Report – Navigation

Environment

The code was tested on workspace in udacity course, using the environment“Banana.x86_64”. This environment is trying to make the agent learn to catch yellow bananas as much as possible while avoiding blue bananas.

Algorithms

The main algorithm is based on Deep Q-learning considering “Experience replay” and “Fixed-Q-Targets”. Specifically, the policy model is a neural network, which has 37 states as input and 4 action as output. There are two linear hidden layers (64 neurons), using ReLU as activation function. The concrete architecture is shown below in Figure 1.

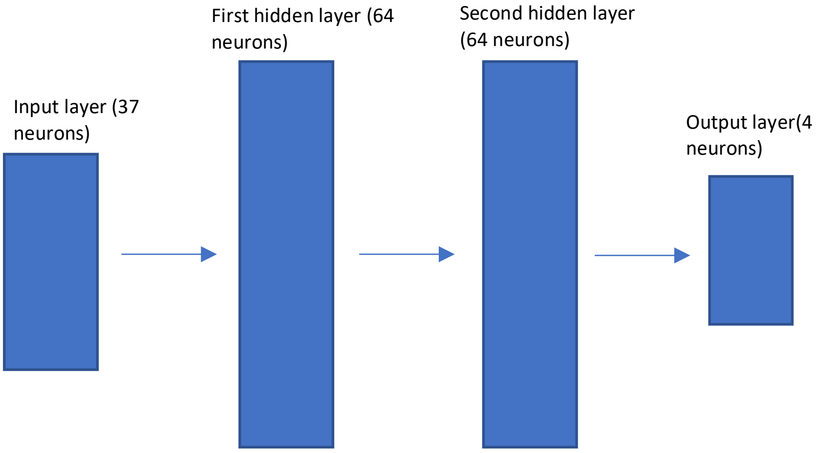

First hidden layer (64 neurons)
Second hidden layer (64 neurons)
Input layer (37 neurons)
Output layer(4 neurons)

Figure 1

Next step is to train the agent. We use experience replay, namely storing (states, actions, rewards, next steps, done) into a Replay Buffer, to help the agent learn all possible situation. Since the weight of the Q-networks is the same as the weights of the goal, it’s difficult to converge. Therefore, a fixed Q-target is introduced, which trains local Q-Network with the same architecture and then soft-update the target Q-Network.

It’s worth to mention that an epsilon-greedy action policy(exploration and exploitation) and a discount value, gamma(0.99) are introduced during learning.

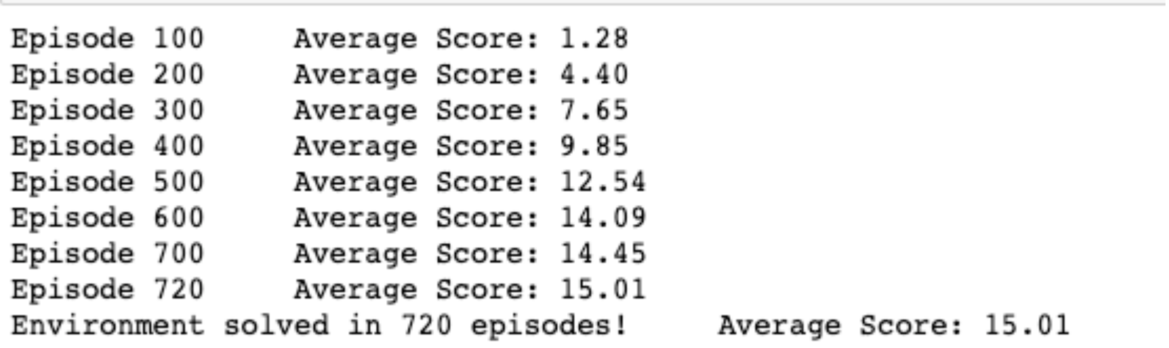
Hyperparameter values

Replay buffer size	100,000
Batch size	64
Discount factor	0.99
Tau for soft update of target Q	0.001
Learning rate for gradient decent	0.0005
Eps for epsilon-greedy policy	From 1 to 0.01 with 0.995 decay
First hidden layer for NN	Linear with 64 neurons
Activation function	ReLU
Second hidden layer for NN	Linear with 64 neurons
Activation function	ReLU

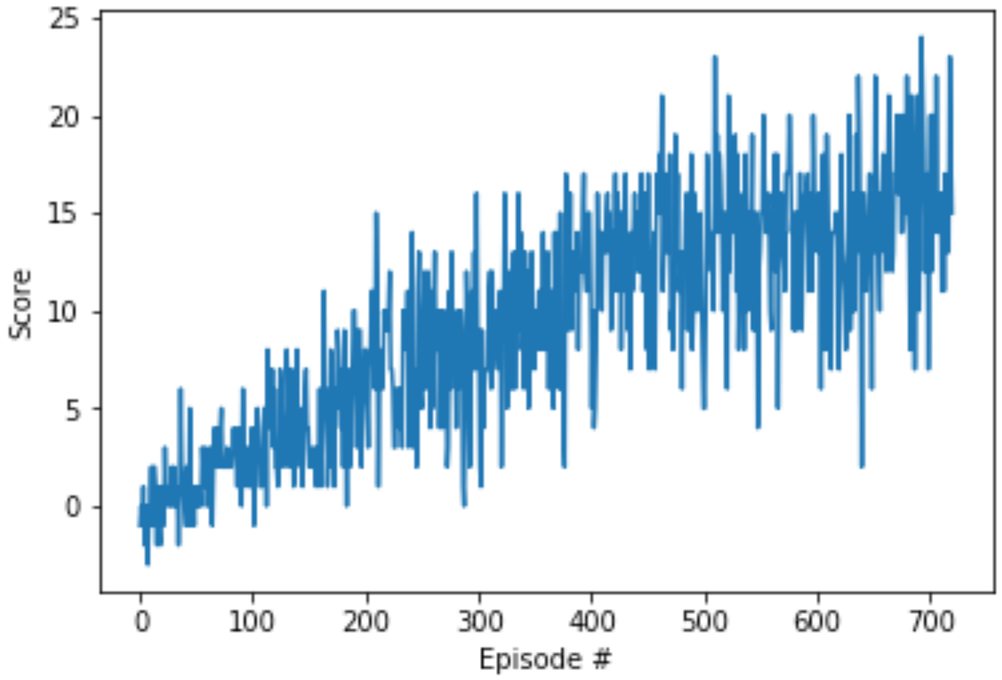
Output layer	Linear with 4 neurons

Results.

With given algorithm, the average score can achieve 15 after 720 episode. See the graphs below.



```
Episode 100     Average Score: 1.28
Episode 200     Average Score: 4.40
Episode 300     Average Score: 7.65
Episode 400     Average Score: 9.85
Episode 500     Average Score: 12.54
Episode 600     Average Score: 14.09
Episode 700     Average Score: 14.45
Episode 720     Average Score: 15.01
Environment solved in 720 episodes!     Average Score: 15.01
```



Future work

There are multiple improvements could be done: eg. Double DQN to improve the robust of the model and Prioritized Experience replay so as to help the agent learn more from the important samples. In addition, a test environment may also help to test the model.