**Practical Bioinformatics**

**Biol 4220**

Fall 2020

|  |  |  |
| --- | --- | --- |
| **Instructor** |  | Dr. Michael J. Landis  [michael.landis@wustl.edu](mailto:michael.landis@wustl.edu)  (314) 935 - 8082  Rebstock 210 (office) |
| **Class hours** |  | Online (Zoom) MW 8:30am – 12:00pm |
| **Office hours** |  | Online (Zoom) or MW 1:00pm – 2:00pm Rebstock 210 (by appointment) |
| ***Covid-19 logistics for Fall 2020*** |  | To protect everyone’s safety, Biol 4220 in Fall 2020 will be taught in an online format that is primarily asynchronous, with some synchronous events. Details are provided towards the end of the syllabus. |
| **Course description** |  | From medicine to genomics to ecology, all fields of biology are now generating large and complex datasets that can only be analyzed using computational approaches. This course introduces computational techniques and perspectives to biologists that are new to computational thinking. Students will learn how to design research workflows, decompose complex problems into simpler solvable units, and apply scientific computing principles to research. In addition, students will practice foundational computing skills, such as how to use the UNIX operating system on research clusters, write custom analysis programs with shell scripts and with Python, and summarize and visualize analysis output. The laboratory exercises build on one another, culminating in the construction of a bioinformatics pipeline that can process and analyze molecular data. Students will apply their newly learned computational skills and use their pipeline to analyze virus sequence evolution and explore evolutionary models. |
| **Course information** |  | 4 units. Intended for upper-division Biology undergraduates and for early-stage Biology graduate students. The course satisfies the Advanced Laboratory Requirement for both the Biology Major and for the Genomics and Computational Biology Track in Biology. The course also satisfies the Advanced Biology Elective requirement for the Bioinformatics Minor. |
| **Expected enrollment** |  | 10-12 students |
| **Prerequisite courses** |  | Biol 2970  Math 132 (Calculus II)  Math 223 (Calculus III) or 2200 (Elementary Probability) |
| **Other suggested courses** |  | CSE 131 (Computer Science I)  Biol 3100 (R Workshop in Biology) |
| **Primary text** |  | “Computing Skills for Biologists” by Allesina & Wilmes (ISBN-13: 978-0691182759) |
| **Suggested texts** |  | “Bioinformatics Data Skills” by Buffalo  (ISBN-13: 978-1449367374) |
| **Computer resources** |  | All work will be performed through remote UNIX-based virtual machines that can be accessed from your person computer through the network. Please contact the instructor if you do not have a computer and/or internet access in your place of residence. |
| **Learning objectives** |  | Students will learn to   * design, build, and apply bioinformatics pipelines “from scratch” to process and analyze molecular datasets using assorted computational resources * compare and contrast how biological inferences and hypothesis tests are sensitive to computational settings, such as data cleaning and choice of methods * identify major features of computer hardware, operating systems, and programming languages * write, modify, optimize, and debug Python and UNIX code to solve biological problems * work with fellow students using modern collaboration tools and software review methods |
| **Assignments** |  | **Reading** assignments are selected to prepare students for each session’s lecture and lab. Reading assignments will typically cover 15-30 pages from either the primary text (“Computing Skills for Biologists” by Allesina & Wilmes) or supplementary materials, such as research papers or online tutorials, that will be shared by the instructor through Canvas. Content from reading assignments will be applied during laboratory exercises.  **Participation** allows students to express their knowledge or curiosity about the course material. Participation includes asking questions during class, in the Canvas forum, and attending office hours.  **Quizzes** review what was taught in previous lecture. Each session will begin with a 5-minute quiz that will be turned in immediately. The five quizzes with the lowest scores will be ignored when computing grades.  **Labs** apply taught concepts to practice and develop problem solving skills. New lab assignments are posted Monday mornings at 8:30am each week. Each lab must be submitted for grading within one week to the GitHub Classroom site. Submitting assignments through our GitHub Classroom site will be covered in class. Instructions for submitting assignments are also found on the GitHub Classroom site for Biol 4220 ([link](https://classroom.github.com/classrooms/42896217-biol-4220)).  **Code reviews** develop peer relationships and communication skills. Students will be assigned a random partner for code review each week. Instructions for code reviews are given on the GitHub Classroom site for Biol 4220 ([link](https://classroom.github.com/classrooms/42896217-biol-4220)).  **Midterm + Final exams** will test students’ understanding of definitions, application of formulas, ability to read and write code, and ability to interpret and critique bioinformatics case studies. The midterm exam will cover topics from the first half of the course, while the final exam will be comprehensive of all material.  **Project + Presentation** will require students to design and execute a small analysis for one or more additional datasets using the analysis pipeline they built. The instructor will provide a list of curated datasets to analyze, which include other infectious diseases but also various clades of plants and animals. Students must get approval from the instructor if they wish to analyze datasets that are not part of the pre-approved list. Students are also encouraged to modify their pipeline to ask novel questions not covered directly in class. Students will deliver a 10-minute presentation on the final day of classes, describing their work to their classmates. Students may choose to emphasize their biological findings, their computational innovations, or their technical challenges. By the time of the final exam, students will submit the GitHub link for analysis and a two-page summary of their work to the instructor. Details for the project are located in the GitHub Classroom site for Biol 4220 ([link](https://classroom.github.com/classrooms/42896217-biol-4220)). |
| **Grading** |  | Grades are primarily determined through assignments and exams:  10% Assessment (quizzes + participation)  50% Labs (exercises + code reviews)  10% Midterm  15% Final exam  15% Project + Presentation  Letter grades typically follow the standard numerical scale: A for 90 to 100; B for 80 to 89; C for 70 to 79; D for 60 to 69; F for 0 to 59. The instructor may manually adjust letter grades based on performance and effort displayed by either individual students or by the class as a whole. |
| **General Policy and Procedures** |  | All students will adhere to the tenets of the Undergraduate Student Academic Integrity Policy ([link](https://wustl.edu/about/compliance-policies/academic-policies/undergraduate-student-academic-integrity-policy/)), which forbids cheating, plagiarism, and other forms of academic dishonesty.  Students will follow the University’s Discrimination and Harassment Policy ([link](https://hr.wustl.edu/items/discrimination-harassment-policy/)). Any student who feels they have experienced harassment or discrimination, or if they witness their peers being subjected to such behavior, that student has options to report the behavior anonymously, without fear of repercussions, as described in the University’s Discrimination and Harassment Policy. |
| **Special Accommodations** |  | The instructor will do everything possible to satisfy any Special Accommodations for students. Students are responsible for requesting Special Accommodations at least two weeks before those accommodations are needed through Disability Resources ([link](https://students.wustl.edu/requesting-academic-accommodations/)). |
| ***Covid-19 logistics for Fall 2020***  ***(continued)*** |  | Biol 4220 has been adapted to an online format for Fall 2020. The goal is to ensure that everyone remains as healthy as possible while retaining a classroom experience that is engaging and enriching for all. At any time, please provide feedback and/or suggestions for ways we could better achieve this goal for you and/or the entire class.  Lectures and lab exercises will be taught online, with lecture slides and lab materials posted online at the start of each class meeting. The first 30-60 minutes of each class will begin with a lecture over Zoom that provides an overview of day’s learning objectives and the general design of that day’s lab exercises. Throughout the remainder of each class, I’ll be available to answer questions, expand on topics, and help with assignments.  Students may work on the lab assignments whenever they prefer. Some lab assignments will be team exercises, meaning students will need to work collaboratively, coordinate efforts, and submit work as groups. Just as with individual exercises, team exercises can take place during the time reserved for the course, our outside of that time. Quizzes will be assigned at each class, and must be submitted before the start of the next class.  Office hours will be held over Zoom with in-person office hours available weekly with an appointment. Students and faculty who meet for in-person office hours must follow WUSTL safety protocols (<https://covid19.wustl.edu/health-safety/>), including self-screening (<https://screening.wustl.edu/Screen>). |
|  |  |  |

**Lecture/Lab Schedule**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topic** | **Lecture** | **Lab** | **Reading** |
| **1** | Introduction  Computer architecture + infrastructure | Introduction, examples of computational problems in biology  hardware, operating systems, networks, servers | Share research interests, skills assessment, ssh into cluster, download virus dataset | TBD |
| **2** | UNIX 1 | UNIX I. Shell, filesystem, commands, scripts | “Hello, world!”, Writing pseudocode. Script to create files, move files, transfer files, delete files | Chapter 1.1 – 1.5 |
| **3** | UNIX 2 | UNIX II. Files and strings, file format principles, text processing, grep, awk, regular expressions | Process virus dataset, identify all unique motifs of length k, count them, sort them | Chapter 1.6  Chapter 5 |
| **4** | Biology 1 | Calling biology software, alignment, motif finder, etc. Alignment basics, motif finder basics.  Submitting cluster jobs. | Questions to test understanding of how changing software settings changes output | BWT paper  F84 paper |
| **5** | Version control | VCS, git, clone, branch, add, rm, commit, merge, rebase, etc. | Make first repo, add your code, clone someone else’s repo, change their code and submit/accept pull request  *Rest of course will use git for grading! Will cover remaining git topics as needed.* | Chapter 2 |
| **6** | UNIX 3 | Scripting basics. More on filesystem, path, alias, variables, control structures, etc. | Design a pipeline to align flu dataset under various settings, generate named output. | Chapter 1.7 – 1.8 |
| **7** | Biology 2 | Molecular phylogenetics, positive selection | Build tree, identify sites under selection, incongruent topologies | dN/dS paper  PAML paper |
| **8** | UNIX 4 | Processing batches of files, pipelines, resource management, cluster queues  **Midterm (Weeks 1 to 8)** | Design pipeline to measure sensitivity of positive selection tests to alignment settings, etc. | TBD |
| **9** | Python 1 | Variables, values, data types, operators, file handling | “Hello, world!” program.  File processing. Syntax exercises. Writing comments. | Chapter 3.1 – 3.4 |
| **10** | Python 2 | Control structures, if, for, while, functions | Write program to translate alignment codons to amino acids. Compute pairwise differences. | Chapter 3.5 -3.7 |
| **11** | Python 3 | Custom programs as scripts. Pipelines with shell, Python, and installed programs. | Develop pipeline to download files, managing processes, tmux, advanced job queues | Chapter 4 |
| **12** | Scientific computing 1 | Data tables, transformations, containers, basics of algorithms complexity  numpy, scipy | Creating numpy containers, simulation and optimization with scipy,  PCA and k-means with scipy  Exercises to measure complexity for speed/storage. Develop two algorithms to transform a dataset. | Chapter 6.1 – 6.2 |
| **13** | Scientific computing 2 | pandas, biopython | Creating and accessing pandas containers, handling sequence data with biopython | Chapter 6.3 – 6.5 |
| **14** | Biology 3 | “Big data” ethics, reproducibility, open science  matplotlib, eve | Plotting exercises with matplotlib, eve  Write documentation for your analysis pipeline.  Presentations? | Chapter 4 (revisited) |
| **15** | -- | **Final exam (Weeks 1 to 14)** | -- |  |