Course Description

target audience

This course will be taught for entry-level graduate students in Statistics (MS and PhD), as well as motivated PhD students in the biological sciences. Students will benefit most from taking this course at the beginning of their graduate program, to use these tools for their research, during their time at UW-Madison and beyond. Only basic statistical knowledge will be assumed.

learning objectives

Acquire a strong basis for carrying out computations for graduate research or on the job. More specifically, learn to

- do computations efficiently: possibly handling very large data sets
- do computations automatically: using scripts to repeat tasks (and avoid manual errors)
- build and manage a computing project reproducibly: by oneself or by others
- build and manage a computing project collaboratively: share work with a version control system.
- demonstrate ability to adapt to change in computing platforms

These skills are essential to Statistics graduate students. They are also becoming essential to a growing number of biology PhD students who need to analyze very large data sets using command-line pipelines created by others, and to collaborate on these analyses with peer lab members.

Change is constant now. New computing tools appear constantly. Assets for successful adaptation include an understanding of computing concepts shared across platforms and programming languages, and a mastery of several programming languages. Students will build confidence in their adaptability, from the experience of learning a new language rapidly drawing on known concepts.

tools and topics

(subject to change)

- the shell: power of command lines to interact with a machine
- version control and collaboration: git and GitHub
- best practices for scientific computing
- scripting: Bash and Python primarily, basics of Julia

hardware requirements

Students will need to bring their own laptop, running either Mac or Linux.

This requirement reflects current practice in statistics and in computational biology. PhD students in biology experience this requirement first-hand when working with large molecular sequence data, to run available analysis pipelines. Computing servers in Statistics are all Linux servers. These servers are accessible remotely, however.

Students with a Windows machine have the following options, none of which are recommended nor supported:

- buy an Intel-based chromebook running Linux. They start around $180, so you can consider that this in lieu of a textbook. I started experimenting with a chromebook, see my notes here.
• connect to a remote Linux server in Statistics. An important downside is the lack of admin (super-user) permission, which may prevent the upgrade or installation of necessary software.

• if you have a 64-bit machine running Windows 10, you may try the “Windows Subsystem for Linux”, which offers the Bash shell. But all Linux features are not available, so this option is not recommended and will not be supported during the course.

• install bash and a set of Unix commands on your machine. This will likely involve some trial-error and frustration, but see here to install the bash shell with git, and Python with Anaconda. Also install Rtools, which will provide key utilities like GNU “make” and “gcc”.

• use cygwin or install a desktop virtual machine like Virtual Box VM, but these don’t work with git as well. Again, I will not support this option during the course.

assessments

Because one cannot master computational tools without practice or without trouble-shooting errors, the course will be primarily project-based. All the topics will be taught in the context of practical examples, and case studies in biology. The final project will be collaborative in groups of 3 or 4.

Each item will be graded with a point system. Points will be averaged with the following weights:

• [10%] participation: in lecture, in-class group work, response to project reviews
• [10%] peer reviews
• [40%] homework exercises
• [40%] final project

Final letter scores will be curved. Grades will be posted on canvas

Students may discuss homework problems and case studies with others including peers, the TA and instructor. But each student should write her/his own computer code and project documentation, and should obtain her/his computer output independently.

Homework exercises, projects and peer reviews will be submitted online, via GitHub. Late submissions will receive a maximum of 50% of the total number of points, if submitted less than a week after the deadline. Work submitted later than a week after the deadline will not receive any credit. Deadline extensions will be granted under extenuating circumstances. Vacation travel does not constitute extenuating circumstances.