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

Summer 98 Vol.9 No.1

Statistical COMPUTING & GRAPHICS

A WORD FROM OUR CHAIRS

Statistical Computing

Karen Kafadar is the 1998 Chair of the Statistical Computing Section. In her column she considers the Section Charter and the opportunities it represents.

It is a pleasure to serve as your Section Chair this year along with the other members of the Executive Board. During the time since you received your last Newsletter, the Section has been involved in several activities. Our Newsletter editors, Mark Hansen and Antony Unwin have coordinated some extremely fine articles into this issue. Daryl Pregibon provided superb leadership during his (second) tenure as Chair last year. Russ Wolfinger organized an excellent technical program of invited

CONTINUED ON PAGE 2

Statistical Graphics

Michael Meyer is the 1998 Chair of the Statistical Graphics Section. He encourages Section members to contact the Executive Committee with ideas for new Section initiatives.

The Newsletter has been in transition this year so this is my first chance to write to our Section members. It has also been a transition year for me. I am in the process of leaving Carnegie Mellon, after more years than I care to count, and starting a new position with Boeing in Seattle.

There are many ways for Section members to express opinions about the Section and to suggest worthwhile Section projects. A good way is to send email to the

CONTINUED ON PAGE 3

SPECIAL FEATURE ARTICLE

Interactive Education: A Framework and Toolkit

By Deborah Nolan and Duncan Temple Lang

In the past, the teaching of statistics was centered around textbooks and lectures. Concepts were introduced in a series of lectures with a more detailed discussion available in a chapter of the text. Frequently, the chapter ended with short questions aimed at helping the student gain experience with the material. These exercises were typically oriented towards the practice of statistical computation, and involved unnaturally small datasets. Indeed, the data were often presented as summary tables sufficient for performing formulaic computations, and visualizing data required graph paper and accordingly was done infrequently in a course.

The advent of computers allowed larger datasets to be used and students to make individual observations. The machine became responsible for the mechanics of the computations and the students were in charge of determining what computations were done. Visualization was primitive in comparison with the current methods but nevertheless a powerful new feature. Simulations were possible as a way to illustrate the nature of randomness. One challenge with this newer technology was to connect the concepts described in a textbook to the interaction with the machine. The students perhaps spent more time learning the details of the computing environment than the statistical concepts they were supposed to explore.

And this brings us to the present era in which we have a new computer interface – multimedia. The explosion of Internet activities and the new GUI and Web-oriented programming languages such as Java and the prevalence of inexpensive windows-oriented hardware offer exciting opportunities for developing new instructional

CONTINUED ON PAGE 4

EDITORIAL

Introducing the new Graphics Section Editor

[From Mark] As you might have noticed from the back page, the new Graphics Section Editor of the Newsletter is Antony Unwin from the University of Augsburg. Many of you are probably familiar with Antony and his impressive work on statistical graphics and data visualization. I had the opportunity to meet him at a recent conference on "Data Visualization in Statistics" hosted by Andreas Buja at AT&T Labs, Research. I am confident that we will be able to maintain the high quality you have come to expect from the Newsletter.

This edition of the Newsletter also marks the first I have produced without Mario Peruggia. It was actually quite difficult to envision pulling this issue together without him. During our two years as co-editors we have introduced a number of Newsletter innovations. For example, we have made a commitment to color content as well as creating a complete PDF version of the Newsletter (available through `cm.bell-labs.com/who/cocteau/newsletter`). Mario has given quite a lot during these last two years and I am going to miss working with him. Please join me in thanking Mario for a job very well done.

Finally, I want to mention a conference on "Statistical Science and the Internet" that took place just after the visualization meeting I alluded to above. I think it deserves a comment (and not solely because I had a hand in organizing the affair). In short, this workshop brought together several influential members of the computing community to assess the impact that emerging Internet-related technologies will have on the practice of statistics. Our goals for this meeting fall in line with items 1, 3 and 5 of the Computing Section Charter discussed by Karen Kafadar in her column. For more details about the workshop, check out `cm.bell-labs.com/who/cocteau/comsci`.

[From Mark and Antony] In this issue, we feature an article by Deborah Nolan and Duncan Temple Lang on TILE, a Toolkit for an Interactive Learning Environment. In addition to some impressive pedagogy, Deb and Duncan provide us with a nice example of how software of this type *should* be designed and implemented. Next, we have an article by Martin Theus and Matthias Schonlau on detecting intrusion into computer systems based on structural zeroes of user/command contingency tables. John Emerson presents an S-PLUS implementation of mosaic displays and illustrates their use on television viewer data from Nielsen Media Research. Finally, Dan Carr and Company introduce us to linked micromap plots, a template for the display of spatially indexed statistical summaries.

As usual, we would like to thank our contributors. Their efforts provide a tremendous service to the Sections.

Mark Hansen
Editor, Statistical Computing Section
Bell Laboratories
`cocteau@bell-labs.com`

Antony Unwin
Editor, Statistical Graphics Section
Universität Augsburg
`unwin@uni-augsburg.de`

FROM OUR CHAIRS (Cont.). . .

Statistical Computing

CONTINUED FROM PAGE 1

and contributed paper and poster sessions at the Joint Statistical Meetings in Dallas (see p.33), and Tom Devlin has completed his first year as Electronic Communications Liaison (check out the Section's Web page, `www-stat.montclair.edu/asascs`, reachable through ASA's home page). Merlise Clyde has provided us with excellent and timely information during this second year of her term as Secretary and Treasurer, and Lionel Galway, our first Awards Officer, has coordinated this year's student paper competition and also instituted a "Best JSM paper" award to be given for the first time in Dallas. It's been a real pleasure to work with such a talented group of people.

When my term started, I reread our Section's charter (also available on the Web), to remind myself of our mission within the organization:

1. Encourage the application of computer hardware, software, and systems to statistical problems;
2. Encourage the application of statistical techniques in the design, maintenance, and evaluation of computer hardware, software, and systems;
3. Encourage the joint application of statistical techniques and computer technology in other fields;
4. Serve as the focal point of computer-oriented activities within the ASA including cooperation and liaison with computer-oriented organizations and developers and vendors of statistical software;

5. Encourage research in statistical computing and communication of the results of this research.

Thanks to the many efforts of my predecessors over the years, evidence of point #4 lies in our excellent relations with software vendors particularly at the JSM and efforts such as Daryl Pregibon's participation at the KDD'97 conference following last year's JSM. Many members are involved with meetings such as the Interface Symposium, along the lines of the first three points. It seems to me that point #5 continues to offer our Section some exciting opportunities to influence statistical methodology across the entire organization. As Trevor Hastie noted in his column as Chairman four years ago (December 1994, 5(3), p.4), the impact of computing on statistical practice is significant over the past decade and continues to be into the millennium. Many of the commonly used tools depend, by the way they are defined as well as implemented, on computing, among them the bootstrap, Markov chain Monte carlo, wavelets, generalized estimating equations, classification and clustering methods. As the largest single section of the ASA, we have a real opportunity to influence the organization by encouraging the continued use and development of statistical computing methods in teaching and research. This encouragement comes in part from the Executive Board's support of activities such as the 1997 Data Exposition at the JSM, the Undergraduate Data Analysis contest, student awards for travel to meetings, and the New Researcher's conference held in the fall of 1997. But it also comes in part, and more importantly, from member participation in these Section-sponsored events and activities, such as attendance and presentation at Statistical Computing sessions at the JSM and the Symposium on the Interface, submission of articles to the Newsletter, and simply using such methods to promote their use in teaching and applications. Collectively, your individual participation combines to have a major impact on the directions that are taken by the entire organization. If you have other suggestions on how your Section can further stimulate the development of statistical methods, please let us know; we welcome your participation. We look forward to hearing from you!

Karen Kafadar
University of Colorado at Denver
kk@tiger.cudenver.edu

Statistical Graphics

CONTINUED FROM PAGE 1

current Section Chair or anyone on the Executive Committee. Another good way is to attend the joint Computing and Graphics Mixer at the annual meetings. This year the mixer is again on Monday evening at 6:30pm. Check your program for the location. The Graphics Section Executive Committee meets at 7:00am (just great for the west coast members) that same day. If you wish to make a proposal or presentation at that meeting, please contact me.

I would like to congratulate the newly elected Section officers. The terms of these officers start in January 1999, so the 1999 Chair-Elect is really the Chair for 2000. The new officers are 1999 Chair-Elect, Edward J. Wegman, George Mason University, 1999 Program Chair-Elect, Daniel Carr, George Mason University, 1999 Secretary/Treasurer, Robert L. Newcomb, University of California, 1999 Publications Officer, Michael C. Minnotte, Utah State University, and (1999–2001) Council of Sections Representative, Bradley Jones, SAS Institute. Special thanks are due to Bob Newcomb and Michael Minnotte who are continuing in their offices.

In addition to being Chair-Elect-Elect, Ed Wegman has been our 1998 Program Chair. He has put together an interesting program of invited and special contributed papers. The full program of Graphics papers including abstracts are available in the JSM section of the ASA Web site. I encourage all of you to attend as many sessions as possible, and I especially encourage you to contact the new Program Chairs (Deborah Swayne for 1999, and Dan Carr for 2000) with suggestions and offers for future programs.

Finally I wish to make a plea to you about the use of Section funds. Over the last few years we have tried to use Section funds to support the use of special audio-visual equipment at the annual meetings and in various ways to support undergraduate and graduate students who are working in statistics. We are always open to other suggestions for the use of Section funds. Again, contact anyone on the Executive Committee if you have any ideas.

Mike Meyer
Carnegie Mellon University (for now)
mikem@stat.cmu.edu

SPECIAL FEATURE ARTICLE (Cont.)

CONTINUED FROM PAGE 1

material. Hypertext, images, interactive interfaces and components, sound and so on change the way we interact with machines. Also, current standards change a user's expectations of software in all realms of life, including education. It seems unlikely that a simple translation of the material designed for such different media will be effective in or take advantage of the new possibilities offered by multimedia. A more serious consideration of how to teach in this new environment is necessary and an innovative curriculum needs to be developed.

So how do we develop such a curriculum? Statisticians have high level languages in which they do their daily work, however it seems that few of these systems provide, or even easily allow, a convenient user interface for introductory statistics students. We don't want a student's attention to be focused on the software, nor do we want to teach them how to use software they will never use again. We want to teach statistical concepts with software that doesn't get in the way and is intuitive to the user. But we still want access to many of the statistical procedures available in the common systems. Programming languages such as Matlab and S are too complicated for many students; Jump and Systat and others are more accessible but provide more of a recipe book for students who already know what they want to do. We would like to put the concepts more in the context of a problem in which statistical reasoning and techniques are used rather than simply teaching techniques. The multimedia approach seems ideal for this sort of interaction.

Some commercial groups have recognized the limitations of practitioner-oriented software for teaching statistics, and have developed applications with different educational focuses. These organizations have greater technical capabilities to create software than most university departments, but the latter should have a better understanding of the pedagogical issues. It would seem ideal to have professors provide the content to the software developers. For this to happen, technical capabilities need to be provided at a high level to universities so that faculty might easily customize and modify content while taking advantage of a well designed interface.

With all these issues in mind, we would like to describe some of our efforts in the area of statistical education. These efforts have been in two directions: developing course material for a multimedia computing environment; and providing instructors with flexible and extensible tools to support development of this material. They represent work done part-time over the last one and a half years and from an earlier incarnation of the same project (both funded by the National Science Foundation) with Leo Breiman and Roger Purves. In that early stage, the focus was on developing numerous computer labs to help teach some of the basic concepts of statistics to students in introductory non-calculus statistics courses. Those discussions have had a significant impact on our ideas and also illustrated that different instructors have different pedagogical philosophies that could not be satisfied by a single set of labs. Instead, we need a way for each instructor to define his own material when necessary and to share other labs where available. Accordingly, the focus of our current project both illustrates a more ambitious pedagogical philosophy suited to the multimedia environment and provides the tools to build and customize the different types of labs for all philosophies. Our work in these two arenas have gone hand-in-hand, with technology opening the door to an interactive and individualized learning environment for students, which in turn shaped the design of a toolkit to support instructor development of course material through simple scripting languages and properties files.

We have developed labs that capitalize on the pedagogical advantages of the computer, by making extensive use of real data sets, numerical simulation, and visualization. In the process, we have found one of the most difficult aspects of designing a lab is to determine how to combine these capabilities with the new multimedia technology in order to convey statistical ideas in an interesting, useful, and fun way. We would like to outline a few of our considerations in the development of four labs that explore different statistical concepts and pedagogical techniques through a description of one of these labs. We would also like to give a flavor for how the toolkit supports several different programming levels and interfaces to facilitate developing and modifying a lab. We will do this by describing three of its more unique modules.

Pedagogical Considerations

The main goal in developing these labs is to teach students how to think critically (and statistically) about quantitative problems that are real and important to them. We are not concerned as much with statistical methodology, or software expertise, as with concepts. We follow in spirit the goals of *Statistics*, by Freedman, Pisani, and Purves (1997), but the labs can be used in

conjunction with a variety of statistics courses and textbooks. They are stand-alone units that are meant to be used outside the classroom, not as a replacement for the instructor. Although we think that multimedia is useful to the curriculum, we also believe that value remains in having instructors teach courses using textbooks.

Our efforts have concentrated on the translation of teaching activities into an effective interactive, educational, computing environment. We have focused on how to employ topical contexts that naturally emphasize statistical concepts, and on how to design an interface that is helpful, intuitive, flexible, and educational. We have made the labs both context-based and context adaptable so instructors can tailor the material to the interest of the students.

Each lab is substantial, and more involved than traditional questions concluding the chapter of a text. Students may complete them over several computer sessions. Accordingly, they are not calculator-like environments with which a student simply performs a statistical procedure. Rather than being an interactive component of a textbook, these labs extend the syllabus by putting techniques learned in the classroom in a more interactive and applied setting, where students have the opportunity to come up with their own solutions.

We have found animations are useful for representing a behind-the-scene calculation such as the simulation of 1,000 draws from a pool of tickets. Animation can also be effectively used to run a virtual experiment in order to see how data are created, and to connect a data point with a measurement taken on a subject.

Simulation is a powerful tool for demonstrating probability laws such as the central limit theorem and for finding the distribution of a test statistic. However, simulations can also be boring and not intuitive. We have taken care to include animations with simulations to help explain what is happening, and to limit the use of simulations used only as a computational device.

To maintain consistency across labs, we settled on a single interface for the labs that minimizes the complexity of the application. This uses a work and a help window and provides easy navigation between these and the elements of the work window. Additionally, quiz and animation controls are presented in a common way. The overall interface is similar to several Internet browsers with which many students are already familiar. The navigation system allows students to easily move between exercises, and the help system supports dynamic hints and updates on status. The quiz system gives a student customized feedback, where quiz questions are immediately graded and students receive feedback that depends on the responses given.

The labs are designed to support a variety of levels of expertise. For example, the lab described in the next section uses several practice rounds to introduce the ideas and a more difficult challenge round to encourage the student to think more deeply about the concepts. Another lab on observational studies is similar in spirit to an adventure game offering tasks at different levels of expertise.

We provide more concrete examples of these pedagogical considerations in the example below.

An Example - Design of Experiments

We developed this lab in an attempt to convey to students some basic elements of an experiment, be they medical or industrial, etc. For the source of our narratives, we use excerpts of newspaper articles, accompanied by summaries of journal articles. In our experience working directly with news and journal articles in the classroom (Gelman, Nolan, Men, Warmerdam and Bautista, 1998) we have found this medium to be effective in developing students quantitative literacy. It's also popular among students, because they can relate the stories to their own lives. By comparing and contrasting news and journal reports on an experiment, students hone their skills in critical thinking, synthesizing multiple sources and identifying important statistical elements.

From a nontechnical description of an experiment, we want the student to be able to answer the following questions:

Who were the subjects?

How were they split into treatment groups?

What was the treatment?

What response was measured on the subjects?

Additional important lessons we expect students to take away from this lab address the importance of having comparison groups; randomization; blindness; and generalizing results to populations other than those studied.

We try to present experiments on interesting topics with unusual features or that offer important lessons on the above topics. For example, many students have trouble identifying the subjects of an experiment concerned with the effect of equipping fishing nets with acoustic alarms on the number of porpoises that get entangled in the nets (see Figure 2). The students are more likely to think the porpoises are the subjects of this experiment than the nets. After all, its the porpoises that are getting caught in the nets!

To begin, the lab starts with a brief set of instructions and a few sentences on the topic of the lab. Students are then presented with a set of experiments in the Topics window (Figure 1). In the example shown here the student is presented with three experiments. Those on fishing and surgeons are practice rounds and that on caffeine is what we call the challenge round. The instructor can choose any number of practice rounds for the lab, and the experiments can be selected randomly for each student from a subset of those available for the lab. With random selection, students have less incentive to share results and get a greater diversity of scenarios from which to learn the statistical abstractions.

The enclosing window interface in Figure 2 is used in all of our labs, in an effort to provide a familiar and consistent appearance to the student. This helps the student get started quickly on the content of the lab rather than wasting time trying to figure out the controls. Standard tools (quit-and-save-session, print, help, etc.) are available in the menu bar along with lab-specific actions. A message area and assistant, from which the student can obtain hints, are located at the bottom of the window. The experiment icons below the menu bar comprise the navigation bar allowing the student to move between experiments in any order. Stacking the experiments allows us to use a single work window. The only other window displays the hypertext help system; it contains pages on the different topics in the lab and a dictionary of statistical terms. Finally, tooltips or messages appear over most components.

To begin work on an experiment, the student selects an experiment from the topics window. This brings her to the main work area shown in Figure 2. In the practice rounds, the design is displayed on the right hand side in the form of a tree. The different levels of the tree represent the different stages in the experiment - enrollment, randomization, treatment and measurement. Where there are two boxes in a given level, the observational units have been separated by some earlier splitting criteria giving us different groups.

When the student first sees the design, the boxes are empty. The task is to fill in the boxes in the tree with the details of the experiment by relating the content of the newspaper and journal articles to the different stages. They do this by identifying text in the articles that relates to the particular box they are working on. For example, the phrase “number of porpoises caught” is represented by the porpoise icon in the Measure box, and “randomly” by a coin in the Split stage. When the student clicks on the phrase, it is entered into the box pictorially as an icon. All the phrases in the text relating to this icon are highlighted in the color of the box as if the student used a high-liter to mark important passages in the text. If the student selects text that doesn’t go into the box, a message giving some indication why the match is inappropriate is displayed in the message bar at the bottom.

After completing the contents of the design, the student moves to the animation component of the experiment. Here the design is displayed in a cartoon-like way and the student sees the subjects - shown as little green faced fellows in Figure 3 - participate in the experimental process. Some are rejected at the enrollment gate. Others move through the different stages being randomized in the centrifuge, treated in the blue box, and measured on the scale. Finally, the subjects fall into the appropriate histogram, adding their value to the appropriate bin. This animation helps to illustrate the whole process to the student and connect the data point to a particular subject. This allows the student to think of data as coming from a process rather than a table at the end of a textbook’s chapter.

Once the virtual experiment is complete, the student proceeds to answer questions about the design. The purpose of these questions is to focus the student on some of the more important aspects of the experiment. For example, one question that follows an experiment about emergency medical treatment for stab and gunshot victims asks about randomization. As it happens, the treatment a wounded individual receives is determined according to whether the date is even or odd, not by randomization. Students sometimes confuse the deliberate alternation of treatment assignments with a chance process.

The questions in the wrap–up of the experiment are not intended to be graded. They are grouped into pages, and the student submits one page of questions at a time. Her answers are checked immediately; a profile of the student is built from her responses; and she receives individualized and detailed comments to help clarify important issues before proceeding to the next page of questions. In the stab-wound example, the student is asked in a subsequent page for reasons why the experimenter may have chosen a nonrandom assignment procedure.

The student may work on the practice experiments in any order, completing them sequentially or simultaneously. Once she has finished all the practice rounds, she can move to the challenge round, where the goal is to designs her own experiment (Figure 4). She is given a glossary of terms each identifying: a characteristic of the subject, a procedure for splitting the subjects into groups, a type of treatment, etc. She is also offered

Figure 1 (Topics Screen). Describes the practice and challenge experiments and allows the student to navigate between them.

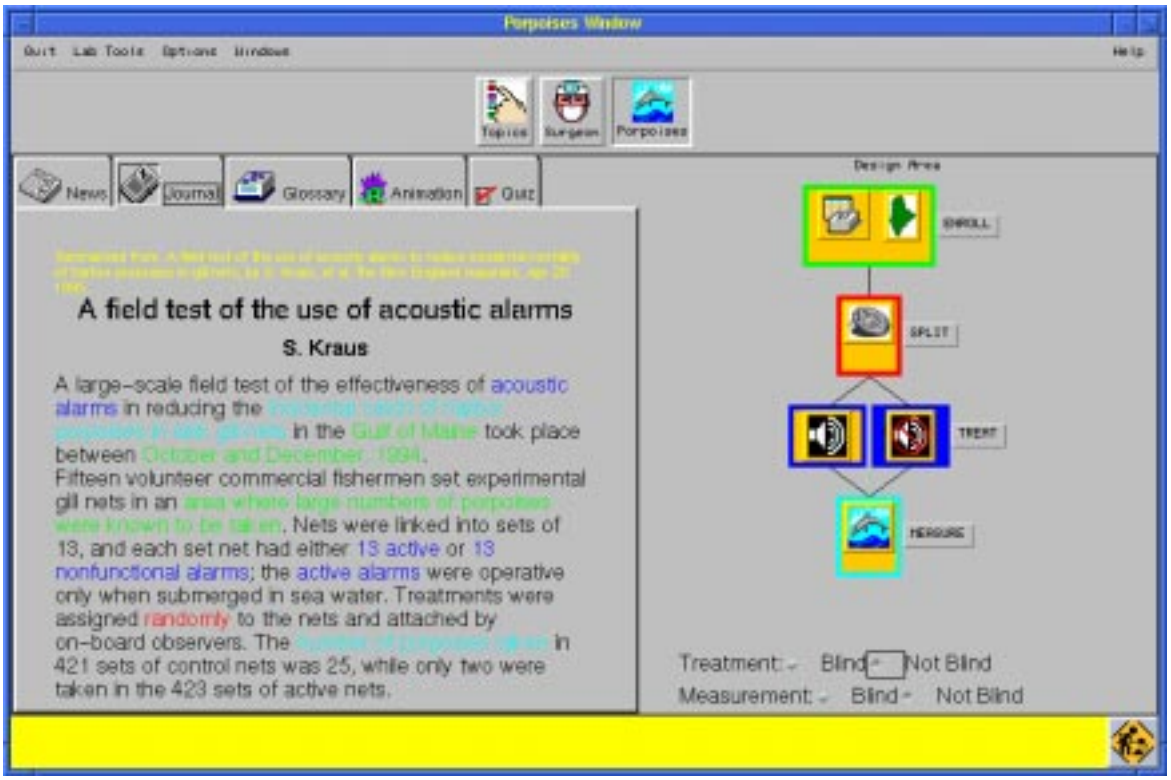


Figure 2 (Practice Round). Students add icons to the boxes in the design by clicking on phrases in the articles on the left. The phrases are higlighted in the color of the box to which they have been added.

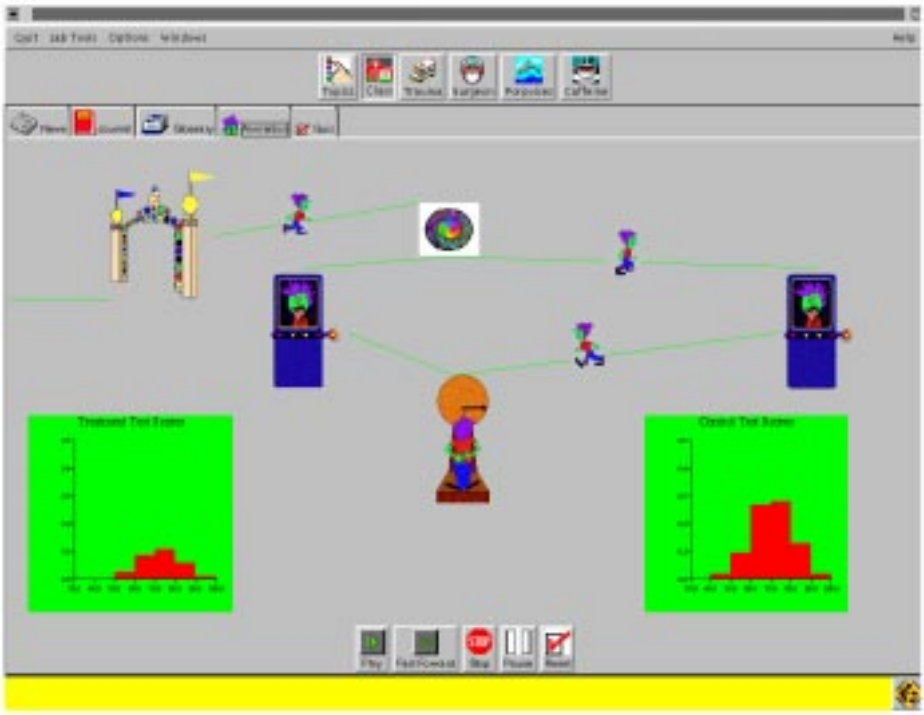


Figure 3 (Animation). Subjects are rejected or enrolled, randomized, treated and measured, building the histograms incrementally.

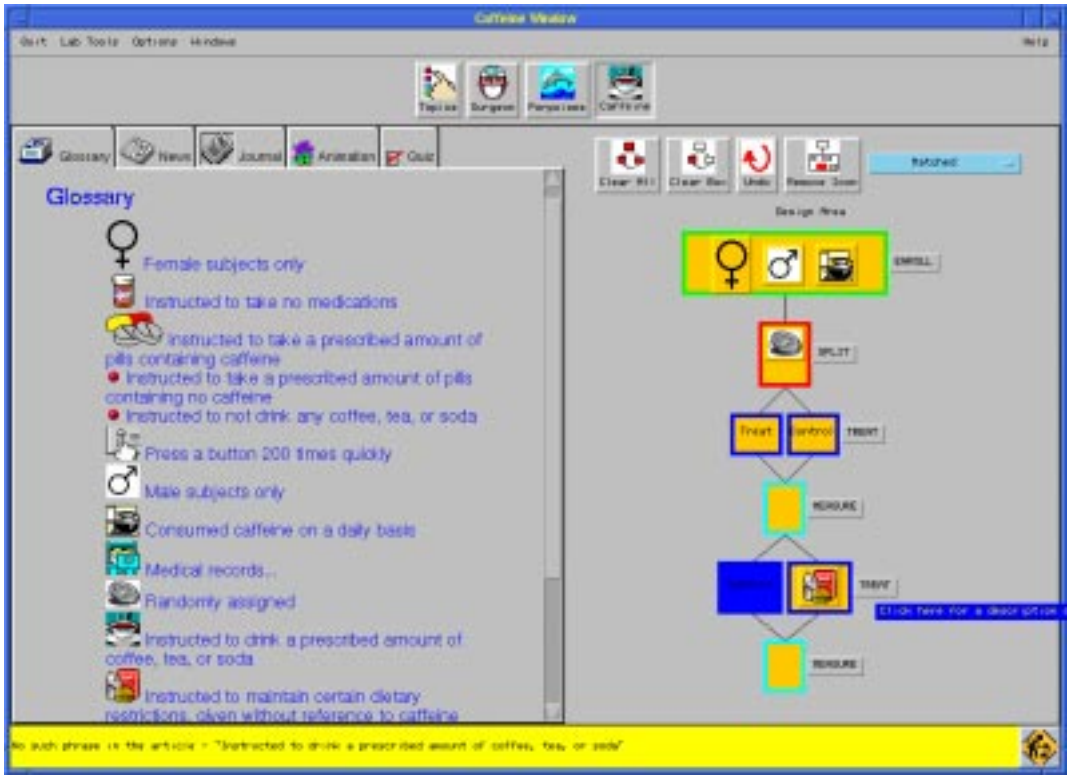


Figure 4 (Challenge Round). The student selects a design and specifies the details from available variables in the glossary on the right.

several designs for her experiment. Once the design is filled in, she runs the animation, which produces responses at random according to a simple model for biases and interactions that is determined by the design. Finally, after the virtual experiment is run, a news story that summarizes and critiques her design is dynamically generated.

Toolkit

We now turn our attention to implementing the example and other labs in general. Producing a lab consists of 4 steps:

1. designing the pedagogical ideas,
2. designing the interface for the student,
3. programming both of these, and
4. creating input files (e.g. experiments, images).

The previous section discussed the first of these two steps for the Design of Experiments lab. Most designers will agree that these two steps are complicated and that using simple prototypes in designing labs would be beneficial. As we started to design the labs, the need for a prototyping language became apparent. Since resources were scarce, we wanted to be able to reuse as much of the prototypes as possible in the final version of the lab. From our earlier experience considering over a dozen different lab designs, we were able to distill components common to several applications. Accordingly, to aid the third development step, we decided to create a library of these common tools and components. To avoid having the lab designer and instructors return to the program developer for lab-specific changes, we decided that an important feature of the components was to allow the instructors to configure and script elements of the lab. A further constraint was to develop the library and labs in a platform-neutral manner since it was unclear then (and still is) what form computing environments might take in the near future. In short, the primary intent of the third step was to

> *provide a flexible and extensible environment in which to develop serious "portable" teaching applications supporting several different levels of programming competence.*

This comes in the form of a toolkit which we call TILE – *a Toolkit for an Interactive Learning Environment*.

Much of the portability issue was solved by using Java as the programming language. Our initial experience with Java was frustrating as we started to use it within months of the first release, and we were coming from a C++ and X windows background. After its initial maturing, we have found it to be an extremely useful tool with which to develop these more involved graphical applications relatively quickly. The choice of Java also allows us to run the applications within browsers even though our focus is on stand-alone applications.

The toolkit is a collection of 16 Java packages[1] or modules and consists of almost 300 reusable classes developed by us and depending on other freely distributable components. The modules provide a relatively comprehensive suite of tools with which one can create a complete lab. The work window has been described earlier and acts as a container into which the lab-specific components are placed. Its services include tooltips for any of the components, navigation between the different work pages, an assistant providing hints for these pages and the basic menu items (such as Print, Quit, etc.). Also, the internal framework takes care of processing command line arguments; providing hooks to connect to different course management tools; saving and restoring a student's session; finding input files; etc. The help window is a hypertext system that displays HTML files supplied by the instructor, etc.

The other modules provide tools that can be used to build up the work areas of the lab. These include classes for different plot types, an SGML[2] rendering component and parser, a dynamic animation rendering facility, a scripting language for generating dynamic text, a quiz component that provides automated grading and feedback, statistical tools for sampling, random number generation, etc. Additionally, there are many lower level tools such as a generic mechanism for managing asynchronous object creation for improved responsiveness in a lab.

Perhaps the most unique thing about the toolkit relates to the final clause in the italicized sentence above. An enormous effort has been made to allow the content of each lab to be specified at run time through simple configuration files. These typically involve pairing names that the application understands and values which specify the content. These are basically *properties* of the application. In the above example, they control application-level properties such as the list of experiments to select from, and experiment-level properties such as the files containing the newspaper article and the layout of the design area for a specific experiment. The application level properties are processed in a hierarchical manner allowing the toolkit properties to be overridden by the instructor at both the course and lab level.

While the configuration allows the instructor to specify different fixed sets of inputs, often greater resolution is required. For these cases, we have developed

a general way of creating simple scripting languages which allow instructors to control the application without programming in Java. We have used this mechanism to provide several such languages to control dynamic feedback and visual animation, for example. The languages use SGML as the syntax since instructors are familiar with HTML. Each language introduces new tag names and attributes for those tags which are used to control the application in some prescribed manner. For example, a quiz uses the `question` tag, and elements of an animation scene might be introduced by the `component` tag, which has attributes such as *image*, *bbox*, etc. These tags and attributes in the input document are made available to the Java application using a single method. This approach has proven to be very effective, as it has allowed us to develop a lab concurrently with one of us doing the lower-level Java development to provide the framework and the other controlling the content. Given the relative ease of creating new input languages based on the SGML/XML syntax and the tree traversal mechanisms in the toolkit, we believe that such scripting languages significantly help in developing, maintaining, and customizing individual labs.

In addition to the labs, we have developed small applications which are interactive interfaces to these languages and modules. These serve as debugging tools for instructors. They allow an instructor to visualize and interact with a succession of quiz pages or an animation scene, for example, and refine the inputs to obtain the desired effect.

We now turn our attention to 3 of the modules in the toolkit. These are of interest because they illustrate how the different programming levels are integrated to create a lab. Also, the dynamic text module is quite unusual.

Plotting Classes

The toolkit provides different plot types such as time series, scatter plots, histograms, bar charts, etc. All plots can be optionally made interactive and support facilities for: identifying data elements by clicking and rubberbanding; stretching axes for zooming; etc. From an instructor's point of view, a useful feature of the plotting classes is the ability to specify attributes in input property files. The plots understand a common set of attributes such as background and plotting colors, background image, axes ranges, number of tick marks, legends, fonts for the different labels, etc. Also, each class understands attributes that have meaning just for it, such as number of bins or bin widths for a histogram.

From a programming point of view, the plots are interesting because they can be used anywhere (e.g. in a button) since they are self describing objects. This approach differs from having a GUI component for one or more plots. The classes support the Java listener event model and can notify other objects when the plot has changed or the user has interacted with it in different ways. Also data can be added incrementally – a feature used in animations to show histograms and scatter plots being constructed dynamically. Finally, the plot classes are designed to be extensible due to the rich class hierarchy making it easy to create new types that inherit all of the features mentioned in here.

Dynamic Text Creation

As mentioned earlier, a major goal in designing the labs is to provide personalized questions and feedback to the student based on what he has done up to that point. While the instructor can change which file to display before the student runs the lab, elements of the document cannot be changed to include information that is only known when it is displayed. An obvious way to do this is to program the dynamic construction of the text in Java since this is where the variables are stored. This however is inconvenient and makes the application relatively rigid as the instructor can't easily change the phrasing and formatting of the text without assistance from the developer.

Our approach is to use an SGML tag through which the instructor identifies a location in the document at which text will be conditionally inserted at different times in the application. This `replace` tag supports different attributes that indicate the conditions under which the text should be replaced and the actual content of the additional text. Each of these attributes can refer to variables in the Java application that are easily made available by the developer as needed by the designer. The conditions and substitution are defined in a simple language that supports most of the usual comparison and logical operations. A few other attributes of the `replace` tag and features of the comparison language make this a very simple but powerful scripting language and allows the instructor to specify control flow of parts of the lab without using Java. The developer need only provide access to the variables by putting them in different tables and arranging to process the document at the appropriate times.

This mechanism is used in several labs. Quiz grading can provide feedback to the student based on his responses. In the challenge round of the Design of Experiments lab, the newspaper article is created by determining the type of design selected and the icons present in the design and then constructing sentences accordingly.

A novel use of the mechanism in another lab allows us to transfer information from a plot to a table in a flexible and extensible manner with a minimal amount of programming.

Animation Module

We use the animation facility in a variety of places throughout the labs. For example, the sampler module illustrates the process of drawing marked tickets from a pool with or without replacement to generate a sample and a statistic. In the example here, the experiment is animated to generate the data for the subjects.

Such interactive dynamic displays are complicated and time consuming to program. Accordingly, we provide a simple abstraction of the animation scene by categorizing the elements into 3 groups: background/global attributes, static components, and elements that move between the static components. This has been sufficient to characterize all of the animations we have considered so far. Each element in the scene is identified by a particular SGML tag defining to which of the 3 categories it belongs. Attributes in the tag define the characteristics of the element such as its position and size, or a sequence of images to be displayed when it is animated, etc. The coordinates of the components are interpreted relative to the instructors scale specified in the `animation` tag. When the scene is rendered, the coordinates are scaled relative to the available space on the screen. The instructor can specify whether a component's bounding box is also scaled or fixed.

Most of the static components are simply images but there is also a mechanism to use an arbitrary Java object. We use this, for example, to allow the student to select a ticket from the sampler pool and to allow histograms to be placed as nodes in the scene (Figure 3). The attributes are passed to these objects which allows us to configure the histogram from the animation input file. This facility allows the animation to be used in a variety of situations as a primitive programming language.

The SGML syntax can also be used to define relationships between components and is used to describe design trees in the Design of Experiments lab.

The elements in the input file allow the scene to be rendered statically. The final part of the animation is the part that controls the dynamic action. This is programmed in Java where each of the moving components runs in its own thread. When each component arrives at an end point of its sub-animation it sends a message to the centralized manager which then decides globally what will happen next. This allows developers to extend the animation control relatively easily since the objects work independently with a single point of synchronization. Different variants of the basic manager class control along which paths the dynamic components move. In our example above, the subjects are dynamic components and the paths they travel relate to the treatment which in turn is specified in input files as a stream of data for the 2 histograms. Finally, the animation controller randomly selects values using the random generator module and associates them with different subjects. When the subject reaches the end of the design, its value is added to the histogram. The animation classes provide the student with controls to start, stop, pause, fast forward and restart the dynamic animation.

Summary and Status

There are a few lessons we have learned from our experiences with this project. The following are perhaps the most important.

1. Educational material to be used in a multimedia environment, and especially graphical interfaces, require a great deal of consideration during the design.
2. Statistical concepts are the important aspects to emphasize rather than the computational techniques associated with data analysis.
3. And finally, careful software design and development with an emphasis on external configuration and extensibility makes for a longer shelf-life and greater reuse. While the cost appears to be longer development time, developing labs individually without such a design takes almost as long, especially if maintenance and customization is considered.

This project is still work in progress, and should be completed by the end of this year. Our efforts so far have concentrated almost exclusively on developing the labs and toolkit. Four labs have been designed and approximately two and a half have been implemented so far. We have made one of the labs available to a handful of students at Berkeley and received valuable suggestions. The responses have been favorable but emphasized the need for a polished and thorough visual interface. The pedagogy has been greeted enthusiastically. Strategies to evaluate the software more formally are also underway.

All of the components of the toolkit are now in place with some of the minor details and enhanced features existing as stubs. The testing and debugging phases should be complete by this Fall. Some of the compo-

nents will be rewritten to take advantage of the new functionality available in Java (e.g. the 2 and 3D packages and the Java Foundation Classes). However, we are confident that the overall design of these modules will remain unaffected except for minor details.

Documentation for the configuration files and scripting languages is available. More descriptive papers on the pedagogical philosophy and a more complete description of the programming interface for the toolkit classes are forthcoming. Also we have considered developing an additional lab as a way of detailing the 4 steps in the lab development cycle.

Information on the project will continue to be made available at the URL `www.stat.berkeley.edu/users/nolan/TILE`.

Notes

[1] A Java package is a collection of classes which are grouped together to provide some higher level functionality.

[2] SGML stands for a Structured General Markup Language and is the generic framework in which HTML is a particular instance. The next generation of Web browsers will use XML which is a subset of SGML. Information on SGML & XML can be found at the Web sites `www.sil.org/sgml` and `www.w3.org`

References

Freedman, D., Pisani R., and Purves, R. (1997), *Statistics*, Norton.

Gelman, A., Nolan, D., Men, A., Warmerdam, S. and Bautista, M. (1998), "Student projects on statistical literacy and the media," *The American Statistician*.

Deborah Nolan
U.C. Berkeley
`nolan@stat.Berkeley.edu`

Duncan Temple Lang
Bell Laboratories
`duncan@research.bell-labs.com`

GETTING TO SLEEP AT NIGHT

Intrusion Detection Based on Structural Zeroes

By Martin Theus and Matthias Schonlau

1. Introduction

Since computer intrusion detection was first investigated (Anderson, 1980) much has happened. Several intrusion detection products are available commercially (e.g. Netranger, 1998; Emerald, 1998) and a number of academic institutions have research teams in this area (e.g. COAST project at Purdue University, 1998; Intrusion Detection for large networks, U.C. Davis Research Group, 1998; Computer Immune Systems, University of New Mexico, 1998).

There are two main approaches to intrusion detection: misuse detection and anomaly detection. Intruders often gain access to computer systems by exploiting software bugs. In misuse detection one tries to detect such attempts by recognizing attack patterns corresponding to known software bugs. The difficulty here is that not all software bugs are known, and that it is not useful once the intruder has already gained access to the system. The second approach, anomaly detection, attempts to detect deviations from past computer usage and flags them for investigation. Neither approach appears to be uniformly superior over the other. The approach presented here falls in this second category.

Our data consist of user names and commands (e.g. `ls, man, java`) from a UNIX operating system. Many cells of the user/command contingency table are empty. Since most users are only aware of some subset of the set of distinct commands or choose never to use commands they are aware of, most empty cells are structural zeroes (as opposed to sampling zeroes). Statistical techniques (e.g. principal components, regression analysis, etc.) generally assume that all combinations of variables/observational units are feasible and can thus take non-zero values. While such techniques still function with a large number of zero cells, they do not incorporate structural information.

Too many empty cells may also lead to estimation problems. For example, it is difficult to estimate transition probabilities from one command to a second command when the first command is never observed. Therefore it is tempting to exclude commands that are used by very few users from further consideration. DuMouchel and Schonlau (1998) for example, use only the 100 most fre-

quently used commands (about 26% of all commands used by a group of 45 users) and combine all remaining commands into one category.

The outline of this article is as follows. In Section 2 we further motivate the importance of structural zeros. Section 3 introduces a heuristic test statistic based on this idea. Section 4 describes an experiment to test the statistic and gives results. Section 5 concludes with a discussion.

To facilitate explanations in later sections we discuss at this point our data and make some definitions.

Data

Command level data consisting of user names and commands (without arguments) were generated from output of the UNIX `acct` auditing mechanism. Some commands recorded by the system are implicitly generated and not explicitly typed by the user. For example, each execution of the `.profile` file or a `make` file generates commands contained in these files that are also recorded in the data stream.

During two separate time periods (training and test data) the first 1000 commands for each of 45 users were recorded. For some of the analysis reported below, we split each set of 1000 commands into 10 replications of 100 commands.

Other sources of data (e.g. packet data) are conceivable. In this paper we are not concerned about what source of data would be most appropriate; rather we take the stance that we have to deal with the data available.

Definitions

A *unique command* is a command that is used by only one user in a training data set. A *rare command* is a command that is used by only a few users in a training data set. We further make the distinction between the *training data* which establishes, for example, how rare commands are and *testing data* which is used to test statistics. We denote the total number of users in any data set by U.

2. Motivation

In this section we explore structural zeroes in our training data, that is, we look at how rare or unique commands are.

We first look at the question of what fraction of commands are used by only one user, used by exactly two users, etc. for a certain community of users. It is then interesting to plot the the fraction of commands used by exactly i users versus i. To overlay plots for a varying sizes of user communities, we standardize and use the fraction of users i/U instead. This plot is shown in Figure 1 for user communities of size 4, 10 and 45. Almost half of all commands are unique, in the sense that they are only used by one out of the 4, 10 or 45 users, respectively.

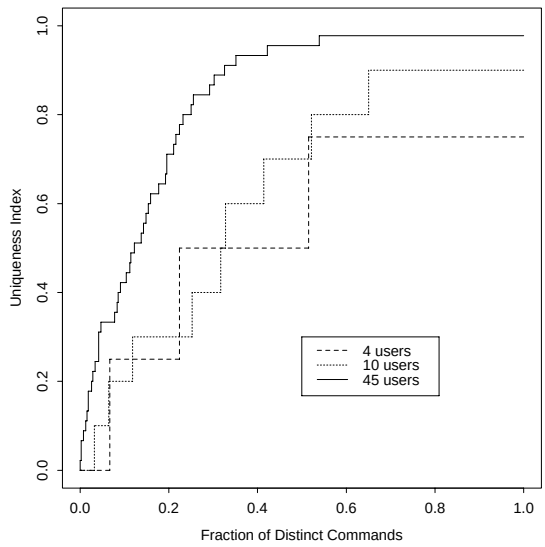


Figure 1. Stepplot of uniqueness index ($1-$ fraction of number of users) versus fraction of distinct commands for 4, 10 and 45 users.

While Figure 1 shows that there are many unique and rare commands, it is unclear whether the unique/rare commands account for a large proportion of the total commands. Instead of the fraction of distinct commands, Figure 2 uses the fraction of the total frequencies on the horizontal axis. Two conclusions can be drawn from Figure 2: First, the percentage of the data consisting of commands used only by a certain percentage of users is approximately the same regardless of whether the user community consists of 4, 10 or 45 users. Second, since the lines are almost on the 45 degree line, the percentage of the data consisting of commands used only by a certain percentage x of users is approximately that percentage x. For example, about 20% of the command data contain only commands that are used by at most 20% of the users (those with a high uniqueness index in Figure 2).

The commands can be grouped such that each group contains all commands that are used by exactly i users ($1 \leq i \leq U$). We assign an ID to each command such that commands from groups with rare commands are assigned lower ID's than commands from groups with less rare commands. The order within a group is arbitrary. When plotting the command ID versus their order in the audit stream the usage pattern of unique/rare commands

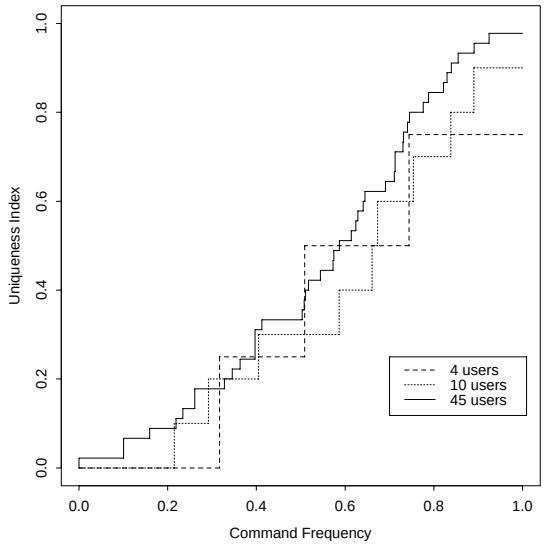


Figure 2. Stepplot of Uniqueness Index (1− Fraction of Number of Users) versus total Command Frequency for 4, 10, and 45 users.

emerges. Such a plot is shown in Figure 3 for each of 10 users. Plots of different users are clearly distinguishable from each other. The fourth, seventh and the tenth plot are more regular than the other ones. They belong to UNIX processes (rather than to human users).

The same plot can be seen in Figure 4 for 45 users. (Figure 3 corresponds to the first 10 columns of Figure 4). While the time series aspect is now difficult to make out because each column is very narrow, the column patterns are clearly distinct. A classification of users based on which commands are used and how often they are used appears promising.

3. Intrusion Detection by Uniqueness

We are about to define a test statistic that depends on two quantities, W_{ij} and U_j. These quantities are estimated from the training data set. The test data is then used to evaluate the test statistic.

Let N_i denote the length of the test data sequence for user i and N_{ic} the number of times command c appears in user i's test data. Then $N_i = \sum_{c \in Test(i)} N_{ic}$.

We now define

$$\begin{matrix}\text{Uniqueness} \\ \text{Score User i}\end{matrix} = \frac{1}{N_i} \sum_{c \in Test(i)} W_{ic}\, U_c\, N_{ic} \qquad (1)$$

where c indexes the set of (distinct) commands $Test(i)$, the weights W_{ic} are

$$W_{ic} = \begin{cases} 1 & \text{if user } i\text{'s training data contains} \\ & \text{command c} \\ -1 & \text{otherwise} \end{cases}$$

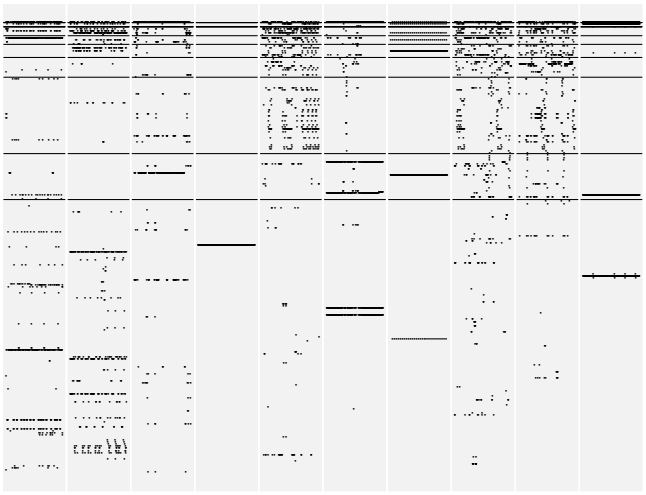


Figure 3. Plots of command ID vs command order in the audit stream for each user. Commands are grouped by the number of users that have used them. Commands used by only one user (unique commands) have lowest ID's, commands used by all users have the highest. Within each plot, horizontal lines are drawn to separate groups.

and where the uniqueness index U_c is defined as

$$U_c = \begin{cases} 0 & \text{if } c \text{ used by all users} \\ 1/U & \text{if } c \text{ used by all but one users} \\ \cdots & \\ (U-2)/U & \text{if } c \text{ used by only two users} \\ (U-1)/U & \text{if } c \text{ used by only one users} \\ 1 & \text{if never used by any users} \end{cases} .$$

The quantity U_c is 1 minus the fraction of users that have used command c in the training data. It is the same quantity that has been plotted on the vertical axis in Figures 1 and 2.

For each new command the score in (1) either increases or decreases, depending on whether the associated user has used that command before in the training data or not. The amount of increase/decrease U_c is higher for rare commands than for common commands. Hence a user will tend to score high if he/she uses similar commands to the ones he/she used in the training data. The order in which the commands appear does not matter.

Weighted Uniqueness Scores

Discrimination among users based on (1) works well, but we improve on it for the following reason: Suppose user A uses a rare command only a few times, and user B uses that rare command often. In this case test data from user B with many incidences of that same rare

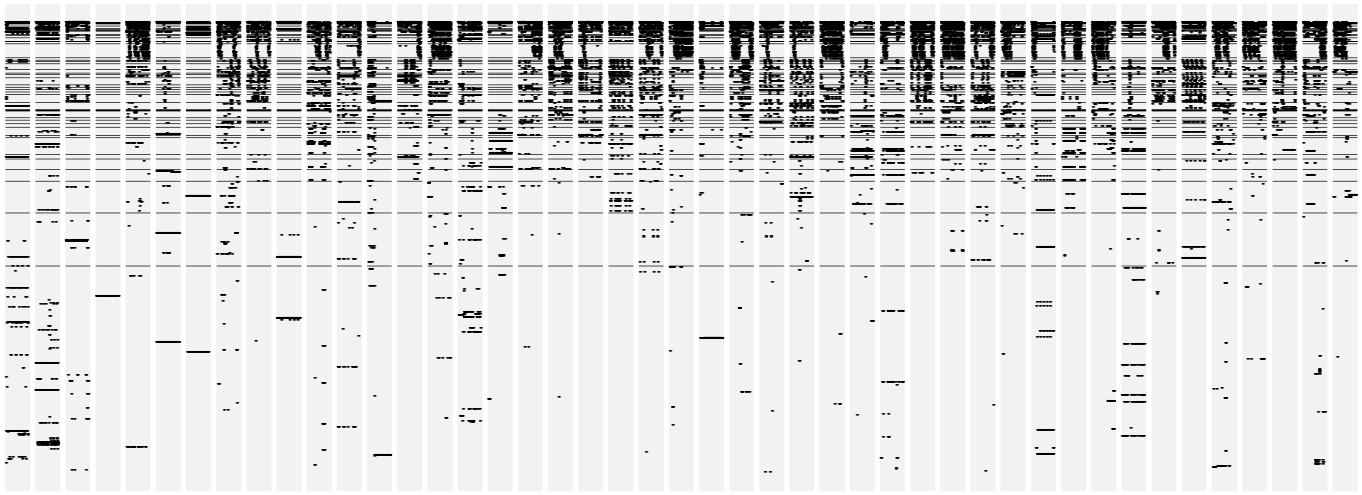


Figure 4. Plots of command ID vs command order in the audit stream for each user. Commands are grouped by the number of users that have used them. Commands used by only one user (unique commands) have the lowest ID's, commands used by all users have the highest. Within each plot, horizontal lines are drawn to separate groups.

command will be assigned a high score when trained on data of user A. The use of that rare command is indicative of user A and should therefore lead to an increase in the score. However, since user A does not use that command very much relative to other users, the use of that rare command is only moderately indicative and the score increase should not be very large.

We modify (1) by changing the definition of the weights:

$$\begin{matrix}\text{Uniqueness} \\ \text{Score User i}\end{matrix} = \frac{1}{N_i} \sum_{c \in Test(i)} W^*_{ic}\, U_c\, N_{ic} \qquad (2)$$

where U_c is defined as before,

$$W^*_{ic} = \begin{cases} p_{ic}/p_{.c} & \text{if user i's training data contains} \\ & \text{command c} \\ -1 & \text{otherwise} \end{cases}$$

and where

$$p_{ic} = N_{ic}/N_{i.}\,.$$

Commands previously associated with a weight of 1 are now given a smaller weight – how small depends on command usage relative to other users. The weight for a particular command c for user i is high when user i uses command c a lot *relative to other users*, i.e. when the use of command c is especially characteristic of user i. As a consequence, a user using a particular rare command much more often than the legitimate user will score more often for that command when tested as the legitimate user, but the impact on the overall score is lower due to a low weight.

4. Results

To evaluate the method presented in the previous section, we compute the test data score (2) of user i ($i = 1, \ldots, 45$) based on the training data for user j ($j = 1, \ldots, 45$) for all pairs of users.

The resulting 45×45 scores are graphically displayed in Figure 5 (only scores greater than zero are shown). Darker shading corresponds to higher scores, lighter shading to lower scores. For the diagonal entries both the test data and the training data correspond to the same user. Ideally, we would be able to perfectly discriminate between diagonal entries and off-diagonal entries, that is all diagonal entries should be darker than all off-diagonal elements.

By varying the threshold at which an alarm is raised one can obtain different rates of false alarms (false positives) and missing alarms (false negatives). Figure 6 shows this tradeoff between false positives and false negatives for scores based on 100 commands and 1000 commands. Figure 6 shows two curves in both cases to give an idea of the variability of the curves for different data sets; the second curve in both cases is obtained by interchanging the training and the test data. (The fact that the curves based on 1000 commands do not go all the way to the vertical axis on the left is due to a discreteness effect. The midpoint between zero and one out of 45 possible false alarms corresponds to $1/90 = 1.1\%$ on the horizontal axis.)

The lower left corner on this plot represents the ideal situation: no false alarms, and no missing alarms. For our method no false alarms correspond to about 10%

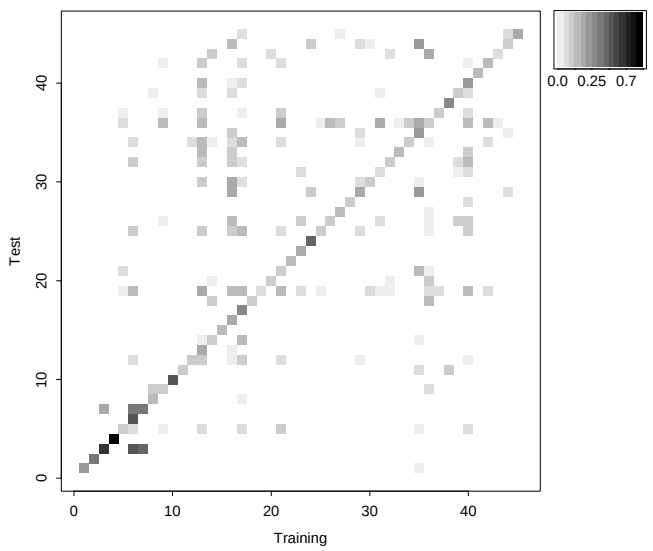


Figure 5 Visual display of the 45×45 scores from (2) based on 1000 commands. Only scores greater than zero are shown. The darker the shading, the higher the score.

missing alarms based on 100 commands or 5% based on 1000 commands. As expected the discrimination among users based on 1000 commands is easier than based on 100 commands. As the rate of false alarms increases, the rate of missing alarms initially drops only slowly. Incidentally, the threshold of score $= 0$ corresponds to the situation when no false alarms are generated. Decreasing the threshold below zero leads to an increase of missing alarms without changing the false alarm rate. This implies that based on this data the legitimate user always has a score above zero, whereas most but not all of the other users have a score smaller than zero.

5. Discussion

Intrusion detection based on command uniqueness is conceptually very simple. It requires very little storage (W_{ij} for all i, j and U_j for all j) and, since (2) is a weighted average, it can be easily updated. Moreover, each update is computationally fast requiring only a few multiplications. We would also like to point out that while (2) constitutes a quantitative improvement over (1), the uniqueness index is qualitatively more important than the weights.

When too many of the alarms generated are false, they tend to be ignored altogether after a while. Therefore, a low false alarm rate is particularly important in intrusion detection. Choosing a threshold of 0, based on 100 commands we are able to avoid false alarms altogether in our experiment. We have not seen another intrusion

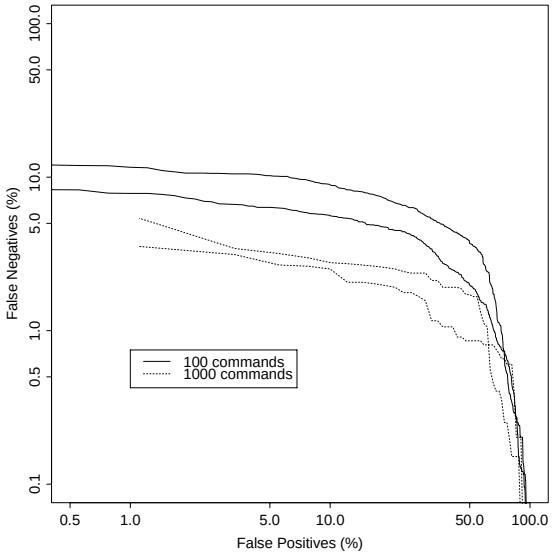


Figure 6 Tradeoff between false negatives and false positives (ROC curve) based on (2) for 100 and 1000 commands. Two curves are obtained by interchanging the training and the test data. Both axes are on a logarithmic scale.

detection method that has virtually no false alarms at an acceptable rate of missing alarms.

The uniqueness approach may work especially well in the diversity of a research environment. The UNIX operating system allows and even encourages this diversity. The uniqueness approach may not work as well in an environment where everybody's job description is very similar.

The choice of using 100 commands for the scores was arbitrary. We are investigating whether we can discriminate users based on even fewer commands.

It is difficult to compare various intrusion detection methods due to the lack of a common yardstick. Different methods are based on different data sources and are often too complicated to reimplement for comparison purposes. Based on Figure 6, the uniqueness approach compares favorably with DuMouchel and Schonlau (1998) who report false negative rates between 10% and 50% at 5% false positives for comparable data sources.

Acknowledgements

M. Schonlau's work is funded in part by NSF grants DMS-9700867 and DMS-9208758. We are grateful for feedback from our network intrusion group with members from AT&T Labs Research, The National Institute of Statistical Sciences and Rutgers University. Their comments have led to a number of improvements.

References

Anderson, James P. (1980), "Computer security threat monitoring and surveillance," Technical report, James P. Anderson Co., Fort Washington, PA, April 1980.

Computer Immune Systems,
www.cs.unm.edu/~forrest/
(accessed June 23, 1998)

COAST project,
www.cs.purdue.edu/coast/
(accessed June 23, 1998)

DuMouchel, W. and Schonlau, M. (1998), "A comparison of test statistics for computer intrusion detection based on principal components regression of transition probabilities," In *Proceedings of the 30th Symposium on the Interface: Computing Science and Statistics*, (to appear).

Emerald,
phlox.csl.sri.com/emerald/
(accessed June 23, 1998)

Intrusion Detection for Large Networks,
seclab.cs.ucdavis.edu/arpa/
(accessed June 23, 1998)

Netranger,
www.wheelgroup.com/netrangr/1netrang.html
(accessed June 23, 1998)

Martin Theus
AT&T Labs-Research
theus@research.att.com

Matthias Schonlau
AT&T Labs-Research and
National Institute of
Statistical Sciences
schonlau@research.att.com

NEW SOFTWARE TOOLS

Mosaic Displays in S-PLUS: A General Implementation and a Case Study.

By John W. Emerson

Introduction

Hartigan and Kleiner (1981) introduced the mosaic as a graphical method for displaying counts in a contingency table. Later, they defined a mosaic as "a graphical display of cross-classified data in which each count is represented by a rectangle of area proportional to the count" (Hartigan and Kleiner 1984). Mosaics have been implemented in SAS (see Friendly 1992) as a graphical tool for fitting log-linear models. Interactive mosaic plots (see Theus 1997a, b) have been implemented in Java. A third implementation is available in MANET, a data-visualization software package specifically for the Macintosh. No general implementation has been available in S-PLUS, one of the most popular statistical packages.

The implementation presented in this article, while lacking the modelling features of Friendly's SAS implementation, provides a simply specified function for mosaics displaying the joint distribution of any number of categorical variables. As an illustration, this article examines patterns in television viewer data. A four-way table of 825 $(5 \times 11 \times 5 \times 3)$ cells represents Nielsen television ratings (number of viewers) broken down by day, time, network, and switching behavior (changing channels, turning the television off, or staying with the current channel) for the week starting November 6, 1995. Simple patterns in the data appearing in the mosaic support intuitive explanations of viewer behavior.

The Data

Nielsen Media Research maintains a sample of over 5,000 households nationwide, installing a Nielsen People Meter (NPM) for each television set in the household. The sample is designed to reflect the demographic composition of viewers nationwide, and uses 1990 Census data to achieve the desired result. Nielsen summarizes the stream of minute-by-minute measurements to provide quarter-hour viewing measurements (defined as the channel being watched at the midpoint of each

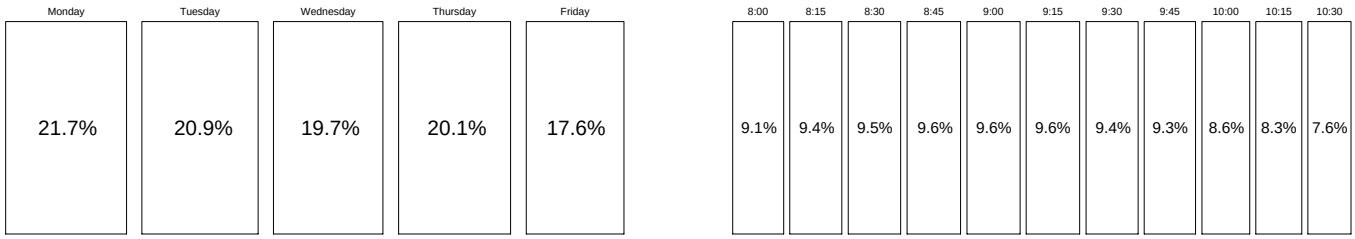


Figure 1. (a) Mosaic of the week's aggregate audience by day (lefthand panel); and **(b)** mosaic of the week's aggregate audience by time (righthand panel).

quarter-hour block) for each viewer in the sample. (Details are presented in Nielsen's *National Reference Supplement* 1995.)

A "TV guide" of the prime-time programming of the four major networks (ABC, CBS, NBC, and FOX) for the weekdays starting Monday, November 6, 1995 appears in Figure 2. During any quarter-hour, the individual is observed watching a major network channel, a non-network channel, or not watching television. At 10:00 however, FOX ends its network programming, so Nielsen does not record individuals watching FOX after 10:00. I confine this study to a subset consisting of 6307 East coast viewers in 2328 households.

Creating Mosaics in S-PLUS

Friendly (1994) describes the complete algorithm used to construct a mosaic for a general four-way table, alternatively dividing horizontal and vertical strips of area into tiles of area proportional to the counts in the remaining sub-contingency table. Without repeating the description of a general mosaic display, I note the important features of my S-PLUS implementation, which help explore various aspects of any cross-classified data set:

- Any number of categorical variables may be included in the mosaic, though in practice even a five-way table may be sufficiently complicated to defy explanation.
- Empty cells of the contingency table are represented (where possible) by a dashed line segment.
- The order in which the variables are represented may be specified, allowing simple exploration of any marginal or conditional frequencies on any subset of variables without physically manipulating the raw contingency table itself.
- The direction (horizontal or vertical) used in dividing the mosaic by each variable may be specified, allowing more flexibility than the traditional alternating divisions.
- Shading of the tiles resulting from the inclusion of the final variable in the mosaic may be specified, if desired. The amount of space separating the tiles at each level of the mosaic may also be customized.

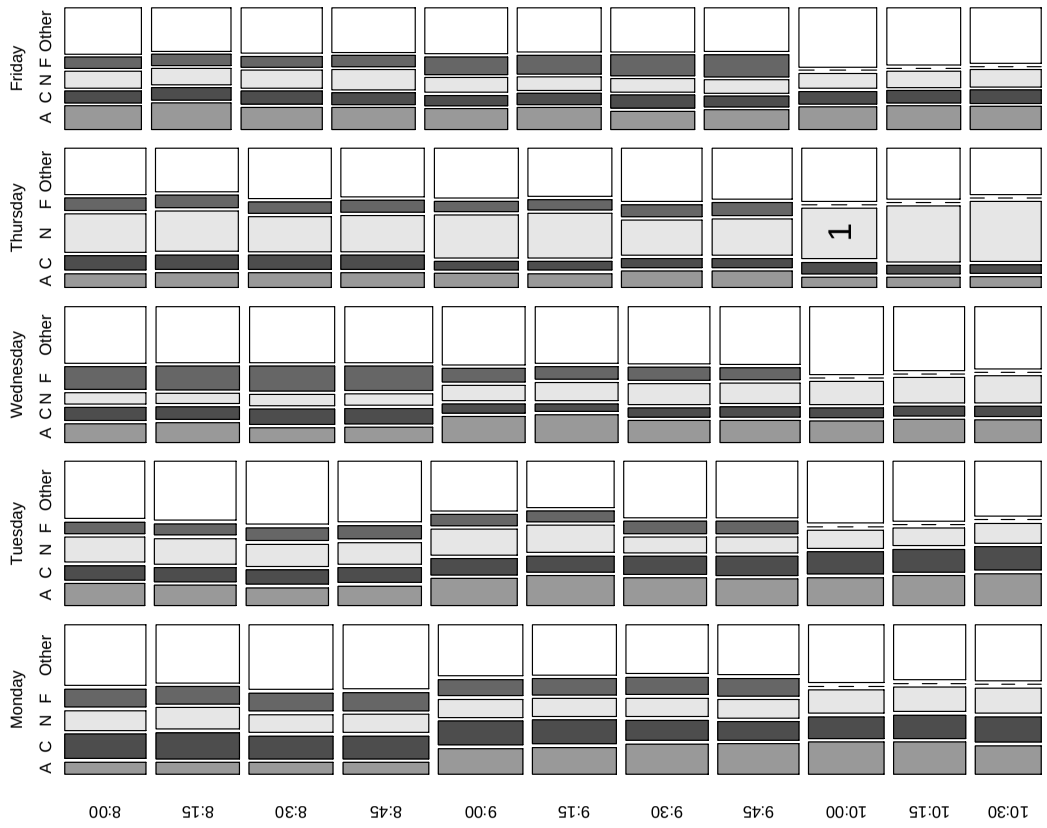
The documentation and S-PLUS code are available – details are provided at the end of the article. The basic algorithm, an efficient recursive procedure, proceeds as follows:

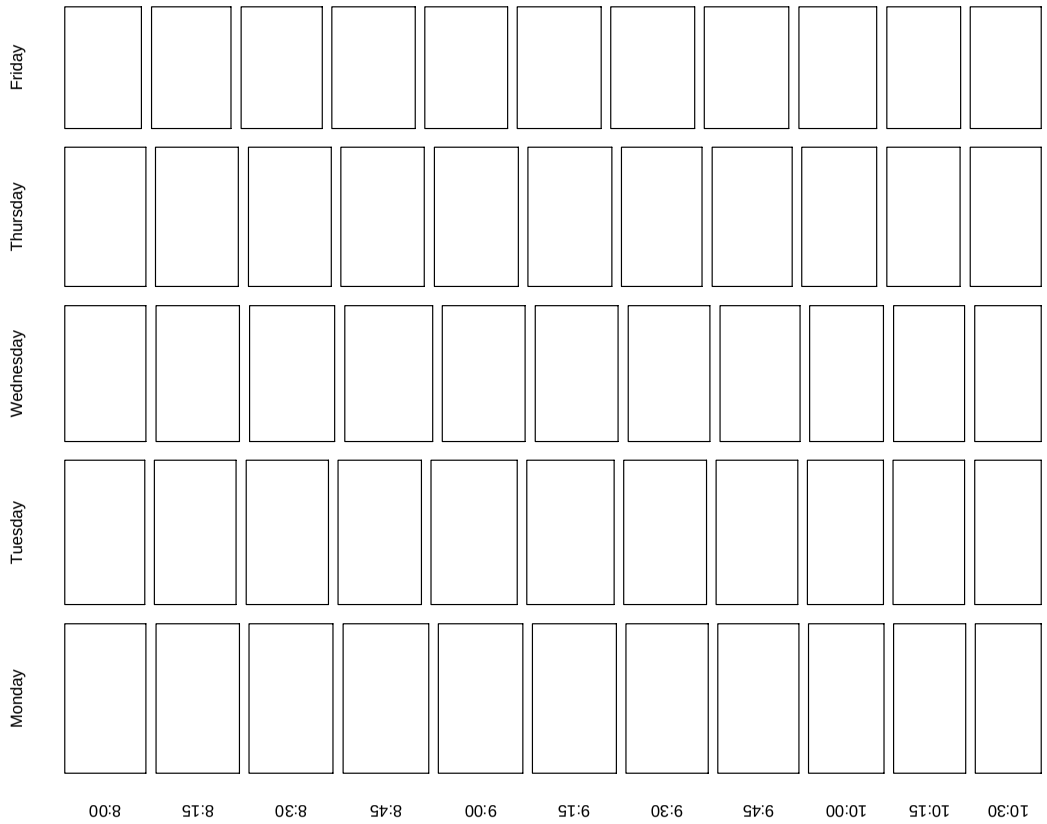
1. Initialize the parameters and the graphics device – the lower left and upper right corners of the plot area are (x_1, y_1) and (x_2, y_2). The term "parameters" refers to a collection of counts from the contingency table, labels, and values associated with features discussed above.
2. Call the recursive function
   mosaic.cell$((x_1, y_1), (x_2, y_2),$ all parameters$)$.
3. Recursive function
   mosaic.cell$((a_1, b_1), (a_2, b_2),$
   selected parameters for the current tile$)$:
   (a) Divide the current tile, given by (a_1, b_1) and (a_2, b_2), into sub-tiles, taking into account the spacing and split direction arguments of the parameters.
   (b) Add labels if the current variable is one of the first two divisions of the axis.
   (c) If this division corresponds to the last variable of the contingency table, draw the sub-tiles. Otherwise, call *mosaic.cell*() once for each of the current sub-tiles, with the appropriate sub-tile coordinates and subsets of the current parameters.

Results: Television Viewer Behavior

Simple mosaics dividing the week's aggregate audience by day and time are presented in Figures 1a and b, respectively. Though they serve the same purpose as histograms, their tile areas are more difficult to compare than the tile heights in histograms. The advantage of mosaics does not appear until at least two categorical

Figure 2. TV Guide, 11/6/95 – 11/10/95.


Friday
Thursday
Wednesday
Tuesday
Monday
A C N F Other
1
8:00
8:15
8:30
8:45
9:00
9:15
9:30
9:45
10:00
10:15
10:30


Friday
Thursday
Wednesday
Tuesday
Monday
8:00
8:15
8:30
8:45
9:00
9:15
9:30
9:45
10:00
10:15
10:30

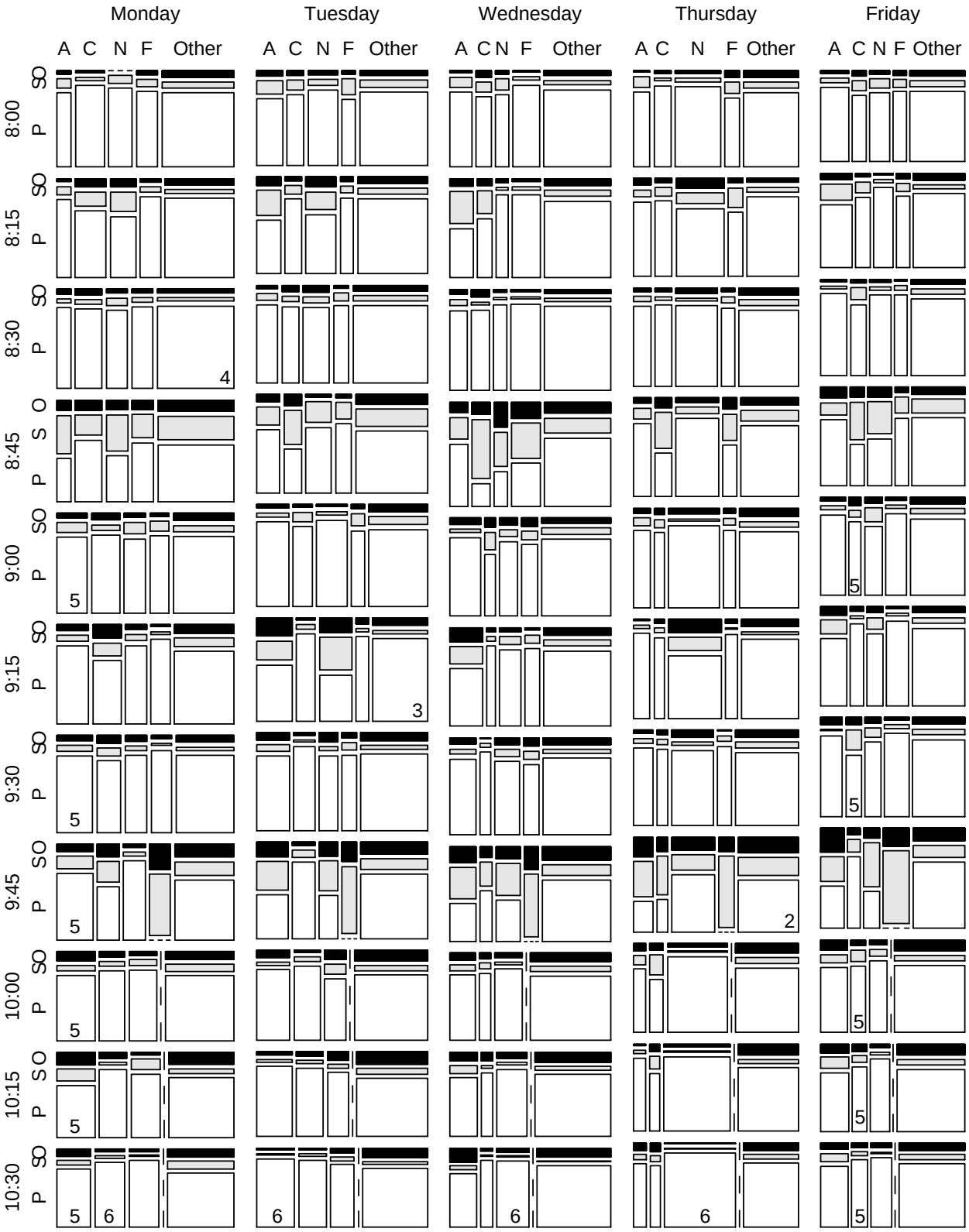


Figure 5. Mosaic of network shares and audience transitions. P = persistent, S = switch, O = off. Numbered tiles are discussed in Section 4.

Thursday	Transition			
9:45 Network	Off	Persist	Switch	9:45 Network Total
ABC	54	86	99	239
CBS	21	47	56	124
NBC	80	349	94	523
FOX	31	0	156	187
CABLE	135	443	152	730
Transition Total	321	925	557	1803

Table 1. Thursday 9:45 Contingency Table.

variables are included. These one-way mosaics show that the aggregate audience is smaller later in the week (Figure 1a) and later in the evening (Figure 1b). The mosaic corresponding to the two-way table of the aggregate audience, divided first by day and then by time, appears in Figure 3 just for clarity of exposition – in this example, interesting analysis begins with the addition of specific network counts by day and time.

As we add the network variable (to simplify exposition, the term "network" will include the aggregate cable, or non-network, alternative) and the transition categories to the mosaic (Figures 4 and 5, respectively), several points illustrate the use of these mosaics in studying television viewer behavior. The following numbers are marked in the relevant tiles in the mosaics.

1. When the network variable is added to the two-way mosaic in Figure 3 to form a three-way contingency table, the resulting mosaic tiles at each day and time represent the network ratings, or share of the viewing audience (Figure 4). For example, on Thursday at 10:00, 685 of 1692 viewers watching television were tuned into NBC's hit *E.R.*, so the NBC rectangle occupies 40.4% of the area in Thursday's 10:00 tile.
2. Figure 5 includes an additional variable with three categories: among the viewers watching a certain network (at time t on day d), some turn the TV off and do not watch anything at time $t+1$ (represented by the black tiles); others switch networks at time $t+1$ (shaded tiles), while the remaining viewers watch the same network, or *persist* (unshaded tiles). For example, consider the NBC viewers in the Thursday 9:45 tile who watch the end of *Caroline in the City*: 523 of 1803 viewers watching television then tuned into the end of *Caroline in the City* – the NBC tile is 30% of the area of the Thursday 9:45 tile. Of the 523 viewers, only 80 turned the television off at 10:00 (black tile), 94 switched to a different network at 10:00 (shaded tile), and the remaining 349 watched the beginning of *E.R.* on NBC (persisting in their viewing of NBC, the unshaded area). Table 1 presents the two-way contingency table for the viewers watching television at 9:45 classified by network choice and viewing behavior after the quarter-hour. Note that there can be no viewers persisting in watching FOX from the 9:45 quarter-hour – these FOX viewers must either turn the TV off or switch channels. This empty cell corresponds to the empty transition tile in the FOX 9:45 tile. Similarly, all FOX tiles after 10:00 are empty.
3. A quick study of the TV schedule in Figure 2 and the mosaic in Figure 5 shows that viewer persistence is higher when there is show continuity. For example, on Tuesday night after the 9:15 quarter-hour, CBS and FOX have continuations of longer shows (both are movies) while ABC and NBC start new shows at 9:30 (competing half-hour comedies). This tile shows a striking example of high persistence with show continuity and lower persistence going into new programming: ABC and NBC have lower persistence rates of roughly 60% and 50%, while CBS and FOX enjoy high persistence rates of close to 90% each. Note the uniformly high degree of switching at 8:45 and 9:45 in Figure 5.
4. It is also evident from the mosaic that persistence during the odd quarter-hour transitions (that is, always during a show) is fairly uniform between the networks, and usually high compared to other transitions. The 8:30 frame on Monday, for example, shows uniformly high flow of viewers persisting into 8:45.
5. These mosaics provide insight into different sources of viewer persistence. The primary trend appears to be higher persistence during shows (and lower persistence at end of shows), but more specific elements of persistence are also evident in the mosaics. First, consider *Monday Night Football* on ABC after 9:00. There is unusually

low persistence given the show continuity, particularly after 10:00, because sports and news programs fail to maintain the audience as effectively as other programs. CBS's *Ice Wars* figure skating event on Friday also has a slightly lower persistence rate given the show continuity.

6. Finally, consider dramas such as *Chicago Hope* (Monday at 10:00 on CBS), *NYPD Blue* (Tuesday at 10:00 on ABC), *Law & Order* (Wednesday at 10:00 on NBC), and *E.R.* (Thursday at 10:00 on NBC). All have particularly high persistence into the final quarter-hour – viewers watching the later parts of these popular dramas tend to finish watching rather than turning away before the climax.

It should be noted that although these mosaics focus attention on network persistence, viewer persistence in the other alternatives must also be addressed. Persistence in the aggregate non-network category is understandably high, since only switches from a non-network alternative into a major network and back again are observed (no switching between non-network alternatives can be studied). A detailed study of overall rates of switching would require a richer data set. Viewers not watching television also persist in not watching television, though these counts are not included in this study.

Mosaics are a promising method for displaying multivariate categorical data, and it is hoped that this S-PLUS implementation will be useful to the statistical community.

Acknowledgements

The authors wishes acknowledge the guidance of John Hartigan in refining these mosaics, and Ron Shachar (Yale School of Management), and Greg Kasparian and David Poltrack of CBS for their help in obtaining the data for this study.

Additional Resources

Additional resources associated with this article – both the software and the data – are available at the Web site `www.stat.yale.edu/~emerson/JCGS/`.

References

Friendly, M. (1992), "User's guide for MOSAICS: A SAS/IML program for mosaic displays," Technical Report 206, Department of Psychology, York University.

See also `www.math.yorku.ca/SCS/friendly.html`

Friendly, M. (1994), "Mosaic displays for multi-way contingency tables," *Journal of the American Statistical Association*, **89**, 190–200.

Hartigan, J. A., and Kleiner, B. (1981), "Mosaics for contingency tables," *Proceedings of the 13th Symposium on the Interface between Computer Science and Statistics*.

Hartigan, J.A., and Kleiner, B. (1984), "A mosaic of television ratings," *The American Statistician*, **38**, 32–35.

Nielsen Media Research (1995), *National Reference Supplement*, A.C. Nielsen.

Theus, M. (1997a), "Visualization of categorical data," in *Advances in Statistical Software 6*, 47–55, Lucius & Lucius.

Theus, M. and Lauer, St. R. W. (1997b), "Visualizing log-linear models," submitted to *Journal of Computational and Graphical Statistics*.

The Web site `www.research.att.com/~theus/Mondrian/Mondrian.html` has more information on these tools.

John W. Emerson
Yale University
`emerson@stat.yale.edu`

Linked Micromap Plots: Named and Described

By Daniel B. Carr, Anthony R. Olsen,
Jean-Yves P. Courbois, Suzanne M. Pierson,
and D. Andrew Carr

1. Introduction

In this article we describe a new template for the display of spatially indexed statistical summaries and give it a name: linked micromap plots (LM plots). Readers can obtain a general notion about the template from a quick glance at the three examples in this article. The LM template emerged from our environmental graphics research and has quickly jumped to other applications such as the display of federal statistics summaries. We foresee a wide range of applications that extend even to the study of gene communications networks. Our definitive paper on the template emphasizes diverse environmental examples and is nearing completion. Our enthusiasm is such that we want to promote the template immediately. Thus we are adapting some of our material for this article.

LM plots resulted directly from Tony Olsen's challenge to link row-labeled plots (Carr 1994a, Carr 1994b, and Carr and Olsen 1996) to large maps. Micromaps were our linking solution, but we immediately observed that micromap sequences were informative without the large maps. Our first big presentation combining micromaps and large maps (Olsen et al, 1996) was a 4 x 8 foot poster. While we have shown this poster summarizing millions of observations at many conferences, we have been slow to publish. What journal would publish a large poster? The newsletter article by Carr and Pierson (1996) and a paper by Carr (1997) provide the first published example LM plots. Neither article provides a name for the template nor calls attention to the scope of design variations and applications. This sequel does both.

The structure of the article is as follows. Section 2 describes the LM template in the context of three examples. Section 3 revisits the examples to make comments about plot interpretation and continuing design issues. Section 4 comments on the newness of the template, credits many that have influenced our thinking and closes with indications of software available and challenges for the future.

2. The Template for Linked Micromap Plots

The template for LM plots has four key features. First, LM plots include at least three parallel sequences of panels that are linked by position. As illustrated in Figure 1, one sequence consists of micromaps, one sequence consists of linking legends, and one sequence consists of familiar statistical graphics, in this case times series panels. As in Figure 2, the template can include more sequences of panels as long as it has these three types of panels. Sorting the units of study (nations in Figure 1 and states in Figures 2 and 3) is the second feature. The third feature partitions the study units into panels to focus attention on a few units at a time. The fourth feature links representations for highlighted study units across corresponding panels of the sequences. We discuss these features below. With the three types of parallel sequences in mind, we discuss sorting and grouping of study units first, and end this section with discussions of across panel linking and LM plot labeling.

2.1 Study Units, Sorting, and Perceptual Grouping

The study units in LM plots have names, map locations, and summary statistics. In Figures 1, 2 and 3 the study units are nations, states, and states, respectively. One goal of LM plots is to group information into manageable units for human interpretation. Thus the number of study units in LM plots is typically modest, often 50 or fewer. The desire to represent many more study units suggests a hierarchical organization of multiple LM plots. For example one national LM plot might show summaries for all the U.S. states while multiple state LM plots show summaries for the counties within each state. Since overview summaries can show extrema as well as central tendency, say via boxplots with outliers, overviews do not necessarily hide interesting clues about where to look next.

LM plots sort the study units. Often, plotted summary estimates provide the basis for sorting. The unemployment ratio provides the basis for sorting states in Figure 3. The time series in Figures 1 and 2 are multivariate summaries. In both examples we sort by the time series average value. Many other multivariate sorting options are available. Cleveland (1993) suggests using the median, while Carr and Olsen (1996) discuss additional methods such as spanning tree traversals and sorting by the first principal component. Sorting can also be based on variability, spatial position and many other variables. Different sorting methods can reveal and accentuate different relationships. An interactive setting facilitates rapid re-expression using different sorting criteria. However, the easy ability to re-express LM

plots does not obviate the need to think about the sorting order and the choice of panel representations. For example, carefully sorting two parallel sequences of bar plots may fail to show structure as clearly as a simple scatterplot.

Kosslyn (1994) recommends creating small perceptual groups so that we can focus attention on a manageable number of elements. He cites literature indicating four items as a good number for direct comparison. We often chose to emphasize five items per panel since counting in fives is convenient. In the discussion below we refer to the emphasized items in a panel as highlighted items. In some cases, such as the middle panels of Figure 3, we include six items. This is still consistent with the cartographic literature that recommends using six or fewer classes for classed choropleth maps. Six items makes color discrimination and other tasks substantially more difficult than five. Still six is within reason when space constraints are present as in Figure 3.

Creating small perceptual groups is closely related to the chunking of information that appears in the psychological literature. Creating small perceptual groups is also consistent with the human computer interface mantra of focus and context. In LM plots, an individual panel focuses attention on a few study units while the full sequence of panels provides one facet of context.

Creating small perceptual groups can be done in different ways. Carr, Somogyi, and Michaels (1997), for example, grouped genes by gene function first and then created subgroups with four or fewer genes. Using logical criteria for sorting and grouping is often helpful. The cartographic practice of using gaps between values of the sorting variable to partition elements is often advantageous for bringing out spatial patterns. Different objectives motivate different groupings.

Breaking a long list into smaller perceptual units can simplify visual appearance and provide additional visual entry points that might be of interest. However, many small perceptual units can still constitute a long list. Thus small perceptual units can be grouped into larger perceptual units. The design strategy of Carr and Pierson (1996) reflected in Figure 3 uses a 5-1-5 grouping of panels. To our knowledge there is no general theory concerning iterated grouping for long lists.

2.2 Micromaps

The primary task of micromaps is to show the spatial location that corresponds to a statistical estimate. We use five distinct saturated hues as a rapid link between statistical estimates, study unity names, and spatial locations. The cyclic use of five distinct hues often causes confusion when people first encounter LM plots. We discuss this further in Section 2.4. This coupling of name and location in micromaps serves as a reminder for some readers and as an educational device for others. When working with states, some people need an occasional reminder. Went encountering U.S. ecoregions for the first time, the micromaps are instructive.

Carr and Pierson (1996) indicate that micromaps are often spatial caricatures designed to serve specific purposes. Figures 2 and 3 show simple, generalized state boundaries. The design purposes in the two figures include providing sufficient shape detail for state recognition, preserving neighbor relationships (DC is an exception), and enlarging small states to facilitate perception of color.

Micromaps can handle multiple tasks such as showing supplemental information. In environmental monitoring applications involving small areas, micromap details may include streams and various landmarks. Figure 1 illustrates display of three information layers, foreground, middle ground and background. The foreground layer consists of highlighted nations appearing in saturated color. The middle layer uses light yellow with black outlines to indicate OECD nations that are not highlighted in the particular panel. The background layer uses light gray with white outlines to indicate non-OECD nations.

Micromaps are particularly effective when the sequential highlighting of sorted study units reveals spatial patterns. We can often augment micromaps with contours to bring out spatial patterns. Figures 2 and 3 use light yellow and gray to distinguish states above the median from states below the median. In Figure 3 the light yellow contour serves as a spotlight that emphasizes high unemployment states in the upper half of the plot and low unemployment states in the lower half of the plot. In the lower half of the figure the Midwest, Northern Plains, and Southern Coastal states are readily evident as parts of the low rate contour.

The color encoding in Figure 3 is not immediately obvious to all readers. The double use of color for contouring and highlighting can be confusing at first glance. Still, the distinction between unsaturated and saturated color is easy to make and interpretation is easy once the encoding is understood. Using light yellow to emphasize above the median contour for states at the top of the page and the below median contour at the bottom of the page can also be a confusing encoding. An alternative is to use a light blue or other unsaturated color for the below median contour. We are reluctant to introduce yet another color. We conjecture that the notion of shifting

a light yellow spotlight is easily learned. Our purpose is to introduce a middle ground contour rather than two equal emphasis background contours.

Figure 3 calls attention to just two contours, one above and one below the median. This reinforces the statistical concept of median and targets a broad portion of the public that has at least some interest in statistical summaries. Examination of the parallel dot plot panels shows many dots close to the median. The display of other contours would be more consistent with gaps in the unemployment rate distribution.

We indicate three color options of many for showing more contours in Figure 3 to sophisticated audiences. The first uses a shifting light yellow spotlight to focus on the region that combines highlighted units in panels immediately above and below the current panel. The second uses a multicolor spotlight to cover units from more panels. The third approach uses a different color for each state. One such pattern starts with a spectral hue sequence for the panels. Warm colors are on top, bright yellow represents the median and cool colors are below. A lightness ramp within each panel attempts to distinguish the states. Distinguishing states is difficult with fifty-one color schemes, so at present we retain our five distinct hue approach.

Readers might be surprised at the mention of spectral ordering. After citing extensive literature arguing against spectral order, Brewer (1997) cites perceptual studies indicating that some instances of spectral order work quite well. Lightness remains the primary basis for ordering. The spectral order works when used as divergent spectral scheme with bright yellow in the middle. Brewer also discusses color scales for the color blind so the paper is of considerable interest.

In some cases the micromap design itself is less than ideal for showing spatial patterns. For example, Figure 1 illustrates a caricature developed to show OECD nations. The topological distortion and use of two insets is not conducive to properly observing spatial patterns. The Figure 1 micromap sequence might even be deleted since spatial location is not central to the story.

Our notion of micromaps is meant to be general. The examples here involve area representations. Some environmental applications involve monitoring sites that are represented as points. A map may be something other than areas or points on the surface of the earth. For example locations might be position within a building, nodes or links in a formal graph, or even a position in a transition matrix.

Students at George Mason University have developed some of their own micromap variations. A student in a Statistical Graphics and Data Exploration class won an external poster competition by showing sequences of Virginia maps overlaid with pie glyphs at county centroids. The pie glyphs represented crime rates for different classes of crimes summarized at the county level. (The class did not promote pie glyphs for making comparisons). A student in a Scientific and Statistical Visualization class provided a much better example. He redesigned a statistical summary from World War I concerning the effects of mustard gas. The micromaps were caricatures of the human body and clearly showed the susceptibility of exposed and moist locations. Micromaps can take many forms.

2.3 Statistical Summary Panels

The statistical summary panels can take many forms such as dotplots, barplots, boxplots, times series plots, scatterplots, cdf plots, perspective views, stereo pairs plots and so on. While most of these plots are familiar, that does not mean there is a lack of graphical design issues to address.

Statistical summary panels are typically small, so overplotting remains a problem even though the number of highlighted elements in a panel is small. In Figure 1, the time series overplot substantially. Since there are no missing values the reader can infer values for hidden points. When there is missing data, as in Figure 2, sometimes the overplotting is not too bad. When something must be done, less than elegant solutions include plotting dots of different size with large dots plotted first, plotting symbols that remain identifiable when overplotted, staggering plotting locations and so on.

Space constraints continually come into play. It is advantageous to keep a LM plot to one page. We often forego Cleveland's (1993) guidance about banking to 45 degrees and are tempted to skimp on labeling. Uncomfortable compromise, of course, is not unique to LM plots.

Scaling and resolution issues are recurrent in statistical summary panels. Figure 1 deals with the scaling and resolution issue in two ways. First, the selected unit of measure is tons per person. This reduces some of the disparity between small and large population countries. Second, the top panel has a different scale than the other panels. This compromise is problematic. It is difficult to compare time series between panels on different scales. Since we ordered the nations by the time series mean it is not necessarily the case that values for a specific year in the top panel are above those in the second panel. A helpful option is to show the times series from all panels

in the middle ground, clipping the series appropriately for the scale of each panel. The LM plot suggests the presence of a different scale by the separation between the two panels where the change occurs but this is too subtle. Some addition labeling or special scale warning convention would be helpful.

Focus and context issues apply to a variety of statistical summary panels as well as to a variety of micromaps. As another summary panel example, it can be advantageous to plot all the points in all scatterplots as a middle layer and then overplot the highlighted points for the particular panel. In some cases a translucent middle layer helps by keeping background grids visible.

While there are many design issues, statistical summary panels can nonetheless accommodate much information that helps in interpretation. Carr and Pierson (1996) discuss the design of confidence bounds shown in Figure 3. Note also the dashing line that represents U.S. average as a reference value. While statistics are beyond some segments of the public, the LM plot design makes the notion of median almost self-explanatory and the difference between the median state value and the national average is readily evident.

2.4 Linking Legends and Visual Guides

The linking legend typically uses close juxtaposition to link the name of each unit of study to a symbol such as a dot. In the three figures dot color links the names to micromap regions and to elements in the statistical summary plots. When the study units have point locations, symbol shape may also serve as a link. In Figure 3, the statistical panels are dot plots. For this case the state names link directly to statistical summary elements by vertical position as well as by color. People often have a difficulty when they first encounter LM plots because saturated color only links across panels highlighting a few units of study. Their prior experience suggests that color will retain the same meaning for the full sequence of micromaps and statistical graphics panels. This adds to the learning curve for LM plots but the learning curve is usually short.

Figure 2 illustrates a different LM plot design that includes visual guides (lines) from names to symbols. Something needed to be done to address over-plotted values for annual estimates. Rather than stagger the plotting locations, the design choice uses connecting lines from the state names and to annual estimates. Extending the lines to the right of the plot and placement of the tic labels between panels enables quick reading of annual values.

2.5 LM Plot Labeling

Labeling is one of the most difficult challenges in graphical design. An ill-chosen word can confuse the reader. Lack of explanation can leave the reader confused or with a totally wrong interpretation. At the same time space is at a premium and words can interfere with the power of the eye brain system to perceive graphical patterns. We do not claim to have a general solution, but attempt to address problems as they arise. In Figure 3 a legend concerning confidence bounds and reference line now appears at the bottom of the panel sequence in which they appear. There is now a label for the median state. Labels also suggest that the light yellow calls attention to states above the median for panels about the median and to states below the median for panels below the median.

Labeling placement can also be used to create perceptual groups. Three columns appear simpler than four columns. In Figure 3 we attempt to combine the micromaps and linking labels into one perceptual unit by centering the label over the two columns of panels. For most people this may make little difference, but if we can invite even a few more people into the world to statistical graphics, that is good.

3. Comments on Plot Interpretation

This article emphasizes the LM template. However, some comments about plot interpretation and remaining design issues for the examples seem appropriate.

3.1 Figure 1

Figure 1 shows an OECD (Organization for Economic Co-operation and Development) time series of per capita CO2 emissions for energy use. The data on CO2 emissions for energy use and population size comes from the OECD Environmental Data Compendium 1995, pages 39 and 283 respectively. The population data is incomplete, so we interpolated population values as necessary to produce yearly per capita estimates.

In terms of interpretation, population interpolation is a minor concern. While tabling values in OECD reports provides some pressure for international consistency, this pressure is minor compared to limitations in available methodology, limitations in assessment resources, and pressures of maintaining national images. Thus, comparisons across nations without a deep understanding of the nation specific estimation process can be misleading. With times series data, there is some hope for consistency for individual nations over time.

When first looking at Figure 1, the magnitude of 20 tons of CO2 emissions per person in the U.S. was a shock.

That the U.S. was not top on the list was also a surprise. However, those familiar with Luxembourg describe special circumstances consistent with high values. The unusual value for Iceland is a rounding artifact with the tabled numerator being represented by a single digit. An encouraging hint of declining values appears for Germany, France, and Sweden. (German unification leads to interesting accounting issues in regard to future improvement.) The figure suggests increasing per capita rates for nations such as Ireland, Japan, New Zealand, Greece and Portugal.

In terms of total emissions, even flat per capita patterns are cause for concern when coupled with increasing population. For example the Mexican population is increasing rapidly. Of course the U.S population continues to increase due to immigration and U.S. per capita values are much higher than Mexican values. Our perhaps injudicious interpretation is that nations tend to have CO2 emission styles that are relatively stable for the reporting period, that total emissions are linked to population, and that as far as we know population growth is not under control.

As indicated by Wood (1992) plots reflect some agenda. The agenda behind showing amount per capita was both to reduce the tremendous range of values and to present the information on a personal level. In a chance airplane conversation, a consultant for U.S. utilities looked at the plot and suggested reporting CO2 emissions per gross domestic product. His agenda was to make the U.S. appear as a waste minimizing (efficient) energy producer. The verbal battle over greenhouse gas emissions will continue.

Developing a micromap for OECD nations was a design challenge. Figure 1, takes a variety of liberties, not only slicing a way most of the Atlantic and enlarging small countries, such as Luxembourg, at the expense of others, but also by using two insets, one for Australia and New Zealand and one for Japan. Is the distortion too much for those familiar with maps of Europe? Will the inset placing Japan on Russia arouse political sensitivities? We have not addressed such issues, but note the micromap will need to be revised to incorporate the three new OECD nations that appear in the 1997 compendium. Revising an already published OECD view (The State of the Environment, OECD 1991, page 134) may provide a solution, but showing small nations is still a challenge.

3.2 Figure 2

Figure 2 re-expresses a portion of a table published on page A-29 in Agriculture Prices (release date January 31, 1994) by The National Agriculture Statistics Service. The micromaps call attention to the higher wheat prices in the West Coast and Northern States. Are transportation costs involved? Is the total amount available at different times during the year a major factor determining price factor? The time series indicate missing data. Why is it missing? A big question that jumps out in the graphical representation concerns the mismatch between the marketing year average for each state (footnoted as being preliminary) and the monthly time series values. In some cases the marketing year average for a state is near extreme values for the state, and that implies heavy weighting of specific months. An explanation about the weighting seems appropriate. Likely the weighting explanation is available in related Department of Agriculture documents.

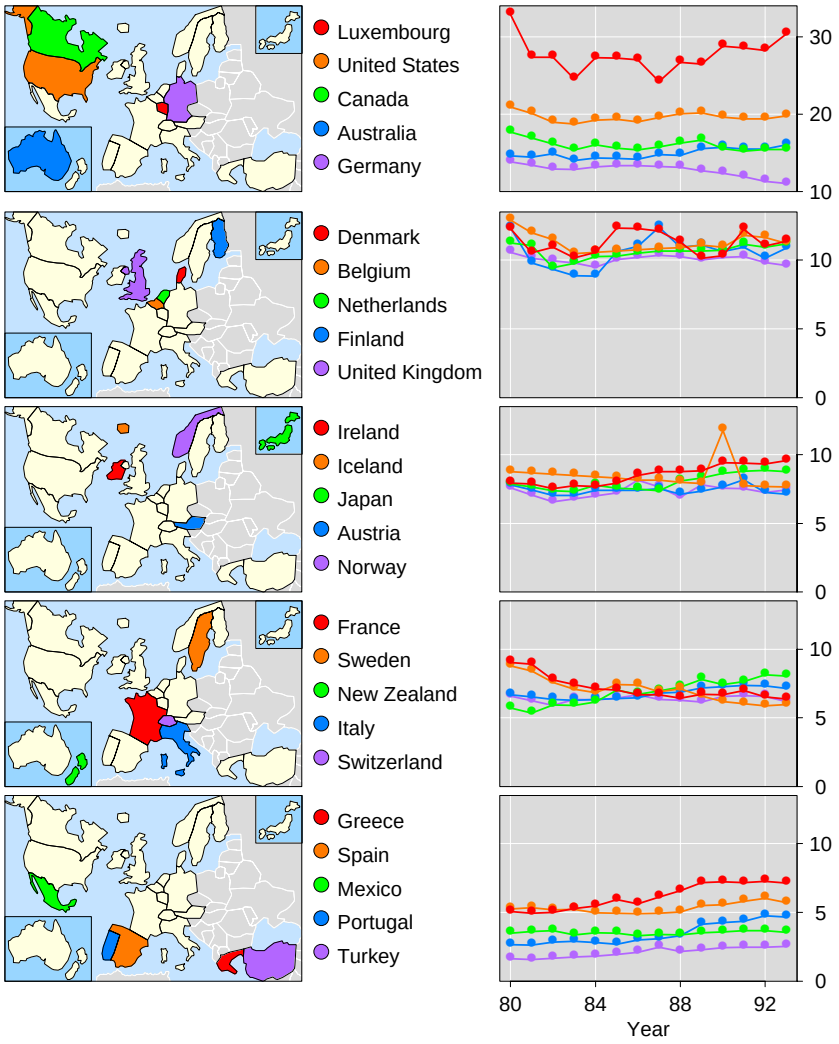
3.3 Figure 3

The primary source for Figure 3 is the Geographic Profile of Employment and Unemployment, 1995, U. S. Department of Labor Bureau of Labor Statistics, Bulletin 2486. Carr and Pierson (1996) propose a LM plot as a replacement for a choropleth map in that document. They discuss the relative merits of choropleth maps and LM plots and we encourage readers to read the article. Figure 3 takes a step further from the previous LM plot, showing contours, the U.S. average as a dashed line, and more labeling. The plot tells a reasonably complete story indicating estimates, estimate precision, estimate importance (the number unemployed), and estimate location. A deeper interpretation item that is not shown concerns the determination of who is excluded from the numerators and denominators in the determination of rates.

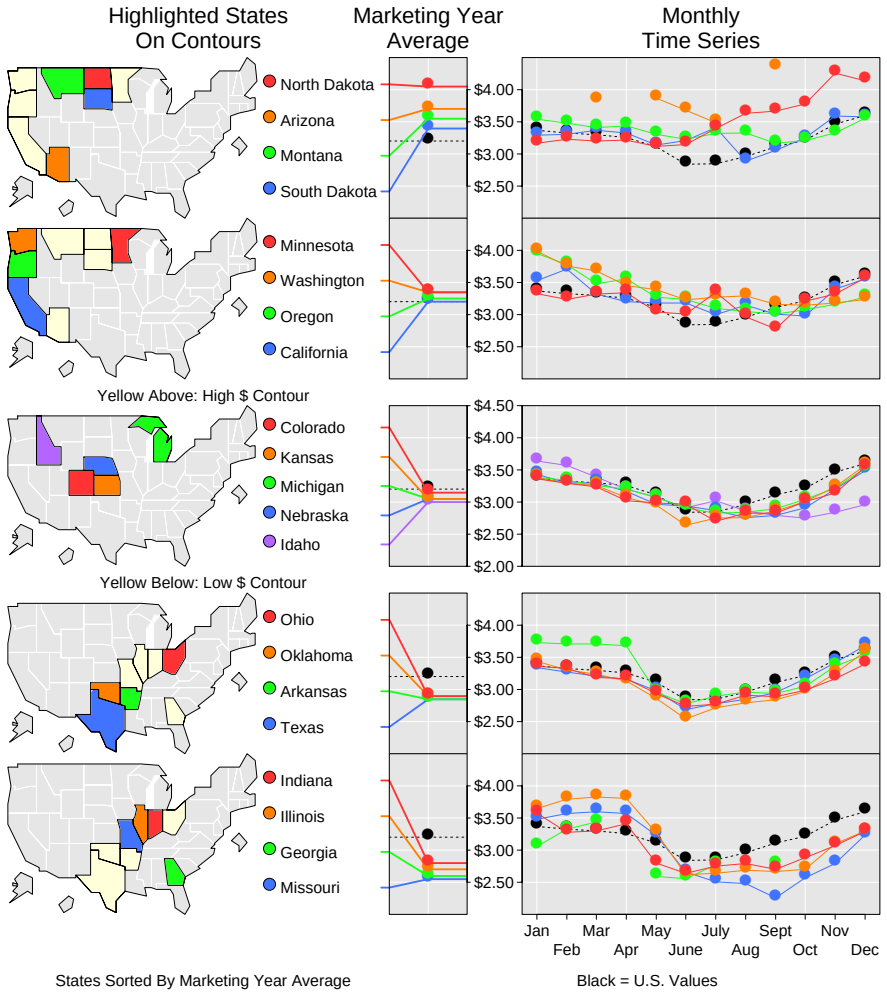
4. LM plot history, Connections to other Research and Challenges

We claim that LM template is new, but there are, of course, many connections to previous graphics and conceptualizations. While we were intrigued by the thumbnail images of Eddie and Mockus (1996), a stronger connection is to the work of Edward Tufte. The LM plots belong to class of graphics that Tufte (1983, 1993, 1997) calls small multiples. In "The Visual Display of Quantitative Data," his eloquent description of well-designed small multiples include phrases such as "inevitably comparative", "deftly multivariate", "efficient in interpretation", and "often narrative in content." We designed LM plots with the hope that such phrases would apply. In "Visual Explanations," Tufte calls particular attention to explanatory power of parallelism. While our use of parallelism precedes this book, Tufte's

Annual CO2 Emissions From Energy Use
Units = Tons Per Person



Wheat Prices Received By State: 1993
Dollars Per Bushel



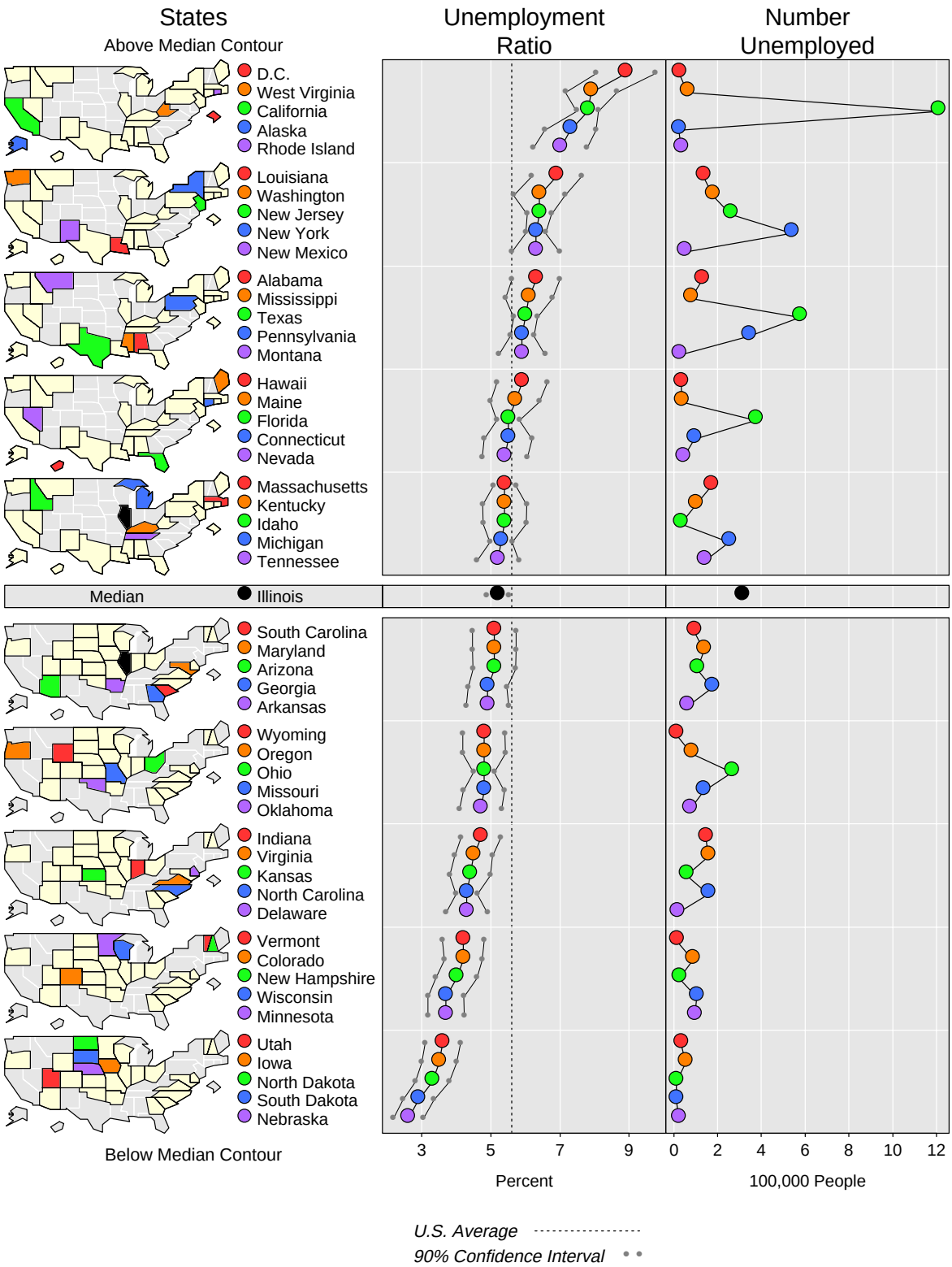


Figure 3. Linked Micromap of 1995 unemployment figures taken from the Bureau of Labor Statistics.

earlier examples may well have guided us to make parallelism a fundamental part of the LM plot design.

As indicated in the introduction, LM plots emerged as at way of linking row-labeled plots to maps. The row-labeled plots in turn build upon of Cleveland (1985) and Cleveland and McGill (1984). In fact the development of row-labels plots was part of an effort to encourage EPA staff to use Cleveland's dotplots in EPA graphics. Dissatisfaction with the look of early S-plus dotplots (see Cleveland 1993b for an example) and the promise of multiple panel layouts for expressing complex tables as plots lead to development of new S-plus functions (Carr 1994a and Carr 1997). While the row-labeled plot development was independent of the Trellis graphics development, there are similarities. This is not surprising since Cleveland's design ideas were important in both and S-plus was a common computing environment.

The linking of maps and statistical graphics also builds upon the work of Monmonier (1988) who connected contemporary methodology from cartographic and statistical graphics communities. One of his many interesting examples has a map on the left, labels in the middle and bar plots on the right. This example is a precursor to our LM plots. We have followed Monmonier's work over the years. The statemap caricature in Carr and Pearson (1996) was specifically inspired by a more recent reading of Monmonier (1993) and adapted from coordinates that he graciously supplied.

Despite connections to other research, we claim the LM template is new. Many people have said "There is nothing new under the sun." The truth of this statement always depends on ignoring distinctions. Presumably the speakers of the statement are not identical but if so they collectively get just one vote. We note that the use of parallel sequences of small multiples is relatively uncommon and have called attention to defining features in Section 2. Ultimately others will have to judge the distinctiveness of specific examples and the template in general. What is most important, however, is not the newness of the template, but rather its utility, community awareness of its relative merits, plot production convenience, and statistical graphics literacy.

Plot production convenience remains a big issue. If LM plots are to be used they need to be easily produced. The general S-plus tools we developed (anonymous ftp to `galaxy.gmu.edu` and change directory to `pub/dcarr/newsletter/lmplots`) are flexible building blocks but not easy push button tools. The software also includes a Visual Basic front end that Andrew Carr developed to simplify production of LM plots similar to Figure 3. This is a start toward simple production. Much work remains to design micromaps for new applications and to develop software that makes it easy to produce a wide range of LM plots. Much research is appropriate concerning compromises and variations that are motivated by plot purpose, audience, specific data and metadata.

The other big recurrent issue is statistical literacy. Carr and Pierson (1996) suggest that if federal agencies distribute estimates with confidence boundaries, then the Web literate public will grow comfortable with the general idea. Similarly, medians and other statistics can become familiar. A big challenge is to start the ball rolling with federal statistical graphics distributed on the Web (see Carr, Valliant and Rope 1996), a topic to be revisited in future articles.

Acknowledgements

The majority of the work behind this paper was supported by the EPA under cooperative agreement No. CR8280820-01-0. Some facets of this work have been supported by BLS, NASS, and NCHS. The article has not been subject to review by EPA, BLS, NASS, and NCHS so does not necessarily reflect the view of the agencies, and no official endorsement should be inferred.

We wish to thank Wing K. Chong who helped in developing the OECD micromap and the many people commenting at our past presentations.

References

Brewer, C. A. (1997), "Spectral Schemes: Controversial Color Use on Maps," *Cartography and Geographic Information Systems*, Vol. 24, No. 4, pp. 203-220.

Carr, D. B. (1994a), "Converting Plots to Tables," Technical Report No. 101, Center for Computational Statistics, George Mason University, Fairfax, VA.

Carr, D. B. (1994b), "A Colorful Variation on Boxplots," *Statistical Computing & Graphics Newsletter*, Vol. 5, No. 3, pp. 19-23.

Carr, D. B. (1997), "Some Simple Splus Tools for Matrix Layouts," Bureau of Statistics Statistical Note Series, No. 42.

Carr, D. B. and A. R. Olsen (1996), "Simplifying Visual Appearance By Sorting: An Example Using 159 AVHRR Classes," *Statistical Computing & Graphics Newsletter*, Vol. 7 No. 1 pp. 10-16.

Carr, D. B., R. Somogyi and G. Michaels (1997), "Templates for Looking at Gene Expression Clustering," *Statistical Computing & Graphics Newsletter*, Vol. 8, No. 1, pp. 20-29.

Carr, D. B. and S. Pierson (1996), "Emphasizing Statistical Summaries and Showing Spatial Context with Micromaps," *Statistical Computing & Graphics Newsletter*, Vol. 7, No. 3, pp. 16-23.

Carr, D. B., R. Valliant, and D. Rope (1996), "Plot Interpretation and Information Webs: A Time-Series Example From the Bureau of Labor Statistics," *Statistical Computing & Graphics Newsletter*, Vol. 7, No. 2, pp. 19-26.

Cleveland, W. S. (1985), *The Elements of Graphing Data*, Hobart Press, Summit, NJ.

Cleveland, W. S. (1993a), *Visualizing Data*, Hobart Press, Summit, NJ.

Cleveland, W. S. (1993b), "Display Methods of Statistical Graphics," *Journal of Computational and Graphical Statistics*, Vol. 2., No. 4, pp. 327.

Cleveland, W. S. and R. McGill. (1984), "Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods," *Journal of the American Statistical Association*, Vol. 79, pp. 531-554.

Eddy, W. F. and A. Mockus (1996), "An Interactive Icon Index: Images of the Outer Planets," *Journal of Computational and Graphical Statistics*, Vol. 5., No. 1, pp. 101-111.

Kosslyn, S. M. (1994), *Elements of Graph Design*, W. H. Freeman and Company, New York, NY.

Monmonier, M. (1988), "Geographical Representations in Statistical Graphics: A Conceptual Framework," American Statistical Association 1988 Proceedings of the Section on Statistical Graphics, American Statistical Association, Alexandria VA. pp. 1–10.

Monmonier, M. (1993), *Mapping It Out*, The University of Chicago Press, Chicago, IL.

Olsen, A. R., D. B. Carr, J. P. Courbois, and S. M. Pierson (1996), "Presentation of Data in Linked Attribute and Geographic Space," Poster presentation, 1996 ASA Annual Meeting, Chicago, Il.

Tufte, E. R. (1983), *The Visual Display of Quantitative Information*, Graphics Press, Cheshire, CT.

Tufte, E. R. (1990), *Envisioning Information*, Graphics Press, Cheshire, CT.

Tufte, E. R. (1997), *Visual Explanations*, Graphics Press, Cheshire, CT.

Wood, D. (1992), *The Power of Maps*, The Guilford Press, NY.

Dan Carr
Institute for Computational Statistics and Informatics
George Mason University
dcarr@galaxy.gmu.edu

Tony Olsen
EPA National Health and Environmental Effects Research Laboratory
tolsen@mail.cor.epa.gov

Pip Courbois
Oregon State University
courbois@stat.orst.edu

Suzanne M. Pierson
OAO Corporation
spierson@mail.cor.epa.gov

D. Andrew Carr
Bureau of Labor Statistics
carr_a@bls.gov

Statistical Computing Activities at the JSM

By Russell D. Wolfinger

The Statistical Computing Section is sponsoring an information-packed slate of invited and contributed sessions, posters, and roundtables for the Joint Statistical Meetings, August 9-13, 1998, in Dallas, Texas. The invited sessions are as follows:

- Data Mining, Sun 2:00-3:50, chaired by Don Sun, featuring Andreas Buja, Chidanand Apte, and Robert Chu
- Computing for Large Mixed Models, Mon 10:30-12:20, chaired by Russ Wolfinger, featuring David Harville, Yurii Bulavsky, and Stephen Smith
- Bootstrapping Time Series, Tues 8:30-10:20, chaired by Tim Hesterberg, featuring Joseph Romano, Hans-Ruedi Kuensch, and Lori Thombs
- Smoothing Methods and Data Analysis, Wed 10:30-12:20, chaired Michael O'Connell, featuring Steve Marron, Doug Nychka, and Chong Gu
- Internet Developments, Wed 2:00-3:50, chaired by Balasubramanian Narasimhan, featuring P.B. Stark, R. Todd Ogden, Jan de Leeuw, Duncan Temple Lang, and Jim Rosenberger
- Computer Algebra Systems, Thur 8:30-10:20 chaired by John Kinney, featuring Eliot Tanis, John Kinney, and Barbara Heller

The contributed sessions are as follows:

- Density Estimation and the Bootstrap, Sun 4:00-5:50, chaired by Ed Wegman, featuring David Scott
- Computing in Time Series, Mon 8:30-10:20, chaired by Michael Leonard
- Computing and Statistical Inference, Mon 2:00-3:50, chaired by Morteza Marzjarani
- Prediction and Simulation Algorithms, Tues 8:30-10:20, chaired by Anwar Hossain
- Computing in Psychometrics, Tues 10:30-12:20, chaired by Yiu-Fai Yung
- Regression Computation, Wed 8:30-10:20, chaired by Robert Cohen
- EM Algorithm and Imputation, Thurs 10:30-12:20, chaired by Dan McCaffrey

The roundtables are as follows:

- The Future of Web-based Computing, R. Webster West
- Bayesian Computation, Merlise Clyde
- Information Theory and Statistical Reasoning, Bin Yu
- Directions in Computing for Statistics, John Chambers

The Student Award Winners session will be Thurs 10:30-12:20 and chaired by Lionel Galway. The winners this year are Alessandra Brazzale, Matthew Calder, Yan Yu, and Steven Scott. Six posters will also be presented at the meetings. Finally, several other ASA sections are sponsoring invited and contributed sessions that may be of interest to you. Statistical Computing will be listed as a co-sponsor of these sessions in your JSM program.

Please mark your schedules to attend these "state-of-the-art" computing activities!

Russell D. Wolfinger
SAS
sasrdw@sas.com

Statistical Graphics at the JSM

By Ed Wegman

The Statistical Graphics Section is sponsoring three invited sessions, two special contributed sessions as well as co-sponsoring a regular invited session at JSM 98. Session 90, Real Success Stories of Statistical Graphics, was designed to highlight success stories for graphical methods. Often, graphics papers tend to show how graphical methods can uncover facts already known by other methodologies. Here we are intending to shed light on facts that have not or could not be known by other methods.

With the ubiquitous presence of the World Wide Web and its potential for creating a revolution in the scientific investigation process, in Session 138, Statistical Graphics on the Web, we intended to explore the Web's potential for statistical graphics. Clearly animation, color and even three-dimensional, stereoscopic graphics are possible on the Web and our authors in this session explore

those possibilities. Along with the Web, data mining has become a popular theme in recent years. In Session 277, Graphics for Data Mining, we examine the potential of graphical methods, especially high interaction graphical methods for data mining.

Session 12, Applications of Graphics, is a special contributed session focusing on graphical methods in several different application areas including survey research, drug design and psychometrics. Session 53, Visualization & Smoothing, is a second special contributed session which not only features traditional themes for the Statistical Graphics Section, but which also features the winners of the Student Paper Competition. Finally, Session 36, Density Estimation and the Bootstrap, a regular contributed session jointly sponsored with the Statistical Computing Section features a number of well-known speakers including Professor David Scott, the overall program chair of JSM 98. In addition to the sessions mentioned here, Statistical Graphics is co-sponsoring a number of other invited and contributed sessions with graphics and visualization themes. The complete list of invited sessions follows.

JSM Session 90 Real Success Stories of Statistical Graphics, 8/10 2:00 PM–3:50 PM

- Applications of Statistical Graphics to Discriminant Analysis By Jeffrey L. Solka and David J. Marchette
- Counts - Proportions - Interactions: A View on Categorical Data By Martin Theus and Adalbert Wilhelm
- Large Scale Genome Analysis Approaches: Integrating Bioinformatics and Data Mining By Roland Somogyi and George Michaels

JSM Session 138 Statistical Graphics on the Web, 8/11 10:30 AM–12:20 PM

- Web-Based Visualization By Stephen G. Eick
- Untangling the Web: Java Beans for Disseminating Information Through Interactive Graphics By Daniel J. Rope and Daniel Carr and Leland Wilkinson
- The Data Image: A Tool for Exploring High Dimensional Data Sets By Michael C. Minnotte and R. Webster West

JSM Session 277 Graphics for Data Mining, 8/13 10:30 AM–12:20 PM

- A Successful System Using Data Mining for Direct Marketing Response Prediction By James G. Wendelberger
- Mining the Sands of Time By Edward J. Wegman and Adalbert Wilhelm and Juergen Symanzik
- Crystal Vision: A Graphical Tool for Data Mining By Qiang Luo

Ed Wegman
George Mason University
ewegman@gmu.edu

1998 Student Paper Competition: The Graphics Section

By Lorraine Denby

The winners of the Statistical Graphics Section first annual student paper competition will be featured on Monday August 10 at the 8:30 session in Dallas. John W. Emerson from Yale University will be discussing "Examining Patterns in Television Viewing with Mosaics in S-PLUS." His paper offers a general S-PLUS implementation of mosaic displays, and uses mosaics to explore television viewer data from Nielsen Media Research. The mosaics show interesting patterns in viewer persistence – viewers remaining with their current channels rather than switching channels. Elizabeth R. Brown from University of Colorado is talking about "A Bivariate Smoothing Spline Model of Changes in Plasma Viral Load and CD4+ Lymphocyte Count in HIV/AIDS Patients." She uses a cubic smoothing spline to model unequally spaced measures of CD4 count and viral load for HIV/AIDS patients and displays the results in interesting graphs.

We'd like to congratulate these students on their fine work and encourage you to attend this session.

The deadline for next year's competition is January 1999. It is not too early to start encouraging your students to start working towards submitting something for next year. The winners receive up to $1000 towards expenses for attending the JSM.

Lorraine Denby
Bell Laboratories
ld@research.bell-labs.com

SECTION OFFICERS

Statistical Graphics Section - 1998

Michael M. Meyer, Chair
412-268-3108
Carnegie Mellon University
MikeM@stat.cmu.edu

Dianne H. Cook, Chair–Elect
515-294-8865
Iowa State University
dicook@iastate.edu

Sally C. Morton, Past–Chair
310-393-0411 ext 7360
The Rand Corporation
Sally_Morton@rand.org

Edward J. Wegman, Program Chair
703-993-1680
George Mason University
ewegman@gmu.edu

Deborah Swayne, Program Chair–Elect
973-360-8423
AT&T Labs – Research
dfs@research.att.com

Antony Unwin, Newsletter Editor (98-00)
49-821-598-2218
Universität Augsburg
unwin@uni-augsburg.de

Robert L. Newcomb, Secretary/Treasurer (97-98)
714-824-5366
University of California, Irvine
rnewcomb@uci.edu

Michael C. Minnotte, Publications Liaison Officer
801-797-1844
Utah State University
minnotte@math.usu.edu

Lorraine Denby, Rep.(96-98) to Council of Sections
908-582-3292
Bell Laboratories
ld@bell-labs.com

David W. Scott, Rep.(98-00) to Council of Sections
713-527-6037
Rice University
scottdw@rice.edu

Roy E. Welsch, Rep.(97-99) to Council of Sections
617-253-6601
MIT, Sloan School of Management
rwelsch@mit.edu

Statistical Computing Section - 1998

Karen Kafadar, Chair
303-556-2547
University of Colorado-Denver
kk@tiger.cudenver.edu

James L. Rosenberger, Chair–Elect
814-865-1348
The Pennsylvania State University
JLR@stat.psu.edu

Daryl Pregibon, Past–Chair
908-582-3193
AT&T Laboratories
daryl@research.att.com

Russel D. Wolfinger, Program Chair
919-677-8000
SAS
sasrdw@sas.com

Mark Hansen, Program Chair–Elect
908-582-3869
Bell Laboratories
cocteau@bell-labs.com

Mark Hansen, Newsletter Editor (96-98)
908-582-3869
Bell Laboratories
cocteau@bell-labs.com

Merlise Clyde, Secretary/Treasurer (97-98)
919-681-8440
Duke University
clyde@isds.duke.edu

James S. Marron, Publications Liaison Officer
919-962-5604
University of North Carolina, Chapel Hill
marron@stat.unc.edu

Janis P. Hardwick, Rep.(96-98) Council of Sections
313-769-3211
University of Michigan
jphard@umich.edu

Terry M. Therneau, Rep.(97-99) Council of Sections
507-284-1817
Mayo Clinic
therneau@mayo.edu

Naomi S. Altman, Rep.(97-99) to Council of Sections
607-255-1638
Cornell University
naomi_altman@cornell.edu

INSIDE

Statistical COMPUTING & GRAPHICS

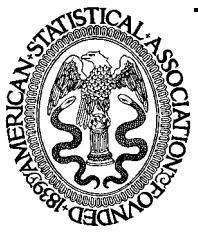
The *Statistical Computing & Statistical Graphics Newsletter* is a publication of the Statistical Computing and Statistical Graphics Sections of the ASA. All communications regarding this publication should be addressed to:

Mark Hansen
Editor, Statistical Computing Section
Statistics Research
Bell Laboratories
Murray Hill, NJ 07974
(908) 582-3869 • FAX: 582-3340
cocteau@bell-labs.com
cm.bell-labs.com/who/cocteau

Antony Unwin
Editor, Statistical Graphics Section
Mathematics Institute
University of Augsburg
86135 Augsburg, Germany
+49-821-5982218 • FAX: +49-821-5982280
unwin@uni-augsburg.de
www1.math.uni-augsburg.de/~unwin/

All communications regarding membership in the ASA and the Statistical Computing or Statistical Graphics Sections, including change of address, should be sent to:

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



Nonprofit Organization
U. S. POSTAGE
PAID
Permit No. 50
Summit, NJ 07901

American Statistical Association
1429 Duke Street
Alexandria, VA 22314-3402
USA

This publication is available in alternative media on request.