CS221 Exam

CS221
November 27, 2018

Name: ____________________
by writing my name I agree to abide by the honor code

SUNet ID: ____________________

Read all of the following information before starting the exam:

- This test has 3 problems and is worth 150 points total. It is your responsibility to make sure that you have all of the pages.
- Keep your answers precise and concise. Show all work, clearly and in order, or else points will be deducted, even if your final answer is correct.
- Don't spend too much time on one problem. Read through all the problems carefully and do the easy ones first. Try to understand the problems intuitively; it really helps to draw a picture.
- You cannot use any external aids except one double-sided $8\frac{1}{2}$" x 11" page of notes.
- Good luck!

Problem	Part	Max Score	Score
1	a	10	
	b	10	
	c	10	
	d	10	
	e	10	
2	a	10	
	b	10	
	c	10	
	d	10	
	e	10	
3	a	10	
	b	10	
	c	10	
	d	10	
	e	10	

Total Score: ☐ + ☐ + ☐ = ☐

1. **Wildlife** (*50 points*)

You are working in a wildlife conservation group, where you are installing n sensors to detect the presence of a wild animal called a pangolin. Let $Y \in \{0,1\}$ denote whether there is actually a pangolin (in a given location), and let $X_1, \ldots, X_n$ denote the predicted outputs of the n sensors, where each $X_i \in \{0,1\}$. We assume the following:

- There is a natural rate of pangolin appearance $p(y=1) = h$.
- All sensors have the same false positive rate of $p(x_i = 1 \mid y = 0) = \alpha$.
- All sensors have the same false negative rate of $p(x_i = 0 \mid y = 1) = \beta$.
- All sensor outputs are conditionally independent given the actual appearance (see Figure 1).

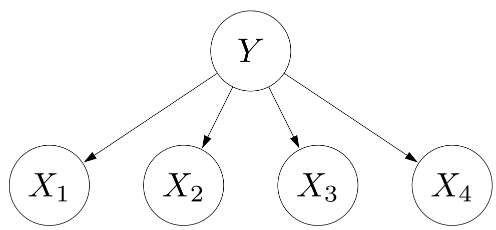

Figure 1: A Bayesian network with $n = 4$ sensors relating the presence of a pangolin Y and sensor outputs $X_1, \ldots, X_4$.

a. (*10 points*)

The specification sheet for the sensors does not provide the false positive rates or the false negative rates, so you have to estimate them. Each week, you install a new sensor, record all the outputs of all the sensors installed thus far. Then you go out into the field and observe whether there is a pangolin or not (Y). Below is the data you have collected, where "-" indicates there is no data for that sensor for that week.

	X_1	X_2	X_3	X_4	Y
Week 1	0	-	-	-	0
Week 2	0	1	-	-	0
Week 3	1	1	0	-	1
Week 4	0	0	1	0	0

Compute the maximum likelihood estimate of the parameters (h, α, β) of the Bayesian network:

$h =$

$\alpha =$

$\beta =$

b. (*10 points*)

After experiencing the dangers of data collection in the wild for four weeks, you decide that it's not really your true calling. Instead, you're just going to use your estimated parameters (h, α, β) to predict Y for future weeks.

Suppose that you have n sensors installed. During one week, you observe that all the sensors output 1. What is $\mathbb{P}(Y = 1 \mid X_1 = \cdots = X_n = 1)$, i.e. the probability that there is a pangolin given the observations?

Your answer should be an expression defined in terms of (h, α, β, n).

c. (*10 points*)

Your boss asks you to install more sensors, but you first want to understand what happens as the number of sensors goes to infinity. Note: your solution will depend strongly on your formula for part (b).

(i) [5 points] Suppose $\alpha + \beta < 1$. What does $\mathbb{P}(Y = 1 \mid X_1 = \cdots = X_n = 1)$ converge to as $n \to \infty$? You should make explicit how your answer depends on h (your answer should cover every possible value of $0 \leq h \leq 1$). You should also give a rigorous mathematical justification for your answer.

(ii) [5 points] Suppose $\alpha + \beta = 1$. What does $\mathbb{P}(Y = 1 \mid X_1 = \cdots = X_n = 1)$ converge to as $n \to \infty$? You should make explicit how your answer depends on h (your answer should cover every possible value of $0 \leq h \leq 1$). You should also give a rigorous mathematical justification for your answer.

d. (*10 points*)

Let us try to solve the classification problem (predicting Y from $X_1, \ldots, X_n$) in a different way now. We could use the standard hinge loss or logistic loss, but we want to design our loss function to capture the fact that false negatives are much more unacceptable than false positives (we don't want to miss any pangolins!).

Let us suppose we define a linear classifier as follows:[1]

$$f(x) = [\mathbf{w} \cdot \phi(x) \geq 0]. \tag{1}$$

Define a loss function as follows:

$$\text{Loss}(x, y, \mathbf{w}) = \begin{cases} \ell_1(x, y, \mathbf{w}) & \text{if } y = 1 \\ \ell_0(x, y, \mathbf{w}) & \text{if } y = 0. \end{cases} \tag{2}$$

Design ℓ_0 and ℓ_1 based on the hinge loss so that the following properties hold:

- If the margin of a point (x, y) (defined as $\mathbf{w} \cdot \phi(x)$ for $y = 1$ and $-\mathbf{w} \cdot \phi(x)$ for $y = 0$) is at least 1, then the loss is zero on that point.
- If two points $(x, 1)$ and $(x, 0)$ have the same margin, then the loss of $(x, 1)$ is 5 times that of $(x, 0)$.

(i) [5 points] Specify the loss function:

$\ell_0(x, y, \mathbf{w}) =$

$\ell_1(x, y, \mathbf{w}) =$

[1] Note that the output label is $y \in \{0, 1\}$ rather than $y \in \{1, -1\}$ as we've seen in class.

(ii) [5 points] As usual, we need to compute the gradient to optimize this objective. Compute the gradient of the loss function with respect to $\mathbf{w}$. Express your answer in terms of $\mathbf{w}, \phi(x), y$.

$\nabla \text{Loss}(x, y, \mathbf{w}) =$

e. (*10 points*)

Suppose that the feature vector is $\phi(x) = [1, \sum_{i=1}^{n} x_i]$, where $x_i \in \{0, 1\}$ is still the output of sensor i. Consider the 4 examples from part (a), where an unobserved value is treated as a zero:

	X_1	X_2	X_3	X_4	Y
Week 1	0	-	-	-	0
Week 2	0	1	-	-	0
Week 3	1	1	0	-	1
Week 4	0	0	1	0	0

For example, for week 3, $\phi(x) = [1, 2]$.

For this two-dimensional feature vector, the weight vector is $\mathbf{w} = (w_1, w_2)$. If we set $w_1 = -5$, what are all the possible values of w_2 for which the resulting weight vector $\mathbf{w}$ achieves zero training loss according to the loss function you defined in part (d)?

2. Driving (*50 points*)

Optimal Driving Corporation (ODC) has hired you as a consultant to help design optimal algorithms for their new fleet of self-driving cars. They are currently focused on a particular stretch of freeway with H lanes and length W (see Figure 2). There are $K+1$ cars on the road: Your car (the one being controlled algorithmically) starts at position $x_0^{(0)} = (1,1)$, and the K other cars start at positions $x_1^{(0)}, \dots, x_K^{(0)}$.

Each car (yours and others) has the same set of four possible actions (the successor states below are for the car in $(5,2)$ from Figure 2):

- $+(1,0)$: Move forward one step (ending up in $(6,2)$)
- $+(2,0)$: Move forward two steps (ending up in $(7,2)$)
- $+(1,-1)$: Move forward one step and to the left (ending up in $(6,1)$)
- $+(1,1)$: Move forward one step and to the right (ending up in $(6,3)$)

A car cannot take an action that takes it out of one of the lanes; for example, a car in $(1,3)$ cannot take action $+(1,1)$.

If a car leaves this stretch of freeway (e.g., taking action $+(1,0)$ from $(7,2)$), then the car is teleported to a special end position $\odot$.

All cars take turns moving, starting with your car, followed by cars 1 through K, and then your car, etc. The game ends when all cars have left the stretch of freeway (moved in position $\odot$).

Each action has a base cost of 1. There are also a set of potholes P. If at the end of an action, a car lands on a pothole or in a position occupied by another car, it incurs an additional cost of c (and does not affect any other cars). Each car's goal is to minimize its own total cost. For example, if your car takes actions that lead to the following state sequence: $(1,1) \to (2,2) \to (4,2) \to (6,2) \to (7,3)$, the cost incurred by you is $1+1+(1+c)+1 = 4+c$. Note that cost is only incurred based on where a car lands, so the cost of $(2,2) \to (4,2)$ above is 1, even though there is a pothole at $(3,2)$.

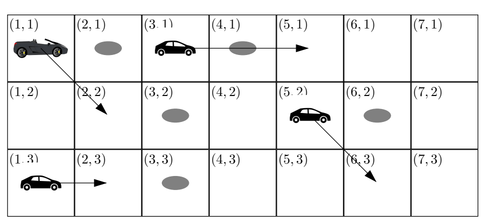

Figure 2: An example of the driving scenario, where we have a freeway of length $W = 7$ and $H = 3$ lanes. Your car starts at $x_0^{(0)} = (1,1)$ and the other $K = 3$ cars start in $x_1^{(0)} = (1,3)$, $x_2^{(0)} = (3,1)$, and $x_3^{(0)} = (5,2)$. The potholes $P = \{(2,1),(3,2),(3,3),(4,1),(6,2)\}$ are shown as gray ellipses. The arrows show one possible action that each of the cars can take.

a. (*10 points*)

Let us formulate this as a $(K+1)$-player game. Define the state to be $(x_0, \dots, x_K, j)$, where x_0 is the position of your car, $x_1, \dots, x_K$ are the positions of other cars, and $j \in \{0, \dots, K\}$ specifies which car's turn it is.

Rather than defining only a utility for end states for each car as in class, we define a cost for all state-action pairs: in particular, let $\mathrm{Cost}_j(s, a)$ be the cost incurred by player j when action a is taken from state s. Complete the game specification below.

- $s_{\text{start}} = (x_0^{(0)}, \dots, x_K^{(0)}, 0)$
- $\mathrm{Player}((x_0, \dots, x_K, j)) = j$
- $\mathrm{Actions}((x_0, \dots, x_K, j)) = \{a \in \{+(1,0), +(2,0), +(1,-1), +(1,1)\} : \mathrm{InLane}(x_j + a)\}$, where $\mathrm{InLane}((u,v)) = [(u,v) = \odot \text{ or } 1 \le v \le H]$. By convention, we define $\odot + a = \odot$, so that cars that have left the stretch of freeway stay in the same end state.
- $\mathrm{Succ}((x_0, \dots, x_K, j), a) =$
- $\mathrm{Cost}_j((x_0, \dots, x_K, j), a) =$
- $\mathrm{IsEnd}(x_0, \dots, x_K, j) =$

b. (*10 points*)

Assume the other K cars follow the deterministic policy of always moving forward one step (choosing $a = (1, 0)$).

(i) [5 points] Compute the maximum possible number of unique states $(x_0, \ldots, x_K, j)$ which are reachable from a given s_{start}, where your car can take any action and the other K cars take only the action given by the fixed policy. Your answer should be as tight an (asymptotic) upper bound as possible. You must provide the expression that has the correct (smallest) dependence on W, H, K, but don't worry about constants (if the true answer is $W - 1$, then $O(W)$, $2W$, W are all acceptable, but W^2 is not).

(ii) [5 points] What if your car was no longer allowed to move forward two steps (i.e., $(2, 0) \notin \text{Actions}((x_0, \ldots, x_K, 0))$)? Compute the (asymptotic) maximum possible number of unique states that are reachable in this case.

c. (*10 points*)

It is Sunday morning and there are no other cars on the road ($K = 0$). In this case, the problem is simply a search problem over the state x_0. You realize that you might be able to compute the optimal policy faster by leveraging A*. But of course to do that, we need a consistent heuristic.

(i) [5 points] Design a non-trivial consistent heuristic that can be computed in closed form, and prove why your heuristic is consistent. Here, your car position $x_0 = (u, v)$.

$h((u, v)) =$

(ii) [5 points] Prove that the following heuristic is inconsistent:

$$h((u, v)) = \sum_{u'=u+1}^{W} c \cdot \mathbb{1}[(u', v) \in P], \quad (3)$$

which simply counts the number of potholes between $(u + 1, v)$ and (W, v) and multiplies by the cost c of landing on a pothole (recall that $\mathbb{1}$ is the indicator function). Hint: try to construct a simple example.

d. (*10 points*)

It turns out that your assumption that all other cars just move forward by one step ($a = +(1, 0)$) is wrong (shockingly). You still assume that each car has the same policy, but this policy chooses an action only depending on what's at the four positions a car could move into. Suppose we have a car at position x_j and it's j's turn. Define a feature vector $\phi((x_0, \ldots, x_K, j))$, which has two features for each action a:

1. whether there is a pothole at $x_j + a$, and
2. whether there is another car at $x_j + a$.

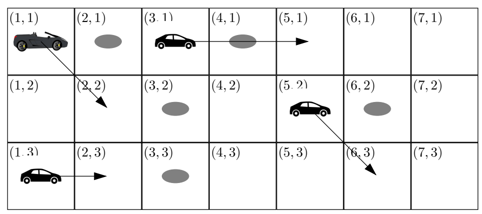

Figure 3: Repeat of Figure 2 for convenience. The training data here is the action taken (shown by the arrow) for each of the three other cars.

For example, for the car at $x_j = (3, 1)$ in Figure 3, which takes action $+(2, 0)$, the feature vector would be the following:

name	value
$+(1, 0)$ has pothole:	1
$+(1, 0)$ has car:	0
$+(2, 0)$ has pothole:	0
$+(2, 0)$ has car:	0
$+(1, -1)$ has pothole:	0
$+(1, -1)$ has car:	0
$+(1, 1)$ has pothole:	0
$+(1, 1)$ has car:	0

As shorthand, write this feature vector as 10000000.

Note that for the purposes of computing features, positions off the road are treated as empty positions.

(i) [5 points] We will first use a "tabular" method for estimating the policy. The idea is that for each value of the feature vector, we will estimate a distribution over possible actions based on the data, which is the actions of the three other cars in Figure 3, given by the arrows. In other words, our training data are 3 (feature vector, action) pairs. Specify only the non-zero probabilities and use the shorthand notation for the features. We have filled in the feature vector and action for one of the cars for you.

Hint: think about how we perform model-based estimation of transition probabilities for MDPs, given observations of (state, action, successor) triples.

feature vector ϕ	action a	estimated probability $\pi(a \mid (x_0, \dots, x_K, j))$
10000000	(+2, 0)	

(ii) [5 points] Next you consider using a linear predictor with the same features ϕ. In one or two sentences, state one advantage and one disadvantage of doing so over using the tabular approach above.

e. (*10 points*)

In practice, we don't know what policies the other cars could be facing. For each setting, specify what is the most specific type of model that could be used to represent it (e.g., search problem, MDP, two-player zero-sum game, CSP, etc.). For example, if something is a search problem, don't say MDP, even though all search problems are special cases of MDPs. What inference algorithm would you use for the model (e.g., uniform cost search, minimax, etc.)? If we have not covered inference algorithms for a particular model, say so. Simply state the model and inference algorithm, and briefly justify your answer.

1. (2 points) Each other car's policy chooses the action a that minimizes its immediate cost (not your cost) plus the distance from $x_i + a$ to (W, H). Any ties are broken randomly.

2. (2 points) Each other car's policy uses the learned policy produced from part (d).

3. (2 points) Each other car's policy is optimally minimizing your cost (which might be the case if you had a siren on your car).

4. (2 points) Each other car's policy is optimally maximizing your cost.

5. (2 points) Each other car's policy is trying to minimize its own cost.

3. **Farm** (*50 points*)

Farmer Kim wants to install a set of sprinklers to water all his crops in the most cost-effective manner and has hired you as a consultant. Specifically, he has a rectangular plot of land, which is broken into $W \times H$ cells. For each cell (i, j), let $C_{i,j} \in \{0, 1\}$ denote whether there are crops in that cell that need watering. In each cell (i, j), he can either install ($X_{i,j} = 1$) or not install ($X_{i,j} = 0$) a sprinkler. Each sprinkler has a range of R, which means that any cell within R Manhattan distance gets watered. The maintenance cost of the sprinklers is the sum of the Manhattan distances from each sprinkler to his home located at $(1, 1)$. Recall that the Manhattan distance between (a_1, b_1) and (a_2, b_2) is $|a_1 - a_2| + |b_1 - b_2|$. Naturally, Farmer Kim wants the maintenance cost to be as small as possible given that all crops are watered. See Figure 4 for an example.

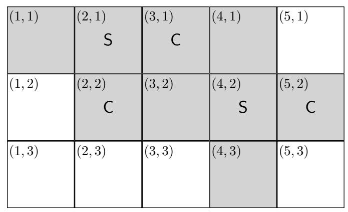

Figure 4: An example of a farm with $W = 5$ and $H = 3$. Each cell (i, j) is marked 'C' if there are crops there that need watering ($C_{i,j} = 1$). An example of a sprinkler installation is given: a cell (i, j) is marked with 'S' if we are placing a sprinkler there ($X_{i,j} = 1$). Here, the sprinkler range is $R = 1$, and the cells that are shaded are the ones covered by some sprinkler. In this case, the sprinkler installation is valid (all crops are watered), and the total maintenance cost is $1 + 4 = 5$.

a. (*10 points*)

Farmer Kim actually took CS221 years ago, and remembered a few things. He says: "I think this should be formulated as a factor graph. The variables should be $X_{i,j} \in \{0,1\}$ for each cell (i,j). But here's where my memory gets foggy. What should the factors be?"

Let $X = \{X_{i,j}\}$ denote an assignment to each variable $X_{i,j}$. Your job is to define two types of factors:

- $f_{i,j}$: ensures any crops in (i,j) are watered,
- f_{cost}: encodes the maintenance cost,

so that a maximum weight assignment corresponds to a valid sprinkler installation with minimum maintenance cost.

$f_{i,j}(X) =$

$f_{\text{cost}}(X) =$

b. (*10 points*)

Every year, Farmer Kim plants a different set of crops in possibly different cells; in other words $\{C_{i,j}\}$ changes year by year. Farmer Kim says: "Maybe I can just start with my assignment $X = \{X_{i,j}\}$ from last year and run Gibbs sampling to try to improve it to a better solution."

(i) [5 points] Recall that Gibbs sampling sets $X_{i,j} = 1$ with some probability p. For convenience, use the notation $X \cup \{X_{i,j} : 1\}$ to denote a modification of X where $X_{i,j}$ has been assigned 1 (analogously for 0). Write an expression for p in terms of the factors (e.g., f_{cost}). Your expression should involve as *few* factors as possible.

$p =$

(ii) [5 points] Recall Gibbs sampling is guaranteed to find the optimal assignment eventually if there is a non-zero probability of reaching any valid assignment X' from the initial assignment X. Prove that this is the case for any X, X'.

c. (*10 points*)

Having installed the sprinklers ($X_{i,j}$), Farmer Kim wants to install water sources on K cells, $\mu_1, \dots, \mu_K$, to power these sprinklers. Each sprinkler ((i, j) for which $X_{i,j} = 1$) is to be assigned to a particular water source $z_{i,j} \in \{1, \dots, K\}$. The *transportation cost* of an installation is the sum of the Manhattan distances from each sprinkler ((i, j) for which $X_{i,j} = 1$) to its assigned water source.

For example, in Figure 4, if $K = 1$, we might install one water source $\mu_1 = (3, 2)$, which would obtain a transportation cost of $2 + 1 = 3$.

Similar to K-means, derive an alternating minimization algorithm that alternates between minimizing the transportation cost with respect to water source assignments (step 1) and minimizing with respect to the location of the water sources (step 2).

(i) [5 points] Step 1: given $\mu_1, \dots, \mu_K$, write an expression for $z_{i,j}$ that minimizes the transportation cost. Notation: let $\mu_k[0]$ and $\mu_k[1]$ be the two coordinates of μ_k.

(ii) [5 points] Step 2: given the sprinkler to water source assignments $\{z_{i,j}\}$, write an expression for μ_k that minimizes the transportation cost. What is the time complexity for calculating ALL μ_k? Give your answer in big-O notation as a function of W, H, K.

d. (*10 points*)

A year has passed after Farmer Kim installed the sprinklers, and now some of the sprinklers have broken. Let $B_{i,j} \in \{0, 1\}$ denote whether a sprinkler at cell (i, j) is broken. Farmer Kim has called you back and is asking for help. He wants to run an irrigation pipe starting from $(1, 1)$ and running through a subset of the broken sprinklers and back to $(1, 1)$.

Every cell that has watered crops generates r dollars of revenue, but connecting a pipe from one cell to an adjacent cell costs c dollars. Farmer Kim naturally wants to maximize profit (revenue minus cost).

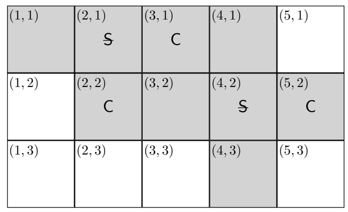

Figure 5: Repeat of Figure 4 for convenience, except now with both sprinklers broken.

For example, in Figure 5, suppose both sprinklers are broken ($B_{2,1} = B_{4,2} = 1$). Here are two options:

1. Fix both sprinklers: build the pipe $(1, 1) \to (2, 1) \to (3, 1) \to (4, 1) \to (4, 2) \to (4, 1) \to (3, 1) \to (2, 1) \to (1, 1)$. This would cost $8c$ and produce all the crops for a revenue of $3r$, so the profit would be $3r - 8c$.

2. Fix the $(2, 1)$ sprinkler: build the pipe $(1, 1) \to (2, 1) \to (1, 1)$. This would cost $2c$ and produce crops at $(3, 1)$ and $(2, 2)$ for a revenue of $2r$, so the profit would be $2r - 2c$.

Your job is to define a search problem where the minimum cost path corresponds to the maximum profit. Farmer Kim also declares that you must fix at least one sprinkler. Farmer Kim has already told you that the state is $((i,j),S)$, where (i,j) is the current position and S is the set of sprinklers (cells) that are working so far. Complete the following search problem specification. *You should give an intuitive description as well as being mathematically precise using the given notation.*

- $s_{\text{start}} = ((1,1), S_0)$, where $S_0 = \{(i,j) : X_{i,j} = 1 \text{ and } B_{i,j} = 0\}$.
- $\text{Actions}(((i,j),S)) = \{a \in \{(-1,0),(+1,0),(0,-1),(0,+1)\} : (i,j)+a \text{ is in bounds}\}$
- $\text{IsEnd}(((i,j),S)) = [i = 1 \text{ and } j = 1 \text{ and } S \neq S_0]$, where we finish when we reach $(1,1)$ having fixed at least one sprinkler.
- $\text{Succ}(((i,j),S),a) =$
- $\text{Cost}(((i,j),S),a) =$

e. (*10 points*)

Suppose that the revenue per crop r is much larger than the cost c; specifically, assume $r > 2(W + H - 2)c$. Assume that sprinklers have been placed $X = \{X_{i,j}\}$ to minimize the maintenance cost as in part (a). In terms of the solution to part (d), will it always be optimal to fix all the broken sprinklers? Either prove the claim or show a counterexample.

Congratulations, you have reached the end of the exam!