Comp579 Assignment3

1 Value-based methods with linear function approximation

1.1 MountainCar-v0 task

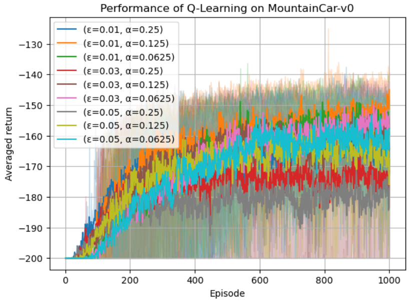


Figure 1: Q-Learning

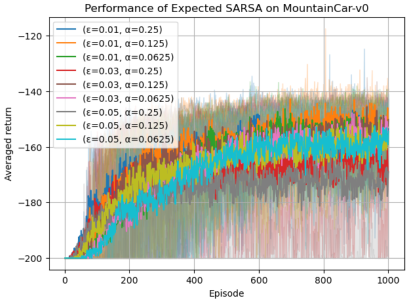


Figure 2: Expected SARSA

- Both algorithms improve performance over time as they learn from the environment. However, Expected SARSA seems to achieve a slightly higher averaged return by the end of 1000 episodes, indicating a potentially better final policy or a more effective learning strategy in this specific task.
- The Expected SARSA appears to show less variance in averaged returns compared to the Q-learning, especially with smaller learning rates (α). This suggests that Expected SARSA might have a more stable learning process over episodes
- For both algorithms, learning rate (α=0.125) performs better than the other two step size, thus it best balances how much new information overrides old information.
- For both algorithms, a smaller exploration rate (ϵ=0.01) appears to balance exploration and exploitation well, as these lines tend to rise towards the higher averaged returns.

1.2 CartPole-v1

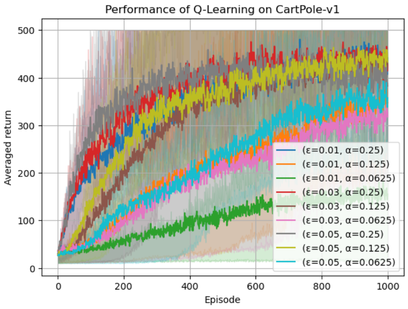


Figure 3: Q-Learning

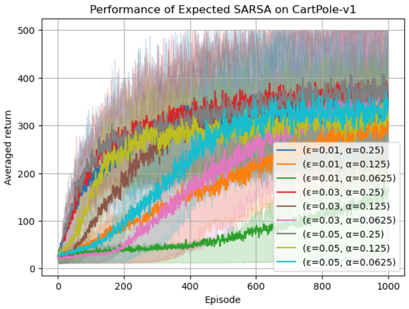


Figure 4: Expected SARSA

- The performance of both algorithms generally improves as the number of episodes increases across all combinations of learning rate and exploration rate. In general, Q-learning does achieve a higher averaged return over time compared to Expected SARSA.
- The overall better performance of Q-learning on CartPole-v1 in most of the parameter combinations could be due to the specifics of the task, where exploiting known good strategies is more beneficial than the more cautious approach taken by Expected SARSA. CartPole is typically less stochastic than MountainCar and might benefit from the more aggressive strategy.
- For both algorithms, a higher exploration rate (ϵ=0.03 or 0.05) appears to balance exploration and exploitation well, as these lines tend to rise towards the higher averaged returns.
- For both algorithms, a higher learning rate (α=0.25 or 0.125) performs better than 0.0625, thus slightly more new information overrides old information works better here.

2 Policy Gradient Theorem

Since

$$\pi(a_i|s) = \frac{e^{z(s,a_i)}}{\sum_{a\in A} e^{z(s,a)}}$$

$$\nabla_{z(s,a_j)}\pi(a_i|s) = \pi(a_i|s)(I(i=j) - \pi(a_j|s))$$

and

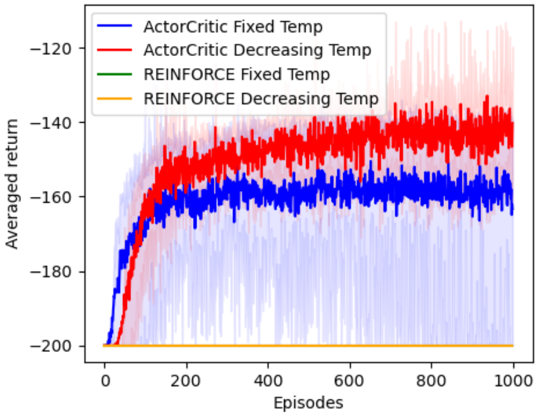
$$\begin{aligned} J(\pi) &= \mathbb{E}_{s\sim d^\pi, a\sim\pi}[Q^\pi(s,a)] \\ &= \sum_{s\in\mathbf{S}} d^\pi(s) \sum_{a'\in A} \pi(a'|s) Q^\pi(s,a') \end{aligned}$$

Thus we have:

$$
\begin{aligned}
\nabla_{z(s,a)} J(\pi) &= \frac{\partial J(\pi)}{\partial z(s,a)} \\
&= d^{\pi}(s) \sum_{a' \in A} \nabla_{z(s,a)} \pi(a'|s) Q^{\pi}(s,a') \\
&= d^{\pi}(s) \sum_{a' \in A} \pi(a'|s)(I(a=a') - \pi(a|s)) Q^{\pi}(s,a') \\
&= d^{\pi}(s)[\pi(a|s) Q^{\pi}(s,a) - \pi(a|s) \sum_{a' \in A} \pi(a'|s) Q^{\pi}(s,a')] \\
&= d^{\pi}(s) \pi(a|s)[Q^{\pi}(s,a) - V^{\pi}(s)] \\
&= d^{\pi}(s) \pi(a|s) A^{\pi}(s,a)
\end{aligned}
$$

3 Policy-based methods with linear function approximation

2.1 MountainCar-v0 task



50

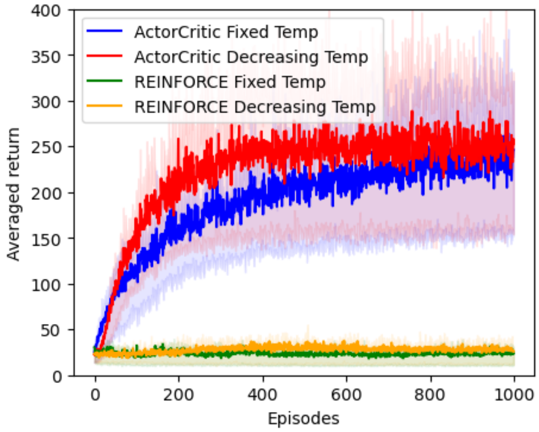
Figure 5: Performance of ActorCritic and REINFORCE on MountainCar-v0

- Actor-Critic method shows improvement in averaged return over time, with both fixed and decreasing temperature configurations. But REINFORCE algorithm did not learn an effective policy for the MountainCar-v0 task during the training episodes under these parameter settings.
- This better performance of Actor-Critic could be attributed to its ability to balance the policy learning (actor) with the value estimation (critic),

which helps in stabilizing the updates and directing the agent more effectively towards the goal. The failure of REINFORCE could be due to insufficient exploration, or the high variance of the REINFORCE algorithm, which can make learning difficult in environments with sparse rewards like MountainCar-v0.

- Decreasing Temp in Actor Critic shows a performance improvement over time. This suggests that reducing exploration as the algorithm learns can be beneficial. Because it allows the algorithm to exploit the better-understood parts of the state space more effectively as learning progresses.

2.2 CartPole-v1 task



50

Figure 6: Performance of ActorCritic and REINFORCE on CartPole-v1

- Actor-Critic method shows improvement in averaged return over time, with both fixed and decreasing temperature configurations. REINFORCE exhibits low performance. It does not show substantial improvement over episodes, suggesting that even with an adaptive exploration strategy, REINFORCE is not suitable for the CartPole-v1 task given the settings used.

- Decreasing Temp in Actor Critic shows a performance improvement over time. This suggests that reducing exploration as the algorithm learns can be beneficial. Because it allows the algorithm to exploit the better-understood parts of the state space more effectively as learning progresses.