Janice Ainembabazi

Southern New Hampshire University

CS-370-12200-M01 Current/Emerging Trends in CS

Dr. Mohammad Habibi

December 8th 2024

**MODULE SIX**

In reinforcement learning (RL), agents learn to make decisions by interacting with their environment, and a common example used to understand RL is the cartpole problem. In this problem, the agent must balance a pole on a moving cart by deciding whether to move the cart left or right. In Module Five, I used a Deep Q-Network (DQN), a value-based approach, to solve this problem. In this paper, I will explore two other ways to solve the cartpole problem: using the REINFORCE algorithm, which is policy-based, and using the Advantage Actor-Critic (A2C) algorithm, which combines both value- and policy-based approaches. By comparing these methods, I hope to deepen my understanding of reinforcement learning.

**With Solving the Cartpole Problem using REINFORCE**

The REINFORCE algorithm is a policy-based method, meaning it directly optimizes the agent’s policy. Instead of learning a value function like DQN, REINFORCE updates the policy parameters to maximize the expected reward. Here's how I would approach solving the cartpole problem using REINFORCE:

1. **Initialize** the policy network with random parameters.
2. For each episode:
   * Starting with the initial state of the cartpole.
   * Collecting data by letting the agent interact with the environment (states, actions, and rewards).
   * Calculating the total reward for each action taken (this is called the return).
   * Updating the policy network by using the following equation: Gradient=−log(action probability)×return\text{Gradient} = -\log(\text{action probability}) \times \text{return}Gradient=−log(action probability)×return
3. Repeating this process for multiple episodes, gradually improving the policy.

However, one downside of REINFORCE is that it can be unstable because of high variability in the random samples used to update the policy. This can slow down training and make it harder to converge to an optimal solution.

**With Solving the Cartpole Problem using A2C**

The Advantage Actor-Critic (A2C) algorithm is a more advanced method that combines both policy-based and value-based approaches. A2C uses two neural networks:

* The **Actor** is responsible for choosing the best action based on the current policy.
* The **Critic** evaluates the chosen action by estimating the value of the state.

Here’s how I would use A2C to solve the cartpole problem:

1. **Initialize** both the Actor and Critic networks with random parameters.
2. For each episode:
   * Starting with the initial state.
   * Collecting data by interacting with the environment (states, actions, and rewards).
   * Calculating the advantage, which is the difference between the Q-value (action-value) and the state-value
   * Updating the Actor’s policy using the advantage
   * Updating the Critic to minimize the error between the predicted value and the actual return.

A2C tends to be more stable than REINFORCE because the Critic helps reduce the variability in the policy updates. It provides more reliable feedback for the Actor, which improves learning efficiency.

**Policy-Based vs. Value-Based Approaches**

Policy-based methods, like REINFORCE, focus on directly improving the policy. They are often better suited for environments with continuous actions, where it’s harder to calculate the best action for each state. However, they can have high variance, making training less stable and slower.

In contrast, value-based methods like DQN learn the value of each state-action pair. The agent then uses this value to determine which actions to take. Value-based methods work well in environments with discrete actions, but they struggle in continuous action spaces.

**Actor-Critic vs. Other Approaches**

Actor-Critic methods like A2C combine both value- and policy-based approaches. The Actor updates the policy, while the Critic evaluates the actions to stabilize learning. This combination allows Actor-Critic methods to learn more effectively in environments with both discrete and continuous action spaces. In my opinion, A2C is a great way to balance exploration and stability during training, making it more efficient than REINFORCE or DQN alone.

I explored two advanced reinforcement learning methods: REINFORCE and A2C. REINFORCE directly optimizes the policy but suffers from high variability and slow learning. A2C, on the other hand, combines the strengths of both policy-based and value-based methods, offering more stability and efficiency. By understanding these methods, I have gained a deeper insight into reinforcement learning and how different approaches can be applied to solve problems like the cartpole challenge.

**REFERENCES**

Surma, G. (2019, November 10). *Cartpole - introduction to reinforcement learning (DQN - deep Q-learning)*. Medium. <https://gsurma.medium.com/cartpole-introduction-to-reinforcement-learning-ed0eb5b58288>

Yoon, C. (2019a, May 23). *Deriving policy gradients and implementing reinforce*. Medium. <https://medium.com/@thechrisyoon/deriving-policy-gradients-and-implementing-reinforce-f887949bd63>

Yoon, C. (2019, July 17). *Understanding actor critic methods*. Medium. <https://towardsdatascience.com/understanding-actor-critic-methods-931b97b6df3f>