CS181 Practice Questions: More MDPs and Reinforcement Learning

1. **Q and V practice**

Consider an MDP with two states (A and B) and two actions (□ and △). You have the following transition and reward functions with $\gamma = 0.5$.

s	a	s'	$P(s' \mid a, s)$
A	□	A	0.5
A	□	B	0.5
A	△	A	1
A	△	B	0
B	□	A	1
B	□	B	0
B	△	A	1
B	△	B	0

s	a	$R(s, a)$
A	□	3
A	△	1
B	□	0
B	△	1

Complete the following table. (Hint: It's pretty easy to set up a spreadsheet to propagate the information for value iteration.)

k To Go	$Q(A, \square)$	$Q(A, \triangle)$	$Q(B, \square)$	$Q(B, \triangle)$	$\pi_k(A)$	$\pi_k(B)$	$V_k(A)$	$V_k(B)$
0	–	–	–	–	–	–	0	0
1								
2								
3								
4								
5								

2. **Stationary Policy**

In the previous problem, is the policy stationary? If not, why?

3. **Value Iteration for Survival (MIT 6.034)**

Consider the following MDP:

Start State	Action	Probability	Next State
hungry, tired	rested	1.0	hungry, rested
	hunt	0.8	hungry, tired
		0.2	full, tired
hungry, rested	rest	1.0	hungry, rested
	hunt	0.8	full, rested
		0.2	hungry, tired
full, tired	rest	1.0	hungry, rested
	hunt	0.8	hungry, tired
		0.2	full, tired
full, rested	rest	1.0	hungry, rested
	hunt	0.8	full, tired
		0.2	hungry, tired

In this world, you care about being rested and well fed. You can either rest or hunt. Your reward function is

State	Reward
hungry, tired	0
hungry, rested	1
full, tired	1
full, rested	2

(a) Given a discounting constant γ, what is the value of the (hungry, tired) state for a policy that always rests? Give your answer in terms of γ. (Hint: use geometric series.)

(b) Assume that $Q((\text{hungry}, \text{tired}), \text{rest}) = a$ and $Q((\text{hungry}, \text{tired}), \text{hunt}) = b$ and that $b > a$. Write an expression for $Q((\text{hungry}, \text{rested}), \text{hunt})$ in terms of these quantities, γ, and $V^\star(\text{full}, \text{rested})$.

4. **Q Learning**

Imagine that you are taking actions in an MDP that are selected according to the optimal policy. Explain whether or not Q-learning will only learn optimal Q-values.

5. **Q Learning**

Suppose you are standing on a linear board on which you can take action L or R to walk left or right, or S to sleep. If you walk, you have probability p_a that you actually walk to the next square, where $a \in \{L, R\}$. With probability $1 - p_a$, however, your cat distracts you and you stay on the same square. Staying on a square gives you reward r_i. Your learning rate is $\alpha = 0.5$ and $\gamma = 0.5$.
You are on square 1, you choose $a = L$ and receive $r = 4$. What is your updated value of $Q(1, L)$? Assume that your initial Q-values are all zero.

6. **Q learning**

Continuing the previous problem, in the next step, you are on square 0, you choose $a = R$, receive $r = 3$ and end up in $s = 1$. What is your updated value of $Q(0, R)$?

7. **Model Based Reinforcement Learning**

How do we compute the Maximum-Likelihood estimate of the expected reward from taking action a in state s?

8. **ϵ-Greedy**

Why do we generally use an ϵ-Greedy algorithm when choosing the current action during the Q-Learning algorithm? Describe how you would change the value of ϵ over time to encourage early exploration/learning and good decision making after the state space was explored.

9. **Convergence of Q-Learning**

Suppose we have a deterministic world where each state-action pair (s, a) is visited infinitely often. Consider an interval during which every such pair is visited. Suppose that the largest error in the approximation of $\hat{Q}_n$ after n iterations is e_n, i.e.

$$e_n = \max_{s,a} |\hat{Q}_n(s, a) - Q(s, a)|$$

Show that e_{n+1} is bounded above by γe_n, where γ is the usual discount factor.