

Computing in the Mathematical Statistics Course

Samuel E. Buttrey, Naval Postgraduate School, Monterey
Deborah Nolan, University of California, Berkeley
Duncan Temple Lang, Bell Labs, Murray Hill
Deborah Nolan, Statistics, U.C.B., Berkeley, CA 94720-3860

Abstract

We have developed a model for teaching mathematical statistics through detailed case studies, and we have had some success with students using the R and S-Plus languages to analyze data from the cases and to perform simulation studies of statistical concepts underlying the methods used in the analysis. We describe a few examples of these simulation studies, and explain how we use the computer to bridge the gap between statistical theory and practice and to develop understanding of the basic ideas in mathematical statistics. Recently, we have embarked on a new approach that builds on the work of the Omegahat project, which has embedded the R statistical software in the Netscape browser. We are designing interactive environments for browsers that use R for performing computations and plotting. The goal is to develop electronic activities that will make it easier for students to explore and develop understanding of the concepts of statistics and that will connect these concepts to the practice of statistics.

Key Words: Education, Interactive documents, Web

Introduction

We use statistical software for teaching mathematical statistics to undergraduates in two ways. Students use software to analyze data from case studies where they work on real-life problems from diverse fields. They also use software for simulations and demonstrations that are designed to help them develop intuition for – and understanding of – key statistical notions.

We have found that when case studies are central to the mathematical statistics course, then the cases motivate students to learn the theory, and they provide a meaningful framework for the material. Case studies give students experience with

how statistics can be used to answer scientific questions, which in turn encourages statistical thinking. Additionally, we have found that computer demonstrations and simulations help students understand statistical concepts, and when tied to an application, they aid students in developing their quantitative reasoning skills.

In this paper, we describe our model for teaching mathematical statistics through case studies, and we provide examples of how we use statistical software to bridge the gap between the theory and the practice of statistics. Next, we discuss our design of new computer activities that use the statistical software R in Netscape to create a richer environment for students to explore and study statistical concepts. This is part of a larger project (Buttrey et al., 2001, www.stat.berkeley.edu/~statdocs/) to develop an interactive, electronic text for teaching mathematical and applied statistics.

Teaching the mathematical statistics course

We teach mathematical statistics using Nolan and Speed (2000), a text that blends mathematical statistics with modern statistical practice. In this text each chapter is centered on a case study, and it is via the case study that the theory is introduced. This leads to an integration of statistical theory and practice in a way not commonly encountered in an undergraduate course. The cases raise scientific questions that are interesting in their own right, and they contain data for use in addressing these questions. The context of the scientific question is the starting point for developing the statistical theory.

The presentation of a case is divided into five main parts: introduction, data description, background material, investigations, and theory. In the

introduction, a clear scientific question is stated, and motivation for answering it is given. The question is presented in the context of the scientific problem, and not as a request to perform a particular statistical operation. We also provide students with documentation for the data collected to address the question, including a detailed description of the study protocol. In addition, students read background material to put the problem in context. This information is gathered from a variety of sources, and presented in nontechnical language. With the background material, we attempt to create a detailed picture of the field of application that is understandable to a broad college audience.

The student's assignment consists of suggestions for answering the question posed in the introduction. These suggestions are written in the context of the problem, and they use very little statistical terminology. Students use R to analyze the data. The data are often complex, the analysis open-ended, and a coherent synthesis of findings required. As a result, the analysis often includes simulation and bootstrap studies.

After completing their analyses, students prepare reports on their findings. We ask students to play the role of a statistician when writing their reports. For example, the student must write an article for a widely read science magazine, a memo to the head of a research group, or a pamphlet for consumers.

We note that the theoretical development of the statistical concepts and methodology required to address the question are introduced after the presentation of the problem as they perform their investigation. This approach to teaching mathematical statistics provides students with a reason to learn statistical theory because it is needed to solve a real problem, and it also generates an interest in learning the material because the problems have depth and merit. Students gain practical experience, problem-solving skills, and the ability to apply what they learn in the theoretical course to the real applications. We view the divide between applied and theoretical statistics as unnecessary and unnatural, and think the integration of the two makes for a more compelling and complete statistics course.

Computing in the mathematical statistics course

We provide three examples of how we use the computer to explore some ideas in mathematical statis-

Time	Bootstrap	
	Count	population
0	57	197
0.1	1	3
0.5	5	17
1	5	17
1.5	1	4
2	14	48
3	3	11
4	1	3
5	1	4
14	2	7
30	1	3
Total	91	314

Table 1: Distribution of time, in hours, spent playing video games in the week prior to the survey for the 91 statistics students in the sample.

tics. These examples are used in the course in a variety of ways: as an exercise in a homework assignment, an in-class demonstration to assist in the understanding of theoretical material, and as part of the data-analysis project described in the previous section. In the first example, we employ the bootstrap to get a picture of the sampling distribution of an estimator. The second example performs a sensitivity analysis on a statistical test. The final example shows students how to search for a maximum of a response surface using first and second order local approximations.

Sampling distribution of the sample average

We teach our students that the average from a simple random sample has an approximate normal distribution when the sample size is large in absolute terms and small relative to the population size. They invariably ask us how large is large and how small is small, and our answer is that it depends on the population. This is not a very helpful answer when faced with actual data, and students are often dissatisfied with this response. The bootstrap addresses this problem.

For example, in a study of the time spent playing video games by students in a large statistics course, a survey of 91 of the 314 students found that on average students played video games for 1.2 hours with an SD of 3.5 hours in the week before the survey was taken. The distribution of time spent playing is highly skewed with 57 of the 91 stu-

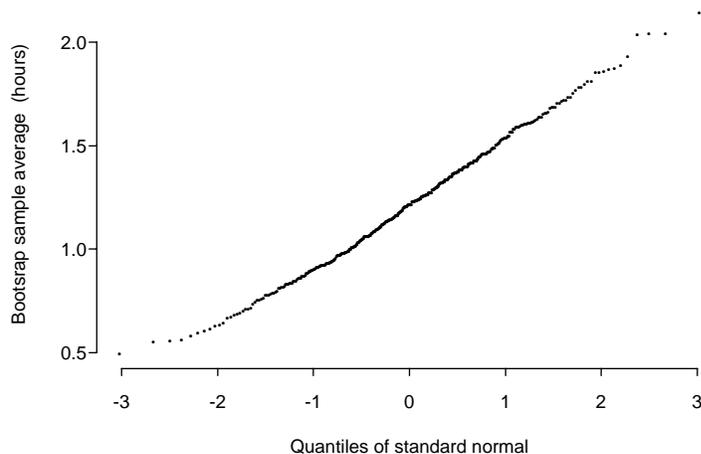


Figure 1: Normal-quantile plot of the bootstrap sample averages, from 400 bootstrap samples of size 91 from the bootstrap population (Table 1), for the 91 statistics students in the sample.

dents spending no time at all playing video games.

The bootstrap helps us assess whether or not the distribution of the sample average approximately follows the normal curve. We build a bootstrap population of 314 based on the sample. For every unit in the sample, we make $314/91 = 3.45$ units in the bootstrap population with the same time value (we round cell counts off to the nearest integer). The sample and bootstrap population appear in Table 1. We draw samples of 91 repeatedly from this bootstrap population to simulate the distribution of the sample average. Figure 1 shows the normal quantile plot of 400 sample averages; that is, the averages of 400 simple random samples of 91 from the bootstrap population. We see that the averages appear to follow the normal distribution. We continue the simulation exercise by comparing the skewness (0.19) and kurtosis (2.67) of the 400 bootstrap sample averages to a simulated distribution of skewness and kurtosis for samples of size 400 from the normal distribution, and find that the observed values are reasonable.

Chi-square test of the uniform distribution

In statistics, we often hypothesize that observations are realizations of independent random variables from a specified distribution, such as the uniform. We do not believe that the data are exactly gener-

ated from such a distribution but that for practical purposes the probability distribution does well in describing the randomness in the outcomes measured. We show our students techniques, such as the χ^2 goodness-of-fit test, for assessing how well a model fits.

In teaching hypothesis testing, we often give our students rules of thumb for when it is appropriate to apply a test. For example, the expected cell counts should not fall below 5 in order for the χ^2 approximation to be appropriate. We have our students perform a simulation study on the sensitivity of the test to low expected cell counts for the uniform distribution. They simulate the distribution of the test statistic for several sample sizes and bin widths. From this simulation study they find that the uniform distribution is not particularly sensitive to violations of this rule.

Searching for the maximum of a response surface

In an experiment to find the dimensions of the best paper helicopter, we have students search the two-dimensional design space (wing width and length) using first-order and second-order local approximations to the response surface (time to descent).

To help them visualize the search, we make contour plots of the locally fitted surface. After building and flying helicopters from these design points,

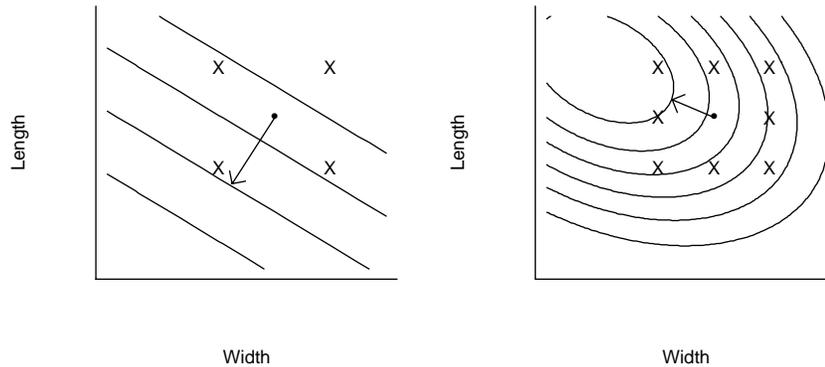


Figure 2: A linear contour plot (left) and quadratic contour plot (right) fitted to the data at the design points, which are marked with crosses. The arrow denotes the path of steepest ascent.

students use our R function to draw the contours of the fitted quadratic and the path of steepest ascent from the center of the design. Then students choose a new point along this line for building helicopters, and continue to explore the design space until they settle on the region where they think the best performing helicopter is located.

A new approach

Recent advances in technology (www.omegahat.org) enable the R software to be plugged into the Netscape browser. This new capability has great potential for creating innovative educational statistical activities (Buttrey et al. 2001). We revisit the three examples described in the previous section, and illustrate how this plug-in can improve these activities.

Figure 3 shows a screenshot of the adaptation of the bootstrap/Central Limit Theorem demonstration to this new approach. We lay out the areas of user input in HTML, handle events through JavaScript, and generate random numbers and figures in R. The student clicks on a radio button to choose whether the sampling is to be with or without replacement, and specifies the sample size. Then to generate a sample, the student clicks the **Sample** button. A JavaScript expression handles this event by retrieving the student's inputs and passing them as values in a call to an R function which generates 400 bootstrap samples and computes the mean for each sample. Then R plots the bootstrapped sampling distribution, and returns its

mean and SD. The JavaScript command displays these numbers in the corresponding text fields under the plot. The visual contrast between the bootstrap population (on the left) and the bootstrap sampling distribution of the average (on the right) is quite striking. Students continue to work with this activity by changing the sample size and by sampling with replacement instead of without to see the effects on the sampling distribution.

Next, for the sensitivity analysis of the χ^2 goodness-of-fit test, we break the activity into two parts (Figure 4). The first part is designed to give students an idea of the variability in the χ^2 goodness-of-fit test for the uniform distribution. The student chooses the sample size and number of bins, then pushes the **Sample** button to generate a sample from the uniform, plot it, and compute the test-statistic and p -value. In the second part of the activity, the student simulates the null distribution of the test statistic for various sample sizes and bins. We provide the student with a visual comparison of the simulated distribution of the test statistic to the χ^2 distribution, and we also provide a comparison of the tail probability for the α -levels of 0.05 and 0.01.

For the third activity, we build a tool to help the student search for the maximum of a response surface (Figure 5). First, the student determines the wing width and length of interest and the search increment. The tool then provides the four design points, i.e. the dimensions of four helicopters to be built and flown. After collecting the flight data, the student enters it in the table and pushes the **Local**

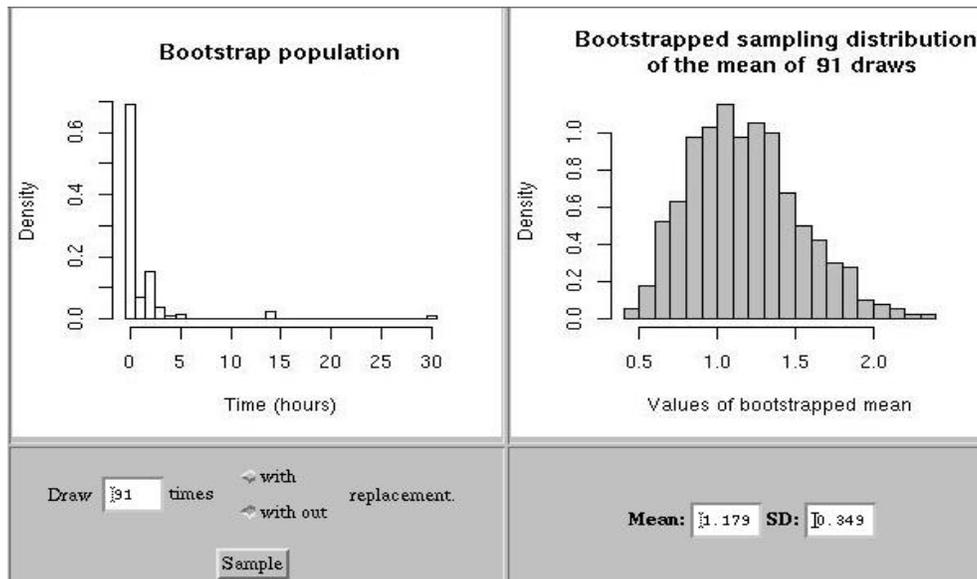


Figure 3: The Central Limit Theorem is demonstrated in this activity. The student enters the parameters (sample size and with/without replacement) to determine the sampling procedure for the bootstrap population. The histogram on the right is the bootstrapped distribution of the sample average.

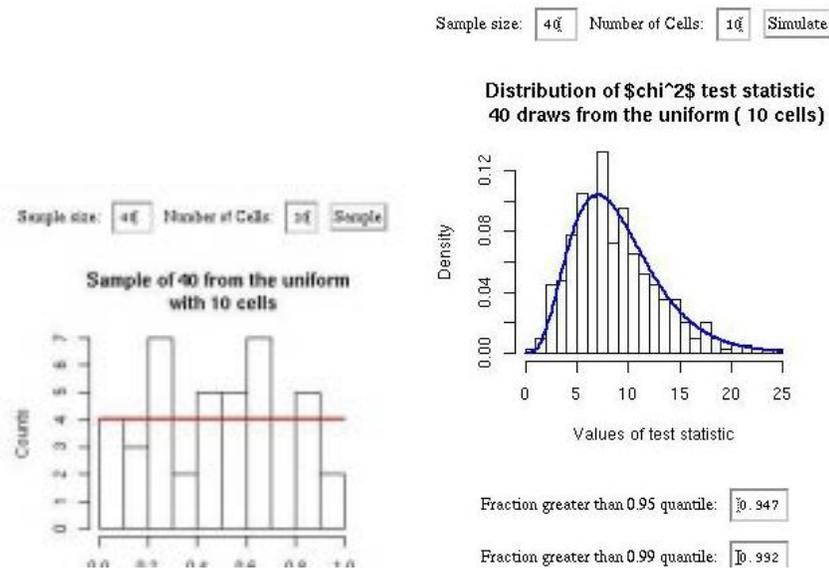


Figure 4: In this interactive browser window, the student performs a sensitivity analysis of the χ^2 goodness-of-fit test to small expected cell counts. Data are generated from the uniform distribution and the goodness-of-fit of the uniform to the data is tested. The student controls the sample size and the number of cells, which effectively controls the expected cell counts.

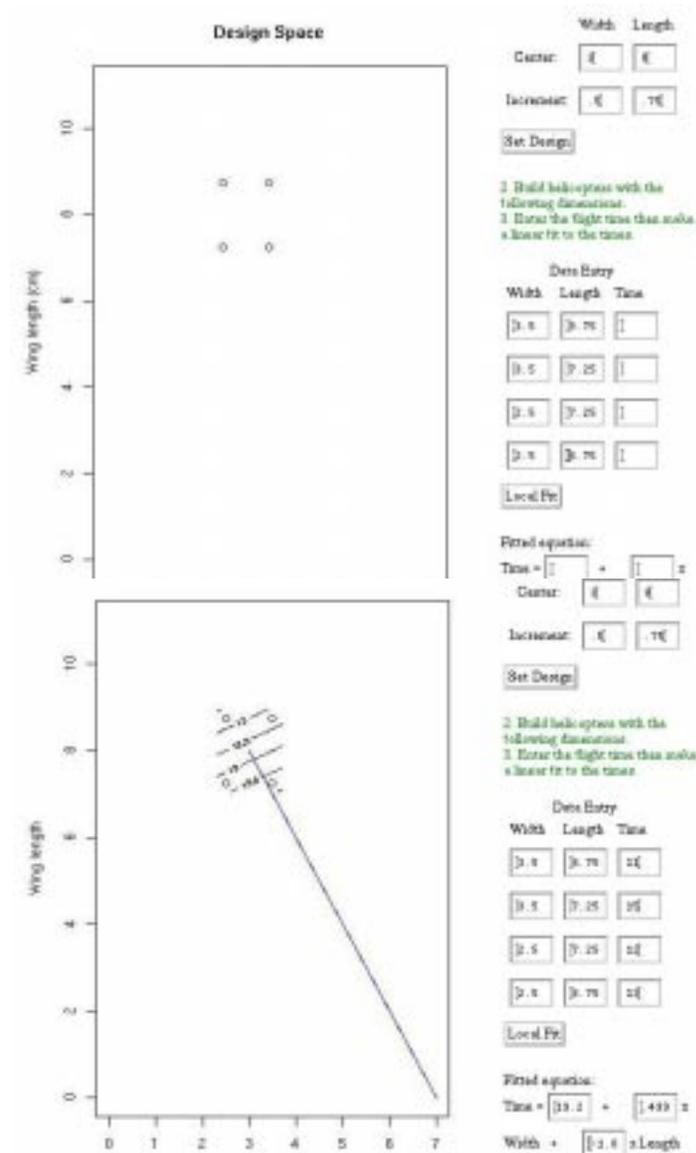


Figure 5: A tool for searching for the maximum of a response surface using the path of steepest ascent. Here measurements are taken at four design points (top). Then a local linear fit is made to the responses (bottom) and the path of steepest ascent is added to the design space.

Fit button. The tool fits a local linear approximation to the response surface, and plots the contours of the approximation and the path of steepest ascent on the design space. The student continues to search the design space by choosing a point along this path, and iterating this process. This tool can be easily adapted to search a known (to the instructor) response surface. For example, it could simulate the flight times at the design points, and allow the student a limited number of local fits before requiring a guess as to the location of the maximum. Then the true location and contours of the response surface could be revealed.

Discussion

For the instructor, there are many advantages to this approach to bringing the computer into the classroom. Most importantly, the instructor can easily extend the documents presented here to meet the specific needs of a class. The documents are formatted using HTML and forms for the radio buttons and text fields, and can easily be modified or included in a larger document. With the `EMBED` tag, we have R's well-tested and widely used code available in the browser. We use R for generating pseudo-random numbers, least squares fitting, and computing test statistics. The instructor can take advantage of R's graphical controls to easily customize the plots and charts.

For example, the bootstrap/CLT example can be modified to allow students to enter their own data, or to simulate from a probability distribution rather than the bootstrap population. The instructor can also add features to explore the sampling distribution of other summary statistics (e.g. median, trimmed mean, quantile). It is also simple to change the HTML that controls the layout and inputs and to include this document in a larger document. In contrast, an applet needs to have been designed to support such extensibility and its elements cannot be placed independently. Further, it is possible to "animate" the simulation, by building the sampling distribution little by little with the use of JavaScript timers.

Finally, with regard to student learning, these new computer activities help the student explore statistical concepts through visualization, play with methodologies to answer what-if questions, and observe the stochastic nature of various random processes. However, we want to emphasize that these activities do not replace the valuable activity of analyzing data using statistical software, which holds

an important place in our mathematical statistics course. Instead, these new activities bring an additional component to the course that helps students develop understanding for the concepts of statistics and connect these concepts to the practice of statistics.

References

- Buttrey, S., Nolan, D. and Temple Lang, D. (2001) An Environment for Creating Interactive Statistical Documents, to appear in *Proceedings of the Interface '01*.
- Nolan, D. and Speed, T.P. (2000), *Stat Labs: Mathematical Statistics through Applications*, New York, NY: Springer-Verlag.