**Yes Bank Stock Closing Price Prediction**

**Shubham Narendra Kadu**

**Data science trainees**

**AlmaBetter**

**Abstract:**

Yes Bank is a well-known bank in the Indian financial domain. Yes Bank Limited is headquartered in Mumbai, India, and was founded by Rana Kapoor and Ashok Kapoor in 2004.

The YES BANK stock market offers excellent investors opportunities by buying a stock and becoming a stockholder to earn from long-term benefits or trading on the stock till 2018 after that the stock price of yes bank actually fell from 2018 onwards which shows the impact. Since 2018, it has been in news because of the fraud case involving Rana Kapoor. Owing to this fact, it was interesting to see how that impacted the stock prices of the company and whether Time series models or any other predictive models can do justice to such situations. This dataset has monthly stock prices of the bank since its inception and includes closing, starting, highest, and lowest stock prices of every month.

**1. Problem Statement**

Yes-Bank is a well-known bank in the Indian financial domain. Since 2018, it has been in the news because of the fraud case involving Rana Kapoor. Owing to this fact, it was interesting to see how that impacted the stock prices of the company and whether Time series models or any other predictive models can do justice to such situations. This dataset has monthly stock prices of the bank since its inception and includes closing, starting, highest, and lowest stock prices of every month.

**2. Goal**

### The main aim of the project is to understand data and visualize data for insight from the monthly stock prices of the bank since its inception and including the closing, starting, highest, and lowest stock prices of every month. The main objective is to predict the stock’s closing price of the month.

### **3. Introducti****on**

The YES Bank stock market is a dynamic and volatile Industry and Investors generally decide to buy or sell the stock based on the company’s past and present performance. Owing to this fact, it was interesting to see how that impacted the stock prices of the company. The dataset contains the monthly stock price details for Yes Bank [‘Date’, ‘Open’, ‘High’, ‘Low’, ‘Close’]. The main objective of this project is to predict the stock’s closing price of the month

**4. Feature Description**

1) Date: Date of the month of stock price.

2) Open: The opening price of the stock on a particular day

3) High: It's the highest price at which a stock traded during a period

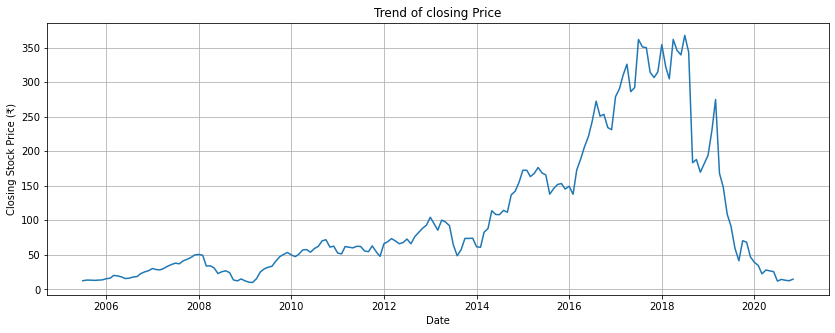
4) Low: It's the lowest price at which stock traded during a period

5) Close: The closing price of a stock at the end of a Trading Day

**4. Exploratory Data Analysis**

Exploratory Data Analysis (EDA) plays a vital role in the analysis of the data variables which are important from the aspect of feature engineering. It will help us to distribute and relate between dependent and independent variables. We have gone through an analysis of every independent as well as the dependent variable to check which independent factor affects the dependent factor.

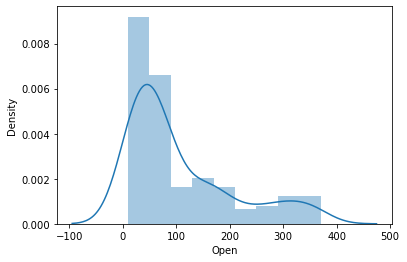
**4.1)** **Visualize the trend of dependent Variable**



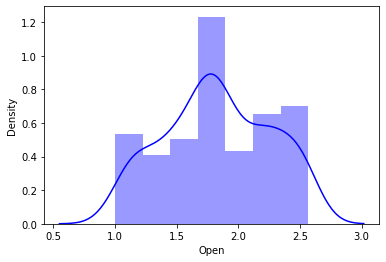
The above plot of Closing prices of different dates gives a very fluctuation in prices regarding different time-duration. After 2018 there is a sudden fall in the stock closing price.

**4.2) Univariate Analysis**

**Visualize the Distribution of Open stock price**

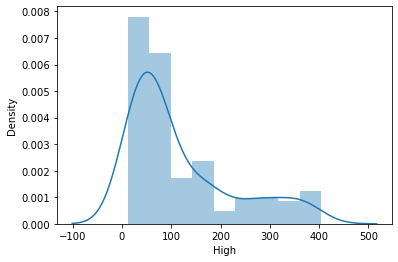
****

The above distribution of Open Stock Price is a positively right-skewed distribution. this is not a perfect normal distribution, so we have to apply some kind of transformation to see if it will look like normal distribution or not. Let’s see after Log Transformation.

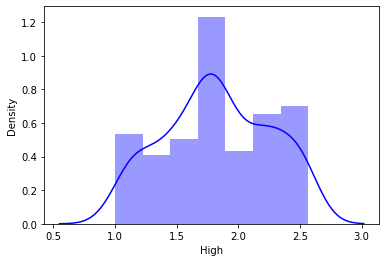


Now we can look at this distribution it is not a perfectly normal distribution but more or less its looking normal distribution.

**Visualize the Distribution of High stock price**

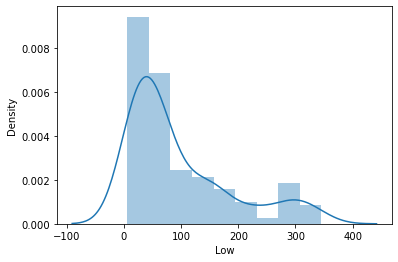


The above distribution of High Stock Price is a positively right-skewed distribution. this is also not a perfect normal distribution again here to apply log transformation

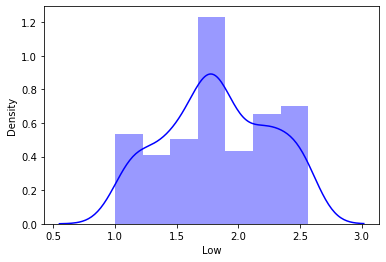


After log transformation, this distribution is also not a perfectly normal distribution but more or less its looking normal distribution.

**Visualize the distribution of low stock price**

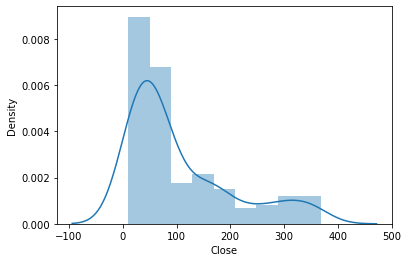


The above distribution of Low Stock Price is a positively right-skewed distribution. this is also not a perfect normal distribution again here to apply log transformation

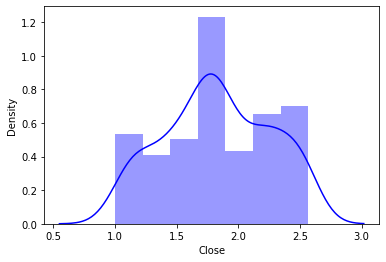


After log transformation, this distribution is also not a perfectly normal distribution but more or less its looking normal distribution.

**Visualize the distribution of Close stock price**



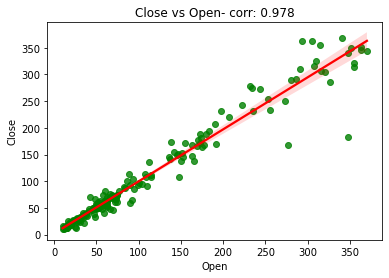
The above distribution of Low Stock Price is a positively right-skewed distribution. this is also not a perfect normal distribution again here to apply log transformation



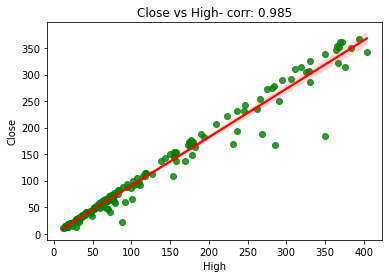
After log transformation, this distribution is also not a perfectly normal distribution but more or less its looking normal distribution

**4.3) Univariate Analysis**

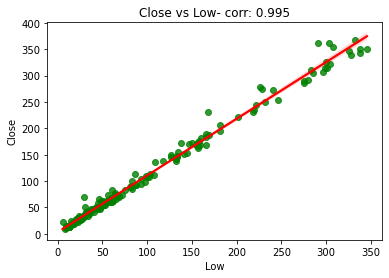
**Close Price vs Open Price**



**Close Price vs High Price**



**Close Price vs Low Price**



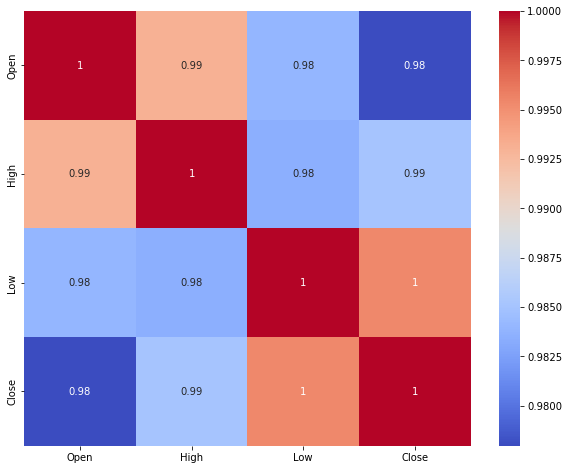
From the above plots, we can conclude that the columns 'Open', 'High', and 'Low' these features are linear relations and high correlations between each independent and dependent variable. that means there is a strong correlation between all the independent variables.

**4.4) Correlation Analysis**

The correlation analysis has been done to get a better understanding of dependent and independent variables’ multicollinearity. Multicollinearity may not affect the accuracy of the model as much but we might lose reliability in determining the effects of individual independent features on the dependent feature in your model and that can be a problem when we want to interpret your model.

**Heatmap**

Let’s check the heatmap plotted concerning independent variables.



* The heatmap shows some high correlations between variables and also us visualize the correlation of each parameter will respect to every other parameter.
* the heatmap, we can see that Every feature is extremely correlated with each other. This means there is high multicollinearity between each independent column.
* High multicollinearity is not good for fitting the model and prediction because a slight change in any independent variable will give very unpredictable results.
* We have measured VIF scores in our dataset which means there is high multicollinearity between these variables., we have dropped one of them. then we fitted the model and make a prediction.

**VIF (Variance Inflation Factor)**

Variance Inflation Factor (VIF) is used to detect the presence of multicollinearity. Variance inflation factors (VIF) measure how much the variance of the estimated regression coefficients is inflated as compared to when the predictor variables are not linearly related. It is obtained by regressing each independent variable.

After measuring the VIF in our dataset which means there is high multicollinearity between these variables., we have dropped one of them. then we fitted the model and make a prediction.

**5) Feature Engineering**

The provided data in its raw form wasn’t directly used as an input to the model. Several feature engineering was carried out where few features were modified, few were dropped, and few were added. Below is a summary of the feature engineering carried out with the provided data set

* In our dataset the Date column which contained the date-time stamp in ‘YYYY-MM-DD’ format was split into individual ‘month’, ‘year’ features.
* Drop date column: Intuitively, there should be no dependency on the date. Hence drop this column
* One Hot Encoding of Month and Year features. And months and year features convert into dummies variable.

**6) Normalization**

The univariate analysis of the distribution of features data shows a positive skewness which would have been a problem while predicting the values on the test data set. So to ensure the minimization of errors we have taken the square root of the rented bike count data which tends the data for equal weightage. The need for normalization is basically for making sure that a table contains only data directly related to the primary key, that each data field contains only one item of data, and that redundant (duplicated and unnecessary) data is eliminated

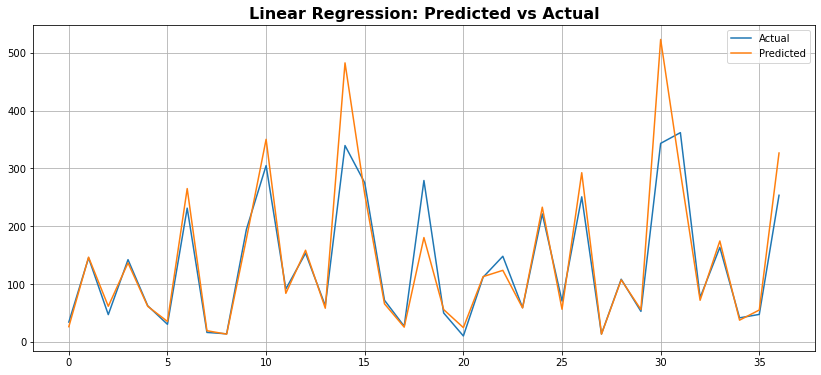
**7) Building Machine Learning Algorithm**

The provided data is first cleaned and transformed using Feature Engineering. We then split the data into the Train set (for Hyperparameter tuning) and Test set (for Model Evaluation). Using MSE as our evaluation metric, we compare various models and select the regression algorithm based on the lowest MSE on the Test data.

**7.1) Train/Test Split**

The train/test split was done as 0.20 % on data with a random state of 0. The final dataset was of shape (185, 33) which was split to (148 17) as Train data and (37, 17) as Test data. To normalize the data after the split, using the Standard Scalar then module will give equal weightage to all the parameters to retain data from one-way deviation.

**7.2) Linear Regression**



In the Linear Regression Model accuracy is moderate for training as well as test data. Therefore, we can conclude that no overfitting.

Our Linear Regression Model predicted the close price with a 0.064% Mean Absolute error.

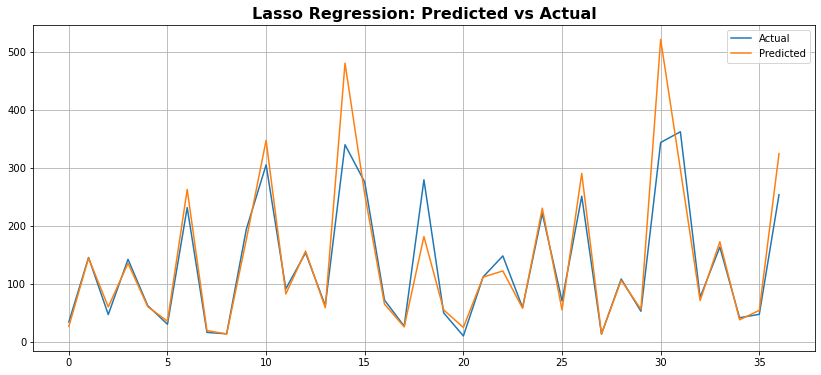
Our model Has a training accuracy of 94.58%.

R2 value for both training and test data is moderate indicating that the model is fit well on both the datasets

R2 value shows that our independence variable can describe 94.95% of our dependent variable.

Adj R2 is about 90.43%.

**7.3) Lasso Regression**



In the Lasso Regression Model accuracy is moderate for training as well as test data.

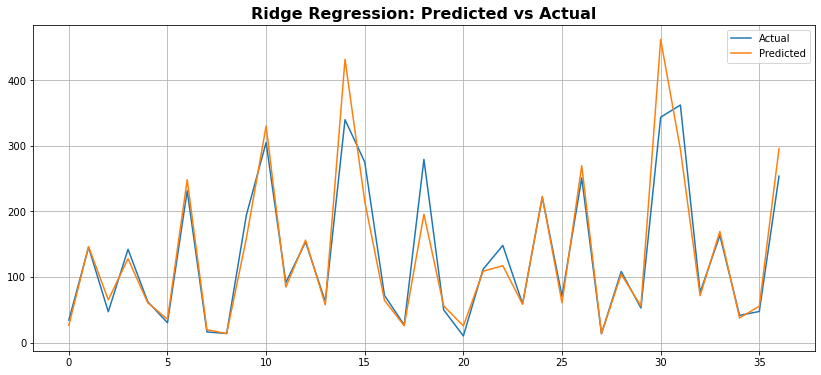
Our lasso Regression Model predicted the close price with a 0.066% Mean Absolute error.

Our model Has a training accuracy of 94.57%.

R2 value for both training and test data is moderate indicating that the model is fit well on both the datasets

Here, R2 is about 94.97% which means the model’s independent features are able to describe our dependent variable and Adj R2 is about 90.48%.

**7.4) Ridge Regression**



In the Ridge Regression Model accuracy is moderate for training as well as test data.

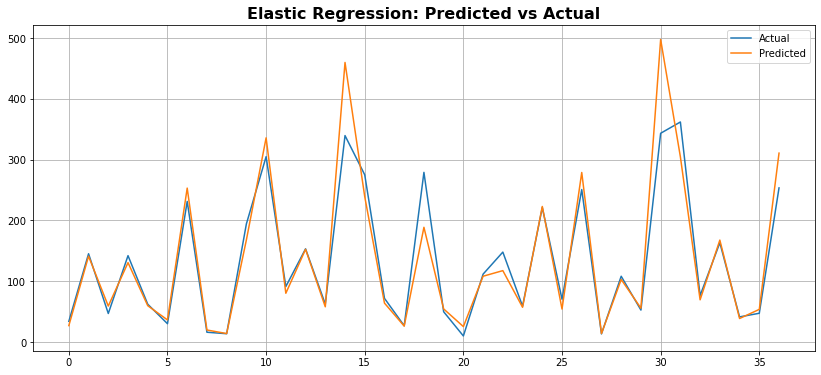
Our Ridge Regression Model predicted the close price with a 0.066% Mean Absolute error.

Our model Has a training accuracy of 94.57%.

R2 value for both training and test data is moderate indicating that the model is fit well on both the datasets

Here, R2 is about 95.25% which means the model’s independent features are able to describe our dependent variable and Adj R2 is about 91.48%.

**7.5) Elastic Net Regression**



In the Elastic Net Regression Model accuracy is moderate for training as well as test data.

Our Elastic Net Regression Model predicted the close price with a 0.063% Mean Absolute error.

Our model Has a training accuracy of 82.31%.

R2 value for both training and test data is moderate indicating that the model is fit well on both the datasets

Here, R2 is about 95.05% which means the model’s independent features are able to describe our dependent variable and Adj R2 is about 90.69%.

**8) Conclusion**

1) first, We started with data inspection, viewed the data distribution

2) In visualization we checked that from 2018 onwards there is a sudden fall in the stock closing price.

3) And Again With the help of a distribution plot we saw that our data is rightly skewed which doesn’t look good in the viewing of the statistical hypothesis. So we applied some kind of transformation Log Transformation to convert it into a normal distribution.

4) Target Variable is strongly dependent on Independent Variables.

5) we have performed VIF to reduce multicollinearity

6) Insights of all the models, A simple linear regression model was built and it was evaluated using accuracy, MSE, RMSE, r2\_score, and Adj\_R2, mean absolute percentage error.

7) Linear Regression, Lasso and Ridge are performing better than Elastic net models with training accuracy of 94.58%, 94.58%, and 94.58% respectively.

8) Apart from Linear Regression, Lasso, and Ridge, Elastic Net is also performing better but has less training accuracy.

9) Ridge and Elastic Net have performed far much better after Cross-validation which is R2 is about 95.25% and 95.09% respectively.

10) R2 and Adjusted R2 are around the range of 95% and 91% in each model.