Case Studies and Computing: Broadening the scope of statistical education

Deborah Nolan
University of California, Department of Statistics
367 Evans Hall
Berkeley, CA 94720-3860, USA
nolan@stat.berkeley.edu

Duncan Temple Lang
Bell Laboratories, Statistics and Data Mining Research
700 Mountain Avenue
Murry Hill, NJ 07974-2070, USA
duncan@research.bell-labs.com

INTRODUCTION

We have found many advantages to incorporating case studies in the advanced mathematical statistics course. And using a computer presents many opportunities, and also many challenges. For example, although the computer enables us to go far beyond the small, artificial examples found in traditional text books, many students become focused on the details of using the statistical software and the statistical concept becomes secondary. Most statistical software was not designed for pedagogical purposes, but instead for professional statisticians or data analysts. For the student, the value of gaining a skill, such as programming in SAS, may not offset the problems and is typically not the primary or initial goal of the instructor. To address these issues, our approach is to develop electronic activities that make it easier for students to quickly start to explore and develop understanding of the concepts of statistics using graphical user interfaces (GUIs) developed on top of existing statistical software (R). By building such GUIs on a flexible and sophisticated interactive programming environment, students with various levels of programming skills can easily move between the easy-to-use and directed GUIs and more free-form exploration using the language itself.

We have had success developing GUIs in the R language (`cran.r-project.org`) to analyze data from case studies and to perform simulation studies of statistical concepts (Buttrey et al., 2001). These GUIs are part of the StatDocs project, (`www.statdocs.org`). Our work builds on the infrastructure from the Omegahat project (`www.omegahat.org`), which provides bindings from R to the Gtk GUI toolkit via the RGtk and related packages. These enable us to program sophisticated, dynamic and interactive interfaces directly in R and to readily access existing and future statistical functionality developed for R. This approach puts the GUI design and development process within the reach of those instructors who know R, but not a programming language such as Java. Also, this approach allows statisticians to reasonably rapidly develop and experiment with convenient GUIs to well-tested code, rather than reimplement standard numeric and graphic facilities (potentially incorrectly). While S/R is a specialized language, it is appropriate for this domain and allows both instructors and students to freely move between GUIs and command-line interaction for different concepts and for students to "grow into" the environment.

This spring, we class tested several GUIs in an advanced undergraduate theoretical statistics course. Sixty students enrolled in this course, including graduate students in biostatistics, business, engineering and demography. The students used the GUIs to, among other things, study the central limit theorem, explore concepts related to regression, and analyze data from five case studies. In this paper, we describe some of these GUIs and provide student reactions

to them (58 of the 60 students completed anonymous course evaluations).

CASE STUDIES

Nolan and Speed (2000) have developed a course that teaches mathematical statistics through in-depth case studies. The approach integrates statistical theory and practice in a way not commonly found in mathematical statistics courses. Each case study centers around a scientific question; it contains a dataset to address the question, and we present statistical theory in order to answer this question. There are three salient aspects to this approach:

- The problem central to the case is introduced first, and background information on the problem and a description of data collected to address the problem are provided before any relevant statistical theory is discussed.
- The solution to the problem raised in the case study is not provided. In fact, there are many possible solutions which use many different types of analyses.
- The student plays the role of a consultant, analyst, government official, textbook author, etc. in developing and presenting the solution to the problem.

The merits of this approach have been discussed elsewhere (see Nolan & Speed, 1999, Nolan, 2002). Here we mention three comments from the student evaluations of the course that are particularly relevant to teaching biostatistics. Most students remarked on the motivation and relevance that case studies brings to the study of mathematical statistics. One noted that, "When put in a scenario it's easier to realize why we need to do certain tasks and the importance of interpretation of statistics." Students also found that teaching mathematics and cases together helped them make the connection between the theory and practice. As one student put it, "It's good to fit mathematical models to actual data. That way I can familiarize myself with the math notation, and what it means to say things like 'Let X_i represent ...' " They also enjoyed the diversity of the cases, as one student explains, "as an environmental engineer I was much more engaged during the radon lab while my biologist classmate was much more interested in the DNA lab. It is a good way to interest non-statistics majors."

COMPUTING

We have offered a case-studies based course for several years, but this is the first time that we have heard very few complaints about the difficulty in learning statistical software and the length of time it takes to analyze the cases. Yet, this spring, students conducted more data analyses and simulation studies than in past classes. We attribute this change to the incorporation of the GUIs into the course. In addition, although the GUIs were designed for ease of use, we maintained our past level of technical support for the students. That is, the teaching assistant met with students each week in groups of twenty to provide technical advice, and we maintained a Frequently Asked Questions web page for the assignments.

Altogether the students completed eight computer assignments. Four were intensive analyses of case studies which required 3-4 page write-ups of their findings. The other four were simulation studies and computer exercises designed to explore a particular statistical concept. We provide an example of each type of assignment.

In the search for unusual clusters of patterns in the DNA of a virus, the students were provided data marking the locations of particular types of patterns (complimentary palindromes) in the DNA. They were to determine whether the homogeneous Poisson process represented an adequate model for the distribution of the locations of these patterns, and if there were any unusual clusters of patterns. To accomplish this task, they conducted χ^2 goodness-of-fit tests of

the uniform and Poisson distributions. They also examined gamma-quantile plots, histograms and "sliding bin" plots with overlapping intervals (see Nolan & Speed, 2000). To assist them in their analyses, we provided a plotting GUI which allowed the students to specify input parameters for the plots, such as interval length and overlap between intervals. We did not provide a GUI for computing the χ^2 tests. Instead, they computed them in a spreadsheet, and included tables of observed and expected counts in their reports.

This example points out the significant difference between software designed for pedagogical purposes versus research purposes. The highly specialized plots, for which a researcher/analyst typically writes code, is provided to the students through an easy-to-use GUI. On the other hand, for the χ^2 test, a professional would perform it with a simple function call, but here the student does the computations "by hand" using basic spreadsheet operations. This means that the student spends time interpreting the plots and understanding how the parameter values effect the plot. He also must spend time figuring out how to compute a goodness-of-fit test from a set of numbers, which includes using a probability model to compute expected counts. The spreadsheet makes these computations intuitive, easy and non-repetitive. The goal of this inversion is to emphasize statistical understanding and thinking.

A simulation GUI helped students explore some of the rules of thumb and conditions for the Central Limit Theorem. In the GUI, the student constructs a population by specifying its values and counts. She then simulates from this population by specifying the sample size, sampling method, statistic to compute (mean, median, SD, min, max). One time the student works only with 0-1 populations to explore the normal approximation to the binomial distribution. Another time, she tinkers with population sizes to explore the effect of the sampling fraction (sample size/population size) in simple random sampling. The student also explores the sampling distribution of other statistics, such as the minimum and SD. For these simulation studies, we have the students provide brief written summaries of their findings along with normal quantile plots of the sampling distribution for various parameter settings.

The simulation GUI shows the power of building GUIs on top of existing statistical software. It is quite easy for the instructor to modify or add features to it. For example, the instructor can use R functions to tailor the simulation study to a particular topic of interest, e.g. she could add the trimmed mean to the list of possible statistics to examine, include a test for normality of the sampling distribution, or overlay a line on the quantile plot. While we have described applications for introductory mathematical statistics, the interactive case-study approach works well for other levels including introductory statistics and advanced graduate classes either as exercises or guided tutorials.

CONCLUSIONS

Overall, the students were very appreciative of the computing environment. They commented on how easy it was to use the GUIs, "The difficulty should come from solving the lab problem rather than struggling with the computer," how effective the visualization tools were for learning, "Graphical displays are a powerful tool in teaching. It allows students to visualize difficult concepts by connecting theory to easier to analyze visual displays," and how helpful the simulation studies were for gaining comprehension of complex statistical concepts, "Simulations let you see things that can't be presented by books. It is hard to imagine how the CLT actually works. Many people read the theorem and they go 'what the heck!' By simulating sampling, we could see that the CLT actually holds."

The students also made several suggestions for course improvements. We present here their general suggestions rather than particular ones on how specific GUIs could be improved. We found them insightful and plan to incorporate them in to our next round of development.

- A few students regretted not having the opportunity to learn how to use the R language to

analyze data. Our future GUIs will incorporate a command-line interface to give students the opportunity to use the R language in addition to the GUI controls.

- A few students said they preferred to use Excel because they were already familiar with it. To this end, We plan to exploit existing facilities from the Omegahat project to connect R to both Gnumeric and Excel spreadsheet applications. We also plan to dedicate a lecture pointing out challenges with statistical computing in Excel.

- Some students requested more lab manuals and helpful hints for using the GUIs.

- The computer work was performed exclusively within sections and not during lectures. Some students thought "It would be nice to have a few lectures where you show us things on the computer. It could really add to the power of the lecture, plus excite the crowd!"

- A related comment focused on the need for tighter communication between instructor and TA, with that change the student said "I think this course would be one of the best experiences in college." In addition to addressing this issue of inter-person communication, we are also in the process of creating guided tutorials in HTML that can incorporate dynamic feedback to the students along with the GUIs. To do this we are using the Gtk GUI toolkit via the Omegahat RGtkHTML package that provides bindings to the gtkHTML library. This package allows us to display HTML with embedded GUIs created in R and to control the actions and contents of the HTML using R. The dynamic content is specified by the instructor using XML and then processed in R via the SQuiz package from StatDocs.

REFERENCES

Buttrey, S., Nolan, D. and Temple Lang, D. (2001) Computing in the Mathematical Statistics Course, In *Proceedings of the Joint Statistical Meetings '01.*

Nolan, D. and Speed, T.P. (1999) Teaching statistics theory through applications, *The American Statistician*, **53**, 370–375,

Nolan, D. and Speed, T.P. (2000), *Stat Labs: Mathematical Statistics through Applications*, New York, NY: Springer-Verlag.

Nolan, D. (2002), Case Studies in the Mathematical Statistics Course, in the *Proceedings of the International Conference on Teaching Statistics VI.*

The Omegahat project, *http://www.omegahat.org.*

RÉSUMÉ

Nous avons développé un modèle pour l'enseignement des statistiques mathématiques à l'aide d'études de cas détaillées. Nous avons réussi à inclure dans cette approche des interfaces utilisateur graphiques pour l'analyse de données de ces cas et pour effectuer des études de simulation de concepts statistiques. Ce travail fait partie du projet StatDocs (`www.statdocs.org`*).*