Machine Learning Engineer Nanodegree

Proposal for a Stock predictor model

Ankit Maurya
June 15, 2018

Domain Background

Investment firms, hedge funds and even individuals have been using financial models to better understand market behaviour and make profitable investments and trades. A wealth of information is available in the form of historical stock prices and company performance data, suitable for machine learning algorithms to process.

In this project we will predict a timeseries composed of stock prices using machine learning . Big investment firms are using more and more of machine learning algorithms to predict the stock prices . The financial advisors are getting replaced with machine learning algorithms. Here we will predict the stock price of a large company listed on NYSE stock exchange , given it's historical performance.

We will use Long short-term memory models which are a special case of RNNs, to predict the stock prices here . Recent research shows the LSTM are the most useful variant of RNNs.

Problem Statement

The challenge of this project is to accurately predict the closing value of a given stock across a given period of time in the future. I will predicting the Adj.Close of a stock listed on NYSE stock exchange. The adjusted closing price is a stock closing price after it has been amended to include any dividend, split, or merge.

I will be using LSTM for the prediction . The goal is to first build a model using linear regression and then use LSTM to improve the prediction.

Datasets and Inputs

Financial data can be expensive and hard to extract, that's why in this project we use the Python library quandl to obtain such information. The library has been chosen since it's easy to use, cheap (XX free queries per day), and great for this exercise, where we want to predict only the closing price of the stock.

Quandl is an API, and the Python library is a wrapper over the APIs.

The format is a CSV, and each line contains the date, the opening price, the highest and the lowest of the day, the closing, the adjusted, and some volumes. The lines are sorted from the most recent to the least. The column we're interested in is the Adj. Close, that is, the closing price after adjustments.

A snapshot of data looks like this :

```
curl "https://www.quandl.com/api/v3/datasets/WIKI/FB/data.csv"
Date,Open,High,Low,Close,Volume,Ex-Dividend,Split Ratio,Adj.
Open,Adj. High,Adj. Low,Adj. Close,Adj. Volume
2018-03-
27,156.31,162.85,150.75,152.19,76787884.0,0.0,1.0,156.31,162.85,150.
75,152.19,76787884.0
2018-03-
26,160.82,161.1,149.02,160.06,125438294.0,0.0,1.0,160.82,161.1,149.0
2,160.06,125438294.0
2018-03-
23,165.44,167.1,159.02,159.39,52306891.0,0.0,1.0,165.44,167.1,159.02
,159.39,52306891.0
2018-03-
22,166.13,170.27,163.72,164.89,73389988.0,0.0,1.0,166.13,170.27,163.
72,164.89,73389988.0
2018-03-
21,164.8,173.4,163.3,169.39,105350867.0,0.0,1.0,164.8,173.4,163.3,16
9.39,105350867.0
```

Solution statement

LSTM looks looks to be the best approach for this problem . We will use tensor flow for easy implementation of LSTM . The project will be programmed in Jupyter Notebook (python 3) for easy presentation . Numpy library will be used for easier management of the data set . The performance measurement will be based on comparison with the benchmark model which will use logistic regression.

Benchmark Model

I am initially tempted to approach the problem as a regression problem. In this case, the regression is very simple: from a numerical vector, we want to predict a numerical value. That's not ideal. Treating the problem as a regression problem, we force the algorithm to think that each feature is independent, while instead, they're correlated, since they're windows of the same timeseries. In order to evaluate the model, we now create a function that, given the observation matrix, the true labels, and the predicted ones, will output the metrics (in terms of mean square error **(**MSE**)** and mean absolute error **(**MAE**)** of the predictions.

Evaluation Metrics

The model will be predicted using the MSE and MAE between the actual and predicted close values for the stock. Also we will compare the delta between the performance of the benchmark model (linear regression model) and the LSTM model.

Project Design

The project will be broken into the following components :

- How to collect the historical stock price information

    1. We will use Python library quandl to obtain the stock data

- Set up the infrastructure

    1. We will use a iPython Notebook
    2. Install the required libraries(like quandl,numpy,matlibplot,tensorflow etc.)
    3. Prepare the data using numpy.

- Use regression to predict the future prices of a stock

    1. Build a benchmark model using regression.
    2. Tune the parameters

- Develop LSTM model

    1. Develop the LSTM model
    2. Tune the hyperparameters

- Visualize the performance

    1. Visualize the performance using Tensor board
    2. Describe results for report