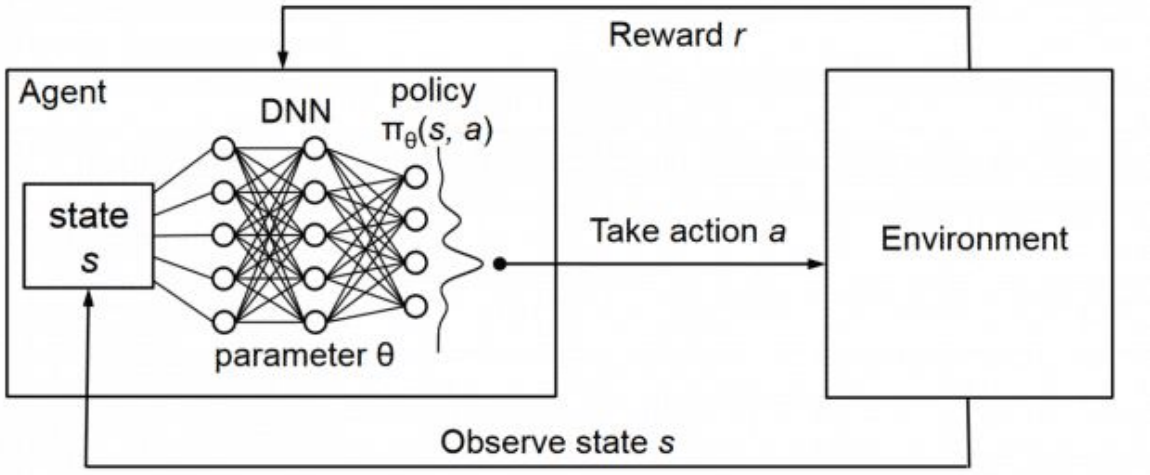
Report
Project 1 Navigation

Description

The learning algorithm used for solution of Banana Collector is a Deep Q Network written in PyTorch. Neural Network has a structure of two fully connected layers. DQN algorithm combines exploration and exploitation of the environment to come up with a better policy than it has right now. At each time step agent chooses either a random action to explore the possibilities or runs forward propagation to estimate best action for this particular state. The loop can be clearly illustrated with the diagram below.



Hyperparameters

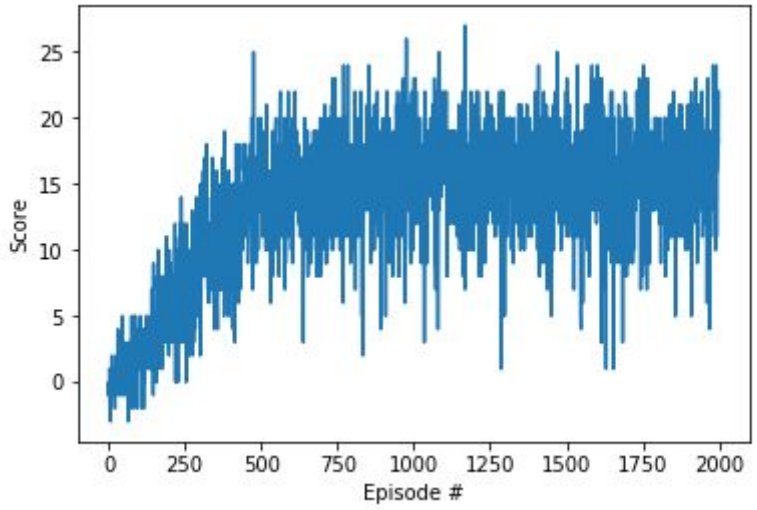
BUFFER_SIZE = 100000
BATCH_SIZE = 64
GAMMA = 0.99
TAU = 0.001
LR = 0.0005
UPDATE_EVERY = 4

Inputs = 37
Layer 1 units = 64
Layer 2 units = 64
Outputs = 4

Epsilon Start = 1
Epsilon End = 0.01
Epsilon Decay = 0.955

DQN is sensitive to hyperparameter tuning, therefore a lot of time needs to be spent on optimization.

Result



The trained model has managed to achieve average reward of 13 around 500th episode.

Future improvements

There are many ways for improving standard DQNs:

1. Dueling DQNs
2. Prioritized experience replay
3. Multi-step bootstrap targets
4. Distributional DQNs
5. Noisy DQN