**Project: FRA Project (Milestone-1)**

Pavan Kalyan B | PGP DSBA Oct\_C’21 | AUG 2022

**Table of contents**

* 1. Outlier Treatment
  2. Missing Value Treatment
  3. Transform Target variable into 0 and 1
  4. Univariate & Bivariate analysis with proper interpretation. (You may choose to include only those variables which were significant in the model building)
  5. Train Test Split
  6. Build Logistic Regression Model (using stats model library) on most important variables on Train Dataset and choose the optimum cut off. Also showcase your model building approach
  7. Validate the Model on Test Dataset and state the performance matrices. Also state interpretation from the model

**Problem Statement:**

Businesses or companies can fall prey to default if they are not able to keep up their debt obligations. Defaults will lead to a lower credit rating for the company which in turn reduces its chances of getting credit in the future and may have to pay higher interests on existing debts as well as any new obligations. From an investor's point of view, he would want to invest in a company if it is capable of handling its financial obligations, can grow quickly, and is able to manage the growth scale.

A balance sheet is a financial statement of a company that provides a snapshot of what a company owns, owes, and the amount invested by the shareholders. Thus, it is an important tool that helps evaluate the performance of a business.

Data that is available includes information from the financial statement of the companies for the previous year (2015). Also, information about the Net worth of the company in the following year (2016) is provided which can be used to drive the labelled field.

Explanation of data fields available in Data Dictionary, 'Credit Default Data Dictionary.xlsx'

**Hints** :

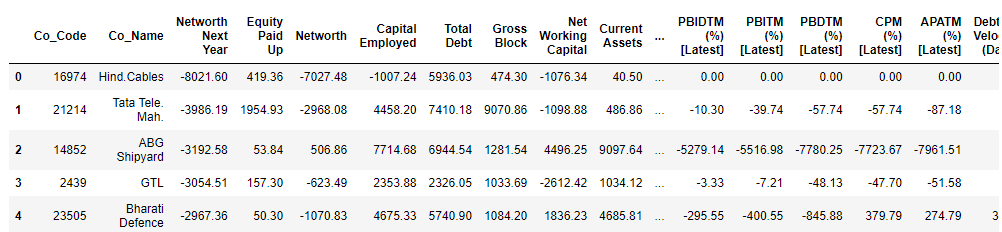
Dependent variable - We need to create a default variable that should take the value of 1 when net worth next year is negative & 0 when net worth next year is positive.

Test Train Split -   Split the data into Train and Test dataset in a ratio of 67:33 and use random\_state =42. Model Building is to be done on Train Dataset and Model Validation is to be done on Test Dataset.

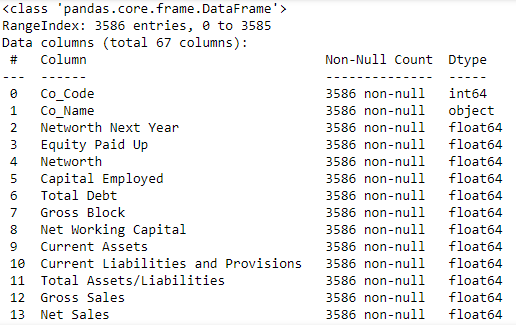
[Credit Risk Dataset](https://olympus.mygreatlearning.com/courses/63586/files/6377907/download?verifier=XyNlomxGTEPmerf1B1FD1w607W981DNFERZaxPKm&wrap=1)

[Data Dictionary](https://olympus.mygreatlearning.com/courses/63586/files/6377910/download?verifier=Jn38BZKwfUDwtZEWsHnY534HOpJCAaObhVR1pVGP&wrap=1)

**Reading the data and exploring.**

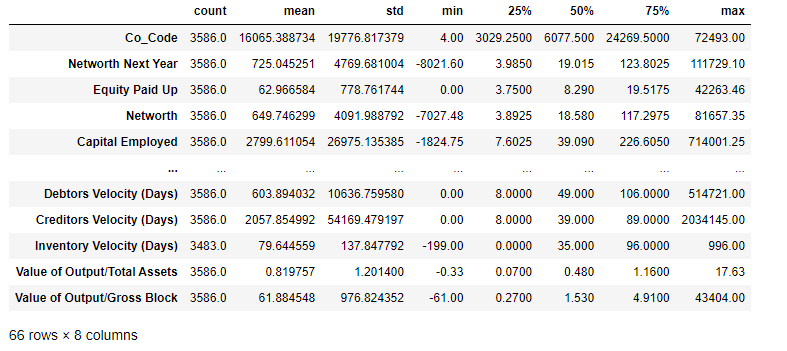
****

**Table 0.1.1 First five rows in the data**

****

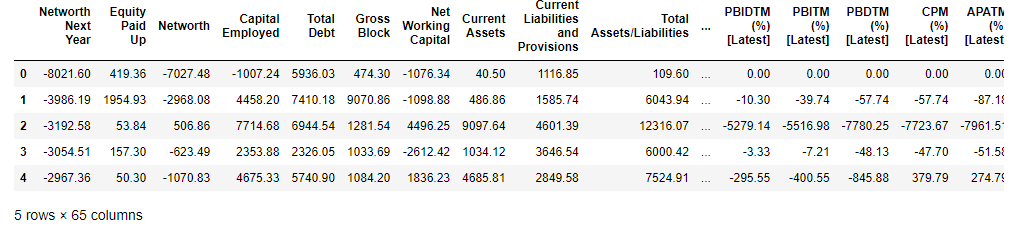
**Table 0.1.2 Data Info**

* We can observe that there are 67 columns and 3586 rows in the data
* It uses memory of 1.8+MB
* We have 63 - float64, 3 - int64, 1 - object data types

****

**Table 0.1.3 Summary of the data**

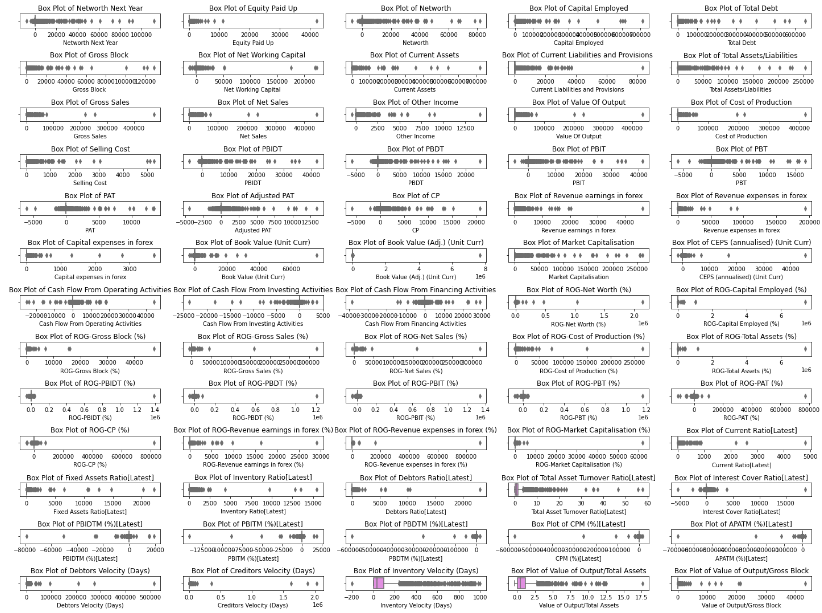
* We can observe that two columns need to be remove which doesn’t make any sense
* Co\_code and co\_name are the one need to be removed

****

**Table 0.1.4 First rows after removing 2 columns**

* We can observe that 65 columns and rest rows are present in the data
* There are no duplicates in the data
  1. **Outlier Treatment**

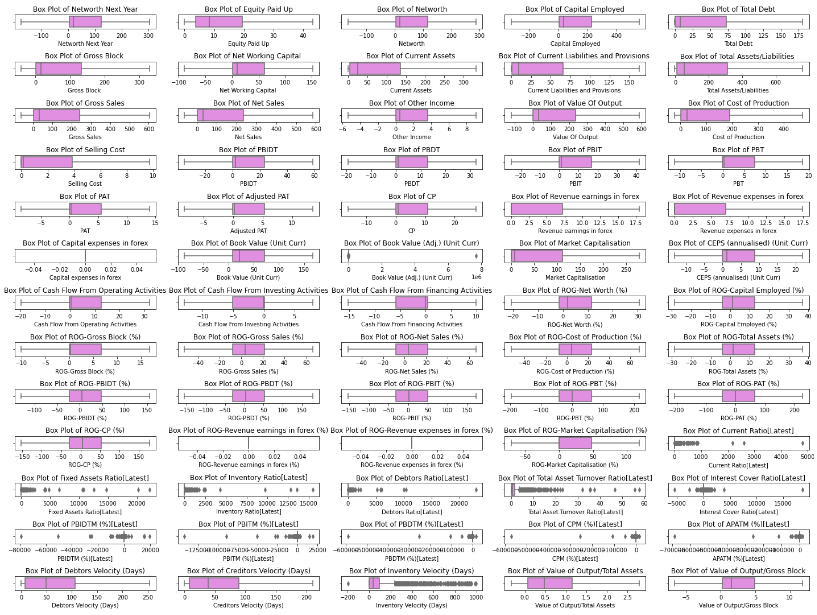
**Before Treating Outliers:**

****

**Fig 1.1.1 Boxplot showing outliers**

* We can observe that most of the columns has outliers in the data which needs to be treated

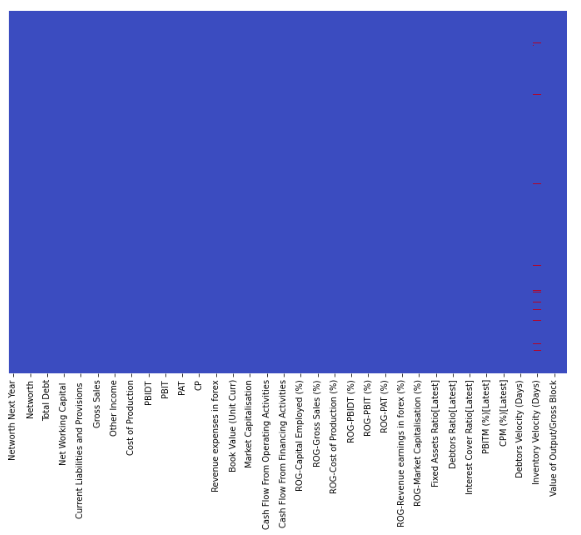
**After Treating Outliers:**

****

**Fig 1.1.2 Boxplot showing no outliers**

* We can observe that there no outliers after the treatment except in few cases
  1. **Missing Value Treatment**

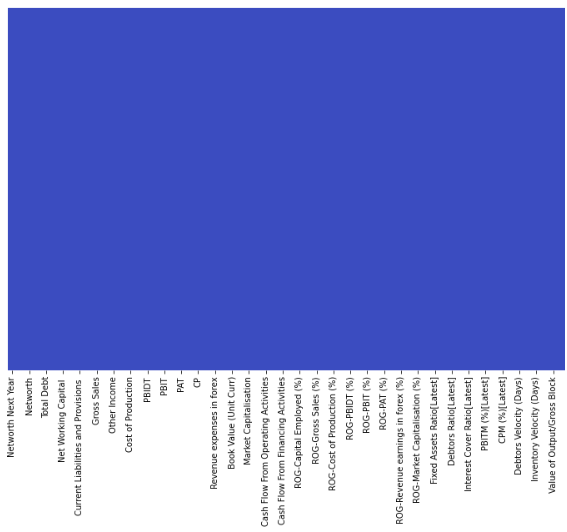
**Before Treating Missing values:**



**Fig 1.2.1 Missing values in the data**

* We can observe that there are missing values in the data in few of the columns
* More number of missing values are identified in the inventory velocity (Days) column and less missing values in rest columns

**After Treating Missing Values:**

****

**Fig 1.2.2 Data without missing values**

* We have replaced the null values with the knn imputation
  1. **Transform Target variable into 0 and 1**

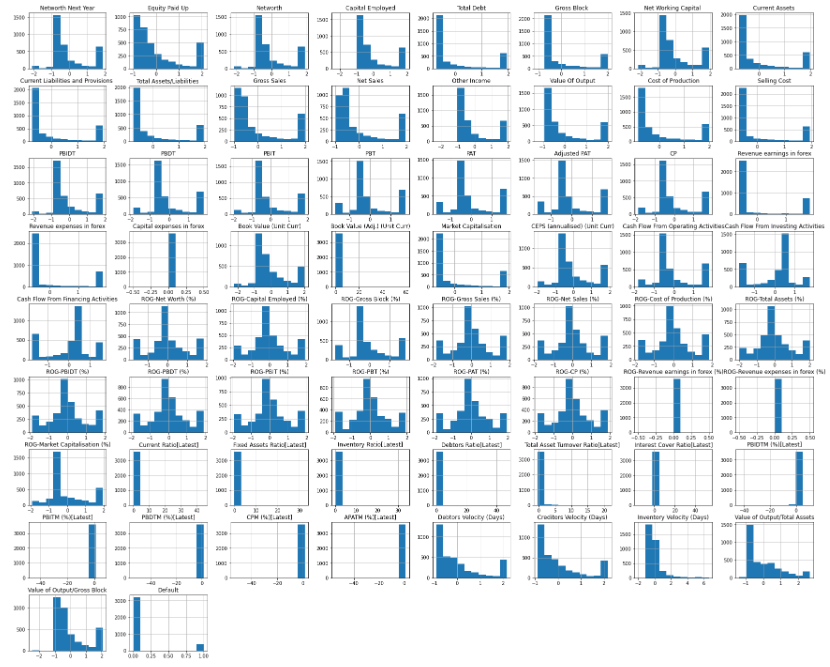
****

* There are 3198 positive values and 388 negative values in the data
* We make positive values to be fixed as 0 and negative values to be fixed as 1

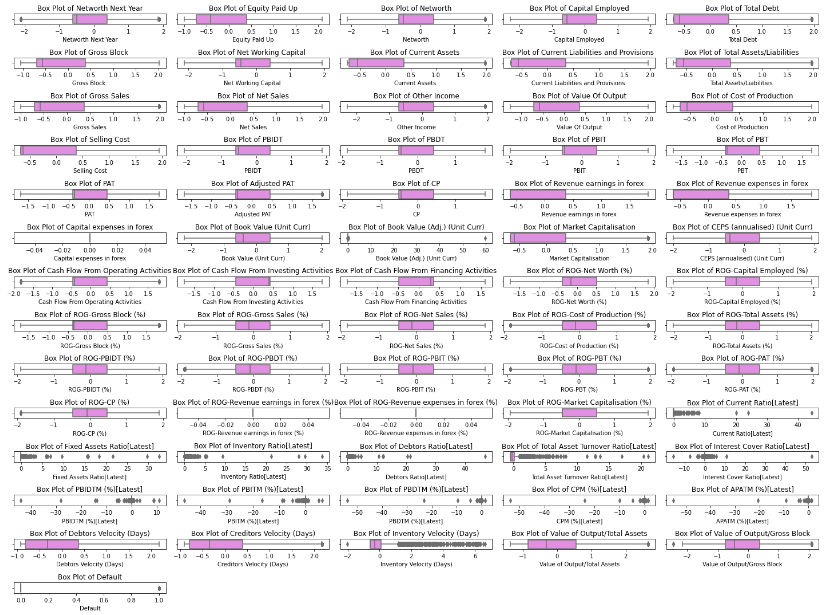


* 1. **Univariate & Bivariate analysis with proper interpretation. (You may choose to include only those variables which were significant in the model building)**

**Univariate Analysis:**

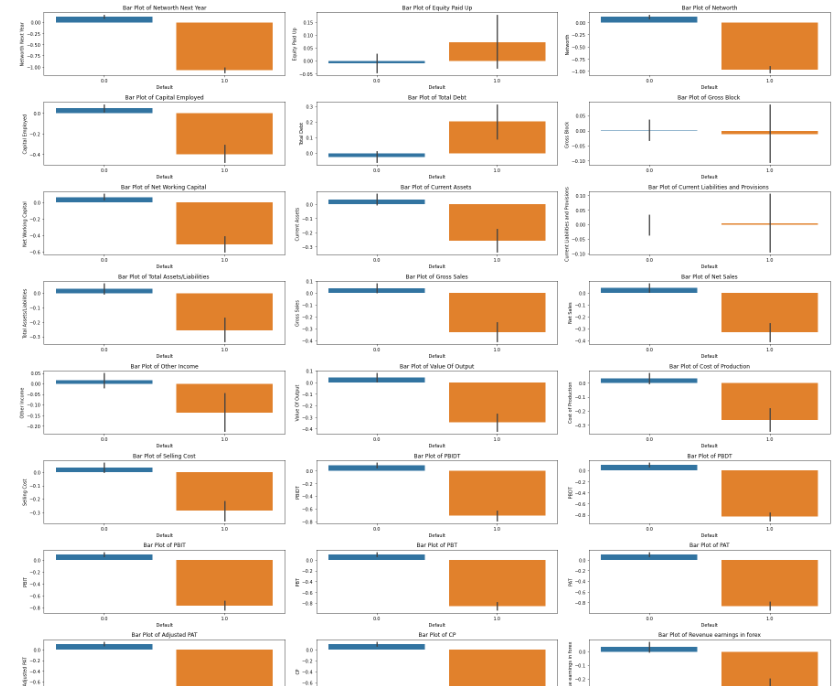
****

**Fig 1.4.1 Histogram for all columns**

****

**Fig 1.4.2 Box plot of all columns**

**Bivariate Analysis:**

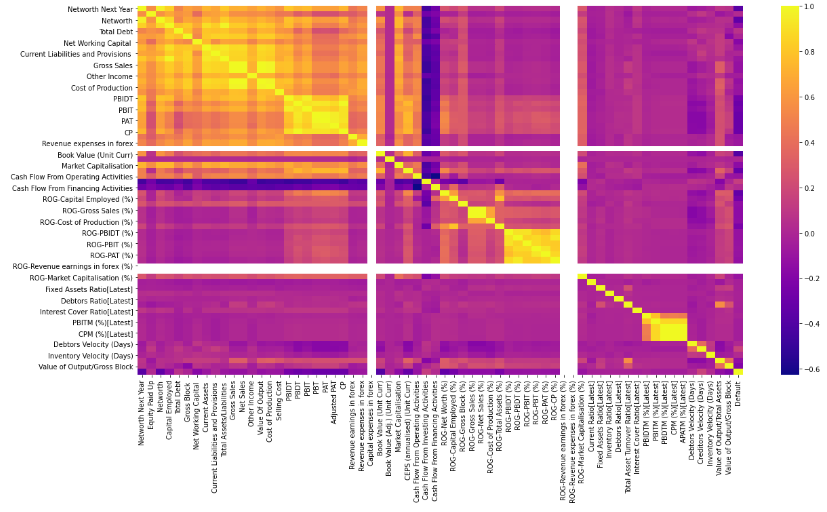
****

**Fig 1.4.3 Box plot shown for few columns**

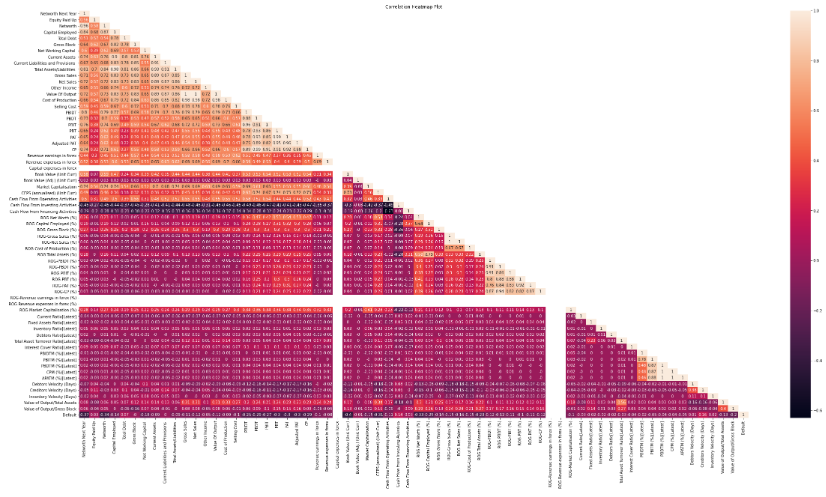
* Created bar plots for few columns based on the new column “Default”

**Multivariate Analysis:**

**Correlation plot:**



**Fig 1.4.4 Heat map**

****

**Fig 1.4.5 Triangular Heat map**

* 1. **Train Test Split**

****

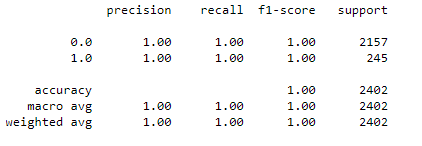
* 1. **Build Logistic Regression Model (using statsmodel library) on most important variables on Train Dataset and choose the optimum cutoff. Also showcase your model building approach**

**&**

* 1. **Validate the Model on Test Dataset and state the performance matrices. Also state interpretation from the model**

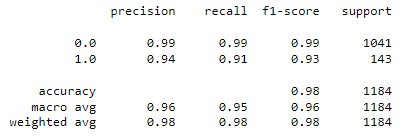
**Logistic Regression - Model 1**

**Classification metrics on training dataset:**

****

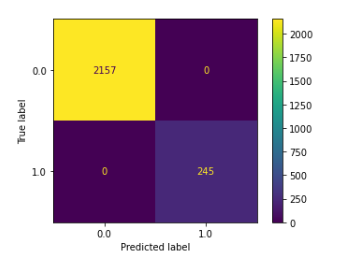
**Table 1.6.1 Training data accuracy**

**Classification metrics on testing dataset:**

****

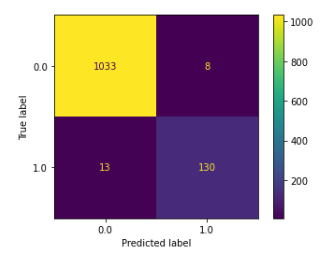
**Table 1.7.1 Testing data accuracy**

**Confusion matrix on training dataset:**

****

**Fig 1.6.1 confusion matrix on training**

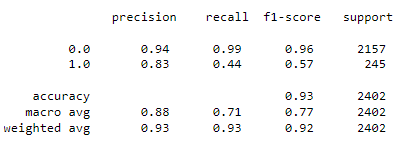
**Confusion matrix on testing dataset:**

****

**Fig 1.7.1 confusion matrix on testing**

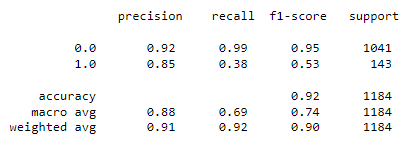
**Linear Discriminant Analysis - Model 2**

**Classification metrics on training data set:**

****

**Table 1.6.2 Training data accuracy**

**Classification metrics on testing dataset:**

****

**Table 1.7.2 Testing data accuracy**

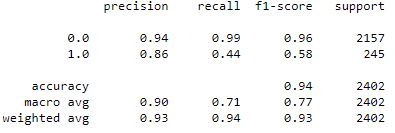
**Confusion matrix on training and testing dataset:**

****

**Fig 1.6.2 and 1.7.2 Confusion matrix on Training and Testing dataset**

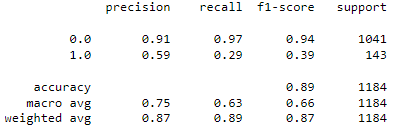
**K-Nearest Neighbor - Model 3**

**Classification metrics on training data set:**

****

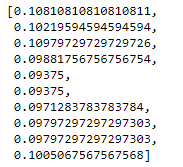
**Table 1.6.3 Training data accuracy**

**Classification metrics on testing data set:**

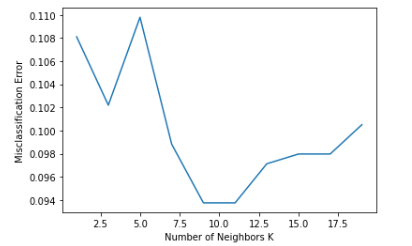
****

**Table 1.7.3 Testing data accuracy**

**Misclassification error:**

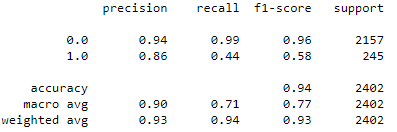
****

**Table 1.6.4 Misclassification error**

****

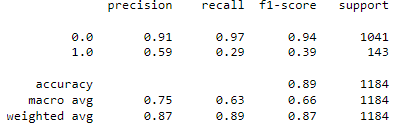
**Fig 1.6.3 Misclassification error graph**

**Classification metrics on training data set with k-value 5:**

****

**Table 1.6.5 Training data accuracy**

**Classification metrics on testing data set with k-value 5:**

****

**Table 1.7.4 Testing data accuracy**

**Conclusions:**

**Based on Model:**

* Based on the Three models comparing the accuracies on the test data set, we can observe that logistic regression performs more better than the LDA and KNN algorithm models
* As the logistic regression gives 100% accuracy on training and 98% on testing part
* We can take model 1 logistic regression as the best model out of these

**Based on Dataset:**

* The dataset was huge and hence we have cleaned and pre-processed it to a neat way
* The outliers were treated and missing values also treated
* Univariate and bivariate analysis are done to give a check
* We used logistic regression model and had a threshold value to predict the output.

**===========================================================================**