← Control with Approximation
Graded Quiz • 40 min

Due Nov 29, 11:59 PM MST

Control with Approximation

TOTAL POINTS 12

1. Which of the following are true? (Select all that apply)

1 point

□ In the tabular case, there is generalization across states, i.e., updates to the value function of one state influences the value function of other states.

□ When using state aggregation or coarse coding, updating the value of one state does not affect the values of other states.

□ In the tabular case, updating the value of one state does not affect the values of other states.

□ When using state aggregation or coarse coding, there is generalization across states.

2. To turn the update of Expected Sarsa algorithm to the update of Q-learning, one must:

1 point

○ Use a neural network to approximate the action-value function.

○ Behave greedily with respect to the action-value function.

○ Expected Sarsa cannot be adapted to represent Q-learning.

○ Use the maximum over all the actions instead of the expectation in the update function.

3. Which of the following are true:

1 point

□ When using function approximation and with discrete actions, there is no straightforward way to turn Sarsa into Expected Sarsa.

□ Exploration is not a problem when using function approximation because learning generalizes across states and actions.

actions.

□ When learning value functions in the mountain car domain and using tiile coding, where all the rewards are -1 except when the terminal state is reached, initializing the weights of a linear function approximation to zero is an example of optimistic initialization.

□ When using function approximation, Q-learning will always result in better performance than Sarsa and Expected Sarsa.

4. Which of the following statements about discounted return and differential return algorithms are true:

1 point

□ Discounted return algorithms can diverge if the average reward term is slightly larger or slightly lower than the true value.

□ Algorithms that maximize the discounted return can suffer from an exponentially large variance when using a large discount factor.

□ There is a set of Bellman equations for the value function corresponding to the discounted return. However, there is no set of Bellman equations for differential value functions.

□ The performance of discounted return algorithms can suffer because the optimal value of gamma is problem dependent.

5. Imagine we have 7 state features and 6 actions and we want to use linear function approximation to compute the action-value function. If we use feature stacking (as explained in Week 3), how many features would we need in total to approximate the action-value function of all the different actions?

1 point

○ 67

○ 42

○ 76

○ 13

6. Imagine we are approximating the action-value function of three different actions (red, green, and blue) using three state features, resulting in the following weights and features:

1 point

$$\begin{bmatrix} x_0(s) \\ x_1(s) \end{bmatrix} \Big\} a_0 \qquad \begin{bmatrix} w_0(s) \\ w_1(s) \end{bmatrix} \Big\} a_0$$

$$\mathbf{X}(s) = \begin{bmatrix} x_0(s) \\ x_1(s) \\ x_2(s) \end{bmatrix} \quad \mathbf{X}(s, a) = \begin{bmatrix} \color{red}{x_2(s)} \\ \color{green}{x_0(s)} \\ \color{green}{x_1(s)} \\ \color{green}{x_2(s)} \\ \color{blue}{x_0(s)} \\ \color{blue}{x_1(s)} \\ \color{blue}{x_2(s)} \end{bmatrix} \begin{matrix} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{green}{a_1} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{blue}{a_2} \end{matrix} \quad \mathbf{w}(s) = \begin{bmatrix} \color{red}{w_2(s)} \\ \color{green}{w_0(s)} \\ \color{green}{w_1(s)} \\ \color{green}{w_2(s)} \\ \color{blue}{w_0(s)} \\ \color{blue}{w_1(s)} \\ \color{blue}{w_2(s)} \end{bmatrix} \begin{matrix} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{green}{a_1} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{blue}{a_2} \end{matrix}$$

Which of these statements is true about our function approximation scheme?

□ The green action will never be selected.

□ The action-value of the green action will be exactly the same as the action-value of the red action for any state.

□ The action-value of the red action will be exactly the same as the action-value of the blue action.

□ There will not be any generalization across states.

7. Consider the following feature and weight vectors corresponding to three state features and three actions (red, green, and blue):

1 point

$$\mathbf{X}(s) = \begin{bmatrix} x_0(s) \\ x_1(s) \\ x_2(s) \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.2 \\ 0.3 \end{bmatrix} \qquad \mathbf{X}(s, a) = \begin{bmatrix} \color{red}{x_0(s)} \\ \color{red}{x_1(s)} \\ \color{red}{x_2(s)} \\ \color{green}{x_0(s)} \\ \color{green}{x_1(s)} \\ \color{green}{x_2(s)} \\ \color{blue}{x_0(s)} \\ \color{blue}{x_1(s)} \\ \color{blue}{x_2(s)} \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{red}{a_0} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{green}{a_1} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \color{blue}{a_2} \end{matrix}$$

$$\begin{bmatrix} \color{red}{w_0(s)} \\ \color{red}{w_1(s)} \end{bmatrix} \quad \begin{bmatrix} \color{red}{30} \\ \color{red}{60} \end{bmatrix}$$

$$\mathbf{w}(s) = \begin{bmatrix} w_2(s) \\ w_0(s) \\ w_1(s) \\ w_2(s) \\ w_0(s) \\ w_1(s) \\ w_2(s) \end{bmatrix} = \begin{bmatrix} 50 \\ 4 \\ 4 \\ 4 \\ 31 \\ 14 \\ 15 \end{bmatrix}$$

What is the approximate action-value for the red action?

○ 4

○ 160

○ 42

○ 16

8. Which of the following statements about epsilon-greedy policies and optimistic initialization are true:

1 point

□ Optimistic initialization results in a more systematic exploration in the tabular case because the agent takes actions it has not taken as often in a state.

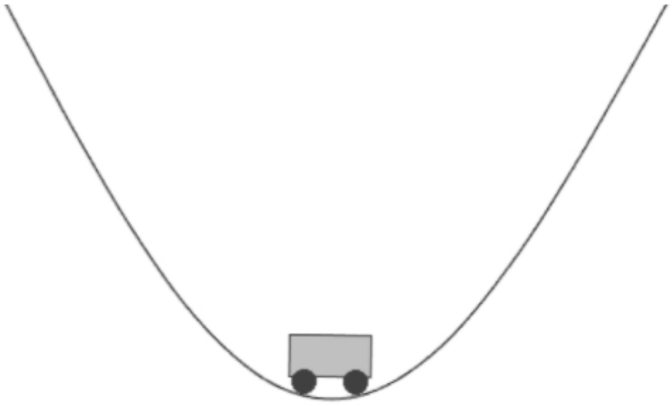
□ Implementing optimistic initialization is always straightforward and simple to implement regardless of the function approximation technique.

□ Optimistic initialization is always preferred over using an epsilon-greedy policy.

□ Epsilon-greedy can be easily combined with any function approximation technique because it only needs to be able to query for the approximate action-values, without needing to know how they are initialized or computed.

9. Consider the mountain car environment where all the rewards are -1 after every action until reaching the flag on top of the right hill:

1 point



Which of the following are examples of optimistic initialization when using linear function approximation?

□ Initialize all the weights so that the action-value function is zero everywhere.

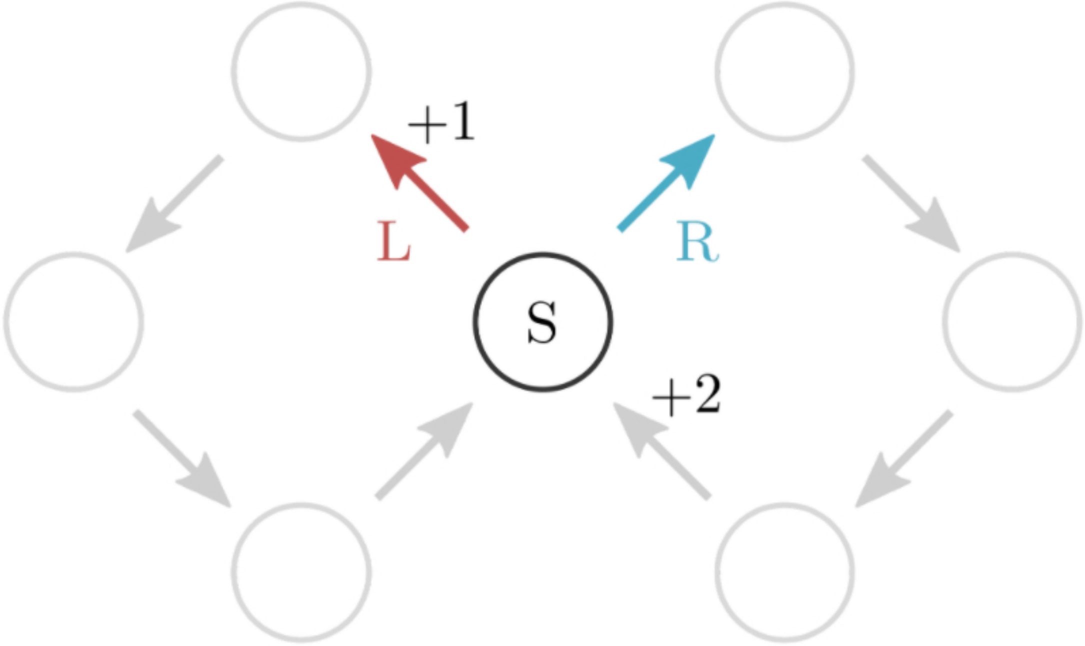
□ Initialize all the weights so that the action-value function is -1 everywhere.

□ Initialize all the weights so that the action-value function is 10 everywhere.

□ Initialize all the weights at random using a uniform distribution between -5 and 5.

10. Consider the following MDP where the Red action (R) at state S results in an immediate reward of +1 and 0 afterwards, and the Blue action (L) results in 0 immediate reward, but a final reward of +2 when returning to state S.

1 point



In this MDP, what is the optimal policy at state S when using discounted return with a value of gamma of 0.5? When using discounted return with a value of gamma of 0.9? When using average reward? (Hint: You can modify the equations used for the task from lesson 3 of this module to get the solution)

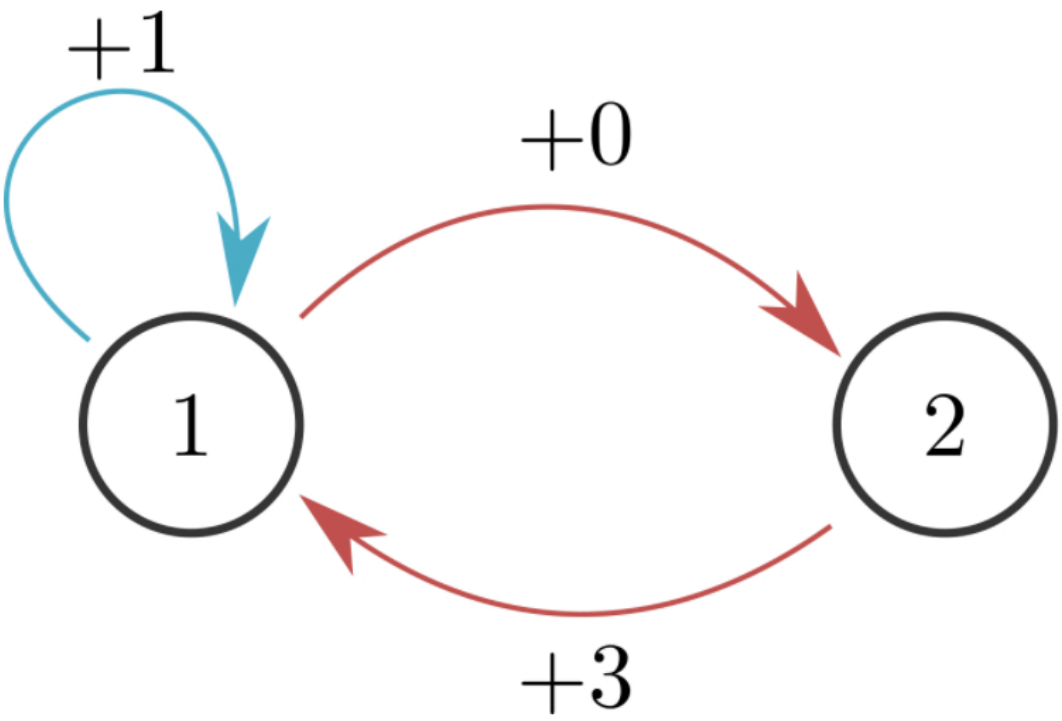
○ The optimal policy is: gamma = 0.5, blue action; gamma = 0.9, blue action; average reward, red action.

○ The optimal policy is: gamma = 0.5, red action; gamma = 0.9, blue action; average reward, blue action.

○ The optimal policy is: gamma = 0.5, blue action; gamma = 0.9, red action; average reward, blue action.

○ The optimal policy is: gamma = 0.5, blue action; gamma = 0.9, red action; average reward, red action.

11. Consider the following MDP with two states. State 1 has two actions: staying in the same state (blue) and switching to the other state (red). State 2 has a single (red) action leading to State 1. The rewards are listed next to each arrow/transition. 1 point



If the agent is following a policy that takes the blue and red actions with equal probability, what is the average reward? (Hint: the state visitation probability is ⅔ for state 1 and ⅓ for state 2, the target policy probability \pi is the random policy, and all the transitions p are deterministic) Recall the formula is:

$$r(\pi) = \sum_{s} \mu_{\pi}(s) \sum_{a} \pi(a \mid s) \sum_{s',r} p(s', r \mid s, a) r$$

○ Average reward = 3/4

○ Average reward = 1

○ Average reward = 2

○ Average reward = 4/3

12. For which of the following tasks is the average reward setting preferable than the discounted return setting if we are interested in maximizing the total amount of reward?

1 point

□ The job scheduling task from Section 10.3 of Sutton and Barto's RL textbook. The agent manages how jobs are scheduled into three different servers according to the priority of each job. When the agent accepts a job, the job runs in one of the servers and the agent gets a positive reward equal to the priority of the job, whereas when the agent rejects a job, the job returns to the queue and the agent receives a negative reward proportional to the priority of the job. The servers become available as they finish their jobs. Jobs are continually added to the queue and the agent accepts or rejects them.

□ An Atari 2600 game where the agent can keep playing until it loses all its lives, which are a finite amount.

□ The episodic mountain car environment.

□ The nearsighted MDP from Week 3. The agent starts at an initial state S that splits into two paths, each leading to a ring that leads back to S after some time. If the agent chooses the left ring, it observes and immediate reward of +1 followed by 0 rewards, whereas choosing the right ring results on immediate reward of 0, but a final reward of +2 when returning to state S.

□ I, **Dhawal Gupta**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Learn more about Coursera's Honor Code

Save

Submit