Project Information

The project involved working in a Unity environment where the agent had to move a double-jointed arm to a target location. A reward of 0.1 is provided for each movement in the right location. Here are the values for the state and action spaces:

- State space: 33 dimensions (velocity, perception of objects in front of the agent, etc.)
- Action space: continuous (4 numbers between -1 and 1, referring to the torques applied)

The environment is considered solved when the agent has achieved an average reward of +30 over 100 episodes.

Learning Algorithm

The learning algorithm I chose was the Deep Deterministic Policy Gradient Algorithm (DDPG). It is a type of actor-critic method, where the actor produces a deterministic policy and the critic evaluates the policy based on the average rewards. The TD error function is used to update the critic. This method is very similar to DQN's, but modified slightly to work with continuous action spaces, like the one in this project. It makes use of many of the same techniques as DQN's, including experience replays. This method helps keep the samples independent from each other. It works by first creating a replay buffer of some fixed size. After a few episodes are run, the results of each of the experiences are stored in the buffer. After some more episodes, some experiences are sampled from the buffer and used to update the weights and losses for the network. This method helps keep each experience independent from each other. Another method used in this algorithm involved soft updates, where the target networks are updated through a mix of the local network weights and the target network weights. Typically, the ratio is around 99% of the target weights and 0.01% of the local weights.

Here are the hyperparameters I used for my network:

- 3 hidden layers
- 700 hidden units
- ReLU activation on all layers except the output layer
- Learning rate of 0.0001 for the actor, 0.001 for the critic

For the other parameters, I set my batch_size to 128, gamma to 0.99, tau to 0.001, and buffer_size to 1e5.

Future Work

- Testing other methods like A3C, A2C, and PPO
- Implementing prioritized replay to focus on specific experiences