Reinforcement Learning Assignment 3

Sun Hao
Student ID:516030910362

1 Introduction

This assignment is to do experiment with model-free control, including on-policy learning (Sarsa) and off-policy learning (Q-learning). I will first build the "Cliff Walking" environment, and search the optimal travel path by Sara and Q-learning, respectively. Besides, I will modify some parameters to see impacts on performance.

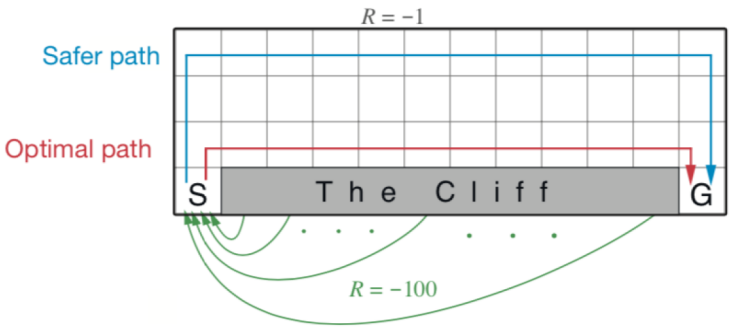


Figure 1: Cliff Walking

Consider the gridworld shown in the Figure 1. This is a standard undiscounted, episodic task, with start state (S), goal state (G), and the usual actions causing movement up, down, right, and left. Reward is -1 on all transitions except those into the region marked "The Cliff". Stepping into this region incurs a reward of -100 and sends the agent instantly back to the start.

2 Method

2.1 Sarsa

Sarsa is an on-policy Temporal-Difference Learning. its basic transaction equation is

$$Q(S,A) \leftarrow Q(S,A) + \alpha\left(R + \gamma Q\left(S',A'\right) - Q(S,A)\right) \tag{1}$$

where $Q(\cdot,\cdot)$ is action value function, R is the instant reward after action A, and γ is the discount factor.

Here comes the sarsa algorithm for on-policy control.

2.2 Q-learning

Q-learning is an off-policy learning. Off-policy, compared with on-policy, is more powerful and general, often of greater variance and slower convergence. Its update equation is that

$$Q(S,A) \leftarrow Q(S,A) + \alpha\left(R + \gamma \max_{a'} Q\left(S',a'\right) - Q(S,A)\right) \tag{2}$$

Q-learning Algorithm for Off-Policy Control is shown as below.

Initialize $Q(s,a)$,$\forall s \in \mathcal{S}$,arbitrarily, and $Q(terminal-state,\cdot)=0$;
for *each episode* **do**
  Initialize S;
  Choose A from S using policy derived from Q;
  for *each step of episode* **do**
    Take action A, observe R,S';
    Choose A' from S' using policy derived from Q;
    $Q(S,A) \leftarrow Q(S,A)+\alpha\left(R+\gamma Q\left(S',A'\right)-Q(S,A)\right)$;
    $S \leftarrow S'$;
    $A \leftarrow A'$;
  end
  if S *is terminal* **then**
    break;
  end
end

Algorithm 1: Sarsa Algorithm for On-Policy Control

Initialize $Q(s,a)$,$\forall s \in \mathcal{S}$,arbitrarily, and $Q(terminal-state,\cdot)=0$;
for *each episode* **do**
  Initialize S;
  for *each step of episode* **do**
    Choose A from S using policy derived from Q;
    Take action A, observe R,S';
    $Q(S,A) \leftarrow Q(S,A)+\alpha\left(R+\gamma \max_{a'} Q\left(S',a'\right)-Q(S,A)\right)$;
    $S \leftarrow S'$;
  end
  if S *is terminal* **then**
    break;
  end
end

Algorithm 2: Q-learning Algorithm for Off-Policy Control

3 Experiment

3.1 Parameters

number of episode(both)	1000
α(Sarsa)	0.5
α(Q-learning)	0.1
γ(both)	1.0
ϵ(both)	0.1

Table 1: Parameters Setting

3.2 Result

The final action functions of two methods:

→	→	→	→	→	→	→	→	→	→	→	↓
↑	↑	↑	↑	↑	↑	←	↑	↑	→	→	↓
↑	↑	↑	↑	↑	↑	←	↑	↑	↑	→	↓
↑	-	-	-	-	-	-	-	-	-	-	-

Table 2: Sarsa actions of each state

←	→	→	→	→	↓	→	→	↓	→	↓	↓
↑	↓	←	→	→	→	→	→	→	↓	↓	↓
→	→	→	→	→	→	→	→	→	→	→	↓
↑	-	-	-	-	-	-	-	-	-	-	-

Table 3: Q-learning actions of each state

Note that because the experiment contains some random process(e.g. ϵ-greedy). The results from each run are not exactly the same. But the path of agent from start position to end position is the same. Here comes a nicer figure showing the result.

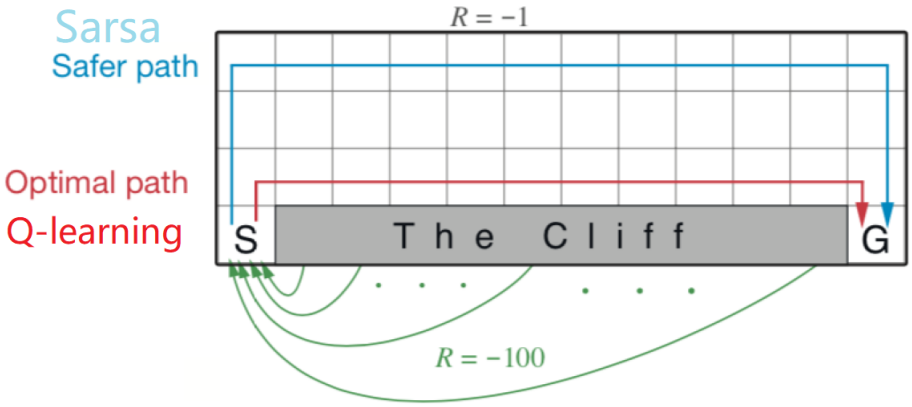


Figure 2: Cliff Walking result of two methods

The experimental result shows that sarsa tends to choose the safer path, while q-learning prefers the optimal(shortest) path.

I draw the averaged award changes over the episodes.

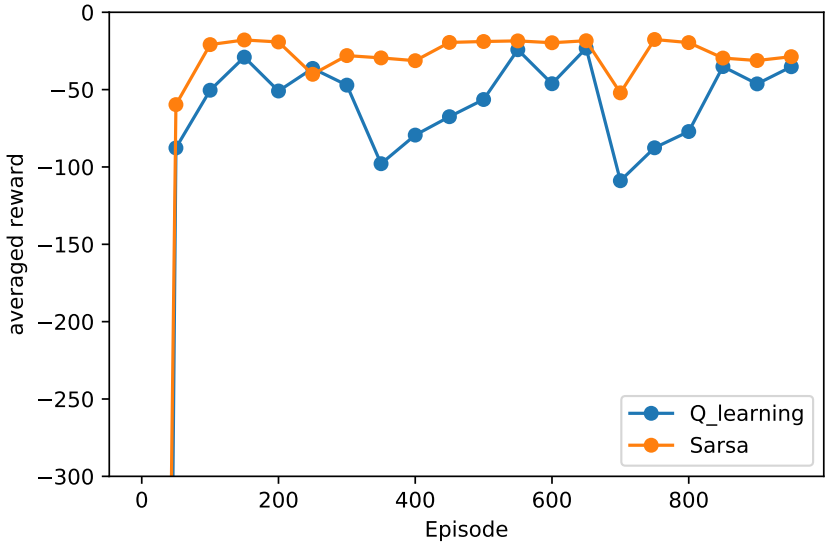
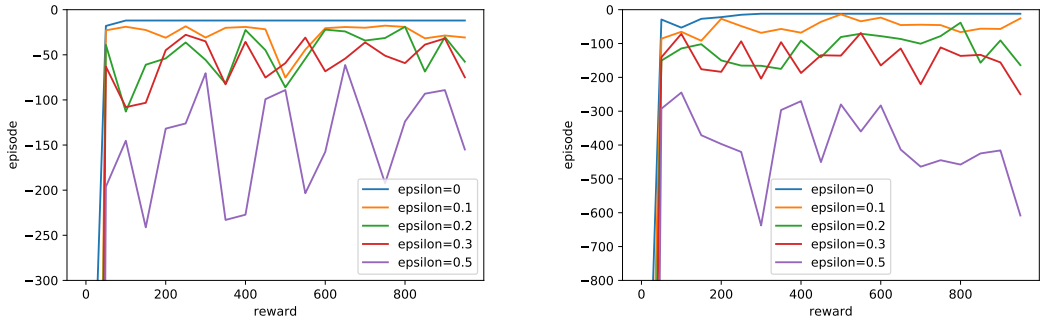


Figure 3: Averaged award of two methods. The figure is drawn by picking one point every 50 episodes.

The figure shows that Sarsa has a higher averaged reward than Q-learning. This is This is in line with our expectations. Q-learning chooses the optimal ways, though agent with Q-learning may have the maximum reward, it is very possible to get into the cliff with ϵ-greedy strategy. That

is also the reason that Q-learning curve has some sudden drops. Conversely, sarsa gets a safer path and it is not so likely to get into cliff.

3.3 Impacts on performance with different ϵ



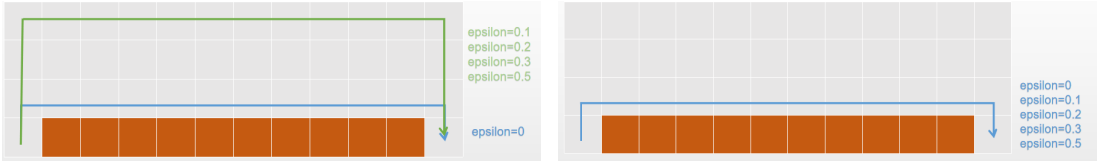
(a) Sarsa (b) Q-learning

Figure 4: The reward changes with the episodes, with different ϵ. The figure is drawn by picking onepoint every 50 episodes. Notice that the left and right graphs have different y ordinate ranges

The figures show that **the reward decreases when ϵ increases**. This is explainable. If ϵ increases, the possibility of getting into the direction of cliff increases. Once getting into the cliff, the reward will drop suddenly by 100, that is also why **the curves with larger ϵ joggle with more amplitude**.

Paths with different ϵ

The path may change when ϵ changes, here is the paths with different ϵ of two methods.



(a) Sarsa (b) Q-learning

Figure 5: Different paths when ϵ changes of two methods

4 Conclusion

In the "Cliff Walking" environment, Sarsa tends to choose the safer path, while Q-learning perfers to choose the optimal path.

The factor to decide random choice ϵ has a great impact on performance. On the one hand, larger ϵ makes the agent easier to go to the cliff, making the reward decrease. On the other hand, different ϵ will get the different path. Specially, when $\epsilon = 0$, Sarsa has the same path(optimal path) as the Q-learning.