Volume 22, July 2011

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

Statistical COMPUTING & GRAPHICS

A Word from our 2011 Section Chairs

RICHARD HEIBERGER
COMPUTING

Statistical Computing, as we in this section all know, is the center of Statistics. The program for the upcoming JSM 2011 in Miami Beach indicates that centrality. The Section on Statistical Computing is the main sponsor of 27 technical sessions and activities, and cosponsor of 50 more for a total of 77. The session and paper titles cover the entire gamut of statistical activity and many scientific and social issues. The cosponsors include most of the sections of ASA and many of the asso-

Continued on page 2 . . .

JÜRGEN SYMANZIK
GRAPHICS

Is there a fifth force in particle physics, in addition to the previously known four interactive forces electromagnetism, strong nuclear force, weak nuclear force, and gravitation? A possible new force, called "technicolor" force, was recently described in an article by three physicists (`http://prl.aps.org/abstract/PRL/v106/i25/e251803`), based on unexpected experimental observations (`http://prl.aps.org/abstract/PRL/v106/i17/e171801`). The discovery of such a new force would

Continued on page 2 . . .

Contents of this volume:

Computing Chair
Continued from page 1.

ciated societies. Topics include: career development, climate, clinical trials, data mining, environment, financial analysis, functional data analysis, genetics, graphics, health policy, high-dimensional methods, mapping, marketing, Monte Carlo simulation, national security, physical and engineering sciences, public affairs, quality and productivity, risk, social media, teaching. There is something for everyone in the collection.

The Program Chair for the Section this year is David J. Poole. With the help of all of you who are presenting he has done a magnificent job. Just the Invited Paper Session Topics are fascinating: Advancement in Hierarchical Models, The Future of Statistical Computing Environments, Computational Methods for Space-Time Correlated Data, Statistics in Computational Advertising (what goes on behind the scenes when a search engine decides which advertisements to display to a specific search item), Statistical Analysis of Actigraphy Data (Actigraphy is a relatively non-invasive method of monitoring human rest/activity cycles.).

In addition to invited papers at the JSM, the Section cosponsors the Data Expo (this year with posters related to the 2010 Deepwater Horizon Oil Spill), the Interface on Computer Science and Statistics, student awards, and Continuing Education courses at the JSM. This year we have two courses: "Data Stream Mining: Tools and Applications" by Simon Urbanek and "Bootstrap Methods and Permutation Tests for Doing and Teaching Statistics" by Tim Hesterberg.

The centrality of Statistical Computing, and computing in general, in our world implies responsibility. Part of our mission is to get everyone who does data analysis to do it well. We therefore must work toward getting high quality statistical computing and graphics systems accessible to everyone on any computer (cell phones included) and at every level from introductory courses to highly complex large and very large applications. Many of the sessions in the upcoming JSM are reflective of the work that our members have done in making high-quality software and thinking processes accessible. We have sessions on software design for teaching, and statistical teaching using high-quality software.

The main specific social event is the annual "Section on Statistical Computing and Section on Statistical Graphics Joint Mixer" on Monday evening. I hope to see many of you there, as well as at the rest of the JSM.

Richard M. Heiberger
Temple University

Graphics Chair
Continued from page 1.

be considered a major revolution in the world of particle physics. But, why does an editorial written by the current chair of the Statistical Graphics section start with discoveries from particle physics? The answer of course is simple: Because of the statistical graphics that supported this discovery.

A less technical summary of these findings and a freely accessible version of the most relevant figure can be found here and here at the NewScientist. Of course, according to our standards, this is a relatively simple statistical figure. Nevertheless, this may be the most important piece of a puzzle that transforms many years of scientific research into a new physical theory and model.

It appears as if statistical graphics have helped to detect the unknown and unexpected — again! Most of us know the classical examples from the last 150 years where statistical graphics have helped to discover the previously unknown. This includes John Snow's discovery that the 1854 cholera epidemic in London most likely was caused by a single water pump on Broad Street, a fact he observed after he had displayed the deaths arising from cholera on a map of London. A second, well–known example is Florence Nightingale's polar area charts from 1857, the so–called Nightingale's Rose (sometimes incorrectly called coxcombs), that demonstrated that the number of deaths from preventable diseases by far exceeded the number of deaths from wounds during the Crimean War. These figures convinced Queen Victoria to improve sanitary conditions in military hospitals. Many additional important scientific discoveries based on the proper visualization of statistical data could be mentioned, but the most important fact is: New discoveries based on the visualization of data can

happen here and now!

This is a message we should carry to our collaborators, students, supervisors, etc.: Statistical graphics (or visual data mining, visual analytics, or any other name you like) typically do not provide a final answer. But, statistical graphics often help to detect the unexpected, formulate new hypotheses, or develop new models. Later on, additional experiments or ongoing data collection as well as more formal methods (and p–values if you really want) may be used to verify some of the original graphical findings.

As can be seen in the examples above, statistical graphics are of universal use. Nevertheless, graphics may also be unique for some particular application areas. This naturally leads me to invite you to the Joint Statistical Meetings (JSM) where new graphical methods and their applications are presented. I do not want to repeat the entire JSM graphics program here, but rather say that Webster West, our 2011 Program Chair, has put together an excellent set of sessions for the 2011 JSM held from July 30 to August 4, 2011, in Miami Beach, Florida, at the Miami Beach Convention Center. So, let me just mention a few highlights here: First, there is the 2011 Data Expo (Mon, 8/1/2011, 2:00pm to 3:50pm) with posters related to the 2010 Deepwater Horizon Oil Spill. Next, there is the Section on Statistical Computing and Section on Statistical Graphics Joint Mixer (Mon, 8/1/2011, 7:30pm to 10:00pm) that only is credited to Statistical Computing in the online program and may be missed at first glance. For everything else (from invited sessions over topic contributed sessions to contributed sessions and roundtables), start at `http://www.amstat.org/meetings/jsm/2011/onlineprogram/index.cfm` and search for details. Enjoy the JSM and/or the beaches!

This may sound strange to many of you, but plans for the JSM 2012 to be held from July 28 to August 2, 2012, in San Diego, California, at the San Diego Convention Center, have far progressed. If you have any suggestions for an invited session for 2012, please contact our 2011 Program Chair–Elect (and 2012 Program Chair), Hadley Wickham (`hadley@rice.edu`), as soon as possible.

Finally, if you always wanted to know what your section membership fees are used for, here is a brief summary provided by John ("Jay") Emerson, our section secretary/treasurer: (1) We have contributed $375 to the 2011 Cavell Brownie Scholars mentoring program (`http://www.amstat-online.org/2010mentoringprogram/CavellBrownieScholarsProgram.php`). (2) We have contributed $2,000 to support graduate students at the Interface 2011 conference. (3) We contributed $1,750 to support color printing of the special 2009 Data Expo issue of JCGS. Just a reminder that the Statistical Graphics section offers color publishing grants that have only received very few applications so far. For details on the application procedure, please refer to `http://stat-graphics.org/graphics/grants.html`. (4) We supported Seth Roberts' invited presentation at the JSM 2010 with $450. (5) Other recurring annual expenses are for food and drinks at the mixer and the officers' meeting at the JSM and for various awards, such as "The Statistical Computing and Graphics Award" and the "Student Paper Competition".

Did you stay with me until the very end? If so, let me reveal one more fact: Another group of researchers from particle physics that conducted the same experiments did not detect any anomalies in their data and figures (`http://www.bbc.co.uk/news/science-environment-13722986` and `http://www-d0.fnal.gov/Run2Physics/WWW/results/final/HIGGS/H11B/`). Currently, researchers from both groups are comparing their results to come to a final joint conclusion.

Jürgen Symanzik
Utah State University

Statistical Graphics and InfoVis — Twins separated at Birth?

Nicholas Lewin-Koh and Martin Theus, Eds.

This volume features two articles both looking at the aspects of "graphical displays of quantitative data". In the first paper "Visualization: It's More than Pictures!" by Robert Kosara, Robert sheds a light from the point of view of an InfoVis person, i.e. someone who primarily learned how to design tools and techniques for data visualization. With the second article "Visualization, Graphics, and Statistics" by Andrew Gelman and Antony Unwin, we get a similar view, but now from someone whose primary training is in math and/or statistics.

Given this set-up, we might think that we have a good idea how both sides would argue, and what would be the assets the one and the other side would claim: computer scientists are good at designing tools for data visualization and statisticians are good at doing the analysis; and consequently, they both don't know much about the expertise of the other discipline.

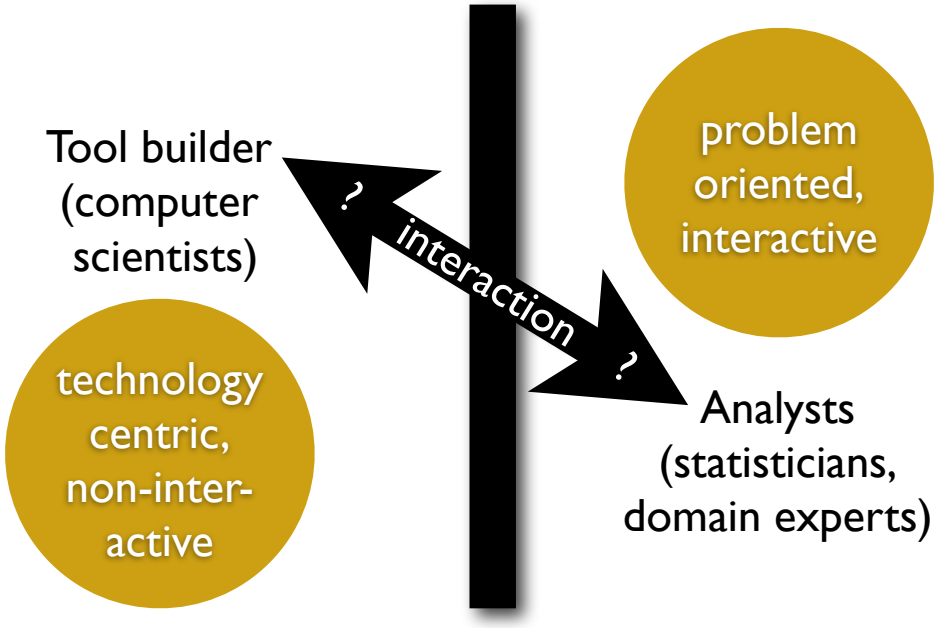


Figure 1: Is there a "wall" between the two promotors of graphical displays? (Taken from `http://www.theusrus.de/blog/the-wall-what-wall/`.)

Reading the two papers you will find out that, while there is certainly some truth behind this simple classification, the overlap and agreement is larger than one would probably think. The common and most important understanding is that there is a story to be told with the data. Graphics are the most powerful tool to do this, no matter what your training and background is.

Nonetheless, there is still a lot to be learned from each other and the one or the other difference or misunderstanding might spur the discussion between the two sides. As a platform for this **discussion** you can use the post at `http://www.theusrus.de/blog/InfoVis-and-StatGraphics/` — we are looking forward to a lively exchange, which might even end up in a collaboration!

Visualization: It's More than Pictures!

Robert Kosara

Introduction

Information visualization is a field that has had trouble defining its boundaries, and that consequently is often misunderstood. It doesn't help that InfoVis, as it is also known, produces pretty pictures that people like to look at and link to or send around. But InfoVis is more than pretty pictures, and it is more than statistical graphics.

The key to understanding InfoVis is to ignore the images for a moment and focus on the part that is often lost: interaction. When we use visualization tools, we don't just create one image or one kind of visualization. In fact, most people would argue that there is not just one perfect visualization configuration that will answer a question [4]. The process of examining data requires trying out different visualization techniques, settings, filters, etc., and using interaction to probe the data: filtering, brushing, etc.

The term *visual analytics* captures this process quite well, and it also gives a better idea of what most visualization is used for: analysis. Analysis is not a static thing, and can rarely be done by looking at a static image. Visualization and visual analytics use images, but the images are only one part of visualization.

Cheap Thrills

It is no wonder that many people think that visualization is primarily about pretty and colorful pictures, even smart people like Andrew Gelman. What readers see on popular websites like FlowingData [8] and infosthetics [3], and what makes them so popular, are the pictures. In many cases, they provide only minimal context, and readers are mostly left to look at the images as images, rather than figure out what they are actually trying to tell them.

Another issue is the blurred boundary between actual visualization and data art, which is often ignored on purpose to have more interesting images to choose from. The result is that the expectation many people have of visualization images is similar to that of a piece of art: that you can look at it and like or don't like it, but don't get any actual information out of it. In fact, I have argued that what Gelman calls "that puzzling feeling" is actually what sets pragmatic visualization apart from data art [2].

Data art clearly has its place, and the more pragmatic visualization community can learn from it. But when we're talking about visualization in the context of statistics and the analysis of data, we need to draw a clear distinction. Visualization is not art any more than statistics is.

Goals

So what does visualization do, then? The main idea is to provide insight into data. This is how scientific visualization got started in the 1980s: the huge amounts of data produced by the then-recent supercomputers required new ways of analysis. Scientific visualization made it possible to see the effects of design changes on the pressure distribution of an airplane wing, for example. The same thing could be done with number crunching in theory, but it was a lot more immediate and obvious where things went wrong when the model was actually shown as an image.

Another, more recent, goal is making data accessible. A lot of data is already available in principle, but not in a form that normal people would want to play with. There is still a difference between data being technically available and actually being accessible to a broad audience. Creating a visualization makes it possible for people to start poking around in the data and perhaps discover interesting facts that nobody has seen before.

Finally, to borrow Tableau's tagline [6], the goal of visualization is *to make analytics fast*. Sure, a lot of questions can be asked of a data warehouse by writing 150-line SQL queries, but changing parameters or exploring variations is going to be difficult this way. An interactive visualization system makes it possible to do that and ask many more questions in much less time. This is not only a worthwhile goal in a business context, but also in the sciences and many other fields: the easier and quicker it is to ask questions, the more questions can be asked.

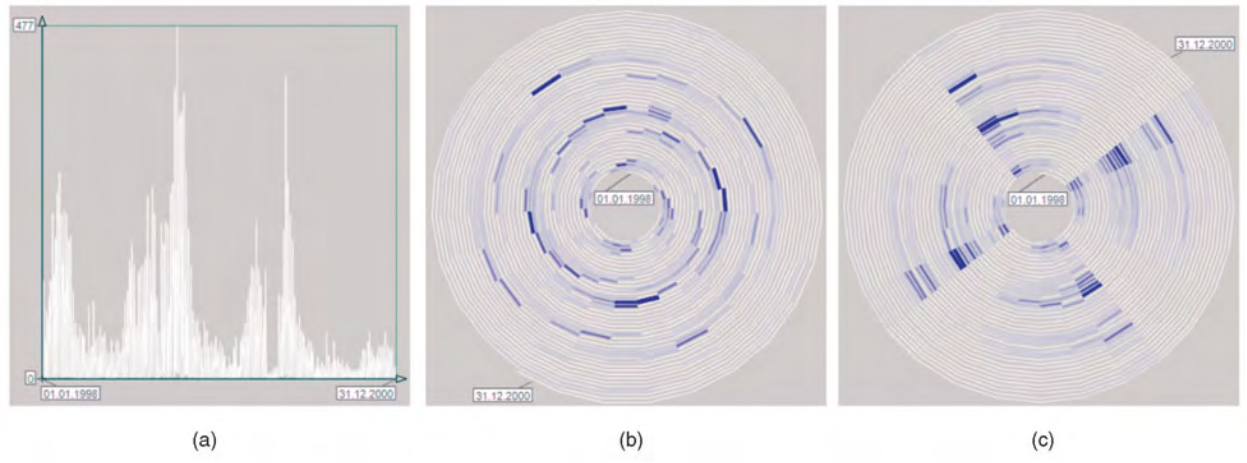


Figure 1: Spirals are useful for finding periodicity in data (from [1]). (a) The bar chart shows no obvious periodic pattern; (b) the spiral set to 25 days hints at a periodic pattern, but this is clearly not the correct time frame; (c) at 28 days, the pattern is very clearly visible.

Example: Perceive Patterns

A common question in time series data is whether the data is periodic, and if yes, what the period is. A common way of finding out is drawing the data on a spiral [1]. By changing the number of data points that is shown per full round the spiral makes (that number is constant, of course), patterns become visible. Figure 1 shows an example of sick leave data that has an interesting periodic pattern: in 28 days, there are four periods, which means that there is a weekly pattern: more people call in sick on Mondays than later in the week.

The way this pattern was discovered is deceptively simple. All it took was to play with a slider that allowed the user to change the number of days on shown on the spiral. Slide it back and forth, and soon you will see a pattern (if there is one). With a bit of practice, you can even tell when you're getting close, as there are telltale signs around the optimal value.

The key here is not just the way of displaying the data, but also the interaction. Without it, it would take much longer to find the correct interval, or require some very educated guessing. The power of visualization is that it allows the user to find things he or she may not have expected, and thus would not have been looking for.

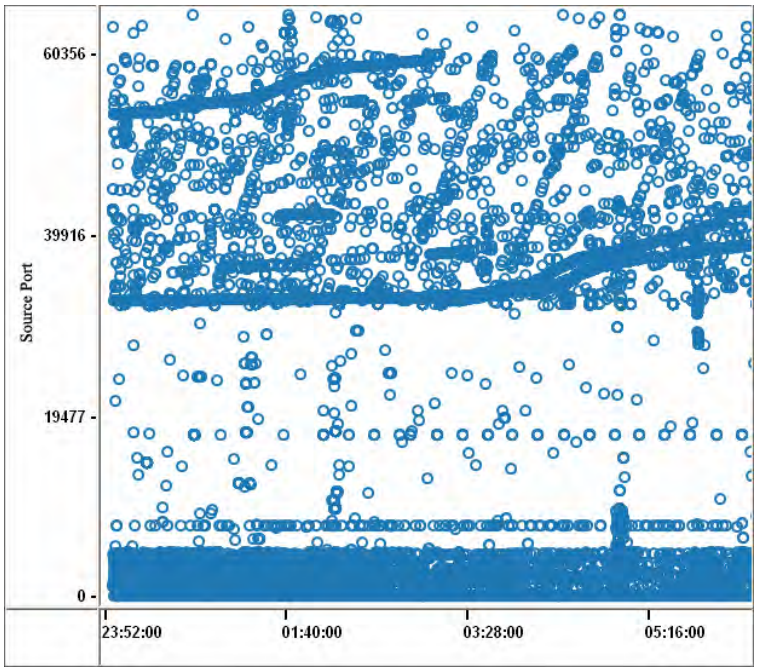
Example: Filter the Flood

A beautiful example of the integration of analysis and visualization is a system for visualizing network traffic data [7]. To be able to deal with the enormous amount of data, the system includes a declarative logic system that can apply rules to find certain patterns in the data. The idea is to identify patterns of known good data, and filter that data out, so that what remains is the data that needs to be examined more closely (Figure 2).

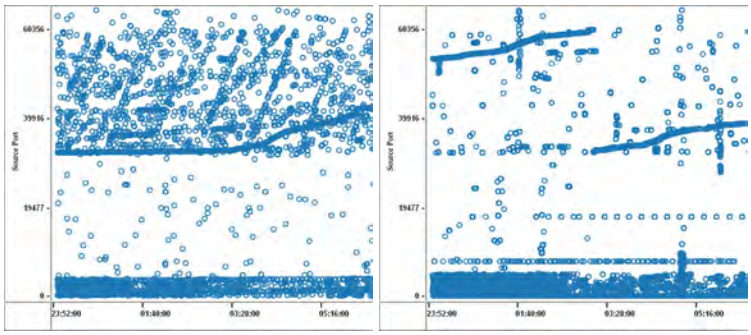
Instead of having to write the declarations by hand, however, the system allows the user to select data points and creates a rule from the selection. The user can then apply that rule to other traffic to see if it matches the right data, and even examine and edit the actual definition directly. Creating and refining definitions of different traffic patterns is relatively straight-forward this way, especially for a network security expert.

One of the most clever design decisions in this system is to focus on the known good traffic, rather than trying to define what is suspicious. New types of scans and attacks are developed all the time, so keeping up with them is practically impossible. Also, defining the bad traffic would defeat a big advantage of the visual part of this system: being able to see new patterns as they emerge.

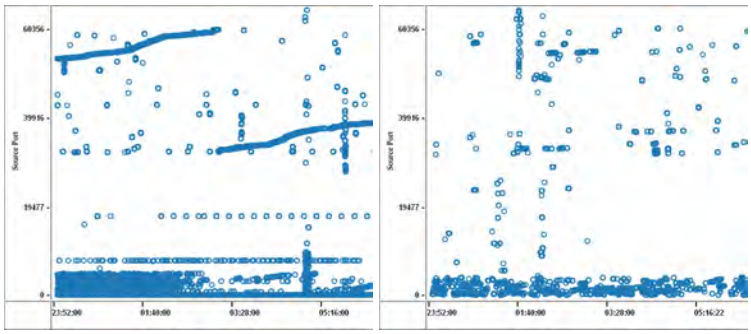
By treating the known good traffic as irrelevant, it can be removed, and the user can focus on the



(a)



(b) (c)



(d) (e)

Figure 2: Event diagrams showing flows during residual analysis (from [7]). (a) Original unidentified traffic (b) Flows with "mail" label (c) The residual after filtering out "mail" from Figure 6a. (d) Flows with the "scan" label (e) The residual after filtering out the "scan" label from Figure 6(c).

parts that may be suspicious. Each part is done by the component that is best suited for it. The machine uses the rules to sift through and filter large amounts of data, and the user tries to understand what remains and tweaks the rules (or finds a way to fend off a break-in attempt).

Conclusions

Visualization cannot exist without visual representations, and those representations need to be designed so that they can be effectively and efficiently perceived. There is no question that more effective visual representations will result in better analysis and easier comprehension of data. But the images aren't everything.

There is also a vast open field of research that makes good use of statistics to enhance visualization. A few attempts at this exist [5], but a lot more can be done. Despite the relatively new field of visual analytics, visualization research is still very strongly focused on visual representation, with too little attention being paid to interaction, analysis, and cognitive effects.

And yet, visualization is much, much more than what it appears to be at first glance. The real power of visualization goes beyond visual representation and basic perception. Real visualization means interaction, analysis, and a human in the loop who gains insight. Real visualization is a dynamic process, not a static image. Real visualization does not puzzle, it informs.

Bibliography

[1] W. Aigner, S. Miksch, W. Müller, H. Schumann, and C. Tominski. Visual methods for analyzing time-oriented data. *Transactions on Visualization and Computer Graphics*, 14(1):47–60, 2008.

[2] R. Kosara. Visualization criticism — the missing link between information visualization and art. In *Proceedings of the 11th International Conference on Information Visualisation (IV)*, pages 631–636. IEEE CS Press, 2007.

[3] A. V. Moere. information aesthetics. `http://infosthetics.com/`.

[4] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualization. In *Proceedings Visual Languages*, pages 336–343. IEEE Computer Society Press, 1996.

[5] C. A. Steed, P. J. Fitzpatrick, T. Jankun-Kelly, and J. E. S. II. Practical north atlantic hurricane trend analysis using parallel coordinates and statistical techniques. In *Proceedings of the 2008 Workshop on Geospatial Visual Analytics*, 2008.

[6] Tableau software. `http://tableausoftware.com/`.

[7] L. Xiao, J. Gerth, and P. Hanrahan. Enhancing visual analysis of network traffic using a knowledge representation. In *Proceedings Visual Analytics Science and Technology (VAST)*, pages 107–114. IEEE CS Press, 2006.

[8] N. Yau. Flowingdata. `http://flowingdata.com/`.

Robert Kosara
UNC Charlotte
`rkosara@uncc.edu`
`http://eagereyes.org/`

Visualization, Graphics, and Statistics

Andrew Gelman and Antony Unwin [1]

Quantitative graphics, like statistics itself, is a young and immature field. Methods as fundamental as histograms and scatterplots are common now, but that was not always the case. More recent developments like parallel coordinate plots are still establishing themselves. Within academic statistics (and statistically-inclined applied fields such as economics, sociology, and epidemiology), graphical methods tend to be seen as diversions from more "serious" analytical techniques. Statistics journals rarely cover graphical methods, and Howard Wainer has reported that, even in the *Journal of Computational and Graphical Statistics*, 80% of the articles are about computation, only 20% about graphics.

Outside of statistics, though, infographics and data visualization are more important. Graphics give a sense of the size of big numbers, dramatize relations between variables, and convey the complexity of data and functional relationships. Journalists and graphic designers recognize the huge importance of data in our lives and are always looking out for new modes of display, sometimes to more efficiently portray masses of information that their audiences want to see in detail (as with sports scores, stock prices, and poll reports), sometimes to help tell a story (as with annotated maps), and sometimes just for fun: a good data graphic can be as interesting as a photograph or cartoon.

We and other graphically-minded statisticians have been thinking a lot recently about the different perspectives of statisticians and graphic designers in displaying data. But first we would like to emphasize some key places in which we agree with the infographics community, some reasons why we and they generally prefer numbers to be graphed rather than written.

- A well-designed graph can display more information than a table of the same size, and more information than numbers embedded in text. Graphical displays allow and encourage direct visual comparisons.
- It has been argued that tables are commonly read as crude graphs: what you notice in a table of numbers are (a) the minus signs, and thus which values are positive and which are negative, and (b) the length of each number, that is, its order of magnitude. In a table of statistical results you might also note the boldface type or stars that indicate statistical significance. A table is a crude form of log-scale graph. If we really must display numbers in tables with many significant figures, it would probably generally be better to display them like this: 3.14_{16}, so as not to distract the readers with those later unimportant digits.
- A graph can tell a story so easily. A line going up tells one story, a line going down tells another, and a line that goes up and then down is yet another possibility. It is the same with scatterplots and more elaborate displays. Yes, a table of numbers can tell a story too—especially in an area such as baseball where, as sabermetrician Bill James wrote, numbers such as .406 or 61 evoke images and history—but in general the possibilities of storytelling are greater and more direct with a graph. Storytelling is important in journalism and advertising (of course) but also in science, where data can either motivate and illustrate a logical argument or refute it.

In short, graphs are a good way to convey relationships and also reveal deviations from patterns, to display the expected and the unexpected.

Now we turn to differences between statistical graphics and infovis. In statistical graphics we aim for transparency, to display the data points (or derived quantities such as parameter estimates and standard errors) as directly as possible without decoration or embellishment. As indicated by our remarks above, we tend to think of a graph as an improved version of a table. The good thing about this approach is it keeps us close to the data. The bad thing is that it limits our audience. We as statisticians think we're keeping it simple and clean when we display a grid of scatterplots, but the general public—and even researchers in many scientific fields—don't have practice reading these

[1] We thank the Institute of Education Sciences for grants R305D090006-09A and ED-GRANTS-032309-005, and the National Science Foundation for grants SES-1023189 and SES-1023176

graphs, and can often miss the point or simply tune out.

In contrast, practitioners of information visualization use data graphics more generally as a means of communication, in competition (and collaboration with) photographs, cartoons, interviews, and so forth. For example, a news article about health care costs might include some reportage (perhaps with some numbers gleaned from government documents), quotes from experts, an interview with a sick person who cannot get health insurance, a photograph of a high-tech MRI machine, a how-much-do-you-know quiz on the prices of medical procedures—and a data visualization showing medical costs and service use in different parts of the country. The visualization is graded partly on how cool it looks: "cool" grabs the reader's attention and draws him or her into the story.

We hope that, by recognizing our different goals and perspectives, graphic designers and statisticians can work together. For example, a website might feature a dramatic visualization that, when clicked on, reveals an informative static statistical graphic that, when clicked on, takes the interested reader to an interactive graphic and a spreadsheet with data summaries and raw numbers.

We illustrate some of our points with two examples. The first is Florence Nightingale's famous visualization of deaths in the Crimean War. Here is Nightingale's graph from 1958 (for more details, see Hugh Small's presentation at `http://www.florence-nightingale-avenging-angel.co.uk/Coxcomb.htm`):

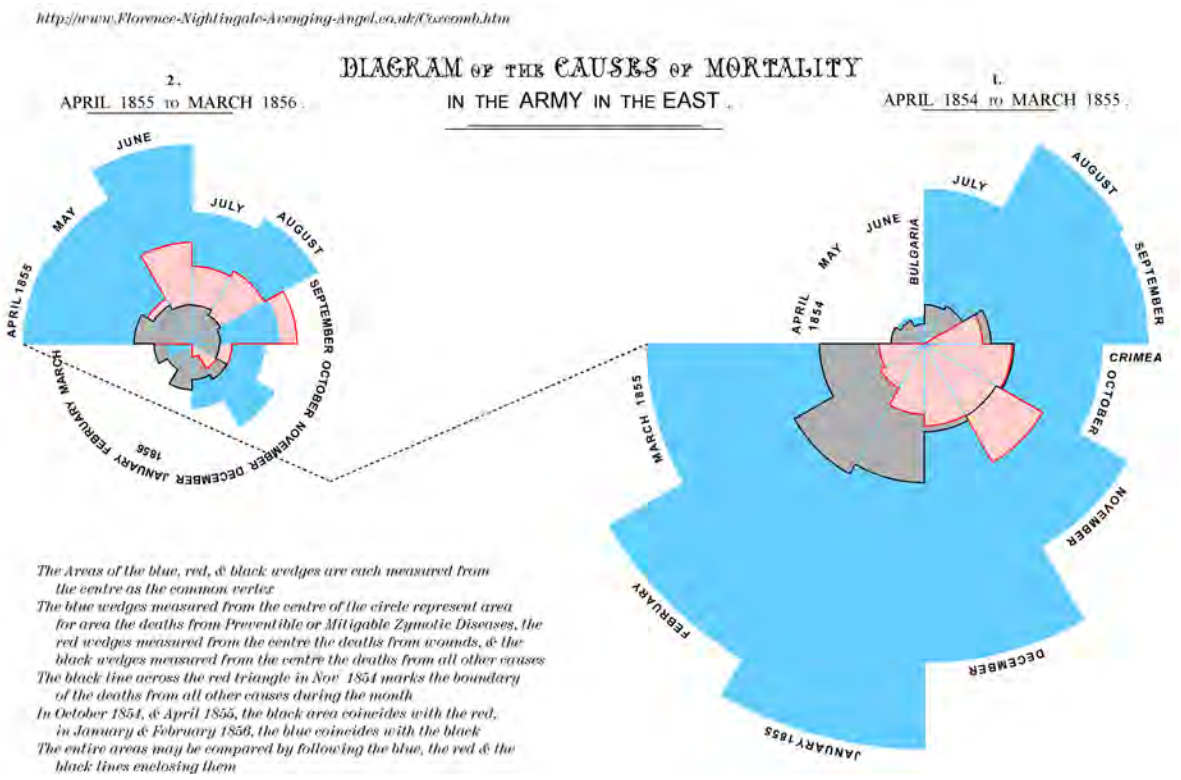


Figure 1: Florence Nightingale's famous visualization of deaths in the Crimean War is attractive and draws the viewer in closer so as to understand what is being conveyed.

And now our presentation of the same information using R:

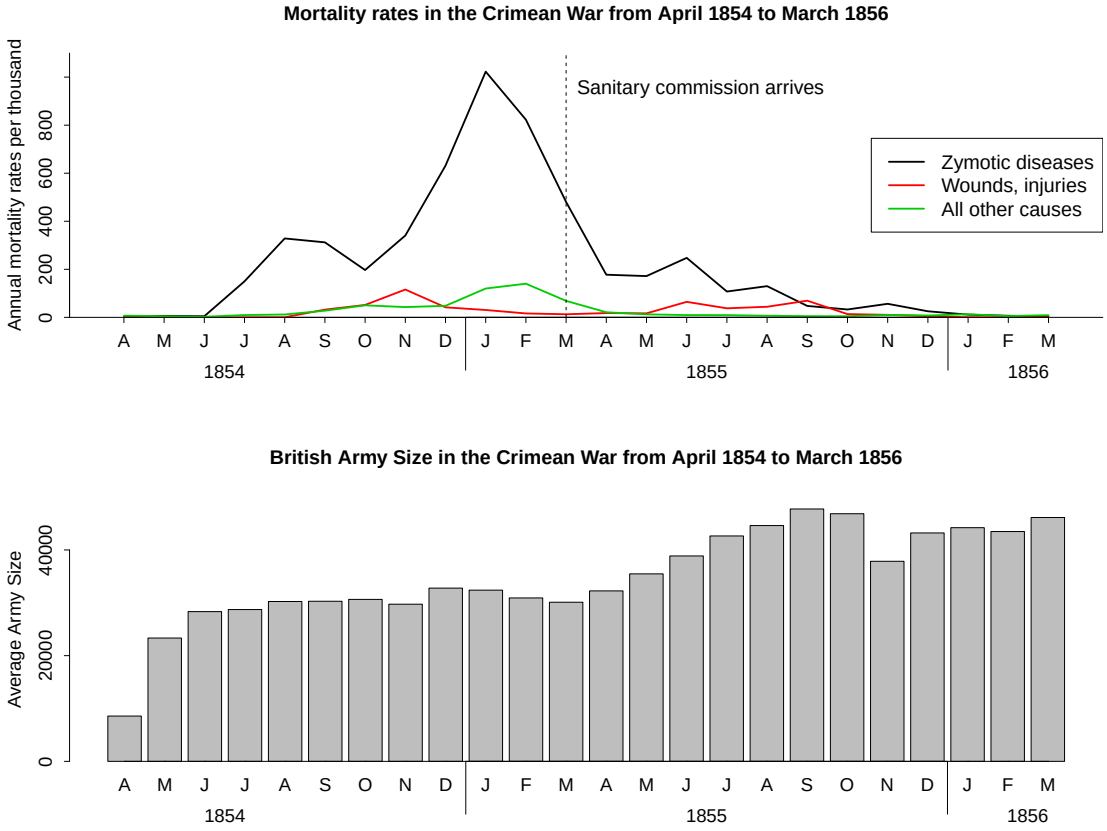


Figure 2: Our re-plotting of Nightingale's data shows the data and their patterns much more clearly, but in a visually less striking way. As is often the case, two smaller plots can show data much more directly than is possible from a single graph, no matter how clever.

Nightingale's visualization and ours both have their strengths. When it comes to displaying the data and their patterns, we much prefer the plain statistical graphs. The most salient visual feature of Nightingale's graph is that a year is divided into twelve months, a fact that we already knew ahead of time. The trends and departures from trend are much clearer when plotted directly as time series. This is no criticism of Nightingale: the standard statistical techniques of today were not so easily available in the mid-1800s, and in any case her graph did the job of attracting attention better than ours do, in any era.

Nightingale's graph is intriguing and visually appealing—much more so than our bland graph—and, as is characteristic of the best infographics, the appeal is centered on the data display itself. A reader who sees this graph is invited to stare at it, puzzle it out, and understand what it is saying. In some ways, the weaknesses of the graph from a statistical point of view—it is difficult to read, the main conclusions to be drawn from the data are not clear, indeed it is a bit of a challenge to figure out exactly what the graph is saying at all—are strengths from the infovis perspective. Given that the graph is attractive enough, and the subject important enough, to motivate the reader to go in deeper, the challenges in reading the graph induce a larger intellectual investment in the viewer and a motivation to see the raw data.

And once policymakers were alerted by Nightingale's dramatic visualization, they were able to scan the columns of numbers directly and understand what was going on: the patterns in

these time series are clear enough that we imagine a careful study of a tabular display would suffice. The role of the graph was to dramatize the problem and motivate people to go back and look at the numbers.

In a modern computing environment, a display such as Nightingale's could link to a more direct graphical presentation such as ours, which in turn could link to a spreadsheet with the data. The statistical graphic serves as an intermediate step, allowing readers to visualize the patterns in the data.

Our second example concerns the survival rates of different groups who sailed on the Titanic's maiden voyage. Here is a doubledecker plot showing the survival rates by sex (males on the left and females on the right) and within sex by class (first, second, third, crew). The widths of the bars are proportional to the numbers in each group, so that we get a rough idea of their relative sizes, though it is the survival rates that are of most interest.

It is easy to see two expected conclusions, that female survival rates were higher than males for all possible comparisons, and that female survival rates went down with class. It is also obvious, though more surprising, that the lowest male survival rate was in the second class. The fact that the male crew survival rate was higher than the male survival rates in the second and third classes must at least partly be due to the lifeboats being manned with crew members to accompany the passengers. All of these conclusions may be drawn directly from the display, but no one would claim it is an attention-grabbing graphic! We looked on the internet (a.k.a. googled) to see if these data had been presented in an infographic display and found several statistical displays, not all either clear cut or easy to read, though no infographic ones. This is a good example where cooperation between statisticians and infographics experts could really pay off: we have an interesting dataset and several interesting conclusions to present and we would like to do it in an attractive and stimulating way without losing any statistical clarity. Just wanting to do that is not enough, we need design expertise, and we look forward to someone from the infographics side taking up the challenge of helping us.

Andrew Gelman
Dep. of Statistics and Department of Political Science
Columbia University, New York
gelman@stat.columbia.edu
http://www.stat.columbia.edu/~gelman/

Antony Unwin
Department of Mathematics
University of Augsburg
unwin@math.uni-augsburg.de
http://www.rosuda.org

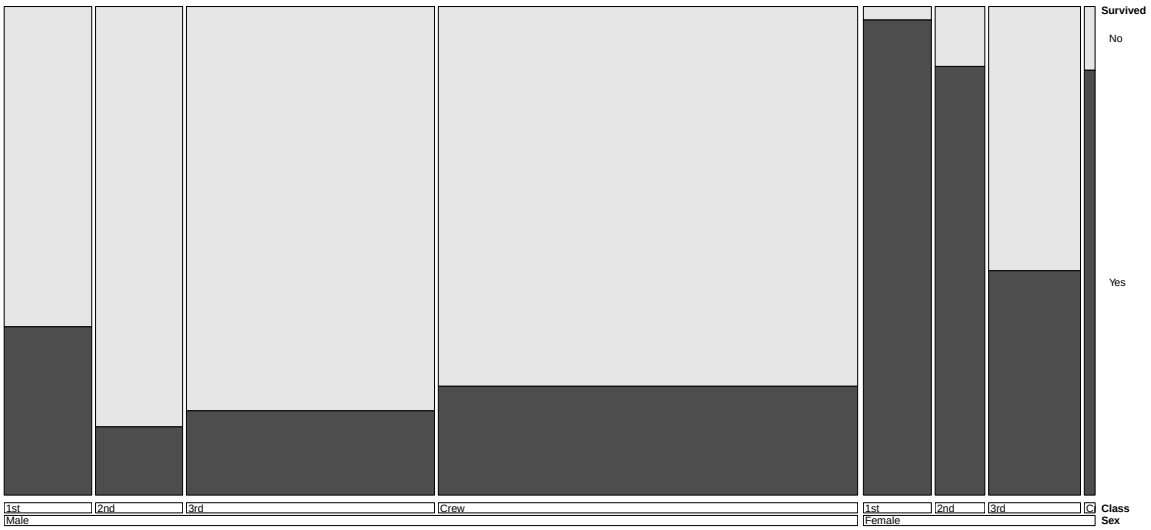


Figure 3: A doubledecker plot showing the survival rates on the Titanic by sex and, within sex, by class. This graph shows several interesting comparisons but could benefit from improvement in graphic design.

Heuristic Methods in Finance

Enrico Schumann and David Ardia [1]

Heuristic optimization methods and their application to finance are discussed. Two illustrations of these methods are presented: the selection of assets in a portfolio and the estimation of a complicated econometric model.

Heuristic methods in Finance

Models in finance

Finance is, at its very heart, all about optimization. In financial theory, decision makers are optimizers: households maximize their utility, firms maximize profits or minimize costs. In more applied work, we may look for portfolios that best match our preferences, or for trading rules that find the ideal point in time for buying or selling an asset. And of course, the ubiquitous estimation or calibration of model parameters is nothing but optimization.

In this note we will describe a type of numerical techniques, so-called heuristics, that can be used to solve optimization models. An optimization model consists of an objective function and possibly a number of constraints, i.e., the model is a precise, mathematical description of a given problem. But the process of financial modeling comprises two stages. We start with an actual problem – such as "how to invest?" – and translate this problem into a model; then we move from the model to its numerical solution. We will be concerned with the second stage. Yet we cannot overemphasize the importance of the first stage. It may be interesting to work on a challenging optimization model, but if the model is not useful than neither is its solution.

It turns out that many financial models are difficult to solve. For combinatorial models it is the size of the search space that causes trouble. Such problems typically have an exact solution method – write down all possible solutions and pick the best one – but this approach is almost never feasible for realistic problem sizes. For continuous problems, issues arise when the objective function is not smooth (e.g., has discontinuities or is noisy), or there are many local optima. Even in the continuous case we could attempt complete enumeration by discretizing the domain of the objective function and running a grid search. But again this approach is not feasible in practice once the dimensionality of the model grows.

Researchers and operators in finance often go a long way to make models tractable, that is, to formulate them such that they can compute the quantities of interest either in closed form or with the computational tools that are at hand. When it comes to optimization, models are often shaped such that they can be solved with "classical" optimization techniques like linear and quadratic programming. But this comes at a price: we have to construct the model in such a way that it fulfills the requirements of the particular method. For instance, we may need to choose a quadratic objective function, or approximate integers with real numbers. To paraphrase John Tukey, when we solve such a model we get a precise answer but it becomes more difficult to say if we have asked the right question.

An alternative strategy is the use of heuristic optimization techniques, or heuristics for short. Heuristics aim at providing good and fast approximations to optimal solutions; to stay with Tukey's famous statement, heuristics may be described as seeking approximate answers to the right questions. (In theory, the solution of a model is the optimum; it is not necessary to speak of optimal solutions. But practically a solution is rather the result that we get from a piece of software, so it is meaningful to distinguish between good and bad solutions.) Optimization heuristics are often very simple, easy to implement and to use; there are essentially no constraints on the model formulation; and any changes to the model are quickly implemented. But of course, there must be a downside: heuristics do not provide the exact solution of the model but only a stochastic approximation. Yet such an approximation may still be better than a poor deterministic solution or no solution at all. If a model can be solved with a classical method, it is no use to try a heuristic. The advantage comes when classical

[1] The views expressed in this paper are the sole responsibility of the authors and do not necessarily reflect those of VIP Value Investment Professionals AG, aeris CAPITAL AG or any of their affiliates. Any remaining errors or shortcomings are the authors' responsibility.

methods cannot solve the given model, i.e., when a model is difficult. Such models are far more common in finance that is sometimes thought.

What is a heuristic?

The aim in optimization is to

$$\underset{x}{\text{minimize}}\, f(x, \text{data})$$

with f a scalar-valued function, and x a vector of decision variables. To maximize a function f, we minimize $-f$. In most cases, this optimization problem will be constrained. Optimization heuristics are a class of numerical methods that can solve such problems. Well-known examples are Simulated Annealing, Genetic Algorithms (or, more generally, Evolutionary Algorithms) or Tabu Search. It is hard to give a general definition of what constitutes a heuristic. Typically, the term is characterized through several criteria such as the following (e.g., Zanakis and Evans [12], Barr et al. [2]): (i) the method should give a "good" stochastic approximation of the true optimum, with "goodness" measured by computing time or solution quality, (ii) the method should be robust to changes in the given problem, in particular the problem size, (iii) the technique should be easy to implement, and (iv) implementing and using the technique should not require any subjective elements. Of course, such a definition is not unambiguous, and even in the optimization literature the term is used with different meanings.

How do heuristics work?

Very roughly, we can divide heuristics into iterative search methods and constructive methods. Constructive methods start with an empty solution and then build a solution in a stepwise manner by adding components until a solution is completed. An example: in a Traveling Salesman Problem we are given a set of cities and the distances between them. The aim is to find the route of minimal length such that each city is visited once. We could start with one city and then add the remaining cities one at a time (e.g., always choosing the nearest city) until a complete tour is created. The procedure terminates once we have found one complete solution.

For iterative search methods, we repeatedly change an existing complete solution to obtain a new solution. Such methods are far more relevant in finance, so we will concentrate on them. To describe an iterative search method, we need to specify (i) how we generate new solutions from existing solutions, (ii) when to accept such a new solution, and (iii) when to stop the search. These three decisions define a particular method; in fact, they are the building blocks of many optimization methods, not just of heuristics. As an example, think of a steepest descent method. Suppose we have a current (or initial) solution x^c and want to find a new solution x^n. Then the rules could be as follows:

(i) We estimate the slope (the gradient) of f at x^c which gives us the search direction. The new solution x^n is then $x^c - \text{step size} \cdot \nabla f(x^c)$.

(ii) If $f(x^n) < f(x^c)$ we accept x^n, i.e., we replace x^c by x^n.

(iii) We stop if no further improvements in f can be found, or if we reach a maximum number of function evaluations.

Problems will mostly occur with steps (i) and (ii). There are models in which the gradient does not exist, or cannot be computed meaningfully (e.g., when the objective function is not smooth). Hence we may need other approaches to compute a search direction. The acceptance-criterion for a steepest descent is strict: if there is no improvement, a candidate solution is not accepted. But if the objective function has several minima, this means we will never be able to move away from a local minimum, even if it is not the globally best one.

Heuristics follow the same basic pattern (i)–(iii), but they have different rules that are better suited for problems with noisy objective functions or multiple minima. In fact, almost all heuristics use one or both of the following principles.

Trust your luck Classical methods are deterministic: given a starting value, they will always lead to the same solution. Heuristics make deliberate use of randomness. New solutions may be created by randomly changing old solutions, or we may accept new solutions only with a given probability.

Don't be greedy When we compute new candidate solutions in the steepest descent method, we choose a (locally) optimal search direction (it is steepest descent after all). Many heuristics put up with "good" search directions, in many cases even random directions. Also,

heuristics generally do not enforce continuous improvements; inferior solutions may be accepted. This is inefficient for a well-behaved problem with a single optimum, but it allows these methods to move away from local minima.

As a concrete example, we look at Threshold Accepting (a variant of Simulated Annealing).

(i) We randomly choose an x^n close to x^c. For instance, when we estimate the parameter values of a statistical model, we could randomly pick one of the parameters and perturb it by adding a bit of noise.

(ii) If $f(x^n) < f(x^c)$ we accept x^n, as before. But if $f(x^n) > f(x^c)$, we also accept it as long as $f(x^n) - f(x^c)$ is smaller than a fixed threshold (which explains the method's name), i.e., we accept a new solution that is worse than its predecessor, as long as it is not too much worse. Thus, we can think of Threshold Accepting as a biased random walk. (Simulated Annealing works the same, but we would accept an inferior solution with a certain probability.)

(iii) We stop, say, after a fixed number iterations.

Stochastic solutions

Almost all heuristics are stochastic algorithms. Running the same technique twice, even with the same starting values, will typically result in different solutions. Thus, we can treat the result (i.e., the decision variables x and the associated objective function value) of a optimization heuristic as a random variable with some distribution D. We do not know what D looks like, but there is a simple way to find out for a given problem: we run a reasonably large number of restarts, for each restart we store the results, and finally we compute the empirical distribution function of these results as an estimate for D. For a given problem (often problem class), the shape of D will depend on the chosen method. Some techniques will be more appropriate than others and give less variable and on average better results. And D will often depend on the settings of the method, most importantly the number of iterations – the search time – that we allow for.

Unlike classic optimization techniques, heuristics can escape from local minima. Intuitively then, if we let the algorithm search for longer, we can hope to find better solutions. Thus the shape of D is strongly influenced by the amount of computational resources spent (often measured by the number of objective function evaluations). For minimization problems, when we increase computational resources, the mass of D will move to the left, and the distribution will become less variable. Ideally, when we let the computing time grow ever longer, D should degenerate into a single point, the global minimum. Unfortunately, it's never possible to ensure this practically.

Illustrations

Asset selection with Local Search

We can make these ideas more concrete through an example, taken from Gilli et al. [5]; sample code is given in the book. Suppose we have a universe of 500 assets (for example, mutual funds), completely described by a given variance–covariance matrix, and we are asked to find an equal-weight portfolio with minimal variance under the constraints that we have only between K_{inf} and K_{sup} assets in the portfolio. This is a combinatorial problem, and here are several strategies to obtain a solution.

(1) Write down all portfolios with feasible cardinality, compute the variance of each portfolio, and pick the one with the lowest variance.

(2) Choose k portfolios randomly and keep the one with the lowest variance.

(3) Sort the assets by their marginal variance. Then construct an equal-weight portfolio of the K_{inf} assets with the lowest variance, then a portfolio of the $K_{\text{inf}} + 1$ assets with the lowest variance, and so on to a portfolio of the K_{sup} assets with the lowest variance. Of those $K_{\text{sup}} - K_{\text{inf}} + 1$ portfolios, pick the one with the lowest variance.

Approach (1) is infeasible. Suppose we were to check cardinalities between 100 and 150. For 100 out of 500 alone we have 10^{107} possibilities, and that leaves us 101 out of 500, 102 out of 500, and so on. Even if we could evaluate millions of portfolios in a second it would not help. Approach (2)

has the advantage that it is simple, and we can scale computational resources (increase k). That is, we can use the trade-off between available computing time and solution quality. Approach (2) can be thought of as a sample substitute for Approach (1). In Approach (3) we ignore the covariation of assets (i.e., we only look at the main diagonal of the variance–covariance matrix), but we only have to check $K_{\text{sup}} - K_{\text{inf}} + 1$ portfolios. There may be cases, however, in which we would wish to include correlation.

We set up an experiment. We create an artificial data set of 500 assets, each with a randomly assigned volatility (the square root of variance) of between 20% and 40%. Each pairwise correlation is set to 0.6. We compute "best-of-k" portfolios (i.e., Approach (2)): we sample 1000 portfolios, and only keep the best one; we also try "best-of-100 000" portfolios and, for intuition, "best-of-1" portfolios (i.e., purely random ones).

Figure 1, in its upper panel, shows the estimated cumulative distribution functions of portfolio volatilities; each curve is obtained from 500 restarts. Such an empirical distribution function is an estimate of D for the particular method. We see that completely random portfolios produce a distribution with a median of about 23.5%. (What would happen if we drew more portfolios? The shape of D would not change, since we are merely increasing our sample size. But our estimates of the tails would become more precise.) We also plot the distribution of the "best-of-1000" and "best-of-100 000" portfolios. For this latter strategy, we get a median volatility below 21%. We also add the results for Approach (3); there are no stochastics in this strategy.

Now let us try a heuristic. We use a simple Local Search. We start with a random feasible portfolio and compute its volatility. This is our current solution x^{c}, the best solution we have so far. We now try to improve it iteratively. In each iteration we compute a new portfolio x^{n} as follows. We randomly pick one asset from our universe. If this asset is already in the portfolio, we remove it; if it is not in the portfolio, we add it. Then we compute the volatility of this new portfolio. If it is lower than the old portfolio's volatility, we keep the new portfolio, i.e., x^{n} replaces x^{c}; if not, we stick with x^{c}. We include constraints in the simplest way: if a new portfolio has too many or too few assets, we always consider it worse than its predecessor and reject it. We run this search with 100, 1000, and 10 000 iterations. For each setting, we conduct 500 restarts; each time we register the final portfolio's volatility. Results are shown in the lower panel of Figure 1.

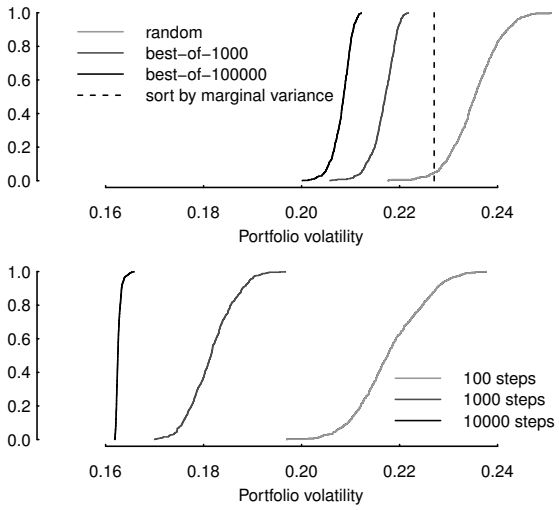


Figure 1: Upper panel: random portfolios. For example: for one restart of the "best-of-100 000" strategy, we sample 100 000 portfolios, and keep the best one. The distributions are estimated from 500 restarts. The sort-by-marginal-variance approach is deterministic, so its result is a constant. Lower panel: Local Search. Each distribution is estimated from 500 restarts.

Already with 1000 iterations we are clearly better than the "best-of-100 000" strategy (though we have used only one-hundredth of the function evaluations). With 10 000 iterations we seem to converge to a point at about 16%. A few remarks: first, we have no proof that we have found the global optimum. But we can have some confidence that we have found a good solution. Second, we can practically make the variance of D as small as we want. With more iterations (and possibly a few other refinements), we could, for all practical purposes, have the distribution "converge". But, third, in many cases we do not need to have D collapse; for financial problems a good solution is fine, given the quality of financial data [6, 7].

Econometric model fitting with Differential Evolution

Our second illustration is taken from Mullen et al. [8], who consider the estimation of a Markov-switching GARCH (MSGARCH) model. MSGARCH are econometric models used to forecast the volatility of financial time series, which is of primary importance for financial risk management. The estimation of MSGARCH models is a nonlinear constrained optimization problem and is a difficult task in practice. A robust optimizer is thus required. In that regard, the authors report the best performance of the Differential Evolution (DE) algorithm compared with traditional estimation techniques.

DE is a search heuristic introduced by Storn and Price [10] and belongs to the class of evolutionary algorithms. The algorithm uses biology-inspired operations of crossover, mutation, and selection on a population in order to minimize an objective function over the course of successive generations. Its remarkable performance as a global optimization algorithm on continuous problems has been extensively explored; see, e.g., Price et al. [9].

Let NP denote the number of parameter vectors (members) $x \in \mathbb{R}^d$ in the population, where d denotes the dimension. In order to create the initial generation, NP guesses for the optimal value of the parameter vector are made, either using random values between bounds or using values given by the user. Each generation involves creation of a new population from the current population members $\{x_i \,|\, i = 1, \ldots, NP\}$, where i indexes the vectors that make up the population. This is accomplished using *differential mutation* of the population members. An initial mutant parameter vector v_i is created by choosing three members of the population, x_{i_1}, x_{i_2} and x_{i_3}, at random. Then v_i is generated as $v_i = x_{i_1} + F \cdot (x_{i_2} - x_{i_3})$, where F is a positive scale factor whose effective values are typically less than one. After the first mutation operation, mutation is continued until d mutations have been made, with a given crossover probability. The crossover probability controls the fraction of the parameter values that are copied from the mutant. Mutation is applied in this way to each member of the population. The objective function values associated with the children are then determined. If a trial vector has equal or lower objective function value than the previous vector it replaces the previous vector in the population; otherwise the previous vector remains. Note that DE uses both strategies described above to overcome local minima: it does not only keep the best solution but accepts inferior solutions, too; the method evolves a whole population of solutions in which some solutions are worse than others. And DE has a chance ingredient as it randomly chooses solutions to be mixed and mutated. For more details, see Price et al. [9] and Storn and Price [10].

We report below some results of Mullen et al. [8], who fit their model to the Swiss Market Index. For the DE optimization, the authors rely on the package **DEoptim** [1] which implements DE in the R language [3]. For comparison, the model is also estimated using standard unconstrained and constrained optimization routines available in R as well as more complex methods able to handle non-linear equality and inequality constraints. The model estimation is run 50 times for all optimization routines, where random starting values in the feasible parameter set are used when needed (using the same random starting values for the various methods). Boxplots of the objective function (i.e., the negative log-likelihood function, which must be minimized) at optimum for convergent estimations is displayed in Figure 2. We notice that standard approaches (i.e., function `optim` with all methods) perform poorly compared with the optimizers that can handle more complicated constraints (i.e., functions `constrOptim`, `constrOptim.nl` and `solnp`). DE compares favorably with the two best competitors in terms of negative log-likelihood values and is more stable over the runs.

Conclusion

In this note, we have briefly described optimization heuristics, but of course we could only scratch the surface of how these methods work and where they can be applied. After all, for most people optimization is a tool, and what matters is how this tool is applied. Heuristics offer much in this regard: they allow us to solve optimization models essentially without restrictions on the functional form of the objective function or the constraints. Thus, when it comes to evaluating, comparing, and selecting models, researchers and operators can focus more on a model's financial or empirical qualities instead of having to worry about how to handle it numerically. We have argued initially that financial modeling comprises two stages: putting an actual prob-

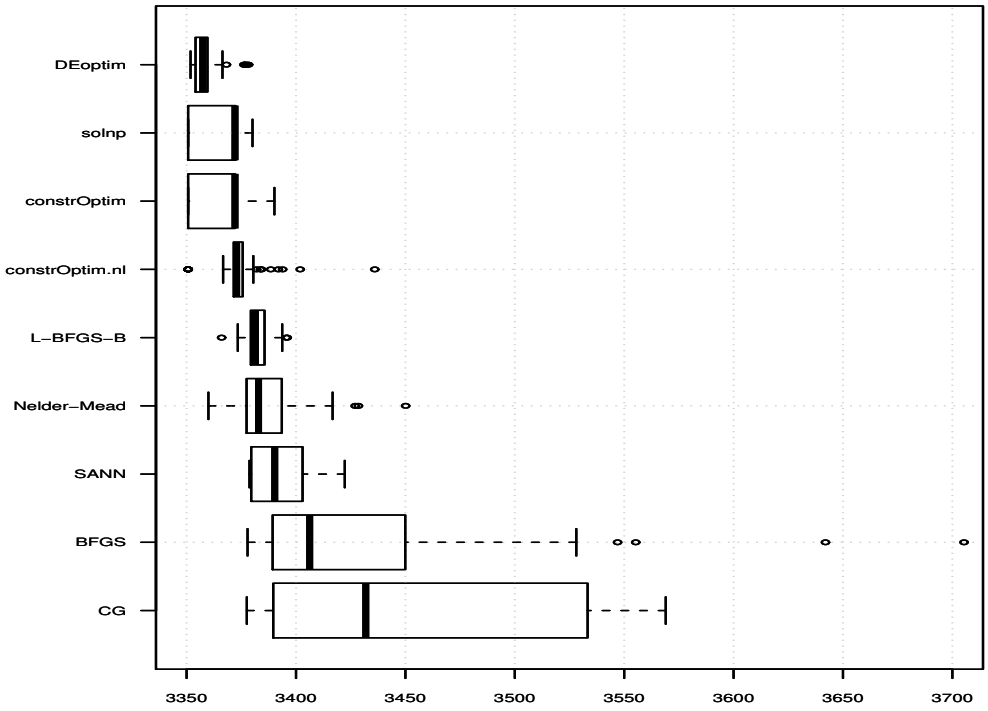


Figure 2: Boxplots of the 50 values of the objective function (i.e., the negative log-likelihood) at optimum obtained by the various optimizers available in R. Function `optim` with method `"Nelder-Mead"` (unconstrained), method `"BFGS"` (unconstrained), method `"CG"` (unconstrained), method `"L-BFGS-B"` (constrained), method `"SANN"` (unconstrained), function `constrOptim` (constrained), function `constrOptim.nl` of the package **alabama** [11], function `solnp` of the package **Rsolnp** [4], function `DEoptim` of the package **DEoptim** [1]. More details can be found in Mullen et al. [8].

lem into model form, and then solving this model. With heuristics, we become much more powerful at the second stage; it remains to use this power in the first stage.

Bibliography

[1] Ardia, D., Mullen, K. M., Peterson, B. G., Ulrich, J., 2011. **DEoptim**: Differential Evolution Optimization in R.
URL `http://CRAN.R-project.org/package=DEoptim`

[2] Barr, R. S., Golden, B. L., Kelly, J. P., Resende, M. G. C., Stewart, W. R., 1995. Designing and reporting on computational experiments with heuristic methods. Journal of Heuristics 1 (1), 9–32.

[3] R Development Core Team, 2009. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0.
URL `http://www.R-project.org`

[4] Ghalanos, A., Theussl, S., 2010. **Rsolnp**: General Non-linear Optimization Using Augmented Lagrange Multiplier Method.
URL `http://CRAN.R-project.org/package=Rsolnp`

[5] Gilli, M., Maringer, D., Schumann, E., 2011. Numerical Methods and Optimization in Finance. Elsevier.

[6] Gilli, M., Schumann, E., 2010. Optimal enough? Journal of Heuristics 17 (4), 373–387.

[7] Gilli, M., Schumann, E., 2010. Optimization in financial engineering – an essay on 'good' solutions and misplaced exactitude. Journal of Financial Transformation 28, 117–122.

[8] Mullen, K. M., Ardia, D., Gil, D. L., Windover, D., Cline, J., 2011. **DEoptim**: An R package for global optimization by differential evolution. Journal of Statistical Software 40 (6), 1–26.

[9] Price, K. V., Storn, R. M., Lampinen, J. A., 2006. Differential Evolution: A Practical Approach to Global Optimization. Springer-Verlag, Berlin, Germany.

[10] Storn, R., Price, K., 1997. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11 (4), 341–359.

[11] Varadhan, R., 2010. **alabama**: Constrained Nonlinear Optimization.
URL `http://CRAN.R-project.org/package=alabama`

[12] Zanakis, S. H., Evans, J. R., 1981. Heuristic "optimization": Why, when, and how to use it. Interfaces 11 (5), 84–91.

Enrico Schumann
VIP Value Investment Professionals AG, Switzerland
`es@vipag.com`

David Ardia
aeris CAPITAL AG, Switzerland
`da@aeris-capital.com`

Section Officers

Statistical Computing Section Officers 2011

Richard M. Heiberger, Chair
rmh@temple.edu
(215) 808-1808

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

Kren Kafadar, Chair-Elect
kkafadar@indiana.edu
(812) 855-7828

Usha S. Govindarajulu, Secretary/Treasurer
usha@alum.bu.edu
(617) 525-1237

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

David J. Poole, Program Chair
poole@research.att.com
(973) 360-7337

Chris Volinsky, Program Chair-Elect
volinsky@research.att.com
(973) 360-8644

Hadley A. Wickham, Publications Officer
hadley@rice.edu
(515) 450-8171

John Castelloe, Computing Section Representative
John.Castelloe@sas.com
(919) 531-5728

Rick G. Peterson (see right)

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

Statistical Graphics Section Officers 2011

Juergen Symanzik, Chair
symanzik@math.usu.edu
(435) 797-0696

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

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

John W. Emerson, Secretary/ Treasurer
john.emerson@yale.edu
(203) 215-3540

Mark Greenwood, GRPH COS Rep 10-12
greenwood@math.montana.edu|
(406) 994-1962

Michael Lawrence, GRPH COS Rep 11-13
michafla@gene.com
(515) 708-3239

Webster West, Program Chair
websterwest@yahoo.com
(803) 351-5087

Hadley A. Wickham, Program Chair-Elect
hadley@rice.edu
(515) 450-8171

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

Rick G. Peterson, ASA Staff Liaison
rick@amstat.org
(703) 684-1221

Martin Theus, Newsletter Editor
martin@theusRus.de

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`

Martin Theus
Editor Statistical Graphics Section
`martin@theusRus.de`

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`