

CS 690U: Computational Biology and Bioinformatics

Meeting Days: T/Th

Times: 11:30 – 12:45pm

Room: CS 142

Credits: 3

Instructional contact hours per week: 3

Delivery mode: In person

Instructor:

Anna Green

annagreen@umass.edu

Office: CSL E449

Pronouns: she/her

TA:

Saishradha Mohanty

saishradhamo@umass.edu

Pronouns: she/her

Office Hours:

Anna Green: Tuesdays 10:30 – 11:30am, CSL E449 (before class)

Saishradha Mohanty: remote, see Canvas for link

Course Description:

This course is designed to provide computer scientists with a comprehensive introduction to the field of computational biology. The course will cover the application of computational techniques to modern research challenges in biology, discussing both foundational algorithms and newly introduced methods. The necessary background on biology will be provided in order to understand the methods. The primary focus will be analysis of genomic data, including genome assembly, genome annotation, sequence alignment, phylogeny construction, mutation effect prediction, population genetics, and genotype-phenotype association studies. We will also cover and protein structure analysis and prediction. Throughout the course, we will emphasize the unique challenges to working with biological data. Through lectures and hands-on programming problem sets, students will develop the necessary skills to tackle computational challenges in the field of biology.

Learning Objectives:

- Gain an understanding of fundamental biological concepts and their relevance to computational biology.
- Develop proficiency in applying computational techniques and algorithms to analyze genomic data.
- Acquire practical skills in programming and data analysis relevant to computational biology.
- Explore and critically evaluate current research in the field of computational biology.
- Foster the ability to design and implement computational approaches for solving research problems in biology.

Prerequisites: No formal prerequisites. This is a graduate-level CS course and thus proficiency in Python programming as well as undergraduate-level understanding of statistics, machine learning, and linear algebra is expected)

Online Resources / Course Management Software

Canvas: will be used for posting lecture slides, problem sets, Echo360 recordings, and announcements.

Piazza: please use for questions related to course logistics, content, and problem sets. If the answer could be useful for another student to know, please post on Piazza. You may post anonymously if you wish, but your name will be visible to instructors. (link posted on Canvas)

Gradescope: Used for turning in your problem sets, and for returning grades to you. Note that all your assignments will be graded by a human being – any grades given by the autograder can be disregarded. Entry Code: (see Canvas)

Textbook: None. Readings will be free electronic materials provided by the instructor, including pdfs of journal and conference articles and pdfs of free textbooks.

Eligibility: This course is aimed at PhD students with a computer science background. The course assumes no prior knowledge of biology. The course may also be suitable for PhD students in biology-related disciplines who have strong computational skills.

Grading Criteria:

60% problem sets. The problem set with the lowest grade will be dropped.

25% project.

15% in-class work (on Feb 26th, April 7th, May 5th, and May 7th)

Bonus 2% participation: students who attend class and office hours regularly may receive up to 2% participation if it will affect their final letter grade

Late work policy: Problem sets are due at 11:59:59 pm on the day specified.

You have four late days that you may use over the course of the semester, but no more than two late days may be used per problem set. **Work that is more than 2 days (48 hours) late will not be graded.** You do not need to request to use late days, they are automatically tracked by Gradescope. However, **you are responsible** for knowing how many late days you have used. Remember that your lowest problem set grade will be dropped (ie, not counted towards your final average), even if that lowest grade is a 0.

If you require an exception to the policies stated above, you must either:

1. Have an accommodation from the Disability Services office which provides you with extra time on assignments, and have discussed the accommodation with me at the beginning of the semester
2. OR Contact CICS Advising, Elizabeth Parolski (MS students) or Eileen Hamel (PhD students), to discuss your need for an exception to course policies due to emergency circumstances
3. OR, Contact the [UMass Dean of Students Office](#), to discuss your need for an exception to course policies due to emergency circumstances

Final Project: Your grade on your final project will be broken into the following components:

5% milestone 1 (one per person)

5% milestone 2 (one per group)

20% milestone 3 / declaration of intent (one per group)

10% data and code available in reproducible github repository

20% in-class presentation

40% written report ($3 + (n-1)$ pages where n is the number of group members)

Please note I reserve the right to modify your final project grade based on feedback from your group members, if you do not contribute equally to the work

Grading Scale: Grading is on a letter scale, listed below. Grades will be rounded to the nearest integer.

Graduate students:

- A: 93-100%
- A-: 90-92%
- B+: 87-89%
- B: 84-86%
- B-: 80-83%
- C+: 77-79%
- C: 74-76%
- F: below 73.5

Attendance policy: Regular attendance and participation is critical to developing an understanding of the material and achieving success on the final project. There is no formal attendance policy. You are expected to attend class on the four days (Feb 26th, April 7th, May 5th, and May 7th) to participate in in-class activities.

Collaboration policy: In corporate and academic settings, it is encouraged that you collaborate with your colleagues and use available resources to complete work. In this spirit, I encourage you to discuss course material with your classmates and use online resources to extend your understanding. However, in order to fairly evaluate your understanding in a classroom setting, **I expect all versions of all assignments to be produced independently by you, in your own words (or code), and to reflect your own understanding of the problem.** Copying any component of an assignment from your fellow students or any other resource (including chatGPT and similar technology) is not permitted. Each assignment will include an option to describe any resources you used or fellow students you discussed with answers with, please answer these honestly. I encourage you to use piazza to discuss all aspects of assignments, including short snippets of implementation.

University policies: regarding **Accommodations, Academic Integrity, and Title IX**, apply to all courses including CS 690U. The policies can be found at:

<https://www.umass.edu/senate/book/non-responsible-employee-required-syllabus-statements>.
Note that I am a “Non-Responsible Employee” under the Title IX definition.

Course Inclusiveness Statement: It is important to me that this course be a welcoming environment to people of all backgrounds. My goal as an instructor is to help you learn the subject material in a way that is useful and empowering, and I believe that the best learning happens on a foundation of mutual respect.

This course will discuss subject matter related to human genetics and evolution. While in an ideal world science would be objective, the reality is that false beliefs about genetic differences between humans have been and continue to be used to justify racism and oppression. I will strive to teach accurate information about the data and techniques used in human genomics while also acknowledging their potential for misinterpretations.

I anticipate that students in this interdisciplinary course may come from different intellectual backgrounds, and thus there may be substantial differences in terms of familiarity with the concepts. I expect you to be patient with your fellow students, and hope that you will help one another in learning the material. I also hope that you will ask questions when something in the course is confusing or unfamiliar! There are no stupid questions.

Please know that my door is open to you if you wish to bring any issues to my attention.

Course Schedule:

Date	Day of Week	Subject	Lecture #	Problem set	Project Milestones due				
29-Jan	Thursday	DNA as information	0						
3-Feb	Tuesday	Sequence Alignment: the problem and classical solutions	1						
5-Feb	Thursday	Sequence Alignment: modern solutions (BLAST)	2						
10-Feb	Tuesday	Sequencing: DNA sequencing technology and variant calling	3	pset 1: Sequence alignment					
12-Feb	Thursday	Sequencing: read mapping, Long read sequencing, graph genomes, pan	4						
17-Feb	Tuesday	Genome Annotation: its importance and classical solutions	5	pset 2: variant calling					
19-Feb	Thursday	NO CLASS due to Monday schedule being followed							
24-Feb	Tuesday	Optional attendance of Dr. Sehi L'Yi's talk							
26-Feb	Thursday	In-class work day: Modeling sequences, and the DGEB paper			PROJECT MILESTONE 1: Attest that you have done the reading				
3-Mar	Tuesday	Genome Annotation: HMMs and PFAM domains	6						
5-Mar	Thursday	Phylogeny: continuous time markov models, jukes-cantor model	7						
10-Mar	Tuesday	Population genetics: basic concepts	8	pset 3: genome annotation					
12-Mar	Thursday	Association studies: basics of association studies (GWAS), controlling f	9						
17-Mar	Tuesday	NO CLASS Spring Break							
19-Mar	Thursday	NO CLASS Spring Break							
24-Mar	Tuesday	NO CLASS Dr. Green Travel		pset 4: GWAS					
26-Mar	Thursday	Association studies: heritability, polygenic risk scores, interpretation of	10						
31-Mar	Tuesday	Mutation effect prediction: basics of mutaiton effects in proteins, class	11		PROJECT MILESTONE 2: Groups formed. Candidate task and project selected				
2-Apr	Thursday	Mutation effect prediction: modern solutions for mutation effect predi	12						
7-Apr	Tuesday	In-class work day: Modeling sequences, and the DGEB paper		pset 5: mutation effect prediction					
9-Apr	Thursday	Interpretability in biological sequence modeling (Why? What is SHAP?)	13						
14-Apr	Tuesday	Protein structure prediction: introduction to protein structure data, clas	14		PROJECT MILESTONE 3: declaration of intent				
16-Apr	Thursday	Protein structure prediction: modern solutions (alphafold)	15						
21-Apr	Tuesday	Mutation effect prediction: in non-coding regions, annotating transcript	16	pset 6: protein structure prediction					
23-Apr	Thursday	Mutation effect prediction: the ENFORMER model for non-coding regio	17						
28-Apr	Tuesday	Introduction to RNA sequencing and sc-RNA sequencing	18						
30-Apr	Thursday	Dimensionality reduction techniques (t-SNE, UMAP, PCA)	19						
5-May	Tuesday	Final presentations group A			Project write-up due				
7-May	Thursday	Final presentations group B			Project presentation				

Course Contents

Module 0: Introduction to computational biology and relevant concepts in biology

Overview of computational biology and its interdisciplinary nature

Introduction to basic biological concepts, including genes, genomes, and genetic variation

Module 1: Sequence search and alignment

Concepts underlying sequence alignment

Probabilistic foundations of alignment

Sequence scoring matrices

Needleman-Wunsch algorithm and extensions (eg Smith-Waterman algorithm)

Fast sequence database search using BLAST

Models and Algorithms: Needleman-Wunsch algorithm

Module 2: From sequencing technology to genome sequences

Introduction to DNA sequencing technologies and read formats

Analysis of short and long-read sequences

Variant calling relative to a reference genome

Module 3: Annotating genome sequences with functional information

Introduction to genome annotation and its importance

Gene prediction algorithms: *ab initio* and comparative methods

PFAM domains

Models and Algorithms: Hidden Markov models

Module 4: Phylogeny Construction

Introduction to phylogenetics and evolutionary relationships

Phylogenetic tree visualization and interpretation

Module 5: Population Genetics and Tests for Selection

Basic concepts in population genetics: allele frequencies, genetic drift, natural selection

Detecting signatures of selection (eg, dN/dS, linkage disequilibrium)

Analysis of genomic variation in populations.

Models and algorithms: random walks, Wright-Fisher Model

Module 6: Genome-Wide Association Studies (GWAS)

Introduction to GWAS and its role in identifying genetic factors associated with traits and diseases

Statistical methods for identifying significant genetic variants

Models and algorithms: linear and logistic regression, significance testing

Module 7: Mutation Effect Prediction in proteins

Introduction to genetic mutations and their impact on proteins and non-coding regions

Functional consequences of genetic variants: missense, nonsense, frameshift, etc.

Prediction of protein structure and function changes

Recent deep learning methods used to predict mutation effects

Interpretability in biological sequence modeling

Models and algorithms: auto-encoders, transformer-based models

Module 8: Mutation effect prediction in non-coding regions

Introduction to non-coding region functions, gene regulation, epigenetics, and chromatin
Laboratory methods for measuring function of non-coding regions
The ENCODE project to annotate non-coding DNA
Machine learning methods to predict function in non-coding regions
Models and algorithms: convolutional neural network, transformers

Module 9: Protein Structure Prediction

Introduction to protein structure and its importance
Protein structure prediction methods: homology modeling, ab initio methods
Recent advances: AlphaFold
Evaluation of predicted protein structures
Disordered proteins and the challenges still to come
Models and algorithms: AlphaFold

Module 10: RNA sequencing

Introduction to and motivation of RNA-seq and sc-RNA-seq
Inferring gene expression level from RNA sequencing data
Dimensionality reduction techniques: PCA, tSNE, and UMAP