Scientific Computation and Preparedness for Undergraduate Research

As a Cognitive Neuroscientist, I teach a number of lab courses in which students design and conduct empirical studies, and analyze, interpret and display the data. Although I do not directly teach computation in my classes, I rely on it as a tool and expect students to have some familiarity with certain techniques before they take them. At St. Olaf, all Psychology majors are required to complete one statistics course and one research methods course prior to taking upper level classes, but student retention of the information is often limited. For many students, there is not sufficient opportunity to work with these tools outside of their initial exposure for them to really stick. Thus, knowing that the students have taken the prerequisite classes does not necessarily tell me much about their abilities in the lab, and I have observed that students have varying levels of comfort with scientific computation. One way that I try to balance student experience is by having students work in small groups, where at least 1 person in each group has a higher level of comfort with computation and analysis. However, this can create additional problems if the group expects the experienced student to do all of the analysis for them, which will also limit what the other students in the group learn from the project. It is also problematic if the group expects the experienced student to teach them how to do the analysis. Some experienced students have seen this as an opportunity, and spent a great deal of time to make sure that each group member understood what they were doing. Others find it faster and easier just to do it themselves. Thus, the outcomes for the less experienced group members have also varies across groups. One challenge that I encounter is to how best assess what skills students bring to the class and how to help get them up to speed. I routinely go through tutorials with the students (how to make graphs, basic data conditioning and analysis, etc.), but am always concerned that I am moving too slow for some students and too fast for others. Review is important to help them internalize the information, but getting the right balance has been tricky, and most review has been done in context or with the particular set of data that the groups are working with which can spread me thin during larger lab classes.

I also run a research laboratory in which we routinely use data collection, analysis, statistical analysis, data visualization and programming. As my program of research has matured, we have added (and built in some cases) new tools to help in our research. MATLAB has become indispensable to our work in the lab. However, few students that I recruit to work in the lab have any experience with the platform. It is not taught in any of our Computer Science courses, and unless students have actively sought it out on their own, few will have any experience with it. Many students are willing to learn the platform, and I have been happy to work with them in any capacity that I can. One technique that has provided some success is to utilize toolboxes and very specific tasks to promote student learning. Two summers ago, I co-supervised a summer research project utilizing mobile electroencephalography (EEG) to understand how people learn, create and perform dance. The two student researchers on the project had no experience with EEG or MATLAB, but were responsible for the data processing and analysis. I suggested they use the EEG Lab Toolbox for MATLAB as an analysis tool, which was met with a little trepidation since neither had worked with MATLAB before. Although an excellent toolbox, the students still need some facility with MATLAB to be able to use it to its full potential. I billed MATLAB as a time saver, and after having them struggle a bit with the analysis in Biopac and Excel, they were eager to try it. What worked here was to allow them to see how the toolbox could be used immediately for the task at hand rather than to try to teach them the platform from scratch, which would have been beyond the limits for a summer project.

In another summer research project, I challenged a student to build a laser pointing device to determine the ones perceptual localization of a sound in the free field. We started with an Arduino system (using the Sparkfun Inventors Kit), where turning a potentiometer knob would generate a voltage which would drive a servo, atop which we mounted a laser. This would allow a participant to indicate a spatial location of a sound source with visual feedback provided by the laser. Once the student had built the system and was comfortable working in Arduino, I asked them to figure out how them integrate the device into MATLAB using the Arduino toolbox. Once they could run the device in MATLAB, I asked them to capture the voltage data from the potentiometer so that they could convert it into an angular position. Once they had accomplished that, we integrated the device into an existing program that handled stimulus delivery. At the start of the project, the student had had zero experience with MATLAB. However, by the end he was much more confident in his abilities and eager to learn more. Again, starting small with a very specific task allowed us to build a more complex system as each previous step was understood.

Using an incremental approach has been extremely helpful when teaching students to work with MATLAB. Since I do not directly teach computation using MATLAB as one would in an introductory course, I find teaching in context to be very successful. Students who might balk at learning a new platform can immediately see how their efforts are going to pay off and are thus more likely to put in the necessary time. Along the way they learn more about the platform and language used, and are often comfortable enough with what they have learned that they want to learn more. Starting with simpler and less threatening platforms (the size and flexibility of MATLAB can often be daunting to new users) and building on easier projects can also help the novice student work up to the task. I have yet to implement this in a formal class, but I would imagine that this approach would yield similar results there.

The lack of experience and confidence in their computational abilities can be a major barrier for students when it comes to attempting to take upper level research classes or get involved in faculty research programs altogether. This can be particularly challenging for first generation college students as well as those from underprivileged backgrounds who may not have had the same opportunities as their peers. Undergraduate research has been repeatedly shown to be a high impact educational practice that produces discipline specific benefits (comfort with research design, data collection and analysis, communication), broader academic benefits (critical thinking skills, communication skills, writing skills) as well as personal benefits (self-confidence, independence, tenacity) (Lopato, 2010). However, if students do not feel competent in statistics and research methods (in Psychology at least), their likelihood of engaging in advanced research opportunities will likely be limited. Earlier exposure followed by consistent implementation and reinforcement across the curriculum may be one way to offset some of these issues, and is something that we are discussing across the STEM departments at St. Olaf.