15-381	Artificial Intelligence	HW9

INSTRUCTIONS

- **Due: Tuesday, 2 April 2019 at 10:00 PM EDT.** Remember that you have NO slip days for Written Homework, but you may turn it in up to 24 hours late with 50% Penalty.
- **Format:** Submit the answer sheet pdf containing your answers. You should solve the questions on this handout (either through a pdf annotator, or by printing, then scanning). Make sure that your answers (typed or handwritten) are within the dedicated regions for each question/part. If you do not follow this format, we may deduct points.
- **How to submit:** Submit a pdf with your answers on Gradescope. Log in and click on our class 15-381 and click on the submission titled HW9 and upload your pdf containing your answers.
- **Policy:** See the course website for homework policies and Academic Integrity.

Name	
Andrew ID	

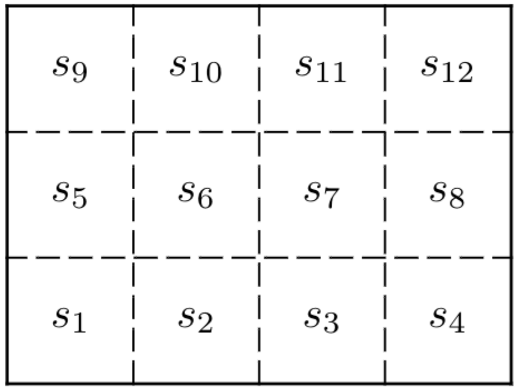
For staff use only

Q1	Q2	Q3	Q4	Total
/22	/30	/25	/23	/100

Q1. [22 pts] Dark Room

A robot mysteriously finds itself in a dark room with 12 states as shown below. The robot can take four actions at each state (North, East, South, and West). We do not know where the interior walls are, except for the walls surrounding the room (represented by solid lines).

An action against a wall leaves the robot in the same state. Otherwise, the outcome of an action deterministically moves the robot in the direction corresponding to the action.



The robot learns the Q-values below. Note that only the maximum values for each state are shown, as the other values (represented with '-') do not affect the final policy.

	N	E	S	W
s_1	59	59	-	-
s_2	-	66	-	-
s_3	-	73	-	-
s_4	81	-	-	-
s_5	66	-	-	-
s_6	-	59	-	59
s_7	-	-	66	-
s_8	90	-	-	-
s_9	-	73	-	-
s_{10}	-	81	-	-
s_{11}	-	90	-	-
s_{12}	100	100	-	-

(a) [4 pts]

Using the learned Q-values, what are the first six actions the robot takes if it starts in state s_7? If there is more than one best action available, choose one randomly. Assume that the robot does not encounter any interior walls.

(b) [9 pts]

We are told the discount factor used during Q-learning was $\gamma = 0.9$.
We are also told there exists a single state, s^* such that $R(s^*, a, s') > 0 \ \forall a, s'$, and for all other states s, $R(s, a, s') = 0 \ \forall a, s'$. Also assume that $R(s^*, a, s')$ are equivalent for all a, s'.

(i) What is s^*, and what is the reward?

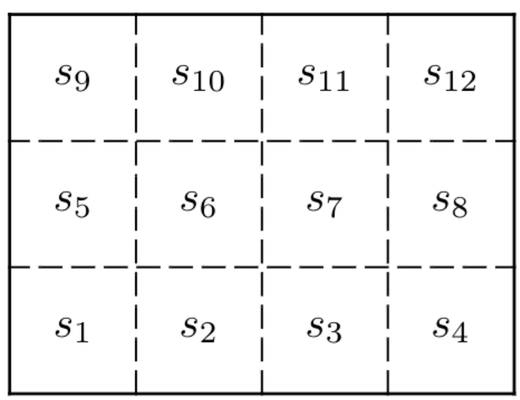
s^***:**	$R(s^*, a, s')$**:**

(ii) Explain how you found your answers.

(c) [9 pts]

Now the robot wants to locate the interior walls within the room.

(i) Where are these walls? Draw them in by filling the corresponding dashed lines below.

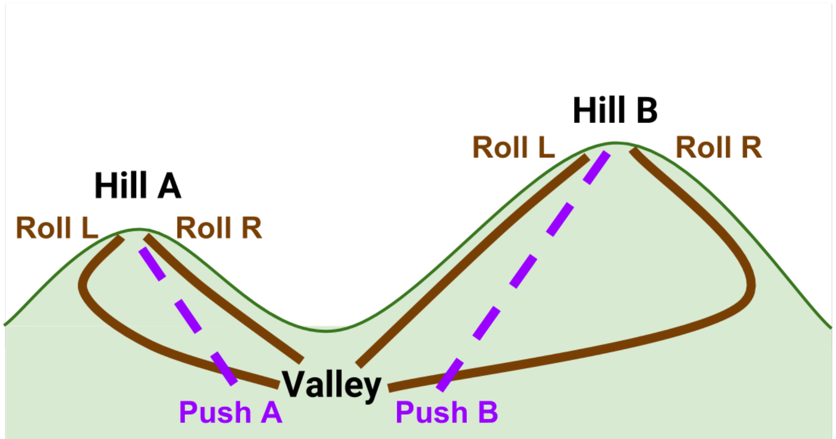


(ii) Explain how you know where the walls are.

Q2. [30 pts] Representation for Buggy Bots

In this problem, we explore how domain representation affects performance and computational time when learning policies for MDPs.

Carnival is upon us, and we are training bots to push our buggy! The buggy bots move between three states: $Valley$, $HillA$, and $HillB$. From the valley, bots can choose to push up to Hill A or push up to Hill B. From the top of either hill, the available actions are to roll left or roll right.



The true transition matrix and reward function are shown in tables below. All actions are deterministic. At both $HillA$ and $HillB$, rolling right gives positive reward and rolling left gives negative reward, but the positive reward in $HillB$ is slightly higher. In $Valley$, both $PushA$ and $PushB$ give negative reward, but $PushA$ gives a slightly less negative reward.

s	a	s'	$T(s,a,s')$	$R(s,a)$
$HillA$	$RollL$	$Valley$	1.0	-1.0
$HillA$	$RollR$	$Valley$	1.0	0.7
$HillB$	$RollL$	$Valley$	1.0	-1.3
$HillB$	$RollR$	$Valley$	1.0	1.0
$Valley$	$PushA$	$HillA$	1.0	-0.5
$Valley$	$PushB$	$HillB$	1.0	-0.7

We consider two representations of the domain by two different robots. Robot One has a sensor that tells it whether it is in state $HillA$, $HillB$, or $Valley$ (allowing it to fully represent the true state). Robot Two has a simpler representation, having only a hill/valley sensor, which allows it to distinguish between $Valley$ and $HillA$ or $HillB$, but it can't distinguish between $HillA$ and $HillB$.

(a) [8 pts] Using a discount factor of $\gamma = 0.9$, Robot One learns the following Q-values:

s	a	$Q(s,a)$
$HillA$	$RollL$	-0.05
$HillA$	$RollR$	1.65
$HillB$	$RollL$	-0.35
$HillB$	$RollR$	1.95
$Valley$	$PushA$	0.98
$Valley$	$PushB$	1.05

(i) What is the optimal policy for Robot One?

$HillA$: ◯ $RollL$ ◯ $RollR$

$HillB$: ◯ $RollL$ ◯ $RollR$

$Valley$: ◯ $PushA$ ◯ $PushB$

(ii) What is the value of this policy? Specify the values with three significant digits. *Note*: You may want to code this up to quickly compute these.

$V^{\pi_1}(HillA)$:	$V^{\pi_1}(HillB)$:	$V^{\pi_1}(Valley)$:

(b) [11 pts] Robot Two uses a uniform exploration strategy while learning, so the observed reward in the *Hill* state is the average of the rewards *HillA* and *HillB*, as each are visited roughly equally. Under this exploration strategy, with a discount factor of $\gamma = 0.9$, Robot Two learns the following Q-values:

s	a	$Q(s, a)$
$Hill$	$RollL$	0.11
$Hill$	$RollR$	2.11
$Valley$	$PushA$	1.39
$Valley$	$PushB$	1.19

(i) What is the optimal policy for Robot Two?

Hill: ○ *RollL* ○ *RollR*

Valley: ○ *PushA* ○ *PushB*

(ii) How is this policy represented in the original domain, i.e. how would Robot One represent this policy?

Answer:

(iii) What is the value of this policy in the original domain, i.e. what would the value of this policy be if it were evaluated by Robot One. Specify the values with three significant digits. *Note*: You will want to code this up to quickly compute these.

$V^{\pi_2}(HillA)$:	$V^{\pi_2}(HillB)$:	$V^{\pi_2}(Valley)$:

(c) [11 pts] Based on the results from Robot One and Robot Two, answer the following questions:

(i) Which of the policies performs better on the original domain?

◯ Robot One ◯ Robot Two

(ii) Can the optimal policy for the original domain be expressed in both domains? Why or why not?

Answer:

(iii) Assume one iteration of value iteration using Robot One's representation takes t seconds. How long, in terms of t, should we expect one iteration of value iteration to take for Robot Two's representation?

Robot Two time:

(iv) What are the trade-offs in choosing the representations for problems with large state spaces?

Answer:

Q3. [25 pts] Approximate Q-learning

A robot is trying to get to its office hours, occurring on floors 3, 4, or 5 in GHC. It is running a bit late and there are a lot of students waiting for it. There are three ways it can travel between floors in Gates: the stairs, the elevator, and the helix.

The **state** of the robot is the floor that it is currently on (either 3, 4, or 5).

The **actions** that the robot can take are *stairs*, *elevator*, or *helix.*

In this problem, we are using a linear, feature based approximation of the Q-values:

$$Q_w(s,a) = \sum_{i=0}^{3} f_i(s,a)w_i$$

We define the feature functions as follows:

Features	**Initial Weights**
$f_0(s,a) = 1$ (this is a bias feature that is always 1)	$w_0 = 1$
$f_1 = f_{\text{time}}(s,a) = (\lvert s-4\rvert + 1)t$, where $t = \begin{cases} 10, a = \text{elevator} \\ 20, a = \text{stairs} \\ 50, a = \text{helix} \end{cases}$	$w_1 = 0.5$
$f_2 = f_{\text{accessibility}}(s,a) = \begin{cases} 2, a = \text{stairs} \\ 5, a = \text{helix} \\ 10, a = \text{elevator} \end{cases}$	$w_2 = 2$
$f_3 = f_{\text{wait}}(s,a) = \begin{cases} 60, a = \text{elevator}, s = 3 \\ 80, a = \text{elevator}, s = 4 \\ 60, a = \text{elevator}, s = 5 \\ 0, \text{otherwise} \end{cases}$	$w_3 = 0.1$

Furthermore, the weights will be updated as follows:

$$w_i \leftarrow w_i + \alpha[r + \gamma \max_{a'} Q_w(s',a') - Q_w(s,a)]\frac{\delta}{\delta w_i}Q_w(s,a)$$

(a) [9 pts] Calculate the following initial Q values given the initial weights above.

Q_w(4, elevator):	**Q_w(4, stairs):**	**Q_w(4, helix):**

(b) [3 pts] The initial Q-values for state 3 happen to be equal to the corresponding initial Q-values for state 5.

(i) As you update the weights, will these values remain equal? Eg. will $Q_w(3, a) = Q_w(5, a)$ given any action a and vector w?

◯ Yes ◯ No

(ii) Why or why not?

Answer:

(c) [3 pts] Given the Q-values for state 4 calculated in part (a), what are the probabilities that each of the following actions could be chosen when using ϵ greedy exploration from state 4 (assume random movements are chosen uniformly from all actions)?

stairs:	**elevator:**	**helix:**

(d) [8 pts] Given a sample with start state 3, action = stairs, successor state = 4, and reward = -2, update each of the weights using learning rate $\alpha = 0.25$ and discount factor $\gamma = 0.6$.

$w_0 =$
$w_1 =$
$w_2 =$
$w_3 =$

(e) [2 pts] What is an advantage of using approximate Q-learning instead of the standard Q-learning? What is a disadvantage?

Answer:

Q4. [23 pts] Probability

"True love is the greatest thing in the world...except a nice MLT: mutton, lettuce, and tomato sandwich, where the mutton is nice and lean..." *–Miracle Max*

You are given the following probability tables for binary random variables M, L, T:

T	$P(T)$
$+t$	0.4
$-t$	0.6

L	T	$P(L \mid T)$
$+l$	$+t$	0.8
$+l$	$-t$	0.25
$-l$	$+t$	0.2
$-l$	$-t$	0.75

L	T	M	$P(M \mid L, T)$
$+l$	$+t$	$+m$	0.95
$+l$	$+t$	$-m$	0.05
$+l$	$-t$	$+m$	0.75
$+l$	$-t$	$-m$	0.25
$-l$	$+t$	$+m$	0.40
$-l$	$+t$	$-m$	0.60
$-l$	$-t$	$+m$	0.10
$-l$	$-t$	$-m$	0.90

(a) [8 pts] Calculate $P(L, T, M)$ from the tables given.

L	T	M	$P(L, T, M)$
$+l$	$+t$	$+m$	
$+l$	$+t$	$-m$	
$+l$	$-t$	$+m$	
$+l$	$-t$	$-m$	
$-l$	$+t$	$+m$	
$-l$	$+t$	$-m$	
$-l$	$-t$	$+m$	
$-l$	$-t$	$-m$	

(b) [5 pts] Which of the following are valid decompositions of the joint probability distribution of M, L, and T, given no assumptions about the relationship between these random variables. Select all that apply.

- ☐ i) $P(M)P(L)P(T)$
- ☐ ii) $P(M)P(M, L)P(M, L, T)$
- ☐ iii) $P(M, L \mid T)P(T)$
- ☐ iv) $P(M \mid L, T)P(L)$
- ☐ v) $P(M \mid L, T)P(T)$
- ☐ vi) None of the above

(c) [10 pts] Let A be a random variable representing the choice of protein in the sandwich with three possible values, $\{mutton, bacon, egg\}$, let B be a random variable representing the choice of bread with two possible values, $\{toast, naan\}$, and let K be a random variable representing the presence of ketchup or not, $\{+k, -k\}$. How many values are in each of the probability tables and what do the entries sum to?

Write '?' if there is not enough information given.

Table	**Num**	**Sum**
i) $P(A, B)$		
ii) $P(A, B, +k)$		
iii) $P(A, B \mid +k)$		
iv) $P(B \mid +k, A)$		