KOÇ UNIVERSITY
College of Engineering

COMP 341: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

Final

Fall 2021, 18/01/2022
Duration: 165 Minutes

Name: Solutions

ID: 00110001 00110000 00110000

- This exam contains 15 pages including this cover page and 9 questions. Check to see if any pages are missing. Put your initials on the top of every page, in case the pages become separated.
- By submitting this exam, you **agree** to fully comply with Koç University Student Code of Conduct, and accept any punishment in case of failure to comply. If you do not submit this exam on time, you will receive 0 credits.
- The exam is **open book** and **open notes**. See the honor pledge for details.
- You are expected to be able provide clear and concise answers. Gibberish will not receive any credit. Your answers need to be readable by a human, illegible writing is not gradable.
- Read each question carefully and make sure to follow instructions. The written instructions take precedence over an answer that the instructor might give you during the exam, unless the instructor makes a class wide announcement.
- Do not write in the table below.

Question:	1	2	3	4	5	6	7	8	9	Total
Points:	12	6	12	12	10	10	14	9	15	100
Score:										

1. (12 points) True or False :

 True Performing hyper-parameter selection outside of the training set reduces the risk of overfitting.

 True Regularization helps with overfitting.

 False It is better to split the data into test and training sets after the pre-processing and feature extraction steps.

 True Markov Decision Processes are used to model problems involving uncertain action outcomes.

 False Value iteration is a model-free method.

 False In Q-learning, the agent must follow its own policy.

 False The value of a state is independent of its neighbors.

 True Given good features, approximate Q-learning helps with generalization.

2. (6 points) Direct Policy Evaluation: You observed the following episodes from an **undiscounted** MDP with two states A and B as below (the numbers denote the reward you receive):

$$(A, +2) \rightarrow (A, +1) \rightarrow (B, -2) \rightarrow (A, +2) \rightarrow (B, -1) \rightarrow terminate$$
$$(B, -2) \rightarrow (A, +2) \rightarrow (B, -1) \rightarrow terminate$$

Estimate the value function using **direct evaluation** (do not use Bellman Equations) and fill in the table below. Make sure to show your work below the table.

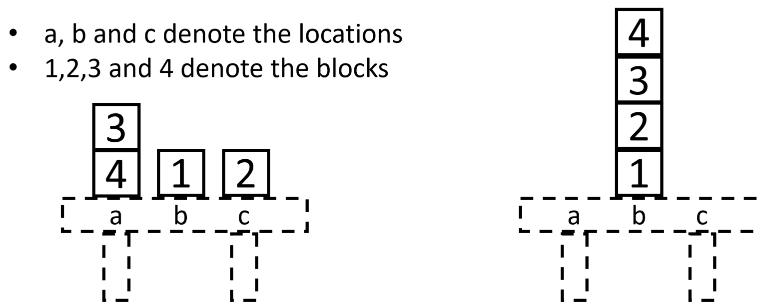
$V(A)$	$V(B)$
1	-1

Average the following for A:
First Episode: (+2+1-2+2-1,+1-2+2-1,+2-1) = (2,0,1)
Second Episode: (+2-1) = (1)
$V(A) = (2 + 0 + 1 + 1)/4 = 1$

Average the following for B:
First Episode: (-2+2-1,-1) = (-1,-1)
Second Episode: (-2+2-1,-1) = (-1,-1)
$V(A) = (-1 - 1 - 1 - 1)/4 = -1$

If you did it in a "first-visit" manner then
V(A) = (2+1)/2 = 1.5
V(B) = (-1+-1)/2 = -1

3. (12 points) Imagine a problem where there are n numbered blocks on top of a table. Blocks can be stacked on top of each other.There are also k discreet locations on the table. The goal is to stack the blocks such that they are ordered from the smallest to the largest on any of the locations. The starting block positions can be random. An example for $n = 4$ and $k = 3$ is given below.



An example starting state. This can be random.

An example goal state. The important thing is the block

You can only move one block at a time. You cannot move a block if there is another block on top of it. If you were to formulate this as a search problem (make sure to provide a general formulation and not something specific to the above example):

(a) (4 points) What would be your state representation? What would be your goal test for this representation?

A 2D $k \times n$ integer array, where the first dimension represents the table locations and the second dimension represents the blocks for each location (3 points). Note that this handles stacks as well. Goal test would be to check whether the order is achieved by looking at each location (1 point). Other answers, such as using linked lists for each location, are possible.

(b) (8 points) What algorithm would you chose to solve this problem? If any, what heuristic would you use and why? Would this heuristic favor solution efficiency or computational efficiency?

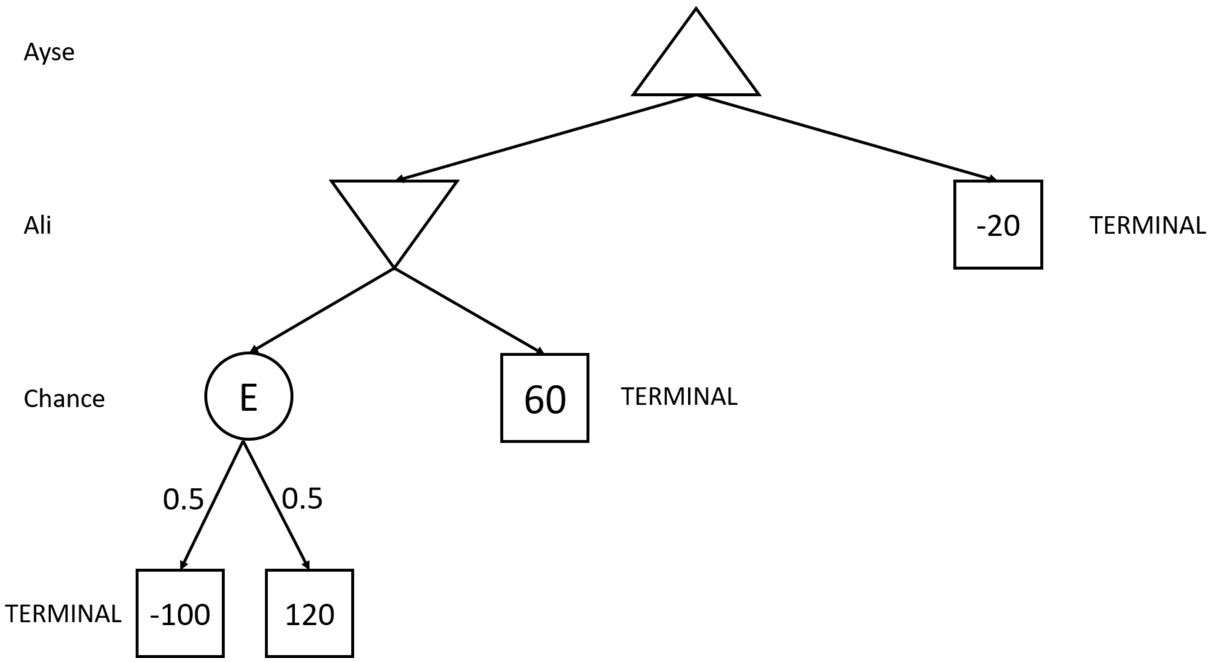
A* search if given a heuristic would get 3 points. A* without a heuristic, Uniform Cost Search, Breadth First Search, Iterative Deepeining Depth First Search would get 2 points. Depth First Search with a justification for computational efficiency get 2 points. Just DFS would get 1 point.

Some A* Heuristics:

- Number of blocks out of place: 1 point by itself. Admissible justification 2 points. Easy to compute additional 1 point.
- In addition to the above, +2 for each block that is not directly above the one it is supposed to. Note that to move a block to its appropriate place in this situation, you must at least make 2 moves. Such a heuristic, along with justification gets at least 2 points.
- Sum of Manhattan Distances to Place: 1 point by itself. Inadmissible but potentially faster to find a (non-optimal) solution justification 1 point. Easy to compute additional 1 point.

There are other heuristics. Grade will depend on the justification and admissibility. If your heuristic looks to be very good, you may get bonus points!

4. (12 points) Ayse and Ali are playing an adversarial game as shown in the game tree below. Ayse (the MAX player) and Ali (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 probability. The tree is given below:



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 Ali will pay Ayse x units of money after the game, so that Ayse's and Ali's utilities will be x and $-x$ respectively.

(a) (3 points) Suppose neither Ayse nor Ali knows the outcome of E before playing. What is Ayse's expected utility? Is this a fair game?

Value of $E = -100 \cdot 0.5 + 120 \cdot 0.5 = 10$. Thus Ali would chose left. Then Ayse would also chose left. Thus Ayse's expected utility is 10. This is not a fair game since Ayse will win more money in the long run (non-zero expected utility).

(b) (1 point) Ceren, a good friend of Ayse's, has access to E and can secretly tell Ayse the outcome of E before the game starts (giving Ayse the true outcome of E without lying). However, Ali is not aware of any communication between Ayse and Ceren, so he still assumes that Ayse has no access to E. What is Ayse's expected utility if Ceren tells her that $E = -100$?

-20 since that Ali would chose left to win 100. Ayse would rather pay 20 than 100 thus she would chose right

(c) (1 point) (Continuing from part b) What is Ayse's expected utility if Ceren tells her that $E = +120$?

120 since that Ali would still chose left and the outcome would be 120.

(d) (1 point) (Continuing from part c) What is Ayse's expected utility if Ceren secretly tells her the outcome of E before playing?

Since both are equally likely (0.5 probability for E's outcomes), Ayse's utility in this case would be: $-20 \cdot 0.5 + 120 \cdot 0.5 = 50$

(e) (2 points) We define the value of *private information* $V_A^{pri}(X)$ 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. Can we assert whether or not $V_A^{pri}(X) \geq 0$? Justify your answer.

Yes. There is no deception so we know that the private information is correct. Also, player A can always choose to ignore this information and act in the same way as if he/she doesn't know this information, thus player A is guaranteed to obtain at least the same utility as before.

(f) (1 point) What is, $V_{Ayse}^{pri}(E)$, the value of private information of E to Ayse in the game tree above?

Follow the definition: 50 - 10 = 40

(g) (1 point) Ahmet also has access to E, and can make a public announcement of E (announcing the true outcome of E without lying), so that both Ayse and Ali will know the outcome of E and are both aware that their opponent also knows the outcome of E. Also, Ayse cannot obtain any information from Ceren now. What is Ayse's expected utility if Ahmet announces that $E = -100$?

Remove the chance effect and follow the minimax algorithm to get -20

(h) (1 point) (Continuing from part g) What is Ayse's expected utility if Ahmet announces that $E = +120$?

Remove the chance effect and follow the minimax algorithm to get 60

(i) (1 point) (Continuing from part h) What is Ayse's expected utility if Ahmet announces the outcome of E before playing?

Equally likely thus 20.

5. (10 points) You are given unique words in an alien language. You want to form sentences using these words of length k (i.e. sentences are formed using k words). You additionally have access to an oracle which can score the sentences. Among all the things you have learned in this course, which method would you use to solve this problem? Formulate it as clearly as possible.

   Among the methods we have learned in class, local search is the most suitable. Hill-climbing (all versions we have seen) or genetic algorithms would be suitable. If there are many words, a stochastic version of hill-climbing would be better. I am going to provide one such example:

   - State: A k dimensional tuple, each dimension representing a word
   - Transition Function: Replacing a single word (see below how) with another word
   - Evaluation Function: The oracle score

   The algorithm:

   1. Initialize the state with random words
   2. Randomly generate l many new states (sentences) by:
      - Randomly pick a position
      - Replace the word in that position with another random word from the alien dictionary
      - Calculate the oracle score
   3. Pick the highest scoring sentence and go back to step 2 until the score is not improved or a pre-determined number of iterations are reached

   You may have tried to use reinforcement learning as well. This is very tricky to do and not needed. Still some partial will be given.

6. (12 points) You are still strying to find your friend in the building (from MT3). F1 is the first floor and F4 is the fourth floor. Your friend randomly moves between the floors. The probability of going up and down by one level at each floor (e.g. your friend can only go to 1st or 3rd floor, or stay in the 2nd floor if he is already in the 2nd floor) is given below. In addition, there is a sound sensor on the first floor with emission probabilities given below.

F4
F3
F2
F1

X_T	Stay	Down	Up
F4	0.6	0.4	-
F3	0.5	0.3	0.2
F2	0.5	0.2	0.3
F1	0.6	-	0.4

S	P(S|X)
-s	0.1+0.2d
+s	0.9-0.2d

d: floor difference from the first floor
X = F1, d = 0
X = F2, d = 1
X = F3, d = 2
X = F4, d = 3

This time you are going to use particle filtering to try to find your friend.

(a) (2 points) The sensor senses a sound ($S = +s$). What is the weight of each particle for a given state after this observation? Show your work below the table and fill the table with your results

F1	F2	F3	F4
0.9	0.7	0.5	0.3

Just plug in $P(+s|X)$ into the emission model $0.9 - 0.2d$!

(Figure repeated here for your convenience)

F4
F3
F2
F1

X_T	Stay	Down	Up
F4	0.6	0.4	-
F3	0.5	0.3	0.2
F2	0.5	0.2	0.3
F1	0.6	-	0.4

S	P(S|X)
-s	0.1+0.2d
+s	0.9-0.2d

d: floor difference from the first floor
X = F1, d = 0
X = F2, d = 1
X = F3, d = 2
X = F4, d = 3

(b) (4 points) You calculated the weights of each particle. Then you decide to add the weights of the particles that are in the same state. You get the following totals after this step (The previous part and this part are independent):

F1	F2	F3	F4
5	3	2	0

You want to sample the next particles with probabilities proportional to their total weights. You are given the following uniform random samples, [0.73, 0.47, 0.15, 0.92]. Assuming the intervals for sampling are formed left to right based on the above table, what are the states of the next 4 particles? Show your work.

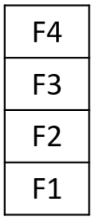
For sampling, we need to normalize the weight totals, i.e., divide them by the total, and then calculate the cumulative distribution:

X	Weights	Normalized	Cumulative
F1	5	0.5	0.5
F2	3	0.3	0.8
F3	2	0.2	1.0
F4	0	0.0	1.0

The next step is to pick the particles based on where the uniform random samples fall. 0.73 yields F2, 0.47 yields F1, 0.15 yields F1 and 0.92 yields F3.

We end up with 2 particles with state F1, 1 particle with state F2 and 1 particle with state F3.

(Figure repeated here for your convenience)



X_T	Stay	Down	Up
F4	0.6	0.4	-
F3	0.5	0.3	0.2
F2	0.5	0.2	0.3
F1	0.6	-	0.4

S	P(S|X)
-s	0.1+0.2d
+s	0.9-0.2d

d: floor difference from the first floor
X = F1, d = 0
X = F2, d = 1
X = F3, d = 2
X = F4, d = 3

(c) (4 points) You have 2 particles with state F1 and 2 particles with state F4 (The previous part and this part are independent). You want to sample next points based on the underlying model dynamics (ie. you want to elapse time). You are given the following uniform random samples, [0.73, 0.47, 0.15, 0.92]. Assuming the intervals for sampling are formed left to right based on the above table, what are the states of the next 4 particles? Show your work.

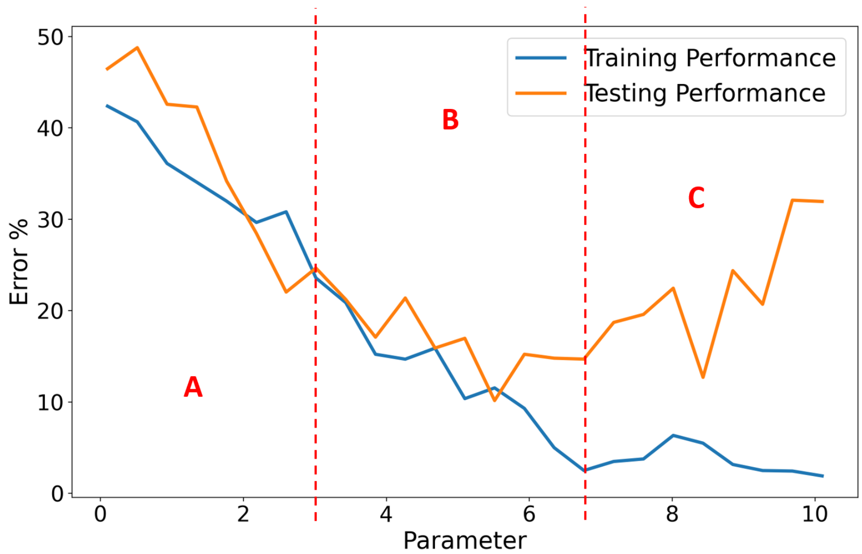
The transition probabilities for F1 and F4:

X_t	X_{t+1}	$P(X_{t+1}\|X_t)$	Cumulative
F1	F1	0.6	0.6
F1	F2	0.4	1.0
F4	F3	0.4	0.4
F4	F4	0.6	1.0

For the first two particles (X_t = F1), 0.73 yields F2 and 0.47 yields F1. For the latter 2 particles, 0.15 yields F3 and 0.92 yields F4. At the end we have 1 particle for each state.

7. (14 points) Machine Learning Potpourri: Answer the questions below

(a) (3 points) Given the figure below match the regions with the "fit type".

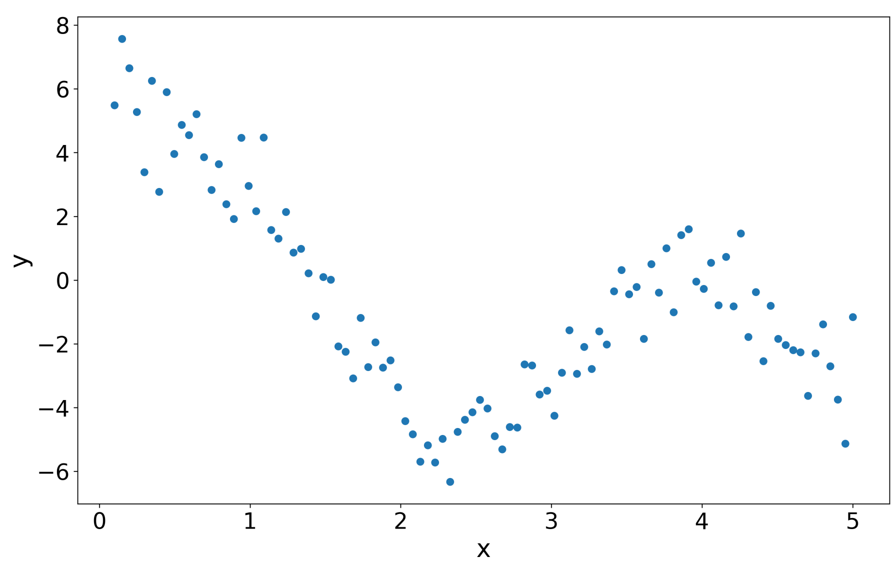


Good Fit: B Under Fit: A Over Fit: C

(b) (2 points) Given only the above figure, what value (approximately) would you chose for the parameter?

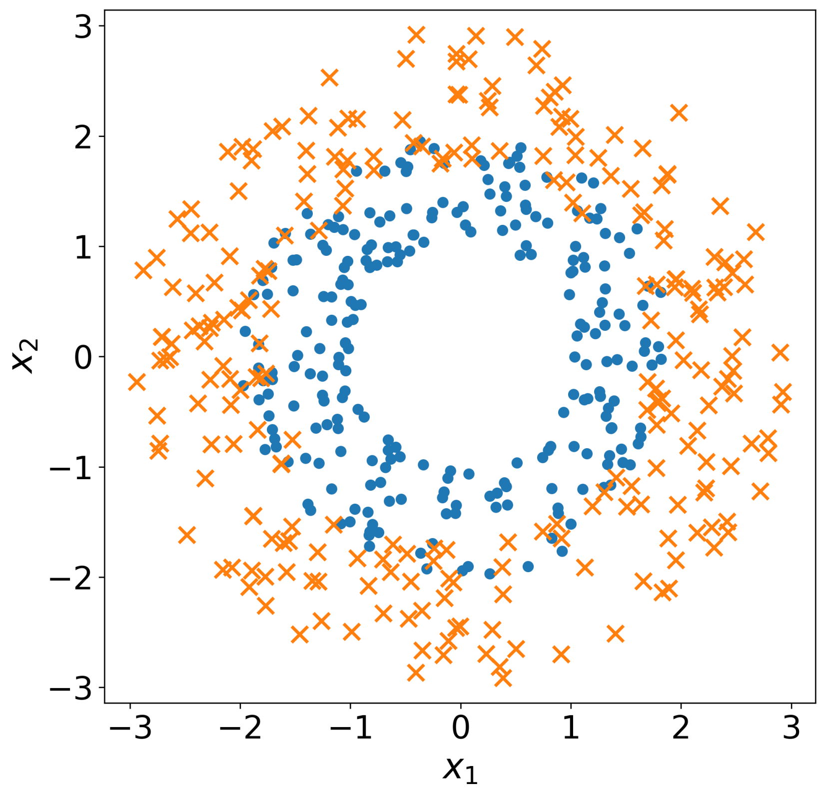
Around 5.5

(c) (3 points) You want to use linear regression to fit a function to the data given below where x is the input and y is the output. How would you formulate this, i.e., what would be the form of the function you fit? Make sure to clearly mark the parameters and inputs.



The figure looks like a third degree or fourth degree (possible to interpret the topleft as curving downward subtly) polynomial. Then we would try to fit $y = f(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3$, where w_i denotes the parameters, x is the input and y is the target.

(d) (2 points) You are given the below data points (x_1, x_2) and their corresponding classes, marked by their color (blue vs orange) and marker shape (dot vs X). Is the kNN method feasible for solving this classification problem using only the given inputs (x_1, x_2)?



Yes

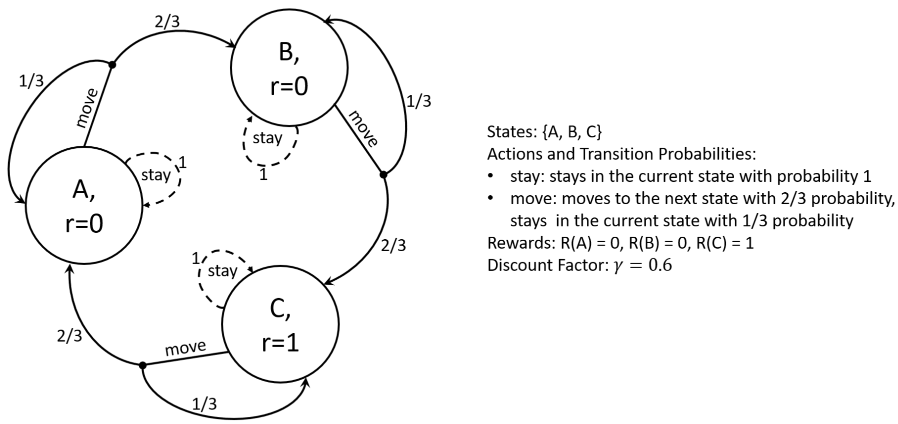
(e) (2 points) Come up with a feature that would be useful for this classification problem? (e.g. assume you want to solve it with a linear method)

The class of the data changes as we move away from the origin, i.e., based on the norm of the data. Thus, this would make a good feature: $x^2 + y^2$

(f) (2 points) Can a realistic **parametric method** reach 100% training accuracy for this problem?

No, because the boundry is very noisy. The model will inetivably be wrong around the boundary

8. (9 points) Dynamic Programming: Answer the questions based on the MDP below



(a) (6 points) Perform one step of value iteration and fill in the table below. Make sure to show your work below the table.

Iteration	$V(A)$	$V(B)$	$V(C)$
0	0	0.4	1.6
1	0.16	0.72	1.96

First expression in the max is for the move action and the other is for the stay action.

$$\begin{aligned}
V^{\pi}(A) &= \max(0 + 3/5 \cdot (1/3 \cdot 0 + 2/3 \cdot 2/5, 0 + 3/5 \cdot (1 \cdot 0)) \\
&= \max(4/25, 0) = 4/25 = 0.16, \textit{move} \text{ action is better} \\
V^{\pi}(B) &= \max(0 + 3/5 \cdot (1/3 \cdot 2/5 + 2/3 \cdot 8/5), 0 + 3/5 \cdot (1 \cdot 2/5)) \\
&= \max(18/25, 6/25) = 18/25 = 0.72, \textit{move} \text{ action is better} \\
V^{\pi}(C) &= \max(1 + 3/5 \cdot (1/3 \cdot 8/5 + 2/3 \cdot 0, 1 + 3/5 \cdot (1 \cdot 8/5)) \\
&= \max(33/25, 49/25) = 49/25 = 1.96, \textit{stay} \text{ action is better}
\end{aligned}$$

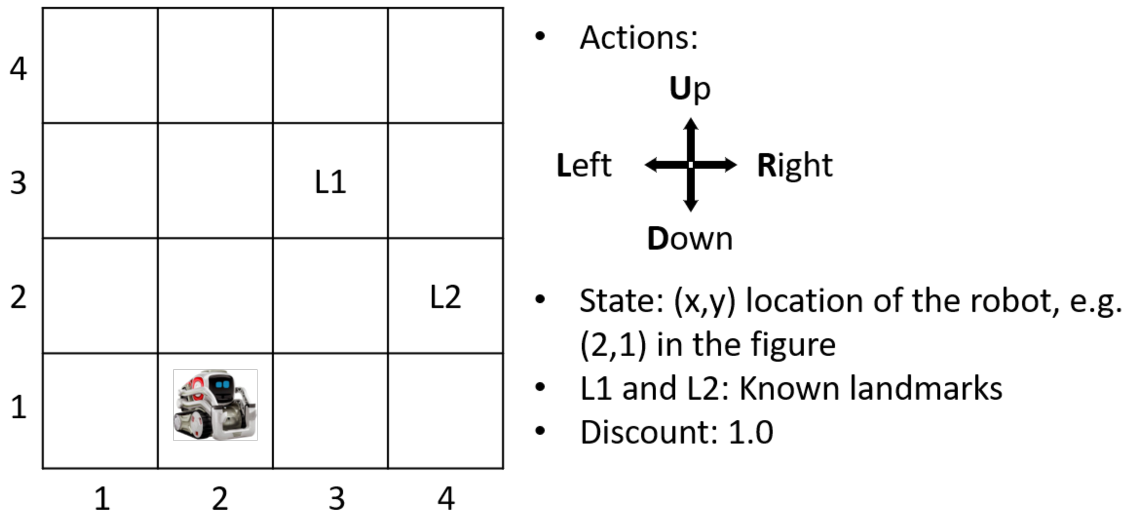
2 points per correct value. 1 point if there is a mathematical error.

(b) (3 points) What is the policy extracted from the calculated these values?

Look at the argmax values.

$\pi(A) = \textit{move}$, $\pi(B) = \textit{move}$, $\pi(C) = \textit{stay}$, 1 point per correct answer

9. (15 points) Value Function Approximation. The robot given below is trying to explore the area and find safe routes to resources. The state of the robot is the grid it is in. Robot can move in four cardinal directions. The landmarks, L1 and L2, signify that there is a resource close-by. The locations of these landmarks are known to the robot (L1 $= (x_{l1}, y_{l1})$ and L2 $= (x_{l2}, y_{l2})$).



The robot wants to use function approximation get the values of each state. It decides to use the following features, given the current state $s = (x, y)$.

- Current x-coordinate: $f_1(s) = x$
- Current y-coordinate: $f_2(s) = y$
- Manhattan Distance to L1: $f_3(s) = |x - x_{l1}| + |y - y_{l1}|$
- Manhattan Distance to L2: $f_4(s) = |x - x_{l2}| + |y - y_{l2}|$

Furthermore, it uses a linear function approximator:

$$\hat{V}(s, w) = w_1 f_1(s) + w_2 f_2(s) + w_3 f_3(s) + w_4 f_4(s) = w^T f(s)$$

The robot then observes the following transitions:

$$(2,1), -0.1 \rightarrow (2,2), -0.1 \rightarrow (2,3), +1$$

Answer the questions below:

(a) (3 points) Calculate the feature vectors of the observed states

$L1 = (3,3), L2 = (4,2)$

$$f((2,1)) = \begin{bmatrix} 2 \\ 1 \\ 3 \\ 3 \end{bmatrix}, \quad f((2,2)) = \begin{bmatrix} 2 \\ 2 \\ 2 \\ 2 \end{bmatrix}, \quad f((2,3)) = \begin{bmatrix} 2 \\ 3 \\ 1 \\ 3 \end{bmatrix}$$

(b) (12 points) Use the observed transitions to update the weights, starting from zero weights with the learning rate $\alpha = 0.2$ and the discount factor $\gamma = 1.0$.

$w = [0,\ 0,\ 0,\ 0]^T$

First Update: $w \leftarrow w + \alpha(r + w^T f(s_2) - w^T f(s_1))f(s_1)$

$$w \leftarrow \begin{bmatrix}0\\0\\0\\0\end{bmatrix} +0.2(-0.1+0-0)\begin{bmatrix}2\\1\\3\\3\end{bmatrix} \Rightarrow w = \begin{bmatrix}-0.04\\-0.02\\-0.06\\-0.06\end{bmatrix}$$

Second Update: $w \leftarrow w + \alpha(r + w^T f(s_3) - w^T f(s_2))f(s_2)$

$$w \leftarrow \begin{bmatrix}-0.04\\-0.02\\-0.06\\-0.06\end{bmatrix} +0.2(-0.1+[-0.04,\ -0.02,\ -0.06,\ -0.06]^T\begin{bmatrix}2\\3\\1\\3\end{bmatrix}-[-0.04,\ -0.02,\ -0.06,\ -0.06]^T\begin{bmatrix}2\\2\\2\\2\end{bmatrix})\begin{bmatrix}2\\2\\2\\2\end{bmatrix}$$

$$= \begin{bmatrix}-0.04\\-0.02\\-0.06\\-0.06\end{bmatrix} + 0.2(-0.1-0.38+0.36)\begin{bmatrix}2\\2\\2\\2\end{bmatrix} \Rightarrow w = \begin{bmatrix}-0.088\\-0.068\\-0.108\\-0.108\end{bmatrix}$$

It is okay if you did it separately for each weight vector w_i, should get the same answer. 1.5 point per weight per step (4 dimensions and 2 steps = 8 answers). 0.5-1 point for calculation error if the path is correct. To grade the second step, assume that the 1st step's output is totally correct.