

# Deep Deterministic Policy Gradients

## **Algorithm –**

The Deep Deterministic Policy Gradient algorithm is an extension of actor critic methods to continuous action spaces. These models directly map input states to actions instead of outputting a probability distribution over discrete actions spaces.

## **Unity ML Agents –**

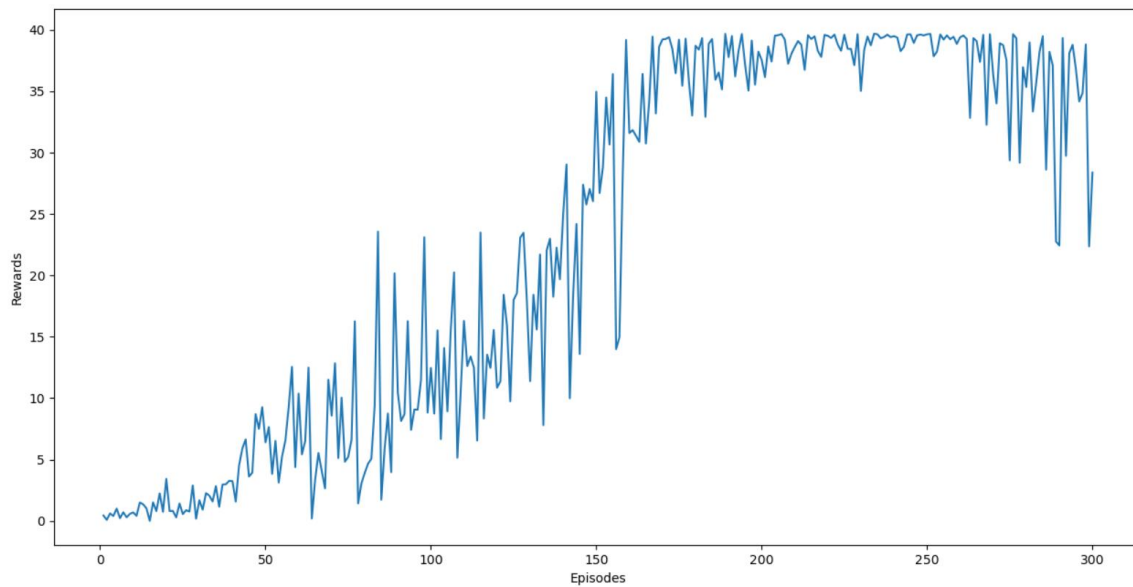
Unity Machine Learning Agents were developed by unity to provide game and simulation environments as training grounds for reinforcement learning. It is an open source plugin which contains already built environments along with the capacity to allow users to create their own environments. In this repository, an environment made by Udacity is used. There won't be a need to download Unity for this project. The implementation won't work with the latest Unity ML-Agents. To reproduce the results, you will need to use this environment.

## **Implementation –**

This algorithm is implemented using PyTorch library. The neural network used for the actor consists of 3 layers with 600, 400, and 200 neurons respectively. The critic network consisted of 2 layers with 400 and 300 neurons. The critic network had a weight decay factor of 0.0001. The agent was trained using Adam optimizer with a learning rate of  $1.5e-4$  for both actor and critic network. The buffer length was  $1e5$  and batch size was 128. Tau used for soft update was  $1e-3$ . The discount factor was set to 0.99. Every episode was run for a maximum of 1000 timesteps. The agent took approximately 5 hours on GPU to train.

## Results –

The environment is considered solved when the agent accumulates an average reward of 30 or more over 100 episodes. It took my network 220 episodes to solve the environment.



## Future Work –

In this project I solved the environment which contained a single agent. For the my future work I would like to solve the environment with 20 agents using the D4PG algorithm.