Syllabus for GEN220: High Throughput Biological Data Processing

Course Description

This course focuses on computational skills for processing data using programming language Python and UNIX environment. No prior programming experience is required, but some basic computer skills will be useful.

With the advancement of high throughput data generation methods, a major challenge that graduate students in life sciences have to face today is to analyze large amount of biological data. The objective of this course is to provide an opportunity for graduate students with no computer science background to learn the basic skills of handling high throughput biological data. It covers the Linux/Unix environment and the importance of the command line interface; the Python programming language; program design, implementation, and testing; BioPython; Strategies for analyzing genome resequencing, RNASeq, and microbiome sequencing data. Students build hands-on skills by analyzing real high throughput biological data through homework assignments and team projects.

Units: 3

Instructor: Jason Stajich (jason.stajich@ucr.edu)

Time and location: W/F 3:00-4:30PM Campbell Hall 104

Office Hours: M (via Zoom)

https://biodataprog.github.io/GEN220_2022/

Prerequisites

- Coursework in genetics or molecular biology or permission of instructor

Resources

None of these texts are required for completion of the course but they will provide a great deal of helpful background and examples that will improve your ability to master UNIX or Programming in Python.

1. *Bioinformatics Data Skills: Reproducible and Robust Research with Open Source Tools*. Vince Buffalo. 2015 O'Reilly & Associates. Available from O'Reilly and Associates, Amazon Free to read on UCR network (or use VPN) - Safari link.
2. *Unix and Perl to the Rescue: A Primer*. Keith Bradnam and Ian Korf. Unix and Perl Primer for Biologists
3. *Unix and Perl to the rescue!* Bradnam and Korf. Amazon

4. Rosalind - An online platform to learn bioinformatics and programming in Python.
5. Software Carpentry - https://software-carpentry.org/ and Data Carpentry - http://www.datacarpentry.org/.
6. Berk Ekmekci, Charles E. McAnany, Cameron Mura. An Introduction to Programming for Bioscientists: A Python-Based Primer. PLoS Comp Bio. DOI: 10.1371/journal.pcbi.1004867
7. Ken Youens-Clark. Tiny Python Projects. https://www.manning.com/books/tiny-python-projects
8. Pat Schloss's Riffomonas Code Club has great videos and links to programming and microbiome analyses.

Grading

- Programming Homework assignments (5 in total): 50% of grade
- Project: 50% of grade

Homework

- Homework is due before class on the date listed in the syllabus or if amended when assignment is given.
- There will be a programming assignment every two weeks during the first half of the course. Programming assignments must be prepared along with any necessary input files or documentation to demonstrate program usage.
- Code should be runnable as turned in. You will deposit your code in your github repository or if not possible, by Canvas. You can make one private personal repository to deposit and should organize a folder for each homework assignment (e.g. hw1, hw2, hw3, hw4). There will be a link to create these through GitHub Classroom and posted in Canvas and Piazza.

Technology Requirements

Because this course takes place online and requires use of a computer, you will need the following:

Hardware

Access to a current Mac or PC (with a fast processor and speakers) Webcam and microphone (to participate in any video components, e.g. live sessions, remote proctoring, video presentations) Software

Operating systems:

- OSX is recommended with Xquartz installed for X11
- UNIX or Linux eg Ubuntu
- Windows with MobaXterm

Other recommended software - a local text editor

- Atom is highly recommended
- Notepad++

Projects

- Project Topics will be discussed in October and teams will select a project idea to focus on.
- Project will be 2-3 individuals working together.
- A presentation will be made by each team - last day(s) of class.
- A final report with the details will be turned in by the group.
- The report needs to detail what each person's contribution is to the project.

Schedule

Date	Day	Lecture Topic	Notes
Sept-24	F	Course Intro / UNIX I: Cmdline, GitHub	
Sept-29	W	UNIX II: Biocluster HPCC, Running programs	
Oct-1	F	UNIX III: Tools for data processing	Homework 1 Due
Oct-6	W	UNIX IV: Advanced UNIX and data processing	
Oct-8	F	Python I - Variables, running, cmdline, strings, math	Homework 2 Due
Oct-13	W	Python II - Logic, loops, lists, iterator; I/O reading/writing files	
Oct-15	F	Python III - Dictionaries, Arrays, functions	Homework 3 Due
Oct-20	W	Python IV - Structured data parsing (GFF, CSV, BED)	
Oct-22	F	Python V - BioPython, Pandas	
Oct-27	W	(No Live lecture class - recorded Video to be posted TBD)	

Date	Day	Lecture Topic	Notes
Oct-29	F	Alignment and Bioinformatics Algorithms; LAST, cmdline & automation	Homework 4 Due
Nov-3	W	Bioinformatics I - Aligning short reads, coverage, identifying variants	
Nov-5	F	Bioinformatics II - RNASeq analyses	
Nov-10	W	Bioinformatics III - Genome Assembly	
Nov-12	F	Bioinformatics IV - Protein Sequence analyses (HMMER, InterPro, SignalP)	
Nov-17	W	Bioinformatics V - Orthology, Phylogenetics and automation	Homework 5 Due
Nov-19	F	Databases - SQLite	
Nov-24	W	Review Bioinformatics and Databases	Extra Topics
Nov-26	F	**NO CLASS - Native American Heritage Day**	
Dec-1	W	Data visualization in R and python	
Dec-3	F	Automation and workflows	
Dec-8	W	Review Topics	
Dec-10	F	Class Presentations	
Dec-15	W	Final Project Reports Due	