

Environment Set Up

Version 1: One (1) Agent

- Linux: [click here](#)
- Mac OSX: [click here](#)
- Windows (32-bit): [click here](#)
- Windows (64-bit): [click here](#)

Version 2: Twenty (20) Agents

- Linux: [click here](#)
- Mac OSX: [click here](#)
- Windows (32-bit): [click here](#)
- Windows (64-bit): [click here](#)

Then, place the file in the `p2_continuous-control/` folder in the DRLND

GitHub repository, and unzip (or decompress) the file.

(For Windows users) Check out [this link](#) if you need help with determining if your computer is running a 32-bit version or 64-bit version of the Windows operating system.

(For AWS) If you'd like to train the agent on AWS (and have not [enabled a virtual screen](#)), then please use [this link](#) (version 1) or [this link](#) (version 2) to obtain the "headless" version of the environment. You will **not** be able to watch the agent without enabling a virtual screen, but you will be able to train the agent. (To watch the agent, you should follow the instructions to [enable a virtual screen](#), and then download the environment for the **Linux** operating system above.)

Learning Algorithm:

I use the optional DRL for finance code for agent (DDPG) and model. A larger hidden units size is needed for more complicated problem. I also introduced two parameters controlling for update frequency.

Agent:

`BUFFER_SIZE = int(1e6) # replay buffer size`

`BATCH_SIZE = 128 # minibatch size`

`GAMMA = 0.99 # discount factor`

TAU = 1e-3 # for soft update of target parameters

LR_ACTOR = 1e-4 # learning rate of the actor

LR_CRITIC = 1e-4 # learning rate of the critic

WEIGHT_DECAY = 0.0 # L2 weight decay

N_LEARN_UPDATES = 10 # number of learning updates

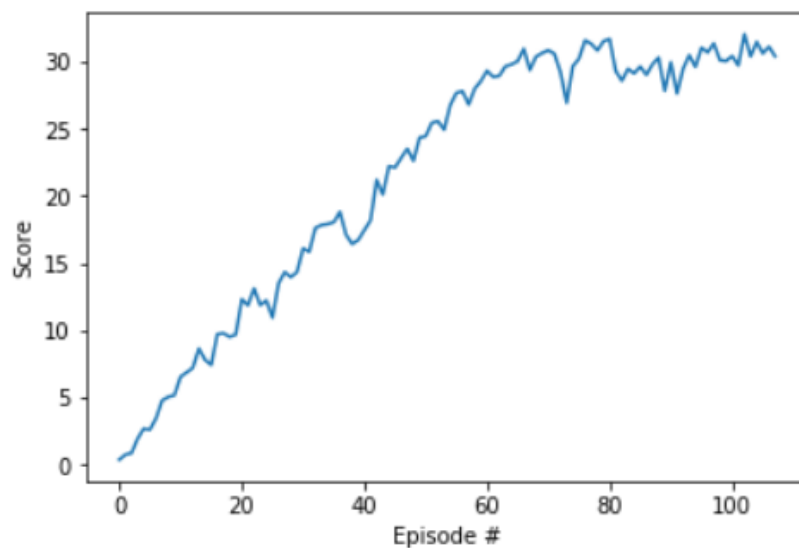
N_TIME_STEPS = 20 # every n time step do update

Model (for both Critic and Actor):

Hidden units fc1_units = 256

Hidden units fc2_units=128

Plot (Version 2): solved in 108 episodes with average score of 30.12959232655111



Future Ideas: Use models like A2C, A3C, GAE