EE 372: Data Science for High-Throughput Sequencing

Winter 2018

Course Description

Extraordinary advances in sequencing technology in the past decade have revolutionized biology and medicine. Many high-throughput sequencing based assays have been designed to make various biological measurements of interest. This course explores the various computational and statistical problems that arises from processing high throughput sequencing data. Specific problems we will study include genome assembly, haplotype phasing, RNA-Seq quantification, single cell RNA-seq analysis, etc. Specific techniques we will learn to solve these problems include spectral algorithms, dynamic programming, the EM algorithm PCA, FDR, etc. Through this course, the student will also get familiar with various software tools developed for the analysis of real sequencing data.

Communication

Course news and assignments will be posted at `ee372.stanford.edu`, which redirects to the course's GitHub website. Each assignment and set of lecture notes will have its own page, and students are encouraged to ask and answer questions by leaving or replying to comments on these pages.

Prerequisites

- Undergraduate level probability
- Some programming experience. We will be using Python.
- Some undergraduate background in algorithms would be beneficial
- No prior background in biology will be assumed

Lecture Times

Tuesday, Thursday 1:30-2:50pm at 540-108

Course Staff

Instructor: David Tse (`dntse@stanford.edu`)
Teaching assistants: Govinda Kamath (`gkamath@stanford.edu`), Jesse Zhang (`jessez@stanford.edu`)

Office hours: Mon 3:00-4:00pm and Thurs 3:15-4:15pm at Packard 264 for instructor

Course Grading

The grading for the course will be broken down as follows:

- Attendance 10%
- Scribing 10%
- Problem Sets 30%
- Project 50%

Attendance

Students are encouraged to participate in class either during lecture or by leaving comments on material posted at the course website.

Scribing

Each student will be responsible for scribing a lecture. To ensure that the notes will be available for students currently in the course, **scribed notes are due within 72 hours after lecture** (no late submissions accepted). A Google Doc will be used for reserving lectures for scribing.

Assignments

There will be 3-4 assignments. The assignments will involve a theory component and a programming component. The programming component is aimed at exposing students to the messiness involved in real data and various tools used in practice. All programming assignments will require only laptop-level computing. The main language used to code will be Python. UNIX/LINUX/OS X may be needed for some of the software packages.

Projects

Projects can be theoretic or practical in nature (ideally a mix of the two). Additional details and a list of possible projects will be put up shortly. Students can also come up with project topics that they are interested in (in consultation with the teaching staff).

Lectures

1. Introduction
2. Biochemistry background and sequencing by Synthesis
3. Base calling for second-generation sequencing
4. Third-generation technologies: Nanopore and Pacific Biosciences
5. Genome assembly I: Formulation
6. Genome assembly II: Read overlap and De Bruijn graphs
7. Long-read assembly: a case study
8. Haplotype phasing
9. Alignment I
10. Alignment II
11. RNA-Seq I: Introduction
12. RNA-Seq II: Quantification vs the EM algorithm
13. RNA-SEq III: differential expression
14. Single-cell RNA-Seq I: technology and challenges
15. Single-cell RNA-Seq II: Dimensionality reduction, clustering and computing representatives
16. Single-cell RNA-Seq III: Dimensionality reduction, clustering and computing representatives
17. Single-cell RNA-Seq IV: Differential expression
18. Cancer genomics
19. Project presentation