14 May 2022

FRA Milestone-1 Project

PGP-DSBA June-Batch

SOOREJ KUNNOOL BALAN

2022

Table of Contents

[1. Problem 3](#_Toc103540780)

[1.1 Outlier treatment 3](#_Toc103540781)

[1.2 Missing value treatment. 8](#_Toc103540782)

[1.3 Transform target variable into 0 and 1. 10](#_Toc103540783)

[1.4 Univariate (4 marks) & Bivariate ( 6marks) analysis with proper interpretation. (You may choose to include only those variables which were significant in the model building). 11](#_Toc103540784)

[1.5 Train Test Split. 25](#_Toc103540785)

[1.6 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? 25](#_Toc103540786)

[1.7 Validate the Model on Test Dataset and state the performance matrices. Also state interpretation from the model. 34](#_Toc103540787)

[THE END 39](#_Toc103540788)

List of Tables List of Figures

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table-1.1 Head of input data | 3 |  | Figure-1.1 Boxplot | | 6 |
| Table-1.2 Tail of input data | 4 |  | Figure-1.2 Boxplot of Equity\_Paid\_Up | | 7 |
| Table-1.3 Renamed columns | 4 |  | Figure-1.3 Boxplot of Networth/Capital\_Employed | | 7 |
| Table-1.4 Concise description | 5 |  | Figure-1.4 Heatmap – Missing data | | 8 |
| Table-1.5 Sum of outliers in columns | 6 |  | Figure-1.5 EDA – Central measures of tendency | | 11 |
| Table-1.6 Descriptive Statistics | 7 |  | Figure-1.6 Histogram/Boxplot- Networth\_Next\_Year | | 12 |
| Table-1.7 Missing data per observation | 9 |  | Figure-1.7 Histogram/Boxplot- Equity\_Paid\_Up | | 12 |
| Table-1.8 Missing data per feature | 9 |  | Figure-1.8 Histogram/Boxplot- Networth | | 13 |
| Table-1.9 Imputed data- no missing value per feature | 10 |  | Figure-1.9 Histogram/Boxplot- Capital\_Employed | | 13 |
| Table-1.10 Target variable | 10 |  | Figure-1.10 Histogram/Boxplot- Total\_Debt | | 14 |
| Table-1.11 VIF - Feature | 26 |  | Figure-1.11 Histogram/Boxplot- Gross\_Block\_ | | 14 |
| Table-1.12 Model 1 - Summary | 27 |  | Figure-1.12 Histogram/Boxplot- Net\_Working\_Capital | | 15 |
| Table-1.13 Model 1 – Confusion Matrix | 28 |  | Figure-1.13 Histogram/Boxplot- Current\_Assets\_ | | 15 |
| Table-1.14 Model 1 – Classification Report | 28 |  | Figure-1.14 Histogram/Boxplot- Current\_Liabilities\_and\_Provisions\_ | | 16 |
| Table-1.15 Model 2 - Summary | 29 |  | Figure-1.15 Histogram/Boxplot- Total\_Assets\_To\_Liabilities\_ | | 16 |
| Table-1.16 Model 2 – Confusion Matrix-Train | 30 |  | Figure-1.16 Histogram/Boxplot- Gross\_Sales | | 17 |
| Table-1.17 Model 2 – Classification Report-Train | 30 |  | Figure-1.17 Histogram/Boxplot- Net\_Sales | | 17 |
| Table-1.18 Model 3 - Summary | 31 |  | Figure-1.18 Histogram/Boxplot- Other\_Income | | 18 |
| Table-1.19 Model 3 – Confusion Matrix-Train | 32 |  | Figure-1.19 Histogram/Boxplot- Value\_of\_Output | | 18 |
| Table-1.20 Model 3 – Classification Report-Train | 32 |  | Figure-1.20 Histogram/Boxplot- Cost\_Of\_Production | | 19 |
| Table-1.21 Model 4 - Summary | 33 |  | Figure-1.21 Histogram/Boxplot- Selling\_Cost | | 19 |
| Table-1.22 Model 4 – Confusion Matrix-Train | 34 |  | Figure-1.22 Histogram/Boxplot- Revenue\_earnings\_in\_forex | | 20 |
| Table-1.23 Model 4 – Classification Report-Train | 34 |  | Figure-1.23 Histogram/Boxplot- Revenue\_expenses\_in\_forex | | 20 |
| Table-1.24 Model 1 – Confusion Matrix-Test | 34 |  | Figure-1.24 Histogram/Boxplot- Capital\_expenses\_in\_forex | | 21 |
| Table-1.25 Model 1 – Classification Report-Test | 35 |  | Figure-1.25 Histogram/Boxplot- Book\_Value\_Unit\_Curr | | 21 |
| Table-1.26 Model 2 – Confusion Matrix-Test | 35 |  | Figure-1.26 Correlation heatmap | 22 | |
| Table-1.27 Model 2 – Classification Report-Test | 36 |  | Figure-1.27 Barplot: Networth vs defaulter | 23 | |
| Table-1.28 Model 3 – Confusion Matrix-Test | 36 |  | Figure-1.28 Barplot: Gross\_Sales vs defaulter | 23 | |
| Table-1.29 Model 3 – Classification Report-Test | 37 |  | Figure-1.29 Barplot: Total\_Assets\_to\_Liabilities vs defaulter | 23 | |
| Table-1.30 Model 4 – Confusion Matrix-Test | 37 |  | Figure-1.30 Barplot: Total\_Debt vs defaulter | 24 | |
| Table-1.31 Model 4 – Classification Report-Test | 38 |  | Figure-1.31 Scatterplot: Total\_Debt vs Current\_Assets | 24 | |
|  |  |  | Figure-1.32 Scatterplot: Revenue\_earnings\_in\_forex vs Revenue\_expenses\_in\_forex | 24 | |
|  |  |  | Figure-1.33 Heatmap- Collinaerity | 25 | |
|  |  |  | Figure-1.34 Model 1 –ROC\_AUC score/ROC curve | 35 | |
|  |  |  | Figure-1.35 Model 2 –ROC\_AUC score/ROC curve | 36 | |
|  |  |  | Figure-1.36 Model 3 –ROC\_AUC score/ROC curve | 37 | |
|  |  |  | Figure-1.37 Model 4 –ROC\_AUC score/ROC curve | 38 | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  | |

# Problem

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 labeled 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.

 

## Outlier treatment

We will be performing the model building using python.

First let us load the data and check the head & tail of the data:

Graphical user interface, text, application, email

Description automatically generated

Table-1.1 Head of input data

Graphical user interface, text, application, email

Description automatically generated

Table-1.2 Tail of input data

We can see that the column names have characters such as %,[,] etc. Let us rename the column names to have only “\_” as the special character in them and no spaces.

Graphical user interface, text, application

Description automatically generated

Table-1.3 Renamed columns

We can see that the data has 3586 rows/observations and 67 columns/variables.

Let us look at a concise description of the data:

|  |
| --- |
| RangeIndex: 3586 entries, 0 to 3585  Data columns (total 67 columns):  # Column Non-Null Count Dtype  --- ------ -------------- -----  0 Co\_Code 3586 non-null int64  1 Co\_Name 3586 non-null object  2 Networth\_Next\_Year 3586 non-null float64  3 Equity\_Paid\_Up 3586 non-null float64  4 Networth 3586 non-null float64  5 Capital\_Employed 3586 non-null float64  6 Total\_Debt 3586 non-null float64  7 Gross\_Block\_ 3586 non-null float64  8 Net\_Working\_Capital\_ 3586 non-null float64  9 Current\_Assets\_ 3586 non-null float64  10 Current\_Liabilities\_and\_Provisions\_ 3586 non-null float64  11 Total\_Assets\_to\_Liabilities\_ 3586 non-null float64  12 Gross\_Sales 3586 non-null float64  13 Net\_Sales 3586 non-null float64  14 Other\_Income 3586 non-null float64  15 Value\_Of\_Output 3586 non-null float64  16 Cost\_of\_Production 3586 non-null float64  17 Selling\_Cost 3586 non-null float64  18 PBIDT 3586 non-null float64  19 PBDT 3586 non-null float64  20 PBIT 3586 non-null float64  21 PBT 3586 non-null float64  22 PAT 3586 non-null float64  23 Adjusted\_PAT 3586 non-null float64  24 CP 3586 non-null float64  25 Revenue\_earnings\_in\_forex 3586 non-null float64  26 Revenue\_expenses\_in\_forex 3586 non-null float64  27 Capital\_expenses\_in\_forex 3586 non-null float64  28 Book\_Value\_Unit\_Curr 3586 non-null float64  29 Book\_Value\_Adj\_Unit\_Curr 3582 non-null float64  30 Market\_Capitalisation 3586 non-null float64  31 CEPS\_annualised\_Unit\_Curr 3586 non-null float64  32 Cash\_Flow\_From\_Operating\_Activities 3586 non-null float64  33 Cash\_Flow\_From\_Investing\_Activities 3586 non-null float64  34 Cash\_Flow\_From\_Financing\_Activities 3586 non-null float64  35 ROG\_Net\_Worth\_perc 3586 non-null float64  36 ROG\_Capital\_Employed\_perc 3586 non-null float64  37 ROG\_Gross\_Block\_perc 3586 non-null float64  38 ROG\_Gross\_Sales\_perc 3586 non-null float64  39 ROG\_Net\_Sales\_perc 3586 non-null float64  40 ROG\_Cost\_of\_Production\_perc 3586 non-null float64  41 ROG\_Total\_Assets\_perc 3586 non-null float64  42 ROG\_PBIDT\_perc 3586 non-null float64  43 ROG\_PBDT\_perc 3586 non-null float64  44 ROG\_PBIT\_perc 3586 non-null float64  45 ROG\_PBT\_perc 3586 non-null float64  46 ROG\_PAT\_perc 3586 non-null float64  47 ROG\_CP\_perc 3586 non-null float64  48 ROG\_Revenue\_earnings\_in\_forex\_perc 3586 non-null float64  49 ROG\_Revenue\_expenses\_in\_forex\_perc 3586 non-null float64  50 ROG\_Market\_Capitalisation\_perc 3586 non-null float64  51 Current\_RatioLatest 3585 non-null float64  52 Fixed\_Assets\_RatioLatest 3585 non-null float64  53 Inventory\_RatioLatest 3585 non-null float64  54 Debtors\_RatioLatest 3585 non-null float64  55 Total\_Asset\_Turnover\_RatioLatest 3585 non-null float64  56 Interest\_Cover\_RatioLatest 3585 non-null float64  57 PBIDTM\_percLatest 3585 non-null float64  58 PBITM\_percLatest 3585 non-null float64  59 PBDTM\_percLatest 3585 non-null float64  60 CPM\_percLatest 3585 non-null float64  61 APATM\_percLatest 3585 non-null float64  62 Debtors\_Velocity\_Days 3586 non-null int64  63 Creditors\_Velocity\_Days 3586 non-null int64  64 Inventory\_Velocity\_Days 3483 non-null float64  65 Value\_of\_Output\_to\_Total\_Assets 3586 non-null float64  66 Value\_of\_Output\_to\_Gross\_Block 3586 non-null float64  dtypes: float64(63), int64(3), object(1)  memory usage: 1.8+ MB |

Table-1.4 Concise description

From table 1.4 we can see that we have missing data in some of the columns.

We can see Networth\_Next\_Year column has no missing values, hence let us create “default” column using condition, default 1, when Networth\_Next\_Year < 0 else 1 (more on section 1.3).

Let us look at proportion of default variable:

We have 3198 non-defaulters and 388 defaulters, which is 89 and 11% respectively.

Let us look at the boxplot of all variables visually.

Chart, bar chart, histogram

Description automatically generated

Figure-1.1 Boxplot

We can see outliers across many columns.

For each column let us determine the Q1 (25th quantile),Q3(75th quantile), IQR(Q3-Q1) values and determine the upper and lower boundary values. Upper boundary is (Q3 + 1.5 \* IQR) and lower boundary is (Q1 – 1.5\*IQR). Any observation w.r.t column, having value greater than upper boundary or lower boundary is an outlier.

Let us look at sum of outliers in few columns:

Text, letter

Description automatically generated

Table-1.5 Sum of outliers in columns

Let us look at the descriptive statistics of some of the columns:

Graphical user interface

Description automatically generated

Table-1.6 Descriptive Statistics

Let us look at the column Equity\_Paid\_Up, which has 75th quantile as 19.52 and maximum at 42263.46.

Chart

Description automatically generated

Figure-1.2 Boxplot of Equity\_Paid\_Up

If we were to treat outliers by limiting all to the quartile upper bound, we will lose the variance in the data and hence affect model’s ability to predict for data, that might come with outliers.

This is the scenario for other columns such as Networth, Capital\_Employed etc.

Chart

Description automatically generated with medium confidence Shape

Description automatically generated

Figure-1.3 Boxplot of Networth/Capital\_Employed

Hence we will substitute all the outlier values with null and determine in null value treatment, the features that we need to retain and use an imputer to replace the null values for the features we need for the model.

## Missing value treatment.

The input dataset had 118 missing data. But after the outlier treatment, where all outlier values, as determined by the columns Interquartile values, has been converted to nulls.

A picture containing diagram

Description automatically generated

Figure-1.4 Heatmap – Missing data

Let us look at missing data per observation.

Table

Description automatically generated

Table-1.7 Missing data per observation

So, we can see that there are occurrences of missing data ranging from 0 to 45.

Let us assume that we need to retain only observations that has at max 5 missing data.

This will lead to a subset of original dataset with 1203 observations against 3586 in input data.

Let us look at the number of defaulters in this new subset, 118.

This means that of the original 388 defaulters, we have 118 in the new subset, which leads to a loss of 70% of the determining data.

Hence the strategy of limiting observations is not a good one.

Let us look at how many missing values we have against feature instead of observation.

Text, letter

Description automatically generated

Table-1.8 Missing data per feature

From the above table we can see that ROG\_Revenue\_expenses\_in\_forex\_perc has 45% of the data as missing. Let us remove features which has more than 30% missing data.

This would mean removing ROG\_Revenue\_expenses\_in\_forex\_perc and ROG\_Revenue\_earnings\_in\_forex\_perc from the dataset.

We will standardize the features using StandardScaler and use KNN imputer to fill the missing values in these features.

After imputing the values, we can see that there are no missing values.

Text, letter

Description automatically generated

Table-1.9 Imputed data- no missing value per feature

## Transform target variable into 0 and 1.

From the problem we know that dependent/target variable should take the value of 1 when net worth next year is negative & 0 when net worth next year is positive.

Hence we will create a new variable “default” which will have value of 0 when Networth\_Next\_Year <= 0 and value of 1 when Networth\_Next\_Year > 0.

Let us look at the default variable:

Table

Description automatically generated Table

Description automatically generated

Table-1.10 Target variable

Let us look at the breakup of default value.

We have 3198 non-defaulters and 388 defaulters. Hence the breakup is 89% and 11%.

11% distribution hints at unbalanced technique and hence while building models we need to check if we should be using SMOTE( Synthetic Minority Oversampling Technique) to oversample the defaulters, so that the model can be trained with sufficiently balanced dataset.

## Univariate (4 marks) & Bivariate ( 6marks) analysis with proper interpretation. (You may choose to include only those variables which were significant in the model building).

The parameters found as significant in the model building are:

* Networth\_Next\_Year
* Equity\_Paid\_Up
* Networth
* Capital\_Employed
* Total\_Debt
* Gross\_Block\_
* Net\_Working\_Capital\_
* Current\_Assets\_
* Current\_Liabilities\_and\_Provisions\_
* Total\_Assets\_to\_Liabilities\_
* Gross\_Sales
* Net\_Sales
* Other\_Income
* Value\_Of\_Output
* Cost\_of\_Production
* Selling\_Cost
* Revenue\_earnings\_in\_forex
* Revenue\_expenses\_in\_forex
* Capital\_expenses\_in\_forex
* Book\_Value\_Unit\_Curr
* Default

**Univariate Analysis:**

Let us first look at the central measures of tendency of all the above variables.

Graphical user interface, application, table, Excel

Description automatically generated

Figure-1.5 EDA – Central measures of tendency

Let us look at the histogram and boxplot of all the above variables.

* Networth\_Next\_Year: Distribution is right skewed, and outliers are present.

Chart, box and whisker chart

Description automatically generated

Figure-1.6 Histogram/Boxplot- Networth\_Next\_Year

* Equity\_Paid\_Up: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated with low confidence

Figure-1.7 Histogram/Boxplot- Equity\_Paid\_Up

* Networth: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.8 Histogram/Boxplot- Networth

* Capital\_Employed: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated with low confidence Figure-1.9 Histogram/Boxplot- Capital\_Employed

* Total\_Debt: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated with medium confidence Figure-1.10 Histogram/Boxplot- Total\_Debt

* Gross\_Block: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.11 Histogram/Boxplot- Gross\_Block\_

* Net\_Working\_Capital: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.12 Histogram/Boxplot- Net\_Working\_Capital

* Current\_Assets\_: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.13 Histogram/Boxplot- Current\_Assets\_

* Current\_Liabilities\_and\_Provisions: Distribution is right skewed, and outliers are present.

Chart, box and whisker chart

Description automatically generated Figure-1.14 Histogram/Boxplot- Current\_Liabilities\_and\_Provisions\_

* Total\_Assets\_to\_Liabilities: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated with low confidence Figure-1.15 Histogram/Boxplot- Total\_Assets\_To\_Liabilities\_

* Gross\_Sales: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.16 Histogram/Boxplot- Gross\_Sales

* Net\_Sales: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.17 Histogram/Boxplot- Net\_Sales

* Other\_Income: Distribution is right skewed, and outliers are present.

Chart, box and whisker chart

Description automatically generated Figure-1.18 Histogram/Boxplot- Other\_Income

* Value\_Of\_Output: Distribution is right skewed, and outliers are present.

Chart, box and whisker chart

Description automatically generated Figure-1.19 Histogram/Boxplot- Value\_of\_Output

* Cost\_Of\_Production: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.20 Histogram/Boxplot- Cost\_Of\_Production

* Selling\_Cost: Distribution is right skewed, and outliers are present.

Chart, box and whisker chart

Description automatically generated Figure-1.21 Histogram/Boxplot- Selling\_Cost

* Revenue\_earnings\_in\_forex: Distribution is right skewed, and outliers are present.

Chart, box and whisker chart

Description automatically generated Figure-1.22 Histogram/Boxplot- Revenue\_earnings\_in\_forex

* Revenue\_expenses\_in\_forex: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated with low confidence Figure-1.23 Histogram/Boxplot- Revenue\_expenses\_in\_forex

* Capital\_expenses\_in\_forex: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.24 Histogram/Boxplot- Capital\_expenses\_in\_forex

* Book\_Value\_Unit\_Curr: Distribution is right skewed, and outliers are present.

Chart

Description automatically generated Figure-1.25 Histogram/Boxplot- Book\_Value\_Unit\_Curr

**Bivariate/Multivariate Analysis:**

* Correlation Heatmap

Chart, bar chart

Description automatically generated

Figure-1.26 Correlation heatmap

Mostly correlation is positive in nature among variables with very few negative correlations.

‘default’ is the target field created based on Networth\_Next\_Year and hence let us not consider this field in this plot.

There is very strong correlation between the following variables:

* Networth & Networth\_Next\_Year
* Capital\_Employed & Total\_Debt/Current\_Assets
* Net\_Sales & Gross\_Sales
* Value\_of\_Output & Net\_Sales/Gross\_Sales
* Revenue\_expenses\_in\_forex & Gross\_Sales/Net\_Sales/Value\_Of\_Output/Cost\_of\_Production

Book\_Value\_Unit\_Curr has very less correlation to other variables.

* Networth vs Defaulter

Created a variable Defaulter corresponding to default column with 0 as ‘No’ and 1 as ‘Yes’.

We can see for Defaulters the average Networth is around -100 whereas for non-defaulters its around 700.

Chart, box and whisker chart

Description automatically generated

Figure-1.27 Barplot: Networth vs defaulter

* Gross\_Sales vs Defaulter

We can see for Defaulters the Average Gross\_Sales is around 150 whereas for non-defaulters its around 1200.

Chart, box and whisker chart

Description automatically generated

Figure-1.28 Barplot: Gross\_Sales vs defaulter

* Total\_Assets\_to\_Liabilities vs Defaulter

We can see for Defaulters the Average Total\_Assets\_to\_Liabilities is around 400 whereas for non-defaulters its around 1900.

Chart, box and whisker chart

Description automatically generated

Figure-1.29 Barplot: Total\_Assets\_to\_Liabilities vs defaulter

* Total\_Debt vs Defaulter

We can see for Defaulters the Average Total\_Debt is around 250 whereas for non-defaulters its around 2100. The value of current debt is not a good indicator for identifying defaulters as non-defaulters has higher debts than defaulters.

Chart, box and whisker chart

Description automatically generated

Figure-1.30 Barplot: Total\_Debt vs defaulter

* Total\_Debt vs Current\_Assets

We can see that most of the datapoints below Current\_Assets value 250,000 have a correspondingly lower debt value, but customers who have Current\_Assets above 250,000 seems to have much higher Total\_Debt and the graph is taking a slow exponential rise.

Chart, scatter chart

Description automatically generated

Figure-1.31 Scatterplot: Total\_Debt vs Current\_Assets

* Revenue\_earnings\_in\_forex vs Revenue\_expenses\_in\_forex

We can see that most of the datapoints above Revenue\_expenses\_in\_forex at 15,000 has much lesser value of Revenue\_earnings\_in\_forex. Seems like customers who deals with large amount of forex are mostly in loss.

Chart, scatter chart

Description automatically generated

Figure-1.32 Scatterplot: Revenue\_earnings\_in\_forex vs Revenue\_expenses\_in\_forex

## Train Test Split.

We will use train\_test\_split module from sklearn library.

The split will be done using test\_size of 0.33 to get a train/test split of 67,33 percentage.

We will execute the split with stratify option to ensure that the defaulters/non-defaulters’ ratio is maintained in both train and test dataset.

After the split we can see that both train and test dataset have the ratio of 89:11.

## 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.

The first step towards model building, should be feature selection. So let us look at the features to see if there are multicollinear features, which could possibly undermine the significance of an independent variable.

Let us look at the collinearity between features through a heatmap:

Chart

Description automatically generated

Figure-1.33 Heatmap- Collinaerity

As we can see in the figure above there are many fields that have strong positive and negative correlation with each other. Net\_Sales and Value\_of\_Output has strong positive correlation whereas Cash\_Flow\_From\_Investing\_Activities and Current\_Assets have strong negative correlation.

Hence we need to identify features with low correlation between each other.

We will determine the VIF (Variance Inflation Factor) of the features and we will choose only features where VIF value is low as high VIF indicates that particular feature is highly collinear with other variables in the model.

|  |
| --- |
| **Feature VIF**  ROG\_Market\_Capitalisation\_perc 1.29  Inventory\_Velocity\_Days 1.29  Current\_RatioLatest 1.30  Revenue\_earnings\_in\_forex 1.38  Debtors\_Velocity\_Days 1.39  Creditors\_Velocity\_Days 1.41  ROG\_Gross\_Block\_perc 1.42  Debtors\_RatioLatest 1.60  Revenue\_expenses\_in\_forex 1.63  Inventory\_RatioLatest 1.64  Cash\_Flow\_From\_Investing\_Activities 1.66  Equity\_Paid\_Up 1.66  ROG\_Cost\_of\_Production\_perc 1.74  Interest\_Cover\_RatioLatest 1.85  Market\_Capitalisation 1.89  Other\_Income 1.99  Cash\_Flow\_From\_Financing\_Activities 2.02  Cash\_Flow\_From\_Operating\_Activities 2.06  Selling\_Cost 2.24  ROG\_Net\_Worth\_perc 2.28  ROG\_Total\_Assets\_perc 2.34  Total\_Debt 2.61  ROG\_Capital\_Employed\_perc 2.64  CEPS\_annualised\_Unit\_Curr 3.49  Net\_Working\_Capital\_ 3.68  Value\_of\_Output\_to\_Gross\_Block 4.51  Fixed\_Assets\_RatioLatest 4.54  APATM\_percLatest 4.61  Gross\_Block\_ 4.70  Book\_Value\_Adj\_Unit\_Curr 5.46  Networth 5.47  ROG\_PAT\_perc 5.67  PBIDTM\_percLatest 5.81  Total\_Asset\_Turnover\_RatioLatest 5.86  Book\_Value\_Unit\_Curr 5.87  Value\_of\_Output\_to\_Total\_Assets 6.24  PBITM\_percLatest 6.76  ROG\_PBIT\_perc 6.79  ROG\_PBT\_perc 6.84  Current\_Liabilities\_and\_Provisions\_ 7.15  ROG\_PBIDT\_perc 7.61  ROG\_CP\_perc 8.33  CPM\_percLatest 9.45  PBIDT 9.68  PBIT 10.83  ROG\_PBDT\_perc 10.84  Current\_Assets\_ 11.28  Adjusted\_PAT 11.69  PBDTM\_percLatest 11.88 |

Table-1.11 VIF - Feature

The above table shows the top 50 features ordered as per VIF value. As a norm VIF values above 5 can be not considered as they indicate that these features introduce collinearity among the other features.

Let us consider the first 30 features in the above table for model creation.

We will be creating the models using logistic regression function logit from statsmodel library.

**Model 1:**

In this model we are considering the first 30 features in table 1.11 as predictors and default as the target variable.

On creating the model and training with the train dataset we get the below summary:

Table

Description automatically generated

Table-1.12 Model 1 - Summary

The R2 value is 0.597 and we can see that the below predictors have significance greater than 0.05 and hence are less effective in explaining the variance of data determining the target variable.

* ROG\_Market\_Capitalisation\_perc
* Inventory\_Velocity\_Days
* Debtors\_Velocity\_Days
* Creditors\_Velocity\_Days
* ROG\_Gross\_Block\_perc
* Debtors\_RatioLatest
* Revenue\_expenses\_in\_forex
* Inventory\_RatioLatest
* Cash\_Flow\_From\_Investing\_Activities
* Equity\_Paid\_Up
* Market\_Capitalisation
* Cash\_Flow\_From\_Financing\_Activities
* Cash\_Flow\_From\_Operating\_Activities
* Selling\_Cost
* ROG\_Total\_Assets\_perc
* Total\_Debt
* ROG\_Capital\_Employed\_perc
* CEPS\_annualised\_Unit\_Curr
* Net\_Working\_Capital\_
* Value\_of\_Output\_to\_Gross\_Block
* Fixed\_Assets\_RatioLatest
* APATM\_percLatest

We will predict the target variable on training data set and check the confusion matrix, against actual train target values.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 2112 | 30 |
| Actual Positive | 87 | 173 |

Table-1.13 Model 1 – Confusion Matrix

Let us look at the classification report for the prediction on training dataset.

Table, calendar

Description automatically generated with medium confidence

Table-1.14 Model 1 – Classification Report

So, we can see that the precision for defaulters is 0.85 and recall is 0.67. In this model we are looking for better recall, as its imperative that we can predict all the defaulters, than incorrectly classifying a non-defaulter as defaulter.

The recall for this model is 67%, that is of all the defaulter’s model can predict 67% of them, which is a low benchmark.

Let us look at further models.

**Model 2:**

This model is an iteration of above model 1, where we will use only the pertinent features and avoid the predictors with p-value greater than 0.05. Hence the features considered for this model are:

* Current\_RatioLatest
* Revenue\_earnings\_in\_forex
* ROG\_Cost\_of\_Production\_perc
* Interest\_Cover\_RatioLatest
* Other\_Income
* ROG\_Net\_Worth\_perc
* Gross\_Block\_
* Book\_Value\_Adj\_Unit\_Curr
* Networth

On creating the model and training with the train dataset we get the below summary:

Graphical user interface, table

Description automatically generated with medium confidence

Table-1.15 Model 2 - Summary

The R2 value 0.5797 is lesser than that of model 1 and it is expected as we have reduced features, which helps in explaining some of the variance in the predictor.

We can see that all features have p-value lesser than 0.05 and hence are significant in explaining the variance of the target variable.

We will predict the target variable on training data set and check the confusion matrix, against actual train target values.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 2116 | 26 |
| Actual Positive | 87 | 173 |

Table-1.16 Model 2 – Confusion Matrix-Train

Let us look at the classification report for the prediction on training dataset.

Table

Description automatically generated

Table-1.17 Model 2 – Classification Report-Train

So, we can see that the precision for defaulters is 0.87 and recall is 0.67. In this model we are looking for better recall, as its imperative that we can predict all the defaulters, than incorrectly classifying a non-defaulter as defaulter.

The recall for this model is also at 67%, that is of all the defaulter’s model can predict 67% of them, which is a low benchmark.

The reason for the low recall could be that data is imbalances as the total number of defaulters in the input dataset is at 11%. Let us do a SMOTE (Synthetic Minority Oversampling Technique) on the training dataset to make the dataset balanced.

We will perform SMOTE on the training dataset with sampling strategy as ‘auto’.

**Model 3:**

In this model we will again consider the first 30 features in table 1.11 as predictors and default as the target variable.

On creating the model and training with the smote-train dataset we get the below summary:

A picture containing table

Description automatically generated

Table-1.18 Model 3 - Summary

The R2 value is 0.6901 and higher than the R2 value of model 1 and model 2, which is better. We can see that the below predictors have significance greater than 0.05 and hence are less effective in explaining the variance of data determining the target variable.

* ROG\_Market\_Capitalisation\_perc
* Debtors\_RatioLatest
* Revenue\_expenses\_in\_forex
* Cash\_Flow\_From\_Investing\_Activities
* Equity\_Paid\_Up
* Cash\_Flow\_From\_Operating\_Activities
* Selling\_Cost
* ROG\_Total\_Assets\_perc
* ROG\_Capital\_Employed\_perc
* Net\_Working\_Capital\_
* Value\_of\_Output\_to\_Gross\_Block
* APATM\_percLatest

We will predict the target variable on smote-training data set and check the confusion matrix, against actual smote-train target values.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 1935 | 207 |
| Actual Positive | 129 | 2013 |

Table-1.19 Model 3 – Confusion Matrix-Train

Let us look at the classification report for the prediction on training dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Table-1.20 Model 3 – Classification Report-Train

So, we can see that the precision for defaulters is 0.91 and recall is 0.94. In this model we are looking for better recall, as its imperative that we can predict all the defaulters, than incorrectly classifying a non-defaulter as defaulter.

The recall for this model is high and the model performance looks good, we can confirm the model after checking the performance again the test dataset too.

**Model 4:**

This model is an iteration of above model 3, where we will use only the pertinent features and avoid the predictors with p-value greater than 0.05. Hence the features considered for this model are:

* Inventory\_Velocity\_Days
* Current\_RatioLatest
* Revenue\_earnings\_in\_forex
* Debtors\_Velocity\_Days
* Creditors\_Velocity\_Days
* ROG\_Gross\_Block\_perc
* Inventory\_RatioLatest
* ROG\_Cost\_of\_Production\_perc
* Interest\_Cover\_RatioLatest
* Market\_Capitalisation
* Other\_Income
* Cash\_Flow\_From\_Financing\_Activities
* ROG\_Net\_Worth\_perc
* Total\_Debt
* CEPS\_annualised\_Unit\_Curr
* Fixed\_Assets\_RatioLatest
* Gross\_Block\_
* Book\_Value\_Adj\_Unit\_Curr
* Networth

On creating the model and training with the smote-train dataset we get the below summary:

Graphical user interface, application, table

Description automatically generated

Table-1.21 Model 4 - Summary

The R2 value is 0.6871 and higher than the R2 value of model 1 and model 2, which is better. Its slightly lesser than model 3 and it’s expected as we have removed some features. We can see that all predictors have p-values lesser than 0.05 and is significant in explaining the deviance towards target variable.

We will predict the target variable on smote-training data set and check the confusion matrix, against actual smote-train target values.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 1948 | 194 |
| Actual Positive | 129 | 2013 |

Table-1.22 Model 4 – Confusion Matrix-Train

Let us look at the classification report for the prediction on training dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Table-1.23 Model 4 – Classification Report-Train

So, we can see that the precision for defaulters is 0.91 and recall is 0.94. In this model we are looking for better recall, as its imperative that we can predict all the defaulters, than incorrectly classifying a non-defaulter as defaulter.

The recall for this model is high and the model performance looks good, we can confirm the model after checking the performance again the test dataset. The performance measures are pretty much as model 3.

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

Let us look at the above 4 model’s performance on the test dataset.

**Model 1:**

We will predict the target variable on test data set and check the confusion matrix, against actual test target values.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 1038 | 18 |
| Actual Positive | 52 | 76 |

Table-1.24 Model 1 – Confusion Matrix-Test

Let us look at the classification report for the prediction on testing dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Table-1.25 Model 1 – Classification Report-Test

Let us also look at ROC\_AUC score and ROC curve:

Chart, line chart

Description automatically generated

Figure-1.34 Model 1 –ROC\_AUC score/ROC curve

So, we can see that the precision for defaulters is 0.81 and recall is 0.59 and ROC\_AUC score is 0.79

The recall for this model is 59%, that is of all the defaulters, model can predict 59% of them, which is a low benchmark and not acceptable model.

**Model 2:**

We will predict the target variable on test data set and check the confusion matrix, against actual test target values.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 1040 | 16 |
| Actual Positive | 49 | 79 |

Table-1.26 Model 2 – Confusion Matrix-Test

Let us look at the classification report for the prediction on testing dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Table-1.27 Model 2 – Classification Report-Test

Let us also look at ROC\_AUC score and ROC curve:

Chart, line chart

Description automatically generated

Figure-1.35 Model 2 –ROC\_AUC score/ROC curve

So, we can see that the precision for defaulters is 0.83 and recall is 0.62 and ROC\_AUC score is 0.8.

The recall for this model is 62%, that is of all the defaulters, model can predict 62% of them, which is a low benchmark and not acceptable model, though it is better than model 1.

**Model 3:**

We will predict the target variable on test data set and check the confusion matrix, against actual test target values. The train dataset for this model has been balanced using SMOTE.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 936 | 120 |
| Actual Positive | 18 | 110 |

Table-1.28 Model 3 – Confusion Matrix-Test

Let us look at the classification report for the prediction on testing dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Table-1.29 Model 3 – Classification Report-Test

Let us also look at ROC\_AUC score and ROC curve:

Chart, line chart

Description automatically generated

Figure-1.36 Model 3 –ROC\_AUC score/ROC curve

So, we can see that the precision for defaulters is 0.48 and recall is 0.86 and ROC\_AUC score is 0.87.

The recall for this model is 86%, that is of all the defaulters, model can predict 86% of them, which is an acceptable benchmark. The recall for the training dataset is 94% and hence the model is neither overfit nor underfit as the recall percent difference is within the threshold of 10%.

**Model 4:**

We will predict the target variable on test data set and check the confusion matrix, against actual test target values. The train dataset for this model has been balanced using SMOTE and the model has only featured in model 3 which are within the p-value of 0.05.

|  |  |  |
| --- | --- | --- |
|  | Predicted Negative | Predicted Positive |
| Actual Negative | 934 | 122 |
| Actual Positive | 18 | 110 |

Table-1.30 Model 4 – Confusion Matrix-Test

Let us look at the classification report for the prediction on testing dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Table-1.31 Model 4 – Classification Report-Test

Let us look at the ROC\_AUC score and ROC curve of model 4 on testing data.

Chart, line chart

Description automatically generated

Figure-1.37 Model 4 –ROC\_AUC score/ROC curve

So, we can see that the precision for defaulters is 0.47 and recall is 0.86, and ROC\_AUC score is 0.87.

The recall for this model is 86%, that is of all the defaulters, model can predict 86% of them, which is an acceptable benchmark. The recall for the training dataset is 94% and hence the model is neither overfit or underfit as the recall percent difference is within the threshold of 10%.

We can choose model 4 as the apt model as it has comparable recall with Model 3 and uses lesser features, thus being cost effective.

Since the input dataset is imbalanced, SMOTE has helped to balance the learning dataset and thus improve the recall statistics as required in this business scenario.

We have achieved a decent recall value without overfitting and ROC\_AUC score is a decent 0.87. Considering the opportunities such as outliers, missing values, and correlated features this is a fairly good model. It can be improved if we get better quality data where the features explaining the default are not missing to this extent

# THE END