Solutions to Reinforcement Learning by Sutton Chapter 10

Yifan Wang

Jan 2020

Exercise 10.1

There are few reasons of why book did not consider any Monte Carlo methods. First, the computational cost of Monte Carlo method is too high to make it practical in complex settings. In Mountain Car problem, Monte Carlo method will not start to learn unless it can complete the first full episode with a random policy. Even if it completes it, the amount of time steps would make the computation cost extremely high. On the other hand, SARSA and other TD methods can learn during progress and converge much faster to make it online.

Second, Monte Carlo method is just an extreme variation of more general group of TD methods. If we write down its pseudocode, it will not be so much different from n-step SARSA except $n = \mathrm{T}$.

■

Exercise 10.2

Repeat the pseudocode as in the box "Episodic Semi-gradient Sarsa for Estimating $\hat{q} \approx q_*$" on page 244. Then, replace the "$\gamma\hat{q}(S', A', w)$" in the 3rd line from bottom with "$\gamma \sum_a \pi(a|S')\hat{q}(S', a, w)$." Finally, swap this line with the line above. Though we can simplify a little bit more, it completes the on-policy Episodic Semi-gradient expected Sarsa.

■

Exercise 10.3

Larger n will tend to value longer history of rewards and thus amplify its variance effect in its first episodes. It is due to the strong dependence on its individual trajectory which usually is chaotic and random in its early stage. In another word, some 'strange' choices made during early steps will affect many other states before, and make the result less general. On the other hand, smaller n such as $n = 1$ only looks into next state S_{t+1}. Even with strange choices, the importance of its result S_{t+1} will be averaged out soon, since it only provides information to S_t.

However, when we have much longer episodes like in Figure 10.3 with 500 episodes, relatively large n will provide nicer result, since it looks into related local variations carefully (by a DP-like way) and thus avoids shaking around the convergence point.

■

Exercise 10.4

Recall the subtle difference between Q-learning and SARSA: Q-learning chooses the greedy action at each timing but SARSA provides the 'old guess' of its (possible) greedy action to the next step.

Thus, we can repeat the pseudocode in the box on page 251, but change the next few lines in the box as following:

Take action A, observe $R, S' \longrightarrow$
Choose A as a function of $\hat{q}(S', \cdot, \mathrm{w})$, observe R, S'.

Choose A' as a function of $\hat{q}(S', \cdot, \mathrm{w}) \longrightarrow$
Choose A' as $\arg\max_{A'} \hat{q}(S', \cdot, \mathrm{w})$.

■

Exercise 10.5

Though I do not fully understand what this question is asking for, I expect to see the update formula for w**:**

$$\mathrm{w}_{\mathrm{t}+1} \doteq \mathrm{w}_t + \alpha\delta_t \nabla \hat{v}(S_t, \mathrm{w}_t)$$

Anyone who knows the meaning of this question please report as an issue, Thanks.

■

Exercise 10.6

I have to clarify the problem setting first due to little bit ambiguity in the original wording. The problem simply state that we have three states A, B, C **with loop and deterministic movement in between. Without losing generality, it is amount to say we have a loop like** $A \to B \to C \to A$ **and reward will be given whenever we reach state** A**.**

The average reward is obviously $\frac{1}{3}$**. Then we have:**

$$\begin{aligned} v_\pi(C) &= 1 - \frac{1}{3} + v_\pi(A) \\ v_\pi(B) &= 0 - \frac{1}{3} + v_\pi(C) \\ v_\pi(A) &= 0 - \frac{1}{3} + v_\pi(B) \end{aligned}$$

An obvious solution for these equations are $v_\pi(C) = \frac{2}{3}$**,** $v_\pi(B) = \frac{1}{3}$ **and** $v_\pi(A) = 0$

■

Exercise 10.7

The average reward is +1.

Since $r(\pi)$ is +1, and $\mathbb{E}_\pi[R_{t+1|S_0=s}]$ is 1 under both A and B, we conclude from (10.13) they are of the same value, 0.

■

Exercise 10.8

The sequence of $R_t - \bar{R}_t$ would be (from state C):

$$\frac{2}{3}, \quad -\frac{1}{3}, \quad -\frac{1}{3}, \ldots \text{(repeat the same pattern)}$$

However, the sequence of δ_t by (10.10) would be (from state C):

$$\frac{2}{3} + 0 - \frac{2}{3} = 0, \quad -\frac{1}{3} + \frac{1}{3} - 0 = 0, \quad -\frac{1}{3} + \frac{2}{3} - \frac{1}{3} = 0, \quad \ldots$$

We observe $\delta_t = 0$ will result no update because we have already have accurate estimate of $\hat{v}$ and $\bar{R}$. This method will give us more stable estimate since it will automatically decrease the speed of oscillation when we are close to convergence and increase the speed of change (by amount $\hat{v}(S_{t+1}) - \hat{v}(S_t)$) when we are in the opposite situation.

■

Exercise 10.9

Replacing all β with β_n. Adding following lines before δ get updated:

$$\beta_n \doteq \beta / \bar{o}_n,$$
$$\bar{o}_n \doteq \bar{o}_{n-1} + \beta(1 - \bar{o}_{n-1}), \text{for } n \geq 0, \text{ with } \bar{o}_0 \doteq 0.$$

■