Reinforcement Learning Lab Report

Assignment 5

Gudepu Venkateswarlu (212011003)
Jayanth S (201081003)
Praveen Kumar N (201082001)
Rishabh Roy (201082002)

1 Algorithms analysed

- Q-learning with Tile Coding function approximation
- SARSA algorithm with Tile Coding function approximation
- Q-learning with Radial Basis function approximation
- SARSA algorithm with Radial Basis function approximation
- Reinforce without Baseline
- Reinforce with Baseline
- Deep Q-learning
- Actor Critic algorithm

2 Environments Considered

2.1 Mountain Car

2.1.1 Environment Description

A car has to travel in a one-dimensional (horizontal) track with the road being similar to a valley between mountains on either side of it. The goal is to reach the top of right side mountain starting in the valley.

- *State space* : 2-D states representing (position, velocity)
- *Action space* :
  1. Push left (accelerate left)
  2. Push right (accelerate right)
  3. Do nothing
- *Reward* :
  - -1 for any action taken
  - 0 on reaching the goal

2.2 Cart Pole

2.2.1 Environment Description

We need to balance a pendulum (pole) upright placed on a cart by applying either force to the left or right of the cart.

- *State space* : 4-D states representing (Cart position, Cart velocity, Pole Angle, Pole Angular velocity)
- *Action space* :
  1. Push cart to left
  2. Push cart to right
- *Reward* :
  - + 1 for each step taken

2.2.2 Additional Information of environment

- *Initial state* : All the values are sampled in the range $(-0.05, 0.05)$ uniformly.
- *Termination* :
  - Pole Angle is greater than ± 12.
  - Cart Position is greater than ± 2.4.
  - Episode length is greater than 200 (we have considered version $v0$)

3 Observations

- We used 16 tiles and 16 tilings with integer hash table size of 8192 for Cart-pole environment for SARSA and Q-learning algorithms with tile coding.
- We used 16 tiles with 8 tiles with integer hash table size of 4096 for Mountain Car environment for SARSA and Q-learning algorithms with tile coding.

- Reinforce with and without baseline did not work for mountain car environment even with several changes in neural network architecture.
- We ran the actor-critic algorithm on $LunarLander - v2$ environment. We were not able to make Deep-Q algorithm work as expected in $LunarLander - v2$ environment.

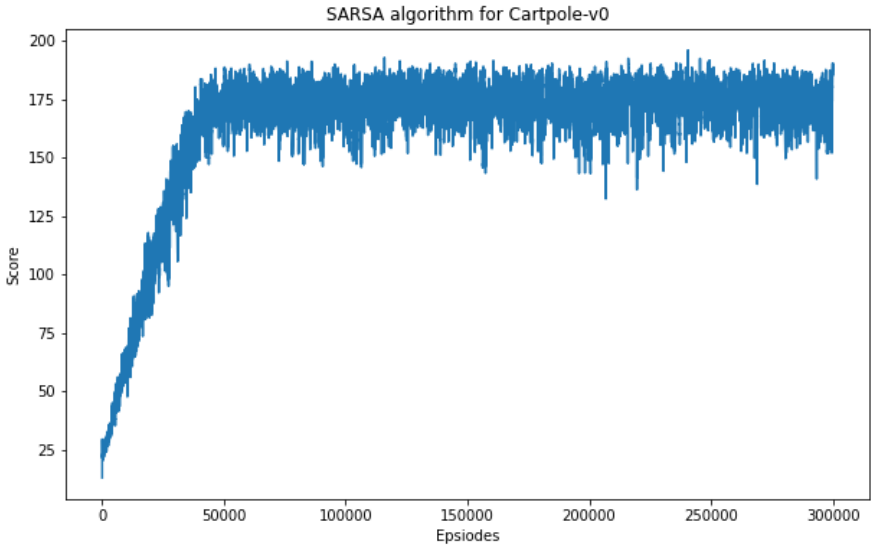

Figure 1: Running average of reward obtained for Cartpole environment using SARSA algorithm with tile coding

4 References

- Open.ai gym environments
- Machine Learning with Phil
- Q-Learning with Value Function Approximation Solution
- Q-Learning with Value Function Approximation

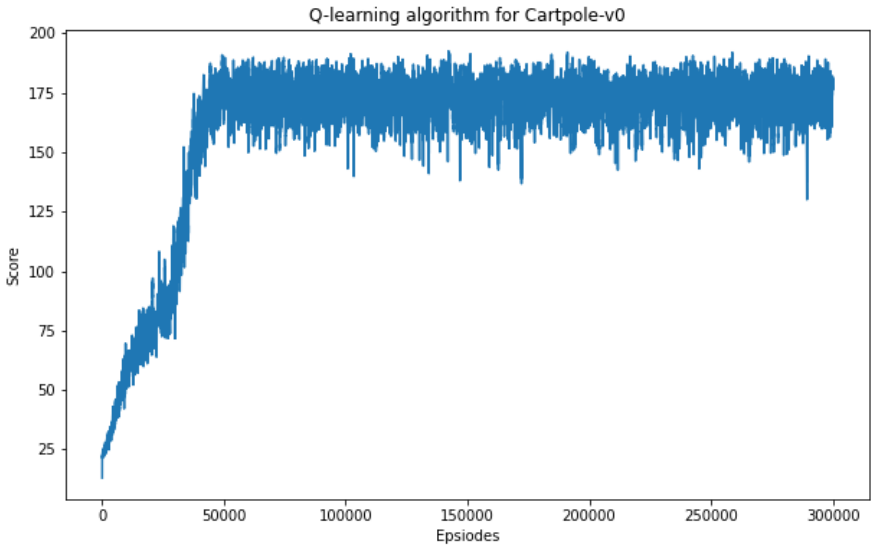

Figure 2: Running average of reward obtained for Cartpole environment using Q-learning algorithm with tile coding

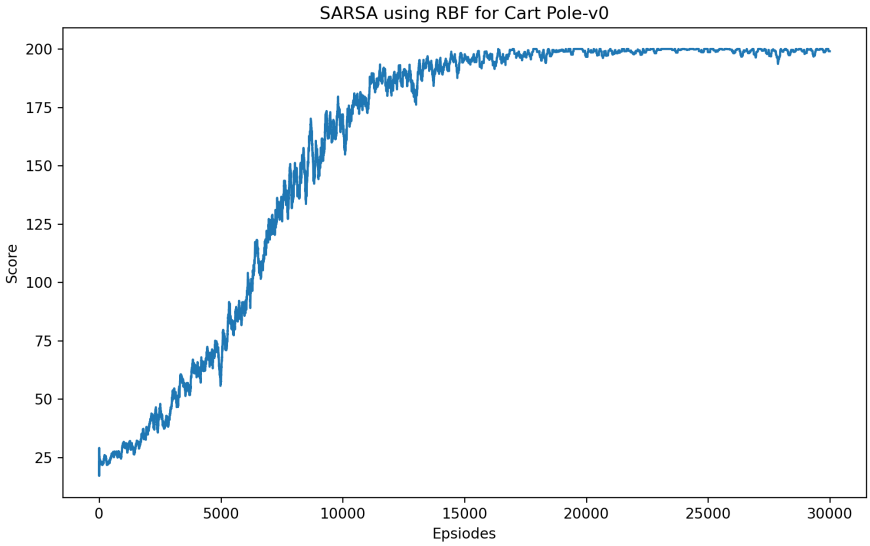

Figure 3: Running average of reward obtained for Cartpole environment using SARSA algorithm with radial basis function approximation

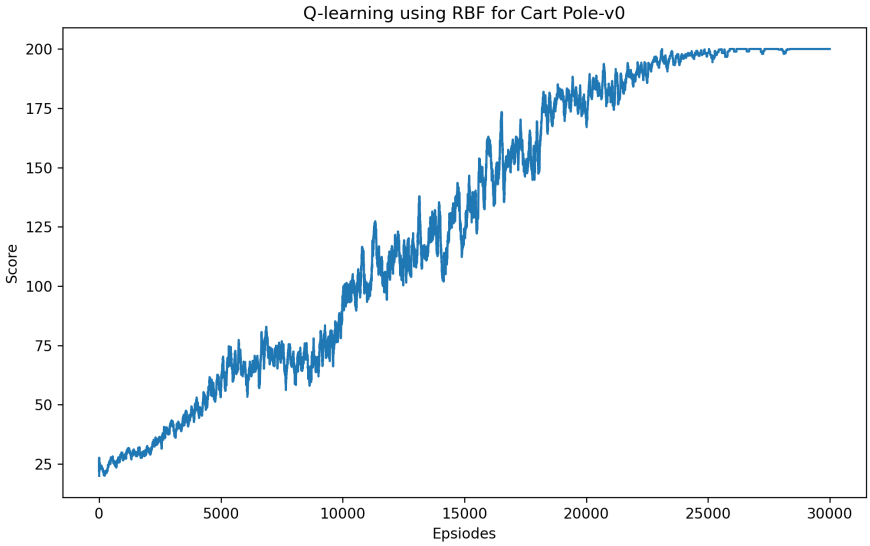

Figure 4: Running average of reward obtained for Cartpole environment using Q-learning algorithm with radial basis function approximation

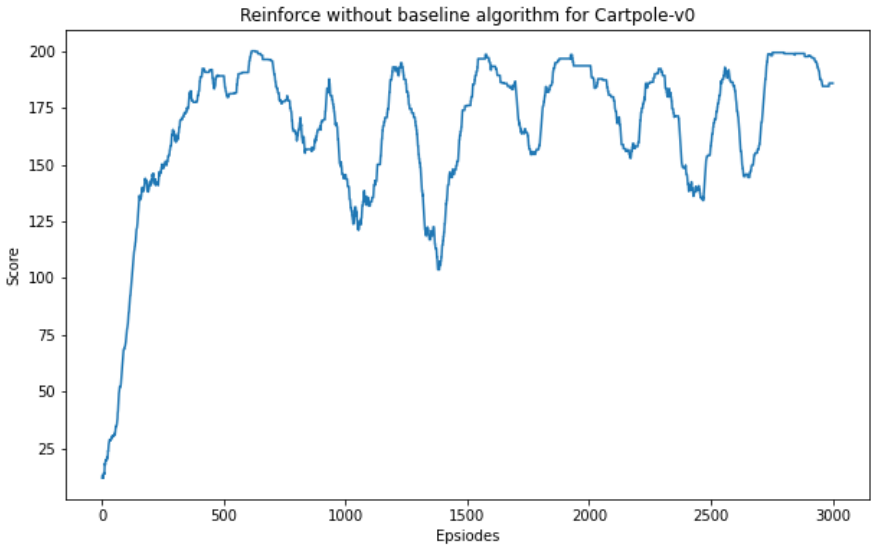

Figure 5: Running average of reward obtained for Cartpole environment using reinforce without baseline algorithm

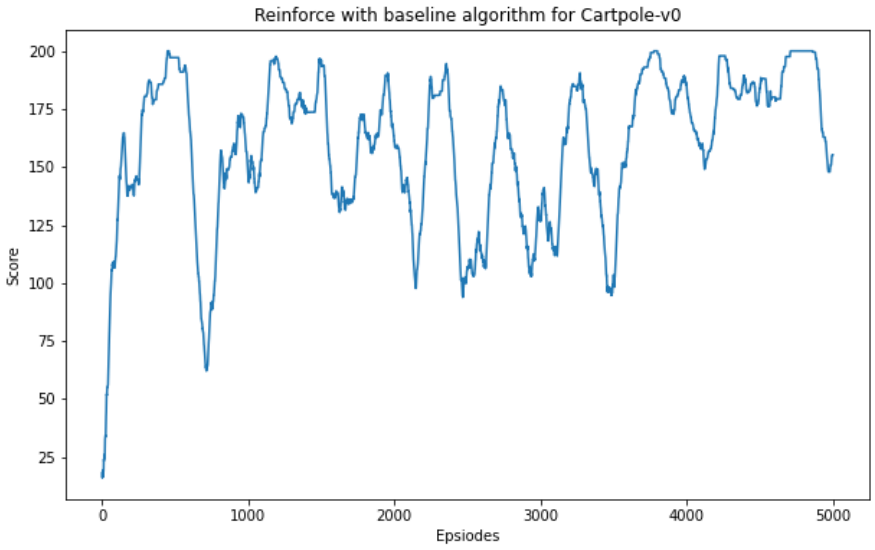

Figure 6: Running average of reward obtained for Cartpole environment using reinforce with baseline algorithm

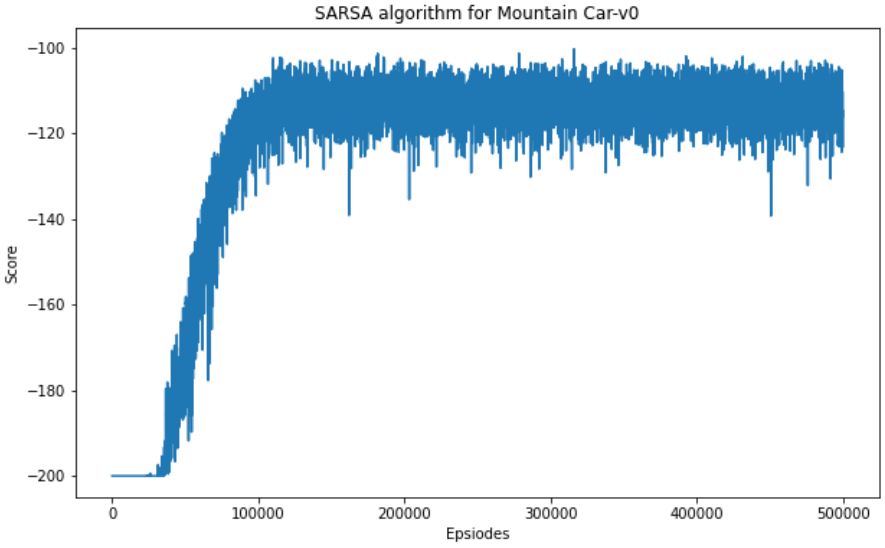

Figure 7: Running average of reward obtained for Mountain Car environment using SARSA algorithm with radial basis function approximation

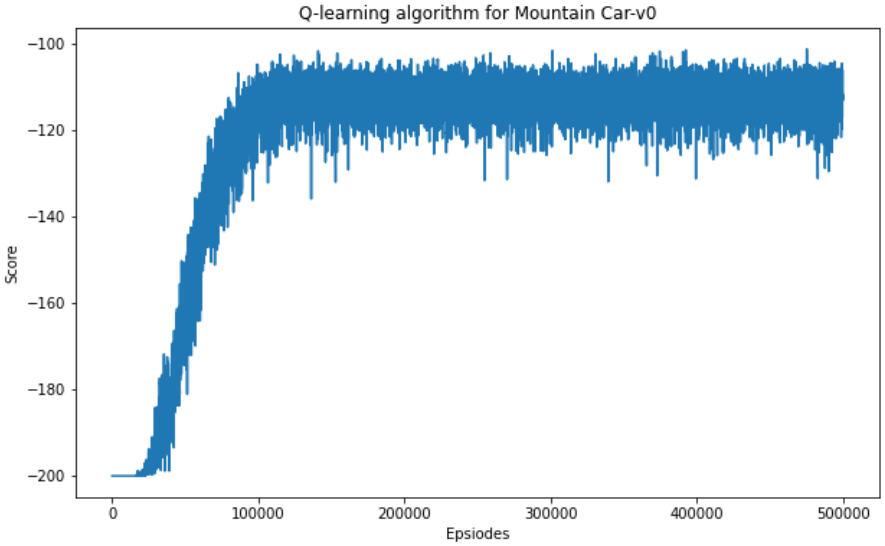

Figure 8: Running average of reward obtained for Mountain Car environment using Q-learning algorithm with radial basis function approximation

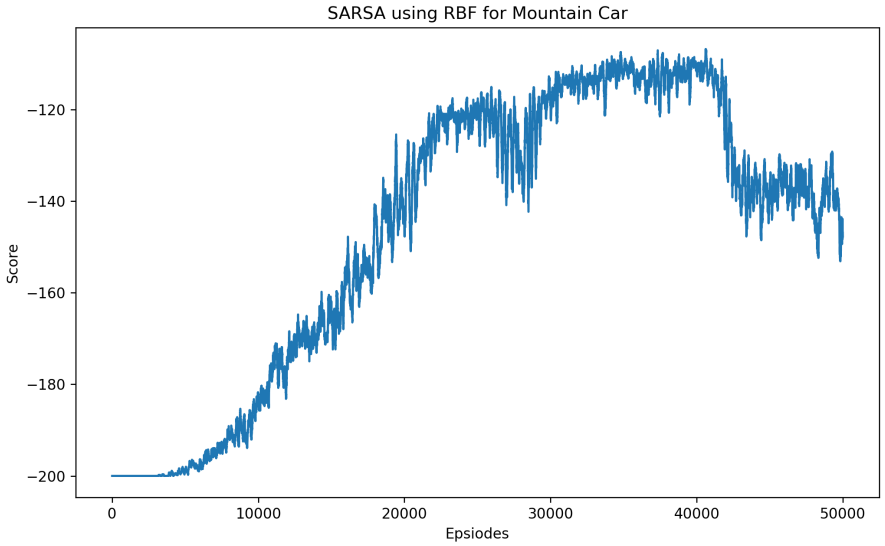

Figure 9: Running average of reward obtained for Mountain Car environment using SARSA algorithm with radial basis function approximation

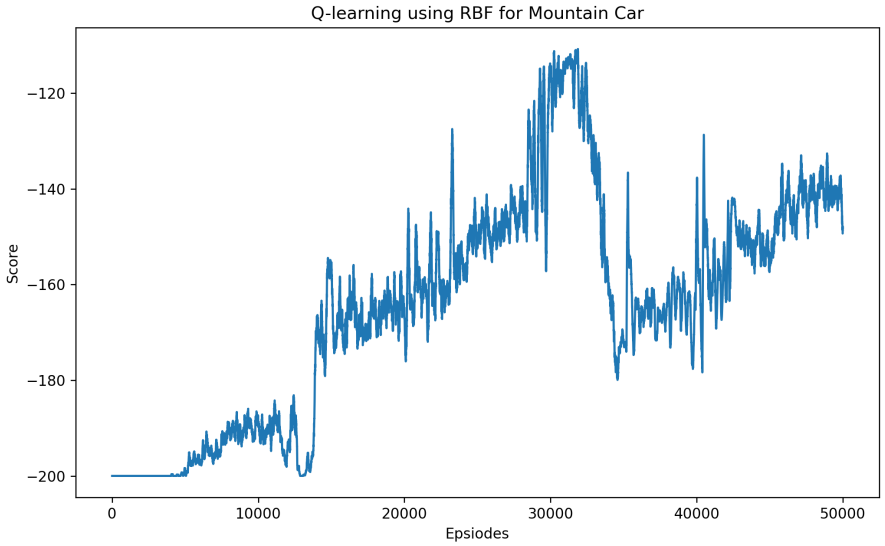

Figure 10: Running average of reward obtained for Mountain Car environment using Q-learning algorithm with radial basis function approximation

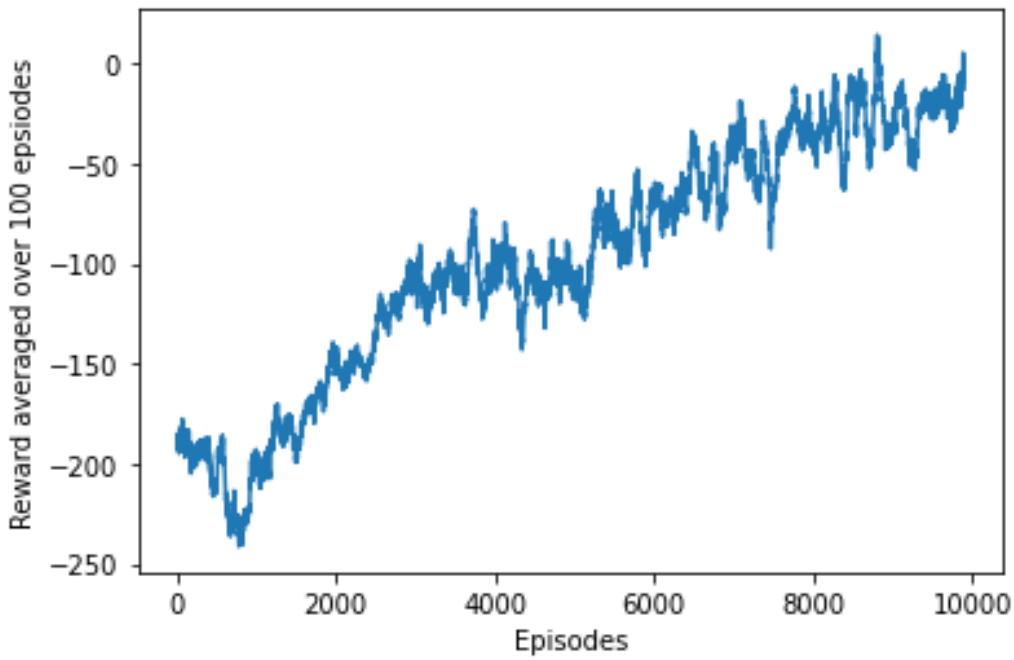

Figure 11: Running average of reward obtained for Lunar Lander environment using Actor-Critic agent.