the order in which they are listed here. The first one that is found is run, and the search immediately stops.

Note: Some path names in this section refer to the "JMP" folder. On Windows, in JMP Pro, the "JMP" folder is named "JMPPro".

On Windows:

1. C:/Users/<username>/AppData/Roaming/SAS/JMP/16
2. C:/Users/<username>/AppData/Roaming/SAS/JMP

On macOS:

1. /Users/<username>/Library/Application Support/JMP/16
2. /Users/<username>/Library/Application Support/JMP

The workspaceStart.jsl script runs only for a particular user on a computer. You can add a script named workspaceStartAdmin.jsl in one of the following places, as appropriate for your operating system. This script is run for every user on a computer. JMP runs workspaceStartAdmin.jsl first if found. Then JMP runs workspaceStart.jsl if found.

On Windows:

1. C:/ProgramData/SAS/JMP/16
2. C:/ProgramData/SAS/JMP

On macOS:

1. /Library/Application Support/JMP/16
2. /Library/Application Support/JMP

Chapter 16

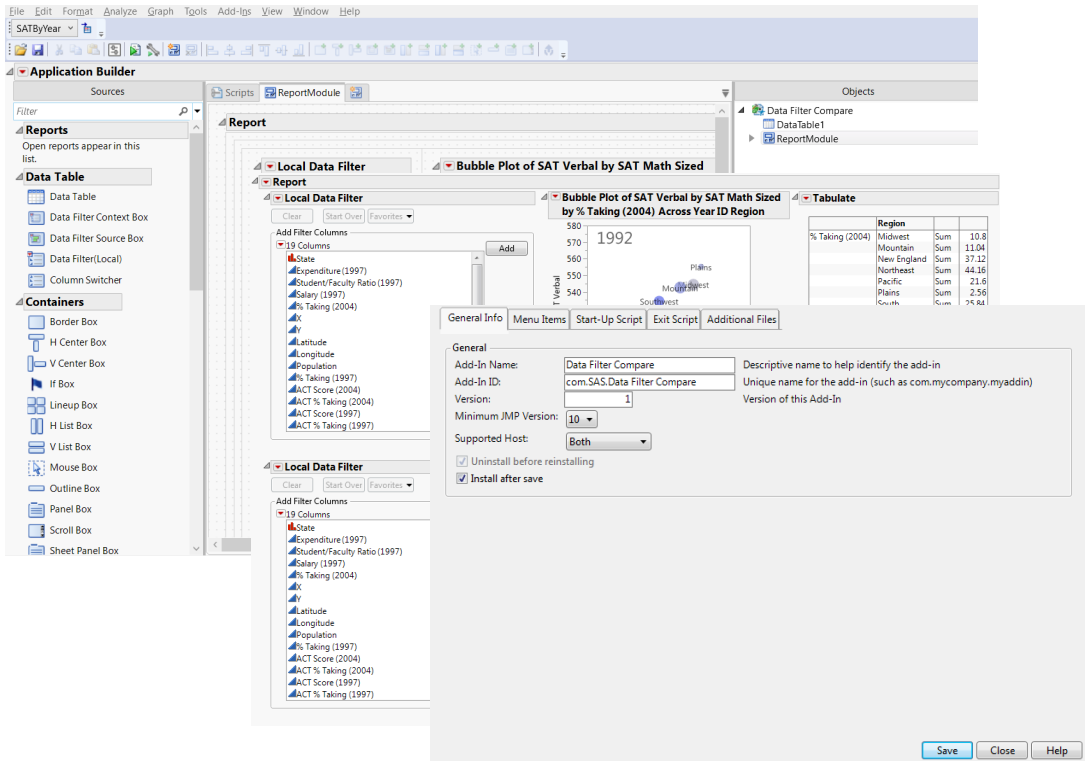
Creating Applications

Application Builder

In addition to providing platforms for quickly analyzing data, JMP lets you create applications to support your specific needs. You often perform the same tasks every day (such as running Distribution and Fit Model analyses on a data table and viewing the results). In JMP's Application Builder, you can create an application that shows the results for both analyses in one window. Application Builder's drag-and-drop interface reduces the amount of scripting required to create an application.

Add-In Builder visually guides you through creating an add-in that users can install in JMP. And after creating an application in Application Builder, use Add-In Builder to save the application as an add-in. This allows users to install the application and quickly run it inside JMP.

Figure 16.1 Creating Customized Reports in Application Builder



Contents

Create Applications with Application Builder 879
Example of Creating an Application 879
Application Builder Terminology 881
Design an Application 882
Application Builder Window 883
Application Builder Red Triangle Options 884
Create an Application 886
Edit or Run an Application 897
Options for Saving Applications 898
Additional Examples of Creating Applications 899
Compile Add-Ins with the JMP Add-In Builder 915
Compile Scripts into an Add-In Using Add-In Builder 915
Edit an Add-In 920
Share an Add-In 920
Installing Multiple Add-Ins 920
Register an Add-In Using JSL 921
Create an Add-In Manually 921
Manage JMP Add-Ins 923

Create Applications with Application Builder

Application Builder's drag-and-drop interface lets you visually design windows with buttons, lists, graphs, and other objects. This saves you the step of writing scripts to create these objects.

Application Builder also helps boost the productivity of experienced JMP Scripting Language (JSL) programmers. You can start designing a program in Application Builder and edit the automatically generated scripts. Integrate your own scripts to create even more powerful custom applications.

You can use Add-In Builder with Application Builder. First, use Application Builder to write complex scripts that create new platforms and interact with users and JMP. Then, use Add-In Builder to easily distribute your complex scripts so that any JMP user can access them without having to open and run the scripts. See "Compile Add-Ins with the JMP Add-In Builder"

Notes:

- Creating simple applications in Application Builder requires little to no JSL programming. However, to create most simple applications, Dashboard Builder might be sufficient. Dashboard Builder provides a subset of features that are found in Application Builder. See *Using JMP*.
- In each version of JMP, Application Builder takes advantages of new features that are not supported in earlier releases. For maximum compatibility, we recommend that you create applications in the earliest version of JMP that the application will be run in.

Example of Creating an Application

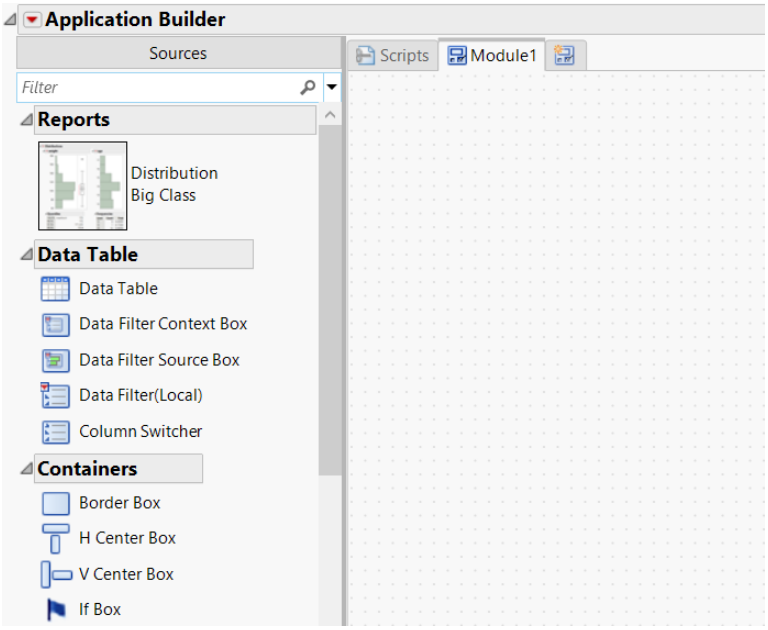
Application Builder helps you create instant applications without writing JSL. One example is a JMP window in which you arrange reports, graphs, and other objects to make them easy to interact with at once.

This example shows how to create a simple instant application for a Distribution report. This application lets you analyze preselected variables rather than having to select them in a launch window.

1. Select **Help > Sample Data Library** and open Big Class.jmp.
2. Run the **Distribution** table script to generate the report.
3. Select **File > New > Application** (or **File > New > New Application** on macOS).
4. Click the Blank Application box.

   A new blank application appears. Notice the Distribution report in the left column.

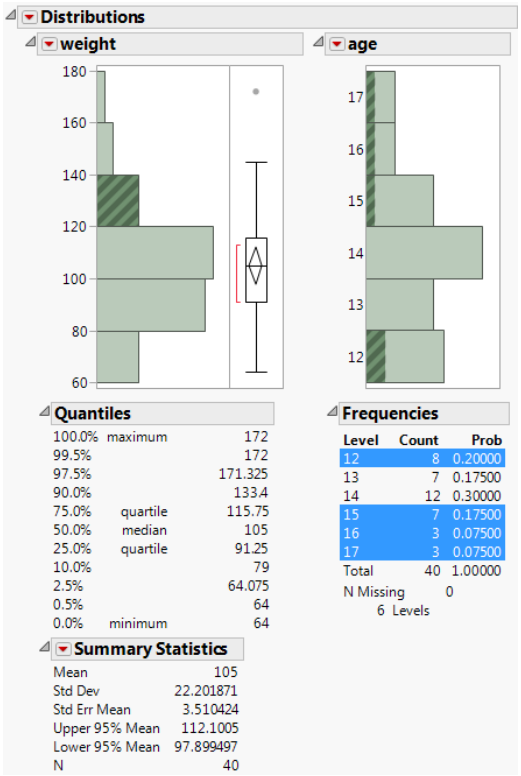
Figure 16.2 New Application with Distribution Report in the Sources Panel



5. In the Sources panel, drag the Distribution report onto the application workspace (the blank area on the Module1 tab).
6. Click the Application Builder red triangle and select **Run Application**.

   The Distribution report appears in a new window. Note that the histograms are interactive and still associated with the data table. When the data changes, you run the application again, and the Distribution report is updated.

Figure 16.3 Example of an Instant Application



	name	age	sex	height	weight
1	KATIE	12	F	59	95
2	LOUISE	12	F	61	123
3	JANE	12	F	55	74
4	JACLYN	12	F	66	145
5	LILLIE	12	F	52	64
6	TIM	12	M	60	84
7	JAMES	12	M	61	128
8	ROBERT	12	M	51	79
9	BARBARA	13	F	60	112
10	ALICE	13	F	61	107
11	SUSAN	13	F	56	67
12	JOHN	13	M	65	98
13	JOE	13	M	63	105
14	MICHAEL	13	M	58	95
15	DAVID	13	M	59	79
16	JUDY	14	F	61	81

Application Builder Terminology

An *application* consists of modules. A *module* contains the objects and scripts that are compiled into an application. When you run an application, JMP can show a separate window for each module. For example, one window might show a **Create Graphs** button. After the user clicks the button, the graphs appear in a new window.

There are terms you should become familiar with before using Application Builder. Knowing these terms helps you understand how a module works, even if you need to minimally edit an automatically generated script.

Application The top-level file that consists of one or more modules.

Module A collection of objects, messages, instances, and other JSL statements that are compiled into an application to create new JMP features.

Instant Application An application that typically consists of reports and does not require JSL scripting. You can also create an instant application, or “dashboard”, in Dashboard

Builder. Similar to Application Builder, Dashboard Builder has a drag-and-drop interface but with pre-configured display boxes.

Custom Application An application that demonstrates custom behavior through JSL scripting.

Objects and Messages An *object* is a dynamic entity in JMP, such as a data table, a data column, a platform results window, a graph, and so on. Most objects can receive messages that instruct the object to perform some action on itself.

A *message* is a JSL statement that is directed to an object. That object knows how to evaluate the message.

Module Instance The occurrence of a module in the application. In a complex application, you might create a start-up script that creates multiple instances of one or more modules.

Namespace A collection of unique variable names and corresponding values that prevents collisions among scripts.

JMP creates the `Application` namespace and the `ModuleInstance` namespace automatically. Symbols in the `Application` namespace are visible only to scripts in the application; the namespace is not available to scripts not created in the application. For more information about namespaces, see "Advanced Scoping and Namespaces".

Variable A unique name to which you assign a value. JMP creates the `thisApplication` and `thisModuleInstance` variables automatically.

- The `thisApplication` variable (used in the `Application` namespace) contains module names, which are available in any script in the application.
- The `thisModuleInstance` variable (used in the `ModuleInstance` namespace) contains box and script names, which are valid only in their own modules.

Container A display box object (such as a tab or outline) into which you drag other objects.

Design an Application

Before creating the application, you must identify its purpose, decide on which graphical elements to include, and figure out which JSL scripts you need to write.

Purpose What do you want the application to accomplish? Which features of JMP do you need to customize (for example, a launch window, report, or graph)?

Graphical Elements How many modules do you need? Look at the list of graphical objects in Application Builder and decide which objects to include in your application. The Sources panel shows dozens of boxes and icons that you can drag and drop into a module. Also look at the preinstalled samples to see whether a similar module exists.

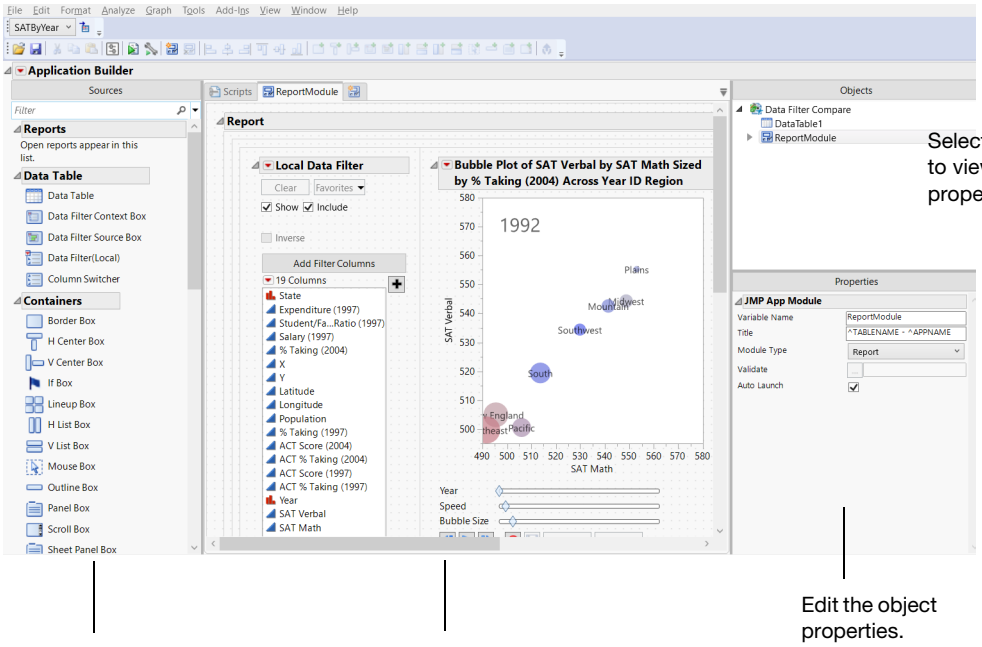
Scripting Depending on the complexity of the application, you write JSL scripts. See "Write Scripts" for more information about writing scripts for an application.

A non-interactive application that only shows objects does not require scripts. One example is a window showing reports that you have already generated in JMP.

Application Builder Window

Figure 16.4 shows an application under development in Application Builder.

Figure 16.4 The Application Builder Window



The Application Builder window provides the following features:

- The toolbar provides quick access to many features, such as aligning objects and inserting common display boxes. (Select **View > Toolbars > Application Builder** to show the toolbar.)
- The Sources panel shows objects that you can include in the application, including open reports and graphs. Right-click an object and select **Scripting Help** for information about the object and its properties.

- The workspace in the middle shows dotted grid lines, which help you line up objects. Drag objects onto one or more Module tabs in the workspace and write scripts on the Scripts tab to control the objects' functionality.
- The Objects panel shows a hierarchical view of the application. Along with the application, each module and its nested objects are listed. You can click an object to show its properties or to select it on the workspace.
- In the Properties panel, you set properties for each object, such as the location, width, or name of an object. The properties vary depending on the type of object.

Tip: To hide the grid and turn off snap-to-grid for all applications that you create, select **File > Preferences > Platforms > JMP App** and deselect these options. You can also deselect these features in the Application Builder red triangle menu.

Application Builder Red Triangle Options

The Application Builder red triangle menu provides options for running and debugging the application, opening sample applications, showing the grid, and more.

Run Application Starts the application. A window for each module opens, and you can interact with the application as the user will.

Debug Application Opens the script in the JSL Debugger to troubleshoot errors. See "Debug or Profile Scripts".

Snap to Grid Aligns (or snaps) objects to the nearest dotted grid lines as you drag them on the workspace. Selected by default.

Show Grid Displays dotted grid lines on the workspace. Selected by default.

Show Properties Shows or hides the Objects and Properties panels.

Auto Scroll Automatically scrolls horizontally or vertically as you drag an object near the edges of the workspace. Selected by default.

Dashboard Mode Displays the application in the simpler Dashboard Builder window. When you drag a report, data filter, or data table to the application, it is automatically placed in splitter and tab page boxes. You cannot add scripts in Dashboard Mode.

Be aware that switching back and forth does not change anything that you have already added to the application. Switching modes changes the way the interface works when you add new content.

Source Panel Shows or hides the Sources panel (the left column in which application components are displayed) and categorizes the icons. Both settings are selected by default.

Save Script Lets you save the application to a data table, journal, script window, template, or add-in. See "Options for Saving Applications".

Table 16.1 Sample Applications Installed with JMP

Six Quality Graphs.jmpappsource	Creates three Control Charts, a Distribution report, and a Capability Analysis report.
Data Filter Compare.jmpappsource	Creates two Bubble Plots. Each plot has its own data filter and Tabulate report. See this application for an example of the `Data Filter Context Box` function.
Data Table Application.jmpappsource	Lets the user select data table columns to stratify and the sampling rate. Display the data table to see the results.
Graph Launcher.jmpappsource	Lets the user enter an equation and axes settings and then create a graph. The initial window remains open, so the user can modify the equation and create new graphs.
Instant App.jmpappsource	Combines two Multivariate reports (Principal Components/Factor Analysis and T Square with All Principal Components).
Instant App Customized.jmpappsource	A modified version of Instant App.jmpappsource. Options for selecting the principal components report, changing the marker size, and showing means are included.
Launcher With Report.jmpappsource	Lets the user select data table columns in a launch window and then creates a graph.
Parameterized Instant App.jmpappsource	Lets the user select data table columns and then creates two Multivariate reports. An argument is assigned to the Y role. This means that the reports can be created from any open data table, not just the table specified in the application.
Parameterized MSA Combo Chart.jmpappsource	Creates a collection of reports for Measurement System Analysis (MSA).
Presentation.jmpappsource	Creates an onscreen presentation, similar to a slideshow, with navigation buttons and an embedded script.

Table 16.1 Sample Applications Installed with JMP *(Continued)*

R Application.jmpappsource	Lets the user select columns and then shows multivariate data in the shape of a face (the Chernoff faces). Requires the R TeachingDemos package.
SAS Application.jmpappsource	Runs a SAS script and then adds the output to a report. Prompts you to log on to a SAS server if you are not already connected.

Create an Application

After writing the application specifications, you create a blank application in JMP and begin adding objects and scripts.

This section provides the basic steps for creating an application:

- "Create a New Application"
- "Arrange and Remove Objects"
- "Customize Object Properties"
- "Write Scripts"

Note: In each version of JMP, Application Builder takes advantages of new features that are not supported in earlier releases. For maximum compatibility, we recommend that you create applications in the earliest version of JMP that the application will be run in.

Create a New Application

To create a new blank application, follow these steps:

1. Select **File > New > Application** (or **File > New > New Application** on macOS).
2. Select the Blank Application.

   The Application Builder window appears.
3. Select **File > Save** and save the file as a JMP Source File. The extension is .jmpappsource.

After you create the blank application, you add open reports, display boxes, and other objects to the workspace. See "Example of Creating an Application".

Manage Modules

When you create a new application, a Module tab appears by default. You can add objects to this module to create interface items. To create additional windows for your application, you can add and customize new modules.

Add Objects to a Module

Follow these steps to add objects to a module:

1. In the Sources panel, select the type of object that you want to add.
2. Drag the object onto the **Module1** tab (or double-click the object).
3. Select the object and update its properties in the Properties panel. See "Customize Object Properties".
4. Add scripts to scriptable objects. See "Write Scripts".
5. (Optional) To prevent a module from launching when you run the application, select the module in the Objects panel and deselect **Auto Launch**. (You might do this while testing one out of multiple modules.)
6. Click the Application Builder red triangle and select **Run Application** to test your application.

   Depending on how the modules are used, it may not be possible to test one module without creating the other modules.

Create a Module

1. Select **Format > Add Module.**
2. In the Properties panel, select the **Module Type**:
   - Dialog
   - Dialog with Menu
   - Modal Dialog
   - Launcher
   - Report
   - Display Box

Tip: When you run an application, the window title is the name of the data table, if applicable, followed by a hyphen and the application name. Change the application name by modifying text in the JMP App object's **Name** property.

Rename a Module

Change the **Variable Name** in the module's object properties.

Delete a Module

Select **Format > Delete Module.**

Delete a Data Table

Select the Data Table object, right-click and select **Delete.**

Note: You cannot delete the data table if it has any associated objects. Delete any objects and then delete the data table.

Modal Dialog Modules

Modal Dialog modules have some special behaviors to be aware of when using them.

- `ret = Module1 << Create Instance()` will not return until the dialog is complete. By this time, the dialog has been destroyed. Therefore, the return value will not be a handle to the Module Instance. Like `New Window()`, the return will instead be of the form `{Button(1 | -1), User Data}`. (1 | -1) indicates whether the OK (1) or Cancel (-1) button was pressed.
- `User Data()` is a new property on the Module Instance. During execution of the dialog, the Module script can set the user data:

  ```
  thisModuleInstance << Set User Data(...);
  ```

  to store something to be parsed by the caller. The user data will be evaluated at the time that it is set. There is also a corresponding `Get User Data()`.
- Like `New Window()`, if there is no OK or Cancel button included, an OK button will be added.
- There is an optional `Validate` script property on the Module Instance. This script is only used for Modal dialogs and behaves like `On Validate()` in `New Window()`. It is called when the OK button is pressed, and returns 1 to accept the input or 0 to disallow the close.

  Tip: If you use the Validate property, you can call `thisModuleInstance << Set User Data()`.

- Unlike other module types, a Modal Dialog will not show until the module script has completed. For other module types, the window is created during the call to `thisModuleInstance << Create Objects`. JMP cannot display a Modal Dialog here because control would stop and there would be no way to initialize the contents of the boxes in the display. Since the window is not created yet, you will not be able to do actions like set the window title or set an on-close script.

Arrange and Remove Objects

Drag objects from the Sources panel onto the workspace or double-click them. A blue border appears around the edge of a selected object. Then you can position the selected object in several ways. You can also change the container of an object or insert an object into a new container.

Tip: To quickly select an object, select it in the Objects panel. This method is particularly helpful when the internal object completely covers the container, as with a border that has an internal text box.

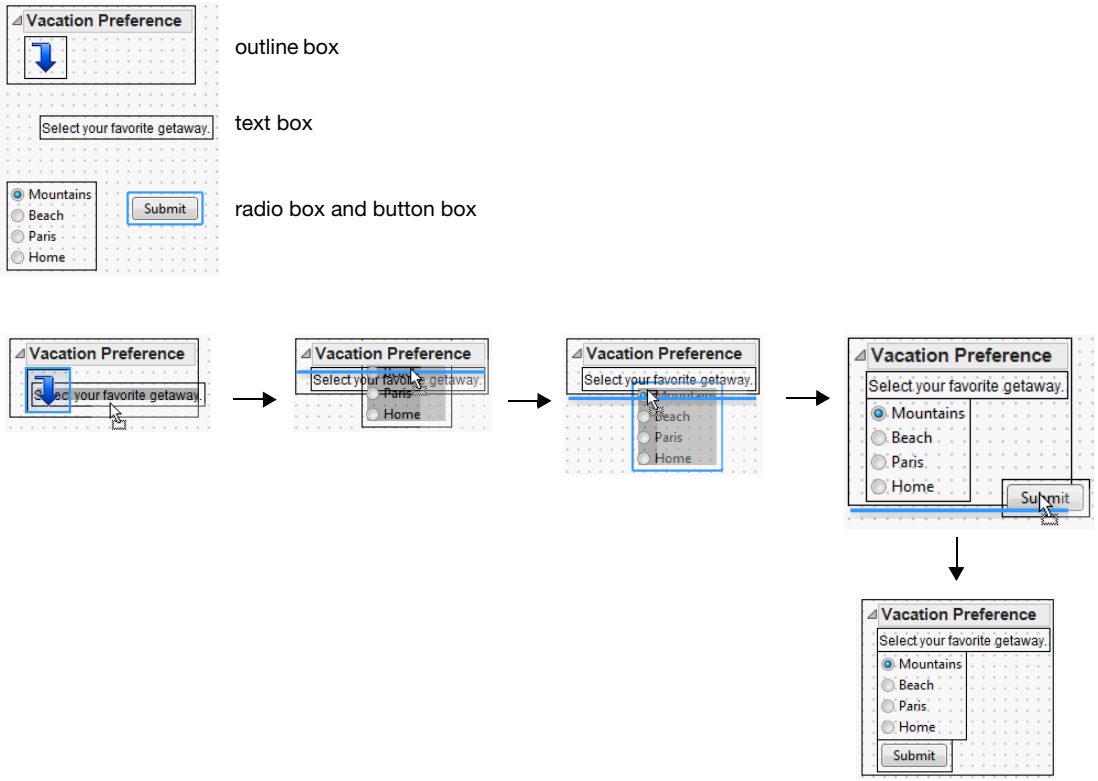
Drag an Object

One way to position objects is by dragging them around the workspace.

- The top and left edges of an object align (or snap) to the nearest dotted grid lines as you drag them on the workspace. For a more precise placement, click the Application Builder red triangle and deselect **Snap to Grid**. You can also turn off **Snap to Grid** for all applications in the JMP Preferences. Select **File (Windows) or JMP (macOS)> Preferences > Platforms > JMP App** to change the option.
- Position objects in containers instead of positioning them independently on the workspace. For example, place two button boxes in a V List Box to align them vertically.
- When you drag an object, press Shift to restrict movement to a single axis (for example, to move the object horizontally without moving the vertical position).
- When you drag an object into a container, the arrow indicates where you can drop the object.
- When you drag an object over another one, a blue line shows where you can drop the object.

Figure 16.5 shows the various methods for dragging objects inside a container.

Figure 16.5 Examples of Dragging Objects into a Container



Change the X and Y Positions on a Container Object

- To position a container object precisely, select the object and then change the **X Position** and **Y Position** properties. After you enter a new X position, press Tab to see the object move to the new position and then enter the new Y position.
- To place an object in the upper left corner, right-click the object and select **Move to Corner**. This is a shortcut to setting the X and Y positions to 0. On macOS, press Ctrl+Command and then select **Move to Corner**.

Line Up Multiple Objects

To line up objects horizontally or vertically, select the objects and then select an option from the **Format > Align Boxes** menu.

Tips:

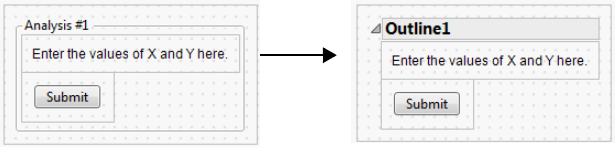
- The **Align Boxes** options are unavailable when you select a container rather than the internal objects.

- If you right-click a container, you might inadvertently select one of the internal objects instead. Right-click in the workspace instead.

Change the Type of Container

After inserting some containers, you can change them to a different type of container without having to re-create the object. Suppose that you create a panel that contains text and a button. To see what the panel looks like as an outline, select the panel, right-click in the module, and then select **Change Container > Outline Box**.

Figure 16.6 Change a Panel to an Outline



In this example, you would also change the outline title to match that of the panel.

Tip: To change the orientation of a list box quickly, select or deselect **Horizontal** in the object properties.

Insert Objects into a New Container

The toolbar at the top of the Application Builder window provides buttons for containers such as border boxes and mouse boxes (Figure 16.7).

To insert selected objects into a container, click the appropriate button on the toolbar or select **Format > Add Container**.

Figure 16.7 Container Toolbar

Duplicate Objects

When you copy and paste an object, the second instance of the object is renamed. The scripts attached to the object are also renamed. Pressing Ctrl (or Command on macOS) and dragging the object also creates duplicates of the originals.

Delete an Object

- Remove one or more objects by selecting them and then pressing Delete.
- Drag the objects outside of the module.
- On macOS, select the objects, press Ctrl+Command, and then select **Delete**.

- On Windows, select **Edit > Clear** to delete all objects in a module.
- Select the object and then press Backspace.

If you no longer need the object's script, you can delete the script also.

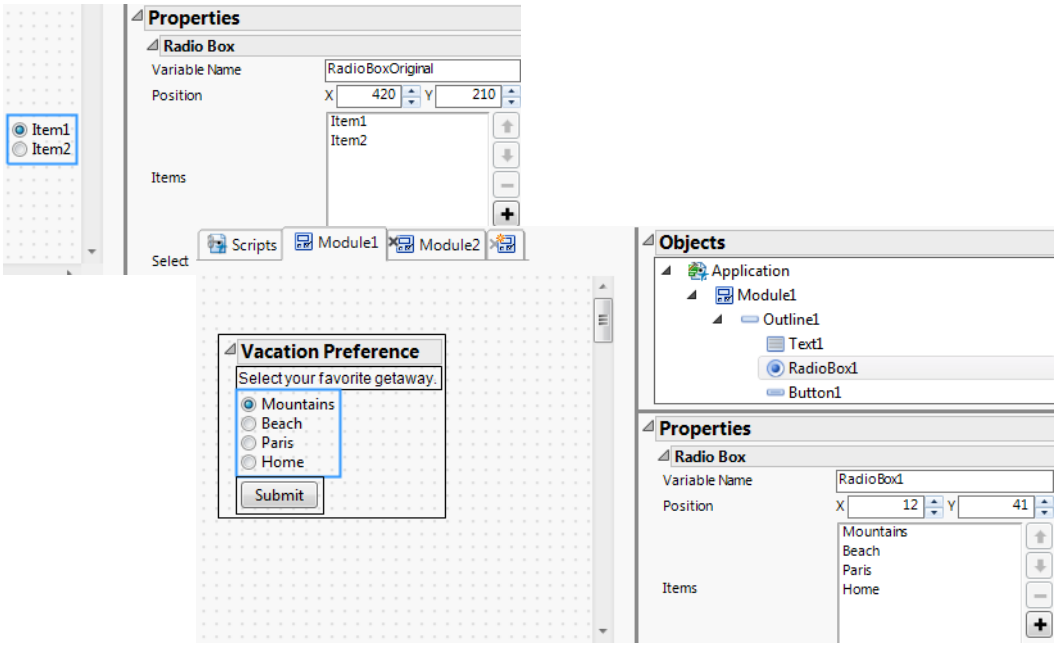
Customize Object Properties

After you drag an object from the Sources panel onto a module, customize the object properties in the Properties panel. Editing the object properties saves you the step of writing JSL for list items, button names, and so on.

Variable names are case- and space-insensitive. When a script contains an object named "Button1", Application Builder warns if you try to rename another object "Button 1".

To create the radio box list items in Figure 16.8, double-click the placeholder items "Item1" and "Item2" and enter the new list item.

Figure 16.8 Radio Box Object Properties



For more information about an object, right-click the object and select **Scripting Help**. The JMP Scripting Index appears, and the object that you selected is highlighted. The Scripting Index often includes a script that you can run to see an example of the object.

To read more about values in the Properties panel, hover over the value.

Application Properties

The Application properties identify the run password, data table name, and more. Select the application in the Objects list.

Name Appears after the data table name in the application's title bar.

Description Enter a description of the application.

Auto Launch Shows a launch window in which the user selects the arguments that have been defined in the application. This launch window is not one of the defined modules; the modules will be instanced based on their own Auto Launch property.

Encrypt Prevents users from editing the application in a text editor. Only applications that the user runs are encrypted (the .jmpapp file and the application in a JMP add-in). Scripts that you save to a data table, journal, and add-in are encrypted. For more information about encryption, see "Encrypt and Decrypt Scripts".

Run Password Enter the password required to run the application. To test the password, run the application outside Application Builder.

Table Module Properties

Inserting the Data Table object creates a data table object. To define properties such as the data table path, select the data table module in the Objects panel. Then modify the following options in the Properties panel.

Variable Name Specifies the name of the data table object. This name appears in the Properties and Objects panels and in the application's JSL script.

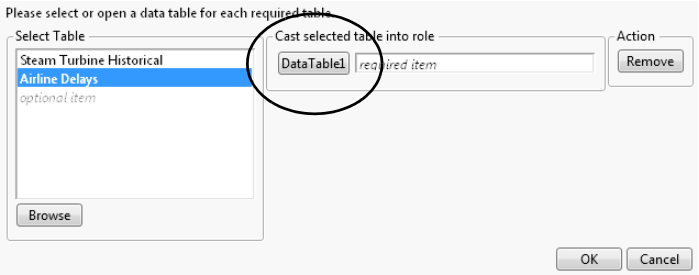
Path Specifies the absolute or relative filepath for the data table used in the application. You can precede the data table name with a path variable (such as $HOME or $USER_APPDATA).

Application Builder opens the specified data table when you edit the application. If the Path property is empty or the data table cannot be found, you are prompted to open the table.

When you close a data table and objects in the application depend on the data table, the objects are removed from the application and a warning message appears. To restore the objects, reopen the application.

Label Specifies the string used when prompting the user to open a data table. Figure 16.9 shows the default value.

Figure 16.9 Label in Data Table Prompt Window



Location Determines how the data table used in the application is selected when the user runs the application.

- Current Data Table: Uses the current data table. If no data tables are open, the user is prompted to open one. This option gives users the flexibility to use their own data.
- Full Path: Uses the data table specified in the Path property.
- Name: Uses the first open data table with the specified name. Otherwise, JMP uses the data table specified in the Path property. Note that if the application is moved to another computer, the link to the data table is broken. Consider storing the data table in a shared location.
- Prompt: Asks the user to select an open data table or browse to select a data table.
- Script: Uses the data table defined in the application or module script.
- Embedded Script: Reads the Source script in the data table that was used to generate the report. The Source script is run when you open the application.

Invisible Hides the data table from view but lists it in the JMP home window. This option is available for the Full Path and Name locations.

Write Scripts

After adding an object to the module, you write a script for the object to provide functionality. For example, you might want the user to click a button, select a directory, and then select a data table. Or you might have an application that displays a different graph based on which radio button the user selects.

Application and Module Namespaces

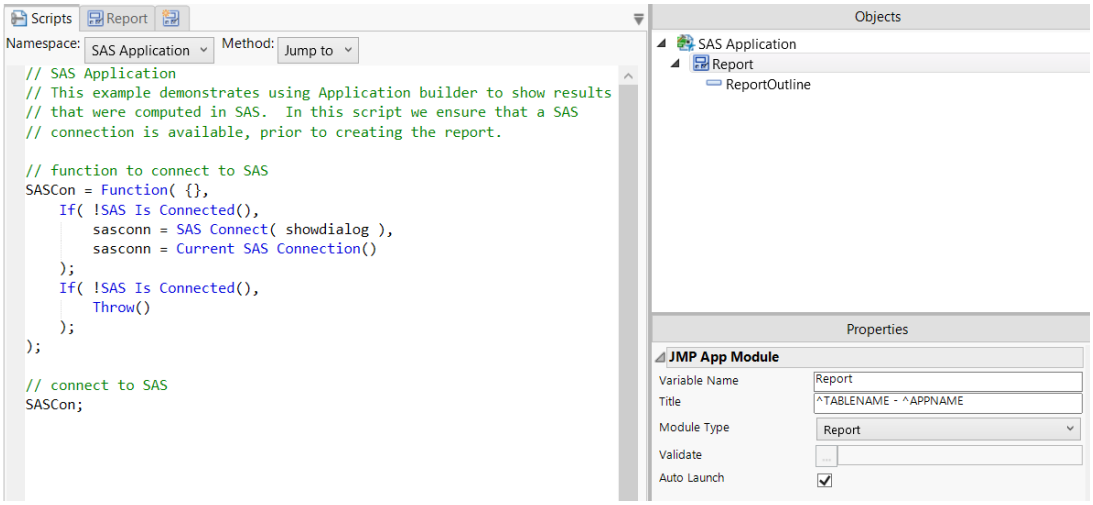
To prevent variable names and values from conflicting among scripts, Application Builder automatically creates the Application namespace and a namespace for each module:

- In the Application namespace, you write scripts that are executed when the user runs the application. Functions defined in the Application namespace can be used in any module.

- In the Module namespaces, you write scripts that are run when an instance of that module is created. If clicking a button in the application opens a new launch window, that launch window is an instance of the module. Two instances of the same module have their own copies of any variables or functions.

To see the scripts in these namespaces, click the **Scripts** tab and then select the namespace in the Namespace list (or in the Objects panel).

Figure 16.10 Application and Module Namespaces



There are two types of scripts: named scripts and anonymous scripts.

Write a Named Script

A *named script* is a function that several controls can use. The `this` argument tells which control is calling the function. In the following example, `Get Button Name` is sent to the `this` argument to print the button name to the log when the button is clicked:

```
Button1Press = Function({this}, Print(this <<Get Button Name))
```

On another button, use `Button1Press` script to produce the same results.

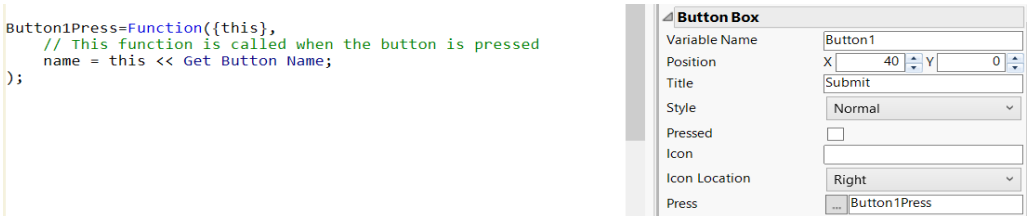
Two other advantages of named scripts: when you add a named script to an object, JMP inserts a placeholder script on the Scripts tab, which you then edit. On the Scripts tab, you can also select a script from the Method list to navigate to that code, which is particularly helpful in long scripts.

Add a named script:

1. Right-click the object, select **Scripts**, and then select the script that you want to add. In this example of creating a button, select the **Press** script. (On macOS, press Control, and then select **Scripts > Press**.)

The object's placeholder script appears on the **Scripts** tab (Figure 16.11). The name of the named script, *Button1Press*, shows up in both the script and the object properties.

Figure 16.11 New Script and Script Properties



2. Edit the object's placeholder script and properties to provide the necessary functionality. For example, the button box shown in Figure 16.11 was renamed. The Title property was changed to *Submit*, the text that appears on the button.

Tip: After you add a script to an object and then delete the object, delete the object's script from the **Scripts** tab if you no longer need the script. This feature prevents scripts that you might want in the future from being deleted.

If you rename the script in the object properties, rename it on the **Scripts** tab also. And if the script is used in another part of the application, rename it there as well.

Write an Anonymous Script

An *anonymous script* is available only to the object that defines it. For example, you might want a simple `Print` statement for one button that is not used elsewhere. By writing an anonymous script, you reduce the number of names to manage in the script. Anonymous scripts also reduce clutter among more important named scripts in the Scripts tab, because you add them to the object properties.

The following examples show two types of anonymous scripts:

```
Print(Button1 <<GetButtonName) // simple anonymous
Function({this}, Print(this <<GetButtonName)) // parameterized anonymous
```

Notice that the simple script sends a message to the "Button1" variable, but `this` is the control in the parameterized script.

Objects such as check boxes might provide additional arguments after `this` that are not otherwise available. One example is an argument that tells which check box in a column of check boxes just changed.

Write an anonymous script:

1. Select the object and click ... in the object properties.

   The anonymous script editor appears.

2. Enter the script and click **OK**.

   The text of the anonymous script appears in the object's properties (rather than the name of a named script).

Tip: Avoid copying and pasting anonymous functions to simplify code maintenance; use a named script instead if the script is needed in more than one place.

Show Specific Scripts

There are several ways to view a script for a specific object:

- Double-click the object on the **Module** tab.
- When an object has multiple scripts, right-click the object, select **Scripts**, and then select the script name. You can also select the **Scripts** tab, select the module name from the **Namespace** list and then select the script name from the **Method** list. Likewise, to see the application scripts, select **Application** from the **Namespace** list.

In each case, the **Scripts** tab appears with the cursor in the first line of the object's script.

Tip: To make a poorly formatted script easier to read, right-click and select **Reformat Script**.

Copy and Paste Objects with Scripts

When you copy and paste an object that has a script, the second instance of the object and script are renamed.

Edit a Platform Script

You can select the object for a platform in an application and edit the script for that platform. For example, you might edit the script for the Multivariate platform to show the scatterplot matrix or to use different columns.

Right-click the platform object on the workspace and select **Edit Platform Script**. After you edit the script and click OK, the object on the workspace is updated.

Edit or Run an Application

To open an application for editing, select **File > Open**, select the .jmpappsource file, and then select **Open**.

Note: If the Table property is empty or the specified data table cannot be found, the application runs anyway but objects requiring the data table will fail to create and JMP displays a warning.

To run an open application, click the Application Builder red triangle and select **Run Application**.

To run a closed application, select **File > Open**, select the .jmpapp file, and then select **Open**.

On Windows, you can also open or run an application from the JMP Home Window by doing one of the following:

- Drag the file from Windows Explorer onto the JMP Home Window or onto a blank application window.
- Double-click the file in Recent Files list.
- Right-click either the .jmpappsource or the .jmpapp file and select either **Edit Application** or **Run Application**.

Options for Saving Applications

JMP provides several options for saving application files.

- When you select **File > Save As** on Windows, you can save the application as an application source file (.jmpappsource), an application (.jmpapp), or a script.
- Selecting **File > Export** on macOS enables you to save the application as an application file or a script. To save the application as an application source file on macOS, select **File > Save As**.

The .jmpappsource file allows an application developer to save the application as is, in order to continue editing the application later. When reopened, the App Builder interface remains in same state as when it was saved. The data tables used to create the application must still be available.

The .jmpapp file is used to distribute applications for others to use. The application runs immediately after opening the file. The application resolves the data tables using the current data tables, files on disk, or by prompting the user, depending on the application settings.

The Save Script red triangle menu provides additional options for saving scripts. When an encrypted script is saved, JMP encloses the script in the `JSL Encrypted()` function to preserve white space and comments.

To Data Table Saves the script to the data table that was used to produce the report. This enables you to run the script again from the data table to recreate the results. When you edit this script from the data table, the application opens in Application Builder, not the script editor.

To Journal Saves a button that runs the script in a journal. The script is added to the current journal. The script contains the path to the data table. Note that if the data table cannot be found, the script does not run.

To Script Window Opens a script editor window and adds the script to it. If you have already saved a script to a script window, additional scripts are added to the bottom of the same script window. Note that if you modify the application significantly in a script window, you might not be able to edit the application in Application Builder. This option is useful when you want to embed the application in a larger JSL process.

Tip: This can also be accomplished by saving as a .jmpapp file and using `Open()` to run the application. However, saving the script can reduce the number of files.

To Add-In Lets JMP users install the application and launch it from a JMP menu. See "Compile Add-Ins with the JMP Add-In Builder".

To Template Saves the application as a template from which you can create new applications. The templates are saved in the following locations based on your operating system and JMP version:

- Windows (JMP): "/C:/Users/<username>/AppData/Roaming/SAS/JMP/<version number>/ApplicationTemplates"
- Windows (JMP Pro): "/C:/Users/<username>/AppData/Roaming/SAS/JMPPro/<version number>/ApplicationTemplates"
- macOS: "/Users/<username>/Library/Application Support/JMP/<version number>/ApplicationTemplates"

Additional Examples of Creating Applications

The following examples illustrate various uses for applications in JMP.

Parameterizing Variables

The following example shows how to create an application with parameterized variables. Users select the variables in a launch window, and then predefined reports are generated.

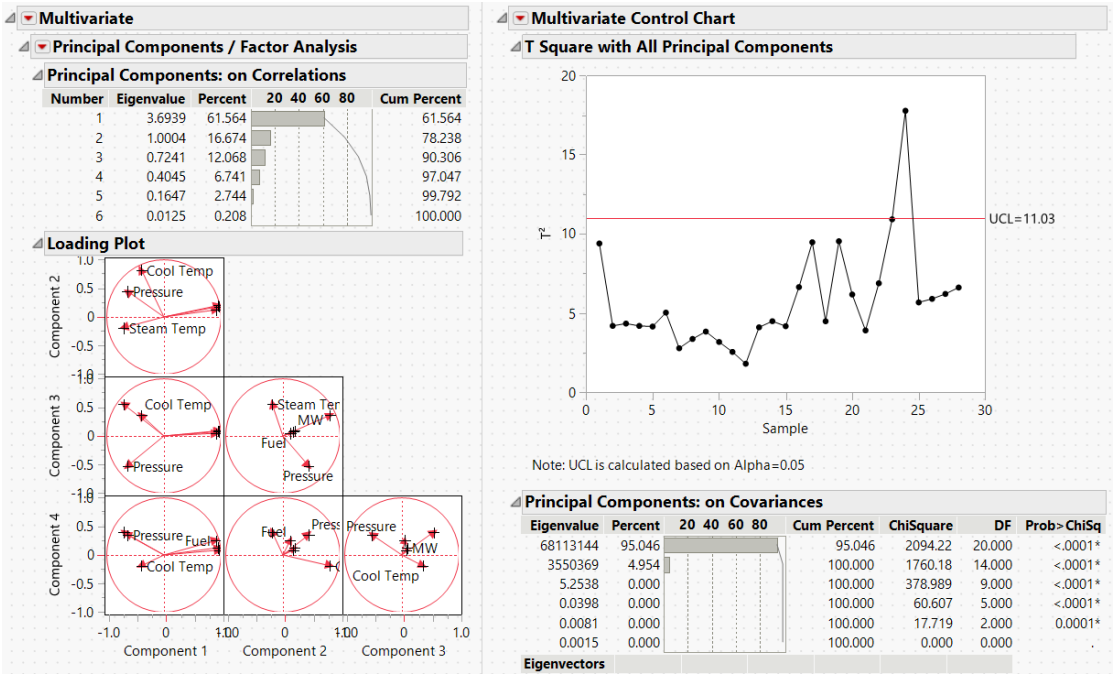
1. Select **Help > Sample Data Library** and open Quality Control/Steam Turbine Historical.jmp.
2. Run the Principal Component Analysis and Loading plot table scripts to generate the reports.
3. Select **File > New > Application**.
4. Select **Blank Application**.

   The Application Builder window appears.
5. Enlarge the window.
6. In the Sources panel, drag each Multivariate report onto the application workspace in a single row.

7. Select **Edit > Select All**.
8. Select **Format > Add Container > H List Box** to align the reports horizontally.
9. Select each report and type `yvar` next to **Y Variable** in the Objects panel.
10. Click the Application red triangle and select **Run Application**.
11. Select the Fuel, Steam Flow, Steam Temp, and MW variables and then click the **Y** button.
12. Click **OK**.

    New Multivariate reports appear in one window.

Figure 16.12 Multivariate Application



Tip: The absolute path to the data table is inserted automatically in the application's Table property. If the data is sample data, the `$SAMPLE_DATA` path variable is automatically used in the path. You can also enter an absolute or relative path. Remember that the user must have access to this path.

Filtering Data in Multiple Reports

In an application that contains several reports, you can select data in one report and then view only that data in all other reports contained in the same window.

To set up the filter, follow these steps:

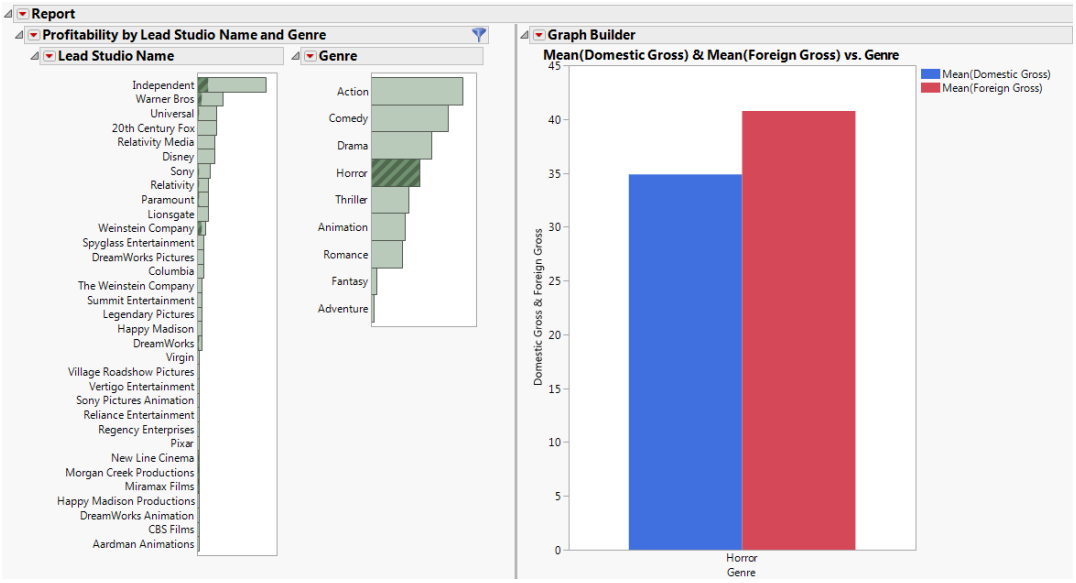
1. Create the application and add two reports to the workspace.
2. Select both reports. From the toolbar, select the HListBox icon.
3. Right-click the primary report and select **Use as Selection Filter**.

   This places the primary report display box in a Data Filter Source Box and the parent report display box in a Data Filter Content Box.
4. Click the Application Builder red triangle and select **Run Application**.

   The reports appear in one window (Figure 16.13).
5. Test the application by selecting a histogram bar in the primary report.

   Only data for the selected bar appears in the second report.

Figure 16.13 Example of Filtered Content



Filtering Data in Separate Reports Using a Local Data Filter

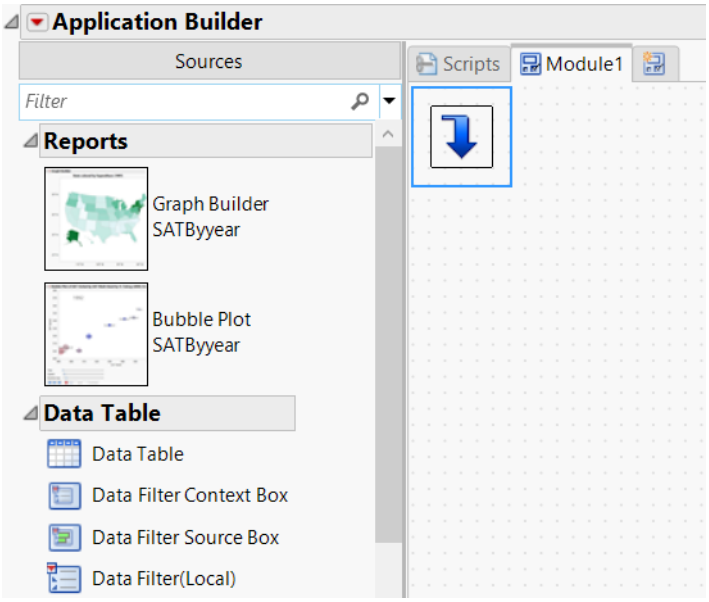
The following example illustrates how to add a Local Data Filter to multiple reports and then filter each report separately.

1. Select **Help** > **Sample Data Library** and open SATByYear.jmp.
2. Run these scripts: Bubble Plot by Region and Graph Builder Map.
3. Resize the report windows to make them smaller, so that a 2x2 arrangement will fit on the screen.
4. Select **File** > **New** > **Application** and choose **Blank Application**.

5. Double-click **Data Filter Context Box**.

   This adds a data filter context box to the workspace. Everything placed in this box will share the same Local Data Filter.

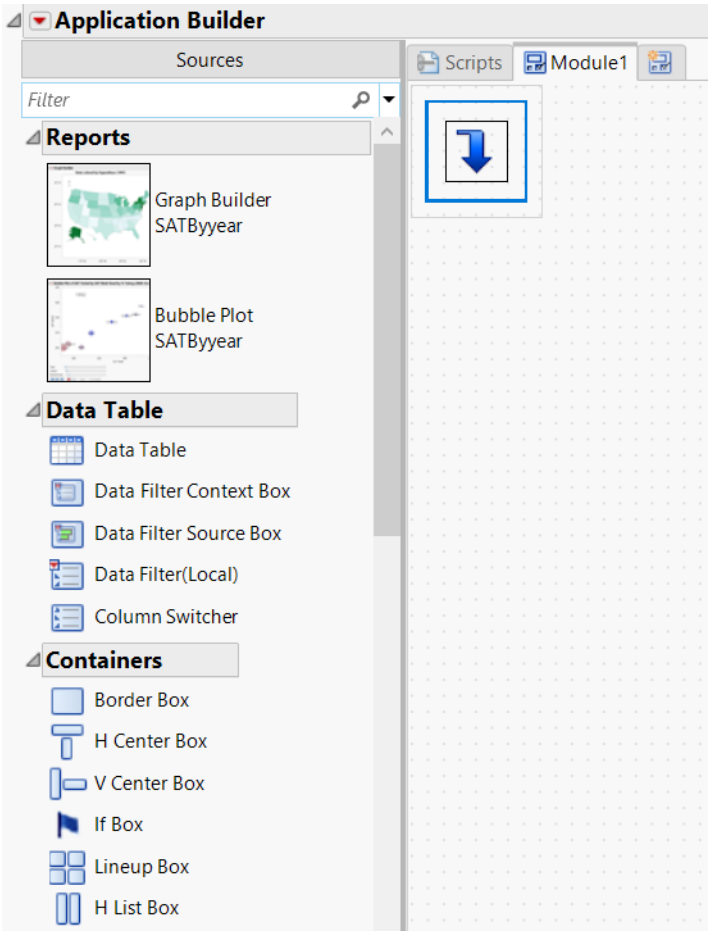
Figure 16.14 Adding a Data Filter Context Box



6. Under Containers, drag a **H List Box** into the Data Filter Context Box.

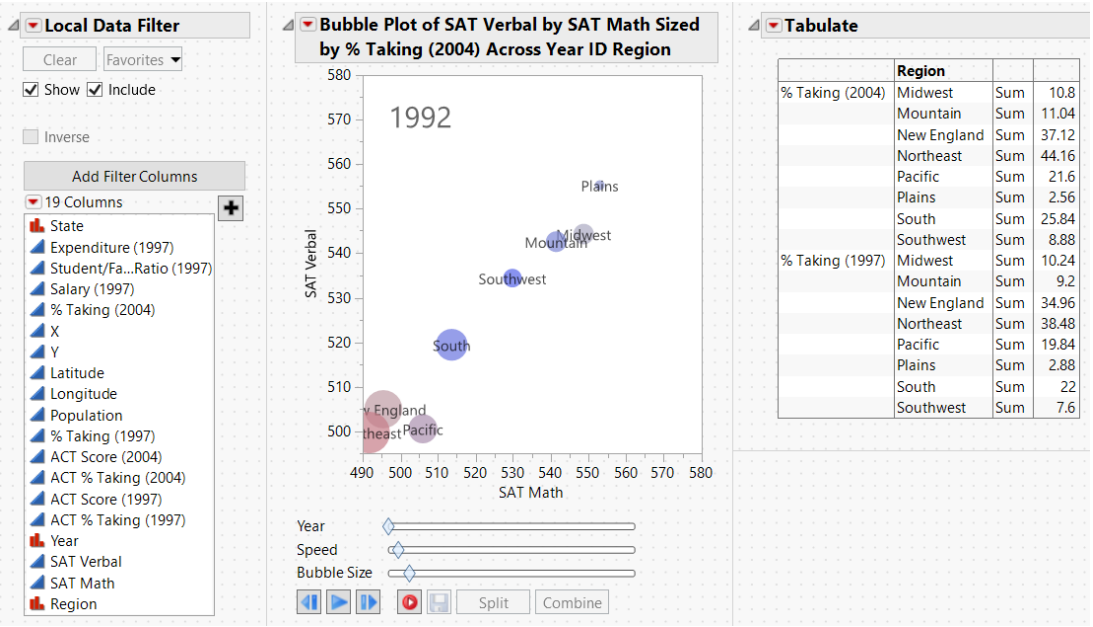
   This allows you to put multiple objects in this box and arrange them horizontally.

Figure 16.15 Adding a H List Box



7. Now drag a Data Filter (local), the Bubble Plot, and the Tabulate report into the H List Box.

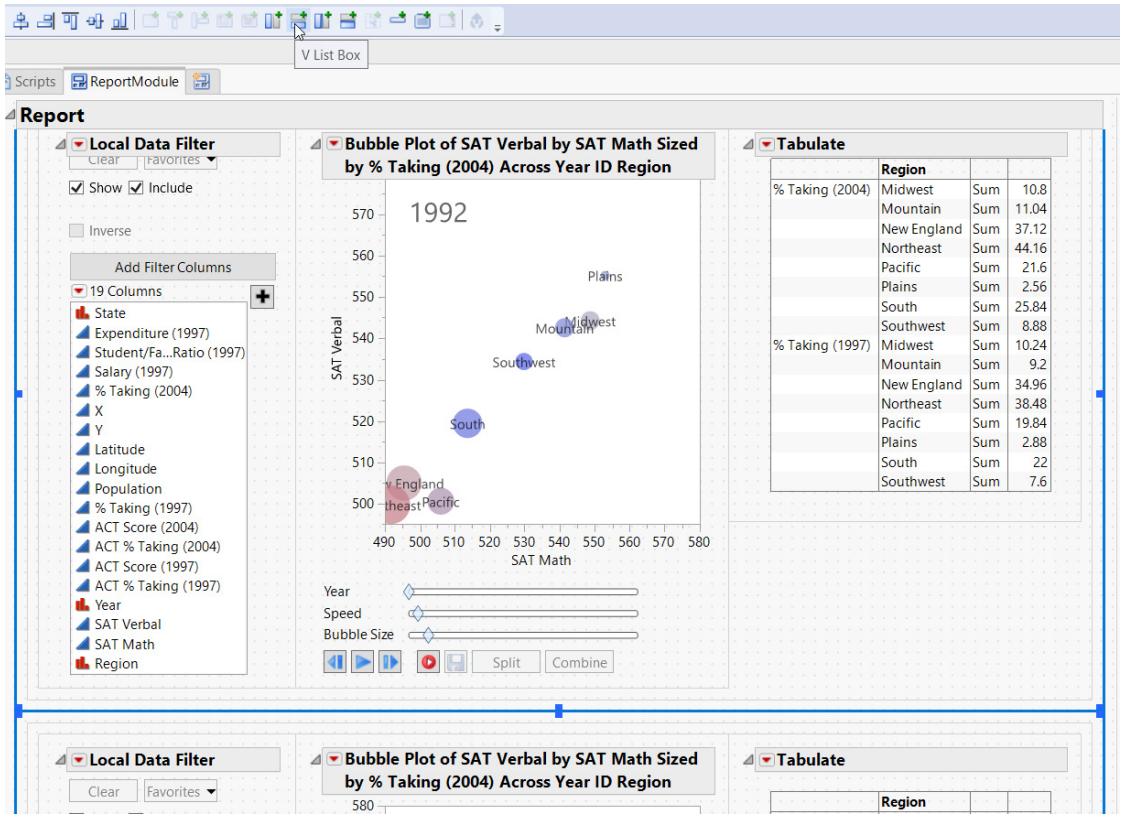
Figure 16.16 Adding Local Data Filter and Two Reports



8. Select the entire Data Filter Context Box. Press Ctrl and drag it below itself, to place a copy below the original set of reports.
9. Select both Data Filter Context boxes, then on the toolbar, click the V List Box icon.

   This arranges both reports into a single list box for display purposes.

Figure 16.17 Selecting Both Data Filter Context Boxes and the V List Box Icon



10. Click the Application Builder red triangle and select **Run Application**.
11. In the Local Data Filter on the top, click SAT Math, then click the Add button . Move the slider to see how it only impacts the reports at top. Do the same for the bottom set of reports.

Selecting Dates

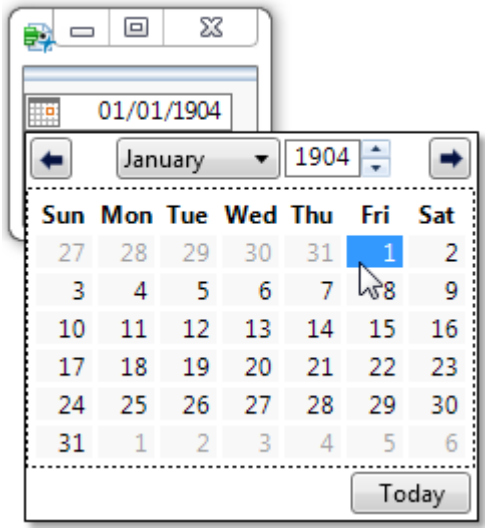
To insert a date selector window into an application, follow these steps:

1. Drag a Number Edit Box from the Sources panel to the workspace.
2. Select the Number Edit Box and then click the button next to Format in the Properties panel.
3. Select **Date** from the list and then select **m/d/y**.
4. Click outside the list.
5. Click the Application Builder red triangle and select **Run Application**.

   The Number Edit Box, which shows a date, appears in a new window.

6. Click the calendar icon to view the date selector window.

Figure 16.18 Example of a Date Selector



The date selector enables you to select the month and year as well as the date and time for the box.

Creating a Launch Window and Report

The Launcher with Report sample application combines a platform launch window with the resulting report. The sample application is displayed when you select **File > New > Application**.

The following example shows you how to arrange objects and include scripts that make up the application.You will recreate the Launcher with Report sample application so that you can create your own applications similarly.

Note: In the Launcher with Report sample application, the variable names of display boxes are customized to describe the display box. For example, “Text1” was renamed “Description” in the sample application. To reduce the number of steps in this example, you do not rename the variables unless one of the scripts refers to the variable name.

1. Select **File > New > Application** and click the **Launcher with Report** sample.

   You will copy and paste JSL from this sample into your copy. Notice that the Iris.jmp sample data table opens when you open the application.
2. Select **File > New > Application** and click **Blank Application**.

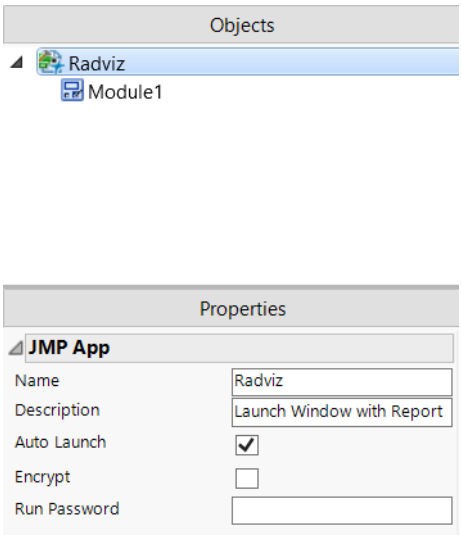
Create the Launch Module

The Launch Module contains objects that create a launch window.

1. In the Objects panel of the blank application, select Application.
2. In the Properties panel:
   - Change the Name from Application to “Radviz” and press Enter.

     This name appears in the title bar when you run the application.
   - Change the Description to “Launch Window with Report”.

Figure 16.19 Customizing the Application

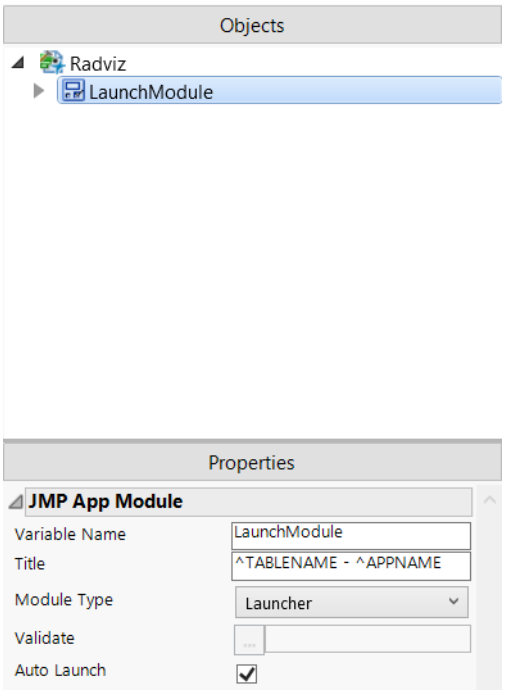


3. Click the **Scripts** tab and select Radviz from the Namespace list.
4. After the green comments, on a new line, type the following expression:

   ```
   dt = Current Data Table();
   ```

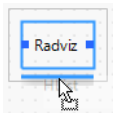
   One of the scripts that you add to this application refers to `dt` to identify the data table.
5. In the Objects panel, select **Module1**.
6. In the Properties panel:
   - Change the Variable Name to “LaunchModule” and press Enter.
   - Change the Module Type to Launcher.
   - Make sure that Auto Launch is selected so that the launch window opens when you run the application.

Figure 16.20 Customizing the LaunchModule



7. Click the **LaunchModule** tab and drag a V List Box (under Containers) onto the workspace.
8. Drag a Text Box (under Display) over the arrow of the V List Box.
9. Double-click Text1, change it to “Radviz”, and then press Enter.
10. Drag an H List Box (under Containers) below the border of the Text Box until a blue line appears.

Figure 16.21 Adding the H List Box



11. Because you have made many changes, select **File > Save** and rename the file. Make sure that .jmpappsource is selected as the file type.

Create the Select Columns Box

1. Drag a Panel Box (under Containers) over the H List Box arrow.
2. In the Properties panel, change the Title to “Select Columns” and press Enter.
3. Drag a Col List Box(All) (under Input) over the Panel Box arrow.

4. In the Properties panel, change the Variable Name to “ColumnList” and press Enter.

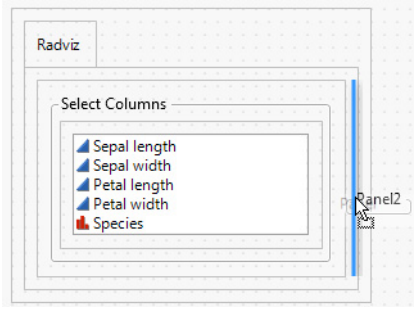
   The script you add later refers to this variable name.
5. Select **DataTable1** in the Objects panel.

   The current data table, Iris.jmp, appears in the Path box.

Create the Cast Selected Columns into Roles Box

1. Drag a Panel Box (under Containers) to the right of the existing Panel Box until a blue line appears.

Figure 16.22 Adding the Panel Box



Note: Make sure that you drag the Panel Box to the H List Box shown in Figure 16.22.

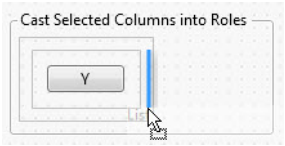
2. In the Properties panel, change the Title to Cast Selected Columns into Roles and press Enter.
3. Drag an H List Box (under Containers) over the Panel Box arrow.

Create the Contents of the Cast Selected Columns into Roles Box

1. Drag a Button Box (under Buttons) over the H List Box arrow.
2. In the Properties panel:
   - Change the Title to “Y” and press Enter.
   - In the Press box, type “SetY” and press Enter.

   You copy and paste the script for this button later.
3. Drag a Col List Box (under Input) inside the right border of the Y Button Box.

Figure 16.23 Adding the Col List Box



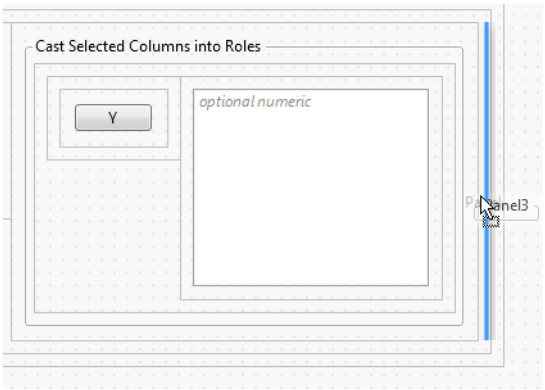
4. In the Properties panel:
   – Change the Variable Name to "YList" and press Enter.
     The script that you add later refers to this variable name.
   – Change the Data type to Numeric.
     The words "optional numeric" appear in the Col List Box.

Tip: Change Min Items to display the minimum number of required items in the Col List Box. If you type 3, then "required numeric" appears three times in the box as a hint to the user.

Create the Action Panel

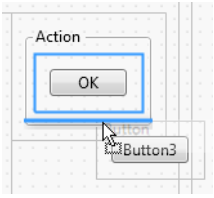
1. Drag a Panel Box (under Containers) to the right of the Panel Box that you just added.

Figure 16.24 Adding a Panel Box



2. In the Properties panel, change the Title to "Action" and press Enter.
3. Drag a Button Box (under Buttons) over the Panel Box arrow.
4. In the Properties panel:
   – Change the Title to OK and press Enter.
   – In the Press box, type "OnOK" and press Enter.
5. Drag another Button Box (under Buttons) below the OK button.

Figure 16.25 Adding a Cancel Button



6. In the Properties panel:
   - Change the Title to “Cancel” and press Enter.
   - In the Press box, type “OnCancel”.
7. Drag a Spacer Box (under Spacers) below the Cancel button.
8. Drag a Button Box (under Buttons) below the Spacer Box.
9. In the Properties panel:
   - Change the Title to “Remove” and press Enter.
   - In the Press box, type “OnRemove” and press Enter.
10. Select the outer border of the objects on the workspace.
11. Right-click and select **Move to Corner**.

    The extra space around the selected object is removed.
12. Select **File > Save** to save your changes in the.jmpappsource file.

Create the Launch Module Script

1. Go back to the Launcher with Report sample application window.
2. Click the Scripts tab.
3. Select LaunchModule as the Namespace and copy the script.
4. Go back to your application window and click the Scripts tab.
5. Delete the script in the LaunchModule namespace and paste the script from the sample application.
6. Select **File > Save** to save your changes in the.jmpappsource file.

Create the Report Module

The Report Module contains an object that builds a graph.

1. In your application window, click the button to the right of the LaunchModule tab to add a new module.
2. In the Properties panel, change the Variable Name to “ReportModule” and press Enter.
3. Make sure that the Module Type is Report.

4. Deselect Auto Launch to prevent the report from opening when the application is run.
5. Drag an Outline Box (under Containers) to the workspace.
6. In the Properties panel, change the Title to “Radviz” and press Enter.
7. Drag a Graph Box (under Display) over the Outline Box arrow.
8. In the Properties panel, change the Variable Name to “Graph” and press Enter.
9. Select the outer border of the objects on the workspace.
10. Right-click and select Move to Corner.

    The extra space around the selected object is removed.
11. Select **File > Save** to save your changes in the.jmpappsource file.

Create the Report Module Script

1. Go back to the Launcher with Report sample application window.
2. Make sure that the Scripts tab is displayed.
3. Select ReportModule as the Namespace and copy the script.
4. Go back to your application window and click the Scripts tab.
5. Delete the script in the ReportModule namespace and paste the script from the sample application.
6. Select **File > Save** to save your changes in the.jmpappsource file.

Your application should look like the Launcher with Report sample application.

Create the Radviz Script

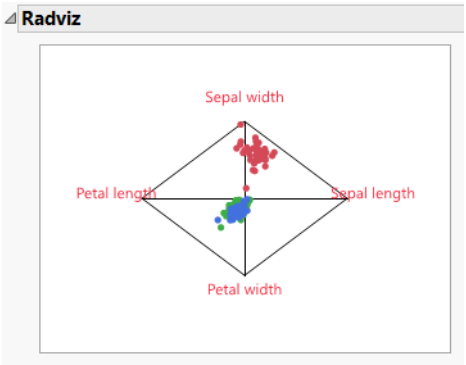
1. Go back to the Launcher with Report sample application window.
2. Make sure that the Scripts tab is displayed.
3. Select Radviz as the Namespace and copy the script.
4. Go back to your application window and click the Scripts tab.
5. Delete the script in the Radviz namespace and paste the script from the sample application.
6. Select **File > Save** to save your changes in the.jmpappsource file.

Test and Save the Final Application

1. In your application window, click the Application Builder red triangle and select **Run Application**.
2. Select the Sepal length and width and Petal length and width columns, click **Y**, and then click **OK**.

A graph of the data should appear.

Figure 16.26 Iris.jmp Graph



If the graph does not appear or errors are generated, make sure that you typed in the correct variable names when necessary. Also make sure that you added the scripts to the Scripts tab for each module.

3. On Windows, select **File > Save As**, select **JMP Application File** from the Save as type list, and then click **Save**.

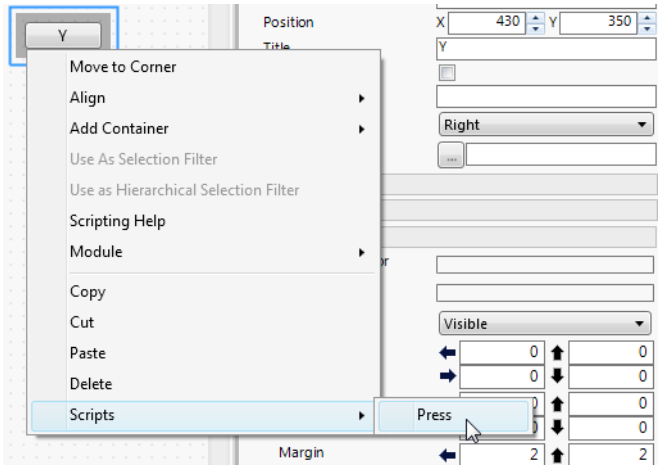
   On macOS, select **File > Export**, select JMP Application, and click **Next** to save the application.

   Your newly created application should match the Launcher with Report sample application.
4. Distribute the .jmpapp file to users. This file cannot be modified.
5. Save the .jmpappsource file so you can edit the application and resave it as a JMP application.

About the Application Scripts

When you create an application, you write scripts to provide functionality to display boxes. You can right-click a display box and select a script from the Scripts menu.

Figure 16.27 Adding a Script Interactively



Selecting a script from the Scripts menu adds a placeholder script to the Scripts tab for the selected module. For example, the Press script shown in Figure 16.27 adds the following script to the Scripts tab:

```
Button5Press=Function({this},
   // This function is called when the button is pressed
   name = this << Get Button Name;
);
```

Fill in the placeholder script with your own JSL. In the application that you previously created, you named two variables `SetY` and `ColumnList`. Rewrite the preceding placeholder script:

```
SetY = Function({},
   // This function will be called when the Y button is
   // pressed

   items = ColumnList << Get Selected;
   YList << Append(items);
);
```

`Get Selected` is passed to the `ColumnList` because you gave the Col List Box this variable name (`ColumnList`) in the object properties. `Get Selected` returns a list of the selected columns. `Append(items)` appends the selected columns to YList, the variable assigned to the Col List Box in the Cast Selected Columns into Roles window.

As shown when you previously created the application, you can write the script from scratch and indicate the script name in the object properties. To get help with syntax for a selected object, right-click the object and select Scripting Index.

Compile Add-Ins with the JMP Add-In Builder

A JMP add-in is a JSL script that you can run anytime from the JMP Add-Ins menu. You can create submenus to group your JMP add-ins and have many levels of menus, if desired.

Experienced JSL script writers can create scripts that extend JMP in many ways (for example, add a custom analytical tool or a user interface to communicate with a database). The JMP add-in architecture simplifies deploying and using these complicated scripts.

You can send co-workers a suite of scripts and tell them how to run them. Another option is sending a single add-in file that any JMP user can install and then use just like any other part of JMP.

As a JMP user, you might be given add-ins to use by co-workers. You can also find add-ins on the JMP website at https://www.jmp.com/addins.

Compile Scripts into an Add-In Using Add-In Builder

To create a JMP add-in, you write the scripts and gather files that are used in the add-in. Then you compile the scripts into an add-in.

- On Windows, select **File > New > Add-In**.
- On macOS, select **File > New > New Add-In**.
- In Dashboard Builder, click the red triangle and select **Save Script > To Add-In**.

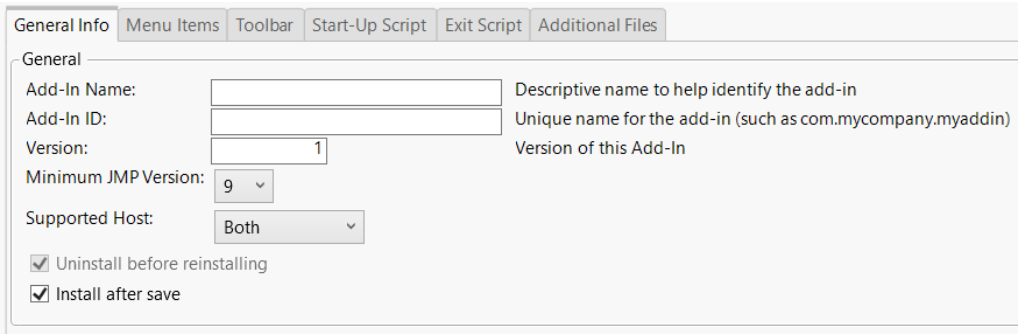
The process of compiling scripts into an add-in involves the following steps:

- "Add General Information"
- "Create Menu Items"
- "Create Toolbars (Windows Only)"
- "Specify Start-up and Exit Scripts (Optional)"
- "Add Additional Files"
- "Save the Add-In"
- "Test the Add-In"

Add General Information

First, in the **General Info** tab, add the general information that identifies and sets up your add-in.

Figure 16.28 Add-In Builder General Info Tab



1. Enter a name for the add-in.

   This is the name of the registered add-in, which appears in the **View > Add-Ins** window.

2. Enter a unique identifier string.

   Unique ID strings are case-insensitive. To ensure uniqueness, it is strongly recommended to use reverse-DNS names (for example, `com.mycompany.myaddin`). The ID string must meet the following requirements:

   – It can be up to 64 characters in length.
   – It must begin with a letter.
   – It should consist only of letters, numbers, periods, and underscores.
   – It should contain no spaces.

   In JSL, use this string to refer to the add-in (for example, in calls to `Get Addin()` or as an argument to $ADDIN_HOME).

3. Enter the version of the add-in.

   If you decide to make changes to the add-in at a later date, you can update the version number and then verify that users have the correct version.

4. Select the minimum version of JMP that the add-in works on.

   Note: Add-ins were introduced in JMP 9, so no prior versions are supported. And when saving an application as an add-in, select 10 or 11 as the JMP minimum version.

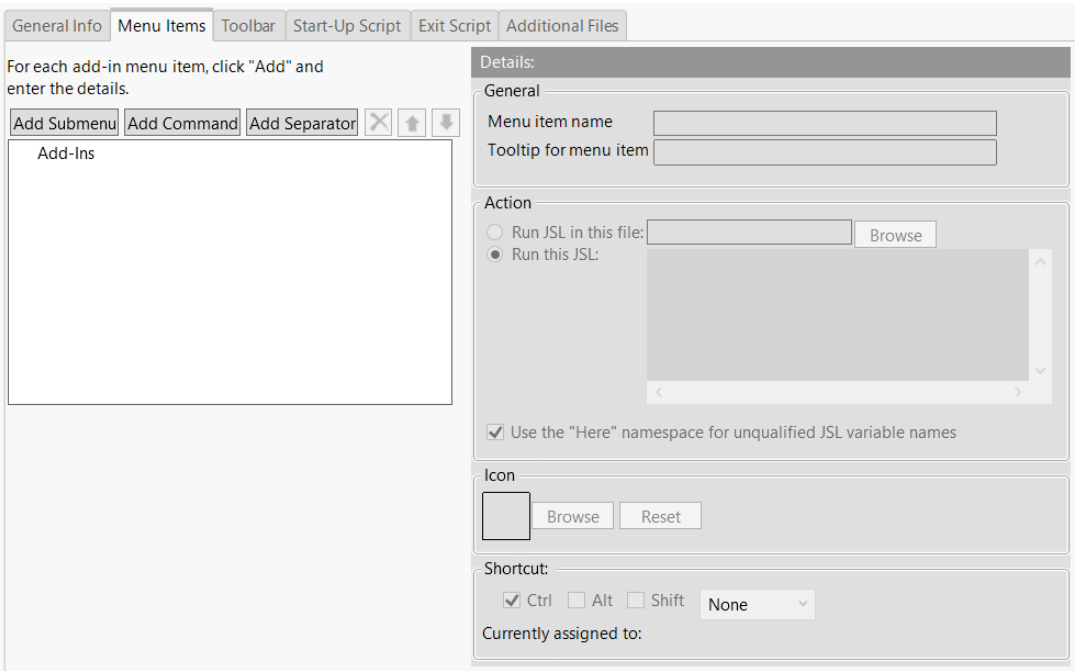
5. Select whether you want the add-in to be supported on Windows, macOS, or both.
6. (Optional) Select the check box next to **Install after save** if you want to install the add-in after saving it.

   If you do not select this option, the add-in is not installed once you save it, and it does not appear as a selectable menu item in the **Add-Ins** menu.

Create Menu Items

1. Click the **Menu Items** tab.

Figure 16.29 Add-In Builder Menu Items Tab



2. (Optional) Click **Add Submenu**.

   If you have multiple add-ins, you can group them under a submenu.
3. If you add a submenu, next to **Menu item name**, enter the name of the submenu.

   This name appears in the **Add-Ins** menu.
4. Click **Add Command**.
5. Next to **Menu item name**, enter the name of the add-in command.
6. (Optional) Next to **Tooltip for menu item**, enter the content that appears as a tooltip when the users place their cursor over the menu item.
7. Add the script. Select either **Run this JSL** and copy and paste in your script, or select **Run JSL in this file** and click **Browse** to find the file containing your script.
8. (Optional) Select **Use the "Here" namespace for unqualified JSL variable names** to ensure that all unqualified JSL variables are in the `Here` namespace, and local only to the script.

Notes:

- If your script creates a custom menu or toolbar, the variables are in the `Here` namespace by default.
- For more information about the `Here` namespace, see "Advanced Scoping and Namespaces".

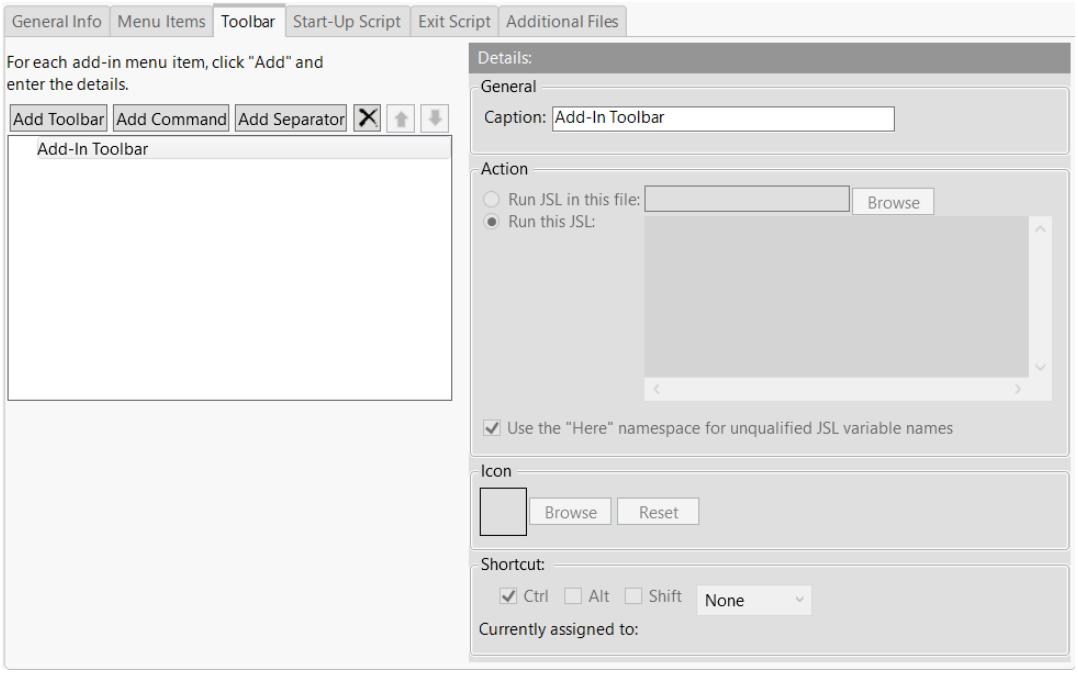
9. (Optional) Browse to add an icon that appears next to the menu item in the **Add-Ins** menu.
10. (Optional, Windows only) Create a keyboard shortcut for the add-in.
11. To add multiple menu items, repeat the steps.

    You can add multiple levels of submenus and add-in commands.
12. Click **Save** and save the add-in to the desired directory.
13. Click **Close**.

Create Toolbars (Windows Only)

1. Click the **Toolbar** tab.

Figure 16.30 Add-In Builder Toolbar Tab



2. Click **Add Command**.
3. Next to Caption, type the name of the toolbar item. This name is only for internal use. The user does not see it.

4. Add the script. Select either **Run this JSL** and copy and paste in your script, or select **Run JSL in this file** and click **Browse** to find the file containing your script.
5. (Optional) Select **Use the "Here" namespace for unqualified JSL variable names** to ensure that all unqualified JSL variables are in the `Here` namespace, and local only to the script.
6. Click **Save** and save the add-in to the desired directory.
7. Click **Close**.

Note: For more information about customizing menus in JMP, see *Using JMP*.

Specify Start-up and Exit Scripts (Optional)

Click the **Start-Up Script** tab to add a script that runs when JMP starts up (and the add-in starts). You can select an existing script (**Run JSL in this file**) or copy and paste in a script (**Run this JSL**). For example, you could provide a message telling the user that the add-in is installed upon start-up.

Click the **Exit Script** tab to add a script that runs when JMP exits or when you disable the add-in. You can select an existing script (**Run JSL in this file**) or copy and paste in a script (**Run this JSL**). For example, you could provide a prompt for the user to export a JMP data table upon exiting or disabling the add-in.

Add Additional Files

If your script calls other scripts, or contains graphics or data tables, add those files here.

Save the Add-In

Save the add-in by clicking the **Save** button on any tab. This effectively creates the add-in.

- If you selected the **Install after save** option in the General Info tab, then the add-in menu item appears in the **Add-Ins** menu immediately.
- If you did not select the **Install after save** option, when you open the saved add-in file, you are prompted to install the add-in.

Test the Add-In

Once your add-in is installed, test your add-in:

1. Select **View > Add-Ins**.
2. Select your add-in and click **Unregister**.
3. Reinstall your add-in by either selecting **File > Open** in JMP, or by double-clicking your .jmpaddin file.
4. Ensure that the menu and toolbar button run your script correctly and that the script itself runs correctly.

Edit an Add-In

To edit a saved add-in:

1. Select **File > Open**.
2. Navigate to and select the add-in file.
3. Select one of the following options:
   - On Windows click the arrow to the right of the **Open** button and select **Open using Add-In Builder**.
   - On macOS, select the **Edit after opening** option.
4. Click **Open**.

   The file opens in the Add-In Builder. Update the arguments and then save the changes.

Share an Add-In

Once you have a .jmpaddin file, you can share the file with other users. Email the file or you place it in a shared location, such as a network folder, or on the JMP File Exchange (located online in the JMP User Community).

When JMP users open the file, the add-in files are extracted into the appropriate folder, and the add-in is registered and installed. The add-in now appears in the user's JMP Add-Ins menu.

Installing Multiple Add-Ins

If you want to install multiple add-ins, copy the add-ins into the following location:

- Add-In files on Windows are located here:
  - %ALLUSERSPROFILE%/SAS/JMP/AddIns (any user on this machine can access the add-in)
  - %APPDATA%/SAS/JMP/AddIns (only the current user on this machine can access the add-in)
- Add-In files on macOS are located here:
  - /Library/Application Support/JMP/AddIns (any user on this machine can access the add-in)
  - ~/Library/Application Support/JMP/AddIns (only the current user on this machine can access the add-in)

When JMP starts, the addinRegistry.xml file is read, which contains information about previously registered JMP add-ins. Then JMP looks in the add-in folders for any other add-ins and installs them automatically.

Notes:

- The Home Folder for discovered add-ins does not have to be the AddIns subfolder in which the addin.def file was found. The addin.def file can be the only file in that subfolder and have a Home setting that points to some other location where the add-in files actually reside.
- If an automatically discovered add-in has the same unique ID as an add-in that was explicitly registered, the automatically discovered add-in is used.

Register an Add-In Using JSL

If your add-in files are not contained within a .jmpaddin file, you can use the `Register Addin()` JSL function to manually register the addin.def file. This installs and registers the add-in.

- For information about the JSL functions, see the *JSL Syntax Reference*.
- For information about creating the addin.def file, see "Create an Add-In Manually".

Notes:

- JMP might find a file named addin.def in the specified home folder. If so, values from that file are used for any optional arguments that are not included in the `Register Addin()` function.
- The addin.def file is used only for values that are not provided in the `Register Addin()` function. This function is useful while developing, but not necessary, since the addin.def file is enough to register an add-in.

Create an Add-In Manually

The addin.def file is a simple text file containing name-value pairs that provide registration information about a JMP add-in. Here are the name-value pairs to include in the addin.def file:

`name` Optional. The name that can be displayed in the JMP user interface wherever add-in names are displayed, instead of the unique ID. This name is displayed if no localized names are provided or when JMP is run under a language for which you did not provide a localized name. Named *Add-In Name* in Add-In Builder.

`name_xx` Optional. Allows the user-friendly name to be localized for different languages, where *xx* is the two-letter ISO 639-1 code for the language. If you include localized names,

you should still include a language-neutral name in case JMP is running under regional settings for which you do not have a localized name. Not available in Add-In Builder.

`id` Required. The unique ID for your add-in. The string can contain up to 64 characters. The string must begin with a letter and contain only letters, numbers, periods, and underscores. Reverse-DNS names are recommended to increase the likelihood of uniqueness. Named *Add-In ID* in Add-In Builder.

`home` Optional. The path to the add-in files. The Home Folder for the add-in is assumed to be the folder where `addin.def` is located. You need to include a setting for home only if the Home Folder is somewhere else (for example, a network shared folder). Not available in Add-In Builder.

`home_win` Optional. The path to the add-in files to be used when JMP is running on Windows. Overrides the value specified for home on Windows, if any. Not available in Add-In Builder.

`home_mac` Optional. The path to the add-in files to be used when JMP is running on the macOS. Overrides the value specified for home on macOS, if any. Not available in Add-In Builder.

`autoLoad` Optional, Boolean. The default value is True (1). Determines whether this add-in is initially configured to load automatically during JMP start-up. Named *Install after save* in Add-In Builder.

`host` Optional. Valid values are `Win` and `Mac`. Named *Supported Host* in Add-In Builder.

`minJMPVersion` Optional. Valid values are integers corresponding to the JMP major version that is the minimum version that the add-in supports. Named *Minimum JMP Version* in Add-In Builder.

`maxJMPVersion` Optional. Valid values are integers corresponding to the JMP major version that is the maximum version that the add-in supports. Use this setting *only* if there is a known incompatibility between your add-in and a specific version of JMP. You should provide a new version of the add-in for later versions of JMP. Not available in Add-In Builder.

Example of a addin.def File

```
id="com.mycompany.myaddin"
name="My Add-In's Friendly Name"
name_fr="My Add-In's French Name"
name_de="My Add-In's German Name"
home="\\server\share\myjmpaddin"
Autoload=1
MinJMPVersion=9
```

Example of a JMP Add-In

A sample add-in named Simple Calculator.jmpaddin is located in one of the following folders:

- On Windows: C:/Program Files/SAS/JMP/16/Samples/Scripts
- On macOS: /Library/Application Support/JMP/16/Sample/Scripts

Note: On Windows, in JMP Pro, the "JMP" folder is named "JMPPro".

To see what the add-in contains, change the extension to .zip and unzip it into a new folder. To see how it works, change the extension back to .jmpaddin and install.

The add-in contains the following files:

addin.def

Provides the specification for JMP to register the add-in. It contains only these two lines:

```
id="com.jmp.sample.calculator"
name="Simple Calculator"
```

addin.jmpcust

Provides the menu customization file that is created when you interactively create a custom menu. Your customizations are shown only to the current user; other users who log on to the computer and open JMP do not see your personal toolbars and menus. This example places the add-in menu item into the default **Add-Ins** menu.

calculator.jsl

A JSL script that creates a basic calculator.

calc_icon.gif

The image used as the calculator's icon.

To download more add-ins, visit
https://community.jmp.com/t5/File-Exchange/ct-p/FileExchange.

Manage JMP Add-Ins

Install Add-Ins

A JMP add-in is a file that has the .jmpaddin extension. You can install it in one of two ways:

1. Select **File > Open**.
2. Navigate to the .jmpaddin file and select it.

3. Click **Open**.

You can also double-click the .jmpaddin file or drag it onto the JMP Home Window.

View Your Add-Ins

Select **View > Add-Ins** to see the add-ins that you have already installed.

Reorder Your Add-Ins

Select **View** > **Add-Ins**, then use the Load Order arrows to move each add-in where you want it to appear in the Add-Ins menu.

Update Add-Ins

If you are given an update to an add-in you already have installed, you can just install the update as you did the original add-in. Doing so overwrites the old version with the new one.

Disable, Enable, and Remove Add-Ins

To temporarily disable an add-in without removing it

1. Select **View > Add-Ins**.
2. Select the add-in that you want to disable.
3. Deselect the **Enabled** check box.

To enable a disabled add-in

1. Select **View > Add-Ins**.
2. Select the add-in that you want to enable.
3. Select the **Enabled** check box.

To unregister an add-in

1. Select **View > Add-Ins**.
2. Select the add-in that you want to remove.
3. Click **Unregister**.

Chapter 17

Common Tasks

Getting Started with Sample Scripts

Examining working scripts line-by-line is one of the best ways to learn JSL. This chapter describes common tasks in JMP, such as converting date/time values and extracting specific values from reports. Sample scripts that address these issues are installed with JMP in the Samples/Scripts folder so you can run them yourself.

Contents

Run a Script at Start Up 927
Convert Character Dates to Numeric Dates 928
Format Date-Time Values and Subset Data 929
Create a Formula Column 930
Extract Values from an Analysis into a Report 932
Create an Interactive Program 936

Run a Script at Start Up

You can run the same script every time you start JMP. For example, you might include the definitions of some utility functions or a namespace that you want available throughout your JMP session. You might also set some preferences in a startup script to keep preferences consistent. Preferences persist across JMP sessions, but you could explicitly reset some preferences in case they were changed at some point in the previous session.

Name the script jmpStart.jsl and place it in one of the following folders, as appropriate for your operating system. When JMP starts, JMP looks for the jmpStart.jsl script in these folders in the order in which they are listed here. The first one that is found is run, and the search immediately stops.

Note: Some path names in this section refer to the "JMP" folder. On Windows, in JMP Pro, the "JMP" folder is named "JMPPro".

On Windows:

1. C:/Users/<username>/AppData/Roaming/SAS/JMP/16
2. C:/Users/<username>/AppData/Roaming/SAS/JMP

On macOS:

1. /Users/<username>/Library/Application Support/JMP/16
2. /Users/<username>/Library/Application Support/JMP

The jmpStart.jsl script runs only for a particular user on a computer. You can add a script named jmpStartAdmin.jsl in one of the following places, as appropriate for your operating system. This script is run for every user on a computer. JMP runs jmpStartAdmin.jsl first if found. Then JMP runs jmpStart.jsl if found.

On Windows:

1. C:/ProgramData/SAS/JMP/16
2. C:/ProgramData/SAS/JMP

On macOS:

1. /Library/Application Support/JMP/16
2. /Library/Application Support/JMP

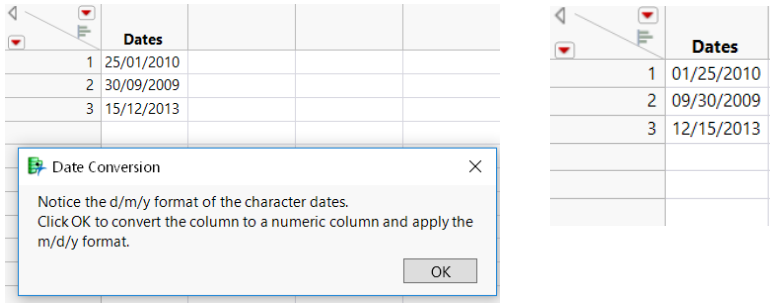
Convert Character Dates to Numeric Dates

Data might appear to be numeric in the data table. However, the column properties may specify a character data type. To manipulate the data as date/time values, convert the column to a numeric column and specify how you want the values to appear.

Convert Dates.jsl creates a data table, specifies the data input format, changes the column to a numeric continuous column, and applies the m/d/y format (Figure 17.1).

```
// Create a data table with character dates.
dt = New Table( "Example",
   Add Rows( 3 ),
   New Column( "Dates",
      Character,
      Nominal,
      Set Values( {"25/01/2010", "30/09/2009", "15/12/2013"} )
   )
);

// Display a modal dialog for the user to confirm the format conversion.
nw = New Window( "Date Conversion",
   <<Modal,
   tb = Text Box(
      "Notice the d/m/y format of the character dates.
Click OK to convert the column to a numeric column and apply the m/d/y
   format."
   )
);

/* Apply the Numeric data type.
Specify the Informat (input format) value "d/m/y".
Specify the Format (display format) value "m/d/y".
Apply the Continuous modeling type */
col = Column( dt, "Dates" );
col << Data Type( "Numeric", Informat( "d/m/y" ), Format( "m/d/y" ) );
col << Modeling Type( "Continuous" );
```

Figure 17.1 Converting Character Dates (Before and After)



When you change the column's data type from character to numeric, defining the format in which the data were entered is important. In this example, `Informat( "d/m/y" )` defines the input format. `Format( "m/d/y" )` defines the new display format. If `Informat()` is omitted, the `Format()` value is applied as both the input and display format. This results in missing values for some data.

Modify Convert Dates.jsl to see for yourself.

1. Open Convert Dates.jsl from the sample scripts folder.
2. Right-click the script window and select **Show Line Numbers**.
3. On line 9, change "25/01/2010" to "01/25/2010".
4. On line 27, delete `Informat( "d/m/y" ),` (including the comma).
5. Run the script.

   `Format( "m/d/y" )` is applied to the column. Only "01/25/2010" appears in the column. The other values are missing; "30/09/2009" and "15/12/2013" are not valid m/d/y values.

Format Date-Time Values and Subset Data

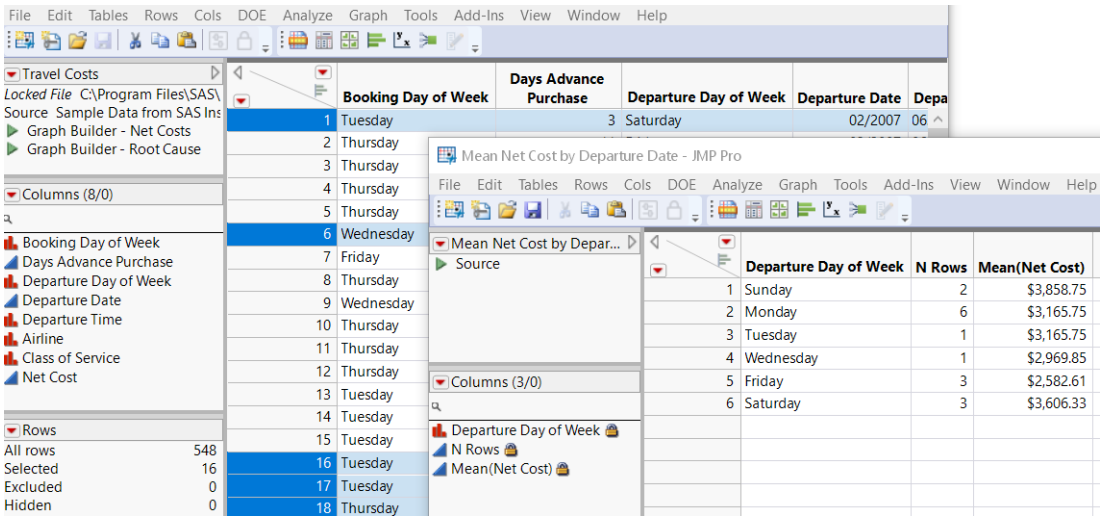
How can you work with dates in JMP? JMP provides a number of formats that you can use to make comparisons and then subset data based on the date.

Select Where Using Dates.jsl applies the Date MDY format to a column of departure dates and subsets the data. A summary table of mean net costs by departure date then appears (Figure 17.2).

```
/*  How can you work with dates in JSL? JMP provides a number of formats
that you can use to make comparisons and then subset data based on the date.
*/

hdt = Open( "$SAMPLE_DATA/Travel Costs.jmp" );
```

```
/* Apply the Date MDY format to Departure Date values and then select only
February dates. */
hdt << Select Where(
      (Date MDY( 02, 01, 2007 ) <= :Departure Date < Date MDY( 03, 1, 2007 ))
      );

/* Subset the selected rows into two tables: one table contains February
departure dates, the other contains all data for those departure dates. */
nt1 = hdt << Subset( Columns( :Departure Date ),
   Output Table Name( "February Departure Date" ) );
nt2 = hdt << Subset( Output Table Name( "February Data" ) );

/* Create a summary table, grouping mean cost by day of week that departure
took place. */
sumDt = nt2 << Summary(
   Group( :Departure Day of Week ),
   Mean( :Net Cost ),
   Output Table Name( "Mean Net Cost by Departure Date" )
);
```

Figure 17.2 The Original Table and the Final Summary Table

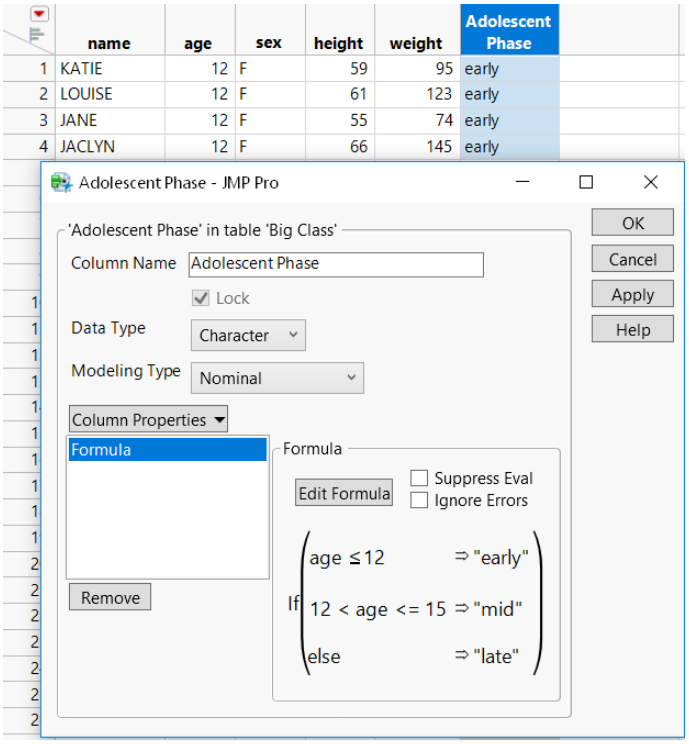


Create a Formula Column

How do you create a formula column that combines conditional expressions with value comparisons? Create a Formula Column.jsl shows how to create a new formula column that evaluates ages in Big Class.jmp and returns the result in the new column (Figure 17.3).

```
/* Scenario:

How do you create a formula column that combines conditional expressions
with value comparisons? This script shows how to create a new formula
column that evaluates ages in Big Class.jmp and returns the conditional
result in the new column. */

dt = Open( "$SAMPLE_DATA/Big Class.jmp" );

/* Create a new character column for the formula.
Insert "early" in the new column if the age is less than or
equal to 12. Insert "mid" if the age is less than or equal to
15 but greater than 12. For ages greater than 15, insert "late".
*/

dt << New Column( "Adolescent Phase",
   Character,
   Formula(
      If( :age <= 12, "early",
         12 < :age <= 15, "mid",
         "late"
         )
      )
);
```

Figure 17.3 Conditional Expression in Formula



Extract Values from an Analysis into a Report

How do you capture specific results of an analysis into a custom report using JSL?

The JMP platforms in the Analyze and Graph menus contain two objects known as the analysis and report layers. Messages are sent to the analysis layer that generate the desired results.

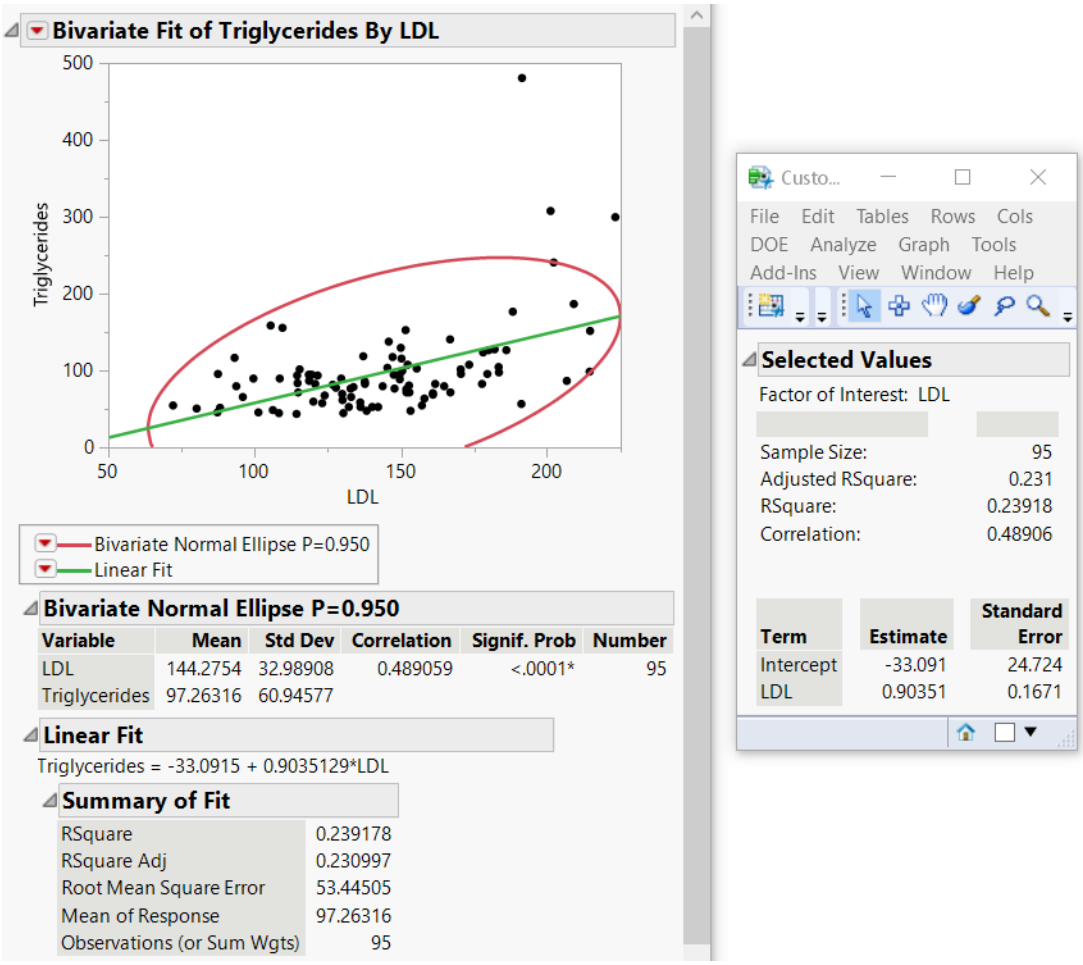
Extract Values from Reports.jsl performs a Bivariate analysis and shows results such as the sample size, RSquare, and Correlation in a new report window. Figure 17.4 shows the Bivariate report along with the customized report, though the Bivariate report window closes after the script is finished.

```
/* Scenario: How do you capture specific results of an analysis
in a report using JSL?

The JMP platforms in the Analyze and Graph menus contain two objects
known as the analysis and report layers.

Messages are sent to the analysis layer that generate the desired results.

This script performs a Bivariate analysis and shows results
such as the sample size, RSquare, and Correlation in a new report window.
*/

sd = Open( "$SAMPLE_DATA/Lipid Data.jmp" );

biv = Bivariate(        // biv is the analysis layer.
  Y( :Triglycerides ),
  X( :LDL ),
  Density Ellipse( 0.95, {Line Color( {213, 72, 87} )} ),
  Fit Line( {Line Color( {57, 177, 67} )} ),
);

/* Make sure the second Outline Box (called "Correlation")
in the Bivariate report is open. You can then see which content
is extracted into the Custom report. */
report(biv) [Outline Box( 2 )] << Close( 0 );

reportbiv = biv << Report;  // reportbiv is the report layer.

/* The density ellipse is generated first.
Extract the correlation coefficient. */
corrvalue = reportbiv[Outline Box( 2 )][Number Col Box( 3 )] << Get( 1 );

/* ...followed by Fit Line
Extract the numeric values from the Summary of Fit report
and place them in a matrix. */
sumfit = reportbiv[Outline Box( 4 )][Number Col Box( 1 )] << Get as Matrix;

// Extract the values of RSquare and AdjRSquare as one by one matrices.
rsquare = sumfit[1];
adjrsq = sumfit[2];
avg = sumfit[4];
samplesize = sumfit[5];

/* Extract the first column of the Parameter.
Estimates report as two objects. */
term = reportbiv[Outline Box( 7 )][String Col Box( 1 )] << Get();
```

```
// Clone the report layer as a String Col Box.
cloneterm = reportbiv[Outline Box( 7 )][String Col Box( 1 )] << Clone Box;

// Extract the Parameter Estimates values as a matrix.
est = reportbiv[Outline Box( 7 )][Number Col Box( 1 )] << Get as Matrix;

// Extract the Standard Error values as a matrix.
stde = reportbiv[Outline Box( 7 )][Number Col Box( 2 )] << Get as Matrix;

dvalues = [];
dvalues = samplesize |/ adjrsq |/ rsquare |/ corrvalue;
sfactor = term[2];

dlg = New Window( "Custom Report",
   Outline Box( "Selected Values",
      /* The Lineup box defines a two-column layout, each of which contains
      a Text Box. */
      Lineup Box( N Col( 2 ),
      Text Box( "Factor of Interest: " ),
      Text Box( sfactor ), ),
      tb = Table Box(
         /* Display an empty string in the first column
            and the text in the second column. */
         String Col Box( "  ",
            {"Sample Size: ", "Adjusted RSquare: ", "RSquare: ",
            "Correlation:"}
         ),
         // Insert a 30 pixel x 30 pixel spacer between the columns.
         Spacer Box( Size( 30, 30 ) ),
         ,
         /* Display an empty string in the first column
         and the dvalues in the second column. */
         Number Col Box( "  ", dvalues )
      ),
      // Insert a 1 x 30 spacer.
      Spacer Box( Size( 0, 30 ) ),
      ,
      Table Box(
         /* Display the cloned String Col Box followed by a spacer.
            Then insert the Parameter Estimates and Standard Error values. */
         CloneTerm,
         Spacer Box( Size( 10, 0 ) ),
         ,
         Number Col Box( "Estimate", est ),
         Spacer Box( Size( 10, 0 ) ),
```

```
            ,
            Number Col Box( "Standard Error", stde )
        )
    )
);
tb << Set Shade Headings( 0 ); // Turn off shaded table headings.
tb << Set Heading Column Borders( 0 ); // Turn off table column borders.
Close( sd ); // Close the data table.
```

Figure 17.4 Customized Report from the Bivariate Analysis



Create an Interactive Program

How do you gather numeric input from the users, perform a calculation on that input, and show the results in a new window?

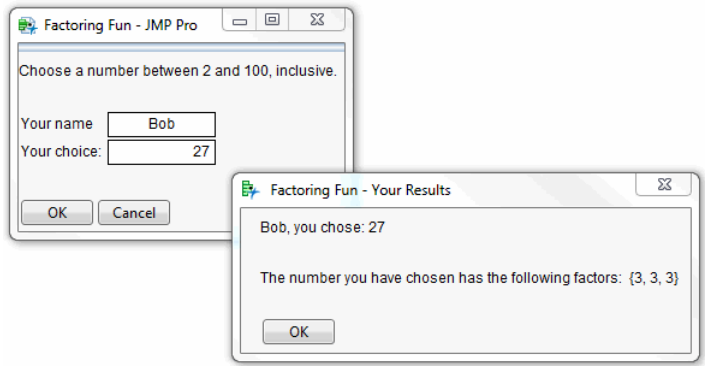
Prime Numbers.jsl asks the user to enter a number and then factors the number or confirms it as a prime number (Figure 17.5). This script is a good example of aligning several types of display boxes, concatenating text, and working with conditional functions.

```
/* How do you gather numeric input from the users, perform a
calculation on that input, and show the results in a new window?

This script demonstrates how to create an interactive program that
asks the user to enter a number and then factors the number or confirms
it as a prime number. */

// Ask the user to enter a name and number.
nw = New Window( "Factoring Fun",

V List Box(
   Text Box( "Choose a number between 2 and 100, inclusive. " ),
   Spacer Box( Size( 25, 25 ) )
    ),
    V List Box(
        Lineup Box(
            2,
            Text Box( "Your name   " ),
            uname = Text Edit Box( "< name > ", << Justify Text( Center ) ),
            Text Box( "Your choice " ),
            uprime = Number Edit Box( 2 )
        ),
        Spacer Box( Size( 25, 25 ) ),
        H List Box(
            Button Box( "OK",
         // Unload responses.
                username = uname << Get Text;
                fromUser0 = uprime << Get;

        // Test input for out of range condition.
                If( fromUser0 <= 1 | fromUser0 > 100,
           // Send message to user that input value is out of range.
                    nw2 = New Window( " Factoring Fun: Message for "
                      || username,
                        <<Modal,
                        Text Box(
                            "The number you chose, " || Char( fromUser0 ) ||
```

```
            " is not between 2 and 100, inclusive.  Please try
            again.  "
        ),
        Button Box( "OK" )

    ),
      /* Else the number is within range.
      Test for a prime number. If not prime, factor it.
      Create a vector which holds the prime numbers
      within specified range. */
    primes = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37,
    41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97];
    // Count the number of primes in the vector.
    p# = N Row( primes );

    isprime = 0; // Set flag.
    // Make a copy of the value for processing.
    fromuser1 = fromuser0;
    factors = {}; // Initialize list.

     /* Process the value by checking for prime then
     factoring if needed. */
    While( isPrime == 0,

        // Compare value to vector of prime numbers.
        If( Any( fromuser0 == primes ),
      // If found, place value in factor list.
            Insert Into( factors, fromUser0 );
            isPrime = 1 // Set condition to exit While loop.
            ;
        ); // End For loop.

        If( isprime == 0,
            For( q = 1, q <= p#, q++,
                If( Mod( fromuser0, primes[q] ) == 0,
                    fromUser0 = fromUser0 / primes[q];
                    Insert Into( factors, primes[q] );
                    q = p# + 1 // End if-then loop.
                    ;
                ); // End If loop.
            ); // End For loop.
        );  // End If/Then loop.

    ); // End while loop.

    cfUser0 = Char( fromUser1 );
```

```
                        nf = N Items( factors );
                        If( nf >= 2,
                            fmsg = "The number you have chosen has the following
                             factors: ",
                           fmsg = "The number you have chosen is a prime number: "
                        );
                // Show message to user about results.
                        nw3 = New Window( " Factoring Fun - Your Results",
                            <<Modal,
                            Text Box( username || ", you chose: " || cfUser0 ),
                            Spacer Box( Size( 25, 25 ) ),
                            Text Box( fmsg || " " || Char( factors ) ),
                            Spacer Box( Size( 25, 25 ) ),
                            Button Box( "OK" )
                        );
                    );
                ), // End the main OK button script.
                // Close the window and the program.
                Button Box( "Cancel", nw << Close Window )
            )
        )
    );
```

Figure 17.5 Factor Numbers Interactively



Appendix A

Compatibility Notes

Changes in Scripting in JMP 16

This appendix provides information about changes that affect compatibility in JSL scripts. Please update your scripts accordingly so that the scripts run correctly in the latest version of JMP.

Compatibility Issues in JMP 16.1

Legend Position

`Position()` has been phased out so that only `Legend Position()` is output in saved scripts. Previously saved scripts that include both the `Position()` and `Legend Position()` commands generate an error even though the script runs fine.

For example, change the following code:

```
{Legend Position( {3, [-4, -1], 5, [0, 1, 2]} ),
Position( {-4, -1, 0, 1, 2} )}
```

It should read as follows to avoid the error:

```
{Legend Position( {3, [-4, -1], 5, [0, 1, 2]} ),
Legend Position( {-4, -1, 0, 1, 2} )}
```

Compatibility Issues in JMP 16.0

Copy File(), Move File(), Rename File(), and Delete File()

In JMP 15.0, copying a file, moving a file, renaming a file, and deleting a file that didn't exist returned 0. Now, these actions throw an error.

Row Selection() and Case Sensitivity

`Match Case`, `Match Case()`, `Match Case(1)` and a missing `Match Case` do a case sensitive match in `Row Selection()`. `Match Case(0)` does a case insensitive match. Previously, `Row Selection()` matched case insensitively if `Match Case` was not supplied.

Target Level Column Property

You can use the new Target Level column property to specify the level of interest in binary logistic regression models. The default level is High. The main effects will be in the slope of the logistic probability curve, the signs of the linear model estimates, the profiler shape, and the ROC target. Be aware that some scripts that assume the target level in Fit Model or Fit Y by X logistic regression models should be checked.

Opening Multiple Data Tables

`Open()` returns a list of data tables if multiple data tables are created (for example, when importing PDF, HDF5, or Excel files). Previously, the `Open()` command returned a single table, even if multiple tables were created in the process.

Format(), Informat(), and Variations

The results of `Format()`, `Informat()` and their variations are written to the log if the format is invalid. Previously, they returned `m/d/y` when the format was invalid.

Graph Background Color Preference in Contour Plots

The Graph Background Color preference no longer affects the Contour Plot legend background color. Previously, the background color was applied to both the graph background and the legend background.

Set Auto Stretching and Get Auto Stretching for Display Boxes are Deprecated

Use `Set Stretch` and `Get Stretch` instead.

Appendix B

Efficient Scripts

Tips for Writing Scripts That Run Faster

This appendix provides ideas for writing scripts that run faster and more efficiently. For example, it is possible to control how much time is allocated to finding an optimal design. You can collapse outlines in a report that are not necessary. Or you can write static matrices, which is faster than developing them dynamically.

Contents

Using Matrices 943
Design of Experiments 944
Scripting Data Tables 944
Scripting Graphs and Display Boxes 945
Scripting Numbers, Strings, Arrays, and Lists 946
Programming Tips 946
Troubleshooting 946

Using Matrices

- The matrix concatenation operators, || and |/, add columns or rows to a matrix dynamically. However, the script runs faster if you define a matrix in a static fashion instead.

```
// faster example
t1 = Tick Seconds();
A = J( 10, 10000, . );
For( i = 1, i <= 10000, i++,
   a2 = J( 10, 1, Random Gamma( 1, 1 ) );
   // define a2 to be a 10 by 1 vector of random gamma values.
        A[0, i] = a2;
       /* populate the ith column of the matrix A to have the
         values in a2.  The first argument, 0, means "all rows". */
);
t2 = Tick Seconds();
Show( t2 - t1 );
   t2 - t1 = 0.0166666666627862;

// slower example
t3 = Tick Seconds();
B = J( 10, 0, . );
For( i = 1, i <= 10000, i++,
   a2 = J( 10, 1, Random Gamma( 1, 1 ) );
   B = B || a2;
);
t4 = Tick Seconds();
Show( t4 - t3 );
   t4 - t3 = 3.1333333333605;
```

- If you know the size of a matrix that you're going to need, it's typically better to allocate the matrix in its entirety first and then populate it. In the example below, the J() function is doing the allocation of the 100000x1 matrix all at the same time.

```
Delete Symbols( a1, a2 );
// faster example
t1 = Tick Seconds();
a1 = J( 100000, 1, Random Uniform() );
t2 = Tick Seconds();
Show( t2 - t1 );
Delete Symbols( a1 );
   t2 - t1 = 0;

// slower example
t3 = Tick Seconds();
a2 = J( 0, 0 );
```

```
For( i = 1, i <= 100000, i++,
   a2 |/= Random Uniform()
);
t4 = Tick Seconds();
Show( t4 - t3 );
   t4 - t3 = 5.05000000004657;
```

- Use matrices rather than a data table to store numbers.

Design of Experiments

In DOE, there are options such as `Number of Starts` and `Design Search Time` that enable you to control how much time is allocated to finding an optimal design.

```
// Number of Starts example
DOE(
   Custom Design,
   Add Factor( Continuous, -1, 1, "X1", 0 ),
   Add Factor( Continuous, -1, 1, "X2", 0 ),
   Add Factor( Continuous, -1, 1, "X3", 0 ),
   Number of Starts( 1000 ),
   Make Design
);
// Design Search Time example
d = DOE( Screening Design );
dt2 = Open( "$SAMPLE_DATA/Design Experiment/Weld Factors.jmp" );
d << Load Factors;
Close( dt2 );
d << {Set Random Seed( 12345 ), Screening Type( 1 ),
Suppress Cotter Designs, Design Search Time( 8 ),
Number of Column Starts( 50 ), Set Sample Size( 12 ), Make Design};
```

Scripting Data Tables

- Displaying a large data table can cause scripts to run slower. To speed things up, you can make a data table invisible, especially if you need to reference numbers in a report that will be used for future calculations or used in another platform. In this case, there isn't a need to see the current data table.

  ```
  dt = Open( "$SAMPLE_DATA/Big Class.jmp", invisible);
  ```

- Drawing objects is slow. Suppose that you have a graph with a lot of data. The platform does complicated calculations or the data table contains a formula. Adding a row can require a lot of processing. Use `Begin Data Update()` so you only redraw the graph once

instead of for each change. `Begin Data Update()` saves all update messages until `End Data Update()` is reached.

```
dt = Open( "MyData.jmp" );
dt << Distribution(
   Column( :"N=1"n, :"N=5"n), :"N=10"n )
);
Wait(1);
dt << Begin Data Update;
dt << Add Rows( 2000 );
dt << End Data Update;
```

- The use of functions might help speed things up, for example, using the `Summarize()` function rather than creating a Summary table. Maintaining a data table can be too much overhead. See "Store Summary Statistics in Global Variables".
- Data tables are great for keeping tabular data.
  - Keep data table strings under 23 characters. Longer strings have additional overhead.
  - Numeric values use less memory than character values.

Scripting Graphs and Display Boxes

- When displaying large numbers in graphs, you can set a preference to use Fast Markers.

  ```
  Preferences( Fast Marker Threshold( 25000 ) ); // 50000 is the default.
  ```

- Collapse some of the outline nodes in a report to speed up the script.

  ```
  SendToReport(
       Dispatch(
          {"Fit Life by X - Hours BY Arrhenius Celsius (Temp) Regression "},
          "Scatterplot",
          OutlineBox,
          {Close( 1 )}
       ),
  ```

- Construct your display boxes outside `New Window()` and then include them by reference in `New Window()`. This method saves the drawing until the end.

  ```
  tb = Text Box( "Select:" );
  lb = List Box( {"One", "Two", "Three"} );
  New Window( "test", tb, lb );
  ```

- The user might not need to see a report if the numbers will be used in another platform or for a future calculation. You can hide the report.

  ```
  dt= Open( "$SAMPLE_DATA/Big Class.jmp" );
  obj = Bivariate( Y( :Weight ), X( :Height ) );
  obj << Show Window( 0 ); // hide the window
  ```

```
obj << Fit Line( {Confid Curves Fit( 1 )} );
```

Submit the following script to see the report window. Notice that the confidence curves were fit even though the window was not shown.

```
obj << Show Window( 1 );
```

Scripting Numbers, Strings, Arrays, and Lists

- Use a faster algorithm. Replacing an O(n^2) algorithm with an O(n) algorithm is faster than anything else. See https://en.wikipedia.org/wiki/Big_O_notation.
- Avoid concatenating long strings. 100s of characters is OK; 1,000,000s of characters will be slow if you build the string in 1,000,000s of operations.
- Use numeric arrays (matrices, not lists).
  - Use numeric arrays with operators that operate on the entire array.
  - Avoid writing loops to manipulate numeric arrays.
- Use associative arrays for lookups.

Programming Tips

- Avoid `Break()`, `Continue()`, and `Return()` functions for speed. Using them might make it easier for you to read the script.
- Functions versus expressions:
  - User-defined functions have overhead for calling them but might make it easier for you to read the script.
  - Expressions have no overhead for calling but might make it more difficult for you to read the script.

Troubleshooting

Use the JSL Debugger to discover where the script is expending the most time. See “Debug or Profile Scripts”.

Appendix C

References

Morgan, J. (2010). "Expression Handling Functions: Part I." *JMPer Cable* 26:15–19. Accessed January 2, 2018. https://www.jmp.com/content/dam/jmp/documents/en/newsletters/jmper-cable/26_winter_2010.pdf.

Appendix D

Glossary

Terms, Concepts, and Placeholders

| In syntax summaries, | means "or" and separates possible choices. Usually choices separated by | are mutually exclusive. In other words, you have to pick one and cannot list several.

argument An argument is something specified inside the parentheses of a JSL function, message, and so forth. Big Class.jmp is the argument in `Open("Big Class.jmp")`.

You can often infer the meaning by the argument's position. For example, the values *200* and *100* in `size(200, 100)` are implicit arguments. The first value is always interpreted as the width; the second value is always interpreted as the height. See also *named argument*.

Boolean A Boolean value is a yes/no value, something that is on or off, shown or hidden, true or false, 1 or 0, yes or no. A value listed as *being a Boolean value* is one that evaluates to true or false (or missing).

col In syntax summaries, a placeholder for any reference to a data table column. For example, `Column("age")`.

command A generic description for a JSL statement that performs an action. The *Scripting Guide* prefers the more specific terms *function* and *message* when they are applicable.

current data table The current data table is the data table that `Current Data Table()` either returns or is assigned.

current row The current row for scripting is defined to be zero (no row) by default. You can set a current row with `Row()` or `For Each Row`, and so forth.

database Although the term is much more general, for JMP's purposes, the word "database" describes any external data source (such as SQL) accessed through ODBC with JSL's `Open Database` command.

Datafeed A Datafeed is a method to read real-time data continuously, such as from a laboratory measurement device connected to a serial port.

db In syntax summaries, a placeholder for any reference to a display box. For example, `report(Bivariate[1])`.

dt In syntax summaries, a placeholder for any reference to a data table. For example, `Current Data Table()` or `Data Table("Big Class.jmp")`.

eliding operator An eliding operator is one that causes arguments on either side to combine and evaluate differently than if the statement were evaluated strictly left to right. For example, `12<a<13` is a range check to test whether *a* is between 12 and 13: JMP reads the whole expression before evaluating. If < did not elide, the expression would be evaluated left to right as `(12<a)<13`. In other words, it would check whether the result of the comparison (1 or 0, for false or true) is below 13, which of course would always yield 1 for true. The << operator (for `object<<message`, which is equivalent to `Send(object, message)` is another example of an eliding operator.

function A function takes an argument or series of arguments inside parentheses after the function name. For example, the infix operator + has a function equivalent `Add()`. The statements `3 + 4` and `Add(3, 4)` are equivalent. All JSL's operators have function equivalents, but not all functions have operator equivalents. For example, `Sqrt(a)` can be represented only by the function. Also see the `Function` operator for storing a function under a name.

global variable A global variable is a name to hold values that exists for the remainder of a session. Globals can contain many types of values, including numbers, strings, lists, or references to objects. They are called globals because they can be referred to almost anywhere, not just in some specific context.

head The head of a JSL expression is the expression without any of its arguments. For example, the head of the expression `Assign( x, 100.1 )` is `Assign()`.

infix operator An infix operator takes one argument on each side, such as + in arithmetic, `3 + 4` , or the = in an assignment, `a=7`.

L-value Something that can be the destination of an assignment. In this manual, L-value describes an expression that normally returns its current value but that can alternatively receive an assignment to set its value. For example, you would ordinarily use a function such as `Row()` to get the current row number and assign it to something else. For example, `x=Row()`. However, since `Row` is an L-value, you can also place it on the left side of an assignment to set its value. For example, `Row()=10`.

list A list is a multiple-item data type entered in special brace { } notation or with the `List` function. Lists enable scripts to work with many things at once, often in the place of a single thing.

matrix A matrix is a JMP data type for a rectangular array of rows and columns of number. In JSL, matrices are entered in bracket `[ ]` notation or with the `Matrix` function.

message A message is a JSL statement that is directed to an *object*, which knows how to execute the message.

metadata In JMP data tables, metadata are data about the data, such as the source of the data, comments about each variable, scripts for working with the data, and so on.

mousedown An event generated by pressing down the mouse button. See "Handle()" and "Mousetrap()".

mouseup An event generated by releasing the mouse button. See "Handle()" and "Mousetrap()".

name A name is a reference to a JSL object. For example, when you assign the numeric value 3 to a global variable in the statement `a=3`, "a" is a name.

namespace A namespace is a collection of unique names and corresponding values. Namespaces are useful for avoiding name collisions between different scripts.

named argument A named argument is an optional argument that you select from a predetermined set and explicitly define. For example, `HexToNumber(hextext, <Base(number)>)` has a single named argument, `Base`. This is the fixed name that must directly be in the `HexToNumber` argument.

ODBC database The Microsoft standard for Open DataBase Connectivity. JSL supports access to any ODBC-enabled data source through the `Open Database` command.

obj In syntax summaries, a placeholder for any reference to an analysis platform. For example, `Bivariate[1]`.

object An object is a dynamic entity in JMP, such as a data table, a data column, a platform results window, a graph, and so forth. Most objects can receive messages telling them to act on themselves in some way.

operator Usually operator refers to a one- or two-character symbol such as + for addition or <= for less than or equal to.

POSIX POSIX is an acronym for Portable Operating System Interface and is a registered trademark of the IEEE. POSIX pathnames enable you to use one syntax for paths for any operating system, instead of having to use a different syntax for each.

postfix operator A postfix operator takes on argument on its left side (before the operator), such as `a++` for postincrement or `a--` for postdecrement.

pre-evaluated statistics Statistics that are calculated once and used as constants thereafter.

prefix operator A prefix operator takes one argument on its right side (after the operator), such as `!a` for negation.

reference A way to address a scriptable object in order to send it messages. For example,. `column("age")` or `Current Data Table()` or `Bivariate[1]`. Typically a reference is stored in a *global variable* for convenience.

row state A data element type to store any combination of the following attributes for data rows: excluded, hidden, labeled, selected, color, marker, hue, shade.

scalar A simple non-matrix numeric value.

scoping operator A scoping operator forces a name to be interpreted as a particular type of data element, for example the `:` operator in `:name` forces *`name`* to be resolved as a column; the `::` operator in `::name` forces *`name`* to be resolved as a global variable. In the rare case that you do want to share specific variables among all opened projects, use the `:::` root operator.

toggle Omitting the Boolean argument for a row state command toggles the setting. If the option is off, the message turns it on. If the option is on, the message turns it off. Sending such a command repeatedly flips back and forth between on and off. If you include the Boolean argument, the command sets an absolute on or off state, and sending the command repeatedly has no further effect. For all other messages, omitting the Boolean argument enables the option.

vector A matrix with only one column or row.

Appendix E

Technology License Notices

- Scintilla is Copyright © 1998–2017 by Neil Hodgson <neilh@scintilla.org>.

  All Rights Reserved.

  Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

  NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved. Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.
- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.
- Made with Natural Earth. Free vector and raster map data @ naturalearthdata.com.
- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr). All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of the WhiteBox nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- iODBC software ia Copyright © 1995–2006, OpenLink Software Inc and Ke Jin (www.iodbc.org). All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  – Neither the name of OpenLink Software Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL OPENLINK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

  3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

- R software is Copyright © 1999–2012, R Foundation for Statistical Computing.
- MATLAB software is Copyright © 1984-2012, The MathWorks, Inc. Protected by U.S. and international patents. See www.mathworks.com/patents. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.
- libopc is Copyright © 2011, Florian Reuter. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and / or other materials provided with the distribution.

  – Neither the name of Florian Reuter nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- libxml2 - Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

  Copyright © 1998–2003 Daniel Veillard. All Rights Reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

- Regarding the decompression algorithm used for UNIX files:

  Copyright © 1985, 1986, 1992, 1993

  The Regents of the University of California. All rights reserved.

  THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group. NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois. All rights reserved. DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS "AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.
- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated. All Rights Reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  – Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  – Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  – Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- dmlc/xgboost is Copyright © 2019 SAS Institute.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

  This file is part of libzip, a library to manipulate ZIP archives. The authors can be contacted at <libzip@nih.at>.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org.

  You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS", WITHOUT WARRANTIES OR CONDITIONS OF

ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- lld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

  Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS”, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

- libcurl is Copyright © 1996–2021, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file. All rights reserved.

  Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

  THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

  Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.