CS 188 Spring 2017 Introduction to Artificial Intelligence Final v0

- You have approximately 165 minutes (2 hours 45 minutes).
- The exam is closed book, closed calculator, and closed notes except your one-page crib sheet.
- Mark your answers ON THE EXAM ITSELF. If you are not sure of your answer you may wish to provide a *brief* explanation. All short answer sections can be successfully answered in a few sentences AT MOST.
- For multiple choice questions with *circular bubbles*, you should only mark ONE option; for those with *checkboxes*, you should mark ALL that apply (which can range from zero to all options)

First name	
Last name	
edX username	

For staff use only:

Total	/??

THIS PAGE IS INTENTIONALLY LEFT BLANK

Q1. [?? pts] Approximate Q-Learning

Consider the following MDP: We have infinitely many states $s \in \mathbb{Z}$ and actions $a \in \mathbb{Z}$, each represented as an integer. Taking action a from state s deterministically leads to new state $s' = s + a$ and reward $r = s - a$. For example, taking action 3 at state 1 results in new state $s' = 1 + 3 = 4$ and reward $r = 1 - 3 = -2$.

We perform approximate Q-Learning, with features and initialized weights defined below.

Feature	Initial Weight
$f_1(s, a) = s$	$w_1 = 1$
$f_2(s, a) = -a^2$	$w_2 = 2$

(a) [?? pts] Write down $Q(s, a)$ in terms of w_1, w_2, s, and a.

$Q(s, a) = w_1 * s - w_2 * a^2$

(b) [?? pts] Calculate $Q(1, 1)$.

$Q(1, 1) = w_1 f_1(1, 1) + w_2 f_2(1, 1) = 1 * 1 - 2 * 1^2 = -1$

(c) [?? pts] We observe a sample (s, a, r, s') of $(1, 1, 0, 2)$. Assuming a learning rate of $\alpha = 0.5$ and discount factor of $\gamma = 0.5$, compute new weights after a single update of approximate Q-Learning.

$$diff = (0 + 0.5 * \max_a Q(2, a)) - (-1)$$

$$diff = (0.5 \max_a (2 - 2 * a^2)) + 1$$

$$diff = (0.5(2 + 2 * \max_a (-a^2))) + 1$$

$$diff = (0.5(2 + 2 * 0)) + 1$$

$$diff = 1 + 1 = 2$$

w_1: $1 + 0.5 * 2 * 1 = 2$

w_2: $2 + 0.5 * 2 * -1^2 = 1$

(d) [?? pts] Compute the new value for Q(1,1).

Q(1,1) = $w_1 * 1 + w_2 - 1^2 = 2 * 1 + 1 * -1^2 = 1$

Q2. [?? pts] Who Spoke When

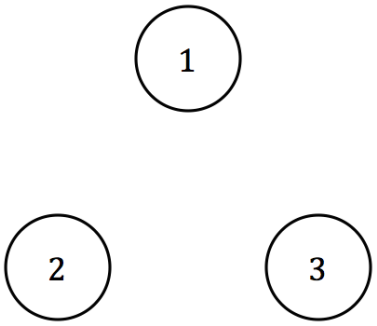
We are given a single audio recording (divided into equal and short time slots) and wish to infer when each person speaks. At every time step exactly one of N people is talking. This problem can be modeled using an HMM. Hidden variable $X_t \in \{1, 2, ..., N\}$ represents which person is talking at time step t.

(a) For this part, assume that at each time step:

- with probability p, the current speaker will continue to talk in the next time step.
- with probability $1 - p$, the current speaker will be interrupted by another person. Each other person is equally likely to be the interrupter.

Assume that $N = 3$.

(i) [?? pts] Complete the Markov Chain below and write down the probabilities on each transition.



Self transitions for all states with probability p and all other transitions with probability $(1 - p)/2$

(ii) [?? pts] What is the stationary probability distribution of this Markov chain? (Again, assume $N = 3$).

$P(X_{\text{inf}} = 1) =$ 1/3

$P(X_{\text{inf}} = 2) =$ 1/3

$P(X_{\text{inf}} = 3) =$ 1/3

1/3 for each state, because of the symmetry.

(b) [?? pts] What is the **number of parameters** (or degrees of freedom) needed to model the transition probability $P(X_t|X_{t-1})$? Assume N people in the meeting and arbitrary transition probabilities.

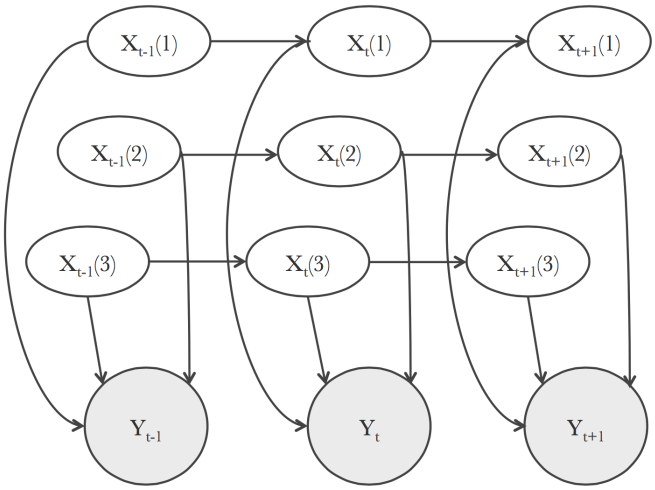
$N(N-1)$ $P(X_t|X_{t-1})$ is $N \times N$ and each row should sum to one. Significant partial credit will be given to the answer N^2.

(c) [?? pts] Let's remove the assumption that people are not allowed to talk simultaneously. Now, hidden state $X_t \in \{0, 1\}^N$ will be a binary vector of length N. Each element of the vector corresponds to a person, and whether they are speaking.

Now, what is the **number of parameters** (or degrees of freedom) needed for modeling the transition probability $P(X_t|X_{t-1})$?

$2^N(2^N - 1)$ We have 2^N different states so we need $2^N(2^N - 1)$ or roughly 2^{2N} parameters.

One way to decrease the parameter count is to assume independence. Assume that the transition probability between people is independent. The figure below represents this assumption for $N = 3$, where $X_t = [X_t(1), X_t(2), X_t(3)]$.



(d) [**??** pts] Write the following in terms of conditional probabilities given from the Bayes Net. Assume N people in the meeting.

Transition Probability $P(X_t|X_{t-1})$

$P(X_t|X_{t-1}) = \prod_{n=1}^{N} P(X_t(n)|X_{t-1}(n))$

Emission Probability $P(Y_t|X_t)$

$P(Y_t|X_t) = P(Y_t|X_t(1), \cdots, X_t(N))$

(e) [**??** pts] What is the **number of parameters** (or degrees of freedom) needed for modeling transition probability $P(X_t|X_{t-1})$? Assume N people in the meeting.

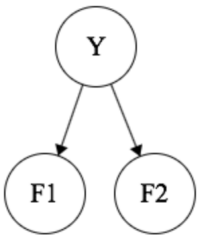
$2N$

We need to define a transition matrix for each person which requires 2 parameters and there are N people. Significant partial credit for answer $4N$.

Q3. [?? pts] Naive Bayes

(a) We use a Naive Bayes classifier to differentiate between Pacmen and Ghosts, trained on the following:

F_1	F_2	Y
0	1	Ghost
1	0	Ghost
0	0	Pac
1	1	Pac



Assume that the distributions generated from these samples perfectly estimate the CPTs. Given features f_1, f_2, we predict $\hat{Y} \in \{Ghost, Pacman\}$ using the Naive Bayes decision rule. If $P(Y = Ghost|F_1 = f_1, F_2 = f_2) = 0.5$, assign $\hat{Y}$ based on flipping a fair coin.

(i) [?? pts] Compute the table $P(\hat{Y}|Y)$.
Value $P(\hat{Y} = Ghost|Y = Ghost)$ is the probability of correctly classifying a $Ghost$,
while $P(\hat{Y} = Pacman|Y = Ghost)$ is the probability of confusing a $Ghost$ for a $Pacman$.

$P(\hat{Y}\|Y)$	$\hat{Y} = Ghost$	$\hat{Y} = Pacman$
$Y = Ghost$	$\frac{1}{2}$	$\frac{1}{2}$
$Y = Pacman$	$\frac{1}{2}$	$\frac{1}{2}$

For each modification below, recompute table $P(\hat{Y}|Y)$. **The modifications for each part are separate, and do not accumulate.**

(ii) [?? pts] Add extra feature $F_3 = F_1 + F_2$, and modify the Naive Bayes classifier appropriately.

$P(\hat{Y}\|Y)$	$\hat{Y} = Ghost$	$\hat{Y} = Pacman$
$Y = Ghost$	1	0
$Y = Pacman$	0	1

(iii) [?? pts] Add extra feature $F_3 = F_1 \times F_2$, and modify the Naive Bayes classifier appropriately.

$P(\hat{Y}\|Y)$	$\hat{Y} = Ghost$	$\hat{Y} = Pacman$
$Y = Ghost$	1	0
$Y = Pacman$	$\frac{1}{2}$	$\frac{1}{2}$

(iv) [?? pts] Add extra feature $F_3 = F_1 - F_2$, and modify the Naive Bayes classifier appropriately.

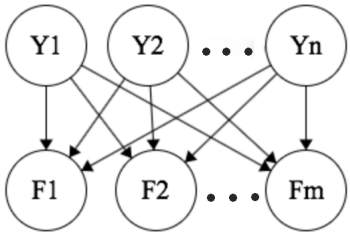
$P(\hat{Y}\|Y)$	$\hat{Y} = Ghost$	$\hat{Y} = Pacman$
$Y = Ghost$	1	0
$Y = Pacman$	0	1

(v) [?? pts] Perform Laplace Smoothing with $k = 1$.

$P(\hat{Y}\|Y)$	$\hat{Y} = Ghost$	$\hat{Y} = Pacman$
$Y = Ghost$	$\frac{1}{2}$	$\frac{1}{2}$
$Y = Pacman$	$\frac{1}{2}$	$\frac{1}{2}$

(b) [?? pts] Now, we reformulate the Naive Bayes classifier so that it can choose more than one class. For example, if we are choosing which genre a book is, we want the ability to say that a romantic comedy is *both* a romance and a comedy.

To do this, we have multiple label nodes $Y = \{Y_1 ... Y_n\}$ which all point to all features $F = \{F_1 ... F_m\}$.



Select all of the following expressions which are valid Naive Bayes classification rules, i.e., equivalent to $\arg\max_{Y_1 ... Y_n} P(Y_1, Y_2, ..., Y_n | F_1, F_2, ..., F_m)$:

☐ $\arg\max_{Y_1 ... Y_n} \prod_i^n \left[P(Y_i) \prod_j^m P(F_j | Y_i) \right]$

☐ $\arg\max_{Y_1 ... Y_n} \prod_i^n \left[P(Y_i) \prod_j^m P(F_j | Y_1 ... Y_n) \right]$

■ $\arg\max_{Y_1 ... Y_n} \prod_i^n [P(Y_i)] \prod_j^m [P(F_j | Y_1 ... Y_n)]$

☐ $\prod_i^n \left[\arg\max_{Y_i} \left\{ P(Y_i) \prod_j^m P(F_j | Y_i) \right\} \right]$

☐ $\prod_i^n \left[\arg\max_{Y_i} \left\{ P(Y_i) \prod_j^m P(F_j | Y_1 ... Y_n) \right\} \right]$

Q4. [?? pts] Tracking a cyclist

We are trying to track cyclists as they move around a self-driving car. The car is equipped with 4 "presence detectors" corresponding to:

- Front of the car (F),
- Back of the car (B),
- Left side of the car (L),
- Right side of the car (R).

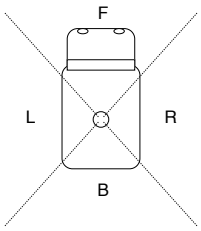


Figure 1: Autonomous vehicle and detection zones

Unfortunately, the detectors are not perfect and feature the following conditional probabilities for detection $D \in \{0, 1\}$ ("no detection" or "detection", respectively) given cyclist presence $C \in \{0, 1\}$ ("no cyclist" or "cyclist", respectively).

Front detector

$P_F(D\|C)$	$d = 1$	$d = 0$
$c = 1$	0.8	0.2
$c = 0$	0.1	0.9

Back detector

$P_B(D\|C)$	$d = 1$	$d = 0$
$c = 1$	0.6	0.4
$c = 0$	0.4	0.6

Left & Right detectors

$P_L(D\|C) = P_R(D\|C)$	$d = 1$	$d = 0$
$c = 1$	0.7	0.3
$c = 0$	0.2	0.8

(a) Detection and dynamics

(i) [?? pts] If you could freely choose any detector to equip all four detection zones, which one would be best?

● The front detector. ○ The detector at the back. ○ The left/right detector.

The front detector features the confusion matrix most concentrated on the diagonal.

Dynamics: We have measured the following transition probabilities for cyclists moving around the car when driving. Assume any dynamics are Markovian. Variable $X_t \in \{f, l, r, b\}$ denotes the location of the cyclist at time t, and can be in front, left, right, or back of the car.

$P(X_{t+1}\|X_t)$	$X_{t+1} = f$	$X_{t+1} = l$	$X_{t+1} = r$	$X_{t+1} = b$
$X_t = f$	p_{ff}	p_{fl}	p_{fr}	p_{fb}
$X_t = l$	p_{lf}	p_{ll}	p_{lr}	p_{lb}
$X_t = r$	p_{rf}	p_{rl}	p_{rr}	p_{rb}
$X_t = b$	p_{bf}	p_{bl}	p_{br}	p_{bb}

(ii) [?? pts] Which criterion does this table have to satisfy for it to be a well defined CPT? (Select all that apply).

■ Each row should sum to 1. ☐ Each column should sum to 1. ☐ The table should sum to 1.

(b) Let's assume that we have been given a sequence of observations $d_1, d_2, \ldots, d_t$ and computed the posterior probability $P(X_t|d_1, d_2, \ldots, d_t)$, which we represent as a four-dimensional vector.

(i) [?? pts] What is vector $P(X_{t+1}|d_1, d_2, \ldots, d_t)$ as a function of $P(X_{t+1}|X_t)$ (a 4×4 matrix written above) and $P(X_t|d_1, d_2, \ldots, d_t)$?

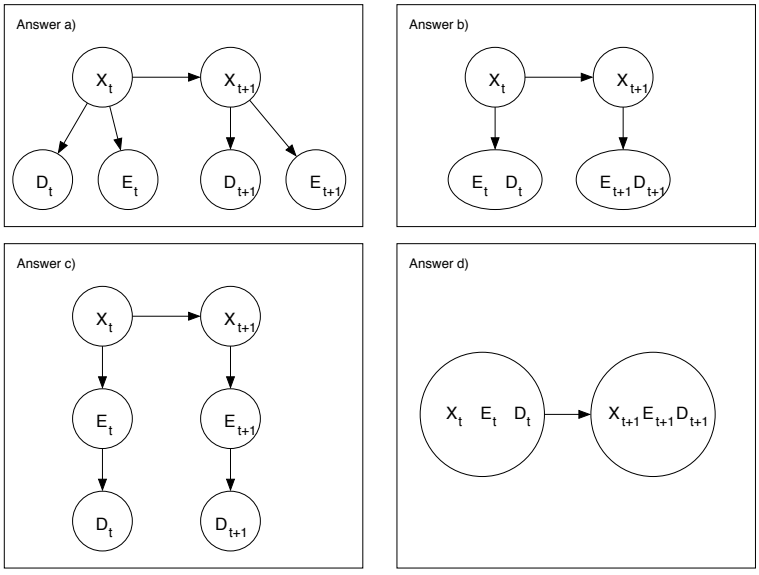
$P(X_{t+1}|D_1, D_2, \ldots, D_t) = P(X_{t+1}|X_t)^T \times P(X_t|D_1, D_2, \ldots, D_t)$
or $P(X_{t+1}|D_1, D_2, \ldots, D_t) = \sum_{x_t} P(X_{t+1}|X_t = x_t) \times P(X_t = x_t|D_1, D_2, \ldots, D_t)$

(ii) [?? pts] What is the computational complexity of computing $P(X_t|D_1 = d_1, D_2 = d_2, \ldots, D_t = d_t)$ as a function of t and the number of states S (using big O notation)?

$O(t \times S^2)$.

Detailed solution: At each time step of the forward algorithm, we need to multiply a vector of size S by a matrix of size S^2 to account for the dynamics entailed in $P(X_{t+1}|X_t)$.
Then, we need to compute the emission probability of each state which here costs $2 \times S$ and normalize (complexity is S). Therefore, the cost of propagating beliefs forward in time through the dynamics dominates as a function of S and is $O(S^2)$ for each time step. Hence the final answer.

(c) **(i)** [?? pts] We now add a radar to the system (random variable $E \in \{f, l, r, b\}$). Assuming the detection by this device is independent from what happens with the pre-existing detectors, which of the probabilistic models *could* you use? If several variables are in the same node, the node represents a tuple of random variables, which itself is a random variable.



Select all that apply.
■ a) ■ b) □ c) ■ d)

(ii) [?? pts] ERRATUM: Which of the following values for Z are correct?

$$P(X_{t+1}|D_1, \ldots, D_{t+1}, E_1, \ldots, E_{t+1}) = \sum_{x=f,l,r,b} \frac{Z \cdot P(X_t = x|D_1, \ldots, D_t, E_1, \ldots, E_t) \cdot P(X_{t+1}|X_t = x)}{P(D_{t+1}, E_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)}.$$

■ $Z = P(E_{t+1}D_{t+1}|X_{t+1}, X_t, D_1, \ldots, D_{t+1}, E_1, \ldots, E_{t+1})$
■ $Z = P(E_{t+1}|X_{t+1})P(D_{t+1}|X_{t+1})$
□ $Z = P(E_{t+1}|X_t)P(D_{t+1}|X_t)$

☐ $Z = P(E_{t+1}|E_t)P(D_{t+1}|D_t)$

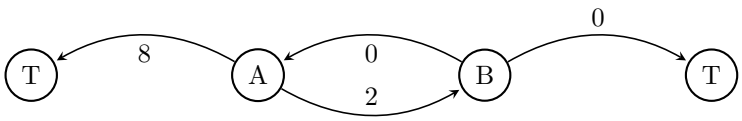
■ $Z = P(E_{t+1}, D_{t+1}|X_{t+1})$

☐ $Z = P(E_{t+1}, D_{t+1}|X_t)$

$$
\begin{aligned}
&P(X_{t+1}|D_1, \ldots, D_{t+1}, E_1, \ldots, E_{t+1}) \\
&= \frac{P(X_{t+1}, D_{t+1}, E_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)}{P(D_{t+1}, E_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)} \\
&= \frac{P(D_{t+1}, E_{t+1}|X_{t+1}, D_1, \ldots, D_t, E_1, \ldots, E_t)P(X_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)}{P(D_{t+1}, E_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)} \\
&= \frac{P(D_{t+1}|X_{t+1})P(E_{t+1}|X_{t+1})\sum_x P(X_{t+1}|X_t = x, D_1, \ldots, D_t, E_1, \ldots, E_t)P(X_t = x|D_1, \ldots, D_t, E_1, \ldots, E_t)}{P(D_{t+1}, E_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)} \\
&= \frac{P(D_{t+1}|X_{t+1})P(E_{t+1}|X_{t+1})\sum_x P(X_{t+1}|X_t = x)P(X_t = x|D_1, \ldots, D_t, E_1, \ldots, E_t)}{P(D_{t+1}, E_{t+1}|D_1, \ldots, D_t, E_1, \ldots, E_t)}
\end{aligned}
$$

Q5. [?? pts] MDP: Left or Right

Consider the following MDP:



The state space $\mathcal{S}$ and action space $\mathcal{A}$ are

$$\mathcal{S} = \{A, B, T\}$$

$$\mathcal{A} = \{\texttt{left}, \texttt{right}\}$$

where T denotes the terminal state (both T states are the same). When in a terminal state, the agent has no more action and gets no more reward. In non-terminal states, the agent can only go left or right, but their action only succeeds (goes in the intended direction) with probability p. If their action fails, then they go the opposite direction. The numbers on the arrows denote the reward associated with going from one state to another.

For example, at state A taking action `left`:

- with probability p, the next state will be T and the agent will get a reward of 8. The episode is then terminated.
- with probability $1 - p$, the next state will be B and the reward will be 2.

For this problem, the discount factor γ is 1. Let π_p^* be the optimal policy, which may or may not depend on the value of p. Let $Q^{\pi_p^*}$ and $V^{\pi_p^*}$ be the corresponding Q and V functions of π_p^*.

(a) [?? pts] If $p = 1$, what is π_p^*? (Select one)

- ○ $\pi_p^*(A) = \texttt{left}$, $\pi_p^*(B) = \texttt{left}$
- ○ $\pi_p^*(A) = \texttt{left}$, $\pi_p^*(B) = \texttt{right}$
- ● $\pi_p^*(A) = \texttt{right}$, $\pi_p^*(B) = \texttt{left}$
- ○ $\pi_p^*(A) = \texttt{right}$, $\pi_p^*(B) = \texttt{right}$

The optimal policy just goes back and forth between A and B getting infinite points.

(b) [?? pts] If $p = 0$, what is $\pi_p^*(A)$? (Select one)

- ○ $\pi_p^*(A) = \texttt{left}$, $\pi_p^*(B) = \texttt{left}$
- ● $\pi_p^*(A) = \texttt{left}$, $\pi_p^*(B) = \texttt{right}$
- ○ $\pi_p^*(A) = \texttt{right}$, $\pi_p^*(B) = \texttt{left}$
- ○ $\pi_p^*(A) = \texttt{right}$, $\pi_p^*(B) = \texttt{right}$

Since $p = 0$, it's the same as $p = 1$ but you just need to take the opposite action.

(c) [?? pts] Suppose $\pi_p^*(A) = \texttt{left}$. Which of the following statements **must** be true? (Select all that apply)
Hint: Don't forget that if $x = y$, then $x \geqslant y$ and $x \leqslant y$.

- ☐ $Q^{\pi_p^*}(A, \texttt{left}) \leqslant Q^{\pi_p^*}(A, \texttt{right})$
- ■ $Q^{\pi_p^*}(A, \texttt{left}) \geqslant Q^{\pi_p^*}(A, \texttt{right})$
- ☐ $Q^{\pi_p^*}(A, \texttt{left}) = Q^{\pi_p^*}(A, \texttt{right})$
- ☐ $V^{\pi_p^*}(A) \leqslant V^{\pi_p^*}(B)$
- ■ $V^{\pi_p^*}(A) \geqslant V^{\pi_p^*}(B)$
- ☐ $V^{\pi_p^*}(A) = V^{\pi_p^*}(B)$
- ■ $V^{\pi_p^*}(A) \leqslant Q^{\pi_p^*}(A, \texttt{left})$
- ■ $V^{\pi_p^*}(A) \geqslant Q^{\pi_p^*}(A, \texttt{left})$
- ■ $V^{\pi_p^*}(A) = Q^{\pi_p^*}(A, \texttt{left})$
- ☐ $V^{\pi_p^*}(A) \leqslant Q^{\pi_p^*}(A, \texttt{right})$
- ■ $V^{\pi_p^*}(A) \geqslant Q^{\pi_p^*}(A, \texttt{right})$
- ☐ $V^{\pi_p^*}(A) = Q^{\pi_p^*}(A, \texttt{right})$

For left to be the optimal action, it must be the case that $Q^{\pi_p^*}(A, \texttt{left}) \geqslant Q^{\pi_p^*}(A, \texttt{right})$. Therefore, we also get that $V^{\pi_p^*}(A) = \max_a Q^{\pi_p^*}(A, a) = Q^{\pi_p^*}(A, \texttt{left})$. Also, it's always the case that $V^{\pi_p^*}(A) \geqslant V^{\pi_p^*}(B)$ for this problem.

(d) Assume $p \geqslant 0.5$ below.

(i) [?? pts] $V^*(B) = \alpha V^*(A) + \beta$. Find α and β in terms of p.

- $\alpha =$ p
- $\beta =$ 0

since $p \geqslant 0.5$, it's always optimal to go left from B. So

$$V^*(B) = Q^*(B, \texttt{left}) = p(0 + V^*(A)) + (1-p)(0 + V^*(T)) = pV^*(A)$$

(ii) [?? pts] $Q^{\pi_p^*}(A, \texttt{left}) = \alpha V^*(B) + \beta$. Find α and β in terms of p.

- $\alpha =$ 1-p
- $\beta =$ 2+6p

$$\begin{aligned} Q^{\pi_p^*}(A, \texttt{left}) &= p(8 + V^*(T)) + (1-p)(2 + V^*(B)) \\ &= (1-p)V^*(B) + 2 + 6p \end{aligned}$$

(iii) [?? pts] $Q^{\pi_p^*}(A, \texttt{right}) = \alpha V^*(B) + \beta$. Find α and β in terms of p.

- $\alpha =$ p
- $\beta =$ $8 - 6p$

$$\begin{aligned} Q^{\pi_p^*}(A, \texttt{right}) &= (1-p)(8 + V^*(T)) + p(2 + V^*(B)) \\ &= pV^*(B) + 8 - 6p \end{aligned}$$

Q6. [?? pts] Take Actions

An agent is acting in the following gridworld MDP, with the following characteristics.

- Discount factor $\gamma < 1$.
- Agent gets reward $R > 0$ for **entering** the terminal state T, and 0 reward for all other transitions.
- When in terminal state T, the agent has no more action and gets no more reward.
- In non-terminal states $\{A, B, C\}$, the agent can take an action $\{Up, Down, Left, Right\}$.
- Assume perfect transition dynamics. For example, taking action *Right* at state A will always result in state C in the next time step.
- If the agent hits an edge, it stays in the same state in the next time step. For example, after taking action *Right* at C, the agent remains in state C.

B	T
A	C

(a) **(i)** [?? pts] What are all the optimal deterministic policies? Each cell should contain a single action $\{Up, Down, Left, Right\}$. Each row corresponds to a different optimal policy. You may not need all rows.

State	A	B	C
Optimal policy 1	Up	Right	Up
Optimal policy 2 (if needed)	Right	Right	Up
Optimal policy 3 (if needed)			

(ii) [?? pts] Suppose the agent uniformly randomly chooses between the optimal policies in (i). In other words, at each state, the agent picks randomly between the actions in the corresponding column with equal probability. The agent's location at each time step is then a Markov process where state $X_t \in \{A, B, C, T\}$. Fill in the following transition probabilities for the Markov process.

- $P(X_{t+1} = B | X_t = A) =$ 0.5
- $P(X_{t+1} = A | X_t = B) =$ 0
- $P(X_{t+1} = T | X_t = C) =$ 1

(b) Suppose the agent is acting in the same gridworld as above, but does not get to observe their exact state X_t. Instead, the agent only observes $O_t \in \{black, green, pink\}$. The observation probability as a function of the state $P(O_t|X_t)$ is specified in the table below. This becomes a partially-observable Markov decision process (POMDP). The agent is equally likely to start in non-terminal states $\{A, B, C\}$.

B 0.5, black 0.5, green	T
A 0.5, black 0.5, pink	C 0.5, pink 0.5, green

(i) [**??** pts] If the agent can only act based on its current observation, what are all deterministic optimal policies? You may not need all rows.

	Black	Green	Pink
Optimal policy 1	Right	Right	Up
Optimal policy 2 (if needed)	Right	Up	Up
Optimal policy 3 (if needed)			

(ii) [**??** pts] Suppose that the agent follows the policy π(Black) = Right, π(Green) = Right, and π(Pink) = Up. Let $V(S)$ be the agent's expected reward from state S. Your answer should be in terms of γ and R. Note that $V(S)$ is the expected value before we know the observation, so you must consider all possible observations at state S.

- V(A) = $\gamma(\frac{1}{2}R + \frac{1}{2}(\frac{R}{2-\gamma}))$
- V(B) = R
- V(C) = $\frac{R}{2-\gamma}$

$V(B) = R$ because we always go right in state B.
$V(C) = \frac{1}{2}R + \frac{1}{2}\gamma V(C) \implies V(C) = \frac{R}{2-\gamma}$.
$V(A) = \gamma(\frac{1}{2}V(B) + \frac{1}{2}V(C))$

Now suppose that the agent's policy can also depend on all past observations and actions. Assume that when the agent is starting (and has no past observations), it behaves the same as the policy in the previous part: π([Black]) = Right, π([Green]) = Right, π([Pink]) = Up. In all cases where the agent has more than one observation (for example, observed Pink in the previous time step and now observes Green), π acts optimally.

(iii) [**??** pts] For each of the following sequences of two observations, write the optimal action that the policy π would take.

Black Pink	Black Green	Green Pink	Green Green	Pink Black	Pink Green
Up	Up	Up	Up	Right	Right

(iv) [?? pts] In this part only, let $V(S)$ refer to the expected sum of discounted rewards following π if we start from state S (and thus have no previous observations yet). As in the previous part, this is the expected value before knowing the observation, so you must consider all possible observations at S.

Hint: since π now depends on sequences of observations, the way we act at states after the first state may be different, and this affects the value at the first state.

- V(A) = γR
- V(B) = R
- V(C) = $\frac{1}{2}(1+\gamma)R$

(c) Boba POMDP May is a venture capitalist who knows that Berkeley students love boba. She is picking between investing in Sharetea or Asha. If she invests in the better one, she will make a profit of \$1000. If she invests in the worse one, she will make no money.

At the start, she believes both Asha and Sharetea have an equal chance of being better. However, she can pay to have students taste test. At each time step, she can either choose to invest or to pay for a student taste test. Each student has a $p = 0.9$ probability of picking the correct place (independent of other students).

(i) [?? pts] What is the expected profit if May invests optimally after one (free) student test?

900

(ii) [?? pts] If she had to invest after one student test, what is the highest May should pay for the test?

400

(iii) [?? pts] Suppose after n student tests, it turns out that all students have chosen the same store. What is her expected profit after after observing these n student tests?

- ○ $1000(0.9^n)$
- ● $1000(\frac{0.9^n}{0.9^n+0.1^n})$
- ○ $1000(0.9^n - 0.1^n)$
- ○ $1000(\frac{0.9^n-0.1^n}{0.9^n})$
- ○ $1000(\frac{0.9^n-0.1^n}{0.9^n+0.1^n})$
- ○ $1000(1 - 0.1^n)$

(iv) [?? pts] How many tests should May pay for if each one costs \$100?

Hint: Think about the maximum possible value of information. How does this compare to the expected value of information?

1

Q7. [?? pts] Graph Search

You are trying to plan a road trip from city A to city B. You are given an **undirected graph** of roads of the entire country, together with the distance along each road between any city X and any city Y: $length(X, Y)$ (For the rest of this question, "shortest path" is always in terms of *length*, not number of edges). You would like to run a search algorithm to find the shortest way to get from A to B (assume no ties).

Suppose C is the capital, and thus you know the shortest paths from city C to every other city, and you would like to be able to use this information.

Let $path_{opt}(X \to Y)$ denote the shortest path from X to Y and let $cost(X, Y)$ denote the cost of the shortest path between cities X and Y. Let $[path(X \to Y), path(Y \to Z)]$ denote the concatenation.

(a) [?? pts] Suppose the distance along any edge is 1. You decide to initialize the queue with A, plus a list of all cities X, with $path(A \to X) = [path_{opt}(A \to C), path_{opt}(C \to X)]$. You run **BFS** with this initial queue (sorted in order of path length). Which of the following is correct? (Select all that apply)

- ☐ You always expand the exact same nodes as you would have if you ran standard BFS.
- ☐ You might expand a different set of nodes, but still find the shortest path.
- ■ You might expand a different set of nodes, and find the sub-optimal path.

Consider a graph of 5 nodes: A,B,C,D,E and edges (A,C), (C,E), (E,B), (A,D),(D,B). Then our initial queue (in order) is

1. C: A-C
2. E: A-C-E
3. B: A-C-E-B
4. D: A-C-A-D

The path returned will be A-C-E-B

(b) [?? pts] You decide to initialize priority queue with A, plus a list of all cities X, with $path(A \to X) = [path_{opt}(A \to C), path_{opt}(C \to X)]$, and $cost(A, X) = cost(A, C) + cost(C, X)$. You run **UCS** with this initial priority queue. Which of the following is correct? (Select all that apply)

- ■ You always expand the exact same nodes as you would have if you ran standard UCS.
- ☐ You might expand a different set of nodes, but still find the shortest path.
- ☐ You might expand a different set of nodes, and find the sub-optimal path.

Regardless of what is on the queue, UCS will explore nodes in order of their shortest-path distance to A, so the set of explored nodes is always {nodes X: dist(A,X) less than dist(A,B)}

Q8. [?? pts] Bayes Net Inference

A plate representation is useful for capturing replication in Bayes Nets. For example, Figure **??**(a) is an equivalent representation of Figure **??**(b). The N in the lower right corner of the plate stands for the number of replica.

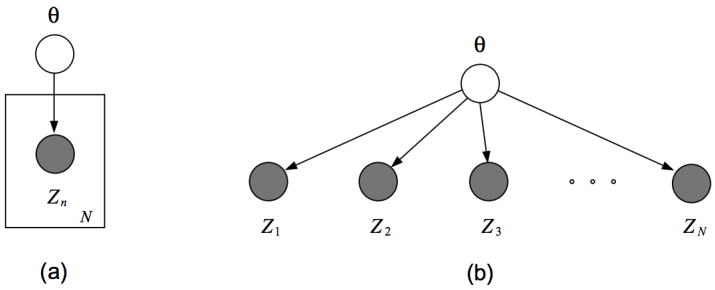


Figure 2

Now consider the Bayes Net in Figure **??**. We use $X_{1:N}$ as shorthand for $(X_1, \cdots, X_N)$. We would like to compute the query $P(X_{1:N}|Y_{1:N} = y_{1:N})$. Assume all variables are binary.

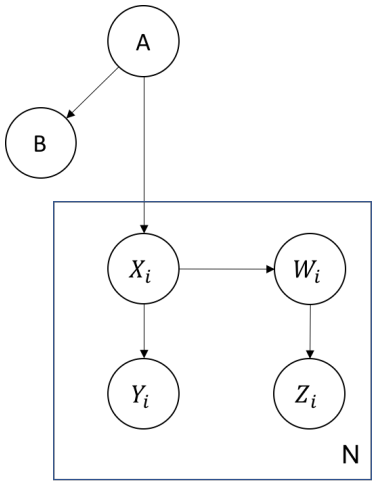


Figure 3

(a) [**??** pts] What is the number of rows in the largest factor generated by *inference by enumeration*, for this query?

○ 2^{2N} ○ 2^{3N} ○ 2^{2N+2} ● 2^{3N+2}

In inference by enumeration, the full joint probability $P(X_{1:N}, Y_{1:N} = y_{1:N}, W_{1:N}, Z_{1:N}, A, B)$ are computed, which has size 2^{3N+2}

(b) [**??** pts] Mark all of the following variable elimination orderings that are optimal for calculating the answer for the query $P(X_{1:N}|Y_{1:N} = y_{1:N})$. (A variable elimination ordering is optimal if the **largest** factors generated is smallest among all possible elimination orderings).

■ $Z_1, \cdots, Z_N, W_1, \cdots, W_N, B, A$
■ $W_1, \cdots, W_N, Z_1, \cdots, Z_N, B, A$
☐ $A, B, W_1, \cdots, W_N, Z_1, \cdots, Z_N$
☐ $A, B, Z_1, \cdots, Z_N, W_1, \cdots, W_N$

The only thing that matters is the size of the maximum factor generated during elimination and the final factor $P(X_{1:N}|y_{1:N})$ has size 2^N. Eliminating anything other than A does not generate a factor which depends on more than one time index i, so as long as A is eliminated last, no factor of size greater than 2^N is generated,

so the first two orderings are both optimal. (In fact, as long as A is eliminated before B, the ordering will be optimal, but it was not necessary to notice this to distinguish among the given options.)

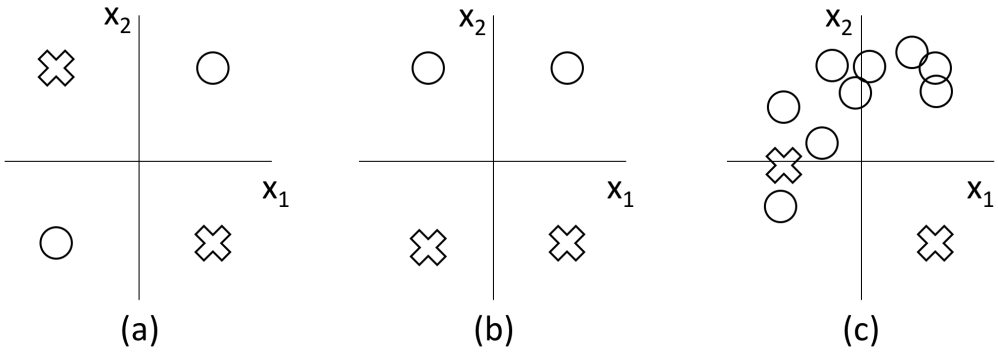
(c) [**??** pts] Which of the following variables can be deleted before running variable elimination, without affecting the inference result? Deleting a variable means not putting its CPT in our initial set of factors when starting the algorithm.

■ W_1 ■ Z_1 □ A ■ B □ *None*

B, W_1, and Z_1 can all be deleted. In general, any variable which has no descendants that are query variables (in this case X_i) or evidence variables (in this case y_i) can be deleted. This is because when we eliminate the subgraph of all the descendants of such a variable, we will end up with a factor in which all the entries are equal to 1 and thus does not affect the results whatsoever when joined with other factors.

Q9. [?? pts] Deep Learning

(a) [?? pts] **Data Separability**



The plots above show points in feature space (x_1, x_2), also referred to as feature vectors $\mathbf{x} = [x_1 \;\; x_2]^T$.

For each of the following, we will define a function $h(\mathbf{x})$ as a composition of some functions f_i and g_i. For each one, consider the decision rule

$$y(\mathbf{x}) = \begin{cases} \times & h(\mathbf{x}) \geqslant 0 \\ \bigcirc & h(\mathbf{x}) < 0. \end{cases}$$

Under each composition of functions h, select the datasets for which there exist some **linear** functions f_i and some **nonlinear** functions g_i such that the corresponding decision rule perfectly classifies the data. (Select all that apply)

A composition of linear functions will always be linear. Parts (i), (ii), (iv) are linear. Plot (b) is linearly separable, and can be separated by linear or nonlinear decision boundaries. Plots (a),(c) require a nonlinear function to perfectly separate them.

(i) $h(\mathbf{x}) = f_1(\mathbf{x})$

(a) ☐ (b) ■ (c) ☐

(ii) $h(\mathbf{x}) = f_2(f_1(\mathbf{x}))$

(a) ☐ (b) ■ (c) ☐

(iii) $h(\mathbf{x}) = f_2(g_1(f_1(\mathbf{x})))$

(a) ■ (b) ■ (c) ■

(iv) $h(\mathbf{x}) = f_4(f_3(f_2(f_1(\mathbf{x}))))$

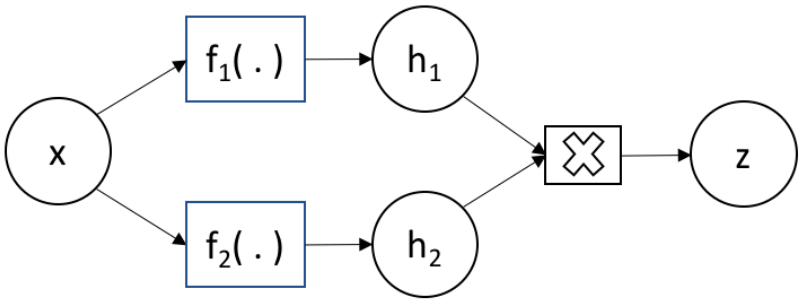
(a) ☐ (b) ■ (c) ☐

(v) $h(\mathbf{x}) = g_2(g_1(\mathbf{x}))$

(a) ■ (b) ■ (c) ■

(b) Backpropagation Below is a deep network with input x. Values x, h_1, h_2, z are all scalars.

$$h_1 = f_1(x), h_2 = f_2(x), z = h_1 h_2 \tag{1}$$



Derive the following gradients in terms of $x, h_1, h_2, \frac{\partial f_1}{\partial x}, \frac{\partial f_2}{\partial x}$.

(i) [?? pts] Derive $\frac{\partial z}{\partial h_1}$

h_2

When taking the partial derivative of z in terms of h_1, we treat h_2 as a constant.

(ii) [?? pts] Derive $\frac{\partial z}{\partial h_2}$

h_1

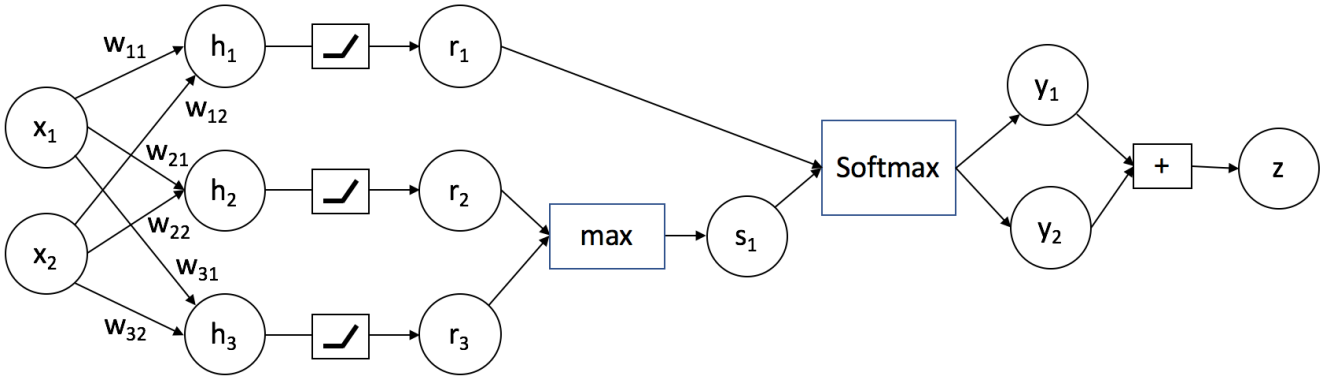
When taking the partial derivative of z in terms of h_2, we treat h_1 as a constant.

(iii) [?? pts] Derive $\frac{\partial z}{\partial x}$

$h_2 \frac{\partial f_1}{\partial x} + h_1 \frac{\partial f_2}{\partial x}$

We use product rule and chain rule.

(c) Deep Network Below is a deep network with inputs x_1, x_2. The internal nodes are computed below. All variables are scalar values.



$$
\begin{aligned}
h_1 &= w_{11}x_1 + w_{12}x_2 \qquad h_2 = w_{21}x_1 + w_{22}x_2 \qquad h_3 = w_{31}x_1 + w_{32}x_2 \\
r_1 &= \max(h_1, 0) \qquad r_2 = \max(h_2, 0) \qquad r_3 = \max(h_3, 0) \\
s_1 &= \max(r_2, r_3) \\
y_1 &= \frac{\exp(r_1)}{\exp(r_1) + \exp(s_1)} \qquad y_2 = \frac{\exp(s_1)}{\exp(r_1) + \exp(s_1)} \\
z &= y_1 + y_2
\end{aligned} \tag{2}
$$

(i) [?? pts] **Forward propagation** Now, given $x_1 = 1, x_2 = -2, w_{11} = 6, w_{12} = 2, w_{21} = 4, w_{22} = 7, w_{31} = 5, w_{32} = 1$, and the same values for x_1, x_2 above, compute the values of the internal nodes. Please simplify any fractions.

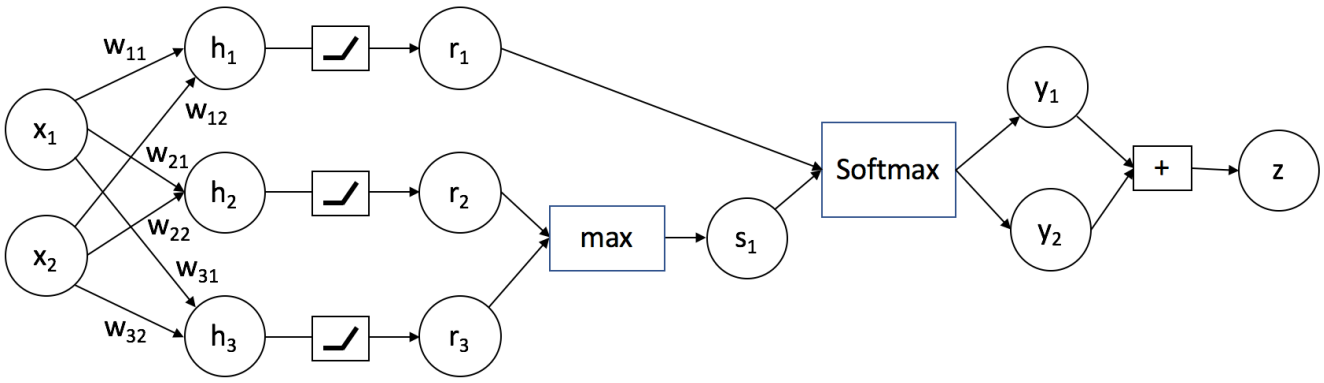
h_1	h_2	h_3	r_1	r_2
2	-10	3	2	0

r_3	s	y_1	y_2	z
3	3	$\frac{1}{1+e}$	$\frac{e}{1+e}$	1

(ii) [?? pts] **Bounds on variables.**
Find the tightest bounds on y_1. $y_1 \in (0, 1)$

The output of a softmax is a probability distribution. Each element of the output is between 0 and 1.

Find the tightest bounds on z. $z = 1$
The sum of the probability distribution is 1.



$$
\begin{aligned}
h_1 &= w_{11}x_1 + w_{12}x_2 \qquad h_2 = w_{21}x_1 + w_{22}x_2 \qquad h_3 = w_{31}x_1 + w_{32}x_2 \\
r_1 &= \max(h_1, 0) \qquad r_2 = \max(h_2, 0) \qquad r_3 = \max(h_3, 0) \\
& \qquad\qquad s_1 = \max(r_2, r_3) \\
y_1 &= \frac{\exp(r_1)}{\exp(r_1) + \exp(s_1)} \qquad y_2 = \frac{\exp(s_1)}{\exp(r_1) + \exp(s_1)} \\
& \qquad\qquad z = y_1 + y_2
\end{aligned} \tag{3}
$$

(iii) [**??** pts] **Backpropagation** Compute the following gradients analytically. The answer should be an expression of any of the nodes in the network $(x_1, x_2, h_1, h_2, h_3, r_1, r_2, r_3, s_1, y_1, y_2, z)$ or weights $w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}$. Hint: Recall that for functions of the form $g(x) = \frac{1}{1+exp(a-x)}$, $\frac{\partial g}{\partial x} = g(x)\,(1 - g(x))$. Also, your answer may be a constant or a piecewise function.

$\frac{\partial h_1}{\partial w_{12}}$	$\frac{\partial h_1}{\partial x_1}$	$\frac{\partial r_1}{\partial h_1}$	$\frac{\partial y_1}{\partial r_1}$
x_2	w_{11}	$1[h_1 > 0]$	$y_1(1 - y_1)$

$\frac{\partial y_1}{\partial s_1}$	$\frac{\partial z}{\partial y_1}$	$\frac{\partial z}{\partial x_1}$	$\frac{\partial s_1}{\partial r_2}$
$-y_1 y_2$	1	0	$1[r_2 > r_3]$

Expanded solutions for selected examples below:

$r_1 = \max(h_1, 0)$. This is known as a ReLU (rectified linear unit) function. When h_1 is positive, $r_1 = h_1$, so the derivative is 1. When h_1 is negative, r_1 is flat, so the derivative is 0.

$y_1 = \frac{\exp(r_1)}{\exp(r_1)+\exp(r_2)} = \frac{1}{1+\exp(r_2-r_1)}$
$\frac{dy_1}{dr_1} = \frac{-1}{(1+\exp(r_2-r_1))^2} \times (-\exp(r_2 - r_1))$, by chain rule
$= \frac{1}{1+\exp(r_2-r_1)} \times \frac{\exp(r_2-r_1)}{1+\exp(r_2-r_1)}$
$= y_1(1 - y_1)$
$= y_1 y_2$

$\frac{dy_1}{dr_2} = \frac{-1}{(1+\exp(r_2-r_1))^2} \times (\exp(r_2 - r_1))$, by chain rule. Notice that this is identical to the case above, but with a negative sign missing on the 2nd term.

$= \frac{-1}{1+\exp(r_2-r_1)} \times \frac{\exp(r_2-r_1)}{1+\exp(r_2-r_1)}$
$= -y_1(1-y_1)$
$= -y_1 y_2$

No matter how x_1, x_2 change, z is always 1, so the gradient with respect to x_1 is 0.

When $r_2 > r_3$, $s_1 = r_2$, so $\frac{\partial s_1}{r_2} = 1$. When $r_2 < r_3$, $s_1 = r_3$, so $\frac{\partial s_1}{r_2} = 0$.

THIS PAGE IS INTENTIONALLY LEFT BLANK