BMI/CS 576 Fall 2016
Final Exam

Prof. Colin Dewey

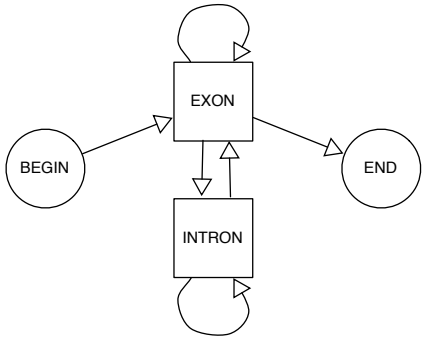
Saturday, December 17th, 2016 10:05am-12:05pm

Name: KEY

Write your answers on these pages and show your work. You may use the back sides of pages as necessary. Before starting, make sure your exam has every page (numbered 1 through 11). You are allowed 2 (double-sided) pages of notes. Calculators are *not* allowed.

Problem	**Score**	**Max Score**
1		25
2		25
3		25
4		25
Total		100

1. (25 points) Recall the HMM below, which you used in HW4 for predicting exons and introns within DNA sequences.



For some particular setting of the HMM parameters, below are the Viterbi, forward, and backward dynamic programming matrices for the sequence `GGACCTG`. The entries in the matrices are $\log_2$ transformed probabilities.

Viterbi

Position	0	1	2	3	4	5	6	7
INTRON	-∞	-∞	-7.3	-10.3	-12.7	-15.0	-18.0	-19.4
EXON	-∞	-2.7	-5.7	-7.4	-12.2	-17.0	-18.5	-21.5
BEGIN	0							-∞
END	-∞							-25.0

Forward

Position	0	1	2	3	4	5	6	7
INTRON	-∞	-∞	-7.3	-9.9	-11.6	-13.8	-16.8	-18.0
EXON	-∞	-2.7	-5.7	-7.3	-12.1	-16.5	-16.8	-19.5
BEGIN	0							-∞
END	-∞							-23.0

Backward

Position	0	1	2	3	4	5	6	7
INTRON	-19.9	-18.5	-17.2	-14.4	-12.0	-9.8	-8.8	-∞
EXON	-22.2	-20.3	-18.0	-16.5	-12.6	-8.0	-6.5	-3.5
BEGIN	-23.0							-∞
END	-∞							0

(a) (10 points) Given the values and traceback pointers in the dynamic programming matrices, give the following:

i. (3 points) The most likely path of hidden states for the sequence.

`BEGIN, EXON, EXON, EXON, EXON, EXON, EXON, EXON, END`

ii. (2 points) The probability ($\log_2$ transformed) of the sequence.

$\log_2 f_{END}(7) = \log_2 b_{BEGIN}(0) = -23.0$

iii. (2 points) The probability ($\log_2$ transformed) of the sequence and the most likely path of hidden states that generated it. In other words, if we let x be the sequence and $\hat{\pi}$ be the most likely path of hidden states, give $\log_2 P(x, \hat{\pi})$.

$\log_2 v_{END}(7) = -25.0$

iv. (3 points) The posterior probability ($\log_2$ transformed) of the 4th position of the sequence being generated by the intron state. Recall that the posterior probability of state k having generated position i of the sequence can be computed with the equation $P(\pi_i = k|x) = \frac{f_k(i)b_k(i)}{P(x)}$.

$$\begin{aligned}
\log_2 P(\pi_4 = intron|x) &= \log_2 \frac{f_{intron}(4)b_{intron}(4)}{P(x)} \\
&= \log_2 f_{intron}(4) + \log_2 b_{intron}(4) - \log_2 f_{END}(7) \\
&= (-11.6) + (-12.0) - (-23.0) = -0.6
\end{aligned}$$

(b) (5 points) One alternative to Viterbi for predicting the path of hidden states for a sequence is *posterior decoding*. With *posterior decoding*, the hidden state predicted to have generated a given position is the state with the highest posterior probability of having generated that position. For the sequence in this problem, is the path of hidden states predicted by posterior decoding the same as that predicted by the Viterbi algorithm? Explain your answer.

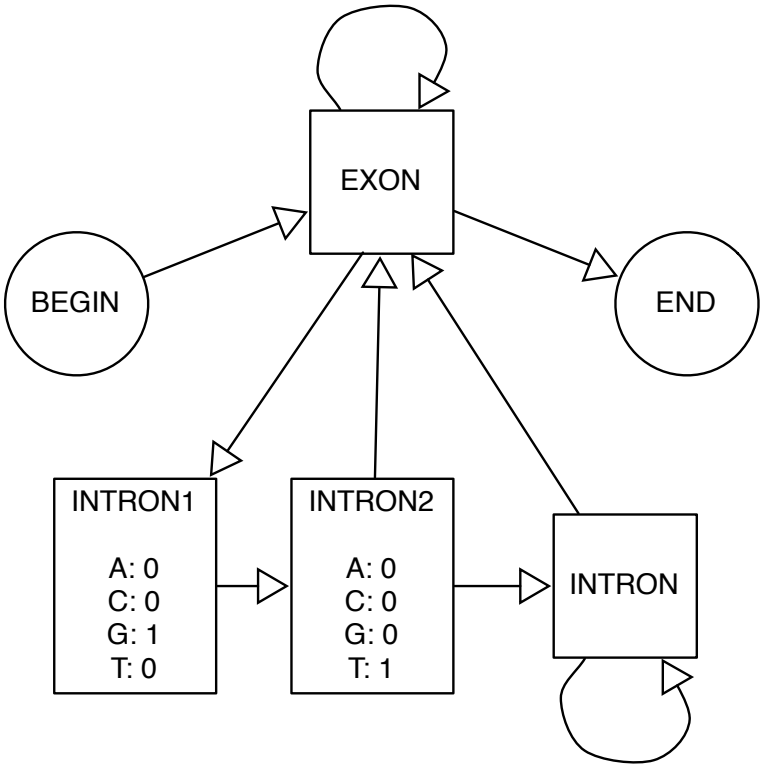
No. For example, for position 4, $\log_2 P(\pi_4 = intron|x) > \log_2 0.5 = -1$ (see 1 (a) iii.), and thus posterior decoding would predict the 4th position to be from the intron state, whereas the Viterbi path predicts that the 4th position is from the exon state.

For completeness, the posterior decoding path for this sequence is `BEGIN, EXON, EXON, EXON, INTRON, INTRON, EXON, EXON, END`

(c) (5 points) Suppose that in exonic segments it is rare for a `T` to follow a `G`. Is the HMM in this problem able to model this feature of exonic segments? Briefly explain your answer.

No. For this HMM, emission probability of a character within the exonic state is only conditioned on the state (exon) for that position, and is not conditional on character that occurred before it. Thus, one cannot model the lower probability of a `T` after a `G` within an exon.

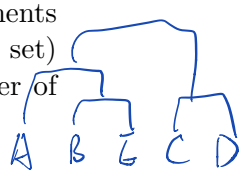
(d) (5 points) It is known that most intronic segments start with the two-nucleotide sequence `GT`. Describe how to modify the HMM to enforce that introns always start with `GT`.



2. (25 points) Suppose we wish to hierarchically cluster some data using a *median link* function for the distance between two clusters:

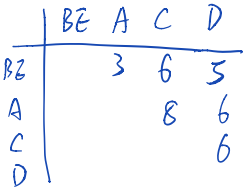
$$dist(c_u, c_v) = median\{dist(a, b) : a \in c_u, b \in c_v\}$$

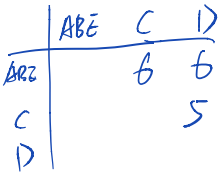
where the median of a set of numeric values may be found by sorting the elements and picking the middle element (if there are an odd number of elements in the set) or taking the arithmetic mean of the two middle elements (for an even number of elements).



(a) (15 points) Given the following distance matrix, show how the data will be joined into a single tree using *median link*. Show your work.

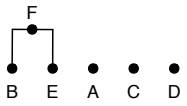
	A	B	C	D	E
A		4	8	6	2
B			6	4	1
C				5	6
D					6
E					





(1) Join B & E --> F

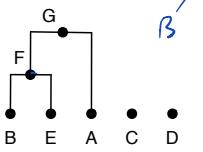
	A	C	D	F
A		8	6	3
C			5	6
D				5
F				

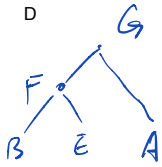


d(A,F) = median{2,4} = 3
d(C,F) = median{6,6} = 6
d(D,F) = median{4,6} = 5

(2) Join A & F --> G

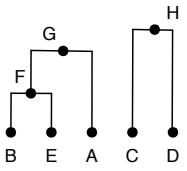
	C	D	G
C		5	6
D			6
G			



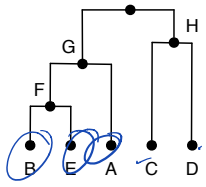


d(C,G) = median{8,6,6} = 6
d(D,G) = median{4,6,6} = 6

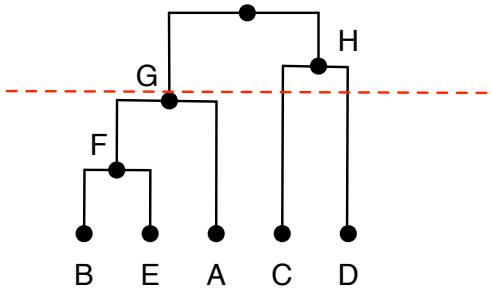
(3) Join C & D --> H



(4) Join G & H --> root



(b) (5 points) Show how your tree would be cut to produce a partitional clustering of $k = 3$ clusters. Give the three clusters that are produced from this process.



clusters: {A,B,E}, {C}, {D}

(c) (5 points) In terms of computational time, is hierarchical clustering with the median link function faster, slower, or the same as with the average link function? Explain your answer.

Slower. For the average link function there exists a constant-time update function for computing the distance from a newly-formed cluster with the other clusters. For median link, we do not have such an efficient update function.

3. (25 points) A hidden Markov model (HMM) may be represented by a Bayesian network. Figure 1 gives the structure of the Bayesian network for an observed sequence $Y = Y_1, \ldots, Y_n$, with the hidden states of each position represented by $X = X_1, \ldots, X_n$. Suppose the state transition diagram (with emission probabilities) for the HMM is that of Figure 2, a simple eukaryotic gene model. There is no end state in this gene HMM as we will not be modeling the length of the sequence generated. Thus, each X_i takes on a value from $\{N, E, I\}$. Each Y_i takes on a value from $\{A, C, G, T\}$.

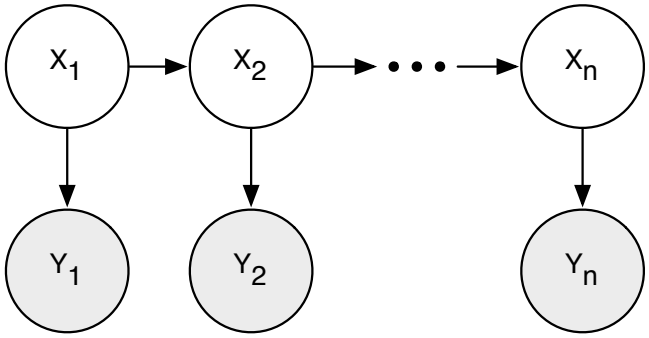


Figure 1: HMM Bayesian network

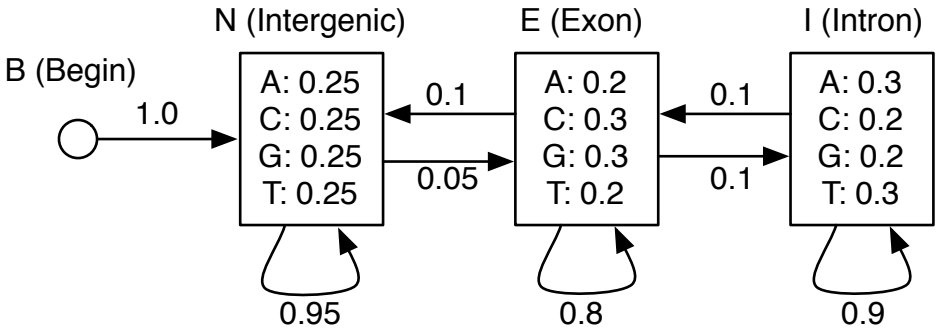


Figure 2: State transition diagram for a simple eukaryotic gene HMM

Give the following (conditional) probability tables for the Bayesian network:

(a) (4 points) $P(X_1)$

X_1

N	E	I
1	0	0

(b) (9 points) $P(X_i|X_{i-1}),\ \forall i > 1$

X_i

X_{i-1}	N	E	I
N	0.95	0.05	0
E	0.1	0.8	0.1
I	0	0.1	0.9

(c) (12 points) $P(Y_i|X_i)$

Y_i

X_i	A	C	G	T
N	0.25	0.25	0.25	0.25
E	0.2	0.3	0.3	0.2
I	0.3	0.2	0.2	0.3

4. (25 points) Looking back on the problems that we discussed during the semester:

   (a) (9 points) Give *one* example of a greedy algorithm that we used during the semester. Why is this algorithm considered "greedy"?

   Various possible answers.

   (b) (8 points) Give *one* example of a dynamic programming algorithm that we used during the semester. What are the *subproblems* that are solved by this algorithm?

   Various possible answers.

(c) (8 points) Give *one* example of an Expectation-Maximization algorithm that we used during the semester. What are computed during (i) the *expectation* (E) step and (ii) fthe *maximization* (M) step of this algorithm?

Various possible answers.

Forward - Backward Algorithm

HMM parameter estimation