Stock Market Prediction Using LSTM, SVM and Linear Regression techniques

**EEN 351 Course Project**

Aayushi Shrivastava Harshit Sharma

15115002 15115055

[ashigwl196@gmail.com](mailto:ashigwl196@gmail.com) [harshit.sharma.3105@gmail.com](mailto:harshit.sharma.3105@gmail.com)

**ABSTRACT**

Predictions on stock market prices are a great challenge due to the fact that it is an immensely complex, chaotic and dynamic environment. There are many studies from various areas aiming to take on that challenge and Machine Learning approaches have been the focus of many of them. There are many examples of Machine Learning algorithms been able to reach satisfactory results when doing that type of prediction.

In this project we applied linear regression, support vector machine and LSTM approach to predict prices of stocks. For predicting the stocks we used the NYSE S&P 500 dataset.

The link to the dataset is given in references.

**INTRODUCTION**

We studied the movement of stock prices for companies listed in NYSE over the course of several years. Machine learning algorithms like Linear Regression and Support Vector Machine were applied along with a deep learning technique i.e LSTM (Long Short Term Memory).

**Dataset**

The S&P 500 (Standard and Poor's 500) is a free-float, capitalization-weighted index of the top 500 publicly listed stocks in the US (top 500 by market cap). The dataset contains information of historical stock prices (last 5 years) for all companies currently found on the S&P 500 index. The dataset contains files of data for individual stocks, labelled by their stock ticker name.

All the files have the following columns: Date - in format: yy-mm-dd

Open - price of the stock at market open

High - Highest price reached in the day

Low Close - Lowest price reached in the day

Volume - Number of shares traded

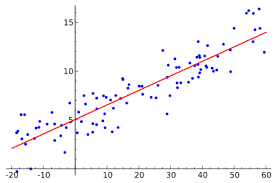
Name - the stock's ticker name.

**Linear Regression**

Linear regression is one of regression method to be used for classifying numerical class . It creates linear function by calculating weight values (w) for each feature (b). The function can be seen as follow:

X= b0\*w0 + b1\*w1 + .... + bn\*wn

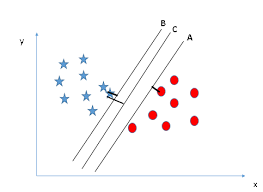
X are regression value for one instance of data. There is a linear line that represents the distribution of data. Distance from each instance data to the linear line is called error or residual. The linear function is created by searching appropriate weight value for each feature to minimize error of each instance data (mean error). A linear regression model looks like one shown in the figure where the blue points are instances or datapoints and the red line is our hypothesis.



**Support Vector Machines**

SVM is classifier which can be used for binary as well as multi class classification. The way it works is that it creates a decision boundary such that most points in one category fall on one side of the boundary while most points in the other category fall on the other side of the boundary. Then elements in one category will be such that the sum is greater than 0, while elements in the other category will have sum less than 0. The optimal hyperplane is such that we maximize the distance from the plane to any point is known as the margin. The points which are situated closest to the boundary are the ones which matter in the selection of the separating hyperplane. These points are called support vectors. The hyperplane is known as Support Vector Classifier (SVC).

SVC is restricted by its linearity i.e it only produces linear boundaries. SVM uses several kernel functions (Linear, RBF, Polynomial) to map the input space to a higher dimensional space in which the data is linearly separable.



The above figure represents a support vector classifier with the line C being the margin and the support vectors the closest points to A and B respectively.

**Kernel Functions**

Kernel is a way of computing the dot product of two vectors **x** and **y** in some high dimensional feature space. Kernels give a way to compute dot products in some feature space without even knowing what this space is and what is the mapping function. The kernels used in the project are linear, RBF and polynomial.

**LSTM**

LSTM stands for Long Short Term Memory.

An LSTM cell consists of input gate, cell state, forget gate, and output gate. It also consists of sigmoid layer, tanh layer and point wise multiplication operation.The various gates and their functions are as follows :

1. Input gate : Input gate consists of the input.

2. Cell State : Runs through the entire network and has the ability to add or remove information with the help of

gates.

3. Forget gate layer: Decides the fraction of the information to be allowed.

4. Output gate : It consists of the output generated by the LSTM.

5. Sigmoid layer generates numbers between zero and one, describing how much of each component should be let

through.

6. Tanh layer generates a new vector, which is added to the state.

The cell state is updated based on the outputs form the gates. Mathematically we can represent it using the following

equations.

ft= σ(Wf.[ht−1, xt] + bf) (2)

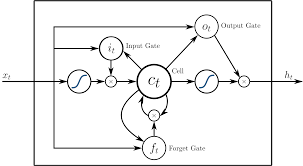
it= σ(Wi.[ht−1, xt] + bi) (3)

ct= tanh(Wc.[ht−1, xt] + bc) (4)

ot= σ(Wo[ht−1, xt] + bo) (5)

ht= ot∗tanh(ct) (6)

where xt: input vector, ht: output vector, ct: cell state vector, ft: forget gate vector, it: input gate vector, ot: output gate vector and W,b are the parameters.

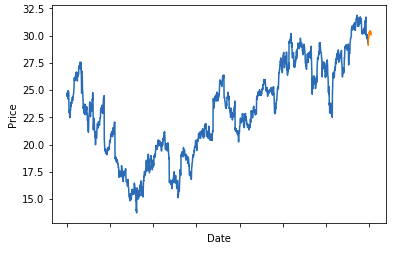


The above figure is labelled with all the layers and gates in an LSTM cell.

**RESULTS**

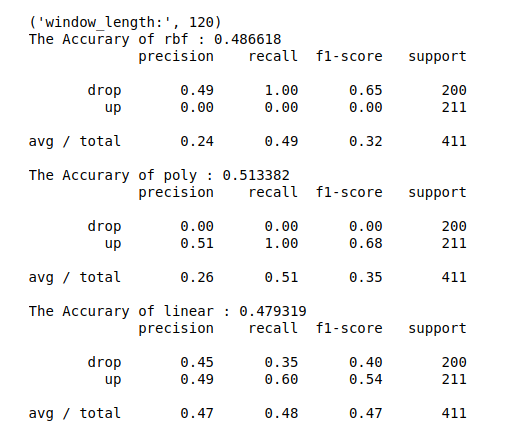
The results of the experiment are attached as jupyter notebooks alongwith the code.

Results are shown here pictorially.

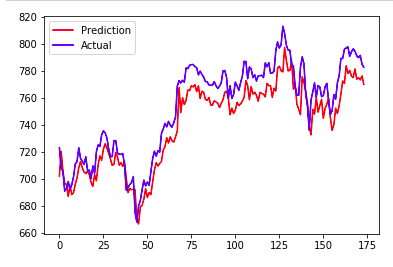
****

The above figure shows the prediction using the linear regression model. The confidence score for the model is 89.26%. The accuracy of the linear model ranges from 45-55 %.

The following image shows the results of different SVM kernels applied to the dataset. We have used 3 kernels to predict the prices, namely RBF, Linear and Polynomial. Among these 3 kernels polynomial kernel has better accuracy than the linear and RBF one. RBF kernel although can map the data to infinite dimensional space, this particular thing is the reason why it isn’t able to achieve a better accuracy. The data actually exists in a manifold which is difficult to determine through a RBF kernel.



The 3rd and final method used is an LSTM network. The network has 2 LSTM layers followed by 2 Fully connected layers. Each LSTM layer consists of 256 cells. A constant dropout of 0.3 has been applied across both these layers. The penultimate layer has ReLU activation and the last layer has linear activation. The model was trained for 100 epochs with a batch size of 512 and a validation split of 0.1.



This model performed way better than the previous 2 models. It incurred a loss of 0.01 RMSE training error and 0.02 RMSE test error.

**CONCLUSIONS**

It is evident from the results that the accuracy of SVM and regression model is almost the same ranging from 50-60% and 45-55% respectively. The accuracy of LSTM model is in excess of 75%. We can safely conclude that the LSTM model is a far better alternative than other traditional machine learning algorithms. With larger datasets like the one used in this experiment, these models (deep learning models) perform exceptionally better than their older counterparts.

**FURTHER WORK**

The advent of deep learning techniques has really enhanced our understanding and capabilities when dealing with larger datasets like the one in this experiment. The accuracies for these kinds of tasks are increasing very frequently because of a large amount of ongoing research in this field. As we move on to better hardware and better in depth understanding of the subject we can keep on improving our techniques and discover some new techniques in the process.

**REFERENCES**

1. Analysis and prediction of Stock Prices <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6740438>

2. Stock market’s price movement prediction using LSTM neural networks <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7966019>

3. Stock price predicition using LSTM, RNN and CNN sliding-window model

<https://www.researchgate.net/publication/321503983_Stock_price_prediction_using_LSTM_RNN_and_CNN-sliding_window_model>

4. Stock market prediction using Support Vector Machine

<https://ieeexplore.ieee.org/document/6703096/>

5. Stock trend prediction and technical indicators using SVM

[http://cs229.stanford.edu/proj2014/Xinjie%20Di,%20Stock%20Trend%20Prediction%20with%20Technical%20Indicators%20using%20SVM.pdf](http://cs229.stanford.edu/proj2014/Xinjie Di, Stock Trend Prediction with Technical Indicators using SVM.pdf)

6. Support Vector Machines

<http://web.cs.iastate.edu/~honavar/hearst-svm.pdf>

7. Regression techniques for prediction of stock price trend

<https://ieeexplore.ieee.org/document/6396535/>

8. Stock trend prediction using regression – A data mining approach

<https://www.researchgate.net/publication/277409163_Stock_Trend_Prediction_Using_Regression_Analysis_-_A_Data_Mining_Approach>

9. LSTM networks architectures for large scale acoustic modelling

<https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/43905.pdf>

10. Linear Regression

<http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm>