Bioinformatics

Example Sheet 3

Petar Veličković

Michaelmas Term 2016

Genome assembly

1. In the *genome assembly* problem, we augment the previously covered sequencing framework with an additional *reference genome*. In what way does this aid sequencing?

2. Explain the basic operations supported by the trie data structure and their complexities. Highlight how it can be deployed into the problem of genome assembly in different ways. Your answer should provide a discussion of *tries*, *suffix tries* and *suffix trees*.

3. Explain how the methods of the Burrows-Wheeler transform (BWT) and run-length encoding (RLE) can be applied to reducing the storage requirements of large genomes. Prove that your scheme is efficiently invertible.

4. How can one efficiently pattern-match to the string obtained by the above? Your answer should contain a discussion of the properties and construction time complexities of *suffix arrays*.

5. Demonstrate all of the techniques outlined in questions 2–4 on the sequence `CATATATAG$`.

6. Implement either a suffix trie, a suffix array or the BWT in a language of your choice, and use it to verify the outcome of applying/constructing it on the same sequence as above.

7. Carefully explain why we might be, in fact, generally more interested in *inexact matchings* of reads to the reference. Outline the *seeding* and BWT approaches to this problem, stating their complexities.

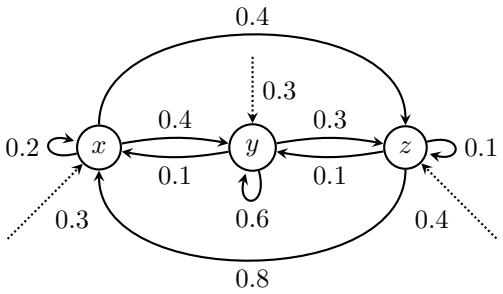
Hidden Markov models

1. Describe the key stateful components of a hidden Markov model (HMM), outlining the differences between it and a Markov chain.

2. Outline the inputs, outputs and time complexities of the following HMM algorithms:

   - Viterbi;
   - Forward;
   - Viterbi training;
   - Baum-Welch.

3. Consider the following three-state HMM[1], modelling an underlying process on DNA sequences (dotted lines represent the probabilities of starting in each of the three states):



   Furthermore, you know the likelihoods of producing each of the nucleotides from each of the states:

	A	T	C	G
x	0.7	0.1	0.1	0.1
y	0.3	0.2	0.4	0.1
z	0.4	0.2	0.2	0.2

   You observed the DNA sequence `CCGAAGTG`.

   (a) What is the most-likely sequence of states that produced it?

   (b) What is the probability that it was produced by this HMM?

   (c) What is the probability that the HMM was in state x when producing the first `G`?

   You may wish to consider implementing some of the required subroutines, rather than computing values by hand.

4. For the following two scenarios, explain (on an abstract level) how you would model the problem using HMMs, and which algorithms would be useful (and in what way):

[1] Setup adapted from a question originally by Sean Holden.

(a) Analysis of transmembrane (located around the cellular membrane) protein secondary structure—namely, for each amino acid of the protein, determining whether it's located *inside the cell*, *inside the membrane*, or *outside the cell*. You are provided with a training set containing transmembrane protein sequences, along with a labelled sequence of the same length, determining the location of each amino acid. You are also aware that any region of a protein within the membrane will consist of at least 5 and at most 25 amino acids.

(b) Classifying patients for presence or absence of a genetic disease, based on their DNA sequences. You are provided with a training set containing DNA sequences labelled as either "patient" or "normal".