Learning algorithm

Owing to the continuous action space, Deep Deterministic Policy Gradients algorithm has been used for this project which performs good in continuous action space. The network is being updated 12 times after 6 time steps.

Network acrhitectures

Actor:

- Hidden layer - 128 units
- Relu
- Batchnorm
- Hidden layer - 64 units
- Relu
- Batch norm
- Fully connected layer
- Tanh

Critic:

- Batchnorm on input
- Hidden layer - 128 units
- Relu
- Hidden layer - 64 units
- Relu
- Fully connected layer

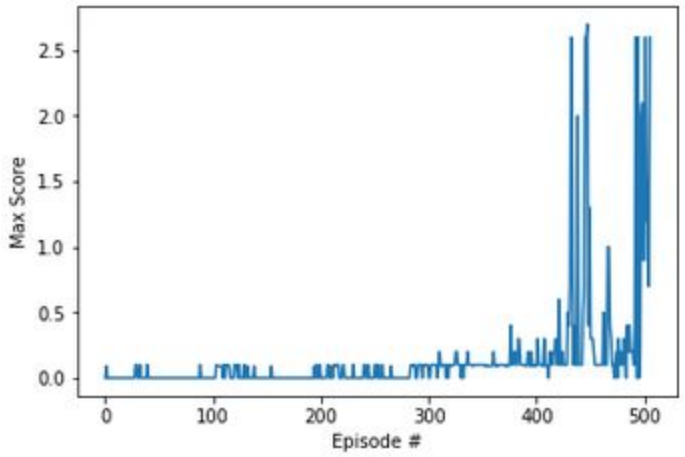
Hyperparameters

Following hyperparameters were used to train the agent.

```
BUFFER_SIZE = int(1e6)  # replay buffer size
BATCH_SIZE = 512        # minibatch size
GAMMA = 0.99            # discount factor
TAU = 1e-3              # for soft update of target parameters
LR_ACTOR = 4e-4         # learning rate of the actor
LR_CRITIC = 1e-3        # learning rate of the critic
WEIGHT_DECAY = 0.0000   # L2 weight decay
```

Reward plot

```
Environment SOLVED at episode 506 Avg Score: 0.52
```



Ideas to improve the performance of the agent

- I experimented only with DDPG. Other algorithms also need to be tried in future to gain improvement in performance like PPO, A3C, MADDPG etc.