CS419 GROUP PROJECT

TOPIC : Stock Price Prediction using Machine Learning

GitHub link : https://github.com/akj-new-era/Stock-Price-Prediction

Group:

- Modi Jay (200020078)
- Mehul Singodia (200020077)
- Parth Dange (200020091)
- Mudke Gourikant (200040083)
- Akshat Jain (200040016)

Abstract

To examine a number of different forecasting techniques to predict future stock returns based on past returns. We do this by applying supervised learning methods for stock price forecasting by interpreting market data. We are primarily looking to apply linear models and later on moving to neural networks.

Introduction

Stock (also known as equity) is a security that represents the ownership of a fraction of a corporation. This entitles the owner of the stock to a proportion of the corporation's assets and profits equal to how much stock they own. Units of stock are called "shares."A stock is a general term used to describe the ownership certificates of any company.

Stock prices change everyday by market forces. By this we mean that share prices change because of supply and demand. If more people want to buy a stock (demand) than sell it (supply), then the price moves up. Conversely, if more people wanted to sell a stock than buy it, there would be greater supply than demand, and the price would fall.

Data Set

To train and test our model we used data from HDFC and ITC from the year 2000 to 2021 which is taken from kaggle(https://www.kaggle.com/datasets/rohanrao/nifty50-stock-market-data). This dataset contains daily stock prices at which they open and close, daily high and low values, company's volume and turnover,previous day stock price,etc.

HDFC

	Date	Symbol	Series	Prev Close	Open	High	Low	Last	Close	\
0	2000-01-03	HDFC	EQ	271.75	293.5	293.50	293.5	293.5	293.50	
1	2000-01-04	HDFC	EQ	293.50	317.0	317.00	297.0	304.0	304.05	
2	2000-01-05	HDFC	EQ	304.05	290.0	303.90	285.0	295.0	292.80	
3	2000-01-06	HDFC	EQ	292.80	301.0	314.00	295.0	296.0	296.45	
4	2000-01-07	HDFC	EQ	296.45	290.0	296.35	281.0	287.1	286.55	

	VWAP	Volume	Turnover	Trades	Deliverable Volume	%Deliverble
0	293.50	22744	6.675364e+11	NaN	NaN	NaN
1	303.62	255251	7.749972e+12	NaN	NaN	NaN
2	294.53	269087	7.925368e+12	NaN	NaN	NaN
3	300.14	305916	9.181669e+12	NaN	NaN	NaN
4	288.80	197039	5.690480e+12	NaN	NaN	NaN

ITC

	Date	Symbol	Series	Prev Close	Open	High	Low	Last	\
0	2000-01-03	ITC	EQ	656.00	694.00	708.50	675.0	708.50	
1	2000-01-04	ITC	EQ	708.50	714.00	729.00	694.3	710.65	
2	2000-01-05	ITC	EQ	712.35	716.25	758.90	660.0	731.00	
3	2000-01-06	ITC	EQ	726.20	741.00	784.30	741.0	784.30	
4	2000-01-07	ITC	EQ	784.30	832.40	847.05	824.0	847.05	

	Close	VWAP	Volume	Turnover	Trades	Deliverable Volume	\
0	708.50	701.81	562715	3.949174e+13	NaN	NaN	
1	712.35	714.16	712637	5.089379e+13	NaN	NaN	
2	726.20	732.43	1382149	1.012325e+14	NaN	NaN	
3	784.30	776.63	721618	5.604266e+13	NaN	NaN	
4	847.05	841.25	231209	1.945046e+13	NaN	NaN	

INFOSYS

	Date	Symbol	Series	Prev Close	Open	High	Low	\
0	2000-01-03	INFOSYSTCH	EQ	14467.75	15625.00	15625.20	15625.00	
1	2000-01-04	INFOSYSTCH	EQ	15625.20	16800.00	16875.25	16253.00	
2	2000-01-05	INFOSYSTCH	EQ	16855.90	15701.00	16250.00	15507.45	
3	2000-01-06	INFOSYSTCH	EQ	15507.45	15256.65	15300.00	14266.85	
4	2000-01-07	INFOSYSTCH	EQ	14266.85	13125.50	13125.50	13125.50	

	Last	Close	VWAP	Volume	Turnover	Trades	\
0	15625.20	15625.20	15625.18	5137	8.026657e+12	NaN	
1	16875.25	16855.90	16646.38	56186	9.352937e+13	NaN	
2	15507.45	15507.45	15786.38	164605	2.598516e+14	NaN	
3	14266.85	14266.85	14462.82	81997	1.185908e+14	NaN	
4	13125.50	13125.50	13125.50	7589	9.960942e+12	NaN	

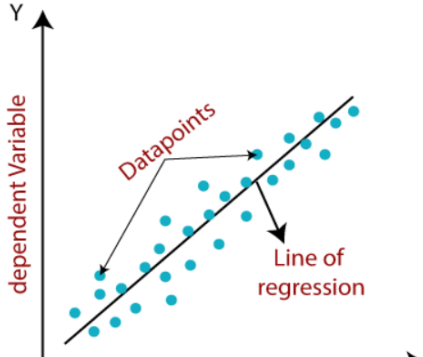
Linear Regression

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as sales, salary, age, product price, etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (xi) variables and shows how the value of the dependent variable is changing according to the value of the independent variables.

In this method we find the best fit line as follows.

1) Define a linear model



$y= a_0+a_1x+ \varepsilon$

Here,

Y= Dependent Variable (Target Variable)

X= Independent Variable (predictor Variable)

a0= intercept of the line (Gives an additional degree of freedom)

a1 = Linear regression coefficient (scale factor to each input value).

ε = random error

Writing it in matrix form, we get

Y = AX + E

2) Define a Cost function

Here we will be using MSE cost function

$$MSE= 1\frac{1}{N}\sum_{i=1}^{n}(y_i - (a_1x_i + a_0))^2$$

Where,

N=Total number of observation

Yi = Actual value

(a1xi+a0)= Predicted value

3) Use Matrix method for minimizing the error between predicted values and actual values

$A = (X^TX)^{-1}X^TY$

This equation will give the optimized value of the Coefficient Matrix (A)

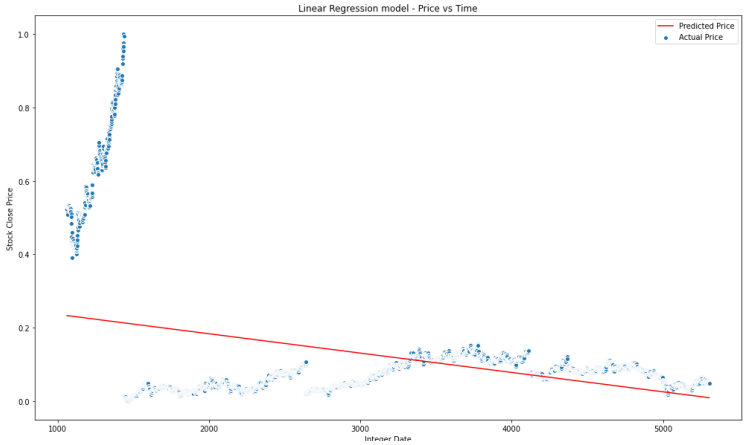
4) Checking the Performance using R Squared/ coefficient of determination

$$\text{R-squared}= \frac{\text{Explained variation}}{\text{Total Variation}}$$

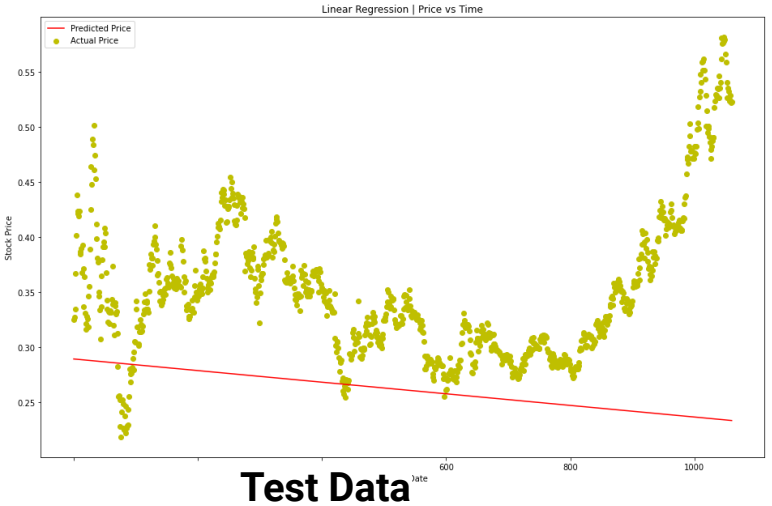
The high value of R-square determines the less difference between the predicted values and actual values and hence represents a good model

Results:

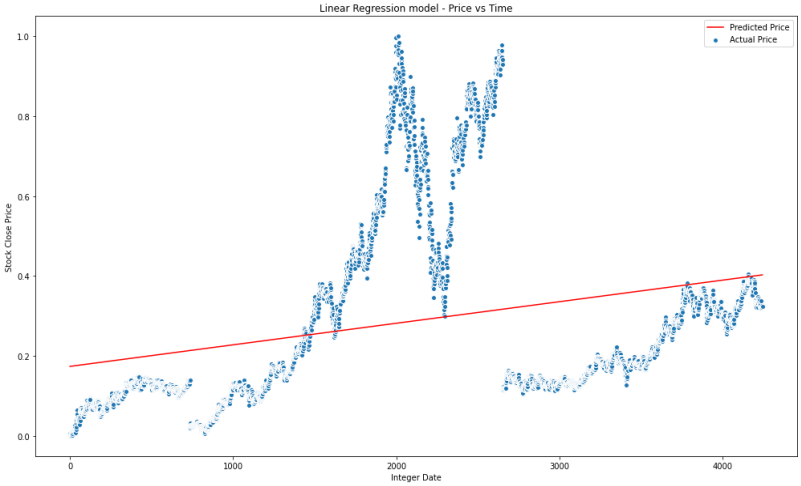
ITC: Train Data



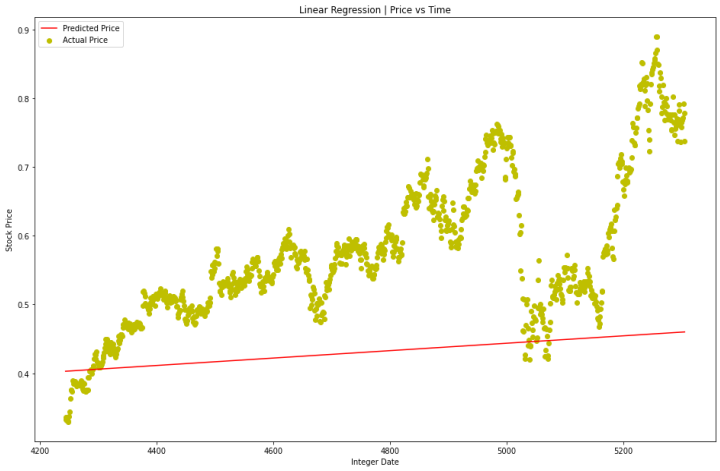
Test Data



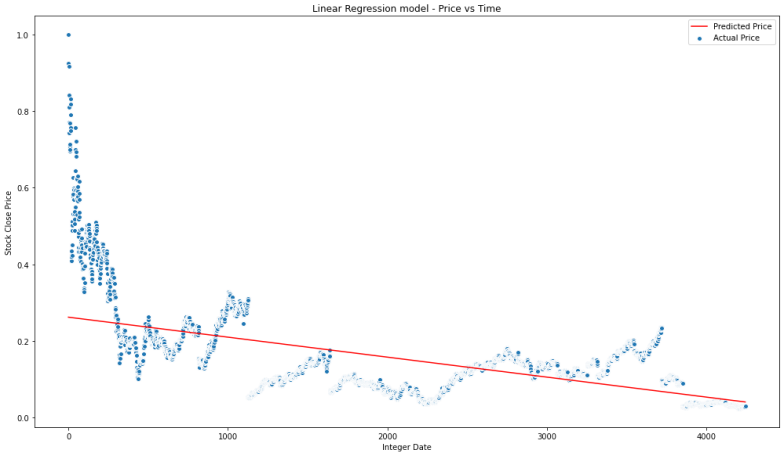
HDFC : Train Data



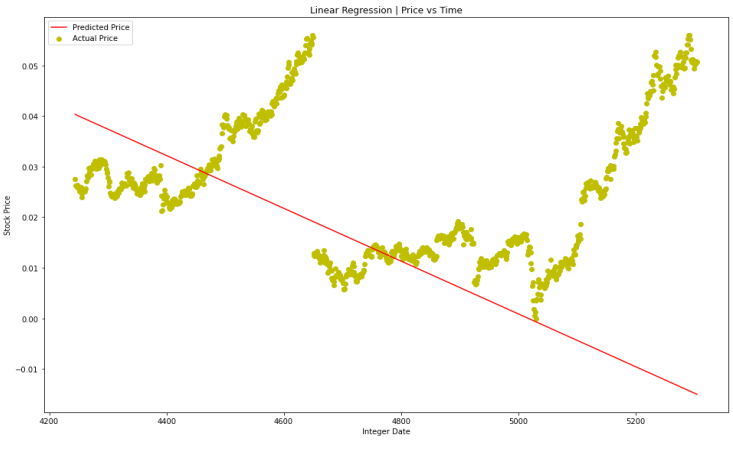
Test Data



INFOSYS : Train Data



Test Data



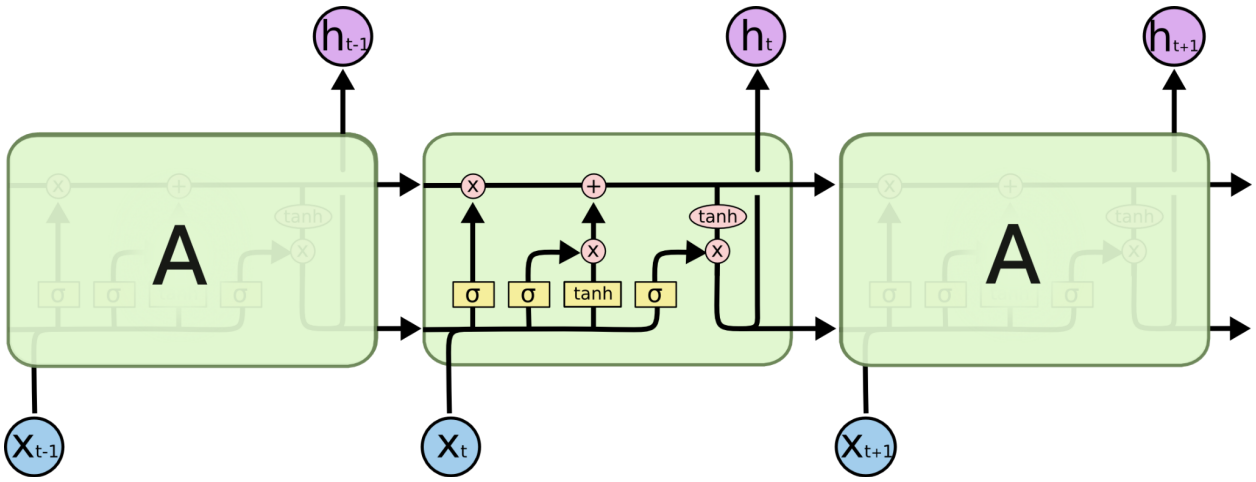
Recurrent Neural Networks

It is a class of neural networks tailored to deal with temporal data. The neurons of RNN have a cell state/memory, and input is processed according to this internal state, which is achieved with the help of loops within the neural network. There are recurring module(s) of 'tanh' layers in RNNs that allow them to retain information. However, not for a long time, which is why we need LSTM models.

LSTM Model

Recurrent neural networks (RNN) have proved one of the most powerful models for processing sequential data. Long Short-Term memory is one of the most successful RNNs architectures. LSTM introduces the memory cell, a unit of computation that replaces traditional artificial neurons in the hidden layer of the network. With these memory cells, networks are able to effectively associate memories and input remotely in time, hence suited to grasp the structure of data dynamically over time with high prediction capacity.

LSTMs have a chain-like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way



The repeating module in a LSTM

Analysis

For analyzing the efficiency of the system we have used the Mean Square Error(MSE). The error or the difference between the target and the obtained output value is minimized by using

MSE value. MSE is the mean/average of the square of all of the errors. The use of MSE is highly common and it makes an excellent general purpose error metric for numerical predictions.

Implementation:

1. Using Scikit Learning (Machine Learning model)
2. Data Preprocessing using dataset
3. Visualization of Dataset
4. Feature Scaling
5. Preparing the Datasets for training
6. Reshaping the datasets
7. Model development
8. Implementation of sequential, dense, LSTM and dropout.
9. Preprocessing the Data
10. Predicting the Output
11. Result visualization

Methodology

Stage 1: Raw Data: In this stage, the historical stock data is collected as per described in the dataset part. and this historical data is used for the prediction of future stock prices.

Stage 2: Data Preprocessing: The pre-processing stage involves

a) Data discretization: Part of data reduction but with particular importance, especially for numerical data

b) Data transformation: Normalization.

c) Data cleaning: Fill in missing values.

d) Data integration: Integration of data files. After the dataset is transformed into a clean dataset, the dataset is divided into training and testing sets so as to evaluate. Here, the training values are taken as the more recent values. Testing data is kept as 5-10 percent of the total dataset.

Stage 3: Feature Extraction: In this layer, only the features which are to be fed to the neural network are chosen. We will choose the feature from Date, open, high, low, close, and volume.Here have chosen Date as feature data.

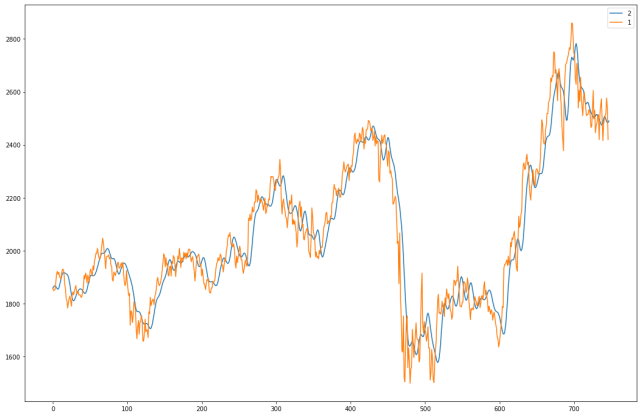
Stage 4: Training Neural Network: In this stage, the data is fed to the neural network and trained for prediction assigning random biases and weights. Our LSTM model is composed of a sequential input layer followed by 2 LSTM layers and a dense layer with ReLU activation and then finally a dense output layer with linear activation function.

Results

ITC Testing Data

Infosys Testing Data

HDFC Testing Data



Errors obtained using Linear

Companies	HDFC	ITC	INFOSYS
Mean Squared Error	245146.57934	53423.3812	159334.2670
Mean Absolute Error	416.1567	228.669	280.6234

Coefficient of Determination	-1.5125	-29.9926	-2.1797

Errors obtained using LSTM

Companies	HDFC	ITC	INFOSYS
Mean Squared Error	5933.854308397654	61.66858715680009	3004.878401029405
Mean Absolute Error	56.56770254532921	5.757666215539615	28.037047148515832
Coefficient of Determination	0.9259315325191522	0.9683613473676155	0.9542569842578266

Conclusion

The popularity of stock market trading is growing rapidly, which is encouraging researchers to find new methods for the prediction using new techniques. The forecasting technique is not only helping the researchers but it also helps investors and any person dealing with the stock market. In order to help predict the stock indices, a forecasting model with good accuracy is required.