Homework 3

ELEN E6885: Introduction to Reinforcement Learning

Due: November 14, 2019

Problem 1 (**n-Step Return, 15 Points**)

The expected value of all n-step returns is guaranteed to improve in a certain way over the current value function as an approximation to the true value function. Prove the following *error reduction property* of n-step returns

$$\max_s \left| E_\pi \left[G_t^{(n)} \middle| S_t = s \right] - V_\pi(s) \right| \leq \gamma^n \max_s \left| V_t(s) - V_\pi(s) \right|,$$

where $G_t^{(n)}$ is n-step return at time t.

Problem 2 (**On-line vs. Off-line Update, 35 Points**)

To distinguish two different ways of making updates in reinforcement learning algorithms (i.e., on-line and off-line updating), answer the following questions.

1. [5 pts] What is the difference between on-line and off-line updating methods?

2. [5 pts] In all following questions, consider an episode: $A, +1, B, +2, A, +1, T$ from an *undiscounted* MDP, where A, B are two non-terminal states, T is the terminal state and the number after each state is an immediate reward. Using a learning rate of $\alpha = 0.1$, and assuming initial state values of 0. What is the total update to $V(A)$ *on-line* every-visit constant-α Monte Carlo method makes after the episode finishes? What about *off-line* every-visit constant-α Monte Carlo method?

3. [5 pts] What is the total update to $V(A)$ *on-line* TD(0) method makes after the episode finishes? What about *off-line* TD(0) method?

4. [10 pts] Assume $\lambda = 0.5$. What is the total update to $V(A)$ *on-line* forward-view TD(λ) method makes after the episode finishes? What about *off-line* forward-view TD(λ) method?

5. [10 pts] Assume $\lambda = 0.5$. What is the total update to $V(A)$ *on-line* backward-view TD(λ) method makes after the episode finishes? What about *off-line* backward-view TD(λ) method?

Problem 3 (**Forward vs. Backward view of TD(λ), 25 Points**)

We know that when using off-line updates, forward-view and backward-view TD(λ) are equivalent, i.e., the total update to a value function at the end of an episode is the same. In other words, the off-line TD(λ) (i.e., backward view) exactly matches the off-line λ-return algorithm (i.e., forward view).

1. [20 pts] As a special case, follow the steps below to prove that off-line TD(1) (i.e., backward view) and off-line every-visit constant-α Monte Carlo method (i.e., forward view) are equivalent.

   a. [5 pts] Consider an episode which terminates after T steps. Prove that the original statement is equivalent to show that for any state s,

   $$\sum_{t=0}^{T-1} \alpha \delta_t E_t(s) = \sum_{t=0}^{T-1} \alpha \left(G_t - V(S_t)\right) \mathbf{1}\left(S_t = s\right), \tag{1}$$

   where $\mathbf{1}(\cdot)$ is the indicator function, which equals to 1 if $S_t = s$ and 0 otherwise.

   b. [5 pts] For any $0 \le t \le T-1$ and state s, prove that the accumulating eligibility trace can be written explicitly as

   $$E_t(s) = \sum_{k=0}^{t} \gamma^{t-k} \cdot \mathbf{1}\left(S_k = s\right). \tag{2}$$

   c. [10 pts] Prove that the equality in (1) holds by plugging (2) into the left-hand-side of (1).

2. [5 pts] Is it possible to construct a version of on-line TD(λ) method (i.e., backward view) that matches the on-line λ-return algorithm (i.e., forward view) exactly? Explain your answer.

Problem 4 (**Linear Function Approximation, 25 Points**)

Consider the small corridor gridworld shown in Fig. 1 below. S and G represents the start and goal (terminal) state, respectively. In each of the two non-terminal states, there are only two actions, *right* and *left*. These actions have their usual consequences in the start state (left causes no movement in the start state). But in the middle state they are reversed, so that *right* moves to the left and *left* moves to the right. The reward is -1 per step as usual. We approximate the action-value function using two features $\mathrm{x}_1(s,a) = \mathbf{1}(a = \textit{right})$ and $\mathrm{x}_2(s,a) = \mathbf{1}(a = \textit{left})$ for all state-action pair (s,a). We sample an episode till the goal by sequentially taking actions *right*, *right*, *right*, *left*. Assume the experiment is *undiscounted*.

1. [5 pts] Approximate the action-value function by a linear combination of these features with two parameters: $\hat{q}(s,a,\mathbf{w}) = \mathrm{x}_1(s,a)w_1 + \mathrm{x}_2(s,a)w_2$. If $w_1 = w_2 = 1$, calculate the λ-return q_t^λ corresponding to this episode for $\lambda = 0.5$.

2. [5 pts] Using the forward-view TD(λ) algorithm with off-line updates and our linear function approximator, what are the sequence of updates to weight w_1? What is the total update to weight w_1? Use $\lambda = 0.5, \gamma = 1, \alpha = 0.5$ and start with $w_1 = w_2 = 1$.

3. [5 pts] Define the TD(λ) accumulating eligibility trace $\mathbf{e}_t$ when using linear value function approximation. Write down the sequence of eligibility traces corresponding to *right* action, using $\lambda = 0.5, \gamma = 1$.

4. [5 pts] Using the backward-view TD(λ) algorithm with off-line updates and our linear function approximator, what are the sequence of updates to weight w_1? What is the total update to weight w_1? Use $\lambda = 0.5, \gamma = 1, \alpha = 0.5$ and start with $w_1 = w_2 = 1$.

5. [5 pts] Based on your results in previous questions, when using off-line updates and linear function approximation, are forward-view and backward-view TD(λ) equivalent to each other?

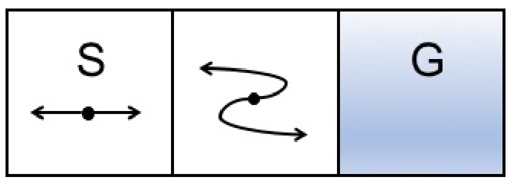

Figure 1: Small corridor gridworld