Paper 232-2012

Getting to the Good Part of Data Analysis: Data Access, Manipulation, and Customization Using JMP®

Audrey Ventura, SAS Institute Inc., Cary, NC

ABSTRACT

Effective data analysis requires easy access to your data no matter what format it comes in. JMP can handle a wide variety of formats. Once the data is in JMP, you can choose from a variety of options to reshape the data with just a few clicks. Finally, customize your data with labels, colors, and data roles so that graphs and charts automatically look the way you want them to. This paper walks through two or three story lines that demonstrate how JMP can easily import, reshape, and customize data (even large datasets) in ways that allow your data to be displayed in vibrant visualizations that will wow your audience.

INTRODUCTION

Tasks such as data cleaning or reshaping data might be up to 80% of the time spent with a particular data set. Data cleaning is the task of managing missing values, adjusting formats, fixing dates and times, adding new columns or formulas, and removing duplicate or erroneous rows. Reshaping data is the task of molding your data into the format that is appropriate for the type of analysis you want to do and might consist of joining multiple data sources, transposing rows and columns, or splitting or stacking columns. Once all this is done, you have finally gotten to the "good part", where you can ask questions of your data.

JMP is an interactive statistical visualization and discovery tool that was developed by SAS. JMP has been available since 1989. The current version of JMP, released in March 2012, is JMP 10. One of the central pieces of JMP is its data table, which is a rich data manipulation tool. JMP makes it easy to get to the "good part" of data analysis.

This paper will summarize the data access capabilities of JMP, and then it will step through three examples starting with raw data to show various ways JMP makes data manipulation easy.

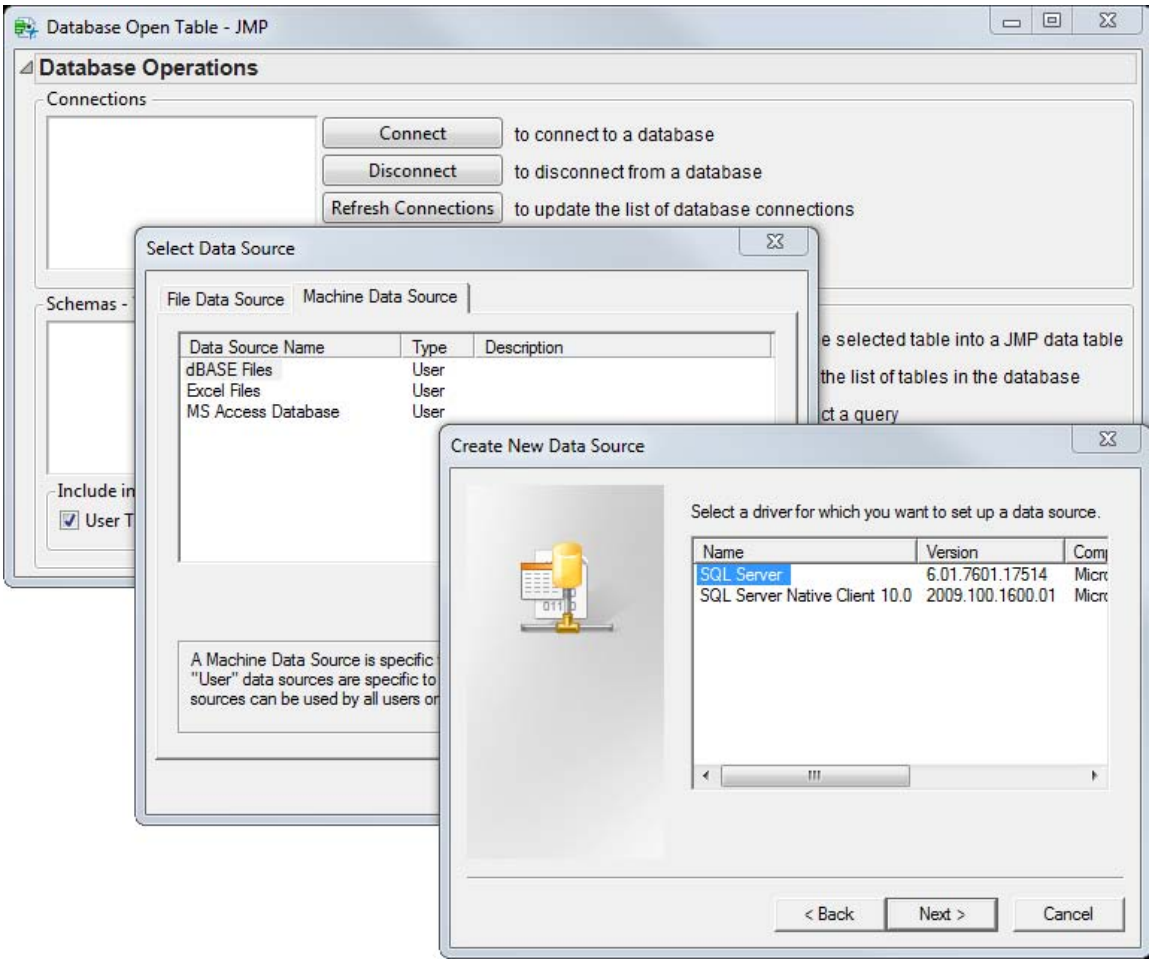
- Data Access
- Example 1 N.C. State Fair Attendance – Internet Open, Value Ordering
- Example 2 U.S. Unemployment – Column Name Add-In, Standardize Column Attributes, Stack
- Example 3 Baseball Attendance – Concatenate, Join, Standardize Column Attributes, Recode

DATA ACCESS

It is worth noting before discussing the examples that JMP can access a wide variety of data types. The examples will showcase only a few ways of accessing data to demonstrate its flexibility. A common way

is copying and pasting data into JMP from another application such as Excel or a text file. JMP can also import many common file types such as Excel, SAS, or text files. JMP 9 introduced mapping capabilities and the ability to read ESRI shapefiles. If you are using a JMP competitor product, such as Minitab or SPSS, it is easy for you to experiment with JMP as you can import those data files directly into JMP. Finally, JMP can connect to most types of databases, since more and more data is stored on database servers.

Figure 1. Database Connections Using JMP



EXAMPLE 1 N.C. STATE FAIR ATTENDANCE DATA

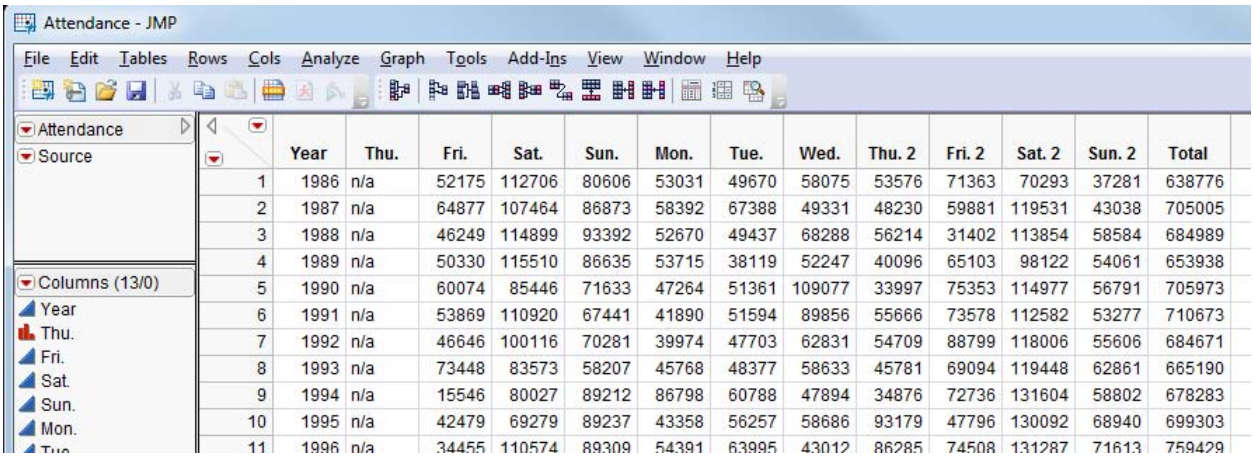
Suppose you are interested in attending the N.C. State Fair, which spans 11 days. Your schedule is flexible and you are interested in attending on a day with low attendance so that the lines will be shorter. A search finds the N.C. State Fair[1] web site, which had attendance data for the last 25 years. You can use **File->Internet Open** in JMP to import the data.

[1] Data from the N.C. State Fair web site is available at http://www.ncstatefair.org/2011/About/Attendance.htm.

Figure 2. Internet Open Dialog Box

The data is imported nicely, with one exception in the first Thursday column, which has been imported as character data. JMP interpreted the "n/a" value as character, but this is easily changed. Also note, the columns named Thu 2, Fri 2, Sat 2 and Sun 2 refer to the second Thursday, second Friday, and so on, since the fair spans two weekends.

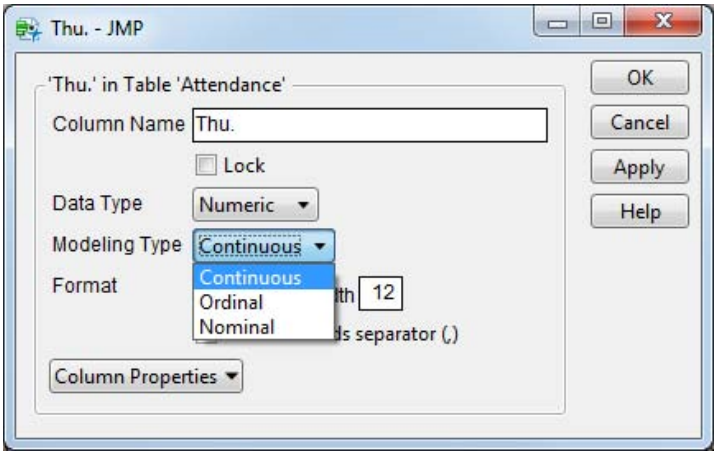
Figure 3. Attendance Data from the N.C. State Fair



	Year	Thu.	Fri.	Sat.	Sun.	Mon.	Tue.	Wed.	Thu. 2	Fri. 2	Sat. 2	Sun. 2	Total
1	1986	n/a	52175	112706	80606	53031	49670	58075	53576	71363	70293	37281	638776
2	1987	n/a	64877	107464	86873	58392	67388	49331	48230	59881	119531	43038	705005
3	1988	n/a	46249	114899	93392	52670	49437	68288	56214	31402	113854	58584	684989
4	1989	n/a	50330	115510	86635	53715	38119	52247	40096	65103	98122	54061	653938
5	1990	n/a	60074	85446	71633	47264	51361	109077	33997	75353	114977	56791	705973
6	1991	n/a	53869	110920	67441	41890	51594	89856	55666	73578	112582	53277	710673
7	1992	n/a	46646	100116	70281	39974	47703	62831	54709	88799	118006	55606	684671
8	1993	n/a	73448	83573	58207	45768	48377	58633	45781	69094	119448	62861	665190
9	1994	n/a	15546	80027	89212	86798	60788	47894	34876	72736	131604	58802	678283
10	1995	n/a	42479	69279	89237	43358	56257	58686	93179	47796	130092	68940	699303
11	1996	n/a	34455	110574	89309	54391	63995	43012	86285	74508	131287	71613	759429

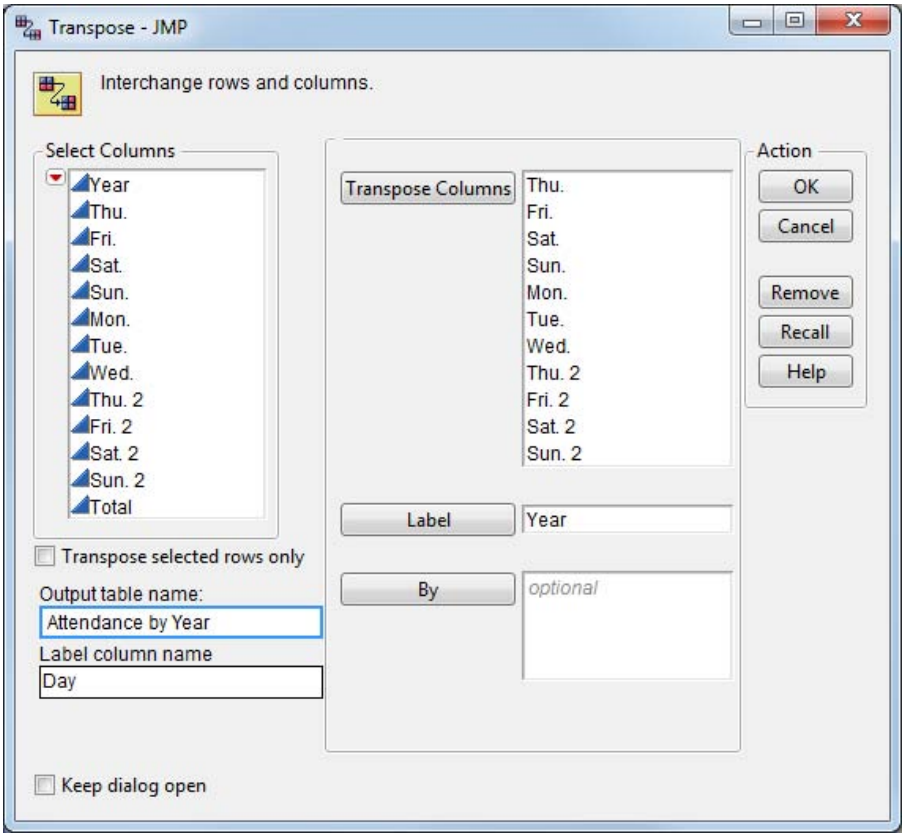
Two actions are needed to analyze attendance per day. First, change Thu to a continuous column through the **Column Info** dialog box.

Figure 4. Change Data Type and Modeling Type in the Column Info Dialog Box



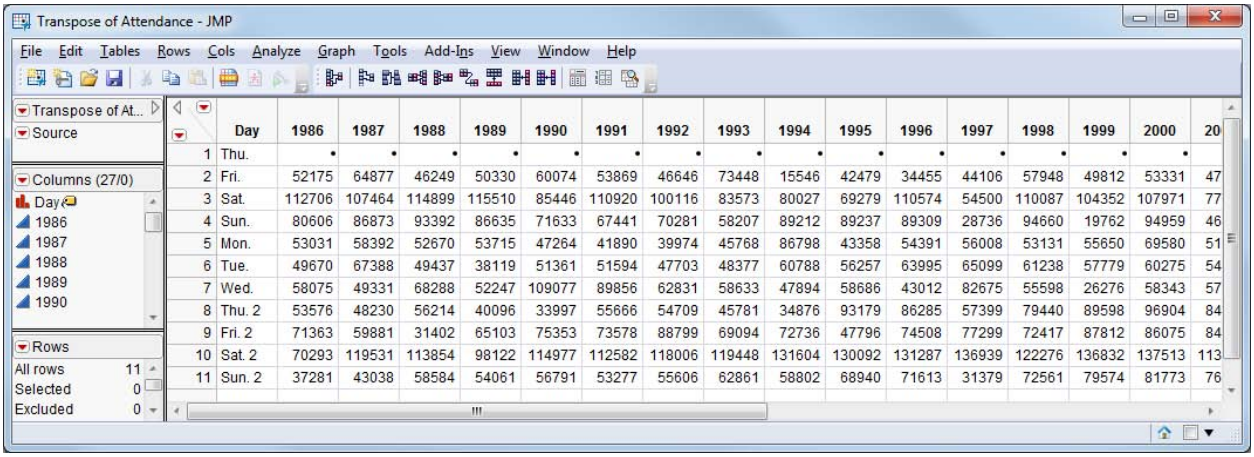
Second, transpose the data so that years are represented in column names and attendance by day is represented on each row. You can access JMP Transpose functionality through **Tables->Transpose**.

Figure 5. The JMP Transpose Dialog Box



Once you press OK, you will have a new data table with the rows and columns transposed.

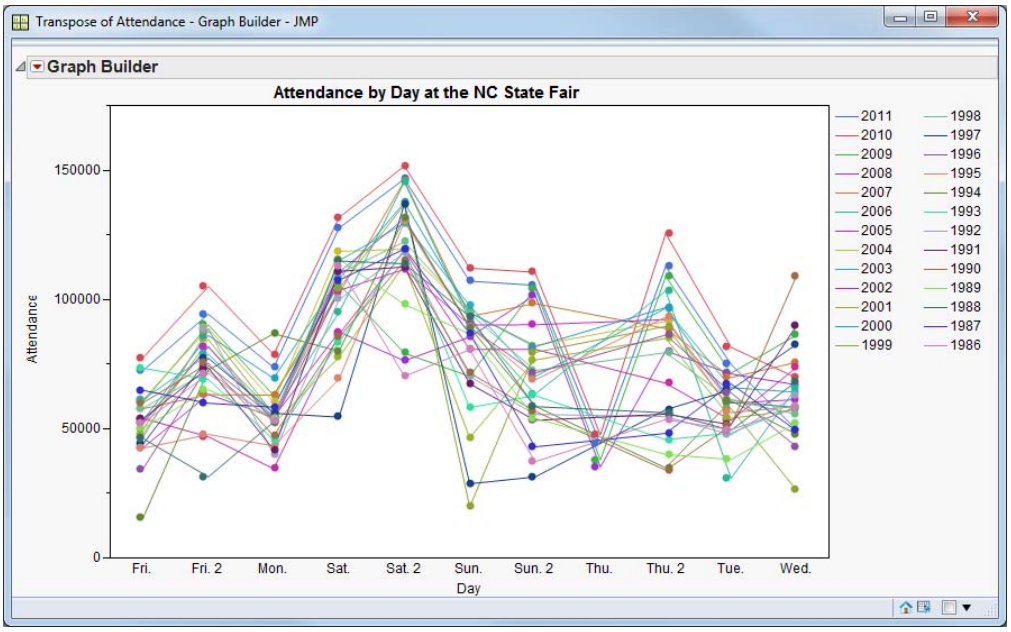
Figure 6. Transposed Attendance Data



	Day	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	20
1	Thu.	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	
2	Fri.	52175	64877	46249	50330	60074	53869	46646	73448	15546	42479	34455	44106	57948	49812	53331	47
3	Sat.	112706	107464	114899	115510	85446	110920	100116	83573	80027	69279	110574	54500	110087	104352	107971	77
4	Sun.	80606	86873	93392	86635	71633	67441	70281	58207	89212	89237	89309	28736	94660	19762	94959	46
5	Mon.	53031	58392	52670	53715	47264	41890	39974	45768	86798	43358	54391	56008	53131	55650	69580	51
6	Tue.	49670	67388	49437	38119	51361	51594	47703	48377	60788	56257	63995	65099	61238	57779	60275	54
7	Wed.	58075	49331	68288	52247	109077	89856	62831	58633	47894	58686	43012	82675	55598	26276	58343	57
8	Thu. 2	53576	48230	56214	40096	33997	55666	54709	45781	34876	93179	86285	57399	79440	89598	96904	84
9	Fri. 2	71363	59881	31402	65103	75353	73578	88799	69094	72736	47796	74508	77299	72417	87812	86075	84
10	Sat. 2	70293	119531	113854	98122	114977	112582	118006	119448	131604	130092	131287	136939	122276	136832	137513	113
11	Sun. 2	37281	43038	58584	54061	56791	53277	55606	62861	58802	68940	71613	31379	72561	79574	81773	76

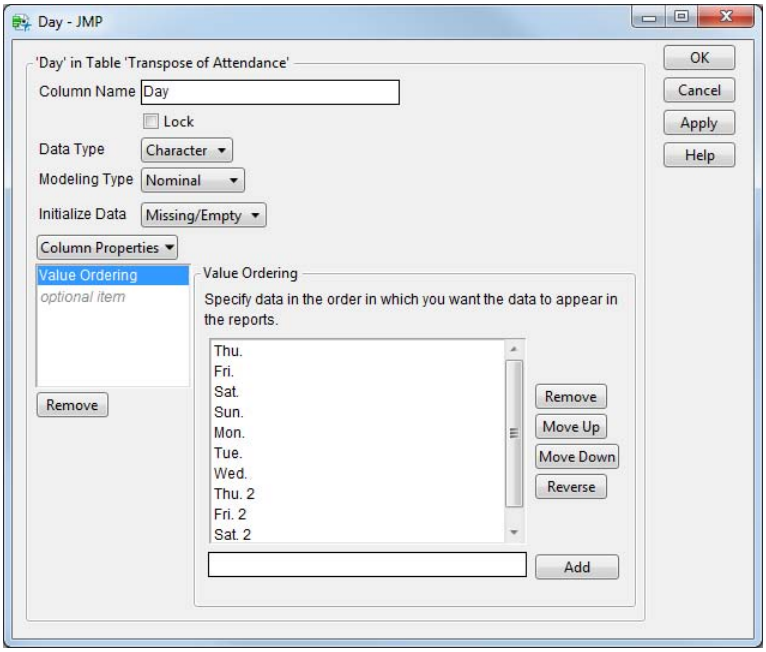
Now you have reached the "good part" and the data is in a format that can be analyzed. Figure 7 shows a line plot of attendance for each day. However, there is one small issue shown in Figure 7: the default in JMP is to alphabetize the days of the week along the x-axis. The graph is not displaying incorrect data, but the days of the week are not in the order you would like to see.

Figure 7. Line Plot of Attendance by Day at the N.C. State Fair (Days Are Alphabetized)



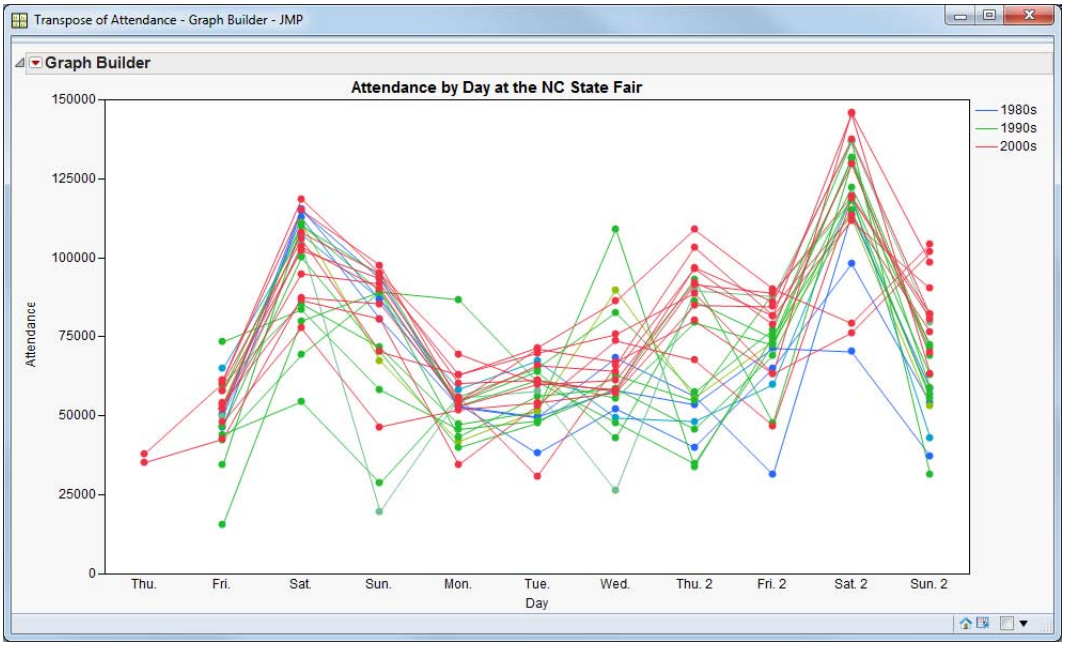
Revisit the data table to add a **Value Ordering** column property to the Day column by way of the **Column Info** dialog box. Value Ordering tells the column what order you would like to display the values.

Figure 8. Example of Adding the Value Ordering Column Property



Once value ordering has been added, now you can recreate the Graph Builder report using the proper order of the days. Additional customization will show you trends of the last three decades.

Figure 9. Line Plot of Attendance by Day at the N.C. State Fair (Days Are in Weekday Order)



You can see from this graph that Saturdays are, not surprisingly, the days with the heaviest attendance. Attendance on the first Thursday is low, but there are also very few data points. Tuesday might be the

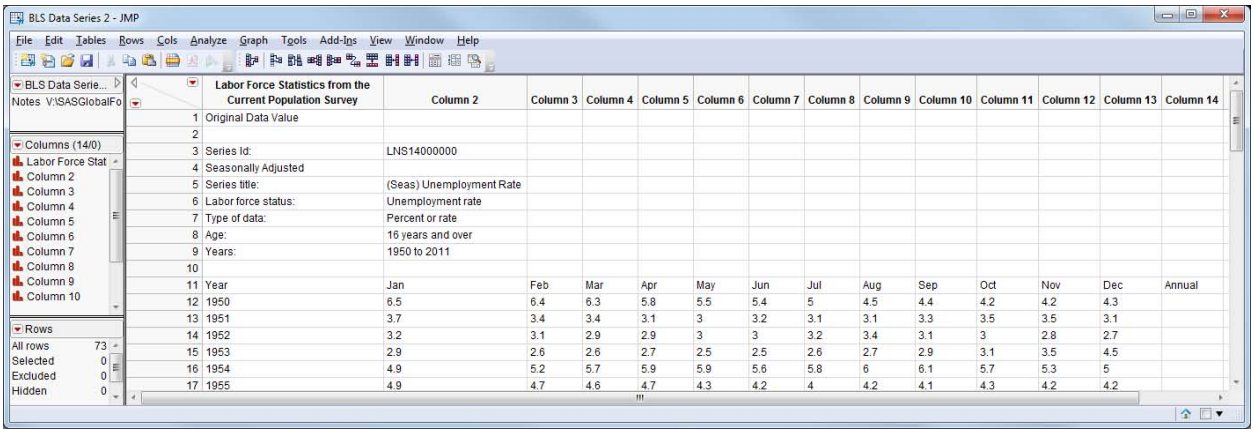
best day to attend the fair based on lowest average attendance in recent years. You could further analyze this data by looking at attendance trends for the last 10 years only.

EXAMPLE 2 U.S. UNEMPLOYMENT DATA (Seasonally Adjusted)

For the next example, suppose you are interested in comparing the recent rates of high unemployment to trends in unemployment rates over the last 60 years. The Bureau of Labor Statistics[2] makes employment data publicly available on their web site. This example uses information from their Current Population Surveys[3] for seasonally adjusted unemployment rates.

The tables available on the Bureau of Labor Statistics web site are not able to be imported using Internet Open in JMP (which you used in Example 1), due to the way queries are processed and returned. However, the tables are easily downloaded into a Microsoft Excel format that can be imported into JMP. The first attempt does not work well, because of 10 rows of notes at the top of the spreadsheet. (See Figure 10.) JMP imports data as columns, and was not expecting the superfluous rows. You have two options. One option would be to close the JMP table and simply copy-and-paste from Excel to JMP. The second option is to proceed and clean up the extra rows in JMP. In this case, you choose the second option.

Figure 10. Unemployment Data Imported into JMP Showing 10 Rows of Notes

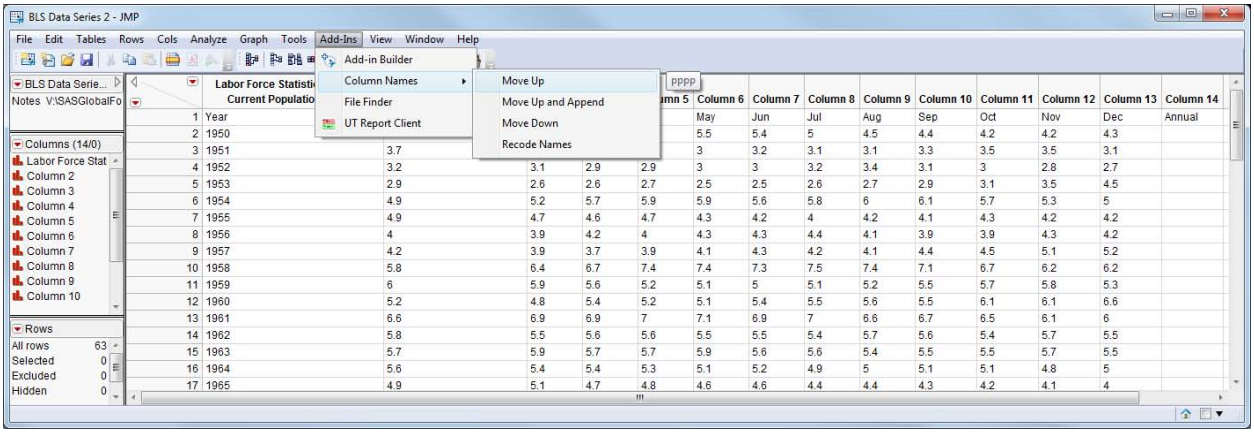


	Labor Force Statistics from the Current Population Survey	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10	Column 11	Column 12	Column 13	Column 14
1	Original Data Value													
2														
3	Series Id:	LNS14000000												
4	Seasonally Adjusted													
5	Series title:	(Seas) Unemployment Rate												
6	Labor force status:	Unemployment rate												
7	Type of data:	Percent or rate												
8	Age:	16 years and over												
9	Years:	1950 to 2011												
10														
11	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Annual
12	1950	6.5	6.4	6.3	5.8	5.5	5.4	5	4.5	4.4	4.2	4.2	4.3	
13	1951	3.7	3.4	3.4	3.1	3	3.2	3.1	3.1	3.3	3.5	3.5	3.1	
14	1952	3.2	3.1	2.9	2.9	3	3	3.2	3.4	3.1	3	2.8	2.7	
15	1953	2.9	2.6	2.6	2.7	2.5	2.5	2.6	2.7	2.9	3.1	3.5	4.5	
16	1954	4.9	5.2	5.7	5.9	5.9	5.6	5.8	6	6.1	5.7	5.3	5	
17	1955	4.9	4.7	4.6	4.7	4.3	4.2	4	4.2	4.1	4.3	4.2	4.2	

JMP can manage the small difficulty posed by the extra rows, as it is an easy thing to delete the 10 rows of notes. Next, you can use the Column Names Add-In[4] to move the header row from row 1 of the data table into the column names area. The Column Names add-in is available on the JMP File Exchange[5].

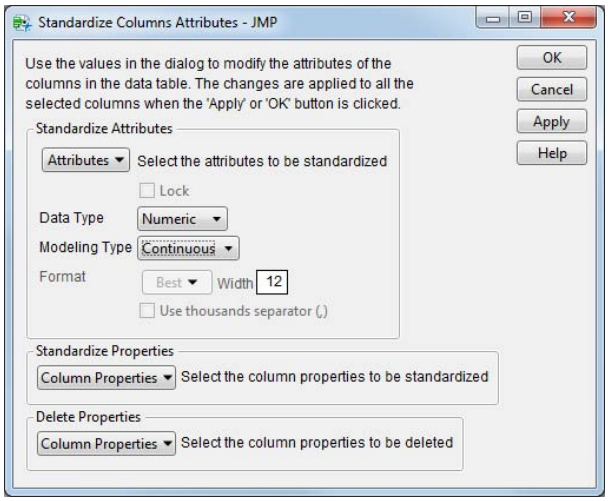
[2] Bureau of Labor Statistics Historical "A" tables are available at http://www.bls.gov/webapps/legacy/cpsatab1.htm.
[3] Current Population Surveys from BLS are available at http://www.bls.gov/cps/#data.
[4] Column Names Add-In is accessible on the JMP File Exchange at http://support.sas.com/demosdownloads/downarea_t4.jsp?productID=110491&jmpflag=Y.
[5] JMP File Exchange can be accessed from http://www.jmp.com/community/.

Figure 11. Using the Column Names Add-In



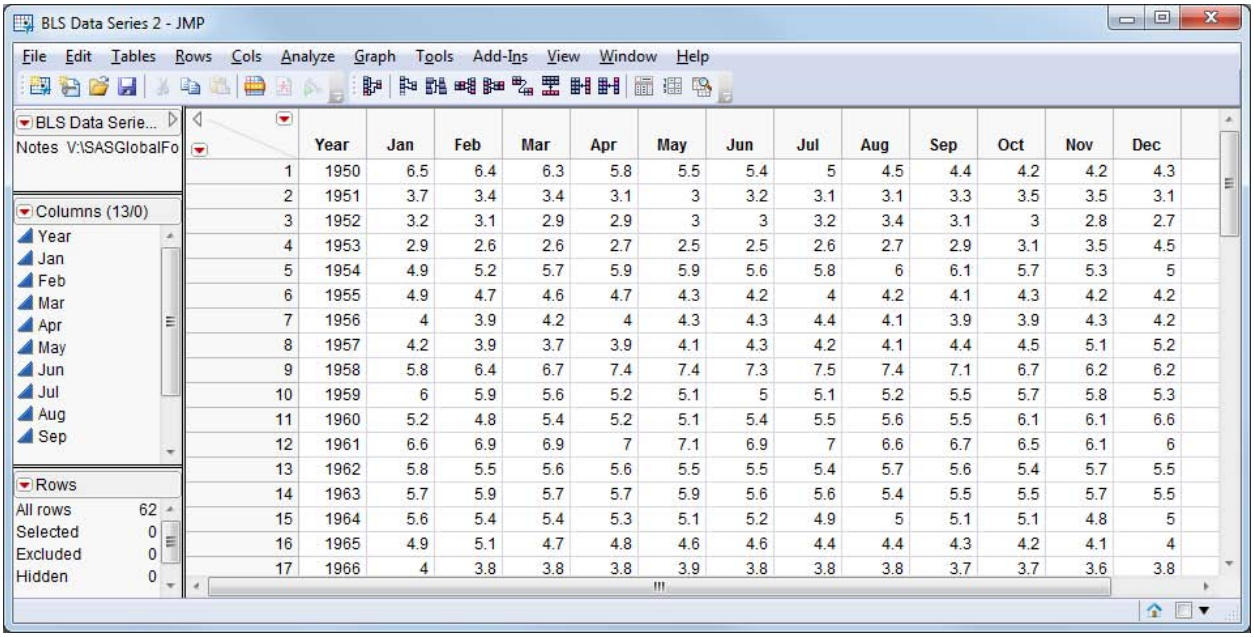
The Annual column is empty for seasonally adjusted data, so it can be deleted. There is also still a small issue to correct in that all the data is considered character data. Use Standardize Attributes functionality in JMP to change the properties of many columns at once. In this case change all the columns to continuous using **Cols->Standardize Attributes**. In this dialog box, change the data type to numeric and change the modeling type to continuous.

Figure 12. Standardize Column Attributes Dialog Box



Now the data table is ready to be analyzed.

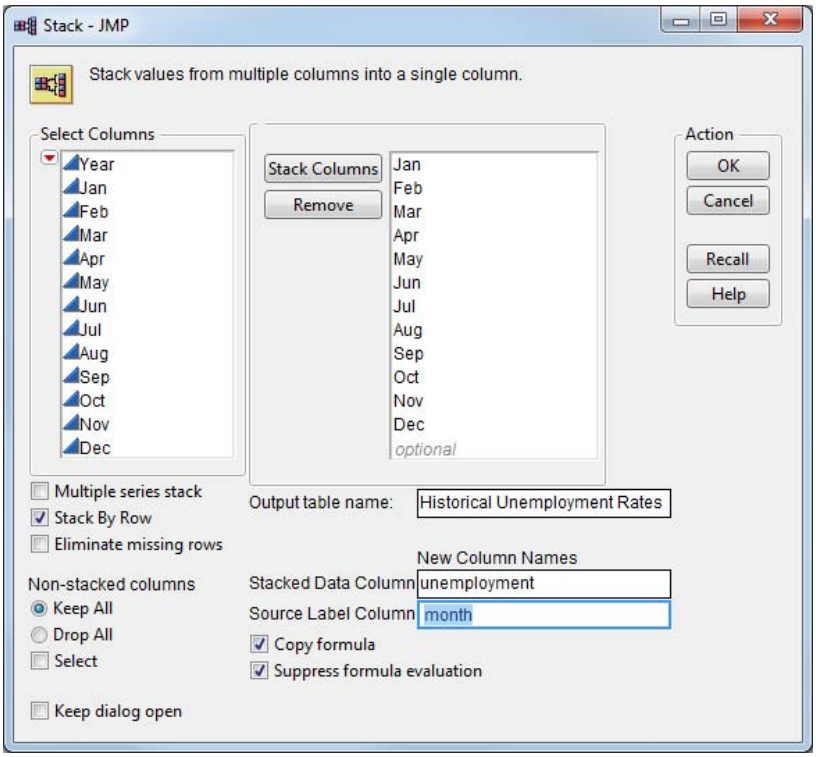
Figure 13. Unemployment Data, After Data Cleaning



	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1	1950	6.5	6.4	6.3	5.8	5.5	5.4	5	4.5	4.4	4.2	4.2	4.3
2	1951	3.7	3.4	3.4	3.1	3	3.2	3.1	3.1	3.3	3.5	3.5	3.1
3	1952	3.2	3.1	2.9	2.9	3	3	3.2	3.4	3.1	3	2.8	2.7
4	1953	2.9	2.6	2.6	2.7	2.5	2.5	2.6	2.7	2.9	3.1	3.5	4.5
5	1954	4.9	5.2	5.7	5.9	5.9	5.6	5.8	6	6.1	5.7	5.3	5
6	1955	4.9	4.7	4.6	4.7	4.3	4.2	4	4.2	4.1	4.3	4.2	4.2
7	1956	4	3.9	4.2	4	4.3	4.3	4.4	4.1	3.9	3.9	4.3	4.2
8	1957	4.2	3.9	3.7	3.9	4.1	4.3	4.2	4.1	4.4	4.5	5.1	5.2
9	1958	5.8	6.4	6.7	7.4	7.4	7.3	7.5	7.4	7.1	6.7	6.2	6.2
10	1959	6	5.9	5.6	5.2	5.1	5	5.1	5.2	5.5	5.7	5.8	5.3
11	1960	5.2	4.8	5.4	5.2	5.1	5.4	5.5	5.6	5.5	6.1	6.1	6.6
12	1961	6.6	6.9	6.9	7	7.1	6.9	7	6.6	6.7	6.5	6.1	6
13	1962	5.8	5.5	5.6	5.6	5.5	5.5	5.4	5.7	5.6	5.4	5.7	5.5
14	1963	5.7	5.9	5.7	5.7	5.9	5.6	5.6	5.4	5.5	5.5	5.7	5.5
15	1964	5.6	5.4	5.4	5.3	5.1	5.2	4.9	5	5.1	5.1	4.8	5
16	1965	4.9	5.1	4.7	4.8	4.6	4.6	4.4	4.4	4.3	4.2	4.1	4
17	1966	4	3.8	3.8	3.8	3.9	3.8	3.8	3.8	3.7	3.7	3.6	3.8

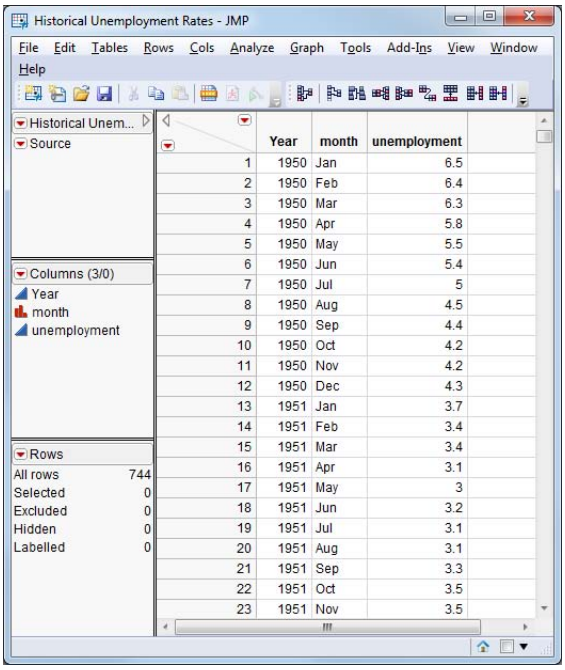
You would like to plot the data over time, but the data is not in the correct format. What you need to do next is *stack* the data, so that each year has 12 rows of data, one for each month. JMP makes this easy with its Stack functionality, which you can invoke through **Tables->Stack**.

Figure 14. The JMP Stack Dialog Box



The resulting table is what you expect. Instead of 62 rows for 62 years of data, we have 744 (12 x 62) rows of data.

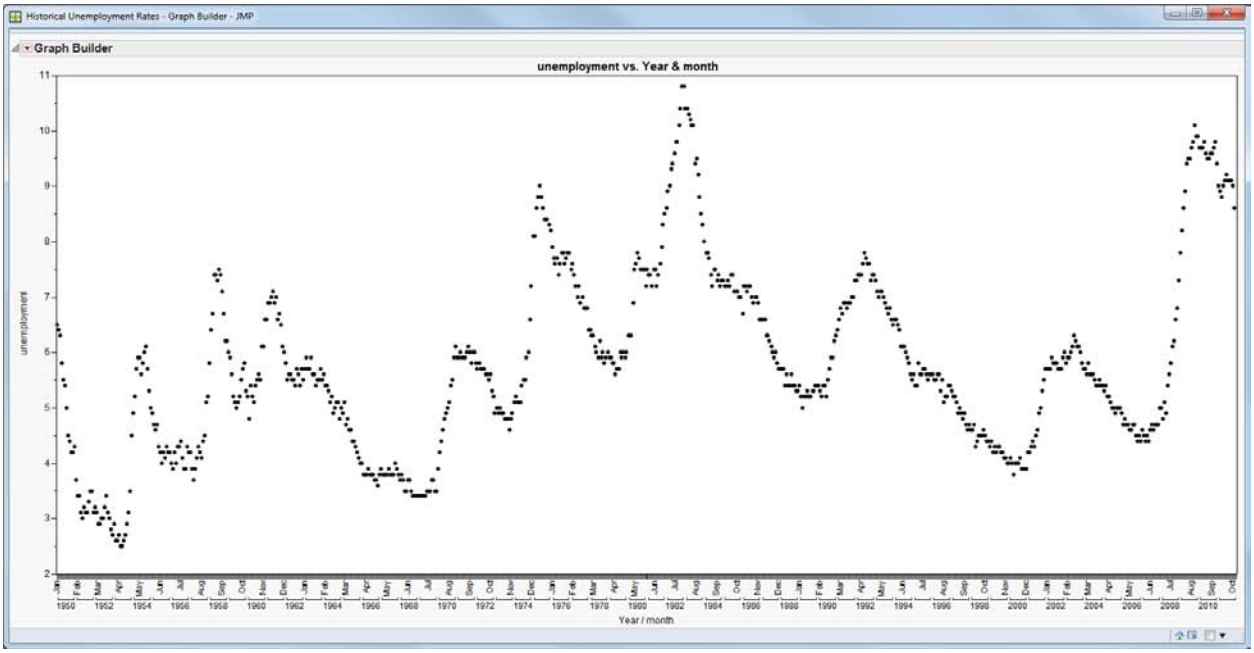
Figure 15. Unemployment Data, After Reshaping Using the Stack Functionality



	Year	month	unemployment
1	1950	Jan	6.5
2	1950	Feb	6.4
3	1950	Mar	6.3
4	1950	Apr	5.8
5	1950	May	5.5
6	1950	Jun	5.4
7	1950	Jul	5
8	1950	Aug	4.5
9	1950	Sep	4.4
10	1950	Oct	4.2
11	1950	Nov	4.2
12	1950	Dec	4.3
13	1951	Jan	3.7
14	1951	Feb	3.4
15	1951	Mar	3.4
16	1951	Apr	3.1
17	1951	May	3
18	1951	Jun	3.2
19	1951	Jul	3.1
20	1951	Aug	3.1
21	1951	Sep	3.3
22	1951	Oct	3.5
23	1951	Nov	3.5

From this newly shaped data, you can now create a visualization of unemployment rate over time. In the JMP Graph Builder platform, drag and drop columns into the X and Y roles to see the rise and fall in unemployment rate over time. Note that the Month and Year values are nested in the x-axis so you did not have to create a merged column that contained both month and year data.

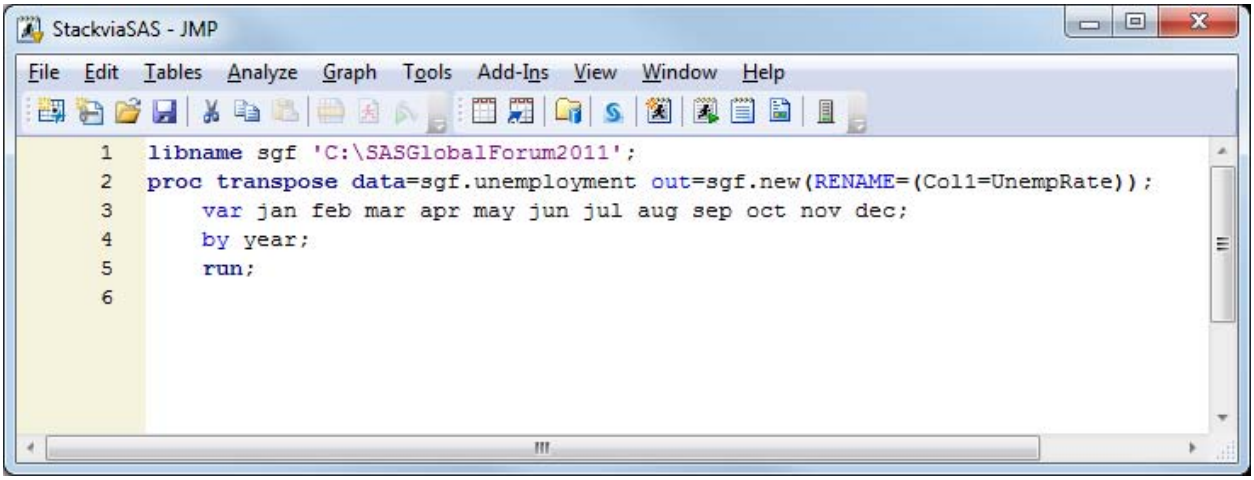
Figure 16. Unemployment Rates over Time



The graph shows the rise and fall of unemployment rates over a 60-year span of time. You can clearly see the rapid rise in unemployment rates at the end of 2008. It is interesting to note that the period of highest unemployment rates occurred in the early 1980s. It is also heartening to see that unemployment appears to be decreasing in recent months.

One more note about this data. If you are an experienced SAS user, you might have already known how to use the DATA step to reshape the data. In this example you used Stack functionality that is available in JMP on the data shown in Figure 13. An alternative would be to use the JMP ability to integrate directly with SAS to create a SAS data set instead. Save the data table from Figure 13 as a SAS data set (for example, C:\SASGlobalForum2011\unemployment.sas7bdat). If you have access to SAS from your machine, you can open the SAS Program Editor in JMP and submit the code to SAS. This code shown below would create a data set named new.sas7bdat that looks identical to the data in Figure 15.

Figure 17. SAS Program Editor within JMP



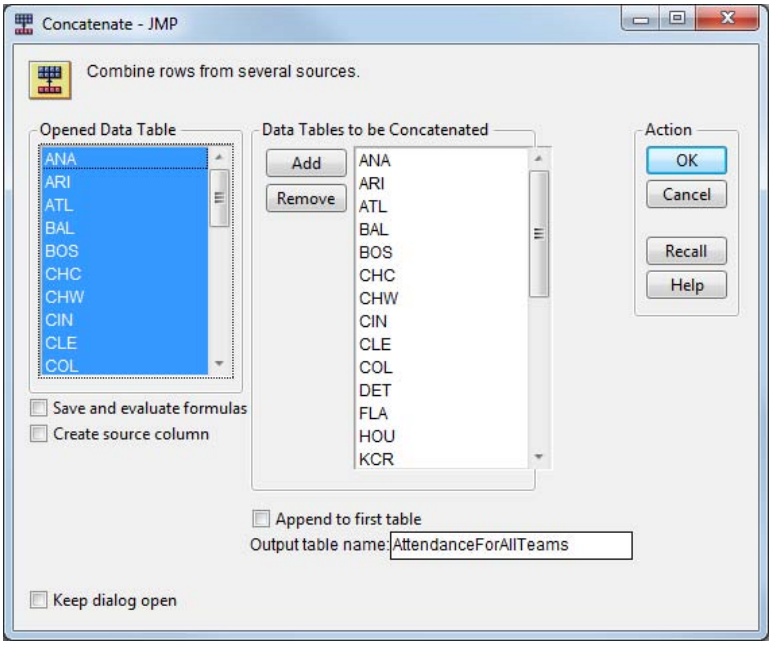
EXAMPLE 3 BASEBALL ATTENDANCE

Many major cities have baseball teams, and some cities have seen teams come and go, and come back again. Suppose you want to analyze attendance per city, to see if you can determine if fans in certain cities are more loyal than others. If you want to analyze data on a per-city basis, you will need information about the different ballparks within each city as well as attendance for each team. Also, you might want to consider the capacity of each ballpark, since sizes vary widely; for example, the Dodgers' stadium seats 56,000 people while Oakland's stadium only seats 37,000.

Attendance data is available online at Baseball-Reference.com[6] for each team. Suppose you saved attendance data for each team into a JMP data table. To more easily work with the data, you should use the Concatenate functionality in JMP to combine all the attendance information into one big table. Use **Tables->Concatenate** to invoke the dialog box.

[6] Data is accessible from http://www.baseball-reference.com.

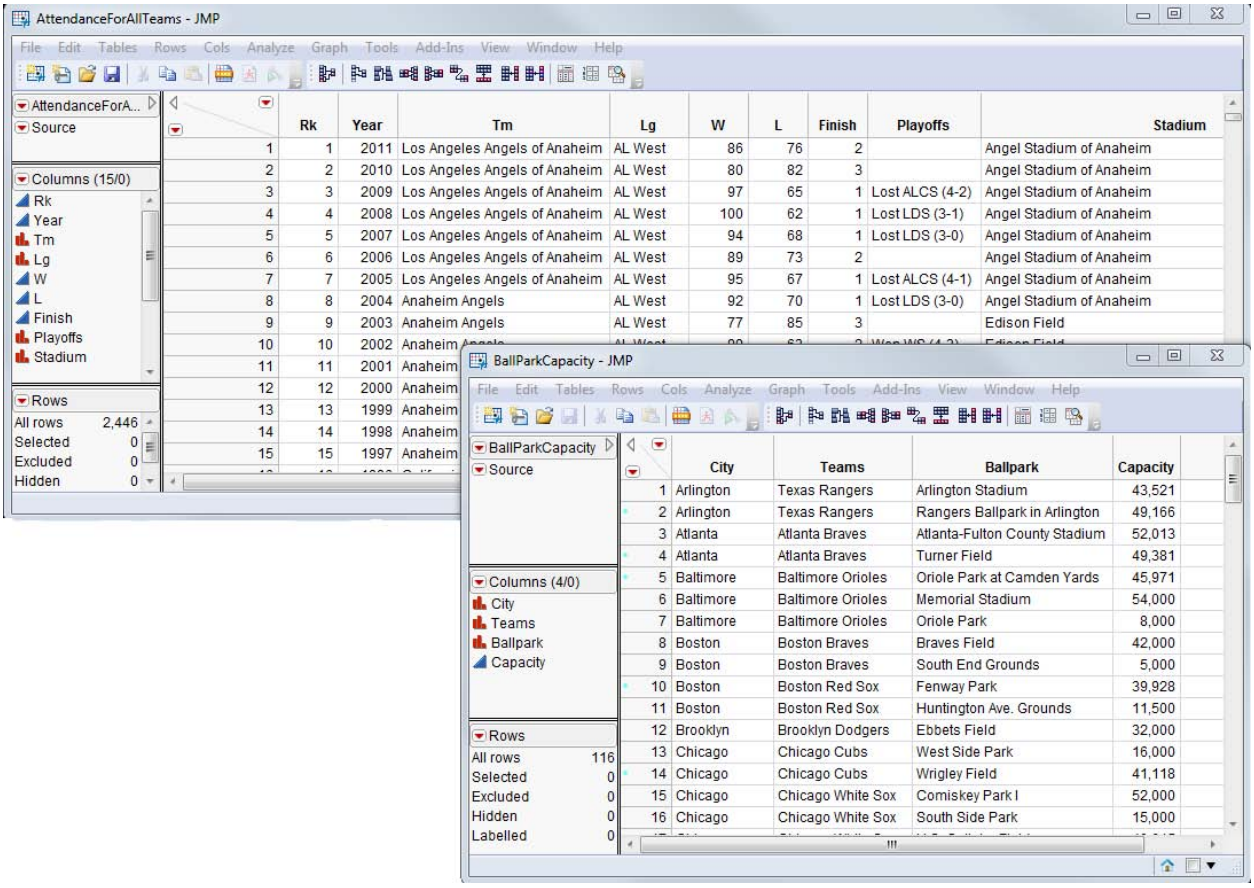
Figure 18. The JMP Concatenate Dialog Box



The capacity of each ballpark is available online at BallParksofBaseball.com[7] and can be imported into JMP using Internet Open or copy-and-paste. Shown below are the two tables you just created – AttendanceForAllTeams, which contains average attendance information for each team and each year, including which ballpark the team played in each year; and BallParkCapacity, which contains a single row for past and present ballparks.

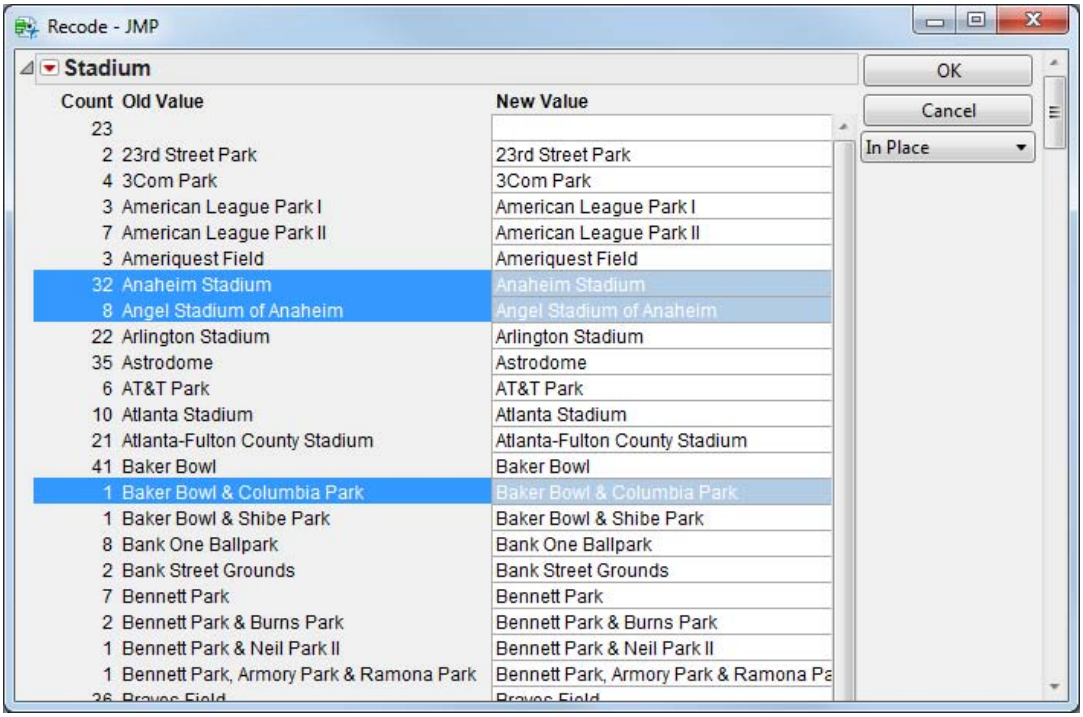
[7] Data can be accessed from http://www.ballparksofbaseball.com.

Figure 19. Tables Showing Concatenated Attendance for All Teams and the Capacity for Each Ballpark



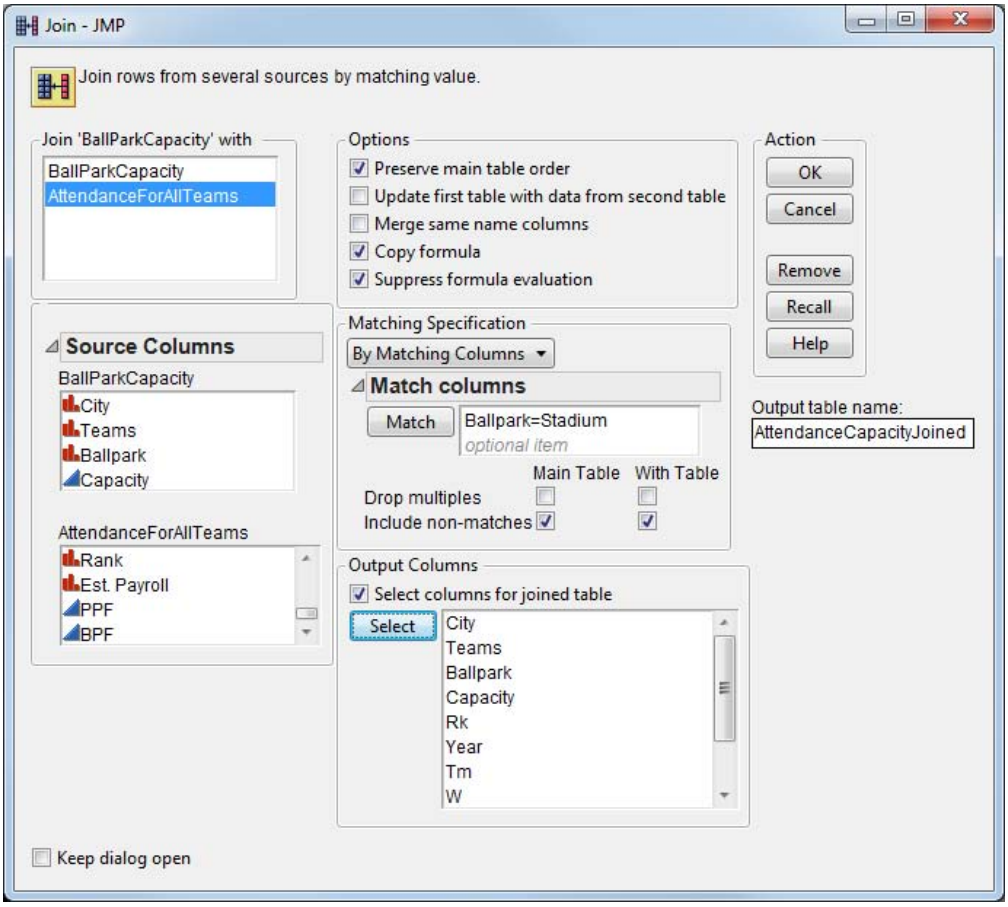
Your next task is to join the two tables so that capacity information for each stadium can be used. The matching columns you will use are the Stadium/Ballpark columns. It is important to examine the values for these two matching columns. Use **Cols->Recode** to look at the values and make judgments about the values of the names. For example, the *Anaheim Stadium* and the *Angel Stadium of Anaheim* are actually the same, but stadiums go through name changes frequently. Recode allows you to modify all values as appropriate. Another example, *Baker Bowl & Columbia Park*, shows that in one year, a team played in two different stadiums. For these cases you would have to decide how to handle the row. The Recode step is necessary as part of data cleaning in any software application and can take time. JMP helps minimize the amount of time spent here.

Figure 20. The JMP Recode Dialog Box



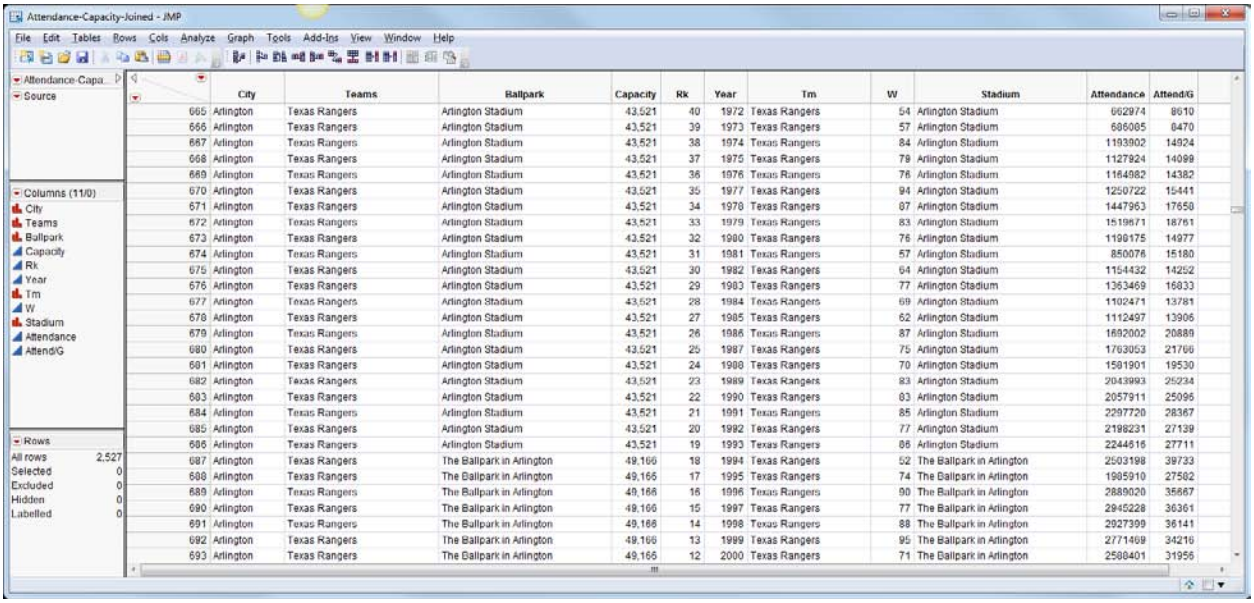
Once you have determined the correct values for stadium names, your next task is to join the two tables. Invoke the Join functionality that is available in JMP using **Tables->Join**. In the dialog box, match Stadiums=Ballparks. You may also select which columns you would like to keep in the resulting table.

Figure 21. The JMP Join Dialog Box



The resulting table contains 2,527 rows, but a few hundred rows contain some missing data because of mismatching in the matching columns we selected. This is primarily due to very old ballparks listed in the attendance data sets (some from the 1800s!) that did not have a corresponding entry in the capacity table.

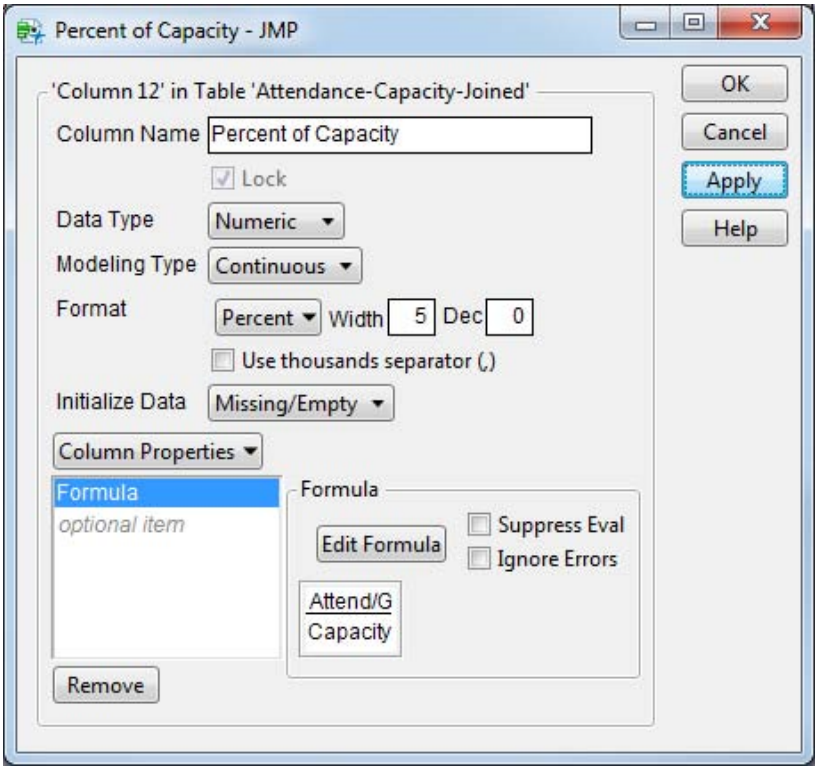
Figure 22. Resulting JMP Data Table after Attendance and Capacity Tables Have Been Joined



	City	Teams	Ballpark	Capacity	Rk	Year	Tm	W	Stadium	Attendance	Attend/G
665	Arlington	Texas Rangers	Arlington Stadium	43,521	40	1972	Texas Rangers	54	Arlington Stadium	662974	8610
666	Arlington	Texas Rangers	Arlington Stadium	43,521	39	1973	Texas Rangers	57	Arlington Stadium	686005	8470
667	Arlington	Texas Rangers	Arlington Stadium	43,521	38	1974	Texas Rangers	84	Arlington Stadium	1193902	14924
668	Arlington	Texas Rangers	Arlington Stadium	43,521	37	1975	Texas Rangers	79	Arlington Stadium	1127924	14099
669	Arlington	Texas Rangers	Arlington Stadium	43,521	36	1976	Texas Rangers	76	Arlington Stadium	1164982	14382
670	Arlington	Texas Rangers	Arlington Stadium	43,521	35	1977	Texas Rangers	94	Arlington Stadium	1250722	15441
671	Arlington	Texas Rangers	Arlington Stadium	43,521	34	1978	Texas Rangers	87	Arlington Stadium	1447963	17650
672	Arlington	Texas Rangers	Arlington Stadium	43,521	33	1979	Texas Rangers	83	Arlington Stadium	1519671	18761
673	Arlington	Texas Rangers	Arlington Stadium	43,521	32	1980	Texas Rangers	76	Arlington Stadium	1199175	14977
674	Arlington	Texas Rangers	Arlington Stadium	43,521	31	1981	Texas Rangers	57	Arlington Stadium	850076	15180
675	Arlington	Texas Rangers	Arlington Stadium	43,521	30	1982	Texas Rangers	64	Arlington Stadium	1154432	14252
676	Arlington	Texas Rangers	Arlington Stadium	43,521	29	1983	Texas Rangers	77	Arlington Stadium	1363469	16833
677	Arlington	Texas Rangers	Arlington Stadium	43,521	28	1984	Texas Rangers	69	Arlington Stadium	1102471	13781
678	Arlington	Texas Rangers	Arlington Stadium	43,521	27	1985	Texas Rangers	62	Arlington Stadium	1112497	13906
679	Arlington	Texas Rangers	Arlington Stadium	43,521	26	1986	Texas Rangers	87	Arlington Stadium	1692002	20889
680	Arlington	Texas Rangers	Arlington Stadium	43,521	25	1987	Texas Rangers	75	Arlington Stadium	1763053	21766
681	Arlington	Texas Rangers	Arlington Stadium	43,521	24	1988	Texas Rangers	70	Arlington Stadium	1581901	19530
682	Arlington	Texas Rangers	Arlington Stadium	43,521	23	1989	Texas Rangers	83	Arlington Stadium	2043993	25234
683	Arlington	Texas Rangers	Arlington Stadium	43,521	22	1990	Texas Rangers	83	Arlington Stadium	2057911	25096
684	Arlington	Texas Rangers	Arlington Stadium	43,521	21	1991	Texas Rangers	85	Arlington Stadium	2297720	28367
685	Arlington	Texas Rangers	Arlington Stadium	43,521	20	1992	Texas Rangers	77	Arlington Stadium	2198231	27139
686	Arlington	Texas Rangers	Arlington Stadium	43,521	19	1993	Texas Rangers	86	Arlington Stadium	2244616	27711
687	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	18	1994	Texas Rangers	52	The Ballpark in Arlington	2503198	39733
688	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	17	1995	Texas Rangers	74	The Ballpark in Arlington	1905910	27582
689	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	16	1996	Texas Rangers	90	The Ballpark in Arlington	2889020	35667
690	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	15	1997	Texas Rangers	77	The Ballpark in Arlington	2845228	36361
691	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	14	1998	Texas Rangers	88	The Ballpark in Arlington	2927399	36141
692	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	13	1999	Texas Rangers	95	The Ballpark in Arlington	2771469	34216
693	Arlington	Texas Rangers	The Ballpark in Arlington	49,166	12	2000	Texas Rangers	71	The Ballpark in Arlington	2568401	31956

You might want to measure attendance by how full the stadium is. Create a new column named Percent of Capacity with a formula of Attend/G divided by Capacity.

Figure 23. Example of the Column Info Dialog Box with a Formula



Using the JMP **Graph Builder** platform, create a graph showing # of Wins and Percent of Capacity over the years. You can use the **Local Data Filter** to limit the years to 1970 or later, since data from those years contain at least 24 teams per year and is thus more interesting. Also use **Local Data Filter** to view each city by itself to see how the Percent of Capacity is affected by the # of Wins. Below are two examples. The Atlanta Braves franchise started in 1966. Its curves follow each other closely, indicating that average attendance in Atlanta is strongly correlated with how well the team is doing. Conversely, Boston Red Sox fans continue to grow more loyal over time, even when the Red Sox hit a rough streak in the 1980s and early 1990s. This might imply that Red Sox fans are more loyal. You could investigate this idea further by using some of the more analytical platforms in JMP.

Figure 24. Graph Builder Chart Showing # of Wins and Percent Capacity over Time for Atlanta

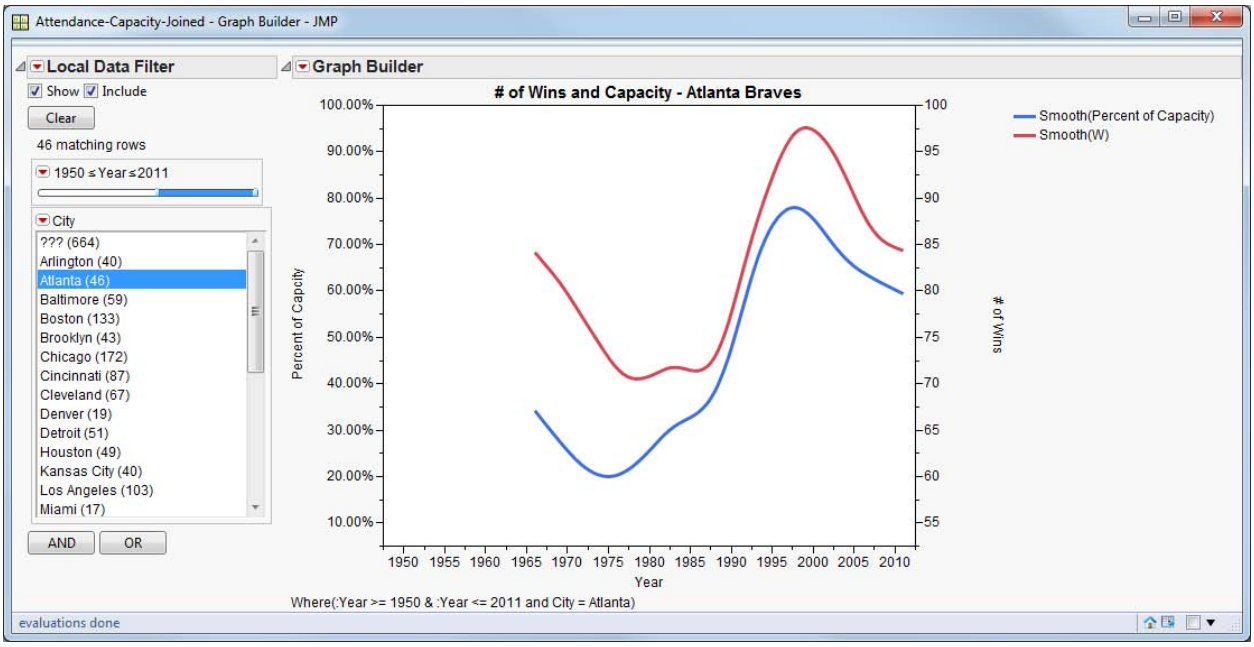
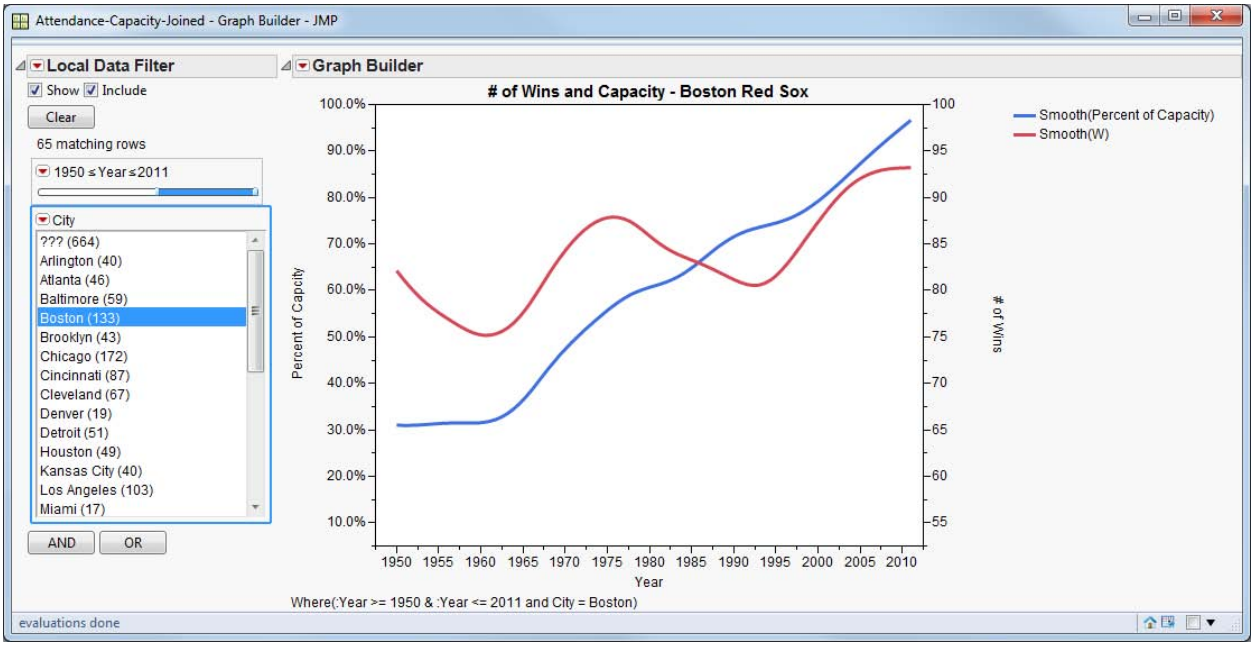


Figure 25. Graph Builder Chart Showing # of Wins and Percent Capacity over Time for Boston



CONCLUSION

JMP 10 continues the tradition of offering varied and easy-to-use ways of accessing and reshaping data. Knowing the ways in which JMP can help analyze data will allow you to quickly take raw data from any source, reshape, and customize it in order to get to the "good part" of data analysis. JMP will help you uncover the answers within your data.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Audrey Ventura
SAS Campus Drive
SAS Institute Inc.
E-mail: Audrey.Ventura@jmp.com