Comparison of SARSA and Expected SARSA on the Taxi Problem

Antoine Voyer 260989010 antoine.voyer@mail.mcgill.ca
Simon Li 260984998 xi.yang.li@mcgill.ca

February 15, 2025

1 Introduction

The purpose of this experiment is to compare the performance of **SARSA** and **Expected SARSA** in the **Taxi-v3** environment from OpenAI's Gym suite. The two algorithms differ in how they estimate future rewards:

SARSA updates its Q-values based on the action actually taken.

Expected SARSA updates based on the expected value of the next state's Q-values under the current policy.

We investigate how their performance is affected by different learning rates and temperature values in softmax exploration.

2 Experimental Setup

The agent operates in a **tabular reinforcement learning setting** with discrete states and actions. The training process follows these steps:

Exploration: Softmax (Boltzmann) exploration is used to select actions based on Q-values.

Hyperparameters:

Learning rate (α): 0.1, 0.3, 0.5, 0.7, 0.9

Temperature (T): 0.1, 0.5, 1.0, 5.0

Discount factor (γ): 0.9

Evaluation Procedure:

Each hyperparameter setting is tested for **10 trials**.

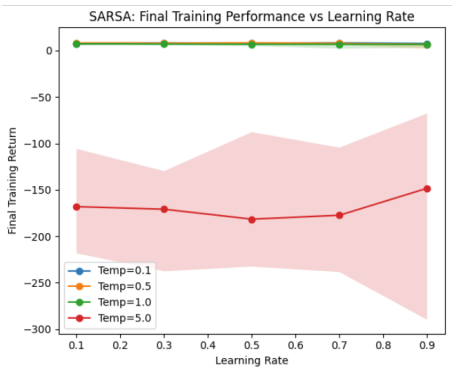
Each trial consists of **500 segments**, each with **10 training episodes** and **1 testing episode**.

The testing episode is run greedily using the best policy learned so far.

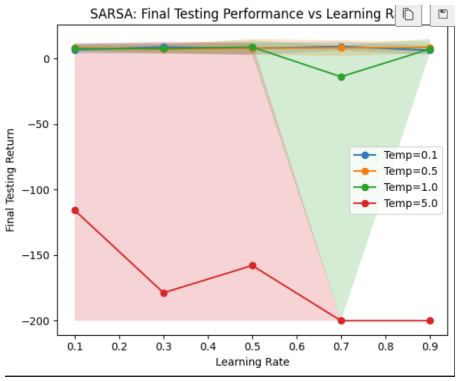
3 Results and Analysis

3.1 Final Training and Testing Performance

Figures 1 and 2 show the effect of different hyperparameter settings on final training and testing performance for SARSA and Expected SARSA, respectively.

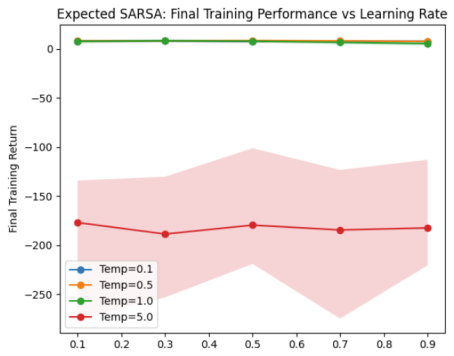


(a) Final Training Performance vs Learning Rate (SARSA)

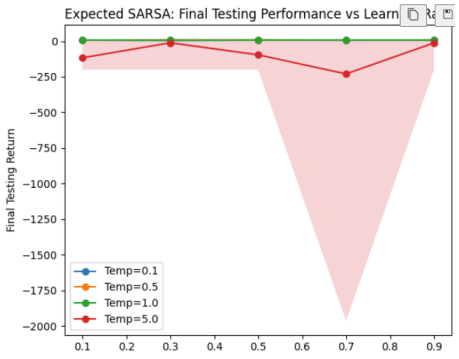


(b) Final Testing Performance vs Learning Rate (SARSA)

Figure 1: Effect of Hyperparameters on SARSA Performance



(a) Final Training Performance vs Learning Rate (Expected SARSA)



(b) Final Testing Performance vs Learning Rate (Expected SARSA)

Figure 2: Effect of Hyperparameters on Expected SARSA Performance

Key Observations:

For SARSA, learning rates between **0.5 and 0.7** with a lower temperature (e.g., 0.1) yield the best performance.

For Expected SARSA, **moderate learning rates (0.3–0.5) and temperatures (0.5–1.0)** give the best results.

High temperatures (T=5.0) lead to significantly worse performance in both cases, likely due to excessive exploration.

3.2 Learning Curves for Best Parameter Settings

Figures 3 and **??** show the learning curves for SARSA and Expected SARSA with their best hyperparameter settings.

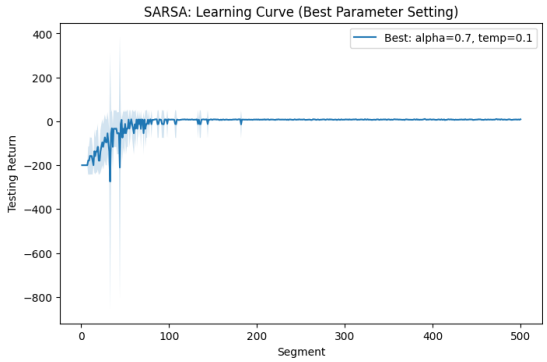


Figure 3: SARSA Learning Curve (Best Parameters: $\alpha = 0.7, T = 0.1$)

Key Observations:

Both algorithms start with high variance in performance but stabilize over time.

SARSA converges slightly faster but Expected SARSA achieves a more stable final performance.

The variance in Expected SARSA is initially higher, but the policy improves more consistently.

4 Mathematical Analysis

4.1 Bellman Equation and Reward Decomposition

We analyze whether we can combine action-value functions for different reward structures linearly. Using the Bellman equation, we show:

- For a fixed policy π, we can express the combined action-value function as a weighted sum:
- For optimal action-value functions, however, linear combination does not generally hold due to the maximization operation:

5 Conclusion

This experiment highlights the trade-offs between SARSA, Expected SARSA, and alternative learning strategies in reinforcement learning. Future work could explore:

Using **epsilon-greedy** instead of softmax exploration.

Extending these ideas to **deep reinforcement learning**.