Reinforcement Learning, Homework 2

Hannah Min: 11011580, Harm Manders: 10677186

September 2020

3.1 Coding Assignment - Monte Carlo

1. $$V_{n+1} = V_n + \frac{1}{n}[W_n G_n - V_n]$$

2. Notebook is uploaded to codegrade.

3. A difference between Monte Carlo methods and Dynamic Programming is that for Monte Carlo we do not need to know the complete probability distributions for all possible transitions. Another difference is that Monte Carlo methods are generally applied to episodic tasks, which means that estimates are updated per episode. In DP we update our estimates per step.

4.3 Coding Assignment - Temporal Difference Learning

1. See submission on codegrade

2. As seen in Figure 1, Q-Learning performs better than SARSA, which is generally speaking also the case. In the Example 6.6 in the book, Q-learning performs worse than SARSA, because Q-learning will find the optimal path without taking taking into account the ϵ-greedy action selection that is used (which results in falling off the cliff sometimes). This will result in a lower average return during training, but will find the optimal path. SARSA does take the action selection policy into account and will find the safer route, therefore increasing the average return value.

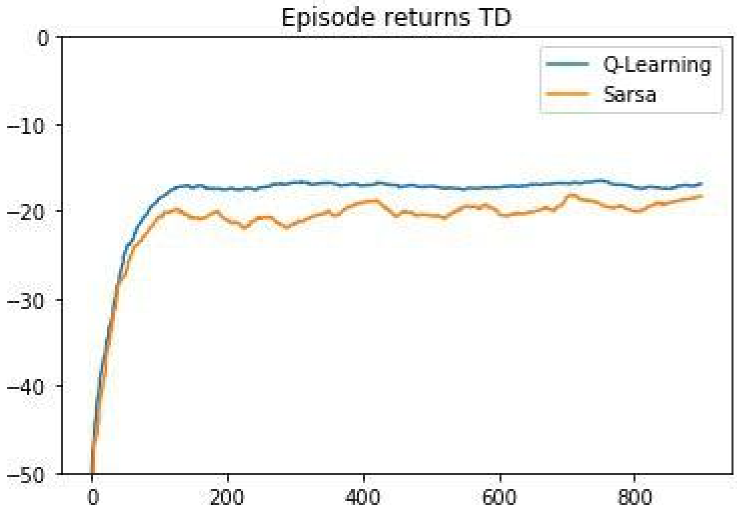

Figure 1: Q-Learning vs SARSA

4.4 Maximization Bias

1. Q-Learning will take the max Q value of the next action. Therefore Q(A,R) will converge to the max Q value in B.

	L	R	a_1	a_2	a_3	a_4
A	0.7	1	-	-	-	-
B	-	-	0	1	0.5	0.5

   SARSA will converge to 0.5 in this case, as the average of the Q value in B is 0.5.

	L	R	a_1	a_2	a_3	a_4
A	0.7	0.5	-	-	-	-
B	-	-	0	1	0.5	0.5

2. After sampling more episodes, the expected reward for every action from B will converge to 0.5

	L	R	a_1	a_2	a_3	a_4
A	0.7	0.5	-	-	-	-
B	-	-	0.5	0.5	0.5	0.5

3. Both Q-Learning and SARSA suffer from maximization bias.
   In this example though, you only see it in Q-Learning. We observe that

$Q(A,R) > Q(A,L)$ while this should not be the case. While Q-Learning inherently has a maximization bias because the max operator is used in the update rule, SARSA can also be affected by maximization bias if the policy is greedy, or in some way includes a maximization operator.

4. With double Q-Learning you use two Q functions that are independent of each other. When updating Q_1 you will use the value from Q_2 of the best action based on Q_1. This makes sure that the Q-values in S_t are not influenced directly by the Q-values in S_{t+1}.
   In this example, if more episodes are generated, Double Q-Learning will ensure that the Q-values in Q(A,R) will converge more quickly towards the real value of 0.5. Because it is less likely that both of the Q functions have observed the same episodes, balancing each other out.