Udacity Deep Reinforcement Learning Nano Degree
Project 3 Collaboration and competition

January 21, 2019

1 Introduction

Collaboration and competition is the third project for the Udacity Nano degree in Deep Reinforcement Learning. In this environment, two agents control rackets to bounce a ball over a net. If an agent hits the ball over the net, it receives a reward of +0.1. If an agent lets a ball hit the ground or hits the ball out of bounds, it receives a reward of -0.01. Thus, the goal of each agent is to keep the ball in play.

The observation space consists of 8 variables corresponding to the position and velocity of the ball and racket. Each agent receives its own, local observation. Two continuous actions are available, corresponding to movement toward (or away from) the net, and jumping.
The task is episodic, and in order to solve the environment, your agents must get an average score of +0.5 (over 100 consecutive episodes, after taking the maximum over both agents). Specifically, after each episode, we add up the rewards that each agent received (without discounting), to get a score for each agent. This yields 2 (potentially different) scores. We then take the maximum of these 2 scores. This yields a single score for each episode.

The environment is considered solved, when the average (over 100 episodes) of those scores is at least +0.5.

2 Learning Algorithm

For the purpose of this submission an implementation of the Multi-Agent DDPG algorithm is included. Thus, to solve the environment two agents are instantiated each with their own actor and critic network. However, the critics are updated using local observations from each agent. The hyperparameters used are as follow: the maximum number of training episodes is 1000, the seed to

initialise the pseudo random number generator is 0, the maximum size of buffer is 1e5, the size of each training batch is 64, the discount factor is 0.99, the interpolation parameter for soft updating is 1e-3, the learning rate for the actor and the critic is 5e-4 and the weight decay parameter is set to 0.

The network architecture for both the agent and the critic consists of a single neural network with two hidden layers consisting of 128 neurons in the first hidden layer and 128 neurons in the final hidden layer. Each layers uses ReLu activation.

2.1 Training Results

The results obtained during training of the agent are

Episode: 654 Score: 2.70 Average Score: 0.52
Environment solved in 654 episodes! Average Score: 0.52

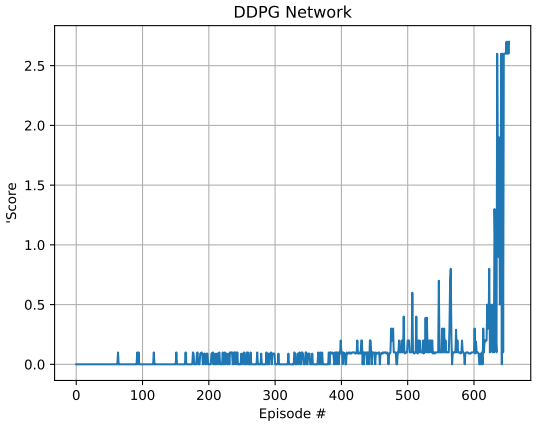

Figure 1: Training performance of Double Deep Q Network

3 Ideas for Future Work

I would definitely like to spend more time learning about multi-agent systems as I found collab and compete to be the most difficult yet interesting project of this nano degree. This would certainly involve looking at how other methods such as PPO, A3C, or D4PG can be used to train multiple agents.