CS6700: Programming Assignment 1
SARSA & Q-Learning

Aaditya Kumar (EE21D411) and Soumen Pachal(CS22D009)

February 25, 2023

1 Introduction

This exercise familiarized us with two popular Temporal Difference Learning algorithms: SARSA and Q-Learning. We are applying both algorithms to solve several variants of the Grid World problem. The observations are described in the upcoming sections.

2 SARSA

The configurations for SARSA are as follows:

- Wind = False:

  1. start state: (3, 6); p = 1.0; exploration strategies = softmax

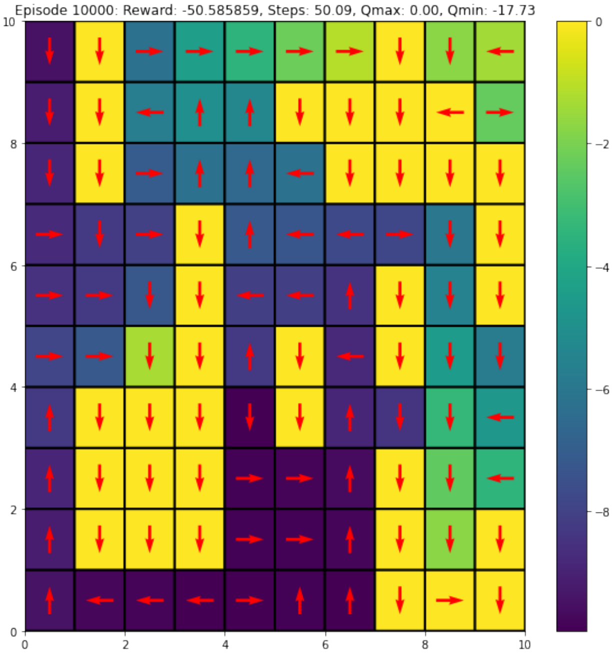

Figure 1: Heatmap of Grid with Q-values

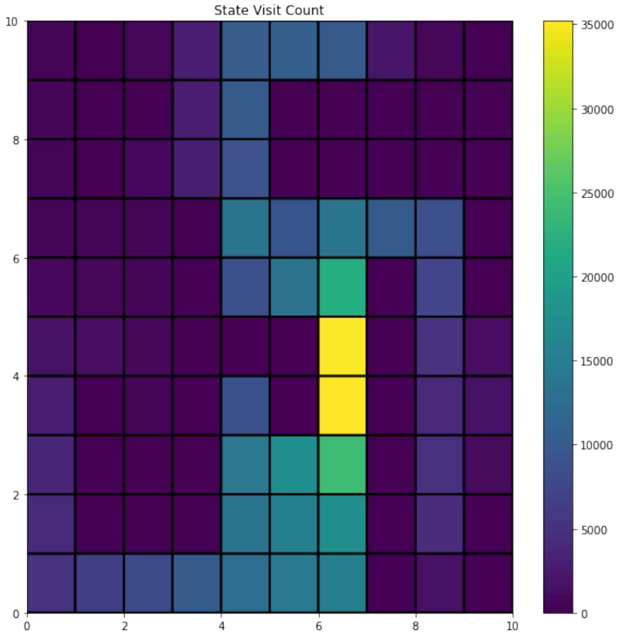

Figure 2: Heatmap of Grid with the state visit counts

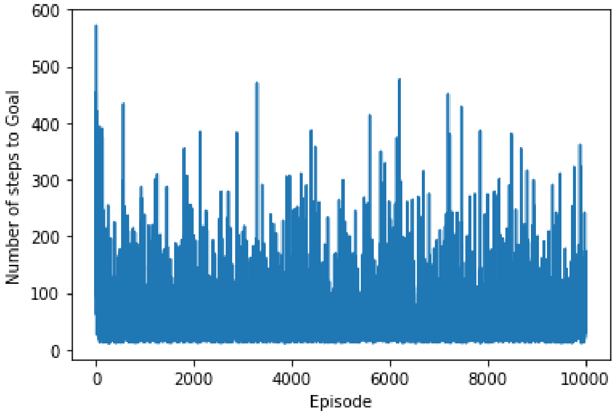

Figure 3: the number of steps to reach the goal in each episode

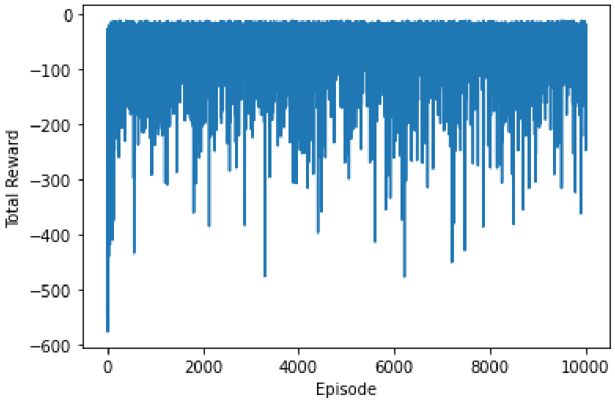

Figure 4: Total Reward

2. start state: (3, 6); p = 1.0; exploration strategies = ϵ-greedy

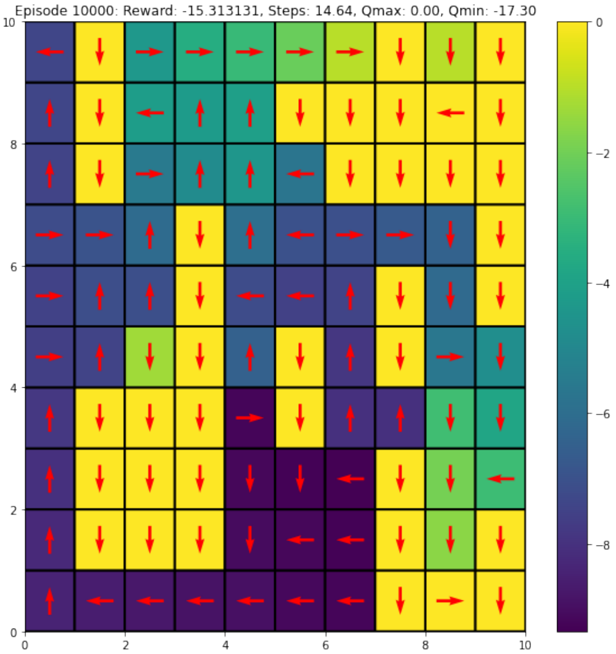

Figure 5: Heatmap of Grid with Q-values

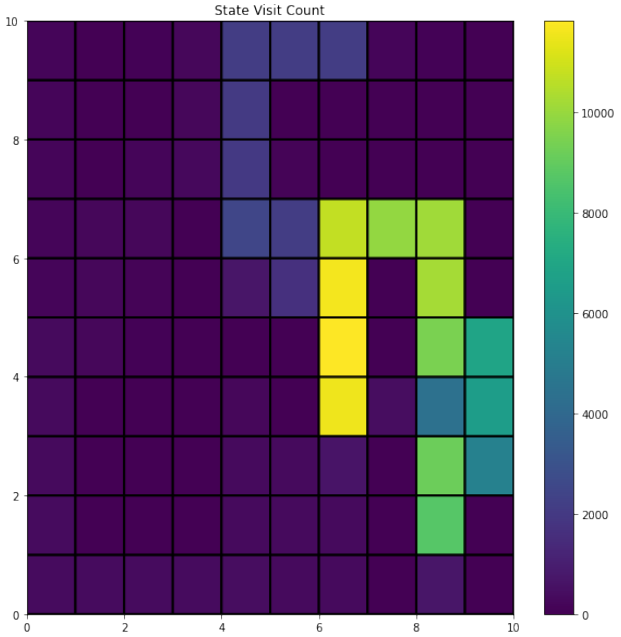

Figure 6: Heatmap of Grid with the state visit counts

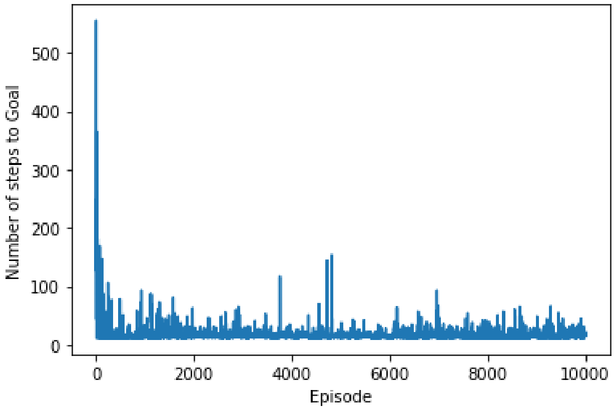

Figure 7: the number of steps to reach the goal in each episode

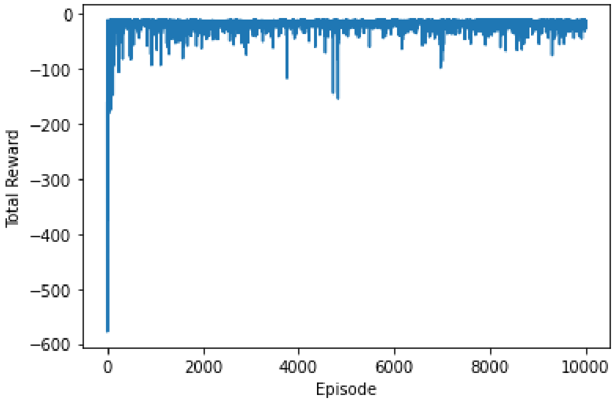

Figure 8: Total Reward

3. start state: (3, 6); p = 0.7 exploration strategies = softmax

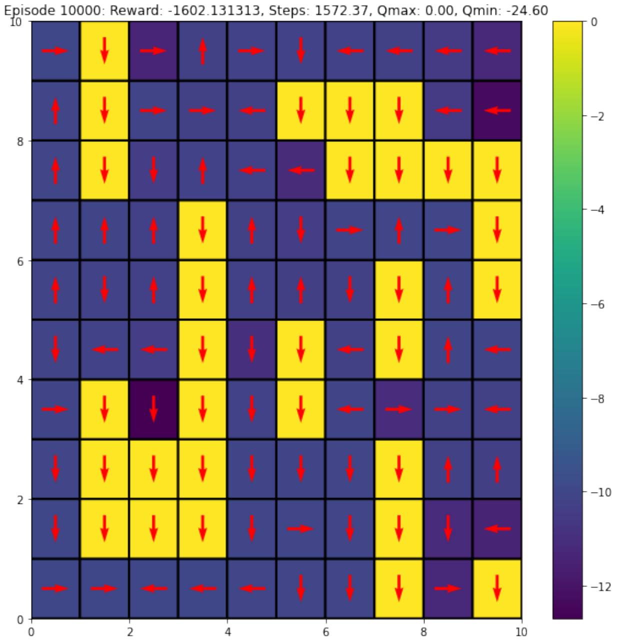

Figure 9: Heatmap of Grid with Q-values

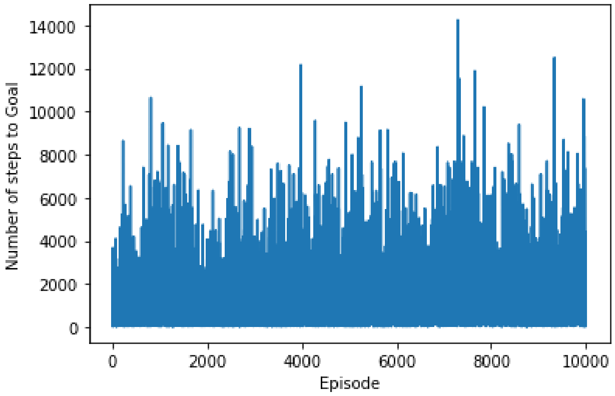

Figure 10: The number of steps to reach the goal in each episode

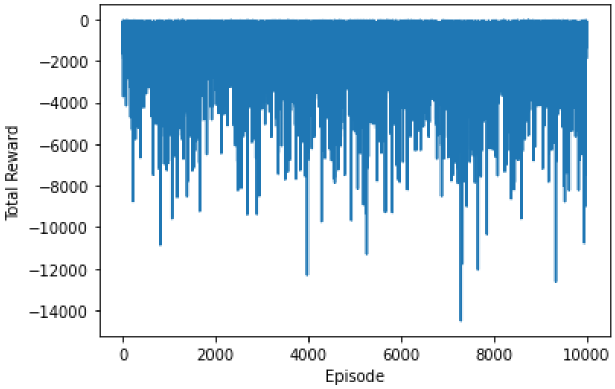

Figure 11: Total Reward

4. start state: (3, 6); p = 0.7; exploration strategies = ϵ-greedy
5. start state: (0, 4); p = 1.0; exploration strategies = softmax

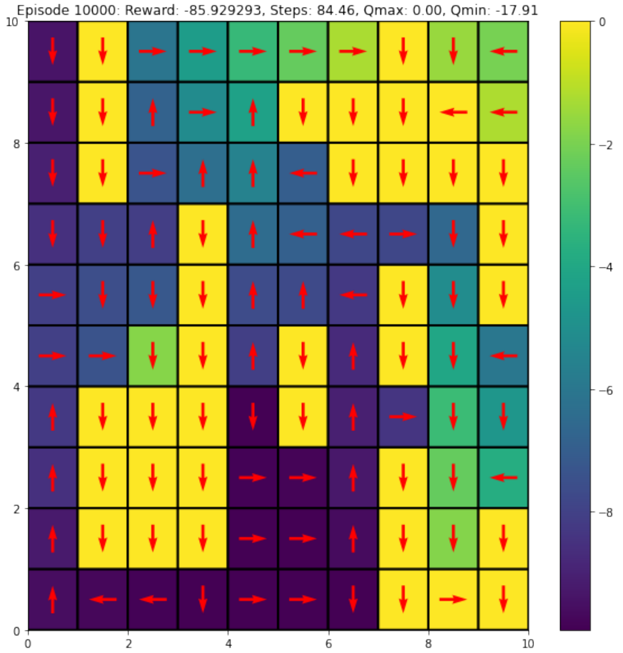

Figure 12: Heatmap of Grid with Q-values

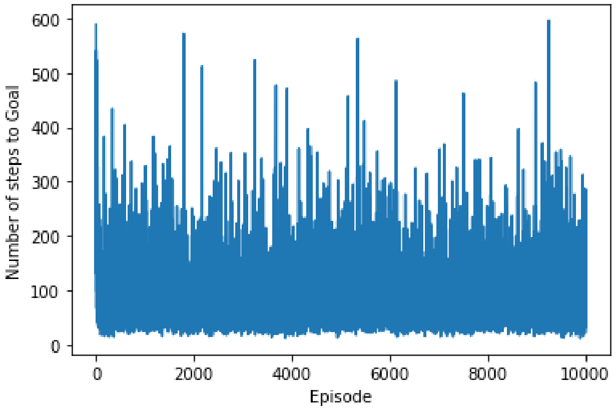

Figure 13: The number of steps to reach the goal in each episode

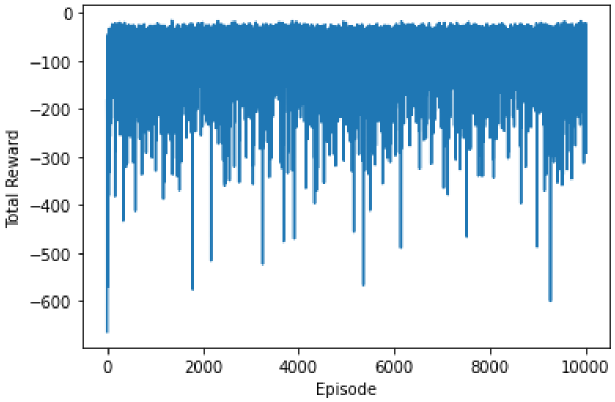

Figure 14: Total Reward

6. start state: (0, 4); p = 1.0; exploration strategies = ϵ-greedy

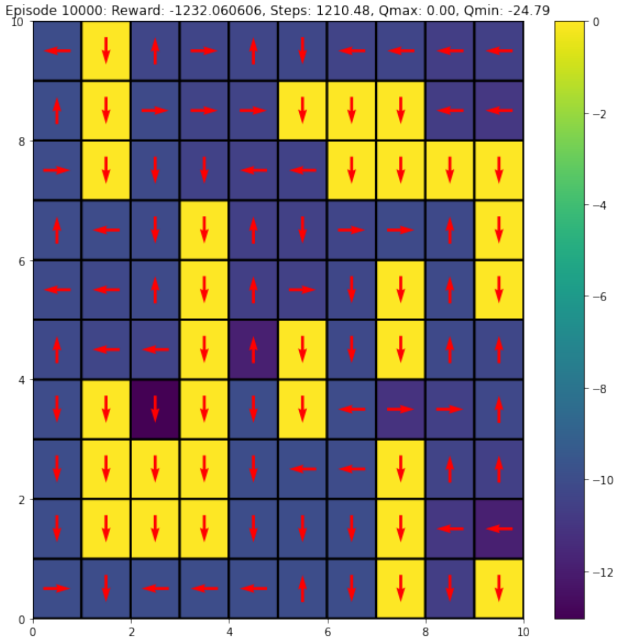

Figure 15: Heatmap of Grid with Q-values

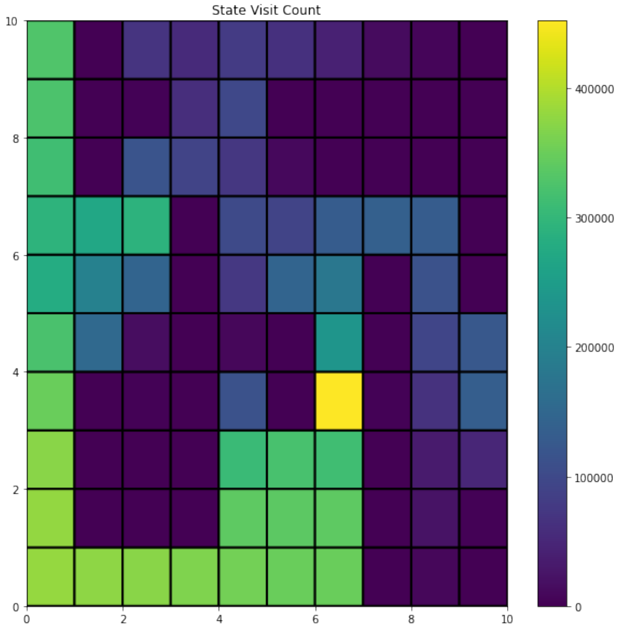

Figure 16: Heatmap of Grid with the state visit counts

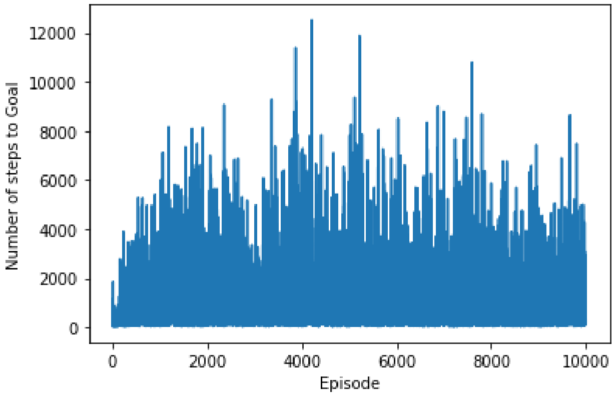

Figure 17: The number of steps to reach the goal in each episode

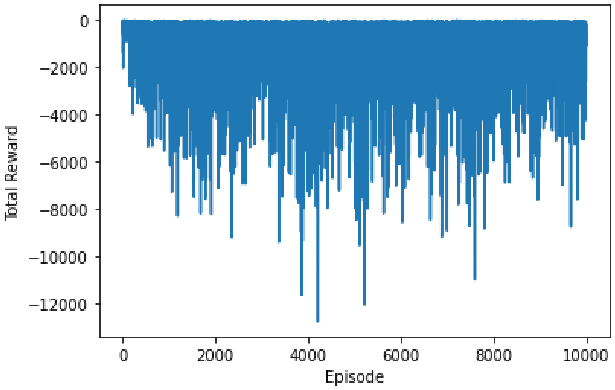

Figure 18: Total Reward

7. start state: (0, 4); p = 0.7; exploration strategies = softmax

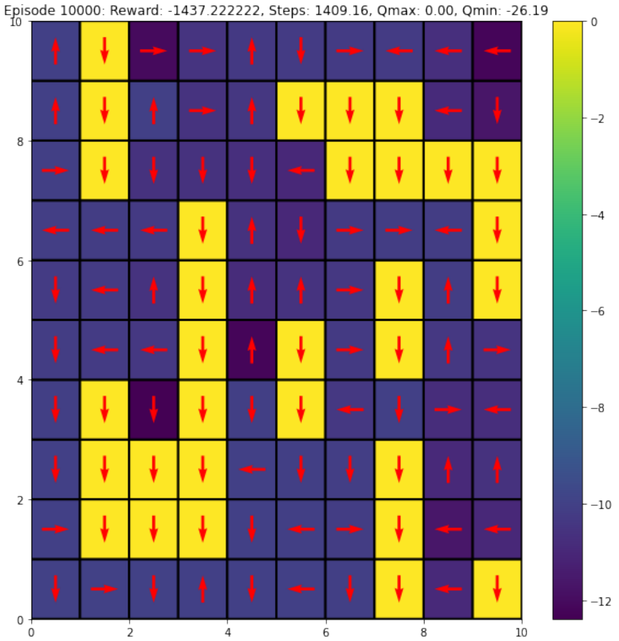

Figure 19: Heatmap of Grid with Q-values

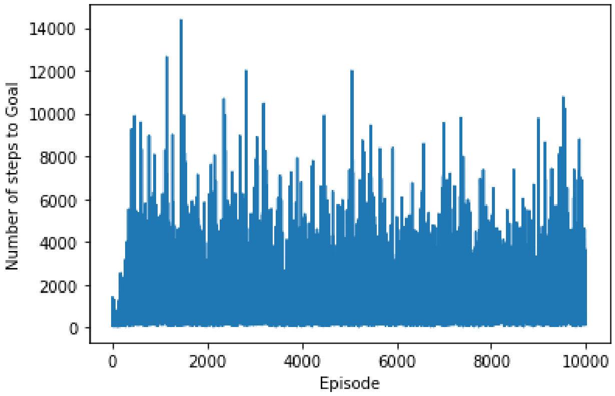

Figure 20: The number of steps to reach the goal in each episode

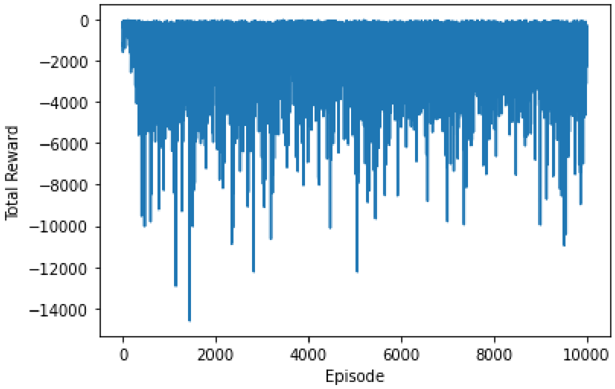

Figure 21: Total Reward

8. start state: (0, 4); p = 0.7; exploration strategies = ϵ-greedy

- Wind = True:

1. start state: (3, 6); p = 1.0; exploration strategies = softmax

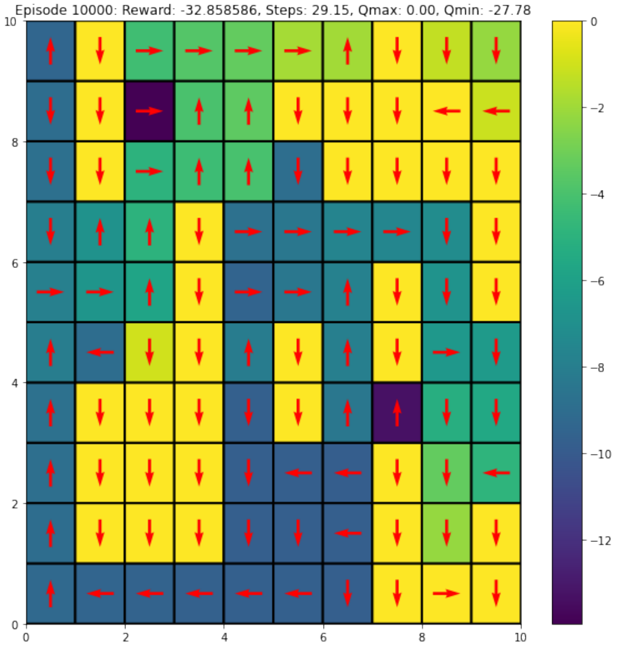

Figure 22: Heatmap of Grid with Q-values

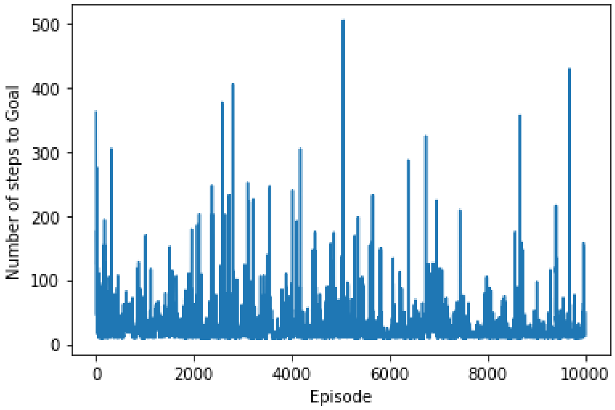

Figure 23: The number of steps to reach the goal in each episode

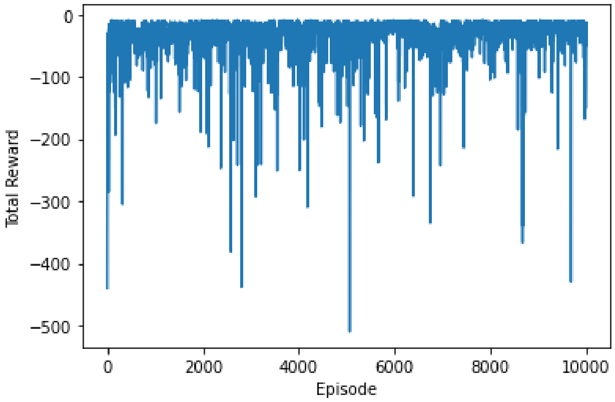

Figure 24: Total Reward

2. start state: (3, 6); p = 1.0; exploration strategies = ϵ-greedy
3. start state: (3, 6); p = 0.7 exploration strategies = softmax
4. start state: (3, 6); p = 0.7; exploration strategies = ϵ-greedy
5. start state: (0, 4); p = 1.0; exploration strategies = softmax
6. start state: (0, 4); p = 1.0; exploration strategies = ϵ-greedy
7. start state: (0, 4); p = 0.7; exploration strategies = softmax
8. start state: (0, 4); p = 0.7; exploration strategies = ϵ-greedy

3 Q-Learning

The configurations for Q-Learning are as follows:

- Wind = False:

  1. start state: (3, 6); p = 1.0; exploration strategies = softmax

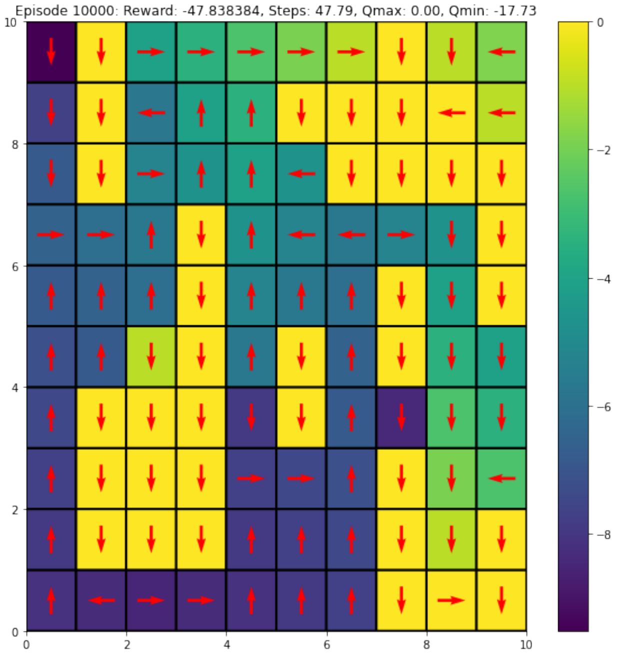

Figure 25: Heatmap of Grid with Q-values

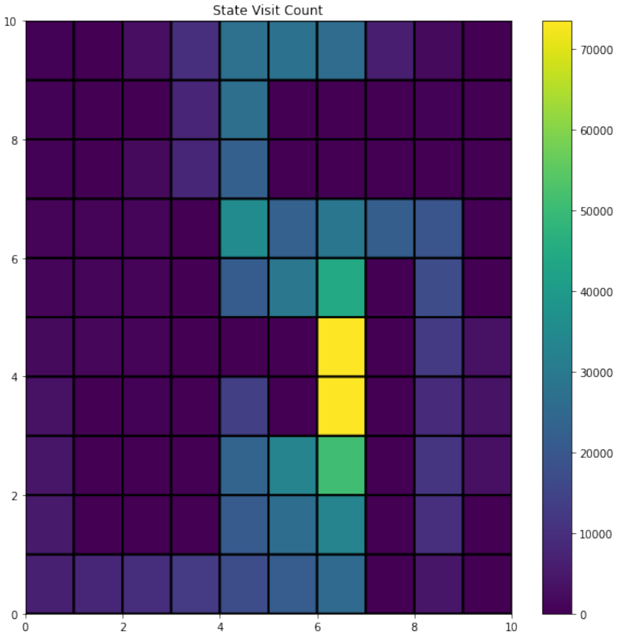

Figure 26: Heatmap of Grid with the state visit counts

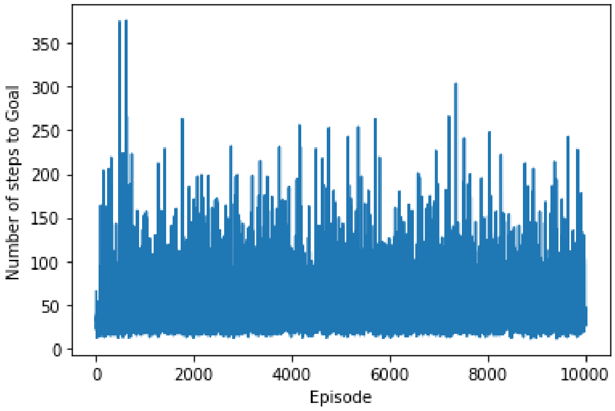

Figure 27: The number of steps to reach the goal in each episode

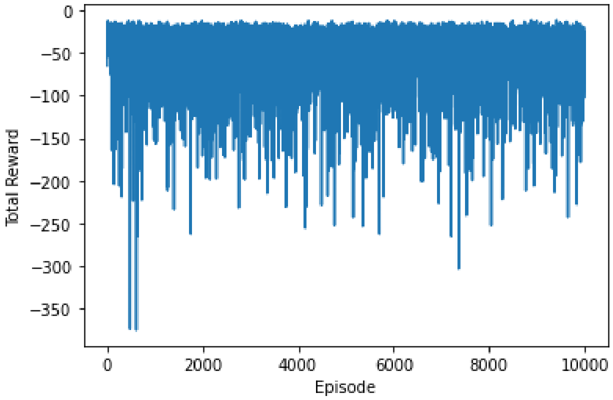

Figure 28: Total Reward

2. start state: (3, 6); p = 1.0; exploration strategies = ϵ-greedy

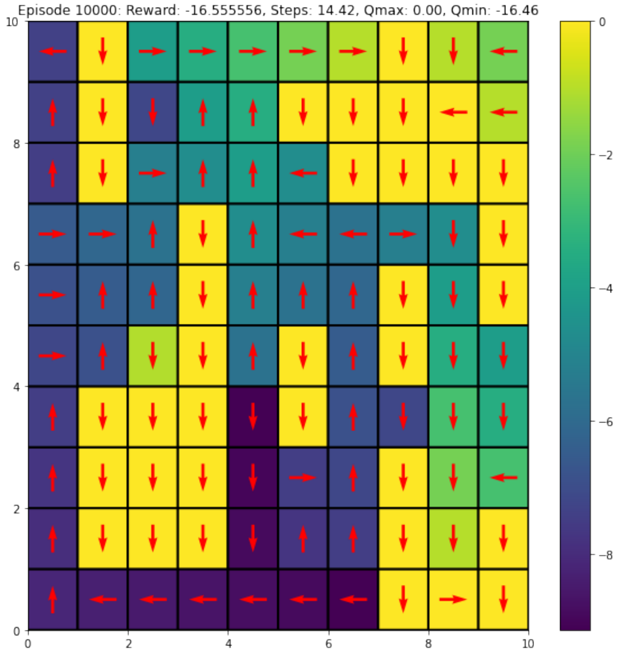

Figure 29: Heatmap of Grid with Q-values

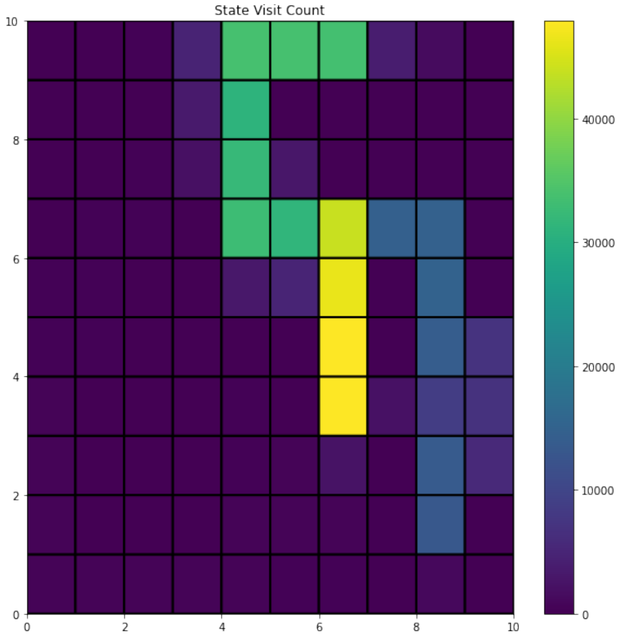

Figure 30: Heatmap of Grid with the state visit counts

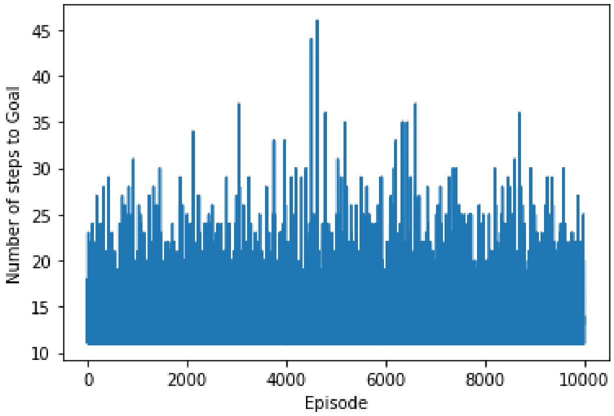

Figure 31: Heatmap of Grid with Q-values

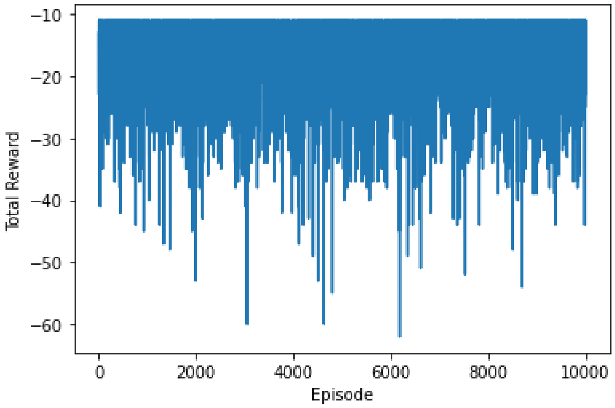

Figure 32: Heatmap of Grid with Q-values

3. start state: (3, 6); p = 0.7 exploration strategies = softmax

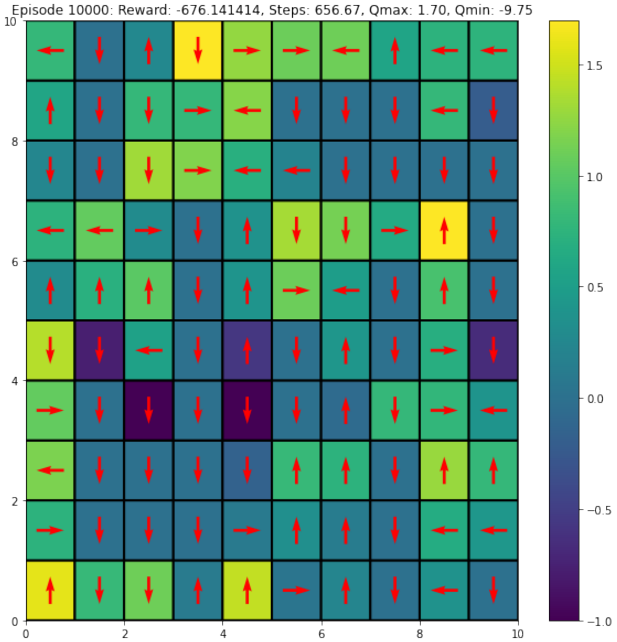

Figure 33: Heatmap of Grid with Q-values

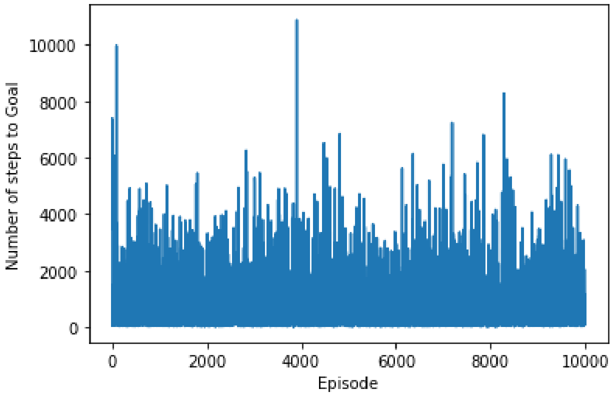

Figure 34: The number of steps to reach the goal in each episode

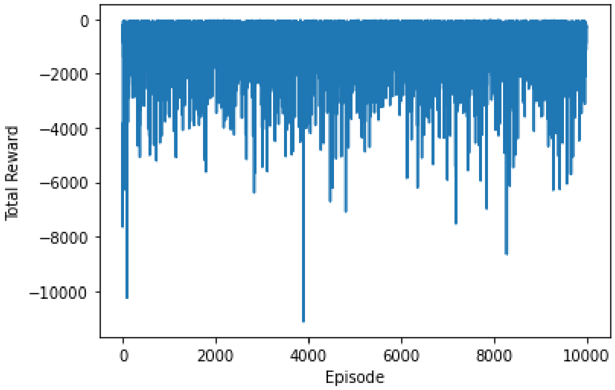

Figure 35: Total Reward

4. start state: (3, 6); p = 0.7; exploration strategies = ϵ-greedy
5. start state: (0, 4); p = 1.0; exploration strategies = softmax

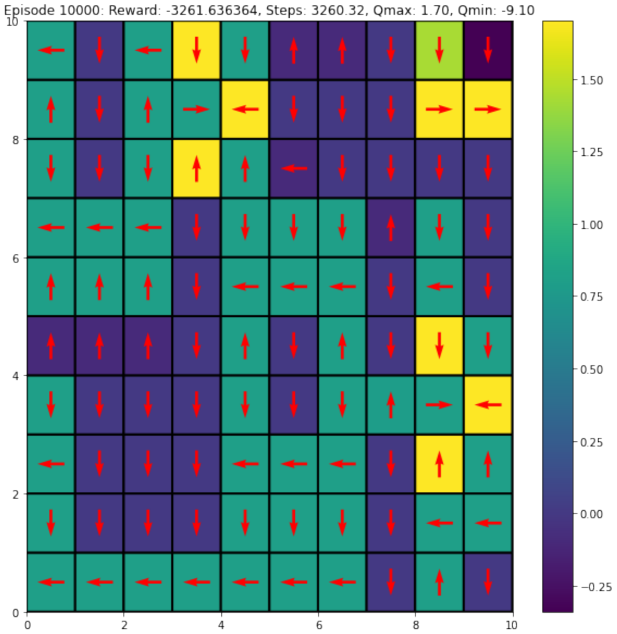

Figure 36: Heatmap of Grid with Q-values

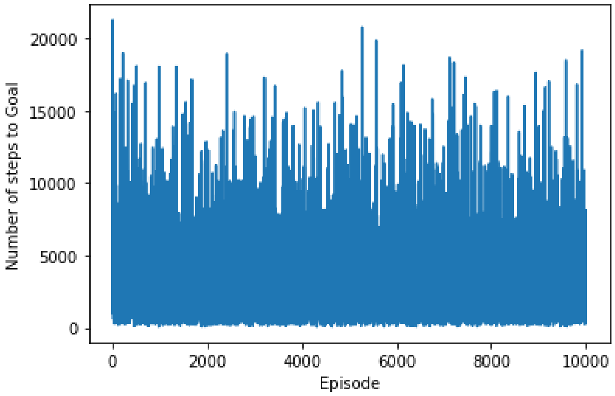

Figure 37: The number of steps to reach the goal in each episode

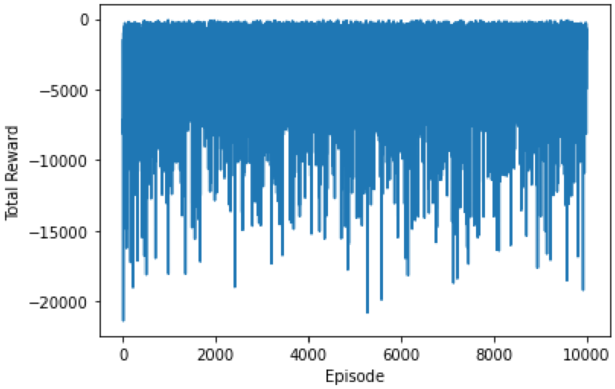

Figure 38: Total Reward

6. start state: (0, 4); p = 1.0; exploration strategies = ϵ-greedy

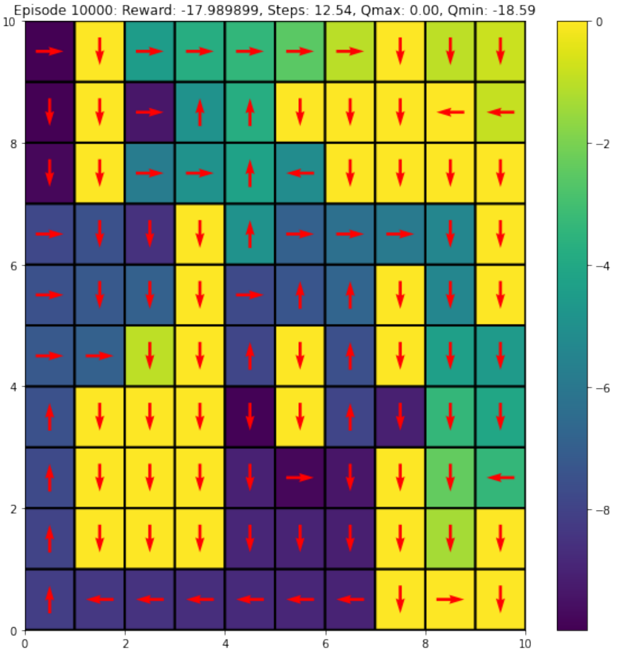

Figure 39: Heatmap of Grid with Q-values

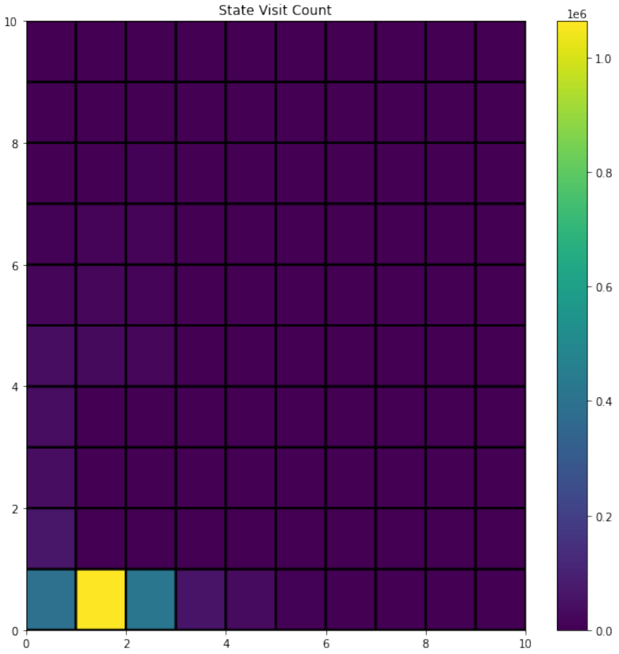

Figure 40: Heatmap of Grid with the state visit counts

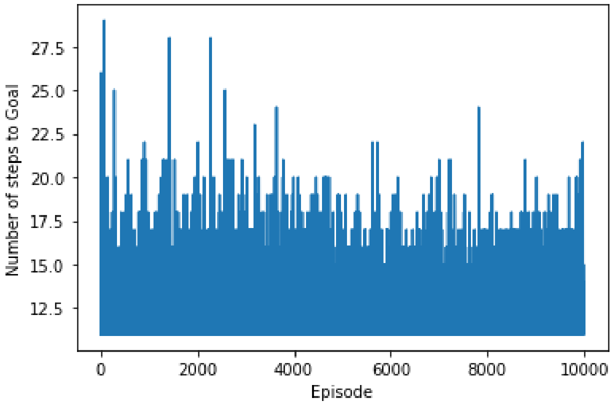

Figure 41: The number of steps to reach the goal in each episode

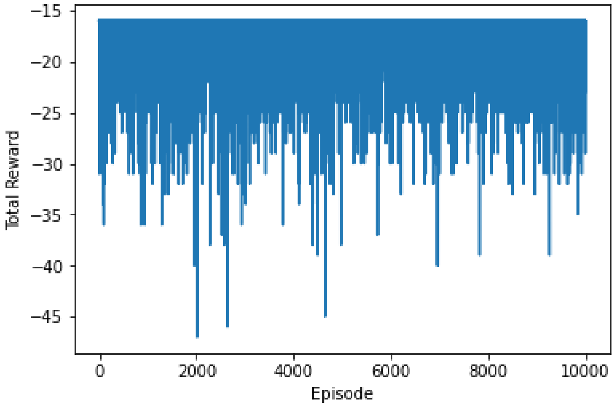

Figure 42: Total Reward

7. start state: (0, 4); p = 0.7; exploration strategies = softmax

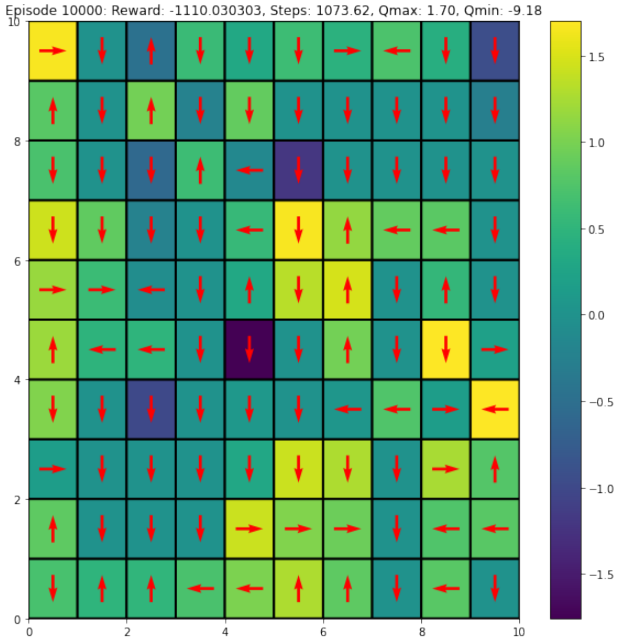

Figure 43: Heatmap of Grid with Q-values

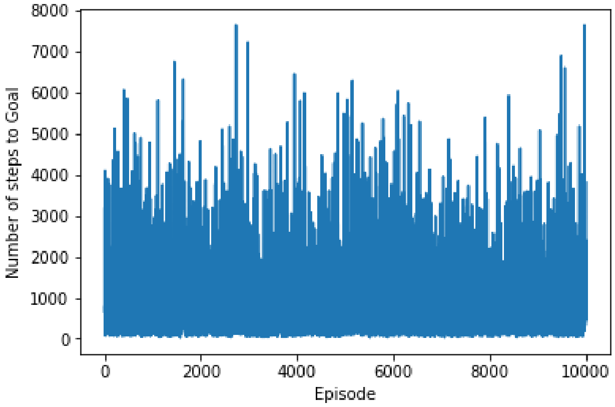

Figure 44: The number of steps to reach the goal in each episode

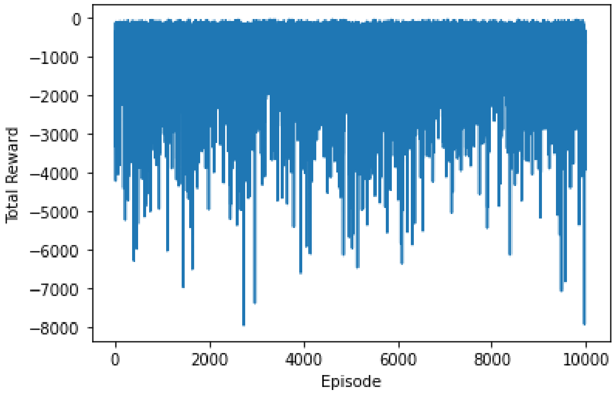

Figure 45: Total Reward

8. start state: (0, 4); p = 0.7; exploration strategies = ϵ-greedy

- Wind = True:

1. start state: (3, 6); p = 1.0; exploration strategies = softmax

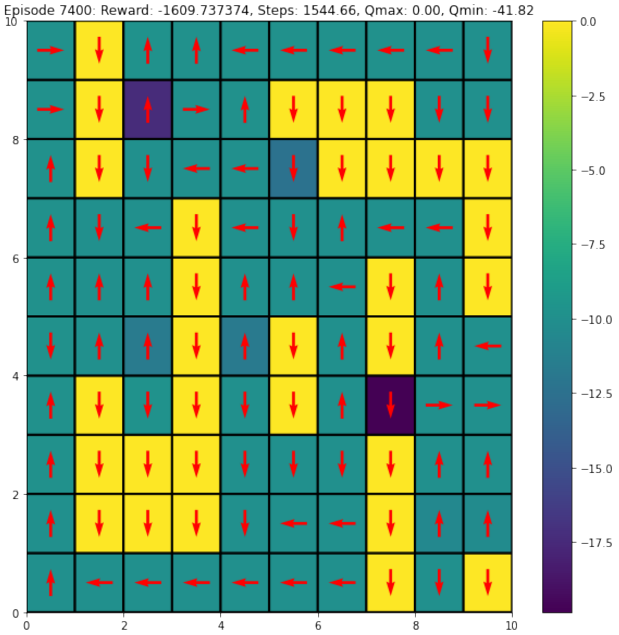

Figure 46: Heatmap of Grid with Q-values

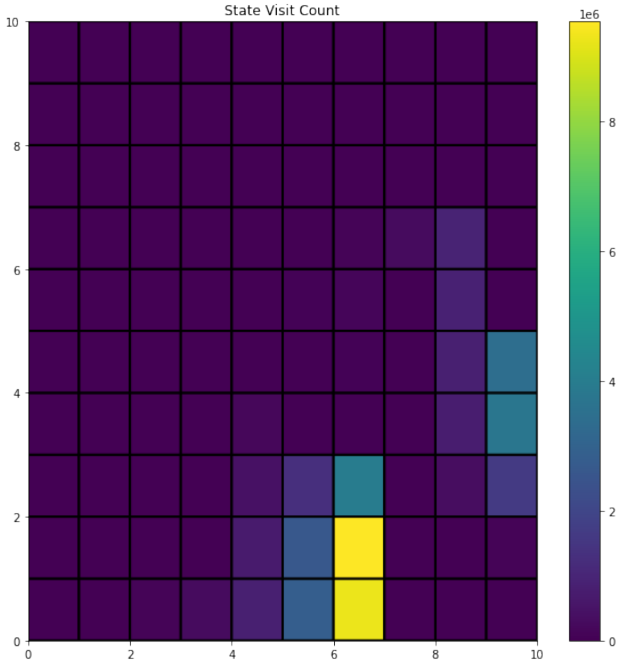

Figure 47: Heatmap of Grid with the state visit counts

Run time error after 7499 episodes.

2. start state: (3, 6); p = 1.0; exploration strategies = ϵ-greedy
3. start state: (3, 6); p = 0.7 exploration strategies = softmax
4. start state: (3, 6); p = 0.7; exploration strategies = ϵ-greedy
5. start state: (0, 4); p = 1.0; exploration strategies = softmax

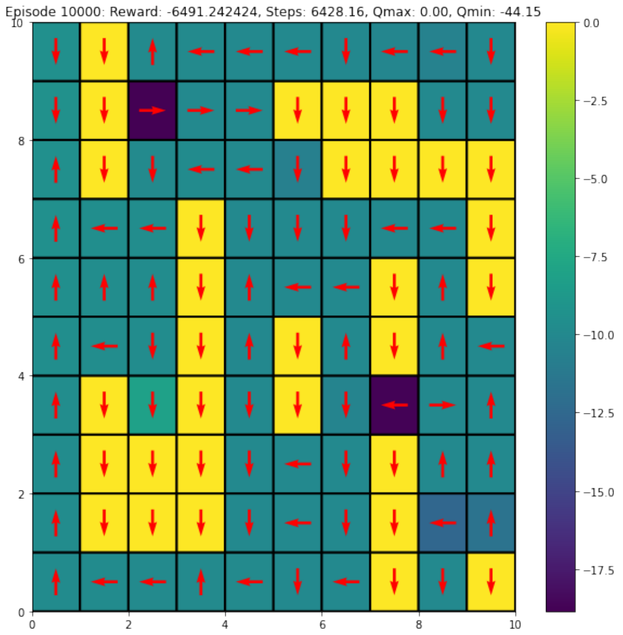

Figure 48: Heatmap of Grid with Q-values

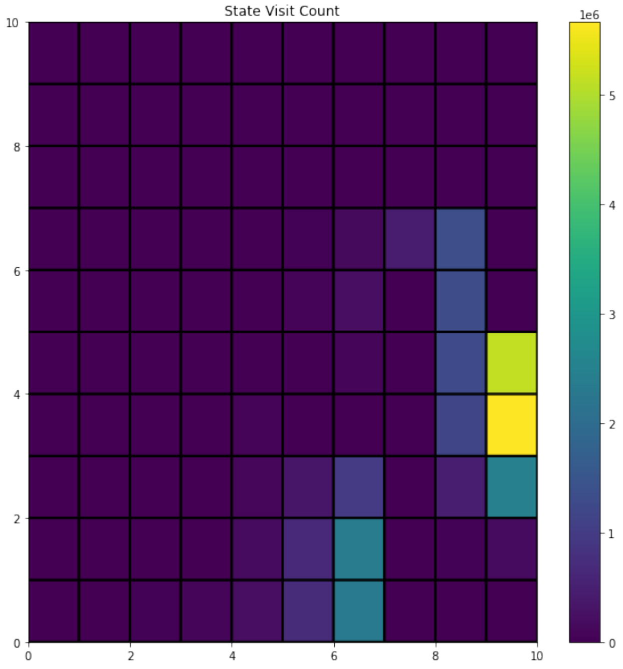

Figure 49: Heatmap of Grid with the state visit counts

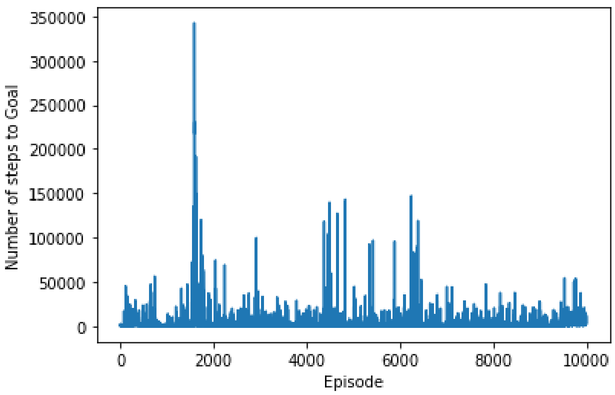

Figure 50: The number of steps to reach the goal in each episode

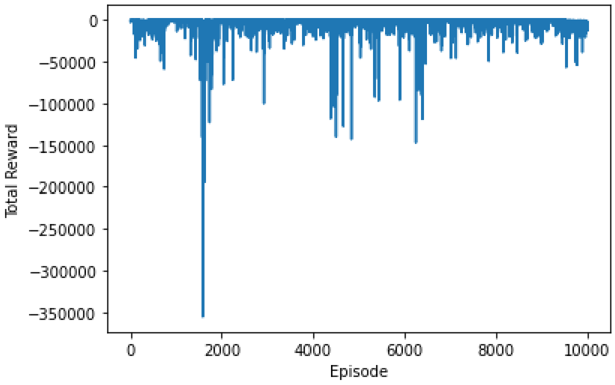

Figure 51: Total Reward

6. start state: (0, 4); p = 1.0; exploration strategies = ϵ-greedy
7. start state: (0, 4); p = 0.7; exploration strategies = softmax
8. start state: (0, 4); p = 0.7; exploration strategies = ϵ-greedy

4 Conclusions

- The performance of both algorithms deteriorates as we increase stochasticity in form of agent transition probability and wind.
- This report above used the following set of hyperparameters:
  1. α (learning rate) = 0.4
  2. γ (discount factor) = 0.9
  3. ϵ: For epsilon greedy exploration = 0.1
  4. β: For Softmax exploration (temperature) = 1
     This could have been tuned to get better performance.

References

[1] Richard S. Sutton and Andrew G. Barto, *Reinforcement Learning: An Introduction* , The MIT Press (1 January 1998).

[2] *Q-Learning*, `https://en.wikipedia.org/wiki/Q-learning`.

[3] *SARSA*, `https://en.wikipedia.org/wiki/State-action-reward-state-action`.

[4] *Tutorial 4* `CS6700 Tutorial-4 Q-Learning and SARSA Code`