Volume 20, December 2009

A joint newsletter of the Statistical Computing & Statistical Graphics Sections of the American Statistical Association

Statistical COMPUTING & GRAPHICS

A Word from our 2009 Section Chairs

JOSÉ PINHEIRO
COMPUTING

The cross-disciplinary nature of Statistical Computing is one of its defining features, but may also be a challenge for people attempting to develop a career in the field. Even though statistical computing methods are increasingly driving innovation in key core areas of Statistics and Computer Sciences, researchers and practitioners specializing in Statistical Computing are at times not properly recognized for their contributions and expertise. **Continued on page 2** . . .

ANTONY UNWIN
GRAPHICS

"Description is Revelation"
(Wallace Stevens, later also Seamus Heaney)

Graphic displays should reveal information and it is always a pleasure to come across graphics that do just that. They describe the information in a graphic way that is often more readily understandable than a set of statistics or the results of a model. **Continued on page 2** . . .

Contents of this volume:

Computing Chair
Continued from page 1.

Establishing a professional identity as an expert in Statistical Computing may be challenging in some companies and academic departments, often leading those individuals to develop additional areas of expertise/activity that are more in line with that of their peers.

This problem is of course not new and has long been recognized by our section. Two of the awards presented by the Statistical Computing section, the Student Paper Competition (jointly sponsored with the Statistical Graphics section) and the John M. Chambers Award, are intended to attract and recognize young talent in the field. Now a new award, the Statistical Computing and Graphics Award, is being created to recognize individuals who, over their careers, have made innovative and impactful contributions in the areas of statistical computing, graphics, and/or software. Initially suggested by Deborah Nolan, the new award is jointly organized and sponsored by the Statistical Computing and Statistical Graphics sections, having been enthusiastically endorsed by the executive committees of both sections. It is intended to be presented bi-annually at the Joint Statistical Meetings (JSM), starting next year. Further details can be found in the award announcement, available at http://stat-computing.org/awards/comp-graphics.

On a different topic, the statistical computing program for next year's JSM is shaping up nicely, under the leadership of Thomas Lumley, the 2010 program-chair. In recent years, there have been fewer roundtables and Continuing Education (CE) courses and computer technology workshops sponsored by our section. Those learning opportunities are highly valued by JSM attendees, provide great visibility and increase the understanding of statistical computing, and are a source revenue for our section. Even though it is too late to submit CE proposals for next year's JSM, I strongly encourage our section members to consider submitting proposals for future meetings.

As this is my final column as section chair, I would like to take the opportunity to thank my fellow officers in the Stat Computing and Stat Graphics sections for making this such an enjoyable experience. Luke Tierney will be taking over as chair in January, ensuring that the future of the section will be in excellent hands.

Finally, on behalf of the section I would like to thank Elizabeth Slate, who is rotating out of the Secretary/Treasurer position at the year-end, for her outstanding contributions to the section over the past two years. A warm welcome to Usha Govindarajulu, who will be taking over the financial helm of the section.

José Pinheiro
Johnson & Johnson

Graphics Chair
Continued from page 1.

But graphics is still more of an art than a science and it is surprising how much tastes vary. I have been visiting Heike Hofmann and Di Cook at Iowa State and Di commented on how much progress computer graphics had made in the last twenty years compared to statistical graphics. She is right, and I think it is because we can all generally agree on which software renders a teapot best (to name one classic example), while few agree on which graphics are best for the iris dataset (to name another classic example of a different type). It is difficult to make progress when the goals are not easy to recognise.

If statistical graphics is to develop, there have to be more students working in the field and the Student Paper Competition (stat-computing.org/awards/student/index.html) is one way the section encourages this. The competition is run jointly with the Statistical Computing section and usually there are far more computing entries than graphics entries. Does anyone have any ideas for changing this state of affairs? My own opinion, doubtless both controversial and contentious, is that it's harder to get a graphics paper accepted. A non-graphics paper will probably involve some new technical idea and be mathematically or computationally sophisticated. It looks impressive, even if it may not achieve a lot. A graphics paper is more likely to involve some insight or simplification that makes the reviewer think it is obvious and therefore not worth publishing, even it the idea has direct application. Whatever the reason, we have far too few graphics entries and I exhort you all to persuade students to think of submitting a paper.

Last year's Data Expo was a great success and I hope any of you who attended the JSM took a look at the submissions (you can still see some of them at http://stat-computing.org/dataexpo/2009/posters/). Hadley Wickham provided an excellent and unusually large dataset on flight delays. The contrast in size of this dataset (about 120 million cases) and one from a previous competition was pointed out by Jürgen Symanzik, who found these instructions for the 1983 Data Expo: "Because of the Committee's limited (zero) budget for the Exposition, we are forced to provide the data in hardcopy form only (enclosed). (Sorry!) There are 406 observations on the following 8 variables...". Note that each participating group had to enter the data by hand before doing any analysis! For the coming JSM in Vancouver Hadley has promised another fascinating dataset and I recommend you keep a look out for that. Bear in mind that the main aim of the Expo is not to give prizes (though prizes for the three most interesting entries are awarded), but to gather together different ways of visualizing the same data. Innovative views of a small aspect of the dataset can make an important contribution, there is no need for every entry to include a full analysis of the whole dataset.

And one final important note, our two sections have agreed to establish a new award, the the Statistical Computing and Graphics Award to recognize an individual or team for innovation in computing, software, or graphics that has had a great impact on statistical practice or research (stat-computing.org/awards/comp-graphics/). Whom would/will you nominate?

Antony Unwin
University of Augsburg

Editor's Note

Nicholas Lewin-Koh (Computing)
Andreas Krause (Graphics)

This issue of Volume 20 of our newsletter contains three articles:

- — High-Flying Graphics at the 2009 Data Expo by Rick Wicklin
- — Hadoop for Statistical Analysis and Exploration by Byron Ellis
- — Statistical Graphics! – Who needs Visual Analytics? by Martin Theus

Our section heads provide interesting views in their column "A word from our section chairs", we look back at the highlights of the JSM 2009, and feature a reader response from Naomi Robbins. These articles all provide interesting reading and we hope you enjoy them.

We are always looking for interesting contributions. Please contact us if you have short articles, software, graphs, or anything that you think might be of interest to the ASA Sections for Statistical Computing and Graphics.

We are seeking a new editor for this newsletter to coordinate the graphics section. Serving as newsletter editor is a great opportunity to get to know members of the statistical computing and graphics communities, building up, enhancing, and revitalizing your network. It is also a valuable professional service function to both, the ASA and the Statistical Computing and Graphics Sections.

Nicholas and Andreas will work closely with the new editor to manage the transition smoothly. Interested people should contact Nicholas (lewin-koh.nicholas@gene.com) or Andreas (Andreas.Krause@actelion.com).

Last but not least: Happy New Year to all of you!

Nicholas and Andreas

Highlights from the Joint Statistical Meetings

Highlights from Stat Computing program

The record number of attendees at this year's JSM in Washington D.C. had an impact on the meeting program (including the Stat. Computing program). The larger than usual number of Contributed sessions, eleven Paper and one Poster sessions, gave clear indication of the high degree of interest in statistical computing topics. The six invited sessions focused on computational topics in spatial and temporal data analysis, Bayesian inference, and collaborative research and applications in statistical computing. Together, these 18 sessions covered a broad range of relevant and interesting topics involving statistical computing and attracted a substantial number of JSM attendees. Kudos to the program-chair, Robert McCulloch, the speakers, discussants, session organizers and chairs.

A couple of areas in which the program could be strengthened for future JSM's are Topic Contributed sessions and Continuing Education (CE) events. Besides the traditional, and always excellent, session featuring the winners of the Student Paper Competition, there was a single additional Topic Contributed session, focusing on computing environments and large datasets, sponsored by our section at this year's meeting. The submission of Topic Contributed session proposals for the 2010 JSM is still open, though the end of January deadline is quickly approaching. All are encouraged to submit proposals.

No CE courses or tutorials were sponsored by the Stat Computing Section in this year's JSM, ditto for roundtables. This is a missed opportunity for promoting and creating greater awareness about Statistical Computing, as well as for generating revenue for the Section. The submission deadline for CE courses has passed, but there is still time for submitting proposals for Computer Technology Workshops and roundtables.

Perhaps influenced by the record JSM attendance, this year's Student Paper Competition had a record number of entries. The large number of high quality submissions made it difficult for the judges to select just four winners, as traditionally done. Upon their request, the Stat Computing and Stat Graphics executive committees agreed to have five winners for this year's competition.

As usual, the Monday night Stat Computing and Stat Graphics mixer was one of the highlights of the meeting, at least for the crowd who attends it every year. A great opportunity to meet old friends and make new ones, while enjoying good food, getting the latest news from both sections, and, with a little bit of luck, winning one of the many door-prizes that are given out. On the latter, a big thanks to the vendors, too many to list here, who, year after year, generously donate books, software, gadgets, etc, for the prizes.

José Pinheiro
Johnson & Johnson

Looking Back — the Graphics program at JSM 2009

As always there was so much going on at the JSM, so many talks, so many discussions, so many contacts, that it's hard to remember what actually took place. One thing I can remember was the annual mixer. For once in my long years of JSM experience I was not looking forward to it. Having part of the responsibility for organising an event is always a sobering influence. Fortunately everyone seemed to enjoy themselves, there were a lot of raffle goodies to distribute, and the main event, honouring our prizewinners, went off very well.

In the conference itself the section cosponsored seven invited sessions, two contributed sessions, a continuing education course (Martin Theus and Simon Urbanek on their super book "Interactive Graphics for Data Analysis"), two roundtable lunches and, most importantly, the Data Expo on flight delays. The Data Expo had a prominent position in the main conference building and was seen by a lot of people. (By the way, if you want to refresh your memory on what all these events were in detail, you can still access all the information from the JSM 2009 program on the web. Just enter under Search by sponsor "Section on Statistical

Graphics".) I wish I could say that I went to all of these sessions and could tell you all about them. In fact, thanks to the regular scheduling rule that there are always some sessions that overlap that shouldn't, I was giving an Introductory Overview Lecture on Visualising high-dimensional data just as one of our invited sessions "Data Display for Large Complex Data Sets" was taking place. The sessions I did manage to get to were very good and we should offer our congratulations to our Program Chair, Steve MacEachern, for his coordinating work and to all the organisers and speakers for their contributions. It would be nice to see more contributed paper sessions (our sister section Statistical Computing has far more of those), but Statistical Graphics is a tricky subject for research and publication, and the short time available for each paper in contributed sessions is not ideal. Do think about it in the future, though.

After the JSM in 2007 Hadley Wickham put up a lot of information about the meeting on the section's website (stat-graphics.org/graphics/). We should rekindle this shortstanding tradition and put stuff up about JSM 2009. We encourage everyone to send their talks, photos, related materials or URL links to Hadley Wickham <h.wickham@gmail.com> for inclusion in the web site.

Antony Unwin
University of Augsburg

High-Flying Graphics at the 2009 Data Expo

Rick Wicklin

Introduction

The Data Expo is a biannual poster session usually sponsored jointly by the ASA Sections on Statistical Graphics and Statistical Computing. The purpose of the poster session is to distribute an interesting data set to many researchers and to challenge them to use statistical graphics to describe and visualize the data concisely on a single poster. The session is always well-attended, and it helps the Sections highlight the importance of statistical graphics in data analysis.

This year's Data Expo was organized by Hadley Wickham, who assembled a truly massive set of data from the Research and Innovative Technology Administration (RITA) which coordinates the U.S. Department of Transportation (DOT) research programs. The data (available from `http://stat-computing.org/dataexpo/2009/`) consist of 123 million records of U.S. domestic commercial flights between 1987 and 2008. For each flight there is information about 29 variables, including the following:

- Dates: day of week, date, month, and year
- Arrival and departure times: actual and scheduled
- Origin and destination: airport code, latitude, and longitude
- Carrier: American, Aloha Air, . . ., US Air

It is fascinating to see the varied approaches and graphical displays presented for these data. You are invited to browse the electronic version of the posters at `http://stat-computing.org/dataexpo/2009/posters/`.

This article describes several graphs in my poster, which was joint work with Robert Allison (6). The poster graphically presents ways in which flight delays and cancellations vary in time, among airports, and among airline carriers. The article also describes graphical methods and features of the data that elicited the most comments from visitors to the Data Expo. For me, the discussions with colleagues during and after the poster session were the most gratifying part of creating the poster.

Data Reduction

When presenting data graphically, it is important to be able to quickly convey the main features in the data. But how can you summarize 21 years worth of data on the delays or cancellations of 123 million flights?

The DOT defines a departing flight as "delayed" if it departs more than 15 minutes after its scheduled departure time. The 15 minute window is also used for defining when an arriving flight is delayed.

For these data, I found it useful to reduce the size of the data by computing a descriptive statistic such as the mean length of a delay or the percentage of flights delayed. These statistics are calculated over an appropriate aggregation unit such as a date, a year, an airport, or a carrier. This often proved to be an important first step in understanding and conveying gross trends in the data. All but two graphs in my poster displayed descriptive statistics, rather than the raw data.

For example, one quantity displayed in several graphs is the "percentage of flights delayed" (PFD) for a given day. (This is a standard quantity measured and reported by the DOT; see http://www.bts.gov/.) If a certain day has 30,000 flights and 6,000 of those are more than 15 minutes late, then the PFD for that day is 20%. Plotting the PFD for each day instead of the original data results in a significant reduction in the volume of data, but at the usual risk of losing details that might be important.

Temporal Effects: A 21-Year Summary

The main graphical technique we used to present how the PFD varies over 21 years is a *calendar plot*, shown in Figure 1 for five years of the data. For each year, there are seven rows that represent the days of the week and usually 53 columns that represent each week in the year. The color of each day represents the value of the variable for that day. The earliest reference I have seen for a calendar plot is Mintz, Fitz-Simons, and Wayland (3). A SAS macro for computing the plot was presented in Zdeb and Allison (8).

The calendar plot is a choropleth map: each year is a "country," each month is a "state," and each day is an "county." By thinking of the calendar plot as a choropleth map, we can use ideas from the cartographic literature to guide the design of the plot. For example, we can follow Pickle et al. (5) and choose a color for each day as determined by the quantile of the PFD for each day.

In Figure 1, the colors are determined by quintiles of the PFD for all days in the five-year time period. The colors are chosen to suggest a traffic light color scheme: the first quintile of the data is colored green for "go"; the second and third quintiles are yellow and pale orange for "caution"; a darker orange and red are used for the upper quintiles to signify extreme delays. The actual colors are based on a diverging color scheme from `ColorBrewer.org` (1). This web site is a valuable resource for creating color schemes because Brewer's schemes have the important property that each color is equally-perceived: no one color catches the eye and draws the viewer's attention. The colors are also designed to be distinguishable to the color blind.

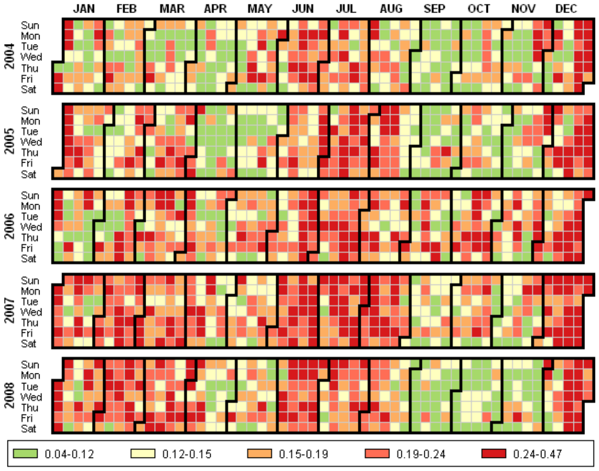


Figure 1: Calendar Plot for Percentage of Flights Delayed (2004–2008)

A few aspects of the data are apparent by looking at the color patterns of columns: the PFD is relatively low in the spring and fall, whereas summer and the last two weeks of December have many days with a high PFD. The PFD in January–March can be quite variable, presumably due to winter storms. If you average the values of the calendar plot for each week (that is, take the average of each column), you obtain the scatter plot in Figure 2 which plots the average weekly PFD versus the week of the year. A loess smoother (adjusted for periodicity in the data) is overlaid on the plot so that it is easier to see that, on average, the PFD is high during the summer and Christmas seasons but low during the spring and autumn. In a paneled layout, this figure can be placed underneath the calendar plot, similar to the arrangement used by Peng (4).

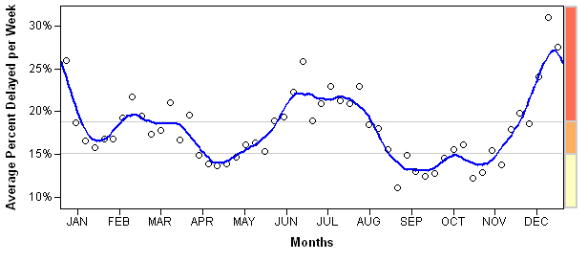


Figure 2: Average Weekly Percentage of Flights Delayed (2004–2008)

A calendar plot for the percentage of flights canceled (PFC) can be similarly constructed and shows similar features. However, the most striking feature of the calendar plot for canceled flights is related to the terrorist attacks on September 11, 2001, as shown in Figure 3. In the two weeks prior to 9/11, an average of 2.3% of flights were canceled. By the middle of October, 2001, the average PFC was down to 1% of flights, due in part to a 17% reduction in the number of scheduled flights per day. The daily PFC remained low until 2004.

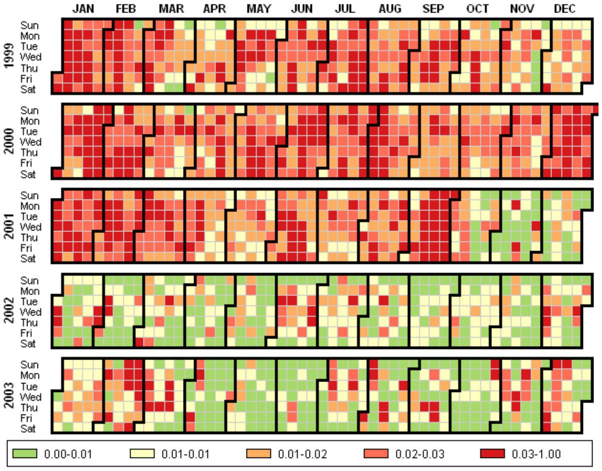


Figure 3: Calendar Plot for Percentage of Flights Canceled (1999–2003)

As suggested by Pickle et al. (5), it is a good idea to plot the distribution of the variable that you are using to color the map. Figure 4 shows the distribution of the PFC variable for the five years shown in Figure 3. (The distribution over the full 21 years is similar, but was not included in my poster due to space considerations.) This distribution exhibits a long tail.

Following Pickle et al. (5), there is a color bar beneath the density plot and the horizontal axis is truncated at the 99th percentile of the data. (The actual maximum is indicated in the graph.) The color bar shows the values that correspond to the colors used in the calendar plot; this gives the viewer a visual impression of the varying lengths of the quantiles. An investigation of the days with the most cancellations reveals that they are primarily related to the 9/11 attacks or to extreme weather events.

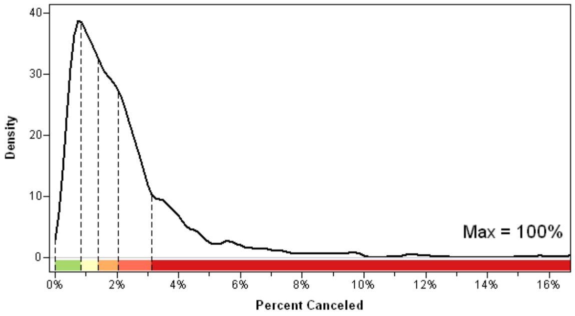


Figure 4: Distribution of the Percentage of Flights Canceled (1999–2003)

In summary, you can use the calendar plot to reveal features and seasonal trends for 21 years of airline delays and cancellations.

Carrier Effects: Multivariate Visualization of Time Series

The calendar plot is useful for showing how a single quantity varies during long periods of time. However, it is not so useful for comparing multiple quantities that vary in time. For example, suppose you are interested in comparing the mean length of delays for flights among the different carriers. Are there some carriers that are almost always on time, while others are habitually late? To investigate this question, we constructed a multivariate time series plot as described by Peng (4).

The plot is displayed in Figure 5 for 20 major airline carriers during 2007. That year is chosen because there was a major winter storm (the “Valentine’s Day Blizzard”) that affected the entire eastern

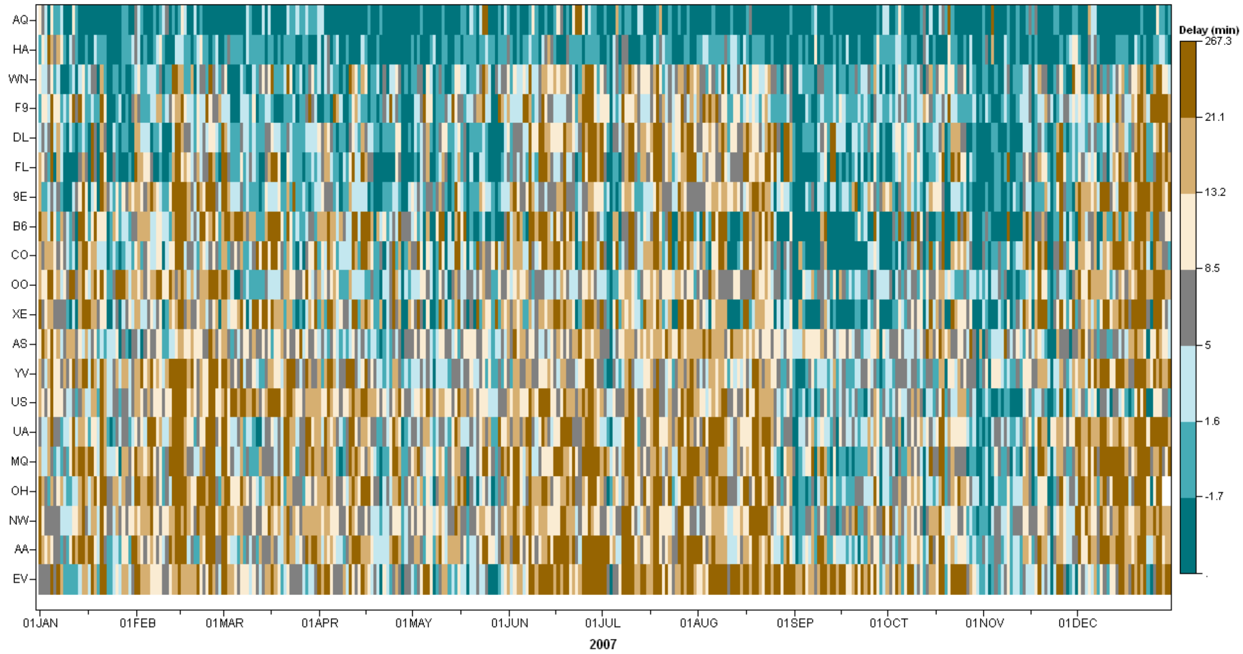


Figure 5: Multivariate Time Series of Mean Delay for Each Carrier (2007)

half of North America and paralyzed transportation in all forms (7). The effect of the storm is visible as a dark brown vertical line in the middle of February.

Following Peng and a suggestion from John Sall (personal communication), the carriers are sorted according to the mean delay for all flights during the year. This makes it easy to see that Aloha Airlines (AQ) and Hawaiian Airlines (HA) have superior on-time performance, presumably because of short flight times and fair weather! The next best performing airlines in 2007 according to this metric were Southwest (WN), Frontier (F9), Delta (DL), AirTran (FL), Express (9E), and JetBlue (B6).

Again, it is useful to use ideas from choropleth maps to choose colors for these data. For this plot the colors are based on seven quantiles and the colors are based on a blue-brown color ramp developed Cindy Brewer and used successfully in Pickle et al. (5). (This double-ended color scheme does a good job of simultaneously depicting both high and low values.) The blue shades correspond to days for which the mean daily delay for a particular carrier was less than five minutes. The brown colors correspond to days and carriers for which the mean delay was between 13 and 21 minutes (light brown) or more than 21 minutes (dark brown). The distribution of mean daily delays is shown in Figure 6. You can see that the median value of the distribution corresponds to a mean delay of seven minutes, but that the distribution has a long tail.

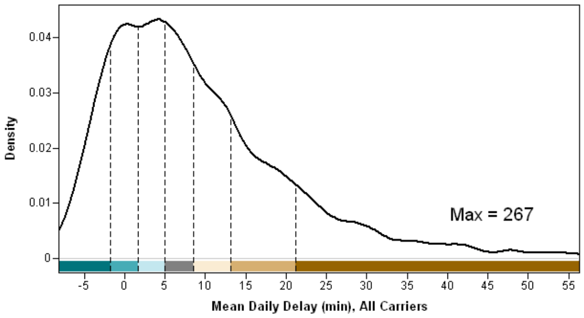


Figure 6: Distribution of the Mean Daily Delay for Each Carrier (2007)

A box plot (not shown) of the mean delays for each day of the year could be placed on the right side of Figure 5 as suggested by Peng (4) and used in a cartographic context by Carr, Wallin, and Carr (2). The box plot enables you to see the within-carrier variation of the daily means throughout the year and conclude, for example, that even though the delays for Frontier airlines are usually low, there is a large amount of variation in the daily means.

Two features of this multivariate time series plot are striking. The first is the "outlying" behavior of Aloha and Hawaiian Airlines. The second is the well-defined vertical bands of brown (long delays) for most carriers in February, in the summer, and in the latter half of December. This is in marked contrast to the faint blue bands in April and May, and the heavier blue bands in September through early November. These qualitative trends can be visualized further by smoothing the time series, similar to Figure 2. The graph (not shown) suggests that the mean delays in the spring and fall are shorter than the mean delays during the summer and during the latter half of December, and that this tends to be true for most carriers (although Atlantic Southeast Airlines (EV) seems an exception to the rule).

In summary, the multivariate time series plot enables you to see gross similarities and differences between the mean daily delays of the 20 carriers.

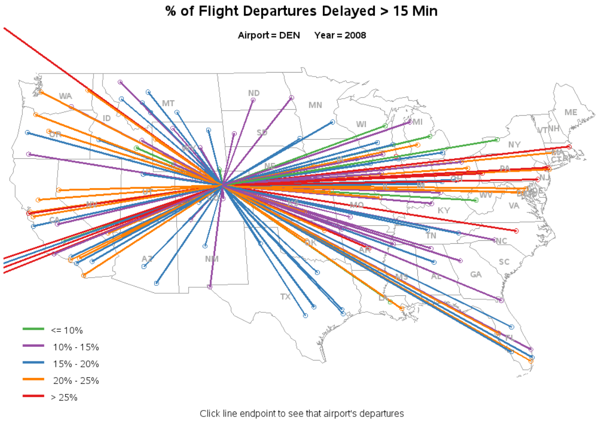
Airport Effects: Interactive Charts

You can create a graph similar to Figure 5 for airports. In fact, you can create two such graphs: one for flights that originate at the airport, and another for flights for which the airport is the destination. However, in my poster I chose to present the relationship between airports and delays by using interactive and dynamically linked graphics.

The interactive graphic takes the form of a map of the continental U.S. with a single airport selected, as shown in Figure 7. I nicknamed this graph the *splay plot* because it reminds me of splayed fingers. All flights that originate from the selected airport are grouped according to their destination and are colored according to the PFD for each destination. Note that some routes are consistently on time, whereas others are frequently delayed.

In the interactive version of Figure 7, you can click on the splay plot to select a different airport. The plot uses SAS/Intrnet® software to detect the click and to run a SAS program to create a splay plot for the new airport. (See my poster for an example.) So that the colors on the splay plot do not change from one airport to another, the colors are hard-coded into five categories: less than 10% of flights delayed (green), between 10% and 15% of flights delayed (purple), and so on, up to more than 25% of flights delayed (red).

The splay plot could be improved in several ways. It could include a density estimate similar to Figure 4 that shows the distribution of the coloring variable for each route. At the Data Expo, Simon Urbanek suggested using semitransparent lines in Figure 7.



Click line endpoint to see that airport's departures

Figure 7: Percentage of Flights Delayed (Denver, 2008)

A second approach to presenting the relationship between airports and delays is to use dynamically linked graphs. Figure 8 shows this approach in the SAS/IML® Studio application (formerly named SAS® Stat Studio). This figure shows the PFD for each major airport and for each year. The linked graphics enable you to select observations according to certain criteria. For example, you can select airports with a large PFD by selecting markers in the scatter plot in the upper-left plot, as shown in the figure. Or you can select certain airports or certain years or both and examine the PFD for those selected observations.

Interactively exploring Figure 8 revealed a difference between "hub" and "non-hub" airports. Hub airports appear to give preferential treatment to inbound flights. An example of this appears in the scatter plot shown in Figure 9 where the percentage of inbound flights that are delayed

(PIFD) is plotted against the percentage of outbound flights that are delayed (POFD) for all major airports and all years. The large blue markers correspond to flights delayed at Chicago O'Hare during 1997–2008. Note that for each year the PIFD is less than the POFD. (The diagonal line is the identity line.)

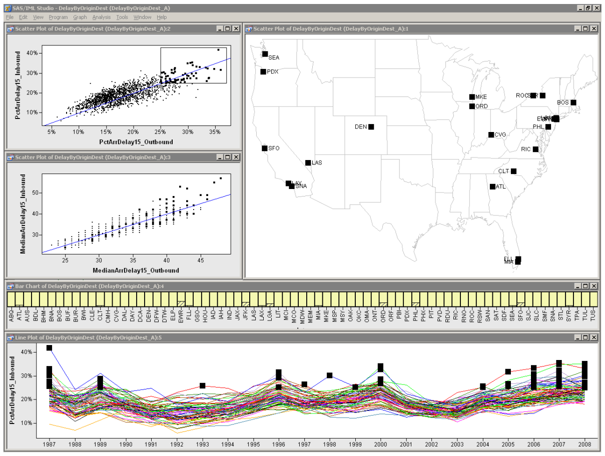

SAS/IML Studio - DelayByOriginDest (DelayByOriginDest_A)
File Edit View Program Graph Analysis Tools Window Help
Scatter Plot of DelayByOriginDest (DelayByOriginDest_A):2
PctArrDelay15_Inbound
40%
30%
20%
10%
5%
10%
15%
20%
25%
30%
35%
PctArrDelay15_Outbound
Scatter Plot of DelayByOriginDest (DelayByOriginDest_A):1
SEA
PDX
SFO
LAS
LAX
SNA
DEN
MKE
ORD
CVG
ROC
BOS
EWR
PHL
RIC
CLT
ATL
FLL
MIA
Scatter Plot of DelayByOriginDest (DelayByOriginDest_A):3
MedianArrDelay15_Inbound
50
40
30
25
30
35
40
45
MedianArrDelay15_Outbound
Bar Chart of DelayByOriginDest (DelayByOriginDest_A):4
ABQ ATL AUS BDL BHM BNA BOS BUF BUR BWI CLE CLT CMH CVG DAL DAY DCA DEN DFW DTW ELP EWR FLL GSO HOU IAD IAH IND JAX JFK LAS LAX LGA LIT MCI MCO MDW MEM MIA MKE MSP MSY OAK OKC OMA ONT ORD ORF PBI PDX PHL PHX PIT PVD RDU RIC RNO ROC RSW SAN SAT SDF SEA SFO SJC SLC SMF SNA STL SYR TPA TUL TUS
Line Plot of DelayByOriginDest (DelayByOriginDest_A):5
PctArrDelay15_Inbound
40%
30%
20%
10%
1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008

Figure 8: Relationships between Delays, Airports, and Years (1987–2008)

Philip Easterling, a former analyst with a major US airline, informed me after the Data Expo that this is deliberate: hub carriers tend to hold outbound flights when an inbound connection is late, so that a single late inbound flight can result in dozens of outbound delays. For this same reason, air traffic controllers give preferential treatment to landing inbound flights at hubs.

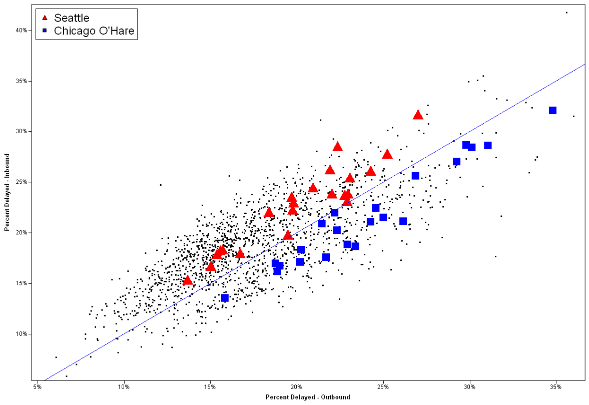


Figure 9: Differences between Hub and Non-Hub Airports (1987–2008)

In contrast, the non-hub airports tend to try to get outbound flights off the ground on time so that they can arrive at the hubs on time. For the specific example of Seattle, shown in Figure 9, Easterling explained that arriving flights are often delayed due to clouds and fog, but that the ground crews work hard to "turn around" these planes so that they depart on time.

Conclusions

The data set used for the 2009 Data Expo is exceedingly rich. It would be easy to fill up *two* posters with graphs of these data! This article briefly describes a few graphs that elicited the most comments from viewers of my poster. The two main approaches used in this article are (1) to reduce the dimensions of the data by plotting descriptive statistics aggregated over dates, years, airports, or carriers; and (2) to use design principles from cartographic research to aid the visualization of multivariate time series. These techniques enable us to create graphs that exhibit relationships among a dozen variables in a data set with 123 million observations.

The ASA promotes poster sessions as a way to increase the interaction between presenters at JSM and their colleagues. The 2009 Data Expo succeeded admirably in that task. If you enjoy making graphs and analyzing real-world data, I hope to see you at the next Data Expo.

Acknowledgements

The graphs in this paper were created by using SAS/IML Studio 3.2, except for Figure 7 which was created by using SAS/GRAPH® software. I thank Robert Allison for help in preparing the original poster and for introducing me to calendar plots. Bob Rodriguez provided helpful comments on the graphics. Peter Borysov helped to write programs in SAS/IML Studio for graphs in the poster that were originally created with SAS/GRAPH software. Linda Pickle reviewed a draft of this article and made helpful suggestions. Finally, I thank the management at SAS, particularly Joe Hutchinson and Radhika Kulkarni, for supporting my participation in the Data Expo.

Bibliography

[1] Brewer, Cynthia A. 2006. ColorBrewer. URL `http://www.ColorBrewer.org`.

[2] Carr, Daniel B., Wallin, John F., and Carr, D. Andrew. 2000. Two new tools for epidemiological applications: Linked micromap plots and conditioned choropleth maps. *Statistics in Medicine*, 19:2521–2538.

[3] Mintz, D., Fitz-Simons, T., and Wayland, M. 1997. Tracking air quality trends with SAS/GRAPH. In *Proceedings of the Twenty-Second Annual SAS, Users Group International Conference*, Cary, NC: SAS Institute Inc.

[4] Peng, Roger. 2008. A method for visualizing multivariate time series data. *Journal of Statistical Software*, 25(1):1–17. URL `http://www.jstatsoft.org/v25/c01`.

[5] Pickle, L. W., Mungiole, M., Jones, G. K., and White, A. A. 1996. *Atlas of United States Mortality*. National Center for Health Statistics, Hyattsville, MD. URL `http://www.cdc.gov/nchs/products/other/atlas/atlas.htm`.

[6] Wicklin, Rick, and Allison, Robert. 2009. Congestion in the sky: Visualizing domestic airline traffic with SAS, software. Poster presentation, Joint Statistical Meetings. URL `http://stat-computing.org/dataexpo/2009/posters/`.

[7] Wikipedia. 2009. February 2007 North America winter storm. URL `http://en.wikipedia.org/wiki/February_2007_North_America_Winter_Storm`.

[8] Zdeb, Mike, and Allison, Robert. 2005. Stretching the bounds of SAS/GRAPH software. In *Proceedings of the Thirtieth Annual SAS, Users Group International Conference*, Cary, NC: SAS Institute Inc.

Rick Wicklin
SAS Institute Inc.
`Rick.Wicklin@sas.com`

Hadoop for Statistical Analysis and Exploration

Byron Ellis

The explosive growth of data available from a variety of sources, particularly social behavior data from the movie watching habits from the Netflix Prize to the millions of posts available from Twitter every day. In more traditional areas of research, such as biology, the introduction of high throughput biology has also increased the amount of available data significantly.

While large , relatively inflexible, data warehousing solutions have long been available the last five years has seen the development of new inexpensive data analysis and storage platforms specifically designed to handle these enormous data sets. One such system is Hadoop, originally developed at Yahoo! and now maintained as an Apache project.

Originally part of the Nutch web indexer, Hadoop provides a distributed storage system and map-reduce processing model based on descriptions of the Google File System (GFS) and MapReduce processing platform published by Google in a series of papers. Using only commodity hardware and networking, it allows organizations to construct distributed, multi-terabyte storage facilities coupled to a processing platform structured to take advantage of the storage system's characteristics.

The processing model is fairly strict in the Hadoop environment and heavily informed by the features of storage model, so we begin with a brief discussion of the Hadoop architecture. From there we will present examples of several use cases. The first case is the most common, data filtering and summarization for downstream analysis and presentation. In the second example, we begin to take the analysis of very large datasets into the Hadoop cluster itself. Finally, we also consider how we might use Hadoop to visualize very large datasets with a simple plotting example.

Architecture and Processing Model

Hadoop is deployed on commodity grid computing environments: a number of individual compute/data nodes controlled by a "master" node with standard (gigabit) networking. Scaling the compute environment is easy, compute/data nodes may enter and leave the system at will and none of the compute hardware is considered to be reliable (at the time of writing the master node is considered to be reliable, but efforts are being made to provide redundancy for master nodes as well) and clusters upwards of 4,000 compute nodes have been documented.

Unlike most grid computing environments, Hadoop does not make use of a standard network filesystem such as NFS. Instead it provides its own implementation of a distributed and replicated filesystem. The storage component of the filesystem is hosted on a set of data nodes, which are usually the same set of nodes that are used for computation. Filesystem metadata, directory and "inode" management, is managed by a single "namenode" that usually runs on one of the grid computing environment master nodes. The primary task of the "namenode" is the manage the mapping of files to blocks and the ordering of blocks withing a file as well as the mapping of blocks to the data nodes.

This storage architecture has several implications. First, operations on the directory structures are not parallel and will have performance somewhat worse than a local filesystem. Second, this filesystem will prefer a small number of large files over a large number of small files. This is reinforced by the typical block sizes—64MB. The advantage is that once a file has been located, reads can be done in a highly parallel fashion.

The Hadoop processing model should be familiar to any `R` programmer who has ever used `tapply`. Like the distributed file system, each compute node runs a local compute manager that is designed to be robust to failure—if one piece of a compute job dies it is rescheduled on a different node—controlled by a management daemon on the master node, equivalent to the data node daemon and the name node daemon. In fact, these will generally be the same set of machines (except perhaps the job control node and the name node).

The processing model consists of two distinct phases, a "map" phase and a "reduce" phase. The first map phase takes advantage of the performance characteristics of the distributed file system by first transporting the map executable to each compute node and then executing parallel reads against each of the blocks of the input data files (i.e. the code is moved to the data rather than the other way around). When the data nodes and the compute nodes are shared hardware the map phases are arranged such that data is kept to the local disk as much as possible. As a result,a given map task may be given several different blocks out of order that happen to reside on the same machine. This lack of ordering of input records is an important feature of the parellelism—it is assumed in the model that a map operation could be performed in parallel for each input record simultaneously with a complete lack of synchronization.

The map phase itself is usually quite simple. Given input tuples, often simply lines of text, the map phase produces a key value pair (k, v) where both the key and the value can be arbitrarily complex data structures. The only restriction is that the keys must be comparable and should be sortable. This key is used to distribute key-value pairs to a specific "reducer" task. Once all of the key-value pairs have been generated each reducer will their assigned keys and then "reduce" the values for each key in this sorted order (note that it is the keys not the values which are sorted). It should be noted that the partition, sort and grouping functions may all be overridden to control this behavior as we will see in the examples.

The reduce itself operates on an iterator that allows the task to pass over each value. Prior to the latest versions of Hadoop there was a restriction that the reducer was only allowed to pass over each value in the iterator exactly once. With version 0.20.0 it seems that this restriction has been lifted so that iterators now support mark/reset functionality. There are no restrictions on what is emitted by a reducer and it also has general access to the filesystem so it can be used for its side effects as we will see in the final example.

In terms of `tapply`, the map phase corresponds to the `INDEX` column while the reduce phase corresponds to the `FUN` parameter. To put it another way, for those familiar with the SQL language, the map phase somewhat like the `GROUP BY` clause, while the reduce phase is like the `SELECT` clause. Using some more advanced techniques it is even possible to implement various kinds of `JOIN` clauses, though that is beyond the scope of this article.

Example: Data Wrangling

The most common use for Map-Reduce platforms is so-called "data wrangling." These tasks are essentially reorganization and summarization of large raw datasets for further analysis or presentation. This use case is very similar to using a relational database system to manage the raw data and extracting components of interest for further analysis in a statistical software package.

To demonstrate this, we will consider a stream of event data where an event type and a timestamp are written out to a file. We make no assumptions that the file is sorted.

```
SECONDS EVENTCLASS
SECONDS EVENTCLASS
```

Given this input data, we would like to do several things. First, we would simply like to get a count of the number of events-per-hour for each class of events so that we can do some simple analysis in R. While Hadoop jobs can be written in any language that can read from `stdin` and write to `stdout` via the Streaming interface the best performance is to be had in its native language Java, which is what we will use for the examples. For this task the map fragment[1] is quite simple:

```
protected LongWritable one =
  new LongWritable(1);
public void map(Object key,Text value,
    Context context) {
  String[] fields = value.toString.split(
      "\t");
  Text      key     = new Text(fields[1]
    +"\t"
    +Long.parseLong(fields[0])/3600));
  context.write(key,one);
}
```

Our reducer is also quite simple: With Hadoop's default configuration the end result is a

[1] Full source code is available on the website

set of tab-delimited text files (one for each reduce node) containing the data. To read the data into, say, R, one might call:

```
out = read.delim(pipe("hadoop␣dfs␣-cat␣
    count_results/part-r-*"),header=F)
```

Alternatively, there is an R package `Rhipe` that interacts directly with Hadoop that should give superior performance to the `pipe` method as well as support for implementing Hadoop jobs in R directly. Upon reading the file, readers will note that event types are not organized within the output files. This is because the default partition function uses a hash of the entire output key, which also contains the hour. To ensure that all data for a given event appears in the reducer (and the same output file as a result) we must override the partition function:

```
public int getPartition(Text key,
    LongWritable value,int numReducers) {
  return key.toString.split("\t")[0].
      hashCode() % numReducers;
}
```

This ensures that all keys containing with the same event class are sent to the same reducer where they will all be sorted together. While this is convenient for reading, if the keys are not well distributed by the partitioner performance can suffer as some reducer nodes will be under utilized or require more data transfer.

Next, rather than grouping events by hour we are interested in calculating statistics about the waiting time between the events. To see how this works, first we consider our desired reduce step:

```
public void reduce(EventTimeWritble key,
    Iterator<LongWritable> times,
  Context context) {
    double sum_x = 0.0,sum_x2 = 0.0;
    double n      = -1.0;
    long   last   = 0;
    for(LongWritable l : times) {
      if(n < 0) {
        last = l.get();
      } else {
        long diff = l.get() - last;
        sum_x += diff;
        sum_x2 += diff*diff;
      }
      n += 1.0;
    }
    double mu = sum_x/n;
    double sd = Math.sqrt(sum_x2/n - mu*
        mu);
    context.write(new Text(key.getEvent()
        ),new Text(mu+"\t"+sd));
}
```

From our reducer we can see that we need to modify our mapper to emit the event type and timestamp as before, but we should also emit the time instead of a counter. Recall from the previous section that the **keys** are sorted before the reduce begins but that the values within each key are not necessarily sorted. By including the value in the overall key we effect a sorting of the values and when combined with a grouping function that only considers the event type we iterate over the timestamps for each event type in sorted order.

This pattern of employing a partition of employing a matched partition and group function along with including a value to be sorted in the key is a common pattern in Hadoop jobs where a well-defined iteration order leads to a calculation that can be performed with only a small amount of state.

Example: Expectation-Maximization in Hadoop

While Hadoop is often used to obtain sufficient statistics for further analysis, sometimes we want to work with the entire dataset. To demonstrate this we will consider the classic mixture-model EM problem. In this case we have observations $\vec{x}_1 = (x_1, y_1), \ldots, \vec{x}_n = (x_n, y_n)$ generated from k bivariate Normal distributions. Each observation $\vec{x}_i = (x_i, y_i)$ has a latent variable z_i identifying the mixture,

$$\begin{aligned} X_i | Z_i = j &\sim N_2\left(\mu_j, \Sigma_j\right) \\ P(Z_i = j) &= \tau_j \text{ and } \sum_{j=1}^{k} \tau_j = 1. \end{aligned}$$

Each iteration t takes the form of a Hadoop job that takes in the current parameters $(\tau_1, \ldots, \tau_k,$ $\mu_1, \ldots, \mu_k,$ $\Sigma_1, \ldots, \Sigma_k)^{(t)}$ as configuration and calculates parameters for $t+1$.

As in the previous section, we begin with the reduce phase, which must produce new values for all parameters at iteration $t+1$, to determine our needs for the map phase as well as any modifications that might be required to the grouping, sorting and combination. Recalling the usual M-step, we have the following equations:

$$
\begin{aligned}
\tau_j^{(t+1)} &= \frac{\sum_{i=1}^n \bar{z}_{ij}}{n} \\
\mu_j^{(t+1)} &= \frac{\sum_{i=1}^n \bar{z}_{ij}\vec{x}_i}{\sum_{i=1}^n \bar{z}_{ij}} \\
\Sigma_j^{(t+1)} &= \frac{\sum_{i=1}^n \bar{z}_{ij}\left(\vec{x}_i - \mu_j^{(t+1)}\right)\left(\vec{x}_i - \mu_j^{(t+1)}\right)'}{\sum_{i=1}^n \bar{z}_{ij}}
\end{aligned}
$$

Calculation of $\tau_j^{(t+1)}$ and $\mu_j^{(t+1)}$ is straightforward, only requiring the $\bar{z}_{ij}$, the likelihood that $\vec{x}_i$ was drawn from mixture j given the current estimates and the observation $\vec{x}_i$ for each mixture. This suggests that mapper should emit each observation k times with $1, \ldots, k$ as keys to allow each mixture to be calculated in parallel. We can also take advantage of the fact that the calculation of $\bar{z}_{ij}$ requires no data other than $\vec{x}_i$ to move that calculation to the mapper, increasing the parallel calculation of that value from k to the total number of map tasks.

Since we can calculate $tau_j^{(t+1)}$ and (μ_j, Σ_j) separately we can emit a key for each with values $\bar{z}_{ij}, \vec{x}_i, \vec{x}_i$ to calculate each of the parameters for the next iteration.

Example: Data Visualization

As a final, lighter, example we consider the problem of plotting the results of the mixture distribution in the last example. If the data set is too large to efficiently perform the EM in the last section, it is probably not feasible to draw, say, a scatterplot of the data.

To do this we will use the map-reduce framework for its side effects rather than the primary output. The basic idea is to partition the dataset into fixed size tiles. If our tile size is, say, 1000 points by 1000 points we can implement the mapper very easily:

```
public void map(Object key,Text value,
    Context context) {
  String[] pos = value.toString.split("\t
      ");
  double x = Double.parseDouble(pos[0]);
  double y = Double.parseDouble(pos[1]);

  Text key = new Text(Math.floor(x/
      1000.0)
    +"\t"+Math.floor(y/1000.0)
    +"\t"+Math.ceil(x/1000.0)
    +"\t"+Math.ceil(x/1000.0));
  context.write(key,value);
}
```

This maps each point into a specific bounding box that can easily be rendered by a reducer and written to an output file on the Hadoop cluster:

```
public void reduce(Text key,Iterator<Text
    > values,Context context) {
  Graphics g;
  // Set up Graphics surface
  for(Text value : values) {
    String[] pos = value.toString.split("
        \t");
    double x = Double.parseDouble(pos[0])
        ;
    double y = Double.parseDouble(pos[1])
        ;
    g.drawOval(x-pointSize/2,y-pointSize/
        2,
      x+pointSize/2,y+pointSize/2);
  }
  //Write graphics to HDFS as a separate
      file
  //Record the file name.
  context.write(key,new Text(outputFile))
      ;
}
```

Since the number of tiles is small relatively to the original data a single Java program can simply place the tiles onto a large canvas using the position of the bounding box to determine where to paint each tile. In this case, each tile is non-overlapping but a graphics format that supports transparency would allow non-overlapping regions to be easily composed into a single enormous canvas.

While the scatterplot is quite simple, it is not much more difficult to draw a graph (assuming straight line edges between vertices) used the same methods. Coupled with a map-reduce implementation of a graph layout algorithm (it seems like stress majorization techniques might be good candidates) it should be a powerful way to visualize very large graph structures.

Conclusions and Final Notes

With a large segment of the Hadoop literature primarily focused on the management of Hadoop clusters, we hope that we have given a few examples of the hows and whys of large data analysis. Furthermore, we hope that we given readers a few reasons to add Hadoop, or other map-reduce platforms, to their large-scale data analysis toolbox. To help develop this toolbox a little bit more quickly, code for all the examples in this article is available at `insert URL here`. Alternatively, a different set of algorithms (primarily related to non-hierarchical clustering at the moment) are available from the Mahout (`http://lucene.apache.org/mahout`) project.

It would be interesting to see the development of a general set of implementations for large scale analysis techniques, for example, it should be possible to implement backpropagation algorithms for neural networks using a technique similar to the one used for the EM example. In general, model fitting approaches that can be implemented in terms of gradient descent seems to be amenable to map-reduce approaches and there already exist methods for finding the first eigenvector of an adjacency matrix courtesy of map-reduce implementations of Google's PageRank algorithm.

Finally, while Hadoop can be easily integrated into an existing grid computing environment, it can also be easily deployed to cloud computing environments such as Amazon's EC2 for as-needed analysis where the expense of a full grid computing environment does not make sense. There is even a company called Cloudera that provides images for Hadoop clusters on EC2 that makes the job of getting started with Hadoop EC2 very straightforward. This may not be cost effective for groups doing a lot of analysis on an on-going basis, but for a specific data set or occasional usage it could be quite convenient.

Byron Ellis
Add Brite
`bellis@adbrite.com`

Statistical Graphics! — who needs Visual Analytics?

Martin Theus

Introduction

Talking to a broader audience at the useR!2009 conference on visualization does probably not call for a presentation showing the bleeding edge research work. Such a presentation should rather give some orientational overview on why and how statisticians may use graphics successfully, and how the way statisticians use graphics may contrast to other disciplines.

A first question should be, what we mean with "visualization" as this term as well as the field is used within many disciplines. There is a broad range of terms connected with visualization reaching from "Visual Communication" - which does "only" communicate qualitative mostly relational information - to "Visual Analytics", which is the latest buzz-word. From a statistician's point of view, visual analytics is exploratory statistical data analysis utilizing the most modern visualization and interaction techniques. As statisticians, we are concerned with visualizations which can generate insights based on data and their distributional properties - an aspect which is only slowly gaining a foothold in the field of InfoVis.

Statistical Graphics and Visual Analytics

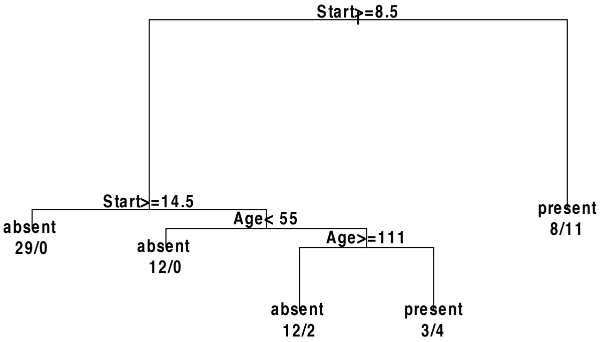


Figure 1: The standard representation of a tree in R

It is not surprising that the statistician's efforts in the area of statistical graphics gets less attention than the filed of Visual Analytics when we look at a simple example in Figure 1 and 2. Figure 1 shows the standard plot of a tree in R and Figure 2 the so called beam trees Admittedly, this is a very extreme example and it is easy to see that more flashy does not necessarily mean "better". Nonetheless, when it comes to "selling" your work, the R-tree will probably fall behind.

(9).

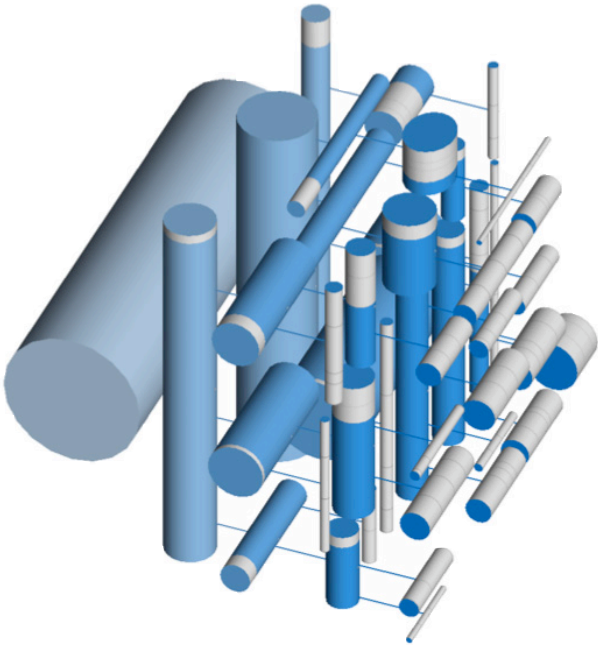


Figure 2: A so called "beam tree", as defined by (9)

For statistical graphics we usually focus on visualizing (mostly statistical) properties, rather than "only" the data itself. Figure 3 shows the Detergent data (7) in a parallel set plot (5), and Figure 4 a mosaic plot (4).

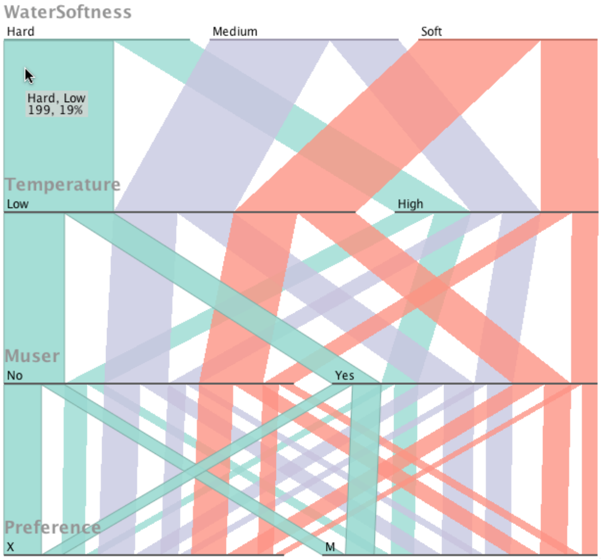


Figure 3: The detergent data set in a parallel sets display

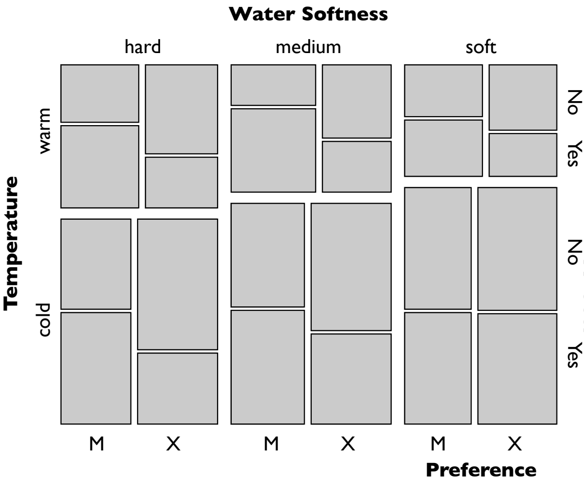


Figure 4: The detergent data set in a mosaic plot

The parallel set hierarchically visualizes the multivariate structure of the data with no particular model in mind. In order to find outliers or special structures this might work well. From a statistical point of view, we want to analyze associations, which might be conditioned upon variables or even combinations of variables. For a dataset with four categorical variables the mosaic plot is clearly the best choice. The plot immediately reveals that the more critical the washing conditions are, i.e., harder water and higher temperatures, the more people tend to stick to their well know detergent 'M' and will avoid experiments with the new brand 'X'.

As trained statisticians we are used to look at graphics and by doing so "virtually" performing statistical tests and estimations graphically without actually calculating the exact numbers. There are graphics or additions to graphics which can support this process directly. E.g., a boxplot is a summary statistics itself, and adding notches will even facilitate elementary inference within the graphics, cf. Figure 5.

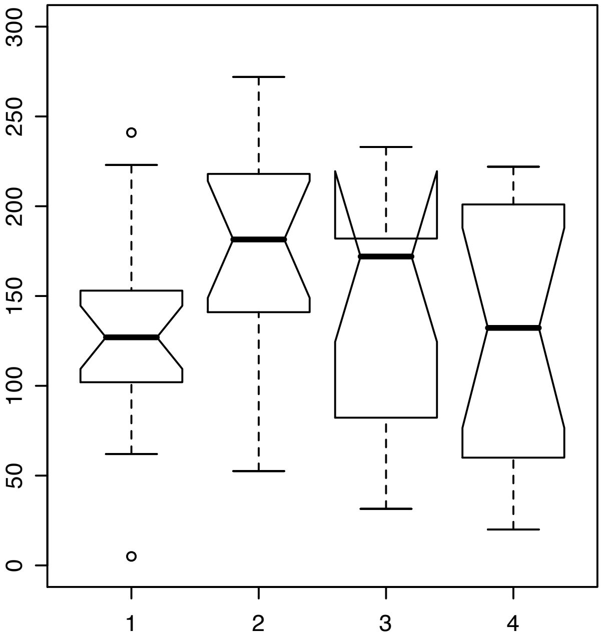


Figure 5: A notched boxplot combines graphical display and basic statistical test.

How do we use Graphics?

To better understand how and why we use graphics, we may classify types of graphics according to their intended use. There are

- **Exploration Graphics**
  Exploration graphics aim at gaining insights. They are mainly used by a single researcher. As we look at very many graphics in short succession during the exploration process, refined scales and legends are not of interest, but a highly interactive set-up which allows a

speedy modification of the graphics gets important.

- **Presentation Graphics**
  As presentations graphics intends to show interpreted results at the end of an exploration process, this kind of graphics makes extensive use of scales and legends. They are usually presented to a broader audience and ofter very much information must be transported in one single graphics — a common problem known by cartographers.

- **Interactive Information Graphics**
  Somewhat in between exploration and presentation graphics (which also covers conventional static info graphics known from newspapers) we find interactive info graphics on the web. There are some degrees of freedom which allow to change groups like states etc. and further information may be queried from the graphics, the basic visualization though, is predefined and can not be changed.

- **Diagnostics**
  For most statistical models and statistical procedure, diagnostic plots are offered to assess the quality of such a procedure. A drawback of such diagnostic plots is that these plots are often not very efficient as they are based on standard plot types and thus do not visualize the properties of the model directly. Furthermore, these plots often make it hard to directly relate back to the raw data or model/procedure properties which would allow for an efficient improvement of the model.

Statistics has its greatest impact on Exploration Graphics and Diagnostic Graphics, although graphics for presentation can benefit from statistical thinking as well.

Whereas the research for improved diagnostic plots may or may not lead to better graphical tools which can improve the building of efficient models, an interactive implementation of diagnostic plots may directly improve the ability to understand the influence of certain cases and/or variables of the model's properties.

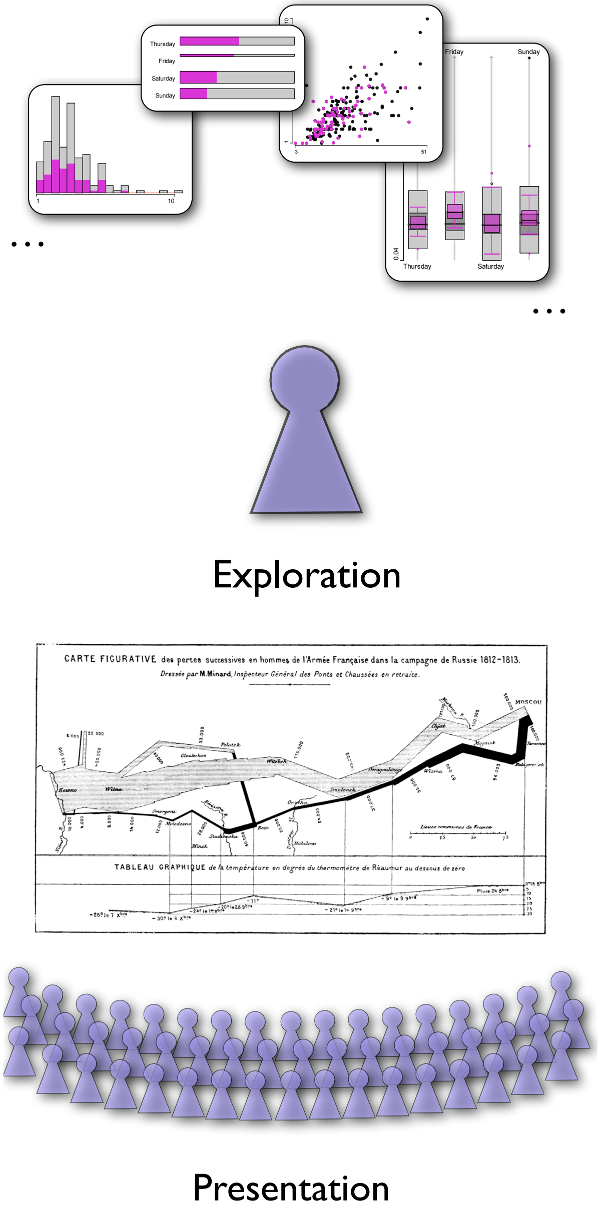

Thursday
Friday
Saturday
Sunday
...
...
Exploration
CARTE FIGURATIVE des pertes successives en hommes de l'Armée Française dans la campagne de Russie 1812–1813.
Dressée par M. Minard, Inspecteur Général des Ponts et Chaussées en retraite.
MOSCOU
TABLEAU GRAPHIQUE de la température en degrés du thermomètre de Réaumur au dessous de zéro
Presentation

Figure 6: Exploration graphics and presentation graphics can be best distinguished by the number of graphics used and by the size of their audiences

Figure 6 depicts the opposite relation between the number of graphics and and the size of the audience for exploration graphics and presentation graphics — a depiction which borrow from Antony Unwin.

Construction Principles for Statistical Graphics

Although there is actually not much to invent regarding standard statistical graphics, it is sometimes desirable to create a visualization which is specialized for a particular dataset. The general problem which must be solved by the design of any visualization is depicted in Figure 7.

Starting with a dataset, we code — nowadays usually with the help of a computer — this data into a graph. Even if this coding is completely lossless, it is still depending on the decoding process whether or not the visualization can successfully depict the phenomenon of interest. In many cases, the exact decoding of particular values is less desirable than the overall qualitative message which might be hidden in a dataset.

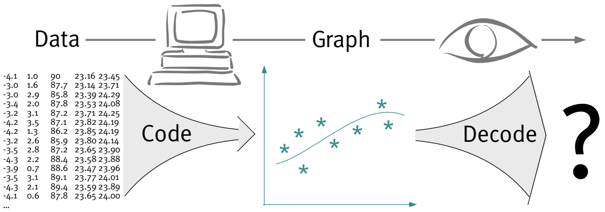


Figure 7: The decoding of the graphics content is the most crucial step in the pipeline.

Cleveland's (3) fundamental work on perceptual issues gives good advice which graphical elements and what use of scales may be most successful. We can not cover these principles here in depth, but even with these construction principles at hand a successful graph design might still be hard. One of the famous graphic designer, Milton Glaser, once put it this in a BBC documentary on the well known London Tube map (1); he said:

> *. . . All design basically is a strange combination of the intelligence and the intuition, where the intelligence only takes you so far and than your intuition has to reconcile some of the logic in some peculiar way. . . .*

[h]

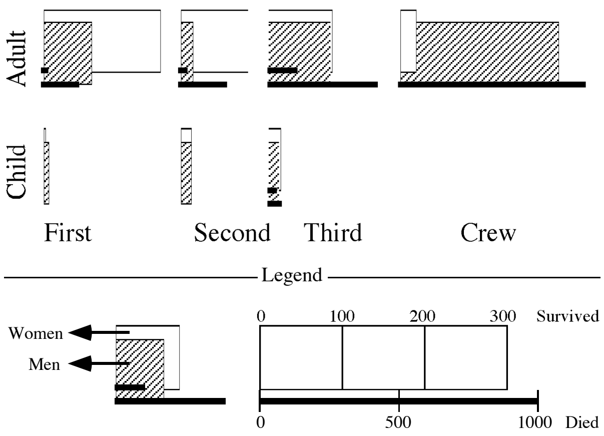


Figure 8: Bertin's proposal for visualizing multivariate categorical data

A good support for our intuition is to stick to certain graphical building blocks.

1. **Points**
   put along a (common) scale are usually the best way to depict continuous data.

2. **Areas** (mostly rectangles)
   should refer to counts of data which are grouped into certain categories.

3. **Lines** (like in profiles)
   shall connect data values which all belong to one entity.

Figure 8 is a good example of what happens when different building blocks are mixed into one graphics, although the underlying data was all measured on the same scale. The example is taken from (2). The original graphics was showing accident victims, the data shown in Figure 8 shows the Titanic survivors data — try to tell the story; only from the graph!

How does R support Statistical Graphics?

There is no doubt that R was never meant to be a graphics package, or a statistical software tool which has its particular strength in creating graphics. Nonetheless, R's structured language and package system allow for an easy extension of existing graphics as well as the creation of a completely new graphics system. E.g., to add a density estimator to an existing histogram, it only take one line of code — something which is usually far harder to achieve in most other statistical packages.

```
> hist(faithful$eruptions, freq=F)
> lines(density(faithful$eruptions))
```

There are three categories where R gets involved in the creation of graphics. There are the general purpose packages `graphics`, `lattice` based on `grid`, `ggplot2` (10) and `iplots` (6) (with its successor `iplots eXtreme` (8). Other packages deal with specific topics like the `party` package which offers advanced plotting methods for trees, or the `vcd` package which has many graphic types for categorical data. The third category where R is supporting graphics can be found with graphics packages which connect to R. `ggobi` has a connection to R, `Klimt` offers advanced interactive graphics for tree models which are generated in R, and `Mondrian` enhances graphics with statistics which are calculated within R.

Whereas the flexibility to enhance or even create new plots has great potential, the growing number of graphics packages seems to lead to a certain lack of clarity under what circumstance it is advisable to use which plot and/or graphics package.

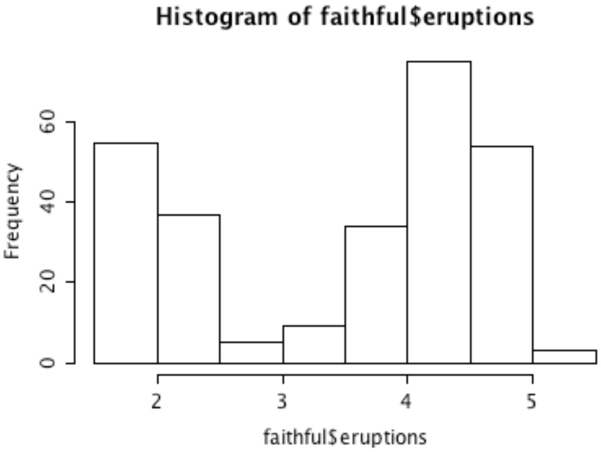


Figure 9: The standard histogram as defined within the base package

The great variety of graphics packages in R has a drawback though. The simple question: *"How to draw a histogram?"* can (to my knowledge) answered in five different ways in R right now. Figure 9 shows the standard histogram within the base package for the Old Faithful data, created by the command `hist(faithful$eruptions)`.

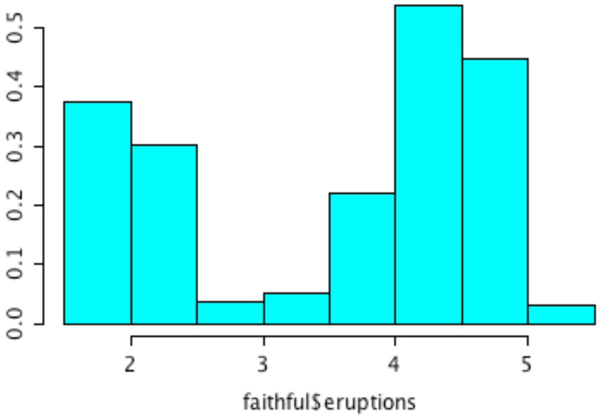


Figure 10: The "truehist" function within the MASS package

Figure 10 shows the same data in a histogram generated by the function "truehist" (`truehist(faithful$eruptions)`) which can be found in the MASS package. Interestingly the bar heights differ, although the breaks used in the two histograms in Figure 9 and 10 seem to be exactly the same.

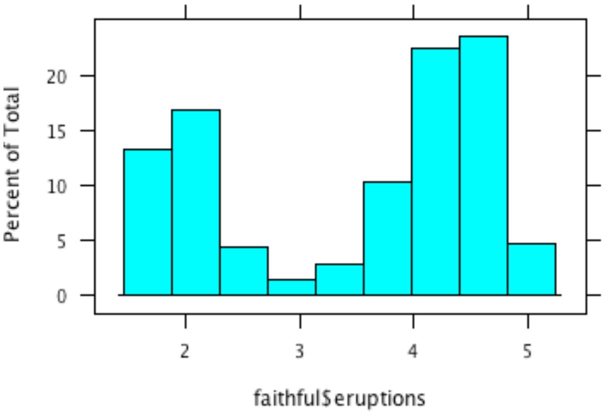


Figure 11: The histogram function from the lattice package

The function `histogram` within the lattice package generates the histogram in Figure 11 by typing `histogram(faithful$eruptions)`. Apart from the less appealing cyan — which was also use by the truehist function — the extensive scalings around all four sides of the histogram effectively reduces the area used to plot the actual data.

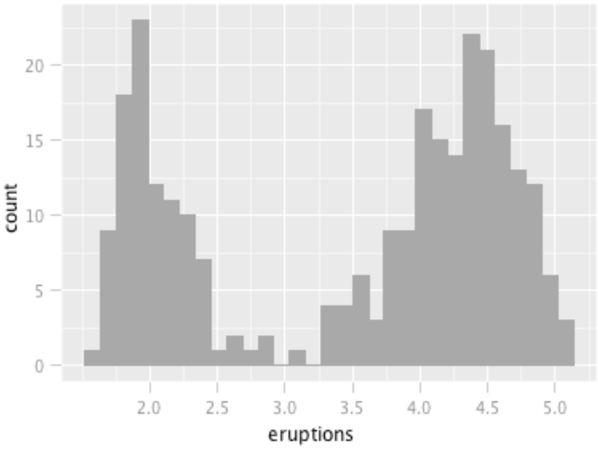


Figure 12: The default histogram within the ggplot2 package

The histogram in Figure 12 was generated with the ggplot2 package. Contrary to the functions in the previous figures ggplot uses a generic plot function, as can be seen from the command:

```
> qplot(eruptions, data = faithful,
 + geom=’’histogram’’)
```

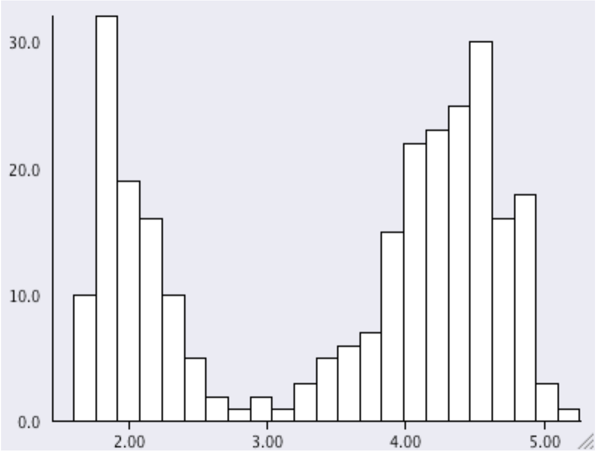


Figure 13: The interactive histogram from the iplots package

Finally in Figure 13 we find the interactive version of a histogram created with `ihist(faithful$eruptions)`.

This “parade” of histograms is not meant to criticize one of the particular implementations, or R’s flexibility in adding new functionality. It is more meant to raise the question whether or not we need some kind of consolidation of functionality within R. Depending on when you got to know R, and who introduced you to it, your answer to “*How to draw a histogram?*” may be very different. This inconsistency can be an obstacle not only to those who start to getting to know R.

Obviously this problem is not restricted to graphics only but can also be found in the context of standard model fitting and other statistical procedures.

Conclusion

At least in the closing of this paper I need to get back to the question in the title.

Statistical graphics — especially interactive graphics for exploratory data analysis — are well developed and the wealth of graphic types guarantees that almost any dataset can be explored with these graphics. The particular power of statistical graphics lies in their design which is usually targeted towards visualizing statistical properties, rather than “just” showing the data itself. I.e., the visualization needs to focus on the potential gain of insight and inference. In this respect, statistical graphics can not learn very much from what is going on within the recent development in visual analytics. What definitely needs to be learned from this community are the technical skills which enable the creation of interactive visualizations which attract the potential audience.

iplots eXtreme can be the perfect toolbox to achieve this within a well known and easy to handle statistical environment, namely R.

Bibliography

[1] BBC. London underground tube map video documentary. `http://infosthetics.com/archives/2008/11/the_london_underground_map_tv_documentary.html`

[2] J. Bertin. *Semiology of Graphics*. University of Wisconsin Press, Madison, 2nd edition, 1983.

[3] W. S. Cleveland. *The Elements of Graphing Data*. Wadsworth, Monetrey, CA, 1985.

[4] H. Hofmann. Mosaic plots and their variants. In C.-h. Chen, W. Härdle, and A. Unwin, editors, *Handbook of Data*

Visualization (Springer Handbooks of Computational Statistics), chapter III.13, pages 617–642. Springer-Verlag TELOS, Santa Clara, CA, USA, 2008.

[5] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *IEEE Trans. Vis. Comput. Graph.*, 12(4):558–568, 2006.

[6] M. Theus and S. Urbanek. iplots: Interactive Graphics for R. *Statistical Computing and Graphics Newsletter*, 15(1), 2004.

[7] M. Theus and S. Urbanek. *Interactive Graphics for Data Analysis: Principles and Examples (Computer Science and Data Analysis)*. Chapman & Hall/CRC, 2008.

[8] S. Urbanek. iplots eXtreme — Next-generation interactive graphics for analysis of large data. `http://www2.agrocampus-ouest.fr/math/useR-2009/slides/Urbanek.pdf`

[9] F. van Ham and J. J. van Wijk. Beamtrees: Compact visualization of large hierarchies. In *INFOVIS '02: Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02)*, page 93, Washington, DC, USA, 2002. IEEE Computer Society.

[10] H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. useR! Springer, New York, 2009.

Martin Theus
Telephónica O_2 Germany
University of Augsburg, Department of Computeroriented Statistics and Data Analysis
martin@theusRus.de

Reader Response

Comments on "Taking it to higher dimensions"

(3) provides a valuable service by reminding readers how to read pseudo-three-dimensional bar charts since so many people misread these charts and are misled by them. He points out that the top of the bars in Figure 1 are below the horizontal lines that represent their values. Therefore, the reader needs to project the bars onto the back wall of the chart that contains the grid lines as shown in his Figure 2. This note provides additional observations on these charts. Although the gap shown in Figure 1 is common, not all pseudo-three-dimensional bar charts include this gap (5). Some software such as the default in PowerPoint draw the figure so that the gap is zero and the back of the bar touches the horizontal line as in Figure 2. Although Excel and PowerPoint come in the same suite of programs from the same company, they use different defaults for the gap. With other software it is the top of the front panel of the bar that touches the line. I find it unacceptable that the way to read a graph depends on the software used to produce it; this makes a strong argument in addition to the gap argument of Krause for not adding an unnecessary pseudo-third dimension. In Excel, the gap is an option which can be changed. (4) points this out in his blog. Click on the series, go to format data series, then options and you will see gap depth. Figure 2 modifies Figure 1 so that the gap depth is zero. These steps apply to both Excel 2003 and Excel 2007.

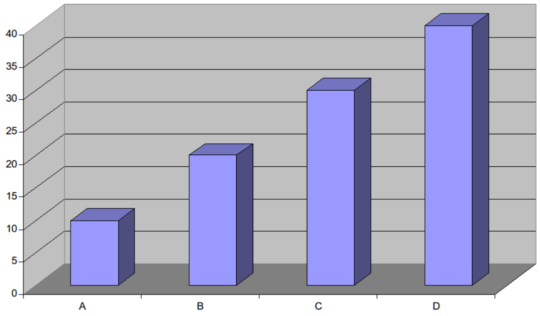


Figure 1: Three-dimensional bar chart modeled after Figure 1 of [2]

Krause further points out that the problem of the gap is magnified when perspective is added to the plot and that this was noted by (1). Haemer discusses perspective in the 1947 paper but his examples use two-dimensional bar charts. He discusses pseudo-three-dimensional figures in a 1951 paper (2).

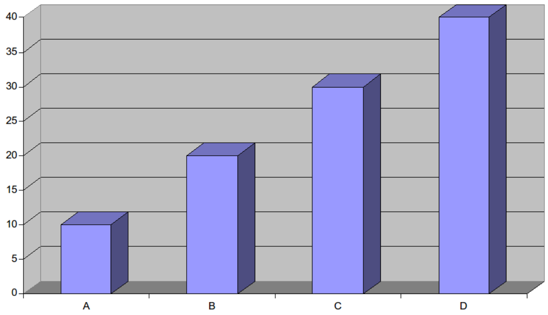


Figure 2: Three-dimensional bar graph with gap of zero.

In summary, not all spreadsheet three-dimensional graphs include the gap and in Excel the gap is an option that can be changed.

Bibliography

[1] Haemer, Kenneth W. (1951) The perils of perspective. *The American Statistician* 1(3):19

[2] Haemer, Kenneth W. (1951) The pseudo third dimension. *The American Statistician* 5(4):28

[3] Krause, Andreas (2009) Taking it to higher dimensions *Statistical Computing and Graphics Newsletter* 20(1), June 2009

[4] Peltier, Jon (2009) PTS blog, `http://peltiertech.com/WordPress/does-excel-suck/#comments`

[5] Robbins, Naomi B. (2005) *Creating More Effective Graphs*. Hoboken: Wiley.

Naomi B. Robbins
NBR
http://www.nbr-graphs.com
naomi@nbr-graphs.com

Announcements

The Statistical Computing and Graphics Award

The ASA Sections of Statistical Computing and Statistical Graphics have established the Statistical Computing and Graphics Award to recognize an individual or team for innovation in computing, software, or graphics that has had a great impact on statistical practice or research. Typically, awards are granted bi-annually.

The prize carries with it a cash award of $5,000plusanallowanceofupto$1,000 for travel to the annual Joint Statistical Meetings (JSM) where the award will be presented.

Qualifications The prize-winning contribution will have had significant and lasting impact on statistical computing, software or graphics.

The Awards Committee depends on the American Statistical Association membership to submit nominations. Committee members will review the nominations and make the final determination of who, if any, should receive the award. The award may not be given to a sitting member of the Awards Committee or a sitting member of the Executive Committee of the Section of Statistical Computing or the Section of Statistical Graphics.

Nomination and Award Dates Nominations are due by December 15 of the award year. The award is presented at the Joint Statistical Meetings in August of the same year. The first award will be given in 2010, and subsequent awards are to be made at most bi-annually according to the discretion of the Awards Committee.

Nominations should be submitted as a complete packet, consisting of:

* nomination letter, no longer than four pages, addressing points in the selection criteria * nominee's curriculum vita(s) * maximum of four supporting letters, each no longer than two pages

Selection Process The Awards Committee will consist of the Chairs and Past Chairs of the Sections on Statistical Computing and Statistical Graphics.

The selection process will be handled by the Awards Chair of the Statistical Computing Section. Nominations and questions are to be sent to the e-mail address below.

Fei Chen
Avaya Labs
233 Mt Airy Rd
Basking Ridge, NJ 07920
`feic@avaya.com`

Section Officers

Statistical Computing Section Officers 2009

Jose C. Pinheiro, Chair
jose.pinheiro@novartis.com
(862) 778-8879

Deborah A. Nolan, Past-Chair
nolan@stat.berkeley.edu
()643-7097

Luke Tierney, Chair-Elect
luke@stat.iowa.edu
(319) 335-3386

Elizabeth Slate, Secretary/ Treasurer
slate@musc.edu
(843) 876-1133

Montserrat Fuentes, COMP/COS Representative
fuentes@stat.ncsu.edu
(919) 515-1921

Robert E. McCulloch, Program Chair
robert.mcculloch1@gmail.com
(512) 471-9672

Thomas Lumley, Program Chair-Elect
tlumley@u.washington.edu
(206) 543-1044

Barbara A Bailey, Publications Officer
babailey@sciences.sdsu.edu
(619) 594-4170

Jane Lea Harvill, Computing
Section Representative
Jane_Harvill@baylor.edu
(254) 710-1517

Donna F. Stroup (see right)
Monica D. Clark (see right)

Nicholas Lewin-Koh, Newsletter Editor
lewin-koh.nicholas@gene.com

Statistical Graphics Section Officers 2009

Antony Unwin, Chair
unwin@math.uni-augsburg.de
+49-821-598-2218

Daniel J. Rope, Past-Chair
drope@spss.com
(703) 740-2462

Simon Urbanek, Chair-Elect
urbanek@research.att.com
(973) 360-7056

Rick Wicklin, Secretary/ Treasurer
Rick.Wicklin@sas.com
(919) 531-6629

Peter Craigmile, COS Rep 08-10
pfc@stat.osu.edu|
(614) 688-3634

Linda W. Pickle, COMP/COS Representative 2007-09
lpickle@statnetconsulting.com
(301) 402-9344

Steven MacEachern, Program Chair
snm@stat.ohio-state.edu
(614) 292-5843

Heike Hofmann, Program Chair-Elect
hofmann@iastate.edu
(515) 294-8948

Donna F. Stroup, Council of Sections
donnafstroup@dataforsolutions.com
(404) 218-0841

Monica D. Clark, ASA Staff Liaison
monica@amstat.org
(703) 684-1221

Andreas Krause, Newsletter Editor
andreas.krause@actelion.com

Statistical COMPUTING & GRAPHICS

The Statistical Computing & Statistical Graphics Newsletter is a publication of the Statistical Computing and Statistical Graphics Sections of the ASA. All communications regarding the publication should be addressed to:

Nicholas Lewin-Koh
Editor Statistical Computing Section
`lewin-koh.nicholas@gene.com`

Andreas Krause
Editor Statistical Graphics Section
`andreas.krause@actelion.com`

All communications regarding ASA membership and the Statistical Computing and Statistical Graphics Section, including change of address, should be sent to

American Statistical Association
1429 Duke Street
Alexandria, VA 22314-3402 USA
TEL (703) 684-1221
FAX (703) 684-2036
`asainfo@amstat.org`