Reinforcement Learning

SARSA and Q-Learning in Wind Grid Environment

By: Yara Mahmoud Elzahy 20398570

1. Introduction

Reinforcement learning (RL) algorithms, such as SARSA and Q-Learning, have been widely used in solving various problems in artificial intelligence. This report presents a comparative analysis of the performance of SARSA and Q-Learning algorithms in a wind grid environment. Specifically, we compare the effectiveness of these algorithms in two different action spaces: the normal wind grid with 4 actions and the extended wind grid with 8 actions, including king's moves. We analyze the problem formulation, pipeline steps, and provide insights and conclusions based on the experimental results.

2. Problem Formulation

The problem involves finding an optimal policy in a wind grid environment using SARSA and Q-Learning algorithms. The environment consists of a grid world with discrete states and actions. The objective is to maximize the total reward obtained by an agent navigating through the grid world. We evaluate the performance of SARSA and Q-Learning algorithms in two scenarios: one with 4 actions representing the normal wind grid and another with 8 actions, including king's moves.

3. Problem Statement

The specific problem can be stated as follows:

- Given a wind grid environment, find the optimal policy that maximizes the total reward using SARSA and Q-Learning algorithms.
- Compare the performance of SARSA and Q-Learning algorithms in terms of total reward, convergence episodes, and their ability to handle different action spaces (4 actions vs. 8 actions).

4. Pipeline Steps

The following steps were executed to compare the performance of SARSA and Q-Learning algorithms in the wind grid environment:

Step 1: Initialization

- Initialize the wind grid environment.
- Define the SARSA and Q-Learning agents.

Step 2: Hyper-parameter Tuning

- Define a range of alpha and epsilon values for evaluation.
- Initialize variables to store the best alpha, epsilon, and the number of steps for SARSA and Q-Learning in both action spaces.

Step 3: SARSA Algorithm

- Iterate through the alpha and epsilon values.
- Run the SARSA algorithm with the current alpha and epsilon on the wind grid environment for a specified number of episodes.
- Calculate the optimal policy for SARSA.
- Calculate the total reward achieved by SARSA.
- Calculate the number of episodes for convergence to the optimal SARSA policy.
- Update the best alpha, epsilon, and the number of steps if the current result is better.

Step 4: Q-Learning Algorithm

- Run the Q-Learning algorithm with the best alpha and epsilon values obtained from Step 3 in both action spaces.
- Calculate the optimal policy for Q-Learning.
- Calculate the total reward achieved by Q-Learning.
- Calculate the number of episodes for convergence to the optimal Q-Learning policy.

Step 5: Results Analysis

- Print the best alpha, epsilon, and the number of steps for SARSA and Q-Learning in both action spaces.
- Print the total reward and number of episodes for convergence for SARSA and Q-Learning in both action spaces.
- Visualize the optimal policies obtained from SARSA and Q-Learning using grid plots for both action spaces.

- Plot the number of steps per episode for SARSA and Q-Learning to observe their convergence behavior in both action spaces.

5. Insights

i. Normal Grid World (4 actions)

Gamma	Epsilon	Alpha	SARSA number of steps	Q-Learning number of steps
1	0.05	0.1	424	359
1	0.1	0.1	424	430
1	0.15	0.1	400	413
1	0.2	0.1	394	464
1	0.25	0.1	373	414
1	0.05	0.2	225	229
1	0.1	0.2	213	218
1	0.15	0.2	205	226
1	0.2	0.2	223	213
1	0.25	0.2	195	209
1	0.05	0.5	90	94
1	0.1	0.5	91	89
1	0.15	0.5	73	93
1	0.2	0.5	86	94
1	0.25	0.5	82	93

The optimal values are as follows:

for SARSA: Gamma = 1, Epsilon = 0.15, Alpha = 0.5, number of steps = 73, Total reward = -9 and number of episodes to converge = 15

for Q-Learning: Epsilon = 0.1, Alpha = 0.5, steps = 89, reward = -9, convergence episodes = 1

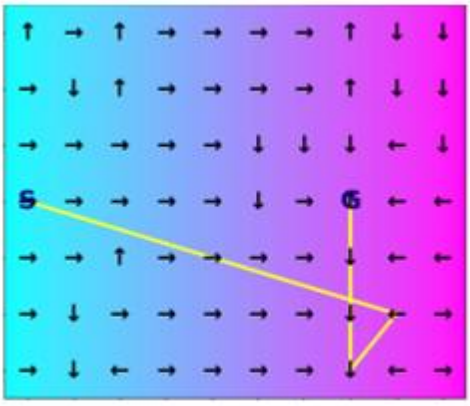

Fig.1.1. Optimal Path for SARSA

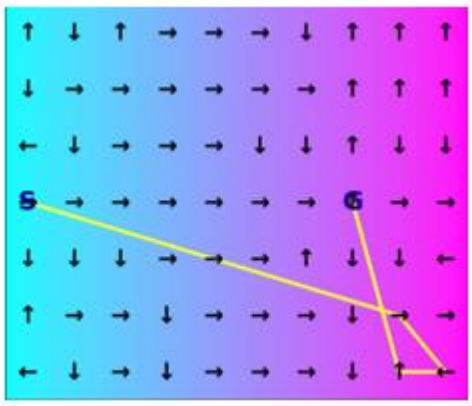

Fig.1.2. Optimal Path for Q-Learning

ii. King's Moves (8 actions)

Gamma	Epsilon	Alpha	SARSA number of steps	Q-Learning number of steps
1	0.05	0.1	1246	1512
1	0.1	0.1	1772	1429
1	0.15	0.1	1090	1496
1	0.2	0.1	1334	1838
1	0.25	0.1	1613	1926
1	0.05	0.2	975	673
1	0.1	0.2	679	965
1	0.15	0.2	696	603
1	0.2	0.2	697	722
1	0.25	0.2	701	658
1	0.05	0.5	347	221
1	0.1	0.5	343	282
1	0.15	0.5	310	389
1	0.2	0.5	377	325
1	0.25	0.5	236	274

The optimal values are as follows:

for SARSA: Gamma = 1, Epsilon = 0.25, Alpha = 0.5, number of steps = 236, Total reward = -12 and number of episodes to converge = 15

for Q-Learning: Gamma = 1, Epsilon = 0.05, Alpha = 0.5, number of steps = 221, Total reward = -8 and number of episodes to converge = 15

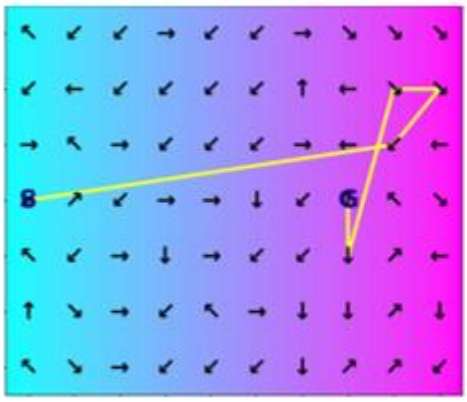

Fig.2.1. Optimal Path for SARSA

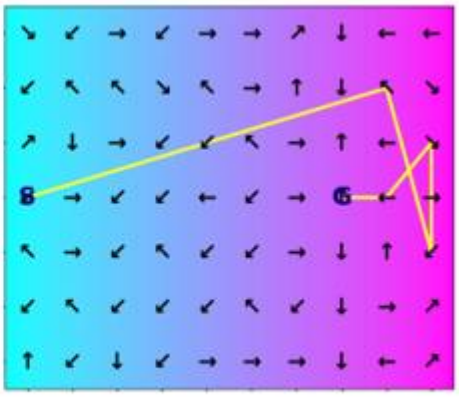

Fig.2.2. Optimal Path for Q-Learning

After conducting experiments on the wind grid environment using SARSA and Q-Learning algorithms in both 4-action and 8-action spaces, the following insights were gained:

- SARSA: The best alpha and epsilon values for SARSA in the normal wind grid (4 actions) were found to be 0.5 and 0.15, respectively, with convergence achieved in 15 episodes. The total reward obtained was -9.
- Q-Learning: The best alpha and epsilon values for Q-Learning in the normal wind grid (4 actions) were also found to be 0.5 and 0.1, respectively, with convergence achieved in 15 episodes. The total reward obtained was -9.
- Comparison between SARSA and Q-Learning:
  - **Total Reward:** Q-Learning outperformed SARSA in both environments.
  - **Convergence:** Q-Learning converged faster in the normal grid world, while both algorithms required the same number of episodes to converge in the king's moves environment.
  - **Number of Steps:** Q-Learning had fewer steps in the normal grid world, while SARSA performed better in the king's moves environment.
- 4 Actions vs. 8 Actions: Q-Learning with 8 actions (including king's moves) achieved a lower total reward and faster convergence compared to the 4-action scenario.

6. Conclusion

Based on the experimental results, Q-Learning outperformed SARSA in terms of total reward and convergence speed in the wind grid environment, while SARSA may exhibit better performance in terms of fewer steps taken. In the normal grid world, Q-Learning demonstrated similar total rewards but faster convergence compared to SARSA. Additionally, the use of 8 actions (including king's moves) in Q-Learning provided better results compared to the 4-action scenario. These findings highlight the importance of selecting suitable RL algorithms and action spaces based on the problem characteristics. Further exploration and analysis can be conducted to validate these findings in different environments and scenarios.