Module 6: CARTPOLE Revisited

CS-370 – Emerging/Current Trends in Computer Science

Jimmy Ferchak

December 8, 2024

**Explain how the cartpole problem can be solved using the REINFORCE algorithm.**

The cartpole problem is a control problem in which a pole is attached to a cart and the goal of the problem is to balance the pole upright by moving the cart back and forth. The previous assignment had me solve this using a DQN solution, but the cartpole problem can also be solved with REINFORCE as well as other approaches.

The REINFORCE algorithm is a Monte-Carlo variant of policy gradients. The agent collects a trajectory τ of one episode using the current policy. The algorithm then uses this to update the policy parameter. REINFORCE is updated in an off-policy way because one full trajectory must be completed to make a sample space. Below is a picture of pseudocode of the algorithm shown in the Medium article “Deriving Policy Gradients and Implementing REINFORCE” by Chris Yoon (Yoon 2019a):

A math equations and formulas

Description automatically generated with medium confidence

The flow of the algorithm is as follows:

1. Do a trajectory roll-out using the current policy
2. Store log probabilities of the policy and reward values at each step
3. Calculate discounted cumulative future reward at each step
4. Compute policy gradient and update policy parameter
5. Repeat

**Explain how the cartpole problem can be solved using the A2C algorithm.**

A2C, or Advantage Actor-Critic is another policy-based reinforcement algorithm. What makes A2C different is that A2C learns the actor and critic function. The critic function evaluates how well the actor did and the validity of its actions. The advantage function is another reason why A2C is different from other reinforcement learning algorithms. The advantage function is the difference between the predicted cumulative reward of taking an action in the current state and the predicted total reward of being in the current state. Belowis a picture of the pseudo code provided by another article made by Chris Yoon titled “Understanding Actor Critic Methods and A2C” (Yoon 2019b):

A black screen with white text

Description automatically generated

**Explain how policy gradient approaches differ from value-based approaches, such as Q-learning.**

Q-learning, which is a value-based approach, tries to adjust the action-value function by computing the expected total reward for a state/action pair. The agent then chooses an action by selecting the action that has the greatest predicted cumulative reward in the current state. The selection is based off prior action-value pairs. The policy gradient approach is more direct in learning the optimal policy and does this without explicitly learning the action-value function. The policy function connects a state to a probability distribution over actions, and the agent then selects actions by sampling from this distribution. A key difference between the two approaches is that policy gradient can handle continuous cations, while value-based are limited to discrete actions.

**Explain how actor-critic approaches differ from value- and policy-based approaches.**

Actor-Critic approaches combine elements of both value and policy based approaches and learn both a policy function and a value function at the same time.

References

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. (2019b, July 17). *Understanding Actor Critic Methods*. Medium. https://towardsdatascience.com/understanding-actor-critic-methods-931b97b6df3f