Financial Risk Analysis

Authored By -

Siddhi Padekar

Problem Statement

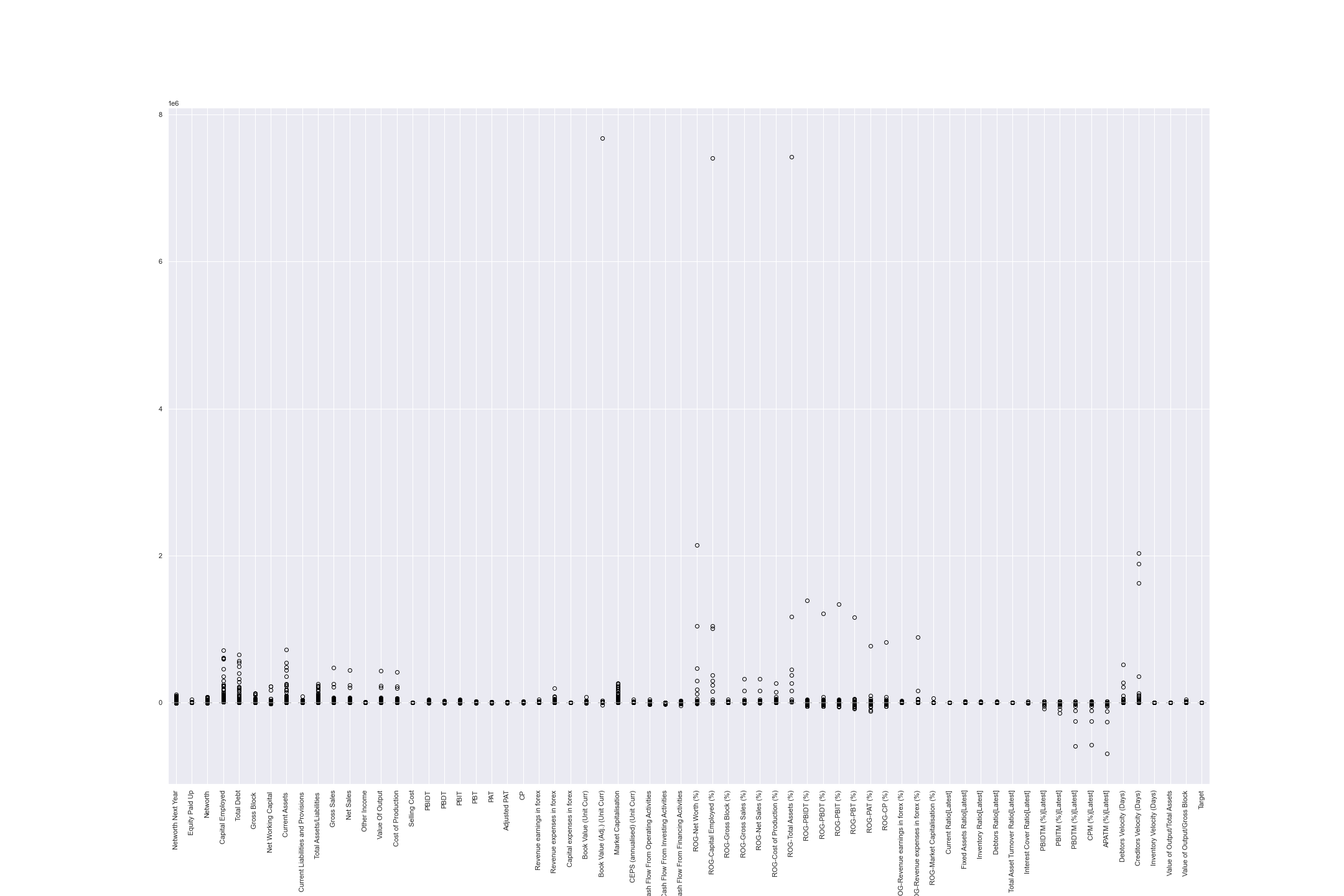
Create an Indian Credit Risk Model

We have been provided with financial data for 3500+ companies and need to evaluate their credit risk performance.

|  |
| --- |
| Particlars |
| This criterion is linked to a Learning Outcome Outlier Treatment |
| This criterion is linked to a Learning Outcome Missing Value Treatment |
| This criterion is linked to a Learning Outcome Check for multicollinearity |
| This criterion is linked to a Learning Outcome Univariate & bivariate analysis |
| This criterion is linked to a Learning Outcome Split the data into train and test (70:30) |
| This criterion is linked to a Learning Outcome Build Logistic Regression Model on most important variables |
| This criterion is linked to a Learning Outcome Predict accuracy of model on train and test datasets |
| Data Split: Split the data into test and train, build classification model CART, Random Forest, Artificial Neural Network |
| This criterion is linked to a Learning Outcome Sort the data in descending order based on probability of default and then divide into 10 dociles based on probability & check how well the model has performed |
| This criterion is linked to a Learning Outcome Build Random Forest Model and predict the accuracy on Train and Test Data |
| This criterion is linked to a Learning Outcome Compare the performance of Logistics Regression and Random Forest |
| This criterion is linked to a Learning Outcome Quality of Business Report |
|  |

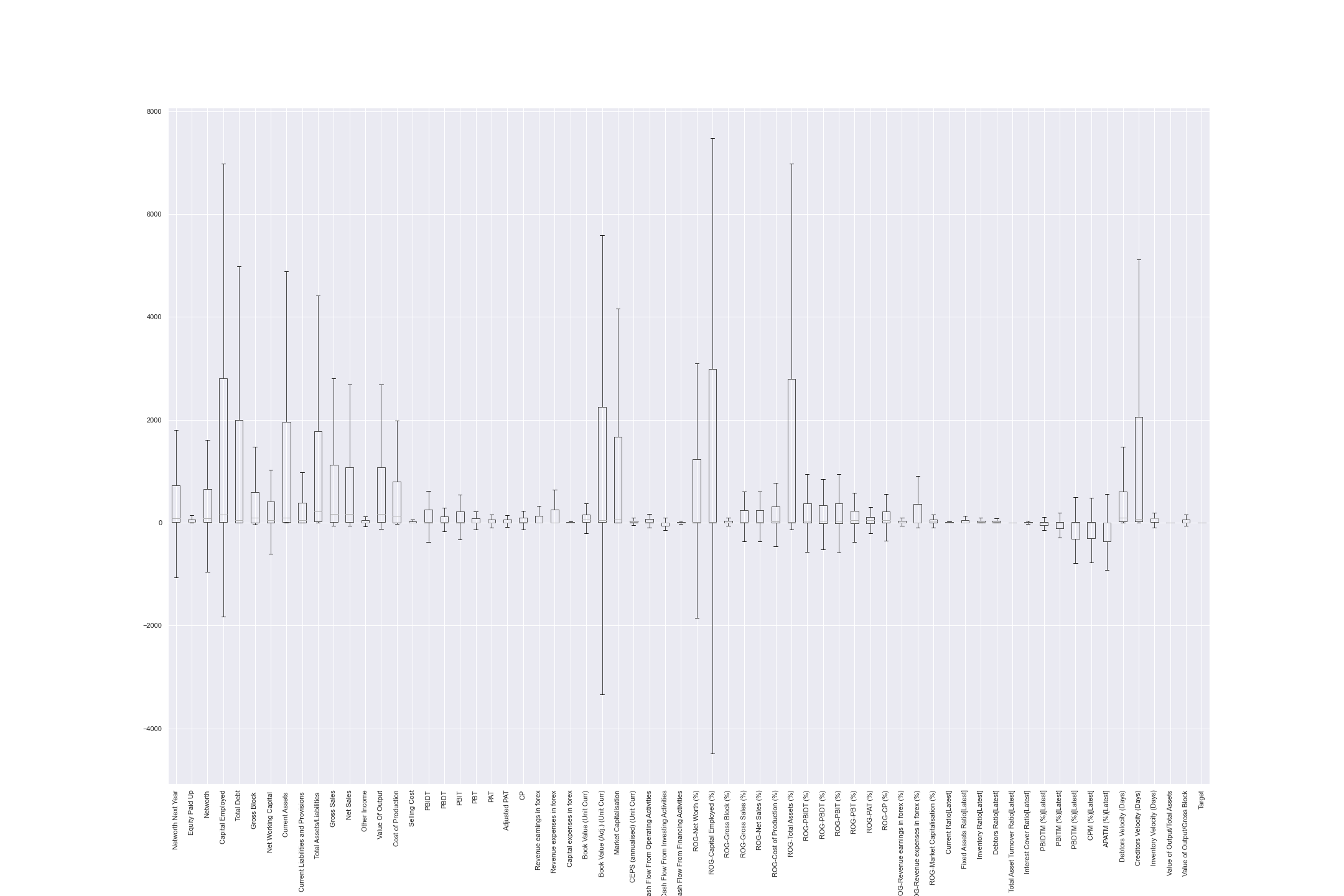
Outlier Treatment

In order to treat the outliers we must first detect them, below we can see that outliers are present in the dataset.



Lets deal with these outliers and later plot the data after outlier removal.

After outlier removal the data looks like this,



For outlier removal, used Inter-Quartile Range (IQR) proximity rule.

The data points which fall below ***Q1 – 1.5 IQR*** or above ***Q3 + 1.5 IQR*** are outliers.

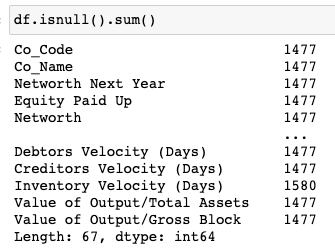
where Q1 and Q3 are the **25th** and **75th percentile** of the dataset respectively, and IQR represents the inter-quartile range and given by Q3 – Q1.

**Filling the missing values**

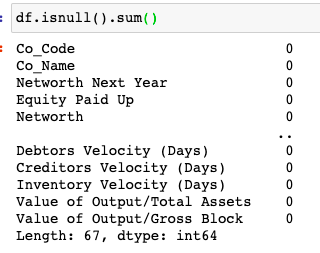
It appears that the data has missing values.

The dataset had roughly around 30% missing values in each column. The Values from Company Code and Company Name serve no purpose. The other column values were filled with the mean value from each column respectively.

Before Filling the missing Values.



After Filling the missing values,



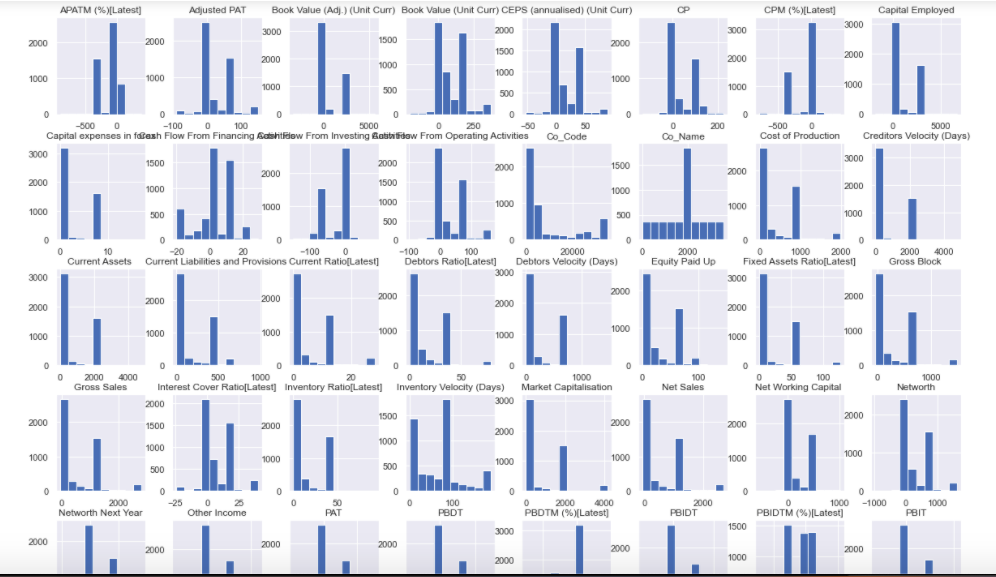
**Creating a Target Variable**

A target variable was created using the values from Next Year Networth.

If The value was negative then new target value will be 1 and if the value was a positive number, then the new target value will be 0.

**Univariate Analysis**

For univariate analysis, plotted histograms for all the columns. The figures shows the distribution of variables.

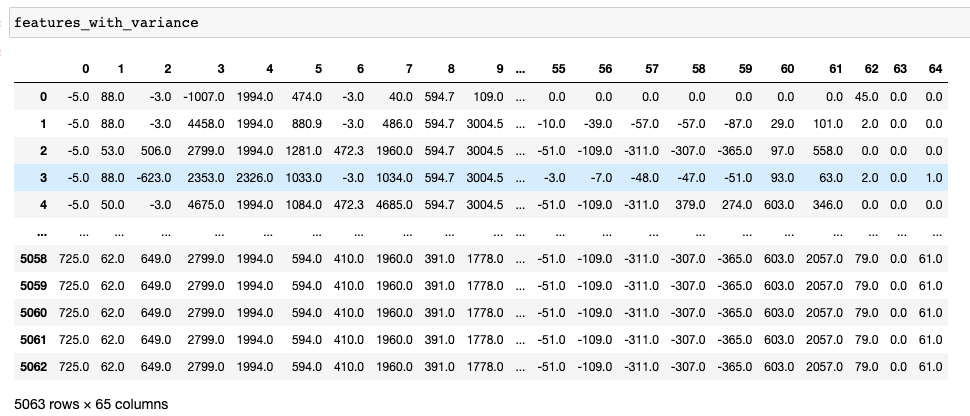


From the figure it is quite clear that some of the numeric columns have a skewed distribution, while others were normally distributed. As a result of this, scaling is needed for those skewed column values to make them normally distributed.

Scaling the data makes it easier for the model to learn and understand the data better.

**Multicollinearity**

Before applying multicollinearity, it is necessary to check If the variance of a feature is zero, then the feature is constant and will not improve the performance of  
the model. In that case, it should be removed.   
  
Sklearn provides with us a class that allows to discard a feature that has a variance below a certain threshold. By default, it discards a feature with zero variance.



But since almost all of our columns are numeric, no feature had a zero variance. Hence all of the features were retained after this step.

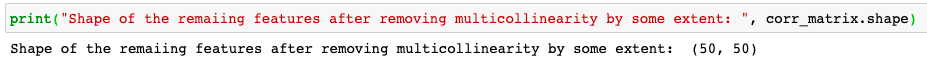
After checking the zero variance, multicollinearity was checked.

There are 3 different methods to check multicollinearity

1. Multicollinearity Analaysis using either Condition Index (CI) or Variance Inflation Factor (VIF).
2. Clustering Analysis of the features (not the data samples)
3. Principal Component Analysis (PCA)

However the first method of using Condition Index (CI) or Variance Inflation Factor (VIF) was followed.

The size of the data was significantly reduced after applying multicollinearity. The initial data had 65+ features. But now the count was brought down to 50 features.



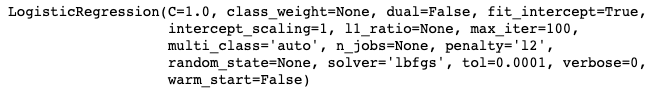
**Modelling**

The final step before modelling included splitting the data into train and test with ratio of 70 and 30 respectively.

Used Two different algorithms for modelling

# **Logistic-Regression**

A very simple Logistic Regression classifier was implemented. The sklearn.linear\_model library is used to import the LogisticRegression class. A classifier object of that class was created and fitted with the X\_Train and Y\_Train variables.



Accuracy score, Auc-Roc curve along with the confusion matrix was implemented to test the prediction accuracy of the classifier.

The model had an accuracy of around 96% and an AucRoc score of 97 %.



**Random Forest**

Random forest classifier was implemented as a second algorithm. The initial model was only given 10 decision trees, which resulted in incorrect predictions. Later changed the number to 50. Once the model was fitted with more the decision trees the number of incorrect predictions grew less.

Similar to the evaluation of logistic regression, Accuracy score, Auc-Roc curve along with the confusion matrix was implemented to test the prediction accuracy of the classifier.

The model had an accuracy and an AucRoc score of 100 %. Might be because of the complexity it serves, the model was able to give 100% accuracy.

Also 100% accuracy, leads to assumption that the data has duplicates in train and test splits. Removing the duplicates can resolve this problem.

