Project Overview

For this project, an agent was trained to navigate (and collect bananas!) in a large, square world.

A reward of +1 was provided for collecting a yellow banana, and a reward of -1 for collecting a blue banana. Thus, the goal of my agent was to collect as many yellow bananas as possible while avoiding blue bananas.

The state space had 37 dimensions and contains the agent's velocity, along with ray-based perception of objects around the agent's forward direction. Given this information, the agent was suppose to learn how to best select actions. Four discrete actions were available, corresponding to:

0 - move forward. 1 - move backward. 2 - turn left. 3 - turn right. The task was episodic, and in order to solve the environment, my agent was suppose to get an average score of +13 over 100 consecutive episodes.

Model used

A Deep Neural Network as a function approximater was used for the Q-Learning purpose. The architecture comprised of 3 linear layers:

1. Input Dimension - 37, Output Dimension - 128
2. Input Dimension - 128, Output Dimension - 64
3. Input Dimension - 64, Output Dimension - 4

Activation function used in each layer was RelU. The model was trained using Gradient Descent using Adam Optimizer.

Agents Deployed

The agent(Q-Learning agent) used was a 3-layered Neuaral Network which directly learned the optimal q-value for the policy in place of switiching between the evaluation and improvemnet.

Results

Environment solved in 611 episodes! Average Score: 15.04

Future Improvements to be made

Continue Researching, and implementing different improvements to DQN such as prioritized experience replay, Duelling DQN, Double DQN