yogesh konduru

In this project, we have built a credit risk model using logistic regression from stats model and the random forest model using sklearn and we have also optimized the models for improving the performance of the models

FRA-PROJECT

PGP-DSBA

Contents

[Problem Statement 2](#_Toc128292651)

[1. Missing Value Treatment 4](#_Toc128292652)

[2. Outlier Detection and Treatment 5](#_Toc128292653)

[3. Checking Correlation amongst the variables using heatmap 6](#_Toc128292654)

[4. Train Test Data Split 7](#_Toc128292655)

[5. 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 7](#_Toc128292656)

[6. Checking the accuracy of the model using confusion matrix for training set 11](#_Toc128292657)

[7. Checking the accuracy of the model using confusion matrix for test set 12](#_Toc128292658)

[8. Build a Random Forest Model on Train Dataset. Also showcase your model building approach 13](#_Toc128292659)

[1.Random Forest model with default parameters 14](#_Toc128292660)

[2.Random Forest model with grid search parameters 14](#_Toc128292661)

[3.Random Forest model with feature selection 15](#_Toc128292662)

[4.Random Forest model with feature selection with grid search 16](#_Toc128292663)

[5.Random Forest model with resampled using smote technique 17](#_Toc128292664)

[6.Random Forest model with resampled using smote technique with grid search 18](#_Toc128292665)

[7.Random Forest model with resampled using smote tomek technique 19](#_Toc128292666)

[8.Random Forest model with resampled using smote tomek technique with grid search 19](#_Toc128292667)

[9. Validate the Random Forest Model on test Dataset and state the performance matrices. Also state interpretation from the model 20](#_Toc128292668)

[1.Random Forest model with default parameters 20](#_Toc128292669)

[2.Random Forest model with grid search parameters 21](#_Toc128292670)

[3.Random Forest model with feature selection 22](#_Toc128292671)

[4.Random Forest model with feature selection with grid search 23](#_Toc128292672)

[5.Random Forest model with resampled using smote technique 25](#_Toc128292673)

[6.Random Forest model with resampled using smote technique with grid search 26](#_Toc128292674)

[7.Random Forest model with resampled using smote tomek technique 27](#_Toc128292675)

[8.Random Forest model with resampled using smote tomek technique with grid search 28](#_Toc128292676)

[10. Comparison of Random Forest model with logistic regression model 30](#_Toc128292677)

[11. Conclusion and Recommendations 31](#_Toc128292678)

# Problem Statement

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

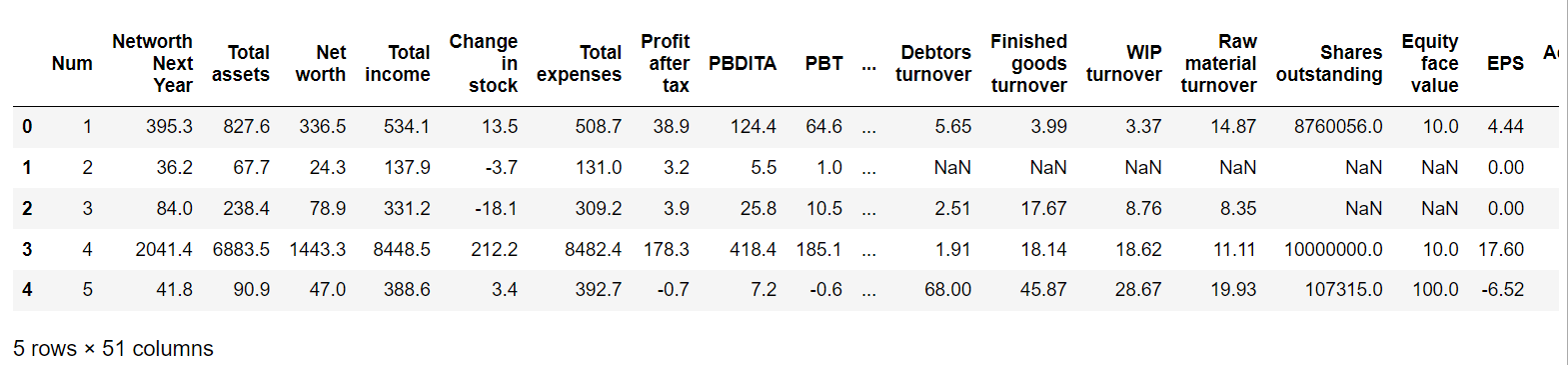
Validation Dataset - We need to build the model on a train dataset and check the model performance measures on the validation dataset.

Dataset for Problem 1: Company(FRA) (4).csv

Data Dictionary: Data\_Dictionary(4).xlsx

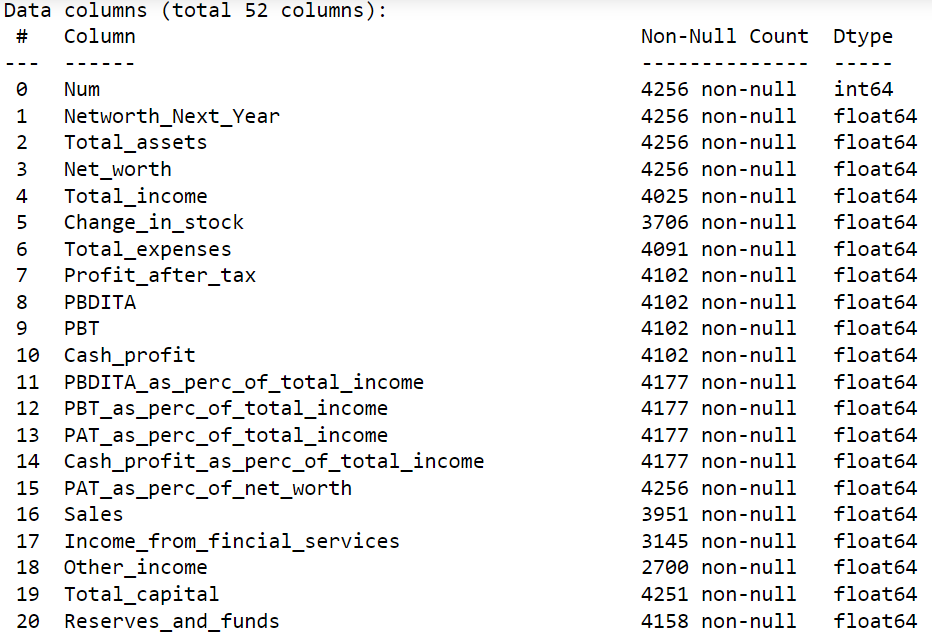
**Loading the Dataset**:

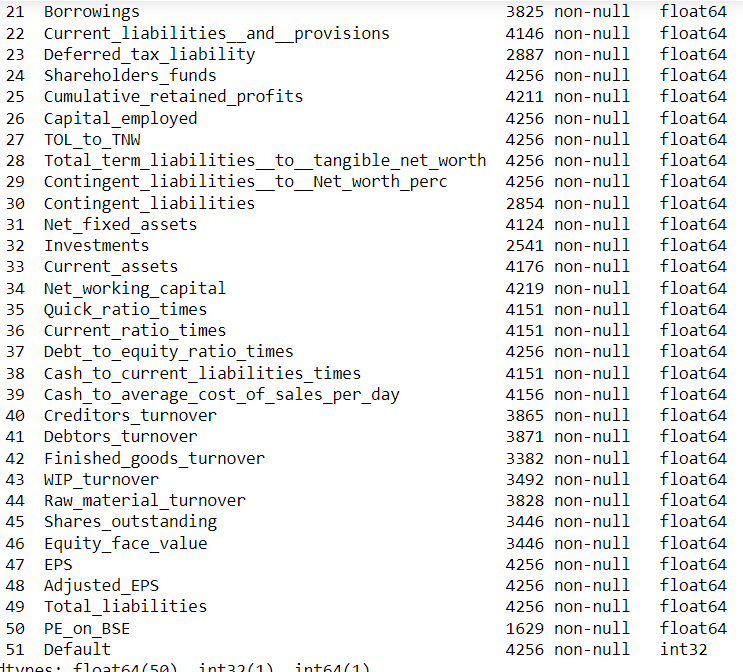
The first 5 rows of the dataset are as follows



**Dataset info:**

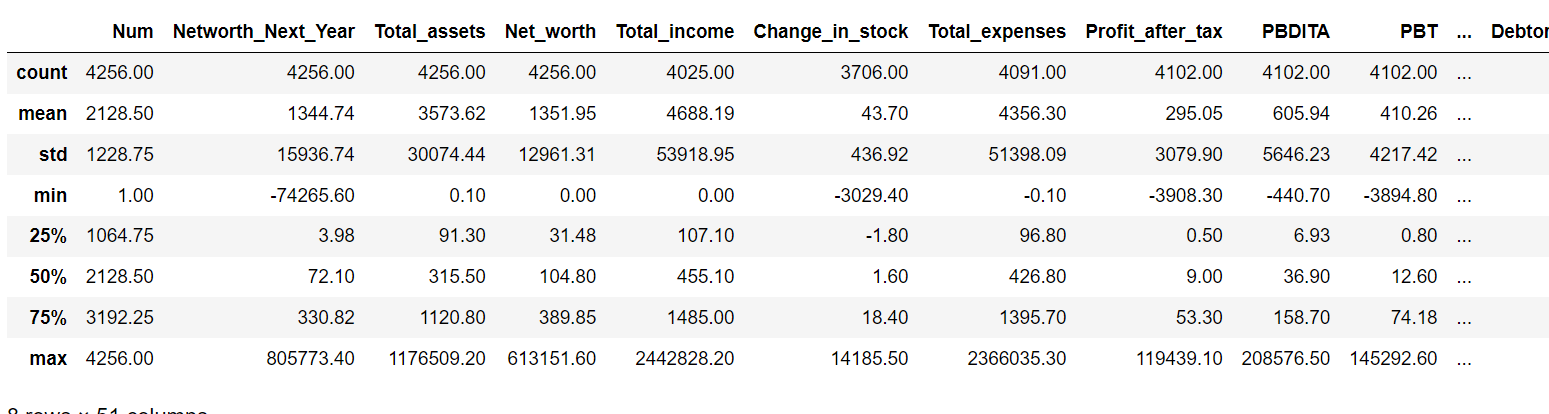
* The dataset contains 4256 records and 51 features in the dataset
* All messy column names are fixed. All spaces, commas, brackets are replaced with either ‘\_’
* All the variables are of the float datatype
* There are no duplicates in the dataset
* There are missing values in our dataset and we need to treat it properly





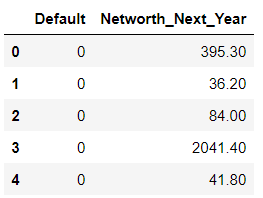
**Dataset Description:**

The dataset description for some of he features is mentioned below



**Default Variable:**

From the given problem statement, We have created a default variable which should take the value of 1 when net worth next year is negative & 0 when net worth next year is positive

****

# 1. Missing Value Treatment

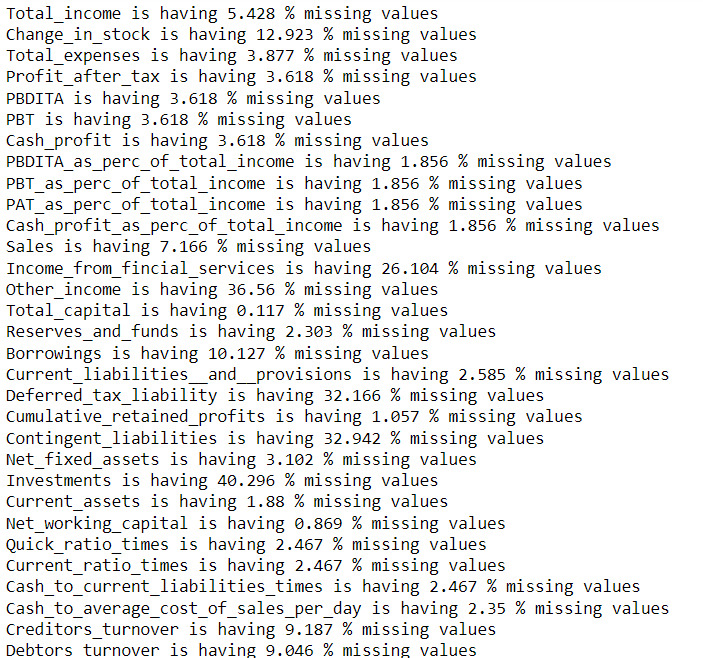
Before going to the missing values treatment, first we need to calculate how much percentage of missing values are present in our dataset

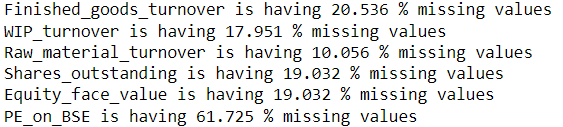
No. of missing values in our dataset = 17778

Total no. of values in our dataset = 217056

percentage of missing values are present in our dataset = (17778/217056) \*100 = 8%

**Missing value percentage with respect to features**

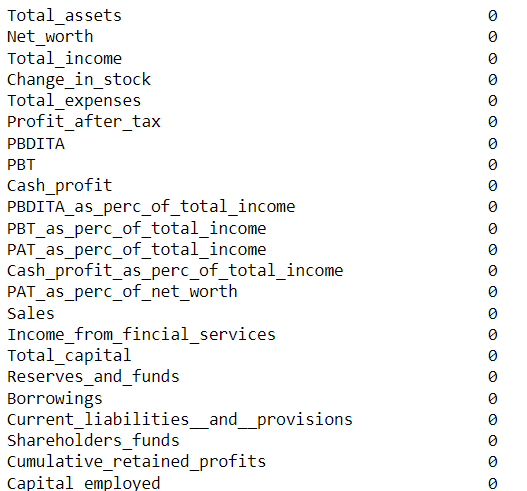




We are dropping the features (Other\_income, Deferred\_tax\_liability, Contingent\_liabilities, Investments, PE\_on\_BSE)that are having more than 30% missing values in the dataset

And for the remaining features we have used the KNN imputer because we have many extremen outliers in our the dataset and It is not good to impute with the median and as well as mean so we have the KNN imputer for this Missing value imputation

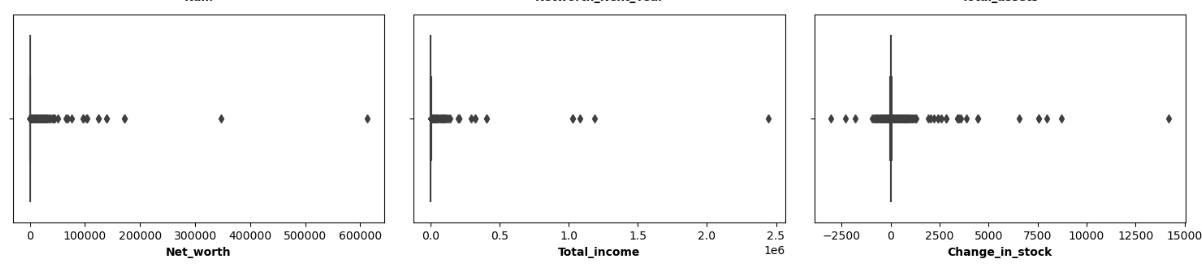
**After missing values imputation:**

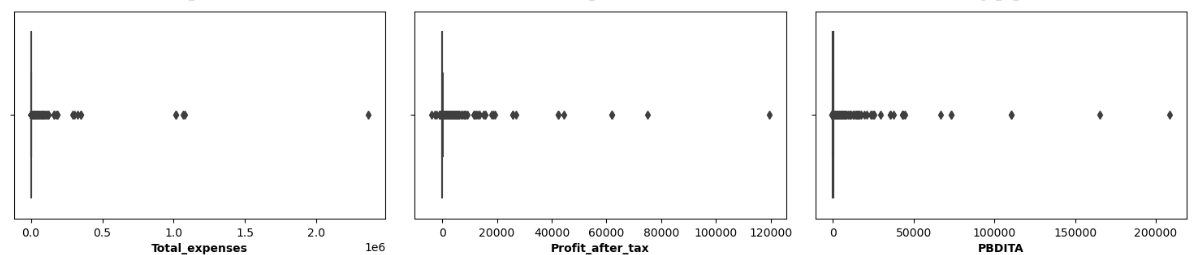


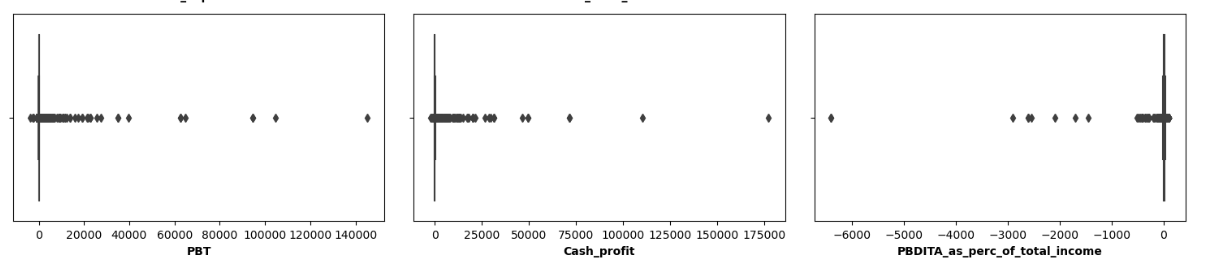
# 2. Outlier Detection and Treatment

**Outlier Detection:**

Let’s look at some of the boxplots in our dataset







Before going to the Outlier treatment, first we need to calculate how much percentage of outlier values are present in our dataset

Conventionally, outliers are identified based on the inter-quartile distance as follows:

Q1 – 25th Percentile

Q3 – 75th Percentile

IQR = Q3 – Q1

Lower Outlier = Value < (Q1 - 1.5 \* IQR)

Upper Outlier = Value > (Q3 + 1.5 \* IQR)

No. of outliers in our dataset = 25946

Total no. of values in our dataset = 217056

percentage of missing values are present in our dataset = (25946/217056) \*100 = 12%

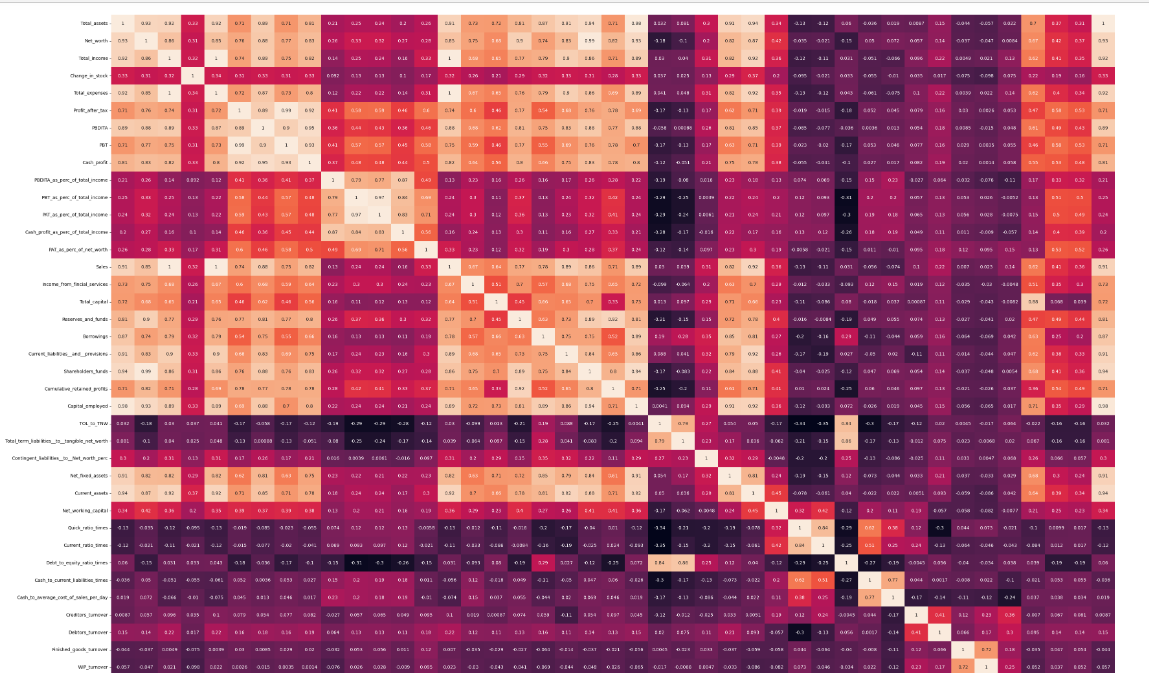
Based on the above calculation, we can say that there are 12% outliers in our dataset

**Outlier treatment:**

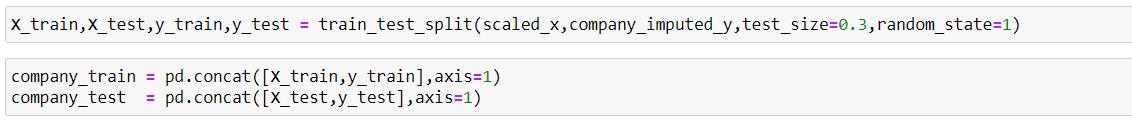
* If Value < (Q1 - 1.5 \* IQR), then replace it with Lower Limit (Q1-1.5 \*IQR)
* If Value > (Q3 + 1.5 \* IQR), then replace it with Upper limit (Q3+1.5\*IQR)

With the above treatment, we have capped the outliers to the lower limit and upper limit

# 3. Checking Correlation amongst the variables using heatmap



# 4. Train Test Data Split

Before splitting the data into the train and test split, we have done the standard scaling and then we have done the train test split with 70:30 raton

# 5. 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

Approach to Model Building:

1. The model is built on all the variables without checking for multi-collinearity

2. In the second step is to address the problem of multi-collinearity. This occurs when the independent variables are highly correlated to each other. We are removing the feature having highest VIF in each iteration and we will iterate this step until we have the features VIF < 5

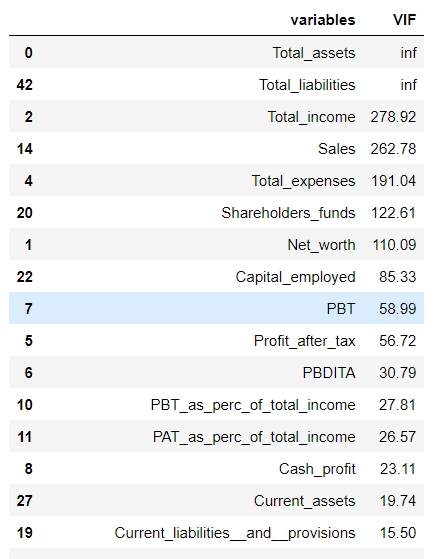
3. The next step is to build the logistic regression model with the remaining variables in an iterative manner. The Null Hypothesis is that there is no relationship between the independent variables and the dependent variable. We will reject the Null Hypothesis if p < 0.05; i.e, all variables whose p values < 0.05 have a relationship with the dependent variable, and are therefore, retained for the next iteration.

4. All variables whose p values > 0.05 do not have an influence on the dependent variable, and therefore, are exclude the feature having highest p-value from the next iteration of the model building exercise

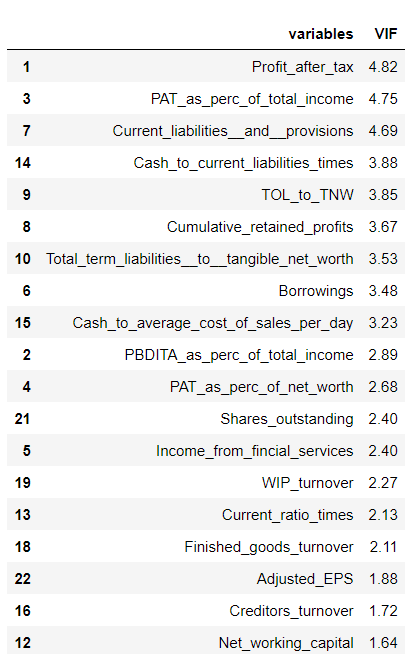
5. We continue this iteration till all the variables have p values < 0.05 4. This model is designated Model A

**VIF Results**

**1st Iteration:**

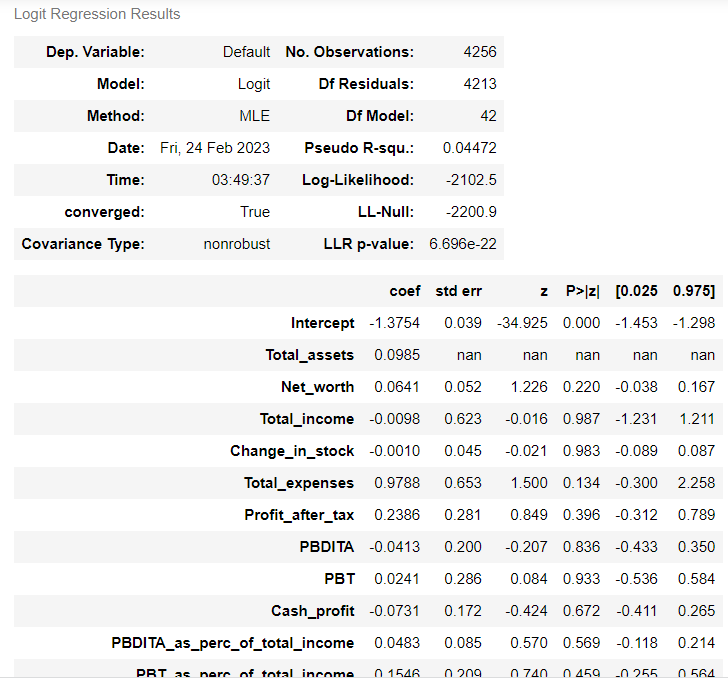


**Last Iteration:**



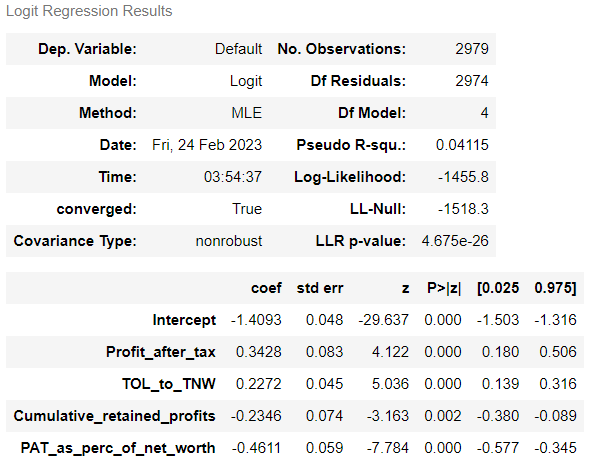
**Iteration 1 model building:**

f\_1 = 'Default ~ Total\_assets+Net\_worth+Total\_income+Change\_in\_stock+Total\_expenses+Profit\_after\_tax+PBDITA+PBT+Cash\_profit+PBDITA\_as\_perc\_of\_total\_income+PBT\_as\_perc\_of\_total\_income+PAT\_as\_perc\_of\_total\_income+Cash\_profit\_as\_perc\_of\_total\_income+PAT\_as\_perc\_of\_net\_worth+Sales+Income\_from\_fincial\_services+Total\_capital+Reserves\_and\_funds+Borrowings+Current\_liabilities\_\_and\_\_provisions+Shareholders\_funds+Cumulative\_retained\_profits+Capital\_employed+TOL\_to\_TNW+Total\_term\_liabilities\_\_to\_\_tangible\_net\_worth+Contingent\_liabilities\_\_to\_\_Net\_worth\_perc+Net\_fixed\_assets+Current\_assets+Net\_working\_capital+Quick\_ratio\_times+Current\_ratio\_times+Debt\_to\_equity\_ratio\_times+Cash\_to\_current\_liabilities\_times+Cash\_to\_average\_cost\_of\_sales\_per\_day+Creditors\_turnover+Debtors\_turnover+Finished\_goods\_turnover+WIP\_turnover+Raw\_material\_turnover+Shares\_outstanding+EPS+Adjusted\_EPS+Total\_liabilities'



**Model 20:**

f\_20 = 'Default ~ Profit\_after\_tax+ TOL\_to\_TNW + Cumulative\_retained\_profits + PAT\_as\_perc\_of\_net\_worth'



**Optimum Cut-off:**

There needs to be a balance between the following 2 opposing factors:

False Positives This means the model predicts the company is a defaulter, but in reality, is not a defaulter. This scenario represents a Lost Opportunity for the investor, since he would not have invested in the company, thinking it was a defaulter.

False negatives This is when the model predicts the company is not a defaulter, but in reality, actually defaults. This is a big loss to the investor, and therefore, needs to be minimized.

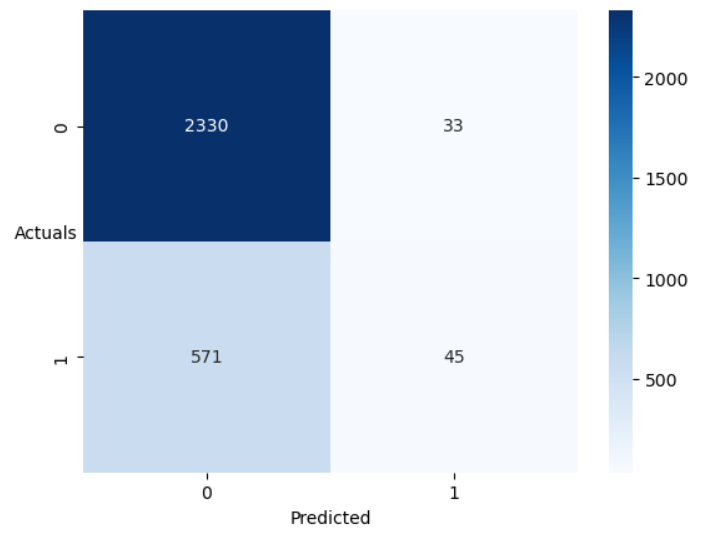
The optimum cutoff is defined by Opt cutoff = Max (tpr – fpr)

For Model 20, which was the Logistic Regression has the cutoff value 0.27

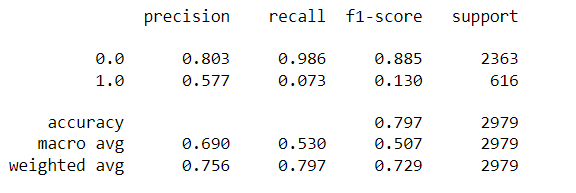
# 6. Checking the accuracy of the model using confusion matrix for training set

**Confusion Matrix for training set:**

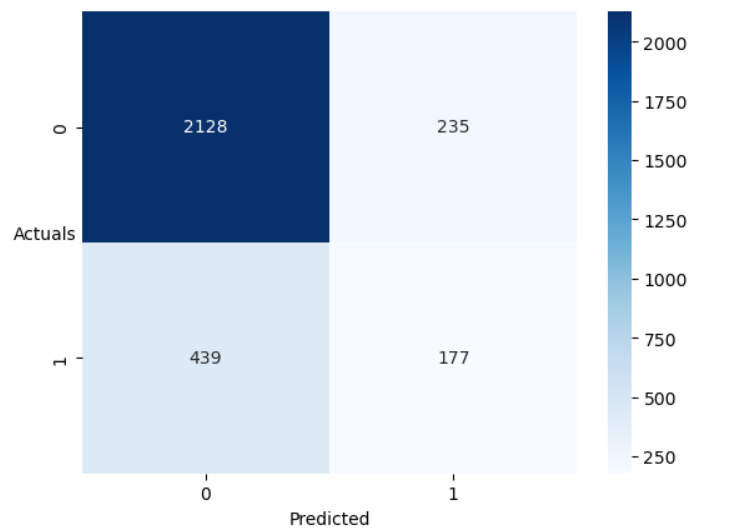
Accuracy = (TP+TN)/(TP+TN+FP+FN) = (2330+45)/(2330+45+33+571) = 2375/2979 = 0.797



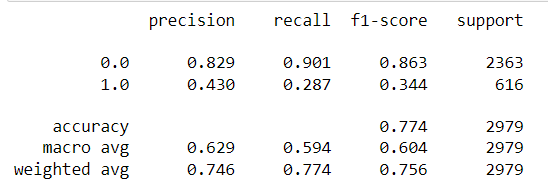
**Classification Report for training set:**



**Confusion Matrix for training set after choosing the optimum cut-off:**



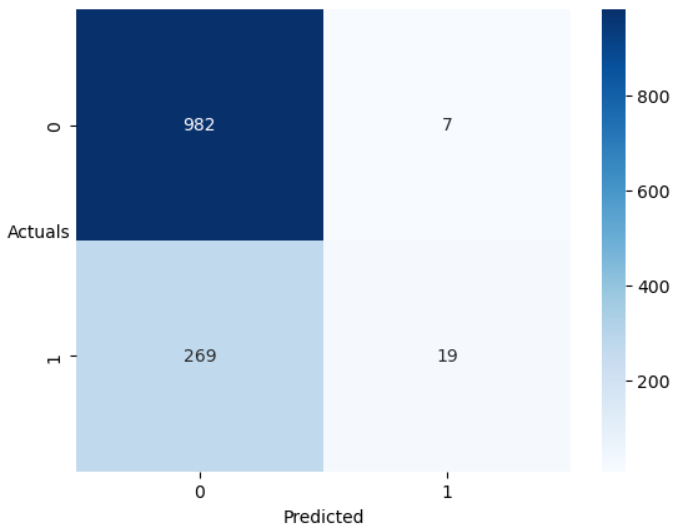
**Classification Report for training set after choosing optimum cu-off:**



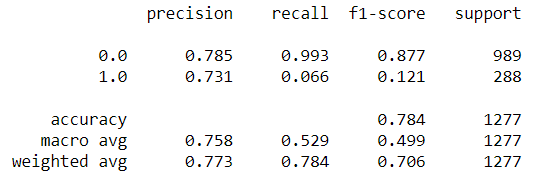
# 7. Checking the accuracy of the model using confusion matrix for test set

**Confusion Matrix for test set:**

Accuracy = (TP+TN)/(TP+TN+FP+FN) = 0.784

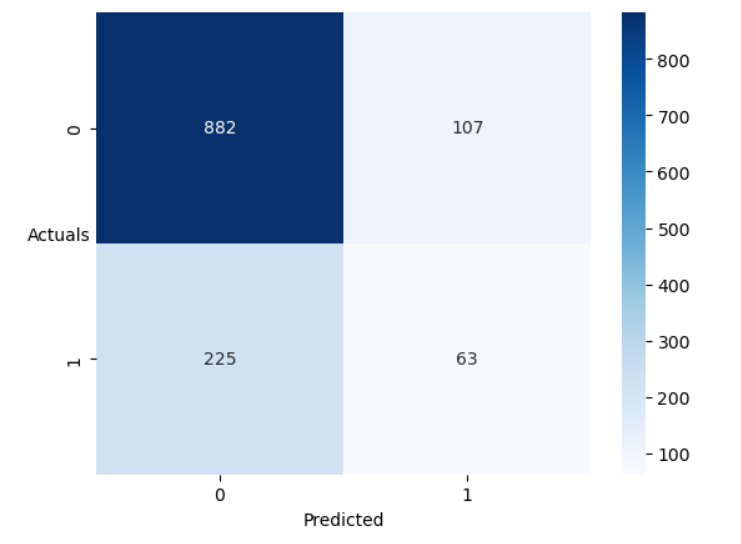


**Classification Report for test set:**

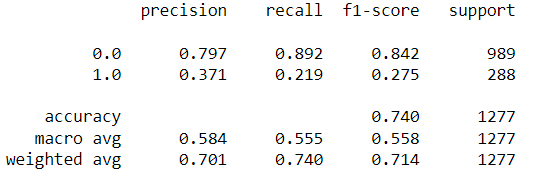


**Confusion Matrix for training set after choosing the optimum cut-off:**

Accuracy = (TP+TN)/(TP+TN+FP+FN) = 0.740

****

**Classification Report for training set after choosing optimum cu-off:**

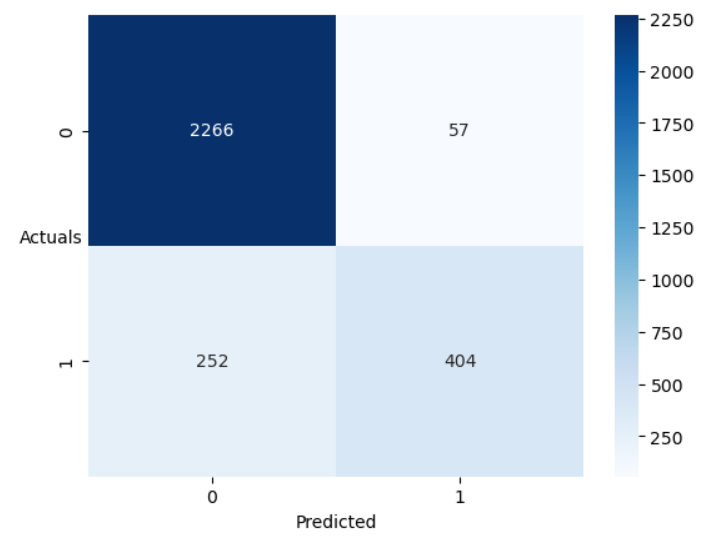


# 8. Build a Random Forest Model on Train Dataset. Also showcase your model building approach

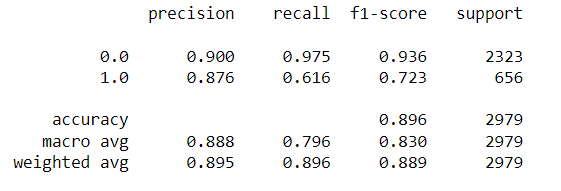
We have built the 8 different models using the Random forest and we have used the dataset that is not treated with the outliers because the random forest are robust to the outliers

## 1.Random Forest model with default parameters

**Confusion matrix on train dataset:**

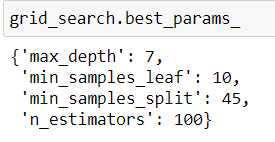


**Classification Report on train dataset:**

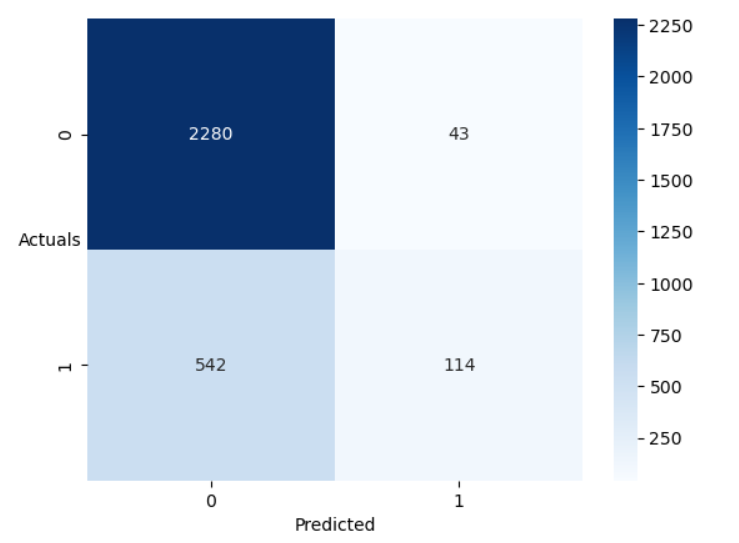


## 2.Random Forest model with grid search parameters

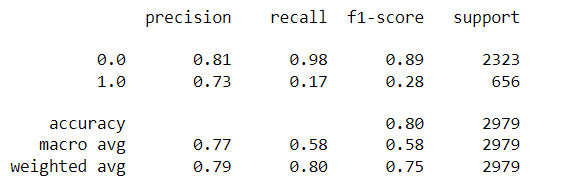
**Best Parameters:**



**Confusion matrix on train dataset:**



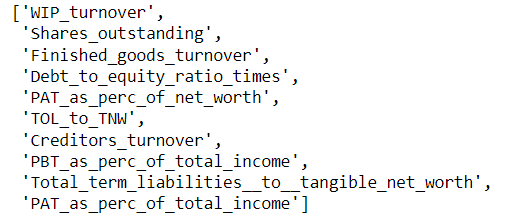
**Classification Report on train dataset:**

****

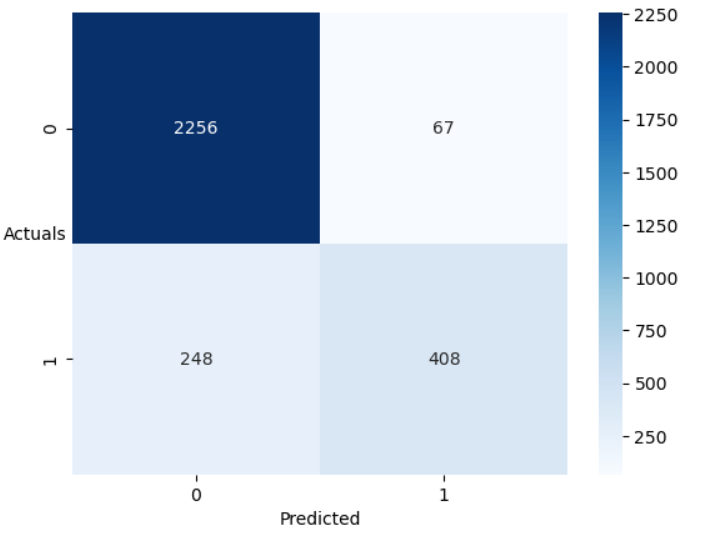
## 3.Random Forest model with feature selection

We have used the ExtraTreesClassifier to select the top 10 features of the dataset

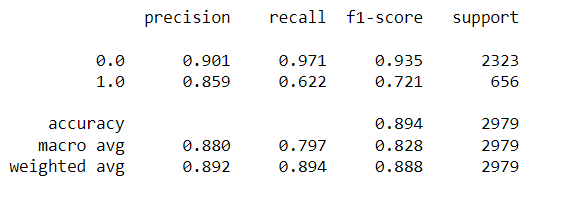
**Best\_features:**



**Confusion matrix on train dataset:**

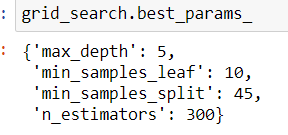


**Classification Report on train dataset**

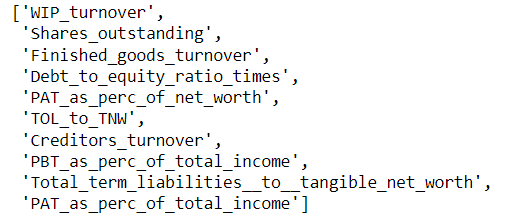


## 4.Random Forest model with feature selection with grid search

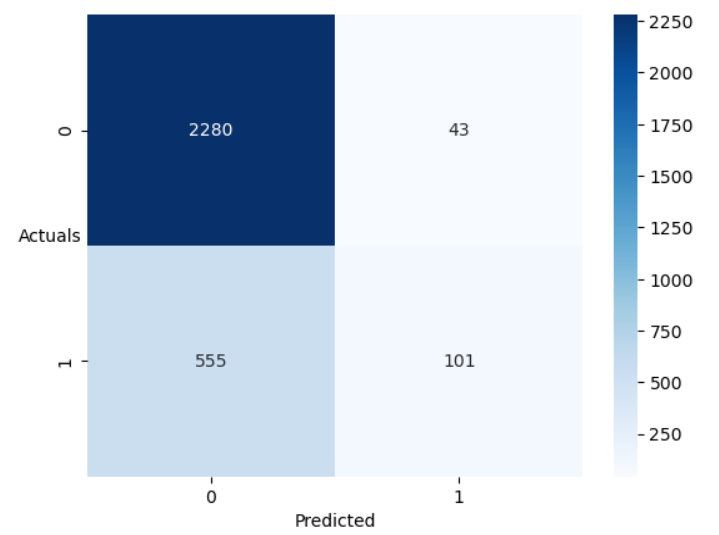
**Best Parameters:**

****

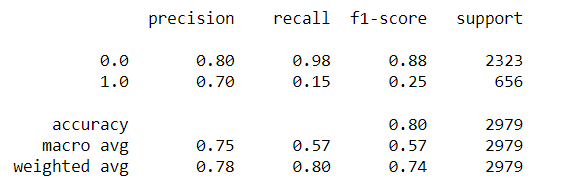
**Best Features:**



**Confusion matrix on train dataset:**

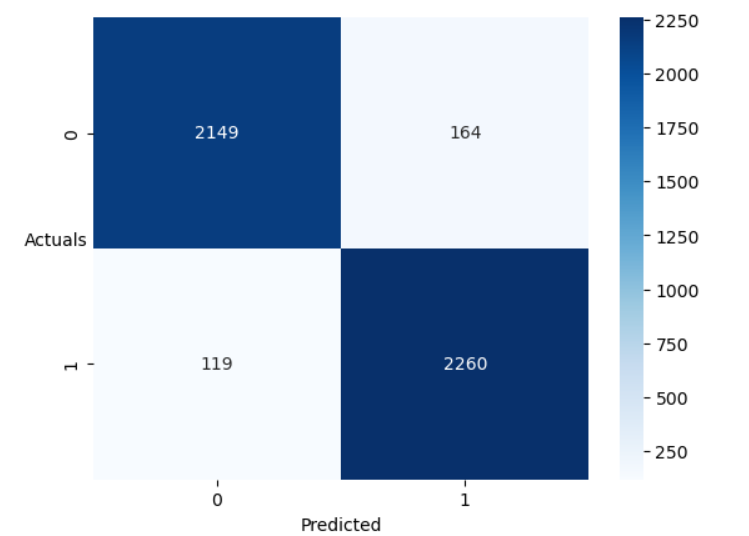
****

**Classification Report on train dataset**

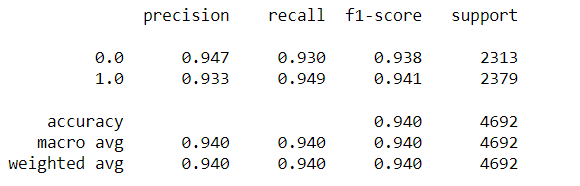
****

## 5.Random Forest model with resampled using smote technique

**Confusion matrix on train dataset:**

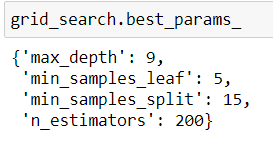
****

**Classification Report on train dataset:**

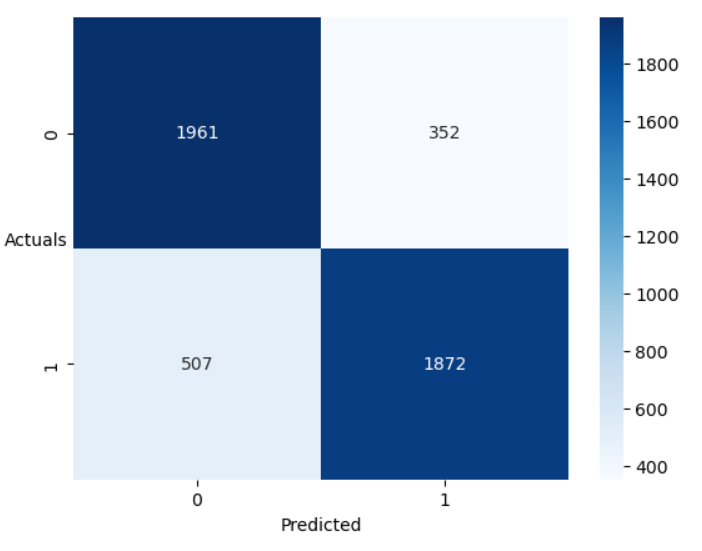
****

## 6.Random Forest model with resampled using smote technique with grid search

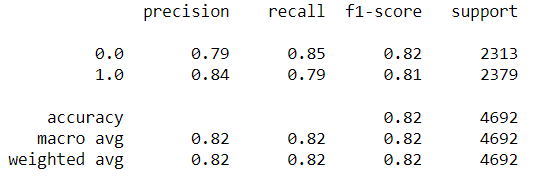
**Best Parameters:**

****

**Confusion matrix on train dataset:**

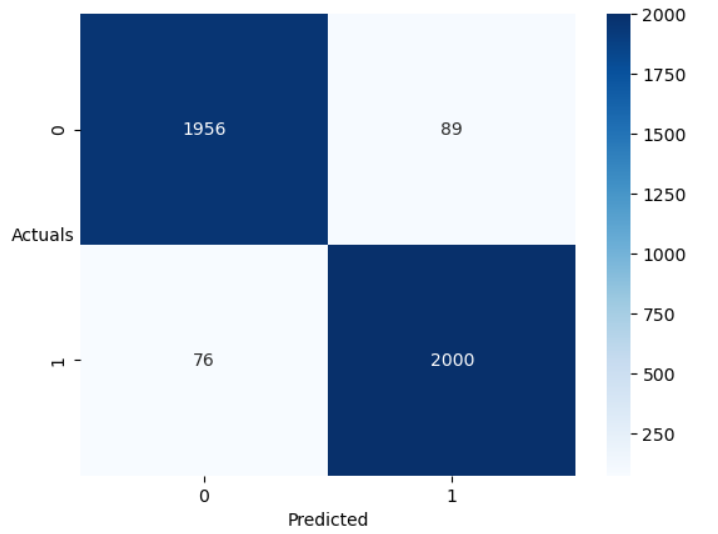
****

**Classification Report on train dataset:**

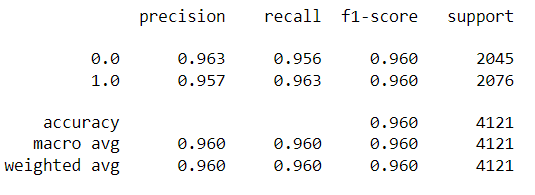
****

## 7.Random Forest model with resampled using smote tomek technique

**Confusion matrix on train dataset:**

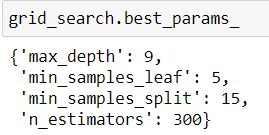
****

**Classification Report on train dataset:**

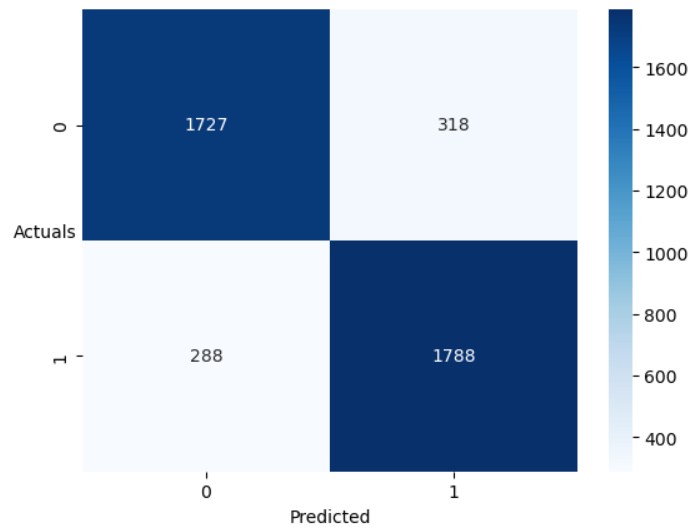
****

## 8.Random Forest model with resampled using smote tomek technique with grid search

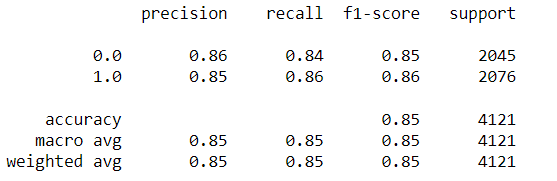
**Best Parameters:**

****

**Confusion matrix on train dataset:**

****

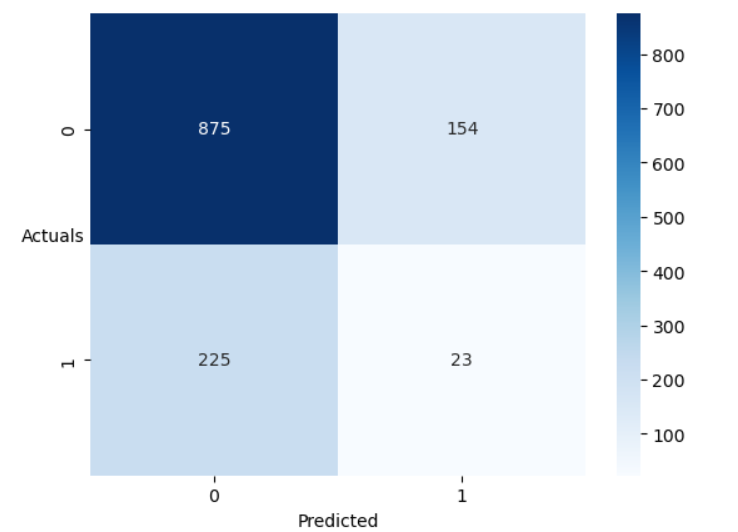
**Classification Report on train dataset:**

****

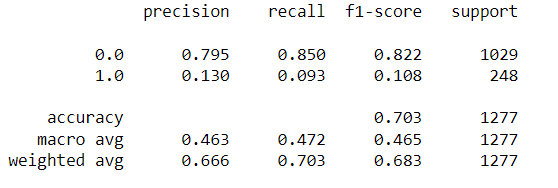
# 9. Validate the Random Forest Model on test Dataset and state the performance matrices. Also state interpretation from the model

## 1.Random Forest model with default parameters

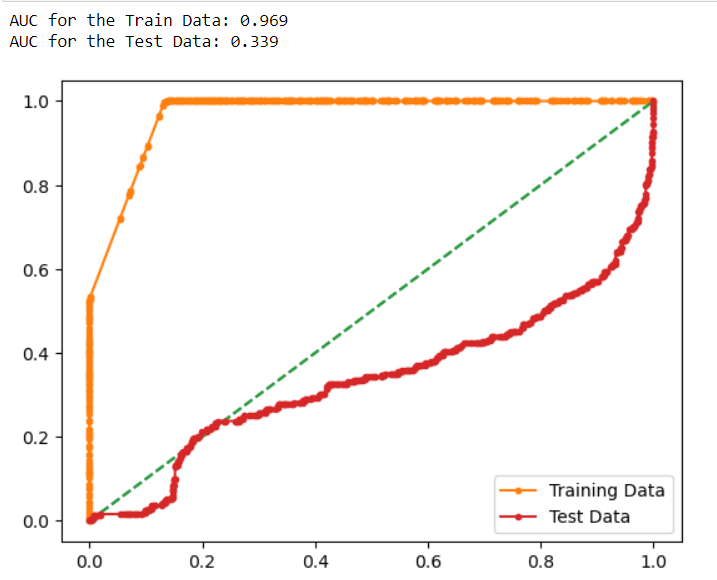
**Confusion matrix on test dataset:**



**Classification Report on test dataset:**

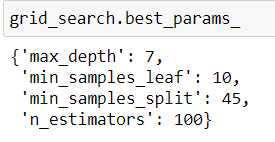
****

**ROC-AUC Curve for both train and test dataset**

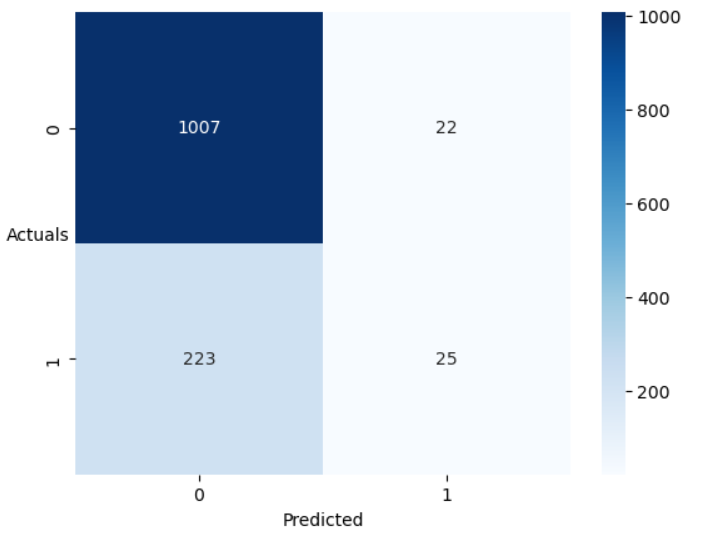
****

## 2.Random Forest model with grid search parameters

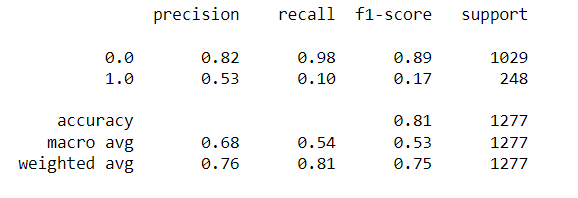
**Best Parameters:**



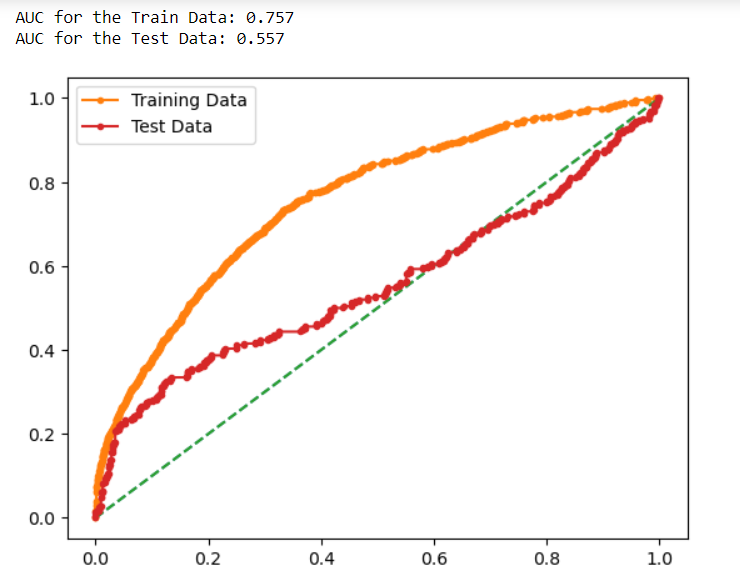
**Confusion matrix on test dataset:**



**Classification Report on test dataset:**

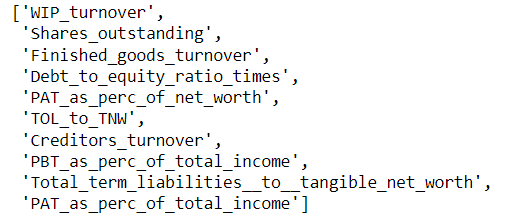
****

**ROC-AUC Curve for both train and test dataset**

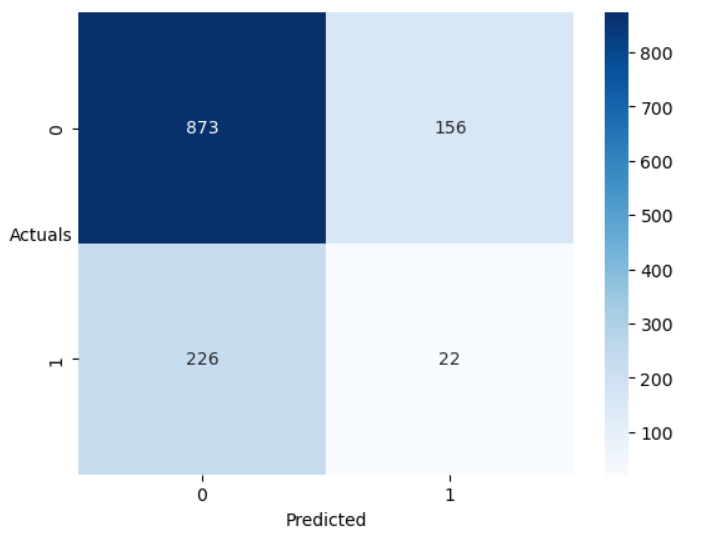


## 3.Random Forest model with feature selection

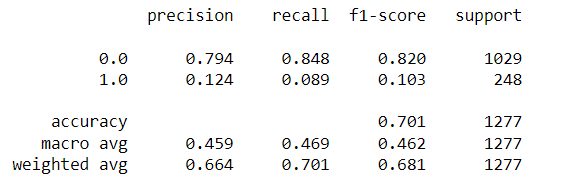
Best\_features:



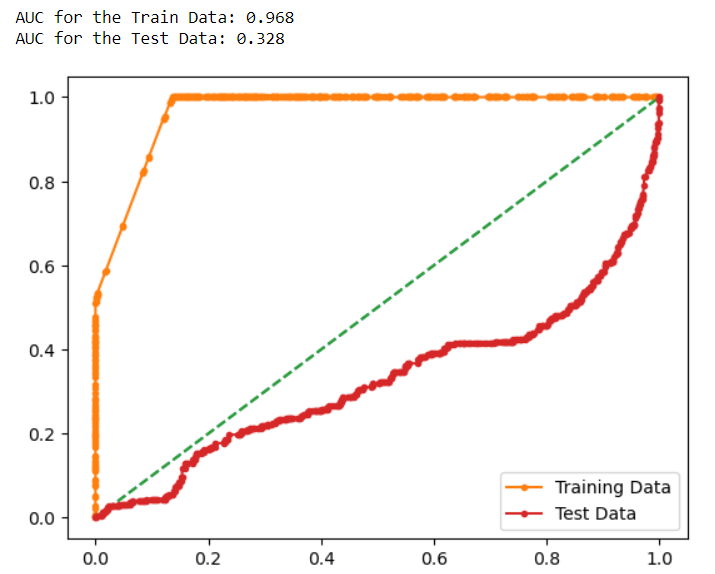
**Confusion matrix on test dataset:**

****

**Classification Report on test dataset:**

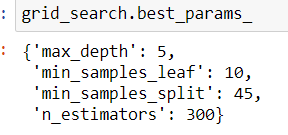
****

**ROC-AUC Curve for both train and test dataset**

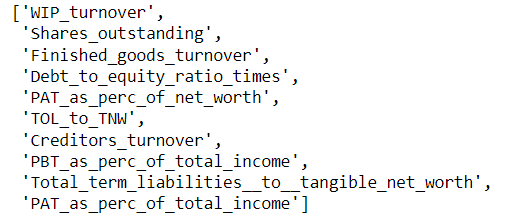
****

## 4.Random Forest model with feature selection with grid search

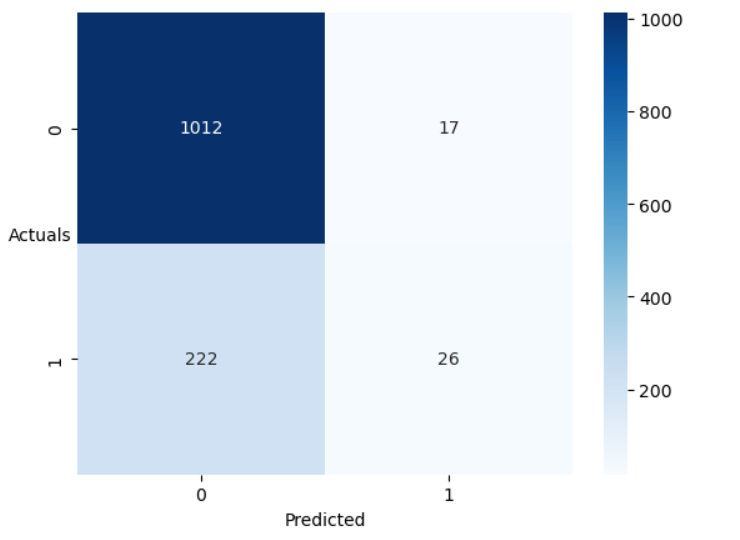
**Best Parameters:**

****

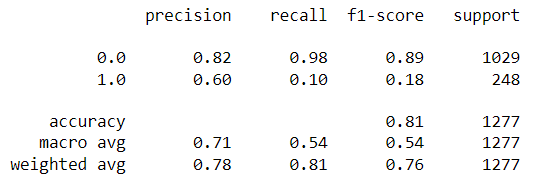
**Best Features:**



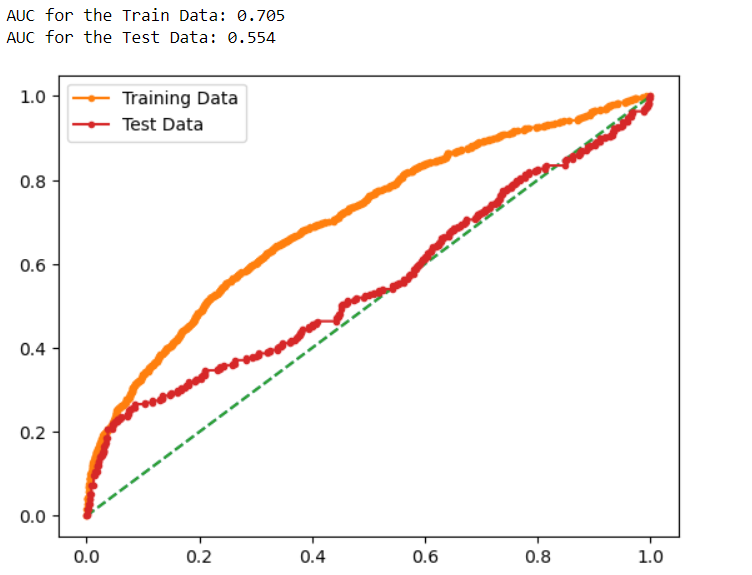
**Confusion matrix on test dataset:**

****

**Classification Report on test dataset:**

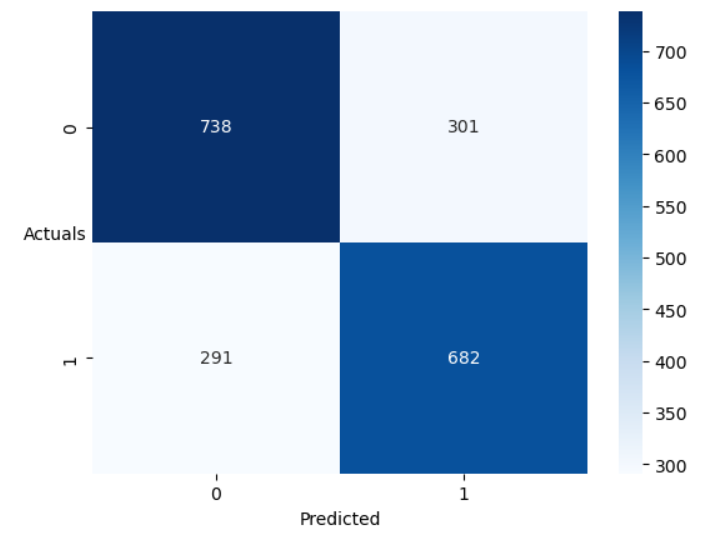
****

**ROC-AUC Curve for both train and test dataset**

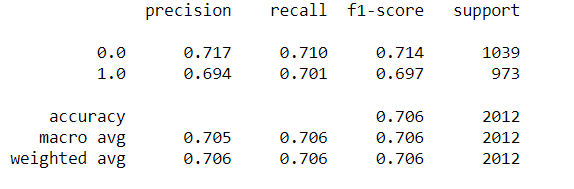
****

## 5.Random Forest model with resampled using smote technique

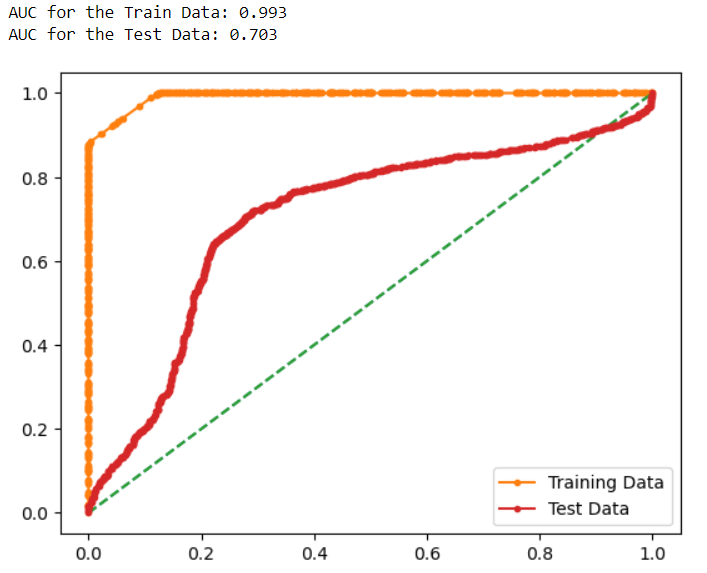
**Confusion matrix on test dataset:**

****

**Classification Report on test dataset:**

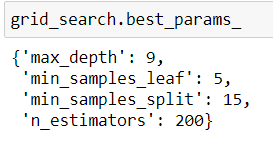
****

**ROC-AUC Curve for both train and test dataset:**

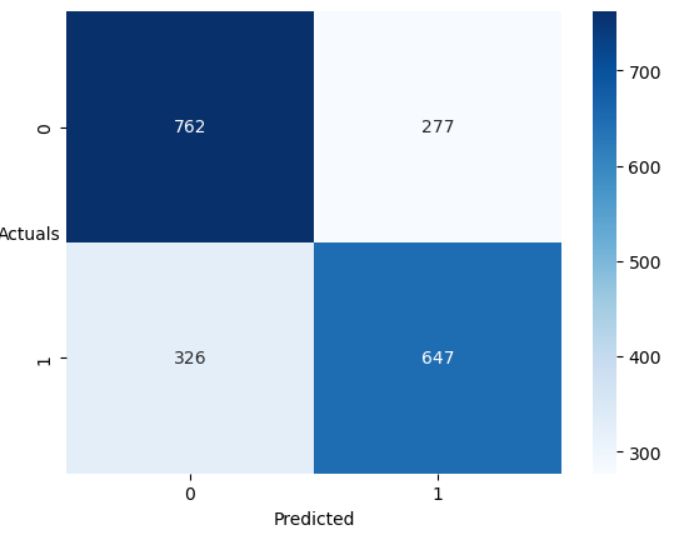
****

## 6.Random Forest model with resampled using smote technique with grid search

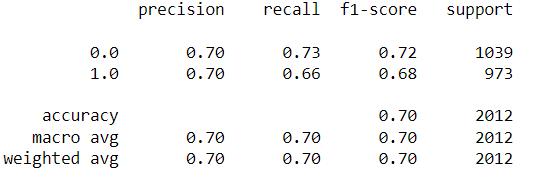
**Best Parameters:**

****

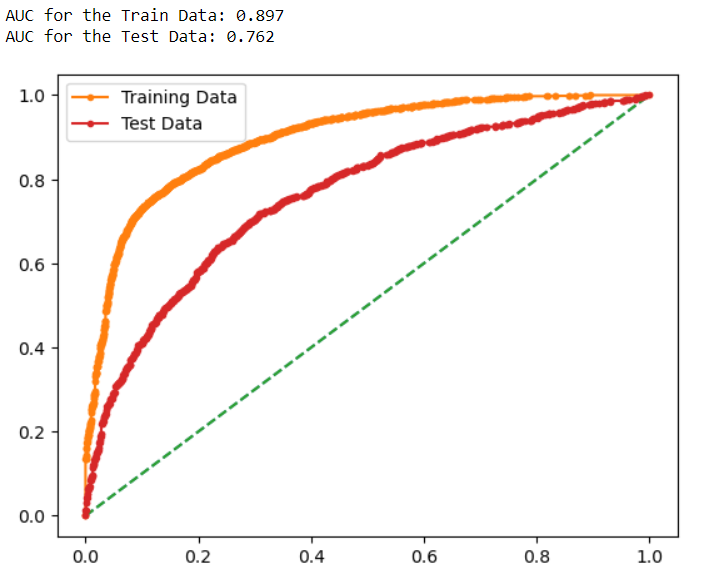
**Confusion matrix on test dataset:**

****

**Classification Report on test dataset:**

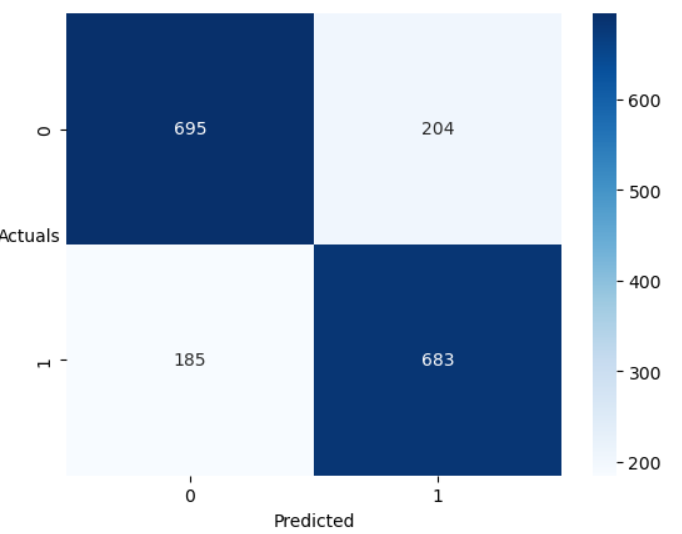
****

**ROC-AUC Curve for both train and test dataset:**

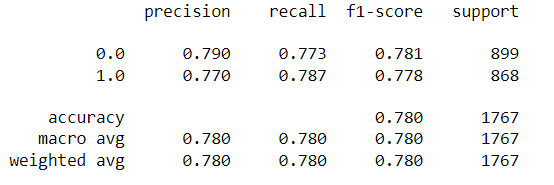
****

## 7.Random Forest model with resampled using smote tomek technique

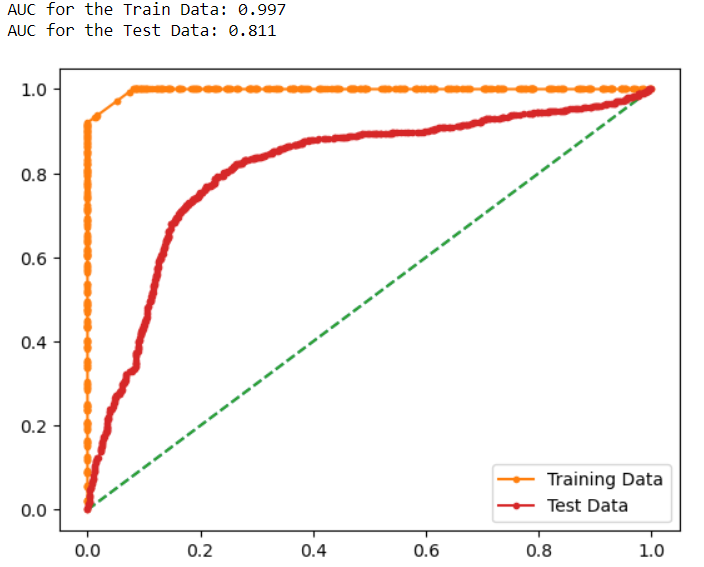
**Confusion matrix on test dataset:**

****

**Classification Report on test dataset:**

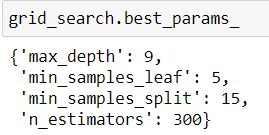
****

**ROC-AUC Curve for both train and test dataset:**

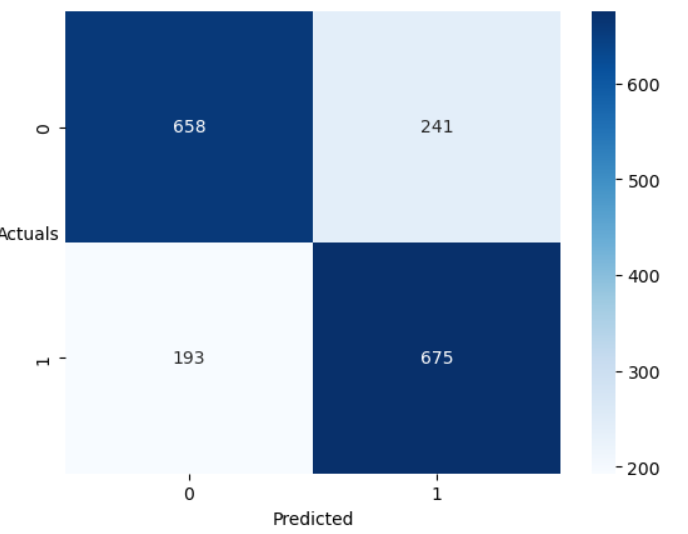
****

## 8.Random Forest model with resampled using smote tomek technique with grid search

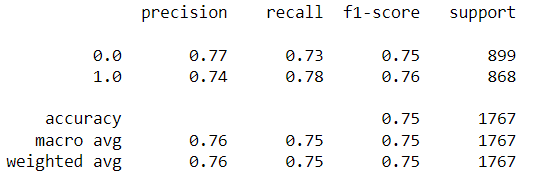
**Best Parameters:**

****

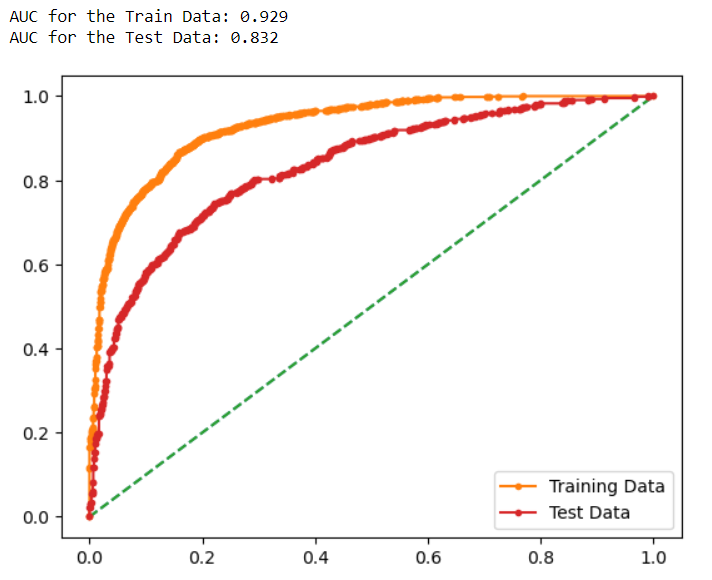
**Confusion matrix on test dataset:**

****

**Classification Report on test dataset:**

****

**ROC-AUC Curve for both train and test dataset:**

****

# 10. Comparison of Random Forest model with logistic regression model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model Name** | **Accuarcy** | **Precision** | **Recall** | **f1** | **Roc\_Auc\_score** |
| **Random Forest with smote-tomek and grid search** | 0.75 | 0.74 | 0.78 | 0.76 | 0.832 |
| **Random forest with smote-tomek** | 0.78 | 0.77 | 0.78 | 0.77 | 0.811 |
| **Random forest with smote and grid search** | 0.70 | 0.70 | 0.66 | 0.68 | 0.762 |
| **Random forest with smote** | 0.70 | 0.69 | 0.70 | 0.69 | 0.700 |
| **Random forest with feature selection and grid search** | 0.81 | 0.60 | 0.10 | 0.18 | 0.554 |
| **Random forest with feature selection** | 0.70 | 0.12 | 0.09 | 0.10 | 0.328 |
| **Random forest with grid search** | 0.81 | 0.53 | 0.10 | 0.17 | 0.557 |
| **Random forest with default** | 0.70 | 0.13 | 0.09 | 0.10 | 0.339 |
| **Logistic regression** | 0.78 | 0.73 | 0.06 | 0.12 | 0.539 |
| **Logistic regression with threshold 0.26** | 0.74 | 0.37 | 0.22 | 0.27 | 0.539 |

# 11. Conclusion and Recommendations

* For the logistic regression we have built, we have very poor recall values and by changing the threshold to the optimum value the recall is increased a bit and the precision is decreased quite significantly and as there always precision-recall trade off exists and we need to choose which is best, but the model has very poor values
* We have built the Random Forest Classifier with different sampling techniques and feature selection and grid search out of these smote tomek with grid seach cv is emerged as the best fitting model with good Roc\_Auc\_score and recall values, so this is our final model which we need to deploy in the production
* The Random Forest Classifier with out any resampling techniques has shown very poor values so we are cannot be used for the production