Interview with Deborah Nolan

Allan Rossman
California Polytechnic State University

Deborah Nolan
University of California – Berkeley

Journal of Statistics Education Volume 23, Number 2 (2015),
www.amstat.org/publications/jse/v23n3/rossmanint.pdf

Interview with Deb Nolan

Deborah Nolan is Professor of Statistics and holds the Zaffaroni Family Chair in Undergraduate Education at the University of California – Berkeley, where she has also served as Associate Dean of Mathematical and Physical Sciences. She is a Fellow of the American Statistical Association and the Institute of Mathematical Statistics.

This interview took place via email on April 1 – November 3, 2015.

Beginnings

AR: Thanks very much, Deb, for agreeing to be interviewed for the Journal of Statistics Education*. Let's start not at the very beginning but when you were 18 years old. Where were you, and what were your career plans and aspirations at that point?*

DN: When I was 18, I was in my freshman year at Vassar College. It was an exciting time to be at Vassar because men were first admitted four years previously and they were going to graduate that year. There was a lot of buzz in the news and on campus about it. There was a strong sense on campus of the importance of women's education, and people wondered how this might change now that Vassar was co-ed. I found it exciting to be a student in this environment.

That year I was eager to try out a lot of new subjects, and I enrolled in courses in economics, philosophy, psychology, and, of course, mathematics. I really enjoyed studying philosophy and nearly majored in it. Math seemed more like a back-up plan for me – it was a subject that I enjoyed and always found a place for in my schedule, but I didn't think math would be my major. At that time, there was only one statistics course offered on campus. It was offered by the math department for students majoring in psychology and didn't count toward the math major. After my brief experience with psychology, I figured it was not the course for me. I loved pure math, especially analysis. I was exposed to statistics while I was an undergrad, but that experience convinced me that statistics was not for me!

AR: There must be a plot twist coming in this story where you learn that statistics is, in fact, for you. Did this revelation come soon after that or not for a while?

DN: Indeed, it was a New Year's resolution four years after I graduated from college that brought me to statistics in a serious way. After graduating from college, I worked for IBM as an applications programmer. My responsibilities included assisting a market research team with their customer data. I learned SAS and helped massage the data into a form the market research cats (that's what we called them) could analyze. Unfortunately, they thought you could simply hand the data over to the computer and it would do everything, including figuring out what type of analysis was appropriate to carry out. This was the same issue that had cropped up in college when I spent one summer at Vassar working for Caroline Bird. She was writing a book called *The Two-Paycheck Marriage* (1979), which was about how the American family was changing now that more married women worked outside the home. Part of her material for the book was to come from a questionnaire that was published in a women's magazine. Thousands of readers filled out the surveys and mailed them in, and my job that summer was to analyze their responses. At the start of the summer, one of my math professors had handed me a copy of a statistics textbook and that was the extent of my guidance. Back then, I realized that I could figure out how to carry out *t*-tests and *chi*-square tests, but I wasn't sure if what I was doing really made sense. It all seemed so final to apply a test and make a decision based on the results, yet not really know if the test was actually answering the question Ms. Bird had posed. Now several years later, I faced the same issues while working at IBM, but this time I appreciated the importance of the field of statistics and took up the challenge of learning more so that I would be a better practitioner. I started taking night classes at Columbia, and after a year of studying nights and weekends, I decided to go full-time to school. The decision happened on New Year's Eve, when my boyfriend came down with the flu and our plans for a romantic celebration were nixed. That night, I had a lot of quiet time and it gave me the opportunity for reflection. I realized that although I had learned a lot in my night courses, I wanted to learn the subject more deeply, and the best way to do that would be to go to graduate school full-time. I'm not the type to make New Year's resolutions, but I did that night and am glad I did. After that, I contacted my old college professors for advice on where to apply and called around to several statistics departments. Despite having missed the deadlines, several schools were willing to take my application. I started graduate school at Yale the next fall.

AR: Wow, that's quite a dramatic and impactful New Year's resolution. I'm glad that your boyfriend came down with the flu. Please tell us about your studies at Yale – was the program quite theoretical or also applied? What did you focus on for your research there?

DN: I really enjoyed my studies at Yale. Given its size, the program had a nice balance of theory and application. When I was there, the senior faculty were Frank Anscombe, John Hartigan, David Pollard, and Richard Savage, and the junior faculty included Kai Yu, Stephan Morgenthaler, and Ross Ihaka. I remember Frank Anscombe had just published a book called *Computing in Statistical Science Through APL* (1981). APL was the first programming language that I had learned at Vassar. It's a neat language that is fun to write code in, and I enjoyed learning from Frank about the computational issues behind statistical methods and graphics. On the other hand, courses with David Pollard invariably involved drawings with colored chalk that gave the intuition behind some theoretical proof; with Richard Savage we would delve into an applied research article and related citations; and John Hartigan's courses were peppered with rules of thumb, general advice on statistical methods, and funny sketches. However, I also learned a great deal from interacting with the faculty outside of the classroom.

The lunch hour was a very fruitful "classtime." Every day, the professors rustled up the students for lunch, and our conversations touched on all sorts of bizarre topics and a fair amount of statistics. Sometimes David and John would puzzle out a proof or a property of a new estimator they were working on or reading about; at other times, we considered various betting schemes that could out-perform confidence intervals. One summer day, John was poking fun at the density estimation fad. In his typical irreverent style, he proposed that we develop our own density estimator over lunch. On the walk to the cafeteria, he dreamed up a density estimator based on convex hulls of subsets of data, and by the time we got back to the department, he (with help from David and lots of questions from the students) had sketched out a proof of its consistency. I'm a shy person and the inclusive community that the faculty created was invaluable to me. I soaked it up and learned a tremendous amount.

Also, we were required to carry out practical work for a semester with a researcher outside the department and under the supervision of a statistics professor. I collaborated with a professor who studied snail fossils. The leading theory was that a snail grew according to a logarithmic spiral, i.e., the curve traced by the shell's spiral followed the relationship $\log(r) = a + b\theta$ in polar coordinates. However, a newly published paper suggested that the growth might be quadratic in θ. My client wanted to examine the growth pattern for a different species of snail. I helped measure several fossils, and when we plotted the data we found some loopy curvature. We figured out that the curvature appeared to be related to a misplaced center of the spiral. The logarithmic spiral was an idealization of how a snail grew, and one difficulty was pinpointing the center of the spiral. We had an unusual case of bias in the measurement process. We came up with the idea of randomly choosing several potential centers and taking a set of measurements for each center. Indeed, after accounting for this problem, the logarithmic spiral was the best fit. That's probably a lot more than you wanted to hear about the applied side of the program!

As for my research, it was quite theoretical. I worked on central limit theory for random functions. You can think of a density estimate as a random function. For example, a kernel density estimate spreads the mass of the observations smoothly over small regions neighboring each point to create a density curve. My research examined the large sample behavior of the kernel density estimate. I found that when properly normalized these random functions behave in a similar way to averages of random variables. That is, the density estimator converges asymptotically to a Gaussian process. I had strong training in analysis at Vassar and for my thesis I gravitated to theory, but from my work experience, I also had an interest in applications. It was easy to explore both sides of statistics at Yale.

AR: Had your interest in teaching already emerged when you were at Yale? Did you gain some teaching experience there?

DN: Most of my teaching appointments at Yale involved homework and exam grading. My first time in front of a classroom was in my third year when my advisor, David Pollard, attended a conference in Oberwolfach. I taught undergraduate probability for him for a week and had use of his office. He made a nameplate for me and put it on his desk. It said "Frau Professor Doktor Nolan," the typical German honorific for someone with a Ph.D. I found out recently that until a few years ago, Americans in Germany could be fined or jailed for calling themselves Doktor if their degree was not from Germany! Anyway, after that week of teaching, I taught the same probability course during the summer. My only other teaching experience was with introductory statistics, where I led a discussion section. We used the text, *Statistics*, by Freedman, Pisani, and Purves (1977), and I learned a lot from reading that book and working the exercises. I highly recommend it. The focus is on concepts, and there are few, maybe no, formulas in the book. Although I enjoyed these few teaching experiences, my keen interest in teaching didn't develop until several years after coming to Berkeley.

AR: Did you go start your career at Berkeley immediately after finishing at Yale? Were you committed to a career in academia at that point, or were you also considering other career paths?

DN: The entire time I was at Yale, I was officially on leave from IBM. I continued to work there for a few summers during graduate school and I expected to return full-time when I graduated. One of those summers I spent working for Lilian Wu at IBM's Watson Research Center. I enjoyed it very much and briefly considered working there when I graduated. It wasn't until my last year in graduate school that I thought about going into academia. John Hartigan advised that it was easier to move from academia to industry than the other way around, so if I was at all interested in academia, now was the time to try it out. I was intrigued by academia. I already knew what it was like to work in industry and thought I might really enjoy choosing my own problems to work on, teaching, and having a flexible schedule. I decided to give academia a try. After graduating from Yale, I started at Berkeley as an Assistant Professor and have been here ever since.

Early Career at Berkeley

AR: Very soon now I'm going to ask how your interest in teaching and education developed at Berkeley, but first I want to ask what Berkeley was like then. I assume that it was already one of the top statistics departments in the country then. Who were the most influential faculty, and what kinds of problems were they working on? On a slightly more personal level, how did the renowned faculty members treat a new hire like yourself?

DN: Yes, Berkeley was a top department back then. At Yale, my first statistics course used the text by Peter Bickel and Kjell Doksum (1977); my advisor and I had read the new book on CART by Leo Breiman, Jerry Friedman, Richard Olshen and Chuck Stone (1984); two of the leading figures in my area of research were Lucien LeCam and Chuck Stone; and as I mentioned earlier, I taught from the influential introductory text by David Freedman, Robert Pisani, and Roger Purves. All of these statisticians influenced my statistical training, and with the exceptions of Friedman, Olshen, and Pisani, they were all at Berkeley, along with several legendary figures like David Blackwell, Erich Lehmann, and Elizabeth Scott. Wandering the hallways and reading the nameplates on the doors felt like I had moved to the Statistics Hall of Fame!

Breiman was continuing his work on topics related to CART, Freedman had just published "From Mouse-to-Man" (Freedman and Ziesel 1988), which was about the extrapolation of results from animal experiments to cancer risks in humans, Doksum's focus was in survival analysis, Stone was working on splines and nonparametric regression, and as usual Bickel was contributing to many research areas, including robust statistics, bootstrap asymptotics, and semi-parametrics. Also here was David Brillinger who was doing applied work in time-series and point processes. Terry Speed arrived the next year as a senior appointment; he was starting in a new field - statistical genetics and genomics. Additionally, the statistics department at Berkeley has long had a strong probability group. When I arrived, Jim Pitman and David Aldous were the leaders. Jim was working on random permutations with Persi Diaconis at Stanford, and they were leading a weekly seminar on the topic at Berkeley. That's not everyone who was here when I arrived in 1986, but it gives you a sense of what it was like.

A few years after I arrived, I learned that I was the first woman hired in the department to a ladder rank faculty position since Betty Scott was hired in 1951. Betty was well known on campus for her fierce support for women. Her famous salary equity study had led to the adjustment of the salaries of women faculty on campus and had become a national model. I remember one time I called the campus admissions office to see if I could get hold of some data because the admissions process was under public scrutiny and one of my students had asked if we could discuss the topic in class. The call started out OK, but when they heard I was from the statistics department, they asked me if Betty Scott had put me up to making the call. There was no convincing them otherwise, and they wouldn't give me anything! There was a kind and caring side to Betty too. She looked after all of the statistics students. She brought cakes every Saturday afternoon for all the students (and faculty) who were working in the department. I was sorry that she retired a year after I arrived and passed away the following year.

For me, the transition from being a student to an assistant professor was not easy. Thinking back on it now, maybe it was in part because I had envisioned returning to industry after graduate school. I didn't start growing into the role of an academic until late in my studies. My colleagues at Berkeley were always supportive of me and my career, but being the only woman in the department was difficult. The connections that colleagues make over the tennis court and soccer field didn't happen for me. I felt this so keenly that I wrote an article on the topic, called "Women in Academe: Mentors Matter" (Nolan 1990). That's why Dimitris Politis and I started the IMS New Researchers Meeting in Statistics and Probability in 1993.

*AR: I see online (*http://depts.washington.edu/imsnrc17/about.html*) that this conference has continued every year with many participants per year and well-known presenters. Let me transition from research to teaching. What were your initial teaching assignments at Berkeley, and how did you approach them?*

DN: My first teaching assignments included introductory statistics for students with a calculus background and the two-course probability/statistics sequence for the undergraduate major. My approach to teaching back then was to work through many examples on the blackboard. These examples were similar to or the same as those found in the assigned textbooks.

AR: I suspect that your teaching approach has evolved since then, but I also suspect that high-quality of teaching undergraduates was not a top priority in how your Berkeley colleagues evaluated your case for tenure and promotion. Am I right that there was little or no incentive for you to devote much attention to teaching undergraduates at the time? If so, what were some of the early influences that led you to care about teaching so much?

DN: I received mixed messages about teaching when I first arrived at Berkeley. The chair of the department took me to lunch and in the conversation let me know that it would be best if I spent as little time as possible on my teaching. After that I was quiet about my interests in teaching and limited my teaching-related conversations to asking for advice on books to use and sample exams. Contrary to this advice, many of the faculty had written undergraduate textbooks, which I take as evidence that they cared about teaching. These texts included Blackwell's *Basic Statistics* (1969), Breiman's *Probability and Stochastic Processes: With a View toward Applications* (1969), Hodges' *Stat Lab: An Empirical Introduction to Statistics* (1975), Pitman's *Probability* (1993), and Stone's *Introduction to Statistical Theory* (Hoel, Port, and Stone 1971). And as I mentioned already, several colleagues had written graduate texts as well. At the campus level, the message was that excellence in all three areas of research, teaching, and service mattered. As an assistant professor it was hard to decipher what that really meant. I think it's much clearer now, and a variety of types of evidence of good teaching (more than student evaluations) are expected to be part of tenure and promotion cases.

Leon Henkin, a logician in the mathematics department, was an early influence. He had a huge impact on my undergraduate teaching. I clearly remember the day he stopped by my office to ask me if I would be interested in running a small seminar on probability for the Summer Math Institute for women at Mills College. He told me that there would be 12 undergraduate women and one graduate student assistant in this 6-week seminar and that the goal was to expose the students to the research process and teach them material not typically encountered in a classroom. That was in 1992, six years after I started teaching at Berkeley. It sounded interesting, and I jumped at the chance. I put together a few introductory lectures on discrete probability and then supplied students with sets of definitions and problems on advanced topics like random permutations. They worked in groups and kept research journals of their conjectures, proofs, etc. Each student also read her "own" research article, and she wrote a short introduction to the results, presented these to the rest of the students, and designed problems for the other students to work out. The atmosphere in the seminar that summer was thrilling and engaging, and I asked myself how I could bring this approach to teaching into my regular classrooms at Berkeley.

That was the beginning. The floodgates had opened and I started to put tremendous energy into new approaches to teaching and learning. My discussions with Terry Speed about the undergraduate theoretical statistics course strongly influenced my overhaul of that course and led to our publication of *Stat Labs: Mathematical Statistics through Applications* (Nolan and Speed 2001). Later, another big influence was an informal lunch-time seminar that Andrew Gelman and

I led with three undergrads, where they daily clipped articles involving statistics from the newspaper, tracked down the sources, and compared the news story to the source material. That experience led to our publication of *Teaching Statistics: A Bag of Tricks* (Gelman and Nolan 2002a). I should also mention Leo Breiman. He included me in an effort to develop video game-like exercises to teach introductory statistics, which helped rekindle my interests in computing and technology.

Book Projects

AR: That's terrific, not least because you've given me so many different directions to pursue follow-up questions. Let me take them in the order that you mentioned them, starting with Stat Labs*. Please give an overview of this project for those who don't know it, and describe how you used it in your teaching.*

DN: *Stat Labs* is a book of case studies for teaching statistics. It attempts to flip the traditional approach to teaching statistics on its head. Typically, we teach our students a method and related theory and then have them apply the method to a set of data that are prepared expressly for that purpose. The students miss out on how to connect the theory with answering the real scientific question. Instead, Terry and I organized each case starting with a question – a question from the domain of the scientific application that is not cast in statistical terms. We give the students data that have been collected to address the question and some investigatory suggestions to help them frame an attack on the problem. We also provide background material to put the problem in context. This information is gathered from a variety of sources and presented for a non-expert, college-level audience. The proposed investigations are written without using any statistical terminology, nor are they simply a list of statistical tasks for them to perform. For the students, an important part of their work is to connect their data analysis to answering the original question, which includes developing the theory and models appropriate for the problem. We also have them write up their results with a particular audience in mind, e.g., an article for a widely read magazine, a memo to the head of a research group, or a pamphlet for consumers.

These cases are developed from real problems, and we have even heard from researchers who have published papers on their solutions to our proposed investigations. To create a case study, we did a lot of prep work. We carried out background reading on the subject area, including reading about why the question is important and what others have contributed to addressing the problem. We met with scientists who own the data and asked them about the problem, data, and their work. We analyzed the data in a variety of ways and tried to piece together an interesting story that undergraduates could conceivably contribute to. We had help from many undergraduates in this process. We organized undergraduate research groups, where students worked with us on all of these tasks. Each case took several months to develop. We abandoned many along the way in our search for the right balance of an interesting scientific problem and accessible open-ended analyses.

In my courses, I typically cover about eight cases in a semester. Half of them are primarily discussed in class, and students work on the others outside of class, often in pairs. For all eight of the cases, I dedicate class time to a discussion of the problem and try to motivate the statistical theory from these discussions. I take a few different approaches to foster discussion of a case study in class. Sometimes I supply an abbreviated list of investigations, and ask the students to develop a plan of attack for addressing these suggestions. They are asked to think of ways to work with the data, but they do not perform any actual data analysis in class. With this approach, I bring to class results from several possible analyses of the data that have been prepared in advance in anticipation of their ideas. We have a whole-class discussion on the ideas that the groups came up with, then I provide my prepared results, and we talk about implications of what we find. For another approach, I provide students with a set of graphs and output from statistical methods and ask them to work in groups to summarize and interpret the output with the goal of addressing the questions from the investigation. The students seem to enjoy working on these problems in class because they receive immediate feedback on their ideas and they get to see the great variety of ideas their classmates come up with. At times I have invited experts to the class – people who have collected and/or analyzed the data. The students are expected to ask the experts questions about the context of the problem, the data, and the interpretation of the students' findings.

I try to teach all of my undergraduate courses this way. At Berkeley, we don't differentiate applied/methods classes from theoretical classes. All of our statistics courses have "discussion sections," where I like to have the students work with data on real problems.

AR: Can you describe an example or two of a case study that you've used in this way recently, perhaps one not in Stat Labs*?*

DN: Sure. One case looks back at the famous 2000 U.S. presidential election when Gore lost to Bush by 537 votes in Florida. If Gore had won Florida, he would have been elected president. As you know, there were voting abnormalities in Palm Beach County (PBC) related to the butterfly ballot. The butterfly ballot was thought to have confused voters, leading some to vote for Buchanan, the Reform candidate, when they meant to vote for Gore. We ask students to consider various types of evidence in their study of voting anomalies *and* to estimate the number of voters who mistakenly voted for Buchanan instead of Gore in PBC to see if the number of miscast votes was enough to have cost Gore the election.

We give the students three sets of data to work with, where each has a different level of granularity. There are county-level data for the entire U.S. linked with demographic information from the census; precinct-level data for Florida with election-day and absentee results split out; and ballot-level data for Palm Beach County. It's rare to have ballot data that shows how an individual voter casts their vote. The ballot data from PBC are very interesting because the absentee voters did not use the butterfly ballot. These data also contain results for the U.S. Senate race, and since we have individual ballots, we can see a voter's choice for both senator and president. The ballot data make for a kind of natural experiment: one group of voters used a butterfly ballot and another group did not; of course, these voters self-selected when they decided to vote absentee or on Election Day.

There are so many different types of analyses that students can do with these three sets of data. They can make simple comparisons of rates between subgroups. For example, they can compare absentee voters vs. election-day voters in Florida counties. In PBC, they can compare absentee voters who voted Democrat in the senate race vs. election-day voters who voted for the

Democrat in the senate race. Depending on the students' background, they can employ more advanced tools such as logistic regression. This case study is inspired by the paper by Wand, Shotts, Sekhon, Mebane, and Brady (2001).

For another case study, we consider the behavior of Web pages. Internet search engines, such as Google, Bing, and Ask, keep copies of Web pages so that when you make a query, they can quickly search their stored pages and return their findings to you. These saved pages are called a Web cache. Of course, if the page has changed since the last time it was stored, then the search engine serves stale pages. In order to keep the cache fresh, Web pages need to be visited regularly and the cache updated if a page has changed. The questions we consider are: How often do Web pages change, and how often should a site be visited to keep the cache fresh? This case study is based on articles by Cho and Garcia-Molina (2003), Grimes and O'Brien (2008), and Grimes and Ford (2008).

To help answer these questions, we study the behavior of 1,000 Web pages. Each of these pages is visited every hour for 30 days. The page is compared to the previous visit, and if it has changed, the cache is updated and time of the visit recorded. These data are censored in the sense that we do not observe the actual time a Web page changes but only the interval in which it changed; if the page changed on two or more occasions between visits, we only know that at least one change occurred.

We can treat the page changes as random events, and develop naïve estimates of the rate of change. We can even try modeling the change of a Web page with a Poisson arrival process, which can help address the problem with multiple changes between visits to the page. Questions about how well a model fits the data and how useful is a model are interesting issues to be considered.

AR: Those are very compelling examples, thanks. Let me ask about another project you mentioned earlier. How did your Teaching Statistics: A Bag of Tricks *book with Andrew Gelman emerge from the informal seminar that you described?*

DN: I don't exactly recall the sequence of events. In those days, Andrew and I talked frequently about teaching. We shared classroom activities that we had tried out; some were our inventions, and others were adapted from activities that we had read about or that others had described to us. At some point we thought that it might be helpful to statistics educators to collect them all in one place, so we decided to write the book.

Actually, we're working on a second edition of the book right now. We're adding several new chapters that reflect how our teaching has evolved in the years since we wrote the book. These new chapters are on graphics, teaching social science students, statistics diaries, a course in statistical communication, and a course in data science. I'm writing the graphics and data science chapters and Andrew is writing the other three. We also plan to add several new sections to the chapters from the first edition. Many of the biggest changes in the way I teach and in what I teach have been a direct result of advances in technology – both advances in technology to manage a course and technology for doing statistics. I want to add this perspective to the second edition. It looks like we're on schedule for a spring 2016 publication date. ☺

AR: I look forward to seeing the new edition, and I'm delighted to include such breaking news in a JSE *interview. As I did with* Stat Labs*, let me ask you to describe a couple of activities from the first edition of* Teaching Statistics: A Bag of Tricks*. But I want to point you in a particular direction this time. One of the all-time great article titles is "You Can Load a Die, But You Can't Bias a Coin" (Gelman and Nolan 2002b). Please tell us about that, and perhaps you could also describe another highlight from your "bag of tricks."*

DN: Funny you should ask about that particular activity. Someone contacted Andrew and me earlier this summer and proposed a bet. He would produce a coin, and we would train an undergraduate to flip the coin according to the description in our article, i.e., "flip the coin straight up, high in the air so it spins rapidly, and with the spinning axis also parallel to the floor; catch the coin midair in the palm of hand." The bet was that every time the coin landed tails, this guy would pay us $101, and if it landed heads, then we had to pay him $99. Plus, he wanted the coin flipped 40,000 times. That's a lot of money!

Now, according to our article, the coin obeys Newton's laws and spends half the time with heads facing up and half the time with heads facing down, so when it lands, the two sides are equally likely. Weighting the coin doesn't make a difference, because even a lopsided coin spins around an axis that passes through its center of gravity, and although the axis does not go through the geometrical center of the coin, there is no difference in the way the biased and symmetric coins spin about their axes. That's what we mean by "you can't bias a coin." However, you can bias the flip, and that's why there are the rules about the flipping. Also, bouncing and spinning the coin on a table are a different matter altogether. In our activity, we have students modify a checker with putty so that when it spins, it lands tails something like ¾ of the time. Then, they flip that same checker 100 times, and find that it lands heads about half the time.

Anyway, this proposition from a complete stranger seemed a bit crazy. In the end, Andrew declined the offer, but he published his exchange with the man on his blog at http://andrewgelman.com/2015/06/19/in-which-a-complete-stranger-offers-me-a-bet/. I highly recommend it. Andrew's final response is funny:

> One of these days in your travels a guy is going to show you a brand-new deck of cards, on which the seal is not yet broken. Then this guy is going to offer to bet you that he can make the jack of spades jump out of this brand-new deck of cards and squirt cider in your ear. But, son, you do not accept this bet. Because as sure as you stand there you're going to wind up with an ear full of cider.
> – Damon Runyon, "The Idyll of Miss Sarah Brown"

Coin-flipping activities are a lot of fun. One that I particularly enjoy is the real vs. fake flips activity. There we split the class into two groups and have one group flip a coin (according to the above instructions) 100 times and write down the sequence of heads and tails. The other group picks someone to make up the sequence of coin flips in their head and write the sequence down. They do this when I am out of the room. When I return, there are two sequences of 100 coin flips on the board, and my task is to identify the fake sequence. It's usually pretty easy to do because people think that a random sequence will not have too many heads or tails in a row, so their "random" sequence has too many switches back and forth and therefore runs of heads and tails that are too short. Hopefully, after this exercise, they are convinced that, e.g., random selection rather than judgment samples are better.

This activity has received some attention in the media. There's an episode of RadioLab on NPR called "Stochasticity" that includes the activity. We have some undergrads at Berkeley flip a coin 100 times, and some statistics grad students make up the coin flips. Even though these grad students are aware of the issues, I can still tell the real from the fake flips. Also, a British TV show called "Bang Goes the Theory" filmed an episode on chance and included a short bit with me picking the fake from the real flips made by people in a café in San Francisco. It was fun.

AR: That sure does sound like fun. Speaking of television appearances, do I remember correctly that the folks at MythBusters *consulted with you on one of their experiments?*

DN: Yes, actually they consulted with me on two of their experiments. The first one was about helium-filled footballs. They wanted to see if they would fly farther than footballs filled with air. Unfortunately, their measurement process was pretty sloppy, so it was not possible to do much with the data. For that experiment, I had a tour of the studio and appeared on the show. The second time, they contacted me when I was on my way out of town, so I consulted on the phone and by email. They wanted to test whether people could smell fear. I made recommendations for the design of experiment and analyzed the resulting data. I don't believe they found a significant effect.

Statistical Computing in Undergraduate Statistics

AR: Shifting gears a bit, I've been intending to ask you several questions about your views on statistical computing and data science. Your 2010 article with Duncan Temple Lang in The American Statistician *(Nolan and Temple Lang 2010) called for a broader definition of statistical computing and for greater inclusion of this topic in undergraduate statistics programs. Could you please summarize your argument here?*

DN: Duncan and I tried to make several points in this article. Our main objective was to encourage faculty and departments to rethink how they were preparing their undergraduates for the future, for both students entering the workforce immediately after graduating and those going on to a graduate program in statistics. The article touches on the why, what, and how; that is, why teach statistical computing, what to teach, and how to teach it. Actually, come to think of it, we also briefly touch on who teaches and when to teach statistical computing.

For the question of why we should teach computing in our courses, we argue in our paper that undergraduate statistics students need the essential technical skills that would enable them to engage in statistical problem solving – to wrestle with data, apply a modern statistical method, and present data and results graphically. If our students must wait for someone to carefully prepare the data before they can get started on the analysis, then they will lose out. Someone else will jump in and do it. Moreover, how we analyze data can impact computational considerations and these consideration can in turn impact the analysis; we want our students to understand these considerations and have the confidence to overcome computational challenges.

As for what we should teach, our paper attempts to broaden the field of statistical computing. Traditionally, statistical computing has focused on algorithms for statistical methods, such as random number generation and linear algebra. This seems a small, albeit important, part of statistical computing. We included many more data-related topics. A few worth mentioning here are information technologies, Web technologies, visualization, and what we called integrated development environments. Briefly, information technologies include working with data in all sorts of forms, like relational databases and unstructured text. We also put Web technologies under information technologies; these include Web scraping and Web services. As for development environments, we mean working outside of the walled-garden environment of a typical laptop. For example, if you needed to read 10,000 files, you wouldn't want to do this using a point-and-click interface. Or, if you are working on a project with several other people, then coordinating the various versions of the code is important. We include shell tools, version control, and text editors under this topic.

Now for the question of how to teach these topics. This is really important. I take issue with the "pick it up on your own" approach to learning computing skills. As a first exposure to programming, it's undesirable for several reasons. It can lead to bad habits, inefficient code, and basic misunderstandings. This approach can limit the way students think about problems and can even make some tasks seem impossible that aren't. Most importantly, I think that it can push the less-confident and less-prepared students away from programming and away from statistics.

In our article we raised three important aspects to teaching data science/statistical computing:

- Teach more than the nuts and bolts. That is, teach the paradigms in a language so that students can understand the fundamental concepts, communicate ideas through code, and reason about computational problems precisely.
- Teach how to learn about new technologies. Teach the skills needed to learn novel technologies, data formats, and programming languages as they are ever-changing, so it's important to know how to keep current.
- Teach in context. Like the case studies approach in *Stat Labs*, it can be a big leap from practicing basic programming skills to embracing problem-solving methodologies and general computing principles. Students, ideally, gain this experience as they behave like quantitative researchers, computing with data in the context of solving a scientific problem.

As for who: if we want to teach computing in context, as proposed, then statisticians should be teaching or co-teaching this material. And for when: if we use a language like R or MATLAB (as proposed in our article), then it's possible to teach data science in a second course, or even a first course.

AR: Do you or your colleagues teach a course like this as a first or second course at Berkeley? Please describe that course: Who takes it? How many students? What are the pre-reqs? What language do you use? What kinds of assignments? What works well, and what needs improving?

DN: Yes, we offer a course called Concepts in Computing with Data. Duncan and I designed the course almost 12 years ago in spring 2004. He had just arrived at UC Davis from Bell Labs. We were simultaneously teaching these courses at Davis and Berkeley.

At Berkeley, the course is one of three required courses for the major; these three are Concepts in Computing with Data, Concepts in Probability, and Concepts in Statistics. We have no prerequisites for Computing with Data, so students can take it as a first course. However, we recommend that students take the course as sophomores, not freshmen, because the projects are open-ended and challenging, and freshmen often aren't prepared for this kind of work. In effect, it's a second course, just not necessarily a second course in statistics. Also, although all statistics majors take this course, many other students take it as well. Concepts in Computing with Data has become quite popular; last year we offered two sections of 150 students each semester and a small class in summer session.

We use R when we teach the course for a few reasons. R is a complete, general-purpose programming language, where students can create new functionality and express statistical ideas and computations at a relatively high level. It also provides an environment where they can, for example, extract data from a relational database with SQL statements, clean and process text data with regular expressions, parse XML documents and locate content with XPath, and scrape data from the Web. Additionally, the faculty who teach other courses in the major typically expect their students to have some facility with R.

We have three types of assignments in the course. I already mentioned the open-ended projects. We typically assign two or three in a semester. A project addresses a modern data problem and includes several different computational challenges. We also have homework and weekly lab assignments. The labs are short. Students can typically finish them in the lab section, which is 2 hours. The labs aim to give guided practice in material that is currently being covered in class. The homework assignments fall between the lab work and projects in terms of directedness and difficulty. We sometimes use them to pace a project. For example, if cross-validation is part of a project, we ask students to write a function or functions to cross-validate for a homework assignment.

One project that I have used many times is related to presidential elections. Each time I use this assignment the data and formats are slightly different. For example, one year, we had census tables in CSV (comma-separated values) format, latitude and longitude data in geographic markup language, and election results in HTML tables. The students extract and merge all of these data sources and carry out an analysis. For the 2008 primary season, they riffed off a regression tree analysis in the New York Times that attempted to distinguish Clinton and Obama voters. They also make maps and other types of visualizations. In fact, Duncan and I just published a set of case studies that capture the flavor of these projects. The book is called *Data Science in R: A Case Studies Approach to Computational Reasoning and Problem Solving* (Nolan and Temple Lang 2015a).

I also enjoy visualization assignments. The topic of visualization works well in this course because students enjoy making beautiful plots. They typically haven't had any formal introduction to the subject, and whether they already know how to program or are novices they invariably are proud of their creations. I think it motivates the newbies to want to learn more about programming and other computational topics, and it shows the "experts" that they have something to learn too.

Something that needs work relates to the enormous growth that we have experienced in enrollments. In the past, I have had several students express their initial trepidation about learning coding but later said they overcame this fear during the course. Now that there are 150 students in the class, I think it's especially important not to lose those who are nervous about computing.

This issue also relates to my dedication to active learning in the classroom. The course is not a typical statistics course so I am slowly building a repertoire of activities, demos, worksheets, and the like to help students learn the concepts. Last semester, I stumbled on Bret Victor's Web essay titled "Learnable Programming: Designing a Programming System for Understanding Programs" (Victor 2012). He has a lot of nice ideas about how to teach programming. They are all in the context of designing systems to teach programming, but I was able to transfer a couple of these ideas into my classroom. I continue to be on the lookout for this kind of material. And, in the process, I am reminded how developing these activities helps me, as an instructor, identify and refine the important concepts.

AR: You mentioned in your article with Temple Lang that one of the goals of your Computing with Data course is to instill in students a habit of mind for computational thinking. How would you describe what computational thinking means, and can you provide an illustrative example?

DN: That's a tough question to answer concisely. A few ideas come to mind. Let's cast them in the context of working with and analyzing data. Suppose we are interested in data from the annual Cherry Blossom Ten Mile Run. Fifteen years' worth of race results are available on the Web at http://www.cherryblossom.org/. Danny Kaplan and I wrote about acquiring, cleaning, and analyzing these data (as one of the case studies in Nolan and Temple Lang 2015a), and this process covers several concepts in computational thinking:

- *Digital Information*: When we inspect the Web pages of race results, we find that they are not HTML tables, but simple plain text files. This discovery impacts how we extract the data. Even though they look roughly the same, an HTML table is different from fixed-width-formatted values, and these are different from a spreadsheet. We don't need to know a lot about the innards of a computer, but it is important to understand a bit (no pun intended) about how digital information is stored.

- *Work Flow*: When we visit the Cherry Blossom site, we find the men's and women's race results are in different files and that the naming conventions for the files vary from year to year. So, we decide to create a file that contains the addresses for all of the files. This way we have a record of our work, and if we need to go back to the site and access the data again, we can do this programmatically using our file of file names. When we work with data, we often want to retrace our steps and try a few different approaches to analyzing our data. This may involve redoing the preparation of our data. If we perform these steps programmatically, then it will be easy for us to redo them. On the other hand, if we, say, cut and paste or doing some computations “by hand”, then we may make mistakes when we try to recreate the process.

- *DRY Principle*: DRY stands for “don’t repeat yourself.” We begin writing code to clean and process the race results for the males for one year. Once we have cleaned one year’s worth of data, then we can repeat the process for other years and for the female runners. Rather than writing similar code and changing small bits for each year, we create a function that handles the task of processing a file, and we call this function repeatedly, passing it each file name.

- *Modularity*: We want to convert the run times into seconds to make our analysis easier so we write a small helper function to perform the conversion. As a separate function, it’s easier for us to test and debug our code. We can also more easily update this function in the future and use it for other purposes. When facing a computational problem, we want to be able to identify the various tasks and develop a set of modular functions for performing these tasks. Often this involves revisiting the problem a few times as we break larger tasks into smaller ones.

- *Exploring, Debugging, and Testing*: Our function to read the files and the helper functions to clean the data seem to process all fifteen years of data without a hitch. However, when we explore the data (using EDA) to confirm that our data are as expected, we find that all of the runners in 2003 were under 10 years old! When we write code, we want to develop a set of test cases to check our code continues to work properly as we extend and modify our code. And, as we uncover new errors, we add more tests to our set. Further, when we do find bugs in our code, we take a systematic approach to searching for and narrowing down the source of the problem.

- *Error Handling*: We all make mistakes when we code, and understanding how an error arises is part of computational thinking. If we understand programming principles and how the particular language that we are coding in works, then the errors occur less frequently and are not as indecipherable.

- *Efficiency*: Hand-in-hand with understanding errors and debugging goes the notion of efficiency. For example, we wrote our time conversion function so that it accepts a vector of run times, rather than an individual’s time. This will be much faster in R than calling the function repeatedly for each runner. If we understand the computational paradigm of a language, then we can write better, more efficient code.

- *Data Structures*: Some people ran in the race for many years, and if we can match their records across years, then we can perform a longitudinal analysis of sorts. Not everyone has run in the same number of races so we ask ourselves: how can we organize these multiple race times for the runners? To answer this question, we need to understand and compare different data structures, such as a data frame versus a list of data frames.

- *Abstraction*: We can think of a few different ways to match records across years, but how do we turn these ideas into code (and functions)? When we write code, we try to abstract the problem and anticipate its use in a more general way. This can be useful when we want to try a few slightly different approaches or, in the future, when we confront a similar task. Of course, there's a balance between abstraction and specificity.

- *Communicating Ideas through Code*: If we want to publish our findings, we may use mathematical notation to express a fitted model and assumptions. Code or pseudo-code can also help us express our ideas and methods in ways that are useful for us and for others.

These are some of the habits of mind for computational thinking that we would like our undergraduates to learn and practice.

AR: Thanks very much. That's very helpful and enlightening to think in terms of those habits of mind. Where does the idea of reproducible research fit in, if at all? Do you think this is an important topic for undergraduate students? Do you address this idea in your Computing with Data course?

DN: Reproducible research is a topic of great interest to me. It fits under the concept labeled Work Flow and also under Communicating Ideas through Code in my previous answer. Actually, what most people call reproducible research, I would call reproducible computations. These are the computations needed to reproduce the results that we publish. A popular format for reproducible documents is Markdown, and the R extension, RMarkdown. These documents incorporate the write-up of findings and the accompanying code for analyzing data, making plots, running simulations, etc. In the Computing with Data class, students prepare some of their homework and projects as RMarkdown documents. I have also used them recently in introductory classes and advanced electives. A similar framework in Python is Jupyter.

There are some exciting ideas and problems being worked on in this area. For example, Peng (2008) considers how to cache analyses. Caching can avoid having to rerun the entire analysis each time you update your document. If you have a large project with, say, lengthy simulations or data processing, then you most likely don't want to run all of the computations when a small change is made to the document. Instead, the document can be analyzed for dependencies between computations and only those that need updating rerun.

Duncan and I wrote our most recent two books (Nolan and Temple Lang 2013 and 2015a) in RDocBook, which is an extension of the XML vocabulary, DocBook, for writing technical reports (http://www.docbook.org/). XML documents have many nice features suitable for reproducible computations (and research). The documents have a hierarchical structure, so text and code are mixed and readily extracted. The content is separated from presentation so, for example, a title is marked as a title, not as text to be rendered in large bold font. This makes it easy to render the title in different styles. Another advantage is that the vocabulary can be easily extended, for example, code chunks for a plot, function, or sequence of expressions can be differentiated. Of course, with all of these extra capabilities comes the downside of verbosity.

What's the difference between reproducible computation and research? Reproducible research can include many additional pieces, not just the finished analysis. These might include tangential analyses that helped shape the final analysis, ideas that were tried out and discarded, mistakes that were made and kept as reminders, notes for the programmer about implementation, high-level summary for a non-technical audience, and tutorials for the student. The hierarchical structure of XML supports the notion of a document-database containing all of the analyst's work that can be rendered into different views, such as a technical appendix, interactive tutorial on a new statistics method, or detailed abstract. Duncan and I have written about the potential for these types of documents in education (Nolan and Temple Lang 2007). A graduate student of Duncan's developed a prototype that was implemented via an R plug-in to a Web browser (Becker 2014). Unfortunately, changes in browser security requirements have limited the development of this approach.

Undergraduate Statistics at Berkeley

AR: You mentioned earlier that Berkeley's undergraduate major in Statistics includes only three required courses. Please tell us a bit about the other two, on concepts in probability and in statistics.

DN: In terms of required courses, I wasn't counting prerequisites for the major. We do require four semesters of math (calculus and linear algebra) before you can declare the major. In addition to these math courses, the major consists of nine upper division courses. Three are the required "concept" courses, three are electives, and three are in an area of application. If you were wondering, yes, you don't need to take a first-year statistics course to be a major.

Now to your question. Concepts in Probability is calculus-based probability. We often use Jim Pitman's text *Probability* for the course. This is a probability course in its own right; meaning, the aim of the course is not solely to prepare students for Concepts in Statistics. Many economics, engineering, math, and other majors allow or require their students to take it as part of their major program. I should also mention that we have a group of about a half dozen probabilists on the statistics faculty, and our statistics major includes a track with a focus on probability.

Concepts in Statistics is our theoretical statistics course. The probability course is a prerequisite, and the computing with data course is strongly recommended. I regularly taught this course, and it was the impetus for developing *Stat Labs*. However, I have not taught either of these courses in a long time because my focus has been on Computing with Data and introductory courses and I was chair of my department for three years and after that associate dean for nine years.

AR: I apologize if you hear this comment a lot, but it strikes me that some readers might be thinking that these courses and this program sound great, but you and your colleagues have the advantage of working with top-notch Berkeley students. My question is: To what degree do you think these courses, especially the Computing with Data course, and your program requirements, can work well for institutions with less stellar students?

DN: I have heard this comment before, but not a lot. I would say that I don't believe the caliber of the students is what makes teaching with *Stat Labs* or *Concepts in Computing with Data* possible. I think what matters is the instructor's dedication to teaching statistics in context with authentic data. It's hard to do, compared to teaching using exercises from a standard textbook. This approach requires considerable preparation and there is less control over the direction students' questions take. From my experience, when an instructor puts in the effort, it can be very rewarding for the instructor and students.

When I first started teaching theoretical statistics from this perspective, math majors taking the course complained to me that this was supposed to be a mathematics course and what I was teaching was not what they expected or wanted. Fortunately, by the end of the course, they were converted and thought they had learned a lot. A similar thing happened with Computing with Data when it was first introduced. We owe it to our students to teach this way because they eventually need to have these kinds of problem solving skills and they're really hard to learn on one's own.

AR: The number of undergraduates majoring in Statistics at Berkeley has increased dramatically in the past decade. Moreover, data compiled by ASA indicate that your school has seen far more undergraduate statistics majors than any other program (http://www.amstat.org/misc/StatsBachelors2003-2013.pdf). To what do you attribute the growing popularity of statistics as an undergraduate major at Berkeley?

DN: Our enrollments have skyrocketed over the past ten years. We awarded 235 degrees in statistics during 2014-15, more than doubling the number in 2012 in the report that you reference. I think there are several reasons for this. We saw two periods of increase in our enrollments. The first was around 2006-2010, and my colleagues and I attribute that increase to the introduction of the Computing with Data course. Since then, we have seen an even sharper rise in the number of majors, which I would attribute to two reasons. I think one reason is that the recent popularity of data science in the media has created an interest in studying statistics. Another reason is that our student demographics have changed in the past seven years as we have admitted more students from out of state. In 2014, about 13% of Berkeley's undergraduates were international, compared to 3% in 2007, and statistics is the second most popular major for these international students (economics is the most popular) with 195 international students in our program.

Guidelines for Undergraduate Programs

AR: You were a member of the group that revised ASA's guidelines for undergraduate programs in statistical sciences (American Statistical Association 2014). The first of the "key points" described in the group's report refers to the increased importance of data science and the need for undergraduates studying statistics to learn about data science. As institutions and departments review and revise their undergraduate statistics programs, do you regard data science skills as one of the top issues to focus on?

DN: Well, I like all four of the key points: increased importance of data science, real applications, more diverse models and approaches, and ability to communicate. Guess that's not much of a surprise given, as you said, I was a member of the group revising the guidelines. I think the top issue facing departments relates to integrating computing into our courses and programs. This integration needs to come in a variety of forms, and data science is one aspect of it. I do think data science skills are very important, and they probably pose the greatest challenge to departments. For these two reasons (the importance and the challenge), I do regard data science skills as one of the top issues for departments to focus on. There are different models for providing our students with these skills. For example, at St Olaf College, the statistics and computer science faculty jointly designed an introductory course in Python and R that meets the needs of both departments. This course, Computer Science for Scientists and Mathematicians, is an introduction to computing concepts and data science skills. The program is described in more detail in Hardin, Hoerl, Horton, and Nolan (2015), along with six other data science courses, including Berkeley's.

Another way computing needs to be integrated into the statistics curriculum is through simulation-based inference (SBI). SBI is another example of how our discipline is shaped by computational resources. George Cobb speaks eloquently on this point in the inaugural issue of *TISE* (Cobb 2007). Allan, your work in this area has been very influential, including your group's blog and textbook, *Introduction to Statistical Investigations* (Tintle et al. 2016). Others have also contributed a lot in this area, e.g., the textbooks by Chihara and Hesterberg (2011) and the Lock family (2012).

One more topic related to computing that I think is not emphasized enough in our introductory and advanced courses is visualization. Cueing off Andrew Gelman's recent articles (Gelman 2011; Gelman and Unwin 2013), we can ask ourselves: If graphs are so great, why don't they have a more prominent place in the introductory textbook? I think it's because making good statistical graphs is hard to do; it often takes an iterative approach where we progressively improve our plot with a change of scale, transformation of a variable, addition of information, color, and reference markers, etc. We may even decide it's best to discard the plot and make a new one altogether. Each of these steps engages students in better understanding data and models, but this creative process is not easily encapsulated in a textbook so a formal treatment of graphics doesn't appear in our courses. However, with advances in computing, it's relatively easy to make a basic plot and engage students in activities to improve a plot so that the story/inference is more clearly and fully conveyed. I liken teaching visualization to teaching reading and composition, which is a vital aspect of undergraduate education on our campuses.

AR: To what degree do you consider ASA's undergraduate guidelines to be descriptive of what good programs are doing now, as opposed to aspirational for even top programs to aim for in the coming years?

DN: That's a very insightful question. We struggled a lot with questions like this. We began our work by reviewing the existing guidelines. These were written 15 years ago, but the committee thought they were very good and debated whether or not they needed updating. Nick Horton, the chair of our task force, did an excellent job leading us through this process. We organized webinars, held open forums for discussion at professional society meetings, and circulated drafts of the guidelines for comments. More to your question, in the end, I do think that the guidelines are aspirational for all programs. Some programs may have already put in place some of these recommendations, but I think there are a lot of good ideas in there for programs to think about and consider implementing. As one example, my department is currently examining how to bring simulation-based inference into our core theoretical statistics course and simulation into our core probability course.

Aside: At one point in the process, Nick asked me to think how an ideal statistics program would look, if given the opportunity to design it from scratch. He thought that might make a nice white paper to accompany the guidelines (several white papers were written by committee members to accompany the guidelines). I struggled with this and in the end did not write anything. One reason was that I wanted to step outside the framing of a program in terms of courses and rethink how we educate our students. Over the past ten to twenty years, our faculty has been shrinking and our enrollments have been increasing. Add to this the possibilities opening up with technology, and we could really change the university. One model that I have been thinking about would be to have a few really large courses where, with the aid of technology, students learn the basics. Along with these mega-courses, we could offer many small seminars where students work closely with faculty to learn how to think about and address problems in their field in a deeper, experiential way. It's like trying to take the best from each of two extremes! All pretty fanciful…

Preparing Future Teachers

AR: I believe that you're also involved with helping students to consider teaching as a career option. How do you advise and help Berkeley students who are intrigued by the prospects of a teaching career?

DN: I am one of two faculty directors of the Cal Teach program (http://calteach.berkeley.edu/), a relatively new program on campus. Ten years ago, the math, science, and engineering faculty decided to start a teacher-preparation program for undergraduates interested in becoming math and science teachers. The impetus came from a STEM teacher initiative of Governor Schwarzenegger and UC President Dynes. It wasn't clear at that time exactly what the program would look like. Berkeley has a graduate school of education with several programs that award a combined Masters degree and teaching credential, but there was no teacher-preparation program for undergraduates.

At first, Cal Teach offered only seminars for freshmen and sophomores. These seminars include a field placement in local elementary and middle school classrooms, where the students visit the classroom once a week and prepare and carry out an activity with the children twice during the semester. In the campus part of the seminar, the students develop and practice their activities, learn about inquiry-based approaches to teaching math and science, and debrief about their classroom experiences. After teaching one of these seminars, I was asked to become a faculty co-director of the program.

Cal Teach has evolved into an experimental credential program. We award math and science credentials for middle and high school teaching. The program is experimental because it does not follow the traditional model for awarding credentials in California. This means that we must establish the effectiveness of our approach in order to become a "regular" credential program. Cal Teach is modeled after the U Teach program at UT Austin.

There are several key elements of the program. One is that our students earn their full disciplinary degrees in math, science, or engineering and simultaneously earn a Cal Teach minor. We want our students to view themselves as mathematicians, scientists, or engineers who have chosen to teach. Relatedly, the minor includes a course in research methods. I developed this course with my co-faculty director, George Johnson, an engineer, and the staff director of the program, Elisa Stone, a biologist and former high school teacher. The program includes optional summer research opportunities as well.

Another key element of the program is that students have field placements in local classrooms throughout all four years of their undergraduate studies. These field placements are connected to seminars and courses in the minor. They begin in elementary school, like the seminar I described earlier, and progress to middle and high school placements. The activities the students develop and carry out in the classroom also increase in complexity; in the last placement, they lead a multi-lesson project.

Once the students complete the minor, they apply to our credential program and have a final semester of apprentice teaching before being awarded the credential. The program is small but growing. We awarded about 25 credentials this year and are on target to reach 50 credentials a year in the next few years.

I am very impressed with the Cal Teach students. They are bright, dedicated, and passionate, and they are bucking the trend at Berkeley by choosing to teach math and science. I really enjoy interacting with them.

AR: That does sound like an inspirational program. Do you think there are special challenges associated with teaching statistics at the K-12 level, as compared to teaching other math and science topics? Do you provide instruction or guidance specific to teaching statistics as part of the Cal Teach program? Do many statistics majors participate in this program?

DN: One big challenge is teachers who don't really know statistics. Cobb and Moore (1997) wrote about the ways in which statistical thinking and mathematical thinking are different. Others have also written on this topic, such as De Veaux and Velleman's article (2008) that compares math to music and statistics to literature. I think this issue is a much bigger problem at the K-12 level than at the college level.

Both the mathematics and statistics departments have tracks of coursework for students interested in becoming K-12 teachers. The mathematics department requires an introductory calculus-based statistics course in its track, and the statistics department requires the three courses in a related field to be in mathematics. Mathematics is the most popular major among Cal Teach students, with biology being the next most popular major. Unfortunately, we have seen only a few statistics students choosing to teach.

Additionally, the research methods course in the Cal Teach minor includes quite a bit of statistics. Each student carries out three experiments of their own design. They collect and analysis data from their experiments and present their findings in several formats, including a written report, oral presentation, and poster. For the third experiment, they typically redo an earlier one, where they have improved its design and analysis. The students also develop a survey instrument and collect and analyze survey responses. Also, in the Project-Based Instruction course, which is the last course in the minor before entering apprentice teaching, the students develop a multi-lesson, cross-disciplinary project. They work on these projects in pairs; typically a science major and math major are paired together. As part of the project the pupils in the 6-12 classroom collect and analyze data. Both of these courses are well aligned with the next generation science standards, particularly the practices dimension that focuses on “the behaviors that scientists engage in as they investigate and build models and theories about the natural world” (http://www.nextgenscience.org/three-dimensions).

Pop Quiz

AR: Now I’d like to begin what I call the “pop quiz” portion of this interview, where I’ll ask a series of short questions for which I request that you confine your answers to a few sentences. First, please tell us about your family.

DN: My husband, Dave, works for the Hotel Employees and Restaurant Employees Union. We have two children, Ben and Sam. Ben works for the City of Berkeley and is considering attending a data science bootcamp next year. Sam just graduated from college with a major in Greek and Roman studies. He has moved to Brooklyn with a couple of friends and just started working in e-commerce.

AR: What are some of your hobbies outside of statistics and education?

DN: I enjoy gardening (weeding, actually), bird watching, yoga, reading mysteries, and KenKen.

AR: What are some books that you’ve read recently?

DN: Vikram Chandra’s *Geek Sublime,* Katherine Boo’s *Behind the Beautiful Forevers*, Erik Larson’s *In the Garden of the Beasts,* Alice Munro’s *Dear Life,* Annie Barrows’ and Mary Ann Shaffer’s *The Guernsey Literary and Potato Peel Pie Society*, and Tim Cahill’s *Lost in My Own Backyard.* Actually, a bunch of these were audiobooks, but I consider that reading.

AR: What are some of your favorite travel destinations? Let me ask for at least one place that you went related to work, and at least one place that was just for fun.

DN: My favorite conference destination was my first *ICOTS* in 2002 in Cape Town. I also enjoy conferences in the Pacific Northwest, like Vancouver and Seattle. My family enjoys traveling to Europe. Italy is a favorite place, and we also have enjoyed visiting Greece and Switzerland. Closer to home, we love going for day hikes in Point Reyes National Seashore. Also, every spring, I spend a week on my own at Stinson Beach, California.

AR: Next I’ll ask some questions that I have used to collect data from students. Let’s start with some binary variables: Do you use a PC or Mac? Do you consider yourself an early bird or a night owl? Do you prefer window or aisle?

DN: Mac, NA! – I don't care much for early mornings or late nights, and aisle.

AR: And now a non-binary categorical variable: On what day of the week were you born? (You can use www.timeanddate.com to produce a calendar for your birth year.)

DN: Tuesday

AR: Next a discrete quantitative variable: How many Harry Potter *books have you read?*

DN: Seven – the first five aloud to my kids, and the last two silently to myself because Ben and Sam wanted to read them on their own.

AR: Here's a continuous quantitative variable: How many miles do you live today from where you were born? (You can use www.distancefromto.net to calculate this distance.)

DN: According to the Web site, 2839 miles.

AR: Here's a fanciful question that I have asked of students. Suppose that time travel were possible, and you could take one trip. You can only observe, not change anything, when you get there. Would you travel to a time in the past or in the future? What time would that be? What are the reasons for your choice?

DN: If I could travel to a time in the past, I would visit Athens and the Peloponnese in ancient times, around 350 BCE. When I have visited the sites of large, ancient, man-made structures, like Stonehenge, Newgrange, the Forum, and the Acropolis, I become transfixed and wander spellbound around the ruins! And yet, I have trouble imaging what these places must have been like back then. That's why I want to visit ancient Greece and also because so much was going on there at the time. I would want to attend the performance of a tragedy in an open-air theater, watch the Olympics, and take a few lessons in math from Aristotle. I would love to experience all of this firsthand.

AR: Here's another question that is completely hypothetical: Suppose that you are offered dinner for four to discuss statistics and/or education anywhere in the world. Who would you invite for your three dining companions, and where would you go?

DN: This is a tough one to figure out. I think that I would like to invite a couple of my mentors to dinner. It wasn't until years later that I realized the tremendous influence that they had on me and my career. Although they have passed, I would like to invite Winifred Asprey and Leon Henkin to dinner. Miss Asprey was a math professor at Vassar. She was responsible for starting the computer science program there. I learned APL (A Programming Language) from her. She was responsible for that first brush with statistics that I mentioned at the beginning of this interview; she recommended me for a summer internship with Caroline Bird to help analyze her survey data. Later, when she found out that I had no plans in place for after graduation, she told me, yes, told me, on a Friday afternoon, to spend the weekend putting together a resume and meet with the IBM representative who was coming to campus the following Monday. That was the one job I applied for. Thank goodness I got it!

I mentioned Leon Henkin earlier. He was a professor of Mathematics at Berkeley, and he asked me to teach one of the summer seminars in the Mills College Summer Math Institute. As I said already that was a major turning point for me in my teaching career. Also, I found it to be very exciting to be working on mathematics in an entirely female program. It came as a surprise to me that it would make any difference, but I felt lighter doing math in this environment.

As for the third person, I would invite Max Beberman. He headed the University of Illinois Committee on School Mathematics in the 1960s. I have only "met" him through the Illinois math textbooks for high school (Beberman and Vaughn 1965). It was through this math program that I discovered my love of mathematics. Unfortunately, my family moved in the summer before my senior year in high school, and my new school offered only the traditional calculus and trigonometry. I took them both, and they nearly squashed my interest in math.

This dinner would give me the chance to thank Miss Asprey, Leon, and the Illinois math guy. I could talk with them about today's students and brainstorm about how to encourage more women into the field of statistics, mathematics, and computer science. We would have dinner al fresco on a hill under an arbor on a farm in Tuscany or Sicily with a lovely view of the valley below. We would bring in a chef to prepare a five-course meal for us to enjoy together.

AR: That would be quite a dinner and conversation. I'm sorry that I can't provide it, but perhaps your mentors will read this interview as a substitute for enjoying that dinner. Now please tell us something about yourself that is likely to come as a surprise to JSE *readers.*

DN: Well, I built a log cabin from scratch with six friends. We are high school friends, and when we were graduating from college, we figured we would be going our separate ways and wanted a place where we could meet and keep our ties. So, we each chipped in $1000 and searched for a piece of land that we could buy and build a cabin on. We bought a piece of a farm in upstate New York, and over the course of a year, we built a log cabin. We chopped down the trees, scraped the bark off with drawknives, spiked the logs together, and so on. There's no electricity, no cellular service, and it takes a lot of pumping to get water from the well. It's a great place to unplug and relax.

AR: What has been your favorite course to teach?

DN: It's a close race between the Cal Teach research methods course for pre-service math and science teachers and Concepts in Computing with Data that we have spoken a lot about. I enjoyed teaching the research methods course for a bunch of reasons. I team-taught it with two colleagues, a biologist and an engineer. We had a lot of fun learning from each other as we developed the course. Since it is an interdisciplinary course there was a great sense of possibilities, and we put a lot of thought into how to shape the course. However, much of what I liked about teaching the course is not repeatable. Team teaching, designing the course from scratch, and the lack of confines of a discipline made teaching the course for the first time a lot of fun. It was a terrific experience. Except for the team teaching aspect, the other qualities are present in the Concepts in Computing with Data course. Duncan and I started from a blank slate and had fun collaborating to create the course, and it has continued to be very rewarding for me to teach. The students seem keen to learn the material and excited about their accomplishments, and it continues to be fun (and challenging) for me to develop new projects and assignments for the course. Like my students, I get a thrill from, say, scraping data from the Web, mashing together data from different sources, and creating a new analysis or visualization.

Conclusions

AR: Thanks very much, Deb, for taking the time to answer my questions so thoughtfully. I'll conclude with just three more questions. The themes for the past two USCOTS conferences have been "making change happen" and "making connections." Please select one of these, and then comment on how you've made this (change or connections) happen in your career.

DN: My choice is "making connections." Three examples come to mind: connections between statistical research and education, statistical practice and education, and computing and statistics. The Mills College Summer Math Institute and the Explorations in Statistics Research Workshop that I helped develop and run are examples of connecting research questions to the teaching of statistics. My book with Terry, *Stat Labs*, aims to connect real-world statistical problems with teaching, and the book with Duncan, *Data Science in R*, has similar goals in the data science arena. Developing the data science course exemplifies the connections between computing and statistics.

AR: Among all of your activities and accomplishments in statistics education, can you name one of which you are most proud?

DN: Hmmm, that's a bit like asking a parent which child is her favorite! I am proud of them all for different reasons ☺. Writing *Stat Labs* was a big deal for me because it departed significantly from traditional textbooks and teaching and was my first (major) effort at creating case studies. Planning and holding the Explorations in Statistics Research workshops has been a lot of fun (Nolan and Temple Lang 2015b). We have had such wonderful researchers participate in ESR, and the students (undergrads and grads) seem to really enjoy themselves. And, I am very excited about my current work in data science education because it feels as though we are at a major crossroads in both statistics and statistics education.

AR: Very good, I certainly agree that these are valuable contributions, with many more exciting and productive developments to come, I'm sure. My final question concerns advice that you would give to those at the beginning of their careers who are interested in statistics education. Let me make the question a bit more specific than that, though. Please imagine someone who is finishing their undergraduate degree and has become excited not only by the discipline of statistics but also by the prospects of teaching and other education-related work. What advice would you offer to this person?

DN: One piece of advice I would offer someone interested in statistics and the prospect of teaching is that statistics is a field in which it takes a long time to develop expertise, and to be a good teacher they need to work at developing this expertise. I would encourage them to be life-long learners in the field, to try their hand at data analysis, and learn how experts approach statistical problems.

References Cited in the Interview

American Statistical Association (2014), *Curriculum Guidelines for Undergraduate Programs in Statistical Science*, available at: www.amstat.org/education/curriculumguidelines.cfm.

Anscombe, F. (1981), *Computing in Statistical Science Through APL*, Springer.

Beberman, M. and Vaughn, H. (1965), *High School Mathematics, Course 2*. D.C. Heath and Company.

Becker, G. (2014), *Rethinking Dynamic Documents for Data Analytic Research*, Ph.D. thesis, University of California, Davis.

Bickel, P. and Doksum, K. (1977), *Mathematical Statistics: Basic Ideas and Selected Topics*, Holden-Day.

Bird, C. (1979), *The Two-Paycheck Marriage: How Women at Work are Changing Life in America*, Wade Publishers.

Blackwell, D. (1969), *Basic Statistics*, McGraw-Hill.

Breiman, L. (1969), *Probability and Stochastic Processes: With a View toward Applications*, Houghton-Mifflin.

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984), *Classification and Regression Trees*, Wadsworth.

Chihara, L. and Hesterberg, T. (2011), *Mathematical Statistics with Resampling and R*, John Wiley and Sons.

Cho, J. and Garcia-Molina, H. (2003), "Estimating frequency of change," ACM Transactions of Internet Technology, Vol. 3, No. 3, 256-290.

Cobb, G. (2007), “The Introductory Statistics Course: A Ptolemaic Curriculum?,” *Technology Innovations in Statistics Education*, 1(1). Available at: http://repositories.cdlib.org/uclastat/cts/tise/vol1/iss1/art1/

Cobb, G. and Moore, D. (1997), “Mathematics, Statistics, and Teaching,” *American Mathematical Monthly*, 104, 801-823.

De Veaux, R. and Velleman, P. (2008), “Math is Music; Statistics is Literature (Or, Why are There No Six-year-old Novelists?,” *Amstat News*, September 2008, 54-58.

Freedman, D., Pisani, R., Purves, R. (1977), *Statistics*, W.W. Norton and Company.

Freedman, D. and Zeisel, H. (1988), "From Mouse-to-Man: The Quantitative Assessment of Cancer Risks," *Statistical Science*, 3, 356.

Gelman, A. (2011), "Why Tables are Really Much Better than Graphs (with discussion)," *Journal of Computational and Graphical Statistics*, 20, 3-40.

Gelman, A. and Nolan, D. (2002a), *Teaching Statistics: A Bag of Tricks*, Oxford University Press.

Gelman, A. and Nolan, D. (2002b), "You Can Load a Die, But You Can't Bias a Coin." *The American Statistician*, 56, 308-311.

Gelman, A. and Unwin, A. (2013), "Infovis and Statistical Graphics: Different Goals, Different Looks (with discussion)," *Journal of Computational and Graphical Statistics*, 22, 2-28.

Grimes, C. and Ford, D. (2008), "Estimation of Web Page Change Rates," *Proceedings of the Joint Statistical Meetings*, American Statistical Association.

Grimes, C. and O'Brien, S. (2008), "Microscale Evolution of Web Pages," *Proceedings of the 17th International World Wide Web Conference*, 1149-1150.

Hardin, J., Hoerl, R., Horton, N., and Nolan, D. (2015) "Data Science in Statistics Curricula: Preparing Students to 'Think with Data,'" *The American Statistician*, 69, to appear.

Hodges, J. (1975), *Stat Lab: An Empirical Introduction to Statistics*, McGraw-Hill.

Hoel, P., Port, S. and Stone, C. (1971), *Introduction to Statistical Theory*. Houghton-Mifflin.

Lock, R., Lock, P., Lock Morgan, K., Lock, E., and Lock, D. (2012), *Statistics: Unlocking the Power of Data*, John Wiley and Sons.

Nolan, D. (1990), "Women in Academe: Mentors Matter," *Statistical Science*, 7, 267-272.

Nolan, D. and Speed, T. (2001), *Stat Labs: Mathematical Statistics through Applications*, Springer-Verlag.

Nolan, D. and Temple Lang, D. (2007), "Dynamic, Interactive Documents for Teaching Statistical Practice," *International Statistical Review*, 75, 295-321.

Nolan, D. and Temple Lang, D. (2010), "Computing in the Statistics Curricula," *The American Statistician*, 64, 97-107.

Nolan, D. and Temple Lang, D. (2013), *XML and Web Technologies for Data Sciences with* R, Springer-Verlag.

Nolan, D. and Temple Lang, D. (2015a), *Data Science in R: A Case Studies Approach to Computational Reasoning and Problem Solving*, CRC Press.

Nolan, D. and Temple Lang, D. (2015b), "Explorations in Statistics Research: An Approach to Expose Undergraduates to Authentic Data Analysis," *The American Statistician*, 69, to appear.

Peng, R. (2008), "Caching and Distributed Statistical Analysis in R," *Journal of Statistical Software*, Vol. 26. Available at: http://www.jstatsoft.org/article/view/v026i07.

Pitman, J. (1993), *Probability*, Springer-Verlag.

Tintle, N., Chance, B., Cobb, G., Rossman, A., Roy, S., Swanson, T., and VanderStoep, J. (2016), *Introduction to Statistical Investigations*, John Wiley and Sons.

Victor, B. (2012), "Learnable Programming: Designing a Programming System for Understanding Programs," available at: http://worrydream.com/LearnableProgramming/.

Wand, J., Shotts, K., Sekhon, J., Mebane, W., and Brady, H. (2001), "The Butterfly Did It: The Aberrant Vote for Buchanan in Palm Beach County, Florida," *American Political Science Review*, 95, 793-810.
