CS 188 Fall 2024 Introduction to Artificial Intelligence

Exam Prep 6 Solutions

Q1. Probability

Use the probability table to calculate the following values:

X_1	X_2	X_3	$P(X_1, X_2, X_3)$
0	0	0	0.05
1	0	0	0.1
0	1	0	0.4
1	1	0	0.1
0	0	1	0.1
1	0	1	0.05
0	1	1	0.2
1	1	1	0.0

1. $P(X_1 = 1, X_2 = 0) = 0.15$

2. $P(X_3 = 0) = 0.65$

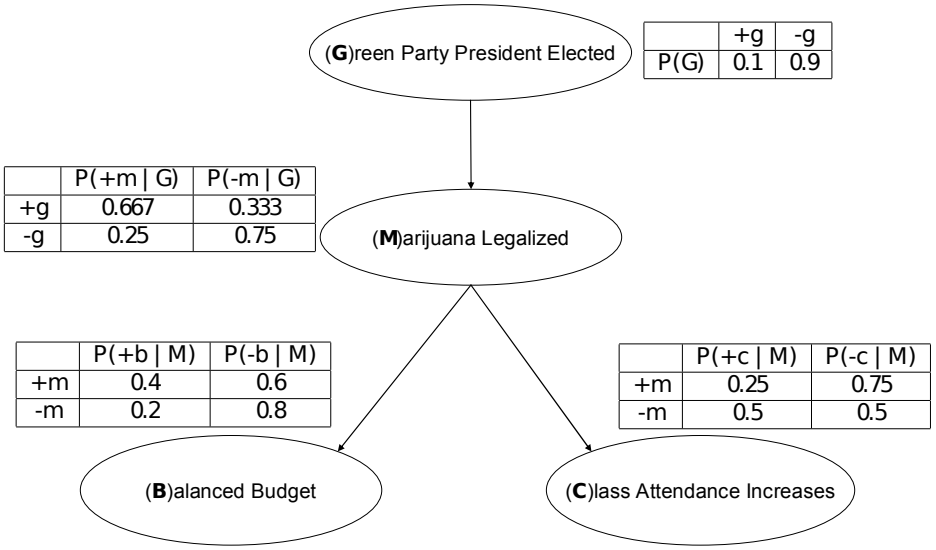
3. $P(X_2 = 1 | X_3 = 1) = 0.2/0.35$

4. $P(X_1 = 0 | X_2 = 1, X_3 = 1) = 1$

5. $P(X_1 = 0, X_2 = 1 | X_3 = 1) = 0.2/0.35$

Q2. Bayes Nets: Green Party President

In a parallel universe the Green Party is running for presidency. Whether a Green Party President is elected (G) will have an effect on whether marijuana is legalized (M), which then influences whether the budget is balanced (B), and whether class attendance increases (C). Armed with the power of probability, the analysts model the situation with the Bayes Net below.



	+g	-g
P(G)	0.1	0.9

	P(+m | G)	P(-m | G)
+g	0.667	0.333
-g	0.25	0.75

	P(+b | M)	P(-b | M)
+m	0.4	0.6
-m	0.2	0.8

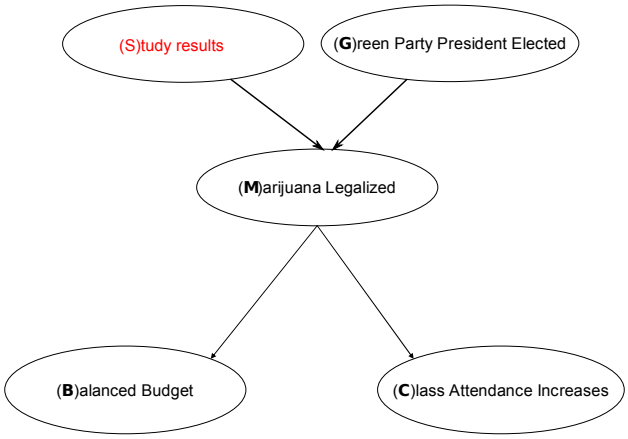
	P(+c | M)	P(-c | M)
+m	0.25	0.75
-m	0.5	0.5

1. The full joint distribution is given below. Fill in the missing values.

G	M	B	C	$P(G, M, B, C)$
+	+	+	+	1/150
+	+	+	-	1/50
+	+	-	+	1/100
+	+	-	-	3/100
+	-	+	+	1/300
+	-	+	-	1/300
+	-	-	+	1/75
+	-	-	-	1/75

G	M	B	C	$P(G, M, B, C)$
-	+	+	+	9/400
-	+	+	-	27/400
-	+	-	+	27/800
-	+	-	-	81/800
-	-	+	+	27/400
-	-	+	-	27/400
-	-	-	+	27/100
-	-	-	-	27/100

2. Now, add a node S to the Bayes net that reflects the possibility that a new scientific study could influence the probability that marijuana is legalized. Assume that the study does not directly influence B or C. Draw the new Bayes net below. Which CPT or CPT's need to be modified?



$P(M|G)$ will become $P(M|G, S)$, and will contain 8 entries instead of 4.

Q3. Policy Evaluation [RL Review]

In this question, you will be working in an MDP with states S, actions A, discount factor γ, transition function T, and reward function R.

We have some fixed policy $\pi : S \to A$, which returns an action $a = \pi(s)$ for each state $s \in S$. We want to learn the Q function $Q^\pi(s, a)$ for this policy: the expected discounted reward from taking action a in state s and then continuing to act according to π: $Q^\pi(s, a) = \sum_{s'} T(s, a, s')[R(s, a, s') + \gamma Q^\pi(s', \pi(s'))]$. The policy π will not change while running any of the algorithms below.

(a) Can we guarantee anything about how the values Q^π compare to the values Q^* for an optimal policy π^*?

- (●) $Q^\pi(s, a) \leq Q^*(s, a)$ for all s, a
- ( ) $Q^\pi(s, a) = Q^*(s, a)$ for all s, a
- ( ) $Q^\pi(s, a) \geq Q^*(s, a)$ for all s, a
- ( ) None of the above are guaranteed

(b) Suppose T and R are *unknown*. You will develop sample-based methods to estimate Q^π. You obtain a series of *samples* $(s_1, a_1, r_1), (s_2, a_2, r_2), \ldots (s_T, a_T, r_T)$ from acting according to this policy (where $a_t = \pi(s_t)$, for all t).

(i) Recall the update equation for the Temporal Difference algorithm, performed on each sample in sequence:

$$V(s_t) \leftarrow (1 - \alpha)V(s_t) + \alpha(r_t + \gamma V(s_{t+1}))$$

which approximates the expected discounted reward $V^\pi(s)$ for following policy π from each state s, for a learning rate α.

Fill in the blank below to create a similar update equation which will approximate Q^π using the samples.

You can use any of the terms $Q, s_t, s_{t+1}, a_t, a_{t+1}, r_t, r_{t+1}, \gamma, \alpha, \pi$ in your equation, as well as $\sum$ and max with any index variables (i.e. you could write $\max_a$, or $\sum_a$ and then use a somewhere else), but no other terms.

$$Q(s_t, a_t) \leftarrow (1 - \alpha)Q(s_t, a_t) + \alpha \left[r_t + \gamma Q(s_{t+1}, a_{t+1})\right]$$

(ii) Now, we will approximate Q^π using a linear function: $Q(s, a) = \mathbf{w}^\top \mathbf{f}(s, a)$ for a weight vector $\mathbf{w}$ and feature function $\mathbf{f}(s, a)$.

To decouple this part from the previous part, use Q_{samp} for the value in the blank in part (i) (i.e. $Q(s_t, a_t) \leftarrow (1 - \alpha)Q(s_t, a_t) + \alpha Q_{samp}$).

Which of the following is the correct sample-based update for $\mathbf{w}$?

- ( ) $\mathbf{w} \leftarrow \mathbf{w} + \alpha[Q(s_t, a_t) - Q_{samp}]$
- ( ) $\mathbf{w} \leftarrow \mathbf{w} - \alpha[Q(s_t, a_t) - Q_{samp}]$
- ( ) $\mathbf{w} \leftarrow \mathbf{w} + \alpha[Q(s_t, a_t) - Q_{samp}]\mathbf{f}(s_t, a_t)$
- (●) $\mathbf{w} \leftarrow \mathbf{w} - \alpha[Q(s_t, a_t) - Q_{samp}]\mathbf{f}(s_t, a_t)$
- ( ) $\mathbf{w} \leftarrow \mathbf{w} + \alpha[Q(s_t, a_t) - Q_{samp}]\mathbf{w}$
- ( ) $\mathbf{w} \leftarrow \mathbf{w} - \alpha[Q(s_t, a_t) - Q_{samp}]\mathbf{w}$

(iii) The algorithms in the previous parts (part i and ii) are:

- [ ] model-based
- [■] model-free