02-251: Great Ideas in Computational Biology

Final Project Assignment

Spring 2024

Learning Objectives

The lectures in this course focus on the algorithmic side of computational biology: modeling biological problems computationally and then formulating algorithms to solve our problems, sometimes under certain assumptions. We want you to have the chance to explore a practical challenge of your own choosing by running a computational analysis on a real biological dataset, and then interpreting the results.

Within this goal, there are two main directions. You may choose to implement an algorithm solving a biological problem and then apply this algorithm to analyze your dataset, or you may use existing software to analyze the dataset. Regardless of which direction you choose, the expectation for the project is on the analysis that you carry out, although consideration will be given to projects that require a great deal of coding.

Furthermore, your work should focus on *replicating* the work of an existing published research paper in computational biology and then communicating that work to an audience of your peers. In some cases, it may be possible for groups to extend a project after replicating existing research, but you should focus most of your work on replicating the paper.

The project will be completed in groups of up to four students. You are free to choose your teammates yourself. You are also free to work solo or with a partner, but in practice, groups of about four students seem to be ideal for projects.

A group-based project has many benefits. It will allow you to practice the communication and interpersonal skills that are vital to workplace collaboration. If your project is well planned and well executed, then it means that you will likely have less work than if you were completing an entire project independently. And it means that we will have the chance to feature *every* student project in project presentations during the last week of the course, which is the best week of the course!

Project Components

We expect the following deliverables as part of a successful project. Each deliverable includes the percentage of the project grade allocated to it. See ”Important Dates” for due dates; materials can be submitted on Canvas.

Note: A group must deliver both a project presentation and complete an essay in order to receive credit for the project.

Deliverable 0: Group Members (0%)

Please identify teammates if applicable and then add yourself to a project group on Canvas (under the "People" tab); if you need a group, please let us know as well.

Deliverable 1: Project Proposal and Team Contract (10%)

You should first find a biological problem that you would like to address. Is there a particular species that interests you? A human disease you would like to investigate? Something from the class that we have already done that has piqued your interest?

You should write a project proposal of at most one page. This proposal should:

- clearly state the scientific problem that your project will aim to address;
- explain why your project is interesting scientifically and computationally;
- discuss briefly your approach to achieving this goal and why you think this is feasible;
- identify the precise scientific effort that you wish to replicate, and which parts of that effort you plan to focus on;
- verify that any data or external resources you will use for the project are available.

It's fine if some of these change a bit during the course of the project, but the proposal should make an attempt to plan out the project. This will help us guide you to shape the project to be successful and avoid a scope that is either too narrow or too broad.

You should also append a completed team contract to the end of your project proposal; please see Canvas for a template document to use for this purpose. We use team contracts because they are backed by research showing that they make for more productive teams that enjoy working together.

If there is any issue with your project as proposed (e.g., see "Potential Pitfalls"), then you will have one opportunity to resubmit a revised proposal.

Deliverable 2: TA check in (5%)

You should meet with your group's assigned TA to discuss your progress and identify any areas of focus moving forward.

Deliverable 3: Progress Report (10%)

You should provide a 1-page description of the progress you have made so far on your project, describing what you have done, what you plan to do in the remaining time, and identifying any problems you are encountering. You should also indicate what individual contributions each member of your team has made.

Deliverable 4: Rough Draft (0%; optional)

Approximately two weeks before the end of classes, you may wish to submit a rough draft of your essay to receive feedback in advance of your final presentation and essay. The rough draft is not required, but we suggest taking advantage of it.

Deliverable 5: Presentation (25%)

A short (maximum 10-minute) in-class presentation to your peers during the last week of the semester explaining what you have done and demonstrating your results. You should explain your work and demonstrate your results to an audience of your peers who are intelligent but unfamiliar with your project.

Deliverable 6: Final Essay and Explanation of Work (50%)

A minimum 10-page essay explaining the technical background of the problem and the algorithm(s) needed to address the problem. The write-up should be clearly written, self-contained and contain citations to relevant previous work. Any algorithms you implement or software that you run should be fully explained on a high level (i.e., do not resort to pasting your code). Then, include an analysis of applying the computational technique to a real dataset. The component of your grade allocated to the written component will be partly graded based on the quality of your exposition and partly graded based on the quality of your scientific work; the latter will be weighted double.

As part of the final essay, each member of your team should submit their own document indicating what each member of the team did in the course of the project. This document is ungraded, but receiving a grade on the final project is contingent upon your submitting this document.

Important Dates

- **Monday, February 12:** due date for establishing project team members (you may form your own groups on Canvas)
- **Monday, February 26:** due date for project proposal along with team contract
- **approximately Monday, March 18:** meetings with TAs
- **Monday, April 1:** due date for written project progress report
- **Wednesday, April 17:** due date for (optional) rough draft
- **Monday, April 22 and Wednesday, April 24:** in-class presentations
- **Wednesday, May 1:** due date for final essay with results included

Potential Pitfalls

In my experience, I have observed a few pitfalls associated with student group projects, and with this project in particular.

With respect to group work in general, research has shown that teams are successful when they establish a clear team contract and meet frequently and regularly; I would suggest meeting at a minimum of once a week on a set schedule, with a clear set of synchronous and asynchronous aims each week. It is also helpful to have clearly defined roles. I would suggest the following three roles as a starting point with respect to meetings: meeting organizer, meeting recorder, and deadline enforcer. You may wish to define your own roles along the way, as well as rotate these roles. Feel free to let me know if I can provide any input! Finally, project groups tend to underestimate the amount of time required to complete a successful project; this is why we have three deliverables along the way to ensure that you are on track.

With respect to this particular project, it is critical to make sure that any source of data that you want to use exists. Authors of papers are often not obligated to publish the data that they used to reach a conclusion, and a few repositories of "public" data require too long to obtain approvals to obtain (e.g., full human genomes, or raw sequencing reads for RNA-sequencing experiments). Fortunately, we live in an era of very plentiful open data, but you should make sure that your particular project has a publicly available dataset.

If you are using existing software, then it is vital that you are able to download and run this software. This sounds obvious, but not all academic software is made alike, and some less commonly used software is very difficult — if not impossible — to install and use. Once you identify a resource that you would like to use, please make sure that you are able to install it and run it on an example dataset.

Finally, I sometimes have to work hard to dissuade students from trying to complete a project that is essentially novel. Do not think that I expect you to complete a research project from scratch! Often, a fantastic starting point for a scientific endeavor is to take what someone else has done, grab their data, and see if you can replicate it, which explains the format of this assignment.

Grading Guidelines

Some guidelines for the presentation and essay to keep in mind when completing the project are as follows. We will not necessarily strictly follow these guidelines in grading, but they are all helpful things to consider and ensure that you are on track.

Essay: Written Quality

- Is there a clearly written introduction that explains the scientific problem for a lay audience?
- Are all figures clearly explained via captions and referenced appropriately from the main text?
- Is the essay structured logically with a clear flow from the beginning of the article to the end?
- Is an abstract provided that clearly articulates the high-level aims and results?
- Does the quality of the exposition accurately reflect the ability of an SCS undergrad?

Essay: Quality of Scientific Work

- Does the essay have the requisite length?
- Are any computational problems addressed in the project very clearly formulated?
- Are the key algorithms used explained on a high level for a wide audience without resorting to copying code into the document?
- Is there a thoughtful results section that is interpreted in the context of the scientific problem introduced and whose results are explained without resorting to appealing to a figure or dataset?

Presentation

Note: Some of the items below are taken directly from the"3 Minute Thesis" competition guidelines (`https://library.cmu.edu/3mt`).

- Does the presentation end on time?
- Does the presentation provide an understanding of the background to the research question being addressed and its significance?
- Does the presentation clearly describe the key results of the research including conclusions and outcomes?
- Does the presentation follow a clear and logical sequence?
- Does the speaker convey enthusiasm for their project?
- Does the speaker capture and maintain their audience's attention?
- Does the speaker avoid scientific jargon, explain terminology and provide adequate background information to illustrate points?
- Does the speaker have sufficient stage presence, eye contact and vocal range; maintain a steady pace, and have a confident stance?
- Does the speaker spend adequate time on each element of their presentation, or did they elaborate for too long on one aspect or was the presentation rushed?
- Do the slides enhance the presentation? Are they clear, legible, and concise?

Sample Projects

The project is deliberately open-ended; you may choose to complete a project on any aspect of computational biology that you find interesting. However, we are providing some off-the-cuff topics that we brainstormed below. You may be interested in one of these, or you may like to pick your own! It is also acceptable if more than one group completes a similar project.

Furthermore, it is perfectly reasonable to begin one's project by examining existing research. In fact, in some cases, replication of an existing paper may be significantly challenging to constitute a project.

There are also many potential SARS-CoV-2 projects of course, although the list below is free of them.

- Compare the quality of the output of a family of different software programs for genome assembly on different types of organisms and read sets.
- Use genotyping data to identify the population structure of a species (e.g., humans) and identify admixture in individuals.
- Classify a large family of cellular or medical images using deep learning with Tensorflow or PyTorch; in the case of medical images, extend this approach to provide an automated diagnosis of a patient based on their image(s).

- Develop a pipeline that quantifies the alpha and beta diversity of metagenomics samples and maps reads against a database.

- Implement an algorithm that will design primers for testing the presence of an arbitrary virus within an individual.

- Apply RNA sequencing to differentiate cells taken from the same tissue in different organisms, or from different tissues in the same organism.

- Build and analyze a gene co-expression graph from RNA-sequencing data.

- Build a transcription factor network connecting transcription factors to the genes they regulate, and infer what biological conclusions can be drawn from the properties of the resulting graph.

- Apply game theory to explore evolutionary dynamics.

- Construct a family of evolutionary trees (perhaps using multiple tree construction algorithms) for a variety of multiple alignments on the same collection of taxa to obtain a collection of "gene trees"; then, design an approach that reconciles these differing trees into a single "species tree" for the collection.

- Extend the evolutionary tree model to handle recombination/horizontal gene transfer (e.g., in influenza viruses).

- Find patterns across human microbiome data in five different tissues for 300 different humans as part of the Human Microbiome Project (`https://hmpdacc.org`). Or, identify which bacteria (or lack thereof) may be implicated in certain diseases.

- Analyze The Cancer Genome Atlas expression data (`https://cancergenome.nih.gov`) to find which expression patterns are associated with differing cancer types.

- Mimic an algorithm from within nature to solve a computational problem.

Bioinformatics Software Resources

If you choose a project on a topic that we have already covered in the course, you may find the following list of software resources useful.

- *De novo* genome assemblers: `https://en.wikipedia.org/wiki/De_novo_sequence_assemblers`

- Sequence alignment software (pairwise/multiple alignments, profile HMMs, read mapping, metagenomics aligners, etc.): `https://en.wikipedia.org/wiki/List_of_sequence_alignment_software`

- Gene prediction software: `https://en.wikipedia.org/wiki/List_of_gene_prediction_software`

- Phylogenetics software: `https://en.wikipedia.org/wiki/List_of_phylogenetics_software`

- RNA Sequencing Software: `https://en.wikipedia.org/wiki/List_of_RNA-Seq_bioinformatics_tools`

Project Ring of Honor

I am providing a few examples of exemplary student projects at http://compeau.cbd.cmu.edu/home/teaching/great-ideas-in-computational-biology/. There is no such thing as a perfect project, but all of these projects are superlative.

External Resources

We require a project proposal because we want to make sure that you are on the right track. If you struggle with the scientific component of your project, please let us know so that we can help.

To help with your written and oral communication skills, the university has a central service called the Global Communication Center (GCC). You should make use of their services throughout your time at CMU; check them out at https://www.cmu.edu/gcc/.