Learning Algorithm:

In this project, I implemented MADPPG which I based on code shared in the class but made many modifications to the code. However, the basic idea is to utilize a multie agent DPPG algorithm. The model weights are stored under model_dir forlder at checkpoint.pt which include actor and critic for both agents.

Hyperparameters:

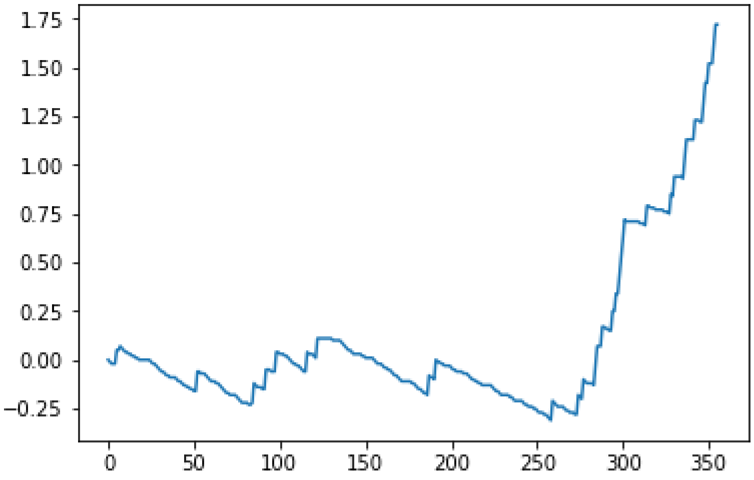
BUFFER_SIZE = 5000000
BATCH_SIZE = 64
GAMMA = 0.99
TAU = .02
LR = .01
UPDATE_EVERY = 4
Noise = 2
Noise Reduction = .9999

Model Architecture:

There are two models: actor and critic. The actor takes a state and outputs 64 nodes, which are then taken by another layer of 64 nodes. The output layer consists of action space of two continuous variables. Each layer is outputted with leaky relu function except output is through tanh. The critic consists of similar layers, however, it outputs 1 variable per agent.

Plot of rewards:

As you see,the algorithm stopped at around 350. Please note that I'm utilizing 2 parallel environments so this is actually 712 episodes of training.



Future work:

This could be improved further with prioritized replay.