Deep reinforcement learning Project 1 report

This is a report for deep reinforcement learning project 1.

Code structure

The structure of the code is split into 3 files, Navigation.ipynb, model.py and agent.py. The main procedures, such as initialising the agent, initialising the unity environment, and looping through the episodes. Model.py contains the Pytorch implementation of neural network. Agent.py contains the q-learning agent implementation, where the q-learning updates and calls model.py.

Chosen hyperparameters:

For dqn:
n_episodes=2000, max_t=1000, eps_start=1.0, eps_end=0.01, eps_decay=0.995

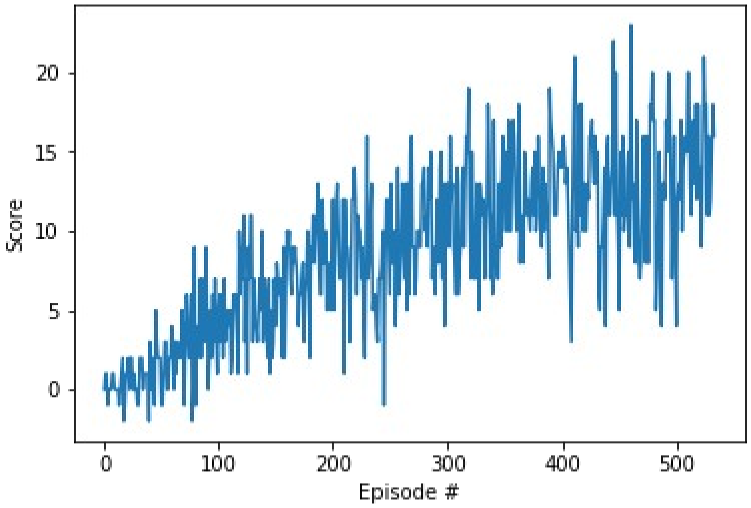
ANN architecture

A total of 4 fully connected networks with relu activation functions. The input dimension and output dimensions are as shown:

1. hidden layer 1
   - input: no. of states
   - output: 128
2. Hidden layer 2
   - input: 128
   - output: 164
3. Hidden layer 3
   - input: 64
   - output: 32

Final results

The agent have to achieve an average score of 13 or more for 100 consecutive episodes.
The model managed to solve the environment in 434 episodes to achieve the score.



Ideas for future work

CNN could be tested out to see if it could take even lesser episodes to achieve the score. Also a larger range of hyperparameters can be tried to see if the agent would be better.