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 com1mand line interface; the Python programming language; program design, implementation, and testing; BioPython; Strategies for analyzing genome resequencing, RNASeq, 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 4:00-4:50PM, F 3:00-4:50PM, Zoom-land

Office Hours:

https://biodataprog.github.io/GEN220_2020/

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 also try UC Library 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 Biosciensts: 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

Grading

- Programming Homework assignments (4 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 iLearn. You can make one private personal repository to deposit and should organize a folder for each homework assignment (e.g. hw1, hw2, hw3, hw4).

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
Oct-2	F	Course Intro / UNIX I: Cmdline, GitHub	
Oct-7	W	UNIX II: Biocluster HPCC, Running programs	
Oct-9	F	UNIX III: Tools for data processing	
Oct-14	W	UNIX IV: Advanced UNIX and data processing	
Oct-16	F	Python 1 - Variables, running, cmdline, strings, math	
Oct-21	W	Python 2 - Logic, loops, lists, iterator; I/O reading/writing files	
Oct-23	F	Python 3 - Dictionaries, Arrays, functions	
Oct-28	W	**NO IN CLASS LECTURE** - Structured data (CSV, XML, GFF, BED)	
Oct-30	F	Python 5 - BioPython, Pandas	
Nov-4	W	Python Review concepts	
Nov-6	F	Alignment and Bioinformatics Algorithms; LAST, cmdline & automation	
Nov-11	W	Bioinformatics 1 - Aligning short reads, coverage, identifying variants	
Nov-13	F	Bioinformatics 2 - RNASeq analyses	
Nov-18	W	Bioinformatics 3 - Genome Assembly	
Nov-20	F	Bioinformatics 5 - Protein Sequence analyses (HMMER, InterPro, SignalP)	
Nov-25	W	Data Plotting and basic R graphics	
Nov-27	F	** Native American Heritage Day; No Class Meeting **	
Dec-2	W	Bioinformatics 6 - ChIP-Seq, ATAC-Seq, 5mC methylaion data	
Dec-4	F	Bioinformatics 7 - Phylogenentic analyses	

Date	Day	Lecture Topic	Notes
Dec-9	W	**NO IN CLASS LECTURE** - Material will be posted - review major topics	
Dec-11	F	Class Presentations	