ECES T580 Homework 3

Bhautik (Brian) Amin

Q1.

A. BEGIN->M1: 0.7
MATCH1: 0.5 (T)
M1->M2: 0.7
M2: 0.4 (A)
M2->M3: 0.7
M3: 0.4 (G)
M3->END: 0.9

Total: 0.7*0.5*0.7*0.4*0.7*0.4*0.9 = **0.0247**

B. TAG
BEG->IN1->M1->D2->M3->END
BEG->IN1: 0.1
IN1: 0.25 (Emitting T)
IN1->M1: 1.0
M1: 0.1 (Emitting A)
M1->D2 0.2
D2: 1.0
D2-> M3: 1.0
M3: 0.4 (G)
M3-> END: 0.9

Total Prob: 0.1 * 0.25 * 1.0 * 0.1 * 0.2 * 1.0 * 1.0 * 0.4 * 0.9 **= 0.0002**

C. The more probable path would be path from part A. It has a 0.0247/0.0002 = 137.22 ratio

D. 6.458807

Q2.

Background:

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.

2 states

Initial Probabilities: [0.5, 0.5]

T = aHH = 0.5, aHL = 0.5, aLL = 0.6, aLH = 0.4

Matrix form: [HL x HL]

	H	L
H	0.5	0.5
L	0.4	0.6

Observations: [T, C, A, G]

Emission: 0.2, 0.3, 0.2, 0.3, and 0.3, 0.2, 0.3, 0.2, respectively

Matrix Form: [H, States x L, States]

	T	C	A	G
H	0.2	0.3	0.2	0.3
L	0.3	0.2	0.3	0.2

Use the Viterbi algorithm to define the most likely sequence of hidden states for the sequence:

GGCACTCAA

Parameters: Use i=0

Initialization states: $V_0(0) = 0$, $V_H(0) = -Inf$, $V_L(0) = -Inf$

Also, for i=1:L use $V_L(i) + \log2(e_L(x_i)) + \max_k(\log2(v_k(i-1)) + \log2(a_{kL})$

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

Problem Start:

Let V (Number of States by Number of Observations) be a table that will store probability values calculated. Let E be the mission matrix and T be the transition matrix

For each index: E(state, observation at current index i) * $\max_k$(V(k, i-1) * T(k, state))

Sequence: **GGCACTCAA**

Index (i=0 start), and Letter	Calculation	Result	Max Pointer
0	$V_0(0) = 0, V_H(0) = -inf, V_L(0) = 0$		
1, G	$V_H(1) = \log(E(H,G)) + \log(T_H(1))$ $V_L(1) = \log(E(L,G)) + \log(T_L(1))$	H: -2.736, L: -3.321	H
2, G	$V_H(2) = \log(E(H,G)) + V_H(1) + \log(T_H(2))$	H = -5.474, L:-6.059	H

	$V_L(2) = \log(E(L,G)) + V_L(1) + \log(T_L(2))$		
3, C	$V_H(3) = \log(E(H,C)) + V_H(2) + \log(T_H(3))$ $V_L(3) = \log(E(L,C)) + V_L(2) + \log(T_L(3))$	H = -8.211, L:-8.796	H
4, A	$V_H(4) = \log(E(H,A)) + V_H(3) + \log(T_H(4))$ $V_L(3) = \log(E(L,A)) + V_L(3) + \log(T_L(4))$	H = -11.532 L = -10.947	L
5, C	$V_H(5) = \log(E(H,C)) + V_H(4) + \log(T_H(5))$ $V_L(5) = \log(E(L,C)) + V_L(4) + \log(T_L(5))$	H = -14.000 L = -14.000	L
6, T	$V_H(6) = \log(E(H,T)) + V_H(5) + \log(T_H(6))$ $V_L(6) = \log(E(L,T)) + V_L(5) + \log(T_L(6))$	H = -17.328 L = -16.481	L
7, C	$V_H(7) = \log(E(H,C)) + V_H(6) + \log(T_H(7))$ $V_L(7) = \log(E(L,C)) + V_L(6) + \log(T_L(7))$	H = -19.539 L = -19.539	L
8, A	$V_H(8) = \log(E(H,A)) + V_H(7) + \log(T_H(8))$ $V_L(8) = \log(E(L,A)) + V_L(7) + \log(T_L(8))$	H = -22.862 L = -22.0135	L
9, A	$V_H(9) = \log(E(H,A)) + V_H(8) + \log(T_H(9))$ $V_L(9) = \log(E(L,A)) + V_L(8) + \log(T_L(9))$	H = -25.657 L = -24.487	L

Traceback: HHHLLLLLL

Q3. For the given HMM, find the probability of the sequence occurring P(x), use both forward and backward algorithm. The transition are all equiprobable, [0.5, 0.5]

The last hidden state transition to the end state must occur with [1,1].

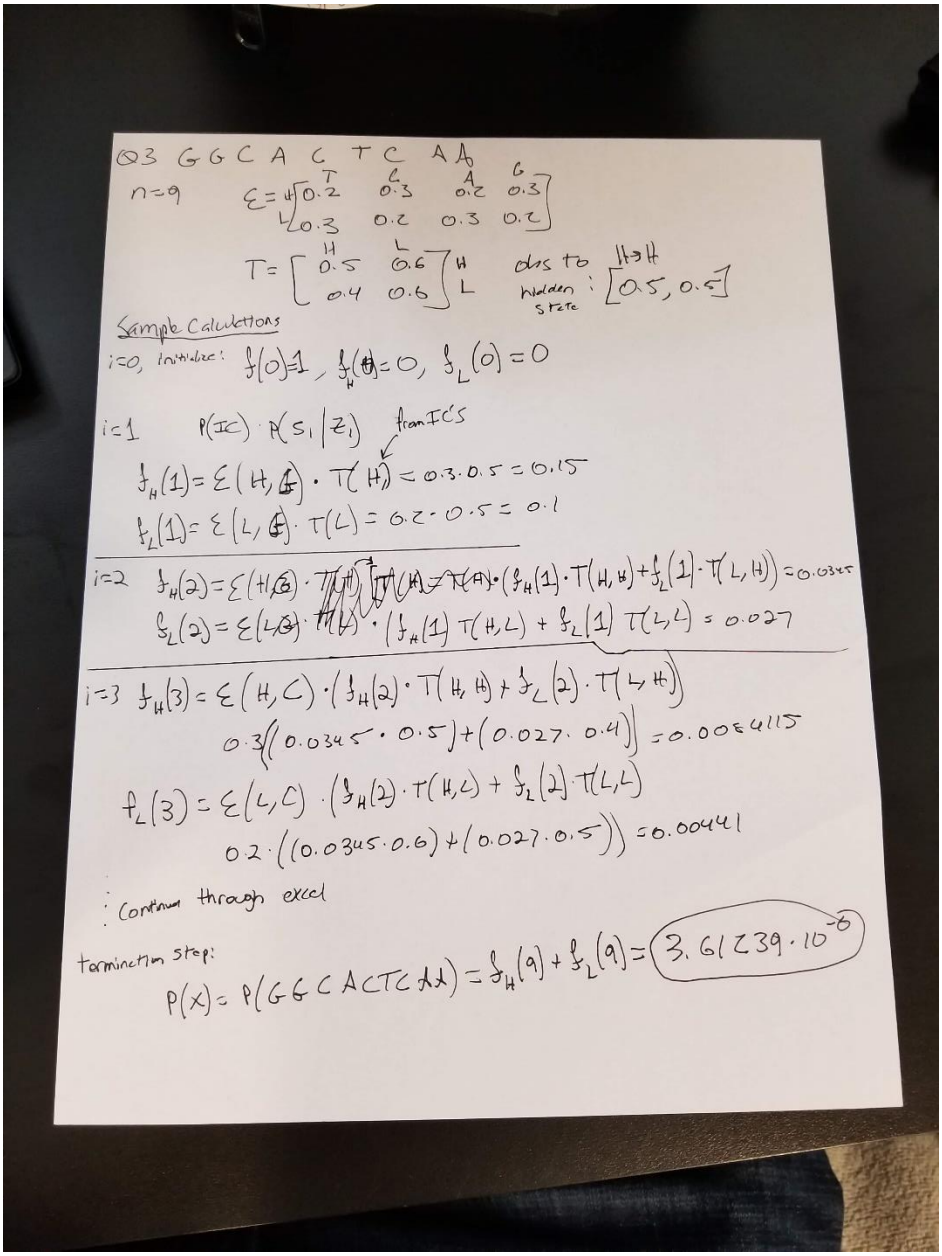
For forward algorithm give P(x(1:i) = fH(i) + fL(i) for each i

Problem Start:

GGCACTCAA

Down below you will see sample hand calculations, the rest was computed using Excel (Table is shown here)

Forward Algorithm:



	A	B	C	D	E	F	G	H	I	J
1	H	L					T	C	A	G
2	0.008415	0.00669				H	0.2	0.3	0.2	0.3
3		H	L		Emission	L	0.3	0.2	0.3	0.2
4	4, A	0.0013767	0.0025182							
5	5, C	0.000508689	0.000417024			Trans	0.5	0.6		
6	6, T	8.42308E-05	0.000154118				0.4	0.6		
7	7, C	3.11287E-05	2.55195E-05							
8	8, A	5.15443E-06	9.43109E-06			Repeat Trans	0.5	0.5		
9	9, A	1.26993E-06	2.34246E-06							
10										
11										
12	P	3.61239E-06								
13										
14										

Backward Algorithm:

The backward algorithm was computed in the same manner using Excel as shown:

Backward Algorithm:		
	H	L
9, A	1	1
8, A	0.28	0.23
7, C	0.0696	0.0566
6, T	0.017148	0.014058
5, C	0.00425916	0.00346356
4, A	0.001049357	0.000860267
3, C	0.000260636	0.000211949
2, G	6.45293E-05	5.24712E-05
1, G	1.59759E-05	1.29906E-05
P	**2.89666E-05**	

We can see that the final value of P(x) is different from the forward algorithm. In an ideal case, both algorithms should be outputting the same final P(x) value. There is likely an error within the backward algorithm calculation

Q4:

For GGCA, the last position is 4. The P(X) was found by looking at the forward algorithm tabulated data from Q3: f4(H) + f4(L) = 0.003895

Following the given formula: (0.0013767 * 1) / 0.003895 = 0.35822 (H)

And for L: 0.0025182 * 1 / 0.003895 = 0.64178

For GGCACTCAA:

H: 0.0013767 * 1) / 3.61239E-06 = 381.1048538

L: 0.0025182 * 1 / 3.61239E-06 = 697.1004887

Q5:

The difference between Viterbi decoding/smoothing and posterior decoding

The Viterbi algorithm finds a sequence of hidden states that maximizes a joint probability of observable data, and the states that correspond with it:

$$P(O, H)$$

Where O is the observed state, and H is the hidden state. The algorithm is a dynamic programming method that populates a table, V, that will contain the calculated emitted and transmitted probabilities as the states move from one another throughout the given observed sequence. Using recursion, this table can be populated, and a traceback step can be used to generate a vector containing the hidden states that correspond to the given observed states. This hidden state sequence is given by:

$$h^* = argmax_k P(O, H)$$

This algorithm is sequential and must be followed in a recursive matter. Whereas, the posterior decoding method can calculate the most likely hidden state in any given point in time. This hidden state sequence path is described for the posterior decoding method as:

$$h^*(t) = argmax_{k,t} P(H_t - k|\{x1 \dots xn\})$$

From the equation above, it should be noted that the posterior decoding method finds the state K at time t that maximizes the probability of the hidden state, given the full set of previous values.