Stock Trading Agent Using Reinforcement Learning

Developer Manual

## Run the code on google colab using following steps

## Training Phase :

Title of the Colab Notebook for training is StocksTraderAgent(Training).

1. Click runtime option from the main menu and select **reset all runtimes** option and wait until it connects to the runtime
2. **Upload** the pickle file(eg. FYP.pkl) in the files section present on the left of the screen using the upload option.
3. Before running the first cell enter the GitHub link of your folder which contains the environment on line 7 of the first cell.
4. Run the first cell of the code which contains all the install and importing commands wait till it completes execution.
5. Again Click runtime option from the main menu but this time select **restart runtime** option.Wait till the notebook reconnects.
6. From the three algorithm cells run **any one** cell depending upon the algorithm among trpo,vpg and ppo the user wants to apply.
7. Profit.txt and action\_seq.txt files will be created and visible in the files section , user can open these files during execution.
8. profit.txt contains the profit earned by the method and the action\_seq.txt specifies the actions 0=buy 1=sell 2 = do nothing , a single line represents actions executed in a single episode.
9. Last cell contains code to plot the profit made by the method.In Order to run this cell you need to have the profit.txt file in the file section.
10. The algorithm cell will completely execute in 2-3 hours.
11. After complete execution download all the generated files from the ./path folder so that trained model can be saved and used, also download the action-seq.txt and profit.txt

## Testing Phase : Title of the Colab Notebook for training is StocksTraderAgent(Testing).

1. In your github file, change the value of stocks\_open variable on the 18th line to d[‘Apple\_test\_open’] and stocks\_close variable on the 19th line to d[‘Apple\_test\_close’].
2. Depending on the data used for testing, change the max\_stride variable and series\_length variable in \_\_init\_\_ function. For Example : For testing the apple stock use max\_stride = 1 and series\_length=220.
3. Click runtime option from the main menu and select **reset all runtimes** option and wait until it connects to the runtime.
4. Create a compressed ‘path’ folder on your local computer using the files downloaded at the end of the training phase. The directory contents should be in the same structure as downloaded during training

The directory structure should be like this:

path/to/

path/to/output\_dir/

path/to/output\_dir/config.json

path/to/output\_dir/progress.txt

path/to/output\_dir/simple\_save/

path/to/output\_dir/simple\_save/model\_info.pkl

path/to/output\_dir/simple\_save/saved\_model.pb

path/to/output\_dir/simple\_save/variables/

path/to/output\_dir/simple\_save/variables/variables.data-00000-of-00001

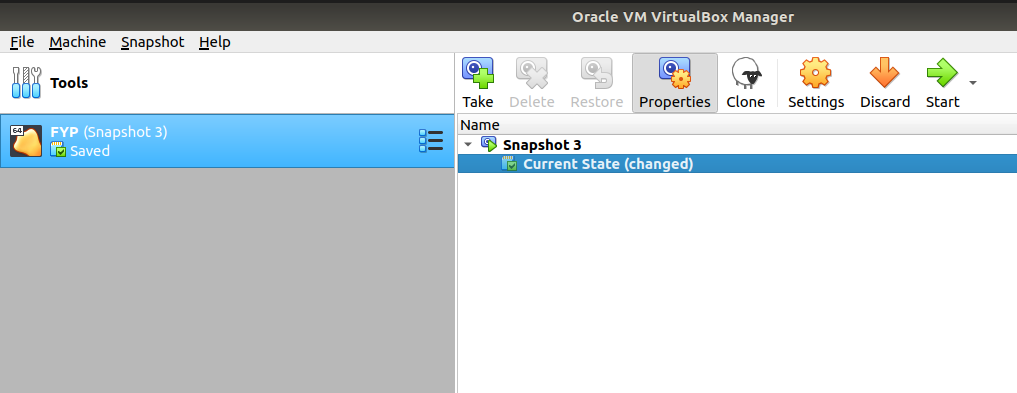
path/to/output\_dir/simple\_save/variables/variables.index

path/to/output\_dir/vars.pkl

1. Upload the path.zip folder in the files section
2. Run the 2nd cell of the notebook which will unzip this file
3. Run the 3rd cell which will be used for testing
4. If there is an error which says ‘environment not found’, run the 4th cell
5. The profit of the 100 episodes (should be the same for all) will be printed in the profit.txt file generated in the files section
6. The action sequence consisting of the trading decisions should be in the act\_seq.txt file generated in the files section

VM Oracle (alternate):

1. Open the VM Virtualbox and install Mint OS in the virtualbox
2. Load the .sav snapshot(click on start button)



1. Open Terminal
2. Go to spinningup folder (type cd spinningup)
3. type python fyp.py
4. Inside stocks folder inside spinningup folder, you will find StocksEnv.py which is the github link for colab.
5. You will also find aplmsfopenclose.pkl which can be replaced with the data you have obtained from other sites.