Project Report

Learning Algorithm

I used Deep Deterministic Policy Gradients for this project on teaching two agents two play tennis.
The action space for the agent is continuous, therefore Deep Deterministic Policy gradients has been used to solve it. The (state, action, reward, next state, done) of both agents are stored in the experience buffer.
The agent is updated 11 times after every 5 time steps. This was set after a lot of experimentation.
The environment is considered solved when agents get an average score of 0.5 over 100 consecutive episodes..

Hyperparameters

I moderately experimented with the hyperparameters as I was able to train the agent. The hyperparameters used are as follows:

- BUFFER_SIZE = int(1e6) # replay buffer size
- BATCH_SIZE = 1024 # minibatch size
- GAMMA = 0.99 # discount factor
- TAU = 1e-3 # for soft update of target parameters
- LR_ACTOR = 1e-4 # learning rate of the actor
- LR_CRITIC = 1e-3 # learning rate of the critic
- WEIGHT_DECAY = 0.0000 # L2 weight decay

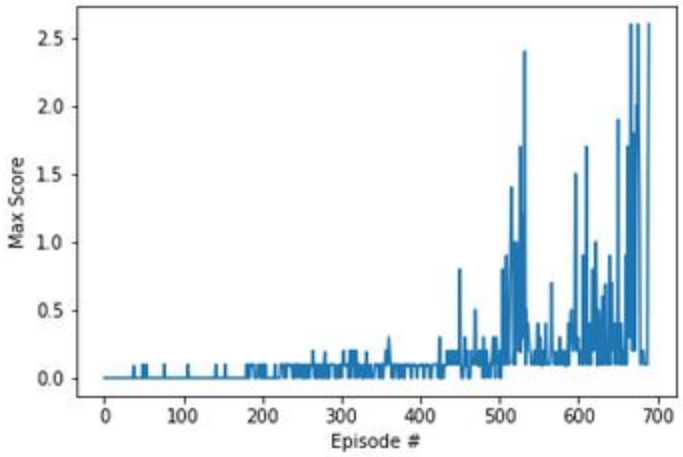
Neural Network Architectures

- Used 2 hidden layers in the **actor network**.
- Batch normalization is applied on input.
- Layer 1 has 128 nodes and layer 2 has 64 nodes.
- Both the layers have relu activation.
- Final fully connected uses tanh activation also.
- **Critic network** has 2 hidden layers.
- Batch normalization is applied on input.
- Layer 1 has 128 nodes, layer 2 has 64 nodes.

- Both the layers use relu activation.
- No activation on final fully connected layer.

Plot of rewards

Average score of 0.5+ was achieved in 691st episode.



Ideas for future works

Several other algorithms like PPO, D4PG have been specified in the classroom to solve the project. It would be good to experiment with these algorithms in future and see how they perform.