Simple Q-Learning for Trading Demo

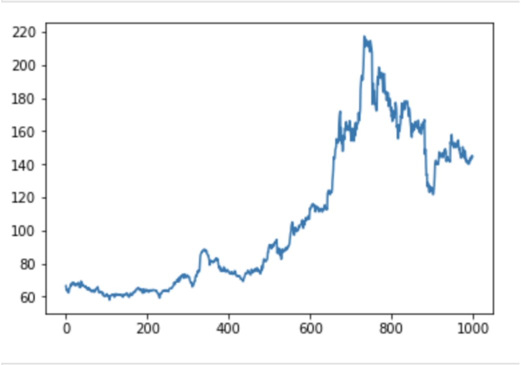
Ce Wang

March 13, 2019

1 Introduction

This is an experiment of using Q-Learning for portfolio trading and optimization. Since no neural network is being considered for value function approximation for large state space, we only consider trading a single stock, by making decisions whether to hold stock or cash on each trading day.

The data we are using is the daily closing prices of Coke during Nov. 7, 2012 to Nov. 7, 2017. There are about 1250 trading days in the above period. We use the first 1000 trading days as the training set, and the rest part as the testing set. The following diagram shows the movement of the stock during the training period:



2 Markov Decision Process

We formulate the problem with the following Markov Decision Process:

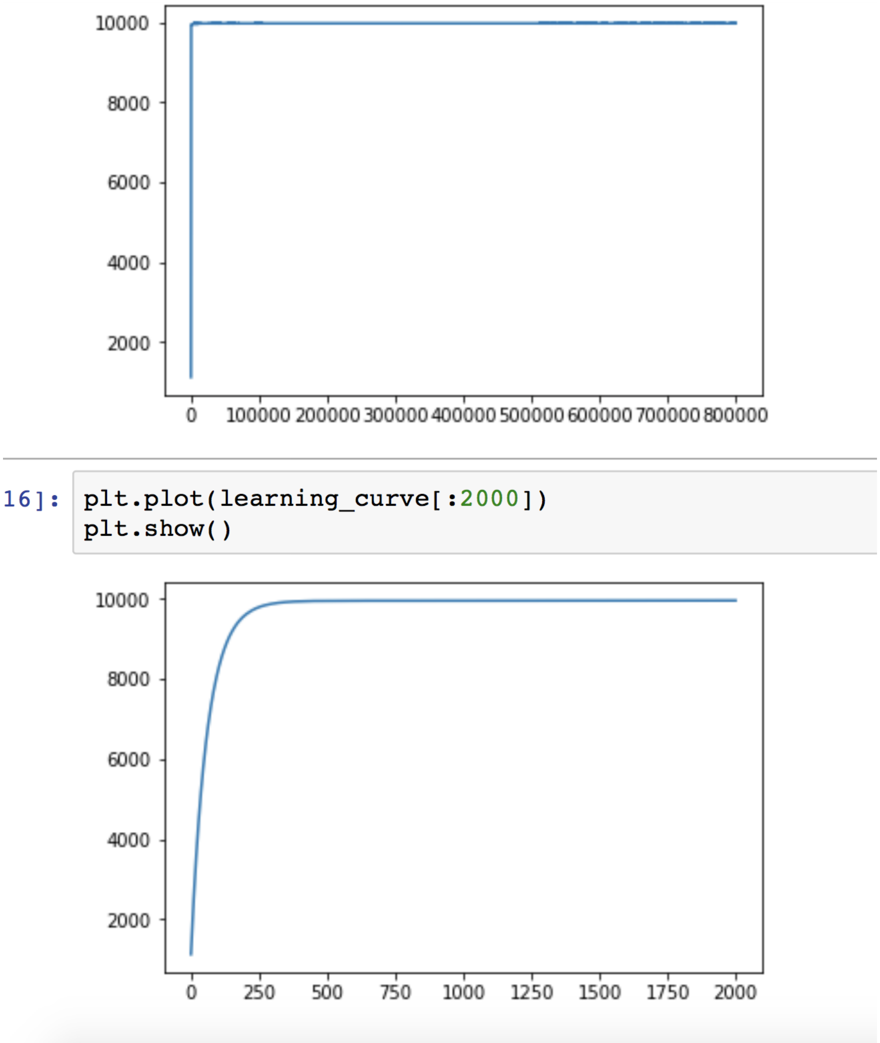
- **State**: We consider 32 different stats. Each state can be represented as $\{s_1, s_2, s_3, s_4, s_5\}$, where $s_1 \in \{0, 1\}$. $s_1, ..., s_4 = 0$ or 1 when the stock is going up or not on 3 days ago, 2 days ago, yesterday, and today. s_5 is 0 or 1 whether the volatility in the past 5 days is low or high.

- **Action**: On each trading day, actions are holding cash or holding the stock.

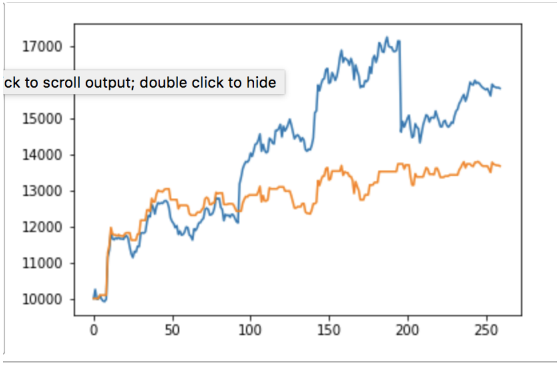
- **Reward**: If action is to hold cash, reward=1. If action is to hold stock, reward is Price next day \ Price today

3 Results and Diagrams

From the learning curve of training iterations, we can see that it's enough to train the Q-table for about 500 iterations. Using the model trained for 500 times to test on the testing set, the model presents the following performance: about 500 iterations. As we see, the performance of our algorithm (orange) did not actually beat the market (blue), ut it does not look too bad.



```
16]: plt.plot(learning_curve[:2000])
     plt.show()
```


ck to scroll output; double click to hide