CISC471 - Homework 5

Hershil Devnani (20001045)
Rayan Shaikli (20059806)

March 26, 2021

1 Programming

Please refer to the submitted python files and README.md for the code implementation of Part 1.

1.1 Implement ConvolutionCyclopeptideSequencing

The *ConvolutionCyclopeptideSequencing* problem has been implemented in *convolution_cyclopeptide_sequencing.py*. The solution to the problem, as describes in the videos and text book chapter four, was a build up from a collection of smaller problems that come together at the end to find a sequence of amino acid masses. This sequence of amino acid masses represents a potential amino acid sequence of a peptide whose spectrum has been experimentally determined using a mass spectrometer. From the spectral analysis of a peptide, and the experimental spectrum given to us via the mass spectrometer, we can build and score amino acid sequence to come up with a potential amino acid sequence that fits the experimental spectrum and can be the basis for an initial sequencing analysis of a given peptide.

These smaller problems were available on Rosalind as well (problem sets ba4c, ba4f, ba4g, ba4h) which were implemented throughout the code and required for the final problem in implementing *Convolution-CyclopeptideSequencing*. Our code for these smaller problem sets works flawlessly with Rosalind data sets, providing us a confident basis to bring these prices together for the larger, and final problem.

Some things we observed after implementing of our algorithm was that there can be multiple, high scoring amino acid sequences, for a given experimental spectrum. Given that many amino acids have similar masses and amino acids sequences of varying lengths and combinations of amino acids can score similarly, we are never restricted to the single output from this algorithm, there are a lot of possibilities for an potential amino acid sequence. This is a very interesting problem, but there is no one correct answer, as implied by the output of the algorithm on Rosalind suggests a sole correct output/answer.

Our algorithm produces a high scoring result that matches the constraints of the problem and the sum of the amino acid sequence masses matches the sum of the parent mass from the experimental spectrum. Our solutions don't always agree with what is given as a correct solution by Rosalind data sets and solutions. In addition, given the fact that we know there can be multiple solutions off a given experimental spectrum, we believe that our algorithm outputs a quality potential amino acid mass sequence, that score high and sums to the parent mass of the experimental spectrum.

These observations, along with the fact that the sub-problems solved in order to complete the overall problem consistently give correct results off of every Rosalind data we tested, we are quite confident that our program's output does produce one of many quality and correct results. As a result, the fact that some of our output's being marked incorrectly on Rosalind may be due to a tight range, or singular output, that Rosalind matches against the specific implementation of their algorithm, when in fact there is a very wide range of potential solutions that all score very high and match the criteria for being a candidate for an amino acid sequence for a peptide, given it's experimental spectrum.

2 Theory

2.1 Question 1

Lesson 5.2: Exercise Break: Find all longest common subsequences of the strings ACTGCA and CATCGC. How many such subsequences did you find?

The longest length subsequence found was of length 4. The following are the 3 determined subsequences:

1. ACGC
   (a) **AC**T**GC**A
   (b) C**ATCGC**
2. ATGC
   (a) **AC**T**GC**A
   (b) C**AT**C**GC**
3. CTGC
   (a) A**CTGC**A
   (b) **CA**T**CGC**

In order to conduct an a search for the longest common sub-sequence of the given strings, we can essentially conduct an alignment between the two strings. The longest common sub-sequence problem, in effect, become an alignment problem, since the longest common sequence is the subset of nucleotides that match given an alignment of contigs. As a result, the nucleotides that match may not be found consecutively in the contigs, but will be found in the same order.

2.2 Question 2

Lesson 5.3: Exercise Break: How many different paths are there from source to sink in a 16×12 rectangular grid?

Convention: According to the Manhattan Tourist problem from the textbook, a 16 x 12 grid represents 17 rows and 13 columns, which is equal to 221 potential nodes we can pass through along the way from the source to the sink. A 16 x 12 grid also indicates that we have to take 16 step in the right direction and 12 steps in the down direction to travel from the source to sink.

There are 16 ways we can move to the right and 12 ways we can move to down. So we need a combination of 16 moves to the right and 12 moves down, for a total of $16 + 12 = 28$ moves we need to make.

Therefore, the number of ways we can move 16 steps to the right and 12 steps down is:

$$\binom{28}{16} = \binom{28}{12} = 30421755 \tag{1}$$

2.3 Question 3

Lesson 5.4: Exercise Break: Construct the alignment of ATGTTATA and ATCGTCC corresponding to the alignment path shown in Figure 1.

Analyzing this problem, from the start to the finish, we move from the top left to bottom right. The solution provided above takes into consideration correct matches (red diagonal arrow ↘), mismatches (violet diagonal arrow ↘), insertions (blue right arrow →), and deletions (teal down arrow ↓). For this reason, the graphs that we will consider in this chapter do not contain directed cycles; such graphs are called directed acyclic graphs (DAGs).

This Manhattan Tourist Problem can be applied to performing alignments, where we can represent an alignment between two contigs as a DAG, with the path we take, and always results in a valid alignment if we start with marking diagonals that match. This becomes the alignment path of the two contigs, and we can use a DAG to represent an alignment, or to reconstruct and alignment given the DAG, as we have done in this problem.

The following is the alignment of the two contigs, ATGTTATA and ATCGTCC corresponding to the alignment path shown in Figure 1 of the assignment outline:


A T G T T A – T – – A
– – A T – C G T C C –
↓ ↓ ↘ ↘ ↓ ↘ → ↘ → → ↓