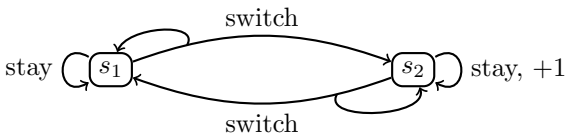
Question 1. (10 points)

Consider the following MDP. Assume that the reward is in the form $r(s,a)$, i.e., $r : S \times A \mapsto \mathbb{R}$. Set $\gamma = \frac{1}{2}$.



Suppose that you have seen the following sequence of states, actions, and rewards:

$$s_1, \text{switch}, s_2, \text{stay}, +1, s_2, \text{stay}, +1, s_2, \text{switch}, s_1, \text{stay}, s_1, \text{switch}, s_1, \text{switch}, s_1, \text{stay}, s_1, \text{switch}, s_2, \text{stay}, +1, s_2$$

1. (4 points) What is $\widehat{U}^{\pi}(s_i)$ calculated by the Every Visit Monte Carlo algorithm?

   Answer:

   For state s_1, we register the following set of 'rewards-to-go':

   $$\frac{1}{2}\cdot 1 + \frac{1}{4} + \frac{1}{512}, \qquad \frac{1}{32}, \qquad \frac{1}{16}, \qquad \frac{1}{8}, \qquad \frac{1}{4}, \qquad \frac{1}{2}.$$

   Average of those is

   $$\widehat{U}^{\pi}(s_1) = \frac{1}{6}\left(\frac{1}{2}\cdot 1 + \frac{1}{4} + \frac{1}{512} + \frac{1}{32} + \frac{1}{16} + \frac{1}{8} + \frac{1}{4} + \frac{1}{2}\right).$$

   In the same manner, we can compute

   $$\widehat{U}^{\pi}(s_2) = \frac{1}{4}\left[\left(1 + \frac{1}{2} + \frac{1}{256}\right) + \left(1 + \frac{1}{128}\right) + \frac{1}{64} + 1\right].$$

   Note:

   If you are consulting with AIMA or other literature, you might find the algorithm under the name Direct Utility Estimation.

2. (2 points) What is transition model P estimated by the Adaptive Dynamic Programming algorithm? (Note that this algorithm is covered only at the tutorials to show alternative approaches, and is not mandatory for the exam.)

   Answer:

   Using the MLE approach, the ADP estimates

   $$P(s_1 \mid s_1, \text{stay}) = \frac{2}{2}, \qquad P(s2 \mid s_1, \text{stay}) = \frac{0}{2},$$
   $$P(s_1 \mid s_1, \text{switch}) = \frac{2}{4}, \qquad P(s2 \mid s_1 \text{switch}) = \frac{2}{4},$$
   $$P(s_1 \mid s_2, \text{stay}) = \frac{0}{3}, \qquad P(s2 \mid s_2, \text{stay}) = \frac{3}{3},$$
   $$P(s_1 \mid s_2, \text{switch}) = \frac{1}{1}, \qquad P(s2 \mid s_2, \text{switch}) = \frac{0}{1}.$$

   It holds that the left and right columns sum to one.

3. (2 points) In the ADP estimates, some of the rare events might have zero probability, even though they are possible. Provide a solution in which the rare events that the algorithm misses during learning have a non-zero probability.

   Answer:

   We should include an assumption of a prior distribution on the set of probabilities. This way, each probability is non-zero, and for an infinite number of samples, the estimates converge to MLE estimates. Such an approach arises under several names - Bayes estimates, Laplace estimates, or pseudo-counts. Using a pseudo-count of 1, the results will be

$$P(s_1 \mid s_1, \mathsf{stay}) = \frac{2+1}{2+2}, \qquad P(s2 \mid s_1, \mathsf{stay}) = \frac{0+1}{2+2},$$
$$P(s_1 \mid s_1, \mathsf{switch}) = \frac{2+1}{4+2}, \qquad P(s2 \mid s_1 \mathsf{switch}) = \frac{2+1}{4+2},$$
$$P(s_1 \mid s_2, \mathsf{stay}) = \frac{0+1}{3+2}, \qquad P(s2 \mid s_2, \mathsf{stay}) = \frac{3+1}{3+2},$$
$$P(s_1 \mid s_2, \mathsf{switch}) = \frac{1+1}{1+2}, \qquad P(s2 \mid s_2, \mathsf{switch}) = \frac{0+1}{1+2}.$$

4. (2 points) What are state values estimated by a Temporal Difference learning agent after two steps? Assume that $\alpha = 0.1$ and all values are initialized to zero.

   Answer:

   After the first interaction, the update to $\widehat{U}^\pi(s_1)$ is made so that

   $$\widehat{U}^\pi_1(s_1) = \widehat{U}^\pi_0(s_1) + \alpha(r(s_1, \text{switch}) + \gamma \widehat{U}^\pi_0(s_2) - \widehat{U}^\pi_0(s_1)) = 0 + 0.1(0 + 0.5 \cdot 0 - 0) = 0.$$

   No update is made to $\widehat{U}^\pi(s_2)$, i.e., $\widehat{U}^\pi_0(s_2) = \widehat{U}^\pi_1(s_2) = 0$.

   After the second interaction, no update is made to $\widehat{U}^\pi(s_1)$, however,

   $$\widehat{U}^\pi_2(s_2) = \widehat{U}^\pi_1(s_2) + \alpha(r(s_2, \text{switch}) + \gamma \widehat{U}^\pi_1(s_2) - \widehat{U}^\pi_1(s_2)) = 0 + 0.1(1 + 0.5 \cdot 0 - 0) = 0.1.$$

[adapted from Richard Sutton's 609 course, see `http://www.incompleteideas.net/book/the-book-2nd.html`]

Question 2. (3 points)

Decide whether the following statement is true or false: *If a policy π is greedy with respect to its own value function U^π, then this policy is an optimal policy.* Explain your decision.

[adapted from Richard Sutton's 609 course, see `http://www.incompleteideas.net/book/the-book-2nd.html`]

Answer:

The statement is true. We can argue by the policy iteration algorithm, which is guaranteed to find an optimal policy. The condition needed for the algorithm to end is that the policy does not change in two consecutive steps - evaluation of a policy and successive calculation of the greedy policy. In other words, the optimal policy found by this algorithm is greedy with respect to its value function.

However, the reasoning is still not complete as the policy found by the policy iteration algorithm might differ from policy π. This discrepancy can be easily fixed by initializing the policy iteration algorithm with π. The algorithm terminates immediately, meaning that the policy is optimal.

Question 3. (5 points)

Do the following exploration/exploitation schemes fulfill the 'infinite exploration' and 'greedy in limit' conditions? Which lead to the convergence of Q-values in Q-learning and which lead to the convergence of Q-values in SARSA. Does anything change if we are interested in the convergence of policy? $n_{s,a}$ denotes the number of times when action a was taken in state s. n_s is defined similarly.

1. a random policy

2.
$$\pi(s) = \begin{cases} a, & \text{if } n_{s,a} \leq 100, \\ \arg\max_a Q(s,a), & \text{otherwise.} \end{cases}$$

3. ε-greedy policy with $\varepsilon = \frac{1}{n_s^2}$

4. ε-greedy policy with $\varepsilon = \frac{1{,}000}{999+n_s}$

5. ε-greedy policy with $\varepsilon = \frac{1}{\sqrt{n_s}}$

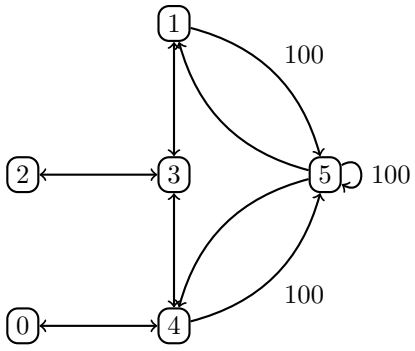
Answer:

Convergence of Q-values in Q-learning is guaranteed when the 'infinite exploration' condition is met. For SARSA, both GLIE conditions are needed. For both Q-learning and SARSA, the policy converges when both GLIE conditions are met. We can, therefore, summarize the results in the following table.

	GL	IE	Q (Q-learning)	Q (SARSA)	π (Q-learning)	π (SARSA)
1.	NO	YES	YES	NO	NO	NO
2.	YES	NO	NO	NO	NO	NO
3.	YES	NO	NO	NO	NO	NO
4.	YES	YES	YES	YES	YES	YES
5.	YES	YES	YES	YES	YES	YES

Question 4. (5 points)

Consider the following MDP with $\gamma = 0.8$, $r(5) = 100$, $r(\cdot) = 0$.



The initial matrix of Q-values is

$$\widehat{Q}(s,a) = \begin{bmatrix} - & - & - & - & 0 & - \\ - & - & - & 0 & - & 0 \\ - & - & - & 0 & - & - \\ - & 0 & 0 & - & 0 & - \\ 0 & - & - & 0 & - & 0 \\ - & 0 & - & - & 0 & 0 \end{bmatrix}.$$

Consider path $1 - 5 - 1 - 3$ and constant learning rate $\alpha = 0.1$. Show changes in Q values after the agent-environment interaction for the Q-learning algorithm.

[adapted from Richard Sutton's 609 course, see `http://www.incompleteideas.net/book/the-book-2nd.html`]

Answer:

After the first interaction, only value $Q(1,5)$ changes:

$$\widehat{Q}_1(1,5) = \widehat{Q}_0(1,5) + \alpha \cdot \left(r(5) + \gamma \cdot \max_{a'} \widehat{Q}_0(5,a') - \widehat{Q}_0(1,5) \right) = 0 + 0.1(100 + 0.8 \cdot \max\{0,0,0\} - 0) = 10.$$

The remaining interactions are analogous.

$$\widehat{Q}_2(5,1) = \widehat{Q}_1(5,1) + \alpha \cdot \left(r(1) + \gamma \cdot \max_{a'} \widehat{Q}_1(1,a') - \widehat{Q}_1(5,1) \right) = 0 + 0.1(0 + 0.8 \cdot \max\{10\} - 0) = \frac{8}{10},$$
$$\widehat{Q}_3(1,3) = \widehat{Q}_2(1,3) + \alpha \cdot \left(r(3) + \gamma \cdot \max_{a'} \widehat{Q}_2(3,a') - \widehat{Q}_2(1,3) \right) = 0 + 0.1(0 + 0.8 \cdot \max\{0,0,0\} - 0) = 0.$$

Question 5. (10 points)

Consider an active reinforcement learning algorithm implemented by SARSA or Q-learning.

1. (2 points) Unlike the temporal difference learning, SARSA and Q-learning algorithms learn Q values instead of U. Why is U not enough?

   Answer:

   The agent needs to know which action leads to the best reward. U requires knowledge of transition probabilities P. Therefore, the agent learns Q to avoid learning the transition model.

2. (3 points) Explain why those algorithms need to balance exploration vs. exploitation. What those terms mean, and which of those is preferred early in the learning.

   Answer:

   Exploration means probing suboptimal actions to find their values. Exploitation is the opposite; the agent greedily tries to maximize its reward. Both cannot be achieved at the same time. Therefore, the agent needs to start exploring so that it does not miss a good action due to a series of unfavorable events. On the contrary, once the values are reliably established, the agent can start to exploit to maximize its reward.

3. (2 points) SARSA and Q-learning are guaranteed to converge to an optimal policy if both:

   - convergence criteria for learning rate α known from TD-learning are met, and
   - convergence criteria on the explore-exploit policy are met.

   What are those criteria placed on the explore-exploit policy?

   Answer:

   The criteria placed on the explore-exploit policy are GLIE. Agent's policy must eventually become greedy in the limit with $t \to \infty$. The second assumption assumes that if a state is visited ∞-many times, then each action is tried in this state ∞-many times.

4. (1 point) Provide an example of an explore-exploit policy that guarantees policy convergence for SARSA and Q-learning.

   Answer:

   The GLIE conditions are fulfilled, for example, by the ε-greedy policy with

   $$\varepsilon(n_s) \in \mathcal{O}\left(\frac{1}{n_s}\right).$$

5. (2 points) Will one of the algorithms learn Q-values even if one of the conditions is not met? If yes, which and why, if not, explain.

   Answer:

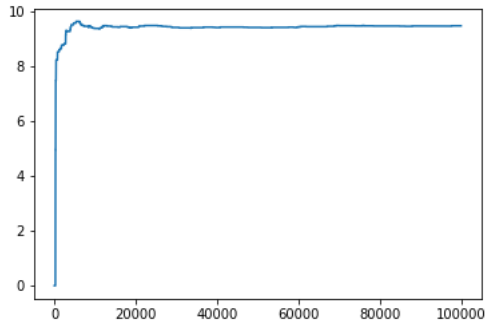
   SARSA, in general, as an on-policy algorithm, cannot learn the true values if one of the GLIE conditions is not met. However, the Q-learning can learn as an off-policy algorithm the Q-values even if the policy does not fulfill the 'greedy in the limit' condition. The algorithm won't act optimally, it will follow a suboptimal policy, but the Q-values will converge to the optimal as long as the 'infinite exploration' condition and learning rate convergence conditions are met.

Question 6. (10 points)

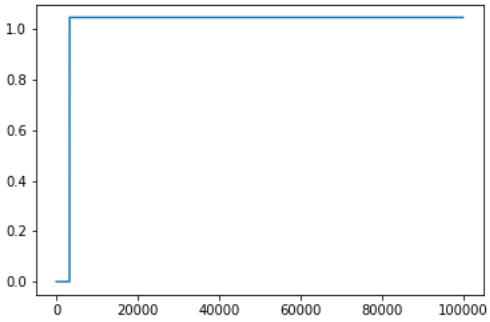
Consider an active reinforcement learning algorithm. You are not told whether it is an instance of SARSA or Q-learning. The implementation met all convergence criteria. All plots shown below are related to the same state-action pair Q-value, i.e., $\widehat{Q}(s, a)$. The action a is **suboptimal** in state s. The used explore-exploit policy was the ε-greedy policy, i.e., with probability ε a random action is selected; otherwise, the agent behaves greedily.

Now, consider four different situations of learning Q-values over 100 000 episodes.

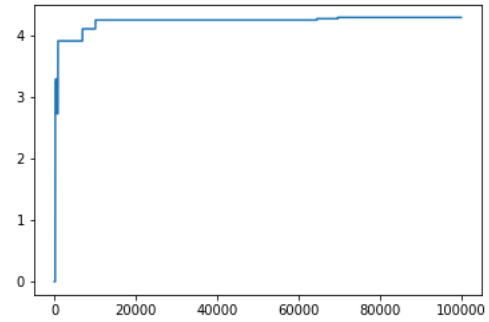
1.



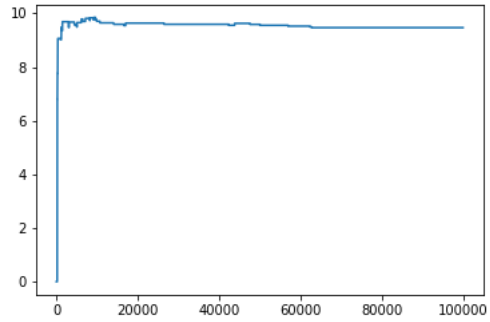
2.



3.

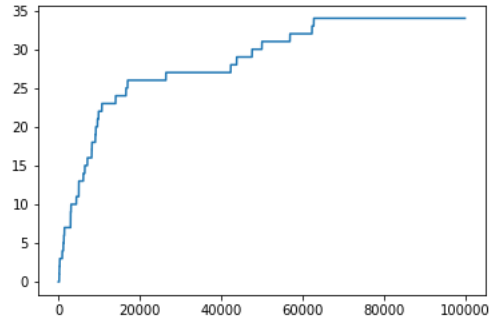


4.

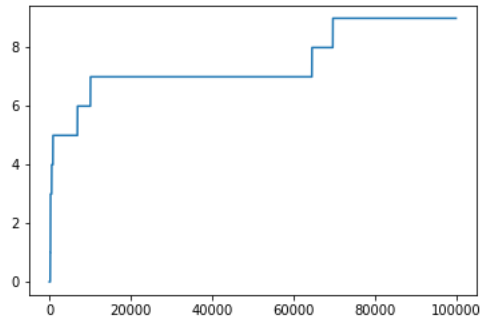


1. (4 points) Four plots below show how many times action a was selected by the agent in state s. For example, point $(1000, 6)$ means that the action a was selected 6 times in state s over the first 1000 episodes.

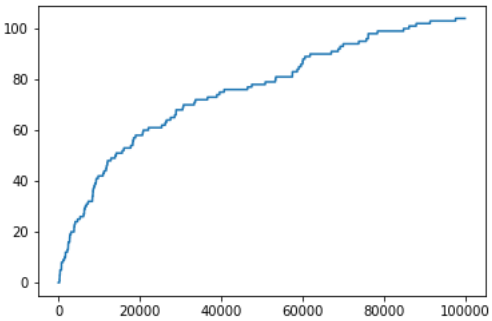
a.



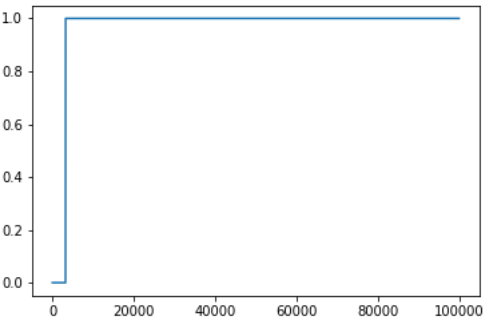
b.



c.



d.



Match figures a-d to figures 1-4. Explain your decision.

Answer:

For both SARSA and Q-learning, value $\widehat{Q}(s,a)$ changes only when state action pair s,a is visited. Therefore, d matches to 2, b matches to 3, a matches to 4, and c matches to 1.

2. (2 points) The ε was set as a function of the number of visits of state s. Relate the following four functions to the figures 1-4.

$$\text{i.}\quad \varepsilon(n_s)=\frac{8}{7+n_s} \qquad \text{ii.}\quad \varepsilon(n_s)=\frac{3}{2+n_s} \qquad \text{iii.}\quad \varepsilon(n_s)=\frac{100}{99+n_s} \qquad \text{iv.}\quad \varepsilon(n_s)=\frac{1000}{999+n_s}$$

Match those policies to figures 1-4. Explain your choice.

Answer:

The agent explores the least in case 2, which corresponds to function ii. We know that action a is suboptimal; therefore, under true Q-values it is chosen only when the agent decides to use random action. Therefore, by sorting by the ε value, we can conclude that case 3 corresponds to i, case 4 to iii, and case 1 to iv.

3. (1 point) Why should agents use different epsilon for different states.

Answer:

The reasoning is the same as in the case of the learning rate. The number of visits is different for different states. Slow learning in states that are visited rarely should not influence the states that are visited often.

4. (2 points) Decide whether the learning algorithm used was SARSA or Q-learning. Explain your decision.

Answer:

From the plots, it is more likely that the algorithm used was Q-learning. Figures 1 and 4 show very little difference even though in figure 4, the agent decides to use random action more often. This indicates that the algorithm is off-policy. SARSA would be influenced more by random decisions. However, as we do not know the environment, we cannot state this for sure. *In fact, Q-learning was used.*

5. (1 point) What is $Q(s,a)$? Explain your answer.

Answer:

$Q(s,a)$ is very likely between 9 and 10, as figures 1 and 4 indicate. Figures 2 and 3 show insufficient learning. Figures 1 and 4 agree on this number.