Report of training the agent

The model was trained for a maximum of 2,000 episodes but the agent was able to solve the environment (i.e. get atleast +0.5 average score over 100 adjacent episodes).

Model

The Actor Network has three dense (or fully connected layers). The first two layers have**256 and 128** nodes respectively activated with **ReLU** activation function. The final (output layer) has**2** nodes and is activated with tanh activation. This network takes in as input the **8** dimensional current state and gives as output**2** to provide the action at current state that the agent is supposed to take.

The Critic Network has three dense (or fully connected layers). The first two layers have**300 and 128** nodes respectively activated with **ReLU** activation function. The final (output layer) has**2** nodes and is activated with linear activation (no activation at all). This network takes in as input the **8** dimensional current state and**2** dimensional action and gives as output a single real number to provide the Q-value at current state and action taken in that state.

Both the neural networks used Adam optimizer and Mean Squared Error (MSE) as the loss function.

The following image provides a pictorial representation of the Actor Network model:

Pictorial representation of Q-Network

The following image provides a pictorial representation of the Critic Network model:

Pictorial representation of Q-Network

The following image provides the plot for score v/s episode number:

Plot for score v/s episode number

Algorithm

The algorithm used to train both of these agents is Multi Agent Deep Deterministic Policy Gradients (DDPG) which is a Multi Agent Actor-Critic model. Each of the two agent has two neural networks - Actor and Critic.

The Actor neural network takes in as input the current state that the agent experiences and gives as output the probability of each action, with the highest probability to the action to be selected at the current state. In a way the Actor model outputs the policy of an agent.

The Critic neural network takes in as input the current state and the action that is taken at that state (which is generated by the Actor model) and computes the Q-value for the particular (state, action) pair.

Thus the Actor and Critic model work in tandem until a near optimal policy is generated by the Actor neural network.

For more information on the Single Agent DDPG model (which is the base of Multi Agent DDPG) follow the link containing tutorial by **OpenAI** :- DDPG OpenAI

Performance

The model was trained on MacBook Air 2017 with 8GB RAM and Intel Core i5 Processor.

- **Number of episodes required to solve the environment** 429 episodes
- **Final score of the agent**: 0.52

Hyperparameters used

Hyperparameter	Value	Description
Buffer size	100000	Maximum size of the replay buffer
Batch size	256	Batch size for sampling from replay buffer
Gamma (γ)	0.99	Discount factor for calculating return
Tau (τ)	0.01	Hyperparameter for soft update of target parameters
Learning Rate Actor	0.001	Learning rate for the actor neural network
Learning Rate Critic	0.001	Learning rate for the critic neural network

Future work

The following multi agent equivalent of these algorithms can be considered for further development of this agent:

- Proximal Policy Optimization (PPO)
- Generalized Advantage Estimation (GAE)
- Advantage Actor-Critic (A2C)
- Asynchronous Advantage Actor-Critic (A3C)