

Twelve Ways to Better Graphs

LeRoy Bessler PhD, Bessler Consulting and Research

ABSTRACT

If you are looking for ways to make your graphs more communication-effective, this tutorial can help. It covers both the new ODS Graphics SG (Statistical Graphics) procedures, and the traditional SAS/GRAPH[®] software G procedures. The focus is on management reporting and presentation graphs, but the principles are relevant for statistical graphs as well. Important features unique to SAS[®] 9.4 are included, but most of the designs and construction methods apply to earlier versions as well. The principles of good graphic design are actually independent of your choice of software.

INTRODUCTION

This tutorial is meant for anyone who creates graphs, or directs that graphs be created, or uses graphs that someone else has created. It is meant for people who create graphs with any software, and who are at any skill level. If using a graph or directing that a graph be created, you should know what you have a right to expect as a communication-effective graph.

I'm sharing lessons learned, as well as principles and methods, developed as a working user of graphics facilities in the SAS System since 1981. This is not a coding tutorial. Appendix A contains code, nowhere else available, for one example. The papers in the bibliography have code for some examples. Code for any example, as well as an index to all my web-available papers, is available upon email request.

WHY GRAPHS ARE NECESSARY, NOT JUST AN OPTION

It is intuitive that, and research has unsurprisingly confirmed that, graphs facilitate and accelerate:

- understanding your data;
- drawing inferences from and about data;
- making decisions based upon data.

However, as will be emphasized and supported with solutions, reliable inferences or decisions requires access to the graph's precise input data, not just a picture, as will be discussed and implemented later.

Though graphs are not self-sufficient, it's important to realize that statistics by themselves are inadequate. In 1973 Francis Anscombe presented what has become known as Anscombe's Quartet, which shows the importance of graphing your data before drawing inferences from its statistics. Here is the data:

I		II		III		IV	
x	y	x	y	x	y	x	y
10	8.04	10	9.1	10	7.46	8	6.6
8	6.95	8	8.1	8	6.77	8	5.8
13	7.58	13	8.7	13	12.7	8	7.7
9	8.81	9	8.8	9	7.11	8	8.8
11	8.33	11	9.3	11	7.81	8	8.5
14	9.96	14	8.1	14	8.84	8	7
6	7.24	6	6.1	6	6.08	8	5.3
4	4.26	4	3.1	4	5.39	19	13
12	10.8	12	9.1	12	8.15	8	5.6
7	4.82	7	7.3	7	6.42	8	7.9
5	5.68	5	4.7	5	5.73	8	6.9

Here are the results from PROC REG:

Analysis of Variance

dset=I					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	27.51000	27.51000	17.99	0.0022
Error	9	13.76269	1.52919		
Corrected Total	10	41.27269			

dset=II					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	27.50000	27.50000	17.97	0.0022
Error	9	13.77629	1.53070		
Corrected Total	10	41.27629			

dset=III					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	27.47001	27.47001	17.97	0.0022
Error	9	13.75619	1.52847		
Corrected Total	10	41.22620			

dset=IV					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	27.49000	27.49000	18.00	0.0022
Error	9	13.74249	1.52694		
Corrected Total	10	41.23249			

dset=I			
Root MSE	1.23660	R-Square	0.6665
Dependent Mean	7.50091	Adj R-Sq	0.6295
Coeff Var	16.48605		

dset=II			
Root MSE	1.23721	R-Square	0.6662
Dependent Mean	7.50091	Adj R-Sq	0.6292
Coeff Var	16.49419		

dset=III			
Root MSE	1.23631	R-Square	0.6663
Dependent Mean	7.50000	Adj R-Sq	0.6292
Coeff Var	16.48415		

dset=IV			
Root MSE	1.23570	R-Square	0.6667
Dependent Mean	7.50091	Adj R-Sq	0.6297
Coeff Var	16.47394		

Parameter Estimates

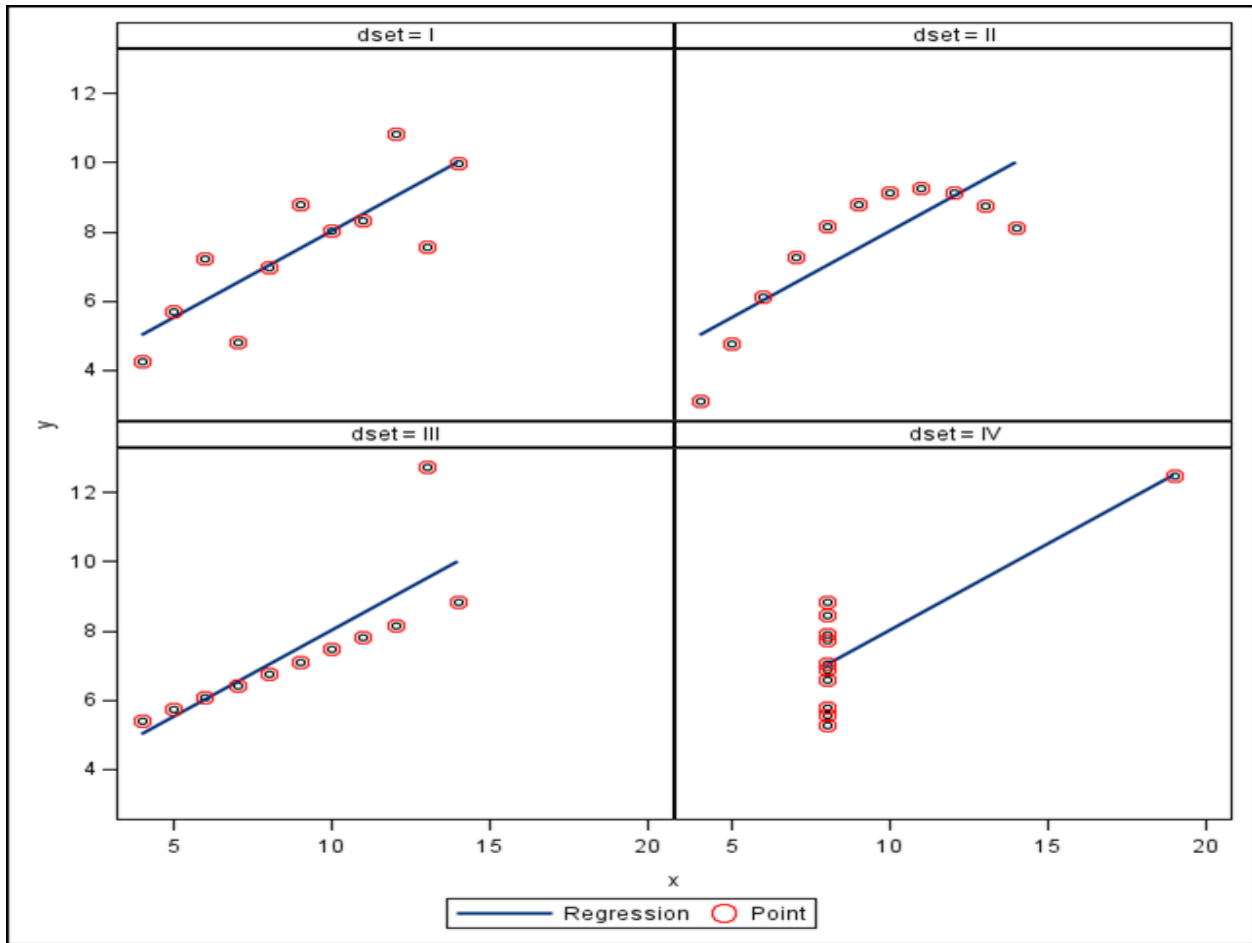
dset=I					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	3.00009	1.12475	2.67	0.0257
x	1	0.50009	0.11791	4.24	0.0022

dset=II					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	3.00091	1.12530	2.67	0.0258
x	1	0.50000	0.11796	4.24	0.0022

dset=III					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	3.00245	1.12448	2.67	0.0256
x	1	0.49973	0.11788	4.24	0.0022

dset=IV		Parameter	Standard		
Variable	DF	Estimate	Error	t Value	Pr > t
Intercept	1	3.00173	1.12392	2.67	0.0256
x	1	0.49991	0.11782	4.24	0.0022

Though the statistics above are nearly identical for all four data sets, here is what their graph looks like:



I learned about this amazing situation from an article by Philip R. Holland in VIEWS Newsletter Issue 54, which you can find online.

Having made the case for graphs, now let's take up twelve of the many ways to make good graphs, where by "good" I mean communication-effective.

But before we proceed, let me emphasize my always guiding design principle:

Accelerate/Facilitate Visual Data Insights with Simplicity

Complexity can distort, impede, and impair communication.

Simplicity is Powerful and Elegant.

Elegance = Everything Needed + Only The Needed.

A sparse image focuses attention—so, a sparse graph is more easily and more quickly interpreted.

Now on to the Twelve Ways.

THE TWELVE WAYS

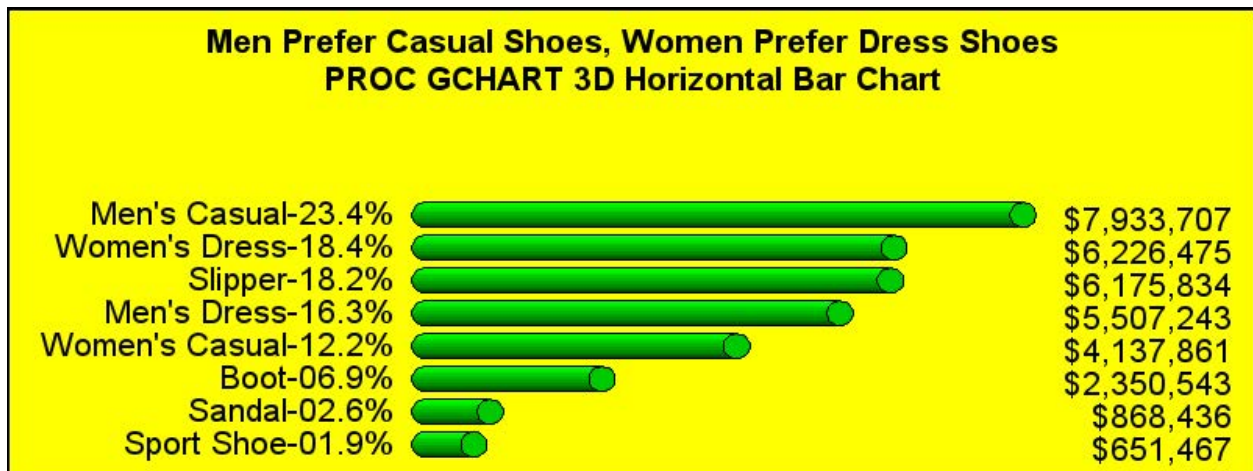
One. Use color to communicate, not to decorate.

I have an entire paper that is not yet in the public domain. Predecessor versions are out there, some of them award-winning, but for the latest edition, simply send me an email. If you are using color to communicate, and you want the color to be distinguishable, then:

- text must be thick enough
- lines must be thick enough
- plot markers must be big enough
- legend samples must be big enough

Two. Whenever possible, make your graph title a headline.

Tell your viewer the inference from, or the revelation in, the graph. The description for the graph might be self-evident or in the axis labels. If an explicit graph description is needed, it can be provided as a subtitle. The headline for the graph below oversimplifies the content, but headlines do that. (**Note: This type of bar chart could serve as a pie chart alternative when there would be too many slices—it delivers category description, value, and percent of whole.**)



Three. Assure Text Readability.

Text readability, which is fundamental to communication, is taken for granted, but not always delivered.

Not all fonts are equally readable. (The serif font Georgia and sans serif Verdana were developed expressly for Microsoft for web page readability.)

Text should be sufficient size.

Text color versus background color should be high contrast. Black on white is not an accidental choice.

There is NO communication benefit from background images, textured background, or a color gradient background. These “embellishments” create visual interference.

A plain solid color background is best.

Use horizontal text, not vertical text. We read horizontally. Instead of using vertical text to label a vertical axis, put the axis label in the graph title or a subtitle. Graph titles often identify the axis variables, in which case an axis label is superfluous.

If your horizontal axis is dense with tick marks, consider not labelling every one, if intermediate labels can be inferred, rather than resorting to vertical text for the tick mark labels.

It's not a matter of readability, but sparse image delivery: a horizontal axis of dates does not need a label.

Now, we finally get to an important, but not necessarily obvious, graphic design principle . . .

Four. Image Plus Precise Numbers Are Always Necessary For Quick/Easy Plus Reliable Inference

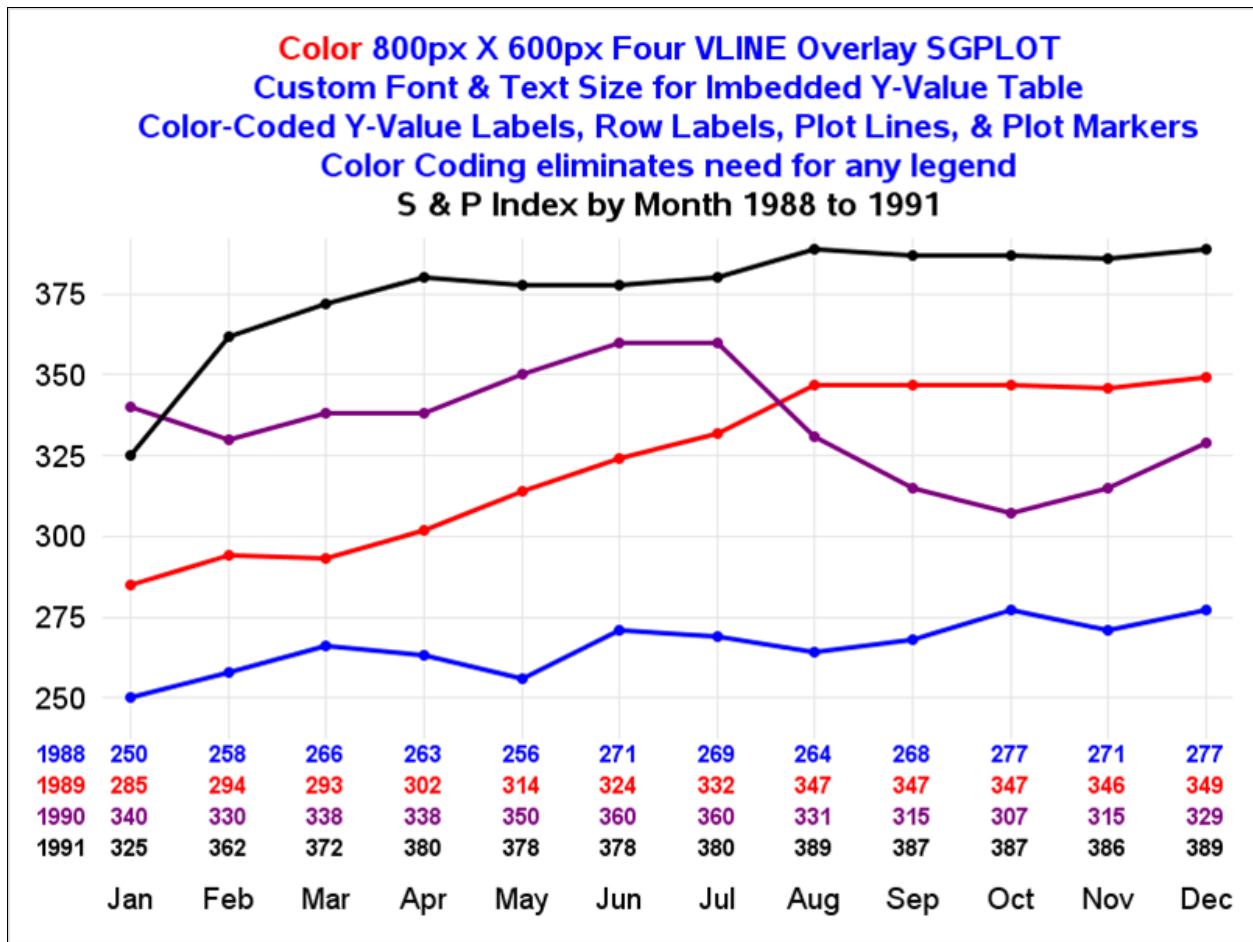
Despite the popularity of the practice,
It is IMPOSSIBLE to reliably determine precise numbers
by comparing plot point locations or bar ends to axis values.

Five. For time series plots, whenever feasible, provide either an on-chart table of y values or annotate the plot points.

Both solutions deliver image plus precise numbers.

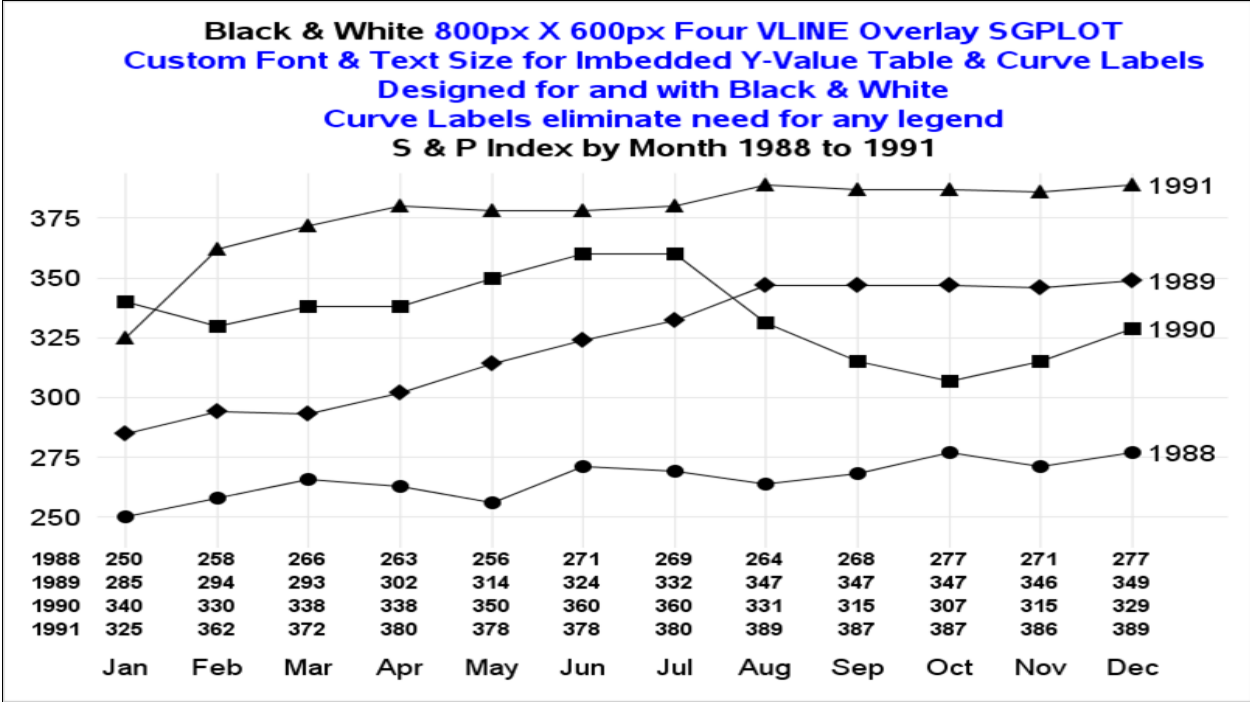
A web-enabled time series plot can provide precise numbers in the pop-up boxes of hover text (for both the y value and the x value of each point, as well as a line identifier if it's a multi-line plot), but the precise information disappears as soon as the mouse moves.

Here is the best solution for a multi-line chart using color:

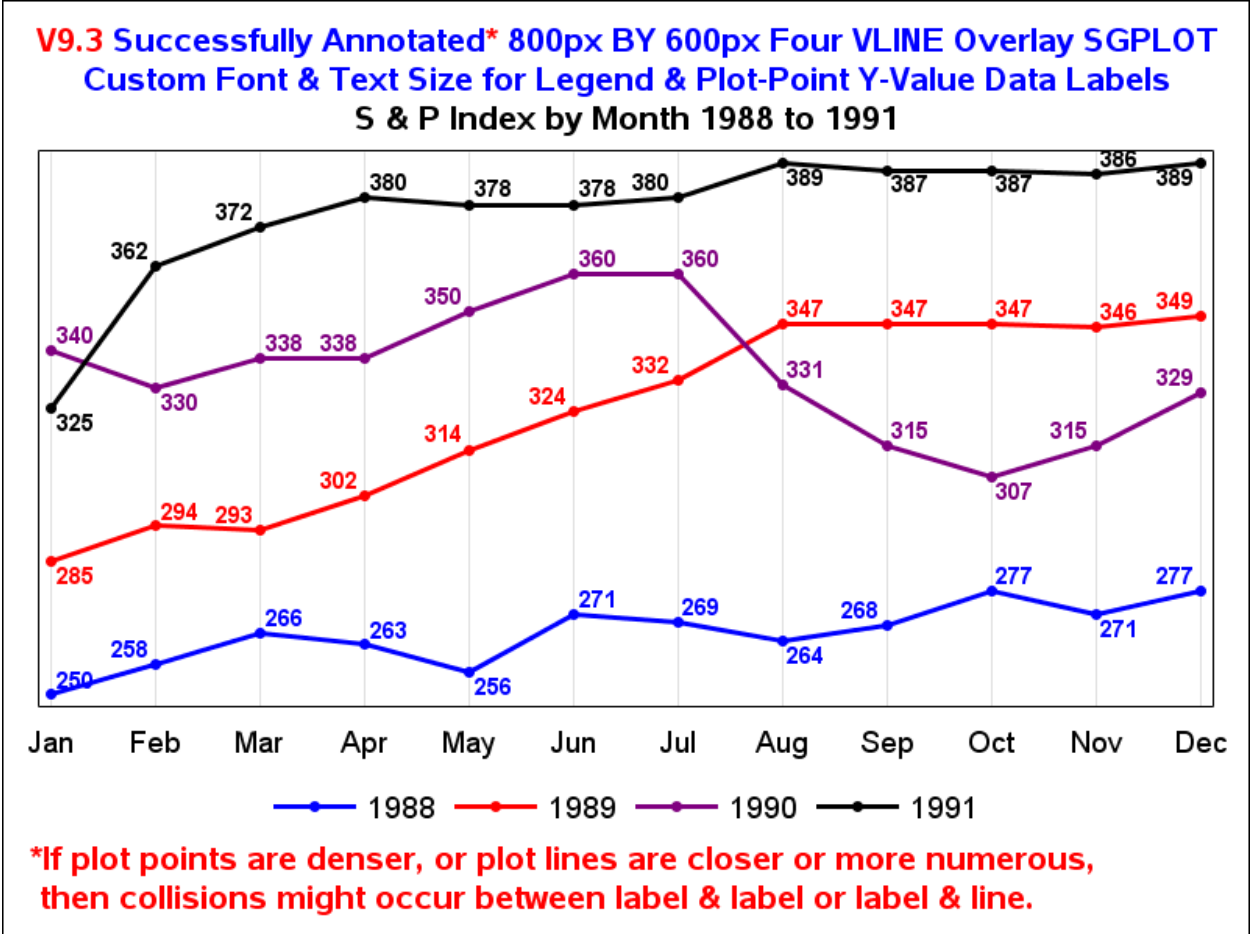


The y-axis values are probably superfluous. So, too, are x-axis grid lines for widely separated values. Note that this plot needs no legend. Row labels for the on-chart table are, in effect, a legend.

For a black-and-white version of this plot, you need to use different plot markers for each year. The commonest solution is to use a legend. However, if the plot line end points are sufficiently separated, curve labels can be used instead. Nevertheless, for a production application, where you do not get to see the result before the end user does, so that you cannot fix any problem, it would be best to use a legend, not curvelabels. Having said all that, on the next page is an example of a black-and-white multi-line plot with curvelabels.



Here is an example of on-graph annotation:



Multi-line overlay plots such as those with on-chart tables and the annotated plot, provide two benefits:

- easy-to-detect seasonality, if any
- no toggling between separate charts for separate years

However, there is no concurrent view of the full range, unless you web-enable the multi-line plot, and add a hyperlink to a full-range plot. We will present that solution later.

For now, let's discuss on-graph annotation, which we just looked at above. On-graph annotation for a plot has inherent limitations. If there are too many points per line, or too many lines per plot, annotations might collide—either label with label or label with line. Several years ago, I developed an algorithm to annotate a line plot with the SAS/GRAPH Annotate facility which performed better than the SAS/GRAPH POINTLABEL OPTION. I characterize my algorithm as collision-resistant, but not collision-proof, due to the inherent limitations of annotation just cited. For ODS Graphics, annotation using Version 9.3 performs better than Version 9.4, and the annotation in both versions for ODS Graphics line plots makes what I regard as suboptimal annotation placements.

Six. Whenever Adequate To the Need, Use Sparse Annotation for Trends / Times Series Plots

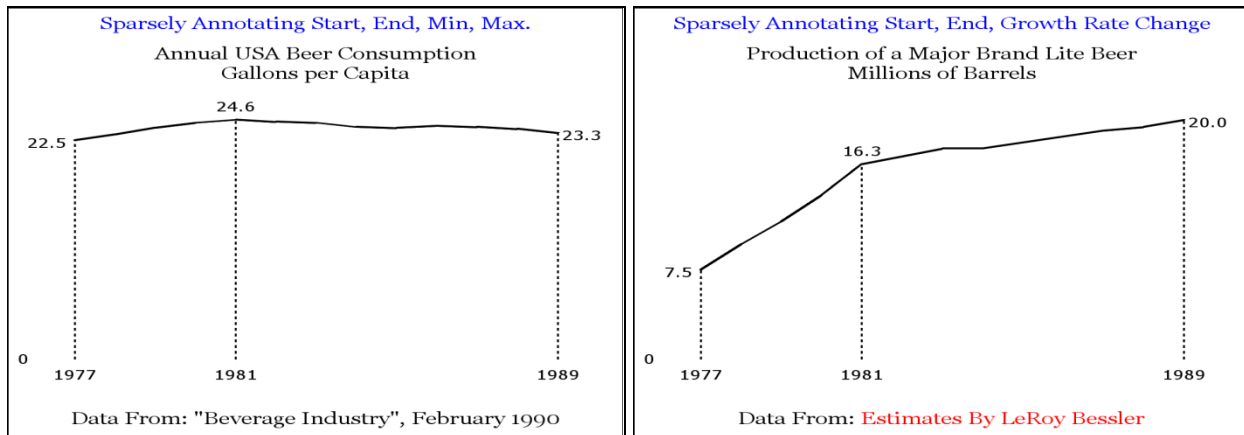
In 1992, I developed what I called Sparse Annotation. It pre-dates the frequently touted Spark Line concept that emerged in 1998.

Sparse Annotation has two benefits: it is **collision-proof**; and it “Shows Them What’s Important.” **Show Them What’s Important** has been one my consistent recommendations since the inception of my work on Communication-Effective Graphic Design.

After looking at trend charts for some real-world business data, I had an epiphany (“an experience of sudden and striking realization”). I realized that, or at least opined that: **When looking at any trend usually the most interesting plot points are Start, End, any intermediate y-value minimum, any intermediate y-value maximum, and, in the case of no minimum or maximum, any point where there is a significant permanent change in general slope.** For the latter, there are four possible cases:

- from rapidly rising to slowly rising
- from slowly rising to rapidly rising
- from slowing declining to rapidly declining
- from rapidly declining to slowly declining

The obvious exception to my case for Sparse Annotation, as usually defined, is when there are any points where significant environmental or operational events occurred.



Seven. Whenever possible, start your vertical axis at zero. It prevents needless anxiety or unjustified elation about changes that might be minor fluctuations. Significance is best measured in absolute magnitude of change or percent change, not the slope of a graph. Of course, if you have a graph where magnifying the granularity is important, then make maximum use of the available vertical space. In the case of the data above, industry observers commented that “beer consumption was flat”, which is the visual effect of the use of the vertical axis as shown.

Eight. Use Web-Enabled Plots: If Too Many Points for On-Chart Table or Full Annotation, Or To Link a Multi-Line Plot of Separate Time Periods and a Plot of the Full Time Range

Web enablement permits you to supply “data tips” (a.k.a. hover text, mouseover text, floatover text, flyover text, tool tips) as pop-up boxes that can contain precise y and x values, formatted however you like, and, if a multi-line plot, the line descriptor, PLUS absolutely anything you want to include—such as a comment about the significance of any point(s) of special interest. By use of a line-break control, you can deliver the information in a stack, rather than just a string of wrapped information. So far, the one thing that HTML does not support is programmer control of font size, font color, or bold, italics, or underline versus normal text.

Data tips are, unfortunately, soft annotation that disappears when you move the mouse. Recall that as the Fourth Way to Better Graphs, I stated:

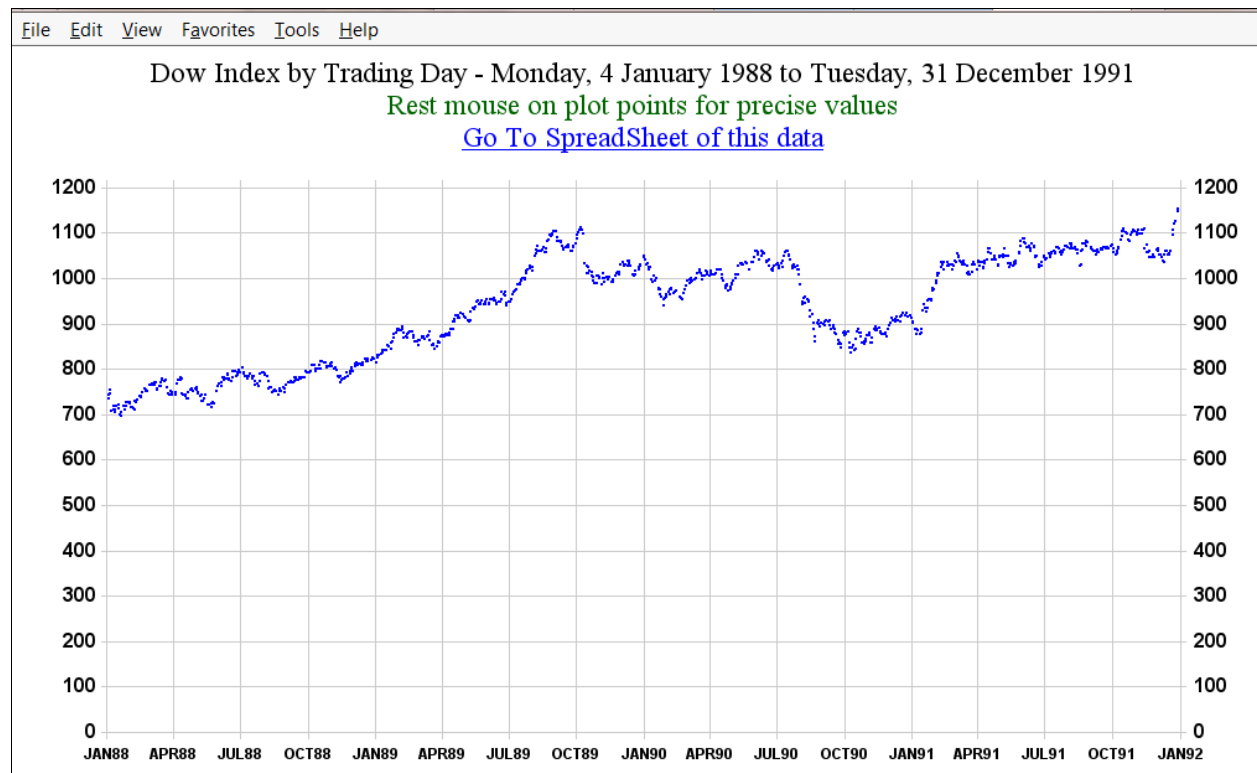
“Image Plus Precise Numbers Are Always Necessary For Quick/Easy Plus Reliable Inference.”

A web-enabled graph can be and should be linked to a spreadsheet of its input data, and the spreadsheet can contain a link back to the graph.

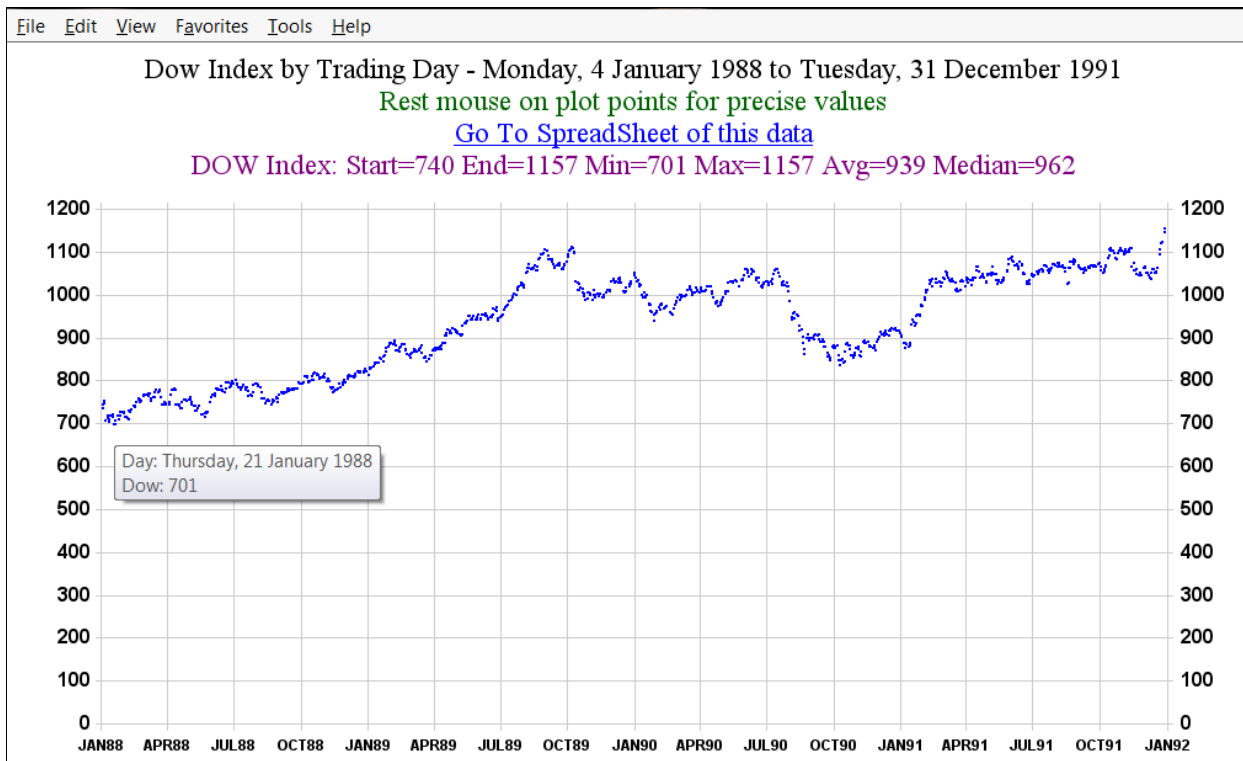
A side benefit of the spreadsheet is that the user can make any desired additional use of the data or simply reformat it to her/his liking, with Excel software that everyone has and knows how to use.

Tip: Size web graphs so that the entire picture can be seen without scrolling. Web browser users want to see the whole picture.

Here is an example of a web graph, with a reminder about the data tips, making no assumption about what a user might realize is available.



On the next page is an enhanced version of this same plot with a dynamically supplied subtitle of significant statistics. It also shows the pop-up box displayed when resting the mouse on a plot point. (For a multi-line plot of the individual years, one could supply a statistics summary for each plot line.)



Below is the spreadsheet that either of the above plots could be linked to. I manually highlighted the row for the plot point with its data tip displayed above. Note the link from the spreadsheet back to the graph.

Dow	Year	Date of Observation
740	1988	Monday, 4 January 1988
747	1988	Tuesday, 5 January 1988
750	1988	Wednesday, 6 January 1988
757	1988	Thursday, 7 January 1988
711	1988	Friday, 8 January 1988
722	1988	Monday, 11 January 1988
715	1988	Tuesday, 12 January 1988
714	1988	Wednesday, 13 January 1988
708	1988	Thursday, 14 January 1988
722	1988	Friday, 15 January 1988
724	1988	Monday, 18 January 1988
718	1988	Tuesday, 19 January 1988
702	1988	Wednesday, 20 January 1988
701	1988	Thursday, 21 January 1988
709	1988	Friday, 22 January 1988

Nine. Show Them What's Important via Ranking and/or Subsetting

- Ranking for bar charts and pie charts
- Subsetting for bar charts

First, a non-graphics example:

Ranking and Subsetting (tabular textual) information

3 of 7 Good Reasons to Smile

- 1. Can make you happy (even if you are not) 😊**
- 2. Makes you more attractive!!!**
- 3. Can make other people happy 😊😊😊😊😊😊😊😊😊😊**
4. Can help you de-stress
5. Can help you land a job
6. Can lead to laughter
7. Just feels good

http://www.sparkpeople.com/resource/wellness_articles.asp?id=1529

Ten: Use Bessler Horizontal Bar Charts To Compare Categorical Data

Here are the characteristics of my Bessler Horizontal Bar Charts:

- Delivers Image + Precise Numbers
- Shows Them What's Important—with Ranking
- Lets Part Stand for the Whole—with Subsetting
- Includes the Option to See It All
- For subsets, reports the significance of the subset

In this era of BIG DATA **hysteria**, let's focus on Enough Data.

Most of us are familiar with the so-called 80-20 Rule, with examples such as these:

- 80% of your product/service sales come from 20% of your customers
- 80% of your IT trouble calls come from 20% of your users

Of course the real numbers will vary from those above, but you get the general idea, and experience bears it out. Similarly, when reporting categorical measures, most of the impact comes from a smallish subset of the full set of categories. So how do we decide how much is **Enough Data**?

Three Ways to Subset Enough Data (I prefer the third way, despite the popularity of the first way):

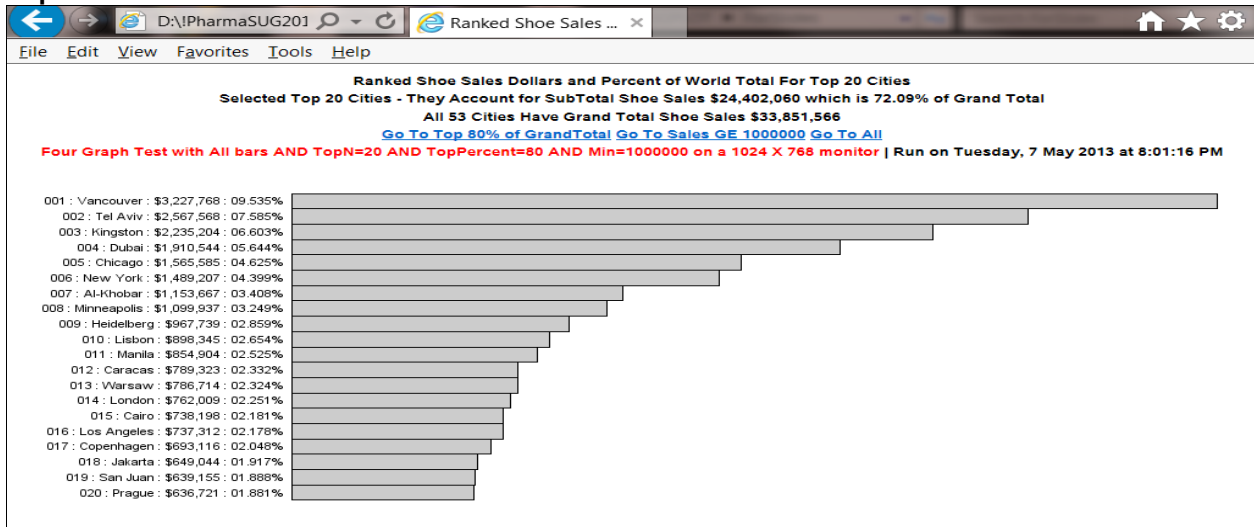
1. Top N (if smallness is the desired characteristic of the measure of interest, this could be Bottom N)
2. CutOff (minimum or maximum)
 - Goal to meet or exceed
 - Threshold to avoid crossing
3. Enough of the Top Values to Account For the Top P Percent of the Grand Total

Above I mentioned that for subsets, you should report the significance of the subset. I do it with dynamically generated title lines to deliver the following:

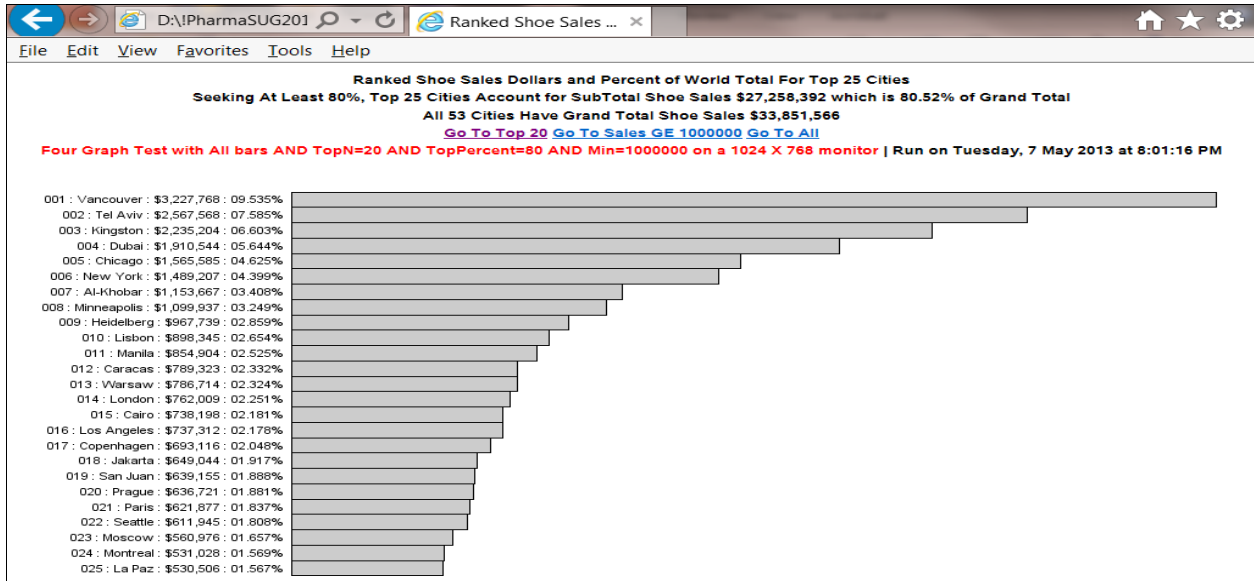
- Selection Criterion (per method 1, 2, or 3 above)
- Observation Count for the Data Selected
- Their Subtotal for the Measure of Interest
- Their Percent Share of the Grand Total of the Measure of Interest
- The Total Number of Observations Available
- The Grand Total of the Measure of Interest

If offering more than one subset of the data, you can include add a title line of links to the other subsets. A further enhancement is an optional final title line for a comment and/or a listing of the Run Day Date Time. On the next page is what Bessler Horizontal Bar Charts look like. The screen image bottoms are trimmed.

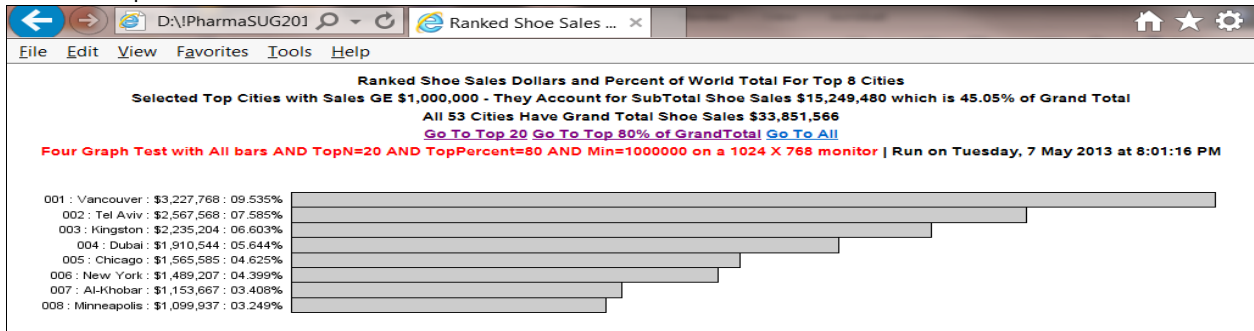
Top 20



Account For At Least 80% of Grand Total



Sales GE \$1000000



The All chart is not shown. Note that the bar widths and font sizes are uniform across the connected set of bar charts.

Here is a close-up view of the Bessler bar chart labels:

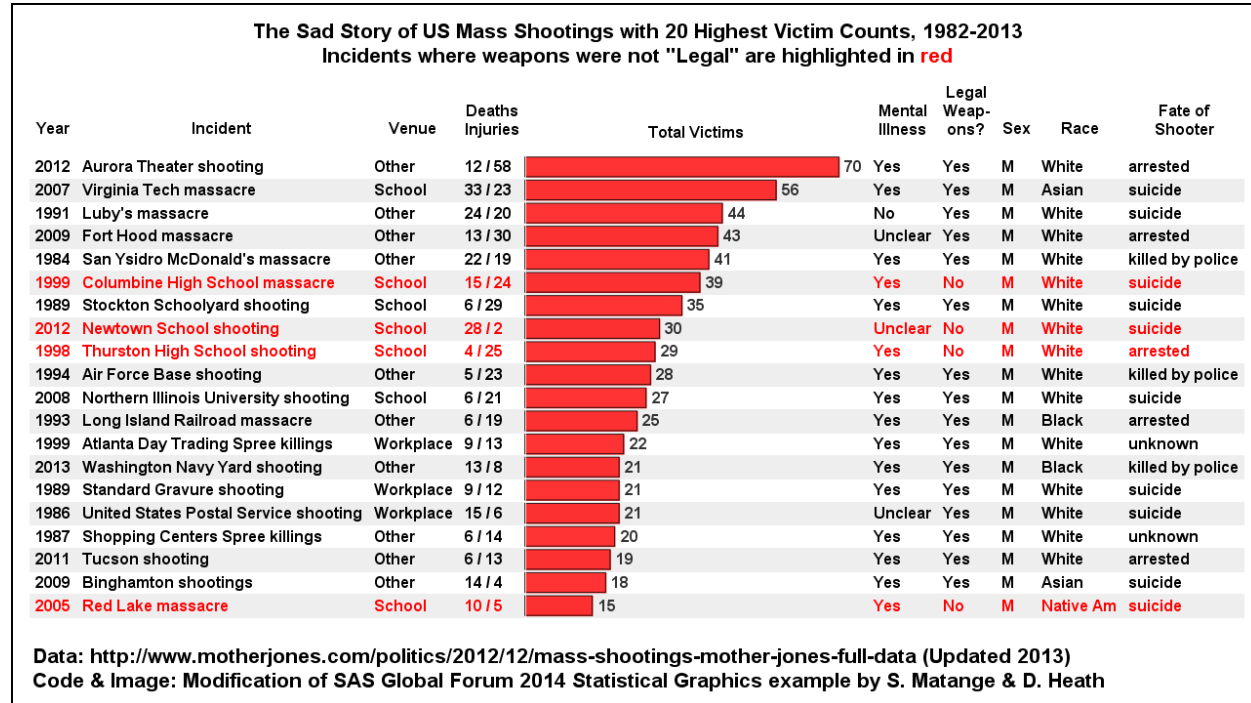
Rank : Description : Value : Pct Of Grand Total

001 : Vancouver : \$3,227,768 : 09.535%
 002 : Tel Aviv : \$2,567,568 : 07.585%
 003 : Kingston : \$2,235,204 : 06.603%
 004 : Dubai : \$1,910,544 : 05.644%
 005 : Chicago : \$1,565,585 : 04.625%
 006 : New York : \$1,489,207 : 04.399%
 007 : Al-Khobar : \$1,153,667 : 03.408%
 008 : Minneapolis : \$1,099,937 : 03.249%

After the repeated appearance of my maximally informative horizontal bar charts in SAS conference papers, the SAS developers came up with their yaxistable feature. Their feature is easier to use than my macro, AND their yaxistables can also be placed at the right-hand-side of a horizontal bar chart. (Their yaxistable feature was similarly preceded by a Bessler method to deliver y-values in a table below the x axis. There, too, the SAS-provided feature is easier, much easier, to use than my method.) Version 9.4 is required for yaxistables.

In a SAS Global Forum 2014 SAS Statistical Graphics Procedures examples handout, I saw a bar chart similar to that below. This example is my version of the chart. For my code, which is nowhere else available, please see Appendix A. The text in the image below looks better in Microsoft Word. Conversion to PDF is unkind to it, for reasons unknown to me.

Intensive Use of YAXISTABLES

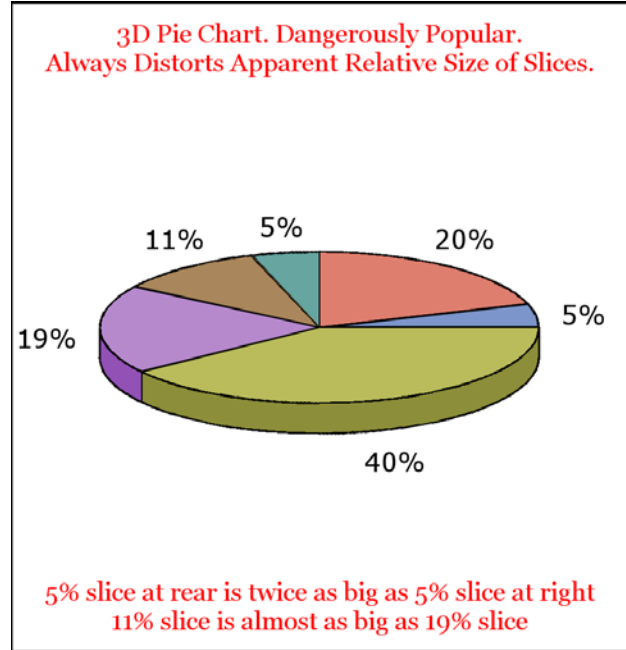


Ranked Pie Charts are taken up later.

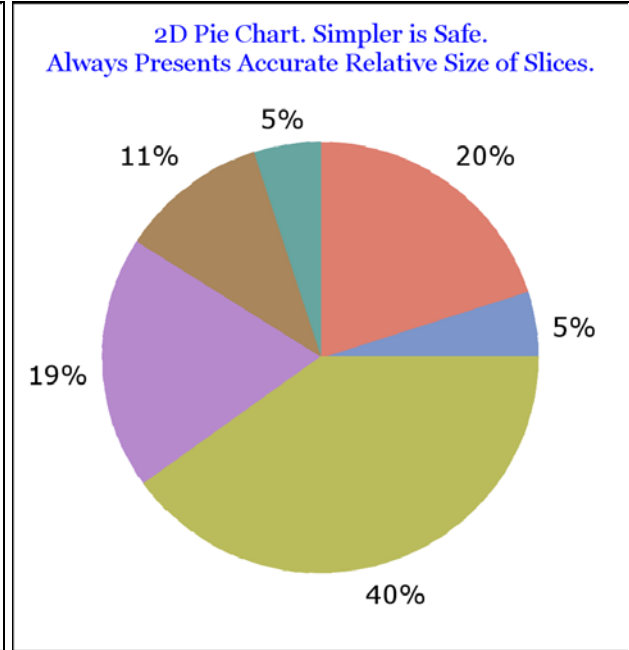
Ten. Use 3D for 3 variables. **Never use 3d pie charts. It is possible to create a safe 3D bar chart.**

Back in the early 1990's I first showed SAS users the problem with 3D pie charts. For MY demonstration, I had to use a non-SAS/GRAPH tool. At the time, SAS/GRAPH users would come up with cunning uses of the Annotate facility in their quest to create 3D pie charts. Eventually, the SAS developers supplied the pie3d statement to satisfy this unfortunate, but understandable, desire. I stumbled across this problem by accident. The slices are ordered by slice name, which has no importance for demonstration purposes.

Problem

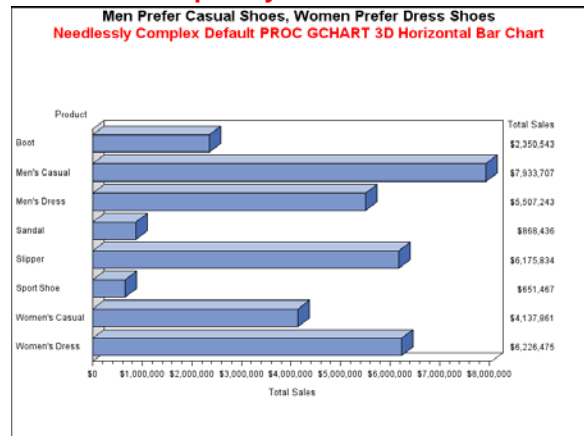


Solution

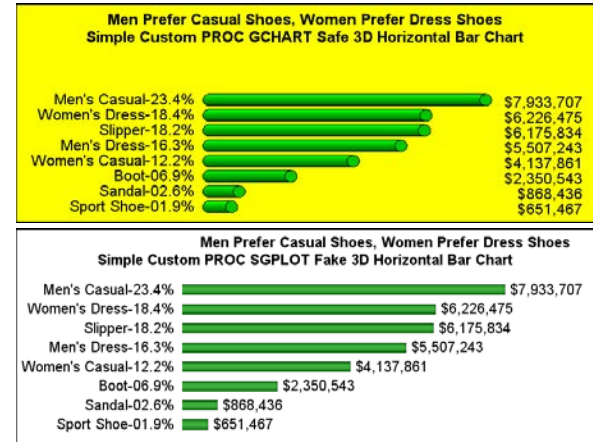


Typical 3D bar charts are not hazardous, just obstructive to easy interpretation. I steadfastly abstained from 3D bar charts until 2004 when I needed to present a proof of concept to executive management, who, it is alleged, are enamored with 3D and other glitzy features. I discovered that I COULD create a safe 3D bar chart using the CYLINDER option in SAS/GRAPH. Compare the 3D bar charts below.

Needless Complexity



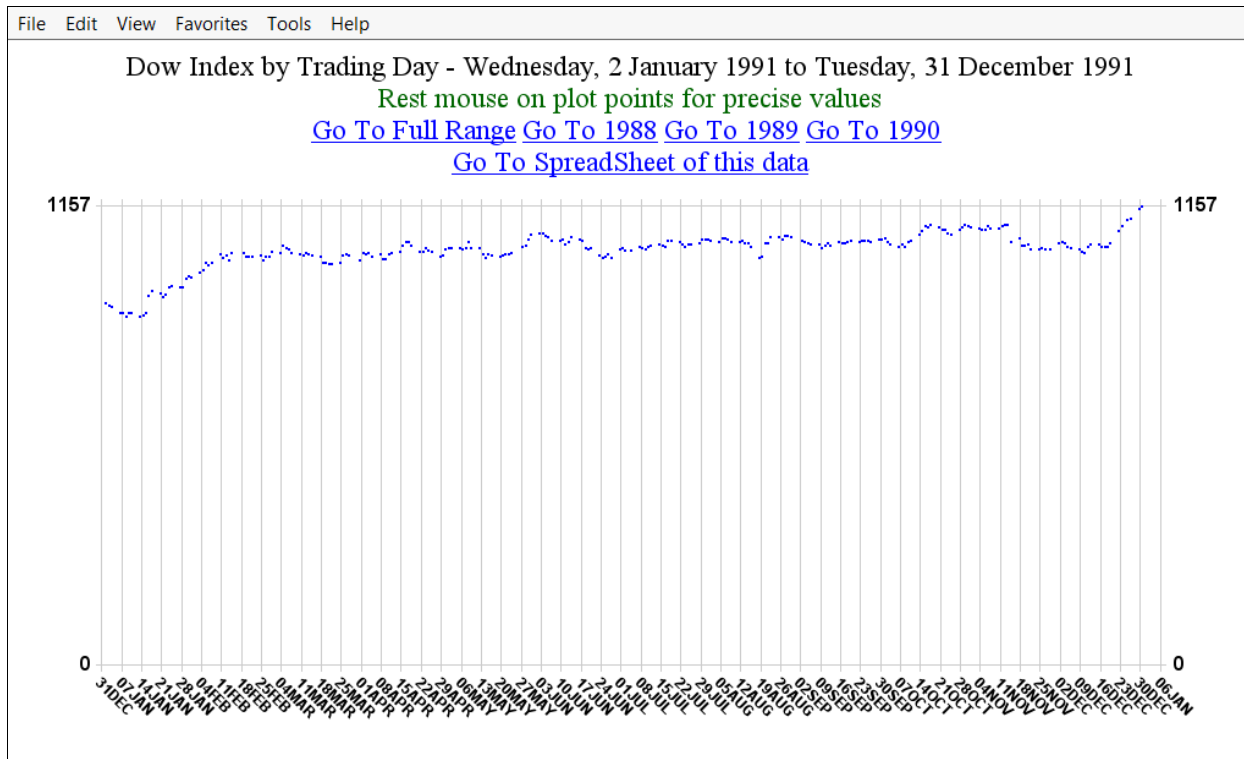
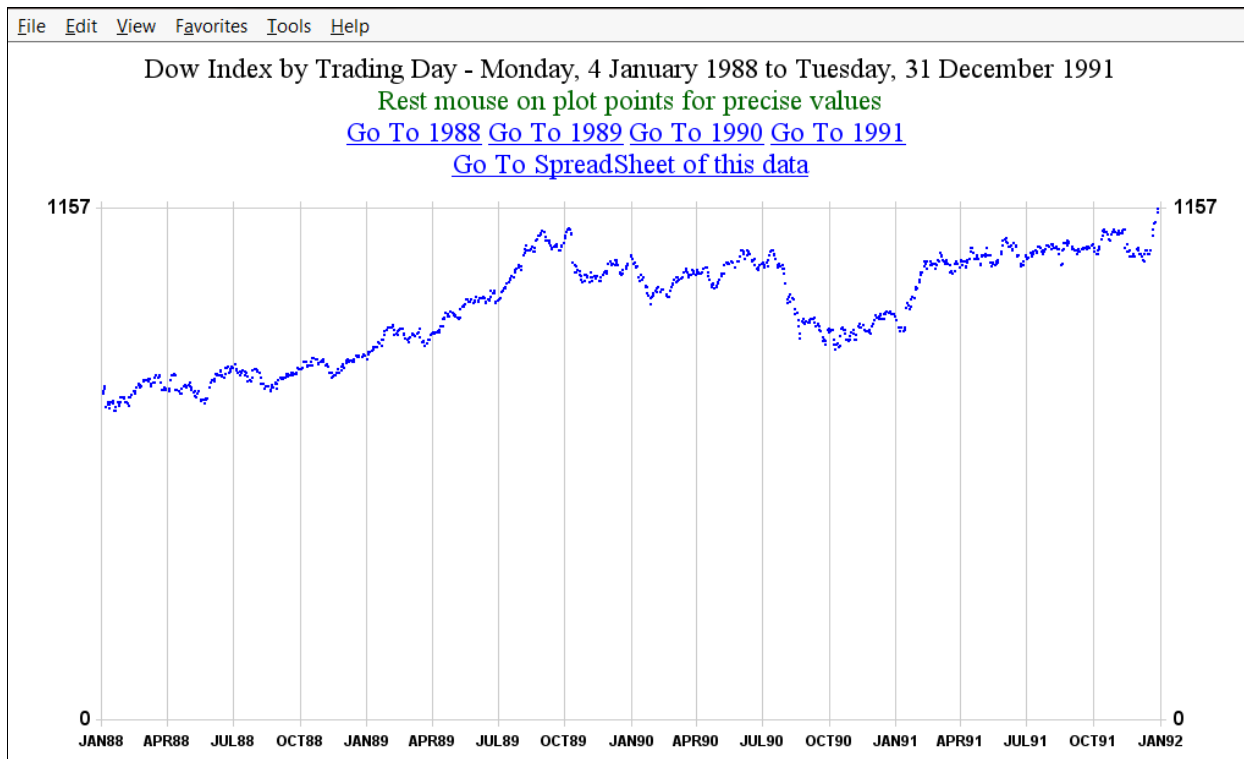
Safe 3D



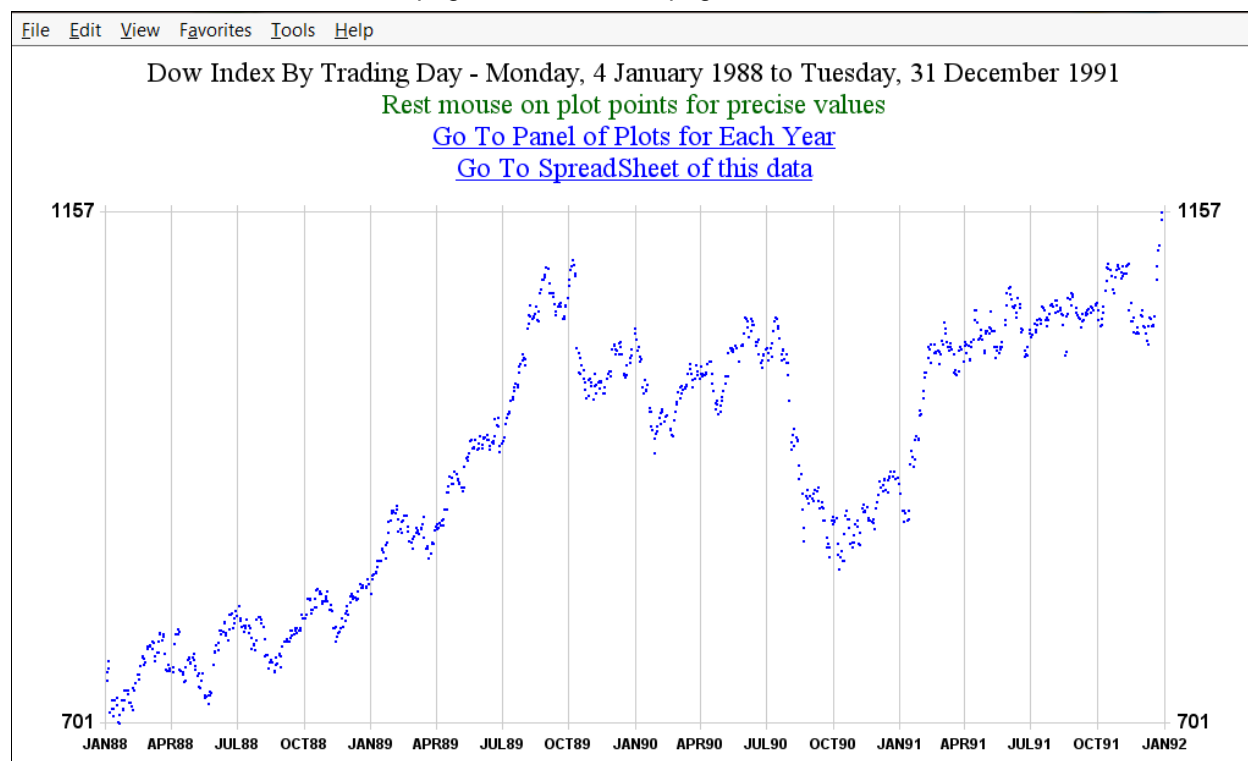
In ODS Graphics, bar charts can use the DATASKIN=PRESSED option to create a simulation of 3D "while maintaining the visual accuracy of a two-dimensional bar chart", but the SAS/GRAPH CYLINDER chart shown above adequately maintains visual accuracy with real 3D. To make the 3D simulation more apparent, the image size needs to be larger, or you need to magnify the page size in your PDF viewer.

Eleven. Link full time series period and its sub-periods.

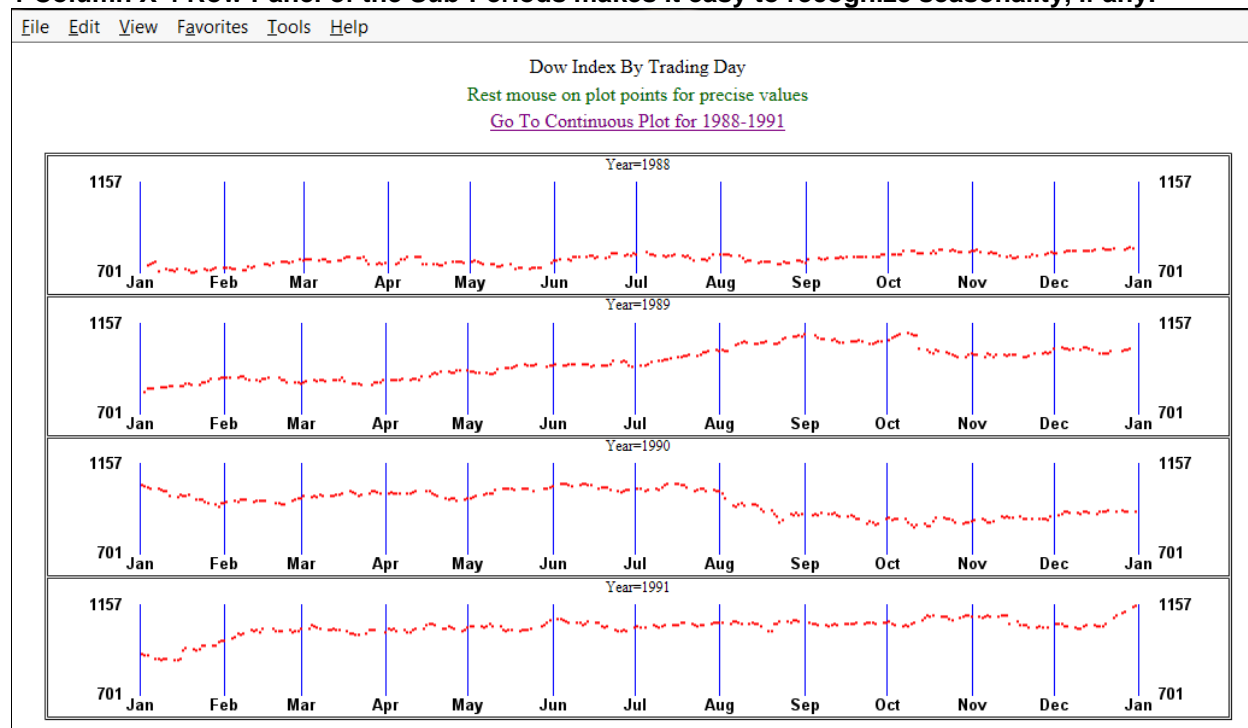
First Method: Interlinks among the Full Period web page and its Sub-Period web pages



Second Method: Full Period web page linked to a web page Panel of the Sub-Periods



1 Column X 4 Row Panel of the Sub-Periods makes it easy to recognize seasonality, if any.

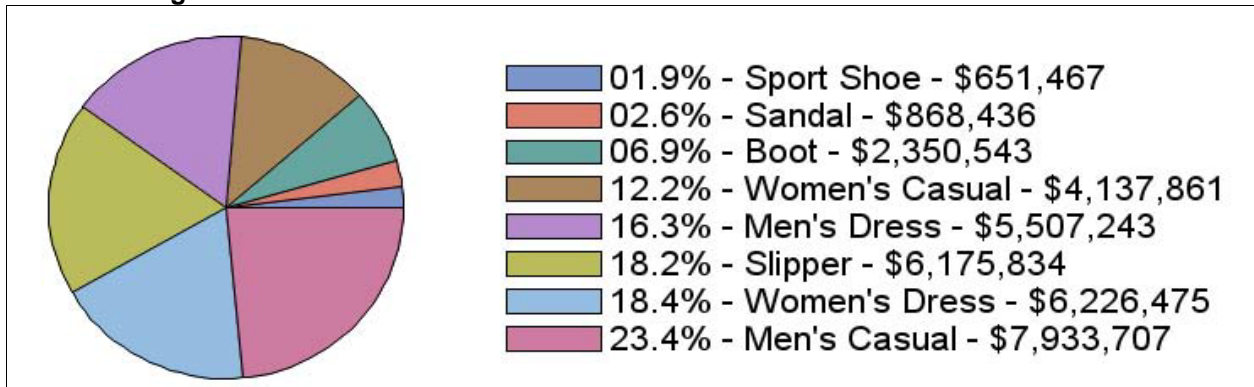


Starting the y axis at zero, as was done for Method 1, would have flattened the yearly plots so much that seasonality, if any, would be hard to perceive.

Twelve. Pie Chart slices should be ordered, and fully explained with category description, value, and percent—either as labels if they fit without collisions, or in a legend.

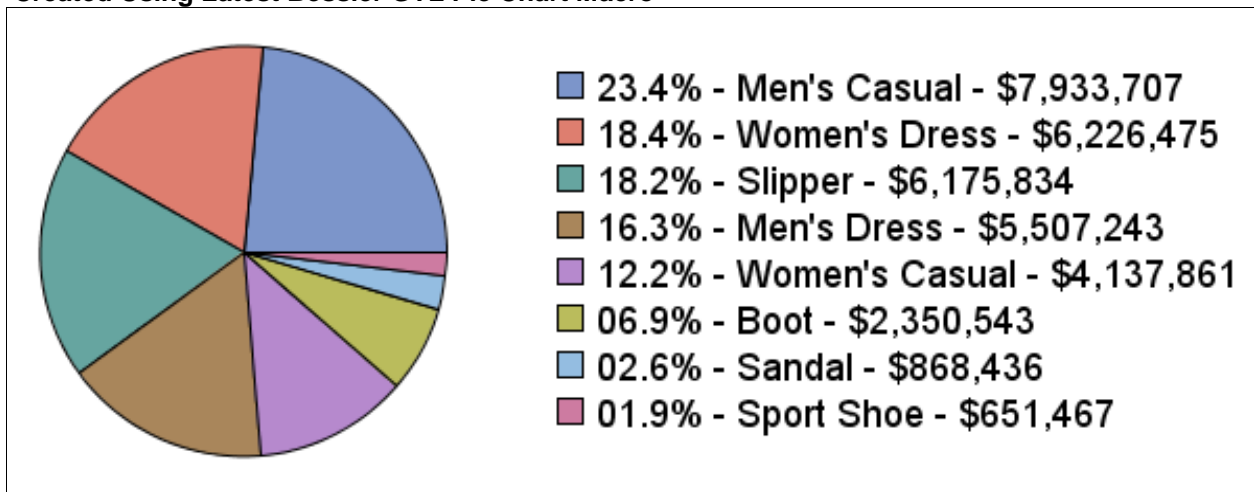
That has been the Bessler Pie Chart Style ever since I first encountered pie chart label collisions. When the SAS/GRAPH developers introduced their legend option, it included only description and value, not percent of whole, which is a key measure of interest inherent to the pie chart concept. The ODS Graphics developers do not support pie charts directly, and require users to resort to GTL (Graphics Template Language) to build them the hard way. Their de-emphasis of pie charts is based on respect for the conclusions from unnecessary academic research, which found that it is easier to visually compare the linear size of bars than the angular size of pie slices. Well, as I have emphasized earlier, reliable inferences or decisions require precise numbers (in this case, from the pie labels or the pie legend), AND I always order pie slices by size. With the right pie chart design and construction, comparing pie slices, both visually, and precise-percent-numerically, is easy and is reliable. It's time for statisticians to accept the (Bessler Style) pie chart as a respectable graphic communication tool, and not to deprecate one of the graphics most frequently seen by regular people every day everywhere. **Ordering pie slices by size** lets you “**Show Them What's Important**”, which is one of my longstanding graphic design principles.

Created Using Bessler SAS/GRAPH Pie Chart Code



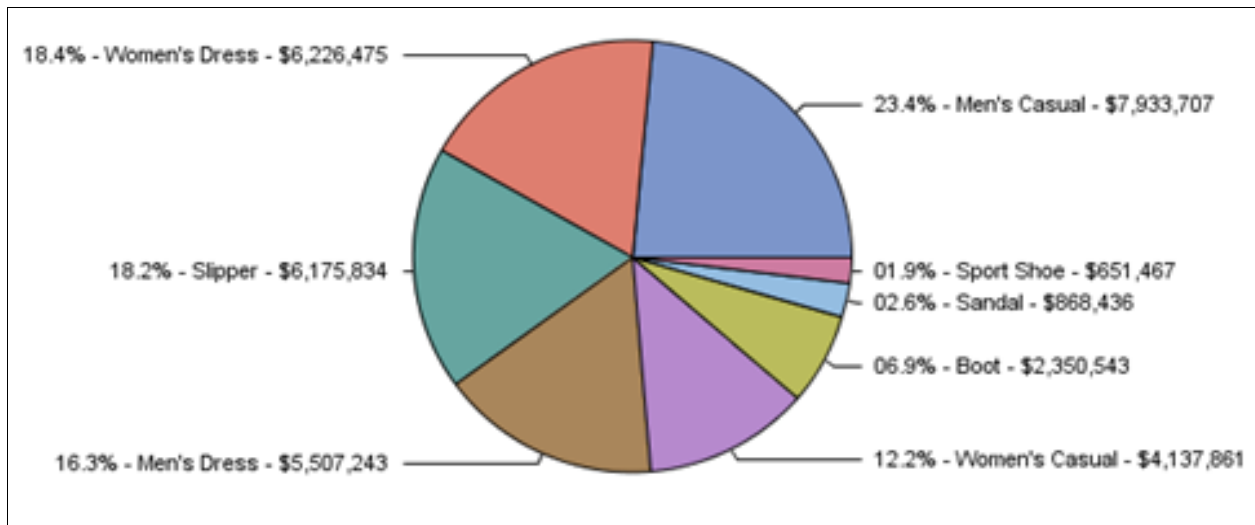
Clockwise ordering of slices above is opposite that of GTL examples, but could have been the same.

Created Using Latest Bessler GTL Pie Chart Macro



With SAS/GRAPH you have absolute control of the size and shape of the legend color samples. In ODS Graphics or GTL, the most you can do is insist that the square color sample be the same height as the sample label. In earlier ODS Graphics versions, there was NO control at all, and those ODS / GTL legends suffer with color samples that are usually very hard to distinguish. Recall my earlier statement that color distinguishability requires lines and text that are thick enough and plot point markers and legend samples that are big enough.

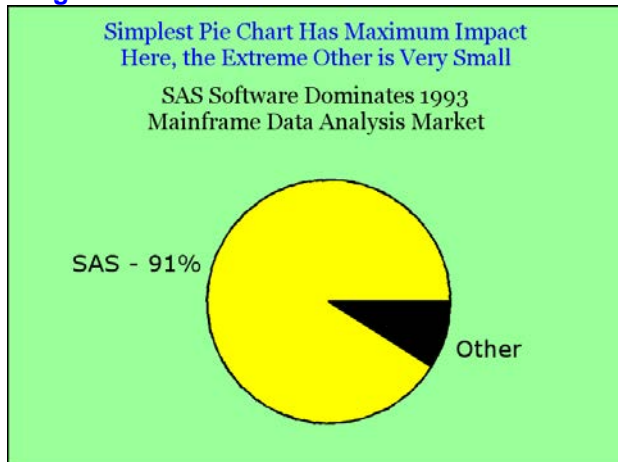
Created Using First Bessler GTL Pie Chart Macro



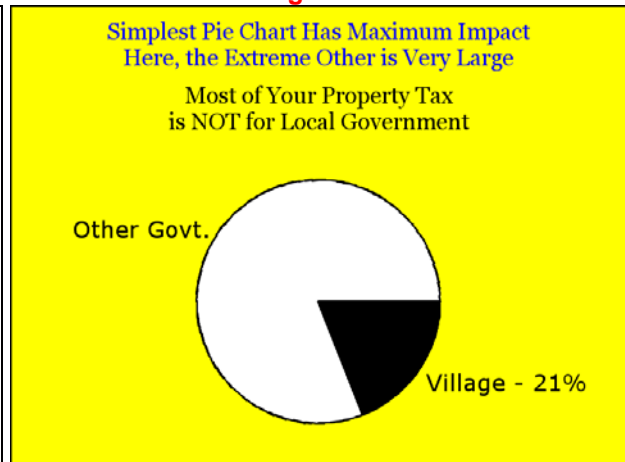
Note: Ordering categories alphabetically is justified, e.g., in a horizontal bar chart of measures for all of the countries of the world, or all 50 states, or all 72 counties in Wisconsin—IF the chart is meant as a convenient lookup tool. However, such a bar chart could be usefully web-linked to a companion bar chart where the categories are ordered by the magnitude of the measure of interest. In a pie chart, it is hard to think of a reason to order slice descriptions alphabetically. If they are so numerous as to make it difficult to find the slice of interest quickly, the pie chart is probably too slice-dense to be useful graphically.

One way that pie chart creators try to deal with the labelling difficulties that might be caused by numerous small slices is to consolidate them into a category called “Other”, but that just invites the question, “What is in Other?” Graphs should anticipate and answer all questions, not create them. Therefore, I had long advocated abstention from use of “Other”. However, in the spirit of “Show Them What’s Important”, sometimes “Other” is a powerful communication tool. Consider what I call “The Extremes of Other”.

Insignificant Other



Viewer Should Investigate Other



I was inspired to present the left-hand pie chart in the early 1990’s. I called it my Pac-Man® Pie Chart. (Pac-Man is a registered trademark of NAMCO BANDAI GAMES INC., Tokyo.) In my estimation, what is in that “Other” just did not matter. I developed the converse pie chart while serving on the Fox Point Village Board. Taxes are paid to the Treasurer, who distributes most of the money to other taxing bodies. Complaints came to him, but taxpayers needed to look more widely if concerned about the size of the bill. (There was a companion table of taxing body, tax amount, one-year change, and five-year change.) I was pleased to eventually see my Pac-Man Pie Chart concept used in my company’s Annual Report.

CONCLUSION

Rather than summarizing the Twelve Ways, here is a reprise, of what you've seen, not necessarily in the sequence of presentation above, in bullet list form:

- Why Graphs Are Necessary. Not Just An Option
- Accelerate/Facilitate Visual Data Insights with Simplicity
- Color Communication Requires Thick Text & Lines, and Big Enough Plot Markers & Legend Samples
- Graph Title as Headline
- **Danger of 3D Pie Chart**, Safe 3D Bar Chart
- **Image + Precise Numbers = Easy, Quick Inference + Reliable Inference**
- Ranking & Subsetting
- Best Use of Vertical Space
- Scrolling Avoidance
- Web-Enablement for Plots
- Web-Enabled Interlinked Subset Plots and the Full-Range Long, Dense Plot
- Dense Plot Linked To Spreadsheet
- Panel of Plot History Subsets
- Facilitation of Seasonality Detection
- Static Overlay Plot with Y-Value Table
- Static Overlay Plot with Point Annotation
- Sparse Line Annotation
- Fully Informing HBar Chart
- InterLinked Subsetted HBar Chart
- A Massively Informing HBar Chart With YAXISTABLEs
- Fully Informing Pie Chart
- HBar Chart As Pie Chart Alternative If Needed
- Pac-Man Pie Chart—The Extremes of “Other”

This paper is a reasonably good presentation of much of the state of my graphic craftsmanship today. The Twelve Ways are a subset of the longer list of design guidelines from my flamboyantly named “Principia Graphika” (a.k.a., “Bessler’s Principles for Communication-Effective Graphic Design: A Tip Sheet for Users of SAS® Graphics Tools or Any Other Data Visualization Software”). Principia Graphika is privately published, and is always being updated, but is not available in the public domain at this time. Presentation attendees receive this quick lookup reference as a Thank You for coming.

I hope you found this paper useful, and enjoyed reading it as much as I enjoyed preparing it, and developing these examples along my graphics journey since 1981, trying to get The Best Results (i.e., the most communication-effective results) out of SAS/GRAPH and now ODS Graphics.

BIBLIOGRAPHY OF MY MOST RECENT RELATED PAPERS

Bessler, LeRoy. 2014. “Communication-Effective Data Visualization: Design Principles, Widely Usable Graphic Examples, and Code for Visual Data Insights.” *Proceedings of the SAS Global 2014 Conference*, Cary, NC, USA: SAS Institute Inc. Available at <http://support.sas.com/resources/papers/proceedings14/1464-2014.pdf>.

Bessler, LeRoy. 2013. “Data Visualization Tips and Techniques for Effective Communication.”, *Proceedings of the Pharmaceutical SAS Users Group 2013 Conference*, Chicago, IL, USA: www.PharmaSUG.org. Available at <http://www.pharmasug.org/proceedings/2013/DG/PharmaSUG-2013-DG10.pdf>.

Bessler, LeRoy. 2013. “Data Visualization Power Tools: Expedite the Easy, Implement the Difficult, or Handle Big Data.”, *Proceedings of the Pharmaceutical SAS Users Group 2013 Conference*, Chicago, IL, USA: www.PharmaSUG.org. Available at <http://www.pharmasug.org/proceedings/2013/DG/PharmaSUG-2013-DG11.pdf>.

For an index to all of my papers, or to request code for any example(s) in this paper, send me an email.

ACKNOWLEDGEMENTS

In the mid-to-late 1980's, when I was supporting three mainframe graphics tools, Alan Paller, who was delivering a seminar for one of the vendors, encouraged me to make myself the graphics how-to resource for all of the users at Miller Brewing Company. In the early 1990's, Chris Potter, Graphics Section Chair at a SAS Users Group International Conference (the predecessor of SAS Global Forum), liked what he saw in my SAS/GRAPH examples when comparing results using the different products, and encouraged me to do what was my first SUGI presentation on communication-effective graphic design and construction. Since my first encounter with SAS/GRAPH, when it was a two-year-old juvenile in 1981, I have repeatedly benefited from the assistance of too many people to mention at SAS Technical Support. Actually, not "too many", but I do not want to inadvertently leave someone out of the long list.

CONTACT INFORMATION

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

Le_Roy_Bessler@wi.rr.com
LeRoy Bessler PhD
Bessler Consulting and Research
Mequon, Milwaukee, Wisconsin, USA
Visual Data Insights™
Strong Smart Systems™

Zum sehen geboren, Born to see,
Zum schauen bestellt. meant to look
— *Goethe*

Ancaro imparo. Still I am learning.
— *Michelangelo*

About the Author: Dr. LeRoy Bessler has presented at software user conferences in the US, Canada, and Europe, on effective visual communication (using graphs, tables, web pages, or color), highly formatted Excel reporting from SAS, custom-developed tools to assist SAS server administrators, users, and managers, and Software-Intelligent Application Development methods to maximize Reliability, Reusability, Maintainability, Extensibility, and Flexibility. His SAS experience includes application development and supporting users, servers, software, and data.

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

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

Visual Data Insights and Strong Smart Systems are trademarks of LeRoy Bessler PhD.

APPENDIX A

The data for this sad story in (abbreviated) graphic form was found at <http://www.motherjones.com/politics/2012/12/mass-shootings-mother-jones-full-data>

I extracted the data from a table and converted it to a CSV file. I can email you the full CSV file upon request. Below is the code for my version of the chart. It is derived from the original handout code of Matange and Heath, with changes, enhancements, and data preparatory processing added.

```
%let Path = C:\! SadStory;

libname sadstory "&Path";

proc import out=SadStory.RawImport
  datafile="&Path.\RecentUnitedStatesMassShootingsHistory_V3.csv" dbms=CSV replace;
  getnames=YES; datarow=2; guessingrows=max; run;

data SadStory.SadStory(drop=Fatalities Injured Total_victims);
length count_fatalities count_injured count_total_victims 3 venue $ 9 gender $ 1;
```

```

length race $ 9; /* truncates "Native American" to "Native Am" */
set SadStory.RawImport;
count_fatalities = input(Fatalities,2.);
count_injured = input(Injured,2.);
count_total_victims = input(Total_victims,2.);
run; /* guessingrows=max prevented truncation of some text variables, but imported some numeric
      variables as text, and imported some text variables over-sized, not right-sized. */
proc sort data=SadStory.SadStory out=SadStorySorted;
by descending count_total_victims; run;

data ToChart;
rename prior_signs_of_possible_mental_i=signs;
length id 3 death_injury $ 7 prior_signs_of_possible_mental_i $ 7;
set SadStorySorted;
id = _N_;
death_injury = trim(left(put(count_fatalities,2.))) || ' / ' ||
               trim(left(put(count_injured,2.)));
run;

proc template;
define style styles.ListingWithNoFrameAndNoYaxis;
  parent=styles.Listing;
  class graphwalls / frameborder=off; /* Remove useless box around the entire graph */
  class graphaxislines / contrastcolor=white; /* Hide the y axis line which
  'yaxis display=(noline)' does not accomplish. This also hides the x axis line,
  but all x axis paraphernalia is suppressed with
  'xaxis display=(noticks novalues nolabel);' */
end; run;

ods noresults; /* Do not send them to Display Manager. */
ods _all_ close; /* In case some non-listing destination is open */
ods listing gpath="&Path\Results" style=ListingWithNoFrameAndNoYaxis; /* Listing is
the default style for the Listing destination. */

ods graphics on / reset=all imagename='SadStory_NoFrameAndNoYaxis_25Mar2015'
width=1280px height=720px;

title1 height=4 PCT
  'The Sad Story of US Mass Shootings with 20 Highest Victim Counts, 1982-2013';
title2 height=4 PCT 'Incidents where weapons were not "Legal" are highlighted in '
  color=red 'red';
footnote1 height=4 PCT bold justify=left
'Data: http://www.motherjones.com/politics/2012/12/mass-shootings-mother-jones-full-data (Updated
2013)';
footnote2 height=4 PCT bold justify=left 'Code & Image: Modification of SAS Global Forum 2014
Statistical Graphics example by S. Matange & D. Heath';

proc sgplot data=ToChart(obs=20) noautolegend;
styleattrs datacontrastcolors=(black red);
hbarparm category=id response=count_total_victims /
  datalabel=count_total_victims transparency=0.2
  datalabelattrs=(size=11 weight=bold color=black) fillattrs=(color=red);
series y=id x=count_total_victims / lineattrs=(thickness=0) x2axis; /* an invisible SERIES line
with thickness=0 must be "drawn" in order to create the label above the bars */
yaxistable year case venue death_injury / label location=inside
  position=left valueattrs=(size=11 weight=bold) labelattrs=(size=11 weight=bold)
  colorgroup=weapons_obtained_legally;
yaxistable signs weapons_obtained_legally gender race Fate_Of_Shooter / label location=inside
  position=right valueattrs=(size=11 weight=bold) labelattrs=(size=11 weight=bold)
  colorgroup=weapons_obtained_legally;
xaxis display=(noticks novalues nolabel);
x2axis display=(noticks novalues) labelattrs=(size=11 weight=bold color=black);
yaxis display=none colorbands=even discreteorder=data
  colorbandsattrs=(color=CXEEEEEE);
label
  case='Incident' venue='Venue' signs='Mental Illness' death_injury='Deaths Injuries'
  count_total_victims='Total Victims' weapons_obtained_legally='Legal Weap- ons?'
  gender='Sex' race='Race' Fate_Of_Shooter='Fate of Shooter';
run;

ods listing close;

```