



バージョン 17

オートメーションリファレンス

「真の発見の旅とは、新しい風景を探すことではなく、
新たな視点を持つことである。」
マルセル・プルースト

JMP Statistical Discovery LLC
SAS Campus Drive
Cary, North Carolina 27513-2414

17.1

The correct bibliographic citation for this manual is as follows: JMP Statistical Discovery LLC 2022–2023.
JMP® 17 Automation Reference. Cary, NC: JMP Statistical Discovery LLC

JMP® 17 Automation Reference

Copyright © 2022–2023, JMP Statistical Discovery LLC, Cary, NC, USA

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

JMP Statistical Discovery LLC, SAS Campus Drive, Cary, North Carolina 27513-2414.

October 2022

March 2023

JMP® and all other JMP Statistical Discovery LLC product or service names are registered trademarks or trademarks of JMP Statistical Discovery LLC in the USA and other countries.

® indicates USA registration.

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

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

JMPを最大限に活用する

JMPの初心者でも経験豊かなユーザでも、JMPについての新しい情報をいろいろなリソースから得られます。

JMP.comのサイトでは以下のリソースをご利用いただけます。

- JMPを使い始めるためのライブおよび収録済みのWebcastセミナー
- 新機能や高度なテクニックを紹介している動画デモやWebcastセミナー
- JMPトレーニングの申し込みに関する詳細
- 各地で開催されるセミナーのスケジュール
- お客様のJMP利用事例
- JMP User Community、アドインやスクリプトの例を始めとするユーザ向けリソース、フォーラム、ブログ、イベント情報など

jmp.com/getstarted

目次

外部アプリケーションによる JMP の実行 35

Visual Basic を使った JMP のオートメーション 35

JMP アプリケーションの起動 35

分析の開始 37

データテーブルを作成し、値を埋め込む 38

プログラム例 39

Visual C++ を使った JMP のオートメーション 44

JMP のオートメーションの手順 45

プログラム例 46

Visual C# を使った JMP のオートメーション 47

JMP アプリケーションの起動 47

分析の開始 47

特定のバージョンの JMP を起動 48

Windows で JMP のオートメーションを行うための Application オブジェクトリファレンス 49

定数 49

Bivariate オブジェクトの定数 49

bivarFitTransformConstants 49

bivarOrthogonalFitConstants 49

fitLoessLambdaConstants 49

Chart オブジェクトの定数 50

chartChartTypeConstants 50

chartOrientConstants 50

chartStatConstants 50

Cluster オブジェクトの定数 51

clusterColormapConstants 51

clusterDistanceConstants 51

clusterOrientationConstants 51

列の定数 51

colDataSourceConstants 51

colDataTypeConstants 52

colFormatConstants 52

colModelTypeConstants 52

colReorderConstants 53
colRoleConstants 53
colValidationConstants 53

ContriChart オブジェクトの定数 53

jmpControlChartAlarms 53
jmpControlChartConstants 54
jmpControlChartRules 54

DataTable オブジェクトの定数 54

dtJoinConstants 54
dtSummaryStatConstants 54
summaryStatColNameConstants 55

Discriminant オブジェクトの定数 55

discrimCanonicalOptions 55
discrimScoreOptions 55
discrimPriorsOptions 56

Distribution オブジェクトの定数 56

distributionFitQuantilePlotConstants 56
distributionSaveConstants 56
fitDistribConstants 56

DOECustom オブジェクトの定数 57

doeChangeDifficultyConstants 57
doeFactorTypes 57
doeModelTypes 57
doeOptimalityConstants 57
doeResponseTypes 58

FitModel オブジェクトの定数 58

fitModelDistributionConstants 58
fitModelEffectAttributeConstants 58
fitModelEmphasisConstants 58
fitModelMacroEffectConstants 58
fitModelPersonalityConstants 59
fitModelRandomEffectMethods 59
fitModelRowDiagConstants 59
fitModelSaveColumnConstants 59
fitModelTransforms 60
fitStepDirectionConstants 60
fitStepRulesConstants 60

ItemAnalysis オブジェクトの定数 61

itemAnalysisModelConstants 61

JMPの定数 61

axisBooleanConstants 61
axisIntervalConstants 61
axisLineRefConstants 61
axisNumericOptionConstants 62
axisScaleConstants 62
commFlowControlConstants 62
commParityConstants 62
frameMarkerSizes 62
internetItemTypes 63
jmpColorConstants 63
jmpGraphicsFormats 63
jmpMarkerConstants 64
jmpScriptConstants 64
jmpWindowTypeConstants 64
nomAxisActions 64
printOrientConstants 65

Neuralオブジェクトの定数 65

neuralControlConstants 65

Onewayオブジェクトの定数 65

OnewayCompareConstants 65
OnewayDisplayConstants 65
OnewayNonParConstants 66
OnewaySaveConstants 66

Overlayオブジェクトの定数 66

overlayLineStyleConstants 67
overlayLineThicknessConstants 67

Partitionオブジェクトの定数 67

partitionCriterionConstants 67
partitionDisplayConstants 67
partitionMissingConstants 67
partitionSaveColumnsConstants 68

行の定数 68

rowStateConstants 68
rowSelectWhereHow 68
rowSelectWhereOperations 68

ScatterplotMatrixPlatformオブジェクトの定数 69

scatterMatrixFormatConstants 69

SurfacePlotオブジェクトの定数 69

surfaceColorConstants 69
surfaceDisplayConstants 69

Survival オブジェクトの定数 70
competingCauseConstants 70

TextImport オブジェクトの定数 70
jmpTIEndOfFieldConstants 70
jmpTIEndOfLineConstants 70

TimeSeries オブジェクトの定数 70
timeSeriesConstraintConstants 70
timeSeriesModelConstants 71

Variability オブジェクトの定数 71
varVarianceComponentConstants 71

Application オブジェクト 71

プロパティ 71

Application 71

FullName 71

Name 71

Parent 72

Visible 72

メソッド 72

ClearLog() 72

CloseAllWindows() 72

CloseWindow() As Boolean 72

CloseWindowsOfType(jmpWindowTypeConstants windowType) 72

CreateDOECustom () As DOECustom 72

CreateTextImportObject(FileName As String, NumberColumns as Integer)
As TextImport 72

EnableInteractiveMode(Flag as Boolean) 72

GetLogContents() As String 72

GetJSLValue 73

GetNumberOfAutomationDatatables()As Integer 73

GetRunCommandErrorString() As String 73

GetTableHandleFromIndex(Integer Index) 73

GetTableHandleFromName(Name as String)As DataTable 73

GetTableNameFromIndex(Integer Index) 74

HasRunCommandErrorString() As Boolean 74

HonorSessionSavePref(Flag as Boolean) 74

InternetOpenItem(String URL, internetItemTypes openHow) As DataTable
74

InternetOpenTextAsData(URL As String) As DataTable 74
NewDatabaseObject() As AUTODB 75
NewDataFeed() As DataFeed 75
NewDataTable(FileName As String) As DataTable 75
OpenDocument(FileName As String) As Document 75
Quit() 75
RunCommand(Command As String) 75
RunJSLFile(FileName As String) 75
SetCurrentDirectory (DirName As String) As Boolean 75
ShowLogHonorPreferences() 75
ShowStartupWindow() 76
ShowLog() As Boolean 76

AUTODB オブジェクト 76

メソッド 76

Connect(ConnectInformation As String) 76
Disconnect() As Boolean 76
ExecuteSql(SQLStatement As String) As Boolean 76
ExecuteSqlSelect(SQLSelectStatement As String) As DataTable 76
GetLastError() 76
OpenTable(TableName As String) As DataTable 77
SaveTable(TablePtr As DataTable, FileName As String) 77

グラフ軸の共通メソッド 77

メソッド 77

AxisBoxAddLabel(Handle As Long, Label As String)As Boolean 77
AxisBoxAddRefLine(Handle As Long, Location As Double, Style As Short, Color As Short)As Boolean 77
AxisBoxBooleanOptions(Handle As Long, Option As Short, Flag As Bool)As Boolean 77
AxisBoxFormat(Handle As Long, Format As Short)As Boolean 77
AxisBoxInterval(Handle As Long, Interval As Short)As Boolean 77
AxisBoxNumberDecimals(Handle As Long, NumDecimals As Short)As Boolean 77
AxisBoxNumericOption(Handle As Long, Option As Short, Number As Double)As Boolean 78
AxisBoxRemoveLabel(Handle As Long)As Boolean 78
AxisBoxRevertAxis(Handle As Long)As Boolean 78
AxisBoxScale(Handle As Long, Scale As Short)As Boolean 78

Column オブジェクト 78

プロパティ 78

DataType 78

FieldWidth 78
InputFormat 78
OutputFormat 79
Locked 79
ModelType 79
Name 79
NumberRows 79
NumDecPlaces 79

メソッド 79

AddFormula(JSLText As String) 79
AddValueLabelToList(Value as String, Label as String) As Boolean 79
CommitValueLabels() As Boolean 80
Exclude() As Boolean 81
GetCellVal(RowNumber As Integer) As String 81
GetDataSource() As Integer 81
GetDataVector() As Variant 81
GetFormula() As String 81
GetRowStateVectorData 81
GetValidation() As Integer 82
InsertDataVector(Data As Variant, AfterRow As Long) As Boolean 82
Hide() As Boolean 82
Label() As Boolean 82
RemoveValueLabels() As Boolean 82
ScrollLock() As Boolean 82
SelectCellMissing(Index as Integer) 83
SelectColumn(Flag as Boolean) As Boolean 83
SetCellVal(RowNumber As Integer, Value as String) 83
SetDataVector(Data As Variant) As Boolean 83
SetCellMissing(Row As Integer) 83
SetCurrencyType(Type As ColCurrencyConstants) 83
SetRole(RoleType As Integer) As Boolean 83

プラットフォーム共通のメソッド 83

メソッド 83

CreateJournal() As Journal 83
CopyGraphicItem(Handle As Long) As Boolean 83
DisplayBoxAppend(SrcHandle as Long, AppendHandle as Long) As Long 84
DisplayBoxPrepend(SrcHandle as Long, AppendHandle as Long) As Long 84
FrameBoxAddGraphicsScript(long handle, script As String) As Boolean 84
FrameBoxSetBackColor(long handle, jmpColorConstants color) As Boolean 84

FrameBoxSetMarkerSize(Handle, frameMarkerSizes size) As Boolean 84
FrameBoxTransparency(alpha as Double) As Boolean 84
GetGraphicItemByName(ItemName as String) As Long 84
GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long 84
GetSubgraphicItemByName(Handle as Long, Name as String) As Long 84
GetSubgraphicItemByType(Handle as Long, BoxName as String, BoxNumber as Short) As Long 85
GetTextOfGraphicItem(Handle As Long) As String 85
JournalGraphicItem(Handle As Long) As Boolean 85
JournalOutput() As Boolean 85
Launch() As Boolean 85
LaunchAdd__(ColumnName As String) As Boolean 85
LaunchAddBy() As Boolean 85
LaunchRemove__(ColumnName As String) As Boolean 85
OutlineBoxGetTitle(Handle as Long) As String 85
NumberColGetHeading(Handle As Integer) 86
NumberColGetItemText(Handle As Long, ElementNumber As Integer) As String 86
NumberColHide, StringColHide(Handle As Integer, Flag As Boolean) 86
NumberColSetHeading, StringColSetHeading(Handle As Integer, Title As String) 86
OutlineBoxSetTitle(Handle As Long, Title As String) 86
PrintPages(From As Integer, To As Integer) As Boolean 86
PrintReport() As Boolean 86
SaveGraphicItem(Handle As Long, FileName As String, GraphicType As Integer) 86
SaveGraphicOutputAs(FileName As String, GraphicFormat As Integer) 86
SaveJournalAs(FileName As String) As Boolean 86
ScriptAction(JSLText As String) 86
SetFrameSize(X As Integer, Y As Integer) 87
SetPrintOrientation(printOrientConstants orientation) As Boolean 87
SetWindowPos(X As Integer, Y As Integer) 87
SetWindowSize(CX As Integer, CY As Integer) 87
StringColGetHeading(Handle As Integer) 87
StringColGetItemText(Handle As Long, ElementNumber As Integer) As String 87
TableBoxMakeDataTable(Handle As Long) As DataTable 87
UseByOutput(WithTitle As String) As Boolean 87

DataFeed オブジェクト 87

メソッド 88

Close() As Boolean 88
Connect(PortName As String) As Boolean 88
Disconnect() As Boolean 88
GetLine() As String 88
SetCommParms(BSTR szCommPort, long baudrate, short parity, short databits, short stopbits, short flow) As Boolean 88

DataTable オブジェクト 88

プロパティ 88

Document 88
NumberColumns 89
NumberRows 89
Visible 89

メソッド 89

Activate() As Boolean 89
AddColumns(Prefix as String, NumToAdd As Integer, Where As Integer, Type As Integer, FieldWidth As Integer) As Integer 89
AddNumericTableVar(Name As String, Value As Double) 89
AddRows(NumberToAdd As Integer, AddAfter As Integer) 89
AddRowsHuge(NumberOfRows as Integer, AddAfterRow as Integer) As Integer 90
SummaryUnlinked() As Datatable 90
AddStringTableVar(Name As String, Value As String) 90
AddToConcatList(ColumnName As String) As Boolean 90
AddToJoinList(ColumnName As String) 90
AddToJoinMatchList(ColumnName As String) As Boolean 90
AddToSortList(ColumnName As String, Ascending As Boolean) As Boolean 90
AddToSplitGroupList(ColumnName As String) As Boolean 90
AddToSplitList(ColumnName As String) As Boolean 90
AddToStackList(ColumnName As String) As Boolean 91
AddToSubList(ColumnName As String) As Boolean 91
AddToSummaryGroup(ColumnName As String) As Boolean 91
AddToSummaryStatList(Stat As Integer) 91
AddToSummarySubGroup(ColumnName As String) As Boolean 91
AddToTransposeList(Name as String) As Boolean 91
AddToTransposeByList(Name as String) As Boolean 91
AddToUpdateMatchList(ColumnName as String) As Boolean 91
ClearRowsSelection() 91
ClearSelectedRowStates() As Boolean 91
ColorByColumn(Name as String) As Boolean 91

Concatenate() As DataTable 92
DeleteColumn(ColumnName As String) 92
DeleteSelectedRows() As Boolean 92
Document() As Document 92
CheckRowState(Index As Integer, rowStateConstants stateToCheck) As Boolean 92
EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer 92
EnumRowStatesGetNextRow() As Integer 92
EnumRowStatesGetRowByIndex(Index as Integer) As Integer 93
ExcludeSelectedRows() As Boolean 93
GetChangedRowStateVector(RowStateToCheck As RowStateConstants) 93
GetColumn(ColumnName As String) As Column 93
GetColumnByIndex(Index As Integer) As Column 93
GetColumnName(Index As Integer) As String 93
GetJSLFunctionErrorString As String 93
GetNumberOfRowsByRowState(rowStateConstants stateToCheck) As Long 94
GetRowStatesChanged() As Boolean 94
GetRowStateVector 94
HasJSLFunctionErrorString As Boolean 94
HideSelectedRows() As Boolean 95
Join(DataTable2 As DataTable, JoinType As Integer, OutputTableName As String) As DataTable 95
LabelSelectedRows() As Boolean 95
MarkerByColumn(Name as String) As Boolean 95
NewColumn(Name As String, Type As Integer, Model As Integer, Width As Integer) 95
PrintTable() As Boolean 95
ReorderColumns(ReorderType As Integer) 95
RowStateBeingMonitoring 95
SelectColumn(Column As String, SelectFlag as Boolean) As Boolean 96
SelectExcludedRows() As Boolean 96
SelectHiddenRows() As Boolean 96
SelectLabeledRows() As Boolean 96
SelectAllMatchingCells() As Boolean 96
SelectMatchingCells() As Boolean 96
SelectRandomly(SampleRate As Long) As Boolean 96
SelectRows(StartRow As Integer, EndRow As Integer) 96
SelectRowsWhere(ColumnName As String, Operation As Integer, SelectHow As Integer, Comparative As String) As Boolean 96
SetJoinMatchOptions(DropMultiples As Boolean, IncludeNonMatches As Boolean) 97

SetJoinMergeColumns(Boolean) 97
SetJoinOptions(UpdateFirstTable As Boolean, CopyFormulas As Boolean,
SuppressFormulaEval As Boolean) 97
SetStackMultipleSeriesN(short N) As Boolean 97
SetTransposeOptions(OutputTableName as String, UseSelectedRows as
Boolean) As Boolean 97
SetWindowPos(X As Integer, Y As Integer) 97
SetWindowSize(CX As Integer, CY As Integer) 97
Sort(Replace As Boolean) As DataTable 98
Split(ColumnID As String, OutputTableName As String,
KeepRemainingCols As Boolean) 98
Stack(idColumnName As String, stackedColumnName As String, TableName
As String) As DataTable 98
Subset() As DataTable 98
SubsetSetRandomSelection(SampleRateOrSize as Double, Shuffle As
Boolean) As Boolean 98
SubsetStratifyAddColumn(Column As String) As Boolean 98
Summary() As DataTable 98
SummarySetStatColumnFormat(summaryStatColNameConstants format) 98
Transpose() As DataTable 99
UpdateTable(DataTable2 as DataTable, IgnoreMissingValues As Boolean)
99

Document オブジェクト 99

プロパティ 99

Application 99
AutoSave 99
FullName 99
Name 99
Path 99
Saved 99
Visible 100

メソッド 100

Activate() 100
Close(SaveChanges as Boolean, FileName As String) 100
CopyToClipboard() 100
CreateBivariate() 100
CreateCluster() 100
CreateNormalMixtures() As NormalMixtures 100
CreatePlatform() 100
Save() 101
SaveAs(FileName As String) 101

Journal オブジェクト 101

メソッド 101

```
GetActiveJournal() As Journal 101
SaveAsHTML(BSTR filename, jmpGraphicsFormats graphicType) As Boolean
101
SaveAsJournal(BSTR filename) As Boolean 101
SaveAsRTF(BSTR filename, jmpGraphicsFormats graphicType) As Boolean
101
SaveAsMSWordDoc(Filename As String) As Boolean 101
```

Text Import オブジェクト 102

メソッド 102

```
ColumnNamesStart(StartLine as Integer) 102
DataStarts(StartLine As Integer) 102
FirstLineIsData(Flag As Boolean) 102
OpenFile() As Document 102
SetColumnType(ColumnNumber As Integer, Type As Integer) As Boolean 102
SetEndOfFieldOptions(Options As Integer) 102
SetEndOfLineOptions(Options As Integer) 103
StripQuotes(Flag As Boolean) 103
```

プラットフォームのメソッド 103

Attribute Chart オブジェクトのメソッド 103

```
EffectivenessReport(Flag As Boolean) 103
```

Bivariate オブジェクトのメソッド 103

```
DensityEllipses(Degree As Double) 103
FitEachValue As Fit 103
FitLine As Fit 103
FitLoess() As Fit 103
FitLoessWeightConstants(fitLoessWeightTricube, fitLoessWeightCosine,
fitLoessWeightEpanechnikov, fitLoessWeightGaussian,
fitLoessWeightCauchy) 104
FitLoessWithParms(fitLoessLambdaConstants Lambda, Alpha as Double,
Robustness as Short) 104
FitMean As Fit 104
FitOrthogonal(OrthogonalFitConstant as Integer, VarianceRatio As
Double) As Fit 104
FitPolynomial(Degree As Double) As Fit 104
FitRobust, FitCauchy(Flag As Boolean) As Fit 104
FitSpline(Degree As Double) As Fit 104
FitTransformed(Xtransform As Integer, Ytransform as Integer,
PolynomialDegree as Integer) 104
```

FitTransformedWithOptions(Xtransform As Integer, Ytransform as Integer, PolynomialDegree as Integer, CenteredPolynomial as Boolean, ConstrainIntercept as Boolean, InterceptValue as Double, ConstrainSlope As Boolean, SlopeValue as Double) As Fit 104
GroupBy(ColumnName As String) As Boolean 105
HistogramBorders(Flag as Boolean) 105
KernelSmoother(Lambda As JMP.fitLoessLambdaConstants, Weight As JMP.fitLoessWeightConstants, Alpha As Double, Robustness As Short) 105
NonParDensity() As FitDensity 105
ShowPoints(Flag as Boolean) 105

Bubble Plotオブジェクトのメソッド 105

AggregateSizeAsSum(Flag As Boolean) 106
AggregateXAsSum(Flag As Boolean) 106
AggregateYAsSum(Flag As Boolean) 106
AllLabels(Flag As Boolean) 106
BubbleSize(Size as Double) 106
BubbleSpeed(Speed as Double) 106
BubbleTimeIndex(Index as Double) 106
CombineAll() 106
Filled(Flag As Boolean) 106
Go() 106
LaunchAddColoring(Name as BSTR) 106
LaunchAddID(Name as BSTR) 106
LaunchAddSizes(Name as BSTR) 107
LaunchAddTime(Name as BSTR) 107
Prev() 107
SelectableAcrossGaps(Flag As Boolean) 107
SplitAll() 107
Step() 107
Stop() 107
Trails(Flag As Boolean) 107

Categoricalオブジェクトのメソッド 107

AgreementStatistic(Flag as Boolean) As Boolean 107
CrosstabFormat(Flag as Boolean) As Boolean 107
CrosstabTransposed(Flag as Boolean) As Boolean 108
Frequencies(Flag as Boolean) As Boolean 108
FrequencyChart(Flag as Boolean) As Boolean 108
LaunchAddResponseRole(ResponseType as jmpCategoricalResponseRoles) As Boolean 108
LaunchAddToResponseList(ColumnName as String) As Boolean 108

Legend(Flag as Boolean) As Boolean 108
RatePerCase(Flag as Boolean) As Boolean 108
ShareChart(Flag as Boolean) As Boolean 108
ShareOfResponses(Flag as Boolean) As Boolean 108
TableFormat(Flag as Boolean) As Boolean 109
TableTransposed(Flag as Boolean) As Boolean 109
TestEachResponse(Flag as Boolean) As Boolean 109
TestResponseHomogeneity(Flag as Boolean) As Boolean 109
TransitionReport(Flag as Boolean) As Boolean 109

Cell Plot オブジェクトのメソッド 109

LaunchOptions(BOOL Scale, BOOL Center) 109
Legend(Flag As Boolean) 109

Chart オブジェクトのメソッド 110

ConnectPoints(Flag As Boolean) 110
LaunchAddY(ColumnName As String, Statistic as Short) 110
Orientation(WhichWay As Short) 110
Overlay(Flag As Boolean) 110
OverlayColor(Color As Short) 110
SeparateAxes(Flag As Boolean) 110
ShowPoints(Flag As Boolean) 110
SpecifyQuantilesVal(Quantiles as Double) As Boolean 110
SpecifyType(ChartType as Short) 110

Cluster オブジェクトのメソッド 111

ClusterCriterion, ClusterSummary, ConstellationPlot (Flag As Boolean) 111
ColorClusters(Flag As Boolean) 111
KMNORMALMixtures(Flag as Boolean) 111
KMPARALLELCoordPlots(Flag as Boolean) As Boolean 111
KMSOMBandwidth(Bandwidth As Double) 111
LaunchSpecifyDistanceFormula(FormulaType As Integer) 111
LaunchSpecifyKMeans(Flag As Boolean) 111
Legend(Flag As Boolean) 112
MarkClusters(Flag As Boolean) 112
NumberOfClusters(Number As Integer) 112
ParallelCoordPlots, ScatterPlotMatrix 112
SaveClusters() 112

Contingency オブジェクトのメソッド 112

Cochran(ColumnName As String) As Boolean 112
Correspondence(Flag As Boolean) 112
Crosstabs(Flag As Boolean) As Crosstabs 112

HorizontalMosaic(Flag as Boolean) 112
MosaicPlot(Flag As Boolean) 112
NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)
113
Tests(Flag As Boolean) 113

Contourオブジェクトのメソッド 113
FillAreas(Flag As Boolean) 113
GenerateGrid(HorizontalSize As Integer, VerticalSize As Integer) As
DataTable 113
LabelContours(Flag As Boolean) 113
ReverseColors(Flag As Boolean) 113
SaveContours() As DataTable 113
SaveTriangulation() As DataTable 113
ShowBoundary(Flag As Boolean) 113
ShowContours(Flag As Boolean) 113
ShowDataPoints(Flag As Boolean) 114
ShowTriangulation(Flag As Boolean) 114

ContourProfilerオブジェクトのメソッド 114
ContourGrid(Low As Double, High As Double, IncrementAs Double) 114
ContourGridWithResponse(low as Double, high as Double, increment as
Double, responseColumn As String) As Boolean 114
SurfacePlot(Flag As Boolean) 114

ControlChartオブジェクトのメソッド 114
BoxChart(Flag As Boolean) 114
CenterColor(Color As Integer) 114
ConnectColor(Color As Integer) 114
ConnectPoints(Flag As Boolean) 114
ConnectThroughMissing(Flag As Boolean) 114
ControlLimits(Flag As Boolean) 115
LaunchAddPhase, LaunchRemovePhase(ColumnName As String) 115
LaunchAddProcess(ColumnName As String) As Boolean 115
LaunchAddSampleLabel(ColumnName As String) As Boolean 115
LaunchAddSampleUnitSize(ColumnName As String) As Boolean 115
LaunchSetChartType(ChartType As Integer) 115
LaunchSetConstantSampleSize(Flag As Boolean, SampleSize As Integer)
115
LaunchSetCStats(various parms as double) As Boolean 115
LaunchSetCusumOptions(TwoSided As Boolean, DataUnits As Boolean) 115
LaunchSetCusumStats(various parms as double) As Boolean 115
LaunchSetEWMAStats(various parms as double) As Boolean 115
LaunchSetEWMAWeight(Weight As Double) 115

LaunchSetIRChartParms(IndMeas As Boolean, MovingRange As Boolean,
Range As Integer) 116
LaunchSetIRStats(various parms as double) As Boolean 116
LaunchSetIRSummarizeParms(PreSummarize As Boolean, Mean As Boolean,
StdDev As Boolean) 116
LaunchSetKSigmaAlphaH(KSigma As Boolean, alpha As Boolean, H As
Boolean, value As Double, beta As Double) 116
LaunchSetNPStats(various parms as double) As Boolean 116
LaunchSetPresummarizeChartTypes(VARIANT_BOOL IndivGroupMeans,
VARIANT_BOOL IndivGroupStdDev, VARIANT_BOOL MovingRangeGroupMeans,
VARIANT_BOOL MovingRangeStdDev) As Boolean 116
LaunchSetPresummarizeStats(double sigma, double meanMeasureGroup,
double meanMeasureStdDev, double meanMovingGroup, double
meanMovingStdDev) As Boolean 116
LaunchSetPStats(various parms as double) As Boolean 116
LaunchSetUStats(various parms as double) As Boolean 117
LaunchSetUWMAMovingAvg(Average As Double) 117
LaunchSetUWMAStats(various parms as double) As Boolean 117
LaunchSetVariableChartParms(Xbar As Boolean, R As Boolean, S As
Boolean) 117
LaunchSetVariableStats(various parms as double) As Boolean 117
Needles(Flag As Boolean) 117
SaveLimits() As Datatable 117
SetAlarm(jmpControlChartAlarms alarmType) As Boolean 117
SetCustomAlarmText(BOOL Speak, BSTR text) As Boolean 117
SetActiveChart(chartNumber as Integer) As Boolean 118
ShowCenter(Flag As Boolean) 118
ShowLineLegend(Flag As Boolean) 118
ShowPoints(Flag As Boolean) 118
ShowZones(Flag As Boolean) 118
Test(TestNumber As Integer, Flag As Boolean) 118
TestsAll(Flag As Boolean) 118
WestgardRule(jmpControlChartRules ruleNumber, VARIANT_BOOL flag) As
Boolean 118

Crosstabs オブジェクトのメソッド 118

CellChiSquare(Flag As Boolean) 118
Col(Flag As Boolean) 118
Count(Flag As Boolean) 119
Deviation(Flag As Boolean) 119
Expected(Flag As Boolean) 119
Row(Flag As Boolean) 119
Total(Flag As Boolean) 119

Diagram オブジェクト 119

メソッド 119

Discriminant オブジェクトのメソッド 119

CanonicalOptions(discrimScoreOptions option, Flag As Boolean) As Boolean 119
SaveDiscrimMatrices 119
ScatterplotMatrix() 119
ScoreData(Flag As Boolean) As Boolean 119
ScoreOptions(discrimScoreOptions option, Flag As Boolean) As Boolean 120
ScoreSelectUncertainRows(Value As Double) As Boolean 120
ShowCanonicalPlot(Flag As Boolean) As Boolean 120
ShowGroupMeans(Flag As Boolean) As Boolean 120
ShowWithinCovariances(Flag As Boolean) As Boolean 120
SpecifyPriors(discrimPriorsOptions option) 120
StepwiseSetup 120

DistribFit オブジェクトのメソッド 120

DensityCurve(Flag As Boolean) 120
GoodnessOffFit(Flag As Boolean) 120
QuantilePlot(Flag As Boolean) 120
QuantilePlotAction(distributionFitQuantilePlotConstants action, VARIANT_BOOL flag) As Boolean 121
Quantiles(UpperLimit As Double, LowerLimit As Double, Target As Double) 121
LabelCumPoints(Flag As Boolean) 121
RemoveFit() 121
SaveDensityFormula() 121
SaveFittedQuantiles() 121
SpecLimits(lower as Double, upper as Double, target as Double) 121

Distribution オブジェクトのメソッド 121

BetaBinomialFit(Sample Size as Integer, Sample Column as String) As Fit 121
BinomialFit(Sample Size as Integer, Sample Column as String) As Fit 121
CapabilityAnalysis(LowerLimit As Double, UpperLimit As Double, Target As Double, Sigma As Double) 122
CDFPlot(Flag As Boolean) 122
ConfidenceInterval(Alpha As Double) 122
CountAxis(Flag As Boolean) 122
DensityAxis(Flag As Boolean) 122
ErrorBars(Flag As Boolean) 122

FitDistribution(FitType As Integer) As FitDistribution 122
FitNormalMixtures(NumberOfClusters as Integer) As FitDistribution 122
Histogram(Flag As Boolean) 122
HorizontalLayout(Flag As Boolean) 122
Moments(Flag As Boolean) 122
MoreMoments(Flag As Boolean) 123
MosaicPlot(Flag As Boolean) 123
NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)
123
NormalQuantilePlot(Flag As Boolean) 123
OutlierBoxPlot(Flag As Boolean) 123
PredictionInterval(alpha as Double, nSamples as Long) 123
ProbAxis(Flag As Boolean) 123
QuantileBoxPlot(Flag As Boolean) 123
Quantiles(Flag As Boolean) 123
Save(Action As Integer) 123
SetQuantileIncrement(Increment As Double) 123
ShowCounts(flag as Boolean) 124
ShowPercents(flag as Boolean) 124
StemAndLeaf(Flag As Boolean) 124
TestMean(meanToTest As Double, Sigma As Double, Wilcoxon As Boolean)
124
TestMeanWithOptions(meanToTest As Double, Sigma As Double, Wilcoxon
As Boolean, PValue As Boolean, Power As Boolean) 124
TestStdDev(stdDeviation As Double) 124
ToleranceInterval(Alpha as double, Proportion as double) 124

DOEオブジェクトのメソッド 124

AddBlockingFactor(NumberOfRuns As Long) As Boolean 124
AddCategoricalFactorWithLevelNames(FactorName as String, LevelNames
as Variant Array of Strings) As Boolean 125
AddBlockingFactorWithName(FactorName As String, NumberOfRuns As Long)
As Boolean 125
AddCategoricalFactor(NumberOfLevels as Long) As Boolean 125
AddCategoricalFactorWithName(FactorName As String, NumberOfLevels as
Long) As Boolean 125
AddContinuousFactorWithBounds(LowerBound As Double, UpperBound As
Double) As Boolean 125
AddContinuousFactorWithName(FactorName as String, LowerBound As
Double, UpperBound As Double) As Boolean 125
AddFactor(factorType As doeFactorType) 125
AddMixtureFactorWithBounds(LowerBound As Double, UpperBound As
Double) As Boolean 125

AddMixtureFactorWithName(FactorName as String, LowerBound As Double, UpperBound As Double) As Boolean 125
AddResponse(ResponseType as doeResponseTypes, Name as String, LowerLimit As Double, UpperLimit As Double, Importance As Double) As Boolean 126
AddTerms(Terms as Variant Array) As Boolean 126
AddTermsWithPowers(Terms as Variant Array, Powers as Variant Array) As Boolean 126
LoadResponses(Table as DataTable) As Boolean 126
LoadFactors(Table as DataTable) As Boolean 127
LoadConstraints(Table as DataTable) As Boolean 127
MakeDesign() 127
MakeModel(ModelType As doeModelTypes) As Boolean 127
MakeTable() As Boolean 127
NumberOfCenterpoints(nCenterpoints As Long) As Boolean 127
NumberOfReplicates(nReplicates as Long) As Boolean 128
NumberOfStarts(nStarts As Long) As Boolean 128
OptimalityCriterion(Criterion as doeOptimalityConstants) As Boolean 128
SaveFactors() 128
SaveXMatrix() 128
SetRandomSeed(Seed As Double) As Boolean 128
ShowDiagnostics() 128
SimulateResponses() 128
SpecifyChangeDifficulty(doeChangeDifficultyConstants difficulty) 128
SphereRadius(Radius as Double) 129
Fitオブジェクトのメソッド 129
ConfidenceFit(Flag As Boolean) 129
ConfidenceIndividual(Flag As Boolean) 129
LineOfFit(Flag As Boolean) 129
PlotResiduals(Flag As Boolean) 129
RemoveFit() 129
SavePredicteds() 129
SaveResiduals() 129
SetAlpha(Alpha As Double) 129
SplineSaveCoeffs() 130
SplineSavePredFormula() As DataTable 130
FitDensityオブジェクトのメソッド 130
FivePercentContours(Flag As Boolean) 130
KernelControl(Flag As Boolean) 130
MeshPlot(Flag As Boolean) 130

ModalClustering(Flag As Boolean) 130
SaveDensityGrid() As DataTable 130

FitLeastSquares オブジェクトのメソッド 130
ContourProfiler(Flag As Boolean) As Boolean 131
CubePlot(Flag As Boolean) As Boolean 131
GetResponse(Name As String) As FitResponse 131
Profiler(Flag As Boolean) As FitProfiler 131

FitLogvariance オブジェクトのメソッド 131
ConfidenceInterval(Alpha As Double) As Boolean 131
LikelihoodRatio(Flag As Boolean) As Boolean 131
MarginalVariances (Flag As Boolean) As Boolean 131

FitManova オブジェクトのメソッド 131
SaveDiscrim() As Boolean 131
SavePredicted() As Boolean 131
SaveResiduals() As Boolean 132

FitModel オブジェクトのメソッド 132
Launch() As Object 132
LaunchAddCrossEffect() As Boolean 132
LaunchAddMacroEffect(fitModelMacroEffectConstants macroType) As Boolean 132
LaunchAddNestEffect() As Boolean 133
LaunchAddToEffectList(Name As String) As Boolean 133
LaunchAddXEffect() As Boolean 133
LaunchAddXEffectWithTransform(transform as fitModelTransforms) 133
LaunchAddYWithTransform(ColumnName As String, transform as fitModelTransforms) As Boolean 133
LaunchGetEffectName(EffectNumber As Integer) As String 133
LaunchGetNumberOfEffects() As Integer 134
LaunchRemoveFromEffectList(Name As String) As Boolean 134
LaunchRemoveSelectedEffects() As Boolean 134
LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean 134
LaunchSpecifyAttributesForSelectedEffects(fitModelEffectAttributeConstants attribNumber) As Boolean 134
LaunchSpecifyDistribution(fitModelDistributionConstants) As Boolean 135
LaunchSpecifyEmphasis(fitModelEmphasisConstants emphasis) As Boolean 135
LaunchSpecifyIntercept(Flag As Boolean) 135
LaunchSpecifyPersonality(fitModelPersonalityConstants personality) As Boolean 135
LaunchSpecifyRandomEffectMethod(method as fitModelRandomEffectMethods) As Boolean 135

UseByFit(Name As String) As Fit 135

FitNominal オブジェクトのメソッド 136

InversePrediction() As Boolean 136

LikelihoodRatioTests(Flag As Boolean) As Boolean 136

OddsRatios(Flag As Boolean) As Boolean 136

Profiler(Flag As Boolean) 136

ROCCurve(Flag As Boolean) As Boolean 136

SaveProbFormula() As Boolean 136

FitOrdinal オブジェクトのメソッド 136

ConfidenceIntervals(Double As Alpha) As Boolean 136

LikelihoodRatioTests(Flag As Boolean) As Boolean 137

SaveExpectedValue() As Boolean 137

SaveProbFormula() As Boolean 137

SaveQuantiles() As Boolean 137

FitParametricSurvival オブジェクトのメソッド 137

ConfidenceIntervals(Flag As Boolean) As Boolean 137

CorrelationOfEstimates(Flag As Boolean) As Boolean 137

CovarianceOfEstimates(Flag As Boolean) 137

EstimateSurvivalProbability() 137

EstimateTimeQuantile() 137

LikelihoodRatioTests(Flag As Boolean) As Boolean 137

FitProfiler オブジェクトのメソッド 137

InteractionProfiler(Flag As Boolean) 137

FitProportional オブジェクトのメソッド 138

メソッド 138

FitResponse オブジェクトのメソッド 138

BoxCoxY(Flag As Boolean) As Boolean 138

CorrelationOfEstimates(Flag As Boolean) As Boolean 138

ExpandedEstimates(Flag As Boolean) As Boolean 138

GetEffectAnalysis(Name As BSTR) As FitEffect 138

InteractionPlots(Flag As Boolean) As Boolean 138

LSMeansPlot(Flag As Boolean) As Boolean 138

LSMeansStudents(Flag As Boolean) As Boolean 139

LSMeansTable(Flag As Boolean) As Boolean 139

LSMeansTukey(Flag As Boolean) As Boolean 139

NormalPlot(Flag As Boolean) As Boolean 139

ParameterPower(Flag As Boolean) As Boolean 139

ParetoPlot(Flag As Boolean) As Boolean 139

RowDiagnostics(fitModelRowDiagConstants diagType, VARIANT_BOOL Flag) As Boolean 139
SaveColumns(fitModelSaveColumnConstants saveType) As Boolean 139
ScaledEstimates(Flag As Boolean) As Boolean 139
SequentialTests(Flag As Boolean) As Boolean 139
TestSlices() As Boolean 139

FitStepwise オブジェクトのメソッド 140

AllPossibleModels() 140
AllPossibleModelsWithParameters(NMaximumTerms As Integer, NBestModelsToSee As Integer, HeredityRestriction As Boolean) 140
EnterAll() As Boolean 140
EnterEffect(EffectNumber As Integer, Flag As Boolean) As Boolean 140
GetEffectName(EffectNumber As Integer) As String 140
GetNumberOfEffects() As Short 140
Go() As Boolean 140
LockEffect(EffectNumber As Integer, Flag As Boolean) As Boolean 140
RemoveAll() As Boolean 140
SetDirection(fitStepDirectionConstants Direction) 141
SetProbToEnter(Value As Double) As Boolean 141
SetProbToLeave(Value As Double) As Boolean 141
SetRules(fitStepRulesConstants Rules) 141
Step() As Boolean 141
Stop() As Boolean 141

Gaussian Process メソッド 141

ContourProfiler(Flag as Boolean) 141
LaunchEstimateNuggetParameter(Flag as Boolean) As Boolean 141
LaunchSpecifyCorrelationType(Type as jmpGaussianCorrelationConstant) As Boolean 141
LaunchSpecifyMinimumTheta(Theta as Double) As Boolean 141
Profiler(Flag as Boolean) 142
SaveJackknifePredictedValues() 142
SavePredictionFormula() 142
SaveVarianceFormula() 142
SurfaceProfiler(Flag as Boolean) 142

Cluster オブジェクトの階層型クラスター固有メソッド 142

ColorMap(clusterColorMapConstants mapType) As Boolean 142
DistanceGraph(Flag As Boolean) As Boolean 142
GeometricXScale(Flag As Boolean) As Boolean 142
LaunchAddLabel.ColumnName As String) As Boolean 142
LaunchAddOrdering.ColumnName As String) As Boolean 142

LaunchRemoveLabel(ColumnName As String) As Boolean 142
LaunchRemoveOrdering(ColumnName As String) As Boolean 142
SaveClusterHierarchy() As Boolean 143
SaveDisplayOrder() 143
SetOrientation(clusterOrientationConstants orientation) As Boolean 143
StandardizeData(Flag As Boolean) 143
TwoWayClustering 143

ItemAnalysis オブジェクトのメソッド 143
LaunchSpecifyModel(Model as itemAnalysisModelConstants) 143
NumberOfPlotsAcross(Number as Integer) 143
SaveAbilityFormula() 143

Cluster オブジェクトの k-means クラスター固有メソッド 143
KMGo() 143
KMSaveMixtureFormulas() 144
KMSaveMixtureProbs() 144
KMSeedWithSelectedRows() As Boolean 144
KMSimulateMixtures(long numberOfRows) 144
KMShiftDistances(Flag As Boolean) 144
KMStep() 144
KMWithinClusterStdDev(Flag As Boolean) 144
LaunchAddFreq(ColumnName As String) As Boolean 144
LaunchAddWeight(ColumnName As String) As Boolean 144
LaunchRemoveFreq(ColumnName As String) As Boolean 144
LaunchRemoveWeight(ColumnName As String) As Boolean 144
StandardizeData (Flag As Boolean) 145

Logistic オブジェクトのメソッド 145
InversePrediction() 145
LiftCurve(Flag as Boolean) 145
LineColor(Color as jmpColorConstants) 145
LogisticPlot(Flag As Boolean) 145
NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean) 145
RateCurve() 145
ROCCurve(Flag As Boolean) 145
ROCSetPositiveLevel(LevelValue As String) As Boolean 145

MatchedPairs オブジェクトのメソッド 146
SignTest(Flag As Boolean) 146
SetAlphaLevel(Alpha As Double) 146
WilcoxonSignedRank(Flag As Boolean) 146

測定システム分析 (MSA) 146

BiasStudySetAlpha(alpha as Double) As Boolean 146
BiasStudySetChartOptions(option as MSAStudyChartOptions) As Boolean 146
LaunchSpecifyAnalysisSettings(maxIterations as Integer, convergenceLimit as Double) As Boolean 146
LaunchSpecifyAlpha(Alpha as Double) As Boolean 146
LaunchSpecifyChartDispersionOptions(option as MSAChartDispersionTypes) As Boolean 146
LaunchSpecifyModelOptions(option as MSAModelTypes) As Boolean 146
RangeChartOption(option as MSARangeChartOptions) As Boolean 147
StandardDeviationChartOption(option as MSASTandardDeviationChartOptions) As Boolean 147
TestRetestStudySetChartOptions(option as MSAStudyChartOptions) As Boolean 147
ShiftDetectionProfiler(Flag as Boolean) 147
AverageChartOption(option as MSASTandardDeviationChartOptions) As Boolean 147
ShowStandardDeviationChart(Flag as Boolean) 147

MultipleCorrespondenceAnalysis 147

LaunchAddResponse(name As String) As Boolean 148
Launch() As Boolean 148
LaunchAddFactor(name As String) As Boolean 148
LaunchAddSupplementaryVariable(name As String) As Boolean 148
LaunchAddSupplementaryID(name As String) As Boolean 148
LaunchAddFreq(name As String) As Boolean 148
LaunchAddBy(name As String) As Boolean 148
LaunchRemoveResponse(name As String) As Boolean 148
LaunchRemoveFactor(name As String) As Boolean 148
LaunchRemoveSupplementaryVariable(name As String) As Boolean 148
LaunchRemoveSupplementaryID(name As String) As Boolean 149
LaunchRemoveFreq(name As String) As Boolean 149
LaunchRemoveBy(name As String) As Boolean 149
CrossTable(flag As Boolean) 149
DisplayOptions(option As MCADisplayOptions, flag as Boolean) 149
SaveCoordinates(nDims As Short) 149
SaveCoordinateFormula(nDims As Short) 149

Multivariate オブジェクトのメソッド 149

ColorMapOnCorrelations(Flag As Boolean) 149
ColorMapOnValues(Flag As Boolean) 149
ClusterOnCorrelations(Flag As Boolean) 150

CorrelationProbability, CIofCorrelation(Flag As Boolean) 150
CorrelationsM(Flag As Boolean) 150
CovarianceMatrix(Flag As Boolean) 150
CronbachsAlpha(Flag As Boolean) 150
Ellipsoid3D(BSTR X, BSTR Y, BSTR Z) As Boolean 150
HoeffdingsD(Flag As Boolean) 150
InverseCorr(Flag As Boolean) 150
KendallsTau(Flag As Boolean) 150
MultivariateSimpleStatistics(Flag as Boolean) 150
OutlierAnalysis(Flag As Boolean) As Outlier 150
PairwiseCorr(Flag As Boolean) 150
ParallelCoordPlot(Flag as Boolean) 151
PartialCorr(Flag As Boolean) 151
PrincipalOnCorrelations As PrincipalComponents 151
PrincipalOnCovariances As PrincipalComponents 151
PrincipalUncentered() As PrincipalComponents 151
SaveTSquare() 151
ScatterPlot(Flag As Boolean) As ScatterPlotMatrix 151
SpearmansRho(Flag As Boolean) 151
StandardizedAlpha(Flag As Boolean) 151
TSquareDistances(Flag As Boolean) 151
UnivariateSimpleStatistics(Flag as Boolean) 151

MultivControl オブジェクトのメソッド 152

PrincipalComponents(Flag as Boolean) 152
SavePrincipalComponents() 152
SaveTargetStatistics() 152
SaveTSquare() 152
ShowCorrelation(Flag as Boolean) 152
ShowCovariance(Flag as Boolean) 152
ShowInverseCorrelation(Flag as Boolean) 152
ShowInverseCovariance(Flag as Boolean) 152
ShowMeans(Flag as Boolean) 152

Neural オブジェクトのメソッド 152

ControlPanelOptions(neuralControlConstants option, Flag As Boolean)
152
Diagram(Flag As Boolean) 153
Go() 153
Profiler(Flag As Boolean) 153
SaveHidden() 153
SaveFormulas() 153

SavePredicted() 153
SaveProfileFormulas() 153
SpecifyConvergeCriterion(Value As Double) As Boolean 153
SpecifyHiddenNodes(Value As Double) As Boolean 153
SpecifyMaxIterations(Value As Double) As Boolean 153
SpecifyNumberOfTours(Value As Double) As Boolean 153
SpecifyOverfitPenalty(Value As Double) As Boolean 154

正規混合のメソッド 154

Biplot, Biplot3D, ParallelCoordinatePlot, ScatterPlotMatrix(Flag as Boolean) 154
BiplotContourDensity(Density as Double) 154
Go() 154
LaunchAddY, LaunchAddFreq, LaunchAddWeight, LaunchAddBy 154
PublishClusterFormulas() 154
SaveClusterFormula, SaveMixtureFormulas, SaveMixtureProbabilities, SaveDensityFormula() 154
SaveColorsToTable 154
SimulateClusters(NRows as Integer) 154
SpecifyNClusters(NClusters as Integer), SpecifyNTours(NTours as Integer), SpecifyMaximumIterations(MaxIterations as Integer), SpecifyConvergeCriterion(Criterion as Double) 155

Onewayオブジェクトのメソッド 155

AnalysisOfMeans(Type as OnewayAofMConstants, Flag As Boolean) 155
CDFPlot(Flag As Boolean) 155
CompareDensities(Flag As Boolean) 155
CompareMeans(Option As Integer, Flag As Boolean) 155
CompositionOfDensities(Flag As Boolean) 155
DisplayOptions(Option As Integer, Flag As Boolean) 155
EquivalenceTest(diffConsideredPracticallyZero as Double) 155
FitRobust, FitCauchy(Flag As Boolean) 155
Histograms(Flag as Boolean) 155
Kolmogorov Smirnov 156
MatchingColumn(ColumnName As String) As Boolean 156
MeansAnovaT(Flag As Boolean) 156
MeansStdDev(Flag As Boolean) 156
NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean) 156
Nonparametric(Option As Integer, Flag As Boolean) 156
NonParametricMultipleComparisons(Type as OnewayNonParMultipleComparisonConstants, Flag As Boolean) 156

NonParametricMultipleWithControl(Type as
OneWayNonParametricMultipleComparisonConstants, ControlValue as String) As
Boolean 156

NormalQuantileLineOfFit(Flag As Boolean) 157

NormalQuantilePlot(Flag As Boolean) 157

NormalQuantilePlotQbyA(Flag As Boolean) 157

NormalQuantileProbLabels(Flag As Boolean) 157

ProportionOfDensities(Flag as Boolean) 157

Save(Option As Integer) 157

SetAlpha(Level As Double) 157

TTest(Flag as Boolean) 157

UnequalVariances(Flag As Boolean) 157

Outlierオブジェクトのメソッド 157

JackknifeDistances(Flag As Boolean) 157

MahalanobisDistances(Flag As Boolean) 158

SaveJackknife() 158

SaveMahal() 158

Overlayオブジェクトのメソッド 158

LaunchAddYWithRightScale(ColumnName As String) 158

LaunchSetSortScaleOptions(XSort as Boolean, XLogAxis as Boolean,
YLogLeftAxis As Boolean, YLogRightAxis As Boolean) 158

LineOptions(overlayLineStyleConstants style,
overlayLineThicknessConstants thickness) 158

Overlay(Flag As Boolean) 158

Range(Flag As Boolean) 158

SeparateAxes 158

YConnectColor(Color As jmpColorConstants) 158

YConnectPoints(Flag As Boolean) 158

YOverlayMarker(Marker as jmpMarkerConstants) 159

YOverlayMarkerColor(Color as jmpColorConstants) 159

YNeedle(Flag As Boolean) 159

YShowPoints(Flag As Boolean) 159

YStep(Flag As Boolean) 159

ParallelPlotオブジェクトのメソッド 159

ReverseScaleOnY(ColumnName as String) As Boolean 159

ShowReverseCheckboxes(Flag as Boolean) 159

Paretoオブジェクトのメソッド 159

AddCauseToCombine(causeName As String) 159

CategoryLegend(Flag As Boolean) 159

CombineCauses() As Boolean 159

CumPercentAxis(Flag As Boolean) 160
CumPercentCurve(Flag As Boolean) 160
CumPercentPoints(Flag As Boolean) 160
HorizontalLayout(Flag As Boolean) 160
Nlegend(Flag As Boolean) 160
PercentScale(Flag As Boolean) 160
PieChart(Flag As Boolean) 160
SeparateCauses() 160
UngroupPlots(Flag As Boolean) 160

Partitionオブジェクトのメソッド 160
ColorPoints(Flag as Boolean) 160
ColumnContributions(Flag as Boolean) 161
Criterion(Option as partitionCriterionConstants, Flag as Boolean) 161
DisplayOptions(Option as partitionDisplayConstant, Flag as Boolean) 161
KFoldCrossValidation(value as Integer) 161
LeafReport(Flag as Boolean) 161
LiftCurve(Flag as Boolean) 161
LockColumns(Flag as Boolean) 161
MinimizeSizeSplit(value as double) 161
MissingValueRule(Option as partitionMissingConstants, Flag as Boolean) 161
PlotActualByPredicted(Flag as Boolean) 161
Prune() 161
ROCCurve(Flag as Boolean) 161
SaveColumns(SaveOperation as partitionSaveColumnConstants) 161
SmallTreeView(Flag as Boolean) 162
Split() 162
SplitHistory(Flag as Boolean) 162

PLSオブジェクトメソッド 162
LaunchSpecifyModelMethod(Method As plsModelMethodConstants) 162
LaunchSpecifyValidationType(valType As plsValidationTypes, valParm As Double) 162
LaunchSpecifyInitialNumberOfFactors(nFactors As Int) 162
LaunchSetRandomSeed(Seed As Double) 162
LaunchAddValidationColumn(Name As String) 163
LaunchRemoveValidationColumn(Name As String) 163
LaunchSpecifyOptions(Centering As Boolean, Scaling As Boolean) 163
LaunchSpecifyImputeMethod(Method As plsImputMethods, Iterations As Int) 163

PercentVariationPlots、LoadingScatterPlotMatrices、Profiler、
VIPVersusCoefficientPlots、CoefficientPlots、
ScoreScatterplotMatrices、SpectralProfiler(Flag As Boolean) 163
CorrelationLoadingPlot(Int nFactors) 163
ConfidenceLines(Flag As Boolean) 163
SaveFormula() 163
SaveOutputs(Flag As Boolean) 163
ShowPoints(Flag As Boolean) 163

PrincipalComponents オブジェクトのメソッド 164

FactorRotation(N As Integer) 164
SavePrincipal(Num As Integer) 164
SaveRotated() 164
Spin(Flag As Boolean) 164

Profiler オブジェクトのメソッド 164

ConfidenceIntervals(Flag As Boolean) 164
Desirability(Flag As Boolean) 164
InteractionProfiler(Flag as Boolean) 164
LaunchAddNoiseFactors(NoiseFactorsColumn As String) As Boolean 164
MostDesirable() 164

Recurrence オブジェクトのメソッド 164

EventPlot(Flag As Boolean) 164
MCFConfidLimits(Flag As Boolean) 165
MCFPlot(Flag As Boolean) 165
PlotMCFDifferences(Flag as Boolean) 165

Scatterplot3D オブジェクトのメソッド 165

BiplotRays(Flag As Boolean) 165
ConnectPoints(BSTR groupingColumn) 165
DropLines(Flag As Boolean) 165
NormalContourEllipsoids(BSTR groupingColumn) 165
PrincipalComponents() 165
RotatedComponents() 165
SavePrincipalComponents(Number as Long) 165
SaveRotatedComponents() 165
ShowPoints(Flag As Boolean) 166
StdPrincipalComponents() 166

ScatterplotMatrix オブジェクトのメソッド 166

DensityEllipses(Flag As Boolean) 166
EllipseAlpha(Alpha As Double) 166
EllipseColor(Color As Integer) 166
Histograms(HorizontalHistogram As Boolean, Flag As Boolean) 166

ScatterplotMatrixPlatform オブジェクトのメソッド 166

DensityEllipses(Flag As Boolean) 166

EllipseAlpha(Alpha As Double) 166

EllipseTransparency(Transparency As Double) 166

LaunchSpecifyMatrixFormat(scatterplotMatrixFormatconstants val) 166

ShowCorrelations、ShowPoints、FitLine、NonParDensity (Flag As Boolean) 167

Screening オブジェクト 167

メソッド 167

SpinPlot オブジェクトのメソッド 167

BiplotRays(Flag As Boolean) 167

PrincipalComponents() 167

RotatedComponents(Number As Integer) 167

SavePrincipalComponents() 167

SavePrincipalComponents2(NumberToSave as Short) 167

SaveRotatedComponents() 167

Spin(pitch As Integer, yaw As Integer, roll As Integer, numTimes As Integer) 168

SpinPitch(Angle As Integer) 168

SpinRoll(Angle As Integer) 168

SpinYaw(Angle As Integer) 168

StdPrincipalComponents() 168

SurfacePlot オブジェクトのメソッド 168

DisplayOptions(option as surfaceDisplayOptions, flag as Boolean) 168

SetItemColor(item as surfaceColorConstants, color as JMPColorConstants) 168

Survival オブジェクトのメソッド 168

CompetingCauseAction(competitingCauseConstants action, Flag as Boolean) 168

CompetingCauses(columnName As String) As Boolean 168

ExponentialEst(Flag As Boolean) 169

ExponentialPlot(Flag As Boolean) 169

LognormalEst(Flag As Boolean) 169

LognormalPlot(Flag As Boolean) 169

MidStepQuantilePoints(Flag As Boolean) 169

ReverseYAxis(Flag As Boolean) 169

SaveEstimates() As DataTable 169

ShowCombined(Flag As Boolean) 169

ShowConfidInterval(Flag As Boolean) 169

ShowPoints(Flag As Boolean) 169

SurvivalPlot(Flag As Boolean) 169
ShowSimultaneousCI(Flag as Boolean) 170
WeibullEst(Flag As Boolean) 170
Weibull-Plot(Flag As Boolean) 170

Ternaryオブジェクトのメソッド 170

LaunchAddFormulaCol(ColumnName As String) As Boolean 170
LaunchRemoveFormulaCol(ColumnName As String) As Boolean 170

Text Explorer 170

LaunchAddTextColumn(name as String) As Boolean 170
LaunchAddID(name as String) As Boolean 170
LaunchAddBy(name As String) As Boolean 170
LaunchMaxWordsPerPhrase(n As Short) As Boolean 171
LaunchRemoveTextColumn(name As String) As Boolean 171
LaunchRemoveID(name As String) As Boolean 171
LaunchRemoveBy(name As String) As Boolean 171
LaunchMaxNumberOfPhrases(n As Long) As Boolean 171
LaunchMinCharactersPerWord(n As Short) As Boolean 171
LaunchMaxCharactersPerWord(n As Short) As Boolean 171
LaunchLanguage(option As textExplorerLanguageOptions) As Boolean - 171
LaunchStemming(option As textExplorerStemmingOptions) As Boolean 171
LaunchTokenizing(option As textExplorerTokenizingOptions) As Boolean 171
LaunchTreatNumbersAsWords(flag As Boolean) As Boolean 172
Launch() As Boolean 172
DisplayOptions(option As textExplorerDisplayOptions, flag as Boolean) 172
LatentClassAnalysis(maxNumTerms As Short, minTermFrequency As Short, numClusters As Short) 172
LatentSemanticAnalysis(maxNumTerms As Short, minTermFrequency As Short, weighting As textExplorerSemanticWeightingOptions, numSingularVectors As Short, centeringAndScaling As textExplorerSemanticCenteringOptions) 172
TopicAnalysis(numTopics As Short) 172
ClusterTerms(flag As Boolean) 172
ClusterDocuments(flag As Boolean) 173
SVDSscatterplotMatrix(numVectors As Short) 173
TopicScatterplotMatrix(flag As Boolean) 173
SaveDocumentTermMatrix(maxNumTerms As Short, minTermFrequency As Short, weighting As textExplorereSemanticWeightingOptions) 173
SaveDocumentSingularVectors(numVectors as Short) 173

SaveDocumentTopicVectors() 173
SaveStackedDTMForAssociation() As JMP.DataTable 173
SaveDTMFormula() 173
SaveSingularVectorFormula() 173
SaveTopicVectorFormula() 174
SaveTermTable() As JMP.DataTable 174
SaveTermSingularVectors(numVectors as Short) 174
SaveTermTopicVectors() 174
ScoreTermsByColumn(columnName As String) 174

TimeSeries オブジェクトのメソッド 174

- ARCoefficients(Flag As Boolean) 174
- Arima(p As Double, d As Double, q As Double, confidenceInterval As Double, intercept As Boolean, constrainFit As Boolean) 174
- Autocorrelation(Flag As Boolean) 174
- ConnectingLines(Flag As Boolean) 175
- MeanLine(Flag As Boolean) 175
- PartialAutocorr(Flag As Boolean) 175
- SaveSpectralDensity() As DataTable 175
- ShowPoints(Flag As Boolean) 175
- SmoothingModel(Model As Integer, Constraints As Integer) 175
- SpectralDensity(Flag As Boolean) 175
- TimeSeriesGraph(Flag As Boolean) 175
- Variogram(Flag As Boolean) 175

Variability オブジェクトのメソッド 175

- AIAGLabels(Flag As Boolean) 175
- BiasReport(Flag As Boolean) 176
- ConnectCellMeans(Flag As Boolean) 176
- DiscriminationRatio(Flag As Boolean) 176
- GageRandR(K As Double, Tolerance As Double) 176
- LinearityStudy(Flag As Boolean) 176
- NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean) 176
- PointsJittered(Flag As Boolean) 176
- ShowBoxPlots(Flag as Boolean) 176
- ShowCellMeans(Flag As Boolean) 176
- ShowGrandMean(Flag As Boolean) 176
- ShowGroupMeans(Flag As Boolean) 177
- ShowPoints(Flag As Boolean) 177
- ShowRangeBars(Flag As Boolean) 177
- ShowStdDevChart(Flag As Boolean) 177

ShowVariabilityChart(Flag As Boolean) 177

VarianceComponents(option As Integer) As Boolean 177

テクノロジーに関する通知 178

外部アプリケーションによる JMP の実行

JMP の大部分は OLE オートメーションを使って操作することができます。

- 35 ページの「[Visual Basic を使った JMP のオートメーション](#)」では、Visual Basic を使った JMP のオートメーションについて説明します。
- 44 ページの「[Visual C++ を使った JMP のオートメーション](#)」では、MFC を使った Visual C++ によるオートメーションを紹介します。
- 47 ページの「[Visual C# を使った JMP のオートメーション](#)」では、C# を使った JMP のオートメーションについて説明します。
- 49 ページの「[Windows で JMP のオートメーションを行うための Application オブジェクトリファレンス](#)」では、Visual Basic や Visual C++ のようなオートメーションクライアントで利用できる JMP のメソッドやプロパティについて詳しく説明します。

「[JMP¥17¥Samples¥Automation](#)」フォルダには、JMP のオートメーション機能を Visual Basic .Net、Visual C# .Net、Visual C++ .Net でプログラミングする例があります。Visual Basic のプログラムには、Visual Studio 2013 以降が必要です。

Visual Basic を使った JMP のオートメーション

JMP アプリケーションの起動

JMP のオートメーション機能における最初のステップは JMP を起動することですが、その前に JMP のメソッドやプロパティで利用可能なリソースについて見ておくことが大切です。JMP には、タイプライブラリがあり、これによって、Visual Basic (以下 VB) のようなオートメーションクライアントは、JMP が公開しているメソッドやプロパティのリスト、および各メソッドで必要となるパラメータを表示できます。このライブラリは JMP.TLB というファイルで提供されています。

JMP のタイプライブラリを VB から利用できるようにするには、次の手順を行います。

1. VB の [プロジェクト] > [参照の追加] を選択します。すると、VB から参照可能なアプリケーションのリストが表示されます。リストに JMP がないときは、[参照] を選択し、ファイル ウィンドウで .tlb (タイプライブラリ) の場所を指定します。JMP ディレクトリ内で、JMP のタイプライブラリのアイコンを探します。このライブラリを選択し、[OK] ボタンをクリックします。
2. VB の [表示] > [オブジェクトブラウザ] を選択してオブジェクトブラウザを開きます。そして、ドロップダウンリストボックスから JMP を選択します。

これで JMP のオートメーションクラスと定数を参照できます。クラスを選ぶと、そのクラスで利用可能なメソッドがオブジェクトブラウザの右側のリストボックスに表示されます。メソッドを選択すると、ウィンドウの下の方に、ヘルプ用の短いストリングが表示されます。このストリ

ングは、そのメソッドで使われるパラメータをリスト表示します。メソッドが特定のパラメータ群を必要とするときには、通常、特定のアクションを表記する定数が使われます。

これでタイプライブラリの情報にアクセスできるようになり、JMP のクラスのインスタンスを生成するのに必要なコードが書けます。これは `CreateObject` を使って行います。VB のプロジェクトのグローバル宣言で、`JMP.Application` タイプの変数を作成します。次のように指定します。

```
Dim MyJMP As JMP.Application
```

この際に、他の変数の作成も行います。`DataTable`、`Distrib`、`Oneway`、および `JMPDoc` がその例です。それぞれ、`JMP.DataTable`、`JMP.Distribution`、`JMP.Oneway`、および `JMP.Document` で指定できます。

JMP のセッションを作成して、セッションを表示し、データテーブルを読み込むには、以下のコードを VB のスクリプトに加えます。

```
Dim JMPDoc As JMP.Document
Set MyJMP = CreateObject("JMP.Application")
MyJMP.Visible = True
Set JMPDoc = MyJMP.OpenDocument("C:\Program
Files\SAS\JMP\17\Samples\Data\Big Class.jmp")
```

`Dim` では変数の型を宣言しています。ただし、この宣言は、VB のプロジェクトの一般的な宣言の中に入れる必要があります。そうしないと、処理の終わりでこの変数が有効範囲外となり、JMP オブジェクトは破棄されてしまいます。

オートメーションのガイドラインで定められているように、デフォルトでは JMP は表示されません。このため、上のコードにあるように、処理の最初のステップで JMP を表示させる必要があります。

複数の JMP インスタンスを開く

デフォルトでは、JMP は複数のクライアントから接続できるオートメーションサーバーとして実行されます。これは、最初のリクエストにより JMP の新しいインスタンスが作成され、その後のクライアントからのリクエストはすべて同じインスタンスを使用することを意味します。

JMP を単一のクライアントのみ接続可能なオートメーションサーバーとして実行することも可能です。その場合、新しいクライアントからのリクエストのたびに JMP が起動されます。JMP を単一クライアントのサーバーとして使用するには、レジストリキーを変更する必要があります。

JMP がオペレーティングシステムと同じビットアーキテクチャで実行されている場合は、次のレジストリキーの値を「Multiple」から「Single」に変更します。

```
HKEY_CLASSES_ROOT\CLSID\{97BCFCC0-7822-11CF-9E68-
0020AF24E9FE}\ServerUse
```

JMP が 32 ビットでオペレーティングシステムが 64 ビットの場合は、次のレジストリキーの値を「Multiple」から「Single」に変更します。

```
HKEY_CLASSES_ROOT\Wow6432Node\CLSID\{97BCFCC0-7822-11CF-9E68-
0020AF24E9FE}\ServerUse
```

警告: レジストリに変更を加えるときは、必ずレジストリのバックアップを作成してから行ってください。詳細については、Windows のヘルプや Microsoft のマニュアル、Microsoft Windows の Web サイトをご覧ください。SAS は、Windows レジストリを編集することに対して責任を負いません。Windows レジストリに変更を加えると、システムが使用不可能になり、オペレーティングシステムの再インストールが必要になる場合があります。

分析の開始

これでデータテーブルが開き、分析を開始し、操作できるようになりました。まず、各分析のオブジェクトを生成します。次に、分析に必要なパラメータを指定します。オプションを設定することもできます。指定が終わったら、分析のプラットフォームを開始します。その後、分析オブジェクトで追加オプションの処理が行われます。

```
Dim Oneway As JMP.Oneway
Set Oneway = JMPDoc.CreateOneway
Oneway.LaunchAddY ("Height")
Oneway.LaunchAddX ("Age")
' 起動前にオプションを指定
Oneway.Quantiles (True)
' 最初の分析出力を生成
Oneway.Launch
Oneway.MeansAnovaT (True)
Oneway.MeansStdDev (True)
Oneway.UnequalVariances (True)
Oneway.NormalQuantilePlot (True)
Oneway.SetAlpha (0.05)
Oneway.Save (oscCentered)
Oneway.Save (oscStandardized)
Oneway.CompareMeans occAllPairs, True
Oneway.CompareMeans occEachPair, True
```

最初に Document クラスの CreateOneway を呼び出して、分析オブジェクトを生成します。次に X の列と Y の列を指定し、Launch をコールして実際の一元配置分析を生成します。各分析プラットフォームのオブジェクトはそれぞれ別のメソッドで生成されますが、これはオブジェクトブラウザの Document で確認できます。多くの場合、オブジェクトの Launch の前にオプションを指定できるため、分析の出力はすでに設定されたオプションに従ったものになります。この例では、オプション処理の多くは分析の起動後に行われています。これはディスプレイの中のオプションのポップアップメニューに表示されます。見てのとおり、メソッドの多くは、メニューから設定するかのような簡単なオプション設定です。SetAlpha には 1 つのパラメータが指定されています。これは、オートメーションの実行中に入力を要求するウィンドウが表示されないようにするためです。CompareMeans はパラメータを 2 つとり、1 つは比較の種類を指定し、もう 1 つはオンまたはオフを示すパラメータを指定します。Save は予め定義された定数（オブジェクトブラウザで確認できます）をとり、一元配置分析で何を保存するかを指定します。

分析手法の多くは上のように処理されますが、Bivariate（二変量）のような手法には、コールされた時点で追加のオブジェクトを生成するものもあります。例：

```
Set Fit = Bivar.FitLine
```

```
Fit.ConfidenceFit (True)
Fit.ConfidenceIndividual (True)
```

ここで、FitLine は Fit 型のオブジェクトを生成します。このオブジェクトは固有のメソッドとプロパティを持ち、それを操作することができます。FitLine で生成された新しいオブジェクトは、その変数が参照可能な範囲内にある間だけ操作できることに注意してください。

あるメソッドから、プログラムで制御のできるオブジェクトが生成された場合、オブジェクトブラウザでこれを確認できます。FitLine の場合、オブジェクトブラウザには戻り値が As Fit 型であると示されます。

これは、short や BSTR のように予め定義された型ではないので、オブジェクトであると推測できます。オブジェクトブラウザをさらに見ていくと、Fit がオブジェクト型であることがわかります。このように、オブジェクトが生成されたことが確認でき、また Fit がサポートするメソッドもわかります。

データテーブルを作成し、値を埋め込む

新しいデータテーブルは、(適当な名前をつけて) Application オブジェクトの NewDataTable というメソッドで作成できます。ファイル名は作成時に割り当てられます。NewColumn メソッドでは列オブジェクトが戻され、これは行を追加していく限りは保持しておく必要があります。デフォルトでは、20 行が作成されます。セルに値を書き込むには SetCellVal メソッドを使います。行を加えるには AddRows を使います。以下はその例です。

```
Dim Col As Object
Set DT = MyJMP.NewDataTable("C:\test.jmp")
Set Col = DT.NewColumn("列 1", dtTypeNumeric, 0, 8)
DT.Visible = True

' テーブルに値を埋め込む前に行を加える
DT.AddRows 20,0

' 1.5 刻みで増える値をセルに埋め込む
For i = 1 To 10
    Col.SetCellVal i, i * 1.5
Next i
DT.Visible = False
For i = 11 To 20
    Col.SetCellVal i, i * 1.5
Next i
DT.Visible = True

' テーブルの先頭に 5 行加える
DT.AddRows 5, 0
' 2 行目の後に 5 行加える
DT.AddRows 5, 2

' すでに指定してあるファイル名でデータテーブルを保存する
```

DT.Document.Save

　' テーブルの 1 行目から 3 行目までだけを取り出したテーブルを作るには
　' 次のようにする

　' メモ : Datatable のメンバー関数 AddToSubList を使って列をリストに加えれば、
　' 特定の列だけを取り出したテーブルを作ることもできる

```
Dim NewDT As JMP.Datatable  
Dim DTDoc As JMP.Document  
DT.SelectRows 1,3  
Set NewDT = DT.Subset
```

　' 新しいテーブルを保存

```
Set DTDoc = NewDT.Document  
DTDoc.SaveAs("C:\MySubset.jmp")
```

プログラム例

「JMP¥17¥Samples¥Automation」フォルダには、JMP のオートメーション機能を Visual Basic .Net、Visual C# .Net、Visual C++ .Net でプログラミングする例があります。Visual Basic のプログラムには、Visual Studio 2005 以降が必要です。

「Analysis」では、ほとんどすべての JMP プラットフォームの単純なオートメーションの例を示しています。例の中のコードはプラットフォームの機能をテストするもので、意味のある統計解析を行うものではありません。このプログラム例の目的は、オートメーション用コードの作成方法を紹介することです。JMP のタイプライブラリが VB のプロジェクトから参照できるようにしておくと便利です。この方法については、このマニュアルの最初の節を参照してください。オートメーションを行うプラットフォームにあるメソッドやプロパティを確認できます。

「Data Table」では、データテーブルのオートメーションに利用できるメソッドの例を示しています。意味のある出力を得ようとしているわけではありません。

「Text Import」では、JMP でテキストファイルを読み込みデータテーブルにする際に必要な手順を示しています。この処理を行うと、「Data Table」での例のようにデータテーブルを操作できるようになり、また「Analysis」での例のように、データについての分析ができるようになります。

「ODBC」では、ODBC アクセスを使って dBase のファイルを JMP に読み込む簡単な例を示しています。

「Word Demo」では、JMP のレポートからグラフの部分をクリップボードにコピーし、Microsoft Word の文書に挿入するためのコマンドを示しています。

「FitModel」と「DOE」の例では、JMP のそれぞれ該当する領域に固有の演算子を示しています。これらのプラットフォームの演算子は、他のプラットフォームとは若干異なっています。

サンプルのプログラム例では、データファイルがデフォルトのサンプルデータのディレクトリにあることを前提としています。データファイルの場所を移動した際は、VB のプログラム例の中にあるパスの部分を変更してください。

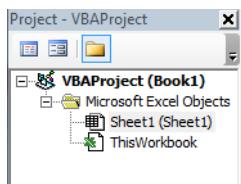
このマニュアルにある Visual Basic のコード例と、サンプルプログラムの例に違いがある場合は、サンプルプログラムのコードのほうを参考にしてください。

Microsoft Excel 2013を使ったJMPのオートメーション

この例では、Microsoft Excel 2013 ワークシートのマクロによって JMP を操作します。マクロコードは、Visual Basic で記述されます。このマクロは、Microsoft Excel ワークシートが開かれたときに、JMP を表示して開始します。次に、ODBC オートメーションインターフェースを使って、Microsoft Excel ワークシートを読み込みます。ワークシートのデータが JMP に読み込まれると、個々のワークシートのセルに加えた変更が JMP に送られ、JMP のデータテーブルが変更されます。

初めて Microsoft Excel で行の値が変更されたときに、JMP により管理図が生成されます。それ以降 Microsoft Excel のワークシートに変更が加えられると、管理図も変更されます。この例では、Microsoft Excel が変更されると JMP データテーブルが更新され、この JMP データテーブルの変更が直接リンクされている管理図の出力に反映されるからです。Microsoft Excel ワークシートが 5 回変更されるごとに、管理図の .PNG ファイルを生成するメソッドが JMP で呼び出されます。これにより、ユーザは JMP を使わずに Web ブラウザに出力を表示することができます。最後に、Excel ワークシートが閉じるときに、JMP もオートメーション機能によってシャットダウンします。

Microsoft Excel を開くところから始めてみましょう。Microsoft Excel のブックで使う Visual Basic スクリプトを作成するには、[開発] リボンから [Visual Basic] を選びます。Visual Basic Editor が別のウィンドウに表示されます。Visual Basic Editor の左側には、「VBAProject」というペインがあり、ブックと、Visual Basic コードが関連付けられているシートが表示されています。



ブック用に記述されたコードは、通常ブック内のすべてのシートで動作します。

以下の例では、3 つのセクションに分けてコーディングていきます。まず最初に、`module1.bas` ファイルでいくつかの変数をグローバル変数として宣言します。これで、宣言した変数が他のコードのモジュールで参照されるようになります。VBA のプロジェクトアイコンでコンテキストクリックし、[挿入] > [標準モジュール] を選ぶと、モジュールを Visual Basic プロジェクトに挿入できます。以下のコードをモジュールに入力します。このコードでは、JMP アプリケーションのインスタンス、JMP データテーブル、およびドキュメントが開いているかどうか確認するためのフラグを宣言しています。

```
Public MyJMP as JMP.Application      ' JMP アプリケーションオブジェクト
Public DT As JMP.DataTable           ' JMP データテーブルオブジェクト
Public DocOpen as Boolean           ' 「JMP テーブルが開いている」ことを示すフラグ
```

次のセグメントでは、Microsoft Excel ワークシートのセルが変更されると、JMP が更新されるようにします。セルが変更、削除、追加されると、必ず Microsoft Excel では `Worksheet_change` イベントが生成されるので、このセグメントは自動的に呼び出されるようになります。

Microsoft Excel の VBA プロジェクトブラウザには、現在ブックに含まれているワークシートが表示されます。以下のコードを、JMP にデータを送るワークシートに入力します。VBA プロジェクトウィンドウでワークシートアイコンをダブルクリックし、そのワークシートのコードのウィンドウを表示します。

```
Private Sub Worksheet_change(ByVal Target as Range)
    Dim Col as JMP.Column
    If (DocOpen) Then
        If(Target.Row = 1) Then
            Return
        EndIf
        If(DT.NumberRows < Target.Row - 1) Then
            DT.AddRows Target.Row - DT.NumberRows - 1, Target.Row
        EndIf
        If(Not IsArray(Target.Value) And Not IsEmpty(Target.Value))
    Then
        Set Col = DT.GetColumnByIndex(Target.Column)
        Col.SetCellVal Target.Row - 1, Target.Value
    EndIf
    EndIf
End Sub
```

このコードでは、まず JMP でデータテーブルが開いていることを確認します。最初の行が変更されたときには、この行は JMP の列名なので無視されます。そして、Microsoft Excel で列名が変更されても、その変更は JMP には反映されません。見出しを変更するコードをここに挿入することができますが、この例では省略しています。

次に、変更された行が、データテーブルで JMP が現在確認している行番号を超えている場合には、`AddRows` メソッドが呼び出されて行が追加されます。

最後に、処理が 1 つの値について行われていて、削除を示していない場合には、JMP データテーブルのセルの値は `Worksheet_Change` に渡された値に変更されます。

メインのモジュールは、ブックに関するものです。VBA プロジェクトブラウザでは、ブックコード領域には通常 `ThisWorkbook` という名前が付いていますが、この名前は簡単に変えられます。次のコードをこの領域に入力します。

```
' ブックの全サブルーチンにアクセスできる Public (グローバル変数) 宣言
Public Counter As Integer '5 回変更されるたびに管理図を更新するためのカウンタ
Public JMPDoc As JMP.Document 'JMP ドキュメントのインスタンス
Public CChart As JMP.ControlChart '管理図のインスタンス
Public ChartOpen as Boolean '管理図が開いているかどうかを設定するフラグ
Public DB As AUTODB
```

```
' ブックを閉じる前に JMP をシャットダウンする
```

```
Private Sub Workbook_BeforeClose(Cancel as Boolean)
    DocOpen = False
    MyJMP.Quit
End Sub

' [ファイル] > [開く] でブックを開くと、オートメーションの JMP をロードする
Private Sub Workbook_Open()
    Set MyJMP = CreateObject( "JMP.Application" ) ' JMP のインスタンスを作成する
    MyJMP.Visible=True ' この JMP のインスタンスを表示させる
    Counter = 0 ' 変更をカウントするカウンタを初期化する
    DocOpen = False ' まだドキュメントを開いていない
    ChartOpen = False ' チャートも開いていない

    'Excel ワークシートを指すようにこのパスを変更する
    Set DB = MyJMP.NewDatabaseObject
    DB.Connect ("DSN=Excel Files;DBQ=C:¥Book2.xls;")
    Set DT = DB.ExecuteSQLSelect("SELECT * FROM ""Sheet1$""")
    DB.Disconnect
    Set JMPDoc = DT.Document
    DocOpen = True ' ドキュメントが開いていることを示すフラグを設定する
End Sub

' ここが最も重要な部分である。
' 最初のデータが変更されたら、管理図を作成する。
'Excel ワークシートのセルに 5 回変更が加えられるたびに、管理図の PNG ファイルを作成する。

Private Sub Workbook_SheetChange(ByVal Sh As Object, ByVal Source As Range)
    Counter = Counter + 1
    ' 要素が 5 つ更新されるたびに、管理図を PNG ファイルに保存する
    If (Counter Mod 5 = 0 Or Counter = 1) Then

        ' 管理図が作成されていなかったときには作成する
        If Not (ChartOpen) Then
            Set CChart = JMPDoc.CreateControlChart ' 管理図を作成する
            CChart.LaunchAddProcess "列1" ' 列を追加する
            CChart.LaunchAddSampleUnitSize 5
            CChart.LaunchSetChartType jmpControlChartVar
            CChart.Launch ' 管理図を起動する
            ChartOpen = True ' 管理図が開いていることを示すフラグを設定する
        EndIf
        CChart.SaveGraphicOutputAs "C:¥ControlChart.png", jmpPNG
    EndIf
End Sub
```

End Sub

Microsoft Excel のテーブルが最初にロードされるときに、`Workbook_Open` サブルーチンが呼び出されます。このサブルーチンでは、変数を初期化し、JMP を起動し、現在 Microsoft Excel 2013 にロードされているものと同じ Microsoft Excel ファイルを (ODBC を使って) JMP で開くように、JMP に指示します。JMP では、Microsoft Excel ファイルがファイルとしてではなく、データベースオブジェクトとして開かれます。JMP では別のアプリケーションですでに開いているファイルを開けないため、これは必要な対処です。

ユーザがブック内のワークシートでセルのデータを変更するたびに、`Workbook_Change` イベントが生成されます。このコード例では、ブックにあるアクティブなワークシートは 1 つだけと仮定しています。初めてユーザがワークシートのセル値を変更したときに、`Workbook_Change` サブルーチンは、現在のデータテーブルを使って JMP に管理図を作成します。

この例では、`Workbook_Change` サブルーチンは、管理図出力用の PNG グラフィックファイルを作成し、ブックに 5 回目の変更が加えられるたびにディスク上のそのファイルを更新します。これは、Microsoft Excel イベントと JMP オートメーションを一緒に使って結果を出力する方法をわかりやすくするためです。

最後に、Microsoft Excel ブックのウィンドウが閉じる直前に、`Workbook_BeforeClose` サブルーチンが呼び出されます。このサブルーチンは、JMP にウィンドウを閉じるよう指示するコードです。

このプログラム例にはいくつか制限があります。Excel でのデータ操作において、データの追加または変更の場合にだけ、このプログラム例はうまく機能します。しかし、Microsoft Excel の `Sheet_Change` イベントには、データの追加や変更以外のイベントも含まれています。特に、削除、ドラッグ＆ドロップ、ロック反復を使用する必要がある場合、JMP のデータテーブルをセル単位で更新することが難しくなります。

このような場合には、強引な方法を採用するほうがおそらく賢明です。その 1 つの方法としては、ある一定の回数変更されるたびに、JMP にデータを再読み込みする方法があります。次にその例を示します。

```
Private Sub Workbook_SheetChange(ByVal Sh As Object, ByVal Source As Range)
    Counter = Counter + 1
    If (Counter Mod 10 = 0) Then
        ' 以前の管理図のテーブルが開いている場合は、まずそれを閉じる
        If (DocOpen) Then
            JMPDoc.Close False, ""
            CChart.CloseWindow
        EndIf

        Set JMPDoc = MyJMP.OpenDocument(InstallDir + "C:¥BOOK1.XLS")
        Set DT = JMPDoc.GetDataTable
        DocOpen = True

        ' 管理図を作成する。
    EndIf
End Sub
```

```
' これは、「列 1」のデータに入る。  
' 5 つ以上の値が変更されたとき、  
' JMP は新しい管理図を生成し、それを  
' PNG ファイルとしてディスクに保存する。  
' PNG ファイルは Internet Explorer で表示可能。  
  
Set CChart = JMPDoc.CreateControlChart  
CChart.LaunchAddProcess "列 1"  
CChart.LaunchAddSampleUnitSize 5  
CChart.LaunchSetChartType jmpControlChartVar  
CChart.Launch  
CChart.SaveGraphicOutputAs "C:\ControlChart.png", jmpPNG  
EndIf  
End Sub
```

この例では、Microsoft Excel のブックが 10 回変更されるたびに、データを再読み込みしています。まず、このコードでは、既存の JMP 管理図とデータテーブルを削除しています。次に、新しいデータをロードして管理図を作成しています。

このサンプルコードは、少ないデータ量の場合に最もうまく動作します。この方法では JMP にテーブルを再読み込みしているので、膨大な Microsoft Excel ファイルを読み込むときには効率的ではありません。

Visual C++ を使った JMP のオートメーション

C や C++ を使ってオートメーションクライアントを作成するのは、時間のかかる退屈な作業です。しかし、Microsoft Visual C++ の MFC の機能を使えば、作業は格段に楽になります。オートメーションサーバーアプリケーション（この場合は JMP）を起動できる状態になるまでには、いくつかの手順を踏む必要があります。Microsoft のサンプルアプリケーションである「CALCDRIV」にも、MFC ベースのオートメーションクライアントが示されています。「CALCDRIV」は、通常、Visual C++ や MSDN の CD に含まれています。

AutoClient では、JMP の起動方法と、二変量の分析やデータテーブルを扱う方法を示しています。このサンプルは、Visual Basic の他のサンプルよりも小さいものですが、すべてのオートメーションの裏の仕組みは、ここにある二変量やデータテーブルを使用した例と同じと考えてください。以下の手順は Visual C++ Version 5.0 のユーザーインターフェースに準拠しています。

JMP のオートメーションの手順

1. App Wizard を使うか、または手動でアプリケーションを生成します。まず OLE オートメーションのサポートを指定します。自分で作ったアプリケーションをオートメーションにするわけではなくても、OLE のヘッダと初期化コードをインクルードする必要があります。既存のアプリケーションを作り変える場合には、必ず OLE サポートを含める必要があります。このためには、通常、アプリケーションに `afxole.h` をインクルードして、アプリケーションの `InitInstance` ルーチンで `AfxOleInit()` をコールします。詳しい方法については、MFC OLE のマニュアルを参照してください。
2. Class Wizard を立ち上げ、オートメーションのタブを選択します。[クラスの追加] のドロップダウンリストで [タイプライブラリから] を選びます。JMP インストールディレクトリを検索し、「JMP.TLB」を選択します。
3. プロジェクトで使うクラスを確認するためのダイアログボックスが表示されます。どのオブジェクト（およびインターフェース）を選べばいいのかがわからない場合は、Shift キーを押しながらクリックしてすべてを選択します。ClassWizard がインターフェースのスタブとヘッダ情報を生成するファイルの名前を選びます。ClassWizard は、MFC `ColeDispatchDrive` クラスに基づいたラッパークラスを生成します。これで、技術的な詳細を知らなくても、OLE オートメーションの関数 `Invoke` に簡単にアクセスできるようになります。[OK] を選択します。クラスウィザードは2つのファイル（.h と .cpp）を作成します。たとえば View クラスを実装するファイルのように、.cpp ファイルで JMP のオートメーションオブジェクトが使われる場合は、必ず .h ファイルを含めます。ファイルをインクルードしてください。
4. ワークスペースの Class View には、読み込まれたインターフェースクラスが表示されているはずです。この Class View を使うと、各クラスのメソッドとプロパティを調べることができます。
5. JMP を起動するには、オートメーションのセッションの間有効となる `IJMPAutoApp` 型の変数を定義します。次いで、この変数について `CreateDispatch` をコールし、単独のパラメータとして JMP ProgID ("JMP.Application") を渡します。この時点でコードが実行され、JMP が起動します。
6. ステップ5で生成された JMP オブジェクトについて `SetVisible(TRUE)` をコールします。JMP の実行状況を見る必要がなければこれは不要です。ただし、デバッグの際には必要になります。
7. これで JMP のアプリケーションオブジェクトを使ってさらにオブジェクトを生成できます。できたオブジェクトからさらにオブジェクトを生成することもできます。まず、データテーブルをロードします。既存の JMP データテーブルをロードするには、ステップ5で作成された JMP オブジェクトについて `OpenDocument` メソッドをコールします。成功すれば、メソッドはディスパッチポインタを戻します。このポインタは `AttachDispatch` メソッドを使って `IJMPDoc` 型のオブジェクトに付加できます。
8. `IJMPDoc` オブジェクトには、「分析」プラットフォームと「グラフ」プラットフォームを起動するメソッドがあります。分析のオブジェクトを生成し、ディスパッチポインタを貼り付けると、分析で使われるデータテーブル列が指定でき、分析を起動できます。いったん分析を起動すると、その分析の種類に合ったプロパティやメソッドを使って操作ができるようになります。以下は、ステップ5～8を説明するアプリケーション例のコードです。

プログラム例

```
// メモ：この例ではエラー処理はなし
IJMPAutoApp m_DispDriver;
IJMPDoc m_Doc;
IAutoBivar m_Bivar;
IAutoFit m_FitLine;

// IJMPAutoApp のインターフェース設定 (jmpauto.h から取得) を使った
// 初期ディスパッチドライバを生成する
m_DispDriver.CreateDispatch("JMP.Application");

if (m_DispDriver)
{
    // JMP がうまく起動されたら、それを表示させる
    m_DispDriver.SetVisible(TRUE);

    // データテーブルをドキュメントとして開く。ドキュメントのインターフェース
    // ポインタが戻され、これが IJMPDoc インターフェース設定を使った
    // Doc ディスパッチドライバクラスに付加される
    m_Doc.AttachDispatch(m_DispDriver.OpenDocument(
        "C:\\\\JMPDATA\\\\BIGCLASS.JMP"));
}

// 最後に Doc インターフェースについて CreateBivariate をコールし
// 二変量の分析へのディスパッチオブジェクトを作成する。
// 以前のディスパッチインターフェースが m_Bivar にすでにある場合、
// MFC はそれを AttachDispatch で解放する。
m_Bivar.AttachDispatch(m_Doc.CreateBivariate());

// 分析対象の列として身長と体重を加える
m_Bivar.LaunchAddX("Height");
m_Bivar.LaunchAddY("Weight");

// 分析を起動する
m_Bivar.Launch();

// FitLine を作成する。Fit はオートメーション可能なので、
// FitLine() から戻されるディスパッチポインタを
// DispatchDriver オブジェクト
m_FitLine.AttachDispatch(m_Bivar.FitLine()); にアタッチする。

// 別のあてはめを実行する。これらのあてはめオブジェクトは、
// オートメーションをサポートしているが、ここでは行っていない
m_Bivar.FitPolynomial(3.0);
m_Bivar.FitSpline(1000.0);
```

```
// 最初の FitLine オブジェクトを操作する
m_FitLine.ConfidenceFit(TRUE);
m_FitLine.ConfidenceIndividual(TRUE);
```

Visual C# を使った JMP のオートメーション

ここでは、Visual Studio 2008 以降を使用することを前提とします。

JMP アプリケーションの起動

C# を使った JMP のオートメーションでは、まず JMP を起動します。JMP を起動するには、あらかじめ、リソースに JMP ライブラリを追加しておく必要があります。JMP に用意されているライブラリを利用すると、ユーザは、JMP で使用するプロパティやメソッドを、オートメーションコントローラを使って表示できます。

JMP ライブラリを追加するには、次の手順に従います。

1. [プロジェクト] > [参照の追加] を選択し、[COM] タブをクリックします。
2. 下の方へスクロールし、「JMP」を選んで [OK] をクリックします。
3. JMP が追加されたことを確認するには、[表示] > [オブジェクトブラウザ] を選択します。
「Interop.JMP」があることを確認してください。
4. これでタイプライブラリの情報をアクセスできるようになり、JMP を開くためのコードが書けます。C# のプロジェクトのグローバル宣言で、*JMP.Application* タイプの変数を作成します。

```
private JMP.Application myJMP;
```

5. JMP のセッションを作成して、セッションを表示します。

```
myJMP = new JMP.Application();
myJMP.Visible = true;
```

分析の開始

この例では、「Big Class.jmp」サンプルデータで二変量の分析を行う方法を説明します。

分析を開始するには、次の手順に従います。

1. ドキュメントの変数を作成します。

```
JMP.Document doc;
```

2. 「Big Class.jmp」サンプルデータを開きます。

```
doc = myJMP.OpenDocument("c:\Program Files\SAS\JMP\17\Samples\Data\Big
Class.jmp");
```

3. 二変量の分析の変数を定義します。

```
JMP.Bivariate biv;
```

4. 二変量オブジェクトを作成します。

```
biv = doc.CreateBivariate();
```

- 必要な列と値を追加して、使用するデータを指定します。二変量の場合、プラットフォームを起動して、*x* と *y* の値を指定します。

```
biv.LaunchAddX("weight");
biv.LaunchAddY("height");
```

- 「二変量」プラットフォームを起動します。

```
biv.Launch()
```

- 直線のあてはめのオブジェクトを作成します。

```
JMP.Fit fLine;
fLine = biv.FitLine();
```

ここで、`FitLine` は `Fit` 型のオブジェクトを生成します。このオブジェクトは固有のメソッドとプロパティを持ち、それを操作することができます。

- 次の `fline.function(Boolean)` 関数を使用して、Confidence Fit、Line of Fit、Plot Residuals などのあてはめ線オプションを表示します。

```
fLine.ConfidenceFit(true);
fLine.LineOfFit(true);
fLine.PlotResiduals(true);
```

このような方法で、JMP のプラットフォーム、データテーブル、およびデータアクセス機能の多くを操作できます。49 ページの [「Windows で JMP のオートメーションを行うための Application オブジェクトリファレンス」](#) では、JMP のすべてのメソッドおよびプロパティについて説明しています。

特定のバージョンの JMP を起動

Windows オートメーションでは、あるアプリケーションのうち特定のバージョンのものを選択して起動するということができません。複数のバージョンの JMP がインストールされている場合の動作は次のようになります。

- JMP が起動していない場合、オートメーションプログラムはファイルの関連付けが設定されているバージョンの JMP を起動します。
- 特定のバージョンの JMP に明示的にファイルの関連付けを設定していない限り、最後にインストールされたバージョンの JMP が使用されます。
- すでにいずれかのバージョンの JMP が起動している場合、オートメーションプログラムは、そのバージョンにかかわらず動作中の JMP のインスタンスを使用します。

WindowsでJMPのオートメーションを行うためのApplicationオブジェクトリファレンス

次のセクション以降で、Visual Basic、Visual C++、Visual C# といったクライアントから Windows 版 JMP を操作するためのメソッドやプロパティを説明します。

定数

定数と関連するメソッドを示します。

Bivariate オブジェクトの定数

bivarFitTransformConstants

この定数を使うメソッド

`Bivariate.FitTransformed()`

値

- None
- Log
- Sqrt
- Square
- Reciprocal
- Exp

bivarOrthogonalFitConstants

この定数を使うメソッド

`Bivariate.FitOrthogonal()`

値

- Estimated Variances
- Equal Variances
- Fit Y to X
- Specified Variance Ratio

fitLoessLambdaConstants

この定数を使うメソッド

`Bivariate.FitLoessWithParms()`

値

- Linear
- Quadratic

Chart オブジェクトの定数

Chart プラットフォームは廃止されました。Chart プラットフォームを呼び出すオートメーションのコードは、Graph Builder（グラフビルダー）を使用してグラフを作成します。このため、以前のバージョンとグラフの外観が異なる場合があります。

chartChartTypeConstants

この定数を使うメソッド

`Chart.SpecifyType()`

値

Bar
Line
Needle
Point
Pie

chartOrientConstants

この定数を使うメソッド

`Chart.Orientation()`

値

Horizontal
Vertical

chartStatConstants

この定数を使うメソッド

`Chart.LaunchAddY()`

値

Data
N
% of Total
N Missing
Min
Max
SumWgt
Sum
Mean
Standard Deviation
Standard Error
Median
Range
Quantiles
Variance
CV

Cluster オブジェクトの定数

clusterColormapConstants

この定数を使うメソッド

`Cluster.ColorMap()`

値

- No Map
- Green to Black to Red
- Green to White to Red
- White to Black
- Blue to Gray to Red
- Blue to Green to Red
- Spectral

clusterDistanceConstants

この定数を使うメソッド

`Cluster.LaunchSpecifyDistanceFormula()`

値

- Average
- Centroid
- Ward
- Single
- Complete

clusterOrientationConstants

この定数を使うメソッド

`Cluster.SetOrientation()`

値

- Left
- Right
- Top
- Bottom

列の定数

colDataSourceConstants

この定数を使うメソッド

`Column.GetDataSource()`

値

- Data
- Formula

colDataTypeConstants

この定数を使うメソッド

```
TextImport.SetColumnType()  
Column.DataType  
DataTable.NewColumn()
```

値

```
Unknown  
Numeric  
Character  
RowState
```

colFormatConstants

この定数を使うメソッド

```
AxisBoxFormat  
AxisBoxScale  
Column.OutputFormat  
Column.InputFormat
```

値

```
Best  
Short  
Long  
Abbrev  
Date/Hr/Min  
Date/Hr/Min/Sec  
Days/Hrs/Mins  
Days/Hrs/Mins/Secs  
MMDDYYYY  
MM/YYYY  
DD/MM/YYYY  
DDMMYYYY  
DDMonYYYY  
DD/MM/YYYY HH:MM  
DD/MM/YYYY HH:MM:SS  
YYYY/MM/DD  
YYYYMMDD  
H:M:S  
MonDDYYYY  
MonDDYYYY H:M  
MonDDYYYY H:M:S  
DDMonYYYY H:M  
DDMonYYYY H:M:S  
YYYY/MM/DD H:M  
YYYY/MM/DD H:M:S  
MM/DD/YYYY H:M  
MM/DD/YYYY H:M:S
```

colModelTypeConstants

この定数を使うメソッド

```
DataTable.NewColumn()
```

`Column.ModelType`

値

Continuous
Ordinal
Nominal

colReorderConstants

この定数を使うメソッド

`DataTable.ReorderColumns()`

値

Original
By Name
By Datatype
By Modeling Type
Reverse

colRoleConstants

この定数を使うメソッド

`Column.SetRole()`

値

None
X
Y
Weight
Freq

colValidationConstants

この定数を使うメソッド

`Column.GetValidation()`

値

Unknown
None
List
Range

ControlChart オブジェクトの定数

jmpControlChartAlarms

この定数を使うメソッド

`ControlChart.SetAlarm()`

値

Write
Speak

Write with Explanation
Speak with Explanation

jmpControlChartConstants

この定数を使うメソッド

ControlChart.LaunchSetChartType()

値

Variable
IR
P
N
C
U
UWMA
EWMA
Cusum
LeveyJennings
Presummarize

jmpControlChartRules

この定数を使うメソッド

ControlChart.WestgardRule()

値

All Rules
Rule 1 2S
Rule 1 3S
Rule 2 2S
Rule R 4S
Rule 4 1S
Rule 10 X

DataTable オブジェクトの定数

dtJoinConstants

この定数を使うメソッド

DataTable.Join()

値

By Row Number
Cartesian
Matching Columns

dtSummaryStatConstants

この定数を使うメソッド

DataTable.AddToSummaryStatList()

値

```
Data
N
% Of Total
N Missing
Min
Max
Sum Wgt
Sum
Mean
Variance
Std Dev
Std Err
Median
Range
Quantiles
CV
```

summaryStatColNameConstants

この定数を使うメソッド

```
DataTable.SummarySetStatColumnFormat()
```

値

```
StatColumn Format
Column Format
Stat of Column Format
Column Stat Format
```

Discriminant オブジェクトの定数

discrimCanonicalOptions

この定数を使うメソッド

```
Discriminant.CanonicalOptions()
```

値

```
Show points
Show ellipses
Show rays
Show contours
Show details
Save canonical scores
Color points
```

discrimScoreOptions

この定数を使うメソッド

```
Discriminant.ScoreOptions()
```

値

```
Show interesting rows
Show all distances
```

Show all probabilities
Show classification counts
Select misclassified rows
Save formulas

discrimPriorsOptions

この定数を使うメソッド

`Discriminant.SpecifyPoints()`

値

Equal Probabilities
Proportional to Occurrence

Distribution オブジェクトの定数

distributionFitQuantilePlotConstants

あてはめが行われた後で分位点プロットに追加できる定数です。

この定数を使うメソッド

`DistribFit.QuantilePlotAction()`

値

Rotate
Confidence Limits
Line Of Fit
Mean Reference Line
Probability Labels

distributionSaveConstants

「一変量の分布」プラットフォームで保存できる定数です。

この定数を使うメソッド

`Distribution.Save()`

値

Level Numbers
Level Midpoints
Ranks
Ranks Averaged
Prob Scores
Normal Quantiles
Standardized
Spec Limits

fitDistribConstants

この定数を使うメソッド

`Distribution.FitDistribution()`

値

```
Normal
Log Normal
Weibull
Weibull With Threshold
Extreme Value
Exponential
Gamma
Beta
Poisson
SmoothCurve
GammaPoisson
GLog
```

DOECustom オブジェクトの定数

doeChangeDifficultyConstants

この定数を使うメソッド

```
DOECustom.SpecifyChangeDifficulty()
```

値

```
Easy
Hard
```

doeFactorTypes

この定数を使うメソッド

```
DOECustom.AddFactor()
```

値

```
Continuous
Categorical
Mixture
```

doeModelTypes

この定数を使うメソッド

```
DOECustom.MakeModel()
```

値

```
Linear
Interactions
RSM
```

doeOptimalityConstants

この定数を使うメソッド

```
DOECustom.OptimalityCriterion()
```

値

```
Recommended
```

D-Optimal
I-Optimal

doeResponseType

この定数を使うメソッド

DOECustom.AddResponse()

値

Maximize
Match Target
Minimize
None

FitModel オブジェクトの定数

fitModelDistributionConstants

この定数を使うメソッド

FitModel.LaunchSpecifyEmphasis()

値

Weibull
LogNormal
Exponential

fitModelEffectAttributeConstants

この定数を使うメソッド

FitModel.LaunchSpecifyAttributesForSelectedEffects()

値

Random Effect
Response Surface Effect
LogVariance Effect
Mixture Effect
Excluded Effect

fitModelEmphasisConstants

この定数を使うメソッド

FitModel.LaunchSpecifyEmphasis()

値

Effect Leverage
Effect Screening
Minimal Report

fitModelMacroEffectConstants

この定数を使うメソッド

FitModel.LaunchAddMacroEffect()

値

- Full Factorial
- Factorial to Degree
- Factorial Sorted
- Response Surface
- Mixture Response Surface
- Polynomial to Degree
- Scheffe Cubic

fitModelPersonalityConstants

この定数を使うメソッド

`FitModel.LaunchSpecifyPersonality()`

値

- Standard Least Squares
- Stepwise
- MANOVA
- Loglinear Variance
- 名義ロジスティック
- 順序ロジスティック
- 比例ハザード
- Parametric Survival

fitModelRandomEffectMethods

この定数を使うメソッド

`FitModel.LaunchSpecifyRandomEffectMethod()`

値

- REML - 推奨
- EMS - 従来

fitModelRowDiagConstants

この定数を使うメソッド

`FitModel.LaunchSpecifyAttributesForSelectedEffects()`

値

- Plot Actual by Predicted
- Plot Effect Leverage
- Plot Residual by Predicted
- Plot Residual by Row
- PRESS
- DurbinWatson

fitModelSaveColumnConstants

この定数を使うメソッド

`FitStepwise.SetDirection()`

値

- Prediction Formula
- Predicted Values
- Residuals
- Mean Confidence Interval
- Individual Confidence Interval
- Studentized Residuals
- Hats
- Standard Error of Predicted
- Standard Error of Residual
- Standard Error of Individual
- Effect Leverage Pairs
- Cook' s D Influence
- Standard Error of Predicted Formula

fitModelTransforms

この定数を使うメソッド

- FitModel.LaunchAddXEffectWithTransform()
- FitModel.LaunchAddYWithTransform()

値

- No Transform
- Log
- Square Root
- Square
- Recip
- Exponential
- Arrhenius
- Arrhenius Inverse

fitStepDirectionConstants

この定数を使うメソッド

- FitStepwise.SetDirection()

値

- Forward
- Backward
- Mixed

fitStepRulesConstants

この定数を使うメソッド

- FitStepwise.SetRules()

値

- Combine
- Restrict
- No Rules
- Whole Effect

ItemAnalysis オブジェクトの定数

itemAnalysisModelConstants

この定数を使うメソッド

ItemAnalysis.LaunchSpecifyModel()

値

Logistic 1PL
Logistic 2PL
Logistic 3PL

JMP の定数

アプリケーションレベルのメソッドやプラットフォームのメソッドに使用できます。

axisBooleanConstants

この定数を使うメソッド

AnalysisPlatform.AxisBoxBooleanOption()

値

Show Major Ticks
Show Minor Ticks
Show Major Grid Lines
Show Minor Grid Lines
Show Labels
Rotate Labels

axisIntervalConstants

この定数を使うメソッド

AnalysisPlatform.AxisBoxInterval

値

Numeric
Year
Month
Week
Day
Hour
Minute
Second

axisLineRefConstants

この定数を使うメソッド

AnalysisPlatform.AxisBoxAddRefLine()

値

Solid
Dashed

Dotted

axisNumericOptionConstants

この定数を使うメソッド

AnalysisPlatform.AxisBoxNumericOption()

値

Axis Minimum
Axis Maximum
Number of Minor Ticks
Increment between Ticks

axisScaleConstants

この定数を使うメソッド

AnalysisPlatform.AxisBoxScale()

値

Linear
Log

commFlowControlConstants

この定数を使うメソッド

DataFeed.SetCommParms()

値

None
DTR/DSR
RTS/CTS
XOn/XOff

commParityConstants

この定数を使うメソッド

DataFeed.SetCommParms()

値

None
Even
Odd

frameMarkerSizes

この定数を使うメソッド

AnalysisPlatform.FrameBoxSetMarkerSize()

値

Dot
Small
Medium

Large
XL
XXL
XXXL

internetItemTypes

この定数を使うメソッド

`Application.InternetOpenItem()`

値

HTML
Edit HTML/Text
HTML with tags stripped
JMP Table
JMP Table from HTML
Run JSL file on Web

jmpColorConstants

この定数を使うメソッド

`AnalysisPlatform.AxisBoxAddRefLine()`
`AnalysisPlatform.FrameBoxSetBackColor()`
`Chart.OverlayColor()`
`ControlCharts.ConnectColor()`
`ControlCharts.CenterColor()`
`ControlCharts.LimitsColor()`
`SurfacePlot.SetItemColor()`

値

Black
Red
Green
Blue
Orange
Purple
Yellow
Magenta

jmpGraphicsFormats

この定数を使うメソッド

`Journal.SaveAsHTML()`
`Journal.SaveAsRTF()`
`AnalysisPlatform.SaveGraphicOutputAs()`
`AnalysisPlatform.SaveGraphicItem()`

値

PNG Format
JPEG Format
Windows Metafile

jmpMarkerConstants

この定数を使うメソッド

Overlay.YOverlayMarker()

値

```
Dot
Plus
X
Hollow Square
Diamond
Triangle
Y
Z
Hollow Circle
Hollow Flat Rectangle
Hollow Tall Rectangle
Star
Solid Circle
Solid Flat Rectangle
Solid Tall Rectangle
Solid Square
```

jmpScriptConstants

この定数を使うメソッド

AnalysisPlatform.ScriptAction()

メモ : Save To File はサポート対象外になりました。定数値の 2 は無効です。

値

```
Redo Analysis
Save To File
Save To Data Table
Save To Report
Save To Window
```

jmpWindowTypeConstants

この定数を使うメソッド

Application.CloseWindowsOfType()

値

```
Datatables
Reports
Journals
JSL Output
Scripts
```

nomAxisActions

この定数を使うメソッド

nomAxisBooleanOption()

値

Rotate Ticks
Dividers
Lower Frame

printOrientConstants

この定数を使うメソッド

AnalysisPlatform.SetPrintOrientation()

値

Portrait
Landscape

Neural オブジェクトの定数

neuralControlConstants

この定数を使うメソッド

Neural.ControlPanelOptions()

値

Log the Tours
Log the Iterations
Log the estimates
Save the iterations

Oneway オブジェクトの定数

OnewayCompareConstants

一元配置分析での4つの多重比較方法を表す定数です。

この定数を使うメソッド

Oneway.CompareMeans()

値

Each Pair
All Pairs
With Best
With Control, Dunnett's

OnewayDisplayConstants

「一元配置」レポートでオン／オフを切り替えられる定数です。

この定数を使うメソッド

Oneway.DisplayOptions()

値

All Graphs
Points

```
Quantile Boxes
Means Diamonds
Means Dots, Error Bars
Grand Mean
Standard Deviation Lines
Comparison Circles
Connect Means
X Axis Proportional
Jitter
Matching Lines
Quantile Fit Lines
V Axis
H Axis
Mean Lines
Mean CI Lines
Mean of Means
Points Spread
```

OnewayNonParConstants

一元配置分析での3つのノンパラメトリックな検定を表す定数です。

この定数を使うメソッド

```
Oneway.Nonparametric()
```

値

```
Wilcoxon
Median
van der Waerden
```

OnewaySaveConstants

「一元配置」レポートの値を保存するための3つのオプションです。Template と Normal Quantiles は同じオプションです。Template は以前から使用していたオプションですが、追加された Normal Quantiles オプションは、プラットフォームで使用される名称と一致します。

この定数を使うメソッド

```
Oneway.Save()
```

値

```
Centered
Standardized
Template
Normal Quantiles
```

Overlay オブジェクトの定数

Overlay プラットフォームは廃止されました。Overlay プラットフォームを呼び出すオートメーションのコードは、Graph Builder (グラフビルダー) を使用してグラフを作成します。このため、以前のバージョンとグラフの外観が異なる場合があります。

overlayLineStyleConstants

この定数を使うメソッド

`Overlay.LineOptions()`

値

`Solid`
`Dotted`
`Dashed`
`Dash Dot`
`Dash Dot Dot`

overlayLineThicknessConstants

この定数を使うメソッド

`Overlay.LineOptions()`

値

`Regular`
`Thicker`
`Thickest`

Partition オブジェクトの定数

partitionCriterionConstants

この定数を使うメソッド

`Partition.Criterion()`

値

`Maximize Split Statistic`
`Maximize Significance`

partitionDisplayConstants

この定数を使うメソッド

`Partition.DisplayOptions()`

値

`Show Points`
`Show Tree`
`Show Graph`
`Show Split Stats`
`Show Split Candidates`
`Sort Split Candidates`
`Show Split Bar`
`Show Split Probability`

partitionMissingConstants

この定数を使うメソッド

`Partition.MissingValueRule()`

値

Closest
Random

partitionSaveColumnsConstants

この定数を使うメソッド

Partition.SaveColumns()

値

Save Residuals
Save Predicteds
Save Leaf Numbers
Save Leaf Labels
Save Predicted Formula
Save Leaf Number Formula
Save Leaf Label Formula

行の定数

rowStateConstants

この定数を使うメソッド

DataTable.GetNumberOfRowsByRowState()

値

Selected
Hidden
Excluded
Labeled

rowSelectWhereHow

この定数を使うメソッド

Datatable.SelectRowsWhere()

値

Clear Previous Selection
Extend Current Selection
Select From Within Current Selection

rowSelectWhereOperations

この定数を使うメソッド

Datatable.SelectRowsWhere()

値

Equals
Not Equals
Greater Than
Greater Than or Equals
Less Than

Less Than or Equals
Contains
Does Not Contain

ScatterplotMatrixPlatform オブジェクトの定数

scatterMatrixFormatConstants

この定数を使うメソッド

ScatterplotMatrixPlatform.LaunchSpecifyMatrixFormat()

値

Lower Triangular
上三角
Square

SurfacePlot オブジェクトの定数

surfaceColorConstants

この定数を使うメソッド

SurfacePlot.SetItemColor()

値

Grid Color
Mesh Color
Axis Color
Value Color
Name Color
Contour Color

surfaceDisplayConstants

この定数を使うメソッド

SurfacePlot.DisplayOptions()

値

Show X Axis
Show Y Axis
Show Z Axis
Show X Value
Show Y Value
Show Z Value
Show X Name
Show Y Name
Show Z Name
Show X Grid
Show Y Grid
Show Z Grid
Show Lights Border
Show Control Panel
Show Surface
Show Mesh

Show Contour
Lock Z Scale
Show Data Points

Survival オブジェクトの定数

competingCauseConstants

この定数を使うメソッド

Survival.CompetingCauseAction()

値

Omit Causes
Save Cause Coordinates
Weibull Lines
Hazard Plot

TextImport オブジェクトの定数

jmpTIEndOfFieldConstants

この定数を使うメソッド

TextImport.SetEndOfFieldOptions()

値

Tab
Space
Spaces
Comma

jmpTIEndOfLineConstants

この定数を使うメソッド

TextImport.SetEndOfLineOptions()

値

Carriage Return+Line Feed
Carriage Return
Line Feed
Semicolon

TimeSeries オブジェクトの定数

timeSeriesConstraintConstants

この定数を使うメソッド

TimeSeries.SmoothingModel()

値

ZeroToOne
Unconstrained

Stable

timeSeriesModelConstants

この定数を使うメソッド

`TimeSeries.SmoothingModel()`

値

- Simple Exponential
- Double Exponential
- Linear Exponential
- Damped Trend
- Seasonal Exponential
- Winters Method

Variability オブジェクトの定数

varVarianceComponentConstants

この定数を使うメソッド

`Variability.VarianceComponents()`

値

- Nested
- Crossed
- Crossed then Nested
- Nested then Crossed

Application オブジェクト

Application オブジェクトを使うと、JMP の実行やデータテーブル等のファイルの読み込みに関して、高レベルの処理ができます。オートメーションセッションには、この Application オブジェクトが不可欠です。

プロパティ

Application

JMP オブジェクトへのディスパッチポインタを戻す。プロパティにアクセスするには、このオブジェクトが先に作成されている必要があります。

FullName

JMP アプリケーションの名前をストリングで戻す（たとえば、"JMP"）。

Name

Name は、FullName と同様に、JMP アプリケーションの名前をストリングで戻す。

Parent

1つ上のレベルのオブジェクトを戻す。Application オブジェクトは最上位のレベルにあるので、その場合は Application オブジェクトを戻します。

Visible

`True(1)` ならば JMP セッションを表示し、`False(0)` ならば表示しない。デフォルトは `False` です。

`Visible` を `True` に変更した場合、変更前に開いていて非表示となっていたウィンドウも含め、すべてのウィンドウが表示されるようになります。

メソッド

ClearLog()

ログウィンドウの内容をクリアする。

CloseAllWindows()

現在開いているウィンドウをすべて閉じる。

CloseWindow()As Boolean

JMP が終了するのを待たずに、分析ウィンドウをただちに閉じる。処理に成功したときは `True`、失敗したときは `False` を戻します。

CloseWindowsOfType(jmpWindowTypeConstants windowType)

`Journal` や `Datatable` など指定された種類のウィンドウで、現在開いているウィンドウをすべて閉じる。`jmpWindowTypeConstants` には、閉じることのできるウィンドウの種類を指定します。

CreateDOECustom () As DOECustom

`DOE Custom Design` オブジェクトを作成する。このオブジェクトは、因子の追加、モデルの追加、計画の作成を行うためのメソッドで使用され、テーブルが作成されます。詳細については、『実験計画法（DOE）』のオートメーションに関する節を参照してください。

CreateTextImportObject(FileName As String, NumberColumns as Integer) As TextImport

`TextImport` オブジェクトを作成する。その後、このオブジェクトに行と列の情報を設定する必要があります。`FileName` は読み込まれるファイルの完全パスで、`NumberColumns` はデータ内の列の数です。新しい `TextImport` オブジェクトへのディスパッチポインタを戻します。

EnableInteractiveMode(Flag as Boolean)

オートメーションプロセスの間、ログではなくメッセージボックスに情報を表示できる。このオプションでは、バッチモードの処理が事実上オフになります。

GetLogContents() As String

ログウィンドウの現在の内容をストリングとして戻す。ログウィンドウは、分離していても固定されていてもかまいませんが、表示されていない場合は、空のストリングが戻されます。

GetJSValue

Integer、Double、String、またはList（前の3つの値タイプが混在）のJSLグローバル変数の値を取得する。戻り値はVARIANT（Integer、Double、またはString）またはVARIANTの配列になります。

メソッド宣言：

```
GetJSValue (VariableName As String) As Variant
```

Visual Basicでの一般的なコール：

```
result = GetJSValue("MyJSLVariable")
```

結果にアクセスする方法は戻されるタイプによって異なりますが、次のようにになります。

```
A = B * result;
```

値が文字列の場合は

```
MsgBox(result)
```

リストが戻され、その3番目の要素が必要な場合は次のように指定します。

```
MsgBox(result(2))
```

GetNumberOfAutomationDatatables()As Integer

JMP内で現在開いている（表示されている）データテーブルの数を戻す。

GetRunCommandErrorString() As String

RunCommand(Command As String)またはRunJSLFile(FileName As String)の実行後に、JSLエラーテキストを取得する。

HasRunCommandErrorString() As Booleanも参照してください。

これらのメソッドにアクセスするVisual Basicのコードの例：

```
MyJMP.RunCommand (Text1.Text)
If (MyJMP.HasRunCommandErrorString) Then
    MsgBox (MyJMP.GetRunCommandErrorString)
EndIf
```

GetTableHandleFromIndex(Integer Index)

データテーブルの番号（Integer Index）が整数で与えられたとき、そのDataTableオートメーションオブジェクトへのハンドルを戻す。データテーブル番号の範囲は、1からJMP内で開いているデータテーブルの数までです。

GetTableHandleFromName(Name as String)As DataTable

データテーブルを名前で検索し、そのテーブルのオートメーションオブジェクトへのハンドルを戻す。

GetTableNameFromIndex(Integer Index)

データテーブル番号 (Integer Index) が整数で与えられたとき、データテーブルの名前を文字列として戻す。データテーブル番号は、1 から JMP で開いているデータテーブルの数までです。

HasRunCommandErrorString() As Boolean

エラーテキストの有無を問い合わせる。これを使用すると、ストリングが空であるかどうかを確かめる必要がありません。

`GetRunCommandErrorString() As String` も参照してください。

これらのメソッドにアクセスする Visual Basic のコードの例：

```
MyJMP.RunCommand (Text1.Text)
If (MyJMP.HasRunCommandErrorString) Then
    MsgBox (MyJMP.GetRunCommandErrorString)
EndIf
```

HonorSessionSavePref(Flag as Boolean)

オートメーションでは、特定の状況が頻繁に再現され、セッションを保存すると混乱が生じることがあるため、シャットダウン時にセッションが保存されません。オートメーションにおいてシャットダウン時にセッションを保存するようにするには、`Quit` メソッドを呼び出す前に、`True` パラメータを指定してこのメソッドを呼び出します。すると、JMP のセッション保存に関する環境設定が有効になります。`False` パラメータを指定すると、セッションは保存されません。

InternetOpenItem(String URL, internetItemTypes openHow) As DataTable

テキストまたはバイナリファイルを開く。HTML ファイルをテキストエディタで開く、HTML ファイルをタグなしのテキストとして開く、バイナリ JMP ファイルを開く、TABLE タグ (TABEL、TD、TR など) を含む HTML ファイルを JMP データテーブルとして開く、といったオプションがあります。2 番目のパラメータによってアクションが決まります。後の 2 つのオプションは、メソッドの呼び出しが成功した場合は JMP データテーブルオートメーションオブジェクトへのポインタ、そうでない場合は NULL が戻されます。前の 2 つのオプションを使用したときは、常に NULL が戻されます。

InternetOpenTextAsData(URL As String) As DataTable

指定された URL のテキストファイルを JMP スクリプトウィンドウ内で開き、そのテキストを JMP データテーブルとして読み込む。テキストの読み込み時には、テキストの読み込みに関する現在の設定が使われます。メソッドが正常に実行された場合は、`DataTable` オートメーションオブジェクトのメソッドで操作できる JMP データテーブルへのポインタが戻されます。エラーが発生した場合は、NULL が戻されます。

コードの例：

```
Set DT = MyJMP.InternetOpenTextAsData("www.sas.com/MyData/data.txt")
Dim Doc As JMP.Document
Set Doc = DT.Document
Doc.SaveAs ("c:\myData.jmp")
```

NewDatabaseObject() As AUTODB

AUTODB 型のオブジェクトを生成する。このオブジェクトはデータへの ODBC アクセスを制御するためを使われます。

NewDataFeed() As DataFeed

シリアルポートに接続された外部機器からのデータを取るデータフィードオブジェクトを生成する。

NewDataTable(FileName As String) As DataTable

新しい JMP データテーブルを作成し、さらにそれを制御できるようにオブジェクトを戻す。

OpenDocument(FileName As String) As Document

JMP データテーブルをドキュメントとして開く。OpenDocument(FileName As String) As Document は、オートメーションアプリケーションでドキュメントにアクセスするための標準的なルーチンで、JMP でもサポートされています。データテーブルを読み込んだ後、その内容を操作するには、ドキュメントメソッドの GetDataTable を呼び出して、DataTable オブジェクトを取得します。DataTable オブジェクトは、中身を変更できるオブジェクトです。

Quit()

他に JMP を使っているオートメーションアプリケーションがなければ、JMP を終了する。JMP を使用中のアプリケーションがあれば、使用カウント数を減らします。

RunCommand(Command As String)

ストリングの形式で与えられた JSL のコマンドを実行する。

RunJSLFile(FileName As String)

与えられた有効なパス名の JSL テキストファイルを読み込み、これを JMP 内で実行する。

SetCurrentDirectory(DirName As String) As Boolean

JMP 内の現在のディレクトリを設定する。これにより、他のメソッドで相対パスのファイル名を使えるようになります。このメソッドは、既存のオートメーションクライアントコードが JMP インストールディレクトリの使用を前提としている場合は、注意して使う必要があります。

ShowLogHonorPreferences()

ShowLog() から派生したメソッド。オートメーションの場合、通常、ログは表示されません。ただし、JMP の環境設定で、JMP の起動時にログウィンドウが開く設定になっている場合は、この関数を呼び出すことでログが表示されます。環境設定では、その他ログウィンドウが開くタイミングとして、ユーザが明示的に開いたとき、およびテキストが追加されたときの 2 つがあります。

ShowStartupWindow()

JMP の起動ウィンドウ（通常はホームウィンドウ）を開く。ホームウィンドウがすでに開いている場合は、最前面に移動します。

ShowLog() As Boolean

ログウィンドウを開く。ログウィンドウがすでに開いている場合は、何も起こりません。ログが使用できる場合は `True` を、そうでない場合は `False` を戻します。

AUTODB オブジェクト

AUTODB オブジェクトを使うと、ODBCを使った外部データへのアクセスが可能になります。テーブルを操作するには、SQL の知識がある程度必要です。

メソッド

これらのメソッドを使用するには、SQL に関する知識が必要です。

Connect(ConnectInformation As String)

最終的に `SQLDriverConnect` をコールするときに使われる接続情報のストリングを含める。例：

```
DSN=oracledata;DBQ=data_o7555;UID=UserID; pwd=userPassword
```

他の例については、オートメーションのコード例を参照してください。

Disconnect() As Boolean

接続を切断する。

ExecuteSql(SQLStatement As String) As Boolean

SQL のステートメントを実行し、成功または失敗を知らせる布尔値を戻す。

メモ：この SQL ステートメントに、レコードセットを戻す `Select` ステートメントを使用することはできません。`Select` ステートメントを送るには、`ExecuteSqlSelect(SQLSelectStatement As String) As DataTable` を使用します。

ExecuteSqlSelect(SQLSelectStatement As String) As DataTable

SQL の `Select` ステートメントを実行し、新たにロードされたテーブルへのディスパッチポインタを戻す。

メモ：この SQL ステートメントには、必ずレコードセットを戻す `Select` ステートメントを使用します。その他の SQL コマンドには、`ExecuteSql(SQLStatement As String) As Boolean` を使用してください。

GetLastError()

最後に実行された `Connect` または `ExecuteSQL` コールのエラーコードを戻す。

OpenTable(TableName As String) As DataTable

データテーブルを開き、データを保持するために作成された JMP テーブルへのディスパッチポインタを戻す。

SaveTable(TablePtr As DataTable, FileName As String)

JMP データテーブルへのディスパッチポインタが与えられたとき、このテーブルを指定された名前でデータベースに保存する。

グラフ軸の共通メソッド

グラフの軸の設定を制御します。

メソッド

AxisBoxAddLabel(Handle As Long, Label As String)As Boolean

Handle で与えられた軸のラベルを追加する。処理に成功したときは True を戻し、失敗したときは False を戻します。

AxisBoxAddRefLine(Handle As Long, Location As Double, Style As Short, Color As Short)As Boolean

Location で指定された位置に参照線を追加する。Style の値は axisLineRefConstants で、Color の値は jmpColorConstants で取得できます。

AxisBoxBooleanOptions(Handle As Long, Option As Short, Flag As Bool)As Boolean

軸のブール値オプションを設定するための標準呼び出しだけです。指定できるオプションは axisBooleanConstants に含まれ、Show Major Ticks (目盛りを表示)、Show Minor Ticks (補助目盛りを表示)、Show Major Grid (グリッドを表示)、Show Minor Grid (補助目盛りのグリッドを表示)、Show Labels (ラベルを表示)、および Rotated Labels (ラベルを回転) です。どのオプションでも、Flag の値が True ならばオンで、False ならばオフです。戻り値が True ならば成功、False ならば失敗を意味します。

AxisBoxFormat(Handle As Long, Format As Short)As Boolean

軸の目盛りの表示形式を指定する。表示形式の値は colFormatConstants にあり、Best、m/d/y などです。

AxisBoxInterval(Handle As Long, Interval As Short)As Boolean

Inc (目盛り間隔) 値に使われる単位を指定する。間隔の値は axisIntervalConstants にあり、Numeric (数値)、Hour (時間)、Day (日) などです。

AxisBoxNumberDecimals(Handle As Long, NumDecimals As Short)As Boolean

軸の値の表示形式について、小数の桁数を指定する。

AxisBoxNumericOption(Handle As Long, Option As Short, Number As Double)As Boolean

軸の数値オプションを指定するための標準呼び出しだけです。このオプションは `axisNumericOptionConstants` にあり、`Min` (最小値)、`Max` (最大値)、`Inc` (目盛り間隔)、および `Minor Ticks` (補助目盛り) です。

AxisBoxRemoveLabel(Handle As Long)As Boolean

軸のラベルを削除する。

AxisBoxRevertAxis(Handle As Long)As Boolean

軸を元の設定に戻す。戻り値が `True` ならば成功、`False` ならば失敗を意味します。

AxisBoxScale(Handle As Long, Scale As Short)As Boolean

軸のスケールを `Log` (対数) または `Linear` (線形) に変更する。`Scale` の値は `axisScaleConstants` から取得できます。

Column オブジェクト

`Column` オブジェクトを使うと、データテーブルの列を制御できます。`Column` オブジェクトのポインタは、`GetColumn` をコールして `DataTable` オブジェクトからメソッドを使って取得します。

プロパティ

DataType

列のデータタイプが、文字、数値、行の属性、不明のどれであるかを示すプロパティ。値は `colDataTypeConstants` の定義に含まれています。

FieldWidth

列の幅 (桁数) を整数値で示したプロパティ。

メモ: `FieldWidth` プロパティの値を指定するには、まず `OutputFormat` プロパティの値を指定しておく必要があります。`NumDecPlaces` プロパティも指定する場合は、まず `NumDecPlaces` を指定し、次に `FieldWidth` を指定する必要があります。

InputFormat

データの入力形式を示す整数値のプロパティ。これは通常、数値データの場合は `Best` (最適)、日付／時間データの場合は日付／時間形式のどれかです。値は `colFormatConstants` の定義に含まれています。`Long` 形式 (長い形式の日付表示) と `Abbrev` 形式 (簡略な形式の日付表示) は入力には無効なので、使用するとエラーが生じます。

OutputFormat

データの出力形式を整数値で示すプロパティ。これは通常、数値データの場合は **Best** (最適)、日付／時間データの場合は日付／時間形式のどれかです。値は **colFormatConstants** の定義に含まれています。

メモ: `Column.OutputFormat` プロパティを `JMP.colFormatConstants.colFormatCurrency` に設定すると、小数点以下の表示桁数が 2 に設定されます (US ドルの場合)。`Column.NumDecPlaces` に数値を設定している場合は、`OutputFormat` プロパティが無効になり、`Column.OutputFormat` は `colFormatFixedDecimal` に設定されます。つまり、いずれかのプロパティを設定すると他方が無効になるため、通貨値の小数点以下の表示桁数として異なる数値を設定することはできません。

Locked

列がロックされているかどうかを示すブール値 (True / False) をとるプロパティ。

ModelType

列の尺度 (連続、名義、または順序) を示す整数値のプロパティ。値は **colModelTypeConstants** の定義に含まれています。

Name

列の名前を示す文字列プロパティ。

NumberRows

列内の行の数を整数値で示すプロパティ。

NumDecPlaces

数値の列の値が小数点以下何位まで許されるかを示す整数値のプロパティ。

メモ: `NumDecPlaces` を指定するには、`OutputFormat` が `colFormatFixedDec` であることを指定する必要があります。`NumDecPlaces` プロパティは、先に `OutputFormat` プロパティを指定してから指定します。`FieldWidth` プロパティも指定する場合は、`OutputFormat` と `FieldWidth` の間に `NumDecPlaces` を指定する必要があります。

メソッド

AddFormula(JSLText As String)

JSL で記述された有効な計算式を列に追加する。

AddValueLabelToList(Value as String, Label as String) As Boolean

値ラベルをオートメーションリストに追加する。値ラベルは、`CommitValueLabels()` As Boolean が呼び出されるまで、確定 (追加) されません。そのため、確定前に、列に割り当てる値のラベルを複数定義しておくことができます。処理に成功したときは `True` を、失敗したときは `False` を戻します。

`CommitValueLabels()` As Boolean と `RemoveValueLabels()` As Boolean も参照してください。

Visual Basic でこれらのメソッドを使用する例：

’ オブジェクト DT にすでに指定してある「Big Class.jmp」を使用する

```
Dim ColSex As JMP.Column
Dim ColAge As JMP.Column

Set ColSex = DT.GetColumn("sex")
Set ColAge = DT.GetColumn("age")

ColSex.AddValueLabelToList "M", "男性"
ColSex.AddValueLabelToList "F", "女性"
ColSex.CommitValueLabels
ColAge.AddValueLabelToList "12", "Twelve"
ColAge.AddValueLabelToList "13", "Thirteen"
ColAge.AddValueLabelToList "14", "Fourteen"
ColAge.AddValueLabelToList "15", "Fifteen"
ColAge.AddValueLabelToList "16", "Sixteen"
ColAge.AddValueLabelToList "17", "Seventeen"
ColAge.CommitValueLabels
```

CommitValueLabels() As Boolean

前に AddValueLabelToList(Value as String, Label as String) As Boolean を使って列に割り当てたラベルを確定（追加）する。列は、CommitValueLabels が呼び出された後で、新しいラベルで更新されます。処理に成功したときは True を、失敗したときは False を戻します。

AddValueLabelToList(Value as String, Label as String) As Boolean と
RemoveValueLabels() As Boolean も参照してください。

Visual Basic でこれらのメソッドを使用する例：

’ オブジェクト DT にすでに指定してある「Big Class.jmp」を使用する

```
Dim ColSex As JMP.Column
Dim ColAge As JMP.Column

Set ColSex = DT.GetColumn("sex")
Set ColAge = DT.GetColumn("age")

ColSex.AddValueLabelToList "M", "男性"
ColSex.AddValueLabelToList "F", "女性"
ColSex.CommitValueLabels
ColAge.AddValueLabelToList "12", "Twelve"
ColAge.AddValueLabelToList "13", "Thirteen"
ColAge.AddValueLabelToList "14", "Fourteen"
ColAge.AddValueLabelToList "15", "Fifteen"
ColAge.AddValueLabelToList "16", "Sixteen"
ColAge.AddValueLabelToList "17", "Seventeen"
ColAge.CommitValueLabels
```

Exclude() As Boolean

列を除外する。この操作はトグルで制御されます。つまり、このメソッドを 1 回コールすれば設定が有効になり、2 回目にコールしたときには設定が解除されます。戻り値が `True` ならば成功、`False` ならば失敗を意味します。

GetCellVal(RowNumber As Integer) As String

指定された行の値をストリングとして戻す。値が数値である場合、呼び出し側によって値がストリングに変換される必要があります。Visual Basic では、受け取るオブジェクトが数値と判断されると自動的にストリングに変換されます。

GetDataSource() As Integer

列のデータが普通のデータであるか、機器からのデータであるか、また計算式を含んでいるかを確認する。`colDataSourceConstants` の定義に含まれている整数値を戻します。

GetDataVector() As Variant

列のすべての要素を含んだベクトルを可変要素として取得する。

Visual Basic での `GetDataVector` の使用例：

```
' 「名前」列のすべての名前を取得する。最初の 5 つの名前を
'messagebox に表示する
```

```
Dim Col As JMP.Column
Set Col = DT.GetColumn("name")
nameArray = Col.GetDataVector
For i = 0 To 4
    MsgBox (nameArray(i))
Next i
```

GetFormula() As String

列の計算式をストリングとして取得する。

GetRowStateVectorData

行の属性の条件に一致するデータを戻す。このメソッドを使うと、`GetRowStateVector` を実行してからそのインデックスを使って `SetCellVal(RowNumber As Integer, Value as String)` を呼び出す手間が省け、1 回の呼び出しでデータを取得できます。これにより、パフォーマンスが向上します。メソッド宣言は次のようになります。

```
GetRowStateVectorData(rowStateConstants state) As Variant
```

Visual Basic での `GetRowStateVectorData` の使用例：

```
Dim Col As JMP.Column
Set Col = DT.GetColumn("height")
myarray = Col.GetRowStateVectorData(rowStateSelected) ' 選択した行番号のベクトルを取得する
nElements = UBound(myarray)      ' 戻される配列の上限を取得する
```

```
For i = 0 To nElements      ' 選択した行の名前を表示する
    MsgBox (myarray(i))
Next I
```

GetValidation() As Integer

列にリストチェックまたは範囲チェックがかけられているか、あるいはチェックがかけられていないかを確認する。`colValidationConstants` の定義に含まれている整数値を戻します。

InsertDataVector(Data As Variant, AfterRow As Long) As Boolean

列のタイプに基づき、`Double` または `String` の配列の値を、2番目のパラメータで指定された行の次の行以降に設定する。既存のデータがある場合は上書きされます。

たとえば、6～10行に既存のデータがある場合に、最初の引数に8つの値からなる配列を、2番目の引数に5を指定すると、6～10行は上書きされ、新たに3行が追加されます。

2番目のパラメータに0を指定すると、データは列の先頭から追加され、必要な数の行が作成されます。データを1行目から追加する場合は、代わりに `SetDataVector(Data As Variant) As Boolean` を使います。2番目のパラメータに-1を指定すると、データは最後の行の後に追加され、必要な数の行が作成されます。

Hide() As Boolean

列を非表示にする。この操作はトグルで制御されます。つまり、このメソッドを1回コールすれば設定が有効になり、2回目にコールしたときには設定が解除されます。戻り値が `True` ならば成功、`False` ならば失敗を意味します。

Label() As Boolean

列にラベルをつける。この操作はトグルで制御されます。つまり、このメソッドを1回コールすれば設定が有効になり、2回目にコールしたときには設定が解除されます。戻り値が `True` ならば成功、`False` ならば失敗を意味します。

RemoveValueLabels() As Boolean

列のすべての値ラベルを削除する。この操作は直ちに行われる所以、このメソッドが呼び出されるとすぐに列は元の値で表示されます。処理に成功したときは `True` を、失敗したときは `False` を戻します。

`CommitValueLabels() As Boolean` と `AddValueLabelToList(Value as String, Label as String) As Boolean` も参照してください。

ScrollLock() As Boolean

列のスクロールをロックする。この操作はトグルで制御されます。つまり、このメソッドを1回コールすれば設定が有効になり、2回目にコールしたときには設定が解除されます。戻り値が `True` ならば成功、`False` ならば失敗を意味します。

SelectCellMissing(Index as Integer)

指定された行のセルの値を欠測値に設定する。

SelectColumn(Flag as Boolean) As Boolean

Flag が True のときは列を選択、Flag が False のときは選択を解除します。戻り値が True ならば成功、False ならば失敗を意味します。

SetCellVal(RowNumber As Integer, Value as String)

指定された行の値を設定する。列の値が数値であるときは、値が数値に変換されます。

SetDataVector(Data As Variant) As Boolean

列のタイプに基づき、Double または String 配列の値を、1 行目から順に設定する。行が配列の要素数より少ない場合は、テーブルに行が追加されます。処理に成功したときは True を、失敗したときは False を戻します。

SelectCellMissing(Row As Integer)

指定された行番号のセルを欠測値にする。

SelectCurrencyType(Type As ColCurrencyConstants)

列の通貨の種類を設定する。

SelectRole(RoleType As Integer) As Boolean

colRoleConstants の値のどれかを使って列の役割を設定する。

プラットフォーム共通のメソッド

各分析プラットフォームやグラフ作成プラットフォームには、共通のメソッドがあります。そのため、以下のメソッドは、どの分析を実行しているときでもコールできます。

メソッド

プラットフォーム共通のメソッドは、どの分析プラットフォームおよびグラフ作成プラットフォームでも使用できます。

CreateJournal() As Journal

ジャーナルを作成し、そのジャーナルを表すオートメーションオブジェクトへのポインタを戻す。このオブジェクトは、後で出力をディスクに保存するときに使用できます。

CopyGraphicItem(Handle As Long) As Boolean

ハンドル (handle) で指定されたグラフィックのサブ項目を、メタファイルとしてクリップボードへコピーする。

DisplayBoxAppend(SrcHandle as Long, AppendHandle as Long) As Long

ディスプレイボックスを特定のディスプレイボックスの後ろに追加する。

DisplayBoxPrepend(SrcHandle as Long, AppendHandle as Long) As Long

ディスプレイボックスを特定のディスプレイボックスの前に追加する。

FrameBoxAddGraphicsScript(long handle, script As String) As Boolean

ハンドル (GetGraphicItemByType から戻される) で与えられたフレームボックスとグラフィックススクリプトを関連付ける。これは、グラフィックススクリプトを分析フレームに追加する対話形式のダイアログボックスに似ています。処理に成功したときは **True** を、失敗したときは **False** を戻します。

FrameBoxSetBackColor(long handle, jmpColorConstants color) As Boolean

ハンドル (GetGraphicItemByType から戻される) で与えられたフレームボックスの、背景色を設定する。**jmpColorConstants** が色の範囲を定義します。処理に成功したときは **True** を、失敗したときは **False** を戻します。

FrameBoxSetMarkerSize(long handle, frameMarkerSizes size) As Boolean

ハンドル (GetGraphicItemByType から戻される) で与えられたフレームボックスの、マーカーのサイズを設定する。**frameMarkerSizes** 定数がサイズ範囲を定義します。処理に成功したときは **True** を、失敗したときは **False** を戻します。

FrameBoxTransparency(alpha as Double) As Boolean

グラフ内の点の透明度のレベルを指定する。

GetGraphicItemByName(ItemName as String) As Long

JMP で表示されるサブ項目へのハンドルを戻す。サブ項目の例としては、二変量分析の「分散分析」表などが挙げられます。このハンドルは、以下に説明する他の操作でも使用できます。

GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long

項目のタイプに応じて、JMP で表示されるサブ項目へのハンドルを戻す。サブ項目のタイプには、**PictureBox** などがあります。PictureBox1 は、たとえば分析のプロットへのハンドルを戻します。このハンドルは、以下に説明する他の操作でも使用できます。

GetSubgraphicItemByName(Handle as Long, Name as String) As Long

Handle で参照されるディスプレイボックスの下層にあるディスプレイボックスへのハンドルを戻す。ディスプレイボックスのタイトルが戻されます。これは **GetGraphicItemByName(ItemName as String) As Long** に似ていますが、基準となるディスプレイボックスが異なり、**GetGraphicItemByName(ItemName as String) As Long** では、最上位にあるディスプレイボックスが基準に、**GetSubgraphicItemByName** では、ハンドルで参照されるディスプレイボックスが基準になります。

GetSubgraphicItemByType(Handle as Long, BoxName as String, BoxNumber as Short) As Long

Handleで参照されるディスプレイボックスの下層にあるディスプレイボックスへのハンドルを戻す。ディスプレイボックスのタイプと番号が戻されます。これは GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long に似ていますが、基準となるディスプレイボックスが異なり、GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long では、最上位にあるディスプレイボックスが基準に、GetSubgraphicItemByType では、ハンドルで参照されるディスプレイボックスが基準になります。

GetTextOfGraphicItem(Handle As Long) As String

グラフィックのサブ項目のテキストすべてをストリングとして戻す。これは、TableBoxes、TextBoxes、NumberColBoxes など、適当な（テキストだけを含む）サブ項目に対してだけ機能します。

JournalGraphicItem(Handle As Long) As Boolean

グラフィックのサブ項目を JMP ジャーナルにコピーする。

JournalOutput() As Boolean

現在の分析ウィンドウのジャーナルを作成する。それ以降の操作を行うためのオートメーションオブジェクトは戻しません。

Launch() As Boolean

LaunchAdd__ で指定された列に対し、分析を実行する。

LaunchAdd__(ColumnName As String) As Boolean

指定された列名を、特定の役割に追加する。__ は、X、Y、Freq（度数）などです（役割が X の場合は (LaunchAddX 度数) の場合は LaunchAddFreq など）。Launch ルーチンを実行する前に、必ず列を追加しておく必要があります。そうしないと、分析側ではどの列を使って分析を行えばよいのかがわかりません。

LaunchAddBy() As Boolean

By グループ処理に使われる列名を追加する。デフォルトでは、分析が起動されると、その分析またはグラフに固有の関数が最初の By グループの結果を処理します。処理に成功したときは True、失敗したときは False を戻します。

LaunchRemove__(ColumnName As String) As Boolean

指定された列を、分析用の役割のリストから削除する。__ は、X、Y、Freq（度数）などです（役割が X の場合は LaunchRemoveX、度数の場合は LaunchRemoveFreq など）。

OutlineBoxGetTitle(Handle as Long) As String

指定されたアウトラインボックスのタイトルをストリングとして戻す。

NumberColGetHeading(Handle As Integer)

ハンドルで指定された数値列ディスプレイボックスの見出しテキストを戻す。

NumberColGetItemText(Handle As Long, ElementNumber As Integer) As String

NumberColBoxes でのみ使われる。NumberColBox の *i* 番目の要素を取得します。*i* 番は 2 番目のパラメータで指定します。値はストリングとして戻されるため、数値演算で使うときは、数値に変換する必要があります。

NumberColHide, StringColHide(Handle As Integer, Flag As Boolean)

数値列ディスプレイボックスまたは文字列ディスプレイボックスの表示／非表示を切り替える。

NumberColSetHeading, StringColSetHeading(Handle As Integer, Title As String)

数値列ディスプレイボックスまたは文字列ディスプレイボックスの見出しテキストを設定し、そのボックスおよびテキストへのハンドルを渡す。

OutlineBoxSetTitle(Handle As Long, Title As String)

Handle で識別されたアウトラインボックスのタイトルを設定する。アウトラインボックスのハンドルは、あらかじめ GetGraphicItemByType を使って取得しておく必要があります。

PrintPages(From As Integer, To As Integer) As Boolean

レポートのうち、指定された範囲のページを印刷する。

PrintReport() As Boolean

分析レポート全体を印刷する。

SaveGraphicItem(Handle As Long, FileName As String, GraphicType As Integer)

グラフィックのサブ項目を、指定されたファイル名でディスク上に保存する。グラフィックの形式はパラメータ (GraphicType) で決まります。この値は jmpGraphicsFormats の定数から取得します。例は、JPEG、PNG、SVG、TIFF、WMF (Windows metafile) などです。

SaveGraphicOutputAs(FileName As String, GraphicFormat As Integer)

指定された FileName で、分析結果全体をファイルに保存する。グラフィックの形式は jmpGraphicsFormats の定数から取得します。例は、JPEG、PNG、SVG、TIFF、WMF (Windows metafile) などです。

SaveJournalAs(FileName As String) As Boolean

レポートのジャーナルファイルを生成し、指定された FileName でディスクに保存する。

ScriptAction(JSLText As String)

分析に対して JSL を発行する。

SetFrameSize(X As Integer, Y As Integer)

グラフのフレームサイズを、分析ウィンドウ全体に対比させて指定する。X と Y の値はピクセル座標です。

SetPrintOrientation(printOrientConstants orientation) As Boolean

渡されたパラメータに基づいて、分析結果を印刷するときの方向を **Landscape** または **Portrait** のどちらかに設定する。処理に成功したときは **True** を、失敗したときは **False** を戻します。

SetWindowPos(X As Integer, Y As Integer)

分析ウィンドウの位置を、Windows デスクトップからの相対的な位置として指定する。X と Y の値はピクセル座標です。

SetWindowSize(CX As Integer, CY As Integer)

分析ウィンドウのサイズをピクセル値で指定する。

StringColGetHeading(Handle As Integer)

ハンドルで指定された文字列ディスプレイボックスの見出しテキストを戻す。

StringColGetItemText(Handle As Long, ElementNumber As Integer) As String

StringColBoxes でのみ使われる。StringColBox の *i* 番目の要素を取得します。*i* は 2 番目のパラメータで指定します。項目はストリングとして戻されます。

TableBoxMakeDataTable(Handle As Long) As DataTable

新しいデータテーブルを作成し、Handle で記述された TableBox の内容をセルに書き込む。

UseByOutput(WithTitle As String) As Boolean

出力順序が最初でない（2 番目以降の）By グループにユーザがアクセスできるようにする。

文字列は、操作する特定の By グループのタイトルでなければなりません。たとえば、性別の値「女性」と「男性」でグループ化された二変量の出力があるとします。「女性」のレポートが出力の最初にあるときは、このレポートが **Bivariate**（二変量）オートメーション関数による操作対象レポートになります。「男性」のレポートを操作する場合は、メソッド **UseByOutput("性別 = 男性")** を呼び出します。このメソッドから **True** が戻されると、後続の **Bivariate** メソッドが呼び出されたとき、「男性」出力を処理します。「女性」の出力に処理を戻すには、**UseByOutput("性別 = 女性")** を呼び出します。

DataFeed オブジェクト

DataFeed オブジェクトを使うと、シリアルポートからデータを読むことができます。これによって、ユーザは測定機器を接続し、JMP でデータを読み、その値をオートメーションクライアントのプログラムで利用することができます。

メソッド

Close() As Boolean

ポートへの接続を切断し、ポートの情報をクリアして、このデータフィードオブジェクトが終了したことを JMP に通知する。以後、このデータフィードオブジェクトは使用できません。戻り値が **True** ならば成功、**False** ならば失敗を意味します。

Connect(PortName As String) As Boolean

指定されたポートへの接続を確立する。パラメータには、先に **SetCommParms** または JMP で設定済みの値が使われます。成功のときは **True**、失敗のときは **False** を戻します。ポートの名前には「**COM1**」などがあります。

Disconnect() As Boolean

このデータフィードに関連するポートからの接続を切断する。ただし、以後の **Connect(PortName As String) As Boolean** の実行に備え、ポートの情報は保持します。戻り値が **True** のときは成功を意味します。

GetLine() As String

ポートからデータを読み、値をストリングで戻す。

SetCommParms(BSTR szCommPort, long baudrate, short parity, short databits, short stopbits, short flow) As Boolean

シリアルポートのパラメータを設定する。0 が設定されたパラメータに対しては、JMP の環境設定で指定されている値が使われます。環境設定が設定されていない場合は、デフォルトの値が使われます。**szCommPort** は必須パラメータです。成功のときは **True**、失敗のときは **False** を戻します。ポートの名前には「**COM1**」などがあります。パリティ (parity) の値は **commParityConstants** のリストから取得します。フロー制御の値は、**commFlowControlConstants** のリストから取得した値の論理和か、フロー制御をしない場合は 0 です。

DataTable オブジェクト

DataTable オブジェクトによって、データテーブルの機能の多くを制御できます。先に、**Document.GetDataTable** または **AUTODB.OpenTable** のような別のメソッドによって、**DataTable** のディスパッчポインタを取得しておく必要があります。

プロパティ

Document

データテーブルを持つドキュメントオブジェクトを戻すプロパティ。これは、たとえば **SaveAs(FileName As String)** のように、データテーブルの **Document** オブジェクトクラスからメソッドを実行させる場合に有効です。

NumberColumns

テーブルの列の数を示すプロパティ。設定は不可です。

NumberRows

テーブルの行の数を示すプロパティ。設定は不可です。

Visible

データテーブルが表示されている (True) か表示されていない (False) かを示すプロパティ。設定と取得が可能です。

メソッド

Activate() As Boolean

データテーブルを最前面に表示し、アクティブにする。

AddColumns(Prefix as String, NumToAdd As Integer, Where As Integer, Type As Integer, FieldWidth As Integer) As Integer

データテーブルの `Where` で指定された列（例：Column 3）の後に、指定の数（`NumToAdd`）の列を加える。`Type`（タイプ）は、`colDataTypeConstants` から定数として取得できます。この値は、`Numeric`（数値）、`Character`（文字）、`RowState`（行の属性）、または `Unknown`（不明）です。`FieldWidth`（フィールド幅）は、`Type` が `Character` のときだけ使われます。このメソッドは、正しく加えられた列の数を戻します。

AddNumericTableVar(Name As String, Value As Double)

数値のテーブル変数を追加する。

AddRows(NumberToAdd As Integer, AddAfter As Integer)

指定の行（`AddAfter`）の後に、指定の数（`NumberToAdd`）だけ行を加える。このメソッドは、正しく加えられた行の数を戻します。

`AddAfter` を 0 にすると、行はデータテーブルの冒頭に追加されます。`AddAfter` を -1、または現在データテーブルに含まれている行数より大きい数にすると、行はデータテーブルの末尾に追加されます。

`NumberToAdd` は `short integer` です。つまり、一度に追加できるのは 32,767 行だけです。それ以上の行を追加する方法として、存在しない大きな行番号の後に 1 行だけ追加するやり方があります。すると、JMP はまずデータテーブルの最後の行と新しい行の間に必要なすべての行を作成してから、新しい行を追加します。次の例は、999,999 行目の後に 1 行追加することで、空の 1,000,000 行のデータテーブルを作成します。

```
dt.AddRows(1, 999999);
```

AddRowsHuge(NumberOfRows as Integer, AddAfterRow as Integer) As Integer

(最大 20 億の) 膨大な数の行を追加する。このメソッドは、一度に 32,767 行しか追加できない AddRows の代わりに使用できます。

指定の行 (AddAfterRow) の後に、指定の数 (NumberOfRows) だけ行を加える。このメソッドは、正しく加えられた行の数を戻します。

AddAfterRow を 0 にすると、行はデータテーブルの冒頭に追加されます。AddAfterRow を -1、または現在データテーブルに含まれている行数より大きい数にすると、行はデータテーブルの末尾に追加されます。

SummaryUnlinked() As Datatable

Summary に似ているが、Summary の場合と違い、元のテーブルにリンクしていない要約テーブルを作成する。つまり、一方のテーブルにおいてブラシツールで選択した部分は、もう一方のテーブルでは選択されません。また、元のテーブルを閉じても要約テーブルは閉じません。

AddStringTableVar(Name As String, Value As String)

ストリングのテーブル変数を追加する。

AddToConcatList(ColumnName As String) As Boolean

Concatenate() As DataTable メソッドを使って結合する列のリストに、列を 1 つ追加する。

AddToJoinList(ColumnName As String)

列を、Join (結合) の対象となる列として追加する。

AddToJoinMatchList(ColumnName As String) As Boolean

Matched Column Join で使われる列のリストに列を追加する。タイプが Matched Column でない場合は、入力した値は無視されます。Matched Column Join で連結されるそれぞれのデータテーブルで、対応操作に使われる列を指定する必要があります。

AddToSortList(ColumnName As String, Ascending As Boolean) As Boolean

Sort メソッドで行を並べ替えるキーとなる列のリストに、指定の列を追加する。最初に指定された列が、主要な並べ替えの列になり、それ以降の列では、サブグループ内の並べ替えが行われます。Ascending (昇順) フラグが True のときは昇順に並べ替えられ、False のときは降順に並べ替えられます。

AddToSplitGroupList(ColumnName As String) As Boolean

オプションの関数で、新しいテーブル内の各行を一意に特定できる値を持った列を指定する。

AddToSplitList(ColumnName As String) As Boolean

指定された列を、列を分割する際のキーとして使う列のリストに追加する。

AddToStackList(ColumnName As String) As Boolean

指定された列を、「積み重ねる列」のリストに追加する。

AddToSubList(ColumnName As String) As Boolean

指定された列を、Subset() As DataTable コマンドで使われる列のリストに追加する。

AddToSummaryGroup(ColumnName As String) As Boolean

指定された列を、Summary() As DataTable 演算の対象となるグループのリストに追加する。

AddToSummaryStatList(Stat As Integer)

Summary() As DataTable の演算で求められる統計量のリストに、指定された統計量を追加する。
統計量の整数値は dtSummaryStatConstants の定義から得られます。

AddToSummarySubGroup(ColumnName As String) As Boolean

指定された列を、Summary() As DataTable 演算の対象となるサブグループのリストに追加する。

AddToTransposeList(Name as String) As Boolean

データテーブルの転置で使用する列のリストに列名を追加する。これは、「転置」ダイアログボックスで列を選択することと同じです。

AddToTransposeByList(Name as String) As Boolean

データテーブルの転置で使用するグループ変数の列のリストに列名を追加する。これは、「転置」ダイアログボックスで列を選択して [By] をクリックすることと場合と同じです。

AddToUpdateMatchList(ColumnName as String) As Boolean

新しい UpdateTable(DataTable2 as DataTable, IgnoreMissingValues As Boolean) メソッドの場合、対応のある列の処理用に列を追加する。これと同じメソッドによって 2 つ目のデータテーブルに列が追加されます。これは、AddToJoinMatchList(ColumnName As String) As Boolean と似ています。

ClearRowsSelection()

現在の行の選択をクリアする。

ClearSelectedRowStates() As Boolean

SelectRows(StartRow As Integer, EndRow As Integer) で選択された行の属性をクリアする。

ColorByColumn(Name as String) As Boolean

Name で指定された列の値に基づいて、行に色を設定する。

Concatenate() As DataTable

`AddToConcatList(ColumnName As String) As Boolean` で指定された列を結合する。新しく作成されたデータテーブルへのディスパッチポインタを戻すので、作成されたデータテーブルも制御できます。

DeleteColumn(Column Name As String)

指定された列をテーブルから削除する。

DeleteSelectedRows() As Boolean

`SelectRows(StartRow As Integer, EndRow As Integer)` メソッドで選択された行を削除する。戻り値が `True` のときは成功を意味します。

Document() As Document

そのデータテーブルオブジェクトを含んだドキュメントオブジェクトを戻す。

CheckRowState(Index As Integer, rowStateConstants stateToCheck) As Boolean

1つの行の属性を調べる。このメソッドで戻されるのは長いリストではないので、1つの行や数少ない行の属性を調べるときに便利です。たとえば、データテーブルの第 5 行が非表示になっているかどうかを調べる場合、`CheckRowState(5, rowStateHidden)` を呼び出します。戻し値が `True` なら行が非表示、`False` なら表示されていることを意味します。

EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer

呼び出された時点での、特定の行の属性を含んだ行のリストを作成する。たとえば、`stateToCheck` パラメータが `rowStateSelected` の場合、選択された状態にある行の行番号がリストされます。このメソッドは、リストに含まれた、その属性を持つ行の合計数を戻します。

このメソッドは次のメソッドとともに機能します。

- `EnumRowStatesGetNextRow() As Integer`
- `EnumRowStatesGetRowByIndex(Index as Integer) As Integer`

まず `EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer` を呼び出してから、他の 2 つのメソッドを使ってください。

EnumRowStatesGetNextRow() As Integer

`EnumRowStatesBegin` を呼び出した後で使用する。このメソッドは、`EnumRowStatesBegin` で作成されたリストの行番号を戻します。このメソッドを呼び出すたびにリスト内の次の行が戻され、それがリストの最後の行が戻されるまで続きます。たとえば `EnumRowStatesBegin(rowStateSelected)` で作成されたリストに、選択状態にある行として行 1、4、7 が含まれているとしましょう。ここで `EnumRowStatesGetNextRow` を続けて 3 回呼び出した場合、それぞれ 1、4、7 が戻されます。

すべての行が戻されているので、4 回目に呼び出しを行うと、0 が戻されます。

このメソッドは次のメソッドとともに機能します。

- `EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer`
- `EnumRowStatesGetRowByIndex(Index as Integer) As Integer`

まず `EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer` を呼び出してから、他の 2 つのメソッドを使ってください。

EnumRowStatesGetRowByIndex(Index as Integer) As Integer

`EnumRowStatesBegin` を呼び出した後で使用する。`EnumRowStatesBegin` で作成されたリストにある特定の項目を戻します。たとえば `EnumRowStatesBegin(rowStateSelected)` で作成されたリストに、選択状態にある行として行 1、4、7 が含まれているとしましょう。`EnumRowStatesGetRowByIndex(2)` を呼び出すと、行 4 を表す 4 が戻されます。

行 2 の属性を戻すのではなく、リスト内の 2 番目の属性を戻すことに注意してください。

このメソッドは次のメソッドとともに機能します。

- `EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer`
- `EnumRowStatesGetNextRow() As Integer`

まず `EnumRowStatesBegin(rowStateConstants stateToCheck) As Integer` を呼び出してから、他の 2 つのメソッドを使ってください。

ExcludeSelectedRows() As Boolean

`SelectRows(StartRow As Integer, EndRow As Integer)` で選択された行を、以降の計算対象から除外する。

GetChangedRowStateVector(RowStateToCheck As RowStateConstants)

指定された属性（選択、表示しないなど）について、変更されたかどうかのインデックスの配列を戻す。先に、`RowStateBeginMonitoring` を呼び出しておく必要があります。

GetColumn.ColumnName As String) As Column

テーブル列のディスパッチポインタを取得する。このテーブル列は、以降、列オブジェクトを自動制御するのに使用できます。

GetColumnByIndex(Index As Integer) As Column

番号（1、2、3...）で指定されたテーブル列へのディスパッチポインタを取得する。

GetColumnName(Index As Integer) As String

番号（Index）で指定された列の名前をストリングで戻す。

GetJSLFunctionErrorString As String

`Application` オブジェクトの `GetRunCommandErrorString() As String` および `HasRunCommandErrorString() As Boolean` と同じ。`JSLFunction` メソッドでは、呼び出しが成功した場合に `JSL` 戻り値を取得できます。

HasJSLFunctionErrorString As Boolean も参照してください。

これらのメソッドを呼び出す Visual Basic のコードの例：

```
DT.JSLFunction(Distribution.Columns(Height));
if (DT.HasJSLFunctionErrorString) Then
    MsgBox(DT.GetJSLFunctionErrorString)
Endif
```

GetNumberOfRowsByRowState(rowStateConstants stateToCheck) As Long

入力パラメータに指定された属性に応じて、除外された、表示しない、または選択された行数を戻す。

GetRowStatesChanged() As Boolean

GetRowStatesChanged() を前回呼び出したとき以降、またはデータテーブルを最初に開いたとき以降に、データテーブル内の行の属性に変化があったかどうかを示すブール値を戻す。戻り値が **True** のときは行の属性が変化していることを意味し、**False** のときは変化がないことを意味します。選択、非表示、除外またはラベルの属性が変化すると、このフラグは **true** になります。

GetRowStateVector

行が特定の属性条件（選択、表示しない、除外、ラベルあり）に一致するかどうかを調べ、行のインデックスの配列を戻す。これにより、行の属性情報をすばやく取得し、前回の問い合わせの後で変更されたかどうかを確かめることができます。メソッド宣言は次のようになります。

```
GetRowStateVector(rowStateConstants state) As Variant
```

GetRowStateVectorData も参照してください。

Visual Basic での **GetRowStateVector** の使用例：

```
myarray = DT.GetRowStateVector(rowStateSelected)' 選択した行番号のベクトルを
取得する
    ' 行番号は 1 から始まる
nElements = UBound(myarray)    ' 戻される配列の上限を取得する
Dim Col As JMP.Column
Set Col = DT.GetColumn("name")

For i = 0 To nElements      ' 選択した行の名前を表示する
    MsgBox (Col.GetCellVal(myarray(i)))
Next i
```

HasJSLFunctionErrorString As Boolean

この関数は **GetRunCommandErrorString** メソッドおよび **HasRunCommandErrorString** メソッドと同じ。 **JSLFunction** メソッドでは、呼び出しが成功した場合に JSL 戻り値を取得できます。

GetJSLFunctionErrorString As String も参照してください。

これらのメソッドを呼び出す Visual Basic のコードの例：

```
DT.JSLFunction(Distribution.Columns(Height));
```

```
if (DT.HasJSLFunctionErrorString) Then
    MsgBox(DT.GetJSLFunctionErrorString)
Endif
```

HideSelectedRows() As Boolean

SelectRows(StartRow As Integer, EndRow As Integer) で選択された行を非表示にする。

Join(DataTable2 As DataTable, JoinType As Integer, OutputTableName As String) As DataTable

指定の行 (AddToList) を、各テーブルから取り出して結合し、新しいテーブルを作成する。新しいテーブルへのディスパッチポインタを戻します。結合タイプ (JoinType) には、対応する行によるもの、カーティアン積によるもの、対応する列によるものがあり、dtJoinConstants のどれかで指定します。DataTable2 は、連結される 2 つのテーブルのうち、2 番目のテーブルのディスパッチポインタです。Join (結合) を呼び出すデータテーブルの一般的なオプションだけが使用されます。つまり、DataTable2 の一般的なオプションは無視されます。

Join Type が By Matching Columns(対応する列の値で結合) の場合は、SetJoinMatchOptions で両方のテーブルに対して設定されている Matching Column オプションが使われます。オプションが設定されていない場合は、デフォルトの処理が行われます。AddToListMatchList で指定された列は、結合の処理を実行するときに使われます。

LabelSelectedRows() As Boolean

SelectRows(StartRow As Integer, EndRow As Integer) で選択された行にラベルをつける。

MarkerByColumn(Name as String) As Boolean

Name で指定された列の値に基づいて、行にマーカーを設定する。

NewColumn(Name As String, Type As Integer, Model As Integer, Width As Integer)

指定の名前をつけた、指定のタイプの新しい列を追加する。Type は、colDataTypeConstants の値のどれかで指定します。Model (Continuous, Nominal) は colModelTypeConstants の値のどれかで指定します。

PrintTable() As Boolean

システムのデフォルトプリンタでテーブルを印刷する。

ReorderColumns(ReorderType As Integer)

列を、データタイプ、尺度、名前のどれかで並べ替えるか、または順序を逆にする。元の順序に戻すこともできます。パラメータとして渡される colReorderConstants の値によって並べ替えの方法が決まります。

RowStateBeingMonitoring

オートメーション機能が情報を戻せるように、行の属性の変更の追跡を開始する。

SelectColumn(Column As String, SelectFlag as Boolean) As Boolean

SelectFlag が True の場合は指定された名前の列を選択し、そうでない場合は選択を解除する。列が見つからない場合は False が戻されます。

SelectExcludedRows() As Boolean

除外された行を選択する。成功のときは True を、失敗のときは False を戻します。

SelectHiddenRows() As Boolean

表示しない行を選択する。成功のときは True を、失敗のときは False を戻します。

SelectLabeledRows() As Boolean

ラベルのついた行を選択する。成功のときは True を、失敗のときは False を戻します。

SelectAllMatchingCells() As Boolean

選択されている行／列の組み合わせに一致するセルを選択する。

たとえば、「年齢」列の中で値が 58 のセルが選択されている場合、このメソッドは年齢が 58 となっている、その他のセルを選択します。これらのメソッドを呼び出す前に、DataTable オブジェクトまたは列オブジェクトに対して SelectColumn メソッドを使用し、列を選択しておく必要があります。SelectAllMatchingCells は、開いているすべてのデータテーブルに適用されます。

SelectMatchingCells() As Boolean

選択されている行／列の組み合わせに一致するセルを選択する。

たとえば、「年齢」列の中で値が 58 のセルが選択されている場合、このメソッドは年齢が 58 となっている、その他のセルを選択します。これらのメソッドを呼び出す前に、DataTable オブジェクトまたは列オブジェクトに対して SelectColumn メソッドを使用し、列を選択しておく必要があります。SelectMatchingCells は、現在のデータテーブルに適用されます。

SelectRandomly(SampleRate As Long) As Boolean

行をデータテーブルから無作為に選択する。SampleRate の値が 1 より大きい場合は、選択される行の数を表します。SampleRate の値が 0 ~ 1 の範囲にある場合は、データテーブルからその比率分の行数が選択されます。

SelectRows(StartRow As Integer, EndRow As Integer)

行を、マウスを使って強調表示したときと同じように選択する。

SelectRowsWhere(Column Name As String, Operation As Integer, SelectHow As Integer, Comparative As String) As Boolean

[行] メニューの Select Where (Where 条件で選択) の操作を実行する。ColumnName で指定された列は、rowSelectWhereOperations の定数で定義された演算子を使って比較されます。演算には、equals (等しい)、greater than (より大きい)、less than (より小さい) などがあります。SelectHow パラメータによって、先に選択された行をどう取り扱うかを指定します。新

しく選択する範囲は、以前の選択範囲をクリアして選択し直すか、以前の選択に加えて選択するか、または、以前の選択範囲の中から選択するかのどれかです。このパラメータには `rowSelectWhereHow` の定数が使われます。最後に、Comparative パラメータで比較対象となる値を定義します。このパラメータは、選択サブセットを作成するのに使用される名前、数値、その他の値を含んだストリングです。

SetJoinMatchOptions(DropMultiples As Boolean, IncludeNonMatches As Boolean)

Matching Column Join (対応する列による結合) 操作のオプションを設定する。連結されるデータテーブルそれぞれにオプションを設定することができ、Join (結合) は両方のオプション設定を受け付けます。これらのオプションは、「対応する列」ダイアログボックスのオプションと同じ働きをします。デフォルトでは、`DropMultiples` と `IncludeNonMatches` が `False` となっています。

SetJoinMergeColumns(Boolean)

結合の際に同名の列をマージするときはオプションをオン (`True`) にし、マージしないときはオフ (`False`) にする。

SetJoinOptions(UpdateFirstTable As Boolean, CopyFormulas As Boolean, SuppressFormulaEval As Boolean)

Join (結合) 用の一般的なオプションを設定する。結合のしかたの各タイプはすべて、この設定を受け付けます。これらのオプションは、「結合 (Join)」ダイアログボックスのオプションと同じ働きをします。Matching Column による結合が実行されるとき、Join (結合) は呼び出しているデータテーブルに属している一般的なオプションを使います。デフォルトでは、`UpdateFirstTable` が `False`、`CopyFormulas` が `True`、`SuppressFormulaEval` が `True` となっています。

SetStackMultipleSeriesN(short N) As Boolean

N 個の複数系列の積み重ねを可能にする。この関数は、`DataTable` オブジェクトで `Stack` を呼び出す前に実行する必要があります。処理に成功したときは `True` を、失敗したときは `False` を戻します。

SetTransposeOptions(OutputTableName as String, UseSelectedRows as Boolean) As Boolean

転置によって作成される出力テーブルの名前を設定する。また、選択された行だけを使うか、すべての行を使うかを設定します。

SetWindowPos(X As Integer, Y As Integer)

テーブルウィンドウの位置を、Windows デスクトップからの相対的な位置として指定する。X と Y の値はピクセル座標です。

SetWindowSize(CX As Integer, CY As Integer)

テーブルウィンドウのサイズをピクセル座標で指定する。

Sort(Replace As Boolean) As DataTable

`AddToSortList(ColumnName As String, Ascending As Boolean) As Boolean` で指定された列を使ってテーブルの並べ替えを行う。Replace が `True` ならば、既存のテーブル内でデータが並べ替えられ、そのテーブルへのポインタが戻されます。Replace が `False` ならば、並べ替えたデータで新しいテーブルが作成され、そのテーブルへのディスパッチポインタが戻されます。

Split(ColumnID As String, OutputTableName As String, KeepRemainingCols As Boolean)

テーブルを分割する。その際、新しい列名は「`ColumnID`」列を使って識別し、`AddToSplitList(ColumnName As String) As Boolean` で入力された列をデータとし、`AddToSplitGroupList(ColumnName As String) As Boolean` で入力された列を、行を識別するリストとします。

Stack(idColumnName As String, stackedColumnName As String, TableName As String) As DataTable

`AddToStackList(ColumnName As String) As Boolean` で指定された列の値を積み重ねる。その際、新しいテーブルの各行は指定の ID 列名 (`idColumnName`) を使って識別します。積み重ねてできた新しい列の名前は `stackedColumnName` で指定でき、その列を含むデータテーブルの名前は `TableName` で指定できます。新しくできたデータテーブルへのディスパッチポインタが戻されます。

Subset() As DataTable

`AddToSubList(ColumnName As String) As Boolean` で選択された列と `SelectRows` で選択された行を合わせたデータによる、新しいデータテーブルを作成する。`AddToSubList(ColumnName As String) As Boolean` で追加された列がない場合、すべての列をサブセットに含めます。

SubsetSetRandomSelection(SampleRateOrSize as Double, Shuffle As Boolean) As Boolean

`SampleRateOrSize` が 0 より大きく 1 より小さい場合は、標本抽出率として扱われる。1 より大きい場合は、標本サイズとして扱われます。`Shuffle` が 1 に設定された場合、`SampleRateOrSize` は無視され、作成されたテーブル内ですべての行が無作為に並べ替えられます。

SubsetStratifyAddColumn(Column As String) As Boolean

無作為に選択されたサブセットの層化に使用するテーブル列を追加する。複数の列を追加できます。サブセットが呼び出されると、この設定は初期化されます。別のサブセットを実行する場合は、再度列を指定します。

Summary() As DataTable

`AddToSummaryGroup(ColumnName As String) As Boolean`、`AddToSummarySubGroup(ColumnName As String) As Boolean`、および `AddToSummaryStatList(Stat As Integer)` を使って要約テーブルを作成する。新たに作成されたデータテーブルへのディスパッチポインタが戻されます。

SummarySetStatColumnFormat(summaryStatColNameConstants format)

要約統計量により生成された列の列名の表示形式を設定する。

Transpose() As DataTable

データテーブルを単純に転置したものを新しいデータテーブルに入れ、そのデータテーブルへのディスパッチポインタを戻す。

UpdateTable(DataTable2 as DataTable, IgnoreMissingValues As Boolean)

[テーブル] > [更新] を選択した場合と同じ働きをする。テーブルは、2つ目のテーブルの変更された値で更新または結合されます。対応する列は、AddToUpdateMatchList(ColumnName as String) As Boolean メソッドを使って設定します。AddToUpdateMatchList メソッドによって対応する列が追加されない場合は、2つのテーブル間で通常の更新が行われます。

戻り値が `True` ならば成功、`False` ならば失敗を意味します。

Document オブジェクト

Document オブジェクトは、各分析やデータテーブルで使用する共通のプロパティやメソッドを提供します。

プロパティ

Application

Application オブジェクトを参照するディスパッチポインタを含むプロパティ。取得のみ可能で、設定や変更はできません。

AutoSave

データテーブルをもつドキュメントを閉じる前に、そのデータテーブルを自動的に保存するかどうかを指定するプロパティ。`True` ならば保存します。設定と取得が可能です。

FullName

ドキュメントの完全な名前を含む、文字列型のプロパティ。取得のみ可能です。

Name

ドキュメントの短縮名を含む、文字列型のプロパティ。取得のみ可能です。

Path

ドキュメントの完全パスを含む、文字列型のプロパティ。取得のみ可能です。

Saved

最後に保存されてからドキュメントに変更が加えられたかどうかを示すプロパティ。`False` のときは、ドキュメントに未保存の変更があることを示します。`True` のときは、未保存の変更はありません。

Visible

ドキュメントが表示されている (True) か表示されていない (False) かを区別するプロパティ。設定と取得が可能です。

メソッド

Activate()

ドキュメントを含むウィンドウを最前面に表示する。

Close(SaveChanges as Boolean, FileName As String)

ドキュメントを閉じる。SaveChanges を使うと、ドキュメントを最終的に保存する際に、別のファイル名を指定できます。SaveChanges を True に設定したときは、FileName でドキュメントの保存先のパス名とファイル名を指定します。既存のファイルに保存する場合は、Save メソッドを使うことを推奨します。

CopyToClipboard()

ドキュメントのウィンドウの内容をクリップボードにコピーする。分析の内容をコピーする場合は、必ず各分析オブジェクトに用意されている CopyToClipboard メソッドを使うようにしてください。

CreateBivariate()

メソッド名の先頭が Create であるメソッドは、特定の分析オブジェクト (Bivariate オブジェクトなど) を生成する。オブジェクトへのディスパッチポインタが戻され、以降、分析する列を指定したり、分析を起動したり、分析出力を操作したりするのにこのオブジェクトを使用できます。なお、このメソッドを先に呼び出してもおかないと、分析を起動することができません。

CreateCluster()

メソッド名の先頭が Create であるメソッドは、特定の分析オブジェクト Cluster オブジェクトを生成します。オブジェクトへのディスパッチポインタが戻され、以降、分析する列を指定したり、分析を起動したり、分析出力を操作したりするのにこのオブジェクトを使用できます。なお、このメソッドを先に呼び出してもおかないと、分析を起動することができません。

CreateNormalMixtures() As NormalMixtures

このメソッドは、JMP で「正規混合」プラットフォームを使うためのオートメーションオブジェクトを作成する。このメソッドは、「クラスター分析」プラットフォームで KMNormalMixtures を使用するのと同じです。

CreatePlatform()

メソッド名の先頭が Create であるメソッドは、特定の分析オブジェクト (Bivariate オブジェクトなど) を生成する。オブジェクトへのディスパッチポインタが戻され、以降、分析する列を指定したり、分析を起動したり、分析出力を操作したりするのにこのオブジェクトを使用できます。なお、このメソッドを先に呼び出してもおかないと、分析を起動することができません。

Save()

ドキュメントがデータテーブルであれば、そのデータテーブルをディスク上に保存する。データテーブルでない場合は

何もしません。なお、分析結果やジャーナルを保存するメソッドは、それらのそれぞれのオブジェクトのほうで用意されています。

SaveAs(Filename As String)

ドキュメントがデータテーブルであれば、そのデータテーブルに指定のファイル名をつけてディスク上に保存する。データテーブルでない場合は何もしません。なお、分析結果やジャーナルを保存するメソッドは、それらのそれぞれのオブジェクトのほうで用意されています。

Journal オブジェクト

Journal オブジェクトのメソッドを使用すると、ジャーナルを HTML、RTF、MS Word、または JMP ジャーナルファイルとして保存できます。

メソッド

GetActiveJournal() As Journal

現在アクティブになっているジャーナルを参照するハンドルを戻す。これは、現在のジャーナルに対して、何らかの追加の操作を行いたい時に利用します。

SaveAsHTML(BSTR filename, jmpGraphicsFormats graphicType) As Boolean

指定されたファイル名で、ジャーナルを HTML としてディスクに保存する。2 番目のパラメータは、HTML 出力のイメージに使用するグラフィック形式の種類です (PNG、JPEG、METAFILE など)。

SaveAsJournal(BSTR filename) As Boolean

指定されたファイル名で、ジャーナルをディスク上に保存する。ディスクに保存されたジャーナルは、次の JMP セッションで再利用することができます。

SaveAsRTF(BSTR filename, jmpGraphicsFormats graphicType) As Boolean

指定されたファイル名を使って、ジャーナルを RTF (リッチテキスト形式) でディスクに保存する。2 番目のパラメータは、RTF 出力のイメージに使用するグラフィック形式の種類です (PNG、JPEG、METAFILE など)。

SaveAsMSWordDoc(Filename As String) As Boolean

ジャーナルから RTF 形式に出力し、指定された名前で Microsoft Word 文書として保存する。このメソッドは、クライアントマシンに MS Word 2000 以上がインストールされている場合にのみ使用できます。処理に成功したときは True を、失敗したときは False を戻します。

Text Import オブジェクト

TextImport オブジェクトを使うと、テキストファイルを JMP データテーブルとして開くことができます。このとき、テキストファイルで使われているフィールドのタイプや行の区切り文字を、ユーザが指定できます。読み込みの結果作成されたデータテーブルへのポインタが戻されます。ここで挙げたメソッドを使う前に、必ず TextImport オブジェクトを作成しておく必要があります。そのためには、Application オブジェクトで CreateTextImportObject(FileName As String, NumberColumns as Integer) As TextImport をコールします。

メソッド

TextImport オブジェクトメソッドを使うと、テキストファイルを JMP データテーブルとして開くことができます。このとき、テキストファイルで使われているフィールドのタイプや行の区切り文字を、ユーザが指定できます。

ColumnNamesStart(StartLine as Integer)

列見出し（列名）を含んでいる行が上から何行目にあるかを指定する。正の値を指定すると、ファイルに列見出しが含まれていることになるので、さらに FirstLineIsData(False) を呼び出す必要はありません。この列見出しを含む行は、データの先頭行の前にある必要があります。

DataStarts(StartLine As Integer)

データの開始行が上から何行目にあるかを指定する。引数に 1 を指定した場合は、列見出しがないものとみなされるので、FirstLineIsData(True) を呼び出す必要はありません。

FirstLineIsData(Flag As Boolean)

テキストファイルの最初の行がデータとみなすのか、列見出し（列名）とみなすのかを指定します。True ならばデータ、False ならば列見出しだると解釈します。

OpenFile() As Document

すでに実行済みのメソッドで指定されたオプションに従って、テキストファイルを開く。Document オブジェクトのポインタを戻します。この処理によって開かれたデータテーブルのポインタを取得するには、ドキュメントオブジェクトに GetDataTable を適用してください。

SetColumnType(ColumnName As Integer, Type As Integer) As Boolean

列のタイプを強制的に文字または数値にする。2 番目のパラメータには colDataTypeConstants で定義されている値を使いますが、RowState（行の属性）はこの操作には使用できません。

SetEndOfFieldOptions(Options As Integer)

フィールドとフィールドとの間にある区切り文字を指定する。jmpTIEndOfFieldConstants で定義されている値を組み合わせてもかまいません。Visual Basic では、演算子 Or を使って値を組み合わせることができます。

SetEndOfLineOptions(Options As Integer)

行の終わりを示す区切り文字を指定する。`jmpTIEndOfLineConstants` で定義されている値を組み合わせてもかまいません。Visual Basic では、演算子 `Or` を使って値を組み合わせることができます。

StripQuotes(Flag As Boolean)

データを新しいデータテーブルへ挿入する際に、引用符を取り除くかどうかを指定する。`True` ならば取り除き、`False` ならば含めたままにします。

プラットフォームのメソッド

各プラットフォームには、それぞれのプラットフォームを起動・操作するためのメソッドがあります。

Attribute Chart オブジェクトのメソッド

Attribute Chart オブジェクトを使うと、「計数値管理図」 プラットフォームを起動し、操作できます。

EffectivenessReport(Flag As Boolean)

対応するオプションをオン (`True`) またはオフ (`False`) にする。

Bivariate オブジェクトのメソッド

Bivariate オブジェクトを使うと、「二変量」の分析を実行、操作できます。

DensityEllipses(Degree As Double)

与えられた確率で確率密度楕円を描く。

FitEachValue As Fit

値ごとのあてはめを実行する。`Fit` オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitLine As Fit

直線のあてはめを実行する。`Fit` オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitLoess() As Fit

デフォルトのパラメータを使って Loess のあてはめを実行する。`Fit` オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitLoessWeightConstants(fitLoessWeightTricube, fitLoessWeightCosine, fitLoessWeightEpanechnikov, fitLoessWeightGaussian, fitLoessWeightCauchy)

カーネル（局所）平滑化法の重み関数を指定する。

FitLoessWithParms(fitLoessLambdaConstants Lambda, Alpha as Double, Robustness as Short)

指定されたパラメータを使って Loess のあてはめを実行する。Lambda は、0 ~ 2 (0 と 2 を含む) の定数で、通常は Linear または Quadratic のどちらかです。Alpha は、0 ~ 1 (0 と 1 を含む) の範囲の値です。Robustness は、0 ~ 4 (0 と 4 を含む) の範囲の値です。

Fit オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitMean As Fit

平均のあてはめを実行する。Fit オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitOrthogonal(OrthogonalFitConstant as Integer, VarianceRatio As Double) As Fit

指定された分散比で直交回帰分析を行う。1 番目のパラメータに入る値は、bivarFitTransformConstants にある定数のどれかです。Fit オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitPolynomial(Degree As Double) As Fit

指定された次数（例：3.0）で多項式のあてはめを行う。Fit オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitRobust, FitCauchy(Flag As Boolean) As Fit

ロバストまたは Cauchy のあてはめを行い、あてはめのオブジェクトを戻す。その後、さらにオプションを指定することができます。

FitSpline(Degree As Double) As Fit

指定された滑らかさ（例：100）で、スプライン曲線のあてはめを実行する。Fit オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

FitTransformed(Xtransform As Integer, Ytransform as Integer, PolynomialDegree as Integer)

X および Y の変換を伴う、あてはめを実行する。

- Xtransform と Ytransform の値は bivarFitTransformConstants から取得できます。また、多項式の次数（例：3）は FitPolynomial(Degree As Double) As Fit のときと同様です。

FitTransformedWithOptions(Xtransform As Integer, Ytransform as Integer, PolynomialDegree as Integer, CenteredPolynomial as Boolean, ConstrainIntercept as

Boolean, InterceptValue as Double, ConstrainSlope As Boolean, SlopeValue as Double) As Fit

X および Y の変換や制約を伴う、あてはめを実行する。

- Xtransform と Ytransform の値は bivarFitTransformConstants から取得できます。また、多項式の次数（例：3）は FitPolynomial(Degree As Double) As Fit のときと同様です。
- CenteredPolynomial は True または False で、必ず指定します。JMP で行われる多くの分析では、デフォルトは True になっています。
- ConstrainIntercept は切片に制約を課すかどうかを示すブール値です。次のパラメータ InterceptValue に切片の値を指定する場合は、これを必ず True に設定します。ConstrainIntercept が False の場合、InterceptValue は無視されます。
- ConstrainSlope はブール値で、傾きに制約を課したい場合は必ず True に設定します。False に設定した場合、SlopeValue は無視されます。

GroupBy(Column Name As String) As Boolean

指定された列の値でグループ化し、分析を行う。処理に成功したときは True を、失敗したときは False を戻します。

HistogramBorders(Flag as Boolean)

ヒストグラム軸の表示オプションを、オン（True）またはオフ（False）に設定する。

KernelSmoothen(Lambda As JMP.fitLoessLambdaConstants, Weight As JMP.fitLoessWeightConstants, Alpha As Double, Robustness As Short)

指定されたパラメータを使って Loess のあてはめを実行する。Lambda は、0 ~ 2（0 と 2 を含む）の定数で、通常は Linear または Quadratic のどちらかです。Alpha は、0 ~ 1（0 と 1 を含む）の範囲の値です。Robustness は、0 ~ 4（0 と 4 を含む）の範囲の値です。

Fit オブジェクトへの参照を戻します。戻されたポインタを使って、追加の処理を行えます。

NonParDensity() As FitDensity

ノンパラメトリックな密度推定を行い、FitDensity オブジェクトの参照を戻す。戻されたポインタを使って、追加の処理を行えます（ノンパラメトリックな密度のあてはめについては、「FitDensity オブジェクトのメソッド」（130 ページ））。

ShowPoints(Flag as Boolean)

True(1) に設定されている場合はプロット上の点を表示し、False(0) に設定されている場合は表示しない。

Bubble Plot オブジェクトのメソッド

Bubble Plot オブジェクトを使うと、バブルプロットを起動して操作できます。

AggregateSizeAsSum(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

AggregateXAsSum(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

AggregateYAsSum(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

AllLabels(Flag As Boolean)

すべてのラベルをオン (True) またはオフ (False) にする。

BubbleSize(Size as Double)

バブルの円のサイズを倍精度の値で指定する。0 が最小値です。

BubbleSpeed(Speed as Double)

アニメーションの際のバブルの速度を倍精度の値で指定する。0 が最も遅い値です。

BubbleTimeIndex(Index as Double)

アニメーションの開始時点を、0 を基準としたインデックスとして指定する。たとえば、1.0 と指定すると、2 番目に小さい時点から開始します。

CombineAll()

2 つの ID 列が指定されていて、SplitAll() が実行済みの場合、CombineAll() は小さいバブルを元のバブルに再結合する。

Filled(Flag As Boolean)

塗りつぶしをオン (True) またはオフ (False) にする。

Go()

アニメーションを先へ進める。終了したら冒頭にループします。

LaunchAddColoring(Name as BSTR)

起動する前に、色の列を指定する。バブルプロットでは、通常の X 値と Y 値に加えて、色など、プラットフォームに特有の列を指定するための起動メソッドが用意されている。

LaunchAddID(Name as BSTR)

起動する前に、ID の列を指定する。バブルプロットでは、通常の X 値と Y 値に加えて、ID など、プラットフォームに特有の列を指定するための起動メソッドが用意されている。

LaunchAddSizes(Name as BSTR)

起動する前に、サイズの列を指定する。バブルプロットでは、通常の X 値と Y 値に加えて、サイズなど、プラットフォームに特有の列を指定するための起動メソッドが用意されている。

LaunchAddTime(Name as BSTR)

起動する前に、時間の列を指定する。バブルプロットでは、通常の X 値と Y 値に加えて、時間など、プラットフォームに特有の列を指定するための起動メソッドが用意されている。

Prev()

アニメーションを、1 時間単位だけ巻き戻す。

SelectableAcrossGaps(Flag As Boolean)

[ギャップでも選択可能] をオン (True) またはオフ (False) にする。SelectableAcrossGaps を使用するには、Launch メソッドを呼び出す前に時間値が指定されている必要があります。

SplitAll()

2 つの ID 列が指定されている場合に、最初の ID で定義されたバブルを、2 番目の ID で定義された小さい構成要素に分ける。

Step()

アニメーションを、前方に 1 時間単位だけ進める。

Stop()

アニメーションを停止する。

Trails(Flag As Boolean)

軌跡をオン (True) またはオフ (False) にする。

Categorical オブジェクトのメソッド

Categorical オブジェクトを使うと、カテゴリカルプラットフォームを起動し、操作することができます。

AgreementStatistic(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。オプションは特定の応答に固有のもので、応答のタイプによっては機能しないことがあります。

CrosstabFormat(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。これらのオプションは互いに排他的で、1 つをオンにすると、その他はオフになります。

- **CrosstabFormat**
- **CrosstabTransposed**

- TableFormat
- TableTransposed

CrosstabTransposed(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。これらのオプションは互いに排他的で、1つをオンにすると、その他はオフになります。

- CrosstabFormat
- CrosstabTransposed
- TableFormat
- TableTransposed

Frequencies(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。

FrequencyChart(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。

LaunchAddResponseRole(ResponseType as jmpCategoricalResponseRoles) As Boolean

複数の列に、Aligned Responses などの役割を割り当てます。LaunchAddToResponseList で指定されたすべての列に、ここで指定した役割が割り当てられます。役割が割り当てられたら、LaunchAddToResponseList のリストはクリアされます。複数回、処理を繰り返すことにより、異なる役割を割り当てるることができます。Launch が呼び出されて、初めて分析が実行されます。

LaunchAddToResponseList(ColumnName as String) As Boolean

応答の役割を追加するために使用されるリストに、列を追加する。LaunchAddToResponseList を複数回呼び出すことによって、リストに複数の列を追加できます。その後、LaunchAddResponseRole を呼び出すことによって、リストに追加されたすべての列に同じ役割が割り当てられます。

Legend(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。

RatePerCase(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。

ShareChart(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。

ShareOfResponses(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。

TableFormat(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。これらのオプションは互いに排他的で、1つをオンになると、その他はオフになります。

- CrosstabFormat
- CrosstabTransposed
- TableFormat
- TableTransposed

TableTransposed(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。これらのオプションは互いに排他的で、1つをオンになると、その他はオフになります。

- CrosstabFormat
- CrosstabTransposed
- TableFormat
- TableTransposed

TestEachResponse(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。オプションは特定の応答に固有のもので、応答のタイプによっては機能しないことがあります。

TestResponseHomogeneity(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。オプションは特定の応答に固有のもので、応答のタイプによっては機能しないことがあります。

TransitionReport(Flag as Boolean) As Boolean

対応するオプションをオン (Flag は True) またはオフにする。オプションは特定の応答に固有のもので、応答のタイプによっては機能しないことがあります。

Cell Plot オブジェクトのメソッド

Cell Plot オブジェクトを使うと、セルプロットを起動して操作できます。

LaunchOptions(BOOL Scale, BOOL Center)

起動する前に、スケールオプションと中心化オプションを設定する。これらは「セルプロット」起動ダイアログボックスのオプションに対応しています。デフォルトでは、これら 2 つのオプションはオフ (False) です。

Legend(Flag As Boolean)

凡例を表示 (True) または非表示 (False) にする。

Chart オブジェクトのメソッド

Chart オブジェクトを使うと、「チャート」プラットフォームを起動し、操作できます。Chart プラットフォームは廃止されました。Chart プラットフォームを呼び出すオートメーションのコードは、Graph Builder（グラフビルダー）を使用してグラフを作成します。このため、以前のバージョンとグラフの外観が異なる場合があります。

ConnectPoints(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

LaunchAddY.ColumnName As String, Statistic as Short)

このメソッドでは、Y列の値を指定する。最初のパラメータには、通常の列名が入ります。2番目のパラメータには、chartStatConstants にある統計量の1つが入ります。これらの統計量は、チャート起動ダイアログボックスに表示されるものと同じです。統計量ではなく、データそのものをプロットしたい場合は、「Data」を指定してください。

Orientation(WhichWay As Short)

グラフの向きを、縦にするか横にするかを指定する。引数の値は chartOrientConstants に用意されています。

Overlay(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

OverlayColor(Color As Short)

水準の色を jmpColorConstants の値から指定する。

SeparateAxes(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

ShowPoints(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

SpecifyQuantilesVal(Quantiles as Double) As Boolean

分位点統計量の列を追加する場合、このメソッドを使って分位点の値を指定することができます。分位点の値は、デフォルトでは 25.0 です。このメソッドは、LaunchAddY.ColumnName As String, Statistic as Short) で列を追加する前に呼び出してください。

SpecifyType(ChartType as Short)

表示するグラフのタイプ（棒、垂線、点、折れ線、円）を指定する。引数の値は chartChartTypeConstants に用意されています。

Cluster オブジェクトのメソッド

Cluster オブジェクトを使うと、階層的クラスター分析および *k-means* クラスター分析を起動し、操作できます。分析を起動する際の詳細については、「プラットフォーム共通のメソッド」(83 ページ) してください。

メモ：

- クラスター分析に固有の起動メソッドには、重要なものが 2 つあります。
 - `LaunchSpecifyKMeans(Flag As Boolean)`
 - `LaunchSpecifyDistanceFormula(FormulaType As Integer)`
- 階層的クラスター分析と *k-means* クラスター分析にもそれぞれ固有のメソッドがあります。Cluster オブジェクトの階層型クラスター固有メソッドおよび Cluster オブジェクトの *k-means* クラスター固有メソッドを参照してください。

ClusterCriterion, ClusterSummary, ConstellationPlot (Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

ColorClusters(Flag As Boolean)

オン (True) / オフ (False) オプションで、対応するオプションをインタラクティブに操作したときと同じ働きをする。

KMNormalMixtures(Flag as Boolean)

このオプションのサポートは、JMP の「クラスター分析」から削除されました。このオプションはサポートされなくなりました。代わりに、新しい「正規混合」オートメーションプラットフォームを使用してください。「正規混合のメソッド」(154 ページ)

KMParallelCoordPlots(Flag as Boolean) As Boolean

K-Means クラスター分析のためのパラレルプロットを表示する。

KMSOMBandwidth(Bandwidth As Double)

自己組織化マップのバンド幅を指定する。

LaunchSpecifyDistanceFormula(FormulaType As Integer)

クラスターを計算するときの距離計算式を指定する (例: 重心法、Ward 法など)。引数の `FormulaType` (計算式のタイプ) に指定できる値は、`clusterDistanceConstants` に用意されています。

LaunchSpecifyKMeans(Flag As Boolean)

階層型クラスター分析 (False) と、*k-means* クラスター分析 (True) のどちらを実行するかを示す。Launch メソッドによってプラットフォームが起動した後には、階層型と *k-means* のそれぞれのプラットフォームだけで使えるメソッドがあります。それらについては後述します。

Legend(Flag As Boolean)

凡例を表示 (True) または非表示 (False) にする。

MarkClusters(Flag As Boolean)

オン (True) / オフ (False) オプションで、対応するオプションをインタラクティブに操作したときと同じ働きをする。

NumberOfClusters(Number As Integer)

作成するクラスターの数を指定する。

ParallelCoordPlots, ScatterPlotMatrix

対応するオプションをオン (True) にする。

SaveClusters()

データテーブルの新しい列に各行のクラスター番号を保存する。

Contingency オブジェクトのメソッド

Contingency オブジェクトを使うと、分割表分析を起動し、操作できます。

Cochran(Column Name As String) As Boolean

ブロック列として指定された列を使って、Cochran-Mantel-Haenszel 検定を実行する。

Correspondence(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Crosstabs(Flag As Boolean) As Crosstabs

Crosstabs オプションをオン (True) またはオフ (False) にする。このメソッドは Crosstabs (分割表) へのディスパッチポインタを戻し、さらに操作することを可能にします (「Crosstabs オブジェクトのメソッド」 (118 ページ))。

HorizontalMosaic(Flag as Boolean)

モザイク図を横向き (True) または縦向き (False) に設定する。モザイク図を表示するには、MosaicPlot(Flag As Boolean) で True を指定します。

MosaicPlot(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。HorizontalMosaic(Flag as Boolean) で横向きまたは縦向きの表示を設定しない場合、デフォルトで縦向きのモザイク図が表示されます。

NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)

名義尺度軸の表示に関するブール値オプションを設定する。プラットフォームで使用可能なオプションをオン (Flag が True) またはオフ (Flag が False) にします。たとえば、Rotated Tick Labels (目盛りラベルの回転)、Divided Lines (分割線)、Lower Frame (下のフレームの表示) などを設定できます。目盛りラベルの回転は、「一元配置」プラットフォームと「変動性図」プラットフォームでのみ使用できます。このメソッドを呼び出す前に、GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long を呼び出して、名義尺度軸ディスプレイボックスへのハンドルを取得する必要があります。

Tests(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Contour オブジェクトのメソッド

Contour オブジェクトを使うと、等高線図を作成し、操作できます。

FillAreas(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

GenerateGrid(HorizontalSize As Integer, VerticalSize As Integer) As DataTable

新しいデータテーブルを作成し、指定されたグリッド数で、線形補間によって計算した等高線の値を保存する。作成されたデータテーブルを参照するディスパッチポインタが戻されます。新しくできたデータテーブルへのディスパッチポインタが戻されます。

LabelContours(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

ReverseColors(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

SaveContours() As DataTable

新しいデータテーブルを作成し、等高線座標のデータを保存する。作成されたデータテーブルを参照するディスパッチポインタが戻されます。

SaveTriangulation() As DataTable

新しいデータテーブルを作成し、三角要素座標のデータを保存する。作成されたデータテーブルを参照するディスパッチポインタが戻されます。

ShowBoundary(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

ShowContours(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

ShowDataPoints(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

ShowTriangulation(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

ContourProfiler オブジェクトのメソッド

ContourProfiler オブジェクトを使うと、等高線プロファイルを起動し、プラットフォーム共通のメソッドや等高線プロファイル固有のメソッドを使って結果を操作できます。

ContourGrid(Low As Double, High As Double, IncrementAs Double)

指定された最小値、最大値、そして間隔のグリッドに基づいた等高線を描く。

ContourGridWithResponse(low as Double, high as Double, increment as Double, responseColumn As String) As Boolean

この関数は ContourGrid(Low As Double, High As Double, IncrementAs Double) と同じですが、等高線グリッドを描く対象の列を、デフォルトの列ではなく、引数で指定することができます。最後のパラメータに、等高線グリッドを描く列の名前を指定できます。

SurfacePlot(Flag As Boolean)

曲面プロットをオン (True) またはオフ (False) にする。

ControlChart オブジェクトのメソッド

Control Chart オブジェクトを使うと、各種の管理図を起動し、操作できます。Control Chart には、共通の起動メソッドとは異なる、固有の起動メソッドがたくさんあります。

BoxChart(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

CenterColor(Color As Integer)

折れ線の色を設定する。引数に指定できる色は、jmpColorConstants で用意されています。

ConnectColor(Color As Integer)

折れ線の色を設定する。引数に指定できる色は、jmpColorConstants で用意されています。

ConnectPoints(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ConnectThroughMissing(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ControlLimits(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

LaunchAddPhase, LaunchRemovePhase(ColumnName As String)

管理図を作成する前にフェーズ変数を追加または削除する。

LaunchAddProcess(ColumnName As String) As Boolean

管理図にプロットする列を選択する。計量値管理図では、測定値の列を指定します。計数値管理図では、不適合数や、不適合品の割合を指定します。

LaunchAddSampleLabel(ColumnName As String) As Boolean

管理図の X 軸に表示させるラベルを含んでいる列を選択する。

LaunchAddSampleUnitSize(ColumnName As String) As Boolean

行のサブグループを定義する列を選択する。

LaunchSetChartType(ChartType As Integer)

チャートの種類を `jmpControlChartConstants` から選択する。このメソッドは、`CreateControlChart` で `ControlChart` オブジェクトを作成した後、最初に、他のメソッドよりも前に、呼び出してください。

LaunchSetConstantSampleSize(Flag As Boolean, SampleSize As Integer)

最初のパラメータが `True` のとき、標本ラベルの列によってではなく、指定された定数 `SampleSize` を標本サイズとして、サブグループを構成する。

LaunchSetCStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetCusumOptions(TwoSided As Boolean, DataUnits As Boolean)

CUSUM(累積和)管理図で、両側検定とデータ単位のオプションを設定する (`True` ならばオン)。

LaunchSetCusumStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetEWMAStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetEWMAWeight(Weight As Double)

EWMA 管理図で、重みを指定する。

LaunchSetIRChartParms(IndMeas As Boolean, MovingRange As Boolean, Range As Integer)

IR 管理図で、個々の測定値と移動範囲のオプションを設定する。移動範囲を選択した場合、範囲を整数で指定する必要があります。

LaunchSetIRStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetIRSummarizeParms(PreSummarize As Boolean, Mean As Boolean, StdDev As Boolean)

最初のパラメータが `True` ならば、予め集計を行った IR 管理図とする。「On Group Means」(グループ平均で)、「On Group Standard Deviations」(グループ標準偏差で) のどちらか、またはその両方を指定します。`Presummarize` (予め集計) を選択する (`True`) と、標本ラベルの列が指定されていれば、そこから各サブグループが構成されます。標本ラベルの列がない場合、または `LaunchSetConstantSampleSize(Flag As Boolean, SampleSize As Integer)` がコードされている場合は、定数の標本サイズをサブグループが構成されます。

LaunchSetKSigmaAlphaH(KSigma As Boolean, alpha As Boolean, H As Boolean, value As Double, beta As Double)

`KSigma`、`Alpha`、`H`、`Beta` が `True` のときは、そのパラメータの値を `value` に設定する。これらの値は倍精度で指定します。`Beta` と `H` の値は、CUSUM (累積和) 管理図のときにだけ関係します。

LaunchSetNPStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

**LaunchSetPresummarizeChartTypes(VARIANT_BOOL IndivGroupMeans,
VARIANT_BOOL IndivGroupStdDev, VARIANT_BOOL MovingRangeGroupMeans,
VARIANT_BOOL MovingRangeStdDev) As Boolean**

予め集計管理図に対して使用できる 4 つのオプションについて、オン (`True`) / オフ (`False`) を設定する。

LaunchSetPresummarizeStats(double sigma, double meanMeasureGroup, double meanMeasureStdDev, double meanMovingGroup, double meanMovingStdDev) As Boolean

予め集計管理図の統計量を設定する。これは、「管理図」起動ダイアログボックスで設定できるパラメータの値に対応します。

LaunchSetPStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetUStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetUWMAMovingAvg(Average As Double)

UWMA 管理図で、移動平均の範囲を指定する。

LaunchSetUWMASStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

LaunchSetVariableChartParms(Xbar As Boolean, R As Boolean, S As Boolean)

計量値に対する管理図において、Xbar 管理図、R 管理図、S 管理図のいずれを、描画するか (True) か、描画しないか (False) を設定する。

LaunchSetVariableStats(various parms as double) As Boolean

起動に先立って、既知の管理図統計量を追加する。対応する管理図の「管理図」起動ダイアログボックスで設定できるパラメータの値を、引数に指定できます。

Needles(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

SaveLimits() As Datatable

新しいデータテーブルを作成し、管理限界のデータを保存する。作成されたデータテーブルを参照するディスパッチポインタが戻されます。

SetAlarm(jmpControlChartAlarms alarmType) As Boolean

管理図の警告スクリプトを書き込むか、または読み上げるかを設定する。jmpControlChartAlarms 定数は、テストが限界を超えた状態を示したときに呼び出される警告の種類です。

メモ: このメソッドを使用した場合、SetCustomAlarmText(BOOL Speak, BSTR text) As Booleanを使用することはできません。

SetCustomAlarmText(BOOL Speak, BSTR text) As Boolean

標準のメッセージを使わないで、警告テキストを 2 番目のパラメータで渡される文字列に設定する。最初のパラメータは、警告を読み上げ (True) またはログへの書き込み (False) のどちらにするかを指定します。

メモ: このメソッドを使用した場合、SetAlarm(jmpControlChartAlarms alarmType) As Boolean を使用することはできません。

SetActiveChart(chartNumber as Integer) As Boolean

表示された管理図のうち、一番上以外の管理図をアクティブにする。一番上以外の管理図をアクティブにして、何らかの操作を行いたい場合に、このメソッドを使用します。管理図の番号は、上から順に 1 より始まります。このメソッドが呼び出された後に実行されるメソッドは、アクティブな管理図を対象とします。

ShowCenter(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ShowLineLegend(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ShowPoints(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ShowZones(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

Test(TestNumber As Integer, Flag As Boolean)

フラグ (Flag) が True のときは与えられた番号のテストを行い、フラグが False のときはその番号のテストをリセットする。テストがその管理図で適用可能なものでないと機能しません。

TestsAll(Flag As Boolean)

フラグが True のとき、その管理図に対してすべてのテストを実行する。テストがその管理図で適用可能なものであることが条件です。

WestgardRule(jmpControlChartRules ruleNumber, VARIANT_BOOL flag) As Boolean

指定されたルールをオン (True) またはオフ (False) にする。

Crosstabs オブジェクトのメソッド

Crosstabs オブジェクトを使うと、分割表分析の度数表出力を操作できます。

CellChiSquare(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Col(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Count(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Deviation(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Expected(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Row(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Total(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

Diagram オブジェクト

Diagram オブジェクトを使うと、特性要因図を起動することができます。ただし特性要因図はインタラクティブに作成するチャートなので、起動後の操作はインタラクティブに行う必要があります。

メソッド

Diagram オブジェクトに固有のメソッドはありません。

Discriminant オブジェクトのメソッド

Discriminant オブジェクトを使うと、判別分析を起動し、操作することができます。

CanonicalOptions(discrimScoreOptions option, Flag As Boolean) As Boolean

正準プロットのオプションを選択し、それをオン (Flag が True) とオフ (False) のどちらにするか指定します。たとえば、「Show Biplot Rays (バイプロット線の表示)」や「Show Normal 50% Contours (正規 50% 等高線の表示)」などを選択できます。

SaveDiscrimMatrices

このメソッドはパラメータをとりません。

ScatterplotMatrix()

別のウィンドウに散布図行列を生成する。

ScoreData(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

ScoreOptions(discrimScoreOptions option, Flag As Boolean) As Boolean

オプションを選択し、それをオン (Flag が True) とオフ (False) のどちらにするか指定します。たとえば、「Show Classification Counts (度数の表示)」や「Select Uncertain Rows (不確実な行を選択)」などを選択できます。

ScoreSelectUncertainRows(Value As Double) As Boolean

0 や 1 から離れている行を、選択します。引数には、0 または 1 からどれほど点が離れているときに選択するかを表す数値を指定します。

ShowCanonicalPlot(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

ShowGroupMeans(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

ShowWithinCovariances(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

SpecifyPriors(discrimPriorsOptions option)

事前に Equal Probabilities (等しい確率) と Proportional to Occurrence (発生頻度に比例) を設定する。

StepwiseSetup

このメソッドはパラメータをとりません。この StepwiseSetup メソッドは選択パネルを開きますが、パネル内の項目を制御することはできません。そのため StepwiseSetup を呼び出した場合、分析を続けるにはインタラティブな操作が必要になります。これらの点に注意して使用してください。

DistribFit オブジェクトのメソッド

Distribution オブジェクト（「一変量の分布」のオブジェクト）の FitDistribution メソッドから作成される DistribFit オブジェクトを使うと、あてはめた確率分布の結果に対して追加の操作を行えます。

DensityCurve(Flag As Boolean)

表示のオプションを設定 (True) またはリセット (False) する。

GoodnessOfFit(Flag As Boolean)

表示のオプションを設定 (True) またはリセット (False) する。

QuantilePlot(Flag As Boolean)

表示のオプションを設定 (True) またはリセット (False) する。

QuantilePlotAction(distributionFitQuantilePlotConstants action, VARIANT_BOOL flag) As Boolean

分布のあてはめの分位点プロットを操作する。分布のあてはめの分位点プロットを操作する。実行されるアクションの種類（回転や、信頼限界の使用など）は、最初のパラメータで指定します。フラグ (Flag) によって、オプションをオン (True) またはオフ (False) にします。

Quantiles(UpperLimit As Double, LowerLimit As Double, Target As Double)

尺度化も中心化もされていない分布の上側分位点と下側分位点を戻す。

LabelCumPoints(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

RemoveFit()

分析からあてはめを削除する。これをコールすると、オブジェクトポインタは無効となります。

SaveDensityFormula()

データテーブルに新しい列を作成し、密度の値を保存する。

SaveFittedQuantiles()

データテーブルに新しい列を作成し、あてはめた確率分布における分位点の値を保存する。

SpecLimits(lower as Double, upper as Double, target as Double)

工程能力分析を実行する。

Distribution オブジェクトのメソッド

Distribution オブジェクトを使うと、「一変量の分布」の分析を起動し、操作できます。

BetaBinomialFit(Sample Size as Integer, Sample Column as String) As Fit

現在、起動している Distribution オブジェクトにおいて、ベータ二項分布をあてはめる。標本サイズを数値で指定したい場合には、Sample Column には空の文字列 ("") を入力します。標本サイズを含んでいる列を指定したい場合には、Sample Column にその列の名前を指定します。後者の場合には、第 1 パラメータは無視されます。Fit オブジェクトが戻されるので、そのオブジェクトを用いて、追加の処理を行えます。

BinomialFit(Sample Size as Integer, Sample Column as String) As Fit

現在、起動している Distribution オブジェクトにおいて、二項分布をあてはめる。標本サイズを数値で指定したい場合には、Sample Column には空の文字列 ("") を入力します。標本サイズを含んでいる列を指定したい場合には、Sample Column にその列の名前を指定します。後者の場合には、第 1 パラメータは無視されます。Fit オブジェクトが戻されるので、そのオブジェクトを用いて、追加の処理を行えます。

CapabilityAnalysis(LowerLimit As Double, UpperLimit As Double, Target As Double, Sigma As Double)

指定された下側仕様限界、上側仕様限界、目標値、そしてシグマに基づいて、工程能力分析を実行する。

CDFPlot(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ConfidenceInterval(Alpha As Double)

指定された有意水準 (α) の信頼区間を計算する。連続変数の列に対しては、平均と標準偏差に対する信頼区間を計算します。カテゴリカルな列に対しては、割合に対する信頼区間を計算します。

CountAxis(Flag As Boolean)

ヒストグラムの表示オプションで、True ならばオン、False ならばオフ。

DensityAxis(Flag As Boolean)

ヒストグラムの表示オプションで、True ならばオン、False ならばオフ。

ErrorBars(Flag As Boolean)

ヒストグラムの表示オプションで、True ならばオン、False ならばオフ。

FitDistribution(FitType As Integer) As FitDistribution

引数で指定された分布をあてはめる。FitDistribution オブジェクトへのポインタが戻されるので、それによって、あてはめた分布に対する追加の分析を行えます。引数である FitType に指定できる定数は fitDistribConstants に用意されています。

FitNormalMixtures(NumberOfClusters as Integer) As FitDistribution

指定されたクラスター数の正規混合分布をあてはめる。なお、ユーザインターフェイスでは二重正規混合と三重正規混合はメニューで選択できますが、それらをあてはめるには、NumberOfClusters に 2 や 3 を指定してください。

Histogram(Flag As Boolean)

ヒストグラムの表示オプションで、True ならばオン、False ならばオフ。

HorizontalLayout(Flag As Boolean)

フラグ (Flag) が True なら、グラフの向きを縦から横に回転させる。

他にも、連続尺度の変数だけに対して使えるメソッド、および、名義／順序尺度の変数だけに対して使えるメソッドがあります。以下に、それらのメソッドを説明いたします。

Moments(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

MoreMoments(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

MosaicPlot(Flag As Boolean)

このメソッドは、名義／順序尺度の分布に固有で、モザイク図を表示 (True) または非表示 (False) にする。

NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)

名義尺度軸の表示に関するブール値オプションを設定する。プラットフォームで使用可能なオプションをオン (Flag が True) またはオフ (Flag が False) にします。たとえば、Rotated Tick Labels (目盛りラベルの回転)、Divided Lines (分割線)、Lower Frame (下のフレームの表示) などを設定できます。目盛りラベルの回転は、「一元配置」プラットフォームと「変動性図」プラットフォームでのみ使用できます。このメソッドを呼び出す前に、GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long を呼び出して、名義尺度軸ディスプレイボックスへのハンドルを取得する必要があります。

NormalQuantilePlot(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

OutlierBoxPlot(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

PredictionInterval(alpha as Double, nSamples as Long)

予測区間を表示する。このオプションは、*n* 個の標本から計算される平均の予測区間を表示します。

ProbAxis(Flag As Boolean)

ヒストグラムの表示オプションで、True ならばオン、False ならばオフ。

QuantileBoxPlot(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

Quantiles(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

Save(Action As Integer)

各種の分析結果をデータテーブルに保存する。引数のActionに指定できる定数は、**distributionSaveConstants** に用意されています。

SetQuantileIncrement(Increment As Double)

分布が連続量のデータに基づく場合に、分位点の間隔を設定する。このメソッドは、名義尺度や順序尺度のデータには影響しません。

ShowCounts(flag as Boolean)

ヒストグラムに度数を表示する。

ShowPercents(flag as Boolean)

ヒストグラムに割合を表示する。

StemAndLeaf(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

TestMean(meanToTest As Double, Sigma As Double, Wilcoxon As Boolean)

指定された仮説値に対して、平均の検定を行う。

TestMeanWithOptions(meanToTest As Double, Sigma As Double, Wilcoxon As Boolean, PValue As Boolean, Power As Boolean)

平均の検定に、検出力アニメーションまたは p 値アニメーションを実行する機能を加える。PValue または Power パラメータを True にすると、アニメーションが実行され、False になると実行されません。

TestStdDev(stdDeviation As Double)

指定された仮説値に対して、標準偏差の検定を行う。

ToleranceInterval(Alpha as double, Proportion as double)

ユーザインターフェースの Tolerance Interval (許容区間) オプション／ダイアログボックスと同じ機能を持つ。

DOE オブジェクトのメソッド

JMP 6 以降において、実験計画 (DOE) における機能の一部がオートメーション可能になりました。JMP スクリプト言語 (JSL) で使用できるカスタム計画の主な機能は、オートメーションでも使用できます。カスタム計画のオブジェクトを使えるようにするには、まず、DOECustom オブジェクトを生成する必要があります。そのためには、JMP Application オブジェクトの CreateDOECustom () As DOECustom メソッドを呼び出します。

メモ : DOE オートメーションでは順序が重要です。実験計画を作成する操作手順と同じ順序で、メソッドを呼び出す必要があります。たとえば、SimulateResponses() は必ず、テーブルを作成する前に呼び出してください。また、MakeModel(ModelType As doeModelTypes) As Boolean は、MakeDesign() を呼び出す前に呼び出してください。

AddBlockingFactor(NumberOfRuns As Long) As Boolean

ブロック因子を追加する。実験数を指定する必要があります。処理に成功したときは True を、失敗したときは False を戻します。

AddCategoricalFactorWithLevelNames(FactorName as String, LevelNames as Variant Array of Strings) As Boolean

カテゴリカル因子を追加する。因子名と、各水準の名前を指定します。各水準の名前は、文字列型 (String 型) の配列で指定してください。オートメーションクライアントとして使用している開発言語に応じて、たとえば、この配列を Variant 型に宣言してから String 型の配列として再配列化するか、または最初から String 型の配列として宣言してください。JMPとともにインストールされる DOE オートメーションのサンプルプログラムで、メソッドの使い方を確かめてください。

AddBlockingFactorWithName(FactorName As String, NumberOfRuns As Long) As Boolean

ブロック因子を追加する。ブロック因子の因子名として、デフォルトで使われる名前の代わりに、自由に名前を指定できます。第 2 引数には、1 ブロックあたりの実験数を指定してください。

AddCategoricalFactor(NumberOfLevels as Long) As Boolean

カテゴリカル因子を追加する。水準の数を指定する必要があります。

AddCategoricalFactorWithName(FactorName As String, NumberOfLevels as Long) As Boolean

因子名を指定して、カテゴリカル因子を追加する。水準の数を指定する必要があります。

AddContinuousFactorWithBounds(LowerBound As Double, UpperBound As Double) As Boolean

連続尺度の因子を追加する。下限および上限を指定できます。

AddContinuousFactorWithName(FactorName as String, LowerBound As Double, UpperBound As Double) As Boolean

因子名を指定して、連続尺度の因子を追加する。下限および上限を指定する必要があります。

AddFactor(factorType As doeFactorType)

doeFactorType 定数に基づき、因子（連続尺度、カテゴリカル、配合など）をデフォルト設定で追加する。

AddMixtureFactorWithBounds(LowerBound As Double, UpperBound As Double) As Boolean

配合因子を追加する。下限および上限を指定できます。

AddMixtureFactorWithName(FactorName as String, LowerBound As Double, UpperBound As Double) As Boolean

因子名を指定して、配合因子を追加する。下限および上限を指定する必要があります。

AddResponse(ResponseType as doeResponseType, Name as String, LowerLimit As Double, UpperLimit As Double, Importance As Double) As Boolean

応答変数の列を計画に追加する。このメソッドは、`MakeModel(ModelType As doeModelTypes) As Boolean`、`MakeDesign() As Boolean`、`MakeTable() As Boolean` を呼び出す前に実行する必要があります。`doeResponseType` 定数は、目標の種類（最大化、最小化など）を表します。

AddTerms(Terms as Variant Array) As Boolean

交差積（交互作用）やべき乗の項を、モデルに追加する。`AddTerms` の引数は、因子の通し番号を含む配列で指定します。たとえば、 X_1 、 X_2 、 X_3 という因子がこの順番で追加されている場合、 $\{1, 2, 3\}$ と指定すると、 $X_1^*X_2^*X_3$ という交互作用が追加されます。この指定は、`Long` 型の配列で指定する必要があります。オートメーションクライアントとして使用している開発言語に応じて、たとえば、この配列を `Variant` 型に宣言してから `Long` 型の配列として再配列化するか、または最初から `Long` 型の配列として宣言してください。JMPとともにインストールされるDOEオートメーションのサンプルプログラムで、メソッドの使い方を確かめてください。

メモ : `AddTerms` は、`MakeDesign()` を呼び出す前に呼び出す必要があります。

AddTermsWithPowers(Terms as Variant Array, Powers as Variant Array) As Boolean

各因子を何乗するかの指定によって、交差積（交互作用）やべき乗の項を、モデルに追加する。因子の引数 (Terms) と、べき乗の引数 (Powers) は対応している必要があります。たとえば、 $X_1^*X_2^2*X_3^4$ という項を追加したい場合、`Terms` には $\{1, 2, 3\}$ と指定し、`Powers` には $\{1, 2, 4\}$ と指定してください。

各引数は、`Long` 型の配列で指定する必要があります。オートメーションクライアントとして使用している開発言語に応じて、たとえば、この配列を `Variant` 型に宣言してから `Long` 型の配列として再配列化するか、または最初から `Long` 型の配列として宣言してください。JMPとともにインストールされるDOEオートメーションのサンプルプログラムで、メソッドの使い方を確かめてください。

メモ : `AddTermsWithPowers` は、`MakeDesign()` を呼び出す前に呼び出す必要があります。

LoadResponses(Table as DataTable) As Boolean

既存のデータテーブルから計画の応答をロードする。

テーブルを事前に `OpenDocument(FileName As String) As Document` を使ってロードし、`OpenDocument::GetDataTable`から戻されたオブジェクトをこのメソッドに渡す必要があります。戻り値が `True`ならば成功、`False`ならば失敗を意味します。このメソッドは、`MakeModel(ModelType As doeModelTypes) As Boolean`、`MakeDesign() As Boolean`、`MakeTable() As Boolean` を呼び出す前に呼び出す必要があります。

以下の 2 つのメソッドは、このメソッドに関連しています。

- `LoadFactors(Table as DataTable) As Boolean`
- `LoadConstraints(Table as DataTable) As Boolean`

LoadFactors(Table as DataTable) As Boolean

既存のデータテーブルから計画の因子をロードする。

テーブルを事前に `OpenDocument(fileName As String) As Document` を使ってロードし、`OpenDocument::GetDataTable` から戻されたオブジェクトをこのメソッドに渡す必要があります。戻り値が `True` ならば成功、`False` ならば失敗を意味します。このメソッドは、`MakeModel(ModelType As doeModelTypes) As Boolean`、`MakeDesign()`、`MakeTable() As Boolean` を呼び出す前に呼び出す必要があります。

以下の 2 つのメソッドは、このメソッドに関連しています。

- `LoadResponses(Table as DataTable) As Boolean`
- `LoadConstraints(Table as DataTable) As Boolean`

LoadConstraints(Table as DataTable) As Boolean

既存のデータテーブルから計画の制約をロードする。

テーブルを事前に `OpenDocument(fileName As String) As Document` を使ってロードし、`OpenDocument::GetDataTable` から戻されたオブジェクトをこのメソッドに渡す必要があります。戻り値が `True` ならば成功、`False` ならば失敗を意味します。このメソッドは、`MakeModel(ModelType As doeModelTypes) As Boolean`、`MakeDesign()`、`MakeTable() As Boolean` を呼び出す前に呼び出す必要があります。

以下の 2 つのメソッドは、このメソッドに関連しています。

- `LoadResponses(Table as DataTable) As Boolean`
- `LoadFactors(Table as DataTable) As Boolean`

MakeDesign()

計画を作成する。因子を追加し、`MakeModel(ModelType As doeModelTypes) As Boolean` を呼び出して、続いてモデルの項を追加した後に、このメソッドを呼び出してください。

MakeModel(ModelType As doeModelTypes) As Boolean

RSM (応答曲面)、Linear (線形)、Interactions (交互作用) などの、モデルの種類を表す定数に基づき、モデルを作成する。このメソッドは、因子を追加した後に、かつ、`MakeDesign()` と `MakeTable() As Boolean` を呼び出す前に、呼び出す必要があります。

MakeTable() As Boolean

計画を含んだデータテーブルを生成する。このメソッドより前に `SimulateResponses()` が呼び出されている場合は、作成されたデータテーブルの応答に乱数シミュレーションの値が代入される。

NumberOfCenterpoints(nCenterpoints As Long) As Boolean

中心点の数を指定する。このメソッドは、`MakeTable() As Boolean` を呼び出す前に呼び出してください。

NumberOfReplicates(nReplicates as Long) As Boolean

反復の回数を指定する。このメソッドは、`MakeTable() As Boolean` を呼び出す前に呼び出してください。

NumberOfStarts(nStarts As Long) As Boolean

最適計画を探索するのに用いる、ランダムな開始計画の数を指定する。引数には、正の整数を指定してください。このメソッドは、`MakeDesign()` を呼び出す前に実行してください。

OptimalityCriterion(Criterion as doeOptimalityConstants) As Boolean

最適化基準を明示的に指定する。自動的に選択されている基準とは別の最適化基準を選択したいときに使ってください。また、このメソッドは、`MakeDesign()` を呼び出す前に呼び出してください。D-Optimal または I-Optimal を指定できます。

SaveFactors()

計画の因子を新しいデータテーブルに保存する。

SaveXMatrix()

作成されたデータテーブルのテーブル変数に、計画行列を保存する。このメソッドはオンとオフが切り替わります。最初に呼び出したときはオンになります。もう一度呼び出すと、オフになります。

SetRandomSeed(Seed As Double) As Boolean

計画開始点を決めるのに使われる乱数のシード値を、正の整数で指定する。シード値として -1 を入力すると、シード値を入力するためのプロンプトが表示されます。このメソッドは、`MakeDesign()` を呼び出す前に呼び出す必要があります。

ShowDiagnostics()

診断統計量をオンにする。このメソッドはオンとオフが切り替わります。最初に呼び出したときはオンになります。もう一度呼び出すと、オフになります。

SimulateResponses()

計画のデータテーブルを作成するときに、応答の値を乱数でシミュレートする。このメソッドは、`MakeTable() As Boolean` を呼び出す前に呼び出す必要があります。このメソッドはオンとオフが切り替わります。最初に呼び出したときはオンになります。次回に呼び出したときはオフになっています。

SpecifyChangeDifficulty(doeChangeDifficultyConstants difficulty)

DOE オートメーションにおける因子変更の難易度 (Easy (容易)、Hard (困難)) を変更する。なお、因子は、次のようなメソッドによって追加できます。

- `AddCategoricalFactorWithLevelNames(FactorName as String, LevelNames as Variant Array of Strings) As Boolean`

- `AddCategoricalFactorWithName(FactorName As String, NumberOfLevels as Long) As Boolean`
- `AddContinuousFactorWithName(FactorName as String, LowerBound As Double, UpperBound As Double) As Boolean`
- `AddMixtureFactorWithName(FactorName as String, LowerBound As Double, UpperBound As Double) As Boolean`

SphereRadius(Radius as Double)

球面計画における領域の半径を正の数値で指定する。

Fit オブジェクトのメソッド

Fit オブジェクトを使うと、「二変量の関係」におけるあてはめの結果に対して、追加の処理を行えます。このオブジェクトは、Bivariate オブジェクトのメソッドのうちのあてはめを行うものから戻されます。

ConfidenceFit(Flag As Boolean)

平均に対する信頼区間のオプションをオン (1 または `True`)、またはオフ (0 または `False`) にする。

ConfidenceIndividual(Flag As Boolean)

個別の値に対する信頼区間のオプションをオン (1 または `True`)、またはオフ (0 または `False`) にする。

LineOfFit(Flag As Boolean)

あてはめ線のオプションをオン (1 または `True`)、またはオフ (0 または `False`) にする。

PlotResiduals(Flag As Boolean)

パラメータの値が `True` のとき、残差プロットを出力する。

RemoveFit()

二変量の出力からあてはめを削除する。

SavePredicteds()

データテーブル内に新しい列を作成し、Y の予測値を保存する。

SaveResiduals()

データテーブル内に新しい列を作成し、Y の残差を保存する。

SetAlpha(Alpha As Double)

信頼区間の有意水準 (α) を設定する。

SplineSaveCoeffs()

スプライン係数を元のデータテーブルに保存する。スプライン曲線のあてはめによって Fit オブジェクトを得た後に、このメソッドを使用できます。データテーブルオブジェクトを戻します。SplineSavePredFormula() As DataTable も参照してください。

SplineSavePredFormula() As DataTable

スプライン予測式を新しいデータテーブルに保存する。スプライン曲線のあてはめによって Fit オブジェクトを得た後に、このメソッドを使用できます。SplineSaveCoeffs() も参照してください。

FitDensity オブジェクトのメソッド

FitDensity オブジェクトを使うと、ノンパラメトリックな密度のあてはめの出力をさらに操作できます (NonParDensity() As FitDensity を参照)。

FivePercentContours(Flag As Boolean)

オン (True) / オフ (False) オプションで、対応するオプションをインタラクティブに操作したときと同じ働きをする。

KernelControl(Flag As Boolean)

オン (True) / オフ (False) オプションで、対応するオプションをインタラクティブに操作したときと同じ働きをする。

MeshPlot(Flag As Boolean)

オン (True) / オフ (False) オプションで、対応するオプションをインタラクティブに操作したときと同じ働きをする。

ModalClustering(Flag As Boolean)

オン (True) / オフ (False) オプションで、対応するオプションをインタラクティブに操作したときと同じ働きをする。

SaveDensityGrid() As DataTable

密度推定値とその分位点を新しいデータテーブルに保存する。この新しいデータテーブルへのディスパッチポインタが戻され、データテーブルがオートメーション可能になります。

FitLeastSquares オブジェクトのメソッド

以下のメソッドは、FitModel オブジェクトで選択された手法が標準最小2乗の場合に、Launch メソッドの呼び出しから戻されます。FitLeastSquares オブジェクトを使うと、プロファイルを呼び出すことができます。なお、複数の応答がある場合は、Response オブジェクトで応答を選択することにより、特定の応答を操作できるようになります。

ContourProfiler(Flag As Boolean) As Boolean

等高線プロファイルをオン (True) またはオフ (False) にする。処理に成功したときは True を、失敗したときは False を戻します。

CubePlot(Flag As Boolean) As Boolean

キューブプロットをオン (True) またはオフ (False) にする。処理に成功したときは True を、失敗したときは False を戻します。

GetResponse(Name As String) As FitResponse

出力された結果における応答変数のうち、指定された列名の応答変数に対する Response オブジェクトを戻す。モデルの起動前に入力されたすべての Y 列に対して、Response オブジェクトがあります。

Profiler(Flag As Boolean) As FitProfiler

プロファイルをオン (True) またはオフ (False) にする。プロファイル設定をさらに細かく操作できる FitProfiler オブジェクトが戻されます。

FitLogvariance オブジェクトのメソッド

あてはめの手法に Loglinier Variance(対数線形-分散)を指定したときのメソッドを提供します。

ConfidenceInterval(Alpha As Double) As Boolean

信頼区間を指定する。

LikelihoodRatio(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

MarginalVariances (Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

FitManova オブジェクトのメソッド

MANOVA あてはめ手法の「応答の指定」ダイアログボックスは高度に対話型なので、この手法に関してはあまりサポートされていません。ただし、アクティブなデータテーブルに値を保存することはできます。

SaveDiscrim() As Boolean

残差を、アクティブなデータテーブルの列に保存する。

SavePredicted() As Boolean

残差を、アクティブなデータテーブルの列に保存する。

SaveResiduals() As Boolean

残差を、アクティブなデータテーブルの列に保存する。

FitModel オブジェクトのメソッド

FitModel オブジェクトのメソッドは、「モデルのあてはめ」の起動と、それに続く分析結果の操作に使われます。ただし、「MANOVA」の手法に関しては、起動後の操作が高度に対話型でデータ固有であるため、起動後の操作のほとんどがサポートされていません。

メモ:

FitModel は、FitLeastSquares、FitManova などの分析結果のオブジェクトを生成します。ただし、By グループの操作を行うための共通メソッド UseByOutput(ByTitle As String) As Boolean は、結果が複雑であるため、FitModel では使えません。

Launch メソッドは、「モデルのあてはめ」を起動します。なお、LaunchAddY と LaunchAddWeight は、応答と重みの列を指定するのですが、「比例ハザード」と「生存時間分布のあてはめ」プラットフォームに限っては、起動ダイアログの [イベントまでの時間] ボタンと [打ち切り] ボタンに相当します。

次の 4 つのメソッドは、AddxxxEffect メソッドで作成された効果の名前の検索、選択された効果の削除、および枝分かれ効果に使われます。

- LaunchGetNumberOfEffects() As Integer
- LaunchGetEffectName(EffectNumber As Integer) As String
- LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean
- LaunchRemoveSelectedEffects() As Boolean

Launch() As Object

指定された設定に基づいて、「モデルのあてはめ」を起動する。いくつかの手法では、起動後の分析結果を追加で操作できるように、Fit オブジェクトが戻されます。Fit オブジェクトが戻されるものは、FitLeastSquares、FitStepwise、FitNominal、FitOrdinal、FitLogVariance、FitProportional、および FitParametricSurvival です。MANOVA 手法は複雑な対話型な操作が必要であり、オートメーション用のオブジェクトは戻されません。

LaunchAddCrossEffect() As Boolean

LaunchAddToEffectList(Name As String) As Boolean によって指定された列から、交差の効果を作成する。

LaunchAddMacroEffect(*fitModelMacroEffectConstants macroType*) As Boolean

LaunchAddToEffectList(Name As String) As Boolean によって指定された列から、引数で指定されたマクロの効果を作成する。マクロの次数（[次数] のフィールドに相当するもの）は、LaunchSpecifyMacroDegree を使って指定します。MacroType 引数に指定する効果の種類は、*fitModelMacroEffectConstants* という定数に用意されています。次数が指定されていないときのデフォルトは、2 次です。

LaunchAddNestEffect() As Boolean

LaunchAddToEffectList(Name As String) As Boolean によって指定された列を、LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean によって指定された列内からの枝分かれ効果として指定する。

たとえば、各処置Dose内で被験者IDを枝分かれ効果として指定する場合を考えます。まず、列「ID」をX効果として追加します。これには、LaunchAddToEffectList(Name As String) As Boolean の次にLaunchAddXEffect() As Booleanを実行します。次に、LaunchAddToEffectList(Name As String) As Booleanを使って列「Dose」を追加し、LaunchSelectEffectで「ID」を選択します。最後に、LaunchAddNestEffect() As Booleanを実行すると、ID[Dose]が作成されます。

LaunchAddToEffectList(Name As String) As Boolean

Name で指定された列を、効果列のリストに追加する。この効果列のリストは、LaunchAddXEffect、LaunchAddNestEffect、LaunchAddCrossEffect、および LaunchAddMacroEffect メソッドによってモデル効果を作成するときに使われます。これらのメソッドによってモデル効果が作成されると、リストは空にされます。

LaunchAddXEffect() As Boolean

主効果を作成する。これは、「モデルのあてはめ」の起動ダイアログボックスにおける【追加】ボタンを使うのと同じです。効果列のリストに現在含まれている列が使われます。モデル効果として含めたい列を、LaunchAddToEffectList(Name As String) As Boolean によって、効果列のリストに追加しておいてください。

LaunchAddXEffectWithTransform(transform as fitModelTransforms)

列に対して、変換の種類を指定する。たとえば、Exponential (指数) や Square (2乗) などを指定します。変換したい列は、事前にLaunchAddToEffectList(Name As String) As Booleanで追加しておく必要があります。

LaunchAddYWithTransform(ColumnName As String, transform as fitModelTransforms) As Boolean

分析のために Y 変数を追加し、Log (対数) や Sqrt (平方根) などの変換を行う。処理に成功したときは True を、失敗したときは False を戻します。

LaunchGetEffectName(EffectNumber As Integer) As String

特定の効果を識別する文字列を戻す。たとえば、「Height」と「Weight」という列の交差効果として、Height*Weight が戻されます。

このメソッドや次の 3 つのメソッドは、AddxxxEffect メソッドで作成された効果の名前を検索したり、選択した効果を削除したり、また、枝分かれ効果を作成するのに使われます。

- LaunchGetNumberOfEffects() As Integer
- LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean
- LaunchRemoveSelectedEffects() As Boolean

LaunchGetNumberOfEffects() As Integer

各種の `AddEffect` メソッドを使って作成された効果の個数を戻す。現在モデルに含まれている効果の個数を戻します。

このメソッドや次の 3 つのメソッドは、`AddxxxEffect` メソッドで作成された効果の名前を検索したり、選択した効果を削除したり、また、枝分かれ効果を作成するのに使われます。

- `LaunchGetEffectName(EffectNumber As Integer) As String`
- `LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean`
- `LaunchRemoveSelectedEffects() As Boolean`

LaunchRemoveFromEffectList(Name As String) As Boolean

`Name` で指定された列を、効果列リストから削除する。

LaunchRemoveSelectedEffects() As Boolean

`LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean` によって選択されている効果をすべて削除する。削除された効果は、推定するモデルから除かれます。このメソッドを呼び出した後は、選択の状態はリセットされます。

このメソッドや次の 3 つのメソッドは、`AddxxxEffect` メソッドで作成された効果の名前を検索したり、選択した効果を削除したり、また、枝分かれ効果を作成するのに使われます。

- `LaunchGetNumberOfEffects() As Integer`
- `LaunchGetEffectName(EffectNumber As Integer) As String`
- `LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean`

LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean

`EffectNumber` で識別される特定の効果を内部リストに追加する。追加された効果は、後で `LaunchRemoveSelectedEffects() As Boolean` を使って削除したり、

`LaunchSpecifyAttributesForSelectedEffects(fitModelEffectAttributeConstants attribNumber) As Boolean` を使って属性を指定したりできます。

このメソッドや次の 3 つのメソッドは、`AddxxxEffect` メソッドで作成された効果の名前を検索したり、選択した効果を削除したり、また、枝分かれ効果を作成するのに使われます。

- `LaunchGetNumberOfEffects() As Integer`
- `LaunchGetEffectName(EffectNumber As Integer) As String`
- `LaunchRemoveSelectedEffects() As Boolean`

LaunchSpecifyAttributesForSelectedEffects(fitModelEffectAttributeConstants attribNumber) As Boolean

`LaunchSelectEffect(EffectNumber As Integer, OnOffFlag As Boolean) As Boolean` を使って選択されている効果に属性を指定する。これは、「モデルのあてはめ」ダイアログボックスの [属性] ポップアップメニューに対応します。効果には、`Mixture Effect` (配合効果) や `Random Effect` (変量効果) などがあります。

効果の種類は、`fitModelEffectAttributeConstants` のうち 1 つを指定します。効果リストに現在含まれているすべての効果に、この属性が適用されます。その後で、選択状態はリセットされます。

LaunchSpecifyDistribution(fitModelDistributionConstants) As Boolean

`Parametric Survival`（「生存時間（パラメトリック）」）が手法として選択されているときに、その確率分布を指定する。選択できる確率分布は、`Weibull`、`LogNormal`（対数正規）、および `Exponential`（指数）です。

`fitModelDistributionConstants`。手法が「生存時間（パラメトリック）」でない場合、この設定は無視されます。

LaunchSpecifyEmphasis(fitModelEmphasisConstants emphasis) As Boolean

`Standard Least Squares`（「標準最小 2 乗」）が手法として選択されているときに、強調点（`emphasis`）を指定する。強調点は、「モデルのあてはめ」ダイアログボックスのドロップダウンリストに対応します。選択できるのは、`Effect Leverage`（効果てこ比）、`Effect Screening`（要因のスクリーニング）、および `Minimal Report`（最小レポート）です。手法が「標準最小 2 乗」でない場合、この設定は無視されます。

LaunchSpecifyIntercept(Flag As Boolean)

切片をモデルに含めるか（`True`）、または、モデルから除くか（`False`）を切り替える。デフォルトでは、切片なし（`False`）になっています。

LaunchSpecifyPersonality(fitModelPersonalityConstants personality) As Boolean

あてはめの手法を指定する。手法には、`Standard Least Squares`（標準最小 2 乗）、`Loglinear Variance`（対数線形分散）、`Parametric Survival`（生存時間（パラメトリック））などがあります。`Standard Least Squares`（標準最小 2 乗）がデフォルトの手法です。

一部の手法には特定の列タイプが必要です。たとえば、`Ordinal Logistic`（順序ロジスティック）では、応答変数の列が順序尺度でなければいけません。選択された手法に適合しない応答変数が追加されると、JMP はデータに適合するように手法を変更します。指定できる手法は、`fitModelPersonalityConstants` に定数として用意されています。

LaunchSpecifyRandomEffectMethod(method as fitModelRandomEffectMethods) As Boolean

`REML`（推奨メソッド、デフォルト）または `EMS`（従来のメソッド）を指定する。処理に成功したときは `True` を、失敗したときは `False` を戻します。

UseByFit(Name As String) As Fit

指定された名前に関連付けられた `By` グループのあてはめ出力結果を見つけ出し、その `Fit` オブジェクトへの参照を戻す。

たとえば、年齢を「`By`」グループに指定して `FitLeastSquares` を起動したとします。Launch メソッドは、分析結果における最初の `FitLeastSquares` オブジェクトへの参照を戻します。

`UseByFit(Name As String) As Fit`を使うと、他の分析結果のオブジェクトへの参照を戻します。戻されるオブジェクトの種類は、選択された手法によって異なります。たとえば、応答変数が順序尺度の場合、このメソッドは `FitOrdinal` オブジェクトへの参照を戻します。このメソッドは、`Launch` メソッド呼び出しから戻されるオブジェクトではなく、元の `FitModel` オブジェクトのメソッドであることに注意してください。

Fit Model オートメーションのサンプルプログラムに、このメソッドを使った例があります。

FitNominal オブジェクトのメソッド

FitNominal オブジェクトは、あてはめの手法として「名義ロジスティック」を指定したときのメソッドを提供します。

InversePrediction() As Boolean

このアクションはオフにできないことに注意してください。`InversePrediction() As Boolean` は、ユーザ入力が必要なダイアログボックスを表示します。`InversePrediction() As Boolean` の値をオートメーションで設定することはできません。

LikelihoodRatioTests(Flag As Boolean) As Boolean

対応するオプションをオン (`True`) またはオフ (`False`) にする。

OddsRatios(Flag As Boolean) As Boolean

対応するオプションをオン (`True`) またはオフ (`False`) にする。

Profiler(Flag As Boolean)

予測プロファイルをオン (`True`) またはオフ (`False`) にする。

ROCCurve(Flag As Boolean) As Boolean

対応するオプションをオン (`True`) またはオフ (`False`) にする。

SaveProbFormula() As Boolean

このアクションはオフにできないことに注意してください。確率計算式を現在のデータテーブルに保存します。`ConfidenceIntervals(Alpha As Double) As Boolean` は、信頼区間の有意水準を変更できます。

FitOrdinal オブジェクトのメソッド

FitOrdinal オブジェクトは、あてはめの手法として「順序ロジスティック」を指定したときのメソッドを提供します。

ConfidenceIntervals(Double As Alpha) As Boolean

信頼区間の有意水準を指定する。

LikelihoodRatioTests(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

SaveExpectedValue() As Boolean

指定された情報を現在のデータテーブルに保存する。

SaveProbFormula() As Boolean

指定された情報を現在のデータテーブルに保存する。

SaveQuantiles() As Boolean

指定された情報を現在のデータテーブルに保存する。

FitParametricSurvival オブジェクトのメソッド

FitParametricSurvival オブジェクトは、あてはめの手法として「生存時間 (パラメトリック)」を指定したときのメソッドを提供します。

ConfidenceIntervals(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

CorrelationOfEstimates(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

CovarianceOfEstimates(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

EstimateSurvivalProbability()

EstimateSurvivalProbability() および EstimateTimeQuantile() は、対話的な入力フィールドを表示する。2つのオプションのどちらか1つを指定できます。

EstimateTimeQuantile()

EstimateSurvivalProbability() および EstimateTimeQuantile() は、対話的な入力フィールドを表示する。2つのオプションのどちらか1つを指定できます。

LikelihoodRatioTests(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

FitProfiler オブジェクトのメソッド

InteractionProfiler(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

FitProportional オブジェクトのメソッド

「比例ハザードモデル」のオブジェクトには固有のメソッドはありません。ただし、FitModel オブジェクトから戻される FitProportional オブジェクトには、他のオブジェクトと共にしているメソッドが用意されています。

メソッド

FitProportional オブジェクトに固有のメソッドはありません。

FitResponse オブジェクトのメソッド

応答固有のコマンドをメソッドとして提供します。たとえば、Effects Screening(要因のスクリーニング) や Estimates(推定値) などのメニューoptionがあります。

BoxCoxY(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

CorrelationOfEstimates(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

ExpandedEstimates(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

GetEffectAnalysis(Name As BSTR) As FitEffect

標準最小 2 乗法による分析結果において、応答変数内の特定の効果に対する参照を戻す。効果名を指定してこのメソッドを呼び出すことで、各効果の分析への参照を取得でき、その効果に対する追加の処理を行える。

FitEffect オブジェクトのメソッドを使うと、GetEffectAnalysis から戻されるオブジェクトを使って、追加の処理を行えます。これらのメソッドは、分析結果の [効果] ポップアップメニューに対応します。

といった枝分かれ項の効果も取得できます。Silica*Silane*Sulfur といった交互作用項や、drug[Placebo,Gender]。

InteractionPlots(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

LSMeansPlot(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

LSMeansStudents(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

LSMeansTable(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

LSMeansTukey(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。

NormalPlot(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

ParameterPower(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

ParetoPlot(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

RowDiagnostics(fitModelRowDiagConstants diagType, VARIANT_BOOL Flag) As Boolean

特定の診断統計量のオプションをオンまたはオフにする。最初のパラメータに指定できる値は、`fitModelRowDiagConstants` に定数として用意されています。Flag パラメータは、オプションをオン (True) またはオフ (False) にします。

SaveColumns(fitModelSaveColumnConstants saveType) As Boolean

選択された統計量を列に保存する。保存される列名は、通常、接頭部が統計量の種類、接尾部が応答名になっています。使用できる統計量の種類は、`fitModelSaveColumnConstants` に用意されています。

ScaledEstimates(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

SequentialTests(Flag As Boolean) As Boolean

対応するオプションをオン (True) またはオフ (False) にする。このオプションに関する情報は、『基本的な回帰モデル』ドキュメントの「モデルのあてはめ」に関する章に記載されています。

TestSlices() As Boolean

輪切り検定（単純主効果検定）オプションをオンにする。

メモ：これは、一度呼び出したらオフにできないアクションです。

FitStepwise オブジェクトのメソッド

FitStepwise オブジェクトのメソッドを使うと、インタラクティブな方法と同じようにステップワイズ回帰を操作できます。

AllPossibleModels()

すべての組み合わせの線形モデルをあてはめて、その結果を表示する。

AllPossibleModelsWithParameters(NMaximumTerms As Integer, NBestModelsToSee As Integer, HeredityRestriction As Boolean)

AllPossibleModelsWithParameters の引数で指定された設定で、[すべてのモデル] を実行する。NMaximumTerms は、モデルに含める項の最大個数。NBestModelsToSee は、表示する最良モデルの数。HeredityRestriction は、「親子関係」制約のオン／オフを切り替える。

EnterAll() As Boolean

ロックされていない効果をすべてモデルに追加する。

EnterEffect(EffectNumber As Integer, Flag As Boolean) As Boolean

最初の引数に指定された番号に対応する効果を、モデルに追加（フラグは `True`）または削除（フラグは `False`）する。

GetEffectName(EffectNumber As Integer) As String

引数に指定された番号に対応する効果の名前を、文字列で戻す。

GetNumberOfEffects() As Short

「現在の推定値」表にある効果の数を戻す。たとえば、ここから戻された整数をもとに、GetEffectName(EffectNumber As Integer) As String を反復して呼び出すことにより、すべての効果名を取得できます。

Go() As Boolean

変数選択の処理を開始し、完了するまでバックグラウンドで実行する。

LockEffect(EffectNumber As Integer, Flag As Boolean) As Boolean

最初の引数に指定された番号に対応する効果を、ロック（フラグは `True`）またはロック解除（フラグは `False`）する。

RemoveAll() As Boolean

効果をすべてモデルから削除（選択解除）する。

SetDirection(*fitStepDirectionConstants* *Direction*)

変数をモデルに出し入れする方法を指定する。引数 *Direction* に指定できる定数は、*fitStepDirectionConstants* に用意されています。Forward (変数増加)、Backward (変数減少)、Mixed (変数増減) のいずれかを指定できます。

SetProbToEnter(*Value* As Double) As Boolean

変数を追加するときの *p* 値を数値で設定する。これらの値については、『基本的な回帰モデル』の「ステップワイズ回帰」を参照してください。

SetProbToLeave(*Value* As Double) As Boolean

変数を除去するときの *p* 値を数値で設定する。これらの値については、『基本的な回帰モデル』の「ステップワイズ回帰」を参照してください。

SetRules(*fitStepRulesConstants* *Rules*)

「ステップワイズ法」ダイアログボックスでの操作と同じように、「ルール」を指定する。引数 *Rules* に指定できる定数は、*fitStepRulesConstants* に用意されています。Combine (組み合わせ)、Restrict (制限)、No Rules (ルールなし)、および Whole Effect (効果全体) のどれかである必要があります。

Step() As Boolean

ステップワイズプロセスの 1 ステップを処理し、停止する。

Stop() As Boolean

バックグラウンドでの選択処理を停止する。

Gaussian Process メソッド

Gaussian Process オブジェクトを使うと、Gaussian 過程分析を起動し、操作することができます。

ContourProfiler(Flag as Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

LaunchEstimateNuggetParameter(Flag as Boolean) As Boolean

対応する起動オプションをオン (True) またはオフ (False) にする。

LaunchSpecifyCorrelationType(*Type* as jmpGaussianCorrelationConstant) As Boolean

相関タイプに Gaussian 過程または 3 次モデルを指定する。

LaunchSpecifyMinimumTheta(*Theta* as Double) As Boolean

Gaussian 過程の起動ダイアログと同様、起動前に *Theta* 値を指定する。

Profiler(Flag as Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

SaveJackknifePredictedValues()

この計算式を現在のデータテーブルに保存する。

SavePredictionFormula()

この計算式を現在のデータテーブルに保存する。

SaveVarianceFormula()

この計算式を現在のデータテーブルに保存する。

SurfaceProfiler(Flag as Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

Cluster オブジェクトの階層型クラスター固有メソッド

「階層型クラスター」プラットフォームには、階層型クラスター分析だけで使えるメソッドがあります。Cluster オブジェクトのメソッドも使えます。

ColorMap(clusterColormapConstants mapType) As Boolean

データからカラーマップを生成する。このメソッドは、「クラスター分析」プラットフォームから使用できる機能に対応します。

DistanceGraph(Flag As Boolean) As Boolean

距離グラフをオン (True) またはオフ (False) にする。

GeometricXScale(Flag As Boolean) As Boolean

[X 軸を等比スケールに] オプションをオン (True) またはオフ (False) にする。

LaunchAddLabel(ColumnName As String) As Boolean

ラベル列を分析に追加する。処理に成功したときは True を、失敗したときは False を戻します。

LaunchAddOrdering(ColumnName As String) As Boolean

順序列を分析に追加する。処理に成功したときは True を、失敗したときは False を戻します。

LaunchRemoveLabel(ColumnName As String) As Boolean

ラベル列を分析から除外する。処理に成功したときは True を、失敗したときは False を戻します。

LaunchRemoveOrdering(ColumnName As String) As Boolean

順序列を分析から除外する。処理に成功したときは True を、失敗したときは False を戻します。

SaveClusterHierarchy() As Boolean

スクリプトでカスタムの樹形図を作成するときに必要な情報を保存する。このメソッドは、クラスター分析ごとに、joiner (結合者)、leader (結合先)、および result (結果) の3行を、クラスターの中心、サイズ、およびその他の情報とともに戻します。

SaveDisplayOrder()

各行の深さ (順位) をデータテーブルの新しい列に保存する。

SetOrientation(clusterOrientationConstants orientation) As Boolean

樹形図の方向を指定する。方向は、left (左)、right (右)、top (上)、bottom (下) のどれかです。

StandardizeData(Flag As Boolean)

列の平均と標準偏差によってデータを標準化するかどうかを指定する。Launch メソッドを起動する前にこのメソッドを呼び出しておく必要があります。デフォルトは True なので、データの標準化が不要なときは、False の値を指定してください。

TwoWayClustering

変数間クラスターを実行する。

ItemAnalysis オブジェクトのメソッド

LaunchSpecifyModel(Model as itemAnalysisModelConstants)

2PL など、使用するモデルの種類を指定する。このメソッドは、Launch メソッドによって分析を起動する前に実行する必要があります。

NumberOfPlotsAcross(Number as Integer)

縦方向に表示するプロットの数を指定する。

SaveAbilityFormula()

式を現在のデータテーブルに保存する。

Cluster オブジェクトの k-means クラスター固有メソッド

「k-means」プラットフォームには、k-means 法のクラスター分析だけで使えるメソッドがあります。Cluster オブジェクトのメソッドも使えます。

KMGo()

クラスター分析を実行する。

なお、KMShiftDistances(Flag As Boolean) や KMwithinClusterStdDev(Flag As Boolean) で設定を行いたい場合は、このメソッドや KMStep() を呼び出す前に実行しておいてください。

KMSaveMixtureFormulas()

混合分布の確率の計算式を、現在のデータテーブルに保存する。

KMSaveMixtureProbs()

混合分布の確率を、現在のデータテーブルに保存する。

KMSeedWithSelectedRows() As Boolean

クラスター平均の最初の推定値が含まれている行を指定する。このメソッドを呼び出す前に、データテーブル内の行を選択しておく必要があります。

KMSimulateMixtures(long numberOfRows)

[クラスターのシミュレーション] メニュー オプションを操作したときと同じ働きをする。numberOfRows パラメータは、シミュレーションされる行数です。

KMShiftDistances(Flag As Boolean)

点が大きなクラスターに優先的に割り当てられるように指定する。デフォルトは `False` です。

メモ: このメソッドや `KMWithinClusterStdDev(Flag As Boolean)` で設定を行いたい場合は、`KMGo()` や `KMStep()` を呼び出す前に実行しておいてください。

KMStep()

クラスター分析の反復を 1 回実行し、値を検証できるようにする。

なお、`KMShiftDistances(Flag As Boolean)` や `KMWithinClusterStdDev(Flag As Boolean)` で設定を行いたい場合は、このメソッドや `KMGo()` を呼び出す前に実行しておいてください。

KMWithinClusterStdDev(Flag As Boolean)

クラスター内の標準偏差によって距離成分を標準化する。デフォルトは `False` です。

このメソッドや `KMShiftDistances(Flag As Boolean)` で設定を行いたい場合は、`KMGo()` や `KMStep()` を呼び出す前に実行しておいてください。

LaunchAddFreq(ColumnName As String) As Boolean

度数列を分析に追加する。処理に成功したときは `True` を、失敗したときは `False` を戻します。

LaunchAddWeight(ColumnName As String) As Boolean

重み列を分析に追加する。処理に成功したときは `True` を、失敗したときは `False` を戻します。

LaunchRemoveFreq(ColumnName As String) As Boolean

度数列を分析から除外する。処理に成功したときは `True` を、失敗したときは `False` を戻します。

LaunchRemoveWeight(ColumnName As String) As Boolean

重み列を分析から除外する。処理に成功したときは `True` を、失敗したときは `False` を戻します。

StandardizeData (Flag As Boolean)

データを標準化する (True) または標準化しない (False) を指定する。このメソッドは、プラットフォームを起動するための呼び出しの前または後に使います。

Logistic オブジェクトのメソッド

Logistic オブジェクトを使用すると、ロジスティック回帰分析を起動し、操作できます。

InversePrediction()

逆推定を要求し、「逆推定」ダイアログボックスを作成する。

LiftCurve(Flag as Boolean)

リフトチャートの表示をオン (True) またはオフ (False) にする。

LineColor(Color as jmpColorConstants)

ロジスティック曲線の色を変更する。

LogisticPlot(Flag As Boolean)

ロジスティックプロットの作成をオン (True) またはオフ (False) にする。

NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)

名義尺度軸の表示に関するブール値オプションを設定する。プラットフォームで使用可能なオプションをオン (Flag が True) またはオフ (Flag が False) にします。たとえば、Rotated Tick Labels (目盛りラベルの回転)、Divided Lines (分割線)、Lower Frame (下のフレームの表示) などを設定できます。目盛りラベルの回転は、「一元配置」プラットフォームと「変動性図」プラットフォームでのみ使用できます。このメソッドを呼び出す前に、GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long を呼び出して、名義尺度軸ディスプレイボックスへのハンドルを取得する必要があります。

RateCurve()

応答率折れ線を表示する。

ROCCurve(Flag As Boolean)

ROC 曲線を描く。

ROCSetPositiveLevel(LevelValue As String) As Boolean

Y 変数におけるデータ値のうち、いずれが陽性を示しているかを設定する。このメソッドは、ROCCurve メソッドを呼び出す前に、実行してください。処理に成功したときは True を、失敗したときは False を戻します。このメソッドが ROCCurve メソッドより後に呼び出された場合は、何も起りません。

MatchedPairs オブジェクトのメソッド

MatchedPairs オブジェクトを使うと、対応のあるペアの分析を起動できます。また、プラットフォーム共通のメソッドも使用できます。

SignTest(Flag As Boolean)

符号検定をオン (True) またはオフ (False) にする。

SetAlphaLevel(Alpha As Double)

有意水準 (α) を設定する。

WilcoxonSignedRank(Flag As Boolean)

Wilcoxon の符号付順位検定をオン (True) またはオフ (False) にする。

測定システム分析 (MSA)

MeasurementSystemsAnalysis オブジェクトを使うと、測定システム分析を起動できます。変数を指定するための通常の起動メソッドに加えて、以下のようなメソッドがあります。

BiasStudySetAlpha(alpha as Double) As Boolean

バイアス分析レポートの有意水準 (α) を設定する。このメソッドを使用するには、起動プラットフォームを呼び出しておく必要があります。

BiasStudySetChartOptions(option as MSAStudyChartOptions) As Boolean

限界の陰影や垂線グラフなど、繰り返し誤差に関するグラフの表示オプションを設定する。

LaunchSpecifyAnalysisSettings(maxIterations as Integer, convergenceLimit as Double) As Boolean

これは、MSA 起動ダイアログボックスのオプションに対応する。

LaunchSpecifyAlpha(Alpha as Double) As Boolean

Launch メソッドで分析を起動する前に、有意水準 (α) を設定する。

LaunchSpecifyChartDispersionOptions(option as MSAChartDispersionTypes) As Boolean

ばらつき図の種類を、範囲にするか、標準偏差にするかを指定する。

LaunchSpecifyModelOptions(option as MSAModelTypes) As Boolean

Nested または Crossed のモデルオプションを指定する。[測定システム分析] メニューで選択されたときと同じく、オプションをオン (True) またはオフ (False) にします。

起動前または起動後に、出力を操作する方法として以下のメソッドを使用できます。

- ShowAverageChart
- ShowRangeChart

- ParallelismStudy
- EMPStudy
- MeasurementIncrementStudy
- VarianceComponents
- EMPGaugeStudy
- ShowBiasStudy
- ShowTestRetestErrorStudy(Flag as Boolean)

RangeChartOption(option as MSARangeChartOptions) As Boolean

「Show Average Range (範囲の平均を表示)」など、標準偏差図に関連するオプションをオン (True) またはオフ (False) にする。現在のばらつき図が範囲図でない場合は、ログにエラーを出力します。

StandardDeviationChartOption(option as MSAStandardDeviationChartOptions) As Boolean

「Show Average Range (範囲の平均を表示)」など、標準偏差図に関連するオプションをオン (True) またはオフ (False) 標準偏差図に関連するオプションをオン (True) またはオフ (False) にする。現在のばらつき図が範囲図でない場合は、ログにエラーを出力します。

TestRetestStudySetChartOptions(option as MSAStudyChartOptions) As Boolean

限界の陰影や垂線図など、繰り返し誤差に関するグラフの表示オプションを設定する。

出力を操作するメソッド

起動前または起動後に、出力を操作する方法として以下のメソッドを使用できます。

ShiftDetectionProfiler(Flag as Boolean)

変化検出プロファイルをオン (True) またはオフ (False) にする。

AverageChartOption(option as MSAStandardDeviationChartOptions) As Boolean

〔限界を表示〕 オプションなど、平均の管理図に関するオプションをオン (True) またはオフ (False) にする。

ShowStandardDeviationChart(Flag as Boolean)

標準偏差図をオン (True) またはオフ (False) にする。

MultipleCorrespondenceAnalysis

Launch() の前に指定するメソッド

次のメソッドは、Launch() メソッドの前に呼び出す必要があります。

LaunchAddResponse(name As String) As Boolean

目的変数の列を指定する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

Launch() As Boolean

多重対応分析レポートを起動する。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchAddFactor(name As String) As Boolean

説明変数の列を指定する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchAddSupplementaryVariable(name As String) As Boolean

追加変数の列を指定する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchAddSupplementaryID(name As String) As Boolean

追加 ID の列を指定する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchAddFreq(name As String) As Boolean

度数の列を指定する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchAddBy(name As String) As Boolean

By 変数の列を指定する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchRemoveResponse(name As String) As Boolean

目的変数のリストから列を削除する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchRemoveFactor(name As String) As Boolean

説明変数のリストから列を削除する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchRemoveSupplementaryVariable(name As String) As Boolean

追加変数のリストから列を削除する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchRemoveSupplementaryID(name As String) As Boolean

追加 ID のリストから列を削除する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchRemoveFreq(name As String) As Boolean

度数のリストから列を削除する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

LaunchRemoveBy(name As String) As Boolean

By 変数のリストから列を削除する。Name はデータ列の名前です。処理に成功したときは **True** を、失敗したときは **False** を戻します。

Launch() の後に指定するメソッド

次のメソッドは、**Launch()** メソッドの後に呼び出す必要があります。

CrossTable(flag As Boolean)

選択された変数の役割に応じて、Burt 表または分割表を作成する。Flag がオプションをオン (**True**) またはオフ (**False**) にします。

DisplayOptions(option As MCADisplayOptions, flag as Boolean)

多重対応分析レポートの各種表示オプションをオンまたはオフにする。Option はオンまたはオフにする表示オプションを指します。Flag がオプションをオン (**True**) またはオフ (**False**) にします。

SaveCoordinates(nDims As Short)

1つまたは複数の JMP データテーブルの主座標を保存する。nDims は保存する次元の数です。

SaveCoordinateFormula(nDims As Short)

多次元の主座標の計算式を含む列を元のデータテーブルに保存する。nDims は保存する次元の数です。

Multivariate オブジェクトのメソッド

Multivariate オブジェクトを使うと、「多変量の相関」プラットフォームを起動し、操作できます。

ColorMapOnCorrelations(Flag As Boolean)

相関に基づくカラーマップを表示 (**True**) または非表示 (**False**) にする。

ColorMapOnValues(Flag As Boolean)

p 値に基づくカラーマップを表示 (**True**) または非表示 (**False**) にする。

ClusterOnCorrelations(Flag As Boolean)

相関に基づくカラーマップで類似の相関を持つ変数をグループ化 (True) またはグループ解除 (False) する。

CorrelationProbability, C1ofCorrelation(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

CorrelationsM(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。これは、CorrelationsMultivariate (相関係数行列) オプションを意味します。

CovarianceMatrix(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

CronbachsAlpha(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

Ellipsoid3D(BSTR X, BSTR Y, BSTR Z) As Boolean

与えられた 3 つの列で、3 次元橢円を作成する。関数が False を戻した場合は失敗を意味します。指定された列名が無効だった場合などには、失敗します。True は成功を意味します。

HoeffdingsD(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

InverseCorr(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

KendallsTau(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

MultivariateSimpleStatistics(Flag as Boolean)

「多変量の基本統計量」レポートの表示をオン (True) またはオフ (False) にする。

OutlierAnalysis(Flag As Boolean) As Outlier

外れ値分析を生成し、Outlier オブジェクトへのディスパッチポインタを戻す。そのオブジェクトはさらに操作可能です (「Outlier オブジェクトのメソッド」(157 ページ))。

PairwiseCorr(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

ParallelCoordPlot(Flag as Boolean)

パラレルプロットの表示をオン (True) またはオフ (False) にする。

PartialCorr(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

PrincipalOnCorrelations As PrincipalComponents

主成分分析を行い、PrincipalComponents オブジェクトへのディスパッチポインタを戻す。そのオブジェクトはさらに操作可能です(「PrincipalComponentsオブジェクトのメソッド」(164ページ))。

PrincipalOnCovariances As PrincipalComponents

主成分分析を行い、PrincipalComponents オブジェクトへのディスパッチポインタを戻す。そのオブジェクトはさらに操作可能です。

PrincipalUncentered() As PrincipalComponents

主成分分析を行い、PrincipalComponents オブジェクトへのディスパッチポインタを戻す。そのオブジェクトはさらに操作可能です。

SaveTSquare()

T2 乗距離を現在のデータテーブルに保存する。

ScatterPlot(Flag As Boolean) As ScatterPlotMatrix

散布図行列を生成し、ScatterPlotMatrix オブジェクトへのディスパッチポインタを戻す。そのオブジェクトはさらに操作可能です(「ScatterplotMatrixオブジェクトのメソッド」(166ページ))。

SpearmansRho(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

StandardizedAlpha(Flag As Boolean)

対応する表示オプションを設定 (True) またはリセット (False) する。

TSquareDistances(Flag As Boolean)

T2 乗距離の表示をオン (True) またはオフ (False) にする。

UnivariateSimpleStatistics(Flag as Boolean)

「単変量の基本統計量」レポートの表示をオン (True) またはオフ (False) にする。

MultivControl オブジェクトのメソッド

MultivControl オブジェクトは、「多変量管理図」プラットフォームを起動し、操作します。このオブジェクトで使用できるメソッドは、「多変量管理図」のユーザーインターフェイスで使用できるものとまったく同じです。

PrincipalComponents(Flag as Boolean)

起動後において、主成分分析をオン (True) またはオフ (False) にする。

SavePrincipalComponents()

主成分得点を現在のデータテーブルの新しい列に保存する。

SaveTargetStatistics()

目標統計量のデータを現在のデータテーブルの新しい列に保存する。

SaveTSquare()

T2 乗のデータを現在のデータテーブルの新しい列に保存する。

ShowCorrelation(Flag as Boolean)

起動後において、相関行列を表示 (True) または非表示 (False) にする。

ShowCovariance(Flag as Boolean)

起動後において、共分散行列を表示 (True) または非表示 (False) にする。

ShowInverseCorrelation(Flag as Boolean)

起動後において、相関行列の逆行列を表示 (True) または非表示 (False) にする。

ShowInverseCovariance(Flag as Boolean)

起動後において、共分散行列の逆行列を表示 (True) または非表示 (False) にする。

ShowMeans(Flag as Boolean)

起動後において、平均を表示 (True) または非表示 (False) にする。

Neural オブジェクトのメソッド

Neural オブジェクトのメソッドを使うと、「ニューラル」プラットフォームを起動し、操作することができます。

ControlPanelOptions(neuralControlConstants option, Flag As Boolean)

ニューラルネットの設定パネルにある特定のチェックボックスを選択または選択解除する。最初のパラメータは、チェックボックスを指定する定数で、2 番目のパラメータは、オプションをオン (True) とオフ (False) のどちらにするかを表します。Go() メソッドが呼び出されたとき、

これらのオプションを使ってニューラルネットが計算されます。デフォルトでは、すべてのオプションがオフになっています。

Diagram(Flag As Boolean)

ニューラルのダイヤグラムをオン (True) またはオフ (False) にする。デフォルトではオフです。

Go()

ニューラルネットの計算を開始する。

Profiler(Flag As Boolean)

予測プロファイルをオン (True) またはオフ (False) にする。デフォルトではオフです。

SaveHidden()

指定のデータを現在のデータテーブルに保存する。[隠れノードと尺度化した列の保存] メニューオプションと同じ働きをします。

SaveFormulas()

指定のデータを現在のデータテーブルに保存する。[計算式の保存] メニューオプションと同じ働きをします。

SavePredicted()

指定のデータを現在のデータテーブルに保存する。[予測値の保存] メニューオプションと同じ働きをします。

SaveProfileFormulas()

指定のデータを現在のデータテーブルに保存する。[プロファイル式の保存] メニューオプションと同じ働きをします。

SpecifyConvergeCriterion(Value As Double) As Boolean

通常のユーザインターフェースにある設定パネルのオプションと同じ働きをする。Go() メソッドを呼び出すと、設定された値がニューラルネットの計算に使用されます。

SpecifyHiddenNodes(Value As Double) As Boolean

通常のユーザインターフェースにある設定パネルのオプションと同じ働きをする。Go() メソッドを呼び出すと、設定された値がニューラルネットの計算に使用されます。

SpecifyMaxIterations(Value As Double) As Boolean

通常のユーザインターフェースにある設定パネルのオプションと同じ働きをする。Go() メソッドを呼び出すと、設定された値がニューラルネットの計算に使用されます。

SpecifyNumberOfTours(Value As Double) As Boolean

通常のユーザインターフェースにある設定パネルのオプションと同じ働きをする。Go() メソッドを呼び出すと、設定された値がニューラルネットの計算に使用されます。

SpecifyOverfitPenalty(Value As Double) As Boolean

通常のユーザインターフェースにある設定パネルのオプションと同じ働きをする。Go() メソッドを呼び出すと、設定された値がニューラルネットの計算に使用されます。

正規混合のメソッド

「クラスター分析」プラットフォームの KMNormalMixtures メソッドに替えて、「正規混合」オートメーションプラットフォームを使用します。

Biplot、Biplot3D、ParallelCoordinatePlot、ScatterPlotMatrix(Flag as Boolean)

これらのオプションは、Go() メソッドを呼び出した後にオン (Flag が True) またはオフ (Flag が False) にする。

BiplotContourDensity(Density as Double)

バイプロットの密度等高線の値を 0 ~ 1 の間で指定する。

Go()

Specify メソッドで設定した値を使って指定された設定パネルの設定と、Specify の呼び出して変更されなかったデフォルト値を使って「正規混合」を実行する。

LaunchAddY、LaunchAddFreq、LaunchAddWeight、LaunchAddBy

これらの関数は、他のオートメーションプラットフォームの場合と同様に、Launch() を呼び出す前に変数にそれぞれの役割を割り当てる。

PublishClusterFormulas()

クラスター計算式を計算式デポに保存する。計算式は、そこから Python や C など、他のさまざまな言語にエクスポートできます。

SaveClusterFormula、SaveMixtureFormulas、SaveMixtureProbabilities、SaveDensityFormula()

元のデータテーブルに、適切な計算式の値が入った列を保存する。

SaveColorsToTable

クラスターの色を、データテーブルに行の属性の色として保存する。

SimulateClusters(NRows as Integer)

シミュレーションデータを作成する。作成されるオブザベーションの数を指定してください。

SpecifyNClusters(NClusters as Integer)、SpecifyNTours(NTours as Integer)、SpecifyMaximumIterations(MaxIterations as Integer)、SpecifyConvergeCriterion(Criterion as Double)

これらのメソッドは、Normal Mixtures オブジェクトで `Go()` を呼び出す前に使用します。メソッドは、「正規混合」プラットフォームを起動した直後に設定パネルに表示されるオプションを反映します。

Oneway オブジェクトのメソッド

`Oneway` オブジェクトのメソッドを使うと、「一元配置分析」（二変量の関係）プラットフォームを起動し、操作できます。

AnalysisOfMeans(Type as OnewayAofMConstants, Flag As Boolean)

指定された種類の平均分析を実行する。一部の平均分析には、その使用法において一定の制約があります。分析の実行に失敗したときは、ログを参照してください。

CDFPlot(Flag As Boolean)

累積確率プロットを表示 (`True`) または非表示 (`False`) にする。

CompareDensities(Flag As Boolean)

「密度の比較」プロットを表示 (`True`) または非表示 (`False`) にする。

CompareMeans(Option As Integer, Flag As Boolean)

平均の比較に関して、最初のパラメータで指定された検定を、オンまたはオフにする。第 1 パラメータに指定する定数は、`OnewayCompareConstants` に用意されています。Flag には、オン (`True`) またはオフ (`False`) を指定します。

CompositionOfDensities(Flag As Boolean)

「密度の構成」プロットを表示 (`True`) または非表示 (`False`) にする。

DisplayOptions(Option As Integer, Flag As Boolean)

分析のグラフィックに関するいろいろな表示オプションをオン (`Flag` が `True` の場合) またはオフ (`Flag` が `False` の場合) にする。Option に指定する定数は、`OnewayDisplayConstants` に用意されています。

EquivalenceTest(diffConsideredPracticallyZero as Double)

指定された差に基づく、同等性の検定を実行する。指定された差は、実質的にゼロとみなされます。

FitRobust, FitCauchy(Flag As Boolean)

ロバストまたは Cauchy のあてはめをオン (`True`) またはオフ (`False`) にする。

Histograms(Flag as Boolean)

各列のヒストグラムを一元配置グラフの横に表示する。

Kolmogorov Smirnov

ノンパラメトリックな検定の一つである Kolmogorov-Smirnov 検定を実行する。

MatchingColumn(Column Name As String) As Boolean

与えられた変数（列）で、対応のあるモデルの分析を実行する。成功のときは `True` を戻し、列が存在しない場合や他にエラーがあった場合は `False` を戻します。

MeansAnovaT(Flag As Boolean)

追加の分析結果を表示する（`True`）か、しない（`False`）かを指定する。

MeansStdDev(Flag As Boolean)

追加の分析結果を表示する（`True`）か、しない（`False`）かを指定する。

NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)

名義尺度軸の表示に関するブール値オプションを設定する。プラットフォームで使用可能なオプションをオン（`Flag` が `True`）またはオフ（`Flag` が `False`）にします。たとえば、Rotated Tick Labels（目盛りラベルの回転）、Divided Lines（分割線）、Lower Frame（下のフレームの表示）などを設定できます。目盛りラベルの回転は、「一元配置」プラットフォームと「変動性図」プラットフォームでのみ使用できます。このメソッドを呼び出す前に、`GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long` を呼び出して、名義尺度軸ディスプレイボックスへのハンドルを取得する必要があります。

Nonparametric(Option As Integer, Flag As Boolean)

ノンパラメトリック検定の分析結果を表示したり、非表示にしたりする。検定の種類は第 1 引数に指定します。第 1 引数に指定する定数は、`OnewayNonParConstants` に用意されています。第 2 引数には、分析結果を表示する（`True`）か、非表示にする（`False`）かを示します。

NonParametricMultipleComparisons(Type as OnewayNonParMultipleComparisonConstants, Flag As Boolean)

`Flag` が `True` の場合は `Type` に指定された比較を実行し、そうでない場合は比較を実行しない。Dunnett 検定などのコントロール群との比較を行う検定では、コントロール群を指定するためのダイアログボックスが表示されます。Dunnett 検定などでこのダイアログボックスを表示させたくない場合には、`NonParametricMultipleWithControl` メソッドのほうを使用してください。

NonParametricMultipleWithControl(Type as OnewayNonParMultipleComparisonConstants, ControlValue as String) As Boolean

「Steel with Control（コントロール群との比較 Steel 検定）」や「Dunn with Control for Joint Ranks（コントロール群との比較 併合順位の Dunn 検定）」など、コントロール群の指定が必要な比較を実行する。JMP 上においては、コントロール群のデータ値は、X 列のタイプに応じて数値や文字列になっています。しかし、このメソッドでは、"Female" や "15" というように、データ値が数値であっても、文字列で指定する必要があります。

NormalQuantileLineOfFit(Flag As Boolean)

[正規分位点プロット] サブメニューにある [あてはめ線] オプションをオンまたはオフにする。

NormalQuantilePlot(Flag As Boolean)

[正規分位点プロット] サブメニューにある [分位点 - 実測値プロット] オプションをオンまたはオフにする。

NormalQuantilePlotQbyA(Flag As Boolean)

[正規分位点プロット] サブメニューにある [実測値 - 分位点プロット] オプションをオンまたはオフにする。

NormalQuantileProbLabels(Flag As Boolean)

[正規分位点プロット] サブメニューにある [確率ラベル] オプションをオンまたはオフにできるようにする。

ProportionOfDensities(Flag as Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

Save(Option As Integer)

データテーブル内に新しい列を作成し、標準化、中心化、または正規化した分位点を保存する。Option (オプション) パラメータで保存のタイプを指定します。パラメータには `OnewaySaveConstants` の定義に含まれている値を使用します。詳細については、『基本的な統計分析』を参照してください。

SetAlpha(Level As Double)

有意水準を指定する (例 : 0.95)

TTest(Flag as Boolean)

2 群を比較する t 検定をオンまたはオフにする。

UnequalVariances(Flag As Boolean)

追加の分析結果を表示する (True) か、しない (False) かを指定する。

Outlier オブジェクトのメソッド

Outlier オブジェクトは、Multivariate オブジェクトの `OutlierAnalysis(Flag As Boolean) As Outlier` メソッドから作成されます。

JackknifeDistances(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

MahalanobisDistances(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

SaveJackknife()

データテーブルの新しい列に距離を保存する。

SaveMahal()

データテーブルの新しい列に距離を保存する。

Overlay オブジェクトのメソッド

Overlay オブジェクトを使うと、重ね合わせプロットを起動し、操作できます。Overlay プラットフォームは廃止されました。Overlay プラットフォームを呼び出すオートメーションのコードは、Graph Builder (グラフビルダー) を使用してグラフを作成します。このため、以前のバージョンとグラフの外観が異なる場合があります。

LaunchAddYWithRightScale(Column Name As String)

右軸をオンにして、Y 変数を追加する。左軸をオンにする場合は、LaunchAddY を使います。

LaunchSetSortScaleOptions(XSort as Boolean, XLogAxis as Boolean, YLogLeftAxis As Boolean, YLogRightAxis As Boolean)

X 軸で並べ替え、X 軸を対数にする、および左右の Y 軸を対数にする、の各設定をオン (True) またはオフ (False) にする。これは、「重ね合わせプロット」起動ダイアログボックスのオプションに対応します。

LineOptions(overlayLineStyleConstants style, overlayLineThicknessConstants thickness)

ConnectPoints オプションによって折れ線を描いた場合に、折れ線の種類と太さを指定する。

Overlay(Flag As Boolean)

グラフを重ね合わせて表示する (True) か、個々のグラフを表示する (False) かを指定する。

Range(Flag As Boolean)

範囲のプロットを表示する (True) か、しない (False) かを指定する。

SeparateAxes

重ね合わせプロットの対応する表示オプションをオン (True) またはオフ (False) にする。

YConnectColor(Color As jmpColorConstants)

折れ線の色を設定する。

YConnectPoints(Flag As Boolean)

重ね合わせプロットの対応する表示オプションをオン (True) またはオフ (False) にする。

YOverlayMarker(Marker as jmpMarkerConstants)

重ね合わせプロットの点のマーカーのタイプを設定する。

YOverlayMarkerColor(Color as jmpColorConstants)

重ね合わせプロットの点のマーカーの色を設定する。

YNeedle(Flag As Boolean)

重ね合わせプロットの対応する表示オプションをオン (True) またはオフ (False) にする。

YShowPoints(Flag As Boolean)

重ね合わせプロットの対応する表示オプションをオン (True) またはオフ (False) にする。

YStep(Flag As Boolean)

重ね合わせプロットの対応する表示オプションをオン (True) またはオフ (False) にする。

ParallelPlot オブジェクトのメソッド

ParallelPlot オブジェクトを使うと、パラレルプロットを起動し、操作することができます。

ReverseScaleOnY(Column Name as String) As Boolean

起動前に指定されていた Y 列のスケールを逆にする。処理に成功したときは True を、列が見つからなかったときは False を戻します。

ShowReverseCheckboxes(Flag as Boolean)

Y 値のスケールを逆にするためのチェックボックスを表示する。

Pareto オブジェクトのメソッド

Pareto オブジェクトを使うと、パレート図を作成し、操作することができます。

AddCauseToCombine(causeName As String)

リストに、組み合わせる原因の名前を追加する。このメソッドによって原因をリストに追加してから、CombineCauses() As Boolean を呼び出すと、リストに追加されている原因が組み合わせられる。

CategoryLegend(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

CombineCauses() As Boolean

AddCauseToCombine(causeName As String) で追加されたすべての原因を組み合わせる。処理に成功したときは True を、失敗したときは False を戻します。

CumPercentAxis(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

CumPercentCurve(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

CumPercentPoints(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

HorizontalLayout(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

Nlegend(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

PercentScale(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

PieChart(Flag As Boolean)

[パレート図] メニューオプションに対応する表示オプション。オン (True) またはオフ (False) にする。

SeparateCauses()

現在組み合わされているすべての原因を分離する。

UngroupPlots(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

Partition オブジェクトのメソッド

Partition オブジェクトを使うと、「パーティション」プラットフォームを起動し、操作できます。このオブジェクトで用意されているメソッドは、ユーザーインターフェイスで使えるものとまったく同じです。

ColorPoints(Flag as Boolean)

プロット点の色分けのオプションをオン (True) またはオフ (False) にする。このメソッドは、Y 変数が名義尺度か順序尺度の場合にのみ機能します。

ColumnContributions(Flag as Boolean)

列の寄与を表示する (True)、または非表示にする (False)。

Criterion(Option as partitionCriterionConstants, Flag as Boolean)

分岐の基準を定数を使って選択する。

DisplayOptions(Option as partitionDisplayConstant, Flag as Boolean)

表示オプションを定数から選択し、それをオン (True) またはオフ (False) に設定する。

KFoldCrossValidation(value as Integer)

K 分割交差検証の K 値を整数で指定する。

LeafReport(Flag as Boolean)

葉のレポートを表示する (True)、または非表示にする (False)。

LiftCurve(Flag as Boolean)

リフトチャートを表示する (True)、または非表示にする (False)。このメソッドは、Y 変数が名義尺度か順序尺度の場合にのみ機能します。

LockColumns(Flag as Boolean)

列をロックするオプションをオン (True) またはオフ (False) にする。

MinimizeSizeSplit(value as double)

分岐の最小サイズを倍精度で指定する。

MissingValueRule(Option as partitionMissingConstants, Flag as Boolean)

欠測値のルールを定数を使って指定する。

PlotActualByPredicted(Flag as Boolean)

「予測値と実測値のプロット」を表示する (True)、または非表示にする (False)。

Prune()

パーティションで最悪分岐を剪定するコマンドを実行する。

ROCCurve(Flag as Boolean)

ROC 曲線を表示するオプションを表示する (True)、または非表示にする (False)。このメソッドは、Y 変数が名義尺度か順序尺度の場合にのみ機能します。

SaveColumns(SaveOperation as partitionSaveColumnConstants)

統計量の列を、現在のデータテーブルに保存する。保存される統計量は、定数から選択します。

SmallTreeView(Flag as Boolean)

小さいツリーを表示する (True)、または非表示にする (False)。

Split()

パーティションで最良分岐コマンドを実行する。

SplitHistory(Flag as Boolean)

分岐履歴を表示する (True)、または非表示にする (False)。

PLS オブジェクトメソッド

PLS オブジェクトを使うと、PLS 回帰プラットフォームを起動し、操作することができます。このオートメーションプラットフォームの土台となった PLS プラットフォームは JMP から削除され、代わりに、豊富な機能を備えた新しい PLS 回帰プラットフォームが用意されています。オートメーション機能により、既存のオートメーション API のマッピングを通じてできるだけ新しい JMP プラットフォームが使用されます。以前の PLS オートメーションを使用している場合は、出力が適切かどうかをよく確認してください。

162 ページの「Launch() の前に指定するメソッド」と 163 ページの「Launch() の後に指定するメソッド」は、PLS 回帰プラットフォームのメソッドを記述します。163 ページの「以前の PLS オブジェクトのメソッド」は旧 PLS オブジェクトのメソッドです。

Launch() の前に指定するメソッド

次のメソッドは、Launch() メソッドの前に呼び出す必要があります。

LaunchSpecifyModelMethod(Method As plsModelMethodConstants)

定義済みの定数の 1 つを使い、手法を指定する。現在、NIPALS と SIMPLS がサポートされています。

LaunchSpecifyValidationType(valType As plsValidationTypes, valParm As Double)

検証法 (K 分割や保留など) を指定する。2 番目のパラメータに、K 分割の場合、分割数を指定します。保留の場合、2 番目のパラメータには検証データ抽出確率を指定します。詳細については、『多変量分析』の「PLS 回帰」章を参照してください。

LaunchSpecifyInitialNumberOfFactors(nFactors As Int)

PLS 回帰のウィンドウと同様、あてはめを開始する前に初期因子数を指定する。LaunchAddX を使って説明変数を指定した後、初期因子数を指定します。

LaunchSetRandomSeed(Seed As Double)

オプションの乱数シード値を設定する。デフォルトの手法はシード値を使用しません。

LaunchAddValidationColumn(Name As String)

検証列を 1 つ指定する。

LaunchRemoveValidationColumn(Name As String)

Launch を呼び出す前に検証列を削除する。

LaunchSpecifyOptions(Centering As Boolean, Scaling As Boolean)

中心化または尺度化のオプションをオン (True) またはオフ (False) にする。デフォルトではオプションがオンになっています。

LaunchSpecifyImputeMethod(Method As plsImputMethods, Iterations As Int)

定義済みの定数を使って欠測データの補完法を指定する。EM 法の場合は、2 番目のパラメータで最大反復回数を指定します。2 番目のパラメータは必須ではあるものの、平均法では無視されます。

Launch() の後に指定するメソッド

次のメソッドは、Launch() メソッドの後に呼び出すことができます。

PercentVariationPlots、 LoadingScatterPlotMatrices、 Profiler、 VIPVersusCoefficientPlots、 CoefficientPlots、 ScoreScatterplotMatrices、 SpectralProfiler(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

CorrelationLoadingPlot(Int nFactors)

指定した因子数で、負荷量の相関図を表示する。

以前の PLS オブジェクトのメソッド

ConfidenceLines(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

SaveFormula()

予測式を現在のデータテーブルに保存する。

SaveOutputs(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

ShowPoints(Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

PrincipalComponents オブジェクトのメソッド

PrincipalComponentsオブジェクトは、MultivariateオブジェクトのPrincipalOnCorrelations As PrincipalComponents、PrincipalOnCovariances As PrincipalComponents、PrincipalUncentered() As PrincipalComponents メソッドから作成されます。

FactorRotation(N As Integer)

N 個の因子で、因子の回転を実行する。

SavePrincipal(Num As Integer)

指定の数 (Num) の成分を、データテーブルの列として保存する。

SaveRotated()

回転後の因子をデータテーブルの新しい列に保存する。

Spin(Flag As Boolean)

フラグ (Flag) が True ならば、最初の 3 主成分の回転プロットを作成する。

Profiler オブジェクトのメソッド

Profiler オブジェクトを使うと、予測プロファイルを起動し、操作することができます。

ConfidenceIntervals(Flag As Boolean)

信頼区間表示オプションをオン (True) またはオフ (False) にする。

Desirability(Flag As Boolean)

満足度関数表示オプションをオン (True) またはオフ (False) にする。

InteractionProfiler(Flag as Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

LaunchAddNoiseFactors(NoiseFactorsColumn As String) As Boolean

ロバスト性を調査するために、誤差因子の列を追加する。

MostDesirable()

満足度の最大化オプションを実行する。

Recurrence オブジェクトのメソッド

Recurrence オブジェクトを使うと、再生モデルによる分析を起動し、操作することができます。

EventPlot(Flag As Boolean)

イベントプロットを表示する (True) か、しない (False) かを指定する。

MCFConfidLimits(Flag As Boolean)

MCF の信頼限界を表示する (True) か、しない (False) かを指定する。

MCFPlot(Flag As Boolean)

MCF プロットを表示する (True) か、しない (False) かを指定する。

PlotMCFDifferences(Flag as Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

Scatterplot3D オブジェクトのメソッド

三次元散布図のオートメーションは、メニューから実行できる機能のほとんどをサポートしています。

BiplotRays(Flag As Boolean)

バイプロット線を表示 (True) または非表示 (False) にする。バイプロット線は、通常バイプロット線を作成するオプションが実行されている場合にのみ表示されます。

ConnectPoints(BSTR groupingColumn)

プロットの点をつなげる。グループ列を指定しない場合は、空の文字列 ("") を指定する必要があります。

DropLines(Flag As Boolean)

垂線を表示 (True) または非表示 (False) にする。

NormalContourEllipsoids(BSTR groupingColumn)

三変量正規分布の確率楕円体を表示する。グループ列を指定しない場合は、空の文字列 ("") を指定する必要があります。

PrincipalComponents()

主成分をオンにする。

RotatedComponents()

因子分析と回転の方法を指定するさまざまなパラメータのあるダイアログボックスを開く。

SavePrincipalComponents(Number as Long)

入力パラメータで指定された数だけの主成分を現在のデータテーブルに保存する。

SaveRotatedComponents()

RotatedComponents() がすでに実行されている場合、回転後の成分を現在のデータテーブルに保存する。

ShowPoints(Flag As Boolean)

点を表示 (True) または非表示 (False) にする。

StdPrincipalComponents()

標準化した主成分をオンにする。

ScatterplotMatrix オブジェクトのメソッド

ScatterPlotMatrix オブジェクトは、Multivariate オブジェクトの ScatterPlot(Flag As Boolean) As ScatterPlotMatrix メソッドによって作成されます。

DensityEllipses(Flag As Boolean)

対応する表示オプションをオン (True) またはオフ (False) にする。

EllipseAlpha(Alpha As Double)

正規分布に従っているときに楕円の中に入る点のパーセンテージを指定する。

EllipseColor(Color As Integer)

楕円の色を、定数 jmpColorConstants から指定する。

Histograms(HorizontalHistogram As Boolean, Flag As Boolean)

ヒストグラムを散布図行列に表示する。最初のパラメータが True の場合は X 軸上に、False の場合は Y 軸上にヒストグラムが描かれます。Flag でヒストグラムをオン (True) またはオフ (False) にします。

ScatterplotMatrixPlatform オブジェクトのメソッド

これらのメソッドは「散布図行列」プラットフォームをサポートするもので、Multivariate オブジェクトの ScatterPlot(Flag As Boolean) As ScatterPlotMatrix メソッドをサポートするものではありません。

DensityEllipses(Flag As Boolean)

確率楕円をオン (True) またはオフ (False) にする。

EllipseAlpha(Alpha As Double)

楕円の確率の値を 0.0 ~ 1.0 の間で指定する。

EllipseTransparency(Transparency As Double)

楕円の透明度を設定する。値の範囲は 0 ~ 1 です。

LaunchSpecifyMatrixFormat(scatterplotMatrixFormatconstants val)

起動メソッドをコールする前に、散布図行列の形式 (下三角、正方形など) を指定する。

ShowCorrelations、ShowPoints、FitLine、NonParDensity (Flag As Boolean)

対応するオプションをオン (True) またはオフ (False) にする。

Screening オブジェクト

Screening オブジェクトを使うと、「スクリーニング」プラットフォームを起動し、操作できます。

メソッド

Screening オブジェクトに固有のメソッドはありません。

SpinPlot オブジェクトのメソッド

SpinPlot オブジェクトでは、回転プロットを起動し、操作することができます。Launch メソッドによって回転プロットを起動した後に、プロットを回転するためには Spin メソッドを使ってください。

メモ: SpinPlot は JMP 8 から削除されました。SpinPlot 用に書かれたコードは引き続き使用できます。ただし、回転プロットではなく「三次元散布図」プラットフォームが起動します。SpinPlot オートメーションインターフェースの既存のメソッドは、すべて引き続き機能します。

BiplotRays(Flag As Boolean)

バイプロット線を表示する (True) か、しない (False) かを指定する。

PrincipalComponents()

使用した変数から主成分を計算する。オプションの働きは、インタラクティブに操作したときのオプションと同じです。

RotatedComponents(Number As Integer)

指定の数 (number) の回転成分のスコアを計算する。

SavePrincipalComponents()

データテーブル内に新しい列を作成し、現在の主成分を保存する。SaveToPrincipalComponents は、保存する主成分の数を尋ねるプロンプトを表示します。SavePrincipalComponents2(NumberToSave as Short) を使用した場合は、保存する主成分の数が指定できるため、プロンプトは表示されません。

SavePrincipalComponents2(NumberToSave as Short)

保存する主成分の数を指定できる。そのため、プロンプトは表示されません。SavePrincipalComponents() の場合はプロンプトを表示します。

SaveRotatedComponents()

データテーブル内に新しい列を作成し、現在の回転成分を保存する。

Spin(pitch As Integer, yaw As Integer, roll As Integer, numTimes As Integer)

指定されたピッチ (pitch)、ヨー (yaw)、ロール (roll) の各値でプロットを回転させる。回転の回数は最後のパラメータ (numTimes) で決まります。

SpinPitch(Angle As Integer)

指定の方向にグラフを回転させる。角度は、 $-45^\circ \sim 45^\circ$ の範囲で指定しなければなりません。

SpinRoll(Angle As Integer)

指定の方向にグラフを回転させる。角度は、 $-45^\circ \sim 45^\circ$ の範囲で指定しなければなりません。

SpinYaw(Angle As Integer)

指定の方向にグラフを回転させる。角度は、 $-45^\circ \sim 45^\circ$ の範囲で指定しなければなりません。

StdPrincipalComponents()

使用した変数から標準化した主成分を計算する。オプションの働きは、インタラクティブに操作したときのオプションと同じです。

SurfacePlot オブジェクトのメソッド

SurfacePlot オブジェクトを使うと、曲面プロットを操作することができます。

DisplayOptions(option as surfaceDisplayOptions, flag as Boolean)

曲面プロットの表示プロパティに関する 15 以上のオプションをオン (True) またはオフ (False) にする。たとえば、データ点や X 軸のグリッドなどを指定できます。

SetItemColor(item as surfaceColorConstants, color as JMPColorConstants)

さまざまな曲面プロット表示要素の色を変更する。たとえば、Mesh (メッシュ) や Contour (等高線) の色を指定できます。

Survival オブジェクトのメソッド

Survival オブジェクトを使うと、「生存時間／信頼性分析」プラットフォームを起動し、操作することができます。

CompetingCauseAction(competitingCauseConstants action, Flag as Boolean)

競合する原因に関するオプションのどれかをオンにする。オプションは、フラグ (Flag) が True ならばオン、False ならばオフです。

CompetingCauses(columnName As String) As Boolean

競合故障分析を実行する。引数には、故障原因のラベルを持つデータテーブルの列の名前を指定してください。成功のときは True を、失敗のときは False を戻します。

ExponentialEst(Flag As Boolean)

グラフ作成オプションをインタラクティブに操作したときと同じ働きをする。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

ExponentialPlot(Flag As Boolean)

グラフ作成オプションをインタラクティブに操作したときと同じ働きをする。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

LognormalEst(Flag As Boolean)

グラフ作成オプションをインタラクティブに操作したときと同じ働きをする。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

LognormalPlot(Flag As Boolean)

グラフ作成オプションをインタラクティブに操作したときと同じ働きをする。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

MidStepQuantilePoints(Flag As Boolean)

対応するオプションをオン (`True`) またはオフ (`False`) にする。

ReverseYAxis(Flag As Boolean)

この表示オプションは、[プロット] サブメニューをインタラクティブに操作したときと同じ働きをする。フラグ (Flag) が `True` ならばオプションはオン、`False` ならばオフです。

SaveEstimates() As DataTable

新しいデータテーブルを作成し、推定値を保存する。新しいデータテーブルへのディスパッチポインタを戻し、そのデータテーブルの操作を可能にします。

ShowCombined(Flag As Boolean)

この表示オプションは、[プロット] サブメニューをインタラクティブに操作したときと同じ働きをする。フラグ (Flag) が `True` ならばオプションはオン、`False` ならばオフです。

ShowConfidInterval(Flag As Boolean)

この表示オプションは、[プロット] サブメニューをインタラクティブに操作したときと同じ働きをする。フラグ (Flag) が `True` ならばオプションはオン、`False` ならばオフです。

ShowPoints(Flag As Boolean)

この表示オプションは、[プロット] サブメニューをインタラクティブに操作したときと同じ働きをする。フラグ (Flag) が `True` ならばオプションはオン、`False` ならばオフです。

SurvivalPlot(Flag As Boolean)

生存時間プロットのオン／オフを切り替える。

ShowSimultaneousCI(Flag as Boolean)

信頼区間のオプションをオン (True) またはオフ (False) にする。

WeibullEst(Flag As Boolean)

グラフ作成オプションをインタラクティブに操作したときと同じ働きをする。オプションは、パラメータが True ならばオン、False ならばオフです。

Weibull-Plot(Flag As Boolean)

グラフ作成オプションをインタラクティブに操作したときと同じ働きをする。オプションは、パラメータが True ならばオン、False ならばオフです。

Ternary オブジェクトのメソッド

Ternary オブジェクトを使うと、三角図を起動できます。また、プラットフォーム共通のメソッドも使用できます。

LaunchAddFormulaCol.ColumnName As String) As Boolean

等高線計算式の列を追加する。

LaunchRemoveFormulaCol.ColumnName As String) As Boolean

等高線計算式の列を削除する。

Text Explorer

Launch() の前に指定するメソッド

次のメソッドは、Launch() メソッドの前に呼び出す必要があります。

LaunchAddTextColumn(name as String) As Boolean

テキストデータを含む列を割り当てます。Name はデータテーブルにおける列名です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchAddID(name as String) As Boolean

[積み重ねた文書単語行列の保存] オプションによって保存されるデータテーブルにおいて ID となる列を指定します。Name はデータテーブルにおける列名です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchAddBy(name As String) As Boolean

指定した列の水準ごとにレポートが作成されます。Name はデータテーブルにおける列名です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchMaxWordsPerPhrase(n As Short) As Boolean

句リストの句を構成するときの、1句あたりに含める単語数の最大値を指定します。N は最大数です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchRemoveTextColumn(name As String) As Boolean

テキスト列フィールドに割り当てられている列を削除する。Name はデータテーブルにおける列名です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchRemoveID(name As String) As Boolean

ID フィールドに割り当てられている列を削除する。Name はデータテーブルにおける列名です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchRemoveBy(name As String) As Boolean

By フィールドに割り当てられている列を削除する。Name はデータテーブルにおける列名です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchMaxNumberOfPhrases(n As Long) As Boolean

句リストに表示される句の最大数を指定する。N は最大数です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchMinCharactersPerWord(n As Short) As Boolean

1 単語あたりに含める文字数の最小値を指定します。N は最小数です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchMaxCharactersPerWord(n As Short) As Boolean

分析に含める単語の最大文字数を指定する。N は最大数です。処理に成功したときは True を、失敗したときは False を戻します。

LaunchLanguage(option As textExplorerLanguageOptions) As Boolean -

テキスト処理に使用する言語を指定する。このオプションは、JMP がどの表示言語で動作しているかとは無関係です。Option には事前に定義された定数を使って言語を指定します。処理に成功したときは True を、失敗したときは False を戻します。

LaunchStemming(option As textExplorerStemmingOptions) As Boolean

先頭が同じで末尾が異なる語を同じとみなすかどうかを指定します。Option には事前に定義された定数を使って指定をします。処理に成功したときは True を、失敗したときは False を戻します。

LaunchTokenizing(option As textExplorerTokenizingOptions) As Boolean

句を単語（トークン）に分解する方法を指定します。Option には事前に定義された定数を使ってトークン化の方法を指定します。処理に成功したときは True を、失敗したときは False を戻します。

LaunchTreatNumbersAsWords(flag As Boolean) As Boolean

すべて数字で構成される語をトークンとして扱う。このメソッドは、[基本ワード] によるトークン化の場合のみ機能します。Flag の指定によりこの機能を有効または無効にします。処理に成功したときは True を、失敗したときは False を戻します。

Launch() As Boolean

[テキストエクスプローラ] プラットフォームを起動する。処理に成功したときは True を、失敗したときは False を戻します。

Launch() の後に指定するメソッド

次のメソッドは、Launch() メソッドの後に呼び出すことができます。

DisplayOptions(option As textExplorerDisplayOptions, flag as Boolean)

テキストエクスプローラのレポートに関連する各種表示オプションを有効または無効にする。Option には、textExplorerDisplayOption の定数の 1 つを指定します。Flag には、そのオプションを有効にするか無効にするかを指定します。

LatentClassAnalysis(maxNumTerms As Short, minTermFrequency As Short, numClusters As Short)

疎行列ルーチンを使い、2 値の重み付き文書単語行列に基づく潜在クラス分析を実行する。JMP Pro でのみ機能します。

LatentSemanticAnalysis(maxNumTerms As Short, minTermFrequency As Short, weighting As textExplorerSemanticWeightingOptions, numSingularVectors As Short, centeringAndScaling As textExplorerSemanticCenteringOptions)

単語文書行列の疎特異値分解を実行する。weighting は textExplorerSemanticWeightingOptions の定数によって定義されます。centeringAndScaling は、textExplorerSemanticCenteringOptions の定数によって定義されます。JMP Pro でのみ機能します。

TopicAnalysis(numTopics As Short)

文書単語行列の Varimax 回転後の特異値分解を実行し、トピックと呼ばれる単語のグループを作成する。numTopics には、トピックの個数を指定します。このメソッドを使用するには、先に Latent Semantic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

ClusterTerms(flag As Boolean)

単語に対する階層型クラスター分析の表示／非表示を切り替えます。Flag の指定によりこの機能を有効または無効にします。このメソッドを使用するには、先に Latent Semantic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

ClusterDocuments(flag As Boolean)

文書に対する階層型クラスター分析の表示／非表示を切り替えます。Flag の指定によりこの機能を有効または無効にします。このメソッドを使用するには、先に Latent Semantic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

SVDScatterplotMatrix(numVectors As Short)

単語と文書の特異値分解ベクトルの散布図行列の表示／非表示を切り替える。numVectors にはベクトルの数を指定します。このメソッドを使用するには、先に Latent Semantic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

TopicScatterplotMatrix(flag As Boolean)

回転した後の負荷量ベクトルの散布図行列の表示 / 非表示を切り替えます。Flag の指定によりこの機能を有効または無効にします。このメソッドを使用するには、先に Latent Semantic Analysis と Topic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

SaveDocumentTermMatrix(maxNumTerms As Short, minTermFrequency As Short, weighting As textExplorereSemanticWeightingOptions)

文書単語行列の各列を、データテーブルに保存する。WeightingにはtextExplorereSemantic WeightingOptionsオプションの1つを指定します。

SaveDocumentSingularVectors(numVectors as Short)

ユーザが指定した数の主成分スコア（左特異ベクトルに特異値をかけたもの）を、データテーブルの列に保存します。このメソッドを使用するには、先に Latent Semantic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

SaveDocumentTopicVectors()

ユーザが指定した数の、主成分スコアを回転したものを、データテーブルの列に保存します。このメソッドを使用するには、先に Latent Semantic Analysis と Topic Analysis を実行しておかなければなりません。JMP Pro でのみ機能します。

SaveStackedDTMForAssociation() As JMP.DataTable

文書単語行列を積み重ねたものを JMP データテーブルに保存する。作成されたデータテーブルへの参照を戻します。JMP Pro でのみ機能します。

SaveDTMFormula()

ベクトルを値とする計算式列をデータテーブルに保存する。

SaveSingularVectorFormula()

文書特異値分解を含んだ、ベクトルを値とする計算式列をデータテーブルに保存する。JMP Pro でのみ機能します。

SaveTopicVectorFormula()

回転後の特異値分解を含んだ、ベクトルを値とする計算式列をデータテーブルに保存する。このメソッドを使用するには、先に `Latent Semantic Analysis` を実行しておかなければなりません。JMP Pro でのみ機能します。

SaveTermTable() As JMP.DataTable

単語ごとに、その単語の登場頻度、その単語が含まれる文書の数を含んだデータテーブルを作成する。作成されたデータテーブルへの参照を戻します。

SaveTermSingularVectors(numVectors as Short)

[単語テーブルの保存] で保存されたデータテーブルに、単語特異値分解で作成された特異ベクトルのうちユーザが指定した数の特異ベクトルを列として保存する。このメソッドを使用するには、先に `Save Term Table` を実行しておかなければなりません。JMP Pro でのみ機能します。

SaveTermTopicVectors()

[単語テーブルの保存] コマンドで作成されたデータテーブルに、トピックベクトルを列として保存する。このメソッドを使用するには、先に `Save Term Table` を実行しておかなければなりません。JMP Pro でのみ機能します。

ScoreTermsByColumn(columnName As String)

単語ごとにスコアを算出し、[用語テーブルの保存] コマンドで作成された JMP テーブルに保存する。スコアは、指定した列の数値データから算出されます。`columnName` は、指定した列の名前。このメソッドを使用するには、先に `Save Term Table` を実行しておかなければなりません。

TimeSeries オブジェクトのメソッド

TimeSeries オブジェクトを使うと、時系列分析を起動し、操作することができます。

ARCoefficients(Flag As Boolean)

この分析オプションは、対応するオプションをインタラクティブに操作したときと同じ働きをする。パラメータが `True` ならばオプションはオン、`False` ならばオフです。

Arima(p As Double, d As Double, q As Double, confidenceInterval As Double, intercept As Boolean, constrainFit As Boolean)

ARIMA モデルを実行する。スタンダードアロンで実行している場合、パラメータの値は「ARIMA」ダイアログボックスの値を反映しています。

Autocorrelation(Flag As Boolean)

この分析オプションは、対応するオプションをインタラクティブに操作したときと同じ働きをする。パラメータが `True` ならばオプションはオン、`False` ならばオフです。

ConnectingLines(Flag As Boolean)

時系列プロットの表示オプション。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

MeanLine(Flag As Boolean)

時系列プロットの表示オプション。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

PartialAutocorr(Flag As Boolean)

この分析オプションは、[偏自己相関] メニューの項目と同じ働きをする。パラメータが `True` ならばオプションはオン、`False` ならばオフです。

SaveSpectralDensity() As DataTable

新しいデータテーブルを作成してスペクトル密度を保存し、そのデータテーブルへのディスパッチポインタを戻す。そのデータテーブルは、さらに操作できるようになります。

ShowPoints(Flag As Boolean)

時系列プロットの表示オプション。オプションは、パラメータが `True` ならばオン、`False` ならばオフです。

SmoothingModel(Model As Integer, Constraints As Integer)

平滑化モデルと制約を設定する。設定するときは、`timeSeriesModelConstants` と `timeSeriesConstraintConstants` の値を使います。

SpectralDensity(Flag As Boolean)

この分析オプションは、対応するオプションをインタラクティブに操作したときと同じ働きをする。パラメータが `True` ならばオプションはオン、`False` ならばオフです。

TimeSeriesGraph(Flag As Boolean)

時系列プロットの表示オプション。パラメータが `True` ならばオプションはオン、`False` ならば時系列プロット全体を表示しません。

Variogram(Flag As Boolean)

この分析オプションは、対応するオプションをインタラクティブに操作したときと同じ働きをする。パラメータが `True` ならばオプションはオン、`False` ならばオフです。

Variability オブジェクトのメソッド

`Variability` オブジェクトを使うと、変動性図を起動し、操作することができます。

AIAGLabels(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。`True` ならば表示、`False` ならば非表示。

BiasReport(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

ConnectCellMeans(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

DiscriminationRatio(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

GageRandR(K As Double, Tolerance As Double)

インタラクティブに操作したときと同様に、Gage R&R 分析を実行する。

LinearityStudy(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

NomAxisBooleanOption(Handle as Long, Action as Short, Flag As Boolean)

名義尺度軸の表示に関するブール値オプションを設定する。プラットフォームで使用可能なオプションをオン（フラグが True の場合）またはオフ（False）にします。たとえば、Rotated Tick Labels（目盛りラベルの回転）、Divided Lines（分割線）、Lower Frame（下のフレームの表示）などを設定できます。目盛りラベルの回転は、「一元配置」プラットフォームと「変動性図」プラットフォームでのみ使用できます。このメソッドを呼び出す前に、GetGraphicItemByType(TypeName As String, ItemNumber As Integer) As Long を呼び出して、名義尺度軸ディスプレイボックスへのハンドルを取得する必要があります。

PointsJittered(Flag As Boolean)

対応するオプションをオン（True）またはオフ（False）にする。

ShowBoxPlots(Flag as Boolean)

対応するオプションをオン（True）またはオフ（False）にする。

ShowCellMeans(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

ShowGrandMean(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

ShowGroupMeans(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

ShowPoints(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

ShowRangeBars(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならば非表示。

ShowStdDevChart(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならばグラフ全体を非表示。

ShowVariabilityChart(Flag As Boolean)

引数のブール値によって、グラフの表示／非表示を切り替える。True ならば表示、False ならばグラフ全体を非表示。

VarianceComponents(option As Integer) As Boolean

対応するオプションをインタラクティブに操作したときと同じ働きをする。メソッドの引数には、表示する統計量の種類を指定する。統計量の種類を示す定数は、`varVarianceComponentConstants` に用意されています。成功すれば True を、失敗すれば False を戻します。

テクノロジーに関する通知

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

All Rights Reserved.

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

NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

- Progressë® Telerik® UI for WPF: Copyright © 2008-2019 Progress Software Corporation. All rights reserved. Usage of the included Progress® Telerik® UI for WPF outside of JMP is not permitted.

- ZLIB Compression Library is Copyright © 1995-2005, Jean-Loup Gailly and Mark Adler.

- Made with Natural Earth. Free vector and raster map data @ naturalearthdata.com.

- Packages is Copyright © 2009–2010, Stéphane Sudre (s.sudre.free.fr). All rights reserved.

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

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

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

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

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

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

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

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

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

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

- This program, "bzip2", the associated library "libbzip2", and all documentation, are Copyright © 1996–2019 Julian R Seward. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

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

Julian Seward, jseward@acm.org

bzip2/libbzip2 version 1.0.8 of 13 July 2019

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

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

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

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

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

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

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

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

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

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

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

- Regarding the decompression algorithm used for UNIX files:

Copyright © 1985, 1986, 1992, 1993

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

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

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

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
- Snowball is Copyright © 2001, Dr Martin Porter, Copyright © 2002, Richard Boulton.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 - Pako is Copyright © 2014–2017 by Vitaly Puzrin and Andrei Tuputcyn.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 - HDF5 (Hierarchical Data Format 5) Software Library and Utilities are Copyright 2006–2015 by The HDF Group. NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright 1998-2006 by the Board of Trustees of the University of Illinois. All rights reserved. DISCLAIMER: THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS "AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. In no event shall The HDF

Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.

- agl-aglfn technology is Copyright © 2002, 2010, 2015 by Adobe Systems Incorporated. All Rights Reserved.

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

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

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

- dmlc/xgboost is Copyright © 2019 SAS Institute.

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

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

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

- libzip is Copyright © 1999–2019 Dieter Baron and Thomas Klausner.

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

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF

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

- OpenNLP 1.5.3, the pre-trained model (version 1.5 of en-parser-chunking.bin), and dmlc/xgboost Version .90 are licensed under the Apache License 2.0 are Copyright © January 2004 by Apache.org. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - You must cause any modified files to carry prominent notices stating that You changed the files; and
 - You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
 - You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
- LLVM is Copyright © 2003–2019 by the University of Illinois at Urbana-Champaign. Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
<http://www.apache.org/licenses/LICENSE-2.0>
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
- clang is Copyright © 2007–2019 by the University of Illinois at Urbana-Champaign. Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
<http://www.apache.org/licenses/LICENSE-2.0>

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

- libld is Copyright © 2011–2019 by the University of Illinois at Urbana-Champaign.

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

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

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

- libcurl is Copyright © 1996–2021, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS file. All rights reserved.

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

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

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

- On the Windows operating system, JMP utilizes the OpenBLAS library. OpenBLAS is licensed under the 3-clause BSD license. Full license text follows: Copyright © 2011-2015, The OpenBLAS Project

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

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

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

ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.