Capstone Proposal: Predicting Stock Prices with LSTM

Udacity Machine Learning NanoDegree 2020

AJ Miller
March 2020

1. Domain Background

Predicting stock prices or any market's value is a rather popular subject. Individuals and firms have been creating financial models to predict the future for centuries now. The onset of the information age made a plethora of data available, often free, but until recently we couldn't utilize it all let alone efficiently. With the advances in computers and algorithms we are able to use a subset of artificial intelligence (AI), machine learning (ML), to use this information and make more accurate predictions.

2. Problem Statement

Using past data to create a model that can predict how volatile a stock will be in the future. The past data (Input) will include opening price (Open), highest and lowest daily price (High, Low), how many stocks were traded (Volume), closing price (Close) and the adjusted closing price (Adjusted Close) which factors in dividends, stock splits and new stock offerings.

3. Datasets and Inputs

We'll be using 20 random and hand picked stocks from the S&P 500 with at least a 10 year history as our training set. This will help reduce the amount of training data, and should reduce noise from IPOs and junk stocks. The list can be found on Wikipedia: https://en.wikipedia.org/wiki/List_of_S%26P_500_companies

There are many options for getting the stock data, and luckily the data is well labeled. The main thing we have to do to the data will be scaling (normalizing) it between 0 and 1. AlphaVantage offers a free service that includes all data we need and a good history. Some other good options are Yahoo! Finance, Bloomberg API, Quandl.

Below is an example of one day of the Microsoft stock (MSFT). https://www.alphavantage.co/

```
{
   "Meta Data": {
      "1. Information": "Daily Time Series with Splits and Dividend Events",
      "2. Symbol": "MSFT",
      "3. Last Refreshed": "2020-03-03",
      "4. Output Size": "Compact",
      "5. Time Zone": "US/Eastern"
   },
   "Time Series (Daily)": {
      "2020-03-03": {
         "1. open": "173.8000",
         "2. high": "175.0000",
         "3. low": "162.2600",
         "4. close": "164.5100",
         "5. adjusted close": "164.5100",
         "6. volume": "71577263",
         "7. dividend amount": "0.0000",
         "8. split coefficient": "1.0000"
      }
}
```


4. Solution Statement

Since this is time series forecasting we'll create and test a Recurrent Neural Network (RNN) with a specialized layer called Long Short Term Memory (LSTM) to predict a stock's adjusted close from the previous 2 months. This makes the history size of 40, which is the estimated trading days for 2 months, and a target size of 1.

5. Benchmark Model

Since this is a linear regression problem, we'll compare our model to a Support Vector Machine (SVM) and the actual market values which I would argue is just a bunch of models setting the price based on their own predictions.

6. Evaluation Metrics

The two metrics used to evaluate the models are R-square and Root Mean Square Error (RMSE). R-squared is a measure in percent of how close the value is to the fitted regression line. RMSE is the average deviation from the true values.

7. Project Design

First we'll install and import the library's we'll need for preparing the data, visualization, training. After we load, validate, split, standardize and visualize the data we can build, train, test and deploy the model. Once everything is tested we'll create a website to utilize the model.