CS 188 Fall 2018 — Introduction to Artificial Intelligence — Final Exam

- You have 180 minutes. The time will be projected at the front of the room. You may not leave during the last 10 minutes of the exam.
- Do NOT open exams until told to. Write your SIDs in the top right corner of every page.
- If you need to go to the bathroom, bring us your exam, phone, and SID. We will record the time.
- In the interest of fairness, we want everyone to have access to the same information. To that end, we will not be answering questions about the content. If a clarification is needed, it will be projected at the front of the room. **Make sure to periodically check the clarifications**.
- The exam is closed book, closed laptop, and closed notes except your three-page double-sided cheat sheet. You are allowed a non-programmable calculator for this exam. Turn off and put away all other electronics.
- The last two sheets in your exam are scratch paper. Please detach them from your exam. Mark your answers ON THE EXAM IN THE DESIGNATED ANSWER AREAS. We will not grade anything on scratch paper.
- For multiple choice questions:
  - $\square$ means mark ALL options that apply
  - $\bigcirc$ means mark ONE choice
  - When selecting an answer, please fill in the bubble or square COMPLETELY ($\bullet$ and $\blacksquare$)

First name	
Last name	
SID	
Student to the right (SID and Name)	
Student to the left (SID and Name)	

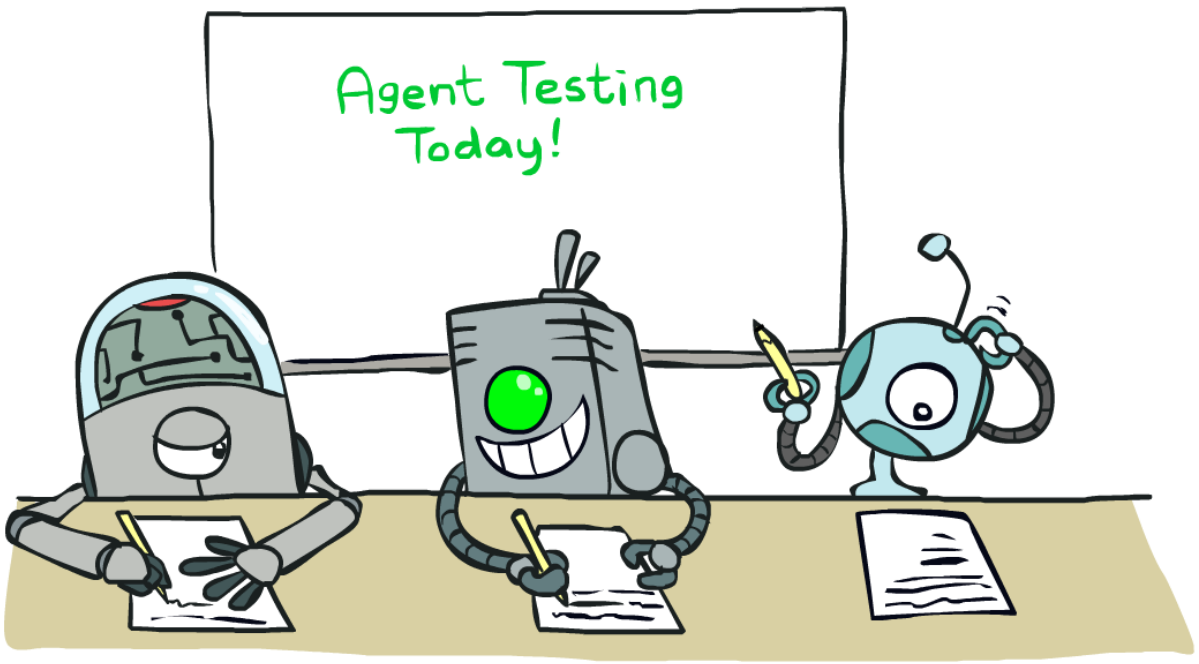
Q1.	Agent Testing Today!	/1
Q2.	Short Questions	/14
Q3.	Treasure Hunting MDPs	/12
Q4.	Approximate Q-learning	/8
Q5.	Value of Asymmetric Information	/16
Q6.	Bayes Net Modeling	/12
Q7.	Help the Farmer!	/14
Q8.	Bayes Nets and RL	/15
Q9.	Decision Trees	/8
	Total	/100

THIS PAGE IS INTENTIONALLY LEFT BLANK

SID: _____

Q1. [1 pt] Agent Testing Today!

It's testing time! Not only for you, but for our CS188 robots as well! Circle your favorite robot below.



Any answer was acceptable.

Q2. [14 pts] Short Questions

(a) [2 pts] Which of the following properties must a set of preferences satisfy if they are rational:

- [x] $(A \succ B)$ OR $(B \succ A)$ OR $(B \sim A)$
- [ ] $(B \succ A)$ AND $(C \succ A) \Rightarrow (C \sim B)$
- [x] $(A \sim B)$ AND $(B \sim C) \Rightarrow [p, A; 1-p, B] \sim [q, B; 1-q, C]$
- [x] $A \succ B \succ C \succ D \Rightarrow [p, A; 1-p, C] \succ [p, B; 1-p, D]$
- [x] $(B \succ A)$ AND $(C \succ B) \Rightarrow (C \succ A)$

The first one is the orderability axiom that all rational preferences must satisfy. The second one is false, you could consider the utility function $U(A) = 1, U(B) = 2, U(C) = 3$ which satisfies the premises but $U(B) \neq U(C)$. The second one is true since A, B, and C are equally preferable any lottery with those preferences is also equally preferable. It follows directly from the substitutability axiom. The fourth and fifth statements follow from the transitivity axiom.

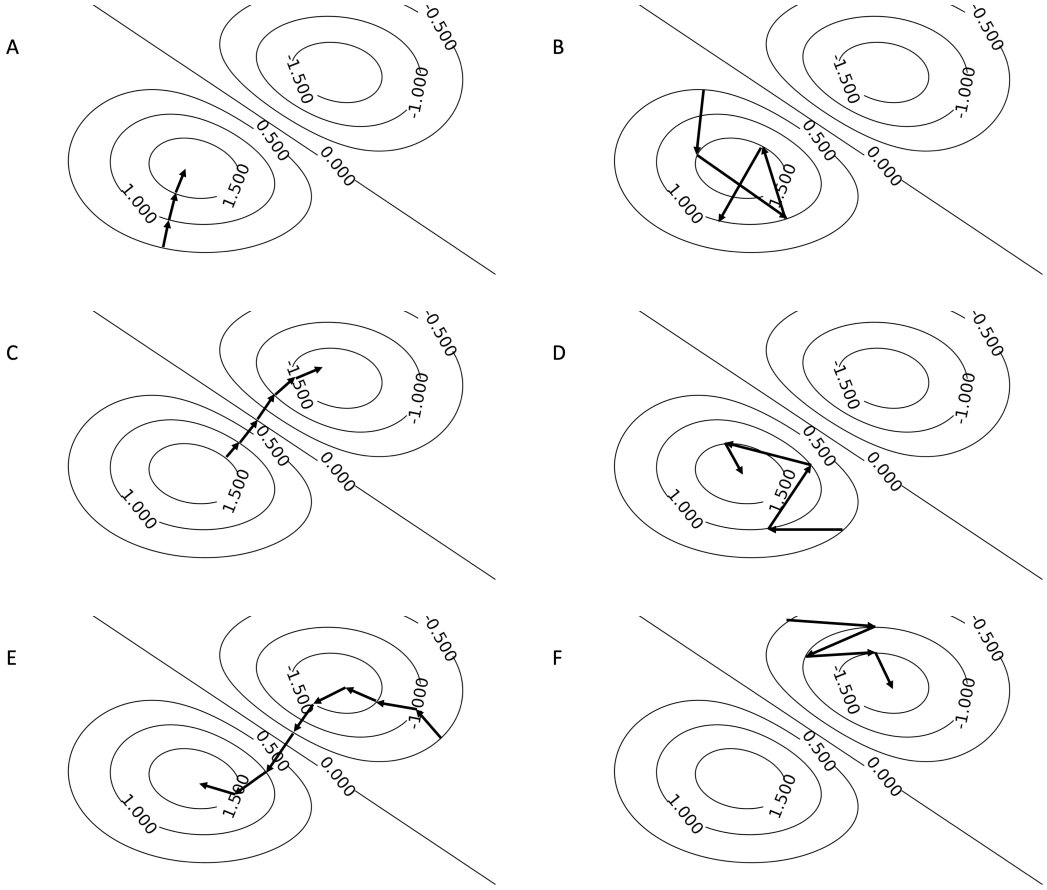
(b) [2 pts] Which of the following are true?

- [ ] Given a set of preferences there exists a unique utility function.
- [x] $U(x) = x^4$ is a risk prone utility
- [ ] $U(x) = 2x$ is a risk prone utility
- [x] For any specific utility function, any lottery can be replaced by an appropriate deterministic utility value
- [ ] For the lotteries $A = [0.8, \$4000; 0.2, \$0], B = [1.0, \$3000; 0.0, \$0]$ we have $A \succ B$

Given a set of preferences there exists an infinite number of utility functions, you may consider a fixed utility function and then apply a monotonic increasing function to it. This will result in a new utility function for that set of preferences. The second statement is risk prone since $U(L) > U(\text{EMV}(L))$. The third statement is risk neutral $U(L) = U(\text{EMV}(L))$. The fourth statement is true since given a lottery and a utility function, we consider the utility of it as the expected utility value for that lottery. The last statement is false: we can consider the utility function $\log(x)$, then we have $0.8 \log(4000) < \log(3000)$.

(c) [2 pts] Which of the following paths is a feasible trajectory for the gradient ascent algorithm?

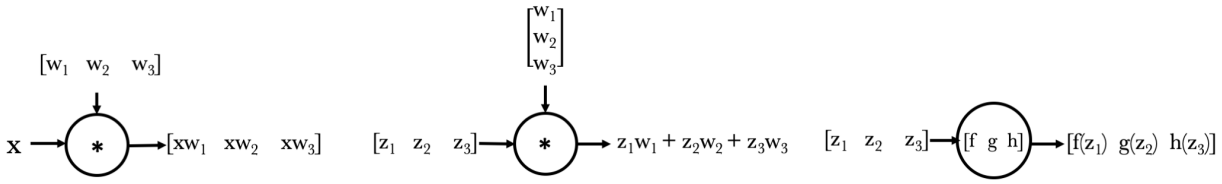
SID: _____



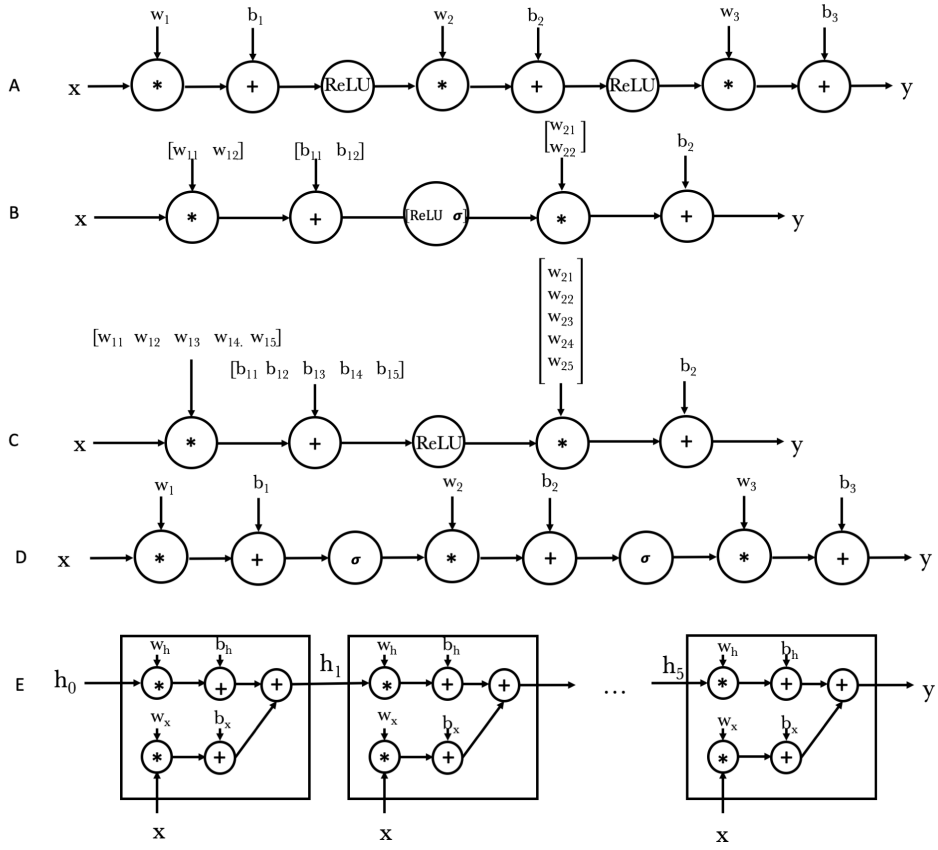
■ A ■ B ☐ C ☐ D ☐ E ☐ F

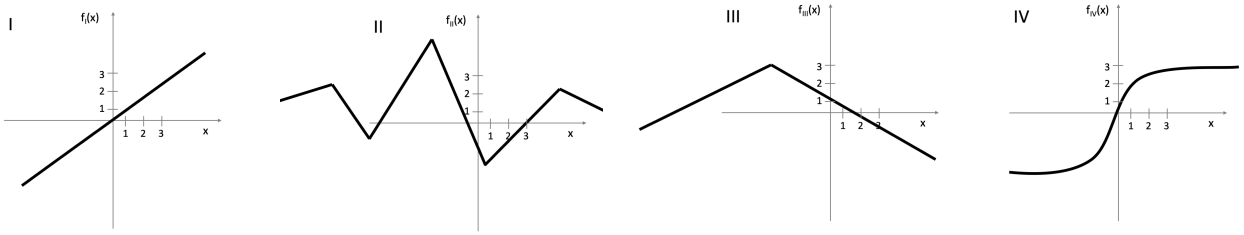
A is a gradient ascent path since the gradient lines are orthogonal to the contours and the point towards the maximum. B is also a gradient ascent path with a high learning rate. C is not because the path is going towards the minimum instead of the maximum. D is not a gradient ascent path since the gradient is not orthogonal to the contour lines. E is not a gradient ascent path since it starts going towards the minimum. F is not since it goes towards the minimum and the gradients are not orthogonal to the contour lines.

(d) We are given the following 5 neural networks (NN) architectures. The operation $*$ represents the matrix multiplication operation, $[w_{i1}...w_{ik}]$ and $[b_{i1}...b_{ik}]$ represents the weights and the biases of the NN, the orientation (vertical and horizontal) its just for consistency in the operations. The term [ReLU σ] in **B** means applying a ReLU activation to the first element of the vector and a sigmoid (σ) activation to the second element. These operations are depicted in the following figures:



Which of the following neural networks can represent each function?





(i) [2 pts] $f_I(x)$: ☐ A ☐ B ■ C ☐ D ■ E

A and B cannot represent this plot since the ReLU activation results in a flat semi-line. C can represent it by having $w_{11} = 1, w_{12} = -1, w_{21} = 1, w_{22} = 1$ and the rest of the parameters being 0. D cannot because the sigmoid activations are non-linear. E is a linear function and therefore can represent the identity.

(ii) [2 pts] $f_{II}(x)$: ☐ A ☐ B ■ C ☐ D ☐ E

This is a piecewise linear function with 6 pieces. As a result you can represent it by lineraly combining 5 ReLU functions. The only possible graph that can represent this function is C.

(iii) [2 pts] $f_{III}(x)$: ☐ A ☐ B ■ C ☐ D ☐ E

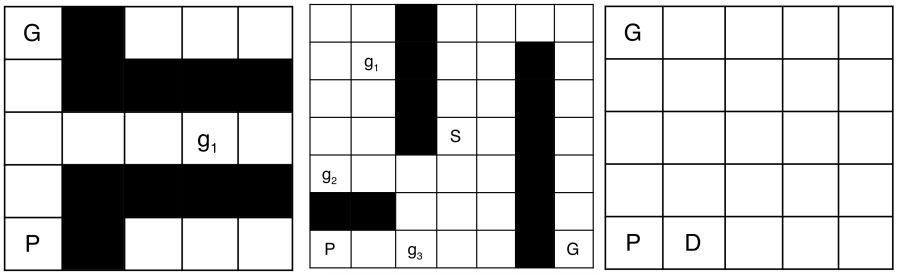
This is a piecewise linear function with 2 pieces. This can be obtained by linearly combining to ReLU functions. The only possible solution is then C. Note that A cannot represent this function since the last ReLU of the network would result in a flat semi-line.

(iv) [2 pts] $f_{IV}(x)$: ☐ A ■ B ☐ C ☐ D ☐ E

This function corresponds to a scaled sigmoid function. A, C, E cannot represent any non-linear or non-piecewise linear functions. B can represent it by setting $w_{11} = b_{11} = 0$. D does not work because the composition of sigmoid with sigmoid is not a sigmoid.

Q3. [12 pts] Treasure Hunting MDPs

In each below Gridworld, Pacman's starting position is denoted as P. G denotes the goal. At the goal, Pacman can only "Exit", which will cause Pacman to exit the Gridworld and gain reward +100. The Gridworld also has K gold nuggets $g_1, ..., g_k$ which will have the properties listed below in each part. Pacman automatically picks up a gold nugget if he enters a square containing one. Finally, define P_0 as Pacman's initial state, where he is in position P and has no gold nuggets.



(a) (b) (c)

(a) Pacman now ventures into (a). Each gold nugget Pacman holds when he exits will add +100 to his reward, and he will receive +100 when he exits from the goal G.

(i) [2 pts] When conducting value iteration, what is the first iteration at which $V(P_0)$ is nonzero?

Answer: 5

The shortest number of iterations is 5 because it takes 4 timesteps for Pacman to go from P to G, and 1 more timestep for Pacman to exit from the goal. It takes far more iterations for Pacman to gain reward by picking up g_1 and then exiting from G.

(ii) [2 pts] Assume Pacman will act optimally. What nonzero discount factor γ ensures that the policy of picking up g_1 before going to goal G and the policy of going straight to G yield the same reward?

Answer: $(\frac{1}{2})^{1/6}$

To solve for γ, we have the equation $200\gamma^{11} = 100\gamma^5$, where the left hand is the reward after picking up g_1, and the right hand is the reward after going straight to and exiting from the goal G.

Solving this equation yields $\gamma^6 = \frac{1}{2}$, $\gamma = \frac{1}{2}^{\frac{1}{6}} \approx 0.891$.

(b) Pacman is now at (b), which contains a Gold Store (S). He will receive +5 per nugget. When at the Store, Pacman can either "Sell" to sell all his gold for +5 per nugget or "Exit" to exit the Gridworld. Exiting from the Store yields 0 reward. Exiting from goal G will give $+100 + 5k$, where Pacman has k nuggets.

Note that Pacman can also only carry one gold nugget at a time.

(i) [2 pts] When conducting value iteration, what is the first iteration at which $V(P_0)$ is nonzero?

Answer: 7

It takes 7 steps for Pacman to pick up g_3 and carry it to the Store to sell it. All other methods of obtaining reward, such as exiting from goal G, selling g_1, selling g_2, or selling some combination of nuggets, takes more time than to directly carry g_3 to the store and sell it. The distance between P and S is 6, as Pacman picks g_3 up on the way, and 1 more timestep is needed to do the action "Sell."

(ii) [2 pts] Now Pacman is in a world with a Store that is not necessarily the Gridworld (b). Assume Pacman is acting optimally, and he begins at the Store. It takes Pacman time T_1, T_2, T_3 to go from the Store, pick up the nuggets g_1, g_2, g_3 respectively, return to the store, and sell each nugget. It takes time T_G to go from the Store and exit from the goal G. Assume $T_1 < T_2 < T_3 < T_G$.

What must be true such that the better policy for Pacman would be to gather and sell all nuggets and exit from the store rather than to gather all nuggets and exit from goal G?

- ● $5(\gamma^{T_1}+\gamma^{T_1+T_2}+\gamma^{T_1+T_2+T_3}) > 115\gamma^{T_1+T_2+T_3+T_G}$
- ○ $5(\gamma^{T_1}+\gamma^{T_2}+\gamma^{T_3}) > 100\gamma^{T_G}$
- ○ $5(\gamma^{T_1}+\gamma^{T_1+T_2}+\gamma^{T_1+T_2+T_3}) > 115\gamma^{T_G}$
- ○ $5(\gamma^{T_1}+\gamma^{T_1+T_2}+\gamma^{T_1+T_2+T_3}) > 100\gamma^{T_G}$
- ○ $15\gamma^{T_1+T_2+T_3} > 115\gamma^{T_G}$
- ● None of the above

The first solution, which was intended but incorrect, as well as "None of the above" were both awarded points due to unclear problem instructions.

The clarification that Pacman cannot use the Store if he holds multiple gold nuggets was added during the exam. The intended solution was to select the first option. However, it is always more optimal to pick up the gold nuggets in a single round trip, rather than returning to the store each time after a nugget is picked up, so the expression $115\gamma^{T_1+T_2+T_3+T_G}$ does not denote the reward if Pacman acts optimally and exits from the goal. Therefore, if we specify some time T'_G that is the optimal time to pick up all round trip gold nuggets and exit via the goal, then the reward when acting optimally and exiting from the goal is $115\gamma^{T'_G}$. However, we never specified any such value T'_G, and the correct answer was not one of the listed five inequalities.

Because the true answer was not included in the options, "None of the above" is the correct answer.

(c) Finally, Pacman finds himself in Gridworld (c). There is no store. However, Pacman finds that there is now a living reward! He gets the living reward for every action he takes except the Exit action. Pacman receives +0 exiting from the Door, and +100 exiting from the Goal. Once in the Door, Pacman can only Exit.

(i) [2 pts] Suppose $\gamma = 0.5$. For what living reward will Pacman receive the same reward whether he exits via the Door or exits via the goal?

Answer: $-\frac{50}{7}$

The rewards are the same if $\frac{1}{2}^4 * 100 + R_L + \frac{1}{2}R_L + \frac{1}{4}R_L + \frac{1}{8}R_L = R_L$.
Solving this yields $\frac{7}{8}R_L = -0.5^4 100$, which reduces to $R_L = -\frac{8*100}{7*16} = -\frac{50}{7}$.

(ii) [2 pts] Suppose $\gamma = 0.5$. What is the living reward such that Pacman receives the same reward if he traverses the Gridworld forever or if he goes straight to and exits from the goal? Hint: $\sum_{t=0}^{\infty} r\gamma^t = \frac{r}{1-\gamma}$

Answer: 50

He would want to go around forever if $\frac{R_L}{1-0.5} = \frac{1}{2}^4 100 + \frac{15}{8}R_L$, which can be reduced to $2R_L = \frac{25}{4} + \frac{15}{8}R_L$, or $\frac{1}{8}R_L = \frac{25}{4} = 50$. The $\frac{15}{8}$ in the first equation is from $R_L + \frac{1}{2}R_L + \frac{1}{4}R_L + \frac{1}{8}R_L$ that Pacman accumulated on the way to the goal G.

Q4. [8 pts] Approximate Q-learning

(a) [2 pts] Pacman is trying to collect all food pellets, and each treasure chest contains 10 pellets but must be unlocked with a key. Pacman will automatically pick up a pellet or key when he enters a square containing them, and he will automatically unlock a chest when he enters a square with a chest and has at least one key. A key can only unlock on chest; after being used, it vanishes.

To finish, Pacman must exit through either goal G_1 or G_2.

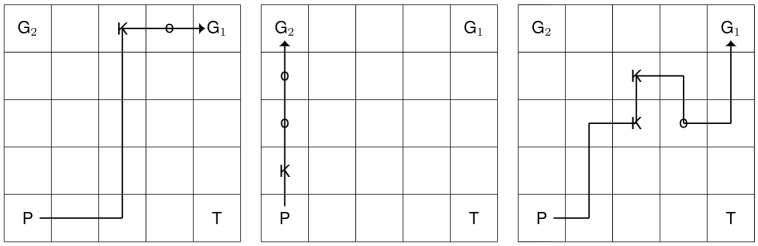
The keys are shown on the map as a K, treasure chests as T, and food pellets as circles. Pacman's starting position is shown as P. The goals are G_1, G_2.

When calculating features for a Q-function $Q(s, a)$, the state the features are calculated with is the state s' Pacman is in after taking action a from state s. The possible features the Q-learning can use are:

- N_{keys}: Number of keys Pacman holds.
- $D_m(K)$: Manhattan distance to closest key.
- N_{chests}: Number of chests Pacman has unlocked.
- $D_m(T)$: Manhattan distance to closest chest.
- N_{food}: Number of food pellets Pacman has eaten.
- $D_m(F)$: Manhattan distance to closest food pellet.
- $D_m(G_1)$: Manhattan distance to G_1.
- $D_m(G_2)$: Manhattan distance to G_2.

Note that the approximate Q-learning here can be any function over the features, not necessarily a weighted linear sum.

Suppose we finished training an agent using approximate Q-learning and we then run the learned policy to observe the following.



Assume we observe all the above paths. What is the minimal set of features that could have been used to learn this policy?

- ☐ N_{keys}
- ■ $D_m(K)$
- ☐ N_{chests}
- ☐ $D_m(T)$
- ☐ N_{food}
- ■ $D_m(F)$
- ■ $D_m(G_1)$
- ☐ $D_m(G_2)$
- ☐ No features

Due to the phrasing of the problem, there is some ambiguity regarding the precise minimal feature set. Full credit was given to answers that chose $D_m(G_1)$, at least one of $\{N_{keys}, D_m(K)\}$, at least one of $\{N_{food}, D_m(F)\}$, and did not pick any of $\{N_{chests}, D_m(T), D_m(G_2)\}$.

In episodes 1 and 3 above, Pacman walks in the direction of G_1; of the features provided only $D_m(G_1)$ could be used to learn this behavior.

Based on episode 3 above, Pacman is willing to move away from G_1 to gather a food pellet. Since the food pellet is just one square away, either N_{food} or $D_m(F)$ could be used to learn this behavior. At the same time, after collecting the first key Pacman chooses to collect the second key instead of picking up the pellet right away; this can be explained by the use of either N_{keys} or $D_m(K)$.

Moving towards/away from the treasure chest is not required to explain Pacman's behavior, so the choices N_{chests} and $D_m(T)$ are incorrect.

The feature $D_m(G_2)$ is not required, either. In episode 2, Pacman begins by collecting the key and two food pellets. At that point, moving either up or right would bring Pacman closer to G_1, so Pacman could have

broken the tie in favor of G_2, at which point Pacman happens to stumble across the second goal and the episode ends.

(b) Suppose Pacman is now in an empty grid of size $M \times M$. For a Q-value $Q(s, a)$, the features are the x- and y-position of the state Pacman is in after taking action a.

Select "Possible with weighted sum" if the policy can be expressed by using a weighted linear sum to represent the Q-function. Select "Possible with large neural net" if the policy can be expressed by using a large neural net to represent the Q-function. Select "Not Possible" if expressing the policy with given features is impossible no matter the function.

(i) [2 pts] Pacman's optimal policy is always to go upwards.

■ Possible with large neural net ■ Possible with weighted linear sum ☐ Not Possible

(ii) [2 pts] We draw a vertical line and divide the Gridworld into two halves. On the left half, Pacman's optimal policy is to go upwards, and on the right half, Pacman's optimal policy is to go downwards.

■ Possible with large neural net ☐ Possible with weighted linear sum ☐ Not Possible

(iii) [2 pts] We draw a vertical line and divide the Gridworld into two equal halves. On the left half, Pacman's optimal policy is to go upwards, and on the right half, Pacman's optimal policy is to go right.

■ Possible with large neural net ☐ Possible with weighted linear sum ☐ Not Possible

The key in this part is understanding that a large enough neural net can approximate any function, so it is always possible to use a large neural net to represent the Q-functions and express a certain policy.

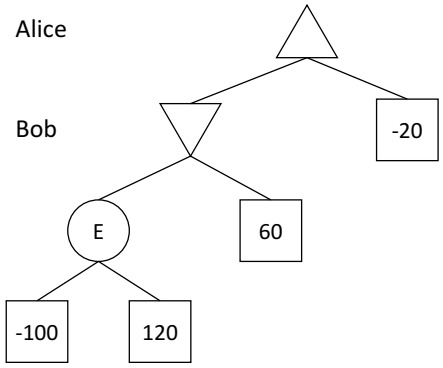
The first is possible with a weighted linear sum, for example, $0 * x + 1 * y$, so Q-values are larger for states with larger y values.

The second is not possible with a weighted linear sum. To make going up optimal, Q-values with larger y-values must be similarly larger. In the right half, this must be the opposite. However, the x features for two positions (x, y+1) and (x, y-1), e.g., one position is higher than the other, are the same, meaning that the x feature can only create offsets between the weighted linear sums for those states and not actually reverse the magnitude to cause (x, y-1) to have a larger Q-value than (x, y+1). That is, for some weights w_x, w_y, the difference in Q-values between (x, y+1), (x, y-1) is $w_x*x+w_y(y+1)-(w_x*x+w_y(y-1)) = w_x*x+w_y*y+w_y-w_x*x-w_y*y+w_y = 2w_y$, and is independent of both x-position and w_x.

The third is not possible with a weighted linear sum. For two positions (x, y+1) and (x + 1, y), we can calculate the difference in Q-values for the given positions given any weights w_x, w_y. The difference is then $w_x * x + w_y(y+1) - (w_x(x+1) + w_y * y) = w_x * x + w_y * y + w_y - w_x * x - w_x - w_y * y = w_y - w_x$. We see then that this is actually completely independent of the positions and the difference will therefore always be fixed; Pacman cannot prefer to go up in some situations and right in others.

Q5. [16 pts] Value of Asymmetric Information

Alice and Bob are playing an adversarial game as shown in the game tree below. Alice (the MAX player) and Bob (the MIN player) are both rational and they both know that their opponent is also a rational player. The game tree has one chance node E whose outcome can be either $E = -100$ or $E = +120$ with equal 0.5 probability.



Each player's utility is equal to the amount of money he or she has. The value x of each leaf node in the game tree means that Bob will pay Alice x dollars after the game, so that Alice and Bob's utilities will be x and $-x$ respectively.

(a) [2 pts] Suppose neither Alice nor Bob knows the outcome of E before playing. What is Alice's expected utility?

Answer: 10 E's expectation is 10. Using minimax, both Alice and Bob should go left.

(b) Carol, a good friend of Alice's, has access to E and can *secretly* tell Alice the outcome of E before the game starts (giving Alice the true outcome of E without lying). However, Bob is not aware of any communication between Alice and Carol, so he still assumes that Alice has no access to E.

(i) [1 pt] Suppose Carol secretly tells Alice that $E = -100$. What is Alice's expected utility in this case?

Answer: -20 Here, Bob will still go left as before since he isn't aware of Alice's access to E. Given this, Alice should now choose to go right when $E = -100$.

(ii) [1 pt] Suppose Carol secretly tells Alice that $E = +120$. What is Alice's expected utility in this case?

Answer: 120 Here, Bob will still go left as before since he isn't aware of Alice's access to E. Given this, Alice should now choose to go left when $E = +120$.

(iii) [1 pt] What is Alice's expected utility if Carol secretly tells Alice the outcome of E before playing?

Answer: 50 E is equally likely to be -100 or $+120$. Averaging the two cases above, $-20 * 0.5 + 120 * 0.5 = 50$.

We define the *value of private information* $V_{\text{A}}^{\text{pri}}(X)$ of a random variable X to a player A as the difference in player A's expected utility after the outcome of X becomes a private information to player A, such that A has access to the outcome of X, while other players have no access to X and are not aware of A's access to X.

(iv) [2 pts] In general, the value of private information $V_{\text{A}}^{\text{pri}}(X)$ of a variable X to a player A

- ☐ always satisfies $V_{\text{A}}^{\text{pri}}(X) > 0$ in all cases.
- ■ always satisfies $V_{\text{A}}^{\text{pri}}(X) \geq 0$ in all cases.
- ☐ always satisfies $V_{\text{A}}^{\text{pri}}(X) = 0$ in all cases.
- ☐ can possibly satisfy $V_{\text{A}}^{\text{pri}}(X) < 0$ in certain cases.

Since player A can always choose to ignore this information and act in the same way as if he/she doesn't know this information, player A is guaranteed to obtain at least the same utility as before, so $V_{\text{A}}^{\text{pri}}(X) \geq 0$.

(v) [1 pt] What is $V^{\text{pri}}_{\text{Alice}}(E)$, the value of private information of E to Alice in the specific game tree above?

Answer: 40 Subtracting the answer of (a) from the answer of (b, iii), $50 - 10 = 40$.

(c) David also has access to E, and can make a *public* announcement of E (announcing the true outcome of E without lying), so that both Alice and Bob will know the outcome of E and are both aware that their opponent also knows the outcome of E. Also, Alice cannot obtain any information from Carol now.

(i) [1 pt] Suppose David publicly announces that $E = -100$. What is Alice's expected utility in this case?

Answer: -20 Using minimax with $E = -100$, Bob will go left and Alice will go right.

(ii) [1 pt] Suppose David publicly announces that $E = +120$. What is Alice's expected utility in this case?

Answer: 60 Using minimax with $E = +120$, Bob will go right and Alice will go left.

(iii) [1 pt] What is Alice's expected utility if David makes a public announcement of E before the game starts?

Answer: 20 E is equally likely to be -100 or $+120$. Averaging the two cases above, $-20 * 0.5 + 60 * 0.5 = 20$.

We define the *value of public information* $V^{\text{pub}}_{\text{A}}(X)$ of a random variable X to a player A as the difference in player A's expected utility after the outcome of X becomes a public information, such that everyone has access to the outcome of X and is aware that all other players also have access to X.

(iv) [2 pts] In general, the value of public information $V^{\text{pub}}_{\text{A}}(X)$ of a variable X to a player A

- ☐ always satisfies $V^{\text{pub}}_{\text{A}}(X) > 0$ in all cases.
- ☐ always satisfies $V^{\text{pub}}_{\text{A}}(X) \geq 0$ in all cases.
- ☐ always satisfies $V^{\text{pub}}_{\text{A}}(X) = 0$ in all cases.
- ■ can possibly satisfy $V^{\text{pub}}_{\text{A}}(X) < 0$ in certain cases.

Player A's utility may decrease if the outcome of X becomes a public information, since other players can now exploit this information to better play against player A, especially in an adversarial setting.

(v) [1 pt] What is $V^{\text{pub}}_{\text{Alice}}(E)$, the value of public information of E to Alice in the specific game tree above?

Answer: 10 Subtracting the answer of (a) from the answer of (c, iii), $20 - 10 = 10$.

(vi) [2 pts] Let $a = V^{\text{pub}}_{\text{Alice}}(E)$ be the value of public information of E to Alice. Suppose David will publicly announce the outcome of E if anyone (either Alice or Bob) pays him b dollars ($b > 0$), and will make no announcement otherwise. Which of the following statements are True?

- ■ The value of public information of E to Bob is $V^{\text{pub}}_{\text{Bob}}(E) = -a$.
- ■ If $b < a$, then Alice should pay David b dollars.
- ☐ If $b > a$, then Bob should pay David b dollars.
- ■ If $b < -a$, then Bob should pay David b dollars.
- ☐ If $b > -a$, then Alice should pay David b dollars.
- ☐ There exists some value $b > 0$ such that both Alice and Bob should pay David b dollars.
- ■ There exists some value $b > 0$ such that neither Alice nor Bob should pay David b dollars.

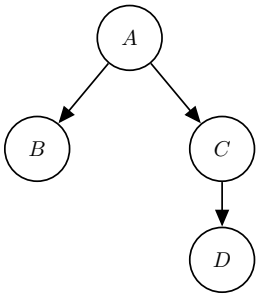
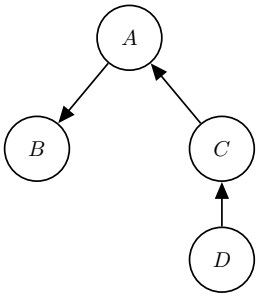
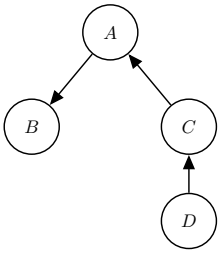
Since Alice and Bob's utilities always sum up to zero, if Alice's utility increases by a after the outcome of E becomes a public information, then Bob's utility will certainly decrease by a, so $V^{\text{pub}}_{\text{Bob}}(E) = -a$.

Alice should pay when $b < V^{\text{pub}}_{\text{Alice}}(E) = a$, and Bob should pay when $b < V^{\text{pub}}_{\text{Bob}}(E) = -a$, which cannot happen simultaneously since $b > 0$. When b is large enough ($b > |a|$), then neither Alice nor Bob should pay for the announcement.

Q6. [12 pts] Bayes Net Modeling

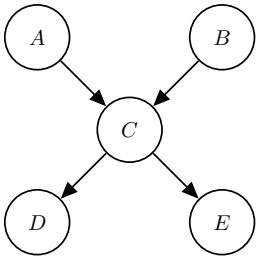
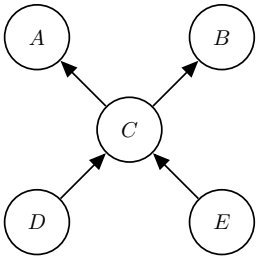
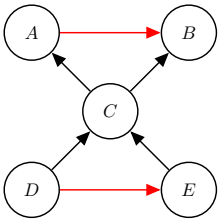
(a) Modeling Joint Distributions For each of the Bayes Net (BN) models of the true data distribution, indicate if the new Bayes Net model is guaranteed to be able to represent the true **joint** distribution. If it is not able to, draw the **minimal** number of edges such that the resulting Bayes Net can capture the **joint** distribution, or indicate if it is not possible.

(i) [2 pts]

BN Model of True Data Distribution	New Bayes Net Model	Can new BN represent joint distribution of True Data Distribution?	If no, draw arrows needed
		● Yes ○ No	 ○ Not Possible

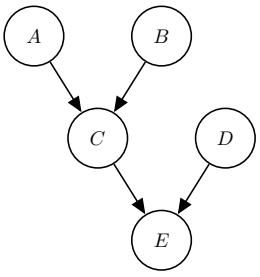
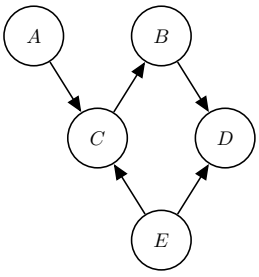
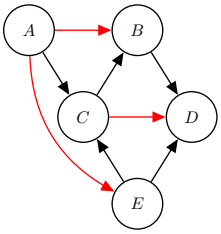
The two Bayes Net models make the same independence assumptions.

(ii) [2 pts]

BN Model of True Data Distribution	New Bayes Net Model	Can new BN represent joint distribution of True Data Distribution?	If no, draw arrows needed
		○ Yes ● No	 ○ Not Possible

The new Bayes Net model encodes the independence assumptions $A \perp\!\!\!\perp B|C$ and $D \perp\!\!\!\perp E$, which are not present in the BN model of the true data distribution. The edge $A \rightarrow B$ removes the assumption $A \perp\!\!\!\perp B|C$, and the edge $D \rightarrow E$ removes the assumption $D \perp\!\!\!\perp E$. Using a different direction for either/both edges is also correct.

(iii) [2 pts]

BN Model of True Data Distribution	New Bayes Net Model	Can new BN represent joint distribution of True Data Distribution?	If no, draw arrows needed
		○ Yes ● No	 ○ Not Possible

The new Bayes Net model encodes the independence assumptions $A \perp\!\!\!\perp B|C$, $A \perp\!\!\!\perp E$, $C \perp\!\!\!\perp D|B, E$, and $A \perp\!\!\!\perp D|B, E$, which are not present in the BN model of the true data distribution. The edge $A \rightarrow B$ removes $A \perp\!\!\!\perp B|C$. The edge $A \rightarrow E$ removes $A \perp\!\!\!\perp E$. The edge $C \rightarrow D$ removes $C \perp\!\!\!\perp D|B, E$ and $A \perp\!\!\!\perp D|B, E$. Using a different edge direction $A \leftarrow E$ is also correct.

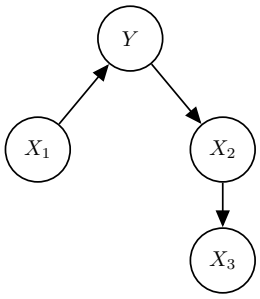
(b) Bayes Nets and Classification Recall from class that we can use Bayes Nets for classification by using the conditional distribution of $P(Y|X_1, X_2, \ldots, X_n)$, where Y is the class and each of the X_i are the observed features.

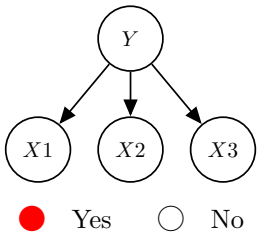
Assume all we know about the true data distribution is that it can be represented with the "True Distribution Model" Bayes Net structure. Indicate if the new Bayes Net models are guaranteed to be able to represent the true **conditional** distribution, $P(Y|X_1, X_2, \ldots, X_n)$. Mark "Yes" if it can be represented and "No" otherwise.

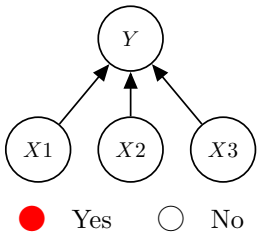
For all subparts of this problem, the answer to the rightmost question (with edges $X_1 \to Y$, $X_2 \to Y$, ...) is "Yes". These models contain a factor $P(Y|X_1, X_2, \ldots, X_n)$ that can represent any conditional distribution.

(i) [2 pts]

True Distribution Model



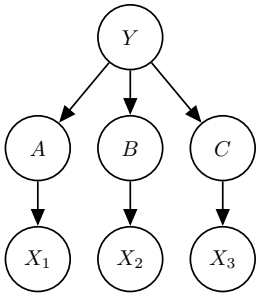


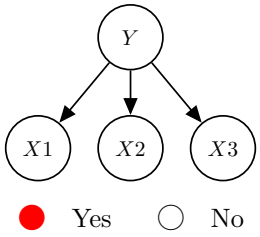


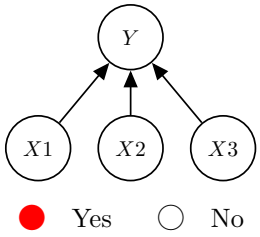
For the left Yes/No question: based on the true distribution model, we can conclude that $P(Y|X_1, X_2, X_3) = P(Y|X_1, X_2)$. In other words, X_3 is not required to model the conditional distribution. If we set $P(X_3 = x_3) = \textit{constant}$ for all values x_3 in its domain, the new Bayes Net model will also have $P(Y|X_1, X_2, X_3) = P(Y|X_1, X_2)$. With X_3 out of the picture, the remaining difference between the two models is the direction of a single edge between Y and X_1, which does not affect the ability to model the true conditional distribution.

(ii) [2 pts]

True Distribution Model



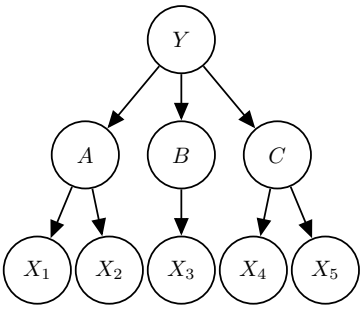


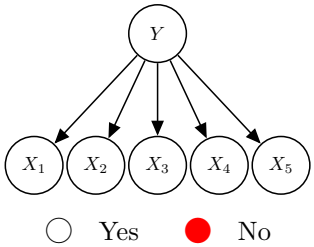


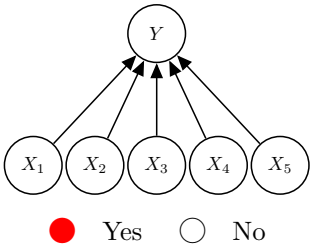
For the left Yes/No question: starting with the true distribution model, we run variable elimination to eliminate A, B, and C. The result would be exactly the model shown in the middle column, so we can conclude that the answer is "Yes": the new model can represent the true conditional distribution.

(iii) [2 pts]

True Distribution Model







For the left Yes/No question: the true data distribution can have the property that $Y = X_1$ XOR X_2, constructed as follows. Let Y, X_1, and X_2 be binary random variables. Let A take on values from the set $\{(0,0),(0,1),(1,0),(1,1)\}$. Let $P(A = (0,0)|Y = 0) = P(A = (1,1)|Y = 0) = 0.5$ and $P(A = (0,1)|Y = 1) = P(A = (1,0)|Y = 1) = 0.5$. Let X_1 equal the first element of A's value with probability 1, and let X_2 equal the second element of A's value with probability 1.

However, the Naive Bayes model has $X_1 \perp\!\!\!\perp X_2|Y$, which can't represent the XOR function. As a result, the Naive Bayes model can't represent the true distribution model.

SID: _____

Q7. [14 pts] Help the Farmer!

Chris is a farmer. He has a hen in his barn, and it will lay at most one egg per day. Chris collects data and discovers conditions that influence his hen to lay eggs on a certain day, which he describes below.

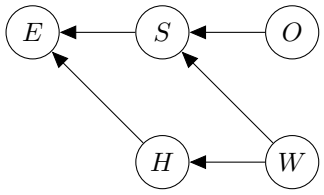
O	P(O)
$+o$	0.1
$-o$	0.9

W	P(W)
$+w$	0.7
$-w$	0.3

H	W	P(H|W)
$+h$	$+w$	0.9
$+h$	$-w$	0.5
$-h$	$+w$	0.1
$-h$	$-w$	0.5

S	W	O	P(S|W,O)
$+s$	$+w$	$+o$	0.6
$+s$	$+w$	$-o$	0.1
$+s$	$-w$	$+o$	0.8
$+s$	$-w$	$-o$	0.1
$-s$	$+w$	$+o$	0.4
$-s$	$+w$	$-o$	0.9
$-s$	$-w$	$+o$	0.2
$-s$	$-w$	$-o$	0.9

E	H	S	P(E|H,S)
$+e$	$+h$	$+s$	0.4
$+e$	$+h$	$-s$	0.8
$+e$	$-h$	$+s$	0.2
$+e$	$-h$	$-s$	0.6
$-e$	$+h$	$+s$	0.6
$-e$	$+h$	$-s$	0.2
$-e$	$-h$	$+s$	0.8
$-e$	$-h$	$-s$	0.4



For a single hen, variables O, W, S, H, and E denote the event of an outbreak (O), sunny weather (W), sickness (S), happiness (H), and egg being laid (E). If an event does occur, we denote it with a $+$, otherwise $-$, e.g., $+o$ denotes an outbreak having occurred and $-o$ denotes no outbreak occurred.

(a) Suppose Chris wants to estimate the probability that the hen lays an egg given it's good weather and the hen is not sick, e.g., $P(+e|+w,-s)$. Suppose we receive the samples:

$$(-o,+w,-s,-h,+e), \quad (-o,+w,-s,+h,-e), \quad (+o,+w,-s,-h,-e)$$

(i) [2 pts] Similar to the likelihood weighing method, Chris weighs each of his samples after fixing evidence. However, he weighs each sample only with $P(-s|+w,O)$, i.e. he omits weighing by $P(+w)$. Chris' method results in the correct answer for the query $P(+e|+w,-s)$.

● True ○ False

Because the probability $P(+w)$ is constant, by not including it, the query after normalizing would be correct.

(ii) [2 pts] Using likelihood weighting with the samples listed above, what is the probability the hen lays an egg given it's good weather and the hen is not sick, or $P(+e|+w,-s)$? Round your answer to the second decimal place or express it as a fraction simplified to the lowest terms.

0.41

The weights are $0.7 * .9$ for the first two and $0.7 * .4$. This gives us the estimation $\frac{0.7*0.9}{0.7*0.9*2+0.7*0.4}$, or a probability of 0.41.

(b) Chris uses Gibbs sampling to sample tuples of (O, W, S, H, E).

(i) [2 pts] As a step in our Gibbs sampling, suppose we currently have the assignment of $(-o,-w,+s,+h,+e)$. Then suppose we resample the "sickness" variable, i.e., S. What is the probability that the next assignment is the same, i.e., $(-o,-w,+s,+h,+e)$? Round your answer to the second decimal point, or express it as a fraction simplified to the lowest terms.

.05

This is asking for the probability $P(+s|-o,-w,+e,+h)$. Mathematically, this is $\frac{P(+s,-o,-w,+e,+h)}{P(+s,-o,-w,+e,+h)+P(-s,-o,-w,+e,+h)} = \frac{0.9*0.3*0.5*0.1*0.4}{0.9*0.3*0.5*0.1*0.4+0.9*0.3*0.5*0.9*0.8} = \frac{0.0054}{0.0054+0.0972}$. This gives us .053.

(ii) [2 pts] What will be the most observed tuple of (O, W, S, H, E) if we keep running Gibbs sampling for a long time? Select one value from each column below to denote the assignment.

☐ $+o$	■ $+w$	☐ $+s$	■ $+h$	■ $+e$
■ $-o$	☐ $-w$	■ $-s$	☐ $-h$	☐ $-e$

The most observed sample should be good weather, no outbreak, no sickness, happiness, and laying an egg. This yields probability of $0.7 * 0.9 * 0.9 * 0.9 * 0.8 = .40824$.

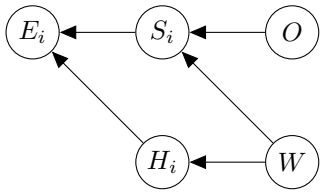
(c) [3 pts] Suppose we adopt a sampling procedure where at each evidence node with probability 0.5 we fix to the evidence, otherwise we sample the outcome and reject if it doesn't match. Upon seeing an evidence node, write an expression for the value we will **multiply into the weight** of the sample to make this procedure consistent. The weight is initialized at the start as `weight = 1`.

Your answer may use the variables s and p, where s is 1 if the coin flip told us to sample the evidence node, and p is the probability of the evidence given its parents.

`weight *=` $(1-s)p+s$

When the coin flip tells us to fix the evidence, we treat it like likelihood weighting and multiply by p. When the coin flip tells us to sample, we treat it like rejection sampling, so we don't need to change the weights. A nice way of writing this is $(1-s)p+s$. Technically, we can also multiply by any constant (e.g. .5 or p) since if every sample weight is scaled this cancels when we normalize weights to estimate probabilities. We also accepted answers written as piece-wise functions.

Now, suppose there are 1000 hens, each independently modeled by the Bayes Net model below. Denote the random variables for sickness, happiness, and laying an egg as S_i, H_i, E_i for hen i. The conditional probability tables are the same as above for each hen.



(d) [3 pts] One day, Chris observed that all the hens lay eggs and the weather is bad. What's the probability of an outbreak happening? Round your answer to the second decimal point. *Hint:* $P(O=+o, W=-w, E_i=+e_i) = 0.0114$ and $P(O=-o, W=-w, E_i=+e_i) = 0.1782$ for all i.

0

$P(+e_i|+o,-w) = 0.0114/P(+o,-w) = 0.0114/0.1/0.3 = 0.38$

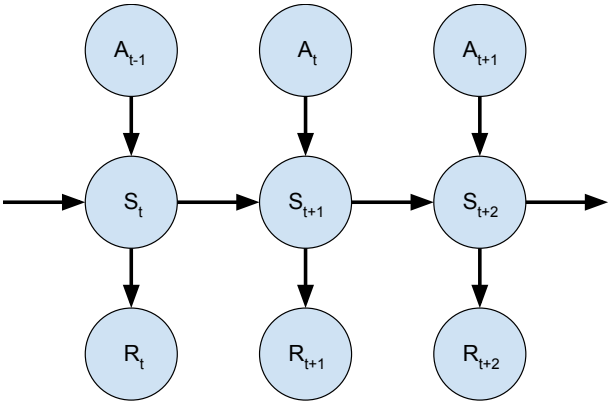
$P(+e_i|-o,-w) = 0.1782/P(-o,-w) = 0.1782/0.9/0.3 = 0.66$

$P(O|+e_1,...,+e_{1000},-w) \propto P(O,+e_1,...,+e_{1000},-w) = \prod_{i=1}^{1000} P(+e_i|O,-w)P(O)P(-w)$

By normalizing this, $P(+o,+e_1,...,+e_1000,-w) = \frac{(0.38)^{1000}*0.1}{(0.38)^{1000}*0.1+(0.66)^{1000}*0.9} = \frac{1}{1+(66/38)^{1000}*9} \approx 0.$

Q8. [15 pts] Bayes Nets and RL

In this question, you will see that variable elimination can solve reinforcement learning problems. Consider the following Bayes net, where $S_t \in \mathcal{S}$, $A_t \in \mathcal{A}$, and $R_t \in \{0, 1\}$:



(a) [2 pts] From the list below, select all the (conditional) independencies that are guaranteed to be true in the Bayes net above:

- [x] $S_{t+1} \perp\!\!\!\perp S_{t-1} | S_t, A_t$
- [x] $R_{t+1} \perp\!\!\!\perp R_{t-1} | S_t, A_t$
- [ ] $R_{t+1} \perp\!\!\!\perp R_t$
- [x] $A_{t+1} \perp\!\!\!\perp R_t | S_t$
- [x] $A_{t+1} \perp\!\!\!\perp R_t | S_t, A_t$
- [ ] None of the above

Let $+r_{t:T}$ denote the event $R_t = R_{t+1} = \cdots = R_T = 1$, and assume that $P(a_t) = 1/|\mathcal{A}|$. Define the following functions:

$$\beta_t(s_t, a_t) = P(+r_{t:T} | s_t, a_t), \qquad \beta_t(s_t) = P(+r_{t:T} | s_t) = \frac{1}{|\mathcal{A}|} \sum_{a_t} \beta_t(s_t, a_t)$$

Perform variable elimination to compute $P(A_t | S_t, +r_{t:T})$.

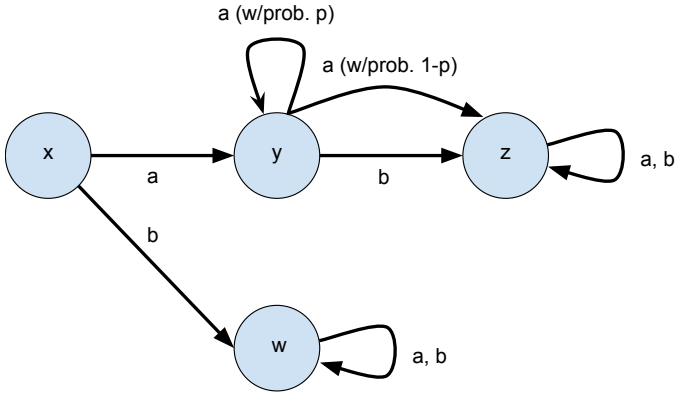
(b) [2 pts] Which of the following recursions does $\beta_t(s_t, a_t)$ satisfy?

- ○ $\beta_t(s_t, a_t) = P(+r_t | s_t) \sum_{s_{t+1}} \beta_{t+1}(s_{t+1})$
- ○ $\beta_t(s_t, a_t) = P(+r_t | s_t) \sum_{s_{t+1}} \beta_{t+1}(s_{t+1}, a_t)$
- ○ $\beta_t(s_t, a_t) = P(+r_{t+1} | s_t) \sum_{s_{t+1}} \beta_{t+1}(s_{t+1}, a_t)$
- ● $\beta_t(s_t, a_t) = P(+r_t | s_t) \sum_{s_{t+1}} P(s_{t+1} | s_t, a_t) \beta_{t+1}(s_{t+1})$
- ○ $\beta_t(s_t, a_t) = \sum_{a_{t+1}} P(+r_{t+1} | s_{t+1}) \frac{1}{|\mathcal{A}|} \sum_{s_{t+1}} P(s_{t+1} | s_t, a_t) \beta_{t+1}(s_{t+1})$
- ○ None of the above

(c) [2 pts] Write $P(a_t | s_t, +r_{1:T})$ in terms of $\beta_t(s_t, a_t)$, $\beta_t(a_t)$, and relevant probabilities from the Bayes net.

- ○ $P(a_t | s_t, +r_{1:T}) = \frac{\beta_t(s_t, a_t)}{\beta_t(s_t)} P(+r_t | s_t)$
- ● $P(a_t | s_t, +r_{1:T}) = \frac{\beta_t(s_t, a_t)}{\sum_{a'_t} \beta_t(s_t, a'_t)}$
- ○ $P(a_t | s_t, +r_{1:T}) = \frac{\beta_t(s_t, a_t)}{\beta_t(s_t)}$
- ○ $P(a_t | s_t, +r_{1:T}) = \frac{\beta_t(s_t, a_t)}{P(+r_t | s_t) |\mathcal{A}|}$
- ○ $P(a_t | s_t, +r_{1:T}) = \frac{P(+r_t | s_t)}{\beta_t(s_t) \beta_t(s_t, a_t) |\mathcal{A}|}$
- ○ None of the above

So far, we have only discussed variable elimination in a certain Bayes net. Now, we will associate the Bayes net with an MDP with two parameters $p, q \in (0, 1)$. The states are $\mathcal{S} = \{x, y, z, w\}$, and the actions are $\mathcal{A} = \{a, b\}$. The transitions are described in this diagram:



All transitions are deterministic except when taking action a starting in state y – this transition is determined by $P(S_{t+1} = y \mid S_t = y, A_t = a) = p$. The rewards are stochastic and depend only on state, taking on values in $\{0, 1\}$ with probabilities

$$P(+r_t|S_t = x) = 1, \quad P(+r_t|S_t = y) = 1, \quad P(+r_t|S_t = z) = 0, \quad P(+r_t|S_t = w) = q$$

Throughout the following questions, assume that $p, q \in (0, 1)$.

(d) [3 pts] Consider running the uniform policy $\pi_{\text{uniform}}(A_t = a) = \frac{1}{2}$ for $T + 2$ timesteps starting in state x. What is $P(A_1 = a|S_1 = x, +r_{1:T+2})$?

- ○ $p^{T+1}/(p^{T+1} + (2q)^{T+1})$
- ○ $(2q)^{T+1}/(p^{T+1} + (2q)^{T+1})$
- ● $p^T/(p^T + 2^T q^{T+1})$
- ○ $2^T q^{T+1}/(p^T + 2^T q^{T+1})$
- ○ $p^T/(p^T + (2q)^T)$
- ○ $(2q)^T/(p^T + (2q)^T)$
- ○ $p^T/(p^T + q^T)$
- ○ $q^T/(p^T + q^T)$
- ○ $p^{T+1}/(p^{T+1} + q^{T+1})$
- ○ $q^{T+1}/(p^{T+1} + q^{T+1})$
- ○ None of the above

Since $P(A_1 = a, +r_{1:T+2}|S_1 = x) = \frac{1}{2}\left(\frac{1}{2}p\right)^T$ and $P(A_1 = b, +r_{1:T+2}|S_1 = x) = \frac{1}{2}q^{T+1}$, we get $P(A_1 = a|S_1 = x, +r_{1:T+2}) = p^T/(p^T + 2^T q^{T+1})$.

(e) [2 pts] Suppose $p > 2q^{(T+1)/T}$. When running π_{uniform}, what is $\arg\max_z P(A_1 = z|S_1 = x, +r_{1:T+2})$?

- ■ $\arg\max_z P(A_1 = z|S_1 = x, +r_{1:T+2}) = a$
- ☐ $\arg\max_z P(A_1 = z|S_1 = x, +r_{1:T+2}) = b$
- ☐ Cannot be determined

If $p > 2q^{(T+1)/T}$, then $P(A_1 = a|S_1 = x, +r_{1:T+2}) > 1/2$, so a is the answer.

(f) [2 pts] Suppose $q > 2^{-T/(T+1)}$. When running π_{uniform}, what is $\arg\max_z P(A_1 = z|S_1 = x, +r_{1:T+2})$?

- ☐ $\arg\max_z P(A_1 = z|S_1 = x, +r_{1:T+2}) = a$
- ■ $\arg\max_z P(A_1 = z|S_1 = x, +r_{1:T+2}) = b$
- ☐ Cannot be determined

If $q > 2^{-T/(T+1)}$, then $P(A_1 = a|S_1 = x, +r_{1:T+2}) < p^T/(p^T + 1) < 1/2$ because $p < 1$. So, the answer is b.

(g) [2 pts] Consider running the optimal policy π^* for $T + 1$ timesteps starting in state x. When is π^* always guaranteed to choose b as its first action?

- (●) $T > \frac{1}{(1-p)q}$
- ( ) $T > \frac{1}{(1-q)p}$
- ( ) $T > \frac{1}{(1-p)(1-q)}$
- ( ) $T > \frac{1}{pq}$
- ( ) $T < \frac{1}{(1-p)q}$
- ( ) $T < \frac{1}{(1-q)p}$
- ( ) $T < \frac{1}{(1-p)(1-q)}$
- ( ) $T < \frac{1}{pq}$
- ( ) None of the above

In state y, the optimal policy will always choose action a, and its value is at most $1/(1-p)$. Meanwhile, in state w, the value is always qT. So, if $qT > 1/(1-p)$, the optimal policy must go to state w, meaning that it must take action b at x.

Q9. [8 pts] Decision Trees

You are given a dataset for training a decision tree. The goal is to predict the label (+ or -) given the features A, B, and C.

A	B	C	label
0	0	0	+
0	0	1	+
0	1	0	+
0	1	1	-
1	0	0	-
1	0	1	-
1	1	0	+
1	1	1	-

First, consider building a decision tree by greedily splitting according to information gain.

(a) [2 pts] Which features could be at the root of the resulting tree? Select all possible answers.

- ■ A
- ☐ B
- ■ C

A and C yield maximal information gain at the root.

(b) [2 pts] How many edges are there in the longest path of the resulting tree? Select all possible answers.

- ☐ 1
- ☐ 2
- ■ 3
- ☐ 4
- ☐ None of the above

Regardless of the choice of the feature at the root, the resulting tree needs to consider all 3 features in a path, so there are 3 edges in that path.

Now, consider building a decision tree with the smallest possible height.

(c) [2 pts] Which features could be at the root of the resulting tree? Select all possible answers.

- ☐ A
- ■ B
- ☐ C

The optimal decision tree first splits on B. For the B=0 branch, the next split is on A; for the B=1 branch, the next split is on C.

(d) [2 pts] How many edges are there in the longest path of the resulting tree? Select all possible answers.

- ☐ 1
- ■ 2
- ☐ 3
- ☐ 4
- ☐ None of the above

As can be seen from the answer to part (c), the optimal tree has two edges per path from the root to any leaf.

THIS PAGE IS INTENTIONALLY LEFT BLANK

SCRATCH PAPER – INTENTIONALLY BLANK – PLEASE DETACH ME

SCRATCH PAPER – INTENTIONALLY BLANK – PLEASE DETACH ME

SCRATCH PAPER – INTENTIONALLY BLANK – PLEASE DETACH ME