A Technical Publication for JMP® Users
Issue 18 Fall 2005

JMPer Cable

Inside This Newsletter

This issue
of JMPer Cable introduces
the release of **JMP 6**

Introducing JMP 6

Introducing JMP 6

By John Sall,
Executive Vice-President, SAS Institute Inc.

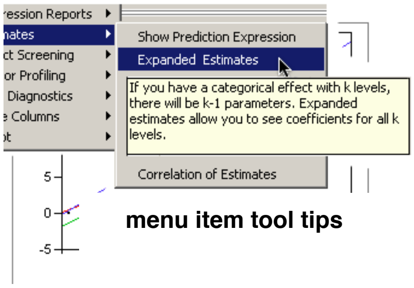
JMP 6 is available now, and you will find it a very compelling upgrade.

Just-in-Time Help

JMP veterans know that JMP is easy to use, but new users face a learning curve. Sometimes users avoid reading manuals but then get frustrated when they can't guess how to complete a task, or when they just don't know the tricks that make something easy. New features in many areas address these frustrations.

The **Help** menu has consolidated several facilities where users are most likely to go for help, and other new help features are distributed where needed.

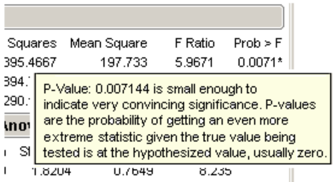
- There is a collection of tutorials that lead you through some sessions to accustom you to the look and feel of JMP. The tutorials are live, connected lessons within JMP, not just static pages of instructions. If your group needs other tutorials, you can write your own as scripts or journals.
- As you progress and want to learn all the hidden shortcuts that make life easier, Tip of the Day helps you to learn new techniques each time you start JMP. For example, Tip 6 shows how to broadcast a command from one analysis to all other similar analyses in the same window (hold down the control key (command key on the Mac)—an invisible feature you could not guess).
- When you are deep in an analysis and click on a hot spot (red-triangle icon), a menu of commands appears, but command names are sometimes not descriptive enough to tell you what the command does. JMP 6 has menu item tool tips. To see them, hover the mouse over a menu item and a brief tip appears, as shown below.
- After you get an analysis report, you might be puzzled as to what the output items mean. Point to something, move the mouse in a little circle, and a tip pops up with a short explanation of that table column, an item in a column, or, in the case of *p*-values, the value and interpretation of the item. We call this feature hover help.



If you have a categorical effect with k levels, there will be k-1 parameters. Expanded estimates allow you to see coefficients for all k levels.

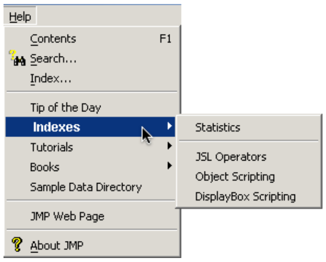
menu item tool tips

hover help



P-Value: 0.007144 is small enough to indicate very convincing significance. P-values are the probability of getting an even more extreme statistic given the true value being tested is at the hypothesized value, usually zero.

There is navigational help to find where to go and how to do a particular analysis. The JMP Starter has long been a table of contents for platforms, and has been improved. However, if you would rather find things in an index, go to **Help** > **Index** to see four different kinds of index browsers:



Statistics—shows how to get a certain statistic in JMP

JSL Operators—describes what each built-in scripting function does

Object Scripting—shows what commands are available for each scriptable object

DisplayBox scripting—describes commands for scripting report display elements

The Statistics index is particularly helpful. For example, to see how to do a Wilcoxon test, scroll in the statistics list (or type the first few letters) to see the Wilcoxon item. Now click on Wilcoxon two groups test to see an explanation. The buttons at the bottom of the index (see *Figure 1*) give more information:

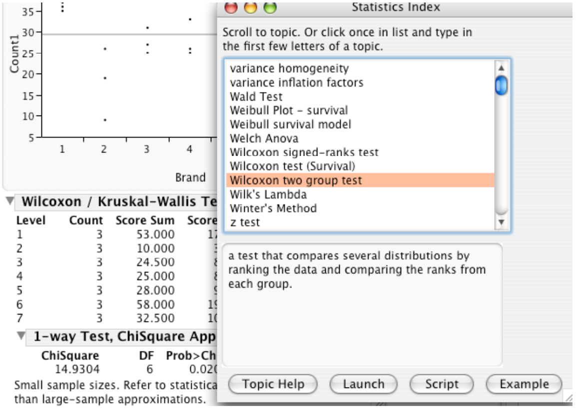
Topic Help shows the appropriate section from the JMP documentation.

Launch launches the platform you need with the options for the specific analysis.

Script shows you a script that can produce the analysis.

Example opens sample data and performs the selected analysis on that data.

Figure 1 Help Menu Statistics Index



Scroll to topic. Or click once in list and type in the first few letters of a topic.

a test that compares several distributions by ranking the data and comparing the ranks from each group.

Small sample sizes. Refer to statistica
than large-sample approximations.

The original help system works as before. From the tools menu, select the question mark cursor and click on output to see help from the documentation for that output. The help menu also includes each book in the JMP documentation set.

Session Management

Before considering the new features for sessions, let's notice that some things you take for granted in JMP that are not in other applications.

- JMP presents graphs and text in the same window. Without this feature, you would have to keep track of the text results and the graphs that go with them, or use a separate mechanism to keep track of them.
- JMP's interactive outlined layout surface organizes results as you open and close sections. Some details are meant to be looked at initially but then hidden while you examine other things. Having a flexible surface allows the document to adapt to the things you need to be visible as you study.
- Intelligent scrolling ensures that titles remain visible until their associated reports scroll off the screen. Without this, you lose context and have to scroll back to identify the current section, or remember what some of the column headings are.
- JMP is document-centric—you see the output almost all the time, rather than modal dialogs and forms.
- The work surface is progressive so that as you dig deeper, you see more context and detail of each analysis. You don't have to switch tools and re-establish context.
- Graphs and plots are interactive, so you can identify each point or select groups of points in any number of graphs and see their corresponding data table observations highlighted.

So what is new in Version 6 that helps you manage your work better? Sometimes the most useful aides are those that help manage mundane challenges such as:

- making work interruptions less painful
- dealing with a messy desktop
- keeping work handy so that you don't have to look for it.

Window List

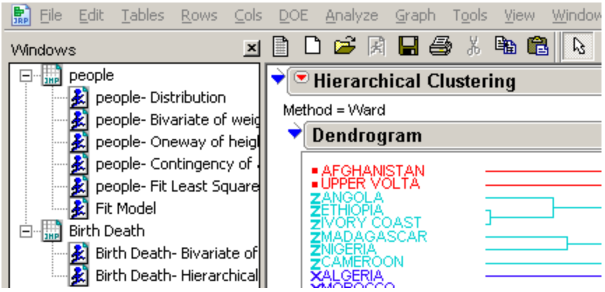
When you analyze data, there are often many analyses and sometimes many open data tables, a situation you might call having a messy desk. To find a particular window, you can always look in the Windows menu.

Unfortunately, that list of windows is not grouped, so all the windows pertaining to a data table are scattered through the menu. In JMP 6 there is a **Window List** command in the **View** menu (not yet available on Mac or Linux) that shows a persistent panel listing all open windows grouped by their associated data table.

The example in *Figure 2* has two open data tables, **people** and **Birth Death**, with multiple analysis windows for each table.

We developed this feature after conducting usability trials, where we and noticed that most novice users, and some veterans, like to work with maximized windows. They could therefore only see one window at a time, and novices had little awareness of which other windows were open. Now, whenever you switch into maximized mode, the window list panel appears so that you can easily refer to windows that don't show. You can then make any window visible by simply clicking on its name in the JMP windows list. This feature is valuable to anyone who has to deal with a desktop that has dozens of open windows associated with several data tables.

Figure 2 Window List Shows when JMP is Maximized



Session save

Sometimes you are interrupted in the middle of a session. You have all the data and analyses in front of you but have to quit JMP and later recreate what you had done. This can be tedious and frustrating.

JMP 6 addresses this time-consuming inconvenience. By default, JMP 6 asks if you want to save your current session when you quit. If you do, then JMP collects scripts from almost every open window and saves a session script. When you start JMP again, it uses that saved script to restore the session. However, if you have unsaved temporary data tables or unsaved temporary scripts, you should save them first so that JMP can reload them in the next session. Assuming default install locations, JMP 6 users can see the script generated by the Session Save facility by looking at these files:

Windows `C:\Documents and Settings\`*`username`*`\Local Settings\Application Data\JMP6 Data\English\session.jsl`

Macintosh `~/Library/Application Support/JMP/session.jsl`

Linux `~/.JMP6/session.jsl`

Sometimes, you might not want to return immediately to that particular session. In that case you can use the command **File > Save Session Script** and save the session script in a specific location. Then, open it later and run it to restore your session.

Project Journal

The previous section showed how to save a session as a script. You can also use a JMP journal to save a session. You can save this journal, and re-open it at any time to duplicate all your windows. The advantage to this method is that you can pick and choose what you want to see.

Previously, JMP only used the journal to save reports, but now journals can also serve as workbooks to save file references and scripts.

- Journals can contain lists of files that you open with a single click.
- You can store scripts from analyses into journal buttons.
- Journals can have buttons that execute any script.
- You can edit, rearrange, and annotate the journal.

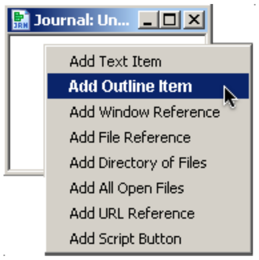
The most spectacular use of these new journals is to make **live** presentations. Imagine that in your next presentation, instead of pasted output, you can create a live analysis at the click of a button. When someone asks a question, you can immediately dig further and show the answer instead of promising to find the answer and report back later.

At SAS, we redesigned our training course, *Modern Design of Experiments*, using a single JMP journal for the whole course. Simulations run at the click of a button. Students have hands-on access to this journal and can run the examples simultaneously with the instructor.

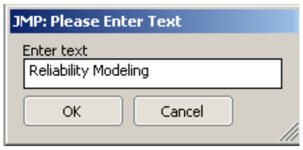
Presentations become more organized as a result of the hierarchical and progressive nature of journals. You open and close outline nodes as you progress through your presentation. The next example shows how to create a presentation using the JMP Journal.

Suppose you want a presentation on reliability. Begin with the **New Journal** command in the **File** menu, then do the following:

1. **Add an outline item.** Right-click on the empty journal window to add an outline item.

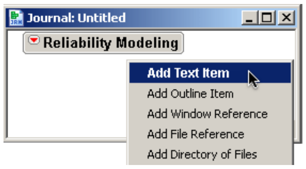


2. **Give it a title**. Type in a title, such as Reliability Modeling, for the outline item.

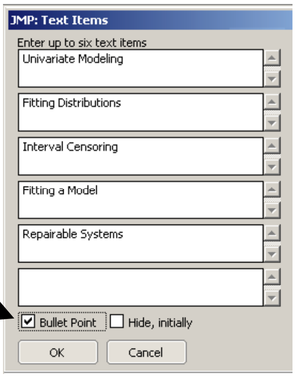


Enter text: Reliability Modeling

3. **Add text.** Right-click the outline box and select **Add Text Item**.



4. **Give the item text bullets (optional).** Click the Bullet Point check box at the bottom of the text items panel.



Enter up to six text items

Univariate Modeling

Fitting Distributions

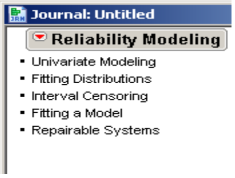
Interval Censoring

Fitting a Model

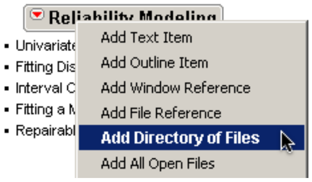
Repairable Systems

☑ Bullet Point ☐ Hide, initially

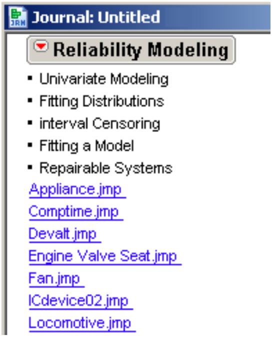
5. **Add file references.** Now you have a bulleted list of topics. The next step is to add links. To add file references, right-click the outline box and choose **Add Directory of Files**. Navigate to the directory you want, open it, and click Select on the Open dialog.



- Univariate Modeling
- Fitting Distributions
- Interval Censoring
- Fitting a Model
- Repairable Systems

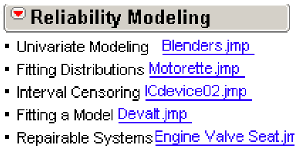


6. **Organize your presentation.** The resulting links open each file in the directory you selected. To place the links with their respective topics, use the selection tool (large plus) and drag file references to bullets. To delete items, highlight them with the large plus and press the delete key.



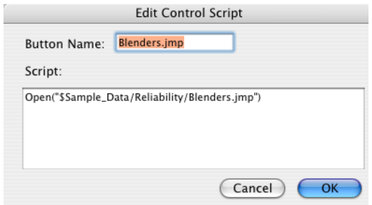
- Univariate Modeling
- Fitting Distributions
- interval Censoring
- Fitting a Model
- Repairable Systems

Appliance.jmp
Comptime.jmp
Devalt.jmp
Engine Valve Seat.jmp
Fan.jmp
ICdevice02.jmp
Locomotive.jmp



- Univariate Modeling Blenders.jmp
- Fitting Distributions Motorette.jmp
- Interval Censoring ICdevice02.jmp
- Fitting a Model Devalt.jmp
- Repairable Systems Engine Valve Seat.jı

7. **Generalize your Presentation.** To make the demo work on other machines, use relative references. Right-click each link, enter the script location, and select **Set Script**. Click **OK** in the dialog. Here, $SAMPLE_DATA refers to the Sample data directory.



Button Name: Blenders.jmp

Script:

```
Open("$Sample_Data/Reliability/Blenders.jmp")
```

Better Data Summarization

JMP 6 introduces several important new platforms for summarizing data, and improved system features.

- Value recode, ordering, and coloring
- Tabulate
- Tree Map
- Missing Value Pattern

Value Recoding, Ordering, and Coloring

Often, a first look at data reveals that it needs cleaning up, as in the histogram shown on the right. The **Recode** command in the **Cols** menu makes this easy. You can see that some months were incorrectly entered in capital letters. When you enter correct values into the Recode dialog and click **OK**, JMP modifies (corrects) the data table. Then, rerun the analysis to see the histogram on the right in *Figure 3*. Note that in JMP 6, when categories come from some natural orderings (such as months), they are automatically displayed in order by JMP platforms. You can also order or color any category as you wish.

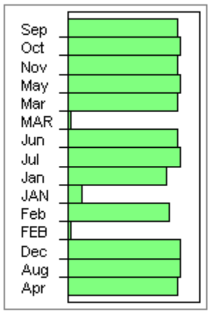
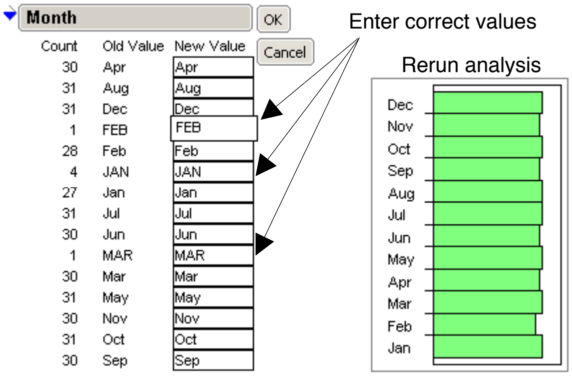


Figure 3 Recode Values to Correct Them



Count	Old Value	New Value
30	Apr	Apr
31	Aug	Aug
31	Dec	Dec
1	FEB	FEB
28	Feb	Feb
4	JAN	JAN
27	Jan	Jan
31	Jul	Jul
30	Jun	Jun
1	MAR	MAR
30	Mar	Mar
31	May	May
30	Nov	Nov
31	Oct	Oct
30	Sep	Sep

Enter correct values

Rerun analysis

In addition to the **Recode** command, other new column properties available in the Column Info dialog for each column help you specify the appearance of values in the data table and on charts and graphs:

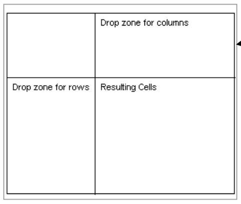
- The labels you specify in the Value Labels dialog appear in the data table instead of the original value, but the original values are not lost. (This feature was available in JMP 5.)
- The Value Ordering property lets you enter column values in the order you want to see them in plots and charts.
- In plots and charts that color levels, such as the

mosaic plot and bar chart, you can specify the colors you want to see with the **Value Colors** property.

Tabulate

If you want bordered tables of summary statistics, then the new Tabulate facility is ideal. Tabulate has a purely drag-and-drop interface, which is different than other platforms. When you drag a categorical column into a row or column border, the classification levels appear there. When you drag a continuous variable and ask for statistics, the statistics for that column appears. Drag a heading from one side to another and the table reorganizes.

Here is a Tabulate example using the **Car Poll** data table from the JMP Sample Data library.



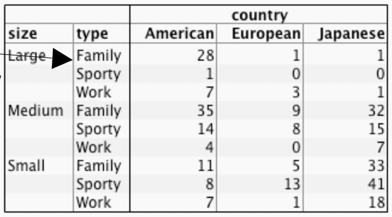
1\. Begin with **Tables > Tabulate**.

2\. Click and drag country to top of the blank table.

	country		
	American	European	Japanese
	115	40	148

2\. Click and Drag **type** to left edge of table.

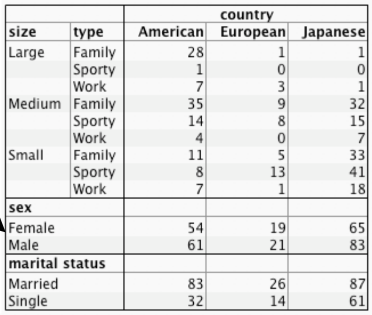
3\. Click and drag **size** to left edge of type.



		country		
size	type	American	European	Japanese
Large	Family	28	1	1
	Sporty	1	0	0
	Work	7	3	1
Medium	Family	35	9	32
	Sporty	14	8	15
	Work	4	0	7
Small	Family	11	5	33
	Sporty	8	13	41
	Work	7	1	18

4\. Click and Drag **sex** to bottom of the table.

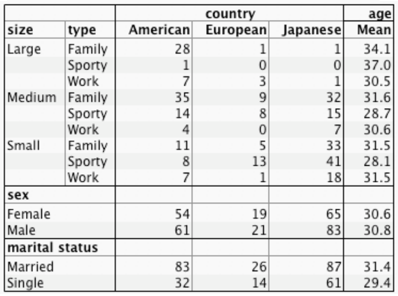
5\. Click and drag **marital status** to bottom of the table.



		country		
size	type	American	European	Japanese
Large	Family	28	1	1
	Sporty	1	0	0
	Work	7	3	1
Medium	Family	35	9	32
	Sporty	14	8	15
	Work	4	0	7
Small	Family	11	5	33
	Sporty	8	13	41
	Work	7	1	18
sex				
Female		54	19	65
Male		61	21	83
marital status				
Married		83	26	87
Single		32	14	61

6 Drag **age** to the far right side of the table and select **New Column Table with Analysis Columns**.

7\. Click and drag **Mean** under **Sum** (to replace the sum with the mean statistic).



		country			age
size	type	American	European	Japanese	Mean
Large	Family	28	1	1	34.1
	Sporty	1	0	0	37.0
	Work	7	3	1	30.5
Medium	Family	35	9	32	31.6
	Sporty	14	8	15	28.7
	Work	4	0	7	30.6
Small	Family	11	5	33	31.5
	Sporty	8	13	41	28.1
	Work	7	1	18	31.5
sex					
Female		54	19	65	30.6
Male		61	21	83	30.8
marital status					
Married		83	26	87	31.4
Single		32	14	61	29.4

Tree Map

Suppose you have hundreds or thousands of category levels and you want to chart frequencies or sums. The Chart platform has difficulty showing that many categories. However, the Tree Map platform (**Graph > Tree Map**) folds categories into tiles that completely fill a rectangular area. The tiles are proportional to the sum or frequency of categories and can be positioned and colored however you want them. Invented by user-interface authority Ben Schneiderman at the University of Maryland, tree maps have become known for their use in disk space mapping, and in web graphs. For an example, see

www.smartMoney.com/marketmap.

The Tree Map platform in JMP supports one or two grouping columns, a sizing column, a coloring column, and one or two sorting columns. The example in *Figure 4* has two sorting columns (*x*, *y* coordinates) that arrange the cities in the chart relative to their geographic location. The population variable determines the box size of each city and the Ozone level variable determines the color.

Missing Value Pattern

Sometimes when you do an analysis you find that the data is sparse (there are lots of missing values). Some columns could be mostly missing, causing many of the observations to be omitted from the analysis. You want to know which columns have the missing data problem.

The **Tables > Missing Value Pattern** command creates a new JMP table that shows the missing value pattern for all or any subset of columns in a source table. Each row in the source table has a pattern of missing values, and the missing value table has a corresponding row for each unique pattern. There is also a column that shows the missing value pattern itself as a list of zeros and ones, and a frequency column with the number of source observations having that pattern.

This new table comes with a Tree Map script that shows (when executed) the composition of the patterns. The Tree Map in *Figure 5* shows only a small section in the upper left with no missing data (all zeros). Below that is a large group with the last four columns missing, and two other large groups. The variables with many missing values show mostly 1s in red at the lower right of the tree map, so if you exclude those variables most observations are used in the analysis.

Figure 4 Tree Map Colored by Ozone Level for Selected Cities, arranged by X, Y Coordinates

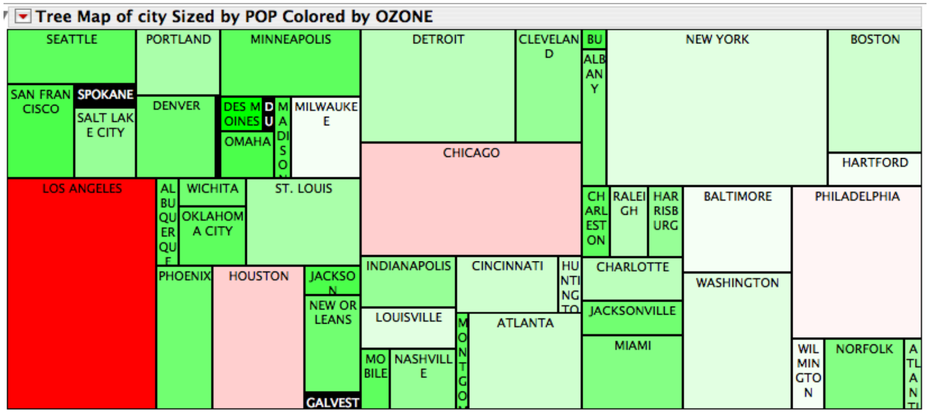
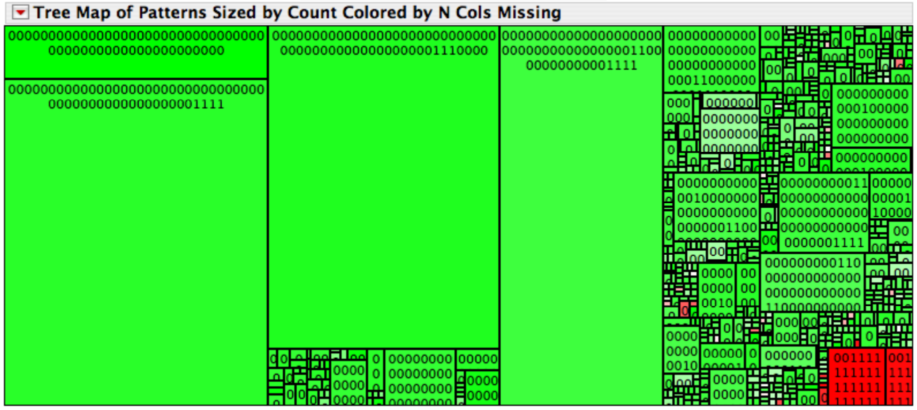


Figure 5 Tree Map Colored by Missing Value Pattern, arranged by Pattern Frequency



Split Plots: They are Everywhere

Consider the following words attributed to Cuthbert Daniel:

> "All industrial experiments are split-plot experiments."

This quote headlines an important added chapter of the new edition of the classic *Statistics for Experimenters,* by Box, Hunter, and Hunter (2005).

Split plots are everywhere, but many experimenters are not trained to recognize them and therefore don't use the correct analysis. Many experiments are inadvertent split plots, and could yield wrong conclusions because they are not recognized and analyzed as split plots.

In one case, a huge investment was made to build an expensive factory to implement a new technology. The decision was based on promising results from an experiment. The new technology did not perform as expected, so the investment was a failure. The experiment, upon review, was found to be a split-plot design that was incorrectly analyzed. When analyzed correctly, the experiment showed that the new technology was not significantly better than the old technology.

Here are some symptoms of a split-plot design:

- You run an experiment in batches where a factor is constant within a batch, such as oven temperature for a batch of units that are heat-treated together.
- A factor is hard to change, so you sort the runs in such a way that you don't have to change that factor often.
- You experiment with individual subjects (lab animals or people) and take multiple measurements on those subjects under different experimental conditions. This is called a repeated-measures design, but it is equivalent to a split plot.
- A factor is applied to larger units. These larger units are subdivided and the subunits are treated differently, like the classic split plots in agriculture.
- You take multiple measurements for a given response as in measurement systems analysis.

The common trait in split-plot designs is that a factor is constant within a block but not across blocks, which means there is a restriction on the randomization of runs.

JMP 6 is revolutionary with respect to split plots in two important ways:

1. JMP can design **optimal** split-plot experiments. No other product does this.
2. JMP can analyze any split plot with the best statistical methods and with a minimum of expertise required from the user.

These features position JMP 6 as a pioneer in split-plot designs. The following sections give more detail about the way JMP handles split-plot designs.

Design

In experimental design, optimal split-plot design is a brand new field and JMP is the only package to offer it. This new field was introduced by Peter Goos (2002) in published papers, and in his excellent book, *The Optimal Design of Blocked and Split-Plot Experiments*. Goos used his own custom programs involving candidate set row exchange. Bradley Jones, a JMP developer, advanced Goos's work by adapting coordinate exchange and other methods to work for split plots.

Goos's book has interesting case studies that compare two recent split-plot designs that were handcrafted by experts in the field and published in *Technometrics*. In both cases, when compared to the optimal split-plot design, the expert's designs were only 77% and 48% efficient. That's like throwing away 23% or 52% of your research budget.

Despite the advanced and pioneering nature of split-plot designs, using JMP to design split-plot experiments requires only two simple design specifications:

- You can specify that a factor is hard to change compared to the other factors. These hard-to-change factors become the whole-plot factors.
- When you decide on the number of runs, you can enter both the number of whole plots (the number of times you are willing to change the hard-to-change factors) as well as the total number of runs in your budget.

You can also specify the ratio of whole-plot to split-plot variance, which sometimes makes a difference. However, Goos showed that the optimal designs are invariant over a surprisingly large range of variance ratios, so the default ratio is generally applicable.

Everything else is automatic.

Analysis

There is long history of method improvements for analyzing split-plot data, but only recently has this research come to full fruition. The main improvements began with the work on REML by Patterson and Thompson, (1971), followed by Kackar and Harville (1984), and most recently Kenward and Roger (1997). The goal is to produce an automated general method for all cases that gets the best possible answer with a minimum of user expertise. The software should do the work and not require the user to do complicated setup, do side-calculations, or depend on balanced arrangements of data. The software should figure out the distribution of the test statistic that best fits instead of counting degrees of freedom by side formulas. The automated method gives correct answers without the requirements of balanced design. Now with today's fast computers, these calculations have become practical. JMP has long supported REML estimates, but only JMP 6 supports the Kackar-Harville and Kenward-Roger improvements that produce the best test in a wide variety of situations. Previous versions of JMP had difficulty counting degrees of freedom when there was a whole plot error that was not a syntactic cross of whole plot factors, and especially when a whole plot factor was continuous. JMP 6 does it right.

Now your only responsibility is to specify a whole plot identifier effect in the model and declare it as a random effect. Everything else is automatic. You don't have to specify the terms in order. The only other software we know that has Kenward-Roger adjustments is SAS, with Proc MIXED and Proc GLIMMIX.

Additionally, the new REML estimates allow negative estimates of the variance components. The variance components can be interpreted as variances but they are really covariance parameters. Traditional (balanced) methods have always allowed negative estimates, but REML methods have had difficulty handling them. It turns out that, though the variances can become negative, the test statistics derived from them are valid. For those REML implementations that constrain the estimates at zero, there are two bad consequences.

- First, the BLUP parameters become non-estimable and counting becomes a problem for non-Kenward-Roger testing routines.
- Second, constraining the estimates becomes problematic on the null distribution of the tests on fixed effects, biasing the size of the tests.

For this reason, the default choice in JMP is to not constrain the variance components to be non-negative.

Now, most negative estimates can be considered as random aberrations in cases when the real variance component is small, but positive.

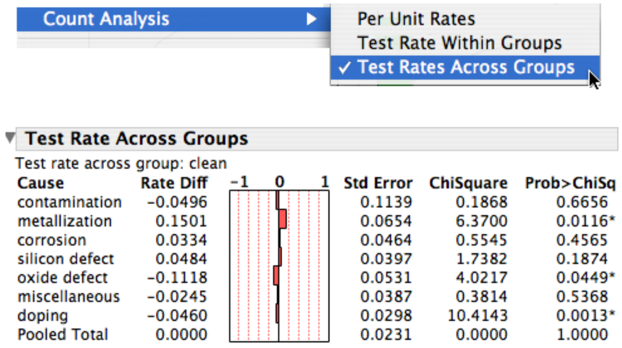
However, it is also possible to introduce un-modeled dependencies within blocks for which a negative variance component estimate is a symptom of the dependency, and thus has diagnostic value, as well as being healthy for the size of the test. Chris Gotwalt developed the new methods in JMP 6.

Other Experimental Design features include improved spherical DOE, nonlinear DOE, a new space-filling method, and enhanced scriptability through JSL.

Complete Six Sigma® Toolkit

The Pareto platform now does defects per unit (DPU) statistics and tests. The example in *Figure 6* uses the **Failure3** sample data table, and looks at the difference in cause of failure rates using the new command, **Count Analysis > Test Rates Across Groups**, in the Pareto platform menu. Other commands give a table of within-group rates, and test those rates for difference between causes.

Figure 6 Test Rates Across Groups



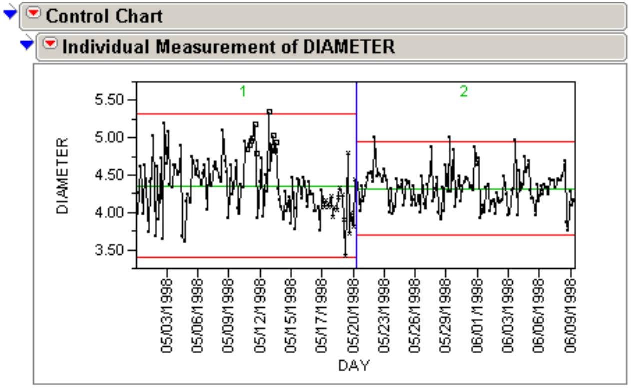
Test Rate Across Groups

Test rate across group: clean

Cause	Rate Diff	-1 0 1	Std Error	ChiSquare	Prob>ChiSq
contamination	-0.0496		0.1139	0.1868	0.6656
metallization	0.1501		0.0654	6.3700	0.0116*
corrosion	0.0334		0.0464	0.5545	0.4565
silicon defect	0.0484		0.0397	1.7382	0.1874
oxide defect	-0.1118		0.0531	4.0217	0.0449*
miscellaneous	-0.0245		0.0387	0.3814	0.5368
doping	-0.0460		0.0298	10.4143	0.0013*
Pooled Total	0.0000		0.0231	0.0000	1.0000

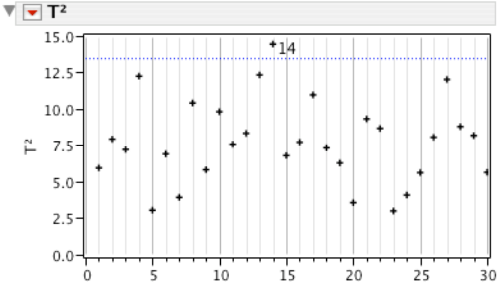
Phase charts are available as part of Shewhart Control charts for variables and attributes. When you have a phase variable, there are side-by-side control charts for each level of the phase variable, as shown in *Figure 7*. You can then save limits for each phase and sigma in a new data table.

Figure 7 Phase Variable in Control Charts



A basic Multivariate control chart, like the T^2 plot in *Figure 8*, is available in the Multivariate platform.

Figure 8 T^2 Multivatiate Control Charts



Also, new in the Distribution platform, is the ability to fit Binomial and Poisson distributions, important in defect rate analysis.

The Variability platform has a number of improvements, including the ability to handle large-but-unbalanced variance component analysis efficiently.

All these improvements will be welcomed by the black belt practitioners of Six Sigma®.

Modeling

When you want to fit counts to a model, you should handle the response as Poisson-distributed instead of using ordinary least squares. In JMP 6, the way to analyze counts is with the Generalized Linear Models personality of the Fit Model platform.

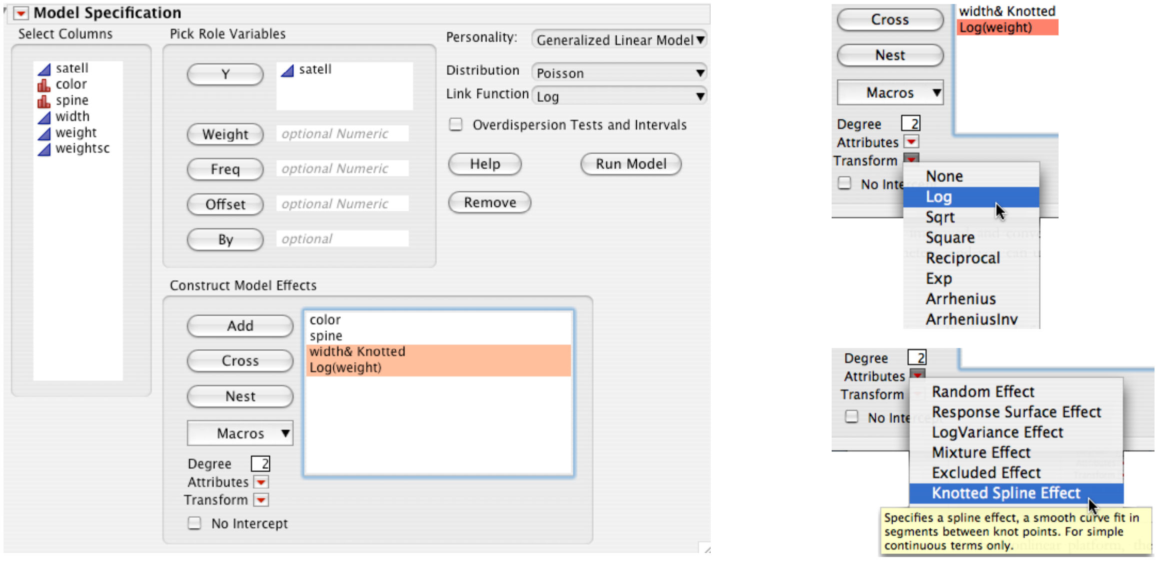
Fit Model can make transformations on the terms in the model, and supports a basic kind of knotted splines to allow flexible curves. *Figure 9* (next page) shows the Model Specification dialog with these new features.

In the Nonlinear platform, the numerical methods are improved and converge more reliably. There are more choices of methods, you can impose bounds on parameters, and you can use the profilers to see how changing the parameter values affect the fit.

Exploratory Studies

The Partition platform (see *Figure 10*, next page), introduced in JMP 5, is a powerful tool for investigating relationships in data. This decision-tree method has been traditionally used for data mining, but the JMP Partition platform is very effective for finding root cause clues for defect rates. JMP 6 improves this tool with better defaults, improved cross-validation facilities, and better diagnostics of fit, such as the Receiver Operating Characteristic (ROC) curve, and Lift curves, shown on the right in *Figure10*. The response variable can be either numeric or character, and the factors can be any combination of numeric and character variables.

Figure 9 Model Specification with New Options



Personality: Generalized Linear Model

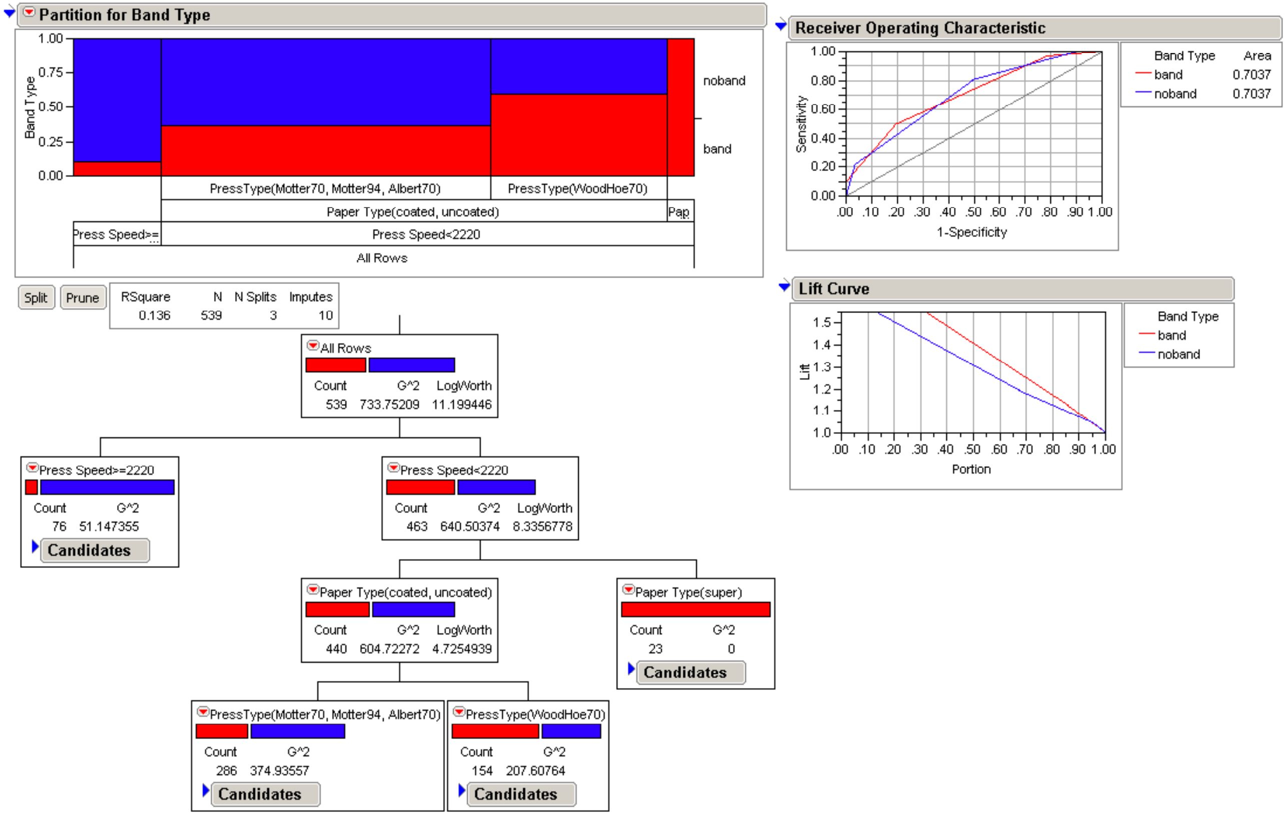
Distribution: Poisson

Link Function: Log

Degree: 2

Specifies a spline effect, a smooth curve fit in segments between knot points. For simple continuous terms only.

Figure 10. Partition Plot with ROC Curve and Lift Curve



RSquare	N	N Splits	Imputes
0.136	539	3	10

Node	Count	G^2	LogWorth
All Rows	539	733.75209	11.199446
Press Speed>=2220	76	51.147355	
Press Speed<2220	463	640.50374	8.3356778
Paper Type(coated, uncoated)	440	604.72272	4.7254939
Paper Type(super)	23	0	
PressType(Motter70, Motter94, Albert70)	286	374.93557	
PressType(WoodHoe70)	154	207.60764	

Band Type	Area
band	0.7037
noband	0.7037

JMP Newswire eNewsletter

Want to stay informed as to what's going on at JMP? Then we invite you to subscribe to JMP Newswire, our free email newsletter. Published every other month, it contains:

- The latest news about JMP software, updates, and training
- A calendar of upcoming tradeshows, webinars, seminars, and SAS User Group conferences featuring JMP
- News about special events and promotional offers
- Application stories about people who are using JMP in new and exciting ways

To subscribe, go to www.jmp.com/news/newsletter/subscribe.shtml

Surface Exploration

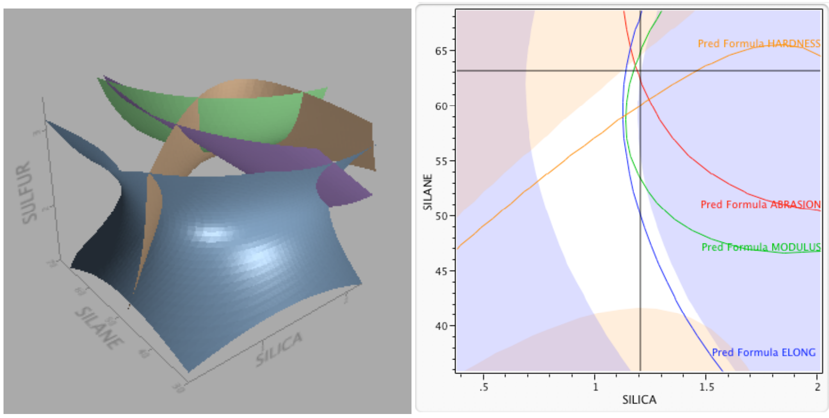
Often you want to know how to set factors to produce results that:

- optimize responses
- are robust (less sensitive to variation in the factors)
- cost less
- perform better.

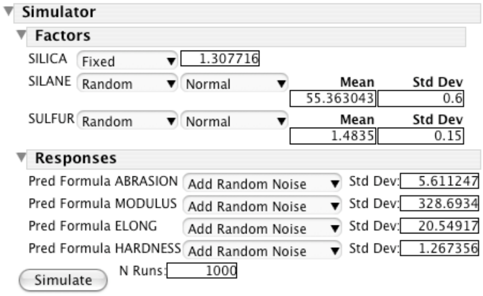
Doing the statistical fit of a model is just the first step. JMP has always been capable of exploring and optimizing fitted surfaces, and JMP 6 takes optimization further.

Surface plots can visualize response surfaces for three factors in 3D space, as shown on the left in *Figure 11*. The Contour Profiler has transparent shading to show feasible regions. Optimization is improved, especially for large categorical problems.

Figure 11 Response Surface and Contour Profiler with Transparent Shading



The **Simulator** command found in the Prediction and Contour profilers allows you to do tolerance-design Monte Carlo explorations on how variation in the factors and the model affect the distribution of the responses. A simulator panel, like the one shown here, lets you add random noise, multivariate noise with assigned correlation among the responses, and standard deviation values.



SILICA Fixed 1.307716

SILANE Random Normal Mean 55.363043 Std Dev 0.6

SULFUR Random Normal Mean 1.4835 Std Dev 0.15

Pred Formula ABRASION Add Random Noise Std Dev: 5.611247

Pred Formula MODULUS Add Random Noise Std Dev: 328.6934

Pred Formula ELONG Add Random Noise Std Dev: 20.54917

Pred Formula HARDNESS Add Random Noise Std Dev: 1.267356

N Runs: 1000

The simulation results are written to a new data table. This new table has a Distribution script ready to show you the results. If you have Specification Limits in your original formulas, these limits are copied into the new table and automatically yield defect rates and capability indices for the simulation.

A fourth profiler, the Custom Profiler, is available for large or complex models that are too big for the graphical profilers to be practical.

Should your company be a JMP Partner? Here are a few of the JMP Partners:

VelQuest Corporation

Velquest provides automation tools that provide record authentication and record maintenance to support FDA 21 CFR Part 11 regulations.

Thomas A. Little Consulting

Specialists in Corporate Training and Implementation Solutions for Six Sigma and Quality Improvement.

Zumatrix, Inc.

Zumatrix is a leading supplier of truly configurable Laboratory Information Management Systems (LIMS) in North America.

Advanced Integrated Technologies Group, Inc.

The AIT Group improves profitability through a combination of great people, hard work, and three proven world class lines of business—Supply Chain Management, Six Sigma, and Lean Production.

Industrial Extension Service of North Carolina State University

NC State's Industrial Extension Service, the first of its kind in the US, was established in 1955 to help North Carolina industries grow and prosper.

iGrafx

iGrafx is a suite of business management tools that help organizations understand, analyze and optimize their processes to reduce costs and increase return on investment.

SigmaFlow

Only SigmaFlow seamlessly integrates the complex data and relationships that make up your organization, and provides an integrated one-stop-shop toolkit for managing your Lean and Six Sigma projects.

Premier Performance Network (PPN)

PPN is a global leader in consulting and training in corporate performance and quality improvement, particularly Six Sigma implementation.

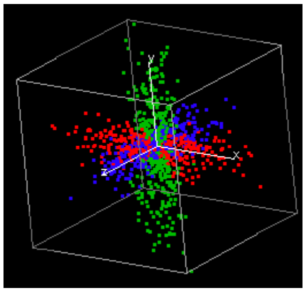
See

www.jmp.com/partners/partners.shtml

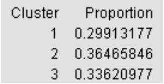
for a complete list of JMP partners.

Normal Mixtures

Clustering helps you see the structure of multivariate data. But what if the clusters overlap? Normal mixtures can overcome this problem in order to get a good picture of the proportion of points in each group, even if many points cannot have their cluster membership identified because they are in overlapping clusters.



JMP 6 has an improved normal mixtures algorithm that does direct optimization and incorporates robust methods to reduce the influence of outliers. To challenge this algorithm, the script shown below creates three clusters in three variables, all of which have the same mean, and produces the spinning plot shown to the right. The only difference in the clusters is in the standard deviation along the coordinate directions: 35% of the data has more variance along *X*, 35% has more variance along *Y*, and 30% has more variance along *Z*. The resulting proportion estimates given by the *k*-means normal mixture routine are very close, given this population size (1000). It is amazing that *any* software can identify groups this well even though the groups have the same means.



Cluster	Proportion
1	0.29913177
2	0.36465846
3	0.33620977

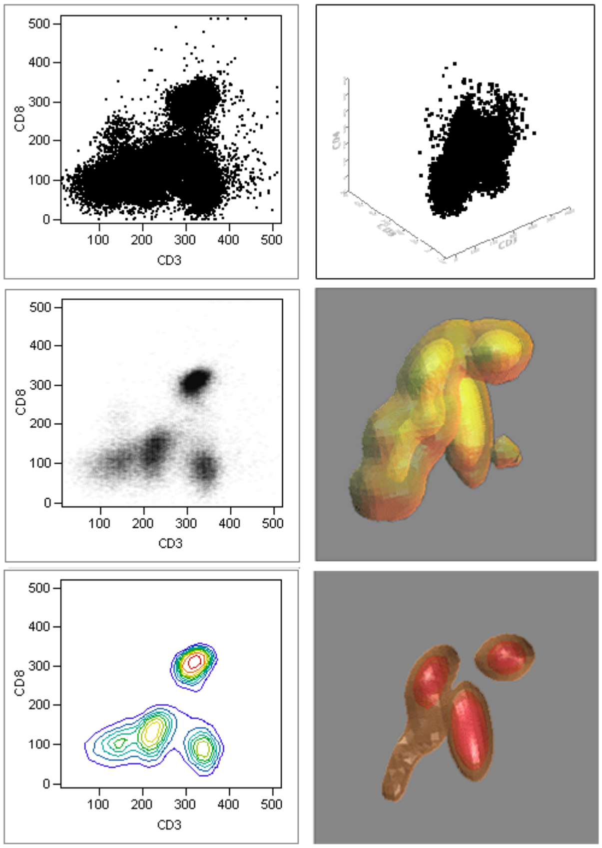
```
newTable("3CL",
    newColumn("X"),  newColumn("Y"),  newColumn("Z"),
    addRows(1000));
    for each row(
        rn = randomUniform();
        if (
            rn<.35,:x=randomNormal(0,4); :y=randomNormal(0,1); :z=randomNormal(0,1),
            rn<.70,:x=randomNormal(0,1); :y=randomNormal(0,4); :z=randomNormal(0,1),
                   :x=randomNormal(0,1); :y=randomNormal(0,1);: z=randomNormal(0,4)
    )
);
SpinningPlot(Y(:X,:Y,:Z));
```

Multivariate Density

Suppose you have a large amount of data for two variables and want to see a revealing picture of the distribution. In *Figure 12*, a simple scatterplot, (top-left plot), is a dense cloud—you can't tell where the densest area is or find patterns in the density. One feature to better judge densities is to make the points transparent so that only the darkest areas build up to a solid color. The middle-left plot shows the same data with transparency set so that the points are only 5% dense. The bottom-left plot shows contours given by the Bivariate platform using normal kernel density estimates with contours on the quantiles. The 10% densest data are in the innermost area and 90% of the data are within the outermost contours.

A JMP 6 enhancement shows density contours in three dimensions. The plot on the upper right in *Figure 12* is the same data in a 3-D scatterplot, but you can't see much structure with a black cloud of points. Kernel density estimation and quantile contour surfaces show the structure much better. The middle-right plot shows three layers of transparent surfaces corresponding to the 50%, 90%, and 95% shells of the density. The bottom-right plot has the outermost shell turned off, which allows you to better see inner shells. These pictures rotate smoothly giving a 3-dimensional view. With a good graphics card and Crystal Eyes® glasses and software, the 3-D effect becomes very real.

Figure 12 Graphical Presentation of Multivariate Data

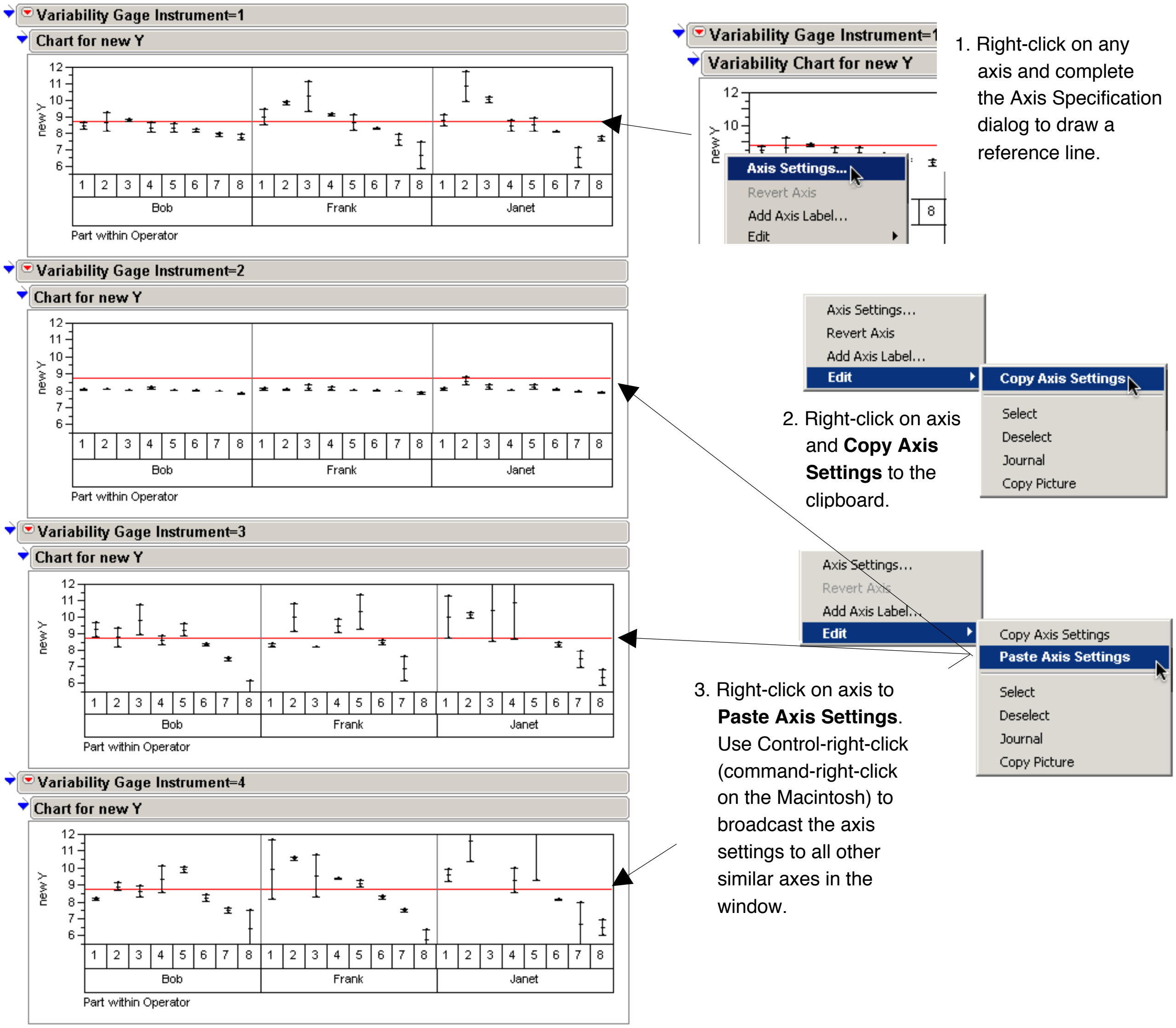


Solving Problems to Scale

Originally, JMP was not designed to handle big problems. The SAS System is designed to solve big problems and should be the tool of choice for them. Nevertheless, computer memory is getting cheaper and computers are getting faster, so the scale of problems you can address with JMP is getting larger. JMP 6 provides some new features that make it more convenient to work on larger problems.

- The old limit of 10,000 columns is eliminated, and we have tested some platforms successfully with 20,000 columns.
- Graph printing has been optimized so when there are millions of points, the image is a bitmap instead of the draw commands to make millions of points. This greatly reduces the time needed to print large, complex plots.
- You can now make massive simultaneous parallel changes to graphs. You can copy axis or frame settings from one graph and paste to another, or control-paste (option-paste on the Macintosh) to broadcast these settings to all similar axes or frames. *Figure 13* illustrates an example. Suppose you have Variability Charts for each level of a By-variable, and want to add a common reference line to all of them.

Figure 13 Use Platform Options to Broadcast Axis or Frame Settings from One Graph to All Others



1. Right-click on any axis and complete the Axis Specification dialog to draw a reference line.

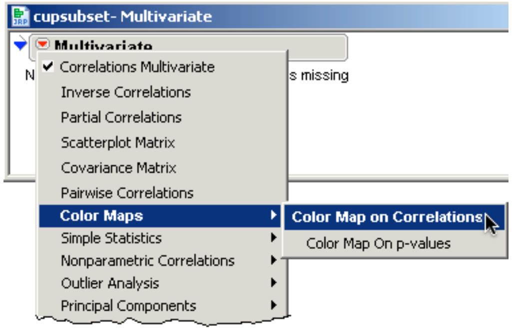
2. Right-click on axis and **Copy Axis Settings** to the clipboard.

3. Right-click on axis to **Paste Axis Settings**. Use Control-right-click (command-right-click on the Macintosh) to broadcast the axis settings to all other similar axes in the window.

The missing data pattern, described earlier, helps you locate missing value patterns. The tree map of the missing data pattern on the left in *Figure 14* is for a subset of the KDD Cup data (Knowledge Discovery and Data mining) for 1998. The data used for this example is cupsubset.jmp, available on www.jmp.com with this edition of the JMPer Cable. It has 79 columns and 60,000 rows, with many missing cells. Using the **Tables** > **Missing Data Pattern** command shows that over 30,000 observations have columns of missing values, and many have more than half the columns with missing values.

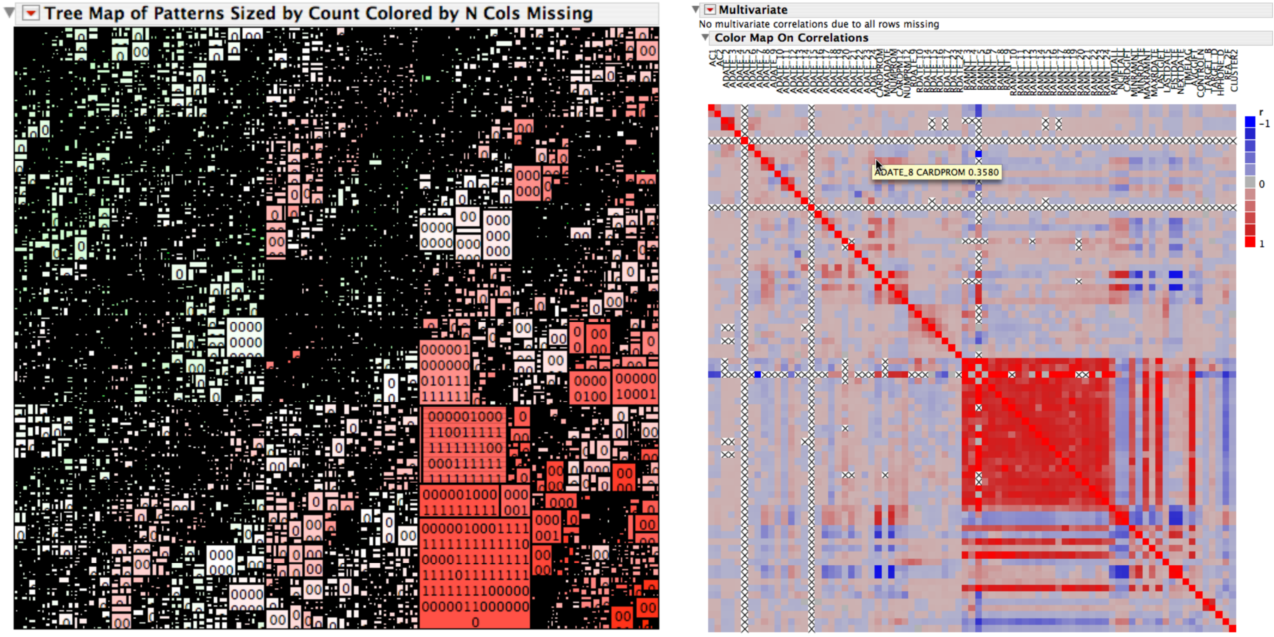
Suppose you want to see correlations among the 79 variables. If you use the **Multivariate** command in the **Analyze** menu, the first result tells you only that there are no multivariate correlations because of missing values. The multivariate correlations are only computed for pairs of variables that have no missing values at all. Instead, you want to see all the pairwise correlations, which are computed for all pairs of nonmissing values.

The best way to see this large number of pairwise correlations is to use the **Color Map on Correlations** command to show a giant pairwise correlation matrix in the form of a cell plot. This command automatically computes and plots all the pairwise correlations.



You can easily spot pairwise all-missing situations, and areas of high correlation for a group. Correlations approaching 1 and –1 show in bright red and blue. Pairs of variables with no correlation are marked with a small x. Hover over a cell to see the pair of variables it represents, as shown below.

Figure 14 Missing Value Tree Map and Cell Plot of Pairwise Correlations for 79 Variables.



No multivariate correlations due to all rows missing

What they are saying…

> "JMP has become the 'Swiss Army Knife' tool in our process maintenance and improvement efforts. With every day that passes, we depend on its robust feature sets more and more."
>
> —Robert Reid
>
> CTS Microelectronics

> "I studied more than 10 products all directed to facilitating industrial experiments. JMP 5 ranked as the best value of all products tested that support full general applications of applied statistics (graphing, summary statistical methods, statistical process control, design of experiments, etc.) in manufacturing."
>
> —Jack Reece, Ph.D., Reece Associates, Ltd.
> Fellow, Statistical Methods Group of SEMATECH (Retired)

Working with SAS, Working with XML

JMP 6 for Windows continues to support reading and writing the latest SAS data sets, and SAS 9 can now create JMP files with Proc Export. If you have a SAS session on your own machine, the IOM interface from JMP to SAS can submit lines to SAS without the SAS Integration Technology. You can even create a SAS job stream to submit data and model to SAS with a shift-click on the hot spot on the Fit Model dialog title bar. You can also use JMP's SAS interface to submit stored procedures to the SAS Metabase.

For those of you that connect JMP to XML-aware applications, JMP scripts can now parse XML text. For example, the SAS Genomics systems construct dialogs using XML specifications. The scripts that show how to parse this XML and create the custom JMP report in *Figure 15* are shown on page 15.

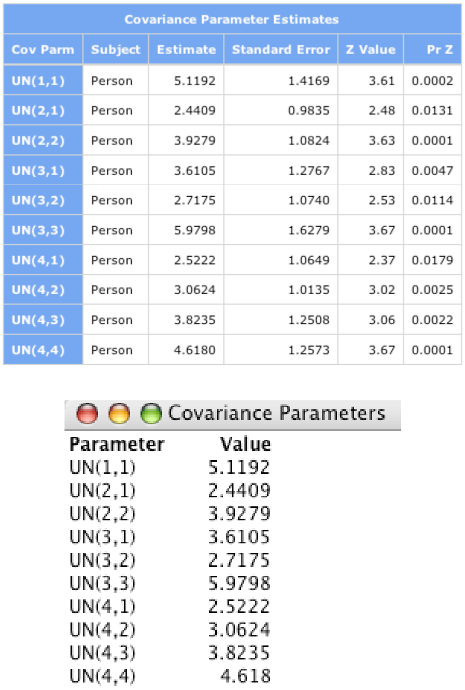
You could make a report of the whole output by using the built-in SAS/ODS/XML mechanism:

```
NewWindow(“mixed”,
loadTextFile("odsMixedFragment.xml",SASODSXML));
```

Once you have the result in a JMP report, it is easy to convert it to a data table with the **Make into Data Table** command, and then use the Chart platform to see a bar chart, as shown in *Figure 16*.

JMP can interface to the SAS Metadata Server with a script, MetaDataAccess.JSL, that we provide on our website.

Figure 15 HTML table (top) and Custom Generated JMP Report (Bottom)



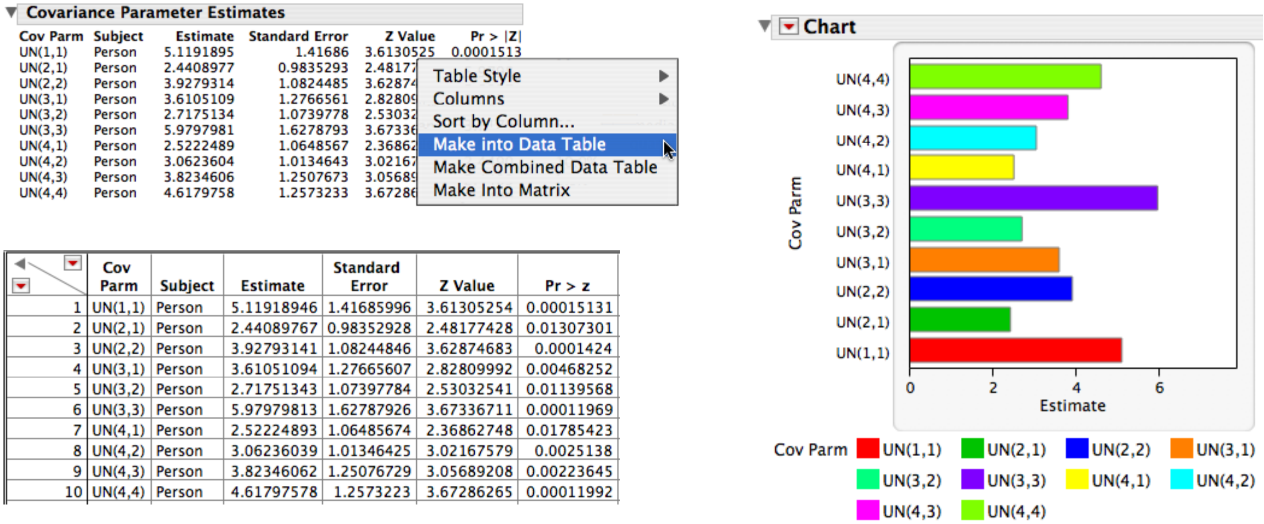
Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr Z
UN(1,1)	Person	5.1192	1.4169	3.61	0.0002
UN(2,1)	Person	2.4409	0.9835	2.48	0.0131
UN(2,2)	Person	3.9279	1.0824	3.63	0.0001
UN(3,1)	Person	3.6105	1.2767	2.83	0.0047
UN(3,2)	Person	2.7175	1.0740	2.53	0.0114
UN(3,3)	Person	5.9798	1.6279	3.67	0.0001
UN(4,1)	Person	2.5222	1.0649	2.37	0.0179
UN(4,2)	Person	3.0624	1.0135	3.02	0.0025
UN(4,3)	Person	3.8235	1.2508	3.06	0.0022
UN(4,4)	Person	4.6180	1.2573	3.67	0.0001

Covariance Parameters

Parameter	Value
UN(1,1)	5.1192
UN(2,1)	2.4409
UN(2,2)	3.9279
UN(3,1)	3.6105
UN(3,2)	2.7175
UN(3,3)	5.9798
UN(4,1)	2.5222
UN(4,2)	3.0624
UN(4,3)	3.8235
UN(4,4)	4.618

Figure 16 Make JMP Output (top) into Data Table (bottom) and Create a Chart (right)



Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr > |Z|
UN(1,1)	Person	5.1191895	1.41686	3.6130525	0.0001513
UN(2,1)	Person	2.4408977	0.9835293	2.48177	
UN(2,2)	Person	3.9279314	1.0824485	3.62874	
UN(3,1)	Person	3.6105109	1.2766561	2.82809	
UN(3,2)	Person	2.7175134	1.0739778	2.53032	
UN(3,3)	Person	5.9797981	1.6278793	3.67336	
UN(4,1)	Person	2.5222489	1.0648567	2.36862	
UN(4,2)	Person	3.0623604	1.0134643	3.02167	
UN(4,3)	Person	3.8234606	1.2507673	3.05689	
UN(4,4)	Person	4.6179758	1.2573233	3.67286	

	Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr > z
1	UN(1,1)	Person	5.11918946	1.41685996	3.61305254	0.00015131
2	UN(2,1)	Person	2.44089767	0.98352928	2.48177428	0.01307301
3	UN(2,2)	Person	3.92793141	1.08244846	3.62874683	0.0001424
4	UN(3,1)	Person	3.61051094	1.27665607	2.82809992	0.00468252
5	UN(3,2)	Person	2.71751343	1.07397784	2.53032541	0.01139568
6	UN(3,3)	Person	5.97979813	1.62787926	3.67336711	0.00011969
7	UN(4,1)	Person	2.52224893	1.06485674	2.36862748	0.01785423
8	UN(4,2)	Person	3.06236039	1.01346425	3.02167579	0.0025138
9	UN(4,3)	Person	3.82346062	1.25076729	3.05689208	0.00223645
10	UN(4,4)	Person	4.61797578	1.2573223	3.67286265	0.00011992

References

Box, G.E.P., Hunter, J.S., and Hunter, W.G. (2005) *Statistics for Experimenters*, New York, John Wiley and Sons, Inc., 336.

Goos, Peter (2002), *The Optimal Design of Blocked and Split-Plot Experiments*, New York, Springer-Verlag.

Kackar, R.N., and Harville, D.A., (1984) “Approximations for Standard Errors of Estimators of Fixed and Random Effects in Mixed Linear Models,” *Journal of American Statistical Association*.

Kenward, M.G., and Roger, J.H. (1997) “Small Sample Inference for Fixed Effects Estimators from Restricted Maximum Liklihood,” *Biometrics* 53, 982-997.

Patterson, H.D. and Thompson, R. (1971) “Recovery of Inter-Block Information when Block Sizes are Unequal,” *Biometrics*, 58, 545-554

Parsing Script

```
// parseODSCovParm.jsl
source = loadTextFile("odsMixedFragment.xml");
// stream states
inCovParm = 0; inBody = 0; inLabel = 0; inValue = 0;
// output values
covParmName = {}; covParmValue = [];
parseXML(source,
             // Look for CovParms Table
             OnElement("output",
               StartTag(inCovParm=If(xmlAttr("name")=="CovParms",1,0)),
               EndTag(inCovParm=0)
             ),
             // Look for output-body
             OnElement("output-body",
               StartTag(inBody=1),
               EndTag(inBody=0)
             ),
             OnElement("header",
               StartTag(inLabel=If(xmlAttr("name")=="CovParm",1,0)),
               EndTag(inLabel=0)
             ),
             OnElement("data",
               StartTag(inValue=If(xmlAttr("name")=="Estimate",1,0)),
               EndTag(inValue=0)
             ),
             OnElement("value",
               StartTag(),
               EndTag(
                       if(inCovParm&inBody,
                               if(
                                      inLabel,InsertInto(covParmName,xmlText()),
                                      inValue,covParmValue=covParmValue|/num(xmlText()))
                               );
                       )
             ),
);
newWindow("Covariance Parameters",
             TableBox(
               StringColBox("Parameter",covParmName),
               NumberColBox("Value",covParmValue),
             )
);
```

XML Specifications

```
<leaf name="CovParms" label="Covariance Parameter Estimates" class="ContentItem" toc-level="2">
<output name="CovParms" label="Covariance Parameter Estimates" clabel="Covariance Parameter Estimates">
<output-object type="table" class="Table">
  <colspecs columns="6">...</colspecs>
  <output-head>...</output-head>
<output-body>
<row>
<header name="CovParm" label="Cov Parm" type="string" class="RowHeader" row="3" column="1">
<value>UN(1,1)</value>
</header>
<header name="Subject" label="Subject" type="string" class="Data" row="3" column="2">
<value>Person</value>
</header>
<data raw-value="QBR6DMO4ymg=" name="Estimate" label="Estimate" type="double" class="Data" row="3"
  column="3">
<value>5.1192</value>
</data>
<data raw-value="P/ardVgwZvM=" name="StdErr" label="Standard Error" type="double" class="Data"
  row="3" column="4">
<value>1.4169</value>
</data>
...
</row>
...more rows...
</output-body>
</output-object>
</output>
</leaf>
```

JMP
SAS Campus Drive
Cary, NC 27513 USA
Tel: (919) 677-8000

About JMPer Cable

Issue 18 Fall 2005

JMPer Cable is mailed to JMP users who are registered users with SAS Institute. It is also available online at www.jmp.com

Contributor, Special Edition
John Sall

Editor
Ann Lehman

Printing
SAS Institute Print Center

Questions, comments, or for more information about JMP, call
1-877-594-6567
or visit us online at
www.jmp.com

To Order JMP Software
1-877-594-6567