< Project 1 – Navigation>

1. Learning algorithm - DQN

The DQN is an algorithm that stores state, action, reward, next state in replay memory and randomly selects one of these samples to train the model. While learning, the local model is constantly updated, but the target model is updated according to the prescribed steps. The goal of the DQN algorithm is to minimize the difference between the target model and the local model. Details of the hyperparameters of the algorithm are as follows.

2. Chosen parameters

Hyperparameters	Value
minibatch size	128
replay memory size	150000
model update cycle	5
discount factor	0.95
learning rate	0.0004
initial exploration	1.0
final exploration	0.005
exploration decay	0.995

3. Model architecture for neural network

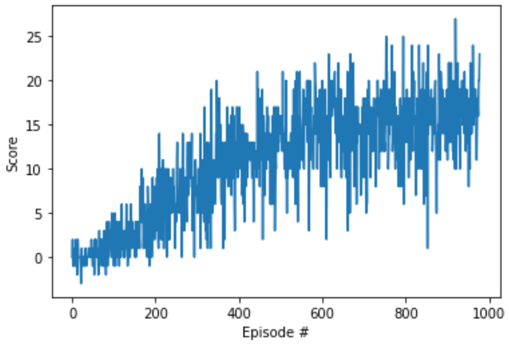
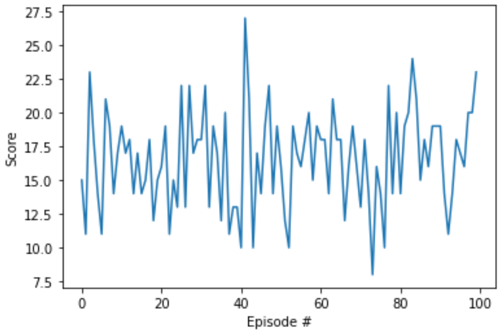
The architecture of neural networks used in target and local models is the same and shown below. Both neural networks used relu as an activation function.

(fc1): Linear(in_features=37, out_features=64, bias=True)

(fc2): Linear(in_features=64, out_features=32, bias=True)

(fc3): Linear(in_features=32, out_features=4, bias=True)

4. Plot of reward per episode

A plot of each Episode	A plot of last 100 episode
	

5. Idea for Future Work

Currently, the model has exceeded the project base point of 13, but it is hard to say that the score is rising steadily. In the future, attempts are needed to create models that are learned reliably.Future algorithm improvement directions are as follows.

- Adjust Hyperparameters

- Apply different DQN algorithms : Duel DQN, Double DQN, Prioritized DQN