Homework 1

CS 1810, Fall 2022

Out: Sept. 16
Due: Sept 29th, 11:59 PM EST

Please upload your solutions on Gradescope. You can use LaTeX or a word document to write up your answers, but we prefer you use LaTeX. You may scan hand-written work or images for parts of solutions **only if** they are extremely clean and legible. Please ensure that your name does not appear anywhere in your handin.

Problem 1: Global alignment table

It is feeding time at the CS181 Aquarium and the animals are hungry! The walrus is especially hungry and is excited to eat his favorite snack, the nautilus (a type of marine invertebrate). After feeding the walrus, the computational biologist decides to perform a global alignment on `NAUTILUS` and `WALRUS`!

Create and fill out the global alignment dynamic programming table, with scores and back pointers, to align `NAUTILUS` and `WALRUS`. For scoring, use the match bonus $+1$, mismatch penalty -1, and indel penalty -1. When there is a tie between a mismatch and an indel alignment, use a back pointer that indicates a mismatch. What is the score of the optimal alignment and to which alignment does this score correspond?

Problem 2: Multiple alignment

In class, you have learned about the global alignment of two strings. A natural question to ask is how we might go about aligning three strings. Your task is to design an efficient algorithm that solves the following problem.

> GLOBAL ALIGNMENT OF THREE STRINGS
>
> **Input:** Strings u, v, and w and a scoring function δ that takes in three arguments.
>
> **Output:** An alignment of u, v, and w for which the score is maximal (as defined by δ) among all alignments of u, v, and w.

Whereas we had to work fairly hard to build up the original binary global alignment algorithm from scratch, it turns out that we can obtain an algorithm for the global alignment of three strings by making some careful modifications to the algorithm you've seen in class.

In your solution, you *must*

- describe how the structure of the dynamic programming table changes in the extension to three strings,
- give a new recurrence relation for finding the score of an optimal alignment of a given combination of prefixes of u, v, and w,
- specify the order in which the new dynamic programming table should be filled out.

If you wish, you may also include diagrams, pseudocode, mathematical expressions, and plain English text in your answer. However, please do not submit a block of code with no accompanying explanation.

Once we know how to extend our global alignment algorithm to three strings, we might consider making the general extension to k strings:

- Show that if we continue to insist on constructing dynamic programming tables, the runtime of our algorithm will be $O((2n)^k)$ for k sequences of length n. Since this approach is exponential in k, we will quickly hit a computational wall in attempting to align sequences of even modest length.

  Hint: How many cells are in the new dynamic programming table and how many operations are in the new recurrence relation?

Bonus: Sketch in a few sentences a reasonable heuristic we might use to skirt this problem if we would still like to align multiple sequences.

Problem 3: Counting global alignments

When implementing algorithms, there are different approaches to solving a problem. Brute force programming is a direct approach to solving a task that depends more on computer processing power and memory, while dynamic programming optimizes by breaking a task into smaller problems and drawing upon solutions to the smaller problems in order to solve the overall task.

A brute force approach to global alignment would iterate through every possible alignment of the two input strings, computing the score and storing the max along the way. To understand the performance of such an approach, we need to know the number of possible alignments between two strings, one of length m and the other of length n. Finding an exact expression for the number of possible alignments is actually fairly difficult. In this problem, we simply ask you to determine whether the number of alignments is

1. polynomial in m and n
2. logarithmic in m and n
3. exponential in m and n

Indicate your answer choice by number and justify your response.

Problem 4: A Fowl Virus

To keep the animals in the CS181 aquarium happy, the TAs wanted to experiment with some new food sources. Their research has revealed that feeding the animals chicken would promote their physical health. To keep up with the demands of all the hungry animals, they started an in-house chicken farm: however, the chickens have started dying at alarming rates! They recruit you, a seasoned computational biologist, to get to the root of the problem. You realize that there is a viral epidemic ravaging the farmer's coops. Luckily, you happen to live next to a sequencing center. You manage to isolate some of the virus and sequence it. Now, with the sequence in your flash drive, it is up to you to save the chickens for the aquarium!

Here, you'll find the sequence for the terrible virus plaguing the fowl. You know that the virus is a retrovirus and is inserting genes into the poor hens' DNA. It's up to you to find out what gene or genes are causing this farmhouse mayhem. Fortunately, you know about BLAST (Basic Local Alignment Search Tool). Navigate to the BLAST website. You're only interested in your chicken's genome so type "chickens" into the species search box and click "search." Now, to find out what gene the virus is infecting your chickens with, copy and paste the viral genome into the box asking for a FASTA sequence and click "BLAST." It will take a short time (1-2 minutes) before NCBI returns your query. Once it does, investigate the alignments it returns and try to get to the bottom of this plague.

a. Figure out what gene your chickens are being infected with and what disease is killing them as a result. Specifically, make sure you check out the alignment for chromosome 20!

Hint: Clicking the "CDS Feature" checkbox in the "Alignments" tab will list genes or proteins coded within each range of the alignment, and selecting "GenBank" for any particular range will provide you with more information about the features found there. If you're not sure whether a feature is significant, try looking up some of the individual words in that feature!

Be sure to provide justification for your answers. The virus itself has an interesting history; if you're interested, try to figure out what virus it is exactly.

Bonus: The general structure of a retrovirus genome is composed of coding regions for the gag-pol-env polypeptides (NOT proteins) and other proteins that assemble the polypeptides into proteins. Using this information, find a closely related virus for the virus above. Give a short description of how you found the related protein.

Problem 5: DNA Sequencing

DNA sequencing has become indispensable to biological research, enabling us to inspect genomes like those you'll see in this class. With constant innovation, this technology has become more readily available, cheaper, and quicker. There are various companies that provide sequencing technology- arguably the most prevalent is Illumina:

> "Illumina is a leading developer, manufacturer, and marketer of life science tools and integrated systems for large-scale analysis of genetic variation and function. These systems are enabling studies that were not even imaginable just a few years ago, and moving us closer to the realization of personalized medicine."

Read Illumina's beginner's guide to next-generation sequencing.

a. In five or less sentences, explain the process of next-generation sequencing.

b. Considering logistics and the everyday decisions of a researcher, give at least 3 specific examples of how human bias or error may impact sequencing output. Possible areas of focus include:

- Workflow Development
- Sample Collection
- Library Preparation
- Sequencing
- Data Analysis
- Handling and Communicating Results

Problem 6: SRC Analysis Toolkit

Ethics in computational biology is of growing importance as many of you continue down career paths in industry, research, medicine, or other related fields. As such, we hope that as you respond to the SRC material (and further down the line in your future careers), that you carefully consider the impacts of data usage and privacy, algorithm development, and modeling accuracy/error. In this section, we hope to introduce a toolkit for analyzing the ethical issues involving computational biology. This will include (1) introducing you to frameworks of ethical analysis and (2) reflecting on the potential considerations that go into constructing computational models that reflect biological systems.

Part 1: Introduction to Ethical Theories

Read this introduction to ethical theories.

a. In 3-4 sentences, compare and contrast the different approaches to evaluating whether an action is ethical?

Part 2: Socially responsible analysis of biological algorithms and models

Read the following document explaining how to approach evaluating biological algorithms in SRC questions.

Definitions: Homology - (*evolutionary biology*) A state of similarity in structure but not necessarily in function between different organisms indicating a common ancestry or evolutionary origin. (*genetics*) A condition denoting to the pair of chromosomes having corresponding genes for a particular trait or characteristic

b. Often, we utilize sequence alignment to get a sequence similarity score (i.e. percent of the DNA nucleotides or amino acids that align exactly). The purpose of the sequence similarity is to infer homology. We infer homology when two sequences or structures share more similarity than would be expected by chance; when excess similarity is observed, the simplest explanation for that excess is that the two sequences did not arise independently, they arose from a common ancestor. Given an example sequence similarity of 18%, how would you interpret the relationship between the two sequences? Can we definitively say whether these sequences are homologous or not? Why or why not?

c. Imagine that you are a lead computational biologist who has developed an algorithm that predicts whether an individual will experience a severe case of COVID-19 based on their lung function. Brainstorm three ways (1-2 sentences each) that you can ensure your algorithm is both biologically accurate and ethically conscious if universally implemented. (Hint: consider the guiding questions in the socially responsible analysis document.)