(Reinforcement Learning using Q-Learning)

Rushabh Shah
(50375759)

1 Abstract:

The goal of this project is to train the agent to learn the trends in stock market prices and use it to perform trades so as to maximize the profit. The agent is provided with an initial amount to work upon. In order to implement the project, the q-learning algorithm is implemented from scratch.

2 Dataset:

Stock price for Nvidia for the last 5 years.The total number of examples are 1258 which have been divided into train and test datasets. The various features included are as follows -

- Opening price
- Closing price
- Intraday high and low
- Adjusted closing price
- Volume of shares traded for each day

3 Environment Description

The stock trading environment has been provided on which the agent trains using the Q-Learning algorithm. The environment describes various details regarding the states, actions and so on.

The environment consists of the following methods -

1) Init - This method is used to initialize the environment and its variables.
2) Reset - resets all the variables and returns the observation/state.
3) Step - This method executes when the agent performs some action and returns the reward and observation as the primary parameters.
4) Render - For plotting total account value over time.

Possible states - There are in total 4 possible observation/states. The state is represented by an integer in the range of 0-3.

- 0 represents an increase in price and the agent does not have stock
- 1 represents an increase in price and the agent has stock
- 2 represents a decrease in price and the agent does not have stock
- 3 represents a decrease in price and the agent has stock

Actions - 3 possible actions (Buy,Sell and Hold)

Goal - To maximize the total account value over the number of days considered.

Rewards - Values that are awarded to the agent in response to an action taken. Calculation of reward takes into consideration the penalty associated when some invalid action is performed. The instances of penalty include when the agent tries to buy the shares with no capital as well as when the agent tries to sell the shares despite having no shares at hand.

4 Introduction

Reinforcement learning is the machine learning technique where the agent interacts and learns from its surroundings/environment.

5 major concepts -

1) Action - possible moves the agent can take.
2) State
3) Reward - immediate reward obtained by the agent.
4) Policy - Strategy taken by the agent to choose the best possible action.
5) Value - Total reward

We make use of the Q-Learning algorithm in order to learn the optimal policy for taking the best action based on the state.

Q-Learning makes use of a 2-D table to store the values for state-action pairs. Initially all the values are set to zero and in subsequent iterations, the table is updated.

After every action the agent takes, the q-table is updated.

During the learning task there are two important tasks that the agent performs-

1) Exploration - It involves the agent taking actions to obtain more data.
2) Exploitation - Using the existing information to select the best possible action.

5 Implementation

We start by initializing the variables for the agent like discount factor, q-table, learning rate etc.

A policy function is defined which returns the action for the agent to perform. The action is chosen either by exploration or exploitation. In order to do this, we define a threshold value above which the agent will explore the environment and below which the agent will exploit to choose the best possible action based on the current information available.

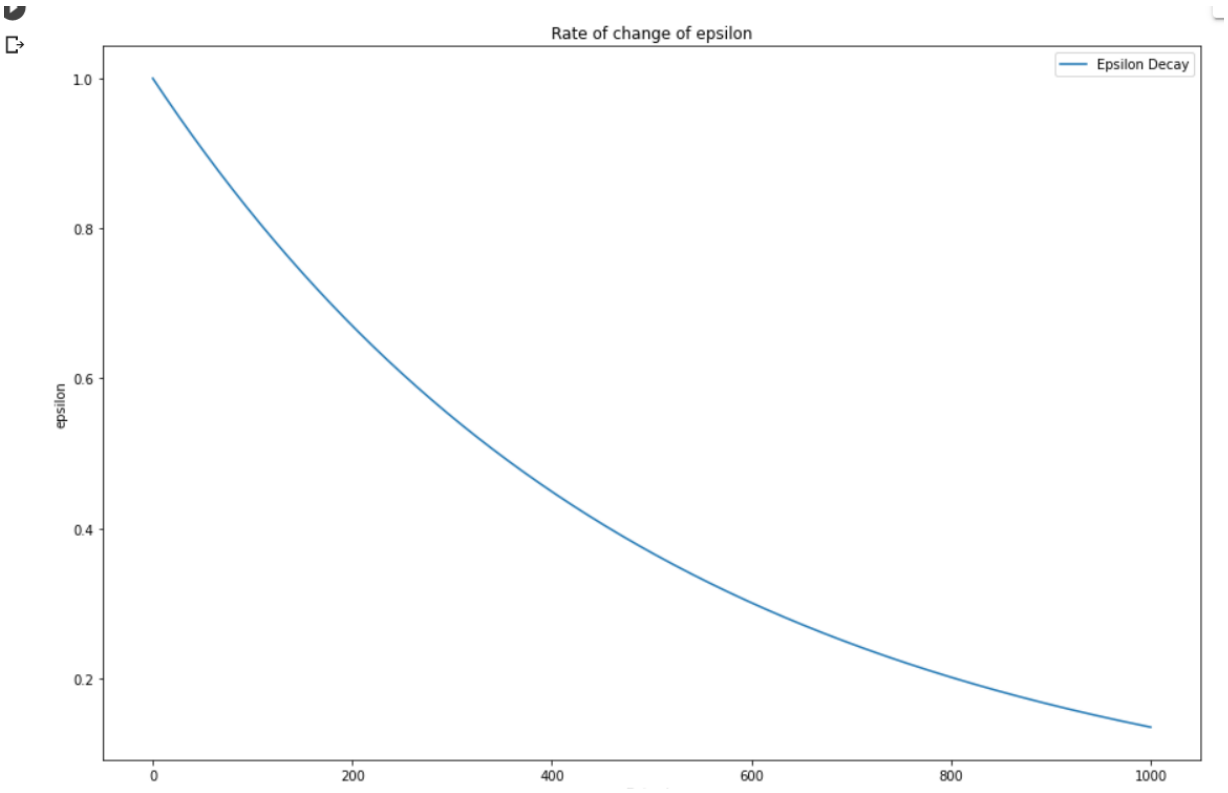
In an ideal scenario, it is prudent to explore during the initial phase of training the agent and then employ exploitation.
We define epsilon and initially set it to 1 which decreases by a factor of 0.998(decay rate) upon completion of each episode. When the epsilon value reaches below the explore_flag, the agent employs exploitation; else it employs exploration.

Agent is then trained and after each step taken by the agent, the q-table is updated according to the following rule

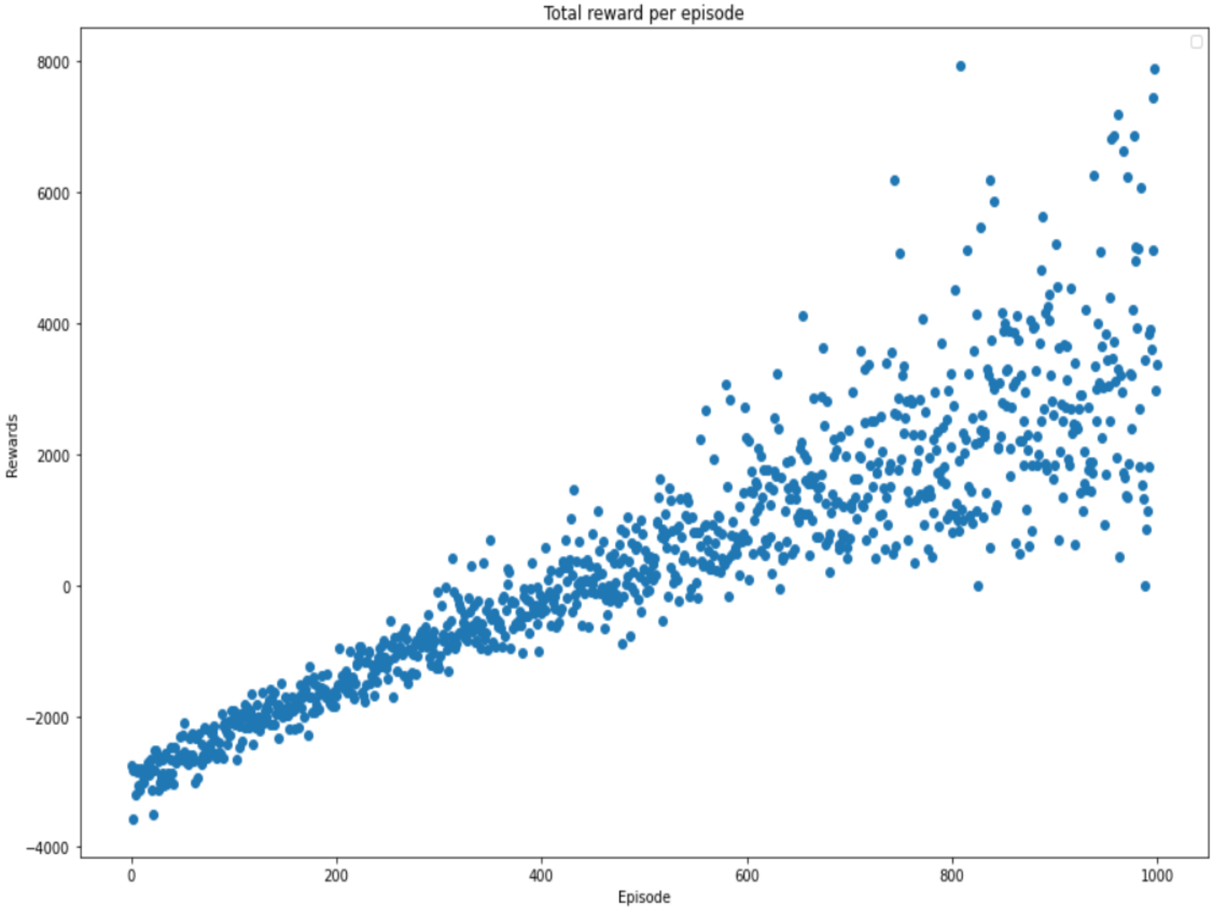
Q’(s,a) = (1 - alpha) Q(s,a) + alpha(reward + discount_factor x Q(s,a))
Where alpha is the learning rate

Results:

The plot for epsilon decay is as shown

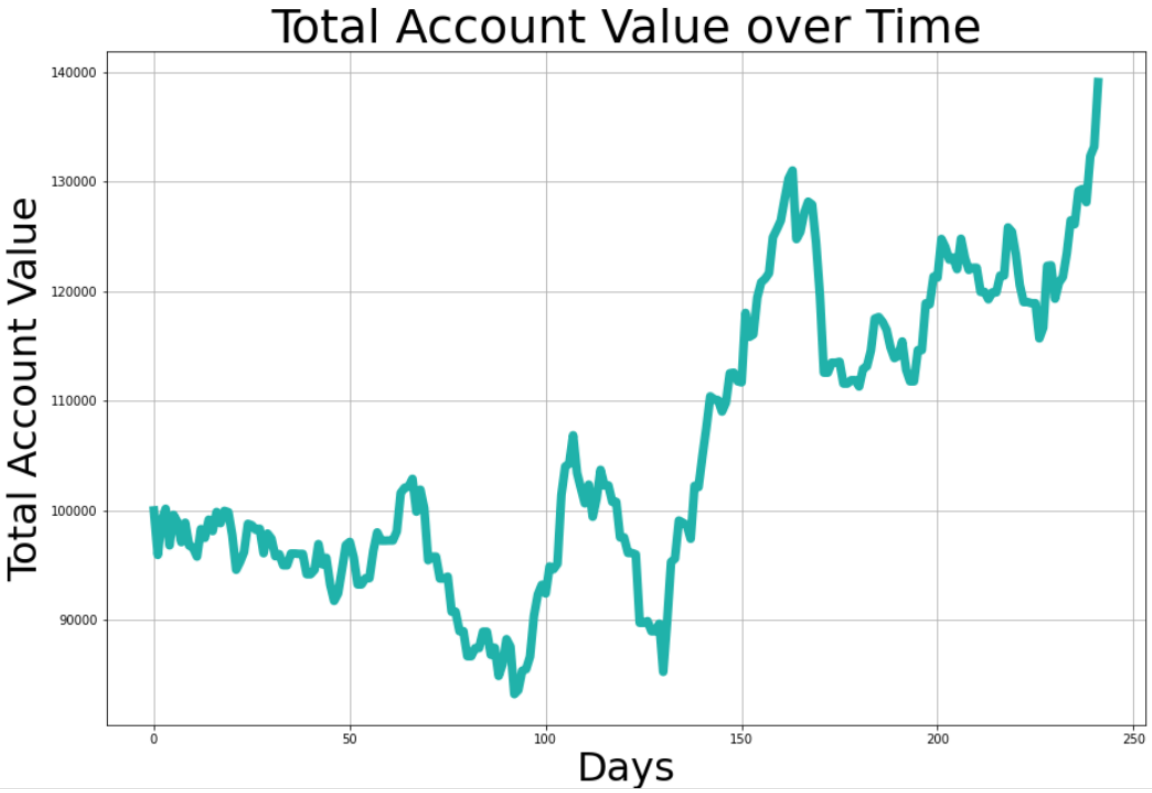


The plot for total reward per episode is as shown



Agent’s performance on test dataset:

The performance of the agent is evaluated on the test dataset by choosing the best action from the current state(greedy policy) using the learnt policy.



Total account value is 139095.30966700005

5 References

1) https://gym.openai.com/docs/#spaces
2) https://numpy.org/doc/stable/reference/random/generated/numpy.random.random_sample.html#numpy.random.random_sample

3) https://en.wikipedia.org/wiki/Q-learning