Q4. [14 pts] RL: Rest and ReLaxation

Consider the grid world MDP below, with unknown transition and reward functions.

A	B	C
D	E	F
G	H	I

The agent observes the following samples in this grid world:

s	a	s'	$R(s, a, s')$
E	East	F	-1
E	East	H	-1
E	South	H	-1
E	South	H	-1
E	South	D	-1

Reminder: In grid world, each non-exit action succeeds with some probability. If an action (e.g. North) fails, the agent moves in one of the cardinally adjacent directions (e.g. East or West) with equal probability, but will not move in the opposite direction (e.g. South).

Let p denote the probability that an action succeeds.

In this question, we will consider 3 strategies for estimating the transition function in this MDP.

Strategy 1: The agent does not know the rules of grid world, and runs model-based learning to directly estimate the transition function.

(a) [1 pt] From the samples provided, what is $\hat{T}$(E, South, H)?

○ 0
○ 1/5
○ 2/5
○ 3/5
○ 4/5
○ 1/3
○ 2/3
○ 1/2
○ 1
○ Not enough information

(b) [1 pt] From the samples provided, what is $\hat{T}$(E, West, D)?

○ 0
○ 1/5
○ 2/5
○ 3/5
○ 4/5
○ 1/3
○ 2/3
○ 1/2
○ 1
○ Not enough information

Strategy 2: The agent knows the rules of grid world, and runs model-based learning to estimate p. Then, the agent uses the estimated $\hat{p}$ to estimate the transition function.

(c) [1 pt] From the samples provided, what is $\hat{p}$, the estimated probability of an action succeeding?

○ 0
○ 1/5
○ 2/5
○ 3/5
○ 4/5
○ 1/3
○ 2/3
○ 1/2
○ 1
○ Not enough information

(d) [1 pt] Based on $\hat{p}$, what is $\hat{T}$(E, West, D)?

○ 0
○ 1/5
○ 2/5
○ 3/5
○ 4/5
○ 1/3
○ 2/3
○ 1/2
○ 1
○ Not enough information

(e) [3 pts] Select all true statements about comparing Strategy 1 and Strategy 2.

□ Strategy 1 will usually require fewer samples to estimate the transition function to the same accuracy threshold.
□ There are fewer unknown parameters to learn in Strategy 1.
□ Strategy 1 is more prone to overfitting on samples.
○ None of the above

The grid world and samples, repeated for your convenience:

A	B	C
D	E	F
G	H	I

s	a	s'	$R(s,a,s')$
E	East	F	-1
E	East	H	-1
E	South	H	-1
E	South	H	-1
E	South	D	-1

Strategy 3: The agent knows the rules of grid world, and uses an exponential moving average to estimate p. Then, the agent uses the estimated $\hat{p}$ to estimate the transition function.

(f) [2 pts] Consider this update equation: $\hat{p} \leftarrow (1-\alpha)\hat{p} + (\alpha)x$

Given a sample (s, a, s'), what value of x should be used in the corresponding update?

○ $R(s, a, s')$

○ 1.0 if the action succeeded, and 0.0 otherwise

○ 1.0 if the action failed, and 0.0 otherwise

○ $V(s)$

○ $V(s')$

(g) [3 pts] Select all true statements about comparing Strategy 2 and Strategy 3.

□ Strategy 2 gives a more accurate estimate, because it is the maximum likelihood estimate.

□ Strategy 3 gives a more accurate estimate, because it gives more weight to more recent samples.

□ Strategy 3 can be run with samples streaming in one at a time.

○ None of the above

The rest of the question is independent from the previous subparts.

Suppose the agent runs Q-learning in this grid world, with learning rate $0 < \alpha < 1$, and discount factor $\gamma = 1$.

(h) [1 pt] After iterating through the samples once, how many learned Q-values will be nonzero?

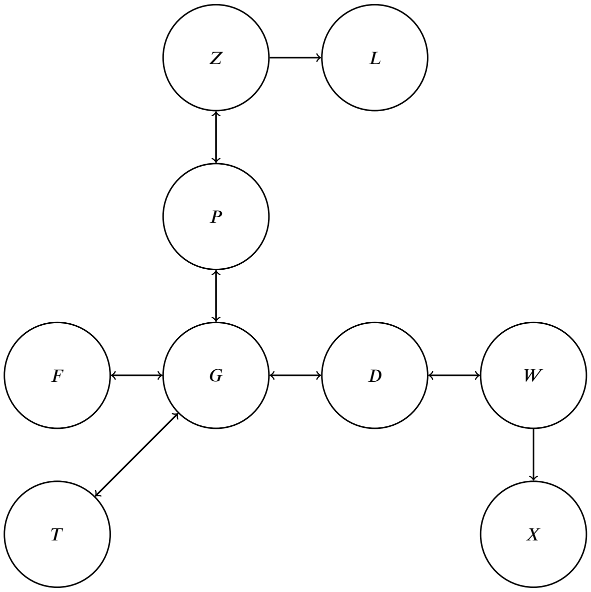
○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ > 4

(i) [1 pt] After iterating through the samples repeatedly until convergence, how many learned Q-values will be nonzero?

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ > 4

Q6. [12 pts] Reinforcement Learning: Island Hopping

Bob wants to traverse different islands to reach the island X. He formulates the problem as an MDP where the islands (nodes) represent the states and the arrows represent the possible actions he can take across the seas. Use the direction of the arrows (up, down, left, right) to refer to the specific actions that can be taken.



(a) [2 pts] Bob's ship doesn't always move in the direction that he wants it to. Despite this, he wants to use MLE to build an estimate of the transition function $\hat{T}$ and the reward function $\hat{R}$ for model-based reinforcement learning. He follows some specified policy and collects some data in the form of (current state, action, next state, reward) tuples shown below:

State (s)	Action (a)	New State (s')	Reward
F	Right	G	20
D	Right	G	-10
G	Up	T	-30
P	Down	Z	-15
G	Right	D	30
W	Down	D	-25
G	Right	D	30
D	Left	G	-5
G	Right	T	-30
W	Down	X	100

(i) [1 pt] What is $\hat{T}$(G, Right, D)?

(ii) [1 pt] What is $\hat{R}$(W, Down, D)?

(b) [3 pts] Bob looks to use temporal difference learning to learn the values of π, where he has the following initial values:

s	F	G	T	P	Z	L	D	W	X
$V^{\pi}(s)$	5	10	-4	8	2	-10	10	30	50

He performs one update step using the sample (G, Right, D, 30). Assume a discount factor $\gamma = 0.5$ and the learning rate $\alpha = 0.2$. What is the updated value of $V^{\pi}(G)$? Show all your work leading to your answer.

(c) [2 pts] Bob wants to have a greedy policy that will minimize exploration and maximize exploitation. Which of the following functions f will do so? Assume k is a positive real number, $N(s,a)$ represents the number of times that the state-action pair is taken, and ϵ is a very small number greater than 0.

□ $f(s,a) = Q(s,a) + \frac{k}{N(s,a)}$

□ $f(s,a) = Q(s,a) + k \cdot e^{N(s,a)}$

□ $f(s,a) = k \cdot Q(s,a) - log(N(s,a))$

□ $f(s,a) = \frac{\epsilon}{k \cdot Q(s,a) \cdot N(s,a)}$

□ $f(s,a) = \frac{N(s,a) \cdot Q(s,a)}{\epsilon}$

○ None of the above

(d) [5 pts] Bob now switches to Q-learning, where he wants to perform approximate Q-learning for $Q(G, right)$. Assume he has w_i, which denotes the ith value of a weight vector w and $f_i(s,a)$, which denotes the value of the ith feature of the Q-state (s,a). He has the following values and observations:

State (s)	Action (a)	New State (s')	Reward (r)
G	Right	D	10
P	Up	Z	1

w_1	w_2	w_3
2	5	10

$f_1(G, Right)$	$f_2(G, Right)$	$f_3(G, Right)$
6	3	4

State	P	Z	D
Q(State, Up)	2	0	0
Q(State, Down)	5	7	0
Q(State, Left)	0	0	2
Q(State, Right)	0	-8	22

(i) [3 pts] What is the initial value of $Q(G, Right)$ based on the above weights and features?

(ii) [2 pts] What is the resulting weight vector after performing the first iteration of the weight update rule for going right on G? This time, assume a discount factor $\gamma = 0.5$ and learning rate $\alpha = 0.5$.