Machine Learning In Finance

Vihaan Sharma [1] , Harpreet Singh Dhoot [2], Bhavna Arora [3]
[1][2]*Student, Department Of Computer Engineering,*
Atharva College Of Engineering, Malad
[3] *Assistant Professor, Department Of Computer Engineering,*
Atharva College Of Engineering, Malad.

Abstract

Data science includes algorithms and processes to extract useful knowledge. Machine learning is the scientific study of algorithms which allows the system the ability to automatically learn and improve from experience without explicitly programmed or without human intervention. This paper illustrates what is data science, machine learning. Other sections of this paper explains how can we apply these two different disciplines in the finance sector. This document also sheds light on possible applications of data science and machine learning models in finance and to automate the tasks.

Keywords: *Finance and Big Data; Machine Learning; Linear Regression, K-nearest Neighbor; Support Vector Method; Decision Tree; Neural Network;Long Term Short Term Memory; Artificial Intelligence; Asset modeling; Forecasting; Investment analysis; Risk management*

1. Introduction

In today's world, the amount of data is increasing exponentially in rate. Main reason for this is due to availability of various sources and media. For example, if consider an media source as Twitter, It processes over 70 million tweets daily I.e. 8TB of data daily. This is a huge collection of data of only one source, there are enormous other sources too. So, this amount of variety and volume of data is coined as Big Data. Big Data possess tremendous potential in a variety of fields like Biology, Advertising Industry, Health care, Finance, etc. Our Paper focuses on the Finance Sector using modern means like Data Science and its branches Machine Learning and Big Data. So, First let's see what Is finance and then we will discuss what is Data Science and the relationship of Data Science and Finance, what is Machine Learning and how it can be collaborated with Finance and its Applications.

2. Theory

2.1. What is Finance

In simple terms, Finance is the study of investment of money and how it is used. It gives an analysis of where, why and how to use money for investment or profit. The Finance Sector works or depends on factors like Product or customer orientation, Service, Efficiency and Risk Control. Product or customer orientation is an analysis done to find, gather and interpret information on customer products. Another important factor is service, as financial products are easily copied to overcome its opposition or competition's service sector places a key role. Efficiency is another factor which indicates how efficiently work-flow is managed. Higher efficiency is the growth of that entity. It can be increased by means of automation of certain sectors. Risk control eyes on the risk which can be encountered like credit risk, market risk, breakdown of info. Or communication. In this all factors technology is playing an important part by various analysis which is done by the means of Data Science as we will discuss in further parts.

2.2. What is Data Science and Relationship of Data Science and Finance

In Simple terms Data Science means organizing, assembling and passing of data. Data Science generally works on three step processes: Data Wrangling/Transformation, Data investigation and Transfer Data. As the name suggests Data Wrangling simply means transforming of data in one form for analysis, Data investigation means examining and evaluating of required data. Transfer of data can be referred to as converting of Real Data to Virtual Data. Big Data is an implementation of the concept of Data Science. Big Data is a high variety and volume of data which enables discovery of data and improved decision making using data. Also It is used for process optimization. But traditionally it was not easy to use Big Data due to factors like Volume i.e. Amount(Large) of Data, Variety I.e. Various sources sharing data, Velocity I.e. speed (High) at which data is been collected and Veracity I.e. Dirty and Noisy Data. But due to modern means of Data Science extracting of data is possible today. Data Science helps in specifically organizing data with the help of various technologies and tools. One of the technologies of data Science that we are focusing on is ML I.e. Machine Learning. ML is considered by researchers as the driver of big data.

2.3. What is ML And Relationship Of ML with Finance

ML I.e. Machine Learning as the name suggests it works on the principle of self-learning from experience I.e. data like Humans learns from their experience. Reason to use ML is pretty simple as it learns from data and provides insight, and provides decisions and predictions. It mainly does statistical analysis, more the data and experience more accuracy is achieved. Thus, using big data with ML makes work efficient. ML mainly has two parts Supervised and Unsupervised. When there is a fixed input and desired output we know then it is labeled as Supervised and when output is unknown then it is labeled as Unsupervised. As data is required for ML to gain experience so Big Data provides enough data and then with the help of the algorithms desired output is obtained. So, with the help of big data and ML one can estimate various decisions and predictions of various parts of different sectors like Finance, too. According to many researchers, ML is the next big tool for finance researchers' toolkit, due to its strong emphasis on self-learning and practical understanding from experiences in the same sense as the humans do in their life. So, further we can discuss algorithms which can be implemented in the field of finance using machine learning I.e. ML. Our paper focuses on some algorithms which can be used like Decision tree[1], K- Nearest Neighbor[2], Linear regression[4] and models which can be considered as basic for the finance sector.

3. Algorithms And Models

There are various algorithms which can be used, our paper focuses on three basic ones I.e. Linear regression, decision tree and k-nearest Neighbor. We will also discuss the Neural Network and Support Vector Method I.e. SVM.

3.1. Decision Tree

Decision Tree [1] is a form of tree which is similar to a regression model to predict target variables from previous data for predictions or decisions. Regression can be defined as an entity which finds relationships between independent variables using dependent variables. It uses a top-down approach without backtracking. It uses greedy search methodology. In this root node is the best starting point and hence at every step branching takes place. Leaf nodes act as target variables for new data. Root nodes are considered with the help of Information Gain or Gini Index. Information Gain is used for categorical organization with use of entropy I.e. randomness/uncertainty of variables and it lies in 0-1.Entropy is calculated in steps as entropy of target and entropy of every branch. Higher the entropy of node and the higher entropy of branch is considered and processed it can be given as **∑Pilog(Pi)**. Gini Index is used for continuous attributes. In this every node is validated to see that the root node is improved or not after this step, If yes then expand else don’t. It can be given as **1-∑(Pi).**It is easy to understand and implement. It helps in classification and predicting or decision making process. Validation is also done. Thus, it is useful in Financial Analysis.

3.2. Regression

Regression [4] as discussed earlier, Regression is finding relation between independent(X) and dependent(Y) variables to predict further value of dependent variable. Regression has mainly five types as Simple regression I.e. One Independent Value, Multiple regression I.e. Two or more independent values, Dependent regression I.e. Continuous/Independent values, Linear regression I.e. Output is straight line and Non-linear regression. I.e. Output is Curve Line. We are discussing

Linear Regression in our Paper. In Linear Regression, Data is placed in a scatter plot. Correlation is to be found. Correlation simply represents strength of relationship between variables, it ranges from -1 to +1(Strong Co-relation-0.5-0.99). Regression lines pass through means of dependent and independent variables. More distance between them means more chances of errors. Best results are taken with the help of least squares method.

This is an example of linear Regression as discussed, Formula can be given as

$$Y= \theta_1 X_1 + \theta_2 X_2 + \dots \theta_n X_n$$

Here, x1, x2,….xn represent the independent variables while the coefficients θ1, θ2, …. θn represents the weights.

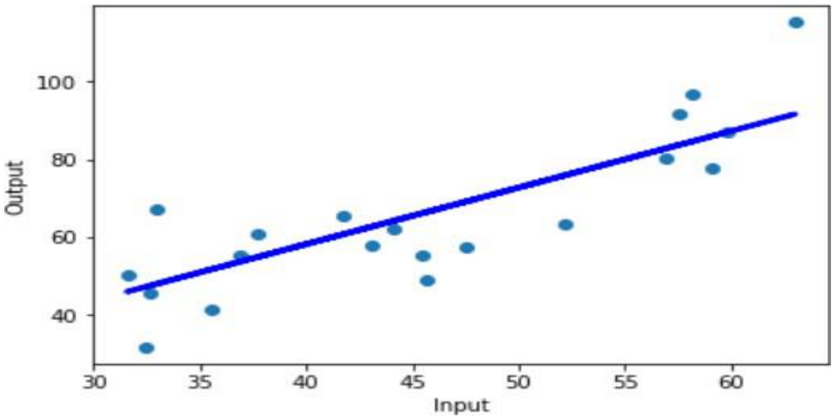


Figure: Showing an example of Linear Regression

Steps can be given as: 1. There is a requirement of linear assumption I.e. linear data is formed.2. The removal of noise and col-linearity is required. 3.At last Prediction and Normalization is achieved. Thus, It is used for Predictive Analytic And Finance.

3.3. K-Nearest Neighbor (KNN)

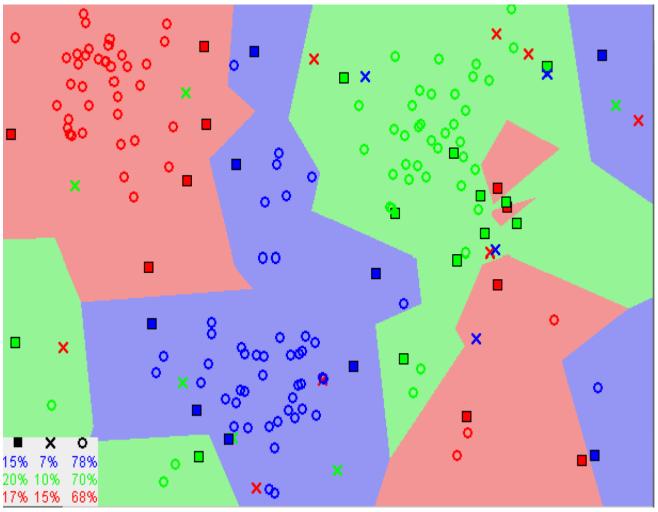


Figure: Explaining Classification with help of KNN

K-Nearest Neighbor (KNN) [2] is an algorithm used for classification of problems. To predict target labels using new test data. In this training data and test data is plotted and it calculates the number of nearest training data points distance with KNN using distance functions like Hamming distance, Euclidean distance, etc. Hamming Distance is used for categorized values and Euclidean distance is used for continuous variables. Data points (xi) and Class label(ci) are plotted. Find the distance between test data with all training data points using any of the distance functions. Arrange in descending order. Now using k variable value to count the number of training points where **$k=n^{1/2}$**.

This above diagram explains classification using KNN as discussed earlier. Thus, Classification is achieved which plays an important role in the financial sector.

3.4. Support Vector Modelling (SVM)

Support Vector Modelling (SVM) [3] is a supervised algorithm used to classify data points. To separate two classes of data points there are many hyperplanes that could be chosen but our main objective is to maximize margin. Margin is to be maximized as it helps to classify distinctly for future points. Data is classified as different classes on either side of the hyperplane. As shown below is an example of hyperplane-

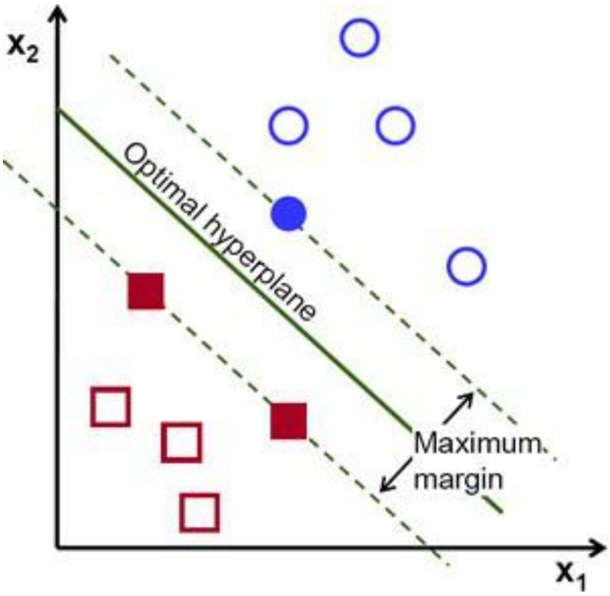


Figure: Explaining Hyperplane in SVM

This diagram is an example of a hyperplane for maximum margin. For n-number of input features, there are n-1 dimensional hyperplanes. Support vectors are data points that are closer to hyperplane and are used to maximize margin. We take output of linear features and squash to range of [0,1], if value is greater than 0.5 then 1 else label 0. In sum, linear function is seen if greater than 1 then class one and if -1 another class Thus reinforced as[1,-1].Loss function is given as-

$$\mathbf{C(x, y, f(x))=(1-y*f(x))}$$

Regularization is to balance maximum margin and loss. Loss function and cost function with regularization is given as-

$$\mathbf{\min_w \lambda \| w \| 2 + \sum(1 - y_i < x_i - w_i >)}$$

Partial derivative is taken with respect to weight to find gradient. If no misclassification is given as,

$w=w-\alpha\ (yx_i-2\lambda w)$,

if there is misclassification then

$w=w+\alpha(yx_i-2\lambda w)$.

Thus, it plays an important role in classification of data which is a very important role in the financial sector.

3.5. Neural Networking

Neural network is just like our normal neural system, Neuron is its basic unit which takes inputs and does some mathematics and gives outputs. It is limited by the Activation function which is used to turn unbounded inputs into outputs that have predictable forms. Neurons are set up in such a way that a network is formed which has three layers which can have n-number of layers in

them. Three layers are Input Layer I.e. Layer which takes inputs, Hidden Layer I.e. Layer which works as an intermediate transmitter and Output Layer I.e. Giving output is done by this layer. It is shown as above in the diagram

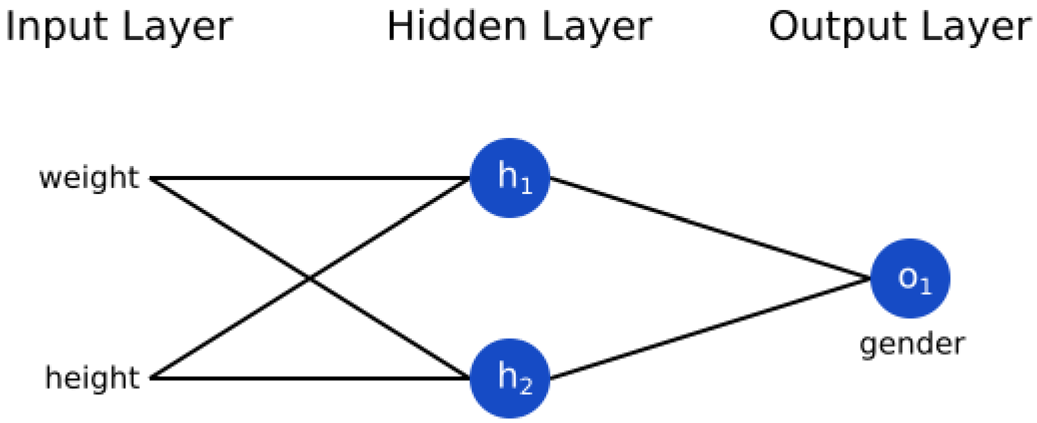


Figure: Explaining Layers in Neural Networking

Loss in transferring the neurons in network in form of Mean Square-

$Error=1/n\sum(y_{true} – y_{predicited})^2$

where y is variable y with true is true value and other id predicted value thus, loss is found by MSE. By changing weights and biases one can change loss rate and increase efficiency. And Activation function can be given as-

$f'(x)=f(x)*(1-f(x))$

Thus, By using neural networks various fields of finance are boosted as complex models, pattern and prediction can be braked down and easily analyzed and predictions accuracy is increased. Thus, Boosting Financial Sectors.

3.6. Short Term Memory And Long Short Term Memory

Short Term memory is an artificial Recurrent Neural Network i.e. RNN, as the name suggests Short Term memory remembers only recent activities and forgets the earlier activities. For E.g. If we are processing a paragraph of text for a decision,Short Term memory will only recall the later half of the paragraph and lose earlier entries. Thus, if we are processing a sequence which is long, then ST will have a hard time carrying all the information from earlier steps to later steps. RNN works on the principle that it processes first input and transfers them into the machine language or readable vector.Then this sequence repeats in sequence one after the other. In RNN it passes previous information/state to the next and thus a neural network is formed. Now, the previous state and current state is combined to form a vector form which has both previous and current step information. Vector goes through tanh activation which is used to regulate values flowing through the network, as tan can have values ranging from -1 to 1. But as discussed Short Memory has limitations. To overcome this Long Short Term Memory was developed.It can be diagrammed as shown-

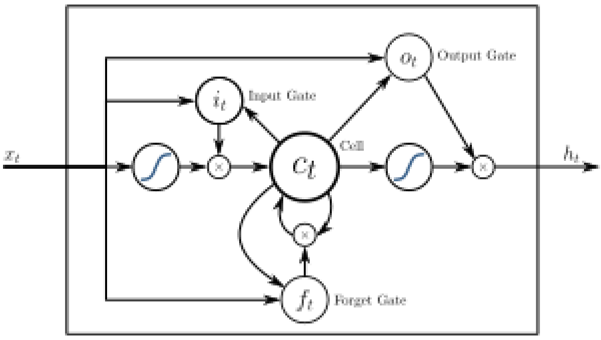


Figure: Long Short Term Memory

Long Short Term Memory overcomes Short Term Memory Issues with the means of internal mechanism i.e. more gates. Gates helps to decide which data in a sequence is important to keep or discard it. It is helpful as it passes information down the long chain of sequence to make predictions. It is similar to short term memory, it has cell state i.e. path, it carries the relevant information throughout processing of sequence. Cell states go on the path and more information is added or removed through gates. Gates in LSTM learns from the data unlike Short Term, and instead of tanh activation sigmoid activation is used as it keeps values validated in the range of 0-1. Gates can be classified in various types- Forget Gates which decides which data is to be kept or forgotten. It makes a decision on sigmoid activation if its values lie near zero then it is discarded else it is kept. Input Gates helps in adding or updating information and it also decides on sigmoid activation function same as in forget gates. Output Gates decides what the next state should be. Next state takes previous and current states and then this all gates, Sigmoid function and call states repeats their functions. Thus, At last this algorithm gives proper output as a prediction which is accurate than most other algorithms in ML.

4. Applications

4.1. Investment analysis

As the name suggests, investment analysis means process and different methods of evaluating investments for income, risk and resale value. It can include charting past returns to predict future performance and here the past data can be exploited by machine learning models for future predictions. We identify four topics in this category, all of which offer an interesting insight into potential future research directions. Topics covered under investment analysis are below.

The first topic is market sentiments. A popular machine learning technique in this topic is SVM. Smailović et al [5] showed that twitter feeds which are based on public sentiment can be used to predict stock price shifts.

Technical analysis is primarily based on quantitative analysis in contrast to the text-based market sentiment topic. The techniques are neural networks and increasingly deep learning variations of neural networks. Ng et al. [6] approach the classic candlestick patterns of technical analysis with a combination of a Radial Basis Function (RBF) neural network and an improved error minimization technique.

The topic of investor behavior fits somewhat to the extant literature on theoretical modeling of agent behavior in finance. Moerland et al. [7] (2018) provides a review of prior literature on the role of emotion in agent / machine interaction, as well as delving into how to integrate emotional understanding in reinforcement learning models.

The final topic is investment decision support. This topic concentrates on application of A.I. (in part of M.L.) to generate information that supports investment decision making. Kampouridis and Otero [8] is the most recent iteration of EDDIE. EDDIE is a software which works on genetic programming specifically genetic decision tree approach to arrive at investment decisions like yes or no or buy or don't buy decisions.

4.2. Asset modeling and forecasting topics

Asset modeling as the name suggests, is the process used to manage what a business organization wants to achieve or complete through the assessment of portfolio assets under specific time. The topics we are covering here are (1) portfolio optimization (2) Foreign exchange forecasting

Portfolio optimization, Shen and Wang (2017) [9] is one of the top matched articles to this topic. They optimized a classic issue by Markowitz-based portfolio selection. According to them the mean-variance portfolio in practice requires long time periods of data for estimation. But because of this long time data, this introduces the risk that the data is obsolete or is not relevant for current portfolios. Shen and Wang (2017) [9] address this issue using ensemble learning, an ML approach where multiple algorithms / sub-samples are tested to improve learning compared to a single ML application.

In Foreign exchange forecasting or sometimes referred as forex forecasting, machine learning in practice can be used because clean and structured data is already available. AmirAskari and Menhai [10], the paper is based on bending of fuzzy logic and neural network, thus forming a new model called Modified Fuzzy Relational Model (MRFM) which as they argue is better modeling for forex relationships and other dynamic structures.

4.3. Risk management topics

This application relates to the area of risk management. Operational risk management, risk forecasting, and risk assessment are the major topics of risk management and where machine learning can be used. Risk modelling is one of the original focus on ML in finance, due to the strong methods in ML for classification and clustering. Classification of acceptable and unacceptable risks is an important task in risk management. Thus, risk management requires highly accurate and a consistent machine learning model.

Marmier et al. [11] introduces an integrated process through a decision tree model in new product development based on project risk. Risk activities and other product development activities are the major part which contribute to any risk in a project, thus this system takes several risk activities and product development activities, and does an assessment of risk which leads to the range of possible scenarios for a project.

5. Implementation

Let's try to implement different machine learning models that were discussed in the paper into real world application - Stock Market Prediction.

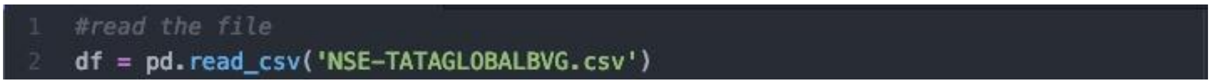
Problem Statement

There are many machine learning models which can be used for stock prediction. Different models have different degrees of accuracy. Hence, study of these models and implementation of the same in a real world application is an important factor deciding which model out of many outperforms the other models.

Implementation of Problem Statement:

The main focus of this project is to use different machine learning models to predict the stocks on a given stocks dataset and to evaluate models by comparing prediction accuracy. The accuracy is calculated using RMSE (Root Mean Square Error) of all the models. We examined a few models including Linear regression, KNN, LSTM.

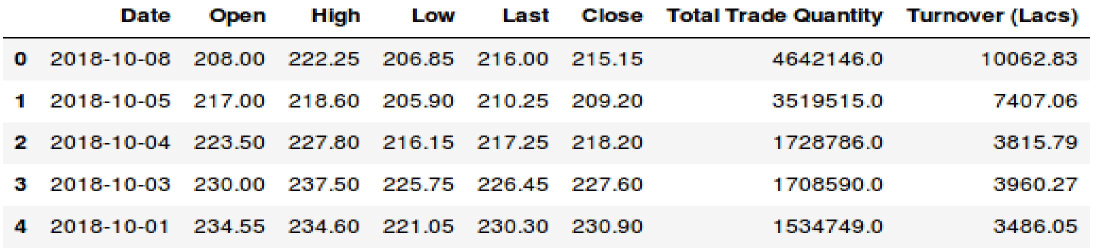
We first need to have a Dataset and target variable. We'll be using a dataset from Quandl and for this particular problem statement, we're using the stock data of 'Tata Global Beverages'. The dataset we've imported is done by:



```
#read the file
df = pd.read_csv('NSE-TATAGLOBALBVG.csv')
```

Figure: Reading CSV file of Dataset

Dataset extracted is as shown by dataset by Quandl is as shown:



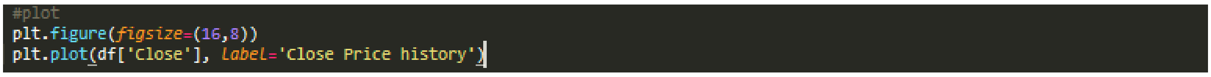
	Date	Open	High	Low	Last	Close	Total Trade Quantity	Turnover (Lacs)
0	2018-10-08	208.00	222.25	206.85	216.00	215.15	4642146.0	10062.83
1	2018-10-05	217.00	218.60	205.90	210.25	209.20	3519515.0	7407.06
2	2018-10-04	223.50	227.80	216.15	217.25	218.20	1728786.0	3815.79
3	2018-10-03	230.00	237.50	225.75	226.45	227.60	1708590.0	3960.27
4	2018-10-01	234.55	234.60	221.05	230.30	230.90	1534749.0	3486.05

Figure: Showing Extracted dataset

Multiple variables are seen from the dataset like Date, Open, High, Low, Last, Close, Total Trade Quantity and Turnover

- The columns *Open* and *Close* represent the starting and final price at which the stock is traded on a particular day.
- *High*, *Low* and *Last* represent the maximum, minimum, and last price of the share for the day.
- *Total Trade Quantity* is the number of shares bought or sold in the day and *Turnover (Lacs)* is the turnover of the particular company on a given date.

Another important thing to note is that the market is closed on weekends and public holidays. Notice the above table again, some date values are missing, of these dates, 2nd is a national holiday while 6th and 7th fall on a weekend. The profit or loss calculation is usually determined by the closing price of a stock for the day, hence we will consider the **closing price as the target variable**. Let's plot the target variable to understand how it's shaping up using our dataset and code as provided below:



```
#plot
plt.figure(figsize=(16,8))
plt.plot(df['Close'], label='Close Price history')
```

ISSN: 2233-7857 IJFGCN

Figure: Plotting of Code and Dataset

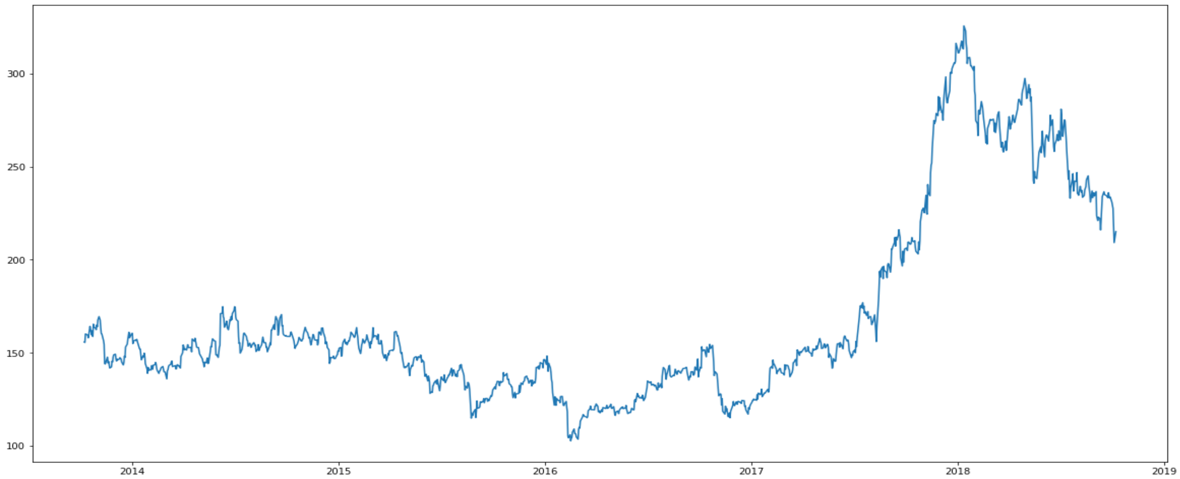
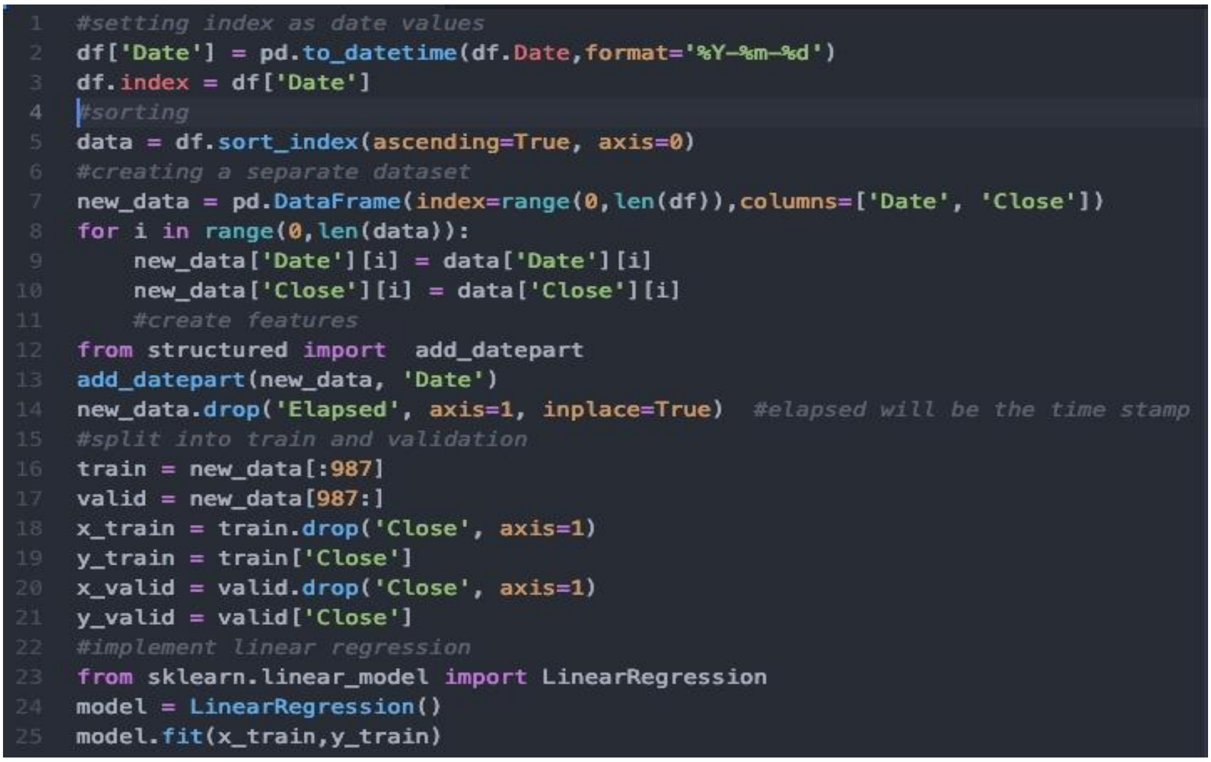


Figure: Plotting of target variable

This is the actual graph of the dataset we've used. In the subsequent subsections we're going to predict the stock using different machine learning models, plot the predicted value on the graph and compare it with the actual value. Note, the orange curve on the graph is the predicted values.

5.1. Linear Regression:

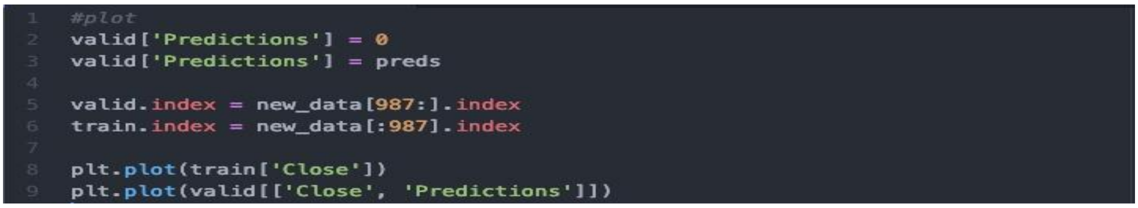
As discussed earlier, Linear Regression is the most basic algorithm of ML and decision trees are based on similar understanding. For our problem statement, we do not have a set of independent variables. We have only the dates instead. Let us use the date column to extract features like – day,



```
#setting index as date values
df['Date'] = pd.to_datetime(df.Date,format='%Y-%m-%d')
df.index = df['Date']
#sorting
data = df.sort_index(ascending=True, axis=0)
#creating a separate dataset
new_data = pd.DataFrame(index=range(0,len(df)),columns=['Date', 'Close'])
for i in range(0,len(data)):
    new_data['Date'][i] = data['Date'][i]
    new_data['Close'][i] = data['Close'][i]
    #create features
from structured import  add_datepart
add_datepart(new_data, 'Date')
new_data.drop('Elapsed', axis=1, inplace=True)  #elapsed will be the time stamp
#split into train and validation
train = new_data[:987]
valid = new_data[987:]
x_train = train.drop('Close', axis=1)
y_train = train['Close']
x_valid = valid.drop('Close', axis=1)
y_valid = valid['Close']
#implement linear regression
from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(x_train,y_train)
```

Figure: Linear Regression Code

month, year, mon/fri etc. and then fit a linear regression model..Now importing the dataset and using this algorithm(above),plotting is done as shown and output is obtained.



```
#plot
valid['Predictions'] = 0
valid['Predictions'] = preds

valid.index = new_data[987:].index
train.index = new_data[:987].index

plt.plot(train['Close'])
plt.plot(valid[['Close', 'Predictions']])
```

Figure: Plotting of Linear Regression Graph

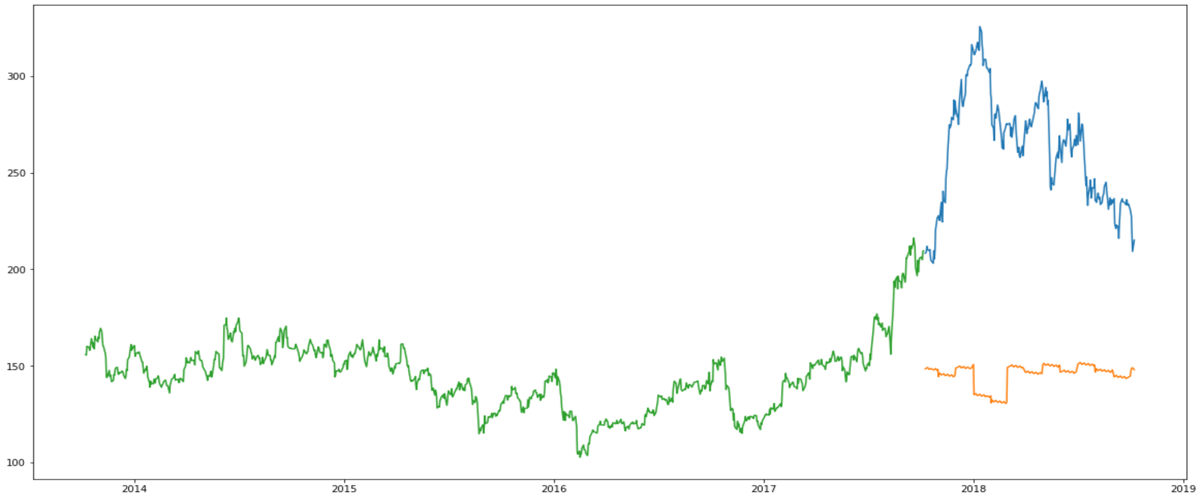


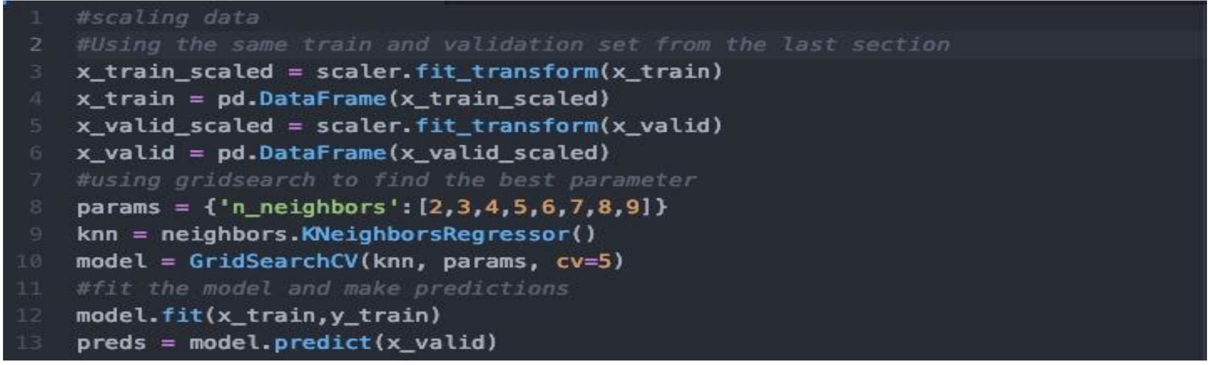
Figure: Prediction using Linear Regression

Here, the green curve is the dataset or historic data. Blue curve is to be predicted and the orange curve is the predicted value by respective models. Thus, it is clear that Linear regression is a simple technique and quite easy to interpret, but there are a few obvious disadvantages. One problem in using regression algorithms is that the model overfits to the date and month column. Instead of taking into account the previous values from the point of prediction, the model will consider the value from the same *date* a month ago, or the same *date/month* a year ago. As seen from the plot above, for January 2016 and January 2017, there was a drop in the stock price. The model has predicted the same for January 2018. A linear regression technique can perform well for problems such as Big Mart sales where the independent features are useful for determining the target value.

RMSE value for Linear Regression is : 121.16382449873643

5.2. K-Nearest Neighbours:

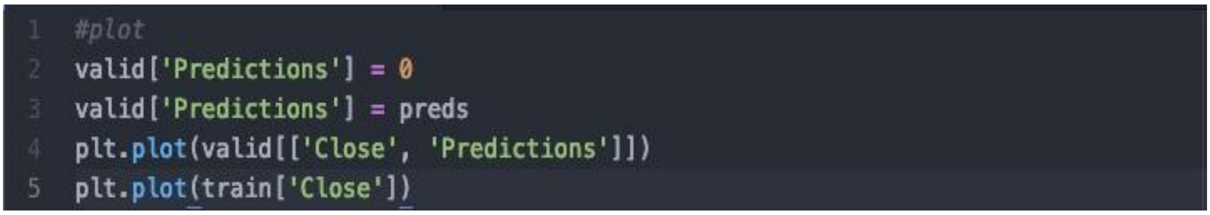
As discussed earlier,it is based on the independent variables,KNN finds the similarity between new data points and old data points.Using the same train and validation set from the last section and using code as given below:



```
#scaling data
#Using the same train and validation set from the last section
x_train_scaled = scaler.fit_transform(x_train)
x_train = pd.DataFrame(x_train_scaled)
x_valid_scaled = scaler.fit_transform(x_valid)
x_valid = pd.DataFrame(x_valid_scaled)
#using gridsearch to find the best parameter
params = {'n_neighbors':[2,3,4,5,6,7,8,9]}
knn = neighbors.KNeighborsRegressor()
model = GridSearchCV(knn, params, cv=5)
#fit the model and make predictions
model.fit(x_train,y_train)
preds = model.predict(x_valid)
```

Figure: K-Nearest Neighbours Code

Now importing the dataset and using this algorithm(above),plotting is done as shown and output is obtained as in form of graph where blue part is actual and orange part is predicted by the given algorithm.



```
#plot
valid['Predictions'] = 0
valid['Predictions'] = preds
plt.plot(valid[['Close', 'Predictions']])
plt.plot(train['Close'])
```

Figure: Plotting of K-NN Graph

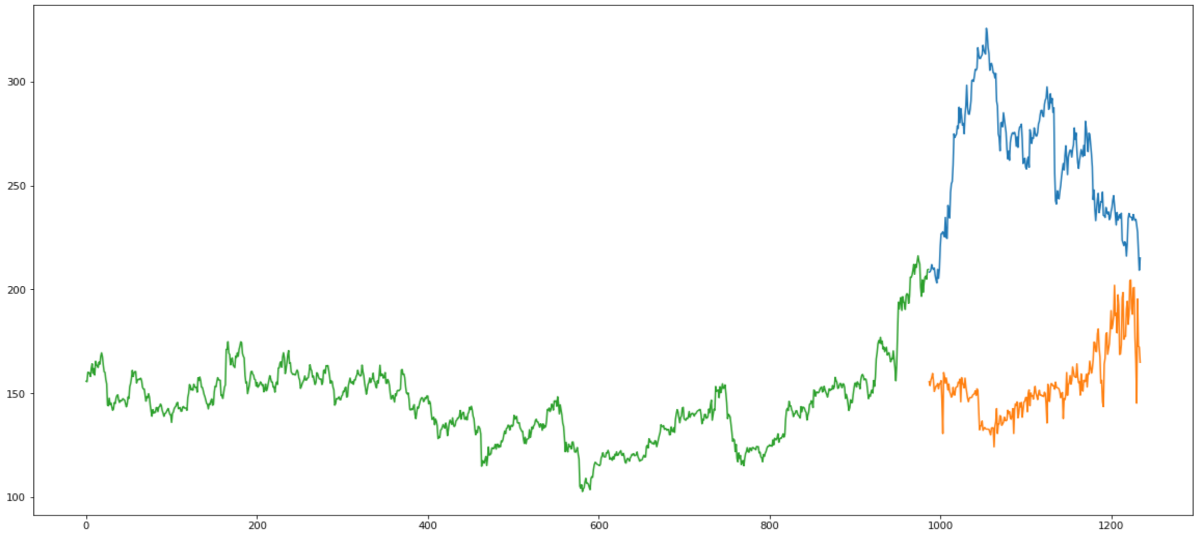


Figure: Prediction using KNN

It is clearly seen that the prediction done using K-NN is somewhat similar to the graph.Prediction is also not what was expected as the same as Linear regression.

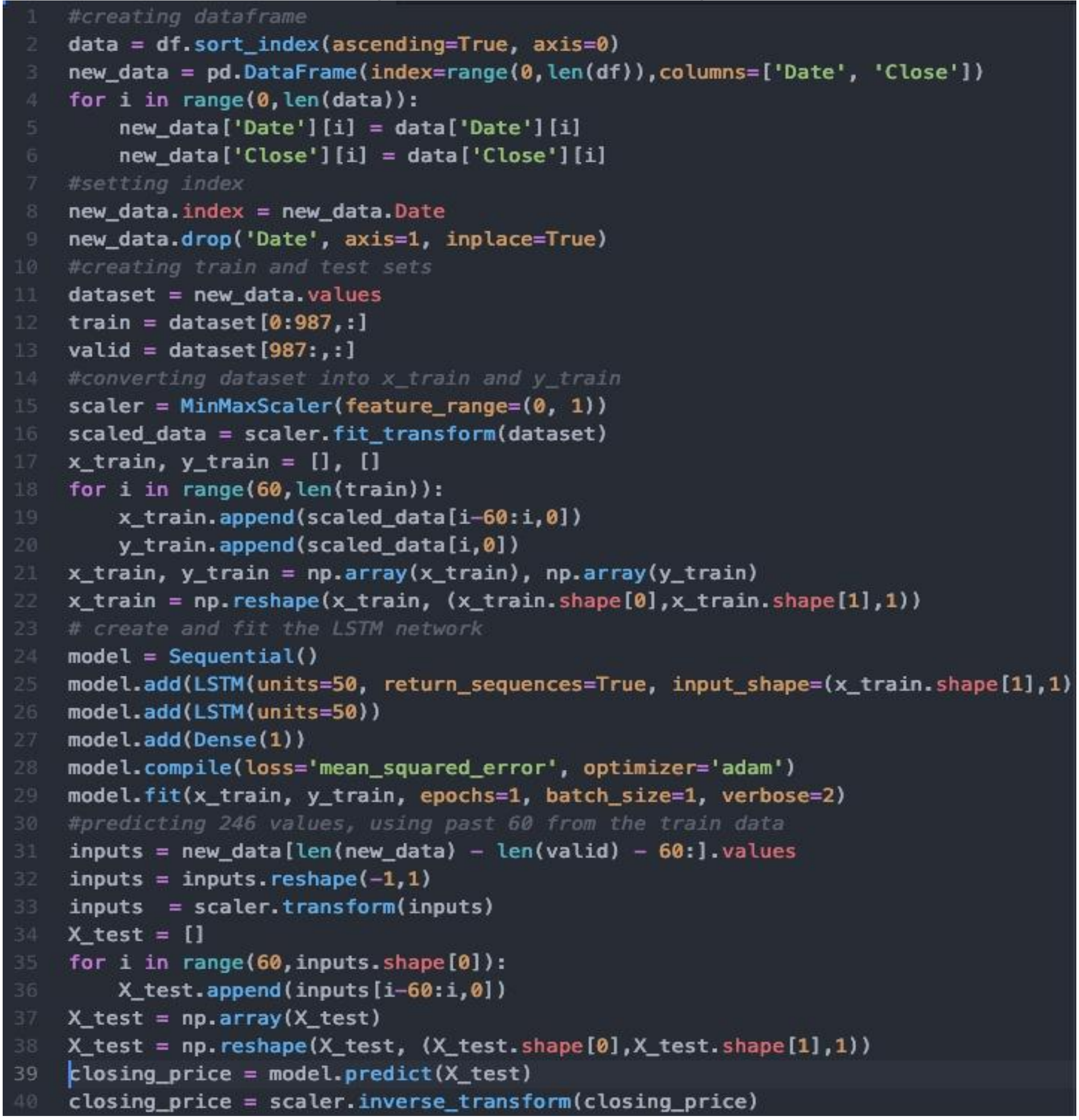
RMSE value of K-Nearest Neighbours is : 114.820692914522

As we can see, the RMSE value of K-Nearest Neighbours is approximately the same as the RMSE value of Linear Regression. This shows that the accuracy of K-Nearest Neighbours and Linear

Regression are the same and that both of the models are not very accurate in predicting the stocks for this data model.

5.3. Long Term Short Term Memory:

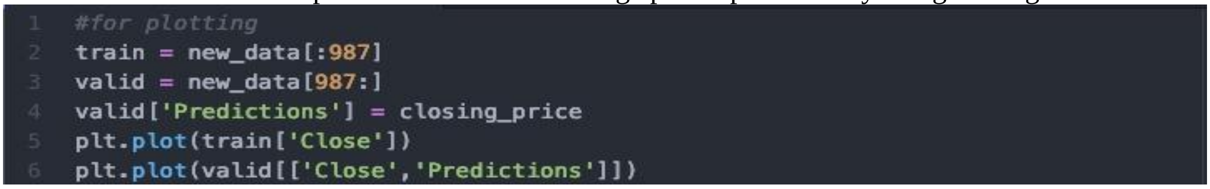
As discussed earlier, it's an application of neural networks. LSTMs are accurate because they store past important information and forget information which is not required or can say which is not important. Here also we are working on the same dataset. Code is given below:



```python
#creating dataframe
data = df.sort_index(ascending=True, axis=0)
new_data = pd.DataFrame(index=range(0,len(df)),columns=['Date', 'Close'])
for i in range(0,len(data)):
    new_data['Date'][i] = data['Date'][i]
    new_data['Close'][i] = data['Close'][i]
#setting index
new_data.index = new_data.Date
new_data.drop('Date', axis=1, inplace=True)
#creating train and test sets
dataset = new_data.values
train = dataset[0:987,:]
valid = dataset[987:,:]
#converting dataset into x_train and y_train
scaler = MinMaxScaler(feature_range=(0, 1))
scaled_data = scaler.fit_transform(dataset)
x_train, y_train = [], []
for i in range(60,len(train)):
    x_train.append(scaled_data[i-60:i,0])
    y_train.append(scaled_data[i,0])
x_train, y_train = np.array(x_train), np.array(y_train)
x_train = np.reshape(x_train, (x_train.shape[0],x_train.shape[1],1))
# create and fit the LSTM network
model = Sequential()
model.add(LSTM(units=50, return_sequences=True, input_shape=(x_train.shape[1],1)
model.add(LSTM(units=50))
model.add(Dense(1))
model.compile(loss='mean_squared_error', optimizer='adam')
model.fit(x_train, y_train, epochs=1, batch_size=1, verbose=2)
#predicting 246 values, using past 60 from the train data
inputs = new_data[len(new_data) - len(valid) - 60:].values
inputs = inputs.reshape(-1,1)
inputs  = scaler.transform(inputs)
X_test = []
for i in range(60,inputs.shape[0]):
    X_test.append(inputs[i-60:i,0])
X_test = np.array(X_test)
X_test = np.reshape(X_test, (X_test.shape[0],X_test.shape[1],1))
closing_price = model.predict(X_test)
closing_price = scaler.inverse_transform(closing_price)
```

Figure: LSTM Code

Now importing the dataset and using this algorithm(above),plotting is done as shown and output is obtained where the bluish part is actual and the orange part is predicted by using this algorithm.



```python
#for plotting
train = new_data[:987]
valid = new_data[987:]
valid['Predictions'] = closing_price
plt.plot(train['Close'])
plt.plot(valid[['Close','Predictions']])
```

Figure: Plotting of LSTM Graph

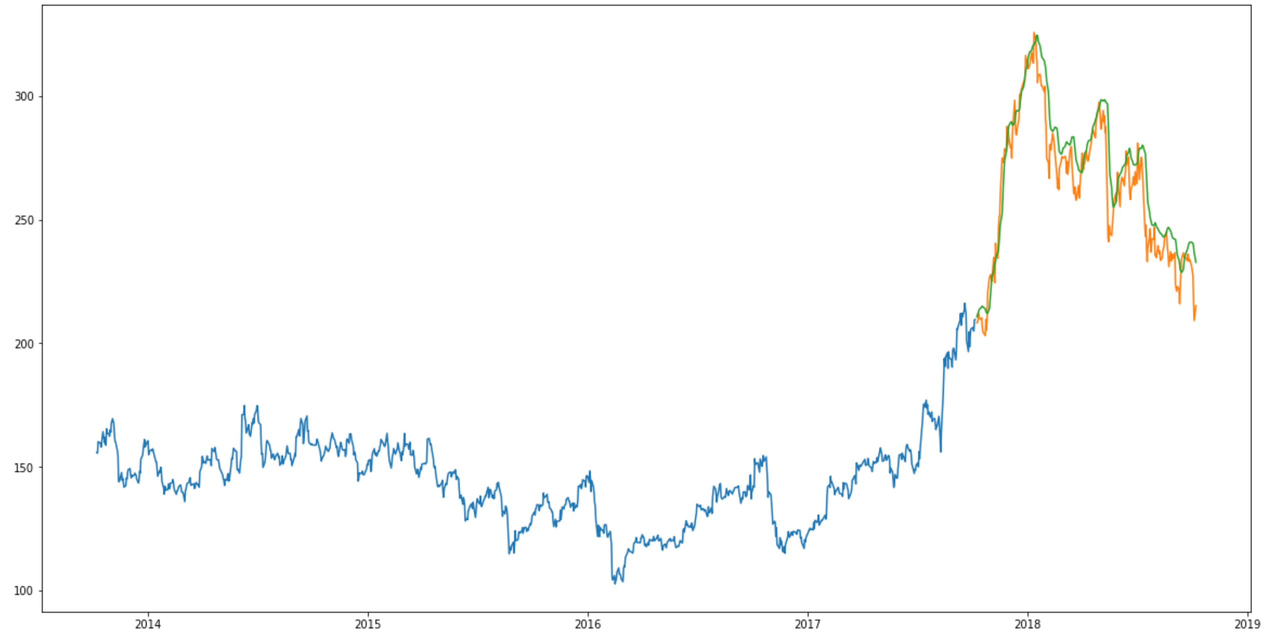


Figure: Prediction Using LSTM

The LSTM model can be tuned for various parameters such as changing the number of LSTM layers, adding dropout value or increasing the number of epochs.Thus, best prediction is brought by our LSTM due to its versatility of learning unlike other ML algorithms.

RMSE value of LSTM is : RMS Value is : 5.9098800103978375
The RMSE value of LSTM is much lower than the other two models. This shows that LSTM is more accurate than linear regression and kNN, and should be preferred for stock prediction.
Thus, we have implemented different machine learning models on our problem statement and compared these models with each other. We've found that LSTM performed very efficiently while Linear Regression performed very poorly. Henceforth, for stock market prediction using machine learning models, LSTM should be preferred.

6. Conclusion

As decision making and prediction plays a pivotal role in the finance sector, this work illustrates how machine learning can be implemented in the finance sector for excelling decision making and forecasting. This paper presented the machine learning models and how these models are applied in the finance sector. Finance sector includes big data and through machine learning models, knowledge (or can say analytical data) is being extracted, this knowledge in turns helps for decision making, analysis and forecasting.. Machine learning models can also be used in predictions. Concluding this paper, machine learning has a very vast scope in the sector of finance and an effective machine learning model implemented on a real world problem statement can give magnificent and accurate results, excelling in the finance sector of any industry or individual.

References

[1] Jafar Tanha, "Semi-supervised self-training for decision tree classifiers", International Journal of Machine Learning and Cybernetics, Volume 8, Issue 1, Page No's: 355-370, January, 2015.

[2] Khadim D, Fleur M and Gayo D, "Large scale biomedical texts classification: a k-NN and an ESA-based approaches", Journal of Biomedical Semantics, 7:40, June, 2016

[3] T. Razzaghi, Oleg R, "Multilevel Weighted Support Vector Machine for Classification on Healthcare Data with Missing Values", PLUS ONE, Page No's:1-18, May 2016

[4] Enrico R, Michel L, "The Counter, a Frequency Counter Based on the Linear Regression", IEEE Transactions on Ultrasonics, Ferroelectrics, Volume 63, Issue 7, Page No's: 961-969, July, 2016.

[5] Smailović, J., Grčar, M., Lavrač, N., and Žnidaršič, M. (2014). Stream-based active learning for sentiment analysis in the financial domain. Information Sciences, 285:181-203.

[6] Ng, W. W., Liang, X.-L., Chan, P. P., and Yeung, D. S. (2011). Stock investment decision support for Hong Kong market using RBFNN based candlestick models. In 2011 International Conference on Machine Learning and Cybernetics (ICMLC), volume 2, pages 538–543. IEEE.

[7] Moerland, T. M., Broekens, J., and Jonker, C. M. (2018). Emotion in reinforcement learning agents and robots: A survey. Machine Learning, 107(2):443–480.

[8] Kampouridis, M. and Otero, F. E. (2017). Heuristic procedures for improving the predictability of a genetic programming financial forecasting algorithm. Soft Computing, 21(2):295–310.

[9] Shen, W. and Wang, J. (2017). Portfolio selection via subset resampling. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, pages 1517–1523.

[10] AmirAskari, M. and Menhaj, M. B. (2016). A modified fuzzy relational model approach to prediction of foreign exchange rates. In 2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA), pages 457-461. IEEE.

[11] Marmier, F., Ioana, F. D., and Didier, G. (2014). Strategic decision-making in NPD projects according to risk: Application to satellites design projects. Computer in Industry, 65(8):1107 - 1114.