Data Science for Biology I: Unix and Python

Fall 2024
BIOL 792 - 1028
Prof: Dr. Thomas Parchman; SFB 209; tparchman@unr.edu
Class: Tuesday/Thursday 6:00 – 7:10 pm FH 219
Office Hours: By appointment

Course Description

Modern Biology is increasingly shaped by data sets that are orders of magnitude larger than most biologists are trained to work with. This is especially the case for the fields of genetics and genomics where major advances in DNA sequencing technology have cut the time and cost investment of DNA sequencing more than 100,000 fold. Spreadsheet software and graphical user interface statistical analysis packages (e.g., Excel, Statistica, JMP) are useless for this scale of data for any scientific discipline. This course will introduce students to basic computational tools (focusing on Unix and Python) to enhance data competency for modern biology. The course will involve hands on approaches and the manipulation and analysis of data collected for graduate student research projects as well as examples from genomic data sets generated in my lab. The course should provide a starting point for students to gain further expertise in simple programming and efficient manipulation and analysis of large-scale data. The tools to be learned in this class are not unique to genomics, and will be of value to research in any scientific field. Prerequisites: must be enrolled as M.S. or Ph.D. student.

Student Learning Outcomes

By the end of the semester, students should have learned enough to start working more confidently with Unix and Python. We will emphasize tools often used in genomics and bioinformatics, but the applications learned will apply generally to data science. This course should also prepare students for the more advanced future courses.

- Students will be able to operate in a Unix computing environment, and will understand the basic use of Unix computing clusters for research.
- Students will be able to write basic programs in Python in order to efficiently manipulate and work with large scale data.
- Students will be able to use basic Unix and Python to manipulate large genomic data sets, and to conduct basic analyses of genome level DNA sequencing data.

Required Materials

- **Computer with Unix/Linux operating system** Students with Mac computers already have machines running Unix and are ready to go. Same goes for students running Linux (Ubuntu, Centos, etc.). Students using computers running a windows OS or will need to figure out how to install Linux or a Linux emulator on their machine.

- **Supplemental primers, readings and assignments** will be announced during the class and provided on the course github page.
- **Practical computing for biologists (*Optional*)** Haddock, S.H.D. and Dunn, C.W., 2011. Sunderland, MA, USA: Sinauer Associates. The book provides an excellent guide to the much of content of the course, is filled with excellent examples and problems, and will also be utilized in Data Science for Biology II during spring semesters.

Course Format

We will meet twice a week (Tues/Thurs 6:00-7:15) but I typically reserve extra time during each window to allow discusion/troubleshooting on coding related to assignments and independent projects. At the beginning of each class, I will introduce new concepts and material that will form the basis of the exercises, assignments, or projects we will work through that week. We will cover questions regarding previous material, and then you will spend at least half of each class working independently, or in small groups, on writing code. All students should come to class having thoroughly read the assigned material and prepared to try new coding exercises.

Course Material Repositoy

All readings, primers, problem set instructions, datasets, as well as ample supplemental materials will be available on the course github page. This will include primers for Unix and Python content, and additional resources for learning more about Python, Unix, and genomic workflows, and information and data sets related to assignments.

Grades

Your grade in this course will be based on the following:

- **Weekly assignments (50%)** Assignments will involve working in the Unix environment, writing simple Bash and Python scripts, and working with a variety of large data sets that will be provided over the course of the semester. Assignments will be evaluated based on completion and effort. You can work in teams of 2 or 3 but will turn in your own Python.py or Bash.sh scripts for each assignment. Code should be annotated, step by step, to explain what you did to complete the task. More guidelines on these files and each specific assignment will be available on the course website. Assignments will be due before class on Tuesdays unless otherwise specified.

- **Participation (30%)** This is a graduate course, with full attendance and participation expected. Participation entails showing up for class prepared and doing your best to work through assigned tasks and programming example problems. Some of the material we cover might be easy and quick to figure out. Other material and tasks will present roadblocks that will be difficult to figure out. No questions will be stupid questions.
- **Independent project (20%)** Everyone will be responsible for an independent project which can be organized either individually, or as a group. This will involve identifying a task or problem in your research (or the research of the group you work within) that either requires, or can be made much more efficient, using Python or Unix scripting. Each group (or individual) will need to turn in a one to two page description of the task and how they will solve it by **week 5**. By week 12, each group will need to turn in final scripts and a one to three page description of how the problem was solved, and how the code works. In addition, each group will give a short (5-10 minute) presentation that describes the data, the problem, and how their scripting tools work. For those without data or ideas, I can supply some options.

Grading scale as follows:

Percentage	Grade
90-100%	A
80-89%	B
70-79%	C
60-69%	D

University Policies

Dropping/Withdrawing

Last day to drop a class and receive a full refund: Sep. 5, 2024
Final day to withdrawal from classes (W, no refund): Oct. 30, 2024

Incomplete Grade

A student may request an “I” if he/she has made satisfactory progress in the majority of the work in the course, but for unavoidable absences or other conditions beyond his/her control, is unable to complete the course. Non-attendance, poor performance or requests to repeat the course are unacceptable reasons for issuance of the “I” mark.

Academic Honesty and Unprofessional Conduct

Academic dishonesty (cheating, plagiarism or other dishonest behavior related to grades and performance) will not be tolerated under any circumstances.

Policy on Taping/Recording Class Lectures

Surreptitious or covert video-taping of class or unauthorized audio recording of class is prohibited by law and by Board of Regents policy. This class may be videotaped or audio recorded only with the written permission of the instructor. In order to accommodate students with disabilities, some students may have been given permission to record class lectures and discussions. Therefore, students should understand that their comments during class may be recorded.

Statement of Disabled Access and Reasonable Accommodation

Qualified, self-identified students with documented physical and learning disabilities have the right to accommodations to ensure equal access to educational opportunities. For assistance, contact the Disability Resource Center (DRC) at 784-6000 to determine eligibility and appropriate accommodations.

SCHEDULE

*Tentative Course Schedule. All contents are subject to change.

Week	Date	Class	Due
Week 1	Aug. 27, Aug 29	Course introduction, Unix I	
Week 2	Sep. 3, Sep. 5	Unix II	
Week 3	Sep. 10, 12	Unix III	Homework 1
Week 4	Sep. 17, 19	Unix IV	Homework 2
Week 5	Sep. 24, 26	Python I	Homework 3
Week 6	Oct. 1, 3	Python II	Homework 4
Week 7	Oct. 8, 10	Python III	Homework 5; *1-2 page project description
Week 8	Oct. 15, 17	Python IV	Homework 6
Week 9	Oct. 22, 24	Python V, Data science careers	Homework 7
Week 10	Oct. 29, Nov. 31	Python VI	Homework 8
Week 11	Nov. 5, 7	Python VII (pandas/numpy)	Homework 9
Week 12	Nov. 12, 14	Jupyter, Rmarkdown	nothing
Week 13	Nov. 19, 21	Python VIII	nothing
Week 14	Nov. 26,	HPC/Pronghorn/Project prep	
Week 15	Dec. 3, 5	Project prep/presentation	projects due

Week	Date	Class	Due
Week 16	Dec. 10	Present Projects	*projects due