CS 559 Final project

Stock Price Prediction using ML

Gaurav Narvwani (CWID 10459113)

Sudarshana Sarma (CWID 10469063)

1.Problem Statement: Analysing the company's future profitability on the basis of its current business environment and financial performance using charts and using statistical figures to identify the trends in the stock market.

2.Machine Learning objective:

To predict the stock price of the company

This project solves this problem using different Machine Learning and Deep Learning Classifiers and make comparisons between these classifiers

3.Business objectives and constraints:

a) To analyse the stock market

b) Correlation between some of the features and the stock market

c) Errors can be very costly

4. Data description:

The dataset was acquired from an open source website called quandl. This is a repository for information regarding the financial sector such as stock prices, etc. We chose the stock dataset of an Indian company called Reliance Industries.

Our dataset consisted of multiple variables, such as – date, open, high, low, close, WAP, number of shares, number of trades, total turnover, deliverable quantity, percentage deliverable quantity to traded quantity, high to low spread, close to open spread.

Our target variable, i.e the attribute to predict is the feature 'close' which represents the final value of the stock. Since the other features can not be known ahead of time, our only feature is the date. We discarded the rest of the features.

With the date, we split the date to make multiple features which are much more valuable. We split the date into the following features – year, month, week, day, day of week.

The first four features are self explanatory. The final feature is important as the value of the stocks also depend on what day of the week it is. For example there is a different trend of the close price on Monday's and Friday's as they are the beginning and end of the week, compared to the rest of the week.

5.Libraries/packages used:

- Numpy
- Pandas
- Matplotlib
- ScikitLearn

- sklearn
- keras
- fastai

6.Train and test datasets:

The data-set is split into two parts train and test. The train set is from 2003 to 2018 and the test set is from 2018 to 2020. Machine learning model is trained upon train set and the performance of the model is tested on the test set.

7. Machine Learning models:

a) Linear Regression: We have implemented linear regression model on the data. The linear regression model returns an equation that determines the relationship between the independent variables and the dependent variable. The equation for linear regression:

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

b) KNN: We have used k-Nearest Neighbours as the second model. It is a non-parametric classification method. It is used for classification and regression.

Based on the independent variables, kNN finds the similarity between new data points and old data points.

c) Long Short Term Memory: We have implemented LSTM model. LSTMs are widely used for sequence prediction problems and have proven to be extremely effective. LSTM stores past information that is important, and forget the information that is not. LSTM has three gates:

The input gate: The input gate adds information to the cell state

The forget gate: It removes the information that is no longer required by the model

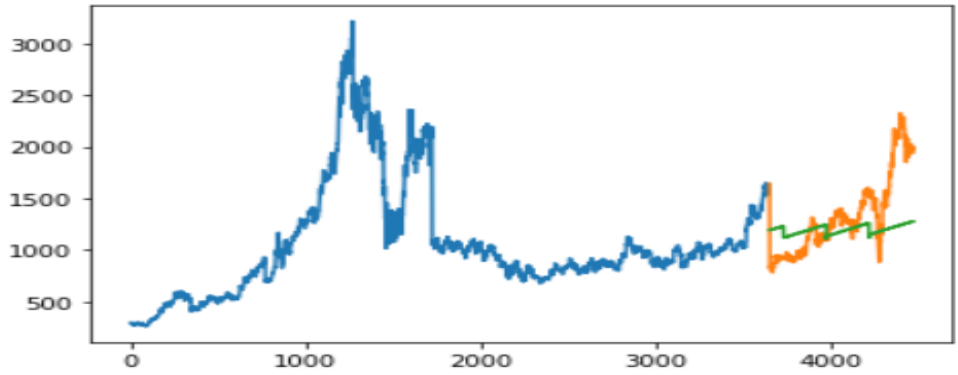
The output gate: Output Gate at LSTM selects the information to be shown as output

Results:

We developed 3 models and checked the root mean squared error to evaluate them. We plotted a graph where the green represents the predicted values and orange the actual ones.

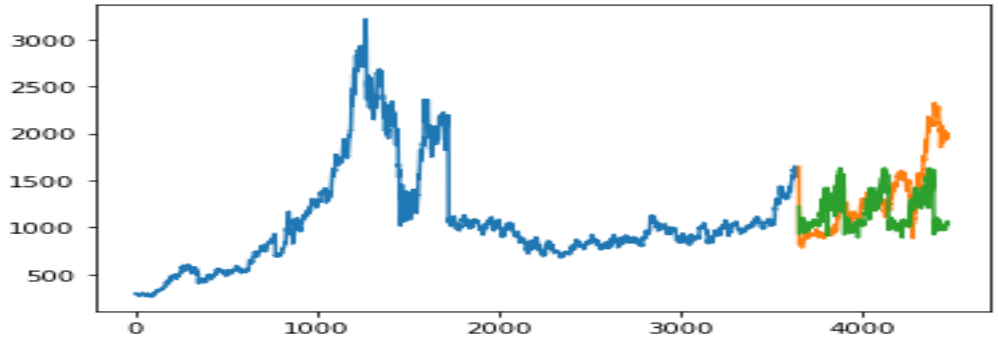
1) Linear Regression

This model required no parameter tuning. The model is a very simple model and hence performed very poorly on the dataset. We got a loss of over 300 which is very poor for any model.



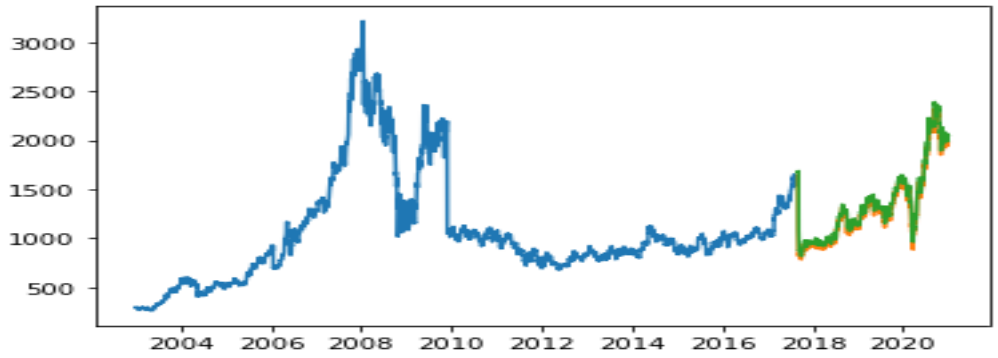
2) K Nearest Neighbors

After performing parameter tuning, we found the best value for the number of neighbours is 6. This model also performed poorly and we got a loss of over 400. It was surprising to see this model perform worse than linear regression which is a much simpler model.



3) Long Short Term Memory

We decided to use a deep learning model since the machine learning models we not performing great. We used LSTM as it is a popular model used for stock price predictions. We set the past to up to 60 training values for this model. To avoid over fitting we only used 2 layers each containing 50 units. After that we passed it into a dense layer to flatten it and used the adam optimizer. We ran it for 10 epochs and kept the batch size 1, thus using stochastic gradient descent. As expected, this model performed the best and gave us a loss less than 0.01.


3000
2500
2000
1500
1000
500
2004
2006
2008
2010
2012
2014
2016
2018
2020