Deep Q learning to solve navigation

In this project I have worked with a unity environment, in which an agent is required to navigate and collect as many yellow bananas as it can. There are blue bananas in the environment, the agent should avoid them.

The implementation is in pytorch, and follows vanilla DQN algorithm.

State size = 37
Action space = 4
Number of layers in network =3
Hidden layer size = 64
The discount factor is set to 0.99
Learning rate is 0.0005

The model achieves a score greater than 13.0 in about 500 episodes.

Ideas for future work:

1. Double Deep Q Networks
2. Prioritized Experience Replay
3. Dueling Deep Q Network