Navigation Banana Environment

Name: Guillermo del Valle Reboul

Algorithm Used

A Deep Q Network DQN was used for this project. The DQN model is built with 4 Linear layers fully connected (nn.Linear size 256) and Relu activation functions.

Hyperparameters

- Replay buffer size = 100000
- Batch size = 256
- GAMMA (discount factor) = 0.998
- Tau = 0.001
- Learning rate = 0.0005
- Update the network evey 16 step

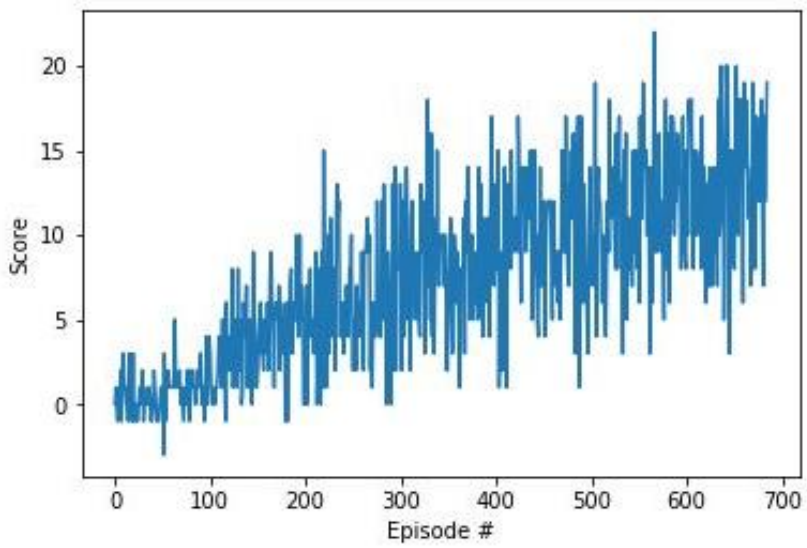
Files Used

- *Navigation.ipynb*: used in training.
- *navigation.py* : used for evaluation.
- *model.py*: contains the Q Network model used for both target and local Q networks.
- *dqn_nav_agent.py*: contains the agent with the DQN algorithm.
- *Checkpoint.pth*: weights saved from training for evaluation.
- *README.md*: for instructions.

Plot of Rewards

DQN Solved the environment in less than 600 episodes (Average score = 13.0). See "*Agent Evaluation.mp4*" to watch trained agent in action (evaluation in navigation.py)

```
Episode 100     Average Score: 0.73
Episode 200     Average Score: 3.68
Episode 300     Average Score: 6.02
Episode 400     Average Score: 8.47
Episode 500     Average Score: 9.820
Episode 600     Average Score: 11.37
Episode 685     Average Score: 13.01
Environment solved in 585 episodes!     Average Score: 13.01
```



Ideas for Future implementations

DQN using raw images 84x84 instead of the states provided by the environment. For that purpose, 2 or 3 convolutional layers must be added to the DQN networks so that the agent is able to recognize properly "yellow" bananas, and avoid "blue" bananas.