Project 1: Navigation
Report

To solve this environment I have adapted the DQN agent from demonstration notebooks. It uses a fully connected neural network with 2 hidden layers, each with 64 neurons. The network is learning from replay buffer using soft-updates after every 4 steps.

For training I have also used ϵ-greedy policy selection to encourage exploration. ϵ starts from 1.0 and goes down to 0.01 with decay factor of 0.995. Other hyper-parameters include: learning rate of the optimizer $\alpha = 5 \times 10^{-4}$, soft-update parameter tau $\tau = 1 \times 10^{-3}$, and the discount factor $\gamma = 0.99$.

Solving the environment means achieving average score over the last 100 episodes of +13. The environment was solved before 1700 episodes. After that, the average score increased slightly but never went over 14. The network weights are saved in `"solved_model.pth"` file.

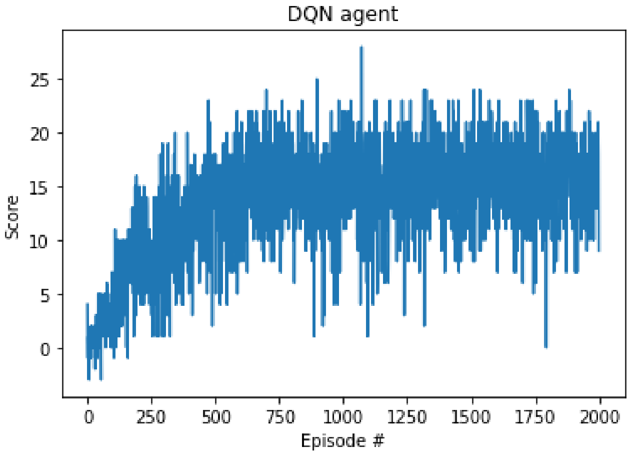

Figure 1: DQN training score over 2000 episodes

There are several potential improvements to this implementation:

1. Learn form pixels
2. Use prioritised experience replay
3. Double DQN
4. Dual DQN