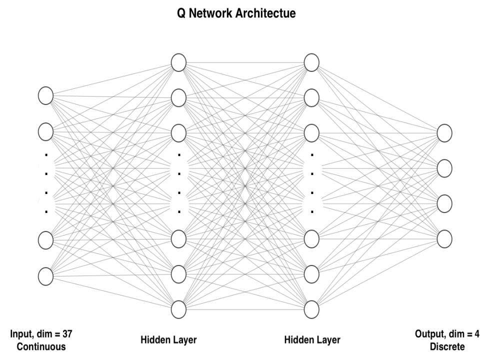
Navigation Project

Two training algorithms Deep Q-learning and Double DQN, are implemented. This report summarizes the training scores, performance with DDQN agent and comparison analysis from these two algorithms.

Q-Network Model Architecture

The value network consists of three fully connected layers as following. `relu` is used after each hidden layer.



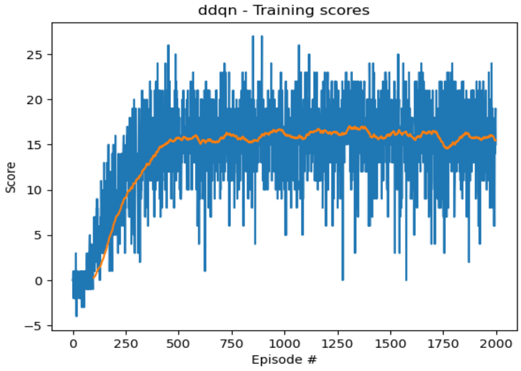
Training

- Training parameters:

```
    - max number of training episodes, n_episodes = 2000
    - maximum number of timesteps per episode, max_t = 1000
    - epsilon-greedy policy:
      - starting value of epsilon, eps_start = 1.0
      - minimum value of epsilon, eps_end = 0.01
      - multiplicative factor (per episode) for decreasing epsilon,
eps_decay = 0.99
```

Double DQN training results

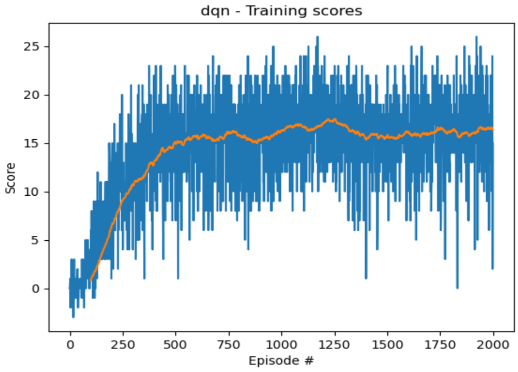
- Average Score of 16 was reached around 500 episode.



- Testing Performance The average score for DDQN agent playing the game is 16.51 for 100 consecutive rounds.

DQN training results

- Training score reached to average of 15 around 550 episode.



Observations

Training with both algorithms was able to reach to score 15 around 500 episodes. Trainnig with Double DQN reached to score 15 slightly faster.

Futher Improvement

Futher turning of the parameters for epsilon policy is one of the areas that could further improve agent performance.

Algorithm wide, Prioritized Experience Replay is an interesting direction to further explore. This implmenation of PER has an efficient `SumTree` for reply buffer implementation.