COSC348: 1st Assignment   Due: 14/Aug/2015   14 marks

Finding frequent and interesting patterns in RNA sequences

The RNA sequence that corresponds to a gene has a "header" and a "trailer", which do not get translated into protein. The technical term for them is 5'-UTRs and 3'-UTRs (UnTranslated Regions), respectively. These regions are used to decide which genes are transcribed and when. Biochemists would like to find common "switches" that are frequently used. This suggests looking for common substrings in a collection of strings. However, if every instance of a common switch was the same, all those genes would turn on or off at the same time. So what we really expect to see are many instances of *similar substrings*, and what we need to do is to find these *common patterns*.

An RNA base is one of A, C, G, or U. There is one wildcard character, i.e. '?' that matches any of A, C, G, or U. Caution: '?' is NOT a fifth character!

DEFINITION: A *pattern* of length $k \geq 3$ is a sequence of characters consisting of:

- a base
- k-2 bases and wildcards, **any mix**
- a base

For example, AC?G?U is one of the patterns for $k = 6$. Another pattern will be: G???AC, etc. Once again, '?' is NOT a fifth character, but it stands for any of A, C, G, or U.

Given a collection of n RNA sequences and a real number $p \in (0,1]$, we say that a pattern is *frequent* in the RNA collection if at least pn sequences contain at least ONE match of the pattern. We do not care how many matches there are for a pattern in a particular sequence, only whether it has at least one match. An *interesting* pattern will be a particular *fixed* k-tuple, e.g. ACGGCU in pn sequences, at least once in each of them.

Note: If you wish, you can call/consider the frequent patterns as variable patterns, and the interesting patterns as fixed patterns.

To complete this assignment you can use any programming/scripting language you want and any algorithms you want. Hand in the written report and submit ***electronically***: the report itself, your codes and their compiled executable files for all the tasks with the README file how to use them.

The report must contain answers to the following TASKS:

(Task 1) There are 2 subtasks:

(a) Provide the list and number of all *frequent (variable)* patterns for k = 3, 4, 5, 6 and 9.
(b) Provide the list and number of all *interesting (fixed)* patterns for k = 3, 4, 5, 6 and 9.

Collection of n = 1000 real RNA sequences given one per line can be found here: http://www.cs.otago.ac.nz/cosc348/alignments/RNAseqs.txt

Note that RNA sequences have different length, i.e. different number of bases. Take p = 0.5.

In the written report ***describe the algorithm(s),*** which you have used in your program and provide the table with the ***total numbers*** of variable and fixed patterns for k = 3, 4, 5, 6 and 9. Include a ***list*** of these patterns in separate file(s). (10 marks for Task 1)

(Task 2) There are two subtasks:

(a) First, estimate experimentally the running time of your algorithm as a function of varying k = 3, 4, 5, 6 and 9, for the total length of the RNA sequences, i.e. the total number of letters in the input file. This can be done at the same time as Task 1. Derive the theoretical formula of the running time of your program as a function of k, e.g. time = exp(k).

(b) Second, estimate experimentally the running time of your program as a function of the varying total number of the RNA characters N, for the fixed k = 4. In other words, you will run the program for k=4 and vary the number of RNA input characters. First, you'll take ¼ of input characters then ½ and then ¾, and finally the time for the whole set (which you already have). Write all these run times into the tables or produce graphs with axes clearly marked with numbers. Derive the theoretical formula of the running time of your program as a function of the total length of RNAs, e.g. time = k * N. (4 marks for Task 2)