ECEV 32000: Computing Skills for Biologists

Stefano Allesina

Winter 2019

E-mail: sallesina@uchicago.edu
Web: allesinalab.uchicago.edu
Office Hours: by request
Class Hours: Wed, 9:30am-12pm
Office: Zoology 403A
Class Room: CSIL 3

Course Description

The class is centered on building a strong computational toolbox for biologists. It encompass a variety of programming languages (Bash, Python, R) and tools (UNIX shell, Git, Regular Expressions, LaTeX, Databases). Each week, a different topic is discussed, a few exercises are solved together in class, and (much) longer exercises are assigned as homework. The homework is not graded, but is essential for getting anything out of the class. Because each topic is treated quite briefly, students are encouraged to read more about the tools presented in class (a reading list is provided for each tool). Rigor and reproducibility of data analysis, as well as good coding style and organization are emphasized. At the end of the Quarter, the students produce a "final project", where they tackle and solve a substantial computational problem stemming from their own research.

Course Objectives

1. Showcase a variety of computational tools.
2. Illustrate the use of the tools using real biological data from published papers.
3. Integrate these tools to produce efficient computational pipelines, automating the analysis, visualization and reporting of biological data.
4. Use appropriate tools for data organization and sharing, debugging, profiling.

Textbook

Allesina, S. and Wilmes, M., **Computing Skills for Biologists**, Princeton UP, 2019

Lecture notes will be provided for the students who don't have a copy of the book.

Course Policy

Computers

The class is taught at the Computer Science Instructional Laboratory (CSIL). Students are required to register with CSIL to get access to the machines, which will have all the needed software already installed. Instructions on how to install the software on personal machines will be provided.

Grading Policy

> *Showing up is 80 percent of life* – Woody Allen, via Marshall Brickman

Ideally, students should take this class for Pass/Fail, rather than a letter grade. I will not grade the homework, and will not take attendance. As such, I will have few occasions to assess whether each student has internalized and mastered the material. Students will however produce a substantial "Final Project", where they take a computational problem in their research or their laboratory, and write code to solve it and automate the analysis. If requested by the graduate program, I will assign a letter grade to the final project.

Attendance Policy

Students should attend all lectures. Students who cannot attend a given lecture should email me beforehand.

Class Schedule

The class meets weekly for 3 hours.

- Week 1: UNIX shell, Version Control
- Week 2: Python (basic programming)
- Week 3: Python (more complex tasks)
- Week 4: Python (debugging, profiling, unit testing)
- Week 5: Python (regular expressions)
- Week 6: Python (scientific computing) + LaTeX
- Week 7: R (basic programming)
- Week 8: R (data wrangling and visualization)
- Week 9: R (recap and more complex examples)
- Week 10: Databases + Wrapping up