Probabilistic Artificial Intelligence
Final Exam

Jan 27, 2015

Time limit:	120 minutes
Number of pages:	16
Total points:	100

You can use the back of the pages if you run out of space. Collaboration on the exam is strictly forbidden.

(1 point) Please fill in your name and student ID.

1. Logic **(17 points)**

a. Propositional logic

(4 points) (i) Write the following two propositional formulas in conjunctive normal form:

- $(A \vee B) \Rightarrow C$.
- $\neg(A \Rightarrow C)$.

(4 points) (ii) Prove, using resolution, that $(A \Rightarrow C)$ can be derived from $(A \vee B) \Rightarrow C$.

b. First-order logic

For this part, use the following first-order vocabulary:

- T, a binary relation that denotes who teaches whom. For two variables x and y, the formula $T(x, y)$ means that x teaches y.
- M, a unary relation, for denoting math students. For a variable x, the formula $M(x)$ means that x is a math student.
- CS, a unary relation, for denoting computer science students. For a variable x, the formula $CS(x)$ means that x is a computer science student.

(6 points) (i) Express the following statements using first-order logic. You may assume that all elements of the universe are students. (Note that there might be students that are neither math nor computer science students.)

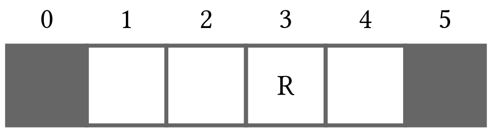
- Every student teaches some student.
- Every student teaches exactly one student.
- No one is both a math student and a computer science student.

(3 points) (ii) Describe a model that satisfies all statements of part (i) and, additionally, satisfies the following formula:

$$\forall x \forall y \forall z \, (T(x,z) \wedge T(y,z) \Rightarrow x = y) \, .$$

2. Robot on a Tape **(15 points)**

A robot moves along the one-dimensional tape shown in the figure below, where cells $1, \ldots, 4$ are free, while cells 0 and 5 are blocked. At each time step t, the robot occupies position X_t and uses two sensors to obtain observations L_t and R_t about the left and right neighboring cells respectively. The values of the observations are either F (free) or B (blocked), but, since the sensors are noisy, each of them is only correct with probability $a < 1$, independently of each other and across time steps t. At each step, the robot moves uniformly at random to one of its neighboring cells. (If the target cell is blocked, no movement is performed.)



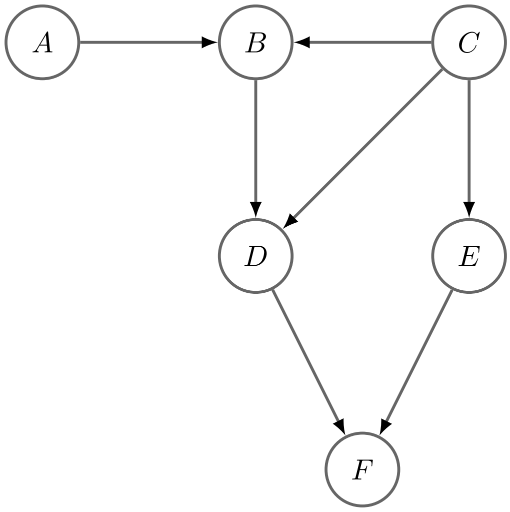
(7 points) (i) Assume that the position of the robot at time t is $X_t = 3$. After one move, the robot observes $L_{t+1} = F$ and $R_{t+1} = B$. Compute the posterior distribution of the robot's new position X_{t+1}.

(4 points) (ii) Draw a Bayesian network that illustrates the relationships between the random variables associated with each of three consecutive time steps of the above setup.

(4 points) (iii) Now, suppose that at each time step, the robot either continues moving along the same direction of its last move, or randomly picks a new direction among the two possible. Draw a Bayesian network representing this new setup. If you use new variables, briefly describe how they would be incorporated into the conditional probability tables of the network.

3. d-Separation (12 points)

Given the Bayesian network shown below, find a minimal set $\mathcal{S}_i$ of variables that have to be observed, so that each of the following d-separation properties hold. Minimal means that removing any elements from $\mathcal{S}_i$ will make the statement false. Note that $\mathcal{S}_i$ can be $\varnothing$, if the property holds without any additional observations. Briefly explain your answers. (We use d-sep$(X; Y \mid \mathcal{Z})$ to denote that X is d-separated from Y, given a set of observed variables $\mathcal{Z}$.)



(4 points) (i) d-sep$(A; E \mid \mathcal{S}_1)$

(4 points) (ii) d-sep$(A; E \mid F, \mathcal{S}_2)$

(4 points) (iii) d-sep$(B; F \mid D, \mathcal{S}_3)$

4. Inference in Bayesian Networks **(22 points)**

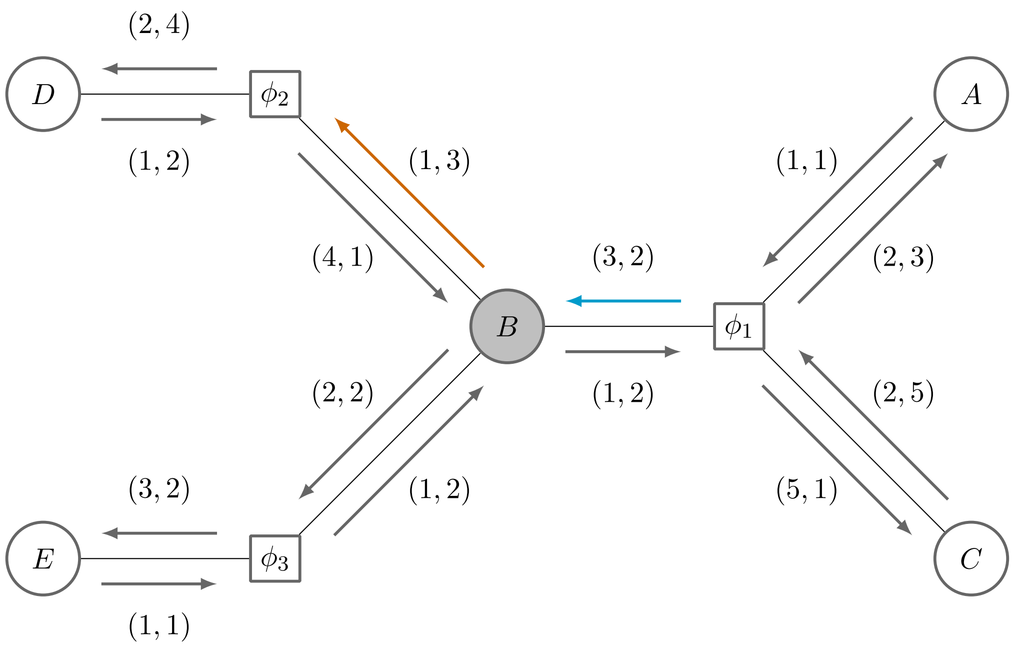
Assume that we have converted a Bayesian network to the factor graph shown below. The structure of the factor graph implies that the joint distribution factorizes as follows:

$$P(A, B, C, D, E) \propto \phi_1(A, B, C)\phi_2(B, D)\phi_3(B, E).$$

All random variables are binary and the factors are defined as follows:

$$\phi_1(A = a, B = b, C = c) = \begin{cases} 0 & \text{if } c = 0 \\ a + b + c & \text{if } c = 1 \end{cases}$$
$$\phi_2(B = b, D = d) = b + d$$
$$\phi_3(B = b, E = e) = b + e.$$

Assume that at the t-th iteration of the belief propagation algorithm the messages shown in the figure below are exchanged between variable and factor nodes in the graph. Each vector (v_0, v_1) defines an (unnormalized) message, for which v_0 corresponds to value 0 and v_1 corresponds to value 1. For example, $\mu^{(t)}_{\phi_1\to B}(0) = 3$, $\mu^{(t)}_{\phi_1\to B}(1) = 2$, and $\mu^{(t)}_{B\to\phi_1}(0) = 1$, $\mu^{(t)}_{B\to\phi_1}(1) = 2$.



(2 points) (i) Will belief propagation converge on the above factor graph? Briefly explain.

(4 points) (ii) From the provided messages compute the approximate marginal distribution of B.

(4 points) (iii) Compute the message $\mu^{(t+1)}_{B\to\phi_2}$ shown in orange. (You do not have to normalize.)

(8 points) (iv) Compute the message $\mu^{(t+1)}_{\phi_1 \to B}$ shown in blue. (You do not have to normalize.)

(4 points) (v) Assume that we are running a Gibbs sampler on the same factor graph and the last sample we drew is $(A = 0, B = 0, C = 1, D = 1, E = 1)$. Now, we want to update the value of variable A. Compute the distribution from which we should draw the new value of A.

5. Who Wants to Be a 'Hundredaire' (15 points)

Consider the imaginary TV game show "Who wants to be a hundredaire", where the host asks a series of questions that, if answered correctly, lead to a reward. If the participant does not want to attempt a question after seeing it, she has the option of taking her current reward and leaving the show. If she answers a question incorrectly, she has to leave the show with 0 reward. Consider such a game with three questions in sequence worth 1CHF, 10CHF, 100CHF respectively. As the questions are progressively more difficult, the probability that the corresponding question can be answered correctly is 0.5, 0.2, 0.05 respectively.

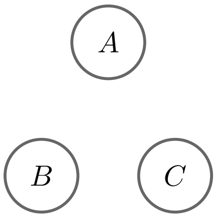
(5 points) (i) Draw the MDP for this game annotating the transitions, transition probabilities and rewards. *(Hint: You may wish to associate rewards with state transitions, i.e., dependent not just on the action and state in which it was taken, but also on where the player moves to.)*

(10 points) (ii) Since the game has at most three rounds, we can consider the problem of determining the optimal expected reward as a finite horizon ($T = 3$), undiscounted MDP. In such MDPs, one can obtain the optimal policy via a variant of the value iteration / dynamic programming algorithm discussed in the lecture.

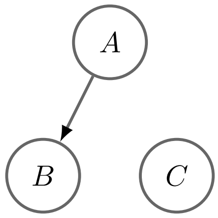
Define $V_t(s)$ as the optimal value of state s, assuming that the game ends after t rounds. Note that $V_0(s) = 0$ for all states s, and for $t \geq 1$, $V_t(s) = \max_a \sum_{s'} P(s' \mid a, s)[r(s, a, s') + V_{t-1}(s')]$. Use this identity to compute the optimal policy.

6. Learning Bayesian Networks **(8 points)**

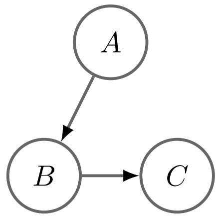
Suppose you want to learn a Bayesian network over a set of random variables. You want to select the best model among the six structures shown below. First, you use your training data to perform maximum likelihood estimation of the parameters of each of the networks. Then, for each of the learned networks, you evaluate the likelihood of the training data L_{train}, and the likelihood of test data L_{test}, i.e., data drawn independently from the same distribution as the training data, but not used to train the model. Both results are specified below each structure.



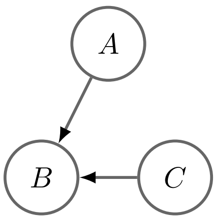
(a)
$L_{\text{train}} = 0.001$
$L_{\text{test}} = 0.0008$



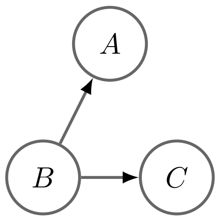
(b)
$L_{\text{train}} = 0.013$
$L_{\text{test}} = 0.011$



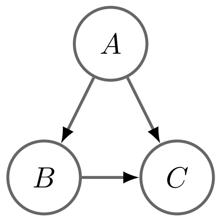
(c)
$L_{\text{train}} = 0.02$
$L_{\text{test}} = 0.018$



(d)
$L_{\text{train}} = 0.015$
$L_{\text{test}} = 0.012$



(e)
$L_{\text{train}} = 0.02$
$L_{\text{test}} = 0.018$



(f)
$L_{\text{train}} = 0.025$
$L_{\text{test}} = 0.01$

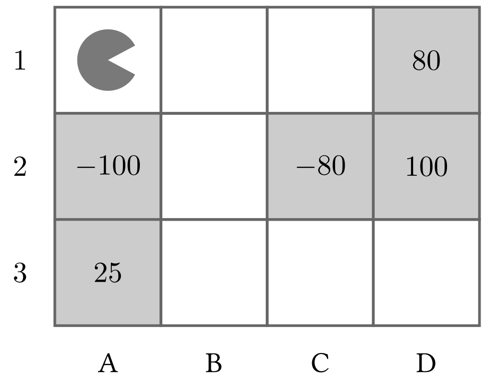
(4 points) (i) Among the six candidate structures, structure (c) and structure (e) have identical likelihoods. Explain why this is the case. (Justify your answer mathematically.)

(4 points) (ii) In comparison to structure $(b) - (e)$, why does structure (f) have the highest likelihood on training data, but low likelihood on test data?

7. Q-Learning **(10 points)**

Consider a grid world, where Pacman wants to learn a policy to maximize its reward. The world is modeled as a deterministic MDP initially unknown to Pacman.

All shaded states are terminal states, i.e., the MDP terminates upon reaching any of those states. At the other states, there are 4 possible actions: *Up, Down, Left, Right*, which deterministically move Pacman to the corresponding neighboring state (or have Pacman stay in place, if the action tries to move out of the grid). An action receives reward 0, unless it results in moving to one of the shaded states, in which case, the corresponding reward is awarded during that transition.



The agent starts from the top left corner (A1) and you are given the following episodes from runs of the agent through this grid-world. Each line of an episode is a tuple (s, a, s', r), containing the state s before transition, action a taken, state s' after transition, and reward r received.

(10 points) (i) Assume the discount factor is $\gamma = 0.5$ and the learning rate for Q-learning is $\alpha = 0.5$. All Q-values are initialized to 0. After Q-learning updates specified by the below episodes, write down the Q-values for the following state-action pairs: (B1, *Right*), (B2, *Down*), (C1, *Right*), (C3, *Right*). *(Hint: You only need to write down the Q-values that have been updated, that is, the non-zero values.)*

Episode 1	Episode 2	Episode 3	Episode 4
A1, *Right*, B1, 0	A1, *Right*, B1, 0	A1, *Right*, B1, 0	A1, *Right*, B1, 0
B1, *Right*, C1, 0	B1, *Down*, B2, 0	B1, *Down*, B2, 0	B1, *Down*, B2, 0
C1, *Right*, D1, 80	B2, *Down*, B3, 0	B2, *Down*, B3, 0	B2, *Down*, B3, 0
	B3, *Right*, C3, 0	B3, *Left*, A3, 25	B3, *Right*, C3, 0
	C3, *Right*, D3, 0		C3, *Up*, C2, −80
	D3, *Up*, D2, 100		