Biol313: Practical Computing for Biologists

Instructor: Patricia E. Salerno
Office: Goodpaster 112
Email: pesalerno@smcm.edu
Office hours: Tuesdays noon-2pm
Course website: https://pesalerno.github.io/computing_SP25/
Course times and locations:

Course Summary

Biological data gets more expansive every year, making it crucial to develop and use computational tools to sort, organize, and analyze large datasets. The main goal of this course is to introduce the students to various tools that are available to them, so that they can independently work towards expertise. The course itself will not aim to make you experts, but rather to introduce you to many issues and tools to work around them. The first few weeks we will focus on unix/shell (computing basics) as well as on text editing (more important than people think!). We will then focus on python, and finally delve a bit into github and remote computing, all crucial tools for today's programmers. A lot of the class will make use of real-life genomics datasets of non-model species, so you will also be learning about genomics, genetics, sequencing platforms, and other biological aspects of data. If time permits, we will also learn about bayesian statistics and apply our command-line knowledge to run some tree-inference analyses. We will end the course with independent projects, so that you all have a chance to apply the knowledge acquired on a subject of interest to you!

Course Learning Outcomes

- Achieve basic proficiency in several computing and programming languages
- Understand the computational tools available and their applications to biological problems
- Become proficient in both literature summaries in the subject and in the independent projects and presentations

Topics to be covered

There is no single way to teach a computing course. The course is focused on a couple of programming languages/tools (bash and python) that are highly important and prominent in biological computing, and it will also offer insights to various other tools (markdown, github, remote computing) that can serve as introductions for the student to later delve into more in their own time. The course will be organized as follows:

1. Basic computing (bash/shell) skills and text editing (Weks 1-5/6)
2. Programming with python (Weeks 6-9)
3. Remote computing (Week 10)
4. Github basics (Week 11)
5. Independent Projects.

IMPORTANT: Although there is a clearly defined schedule for the topics, they are highly subject to change pending progress and interests of the students in the class. When plans change… it's usually because it is meant as an improvement of the original plan based on new information that was received… as in, Bayesian updating! On week 12 we will talk about Bayesian Statistics, as long as we don't run out of time.

Evaluations

I believe all students enrolled in an upper-division course begin the course with an A. If you put the work and time in, turn in assignments, do a good job on your project… you should get an A. This course is not a competition – we don't need to show off if we are better or faster or more skilled. We will learn as a class, as a group, and you will help your peers – rather than leave lab early – if you are done before others are. You learn by teaching, and by doing. I expect you all to do more than the bare minimum to pass this course – it is your responsibility as adults that have chosen to enroll in an upper-division course to make the most of this opportunity. So, forget about your grade (for the most part), and focus on the learning. For those times in which you cannot help but worry about your grade, here is the breakdown:

Laboratory Assignments: 650 points
In-class participation: 50 points
Literature Summaries: 50 points
Literature presentation: 50 pts
Independent project (Proposal): 25 points
Independent project (Code, Paper): 125 pts
Independent project (Presentation): 50 pts

Literature summaries and presentations

A major component of success in the sciences is the ability to read, analyze, interpret, and communicate the importance of primary literature. Literature summaries will be one-page summaries of class readings. You will all do one presentation on primary literature to the class. For this, you will develop a presentation (~10-15 minutes) to explain the paper to the class.

Independent project

The completion of an independent research project will be expected. The project will include the use of techniques learned throughout the course to perform some computational task that is relevant to biology, and will include both a proposal and an official write-up of the analysis in the style of a scientific paper (as outlined in the SMCM style guide). The final paper will include relevant citations from the literature (a minimum of five) to put the work in context. The project will also be presented in class in the style of a scientific conference presentation (10-12 minute presentation with 3- 5 minutes for questions from the audience) during the final week of class (including the final exam time period).

Grade Scale

This course will use the letter grade system below. Grades are an important component of any course, but they should not be viewed as the ultimate meaning for the course. If you are concerned about your grade, please contact me well before the end of the course. Often changes to studying behavior, note taking, or taking advantage of office hours to clarify topics, will substantially improve your grade if adopted early enough.

Final Grades will be given according to the following breakdown of final grade percentage: A >92.50; A- 89.50 – 92.49; B+ 86.50 – 89.49; B 82.50 – 86.49; B- 79.50 – 82.49; C+ 76.50 – 79.49; C 72.50 – 76.49; C- 69.50 – 72.49; D+ 66.50 – 69.49; D 60.00 – 66.49; F < 60.00

Course Materials

We will mostly base the course on the following book:

Haddock, S.H.D., and C. W. Dunn. 2012. **Practical computing for biologists**. Sinauer Associates, Sunderland. http://practicalcomputing.org

>All course material, including primary literature and assignments will be posted on the course website.

Attendance

Attendance is important and will be assessed through a participation component of the grade. If a student misses class, they alone are responsible for making up the material, and, in general, assignment extensions will not be given. There will be a 5-point deduction for missing a lecture period and a 10-point deduction for missing a lab period.

Late assignments

If you have a conflict with an assignment deadline you must tell me as soon as possible by email, phone or in-person. I may ask for written confirmation by an independent party (letter from academic services, etc.) if you have an acceptable excuse that causes you to miss an assignment deadline. This is not meant to punish or cause undue stress to the individual but is necessary to ensure fairness to all students in the class. All assignments are due in class on the assigned day at the beginning of class. Late assignments will be deducted 10 percentage points per day (with days rounded up – 5 minutes late = 1 day late).

Academic Honesty

Academic dishonesty at all levels WILL NOT BE TOLERATED and will be prosecuted to the full extent through official College processes. If you are unsure about what constitutes academic dishonesty, I encourage you to carefully read the SMCM academic honesty policy or speak with me specifically. Plagiarism and cheating are grounds for immediate failure of this course. I expect (and encourage) students to work together on problems and while studying. This does not mean that you can turn in the same answers to questions. You may work together to figure out HOW to answer a question, but anything you submit for a grade MUST be your own. Plagiarism is a highly difficult thing to assess when dealing with open-source computer code. You may use the Internet for help to find solutions to problems, but all the code turned in MUST be written independently. Further, although ChatGPT is an available online resource, you are not allowed to use it for this course unless otherwise explicitly noted and agreed upon - it will be considered the same as plagiarism.