Banana Navigation project using Deep Q-Learning Network

Project report

Learning algorithm

The learning algorithm used is vanilla Deep Q Learning as described in original paper. As an input the vector of state is used instead of an image so convolutional neural network is replaced with deep neural network.

There are two identical networks used, Local and Target. The Local network is the main network that predicts the actions of the agent given a state. The Target network is the baseline network that helps reduce the variance of the weights by predicting the maximum Q-value given the state.

The deep neural network has following layers:

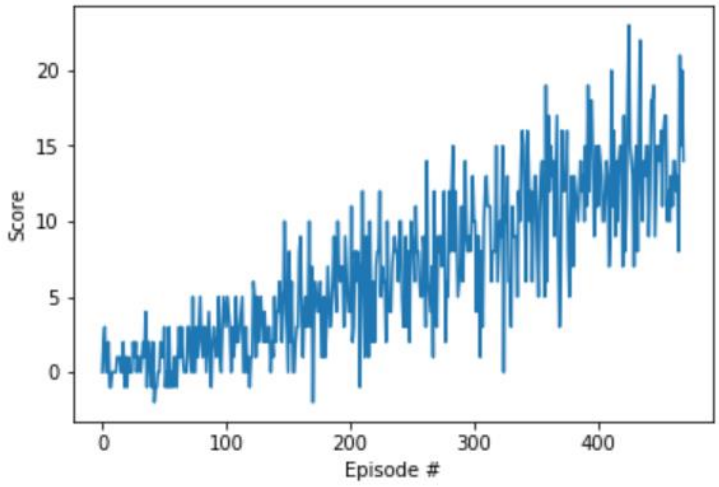
- FC1 = input: 37 (state size), output: 64
- FC2 = input: 64, output: 64
- FC3 = input: 64 output: (action size) 4

Hyperparameters:

- Starting epsilon: 1.0
- Ending epsilon: 0.01
- Epsilon decay rate: 0.995
- Batch size: 64
- Learning Rate: 0.0005
- Gamma: 0.99
- Update Networks every 4 steps

Result:

```
Episode 100     Average Score: 1.08
Episode 200     Average Score: 3.77
Episode 300     Average Score: 7.10
Episode 400     Average Score: 10.63
Episode 470     Average Score: 13.10
Environment solved in 370 episodes!     Average Score: 13.10
```



The environment was solved in 470 episodes with average score of 100 episodes from episodes 371 to 470 as +13.10.

Ideas for future work

I would like to explore ideas from the following research papers.

1. Extensive hyperparameter optimization
2. Double Deep Q Networks
3. Prioritized Experience Replay
4. Dueling Deep Q Networks
5. RAINBOW Paper
6. Learning from pixels