

# JMP<sup>®</sup>er Cable<sup>®</sup>

NEWSLETTER FOR JMP<sup>®</sup> USERS



## FRUSTRATION REDUCTION

**John Sall**  
**Senior Vice President**  
**SAS Institute Inc.**

As work continues on Version 4 of JMP, we find that some of the greatest effort goes not into new features, but into reducing whatever frustration users have with the current features. We have shifted gears a little in how we see our obligation as software developers. This is especially the case as JMP matures and has fulfilled its first charter. JMP will grow more, but we see our effort best spent to make our product do its current job better, not take on too many new jobs.

I'm an aficionado (gourmet reader) of good junk mail. I think that copy writing is a difficult and high art. Three years ago, I received one of the greatest pieces of junk mail ever, a sample newsletter called the *Samarai Sword* from Corporate Visions<sup>®</sup>. Even though I didn't buy their product, the newsletter did change my thinking from that point on. The newsletter was about sales techniques, and the big message was "Find the pain." There are people out there that don't have your product and are *suffering* because of it. You think about a situation in

which having your product makes all the difference, and then all you have to do is tell that story.

For JMP, one story is discovery – that without the graphics and interactivity, you are *painfully* missing those discoveries that would make all the difference.

## IN THIS ISSUE

<b>Frustration Reduction.....</b>	<b>1</b>
<b>Professional Services Update.....</b>	<b>4.</b>
<b>There and Back Again Part 1: Floppy Disks.....</b>	<b>5</b>
<b>There and Back Again Part 2: Sharing on a Network .....</b>	<b>8</b>
<b>Calculating Fitted Values for a Y by X Spline Fit.....</b>	<b>12</b>
<b>The Runs Test: Nonparametric Testing for Randomness in a Series of Runs.....</b>	<b>16</b>
<b>Calculator Corner.....</b>	<b>18</b>
<b>Tips and Techniques.....</b>	<b>20</b>
<b>Upcoming Trade Shows.....</b>	<b>22</b>
<b>JMP Data Discovery Conference...</b>	<b>23</b>



Recently we started reviewing other statistical packages. We needed to find out what kinds of pain the users of these other packages are suffering. This will help us identify ways to present our product. For example, most products don't integrate statistics and graphics into the same window, and we need to show that it's really frustrating to keep track of what graph goes with what text report when they appear in different places.

But of course, in looking at other products we also find out more about our own product, by contrast. So we ask ourselves, what frustrations are our users feeling, or are susceptible to feeling? We want to find problems before our users do, and before our competitors do. If we find them, we can fix things for a future release.

I won't tell you about all the things we are finding because we all have some hypochondriac tendencies, and would suffer more if we were made more aware of our problems. Or, you can consider it your job to tell us what

is bothering you. We welcome your input.

But I might mention a few items, just to prove that we are feeling the pain signals. One problem concerns the Macintosh, one problem concerns Windows, and one problem is portable. All three are good examples of frustrations that we see every day and have learned to cope with, but are improving for Version 4.

On the Macintosh, if you paste a picture from JMP into your word processor, then shrink the picture and print it on a laser printer, some points in plots are distorted and show as rectangles instead of squares; smooth curves don't always have smooth joints. This is a problem that JMP has had from Day 1. Why didn't we fix it? Well the problem is that the MacOS™ Quickdraw does its shrinking with integer coordinates that are 72 dots-per-inch. Whether the square stays a square, or turns into a tall or wide rectangle depends on if a point's coordinates have similar remainders when scaled.

We could have cured that by going to PostScript®-customized graphics but there were big drawbacks. If you had a plot with thousands of points the picture would be huge. We do have some space-efficient custom PostScript for a number of things on the Macintosh including dashed lines,

rotated text, and in certain cases smoothing joined segments.

We also experimented with forms of copying in which we collect the drawing into a bitmap that is four or eight times the original size, and then shrink it into the picture. When it draws on the laser printer it comes out great. The problem is that each picture might use several megabytes. We also considered using the new graphics interface, called Quickdraw GX, but it is something that not all users install.

On Windows, there are analogous problems. For example, when we do rotated text on the side of a plot, it comes out looking funny when pasted into a document. That is because it is done as a rotated bitmap at 300 dots per inch. It looks fine on a laser printer, but on a 72 dot per inch screen, too many pixels get fuzzed in when it displays the high resolution bitmap at the lower resolution.

We tried it both ways under Windows: you have to suffer in one place or the other and we made the call that the printed output was what really counted. In JMP version 4 we will use a different imaging model that will resolve both of these issues more directly. We will make the effort, and most of the effort we make might not show up in anything you notice immediately except in the absence of frustration.

A general frustration concerns scrolling. JMP, unlike most statistical products, does a lot of interaction though hypertext-like popups in the report itself. We have two places where we put popups – on the lower left beside the horizontal scroll bar (stationary icons), and within the report. Popups within a report move with scrolling but stationary icons don't. Each one has a big problem.

- The problem with scrolling controls is that they go off screen and out of sight. To click on them you have to scroll back to find them. Sometimes when the graph is in a small window and popup icons are below the graph you don't even notice them unless you happen to scroll down.

A few releases ago, we noticed that if you asked for some options from a popup, for example to test normality in Distribution, you didn't see any change to the screen because the new report was below the bottom of the window. Now when you request additional reports the window automatically scrolls down to where the report was added. But if you want another report you have to scroll back up to where you were in order to ask for it.

- The stationary icons have a different problem. They work fine when

the report is all about one analysis, but they don't work well when there are several kinds of reports in the same window. For example, Fit Y by X produces four different kinds of output, so the stationary popup menu can't have anything in it because the commands wouldn't be right for all four types of analysis.

We are committed to solving all these problems in Version 4.

I believe that the software industry has much work to do in fixing features that it already has. As users, we get used to things that are badly designed. So it is a challenge to both identify the problems, as well as find new designs to solve them.



In April over 40 million households saw JMP Statistical Discovery Software featured on the nationally syndicated television program, "Technology Today." JMP was selected for the program because of its wide acceptance by engineers as a tool to gain competitive advantage. The program was developed and created by Global Solutions Network. If you missed the broadcast you can visit the Technology Today web site

[<http://www.gsnetwork.com/>](http://www.gsnetwork.com/)

where the feature has been adapted for the internet.

## JMP SERVICES UPDATE

### **Professional Services Division SAS Institute Inc.**

SAS Institute's Professional Services Division is pleased to offer the following JMP Training courses, which will be held publicly in SAS Training centers across the US during 1997.

- **Interactive Data Analysis Using JMP Software**
- **Categorical Data Analysis Using JMP Software**
- **Design and Analysis of Experiments Using JMP Software**
- **Statistical Data Exploration Using JMP Software**
- **Advanced Design and Analysis of Experiments Using JMP Software**
- **Multivariate Statistical Methods Using JMP Software**
- **ANOVA and Regression Methods Using JMP Software**
- **Statistical Quality Control Using JMP Software**

For information about specific locations and dates, call  
919-677-8000 ext 7205.

If you need personalized instruction, Institute staff can come to your location to provide onsite training. You can customize training by choosing segments to meet your specific needs. All instructor-based courses include computer workshops. To schedule JMP onsite training, technical service, or consulting, call JMP Training at  
919-677-8000 ext 7312.

## THERE AND BACK AGAIN:

### A Data Table's Tale (or, Moving Your Data From One Computer To Another) Part I: Floppy Disks

by Michael Hecht  
SAS Institute Inc.

I like my Mac. I prefer to use it for all my work, including the work I do on JMP. However, sometimes I have to use a data table that was created with the Microsoft Windows version of JMP or send a data table from my Mac to a coworker who prefers Windows.

#### JMP Data Tables

We designed JMP to use the same file format for data tables on both the Macintosh and Windows, so transfer between them is quite straightforward. However, there are a few 'gotchas' involved and they all concern how you convince the other system that the file you're moving is really a JMP data table.

The mechanism for this is different on each system. Windows uses the method of the *file extension* — a three-letter code joined to the end of the file name with a period. The file extension defines the contents of the file and, in most cases, what application should be used to open it. For JMP data tables we chose the three-letter code "JMP" because it seemed appropriate. So a JMP data table might be named "BIGCLASS.JMP" on Windows.

With Windows 95, the file name is not restricted to a maximum of uppercase characters, so you are free to name the file "Big Class.jmp". Note that the letter case of the file extension is not important; but, the file extension must still be there. Windows 95 has additional options that cause it to shield the file extension from you once it's been registered with the system.

Instead of file extensions the Mac uses two attributes called *file type* and *file creator*.

- The *file type* is a special four-letter code that is not a part of the file name. In fact, the file type is a feature of the file that you cannot change. The file type only identifies the contents of a file; not what
- This second attribute is defined by the *file creator* — another four-letter code, which is also inaccessible to you. For JMP, the file creator is "SGP ". Note that I said S-G-P-space. The trailing space is significant.

The history of the JMP data table file type code is long and colored, and filled with many wild and conflicting legends. Some say that the developer responsible decided that these files Should Give Predictable Data, and thus chose the fabled letters. Others say that the developer was told by SAS Security to "Stop Getting Pizza Delivered" while working late into

the night, and chose this code as a reminder. But I personally believe that the code is an acronym for the materials used to construct the very first data table, Stellar Globules of Pinkish Dust. Whatever the origin, on a Mac a file with a type code of SGPD represents a JMP data table.

### Moving Tables Via SneakerNet: PC to Mac

The easiest way to get the data from here to there is to put it on a floppy. However, the thing to remember here is that Macs are more accommodating than PCs, so you should use the PC floppy disk format. All PCs use this format, so if you're starting on the PC you just copy the file on the disk and walk it over to the Mac. The only problem is that the file type will not be set properly. There are two ways around this problem. If you're only confronted with the problem once a year or less, you may opt for the first solution. If you do this on a daily basis, I'm sure you'll prefer the second solution.

- 1) Solution number one is to launch JMP, then choose the **Open** command, navigate to the floppy or the folder containing the file, and turn on the **Show all files** check box in the Open dialog. Presto! Your data table appears in the list of files ready to be opened.
- 2) Solution number two is to open the *PC Exchange* Macintosh control panel (see **Figure A** next page)

which lets you map PC file extensions to Macintosh file types. To do this:

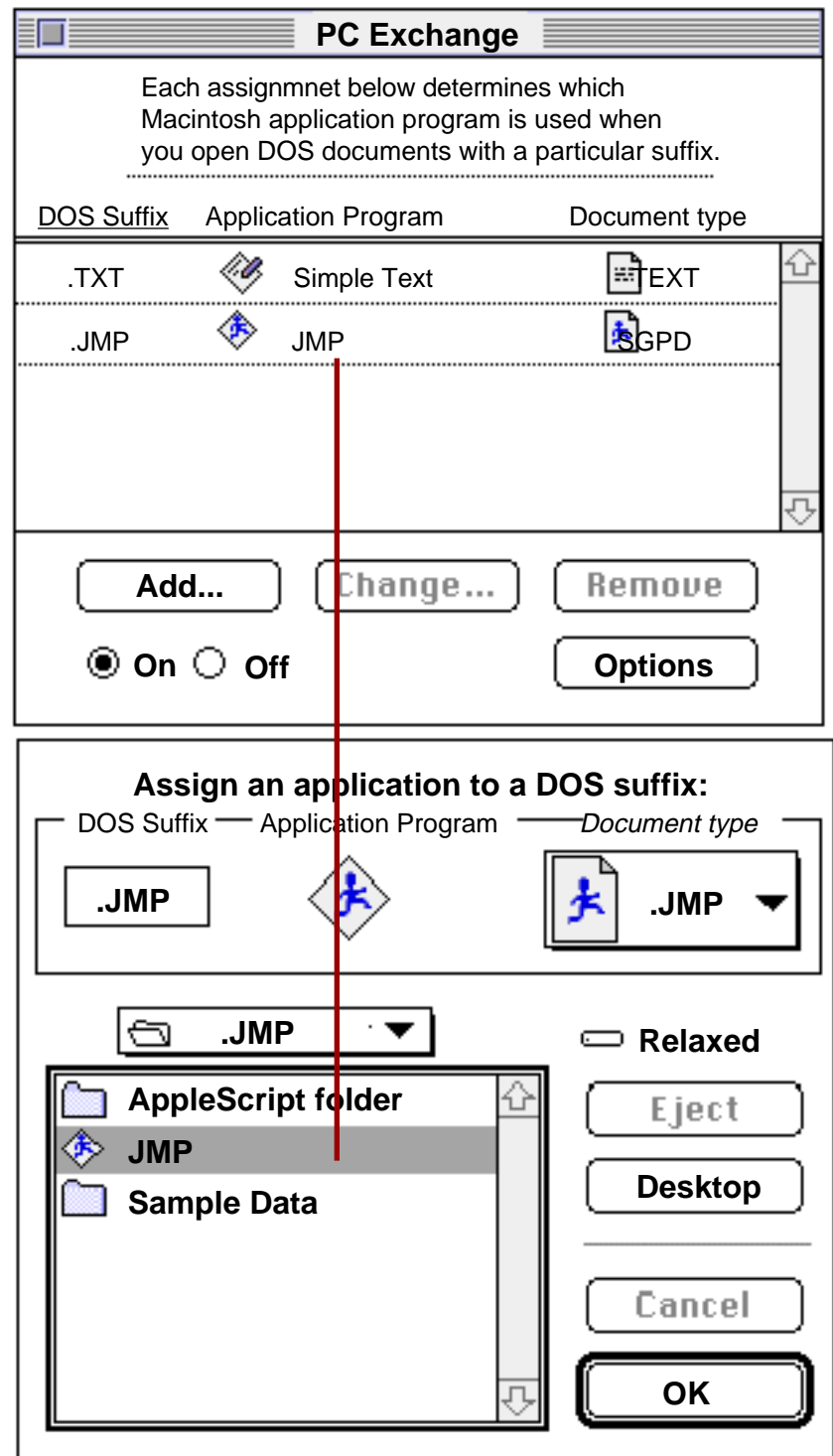
- Click **Add** and in the resulting dialog type ".JMP" into the **DOS Suffix** field.
- Navigate to your JMP application and select it (don't double-click it). The JMP application icon appears under **Application Program** and the **Document Type** popup becomes enabled.
- Click on it and choose "SGPD", which should have the icon of a JMP data table. Click **OK** and you're done! From now on, any file on a PC floppy that ends with ".JMP" automatically gets the JMP data table icon. You can double-click it to launch JMP, just like you can with a data table you created with JMP on your Mac.

### Mac to PC

If you're going the other way (from Mac to PC), you'll have to format the disk so that the PC thinks it's something other than pinkish space dust. To do this, use a high density floppy, which has the stylized "HD" logo and the extra hole. Unlock it, insert it into the Mac, select it, and choose the Finder's **Erase Disk** command. On the dialog that appears choose **DOS 1.4 MB** from the Format popup. You might need to give the disk a new name to suit the rigidity of the PC file system. Click **Erase** and go get a tall glass of fruit juice. When

you back, the disk should have a cute little “PC” embossed on its icon. Just drag your data table onto it and you’re set. Sneaker the disk over to your PC and insert it. From JMP, choose **Open** (or click that snazzy toolbar button) and navigate to the floppy. It’s likely that you’ll need to choose **All files (\*.\*)** from the **Files of type** popup. Even then, you might not recognize our file by name. If your Mac file name won’t fit neatly into the DOS 8.3 convention, the Mac runs it through a meat grinder to make it fit. For example, after copying the data table BIRTH-DEATH SUBSET to a PC floppy the Mac helpfully renamed it to !BIRTH-D.EAT. Lovely. Why it can’t use “.JMP” for an extension, like it knows it should from the PC Exchange control panel, I know not. Oh well, it seems even the Mac has room for improvement. You can see this name mangling on the Mac, too. Just select a file on a PC floppy and **Get Info** for it. Then OPTION-click on the

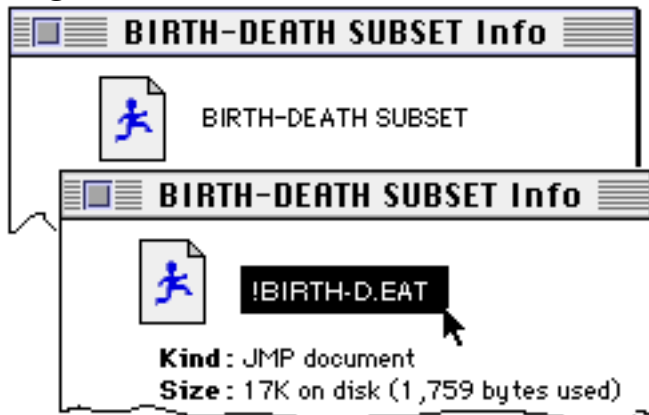
**Figure A** PC Exchange Control Panel



file name in the Get Info window to see what the file name will look like when you plug the disk into a PC.

The example in **Figure B** shows what the Mac will do to modify the data table name Birth-Death Subset ...go figure.

**Figure B** Click in Table Info



The bottom line here is that if you're moving files back and forth, you should put ".JMP" on the end of the file name.



## THERE AND BACK AGAIN

### Part II: Sharing Data Tables on a Real Network

by **Michael Hecht**  
**SAS Institute Inc.**

When your data sharing needs outgrow SneakerNet, you will want to use something a bit more substantial for transferring files, like a file server. Here at SAS Institute we use a Novell NetWare server, which runs on a PC. This server, bless its soul, comes with an extension called *NetWare for Macintosh* that makes it speak the AppleShare protocol. That means our Macs can connect to it

through the Chooser as though it were a real AppleShare File Server. You may have a similar setup at your site. Unfortunately, the DOS file extension to Macintosh file type mappings you set up in PC Exchange for floppy disks (see **There and Back: Part I** in this issue) have no bearing on files placed on servers. Therefore, you are forced to follow these two rules:

- 1) Always put the ".JMP" file extension on JMP data tables that reside on a server.
- 2) On the Mac, you must use the **Show all files** check box to access JMP data tables on the server.

### Tables Across The Internet

If you're connected to the Internet you might want to use it to send your JMP data out into the world. In the realm of the Internet, files are not identified by file extensions or even by file type codes. Instead, yet another standard called MIME is used. MIME stands for the 'Multipurpose Internet Mail Extensions' — Really, I'm not kidding! It's a standard for tagging data which originated as a way to send e-mail that contained styled text, graphics, sound, movies, or other enclosures. The MIME standard has since been conscripted for use with FTP and the World Wide Web.

MIME can define many data types, each of which is assigned a descriptor of the form *type/subtype*. For example, plain old text uses the descriptor

text/plain. Text with embedded markup codes for styles and formatting uses the descriptor text/enriched. In general, the main type text means this data is something a human can read. Some other main types are:

- image for graphics
- audio for noises
- video for moving pictures
- multipart for multiple representations of the same data or data that consists of a conglomeration of different types
- application, for everything else.

The most generic MIME type is application/octet-stream, which means the data is a bunch of bytes for which no interpretation is known or imagined. When received, data of this type is typically just saved to disk. Also, if the MIME type is not recognized, it is usually treated as application/octet-stream.

If you don't tell your computer what MIME type to use when transmitting a JMP data table across the Internet, it is forced to assume that application/octet-stream must be used. The other end then gets untyped data containing the bytes you sent. If the other end is a PC, that's cool—so long as the file it's dumped into has a ".JMP" extension. If the other end is a Mac, that's cool too—but be prepared to use the **Show all files** check box when opening the data table.

A more elegant method is to tag the data as a JMP data table. To do this,

you simply invent a mutually agreed upon MIME type specifically for that purpose. In keeping with the spirit and the specifications of MIME, you should use

application/x-jmp-data

You use the main type application because a JMP data table is not text, image, audio, video, or multipart data. If the receiving application program doesn't recognize the subtype, the main type application tells it to treat the data as application/octet-stream, which is still reasonable. The sub-type for JMP starts with an x- , which means an unregistered, private, or experimental data type. We put jmp- in there because the MIME specification recommends that your data tag include the name of the intended application program.

We could leave it at that, but just in case we ever want to send some other JMP file type (should one ever exist) we put data on the end to identify this as a data table.

Now that we've agreed on a MIME type, how do we tell our computer about it? On the Mac, you use this handy system extension named *Internet Config*. Internet Config manages the preference settings that are commonly used by your various Internet tools, such as your name and e-mail address. In particular, it keeps up with the mappings of MIME types to Macintosh file *type/creator codes*, described in **Part I**.

Here's how to define a mapping from application/x-jmp-data to actual JMP data tables. Launch Internet Config and proceed as shown in **Figure A**.

On Windows, there's no such beast as Internet Config. Each Internet tool you use must be separately configured. Here's how to configure Eudora, a popular e-mail client. Other tools work similarly. Eudora's MIME mappings are found in its *dot-inny* file, which has nothing whatsoever to do with belly buttons—it is a text file named EUDORA.INI.

If you're running Eudora, quit it first; then open EUDORA.INI in a text editor. This file has a section that begins with the line

[Mappings]

and is followed by a countably infinite number of one-line entries that look vaguely like this:

```
both=rtf,MSWD,TEXT,application,rtf
```

Save the .INI file and launch Eudora. When you send someone a JMP data table Eudora consults its MIME mappings and sees that files with the ".JMP" extension are to be tagged as "application/x-jmp-data". If you're sending it to my Mac, my e-mail client (which happens to be Apple's Cyberdog) recognizes this MIME tag as belonging to files with the type/creator pair of "SGPD/SGP". Almost like magic, the enclosure appears to me as a JMP data table. This tells Eudora that both incoming and outgoing data, say an enclosure

in an e-mail message, with a MIME type of application/rtf should be given the ".RTF" extension. The "MSWD" and "TEXT" items happen to be Macintosh file creator and type codes! Why? Because Eudora was ported from the Macintosh to Windows and it retains its Mac interoperability, even on a PC. So, the thing to do is add a line to this file that looks like this:

```
both=jump,SGP ,SGPD,application,  
x-jmp-data
```

## Conclusion

In a perfect world, the issues discussed here would not even exist. It shouldn't matter what I name my file, and I should never need to know about such things as type/creator codes. Unfortunately, life is filled with niggling details such as these. However, with a bit of configuration work we can come close to perfection. Apple could help by picking a cross-platform file type to file extension convention and sticking with it in all situations—removable media, file servers, and networked applications. The Internet is showing us the way with its MIME standard, and Internet Config is a reasonable means of integrating MIME support into the system. Now if only PC Exchange and file servers would make use of it!

Until we achieve cybernirvana, let me leave you with the following rules:

- 1) JMP data tables with legs should be named using the ".JMP" extension.

2) Networking tools should use the MIME type "application/x-jmp-data" for JMP data tables.

For more information see the User's reference material for:

*Cyberdog* <<http://cyberdog.apple.com/>>

*Eudora* <<http://www.eudora.com/>>

*Internet Config* <<http://www.quinn.echidna.id.au/Quinn/Config/>>

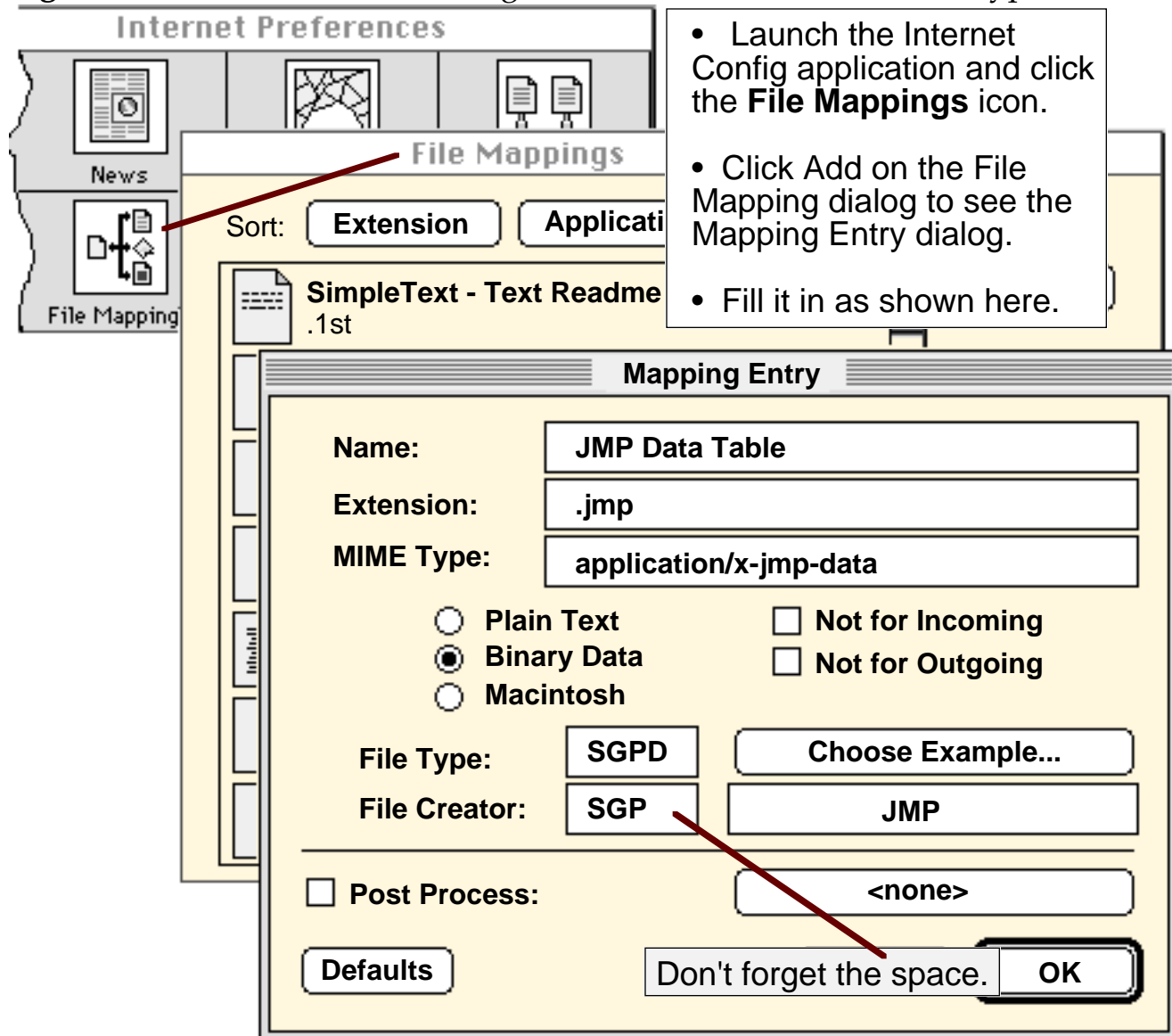
*MIME* <<http://sunsite.auc.dk.RFC/rfc/rfc2045.html>>

*PC Exchange* <<http://www.apple.com.au/Pub/Datasheets/PCEX2.html>>

Share and enjoy!



**Figure A** Use the Internet Config Extension to Define a MIME Type



## CALCULATING FITTED VALUES FROM A Y BY X SPLINE FIT

**By Annette Sanders  
SAS Institute Inc.**

To fit a spline relationship to two numeric variables you first use the Fit Y by X command from the Analyze menu. This example uses the BIG CLASS data table from the SAMPLE data, with height as Y and weight as X. The Fit Y by X platform begins by showing a scatterplot of the X, Y data points. The Fitting popup menu beneath the plot accesses the fitting options shown in **Figure A**.

The Fit Spline option fits a smoothing spline using a smoothing parameter you specify. The spline is displayed on the X, Y scatterplot (weight and height in this example) and a table appears showing the R-Square and Sum of Squares Error.

The Save Predicteds popup menu command for the spline fit creates a new data table column and saves predicted values for each row. However, the spline does not have a prediction equation so you cannot find *fitted values* (predicted values) for data points that are not in the data table.

In order to compute fitted values you need more information about the spline fit; specifically, the coefficients of the spline's prediction formula. To get these you first use the Output Coef Table popup menu command on the

Spline Fit table, which creates a new JMP table and saves a set of spline coefficients for each unique value of the X variable. **Figure B** shows the coefficients table for the height by weight example.

Note that although the BIGCLASS data table has 40 observations the weight (X) variable has only 29 unique values (29 rows) listed in the O column. These values are called *knot points*. The knot points are points at which third degree polynomials are spliced together. The polynomial values and their first derivatives agree at these points, which results in a continuous and smooth curve.

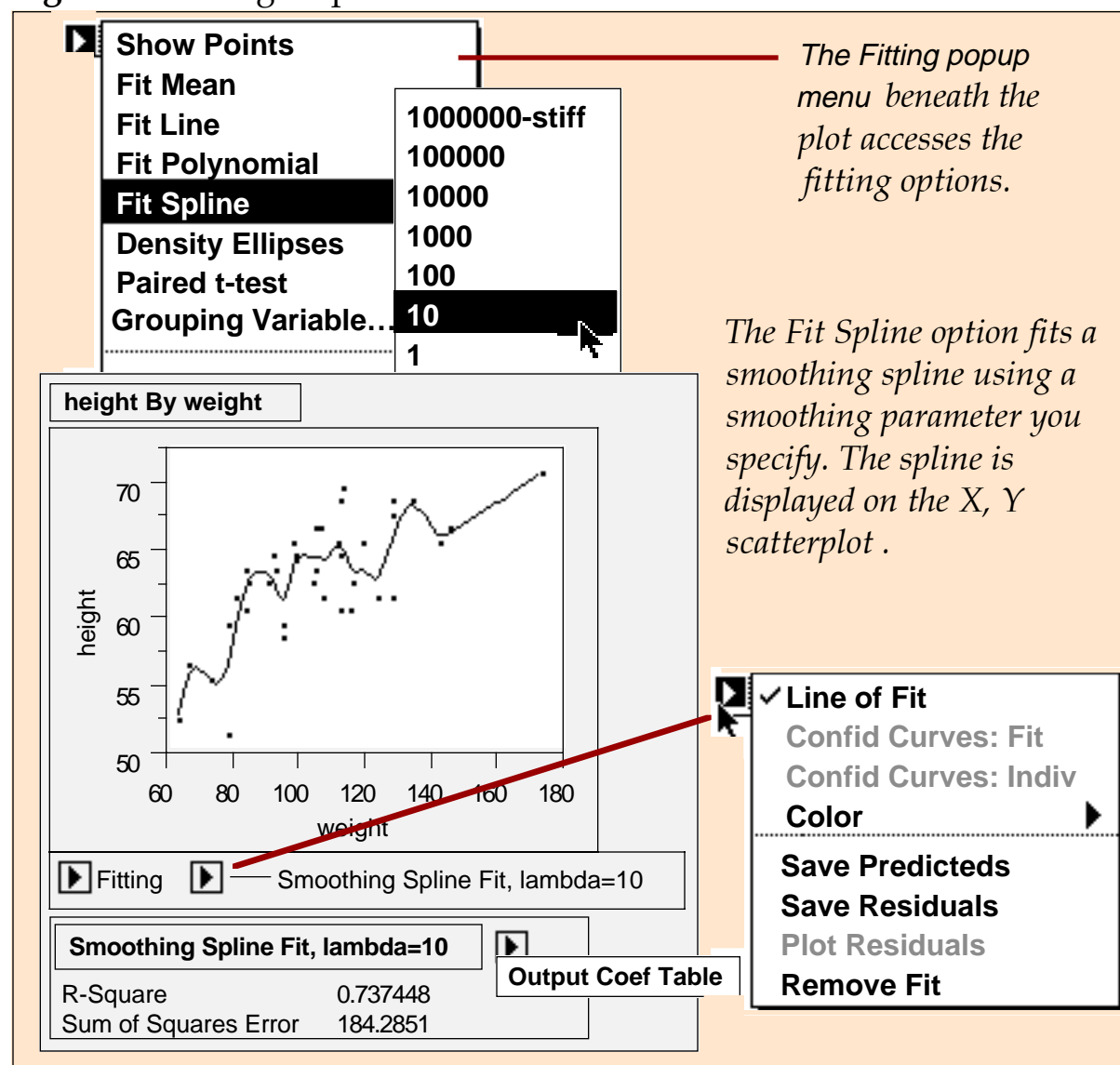
You can use this new table and the JMP calculator to build a formula that will compute fitted values for any X value. The way you do this is to generate the X values you want, and create an index that associates each X with the appropriate row of spline coefficients for that X value.

The fitted values are for any number you choose of equally spaced X values that fall within the range for which there are spline coefficients (the min and max of the weight variable in this example).

Follow these steps to generate the fitted values.

- 1) In the spline coefficients table, add the number of rows so the total rows is the number of fitted

**Figure A** Fitting a Spline Curve



**Figure B** Table of Spline Coefficients

29 Rows	5 Cols				
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	O	A	B	C	D
1	64	52.52	1.05	0.00	-0.02
2	67	55.21	0.59	-0.15	0.01
3	74	54.86	-0.25	0.03	0.01
4	79	56.17	1.11	0.24	-0.06
27	142	65.48	-0.08	0.08	-0.01
28	145	65.73	0.16	-0.00	0.00
29	172	70.00	0.00	0.00	0.00

The **O** column contains unique values of the **weight** variable. The columns **A**, **B**, **C**, and **D** are the spline coefficients associated with each unique **weight** value. They are the constant, linear, quadratic, and cubic coefficients respectively.

points you want to obtain. In this example we added 71 rows (to the 29) in order to obtain 100 fitted points. You can create as many of these new X values as you want at any time by increasing the number of rows in the data table.

- 2) Use Tables→New Column to create a new column of X values (call it newX). In the New Column dialog select Formula as the Data Source.
- 3) Use the calculator Count function with Quantile functions as arguments to generate equally spaced values from the minimum to maximum of the knot points:  

$$\text{count (from quantile}_0\text{O to quantile}_1\text{O in n steps, 1 time)}$$

Recall that Quantile<sub>0</sub> of O is its minimum, Quantile<sub>1</sub> is its maximum, and n is the total number of table rows.

Note that the newX values are in the coefficients data table but their ordering and physical relation to the coefficients are not relevant; you are storing unrelated pieces of information within the same row of the data table. It is likely that the coefficients needed to compute the fitted value for a specific newX value will not come from its own row. The row of coefficients needed is identified by an index variable you need to create, as described next.

- 4) Create an index column (call it index) with calculated values

index) with calculated values between 1 and 29 that point to each unique weight value. This index associates each newX value with the appropriate row of coefficients:

- Use the Sum function to increment the value of index by 1 each time the value of newX is greater than or equal to the next consecutive knot point:

$$\sum_{j=1}^n (O_j \leq \text{newX}_i \text{ and } O_j \neq \bullet)$$

- When the value of newX is less than the next consecutive knot point nothing more is added – that index value identifies the row of coefficients needed to calculate the fitted value.

- 5) Now you can use the JMP calculator to compute fitted height values from newX using the appropriate coefficients. The formula to compute the fitted value from newX with the index value *i* is:

$$\text{fitted value}_i = A_i + d \cdot B_i + d^2 \cdot C_i + d^3 \cdot D_i$$

where d is the difference between newX and the observed weight associated with it by its index value. An efficient way to construct this formula is with a temporary variable and an Assignment function. You construct the formula like this:

- Create a new column called fittedHt.
- Use the calculator and select New Variable from the Variables



## THE RUNS TEST: Nonparametric Testing for Randomness in a Series of Runs

by Annie Dudley  
SAS Institute Inc.

When monitoring process control on a manufacturing line, one problem that occurs is a cyclic run of defective units. By testing for randomness you can identify periodic runs that might not be visible in control charts.

Nonrandomness can occur either with too many or too few runs, where a run is a sequence of like events. The total number of runs in a sample gives an indication of whether the sample is random. If there are few runs, a time trend or grouping of like events due to lack of independence could be occurring. Many runs might indicate some systematic short-period cyclical fluctuation.

As an example, suppose an engineer is monitoring the unit failures off a process line and found the sequence of successes (s) and failures (f) shown in **Figure A**.

To set up a runs test, you first identify the runs. The success and failure events are recorded in a JMP table (event). You identify the number of runs by tagging the first occurrence of each run in the series; that is, tag the row whenever the type of event changes. To do this:

- Create a new column called *r* and change the data source to formula.
- Enter the formula as shown in **Figure A** to assign a 1 to the beginning of each new run.

The total number of runs in the sample is simply the sum of the *r* column.

**Figure A** Event Sequence

2 Cols		<input checked="" type="checkbox"/> N <input type="checkbox"/>	<input checked="" type="checkbox"/> C <input type="checkbox"/>
50 Rows		event	r
1	s		1
2	f		1
3	s		1
4	f		1
5	s		1
6	s		•
7	s		•
8	f		1
9	f		•
10	s		1

In a data table, record sequence of events in the order they occurred.

s f s f s s s f f s f s f s s s s f s f s f s  
s f f f s f s f s f s s f s s f s s s s f s f s s

Tag the beginning of each run.

$$\begin{cases} 1, & \text{if } (event_i \neq event_{i-1}) \\ \square, & \text{otherwise} \end{cases}$$

As a rule of thumb, when either the number of successes or the number of failures is greater than 20, an approximation of the z test can be constructed to test whether there are too many (or too few) runs in a series. This example has 20 failures and 30 successes.

Let  $n_1$  be the number of successes,  $n_2$  be the number of failures,  $n$  be total sample size, and  $r$  be the number of runs (the sum of the  $r$  column). The large sample approximation treats the distribution of  $r$  as normal with

$$\mu_r = \frac{(2 \cdot n_1 \cdot n_2)}{(n_1 + n_2)} \quad \text{and} \quad \sigma_r = \sqrt{\frac{(2 \cdot n_1 \cdot n_2 \cdot (2 \cdot n_1 \cdot n_2 - n_1 - n_2))}{(n_1 + n_2)^2 \cdot (n_1 + n_2 - 1)}}$$

These parameters give the z statistic:









$$z = \frac{r - \mu_r}{\sigma_r}$$

This z statistic formula is bulky but not complicated and can be constructed with the JMP calculator as shown in **Figure B**.

First, create a new column (call it z test) and use the calculator to construct the formula as follows:

- 1) Use Variables in the function browser to create three new temporary variables, called  $n_1$ ,  $n_2$ , and  $r$ .
- 2) Use Assignments from the Conditions functions to assign values to  $n_1$ ,  $n_2$ , and  $r$ .
- 3) Enter the equation for the z-approximation as the results clause.
- 4) The last step is to find the 2-tailed probability associated with the z test value. Create another column (call it p-value) and use the normDist function found in the probabilities functions.

**Figure B** Construction of the z Statistic Formula

		create
		assignment
		clauses
results		
$n_1 \Leftarrow \sum_{j=1}^n (event_j = "s")$		
$n_2 \Leftarrow n - n_1$		assign values to $n_1, n_2$ , and $r$
$r \Leftarrow \sum_{j=1}^n r_j$		
results		
		$n_1 \Leftarrow \sum_{j=1}^n (event_j = "s")$ <p>compute z statistic as result</p> $n_2 \Leftarrow n - n_1$ $r \Leftarrow \sum_{j=1}^n r_j$ $r - \left( \frac{(2 \cdot n_1 \cdot n_2)}{(n_1 + n_2)} + 1 \right)$ $\sqrt{\frac{(2 \cdot n_1 \cdot n_2 \cdot (2 \cdot n_1 \cdot n_2 - n_1 - n_2))}{(n_1 + n_2)^2 \cdot (n_1 + n_2 - 1)}}$

Note that the normDist function returns the probability that a value is less than or equal to its argument, so you use  $1 - \text{normDist}$ , and multiply by 2 to find the desired 2-tailed probability  $2 \cdot (1 - \text{normDist}(|z \text{ test}|))$ . The z test and probability values show as constant columns in the data table as in Figure C).

**Figure C** Runs Test and Probability

4 Cols	N	C	C	C
50 Rows	event	r	z test	p-value
1	s	1	2.979398	0.00288
2	f	1	2.979398	0.00288
3	s	1	2.979398	0.00288
4	f	1	2.979398	0.00288
5	s	1	2.979398	0.00288
6	s	•	2.979398	0.00288
7	s	•	2.979398	0.00288
8	f	1	2.979398	0.00288

You can delete all the rows and save this kind of table as a template. Then whenever you want a large-sample runs test, paste the sequence of events in the event column and the calculations will proceed automatically.

This example is from Siegel (1956), with the sequence being the order of gender in a queue at a theater.

#### References:

Siegel (1956), *Nonparametric Statistics*, McGraw-Hill Book Company, NY

Mendenhall, Scheaffer, and Wackerly (1986), *Mathematical Statistics with Applications*, 3rd Ed., Duxbury Press, CA



## Calculator Corner

by Michael Hecht  
SAS Institute Inc.

### LAGGING BEHIND

The SAS DATA step has a nice feature called *LAG*, which gives you the value a variable had on a previous observation. To do this in JMP you use the Subscript operator on a column. If the subscript's value comes before the current row, you've achieved a lag. JMP's Subscript operator can also subscript rows that come after the current row. This is commonly called a *LEAD*.

### Cumulative Sum

Here is an example that uses a lag variable. Suppose for each row you want to compute the cumulative total of a column called expenses. To do this you create a new column (call it cumulative exp) and give it values with the formula:

$$\text{expenses} + \text{cumulative exp}_{i-1}$$

where  $i$  is the special Terms operator that is the current row number and  $1 \leq i \leq n$ .

What's happening here? Well, for each row, you want the value of cumulative expenses to be the current row's expenses plus the

previous row's cumulative expenses. However, there's a problem condition—for the first row  $i = 1$  and the subscript for cumulative exp becomes 0. The solution is to guard against the border case by using an if clause, as shown in **Figure A**.

### Cumulative Sum by Groups

A more challenging problem is finding a cumulative sum within groups, as well as guarding against missing values anywhere in the expenses column. For example, suppose you have a grouping variable, group, and you want the cumulative values as shown in the data table in **Figure B**.

**Figure A** Accumulate a Total

$$\text{expenses} + \begin{cases} \text{cumulative exp}_{i-1}, & \text{if } i > 1 \\ 0, & \text{otherwise} \end{cases}$$

<input type="checkbox"/> expenses	<input type="checkbox"/> cumulative expenses
1	1
2	5
3	6
4	10
5	15

An efficient formula to do this uses a temporary variable with an assignment condition as follows:

- 1) From the function browser select Variables→New Variable. Name the variable  $e$ .
- 2) From the function browser select Conditions→Assignment to set up

an assignment, which looks like this:

☐  $\leftarrow$  ☐  
results ☐

- 3) Use Conditions→If to give the temporary variable  $e$  the value of zero whenever expenses is missing, and the value of expenses itself when it is not missing:

$e \leftarrow \begin{cases} 0, & \text{if } \text{expenses} = \bullet \\ \text{expenses}, & \text{otherwise} \end{cases}$   
results ☐

- 4) For the assignment's result clause, use another If clause that assigns  $e$  for the 1st row, or when the group changes. Otherwise assign the cumulative,  $e +$  the lagged grouped cum, as shown in **Figure B**.

$e \leftarrow \begin{cases} 0, & \text{if } \text{expenses} = \bullet \\ \text{expenses}, & \text{otherwise} \end{cases}$   
results  $\begin{cases} e, & \text{if } i=1 \text{ or } \text{group} \neq \text{group}_{i-1} \\ e + \text{grouped cum}_{i-1}, & \text{otherwise} \end{cases}$

**Figure B** Grouped Totals

<input type="checkbox"/> expenses	<input type="checkbox"/> group	<input type="checkbox"/> grouped cum
1	A	1
2	A	3
3	B	3
4	B	7
5	B	12
•	B	12
7	C	7



# Tips and Techniques

## SAVING DATA TABLE DISK SPACE

It's easy to calculate how much disk space (bytes) a data table will require.

- 1) First, add together
  - 8 bytes for each numeric variable
  - field width + 1 for each character variable
  - 2 for each row state variable.
- 2) Multiply the total by the number of observations in the table.
- 3) Add 102 bytes overhead for each column.
- 4) To be exact, add in 1 byte for each character used in the Column Info notes and in the Table Info notes.

Under Windows, the number of bytes a table uses shows next to the table name when you select Details from the View menu. To see the number of bytes used by a data table on the Macintosh, click the table icon in the Finder to select it, and use the Get Info command in the File menu.

As an example take the familiar BIG CLASS data table in the sample data folder. It has 40 observations and 5 variables:

- name - width  $12+1 = 13$

- age - 8
- sex - width  $1+1 = 2$
- height - 8
- weight - 8

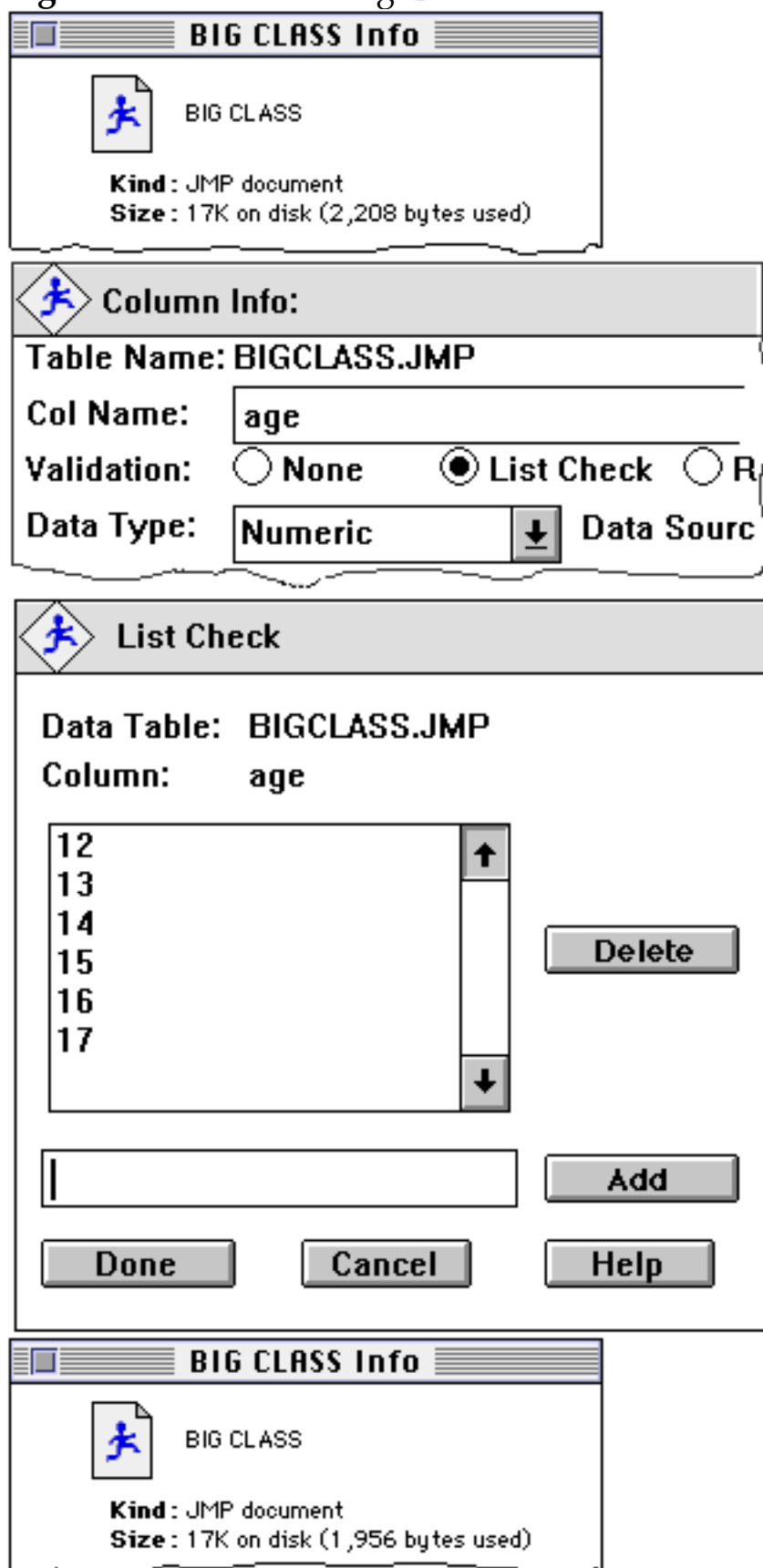
for a total of 39 bytes, giving  $39*40 = 1560$  bytes used by the data. Add in overhead of  $102 \text{ bytes} * 5 \text{ variables} = 510$ , giving a total of  $1560 + 510 = 2070$  bytes used for the table. The Get Info dialog for the BIG CLASS table at the top in **Figure A**, shows 2,208 bytes, which includes the characters used in column and table notes. Windows rounds up and shows 3,000 bytes.

### Using List Check

When you have a variable that has 256 or fewer unique values (either numeric or character) you can use the List Check option in the New Column (or Column Info) dialog. When you click the List Check radio button in the Column Info dialog for an existing column, such as age in this example, the List Check dialog appears and lists the values of the variable as shown in **Figure A**. List Check performs these two functions:

- Validation of data entry is in effect; you can only enter the values listed in the List Check Dialog.
- Data is stored more efficiently; List Check codes each value into a single byte and maps it to the actual value for spreadsheet display and analysis.

**Figure A** Effect of Using List Check



Using List Check on the variable age reduces the number of bytes needed on disk from 2208 to 1956. This is a space reduction of about 11% (see the Get Info dialogs at the top and bottom in **Figure A**). Windows shows a reduction from 3k to 2k.

This may not seem like much of a savings, but consider the effect when the table is very large (say 10,000 observations or more) and there are many codable variables, especially long character variables.

The magnitude of savings became clear to us when we began processing a table with 32,138 rows and 19 columns, which used 4.404 meg of space. Turning List Check on for three character variables of length 2 with 10 values, length 24 with 8 values, and length 35 with 30 values reduced the disk space used to 2.446 meg— a savings of over 44%!

## Enabling Short Numerics

By default, each numeric variable uses 8 bytes; changing the field width only changes the width used for formatting the values in the spreadsheet.

However, a new Preference called **Enable Short Numerics** adds 3 new data types to the Data Type popup menu in the Column Info dialog: Integer 1, Integer 2, and Integer 4. They reduce the number of bytes used to store a numeric variable and can be used when you have these numeric integer values:

- Integer 1 for -126 to +127
- Integer 2 for -32,766 to +32,767
- Integer 4 for -2,147,483,646 to +2,147,483,646

In the BIG CLASS example, using List Check as before, changing height from Numeric to Integer 1, and weight to Integer 2, the file size reduces to 1,440k from the original 2,208. Many survey items have ordinal integer responses that can be stored efficiently with these new Integer data types.



## You are invited to visit and talk with us at these conventions and trade shows

Apr 06-09	Experimental Biology New Orleans, LA
Aug 6-8	MacWorld Boston, MA
Apr 22-24	Quality Expo Chicago, IL
Aug 9-15	ASA Anaheim, CA
May 5-7	ASQC Orlando, FL
Sep 07-11	ACS Fall Las Vegas, NV
May 5-7	Sematech San Antonio, TX
Sep 21-23	SESUG Jacksonville, FL
May 14-17	Interface '97 Houston, TX
Sep 28-30	MWSUG Chicago, IL
Jun 12-15	ICE '97 Minneapolis, MN
Oct. 5-7	NESUG Baltimore, MD
Jun 17-19	PC Expo New York City, NY
Oct. 22-24	WUSS Universal City, CA
Jun 23-25	Drug Information Assoc. Annual conference Montreal, Canada
Nov 9-11	SCSUG Houston, TX
Nov 30-Dec 5	RSNA Chicago, IL



**"The Key isn't  
what you know.  
The key is what  
you can teach  
others and have  
them apply."**

**by Colleen Jenkins  
SAS Institute Inc.**

In November JMP users from across the country attended the first JMP Data Discovery Conference held at SAS Institute Inc.'s corporate headquarters in Cary, NC. This week-long training conference was designed to increase the return on investment that each attendee has made when applying the statistical methods available in JMP to collecting and maintaining their data.

Attendees included industry-leading companies like Procter & Gamble, Motorola, Immunex Corporation, Glaxo-Wellcome, Intel Corporation, Duke Comprehensive Cancer Center, Blue Cross Blue Shield of Florida, The Goodyear Tire and Rubber Company, Dow Chemical Company, Eastman Kodak Company, Hughes Aircraft, and Eli Lilly Company.

Some of the conference objectives were:

- to provide training on key statistical methods and analyses using JMP
- to help client attendees identify methods and strategies to improve their competitive position
- to provide a forum for attendees to interact and share ideas
- to meet with JMP software developers to ask questions and give input about future software development.

The conference kicked off with a keynote speaker, Dr. Tom Little, Director of Engineering Support at Read-Rite Corporation. His presentation, "Ten Keys to Achieving Robust Product and Process Designs," focused on the competitive advantages gained when analytical techniques are used to do more, better and faster! At Read-Rite Corporation Dr. Little is responsible for all SPC applications, use and training of DOE methods, characterization of new products and processes, measurement characterization and control, and wafer fab product engineering.

The objectives of the conference were met through lecture and workshop sessions. Five one-day courses were conducted addressing:

- *categorical data analysis*: investigation of Mosaic plots, frequency tables, odds ratio, the Cochran-Mantel-Haenszel test, logistic regression, and correspondence analysis.
- *ANOVA and regression methods*: evaluation of models with a single continuous response and simple, crossed, or nested categorical and continuous predictors.
- *multivariate statistical methods*: introduction to multivariate modeling, principal components analysis, canonical correlation, and discriminant analysis.
- *advanced design of experiments*: design and analysis of experiments with single or multiple responses including repeated measures, optimization designs, and nonstandard designs, with both fixed and random effects.
- *reliability and survival analysis*: analysis of reliability and survival data, using the Kaplan-Meier method, parametric models, and proportional hazards models.

Other activities included an evening dinner event that gave attendees the opportunity to interact with each other and learn how JMP is used in different corporate environments.

JMP developers and staff participated in breaks and lunches to encourage feedback from customers on needs and wants in future versions of JMP software. Also, round-table lunch discussions hosted by JMP staff focused on specific aspects of the JMP product and its future development.

During 1997 two more JMP Data Discovery Conferences are scheduled to be held at SAS Institute in Cary during the weeks of

**July 15 to July 18**

**October 28 to October 31**

Mark these dates on your calendar as conference enrollment is limited. For more information about the upcoming conferences, or to register, call

919-677-8000 x5005

or send FAX to 919-677-8225



---

Attendees commented,

*"I loved getting to sit around and talk statistics with people who really know what they are talking about. There are a few of us at work that bumble around blindly together, but I've been enlightened this week."*

*"It's obvious that a lot of time and thought went into the courses. It was great fun interacting with such knowledgeable folks!"*

<p><b>EDITOR</b> Ann Lehman</p> <p><b>CONTRIBUTORS</b> Annie Dudley Michael Hecht Colleen Jenkins Ann Lehman John Sall Annette Sanders</p> <p>If you have questions or comments about JMPer Cable write to JMPer Cable SAS Institute Inc. SAS Campus Drive Cary, NC 27513</p>	<p>© Copyright 1997 SAS Institute Inc. All rights reserved.</p> <p>JMPer Cable is sent only to JMP users who are registered with SAS Institute.</p> <p>For more information on JMP, or to order a copy, contact SAS Institute, JMP Sales phone: 919-677-8000 x 5071 FAX: 919-677-8224</p> <p>You can also browse our web site at &lt;<a href="http://www.sas.com/jmp">http://www.sas.com/jmp</a>&gt;</p> <p>SAS, JMPer Cable, and JMP are registered trademarks of SAS Institute Inc. Other brand and product names are registered trademarks or trademarks of their respective companies.</p>
---	--