CS747 - Assignment 3 Report

Shashank Kumar

November 13, 2020

Directory Structure

- *gridworld.py* - File to simulate the environment
- *sarsa.py* - Implementation fof Sarsa(0) agent
- *qlearning.py* - Code for q-learning agent
- $sarsa_expected.py - Agent following expected sarsa policy plot.py - Script to simulate the results$
- *helper.py* - A file for additional tasks like plotting
- *plots/* - Directory containing the plots

Implementation Details

GridWorld

GridWorld class has been implemented to simulate the environment. It has functions to generate wind, computing the next state based on the current state, action and wind, and for providing rewards to the agent. The class takes the following arguments - number of rows and columns in the grid, start state and end state as tuples, seed for the random number generator, and a boolean argument stochastic which determines if the wind is supposed to be stochastic or deterministic.

Rewards

For every step taken, a reward of -1 is provided unless the next state is end state, in which case a reward of 100 is provided to the agent. The task is implemented as an episodic task and completes if the end state is reached.

Agent

The agent implementations extend the GridWorld class. The agent determines next action based on the current state. In the three implementations i.e. Sarsa(0), Q-learning and Expected-Sarsa, epsilon-greedy policy is used to choose next action. Run function is used to simulate the agent interaction with the environment for given timesteps. To simulate King's move, number of actions should be increased to 8 in the agent initialization.

Hyperparameters

The learning rate (alpha) and epsilon are set to be 0.1.

Plots

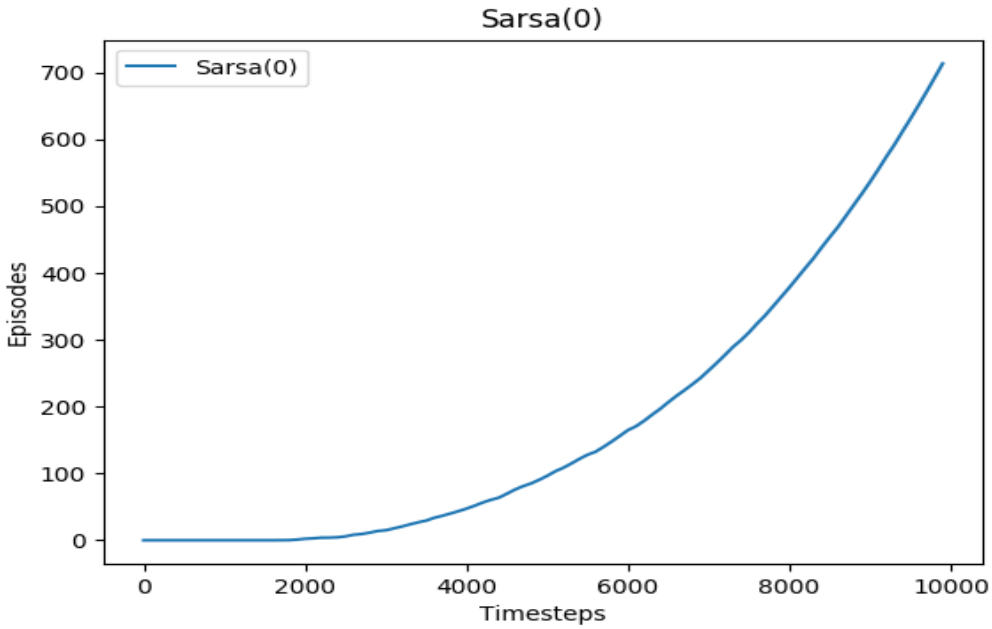

Figure 1: Sarsa(0) agent without King's move and deterministic wind

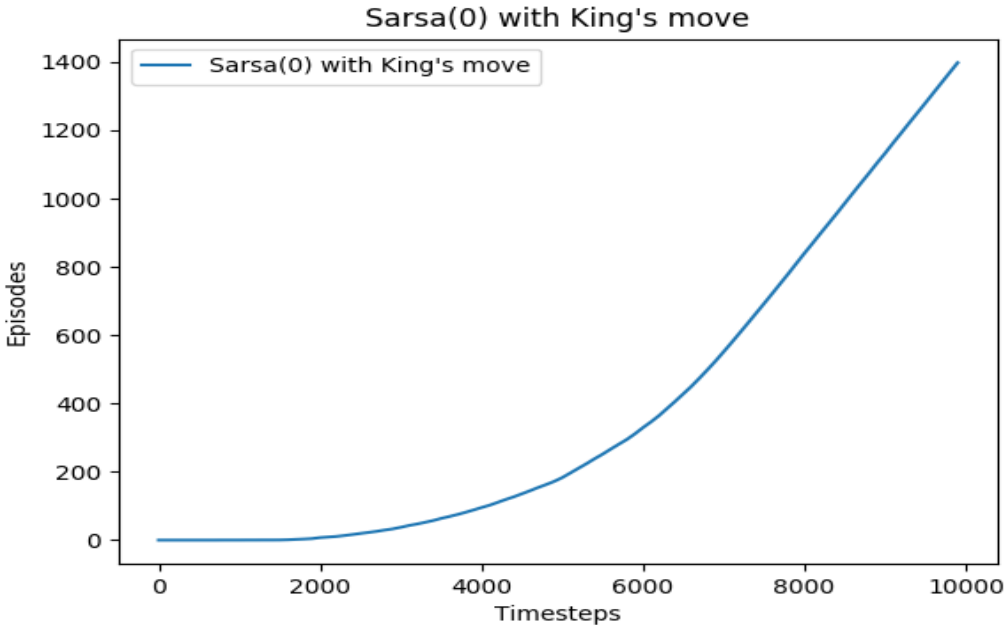

Figure 2: Sarsa(0) agent with King's move and deterministic wind

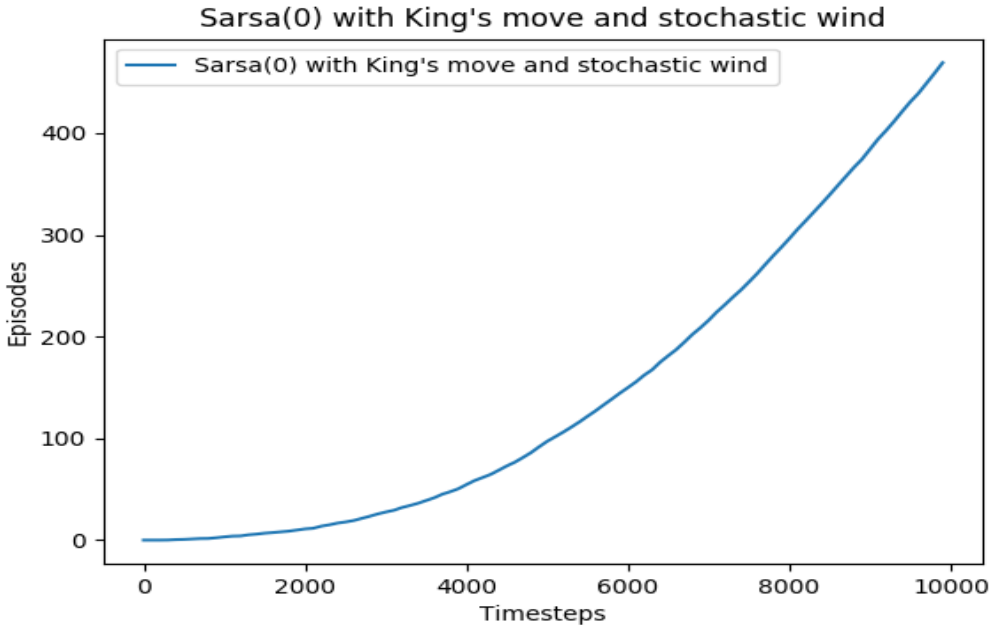

Figure 3: Sarsa(0) agent with King's move and stochastic wind

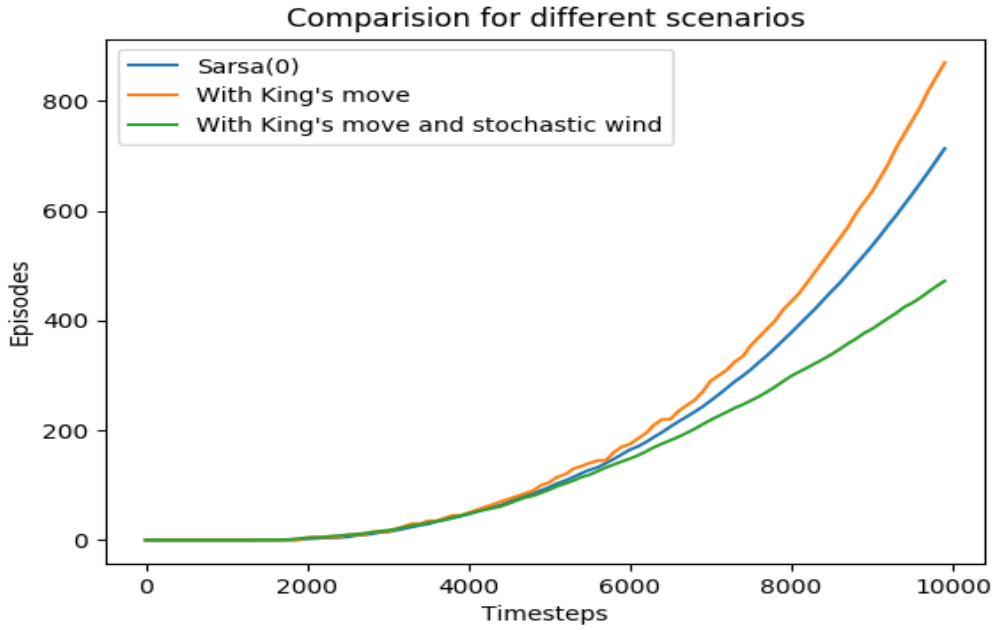

Figure 4: Comparison between different scenarios

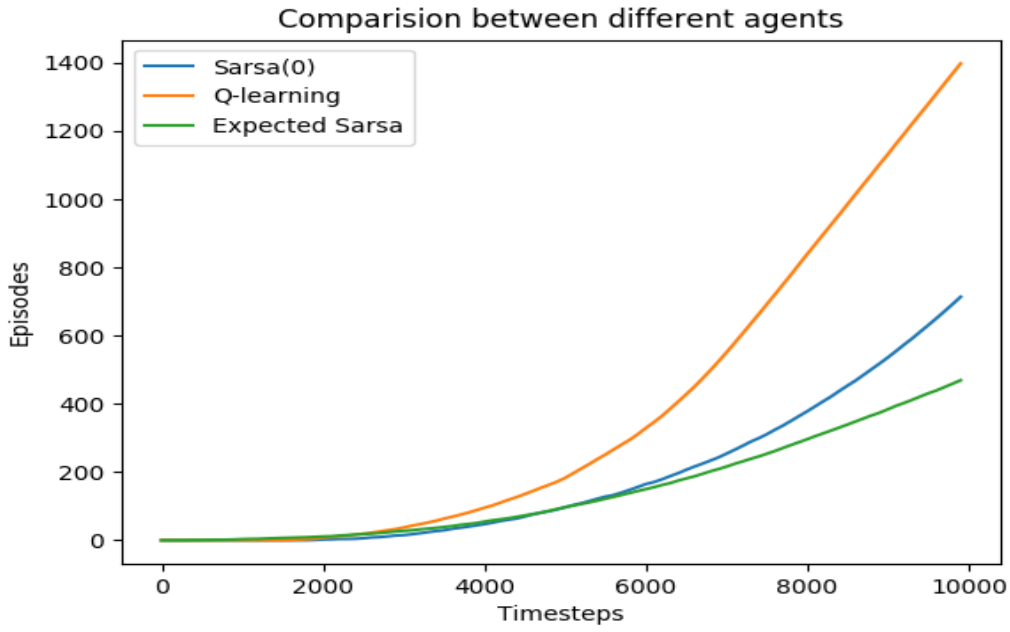

Figure 5: Comparison between agents following different algorithms

Conclusions

Performance comparison for different environment scenarios with Sarsa(0) agent:
Agent with King's move > Normal agent > Agent with King's move and stochastic wind
Performance comparison for different algorithms:
Q-Learning > Sarsa > Expected sarsa

References

[1] Python documentation available at `https://docs.python.org/3/`

[2] Numpy documentation available at `https://numpy.org/doc/`

[3] Matplotlib documentation available at `https://matplotlib.org/api/pyplot`$_{a}pi.html/$
