ECE-S690-501 Genomic Signal Processing: Modeling Sequences with HMMs Homework Assignment 4

Due on:

May 26, 2009

Introduction

In class, we covered how Hidden Markov Models can address three problems: 1) Determining the probability of a set of observations given model parameters, 2) determining the state sequence that best explains the observations given the observations and model parameters, and 3) changing the model parameters to maximize the posterior decoding. Through this homework, you will learn how to use simple hidden markov models to find CpG islands and coding regions in DNA sequences. **You must turn in your Matlab code in your solutions, or you will receive a 0. Start Now! This homework might be longer than the previous ones.**

1

Use Fig. 1 for the following excercise:

- Calculate the probability of the sequence TAG by following a path through the model starting at Begin, going through each of the three match states (red squares), and ending at End.
- Repeat part 1 for a path that, starting at Begin, goes first to the first insert state (green diamond), then to a match state (red square), then to a delete state (blue circle, probability 1 for any character in this state), then to a match state and finishing at End.
- Which of the two paths is the more probable one, and what is the ratio of the probability of the higher to the lower one? The highest scoring path is the best alignment of the sequence with the model and for real sequences is found by dynamic programming.
- Change all the values in each of the states to log odds scores, assuming that the frequency of each base is 0.25. Also change the transition probabilities to log odds; i.e.,

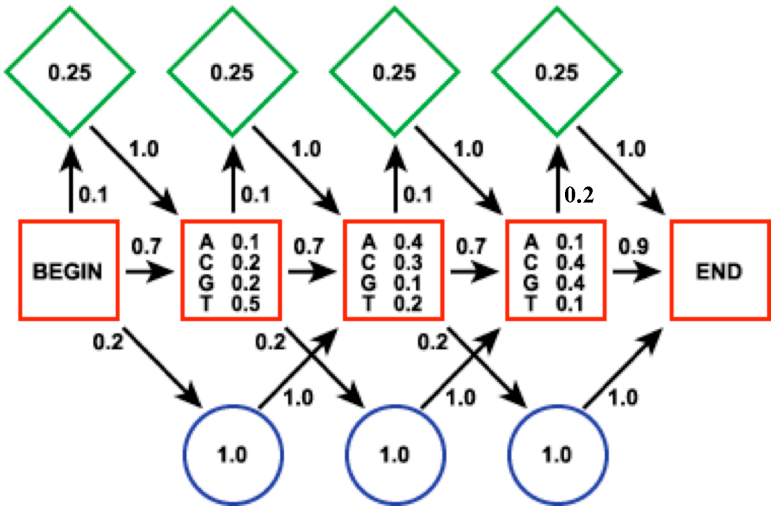

Figure 1: A Markov Model with transition probabilities and probabilities of each state. Red square, match state; green diamond, insert state; blue circle, delete state. Arrows indicate probability of going from one state to the next.

log to the base 2 of the ratio of observed transition probability to background probability. (Note: Transition background is an equal probability of making a transition to each subsequent state and will be calculated by dividing 1 by the number of possible transitions from each state; i.e., the background probability will be one of 1/2=0.5, 1/3=0.33, or 1/1). Now calculate the probability in part 1 as a log odds score.

2

Real DNA sequences are inhomogeneous and can be described by a hidden Markov model with hidden states representing different types of nucleotide composition. Consider an HMM that includes two hidden states H and L for high and lower $C + G$ content, respectively. Initial probabilities for both H and L are equal to 0.5, while transition probabilities are as follows: $a_{HH} = 0.5$, $a_{HL} = 0.5$, $a_{LL} = 0.6$, $a_{LH} = 0.4$. Nucleotides T, C, A, G are emitted from states H and L with probabilities $0.2, 0.3, 0.2, 0.3$, and $0.3, 0.2, 0.3, 0.2$, respectively. **Use the Viterbi algorithm to define the most likely sequence of hidden states for the sequence** $x = GGCACTGAA$.

Use the i=0 initialization states, $V_0(0) = 0$, $V_H(0) = -\inf$, $V_L(0) = -\inf$ and for $i = 1 : L$, use $V_l(i) = log_2(e_l(x_i)) + \max_k(log_2(v_k(i-1)) + log_2(a_{kl}))$. For each i, compute each hidden state, l, parameter: $V_L(i)$ and $V_H(i)$. Remember, k is each hidden state as well that should be computed for each of these (to get the max). The traceback pointer to the hidden state value is the greater of the two. (The Viterbi algorithm is found on p. 56 in Durbin et al.)

3

For the hidden Markov model defined above, **find the probability of the sequence occurring,** $P(x)$, by both the forward algorithm and the backward algorithm (found on pages 57 and 58 in Durbin et al.).

The start-to-hidden-state transitions are equiprobable with $a_{0H} = 0.5$, $a_{0L} = 0.5$. The last hidden state transitions to the end state must occur with $a_{H0} = 1$ and $a_{L0} = 1$. **For the forward algorithm, give the** $P(x_{1:i}) = f_H(i) + f_L(i)$ **for each** i.

4

For the hidden Markov model defined above, find the posterior probabilities of states H **and** L **at the last position of** $x = GGCA$ **and** $x = GGCACTGAA$.

Use the formula:

$$P(\pi_i = k|x) = \frac{f_k(i)b_k(i)}{P(x)} \tag{1}$$

(found on page 59 of Durbin et al.), where $f'_k s$ are from the forward algorithm, $b'_k s$ are from the backward algorithm, and P(x) is the result of the forward/backward calculation.

5

What is the difference between Viterbi decoding and smoothing (π^*) and posterior decoding ($\hat{\pi}_i$)? Write down their mathematical definitions and then explain how they differ.

6

1. In this problem, we will examine part of the Human Chromosome 22 (genbank accession number: NC_000022). Get the first contiguous region. Use the *cpgisland* Matlab command to get all the stretches on the sequence that are CpGIslands. Do not get the ones that are a minimum of 200-length – get the ones that are at least a length of 1.

2. In this first part, we will train HMM parameters given an annotated model.
   Calculate the Transition Matrix for a 1st-order model of first contiguous region of the Human Chromosome 22 sequence. Give your calculated matrix here.
   Calculate the Emission Transition matrix, $Pr(Observed|Hidden\_State)$**, using the CpG Island annotation calculated above as the hidden states and the actual sequence as the observations. Give your calculated matrix here.**

3. Implement the Viterbi algorithm (you may have done this in problem 2). Your Viterbi function must take the observed sequence, the HMM parameters a and e (use the initial states that you used in problem 2), and return the most likely hidden state sequence.

Compare the results of the Viterbi algorithm to the Matlab *cpgisland* (make plots of the CpG content (represent lack of CpG as 0 and the CpG state as 1).

4. Use the Viterbi algorithm to to predict the CpG islands in the test sequences (contigs 4 and 6 of chromosome 22). Compare your predictions against the results of *cpgisland* **Again, make plots of the viterbi output to the** *cpgisland* . How accurate is the Viterbi algorithm (using *cpgisland* as a reference)? How many annotated CpG islands were completely missed by it (false negatives)? There is also a grey area predicted islands may overlap but not entirely coincide with annotated ones; make a decision about how to report these results. Note that exact borders of CpG islands are somewhat arbitrary, and it is possible that some true CpG islands have not been annotated. How accurately are the end points predicted?

5. Now implement and use posterior decoding to make predictions about the locations of CpG islands on contigs 4, and 6. You may need scaling factors to keep the forward and backward probabilities in a numerically good range (p. 78 in Durbin et. al). **Plot the result of the posterior decoding against the Viterbi algorithm AND** *cpgisland*.

6. Compare the predictions based on Viterbi decoding against those made by posterior decoding. Relate your results to the probabilistic interpretation of each method.

7 Bonus

Repeat steps 2,3, and 4 in problem 6 for a 3rd-order HMM (The transition probabilties will now be $Pr(x_i = b|x_{i-1}, x_{i-2}, x_{i-3})$ where x_i is the observation at i and the emission probabilities will be $Pr(x_i = b|\pi_i = k, x_{i-1}, x_{i-2}, x_{i-3})$, where π_i is the state at i (p. 77 in Durbin et al.). **Plot the** *cpgisland*, **1st-order viterbi, and 3rd-order viterbi predictions on the same graph.**