Stock Prediction

By

Gautam.R.A

Srinidhi Bharadwaj

G Phalguna Rajkumar

Submission date: 11-Dec-2021 11:35AM (UTC+0530)

Submission ID: 1727397925

File name: IEEE_Conference-Paper_template_2.doc (955.5K)

Word count: 1094

Character count: 5745

Stock Prediction

Gautam R A
Computer Science
PESU,RR
Bangalore,Karnataka,India
gautamra624@gmail.com

Srinidhi Bharadwaj G R
Computer Science
PESU,RR
Bangalore,Karnataka,India
srinidhi.pes143@gmail.com

G Phalguna Rajkumar
Computer science
PESU RR
Bangalore,Karnataka,India
phalgunagopal@gmail.com

***Abstract*—With the rise in awareness of the stock market amongst the common people, we observe more and more people pouring into the market with an attempt to make money, especially the youth. Hence an efficient way of predicting stock catches the public's eye. We therefore have made an attempt to predict the stocks using the concept of linear regression model and finding a way out with the maximum root mean square error that we are able to achieve.**

***Keywords—Root mean square, linear regression model** (key words)*

I. Introduction

The money market is one of the most famous and well-known markets that the public is aware of and a lot of people here make a whole life out of just predicting the way the stock market goes. One thing everybody knows about the market is that no one can ever predict it a 100%. So we here have tried to refer several previous research papers by others and have tried to build on it by increasing the efficiency of predicting stock as compared to the already known methods of predicting stock. We here have used 2 models to predict the stock by plotting graphs. Both models involve prediction of stock using graphs but with slightly different approaches. The 2 approaches we have used, namely, the linear regression model and the stacked LSTM model, are the methods we thought would best fit our needs. Any stock data we get of any company always 5 or 6 parameters, namely, opening price, closing price, volume, high and low. In the Linear regression model, we have used the data of the apple stocks from previous few years as we will be seeing in detail in the upcoming sections.

II. Predicting techniques

1. Linear regression

With the given data set, we plot a graph between the date and a suitable parameter. We get a scatter plot and hence we plot a line which goes through maximum number of points from the scatter plot called the linear regression line. In reality, we can't draw a line that fits through every single point of the scatter plot because the data isn't perfect. So we plot a line which passes through maximum number of points.

Linear Regression is a model that estimates a relationship between the independent and the dependent variables ,where in the dependent variable is continuous in nature. In our model, the dependent variable we have chosen is the closing price for stocks. Independent variables are model dependent and are chosen according to the model.

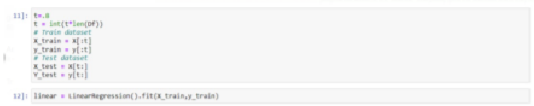
In linear regression the relationship between the dependant and the independent variable is linear in nature,ie for increase in independent varibales we can observe linear increase in dependent varibales.

```
In [8]: DF['5_3'] = DF['Close'].shift(1).rolling(window=3).mean()
        DF['5_9'] = DF['Close'].shift(1).rolling(window=9).mean()
        DF= DF.dropna()
        X = DF[['5_3','5_9']]
        X.head()
```

Out[8]:

Date	5_3	5_9
2008-01-14	6.310833	6.535357
2008-01-15	6.303452	6.458730
2008-01-16	6.199548	6.356349
2008-01-17	6.041191	6.216310
2008-01-18	5.928214	6.140278

We have used rolling window calculations to forecast stock prices as it is a time series data and also used moving average technique as a normalisation technique in the above given module where in the calculations are done for a specific window. In thhe above model we have taken the mean of the window of 3 values.

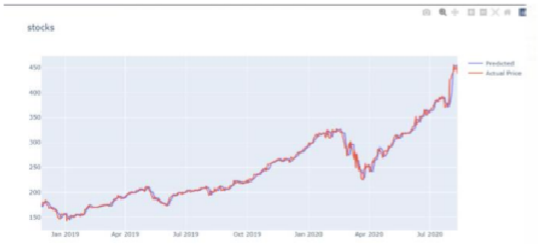


We have split the data set as training and testing data set in the ratio of 0.8:0.2 respectively. The training data set is used to train the linear regression model and later the prediction is made using the testing data set.

By using the data of an apple stock from the given years, we have plotted the graph and compared it's accuracy to the actual data to find the efficiency of our analysis:

```
predicted_price = pd.DataFrame(predicted_price,index=y_test.index,columns = ['price'])
predicted_price.plot(figsize=(10,5))
y_test.plot()
plt.legend(['predicted_price','actual_price'])
plt.ylabel("AAPL Price")
plt.show()
```

Plot:



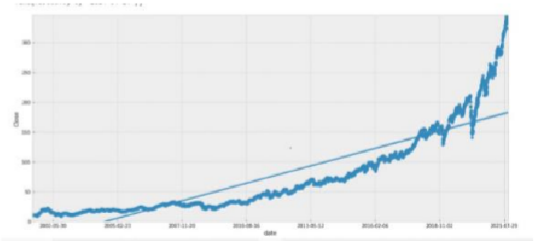
Root Mean Square Error (RMSE) is a standard way to measure the error of a model in predicting quantitative data.

Formally it is defined as follows:

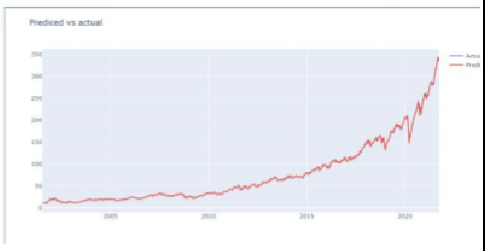
$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

If the RMSE value is between 0.2 and 0.5, we can conclude that this model can predict the prices accurately. The rmse value for the above given model is 1.9817. We conclude the deviation of the rmse value from the ideal range of value is due to the sensitivity of the model to outliers and the parameters chosen did not predict value in an accurate manner.

Due to the inaccuracy of the previous model, we choose to develop another model in which we are choosing 4 independent parameters namely opening price, high, low and volume of stocks traded.



The above figure depicts the best fit line for the data set.



From this figure, we observe that the predicted stock prices and the actual stock prices are relatively more accurate with an rmse value of 0.647. Due to the linearly increasing trend of closing prices, the model could perform well. Though the model could perform reasonably well, we wanted to switch to another model as linear regression model is sensitive to outliers and this model is appropriate only for linearly increasing or decreasing data sets.

2. LSTM

Artificial neural networks refer to the model of We stumbled upon a simple ANN method of processing where the input propogates throughout the neural network to give an output initially giving a garbage value, thus an error

signal is calculated and the random weights are updated, hence learning about the process every iteration. The method used to update is the grwadient descent method. This what we just discussed sums up the feedforward neural networks. The major disadvanatge of this prediction model is that it depends solely on input values and not past events or any memory based occurences which is essential in gthe prediction of stock prices that is we need to know the past trends thus ANN fails without any time dependencies. RNN soolves this very problem where the output depend on input x and also on the previous time stamp. LSTM is better than even RNN for the single reason that LSTM overcomes the vanishing gradient problem(gradient approaches 0). LSTM(long short term memory) can remember and recall information for a prolonged period of time. Therefore the initial input is remebred and passed through out the memory line in the neural networks. Hence this method outshines the previously mentioned methods.

Fig.d

```
def plotting(df, title):
  fig = px.line(title = title)
  for i in df.columns[1:]:
    fig.add_scatter(x = df['Date'], y = df[i], name = i)
  fig.show()
```

Fig.D

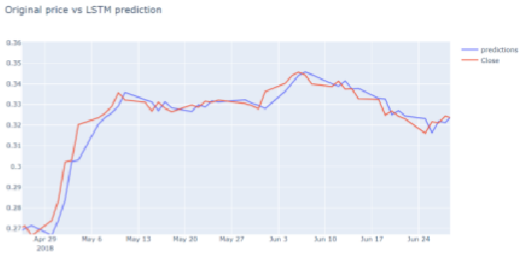


Fig.E

III. References

- Gupta and B. Dhingra, "Stock market prediction using Hidden Markov Models," 2012 Students Conference on Engineering and Systems, 2012, pp. 1-4, doi: 10.1109/SCES.2012.6199099.
- Predicting Stock Prices with Linear Regression in Python - alpharithms (alpharithms.com)
- https://towardsdatascience.com/predicting-stock-prices-with-python-ec1d0c9bece1
- T. Kimoto, K. Asakawa, M. Yoda and M. Takeoka, "Stock market prediction system with modular neural networks," 1990 IJCNN International Joint Conference on Neural Networks, 1990, pp. 1-6 vol.1, doi: 10.1109/IJCNN.1990.137535.
- https://www.datacamp.com/community/tutorials/lstm-python-stock-market
- https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00333-6

Stock Prediction

ORIGINALITY REPORT

6%	6%	2%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	towardsdatascience.com Internet Source	2%
2	ijeee.in Internet Source	1%
3	www.cours-gratuit.com Internet Source	1%
4	link.springer.com Internet Source	1%
5	ebin.pub Internet Source	1%

Exclude quotes	On	Exclude matches	< 5 words
Exclude bibliography	On		