Sean Kross

While earning my bachelor's degree in biology I worked in Dr. Patrick Eichenberger's laboratory during my senior year at New York University growing mutants of *Bacillus subtilis* under different stress conditions in order to measure mRNA expression. This was the first time in my life that I was putting the scientific principles I had been studying for years into practice, and it was exciting to be a part of a team trying to discover something new. I was in awe of our process: the fact that the experiments started with liters of *B. subtilis* growing in liquid media, and through a series of chemical reactions over the course of a week, all that remained was a tiny raindrop – maybe 10 microliters – of a solution of concentrated RNA.

I gained new appreciation cleanliness and carefulness when working at the lab bench. The tiny scars caused by a phenol-chloroform spill on the wrist of a graduate student I worked with served as a constant reminder of how thoughtful I had to be with every aliquot and measurement. Soon I was performing DNA microarray experiments, where weeks of labor by many of my colleagues hinged on setting the microarrays correctly. Overnight the necessary reactions would take place and the computational biologists in our lab would take over the analysis of the results.

Though I had taken courses in biostatistics that taught basic programming skills, it wasn't until I saw the software that was being built, and the results it was producing from our experiments, that I realized the true potential of reproducible biological data analysis. When I saw the data we collected brought to life by computational models I felt like I was getting a glimpse of the future of the field, a future I wanted to be a part of. I started taking apart the R code that was being used in the lab, but ultimately I felt limited in my knowledge of the algorithms being used. I decided that in order to be the biologist I wanted to be I would need to become a better computer scientist, so I decided to pursue a second bachelor’s degree in computer science at the University of Maryland.

After a year of coursework and a solid foundation in programming and algorithms I was introduced to Dr. Mihai Pop and he graciously allowed me to work in his lab. One of Dr. Pop’s main projects is to analyze metagenomic data from children in developing nations who are experiencing diarrhea. It is our hope that the cause of their diarrhea can be identified and potentially treated if their metagenome is compared to the gut microbes observed in their peers. Organisms in the human gut microbiome are identified by the ribosomal RNA that is incorporated into the structure of the ribosome’s 16S small subunit. There are several databases that map sequences of 16S rRNA to an organism’s taxonomy, however these databases are often in disagreement with each other, mapping the same RNA sequence to different organisms. While working in Dr. Pop’s lab I developed an R package called warppipe that highlights the differences in how the same RNA sequence may be assigned differently within a database or between multiple databases. The first release of warppipe is available on GitHub though I am still actively developing the package.

While at the University of Maryland I was introduced to Drs. Jeff Leek, Roger Peng, and Brain Caffo from the Johns Hopkins University School of Public Health. I had the privilege of helping them create the Data Science Specialization on Coursera.org, a series of 9 courses covering topics like R programming, exploratory data analysis, machine learning, and statistical inference. In the past year one million students have enrolled in at least one of the courses in the specialization. In addition to creating lecture materials for many of the classes, I designed and provisioned a Fedora-based operating system for students to use, so that they would have access to all of the software required for the class with minimum configuration.

One the challenges we faced when designing the courses was how to teach R programming effectively. Nick Carchedi (a biostatistics graduate student) and I developed Swirl, an R package that interactively teaches R programming within an authentic R development environment. Swirl presents a dialogue within the R console, instructing the user how to use R by reacting positively when a user types in the correct command, or giving hints if the user is struggling. Swirl can ask the user to answer multiple choice questions, numeric or text-based questions, or Swirl can instruct the user about how to write entire R scripts. Swirl comes packaged with several interactive courses, and Swirl includes tools so that anyone can author their own course. Unlike web browser based interactive programming instruction, Swirl's courses take place inside of the R console, so there is no separation between where the user is taught to use R and where the user can perform a real world analysis with R. Swirl has been downloaded over 100,000 times and we have users all over the world.

In the future I would like to continue developing computational tools for exploring RNA sequencing data and modeling interactions between organisms that share the same microbiome. I'm particularly interested in codifying reproducible research methods so that the largest audience possible can examine the analysis of experimental results, in order to make the analysis more robust and transparent. I plan on continuing to develop Swirl because I believe that tools like Swirl are a large part of the future of education. I hope that one day interactive lectures from great instructors will be available in every school and in every home.