Report for Project 1 – Navigation

The project involves using DQN or its variants to learn an agent that picks up yellow bananas and doesn’t pick up blue bananas.

I followed similar architecture to the one we implemented in the exercise for DQN. I adapted the code slightly to fit the new environment.

Using a DQN already gave me great results with the same settings as used in exercise. I used a network consisting of 2 hidden layers of 64 length each and used ReLu as my activation function.

The model is contained in Model.py.

Dqn\_agent.py describes the agent. It is same as the one in exercise.

Train\_p1.py contains the python script to load unity environment and calls to train the agent. This is exactly same as Navigation notebook.

The hyperparameters used for training are:

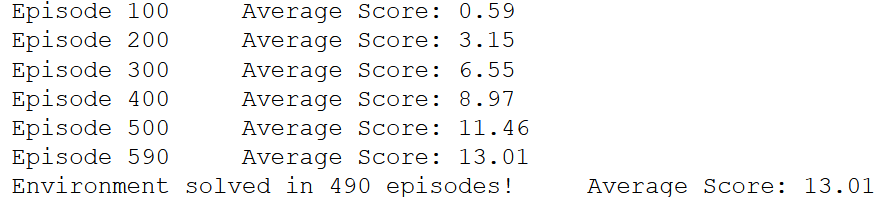
num\_episodes = 2000,

Maximum timesteps = 1000

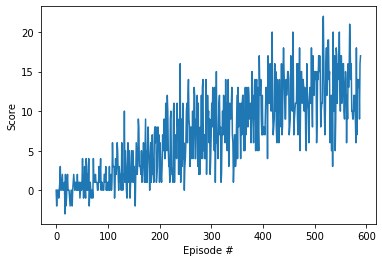
epsilon-greedy action selection has starting value of 1.0, decay of 0.995 per episode and min of 0.01.

Results

These parameters gave me impressive results with reaching the score of >13 within 500 episodes.



The plot of rewards per episode is shown below:



For Future Work, We can implement improvements over DQN such as dueling DQN, double DQN etc.

Some refinement can also be done in hyper parameter space for further improvements.