UC Berkeley, CS 176: Algorithms for Computational Biology (Fall 2013)

Problem Set 5

Instructor: Yun S. Song

Out: November 14, 2013 Due: 12:00 PM, Dec 9, 2013

HMM implementation

The goal of this problem set is to implement the key algorithms for HMM discussed in class. Throughout, we will consider the following interesting biological application:

Meiotic recombination is an important biological mechanism common to most forms of life. As a consequence of recombination, different positions on the same chromosome may have different genealogical histories. For example, given a pair of homologous sequences, different positions may have different times (denoted T_{MRCA}) to the most recent common ancestor (MRCA), as illustrated in Figure 1. Recently, Li and Durbin (*Nature*, **475**:493-496, 2011) used a hidden Markov model to estimate the position-specific T_{MRCA} for a pair of sequences. The transition and emission probabilities in their HMM arise from a stochastic genealogical process (called the coalescent), which you do not need to know to do the problems described below.

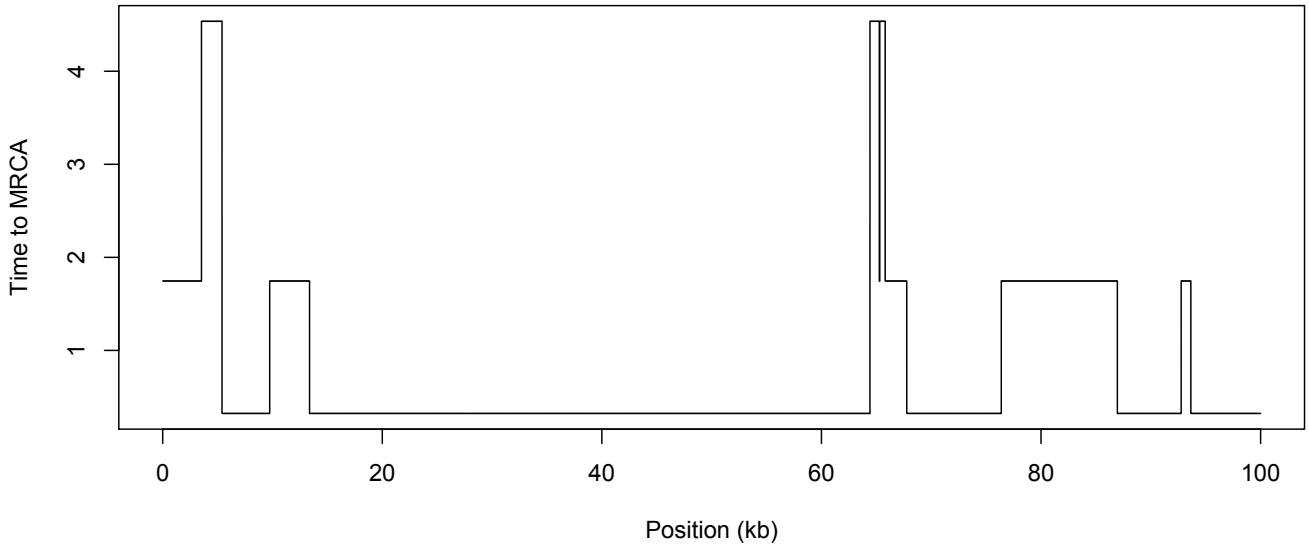

Figure 1: Time to the most recent common ancestor along a pair of homologous sequences, each of length 100 kb. Time is measured in units of $2N_e$ generations, where N_e is the so-called "effective" population size.

Instruction:

- You are strongly encouraged to pair up with a fellow student in class.
- You may use any of the following programming languages: C, C++, Java, Python, Perl, Ruby. Use only the standard libraries for each language.
- You should put all your source code and answers to the questions below into a directory and e-mail us a zipped file. The directory name should be your last name. If you work in a group, use both last names in alphabetical order. (e.g., HardyRamanujan)
- The directory should contain a README file detailing how we can compile AND run your code.

- Download `ps5data.tgz` from the course webpage. Included in the tar archive are sequence files called `sequences_mu.fasta`, `sequences_2mu.fasta`, and `sequences_5mu.fasta`. Each file contains a pair of DNA sequences of length $L = 100,000$ in FASTA format. The three data sets were generated using three different mutation rates, namely μ, 2μ, and 5μ, for some μ. Consider the following HMM:
  - The observed symbol $x_\ell \in \Sigma = \{I, D\}$ at position $1 \leq \ell \leq L$ corresponds to whether the two sequences are identical (I) or different (D) at that position.
  - The hidden state $Q_\ell \in S = \{t_1, t_2, t_3, t_4\}$ at position $1 \leq \ell \leq L$ corresponds to the T_{MRCA} at that position.
  - Assume that the hidden random variables $\{Q_\ell, 1 \leq \ell \leq L\}$ form a homogeneous Markov chain, with transition probabilities a_{ij}, for $i, j \in S$.
  - As usual, the probability of emitting symbol $\sigma \in \Sigma$ from state $k \in S$ is denoted by $e_k(\sigma)$. The parameters of the model are $\Theta = \{a_{ij}, e_i(\sigma), \pi_i\}_{i,j \in S; \sigma \in \Sigma}$, where π_i denotes the marginal probability $\mathbb{P}(Q_1 = i)$.

Remark: We expect $\mathbb{P}(D \mid Q_\ell = t_j) > \mathbb{P}(D \mid Q_\ell = t_i)$, for $t_j > t_i$. Why?

Problems:

1. Implement the forward and backward algorithms.
2. Implement the EM algorithm.
3. For each "`mu`","`2mu`", and "`5mu`" file, do the following (`*` in the file name stands for `mu`, `2mu`, or `5mu`):
   (a) Use the EM algorithm to estimate the parameters Θ of the model. For `sequences_*.fasta`, use the parameters Θ_{initial} provided in `initial_parameters_*.txt` as initialization. Store your estimated parameters $\Theta_{\text{estimated}}$ in a file called `estimated_parameters_*.txt`.
   (b) In `likelihoods_*.txt`, store the log-likelihoods for the initial parameters Θ_{initial} and for your estimated parameters $\Theta_{\text{estimated}}$.
   (c) Using the initial parameters Θ_{initial}, produce both Viterbi and posterior decodings, and compute the posterior mean $\mathbb{E}[T_{\text{MRCA}} \mid \boldsymbol{x}, \Theta_{\text{initial}}]$ for each position. Assume that $S = \{0.32, 1.75, 4.54, 9.40\}$. To identify which hidden state should correspond to which time, think about the remark mentioned above.
      i. Output your results to `decodings_initial_*.txt` in a 3-column format (Viterbi decoding, posterior decoding, posterior mean).
      ii. Plot your results, together with the true T_{MRCA} provided in `true_tmrca.txt`. (In fact, Figure 1 shows the true T_{MRCA} for the data you are analyzing.) Name your figure file `plot_initial_*.pdf`.
   (d) Using your estimated parameters $\Theta_{\text{estimated}}$, produce both Viterbi and posterior decodings, and compute the posterior mean $\mathbb{E}[T_{\text{MRCA}} \mid \boldsymbol{x}, \Theta_{\text{estimated}}]$ for each position.
      i. Output your results to `decodings_estimated_*.txt` in a 3-column format (Viterbi decoding, posterior decoding, posterior mean).
      ii. Plot your results, together with the true T_{MRCA} provided in `true_tmrca.txt`. Name your figure file `plot_estimated_*.pdf`.

Additional exercise (not to be turned in): Try starting the Baum-Welch algorithm with different initial parameter settings. Do you obtain the same final estimates?