CSCI3220 Algorithms for Bioinformatics 2018-2019 First Term
The Chinese University of Hong Kong

Assignment 3

Due date: 11 November 2018 (Sun) 23:59 Full mark: 100

Expected time spent: 3-6 hours

Aims:
1. Get familiar with the details of sequence similarity calculations based on *g*-gapped *k*-mers, basic probability derivations, and HMM algorithms.
2. Experience the difference in time complexity of different algorithms.
3. Generalize some specific concepts studied in lectures.

Description:

In this assignment, you will use different ways to compute sequence similarity based on *g*-gapped *k*-mers, to experience their relative. You will then perform some probability derivations using the three basic rules, and solve some generic problems. Finally, you will implement the forward algorithm for computing data likelihoods according to a first-order hidden Markov model, in a way that your program can handle any type of sequences.

Questions:

1. This question is about *g*-gapped *k*-mers. Suppose we want to compute a similarity score between DNA sequences s_1=`GCGTCCGAC` and s_2=`CGACGCGAC` based on 1-gapped 3-mers (i.e., g=1, k=3).

   (a) How many different 1-gapped 3-mer patterns are there for DNA sequences? Show the steps in your calculation. (3%)

   (b) List all the 1-gapped 3-mers actually supported by s_1 and s_2 and their occurrence counts in the two sequences by filling in the following tables (add more rows if needed). The tables should be sorted in ascending order of the 1-gapped 3-mers, where the wildcard character `*` is ordered before the four nucleotides. (12%)

   Table for s_1

1-gapped 3-mer	No. of occurrence

   Table for s_2

1-gapped 3-mer	No. of occurrence

   (c) Based on your tables in Part b, compute the similarity between the two sequences, defined as the inner product of the two 1-gap 3-mer occurrence vectors. Show clearly the common 1-gap 3-mers of the two sequences in your calculations. (3%)

   (d) Now complete the following table listing the number of mismatches among the 4-mers in the two sequences. The row and column headers should be the 4-mers present in s_1 and s_2, respectively, both sorted ascendingly. If a 4-mer appears multiple times in a sequence, repeat it that number of times in the row/column headers. (8%)

s_2 s_1						

(e) Based on your result in Part d, compute the similarity score between s_1 and s_2 again. Show the steps in your calculation. Do not forget to compare with your result in Part c. (4%)

(f) In practice, it is not easy to decide which values of g and k to use. One possible approach is to try multiple combinations of these values, and use some independent knowledge to evaluate which combination leads to the best similarity scores. Propose two ideas for reducing the total computational time as compared to repeating the above calculations for each combination of g and k values. (6%)

2. This question is about statistical modeling. Suppose Y is a binary variable, and we want to construct a model of Y based on binary features X_1 and X_2, which may not be conditionally independent.

(a) Derive a formula for computing $\Pr(Y=0|X_1=1,X_2=1)$ in terms of $\Pr(X_1=1)$, $\Pr(X_1=1,X_2=0)$ and $\Pr(X_1=1,X_2=1,Y=1)$. Show your derivations step by step. (6%)

(b) [Optional] In general, with three variables X_1, X_2 and Y, there are various possible probabilities of the form $\Pr(\mathrm{V_L}=\mathrm{v_l}|\mathrm{V_R}=\mathrm{v_r})$, where $\mathrm{V_L}$ and $\mathrm{V_R}$ are two disjoint ordered sets of variables (and $\mathrm{V_R}$ can be empty), and $\mathrm{v_l}$ and $\mathrm{v_r}$ are their corresponding values. For example, in the case of $\Pr(X_1=1)$, $\mathrm{V_L}=(X_1)$, $\mathrm{v_l}=(1)$, and $\mathrm{V_R}=\varnothing$. In the case of $\Pr(Y=0|X_1=1,X_2=1)$, $\mathrm{V_L}=(Y)$, $\mathrm{v_l}=(0)$, $\mathrm{V_R}=(X_1,X_2)$, and $\mathrm{v_r}=(1,1)$.

Propose an algorithm that can take one of these probabilities as target (such as $\Pr(Y=0|X_1=1,X_2=1)$) and some of these probabilities as input (such as $\Pr(X_1=1)$, $\Pr(X_1=1,X_2=0)$ and $\Pr(X_1=1,X_2=1,Y=1)$), and determine whether the input is sufficient for inferring the value of the target. (bonus 10%)

(c) Now assume X_1 and X_2 are independent when conditioned on Y. Give a formal definition of this assumption using mathematical symbols. (3%)

(d) Depending on the values of X_1, X_2 and Y, there are 8 probabilities of the form $\Pr(X_1,X_2|Y)$. Given an example set of values for these 8 probabilities such that X_1 and X_2 are not independent when conditioned on Y. Prove that the two variables are not conditionally independent. (5%)

(e) If two variables X_1 and X_2 are independent when conditioned on Y, is it guaranteed that they are also unconditionally independent? Explain why or why not. (5%)

(f) If two variables X_1 and X_2 are unconditionally independent, is it guaranteed that they are also independent when conditioned on Y? Explain why or why not. (5%)

3. Write a computer program called **Forward** in C, C++, Java or Python that implements the forward algorithm to compute the data likelihood of a given sequence based on a first-order hidden Markov model. This program should be able to handle any number of states and any number of emission symbols. Your program should take the following inputs in the specified order, each on a different line:

- The number of states (an integer)
- The initial probabilities of the states (floating-point numbers, one per line), in the order of π_1, π_2, …

- The transition probabilities among the states (floating-point numbers, one per line), in the order of p_{11}, p_{12}, …, p_{21}, p_{22}, …
- The number of emission symbols (an integer)
- The emission symbols (characters, one per line), in the order of b_1, b_2, …
- The emission probabilities (floating-point numbers, one per line), in the order of $e_1(b_1)$, $e_1(b_2)$, …, $e_2(b_1)$, $e_2(b_2)$, …
- The sequence the likelihood of which is to be computed (a string)

You can assume the input data are properly formatted and do not need to check for errors.

The output should be a single floating-point number of the likelihood value. Due to the inexact nature of floating-point numbers, when we check your answers a small amount of leeway may be considered.

The non-comment portion of your program is expected to contain no more than 200 lines of code.

Here is an expected screen shot when a Java program is run on an example from the lecture notes:

```
>java Forward
2
0.5
0.5
0.9
0.1
0.8
0.2
4
A
C
G
T
0.5
0.5
0
0
0.25
0.75
0
0
CAC
0.15156250000000002
```

Your program will be graded based on i) whether it can be compiled/run successfully, ii) whether it follows the input/output formats as specified above, iii) its accuracy on a number of test cases and iv) whether the program is well-documented with appropriate comments added to explain the meaning of the code. (40%)

Submission:

For the programming question, you should first submit your program to our **online judge system**. Please see tutorial notes set 1 for explanations.

For the non-programming question, give all your answers in a single file named <ID>_asmt3.<ext>, where <ID> is your student ID and <ext> is either doc, docx or pdf. We prefer pdf files because it has better portability. Then put your files for both the programming and non-programming questions in a zip file named <ID>_asmt3.zip and submit it to Blackboard.

Both your written and source code files should contain the following header. Contact Kevin before submitting the assignment if you have anything unclear about the guidelines on academic honesty.

```
CSCI3220 2018-19 First Term Assignment 3

I declare that the assignment here submitted is original except for source
material explicitly acknowledged, and that the same or closely related material
has not been previously submitted for another course. I also acknowledge that I
am aware of University policy and regulations on honesty in academic work, and
of the disciplinary guidelines and procedures applicable to breaches of such
policy and regulations, as contained in the following websites.

University Guideline on Academic Honesty:
http://www.cuhk.edu.hk/policy/academichonesty/

Student Name: <fill in your name>
Student ID  : <fill in your ID>
```

Marking Scheme and Notes:

1. Remember to submit your assignment by 23:59pm of the due date. We may not accept late submissions.

2. For the written part, if you submit multiple times, **ONLY** the content and time-stamp of the **latest** one before the submission deadline will be considered. For the program, the version submitted to the online judge system with the highest score will be graded.

University Guideline for Plagiarism

Please pay attention to the university policy and regulations on honesty in academic work, and the disciplinary guidelines and procedures applicable to breaches of such policy and regulations. Details can be found at http://www.cuhk.edu.hk/policy/academichonesty/. With each assignment, students will be required to submit a statement that they are aware of these policies, regulations, guidelines and procedures.