Project – Finance And risk analytics – milestone 1

Name: Adithya Manivannan

PGP Data Science & Business Analytics

Table of Contents

[Project – Finance And risk analytics 0](https://d.docs.live.net/7b7e93144ae82396/Documents/Machine_Learning_Project.docx#_Toc108607484)

[Milestone 1: 3](#_Toc108607485)

[Problem Statement 3](#_Toc108607486)

[Project Objective: 3](#_Toc108607487)

[EDA: 3](#_Toc108607488)

[Information on dataset: 4](#_Toc108607489)

[Summary of the dataset: 5](#_Toc108607490)

[Inference 5](#_Toc108607491)

[1.1 Outlier Treatment 6](#_Toc108607492)

[1.2 Missing Value Treatment 6](#_Toc108607493)

[1.3 Transform Target variable into 0 and 1 7](#_Toc108607494)

[Unique values for categorical variables 7](#_Toc108607495)

[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) 7](#_Toc108607496)

[1.5 Train Test Split 11](#_Toc108607497)

[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 12](#_Toc108607498)

[1.7 Validate the Model on Test Dataset and state the performance matrices. Also state interpretation from the model 15](#_Toc108607499)

[Model Validation on Test Dataset: 15](#_Toc108607500)

[Interpretation of the model: 15](#_Toc108607501)

# List of Figures

Fig. 1. - Networth\_Next\_Year Vs 'Capital\_Employed' 6

Fig. 2. - Capital\_Employed Vs Gross\_Block\_ 7

Fig. 3. - Net\_ Sales Vs Networth\_Next\_Year 7

Fig. 4. - Net\_sales Vs Gross\_Block 7

Fig. 5. - PBIT Vs Value\_of\_output 8

Fig. 6. - Cost\_of\_Production Vs Value\_of\_Output 8

Fig. 7. - Capital\_Employed Vs Net\_Sales 8

Fig. 8. - PBIT Vs PBIDT 9

Fig. 9. - Current Liabilites Vs Provisions and Networth next year 9

Fig. 10 - Networth next year Vs Other income 9

Fig. 11. - Correlation Heatmap Plot 10

Fig. 12. – Confusion Matrix 16

# List of Tables

Table 1. - Head of Dataset 3

Table 2. – Information on Dataset 4

Table 3. – Summary of Dataset 5

Table 4. – Summary table – Logistic Regression Model 13

Table 5. – Final Model 15

Table 6. – Classification Report 16

# Milestone 1:

## 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 Networth of the company in the following year (2016) is provided which can be used to drive the labeled field.

## Project Objective:

The Objective of the report is to explore the dataset "Company\_Data2015.xlsx"in Python) and perform Logistic regression to categorize the companies as defaulters:

• Importing the dataset in jupyter notebook.

• Understanding the structure of dataset.

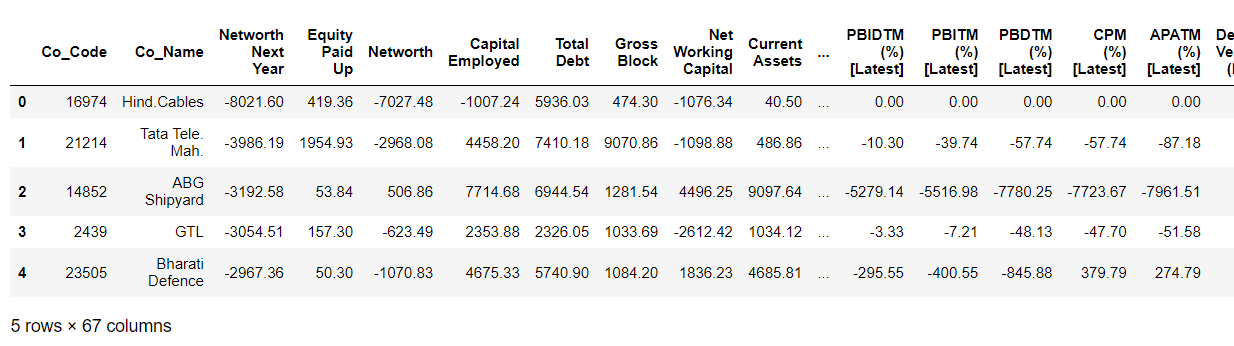
• Exploratory Data analysis

• Graphical exploration

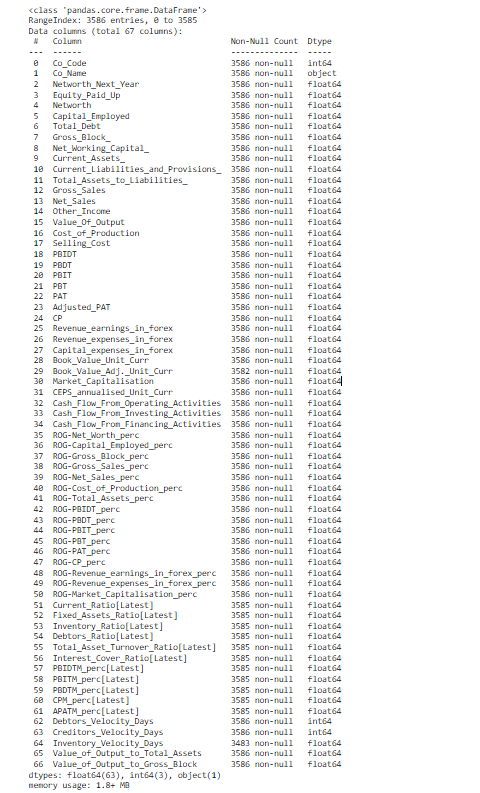
• Prediction using logistic regression

## EDA:

Head of the dataset

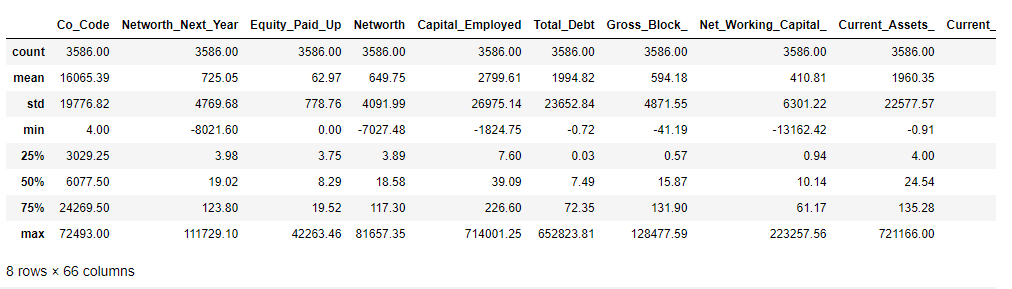


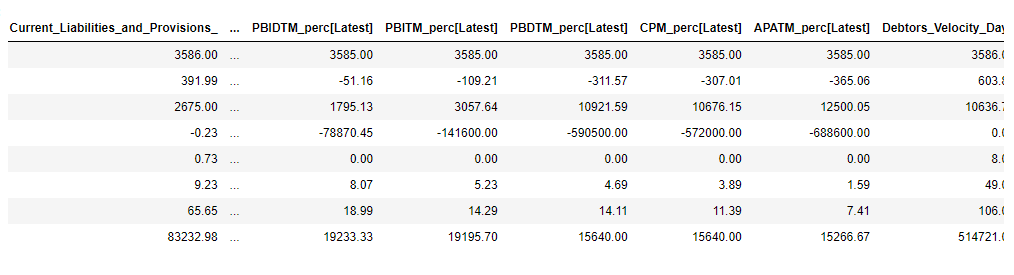
### Information on dataset:

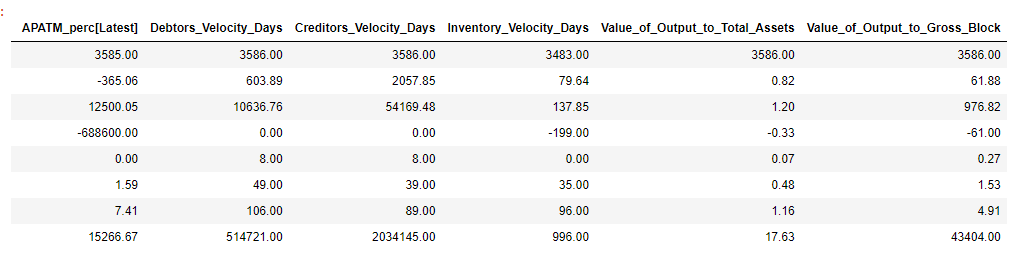


* There are 67 columns and 3586 rows of data.
* Of the 67 columns, only Co\_Name is of Object data type while every other column is either integer of float type. This makes sense since this is a balance sheet of each company.
* It looks like the Inventory\_Velocity\_Days column has quite a few empty rows with no data as it has only 3483 non-null values.
* There are a few other columns with a few negligible count of null values while most of the columns have no non-null values.

### Summary of the dataset:



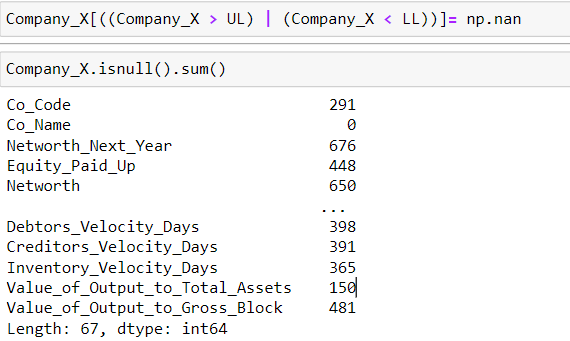
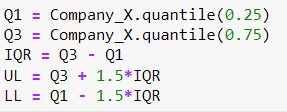




### Inference

* The Networth\_Next\_Year column has a mean value of 725.08 while the median is 19.02. This shows the existence of a large number of outliers.
* Similarly, we can see that most other columns as well have a huge difference between their mean and median values. We suspect these columns to have outliers as well.
* We have a company with a maximum networth next year of 1111729.1 while the minimum networth next year would be a -8021.6.
* The Inventory\_Velocity\_Days column gives the average number of days the company needs to turn its inventory into sales. However, the minimum value here is -199 which does not make sense since we are talking about days here and it cannot be negative.
* The average of Current\_Assets is 1960 while that of Current\_Liabilities is 391. This shows that on an average most companies have more assets than liabilities and that is a good thing.
* It looks like a lot of these values are correlated with each other and we can analyse this further as we proceed.

## Outlier Treatment



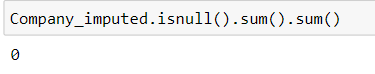
We convert all the outliers into null values so that we can decide how to deal with them. While testing the null values, if we drop all the rows that have more than 5 null columns, we came down to 1177 rows from 3586 rows provided originally, we lose a larger proportion of actual defaulters 388 in the original data vs. only 188 in the subset, therefore, end up losing more than 70% of the actual defaulters. Hence, we cannot drop the null values and instead have to impute them.

## 1.2 Missing Value Treatment

We first drop columns that have more than 20% null values since they do not add much value to our model building. Once we have dropped these columns, we scale the predictor variables and then impute the missing values using KNNImputer.

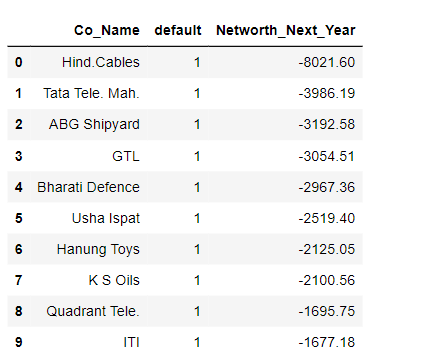
The KNNImputer is used to fill in missing values in a dataset using the k-Nearest Neighbors method. Here, we are using the 10 nearest neighbouts to predict the value. The KNNImputer predicts the value of a missing value by observing trends in related columns.

Now, the missing values have been imputed and there are no missing values now.



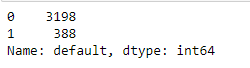
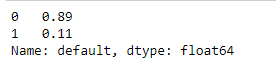
## 1.3 Transform Target variable into 0 and 1



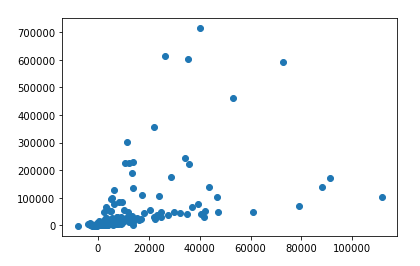


### Unique values for categorical variables

We have converted the target variable in 0 & 1. We can see that in the dataset we have, there is a huge difference in the target variables as we have only 11% of defaulters. This could cause a bias if we do not deal with it.

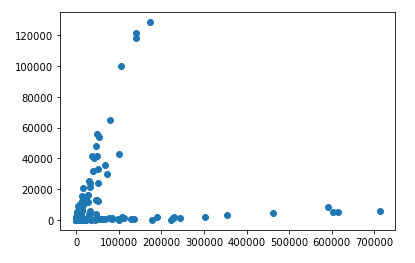
 

## 1.4 Univariate (4 marks) & Bivariate (6marks) analysis with proper interpretation. (You may choose to include only those variables which were significant in the model building)



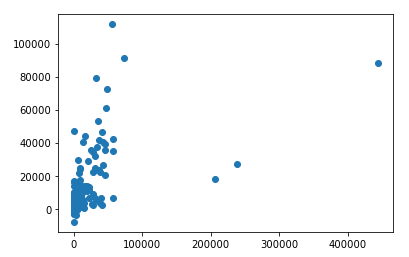
Networth\_Next\_Year Vs 'Capital\_Employed'

We see no direct correlation between the Networth\_Next\_Year and Capital\_Employed.



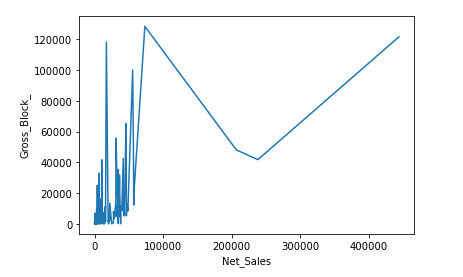
Capital\_Employed Vs Gross\_Block\_

Here, we can see two categories of companies. 1) Group 1: As the capital employed increases either the gross block is high as well. 2) The Capital employed increase does noe increase the gross block as well.

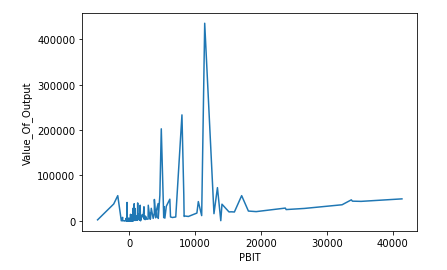


Net Sales Vs Networth\_Next\_Year

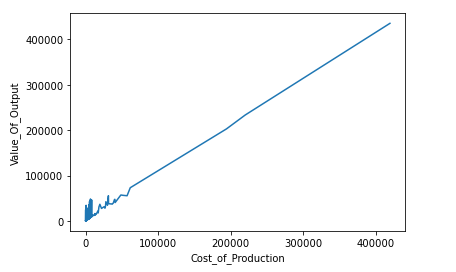
An increase in Net Sales shows an increase in the Networth next year as well just as expected.



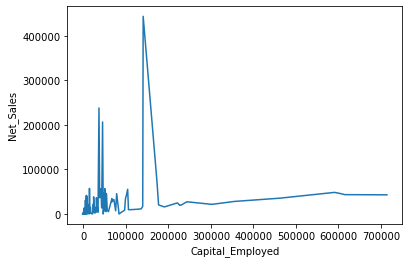
We do see an upward trend in the Net\_sales Vs Gross\_Block chart however there is a zig zag curve with drops here and there.



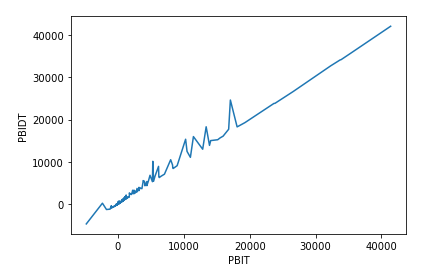
There is no direct relationship between PBIT and Value\_of\_output.



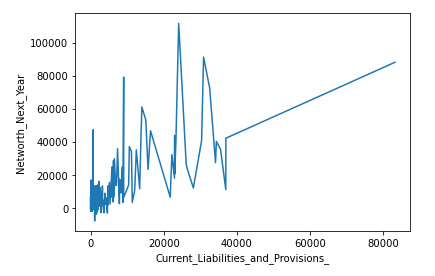
The Cost\_of\_Production and Value\_of\_Output shows a linear relationship i.e. an increase in cost of production shows an increase in value of output.



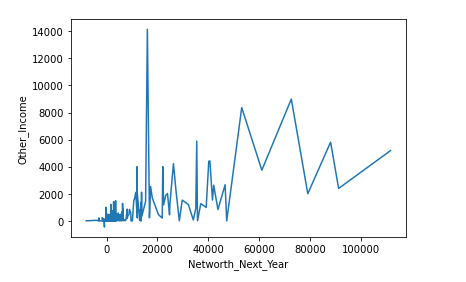
Although there is an upward trend, there is no linear relationship between the Capital\_Employed and Net\_Sales unlike we had expected.



There is a linear relationship between PBIT and PBIDT just as expected.

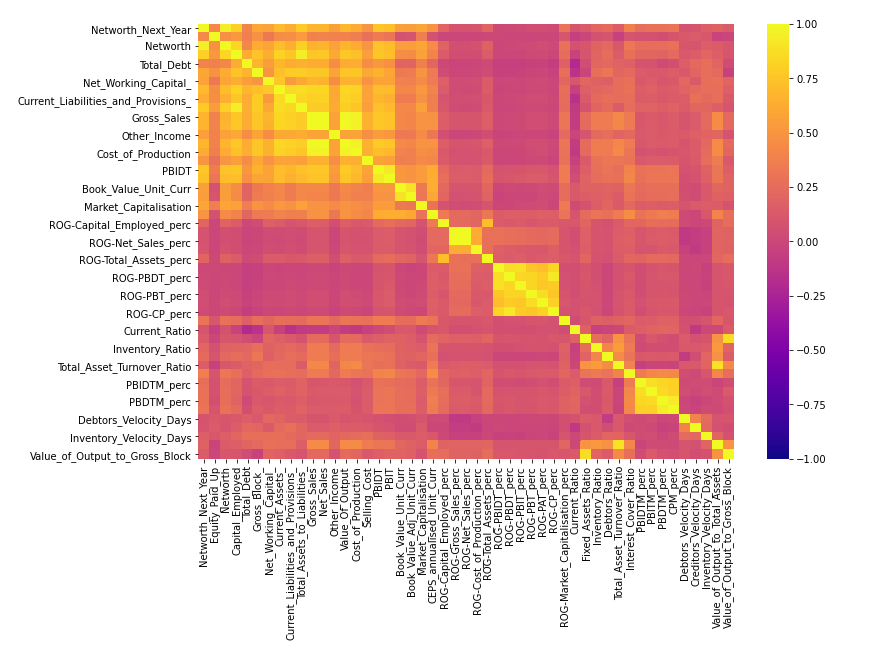


We do see a direct relationship between the Current Liabilites and Provisions and Networth next year but this is an upward trend unlike we expect. We thought that an increase in the liabilities would reduce the networth next year.



Although there is a small increasing trend between Networth next year and Other income, there is no direct correlation.

**Correlation Heatmap Plot**

****

The correlation heatmap shows that after removing the columns that have more than 20% null values, we have the above heatmap that has very minimal correlation between the dependant variables.

There are a few columns such as PBIDT & PBIT, Gross Sales & Net Sales, Gross Sales & Cost of Production are some that have a higher correlation are.

## 1.5 Train Test Split

Separating independent and dependent variables for the logistic regression model

X = independent variables

Y = dependent variable

The training set for the independent variables, X\_train: (2402, 49)

The training set for the dependent variable, y\_train: (2402, 1)

The test set for the independent variables, X\_test: (1184, 49)

The test set for the dependent variable, y\_test: (1184, 1)

We will now use this split data set to train our Logistic regression model using Statsmodel library and then test it to see it’s accuracy and recall measures.

## 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

**Logistic Regression Model**

Logistic regression is the type of regression analysis used to find the probability of a certain event occurring. It is the best suited type of regression for cases where we have a categorical dependent variable which can take only discrete values

Two libraries of Logistic regression

1. sklearn

2. statsmodel

Here for this model, we are using the statsmodel library to perform the Logistic Regression

Statsmodels is a Python module that provides various functions for estimating different statistical models and performing statistical tests

First, we define the set of dependent(y) and independent(X) variables. If the dependent variable is in non-numeric form, it is first converted to numeric using dummies.

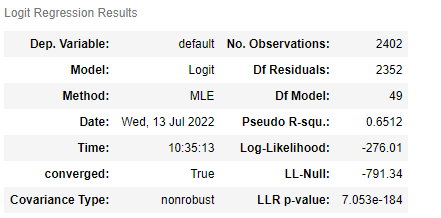
Statsmodels provides a Logit() function for performing logistic regression. The Logit() function accepts y and X as parameters and returns the Logit object. The model is then fitted to the data.

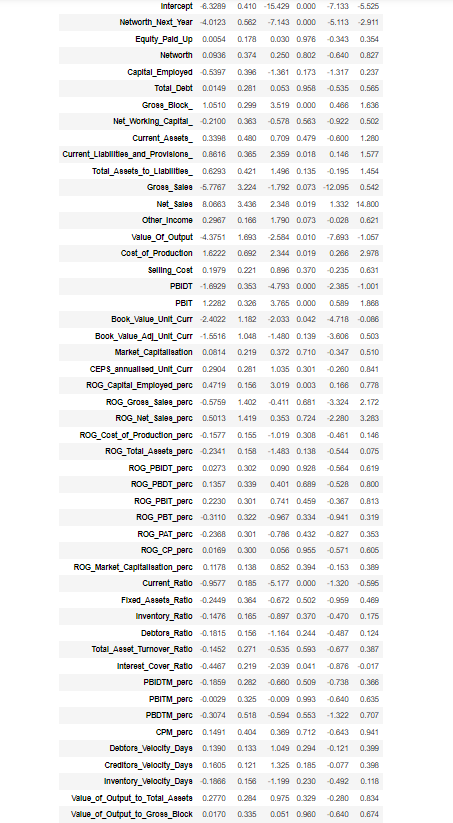
**Output:**



In the output, ‘Iterations’ refer to the number of times the model iterates over the data, trying to optimize the model. By default, the maximum number of iterations performed is 35, after which the optimization fails.

**Summary table:**





coef: the coefficients of the independent variables in the regression equation.

Log-Likelihood: the natural logarithm of the Maximum Likelihood Estimation (MLE) function. MLE is the optimization process of finding the set of parameters that result in the best fit.

LL-Null: the value of log-likelihood of the model when no independent variable is included (only an intercept is included).

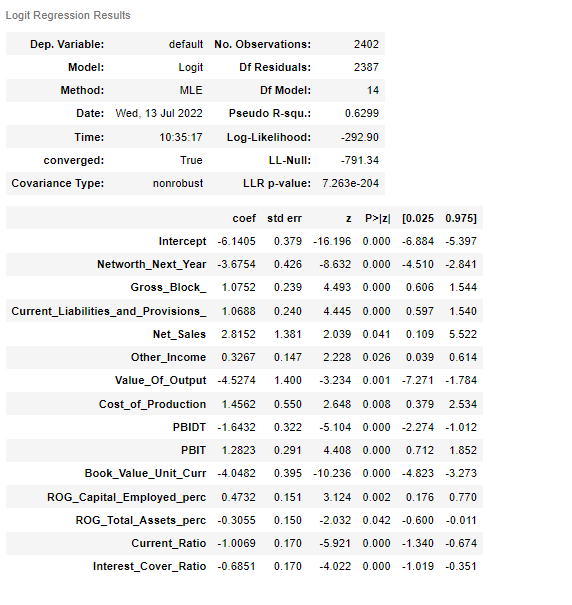
Pseudo R-squ.: a substitute for the R-Squared value in Least Squares linear regression. It is the ratio of the log-likelihood of the null model to that of the full model.

But what we are most focused on is the resultant p-values for each independent variable. If the p-value is greater than 0.05, we understand that is insignificant in the model building and keep eliminating those variables to optimize the regression model.

We iterate multiple models by repeating this process till we have a model where the coeff of each independent variable is less than 0.05.

By performing the same, we arrive at the following model which we decide as final.

**Final Model:**



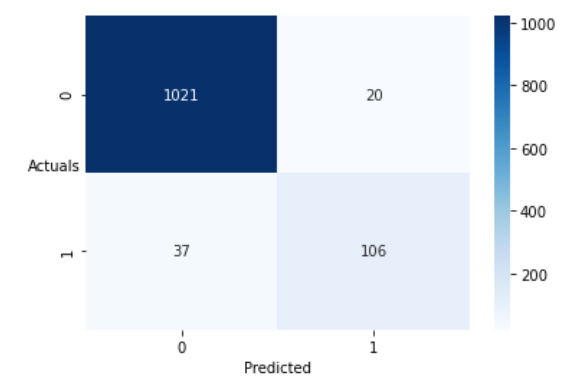
Hence, the most important variables in this logistic regression model are:

* Networth\_Next\_Year
* Gross\_Block\_
* Current\_Liabilities\_and\_Provisions
* Net\_Sales, Other\_Income
* Value\_Of\_Output
* Cost\_of\_Production
* PBIDT
* PBIT
* Book\_Value\_Unit\_Curr
* ROG\_Capital\_Employed\_perc
* ROG\_Total\_Assets\_perc
* Current\_Ratio
* Interest\_Cover\_Ratio

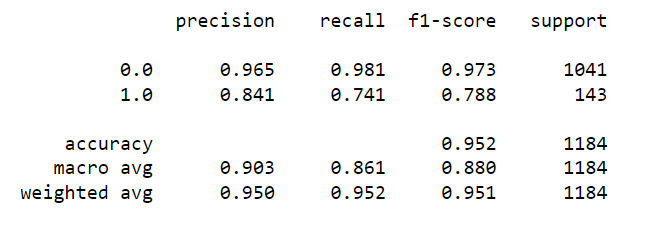
## 1.7 Validate the Model on Test Dataset and state the performance matrices. Also state interpretation from the model

### Model Validation on Test Dataset:

#### Confusion Matrix:



#### Classification Report:



### Interpretation of the model:

Overall, with an accuracy score of 0.952, from the model built, 95% of the predictions made were correct.

With a recall score of 0.741, 74.1% of those defaulted were correctly identified as defaulters by the model. This is a decent model that can be improved by providing more balanced dataset.