Navigation Project report

The environment

The environment is a square world with yellow and blue bananas scattered. Agent's goals is to collect as many yellow bananas as possible while avoiding blue bananas. The environment is considered solved when the agent gets an average score of +13 over consecutive 100 episodes.

Solution

I have used value based method for solving this reinforcement learning problem. I have used the DQN (Deep Q network) algorithm to learn a neural network based function approximator, which maps state to action values. This algorithm was developed by DeepMind team and more details can be found here.

I have used a fully connected neural network with an input layer followed by two hidden layers and then an output layer. Here is the description of layers:

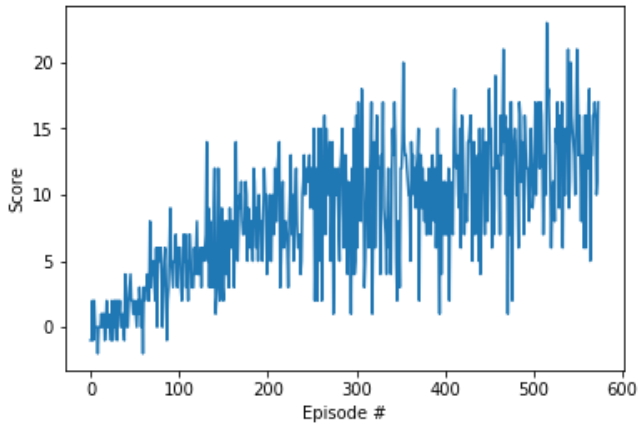
1) Input layer has 37 nodes which is equal to dimension of the state space.
2) First hidden layer has got 64 units with relu activation
3) Second hidden layer also has 64 units with relu activation
4) Output layer with 4 units – one unit for each action

The hyper-parameters used are:

```
BUFFER_SIZE = int(1e5)  # replay buffer size for storing experiences
BATCH_SIZE = 64         # minibatch size for training the network
GAMMA = 0.99            # discount factor
TAU = 1e-3              # for soft update of target parameters
LR = 5e-4               # learning rate
UPDATE_EVERY = 4        # how often to update the network
```

Most of the code is reused from LunarLander coding exercise in **Deep Q-Networks** lesson of Udacity's Deep Reinforcement Learning nanodegree.

The below plot shows how agents average score changes with each episode:



The environment was solved in 474 episodes. Below is the average score after every 100 episodes:

```
Episode 100     Average Score: 2.05
Episode 200     Average Score: 6.48
Episode 300     Average Score: 9.02
Episode 400     Average Score: 10.23
Episode 500     Average Score: 11.48
Episode 574     Average Score: 13.05
Environment solved in 474 episodes!     Average Score: 13.05
```

Future work

The algorithm here uses Experience replay and Fixed-Q targets strategy. The algorithm can be improved by incorporating below items:

- Prioritize experience replay: https://arxiv.org/abs/1511.05952
- Dueling DQN: https://arxiv.org/abs/1511.06581
- Multi-step bootstrap targets: https://arxiv.org/abs/1602.01783
- Distributional DQN: https://arxiv.org/abs/1707.06887
- Noisy DQN: https://arxiv.org/abs/1706.10295
- Above modifications to DQN are combined in algorithm called Rainbow: https://arxiv.org/abs/1710.02298