Name: **Student ID:**

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

01.112 Machine Learning, Spring 2018
Final Exam
(Sample Solutions)

Date: 27 April 2018
Time: 15:00 - 17:00

Instructions:

1. Write your name and student ID at the top of this page.
2. This paper consists of 4 main questions and 14 printed pages.
3. The problems are not necessarily in order of difficulty. We recommend that you scan through all the questions first, and then decide on the order to answer them.
4. Write your answers in the space provided.
5. You may refer to your one-sided A4-sized cheat sheet.
6. You are allowed to use non-programmable calculators.
7. You may NOT refer to any other material.
8. You may NOT access the Internet.
9. You may NOT communicate via any means with anyone (aside from the invigilators).

For staff's use:

Qs 1	/8
Qs 2	/12
Qs 3	/24
Qs 4	/6
Total	**/50**

Question 1. (8 points)
Please indicate whether the following statements are true (**T**) or false (**F**).

1. The EM algorithm can be used for unsupervised learning of a hidden Markov model. *(1 point)*

   Answer : T

2. The Viterbi algorithm involves a forward pass that calculates the score of the optimal path, and a backward pass for recovering/extracting the optimal path. *(1 point)*

   Answer : T

3. When learning Bayesian network structures from data, we should not use the log-likelihood as the criteria for selecting the structure of a Bayesian network. If we do so, we might find that in general, a more complex structure is always preferred over a simpler structure as a more complex structure tends to fit the data better. *(1 point)*

   Answer : T

4. In the Markov decision process, we are interested in learning a policy π, which is a function that specifies for each state an action for the agent to take. *(1 point)*

   Answer : T

5. The Q-value iteration algorithm will always converge eventually, and the policy derived from the converged Q-values is always an optimal policy. *(1 point)*

   Answer : T

6. The hidden Markov model is a special Bayesian network. *(1 point)*

   Answer : T

7. There are some uncertainties associated with the Markov decision process. Specifically, the agent is unable to fully determine its current state at any given time. *(1 point)*

   Answer : F

8. Both the Viterbi and the forward-backward algorithms discussed in class share the same time and space complexity. *(1 point)*

   Answer : T

Question 2. (12 points)
Consider the learning problem for Bayesian networks. Assume we have 9 discrete random variables X_1, X_2, ..., X_9. We consider the following two possible Bayesian network structures: G_1 and G_2.

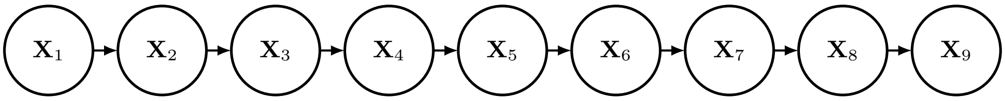

Figure 1: G_1

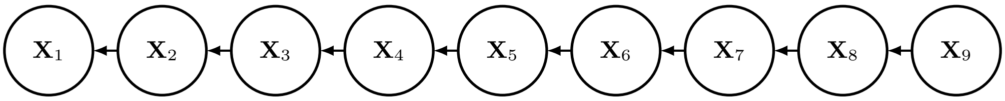

Figure 2: G_2

(a) Assume all variables are taking values from $\{1, 2\}$. What is the number of free parameters for G_1? Answer: 17

(1 point)

(b) Assume all variables are taking values from $\{1, 2, 3\}$. What is the number of free parameters for G_2? Answer: 50

(1 point)

(c) What is the Markov blanket for the variable X_1 in G_1? Answer: X_2

(1 point)

(d) What is the Markov blanket for the variable X_1 in G_2? Answer: X_2

(1 point)

(e) When a large collection of samples is given, you could use the maximum likelihood criterion to estimate the model parameters for both G_1 and G_2. After the learning has completed, you check the final log-likelihood values for both Bayesian network structures and would like to make a comparison. Construct a case (i.e., provide a collection of samples, where each sample is of the form $X_1 = 1$, $X_2 = 2$,...,$X_9 = 2$, for example) where the final log-likelihood obtained for the first structure G_1 would be strictly higher than G_2. If you believe no such case exists, clearly explain why. *(3 points)*

No such case exists.

For any sample $X_1 = x_1, X_2 = x_2, \ldots, X_9 = x_9$, the likelihood would be $p(X_1 = x_1, X_2 = x_2, \ldots, X_9 = x_9)$.
Under MLE, using the learned model parameters, for the first structure, this is equal to:

$$\begin{aligned}
&p(X_1 = x_1) \times \prod_{k=2}^{9} p(X_k = x_k | X_{k-1} = x_{k-1}) && (1)\\
&= \frac{\#(X_1 = x_1)}{\#\text{instances}} \times \prod_{k=2}^{9} \frac{\#\Big((X_k = x_k) \wedge (X_{k-1} = x_{k-1})\Big)}{\#(X_{k-1} = x_{k-1})} && (2)\\
&= \frac{1}{\#\text{instances}} \times \frac{\prod_{k=2}^{9} \#\Big((X_k = x_k) \wedge (X_{k-1} = x_{k-1})\Big)}{\prod_{k=2}^{8} \#(X_k = x_k)} && (3)
\end{aligned}$$

Similarly, for the second structure, we have:

$$\begin{aligned}
&p(X_9 = x_9) \times \prod_{k=2}^{9} p(X_{k-1} = x_{k-1} | X_k = x_k) && (4)\\
&= \frac{\#(X_9 = x_9)}{\#\text{instances}} \times \prod_{k=2}^{9} \frac{\#\Big((X_k = x_k) \wedge (X_{k-1} = x_{k-1})\Big)}{\#(X_k = x_k)} && (5)\\
&= \frac{1}{\#\text{instances}} \times \frac{\prod_{k=2}^{9} \#\Big((X_k = x_k) \wedge (X_{k-1} = x_{k-1})\Big)}{\prod_{k=2}^{8} \#(X_k = x_k)} && (6)
\end{aligned}$$

As we can see, the above two are equal. Thus for any sample the likelihood would be the same. So the overall likelihood would be the same for both structures for any collection of samples.

(f) Now, you would like to use BIC as the criterion for selecting a better structure of the Bayesian network between G_1 and G_2. Construct a case (i.e., provide a collection of samples, where each sample is of the form $X_1 = 1$, $X_2 = 2$,...,$X_9 = 2$, for example) where the final BIC of the first structure G_1 would be strictly higher than G_2. If you believe no such case exists, clearly explain why. *(5 points)*

No such case exists.

The BIC scores for both structures are defined as follows:

$$BIC(D;\theta,G_1) = l(D;\theta,G_1) - \frac{dim(G_1)}{2}\log(m) \tag{7}$$

$$BIC(D;\theta,G_2) = l(D;\theta,G_2) - \frac{dim(G_2)}{2}\log(m) \tag{8}$$

As mentioned above, the log likelihood of 2 structures are same regardless of the collection of samples used for learning. Now the only possible difference would be in the second term in the above two equations. The term $\log(m)$ would be the same for both (since the number of samples m is the same). The only question is the number of free parameters $dim(G_1)$ and $dim(G_2)$ – are they the same? Let us assume each variable X_k can take r_k possible values. Now let us calculate the number of free parameters for G_1 and G_2 respectively.

$$dim(G_1) = (r_1 - 1) + \sum_{k=2}^{9}(r_k - 1)r_{k-1} \tag{9}$$

$$= \sum_{k=1}^{8} r_k r_{k+1} - \sum_{k=2}^{8} r_k - 1 \tag{10}$$

$$dim(G_2) = (r_9 - 1) + \sum_{k=2}^{9}(r_{k-1} - 1)r_k \tag{11}$$

$$= \sum_{k=1}^{8} r_k r_{k+1} - \sum_{k=2}^{8} r_k - 1 \tag{12}$$

As we can see they are the same. Hence the two BIC scores for both structures for any given collection of training samples would always be the same.

Question 3. (24 points)

In this problem, we would like to look at the Hidden Markov Model (HMM).

(a) Assume that we have the following data available for us to estimate the model parameters:

State sequence	Observation sequence
(X, Y, Z, X)	(a, c, a, b)
(X, Z, Y)	(a, b, a)
(Z, Y, X, Z, Y)	(b, c, a, b, c)
(Z, X, Y)	(c, b, a)

Clearly state what are the parameters associated with the HMM. Under the maximum likelihood estimation (MLE), what would be the optimal model parameters? Fill up the following emission probability table. Use the space below to clearly show how each emission parameter is estimated exactly. *(9 points)*

$b_u(o)$ $u\backslash o$	a	b	c
X	0.6	0.4	0
Y	0.4	0	0.6
Z	0.2	0.6	0.2

There are two different types of probabilities: emission probabilities and transition probabilities.

$$b_X(a) = \frac{Count(X \to a)}{Count(X)} = \frac{3}{5}$$
$$b_X(b) = \frac{Count(X \to b)}{Count(X)} = \frac{2}{5}$$
$$b_X(c) = \frac{Count(X \to c)}{Count(X)} = 0$$
$$b_Y(a) = \frac{Count(Y \to a)}{Count(Y)} = \frac{2}{5}$$
$$b_Y(b) = \frac{Count(Y \to b)}{Count(Y)} = 0$$
$$b_Y(c) = \frac{Count(Y \to c)}{Count(Y)} = \frac{3}{5}$$
$$b_Z(a) = \frac{Count(Z \to a)}{Count(Z)} = \frac{1}{5}$$
$$b_Z(b) = \frac{Count(Z \to b)}{Count(Z)} = \frac{3}{5}$$
$$b_Z(c) = \frac{Count(Z \to c)}{Count(Z)} = \frac{1}{5}$$

(b) Now, consider you are given the following new observation sequence with two observations only (b, c), and the following model parameters, compute the following joint probability of the observation sequence using some algorithm that we have discussed in class. Clearly present the steps that lead to your final answer. *(6 points)*

$a_{u,v}$ $u\backslash v$	X	Y	Z	STOP
START	0.1	0.1	0.8	0.0
X	0.2	0.2	0.5	0.1
Y	0.1	0.2	0.2	0.5
Z	0.4	0.4	0.1	0.1

$b_u(o)$ $u\backslash o$	a	b	c
X	0.6	0.4	0.0
Y	0.4	0.0	0.6
Z	0.1	0.3	0.6

$$p(x_1 = b, x_2 = c)$$

We can use the forward algorithm to solve this question. $p(x_1 = b, x_2 = c)$ is the sum of scores of all paths from `START` to `STOP` where the observations are b and c for the first and second positions respectively.

$$\begin{aligned}\alpha_X(1) &= a_{\texttt{Start},X} = 0.1\\ \alpha_Y(1) &= a_{\texttt{Start},Y} = 0.1\\ \alpha_Z(1) &= a_{\texttt{Start},Z} = 0.8\end{aligned}$$

For time step 2, forward scores are updated as follows $u \in X, Y, Z$

$$\begin{aligned}\alpha_u(2) &= \sum_{v \in X,Y,Z} \alpha_v(1) a_{v,u} b_v(b)\\ \alpha_X(2) &= 0.1 \times 0.2 \times 0.4 + 0 + 0.8 \times 0.4 \times 0.3\\ &= 0.104\\ \alpha_Y(2) &= 0.1 \times 0.2 \times 0.4 + 0 + 0.8 \times 0.4 \times 0.3\\ &= 0.104\\ \alpha_Z(2) &= 0.1 \times 0.5 \times 0.4 + 0 + 0.8 \times 0.1 \times 0.3\\ &= 0.044\end{aligned}$$

Required probability is calculated as follows

$$\begin{aligned}p(x_1 = b, x_2 = c) &= \sum_{v \in X,Y,Z} \alpha_v(2) a_{v,\texttt{STOP}} b_v(c)\\ &= 0 + 0.104 \times 0.5 \times 0.6 + 0.044 \times 0.1 \times 0.6\\ &= 0.03384\end{aligned}$$

(c) The HMM discussed in class makes a simple first-order assumption, where the next state only depends on the previous state in the generative process. However, it is possible to extend the model discussed in class to have second-order dependencies. In other words, the HMM can be parameterized in the following way:

$$p(x_1, \ldots, x_n, y_{-1}, y_0, y_1, y_2, \ldots, y_n, y_{n+1}) = \prod_{i=1}^{n+1} p(y_i|y_{i-2}, y_{i-1}) \cdot \prod_{i=1}^{n} p(x_i|y_i)$$

where we define $y_{-1} = y_0 = \texttt{START}$ and $y_{n+1} = \texttt{STOP}$.

In other words, the transition probabilities are changed from $p(y_i|y_{i-1})$ to $p(y_i|y_{i-2}, y_{i-1})$.

Assume you are given a large collection of observation sequences, and a predefined set of possible states $\{0, 1, \ldots, T-1, T\}$, where $0 = \texttt{START}$ and $T = \texttt{STOP}$. Describe the forward-backward algorithm used for inference with such a second-order HMM. In other words, describe the dynamic programming algorithms that compute the following efficiently for such an HMM.

$$\alpha_{u,v}(j) = P(X_1, X_2, \ldots, X_{j-1}, Y_{j-1} = u, Y_j = v; \theta) \quad (13)$$

$$\beta_{u,v}(j) = P(X_j, X_{j+1}, \ldots, X_n | Y_{j-1} = u, Y_j = v; \theta) \quad (14)$$

Clearly write down the algorithms (with the base case, recursive case and final case) for computing these α and β scores. Analyze the time complexity associated with your algorithms (for an observation sequence of length n). *(9 points)*

Forward probability

We first define the transition probabilities as $a_{u,v,w} = P(Y_{k+2} = w \mid Y_k = u, Y_{k+1} = v)$

- Base case:

$$\alpha_{\texttt{START},\texttt{START}}(0) = 1$$

$$\alpha_{\texttt{START},u}(1) = a_{\texttt{START},\texttt{START},u} \quad \forall u \in 1, \ldots, T-1$$

  (providing either one is fine.)

- Recursive case:

$$\alpha_{u,v}(j+1) = \sum_{w} \alpha_{w,u}(j) a_{w,u,v} b_u(x_j) \quad \forall u, v \in 1, \ldots, T-1, j = 1, \ldots, n-1$$

- Final case:

$$\alpha_{u,v}(n) = \sum_{w} \alpha_{w,u}(n-1) a_{w,u,v} b_u(x_{n-1}) \quad \forall u, v \in 1, \ldots, T-1$$

$$\alpha_{u,\texttt{STOP}}(n+1) = \sum_{w} \alpha_{w,u}(n) a_{w,u,\texttt{STOP}} b_u(x_n) \quad \forall u \in 1, \ldots, T-1$$

  (providing either one is fine.)

Backward probability

- Base case:

$$\beta_{u,\texttt{STOP}}(n+1) = 1 \ \ \forall u \in 1,\dots,T-1$$
$$\beta_{u,v}(n) = a_{u,v,\texttt{STOP}} b_v(x_n) \ \ \forall u,v \in 1,\dots,T-1$$

  (providing either one is fine.)

- Recursive case:

$$\beta_{u,v}(j) = \sum_w a_{u,v,w} b_v(x_j) \beta_{v,w}(j+1) \ \ \forall u,v \in 1,\dots,T-1, j = n-1,\dots,1$$

- Final case:

$$\beta_{\texttt{START},u}(1) = \sum_w a_{\texttt{START},u,w} b_u(x_1) \beta_{u,w}(2) \ \ \forall u \in 1,\dots,T-1$$
$$\beta_{\texttt{START},\texttt{START}}(0) = \sum_w a_{\texttt{START},\texttt{START},w} \beta_{\texttt{START},w}(1)$$

  (providing either one is fine.)

From the above we can see that at each time step there are $O(T^2 \times T)$ operations, thus the time complexity is $O(nT^3)$

13

Question 4. (12 points)
Consider the following Markov decision process (MDP). It has states $\{0, 1, 2\}$. In every state, you can take one of two possible actions: A or B. State 0 is the terminal state (i.e., once the agent reaches that state, it stays there no matter what action it takes).

The transition probabilities for action A and B are given as:

$T(s, A, s')$ $s \backslash s'$	0	1	2
0	1.0	0.0	0.0
1	0.8	0.2	0.0
2	0.9	0.0	0.1

$T(s, B, s')$ $s \backslash s'$	0	1	2
0	1.0	0.0	0.0
1	1.0	0.0	0.0
2	0.0	1.0	0.0

For example, $T(2, A, 0) = 0.9$, $T(2, B, 1) = 1.0$.
The reward function is defined as $R(s, a, s') = (s' + 1)^2 - 1$. The discount factor is $\gamma = 0.2$.
Let us consider the Q-value iteration algorithm.

1. Suppose we initialize $Q_0^*(s, a) = 0$ for all $s \in \{0, 1, 2\}$ and $a \in \{A, B\}$. Evaluate the Q-values $Q_1^*(s, a)$ after exactly one iteration of the Q-value iteration algorithm. Write your answers in the table below.

	$s = 0$	$s = 1$	$s = 2$
A	0	0.6	0.8
B	0	0	3

2. What is the policy that we would derive from $Q_1^*(s, a)$? Answer by filling in the action that should be taken at each state in the table below. (In case of draw, the action A is preferred)

$s = 1$	$s = 2$
A	B

3. What are the values $V_1^*(s)$ corresponding to $Q_1^*(s, a)$?

$s = 0$	$s = 1$	$s = 2$
0	0.6	3

4. Will the policy change after the second iteration? If your answer is "yes", briefly describe how. (In case of draw, the action A is preferred)

No