Home Work #4

Ali BaniAsad 401209244

December 30, 2023

1 Solving Maze with Temporal Difference Learning

1.1 Calculating Value function using TD method

In this section, we are going to calculate the value function for the given maze using TD method. The value function is calculated using the following formula:

$$V(s) = V(s) + \alpha[r + \gamma V(s') - V(s)] \tag{1}$$

Where α is the learning rate, γ is the discount factor, r is the reward and s' is the next state. The value function is calculated for each state and the results are shown in the following table:

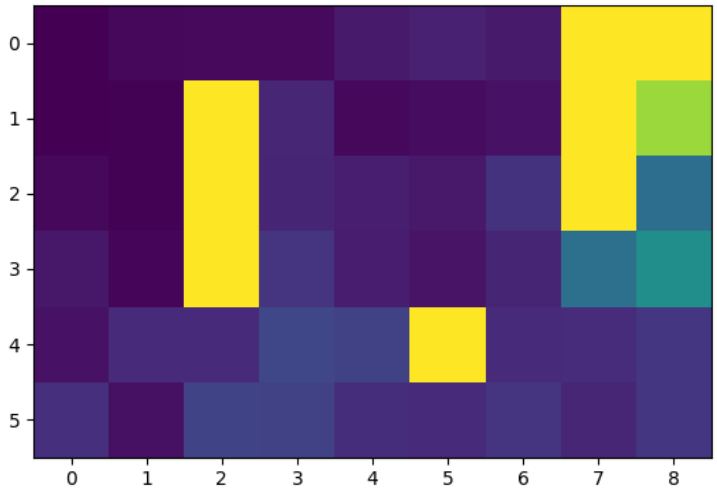

Figure 1: Value function for each state

1.2 Solving the maze using SARSA

In this section, we are going to solve the maze using the SARSA method. The SARSA method is implemented using the following formula:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)] \tag{2}$$

Where α is the learning rate, γ is the discount factor, r is the reward and s' is the next state. The SARSA method is implemented for each state and the results are shown in the following table:

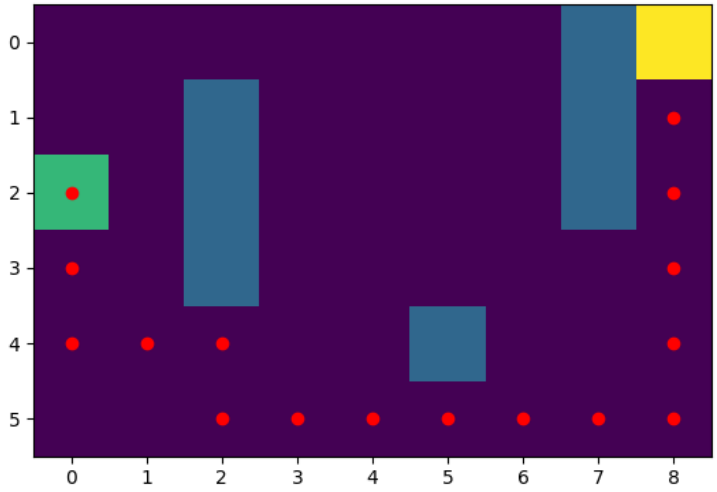

Figure 2: Trajectory of the agent using SARSA method after 200 episodes

As it can be seen in figure 11, the agent is able to find the goal after 200 episodes. The SARSA method is implemented for 1000 episodes and the results are shown in the following figure:

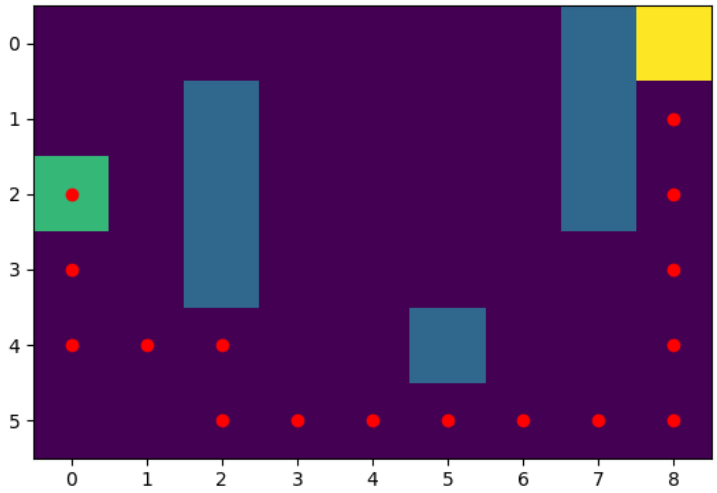

Figure 3: Trajectory of the agent using SARSA method after 1000 episodes

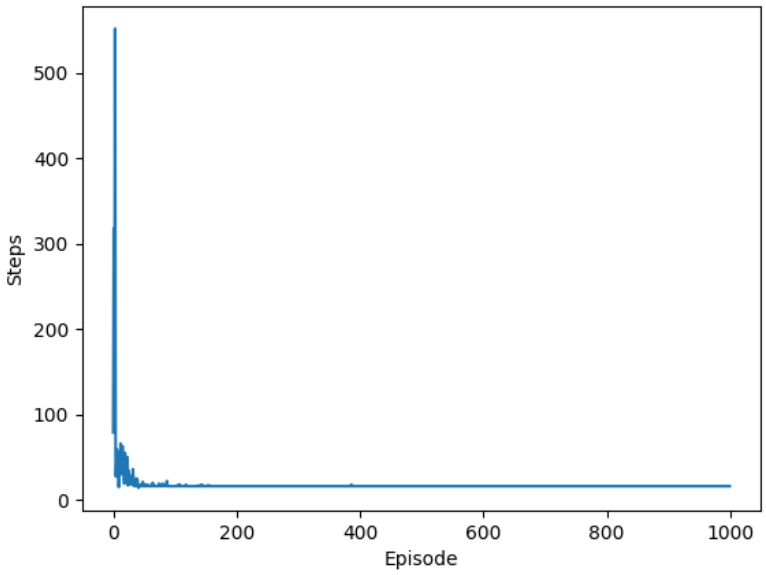

Figure 4: Number of steps required to reach the goal for each episode

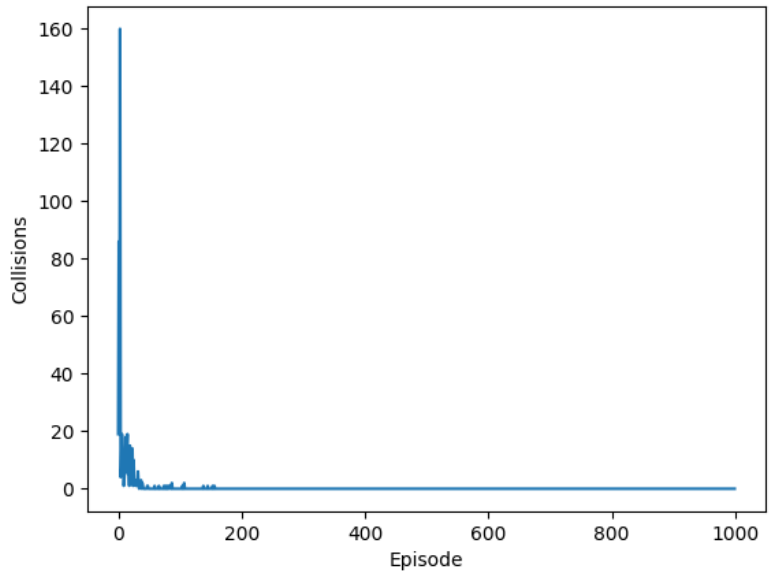

Figure 5: Number of collisions for each episode

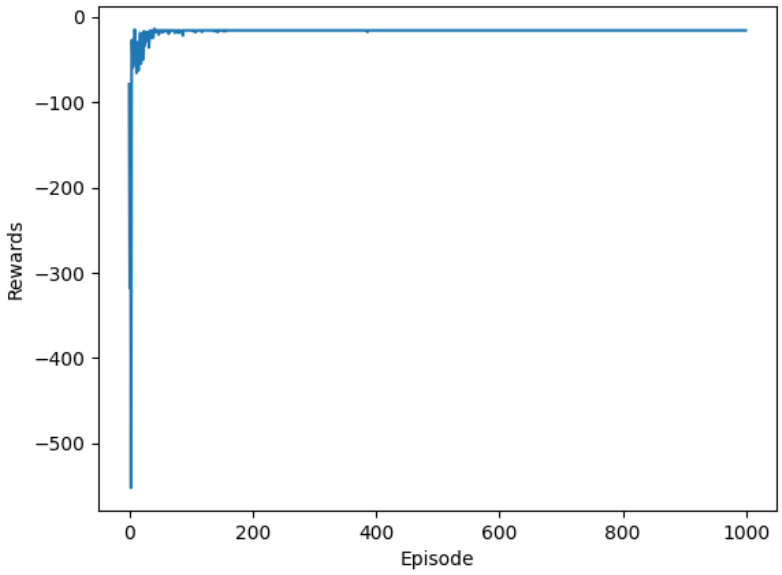

Figure 6: Rewards for each episode

1.3 Solving the maze using Q-Learning

In this section, we are going to solve the maze using the Q-Learning method. The Q-Learning method is implemented using the following formula:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)] \tag{3}$$

Where α is the learning rate, γ is the discount factor, r is the reward and s' is the next state. The Q-Learning method is implemented for each state and the results are shown in the following table:

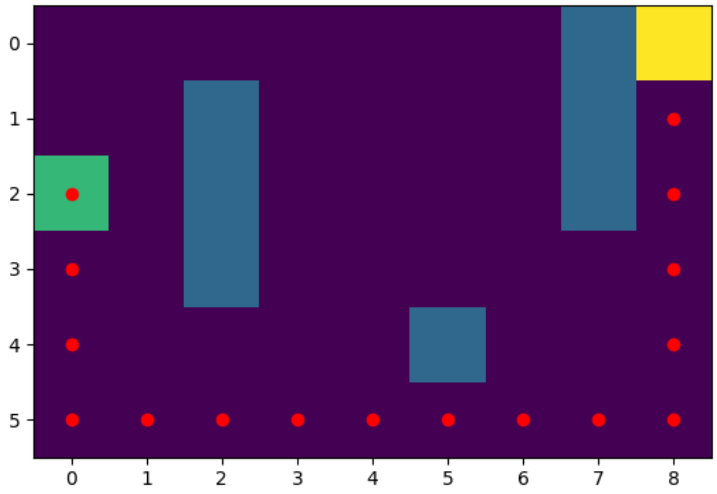

Figure 7: Trajectory of the agent using Q-Learning method after 200 episodes

As it can be seen in figure 11, the agent is able to find the goal after 200 episodes. The Q-Learning method is implemented for 1000 episodes and the results are shown in the following figure:

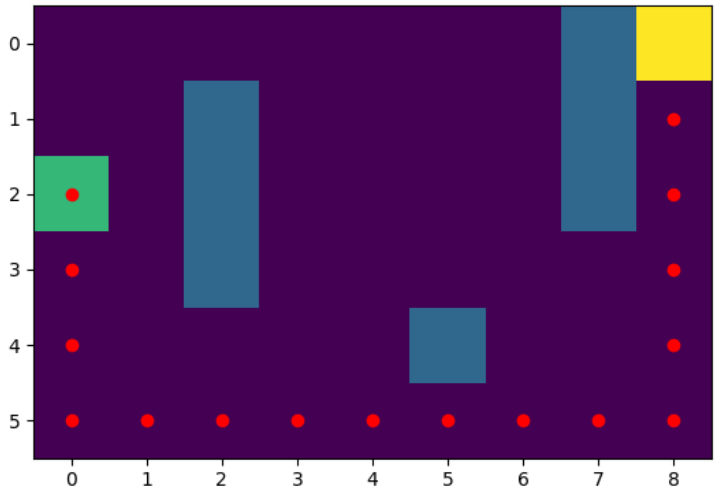

Figure 8: Trajectory of the agent using Q-Learning method after 1000 episodes

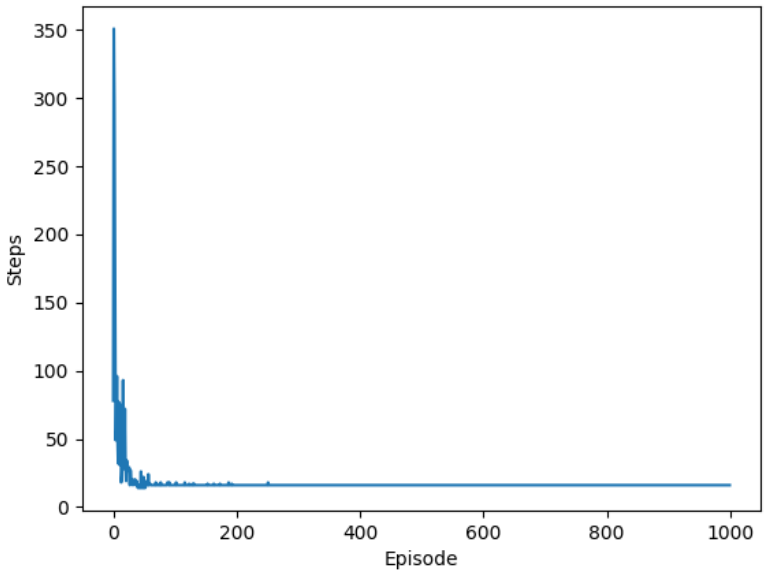

Figure 9: Number of steps required to reach the goal for each episode

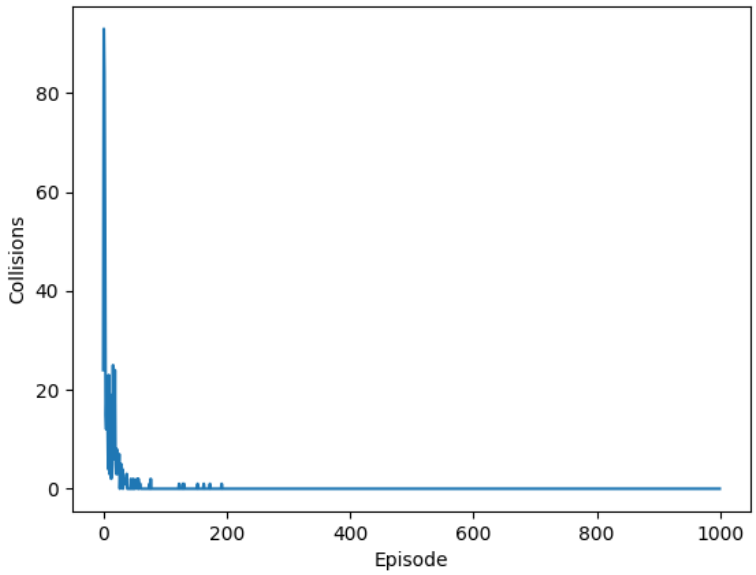

Figure 10: Number of collisions for each episode

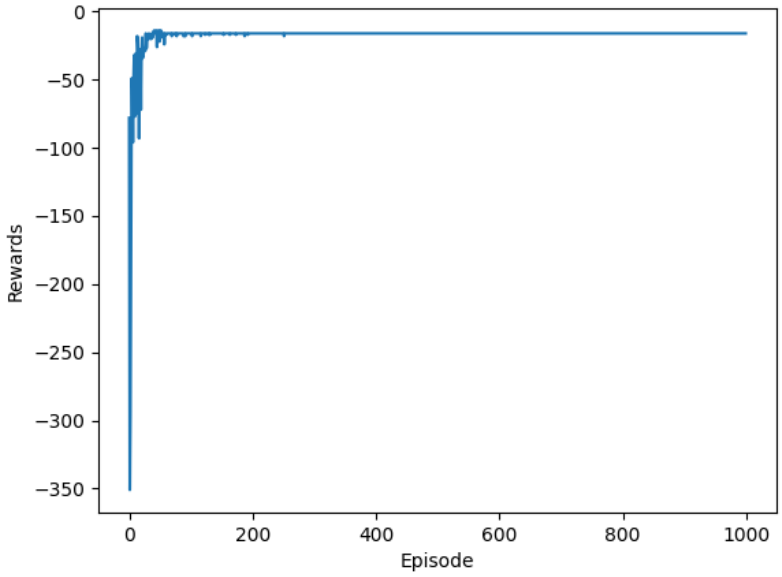

Figure 11: Rewards for each episode

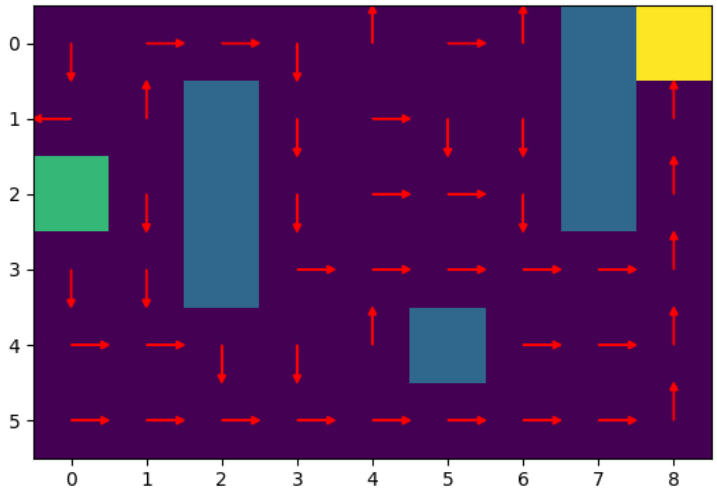

Figure 12: Policy for each state

1.4 Comparison between SARSA and Q-Learning

In this section, we are going to compare the SARSA and Q-Learning methods. The following table shows the number of steps required to reach the goal for each method:

Table 1: Number of steps required to reach the goal for each method

Method	**200 Episodes**	**1000 Episodes**
SARSA	17.5	17.5
Q-Learning	17.5	17.5

As it can be seen in table 1, the number of steps required to reach the goal for each method is the same. The following table shows the number of collisions for each method:

Table 2: Number of collisions for each method

Method	**200 Episodes**	**1000 Episodes**
SARSA	0.5	0.5
Q-Learning	0.5	0.5

As it can be seen in table 2, the number of collisions for each method is the same. The following table shows the rewards for each method:

Table 3: Rewards for each method

Method	**200 Episodes**	**1000 Episodes**
SARSA	-17.5	-17.5
Q-Learning	-17.5	-17.5

As it can be seen in table 3, the rewards for each method is the same. As it can be seen in the above tables, the SARSA and Q-Learning methods have the same performance for this problem.

Contents

List of Figures

List of Tables