Capstone Project -II

YES BANK STOCK CLOSING PRICE PREDICTION

BY

POOJA HOOLAHERI

Content

1. Introduction
2. Problem Statement
3. Data Cleaning
4. Exploratory Data Analysis (EDA)
5. Transforming Data
6. Splitting Data
7. Fitting Different Model
8. Conclusion

YES BANK

Yes Bank is a well-known bank in the Indian financial domain. It has been in the headlines since 2018 as a result of the Rana kapoor fraud case. Due to this, it was interesting to observe how it affected the company's stock prices and whether Time series models or other prediction models could properly reflect for such circumstances. Since the bank's founding, this dataset has included closing, starting, highest, and lowest stock prices for each month.

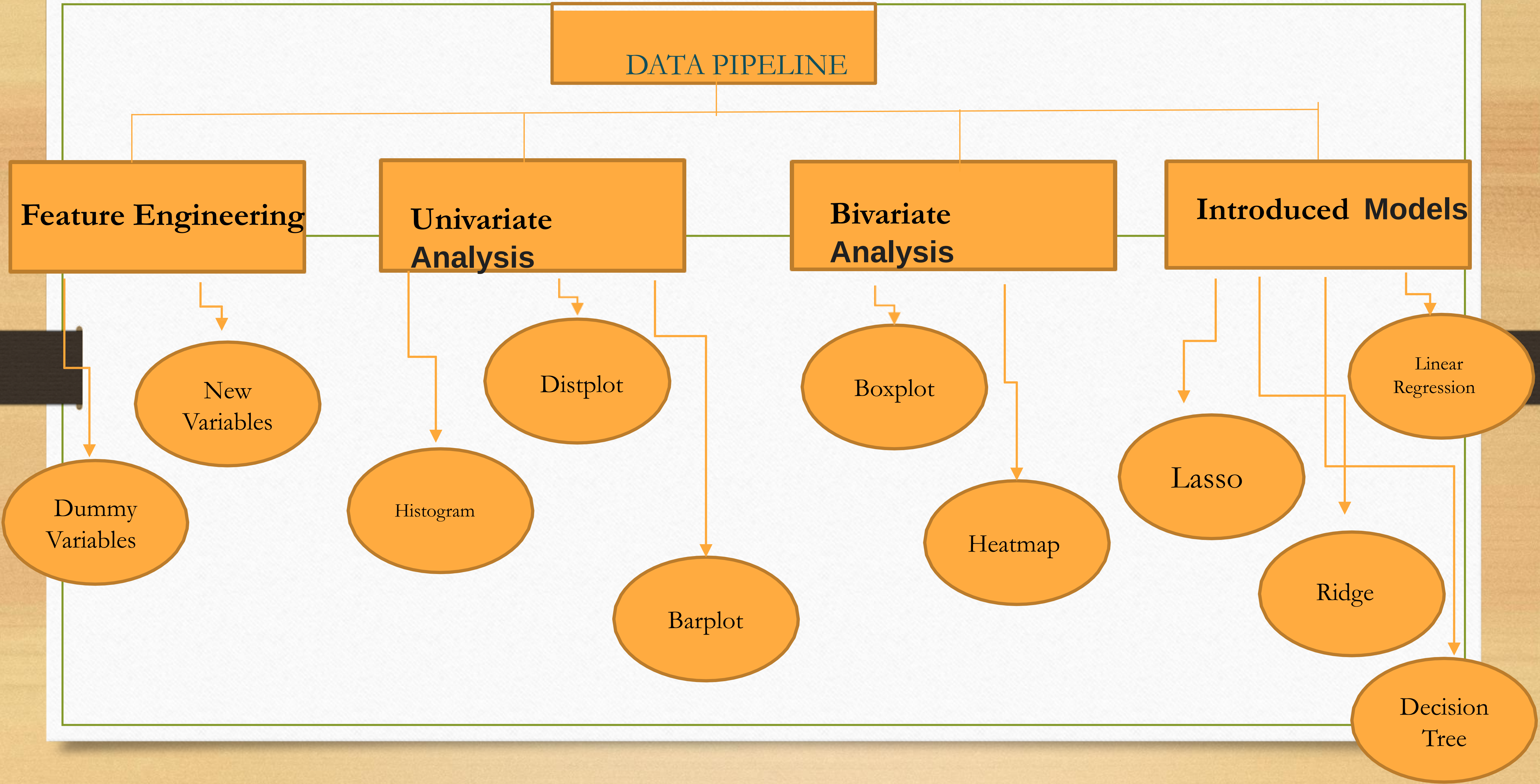
YES BANK STOCK CLOSING PRICE Prediction DATASET

We have 185 rows and 5 columns in our dataset. Here our dependent variable is Close and Independent variable is Open, High and Low.

Date :- It denotes the month and year for a specific pricing.
Open :- The price at which a stock started trading that month is referred to as the "Open."
High :- The highest price for that particular month.
Low :- It describes the monthly minimum price.
Close :- It refers to the final trading price for that month, which we have to predict using regression.

Data Cleaning

- Null Values Treatment
- Duplicated Values Treatment
- Date Format Change (i.e from Jul-05 to 2005-07-01)
- Checking outliers
- So after successfully cleaning the dataset we have 185 columns and 5 rows



DATA PIPELINE

Feature Engineering

New Variables

Dummy Variables

Univariate Analysis

Distplot

Histogram

Barplot

Bivariate Analysis

Boxplot

Heatmap

Introduced Models

Linear Regression

Lasso

Ridge

Decision Tree

Libraries:

1} NumPy

2} Panda

3} Matplotlib

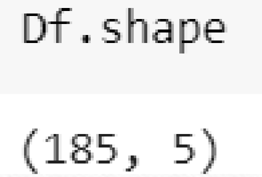
4} Seaborn

5} Datetime

6} Sklearn

Data Wrangling:

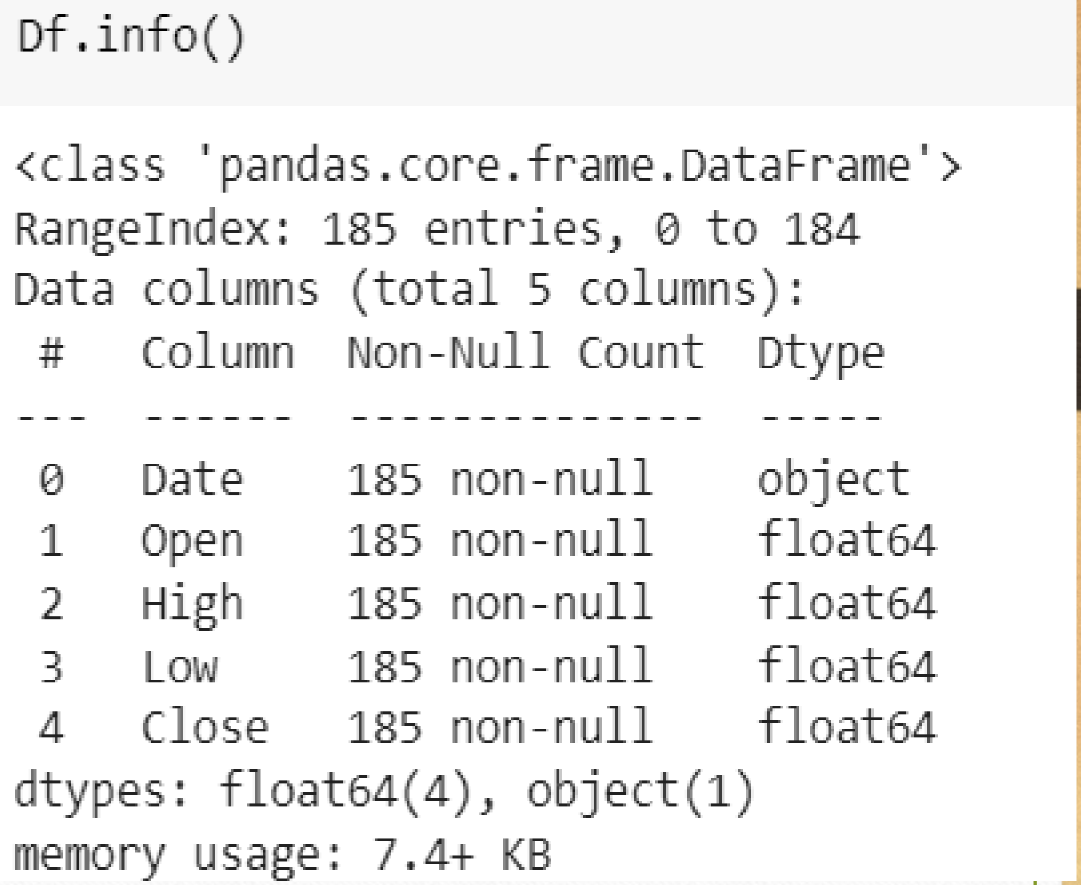
- Shape of the Data



```
Df.shape
```

```
(185, 5)
```

- Datatype in Data Frame

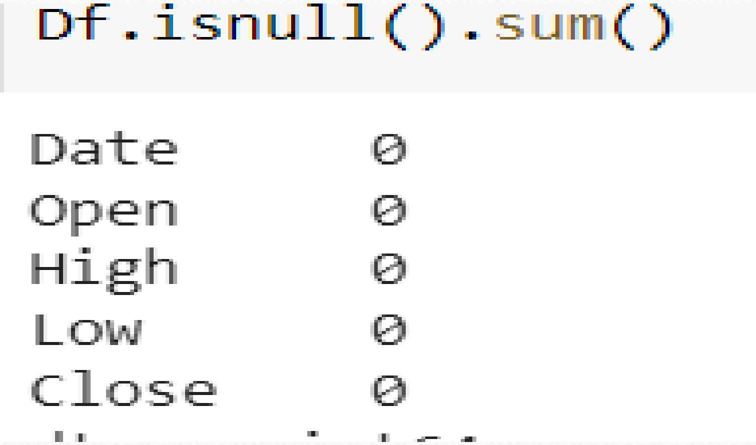


```
Df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 185 entries, 0 to 184
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Date    185 non-null    object 
 1   Open    185 non-null    float64
 2   High    185 non-null    float64
 3   Low     185 non-null    float64
 4   Close   185 non-null    float64
dtypes: float64(4), object(1)
memory usage: 7.4+ KB
```

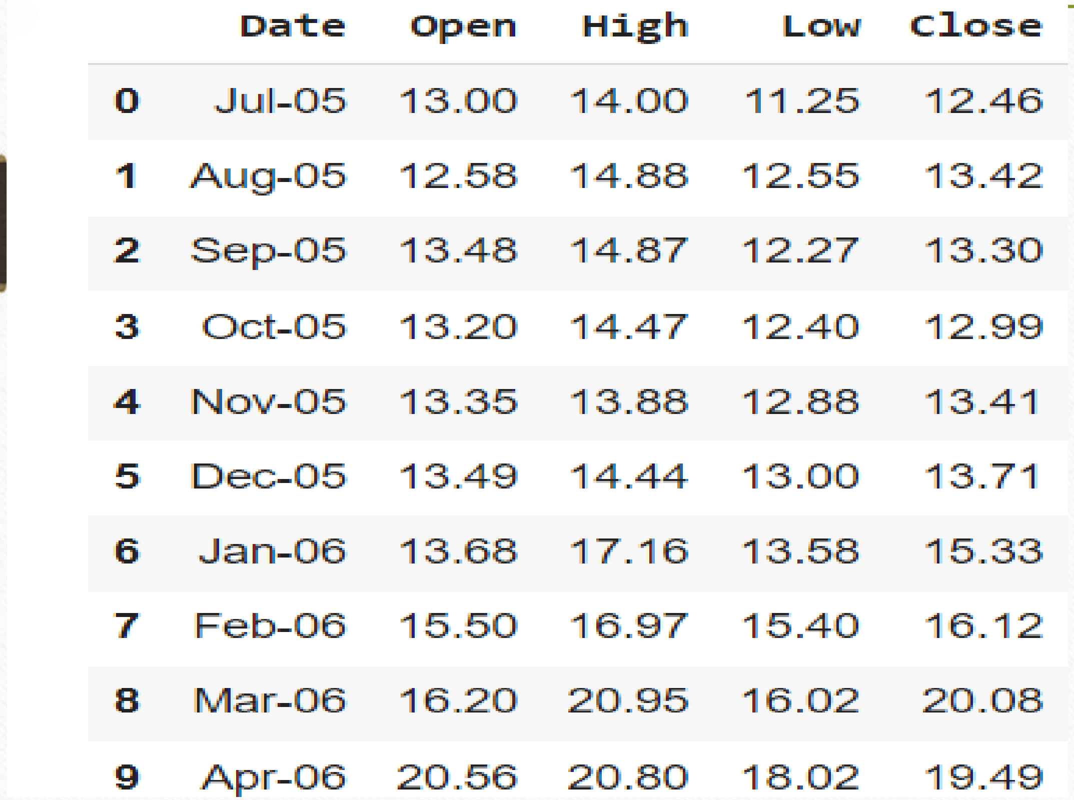
Data Wrangling(cont.)

Finding the Null values:



```
Df.isnull().sum()
```

```
Date        0
Open        0
High        0
Low         0
Close       0
```

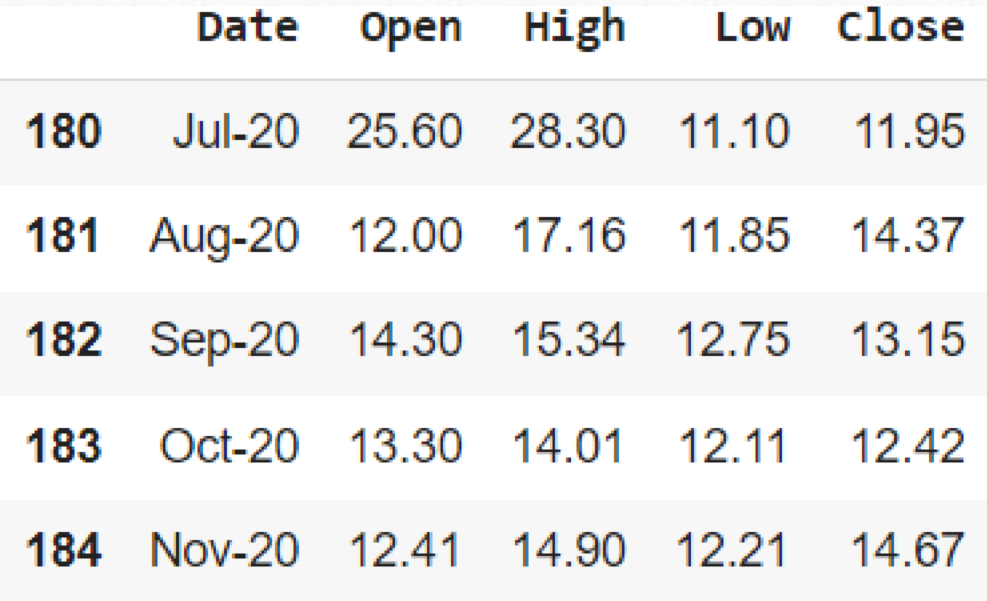


	Date	Open	High	Low	Close
0	Jul-05	13.00	14.00	11.25	12.46
1	Aug-05	12.58	14.88	12.55	13.42
2	Sep-05	13.48	14.87	12.27	13.30
3	Oct-05	13.20	14.47	12.40	12.99
4	Nov-05	13.35	13.88	12.88	13.41
5	Dec-05	13.49	14.44	13.00	13.71
6	Jan-06	13.68	17.16	13.58	15.33
7	Feb-06	15.50	16.97	15.40	16.12
8	Mar-06	16.20	20.95	16.02	20.08
9	Apr-06	20.56	20.80	18.02	19.49

Starting 10 Values

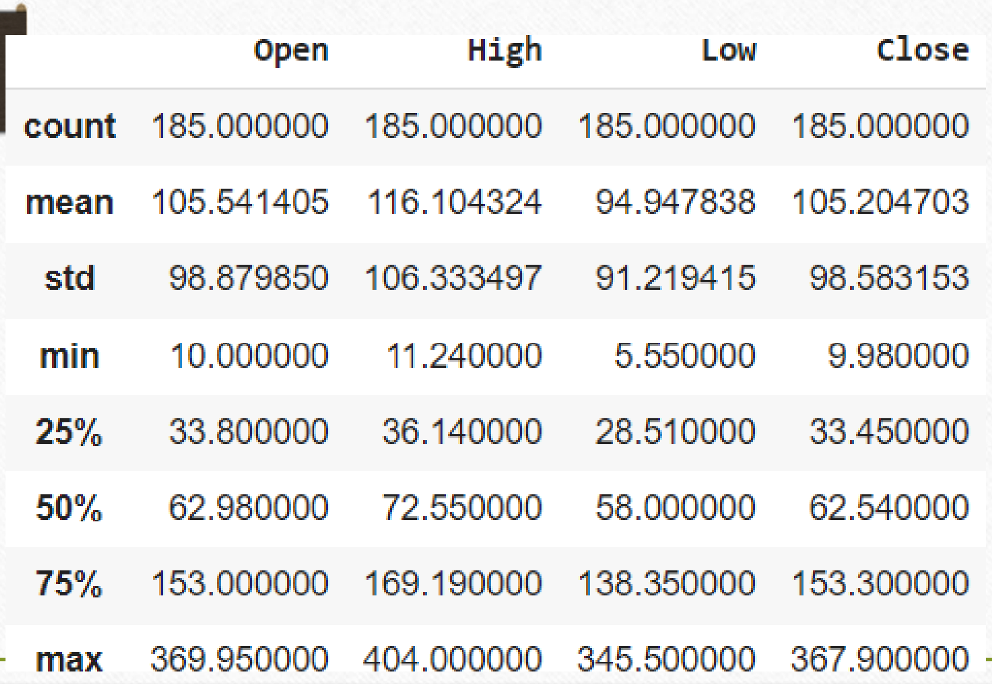
Data Wrangling(cont.)

Last 5 value in the datasets



	Date	Open	High	Low	Close
180	Jul-20	25.60	28.30	11.10	11.95
181	Aug-20	12.00	17.16	11.85	14.37
182	Sep-20	14.30	15.34	12.75	13.15
183	Oct-20	13.30	14.01	12.11	12.42
184	Nov-20	12.41	14.90	12.21	14.67

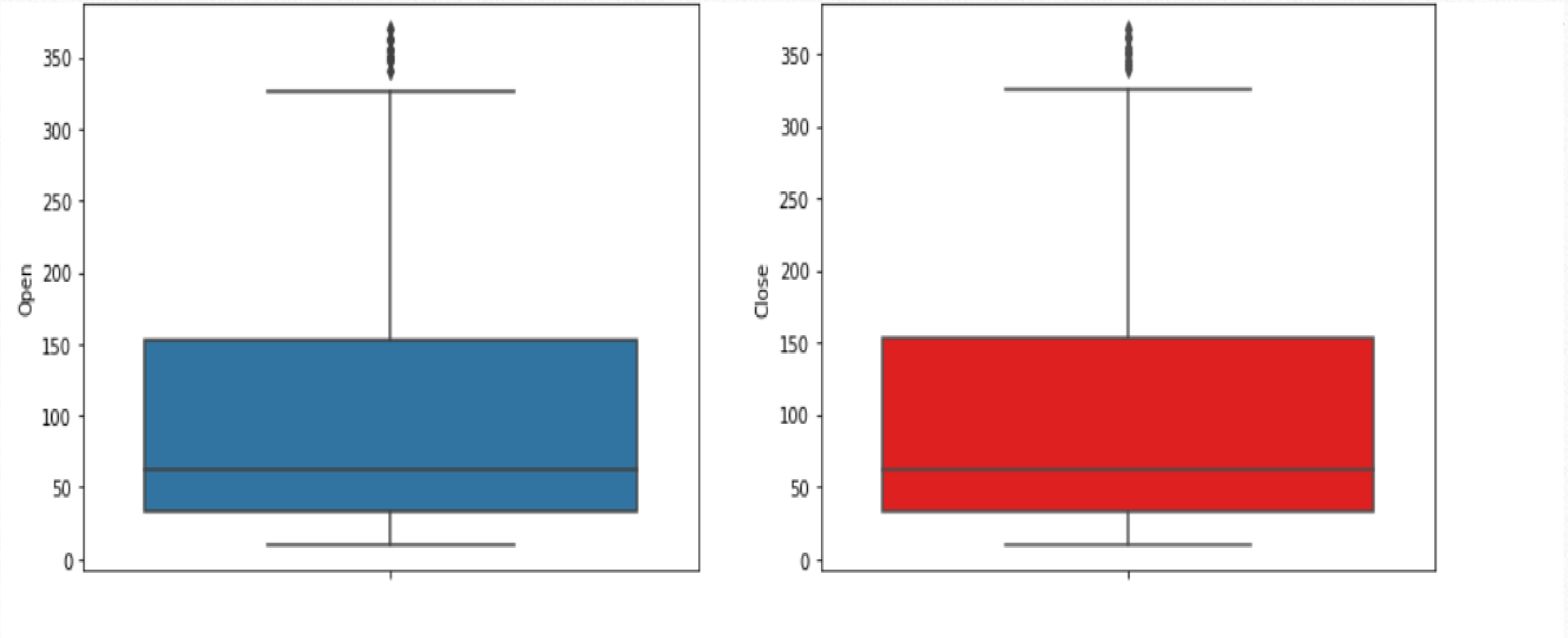
Description of datasets

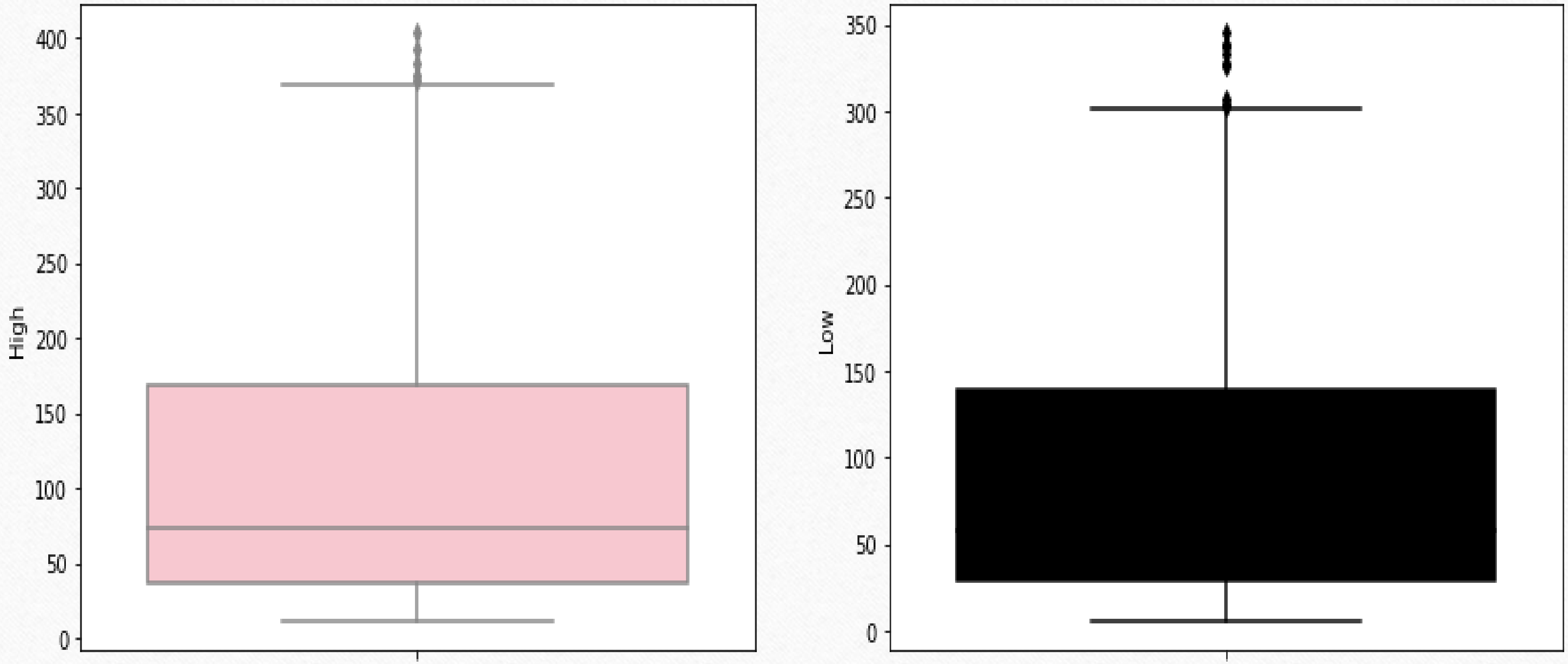


	Open	High	Low	Close
count	185.000000	185.000000	185.000000	185.000000
mean	105.541405	116.104324	94.947838	105.204703
std	98.879850	106.333497	91.219415	98.583153
min	10.000000	11.240000	5.550000	9.980000
25%	33.800000	36.140000	28.510000	33.450000
50%	62.980000	72.550000	58.000000	62.540000
75%	153.000000	169.190000	138.350000	153.300000
max	369.950000	404.000000	345.500000	367.900000

Exploratory Data Analysis

Treating Of Outliers

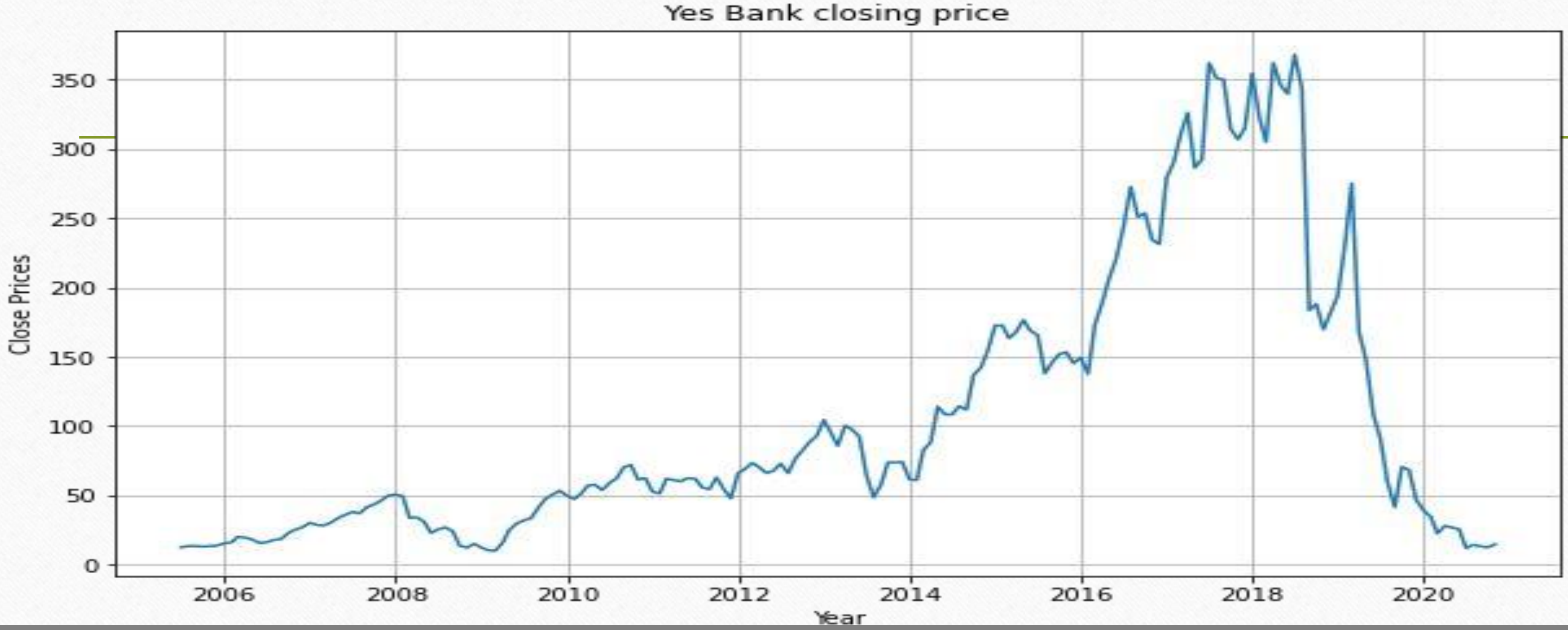




Outliners in the dataset

Exploratory Data Analysis

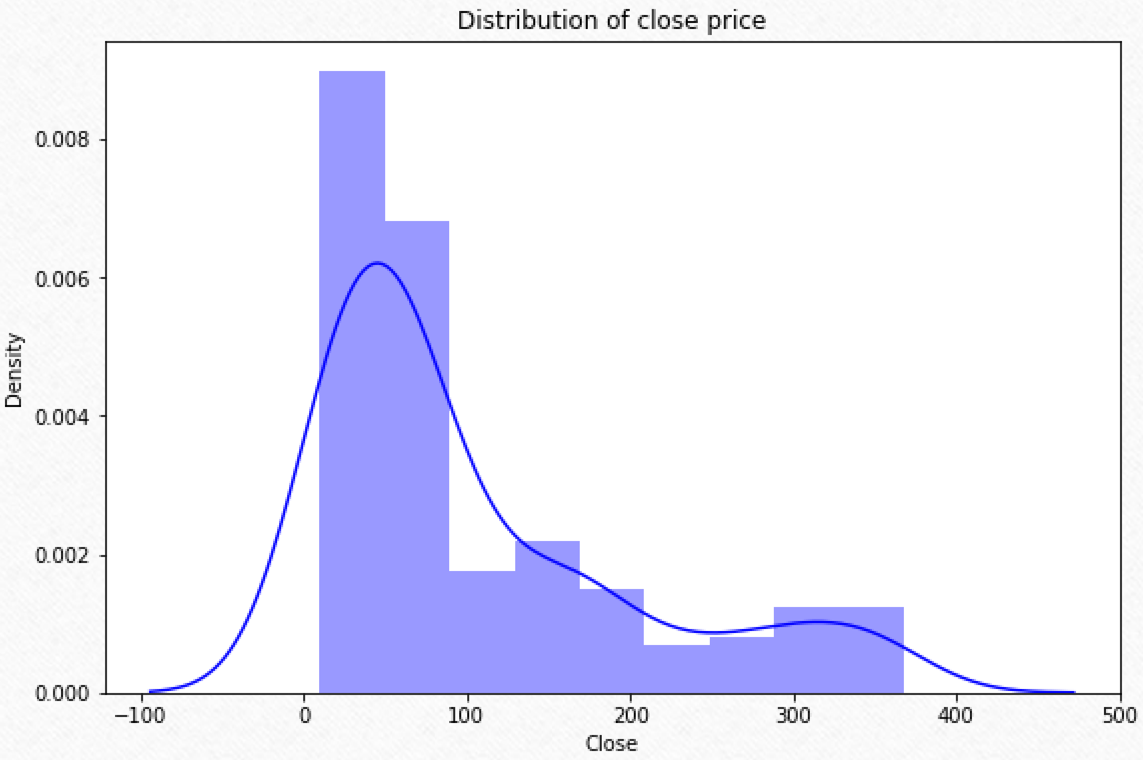
Visualising The Data



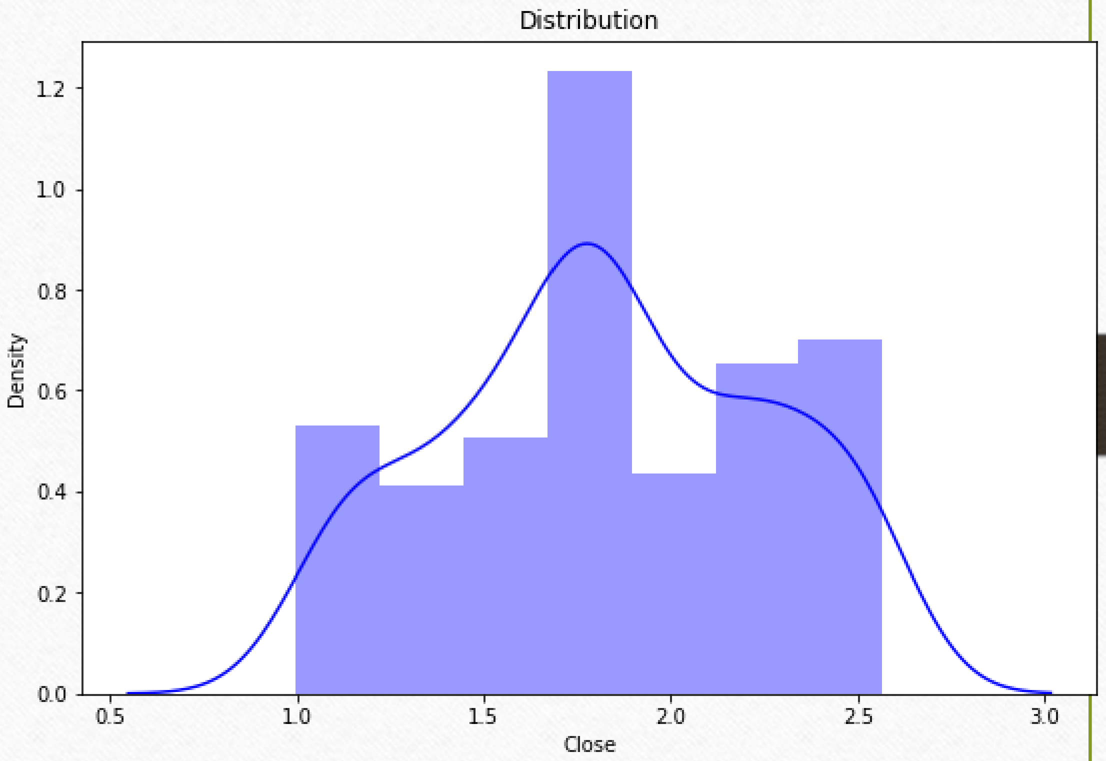
Here, it’s clearly visible form the above plot that the stock prices saw a significant rise from year 2006 to 2018 . However, since 2018 the stock proces saw a major downfall and that is may be due to the fraud case.

EDA (Continued)

- **Distribution of Closing Price :**



- **After Log Transformation :**

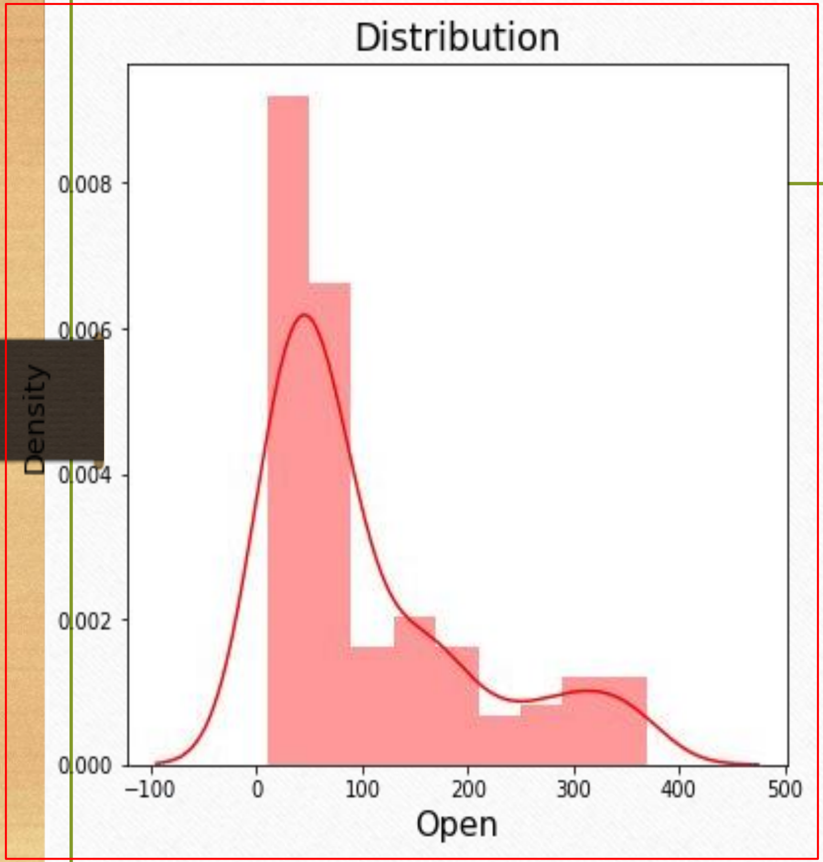


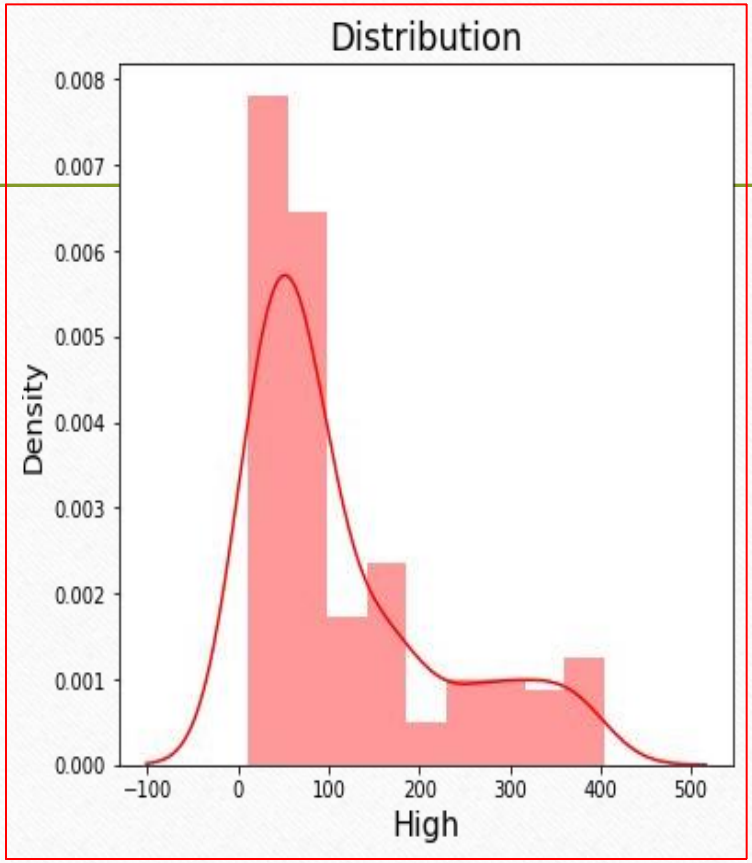
- Distribution of closing price is right skewed.
- We need this distribution to be normal distribution for training algorithm.

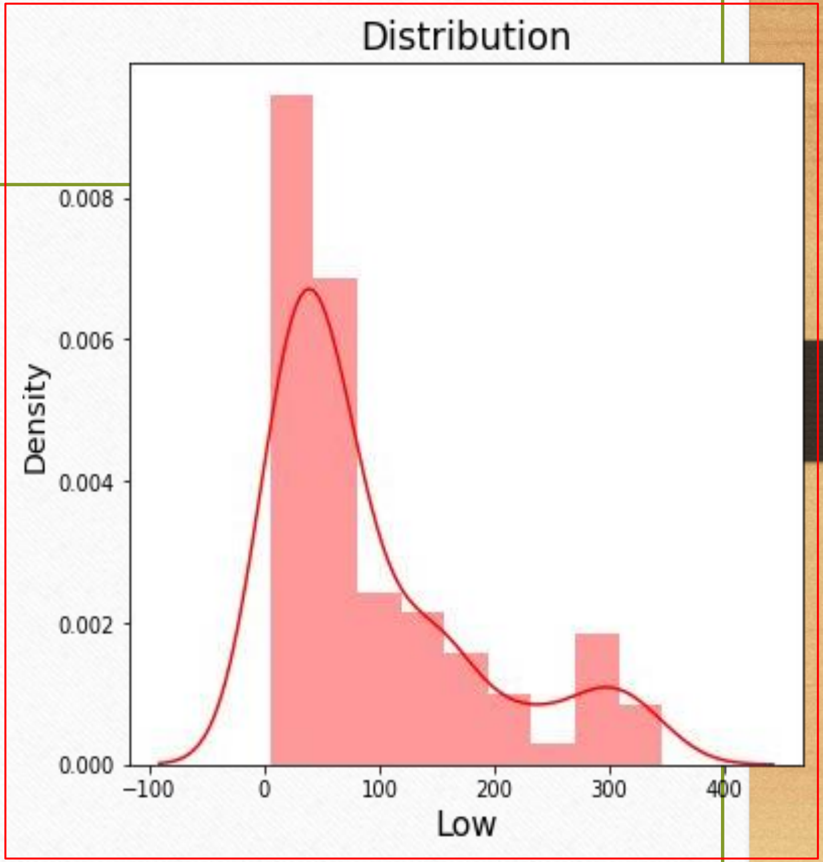
- Distribution of closing price is normal distribution.

EDA (Continued)

- **Distribution of Open, High & Low Price of a stock :**



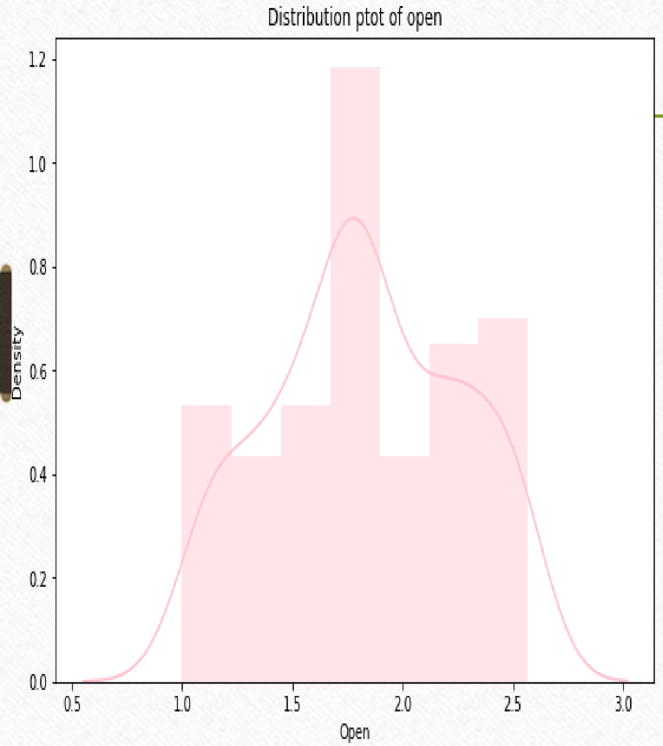




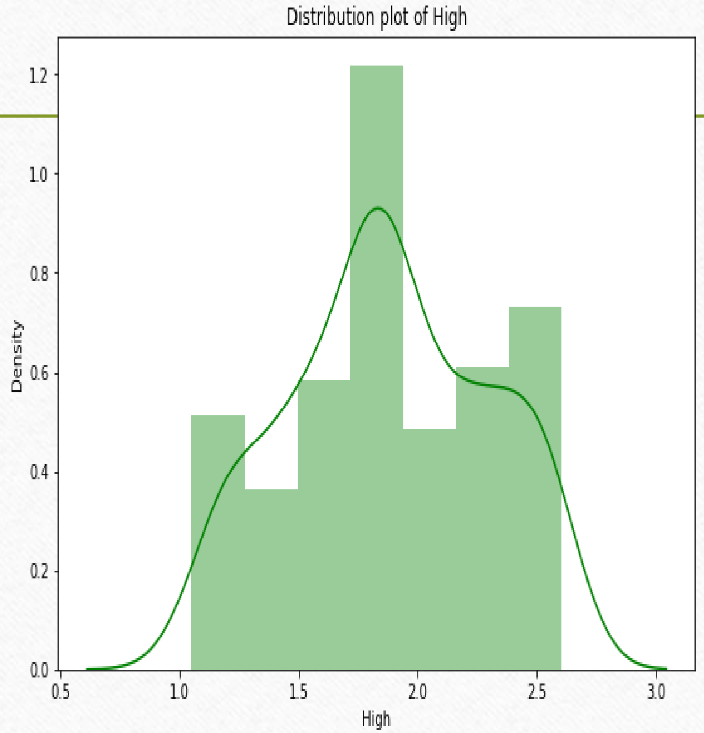
- Distribution of opening price, high price and low price are also right skewed.
- Log transformation applied to make this distribution to normal.

EDA (Continued)

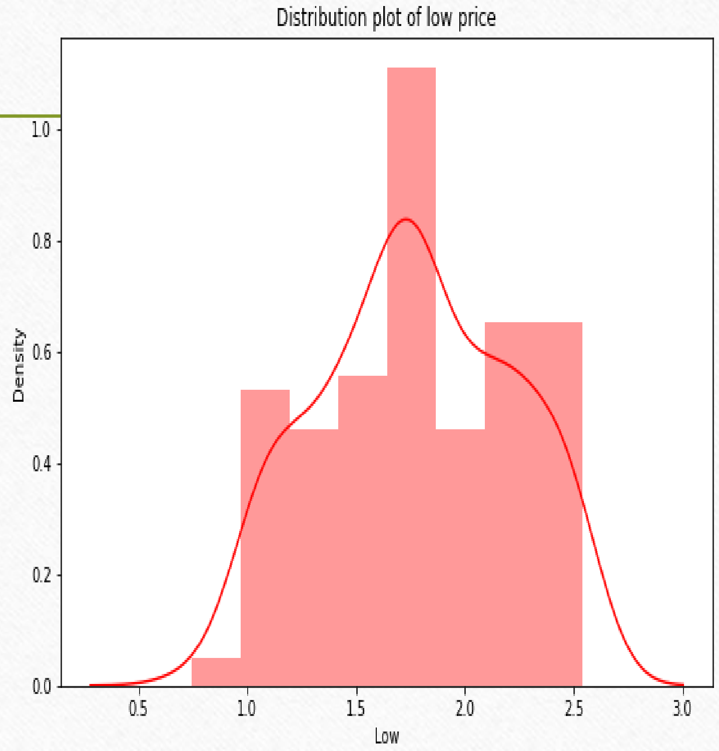
- **Distribution of Open, High & Low Price of a stock after Log Transformation :**



Distribution ptot of open



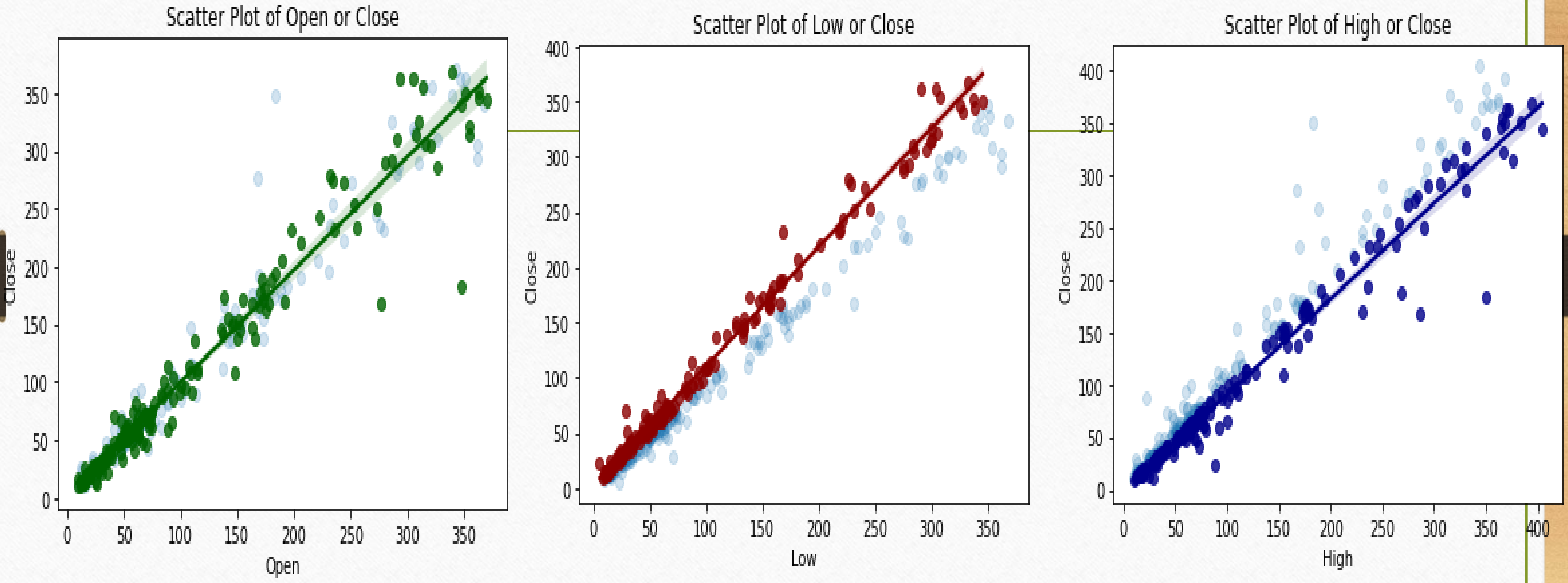
Distribution plot of High



Distribution plot of low price

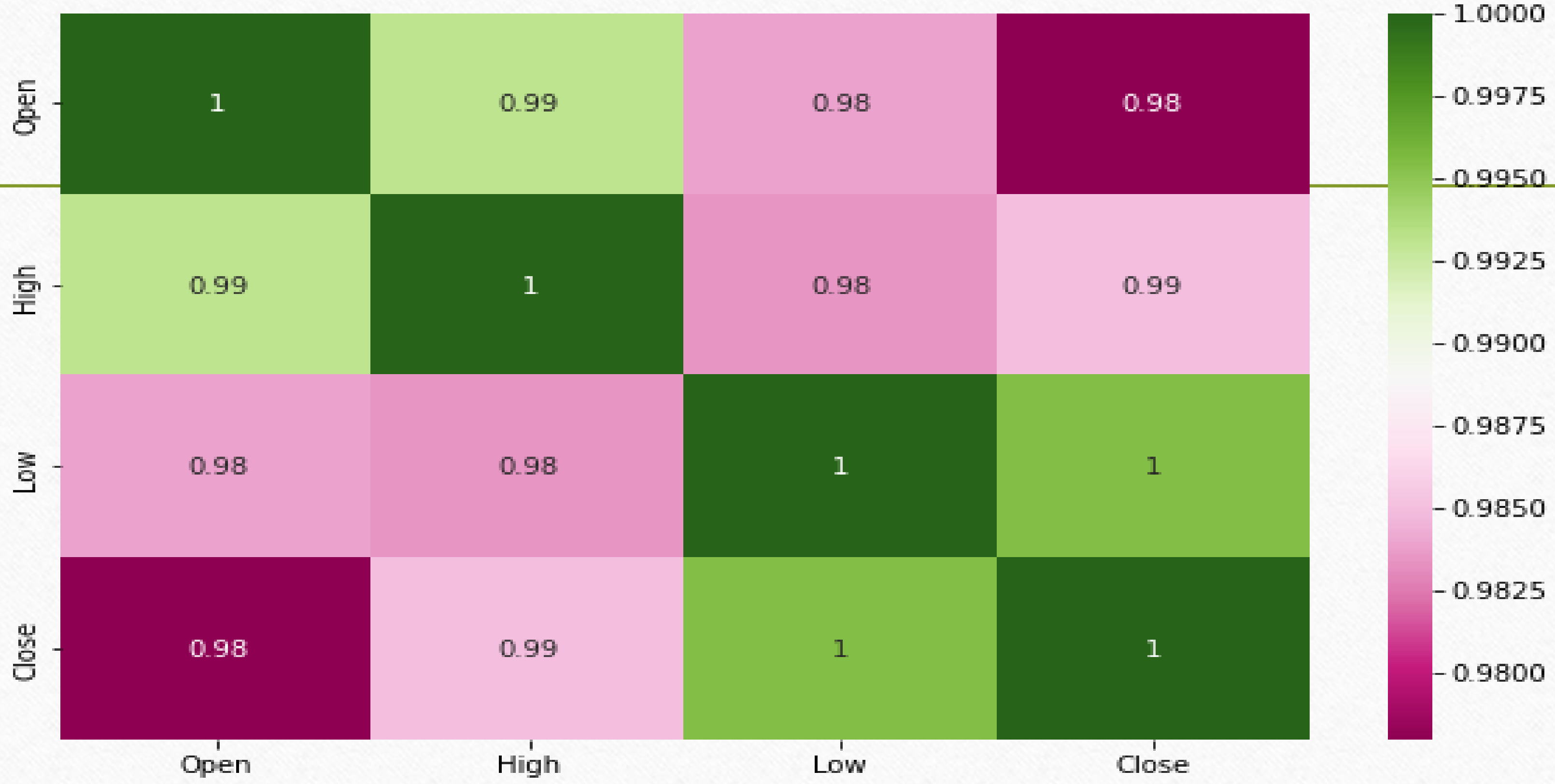
- Distribution of opening price, high price and low price are now normal distribution.

Bivariate Analysis Plots(Relation between the Dependent Variable and Independent Variable)



EDA (Continued)

- **Correlation :**



- All the features are strongly correlated with each other.

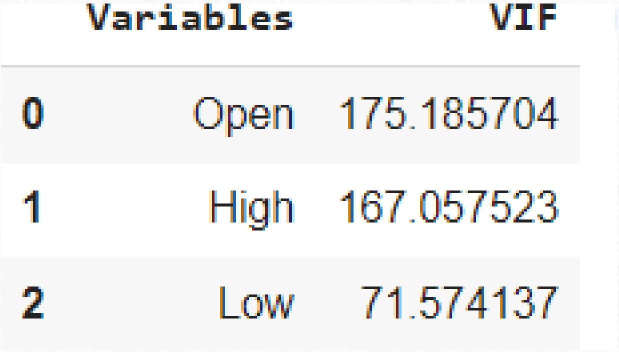
Multicollinearity:

- Even though we have strong VIF ratings, we won't do feature engineering because each feature is critical for this specific use case. Most indicators in the real world consider each of these characteristics to predict future values.

- Due to the fact that each column is equally crucial for prediction, we are not deleting any columns.

- Column removal resulted in the loss of important data (features) that are necessary for the model to make correct
- predictions. It produces a poor model.

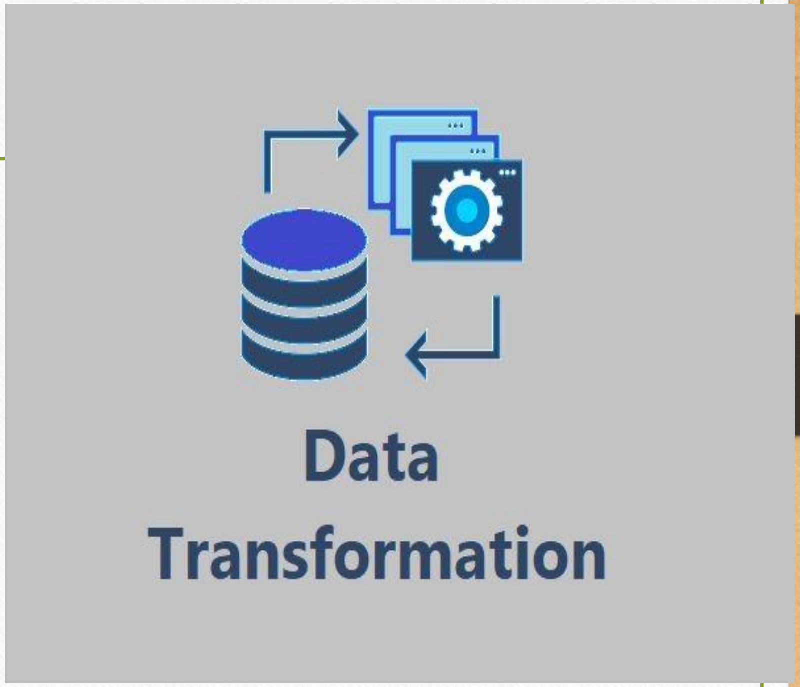
- Therefore, we are not removing any features from the dataset while we attempt to predict the outcome, assess the model's performance with respect to multicollinearity, and make adjustments as necessary.



	Variables	VIF
0	Open	175.185704
1	High	167.057523
2	Low	71.574137

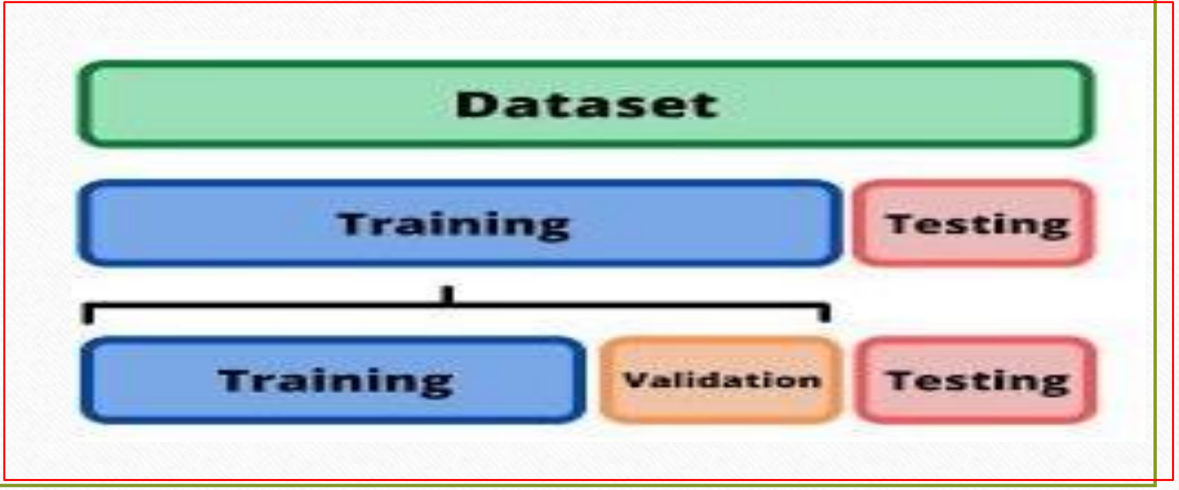
Transformation of Data

- To scale data into a uniform format that would allow us to utilize the data in a better way.
- For performing fitting and applying different algorithms to it.
- The basic goal was to enforce a level of consistency or uniformity to dataset.



Splitting Data

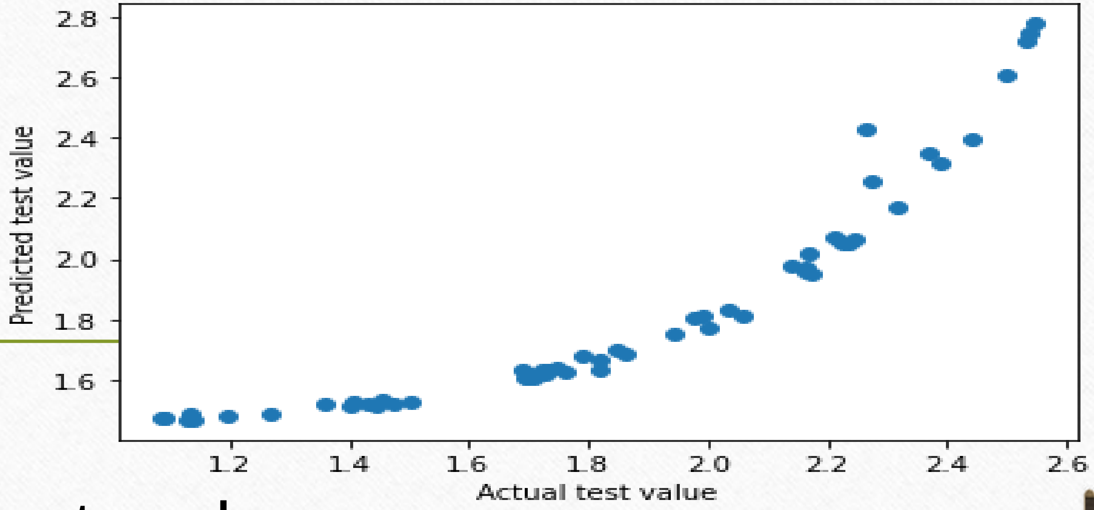
- Data splits into training dataset and testing dataset.
- Training dataset is for making algorithm learn and train model.
- Test dataset is for testing the performance of train model.
- Here 80% of data taken as training dataset & remaining 20% of dataset used for testing purpose.



Create 4 Regression Models for This Data.

- 1} Linear Regression
- 2} Lasso Regression
- 3} Ridge Regression
- 4} Decision Tree Regression

Fitting Different Model

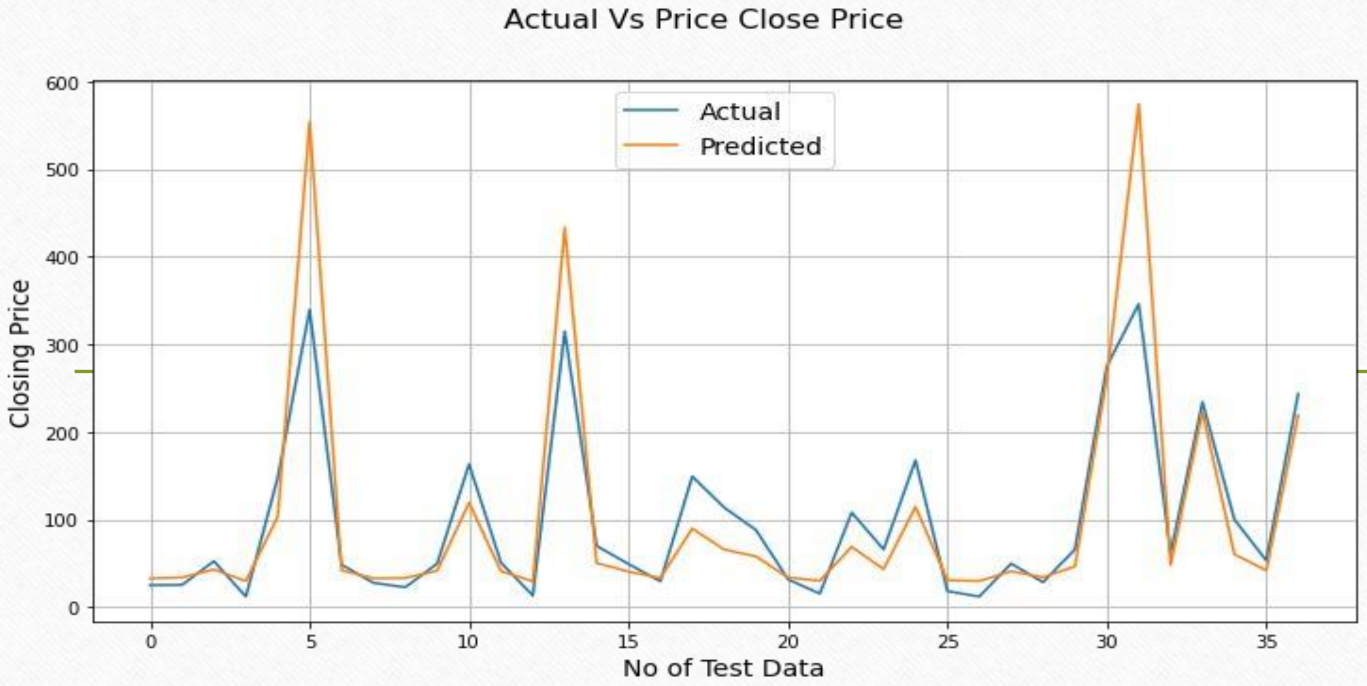


Linear Regression

•Linear regression is one of the easiest and most popular Machine Learning algorithms.

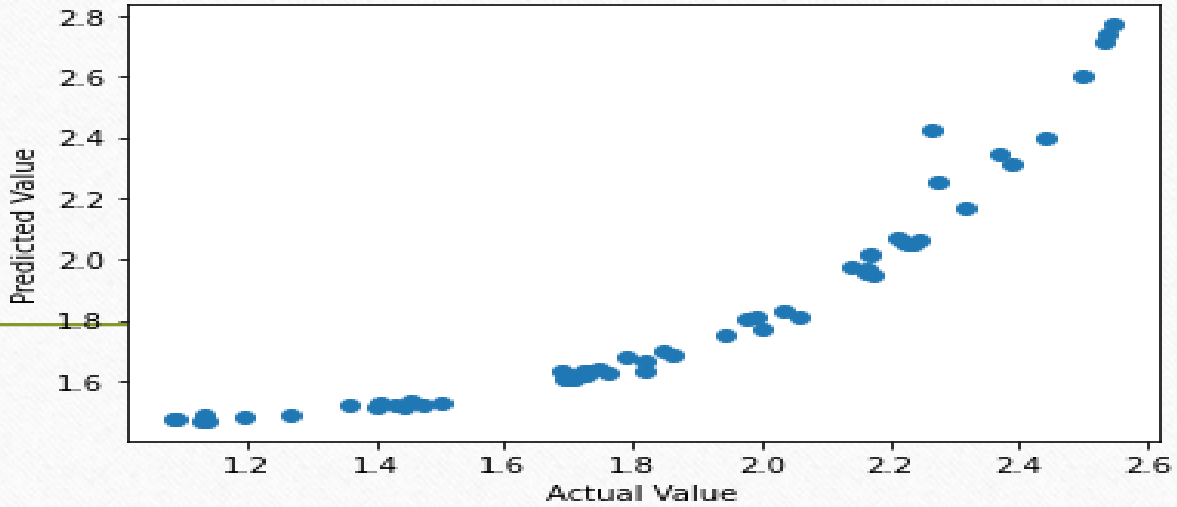
• It is a statistical method that is used for predictive analysis.

•Linear regression algorithm shows a linear relationship between a dependent and independent variable; hence it is called as linear regression.

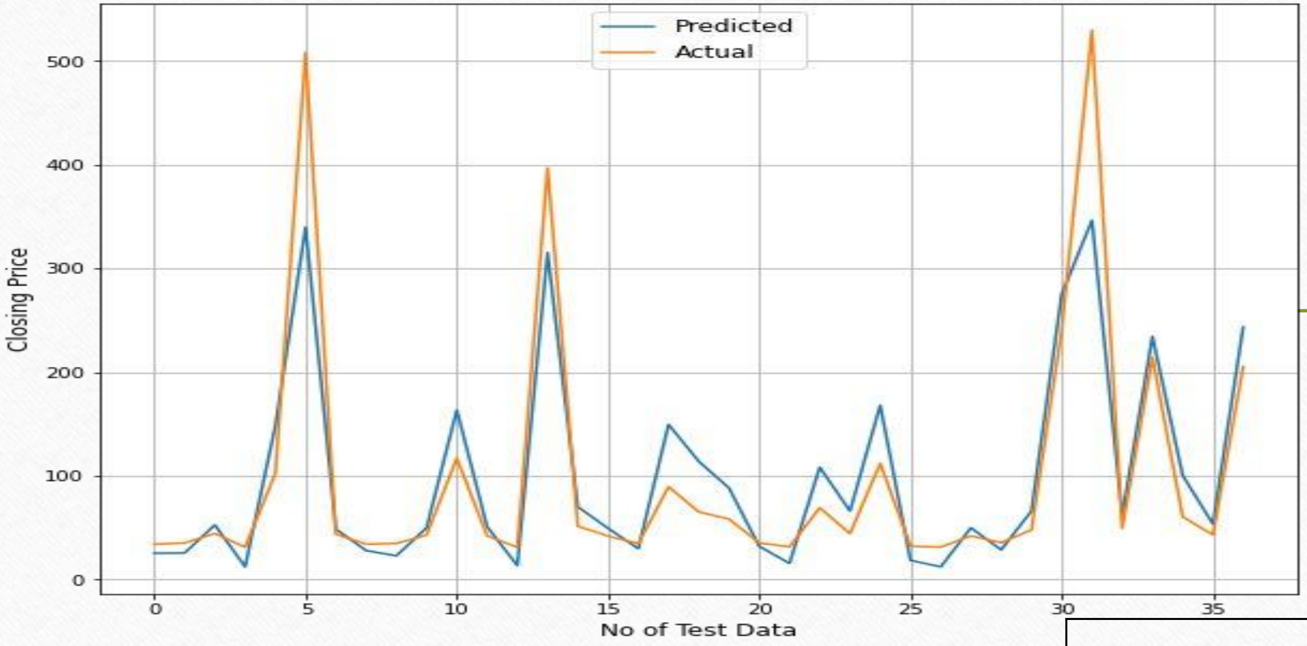


Evaluation Metrics: Linear Regression				
MSE	RMSE	MAE	MAPE	R2
0.0379	0.1814	0.1594	0.0964	0.8103

Lasso Regression



- Lasso: Least Absolute Shrinkage and Selection operator
- It is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the resulting statistical model.
- This method performs L1 regularization.



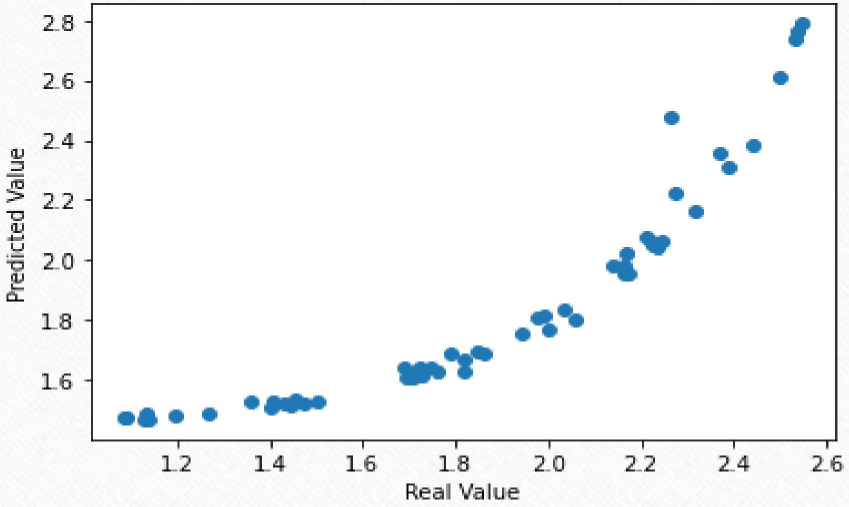
Evaluation Metrics: Lasso Regression				
MSE	RMSE	MAE	MAPE	R2
0.0331	0.1806	0.1598	0.0968	0.8094

Ridge Regression

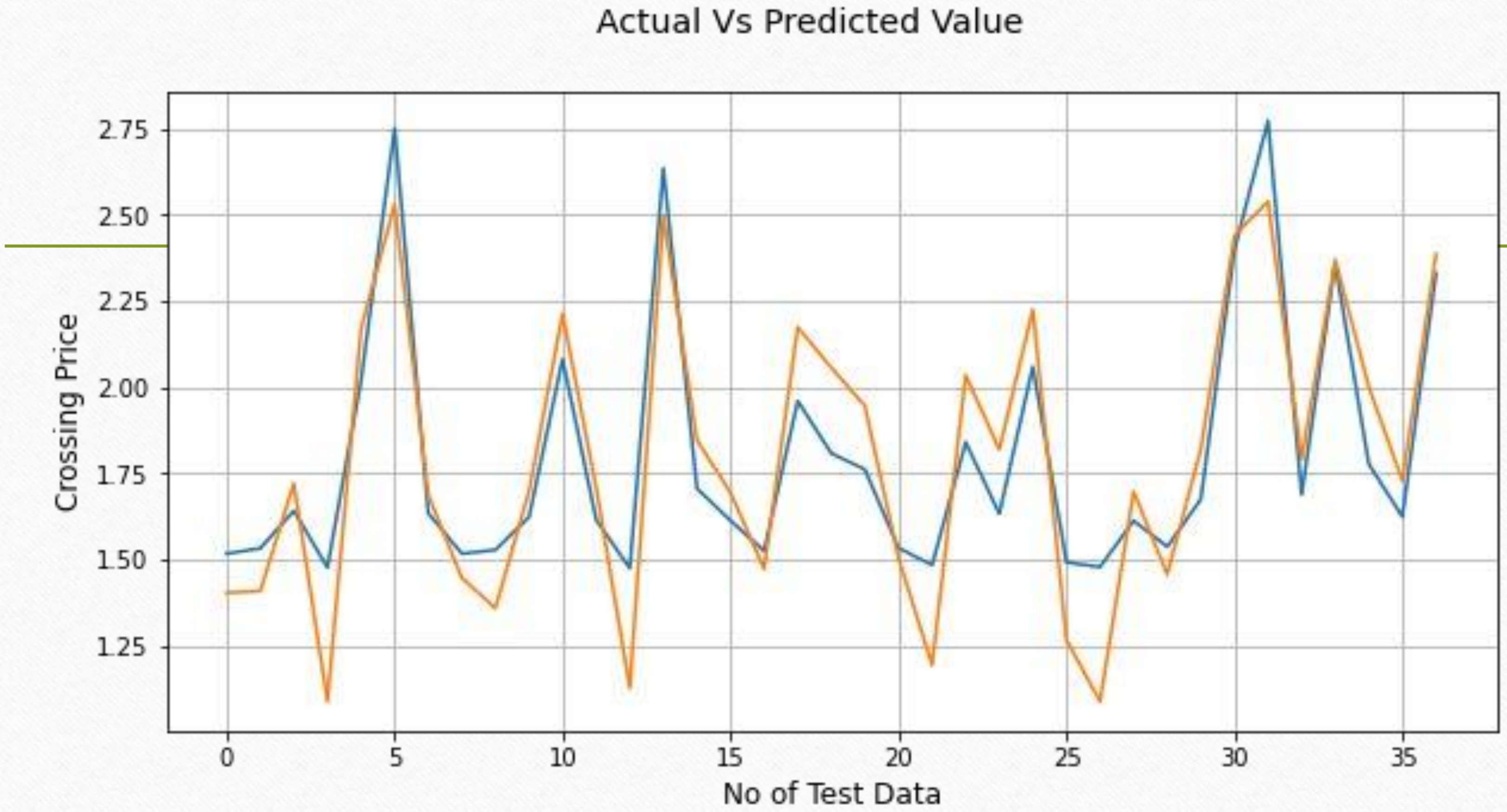
•Ridge regression is a model tuning method that is used to analyses any data that suffers from Multicollinearity.

•When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values to be far away from the actual values.

- This method performs L2 regularization.



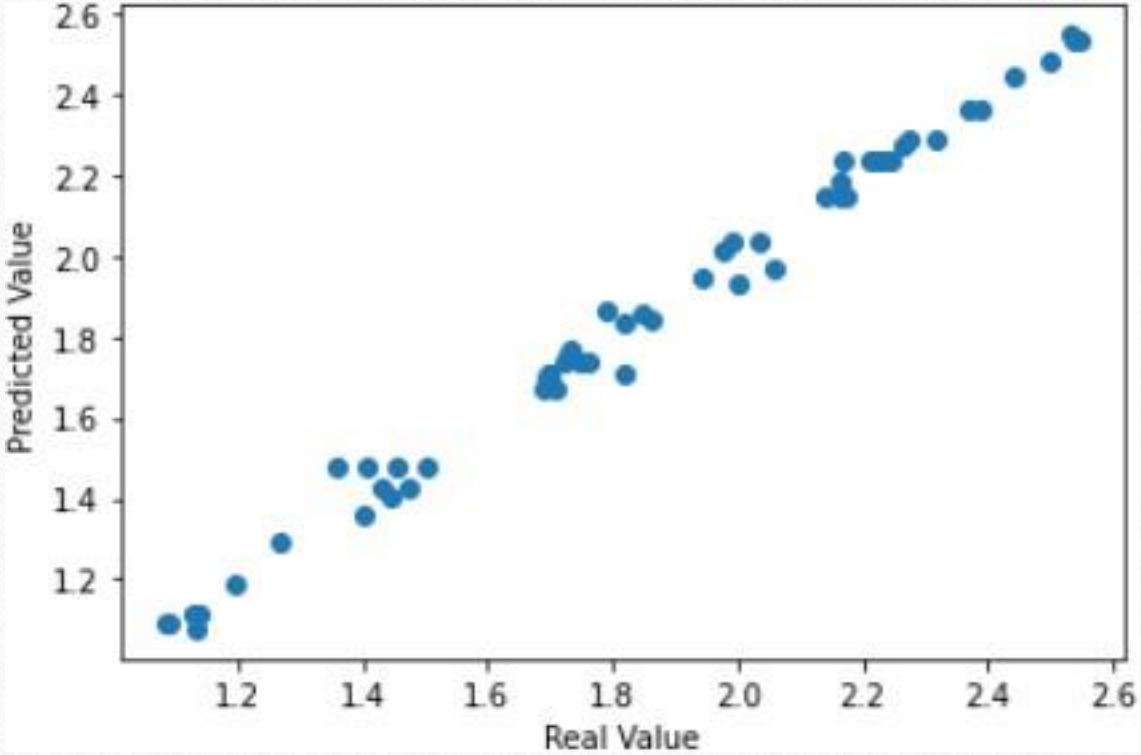
Evaluation Metrics: Ridge Regression				
MSE	RMSE	MAE	MAPE	R2
0.0337	0.1835	0.1614	0.0973	0.8050


Actual Vs Predicted Value
Crossing Price
2.75
2.50
2.25
2.00
1.75
1.50
1.25
0
5
10
15
20
25
30
35
No of Test Data

Decision Tree Regression:

Decision tree regression trains a model in the form of a tree to predict data in the future and generate useful continuous output by observing the properties of an item.

Evaluation Metrics: Decision Tree Regression				
MSE	RMSE	MAE	MAPE	R2
0.002	0.0447	0.0308	0.0175	0.9885



Conclusion:

- The popularity of stock closing is growing extremely rapidly day by day which encourage researcher to find new methods if any fraud happens.
- This technique is used for prediction is not only helpful to researchers to predict future stock closing prices or any fraud happen or not but also helps investors or any person who dealing with the stock market in order to prediction of model with good accuracy.

Both duplicate and null values are absent, as we have seen. But object data type values are available for the Date feature. Therefore, we transformed it to the correct date format, YYYY-MM-DD.

Conclusion:

- The dependent and independent values were found to be linearly related
- The data contained a significant amount of multicollinearity.
- In this work we use linear regression technique, lasso regression, ridge regression, and Decision Tree Regression technique these Four models gives us the following results
- High, low, open are directly correlate with the closing price of stocks
- Target variable(dependent variable) strongly dependent on independent variables

- Decision Tree regression Is best model for yes bank stock closing price data this model use for further prediction

Visualization has allowed us to notice that the closing price of the stock has suddenly fallen starting in 2018. It

seems reasonable that the Yes Bank stock price was significantly impacted by the Rana Kapoor case fraud.

THANK YOU