CSE 546: Reinforcement Learning
Assignment 1

Part 1

Sumeet Milind Aher
Department of Computer Science
UB ID: 50418795
sumeetmi@buffalo.edu

Abstract

This is a submission for Assignment 1 for the course of Reinforcement Learning.

1 Environment(Deterministic and Stochastic)

1.1 Set of Actions

There are 4 allowed actions- a) Up, b) Down, c) Left, d) Right

1.2 State

There are 8 states in this environment:

a) Rivendell: This is where the elves will help our agent recover in the journey.

b) Aragorn: This is where our heir to the throne named Aragorn will help our agent reach the goal

c) Gollum: This is where the agent meets Gollum, a wild and deserted creature living from ages in search for the ring and he will stop the agent.

d) Mountains of Mordor: These are very hard volcanic terrains filled with goblins and orcs making it very hard and risky for our agent to reach the ring.

e) Door of Moria: This door leads the agent to an ancient hidden dwarf maze, it's the door to go anywhere on the map. But, beware! Nobody knows these tunnels, so the agent doesn't know where he will come out from once, he arrives on this state. The agent is thrown on a random location in the environment once it lands on this state.

f) Gandalf: This state is where the agent meets Gandalf and is blessed by his wisdom!

g) Ring: This state is the terminal state where the agent will find the ring and complete his quest.

1.3 Rewards

Each of the state have their own rewards:

a) Rivendell: +2

Figure 1: Rivendell

b) Aragorn: +2

Figure 2: Aragorn

c) Gollum: -2

Figure 3: Golum

d) Mountains of Mordor: -4

Figure 4: Mordor

e) Door of Moria: 0

Figure 5: Moria

f) Gandalf: +4

Figure 6: Gandalf

g) Ring: 20

Figure 7: Ring

2 Main Objective

This is the environment of Lord of the Rings. Our agent (Frodo Baggins) wants to reach to the Ring so he can destroy it. But the ring is far far away in the land of Mordor. To reach, he must take help from the elves at Rivendell, the heir to the throne Aragorn and Gandalf the White wizard. There is also a mystical door which will randomly take the agent anywhere in the environment. And our agent must stay away from Gollum and the mountains of Mordor.

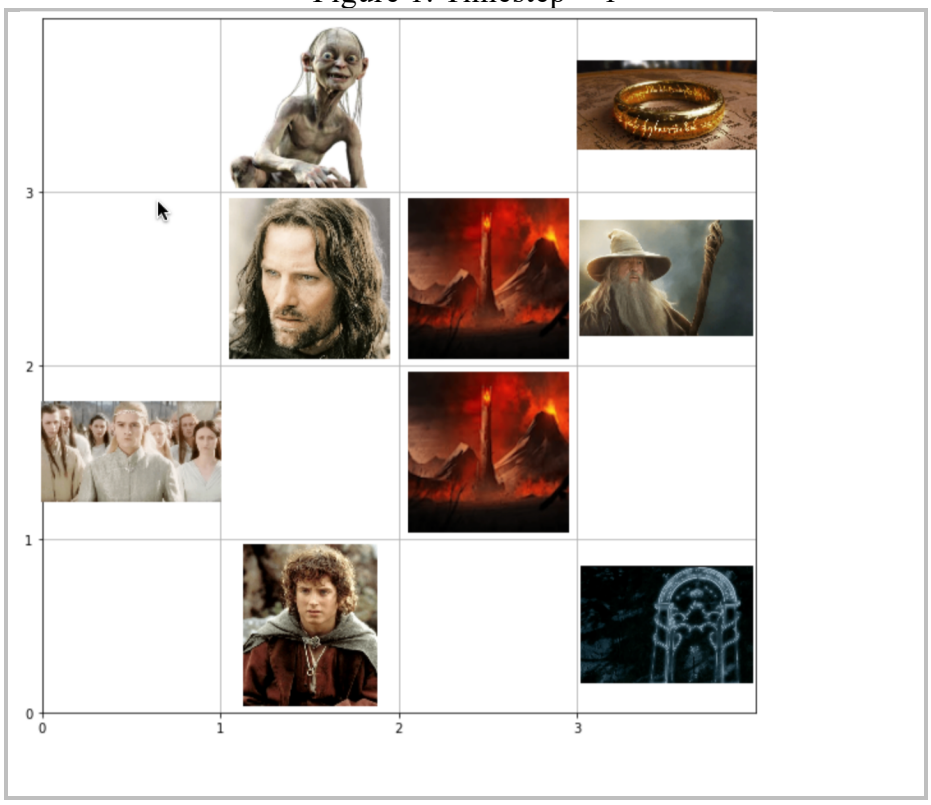
Agent: Frodo Baggins: Our agent is chosen to find the ring in the environment

Figure 1: Agent

3 Visualization

These are some of the timesteps in the environment:

75 Figure 1: Timestep = 1



76 Figure 1: Timestep = 2

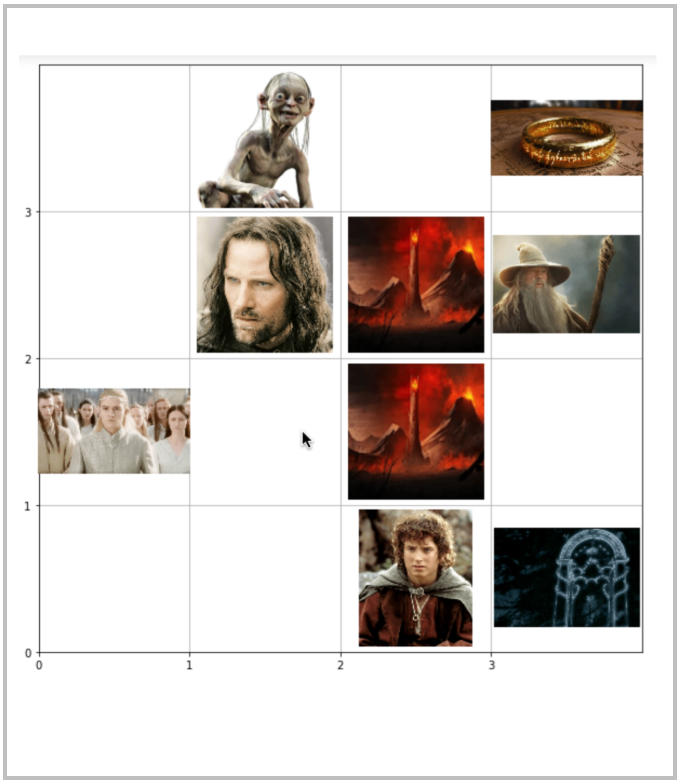


Figure 1: Timestep = 3

4 How did you define the stochastic environment?

In the stochastic environment, whenever the agent decides to take an action, the probability that the agent will be successful with that action is 95%, in the case of the remaining 5%, the agent will choose the UP action instead.

4.1 What is the difference between the deterministic and stochastic environments?

In the deterministic environment, the agent would go right if the action decided is right! And similarly for other actions, the probability of execution of that action is 100%. On the other hand, in a stochastic environment, the probability of execution of a particular action is 95% and in the rest 5%, the agent will have to choose the UP action..

4.2 Safety in AI: Write a brief review explaining how you 99 ensure the safety of your environments.

These are the safety features of my environment:

4.2.1)Whenever the agent takes an action and ends up outside the environment, the code will clip the result between the limits of the grid length and width thus keeping the agent in the environment.

4.2.2) Because of 5.1, the agent might keep stepping outside the grid from a rewarding location, and because of our safety feature, it will again land on the same rewarding location, thus keep on collecting the reward, to stop this, there is an additional safety feature which will give the reward only once.

Part 2

5 Results

All the results are for
alpha =0.15
gamma = 0.95
episodes = 500
steps = 10
epsilon max = 10
epsilon min = 0.01

5.1) a) Applying Q-learning to solve the deterministic environment defined in Part 1. Plots should include epsilon decay and total reward per episode.

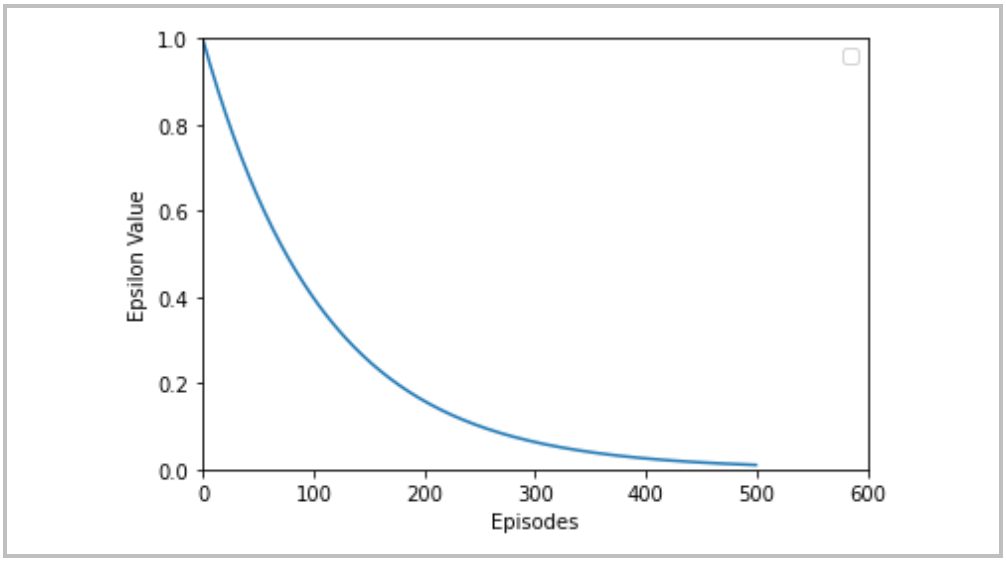


Figure 1: Epsilon Value vs Episode

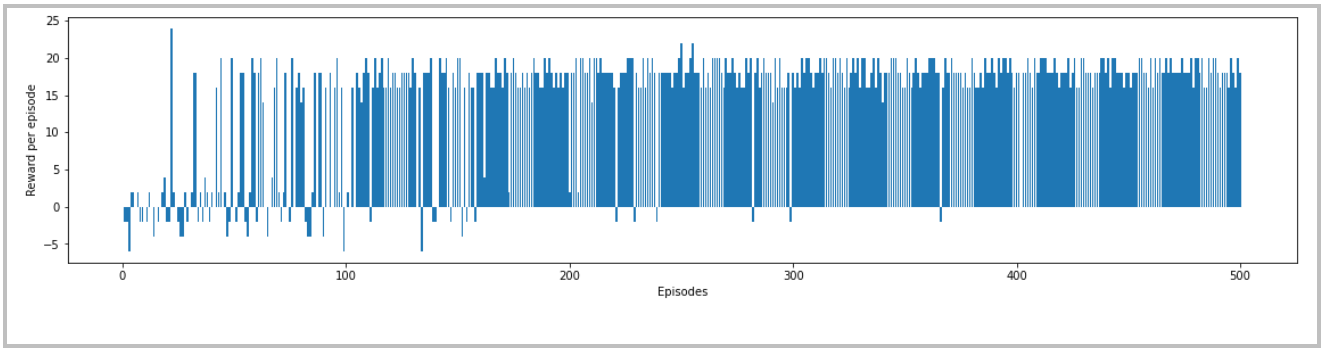


Figure 2: Reward per episodes vs Epsiode for Q Learning Deterministic env

- b) Applying Q-learning to solve the stochastic environment defined in Part 1. Plots should include epsilon decay and total reward per episode.

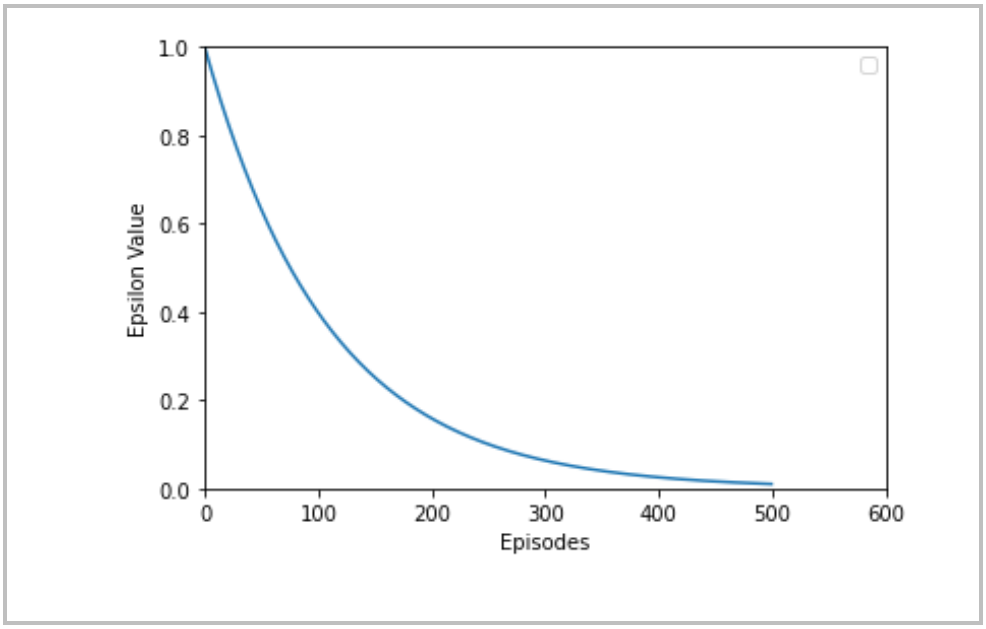


Figure 3: Epsilon decay vs Episode

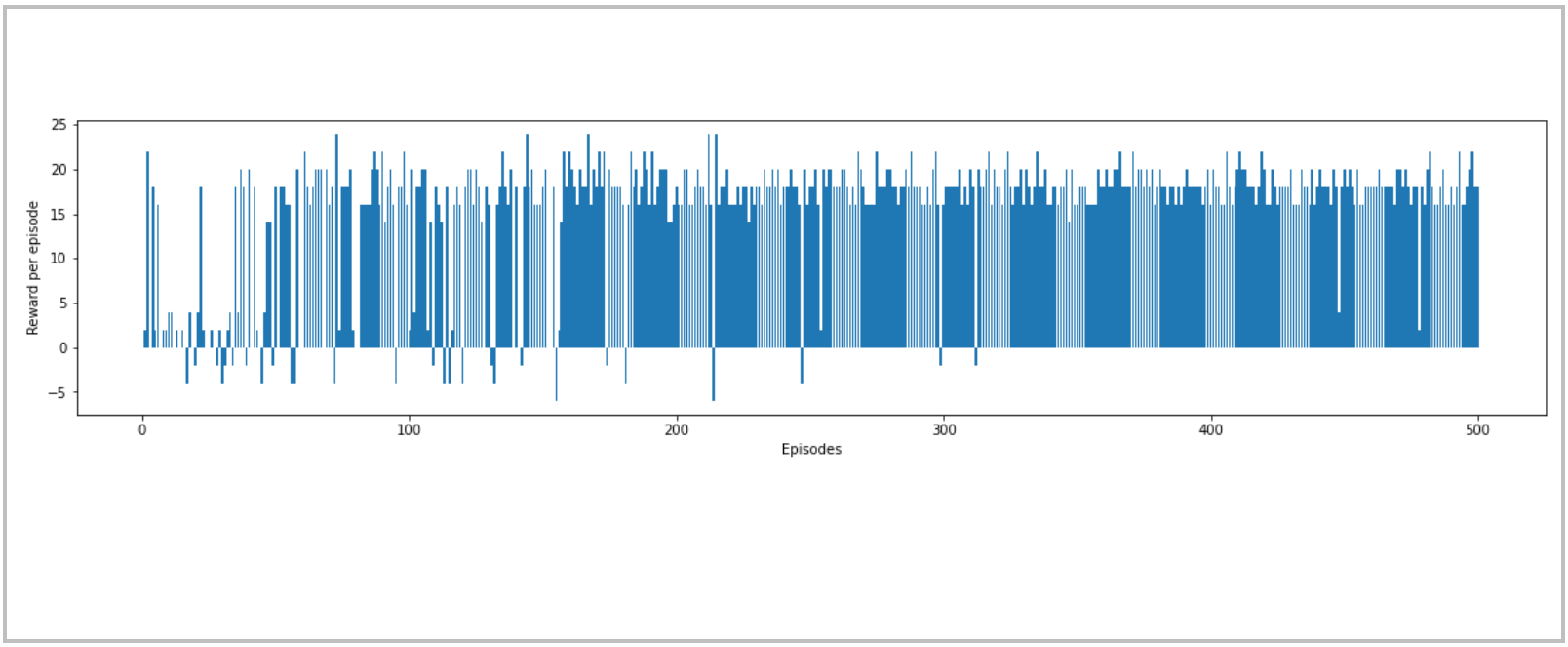


Figure 4: Reward per episode vs Episode For Q Learning Stochastic

- c) Applying any other algorithm of your choice to solve the deterministic environment defined in Part 1. Plots should include total reward per episode.

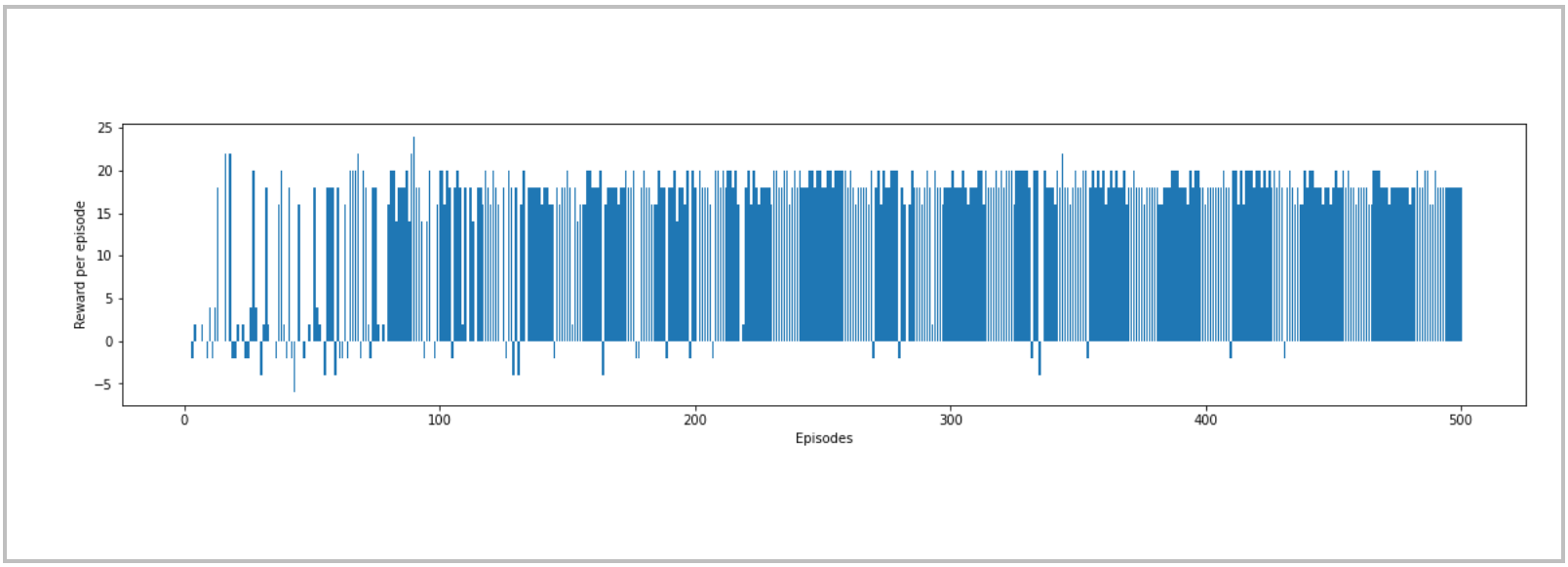


Figure 5: Total Reward vs Episode For SARSA Deterministic env

- d) Applying any other algorithm of your choice to solve the stochastic environment defined in Part 1. Plots should include total reward per episode.

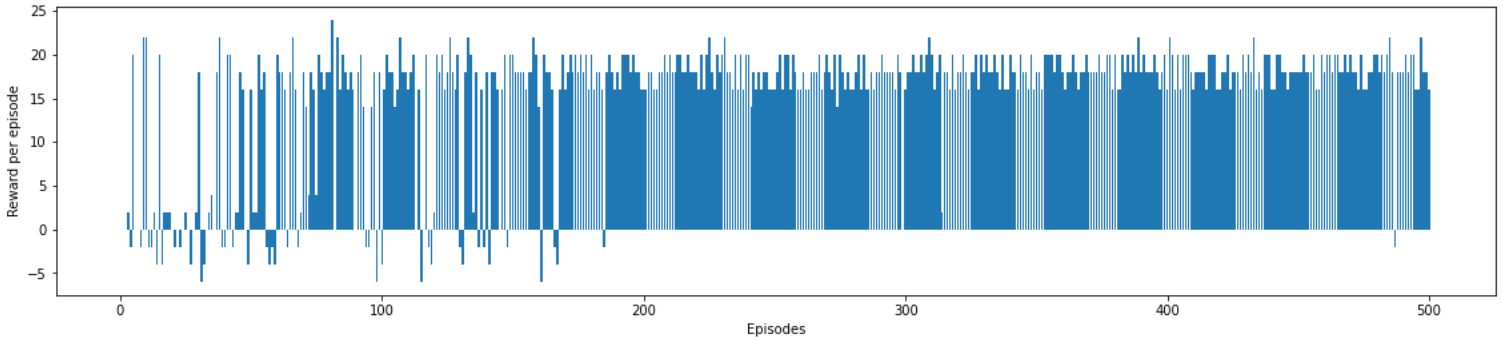


Figure 6: Reward per episodes vs episodes for SARSA Stochastic

- e) Provide the evaluation results. Run your environment for at least 10 episodes, where the agent chooses only greedy actions from the learnt policy. Plot should include the total reward per episode.

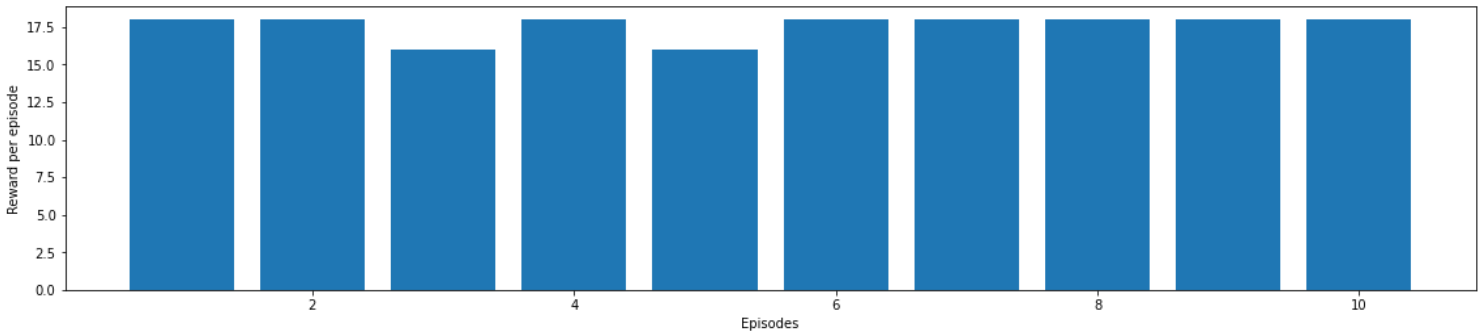


Figure 7: Reward per episode vs Episodes (Deterministic env | Q learning)

Discussion: Stochastic training takes more time both in SARSA as well as Q learning compared to deterministic environment as the action determined through the policy may or may not be executed.

5.2) Compare the performance of both algorithms on the same deterministic environment (e.g. show one graph with two reward dynamics) and give your interpretation of the results.

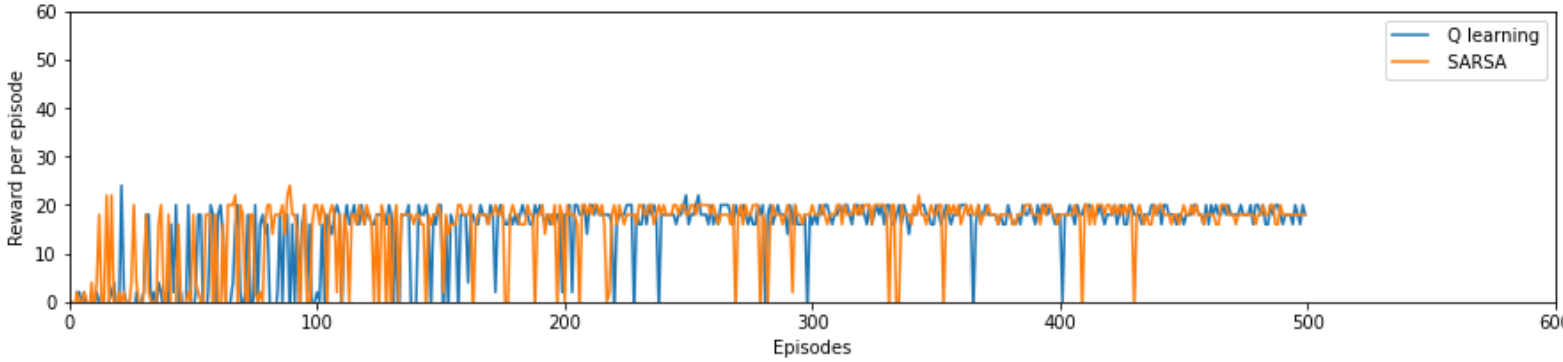
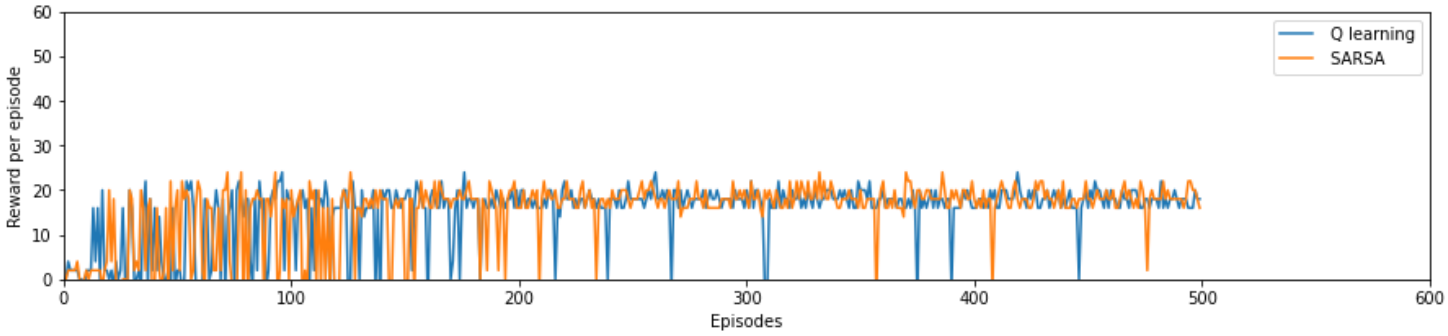


Figure 8: Reward per episode vs Episodes Deterministic env

Discussion: It looks like Q learning is learning faster in this scenario and combination of parameters a it is taking more reward per episode on average.

5.3) Compare how both algorithms perform in the same stochastic environment (e.g. show one graph with two reward dynamics) and give your interpretation of the results.



6.
7.

Figure 9: Reward per episode vs Episodes for Stochastic env

Discussion: It looks like SARSA is learning faster in this scenario and combination of parameters a it is taking more reward per episode on average than Q learning.

5.4) Briefly explain the tabular methods, including Q-learning, that were used to solve the problems. Provide their update functions and key features.

5.4)1. Q -learning – is a model free algorithm which update the value function based on Bellman equation

5.4)2. Update Functions: To calculate the maximum expected future rewards for action at each state , we update the q-value for each state, action pair by adding the learning rate times the target minus the predicted q value.

1. **Q-Learning:**

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha(r_{t+1} + \gamma max_a Q(s_{t+1}, a) - Q(s_t, a_t))$$

5.4)3. Key Features: Q learning is off policy so it learns the optimum value policy independent of agents actions.

5.5) Sarsa – It's a slight variation of Q learning in such a way that it is an on policy algorithm such that the agent learns the value function according to action derived from current policy.

5.5)1. Update Functions: To calculate the maximum expected future rewards for action at each state , we update the q-value for each state, action pair by adding the learning rate times the target minus the predicted q value.

2. **SARSA:**

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha(r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t))$$

5.5)2. Key Features:SARSA stands for STATE ACTION REWARD STATE ACTION, which is literally how the q value is update which is also seen from the equation where the present state, value pair is update based on the expected state action pair obtained from the current policy.

5 Hyperparameter Tuning

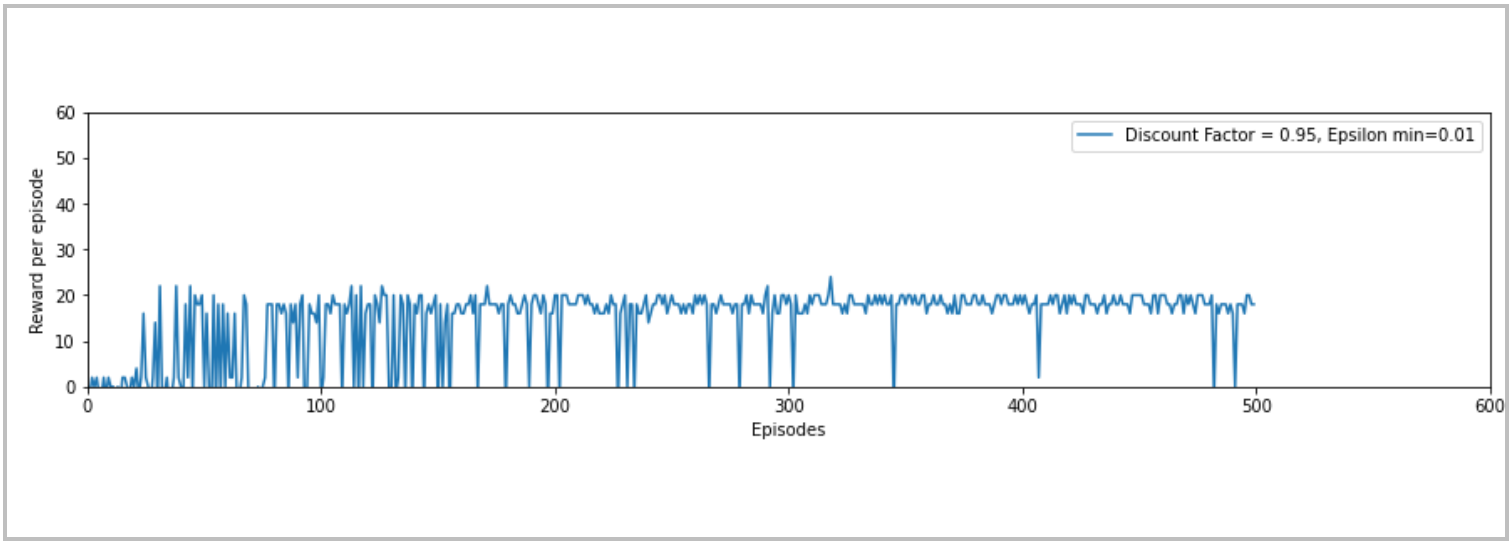
a) Discount factor [0.95, 0.9, 0.8]

c)Epsilon min values [0.01, 0.1, 0.2]

Lets compare the 9 combinations of these hyperparamters:

All the results are for Episodes = 500, steps =10

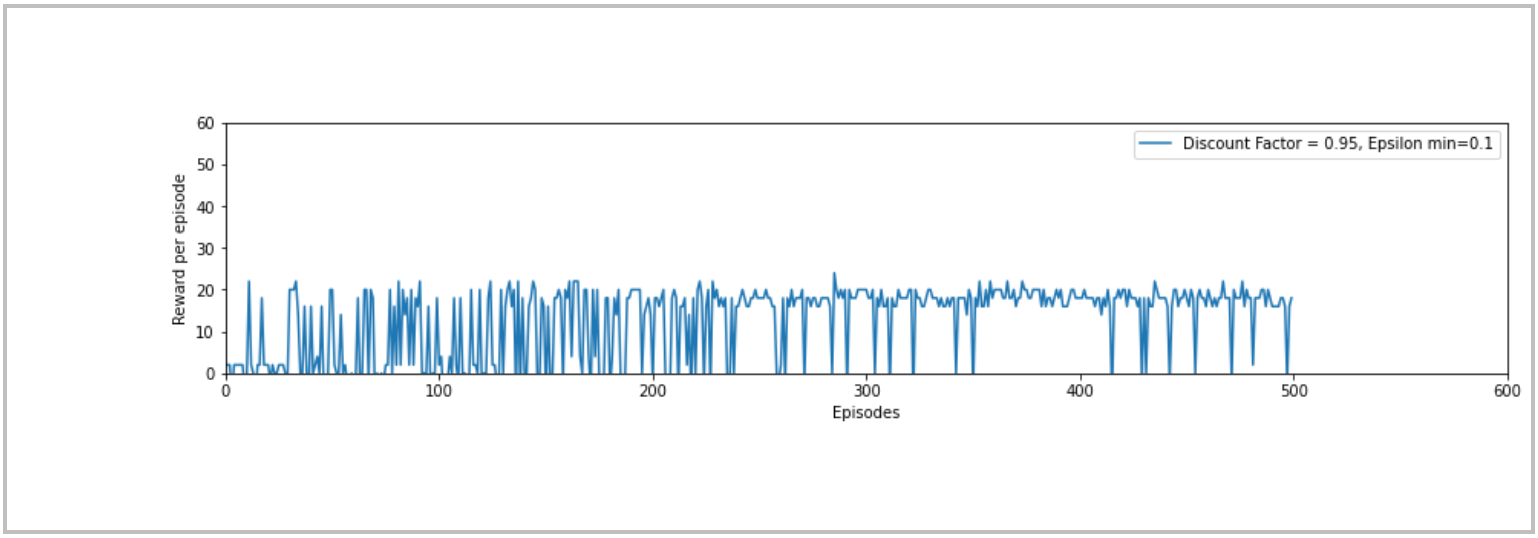
1. Discount factor 0.95 , Epsilon min values 0.01



183 Figure 1: Reward per episode vs Episodes

184

185 2. Discount factor 0.95, Epsilon min values 0.1



186 Figure 2: Reward per episode vs Episodes

187

188 3. Discount factor 0.95 Epsilon min values 0.2

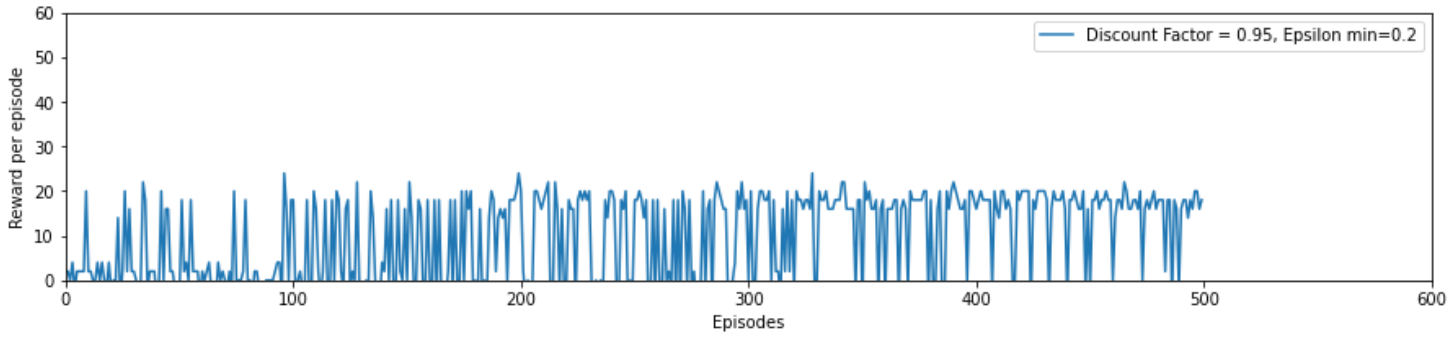


Figure 3: Reward per episode vs Episodes

Discussion: Comparing (1), (2), (3) , the best combination we have is Discount factor = 0.95 and min Epsilon value = 0.01 as it clearly is learning much faster to have the maximum reward per episode

4. Discount factor 0.9 , Epsilon min values 0.01

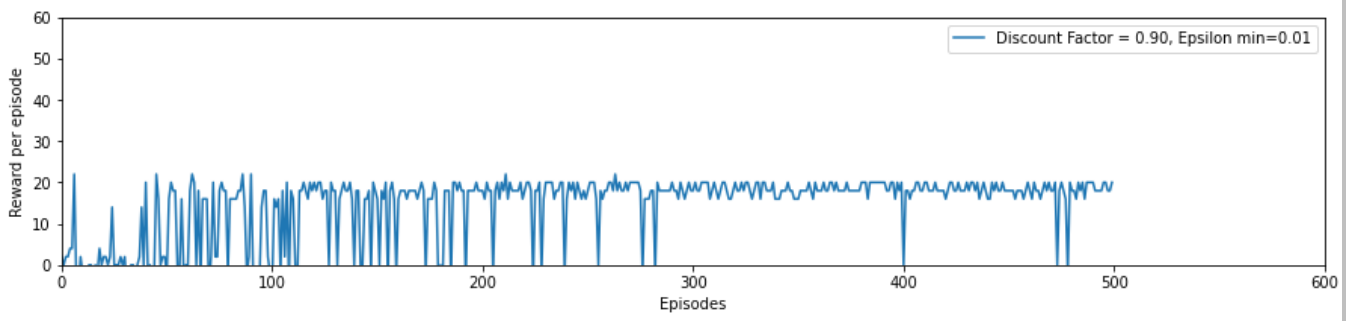


Figure 4: Reward per episode vs Episodes

5. Discount factor 0.9, Epsilon min values 0.1

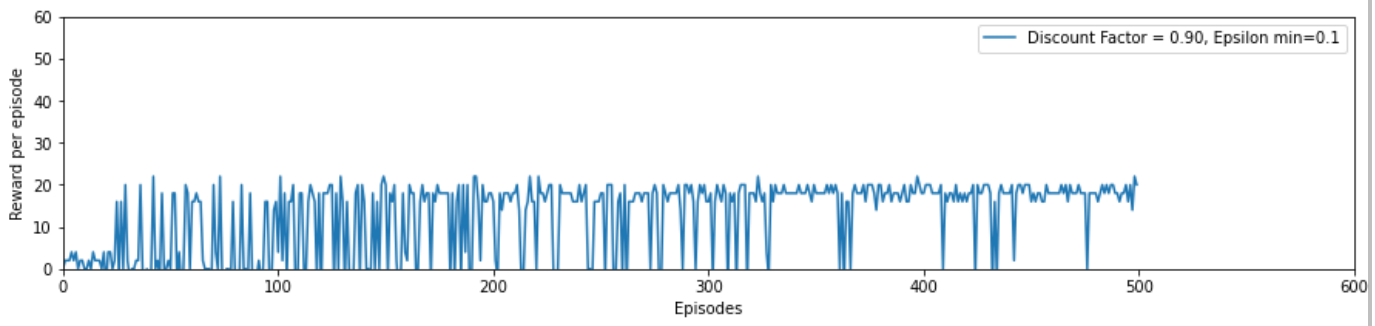


Figure 5: Reward per episode vs Episodes

6. Discount factor 0.9 Epsilon min values 0.2

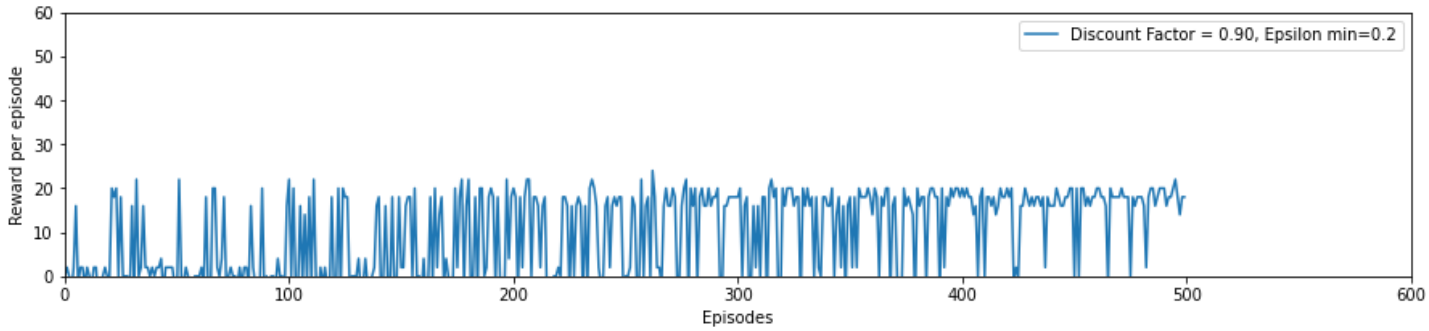


Figure 6: Reward per episode vs Episodes

Discussion: Comparing (4), (5), (6) , the best combination we have is Discount factor = 0.90 and min Epsilon value = 0.01 as it clearly is learning much faster to have the maximum reward per episode

7. Discount factor 0.9 , Epsilon min values 0.01

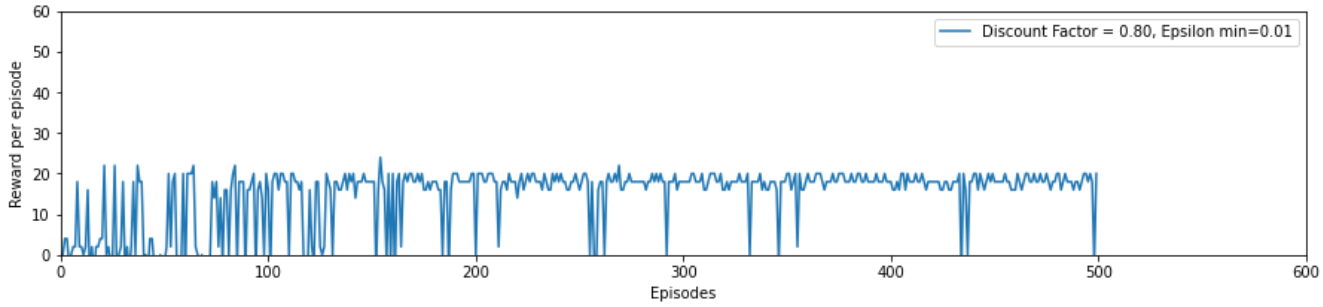


Figure 7: Reward per episode vs Episodes

8. Discount factor 0.9, Epsilon min values 0.1

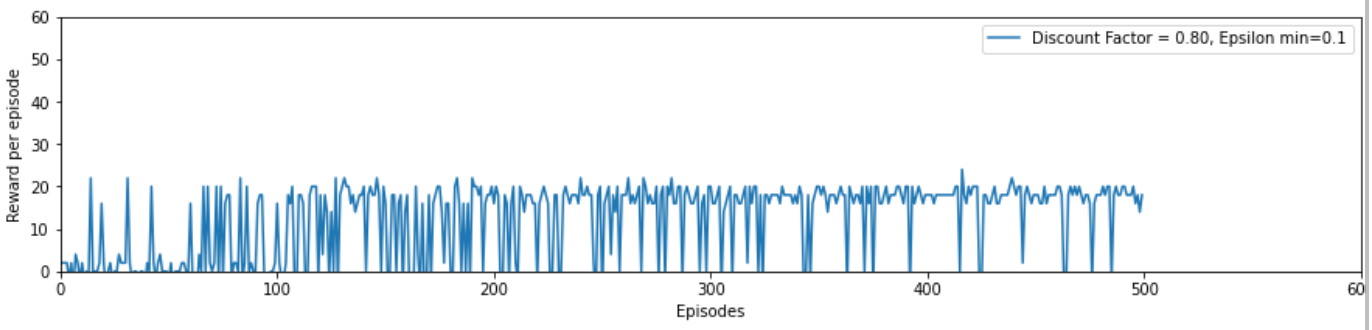


Figure 8: Reward per episode vs Episodes

9. Discount factor 0.9 Epsilon min values 0.2

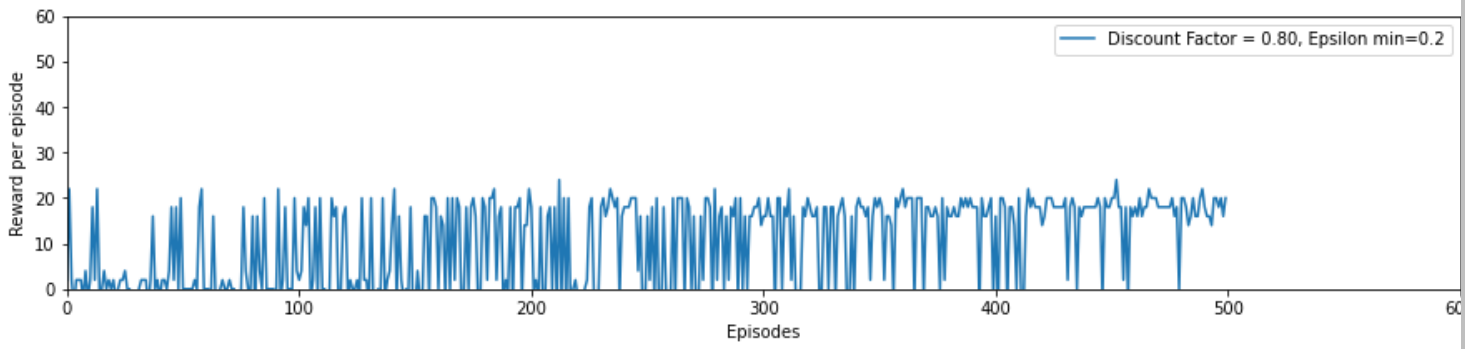


Figure 9: Reward per episode vs Episodes

Discussion: Comparing (7), (8), (9) , the best combination we have is Discount factor = 0.80 and min Epsilon value = 0.01 as it clearly is learning much faster to have the maximum reward per episode

Conclusion: The best combination is Discount factor = 0.95 and min Epsilon value = 0.01 which clearly shows a very fast learning of obtaining maximum reward per episode aming all the 9 combinations.

References

https://gym.openai.com/