Chapter 10 solutions **Hanna Gábor**

1. *We have not explicitly considered or given pseudocode for any Monte Carlo methods in this chapter. What would they be like? Why is it reasonable not to give pseudocode for them? How would they perform on the Mountain Car task?*

   The Monte Carlo is basically n-step Sarsa with $n = T$. I expect it to perform poorly on the Mountain Car example: until it doesn't reach the goal, it doesn't learn anything, so the first episode might be really long. As opposed to this, n-step Sarsa tries new actions after it sees that the previous actions didn't lead to the end of the episode, so it will get to the goal line eventually.

2. *Give pseudocode for semi-gradient one-step Expected Sarsa for control.*

Algorithm 1 Semi-gradient expected Sarsa for control

```
Inputs: a differentiable action-value function parametrization $\hat{q} : \mathcal{S} \times \mathcal{A} \times \mathbb{R}^d \to \mathbb{R}$
Algorithm parameters: steps size $\alpha > 0, \epsilon > 0$
Initialize value-function weights $w \in \mathbb{R}^d$ arbitrarily
for each episode do
    Initialize $S_0$ non-terminal state.
    for $t = 0, 1, 2 \ldots$ do
        Take action $A_t$ according to the $\epsilon$-greedy policy w.r.t. $\hat{q}(S_0, \cdot, w)$
        Observe reward $R_{t+1}$ and next state $S_{t+1}$.
        if $S_{t+1}$ is terminal then
            $w \leftarrow w + \alpha R_{t+1} \nabla \hat{q}(S_t, A_t, w)$
            go to next episode
        else
            $w \leftarrow w + \alpha\Big(R_{t+1} + \sum_a \pi(a|S_{t+1})\hat{q}(S_{t+1}, a, w) - \hat{q}(S_t, A_t, w)\Big)\nabla \hat{q}(S_t, A_t, w)$, where
            $\pi$ is the $\epsilon$-greedy policy w.r.t. $\hat{q}(S_0, \cdot, w)$
            $S_t \leftarrow S_{t+1}$
        end if
    end for
end for
```

3. *Why do the results shown in Figure 10.4 have higher standard errors at large n than at small n?*

   If n is large, it's less predictable where we will end up in n steps. This causes our estimates to have a bigger standard deviation. That in turn, will cause the steps to be more varied among different runs.

4. *Give pseudocode for a differential version of semi-gradient Q-learning* See the algorithm on the next page. I used the average of the seen rewards as $\bar{R}$.

5. *What equations are needed (beyond 10.10) to specify the differential version of TD(0)?*

$$w_{t+1} = w_t + \alpha \delta_t \nabla \hat{v}(S_t, w_t)$$

Algorithm 2 Differential version of semi-gradient Q-learning

```
Inputs: a differentiable action-value function parametrization $\hat{q} : \mathcal{S} \times \mathcal{A} \times \mathbb{R}^d \to \mathbb{R}$
Algorithm parameters: steps size $\alpha > 0, \epsilon > 0$
Initialize value-function weights $w \in \mathbb{R}^d$ arbitrarily.
$\bar{R} \leftarrow 0$
$visits \leftarrow 0$
for each episode do
  Initialize $S_0$ non-terminal state.
  for $t = 0, 1, 2 \ldots$ do
    Choose action $A_t$ according to the $\epsilon$-greedy policy w.r.t. $\hat{q}$
    Observe reward $R_{t+1}$ and next state $S_{t+1}$.
    $w \leftarrow w + \alpha(R_{t+1} - \bar{R} + \max_a(\hat{q}(S_{t+1}, a, w)) - \hat{q}(S_t, A_t, w))\nabla\hat{q}(S_t, A_t, w)$
    $\bar{R} \leftarrow \frac{visits}{visits+1}(\bar{R} + R_{t+1})$
    $visits \leftarrow visits + 1$
    if $S_{t+1}$ is terminal then
      Go to next episode.
    end if
  end for
end for
```

6. *Suppose there is an MDP that under any policy produces the deterministic sequence of rewards +1, 0, +1, 0, +1, 0,... going on forever. Technically, this violates ergodicity; there is no stationary limiting distribution μ_π and the limit (10.7) does not exist. Nevertheless, the average reward (10.6) is well defined. What is it? Now consider two states in this MDP. From A, the reward sequence is exactly as described above, starting with a +1, whereas, from B, the reward sequence starts with a 0 and then continues with +1, 0, +1, 0,.... We would like to compute the differential values of A and B. Unfortunately, the differential return (10.9) is not well defined when starting from these states as the implicit limit does not exist. To repair this, one could alternatively define the differential value of a state as*

$$v_\pi(s) = \lim_{\gamma \to 1} \lim_{h \to \infty} \sum_{t=0}^{h} \gamma^t (\mathbb{E}(R_{t+1} | S_0 = s) - r(\pi)).$$

Under this definition, what are the differential values of states A and B?

The average reward is 0.5.

$$\begin{aligned}
\lim_{h\to\infty}\sum_{t=0}^{h}\gamma^t(\mathbb{E}(R_{t+1}|S_0=A)-r(\pi)) &= \sum_{t=0}^{\infty}\gamma^{2t}-r(\pi)\sum_{t=0}^{\infty}\gamma^t = \frac{1}{1-\gamma^2}-\frac{0.5}{1-\gamma}\\
v_\pi(A) &= \lim_{\gamma\to 1}\frac{1}{1-\gamma^2}-\frac{0.5}{1-\gamma} = \lim_{\gamma\to 1}\frac{1-0.5(1+\gamma)}{1-\gamma^2}\\
&= \lim_{\gamma\to 1}\frac{0.5(1-\gamma)}{1-\gamma^2} = \lim_{\gamma\to 1}0.5\frac{1}{1+\gamma} = 0.25\\
\lim_{h\to\infty}\sum_{t=0}^{h}\gamma^t(\mathbb{E}(R_{t+1}|S_0=B)-r(\pi)) &= \sum_{t=0}^{\infty}\gamma^{2t+1}-r(\pi)\sum_{t=0}^{\infty}\gamma^t = \frac{1}{1-\gamma^2}-\frac{0.5}{1-\gamma}\\
v_\pi(B) &= \lim_{\gamma\to 1}\frac{\gamma}{1-\gamma^2}-\frac{0.5}{1-\gamma} = \lim_{\gamma\to 1}\frac{\gamma-0.5(1+\gamma)}{1-\gamma^2}\\
&= \lim_{\gamma\to 1}\frac{0.5(\gamma-1)}{1-\gamma^2} = \lim_{\gamma\to 1}-0.5\frac{1}{1+\gamma} = -0.25
\end{aligned}$$

7. *Consider a Markov reward process consisting of a ring of three states A, B, and C, with state transitions going deterministically around the ring. A reward of +1 is received upon arrival in A and otherwise the reward is 0. What are the differential values of the three states, using (10.13)?*

$r_\pi = \frac{1}{3}$ in this case.

$$\begin{aligned}
\lim_{h\to\infty}\sum_{t=0}^{h}\gamma^t(\mathbb{E}(R_{t+1}|S_0=A)-r(\pi)) &= \gamma^2\sum_{t=0}^{\infty}(\gamma^3)^t - r(\pi)\sum_{t=0}^{\infty}\gamma^t \\
&= \frac{\gamma^2}{1-\gamma^3} - \frac{1}{3(1-\gamma)} \\
v_\pi(A) &= \lim_{\gamma\to 1}\frac{\gamma^2}{1-\gamma^3} - \frac{1}{3(1-\gamma)} \\
&= \lim_{\gamma\to 1}\frac{3\gamma^2-(1+\gamma+\gamma^2)}{3(1-\gamma^3)} = \lim_{\gamma\to 1}\frac{2\gamma^2-\gamma-1}{3(1-\gamma^3)} \\
&= \lim_{\gamma\to 1}\frac{-2\gamma-1}{3(1+\gamma+\gamma^2)} = -\frac{1}{3}
\end{aligned}$$

$$\begin{aligned}
\lim_{h\to\infty}\sum_{t=0}^{h}\gamma^t(\mathbb{E}(R_{t+1}|S_0=B)-r(\pi)) &= \gamma\sum_{t=0}^{\infty}(\gamma^3)^t - r(\pi)\sum_{t=0}^{\infty}\gamma^t \\
&= \frac{\gamma}{1-\gamma^3} - \frac{1}{3(1-\gamma)} \\
v_\pi(B) &= \lim_{\gamma\to 1}\frac{\gamma}{1-\gamma^3} - \frac{1}{3(1-\gamma)} \\
&= \lim_{\gamma\to 1}\frac{3\gamma-(1+\gamma+\gamma^2)}{3(1-\gamma^3)} = \lim_{\gamma\to 1}\frac{-\gamma^2+2\gamma-1}{3(1-\gamma^3)} \\
&= \lim_{\gamma\to 1}\frac{\gamma-1}{3(1+\gamma+\gamma^2)} = 0
\end{aligned}$$

$$\begin{aligned}
\lim_{h\to\infty}\sum_{t=0}^{h}\gamma^t(\mathbb{E}(R_{t+1}|S_0=C)-r(\pi)) &= \sum_{t=0}^{\infty}(\gamma^3)^t - r(\pi)\sum_{t=0}^{\infty}\gamma^t \\
&= \frac{1}{1-\gamma^3} - \frac{1}{3(1-\gamma)} \\
v_\pi(C) &= \lim_{\gamma\to 1}\frac{1}{1-\gamma^3} - \frac{1}{3(1-\gamma)} \\
&= \lim_{\gamma\to 1}\frac{3-(1+\gamma+\gamma^2)}{3(1-\gamma^3)} = \lim_{\gamma\to 1}\frac{-\gamma^2-\gamma+2}{3(1-\gamma^3)} \\
&= \lim_{\gamma\to 1}\frac{\gamma+2}{3(1+\gamma+\gamma^2)} = \frac{1}{3}
\end{aligned}$$

8. *The pseudocode in the box on page 251 updates $\bar{R}_t$ using δ_t as an error rather than simply $R_{t+1}-\bar{R}_t$. Both errors work, but using δ_t is better. To see why, consider the ring MRP of three states from Exercise 10.7. The estimate of the average reward should tend towards its true value of $\frac{1}{3}$. Suppose it was already there and was held stuck there. What would the sequence of $R_{t+1}-\bar{R}_t$ errors be? What would the sequence of δ_t errors be (using Equation 10.10)? Which error sequence would produce a more stable estimate of the average reward if the estimate were allowed to change in response to the errors? Why?*

   The $R_{t+1}-\bar{R}_t$ sequence would be $\frac{2}{3}, -\frac{1}{3}, -\frac{1}{3}, \frac{2}{3}, -\frac{1}{3}, -\frac{1}{3}, \ldots$. The δ_t error would always be 0. The δ_t errors would produce a more stable estimate, since they don't change the estimate once it's reached, while the $R_{t+1} - \bar{R}_t$ error is never 0, so the estimate would continue to fluctuate.

9. *In the differential semi-gradient n-step Sarsa algorithm, the step-size parameter on the average reward, β, needs to be quite small so that $\bar{R}$ becomes a good long-term estimate of the average reward. Unfortunately, $\bar{R}$ will then be biased by its initial value for many steps, which may make learning inefficient. Alternatively, one could use a sample average of the observed rewards for $\bar{R}$. That would initially adapt rapidly but in the long run would also adapt slowly. As the policy slowly changed, $\bar{R}$ would also change; the potential for such long-term nonstationarity makes sample-average methods ill-suited. In fact, the step-size parameter on the average reward is a perfect place to use the unbiased constant-step-size trick from Exercise 2.7. Describe the specific changes needed to the boxed algorithm for differential semi-gradient n-step Sarsa to use this trick.*

   We would need to keep track of o_n. That is, initialize it to 0 and update it by $o \leftarrow o + \beta(1 - o)$ each time we update $\bar{R}$. (Before updating $\bar{R}$.) Also, when updating $\bar{R}$, use β/o instead of β.