|  |  |
| --- | --- |
| A picture of a winding road and trees  Finance and risk analytics  Project | **Created by-**  **SHUBHASREE SARKAR** |

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr No** | **Topic: Credit Default Data** | **Page No** |
| **1** | **Problem Statement** | **4** |
| **2** | **Data Dictionary** | **5-6** |
| **3** | **Scaling** | **7-8** |
| **4** | **Data Class Imbalance** | **9** |
| **5** | **Performance metrics** | **10** |
| **6** | **1.8 Build a Random Forest Model on Train Dataset. Also showcase your model building approach** | **11-14** |
| **7** | **1.9 Validate the Random Forest Model on test Dataset and state the performance matrices. Also state interpretation from the model** | **15-18** |
| **8** | **1.10 Build a LDA Model on Train Dataset. Also showcase your model building approach** | **19-22** |
|  | **1.11 Validate the LDA Model on test Dataset and state the performance matrices. Also state interpretation from the model** | **23-26** |
| **9** | **1.12 Compare the performances of Logistics, Radom Forest and LDA models (include ROC Curve)** | **27-31** |
| **10** | **1.13 State Recommendations from the above models** | **32-33** |
| **11** | **Topic: Market Risk Data** |  |
| **12** | **Problem Statement** | **34** |
| **13** | **2.1 Draw Stock Price Graph(Stock Price vs Time) for any 2 given stocks with inference** | **35-36** |
| **14** | **2.2 Calculate Returns for all stocks with inference** | **37** |
| **15** | **2.3 Calculate Stock Means and Standard Deviation for all stocks with inference** | **38** |
| **16** | **2.4 Draw a plot of Stock Means vs Standard Deviation and state your inference** | **39** |
| **17** | **2.5 Conclusion and Recommendations** | **40** |
| **18** | **Reference** | **41** |

**Topic: Credit Default Data**

|  |  |
| --- | --- |
| **Description** | **Page no.** |
| **1.Output from Base Random Forest Model on Train dataset (Without Grid Search and Smote)** | **11** |
| **2. Output from Random Forest Model on Train dataset(With Grid Search and without Smote)** | **12** |
| **3. Output from Random Forest Model on Train dataset (With Grid Search and Smote)** | **13** |
| **1.      Output from Random Forest Model on Test dataset (Without Grid Search and Smote)** | **15** |
| **2. Output from Random Forest Model on Test dataset (With Grid Search and without Smote)** | **16** |
| **3. Output from Random Forest Model on Test dataset (With Grid Search and Smote)** | **17** |
| **1.Output from Base LDA Model on Train dataset (Without Grid Search and Smote)** | **19** |
| **2.Output from Base LDA Model on Train dataset (With Grid Search and without Smote)** | **20** |
| **3.Output from Base LDA Model on Train dataset (With Grid Search and Smote)** | **21** |
| **1.       Output from LDA Model on Test dataset (Without Grid Search and Smote)** | **23** |
| **2. Output from LDA Model on Test dataset (With Grid Search and without Smote)** | **24** |
| **3. Output from LDA Model on Test dataset (With Grid Search and Smote)** | **25** |

**List of Figures**

|  |  |
| --- | --- |
| **Description** | **Page no.** |
| **#Check with the output of scaled data-Trained data- Table** | **7** |
| **#Check with the output of scaled data-Test data- Table** | **7** |
| **#Train and Test accuracies for each model** | **26** |
| **#Train and Test Recall for each model** | **27** |
| **#Train and Test Precision for each model** | **28** |
| **#Train and Test F1 score for each model** | **29** |
| **#Train and Test Area under curve for each model** | **30** |

**List of Tables**

**Topic: Market risk data**

|  |  |
| --- | --- |
| **Description** | **Page no.** |
| **The Stock return output table** | **36** |
| **The Stock mean of return output table** | **37** |
| **The Standard Deviation of return output table** | **37** |
| **Volatility vs Average of Stocks- Tabular representation** | **38** |

**List of Figures**

**List of Tables**

|  |  |
| --- | --- |
| **Description** | **Page no.** |
| **Stock Price graph of Infosys** | **34** |
| **Stock Price graph of Mahindra & Mahindra** | **35** |
| **Volatility vs Average of Stocks- Scatter plot** | **38** |

**Topic: Credit Default Data**

**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.

**Data Dictionary**



**SCALING**

Feature scaling in machine learning is one of the most critical steps during the pre-processing of data before creating a machine learning model. Scaling can make a difference between a weak machine learning model and a better one.

Machine learning algorithm just sees number — if there is a vast difference in the range say few ranging in thousands and few ranging in the tens, and it makes the underlying assumption that higher ranging numbers have superiority of some sort. So these more significant number starts playing a more decisive role while training the model.

### **Gradient Descent Based Algorithms**

**Machine learning algorithms like linear regression, logistic regression, neural network, etc. that use gradient descent as an optimization technique require data to be scaled.**

### **Distance-Based Algorithms**

Distance algorithms like **KNN, K-means, and SVM** are **most affected** by the range of features. This is because behind the scenes **they are using distances between data points to determine their similarity.**

### **Tree-Based Algorithms**

Tree-based algorithms, on the other hand, are fairly **insensitive** to the scale of the features.

* Algorithms like **Linear Discriminant Analysis (LDA), Naive Bayes is** by design equipped to handle this and give weights to the features accordingly. Performing features scaling in these algorithms may not have much effect.
* Scaling is critical while performing**Principal Component Analysis (PCA)**. PCA tries to get the features with maximum variance, and the variance is high for high magnitude features and skews the PCA towards high magnitude features.

**Type of Scaling Technique used** -

We will scale the data based on Z-Score method or Standard Scalar in SkLearn

• Standard Scalar standardizes a feature by subtracting the mean and then scaling to unit variance. Unit variance means dividing all the values by the standard deviation.

• Standard Scalar results in a distribution with a standard deviation equal to 1. The variance is equal to 1 also, because variance = standard deviation squared. And 1 squared = 1.

• Standard Scalar makes the mean of the distribution 0. About 68% of the values will lie be between -1 and 1.

Standard Scalar normalizes the data using the formula (x-mean)/standard deviation.

**Z = (value -mean)/standard deviation**

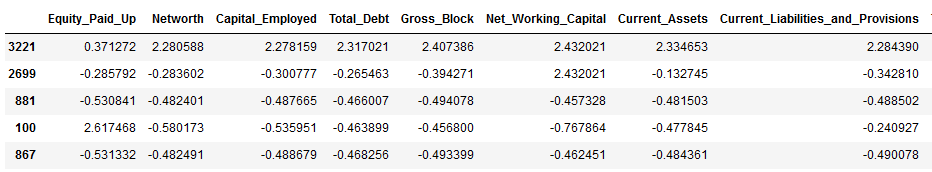
**Why it is to be used?**

### If your variables are of incomparable units (e.g. height in cm and weight in kg) then you should standardize variables, of course. Even if variables are of the same units but show quite different variances it is still a good idea to standardize **Gradient Descent Based Algorithms** and **Distance-Based Algorithms.**

**Explanation with respect to the Variables given**

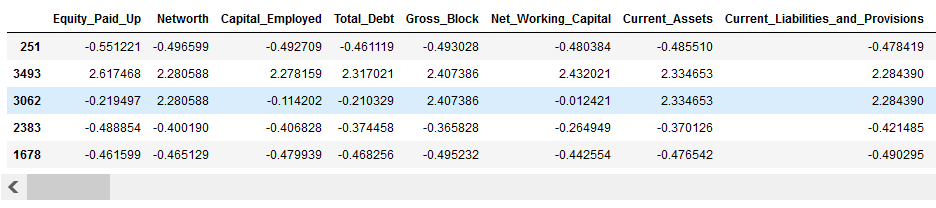
As per the data provided, standard scaling is required as all the different variables are provided in different units, for example, Age is a continuous variable in years. Rest all are categorical in nature. As there are differences in units, hence other values expressed in higher units will outweigh the variables in lower units and can give varied results. This is why scaling is important, and Standard scalar, as mentioned above, normalise the data points with mean 0 and standard deviation 1.

***#Check with the output of scaled data-Trained data- Table***



The above table shows the columns with scaled value for the Continuous data types in the trained dataset

***#Check with the output of scaled data-Test data- Table***



The above table shows the columns with scaled value for the Continuous data types in the test dataset

In this particular dataset, as can be observed from the above tables, we have used Standard scaler to scale all the continuous variable that are present

**DATA CLASS IMBALANCE**

We can observe from Page 27, where the Train and Test data containing the Target Variable –Vote is shown with the Class percentage.

It can be concluded, that there has been high percentage of difference in data between the two classes in Votes, which has separated the dataset into Major and Minor classes based on the weightage.

**What is Class Imbalance?**

When observation in one class is higher than the observation in other classes then there exists a class imbalance.

Class Imbalance is a common problem in machine learning, especially in classification problems. Imbalance data can hamper our model accuracy big time.

## **The Problem with Class Imbalance**

Most machine learning algorithms work best when the number of samples in each class are about equal. This is because most algorithms are designed to maximize accuracy and reduce errors.

However, if the data set in imbalance then In such cases, you get a pretty high accuracy just by predicting the **majority class**, but you fail to capture the **minority class**, which is most often the point of creating the model in the first place

**Type of Ways to Handle Class Imbalance**

## Random Under-Sampling

## Random Over-Sampling

## Synthetic Minority Oversampling Technique (SMOTE)

Here in this problem we have used SMOTE to reduce Class Imbalance

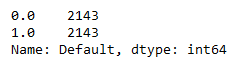
## **SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE (SMOTE)**

This technique generates synthetic data for the minority class.

SMOTE (Synthetic Minority Oversampling Technique) works by randomly picking a point from the minority class and computing the k-nearest neighbours for this point.

The **synthetic points are added** between the chosen point and its neighbours.

After applying SMOTE, the train data set is now balanced.



***Performance Metrices used-***

***ACCURACY***

It is a part of metrices derived from confusion matrix which is basically a **NxN** matrix, where **N** is the **number of classes to be predicted**

It is the **proportion** of the total number of **predictions** that were **correct**.

It is easily suited for **binary** as well as a **multiclass** **classification** **problem** which are **well** **balanced** and **not skewed** or **No class imbalance**.



***PRECISION***

Similar to the Accuracy, it is a metric derived from confusion matrix

**Positive Predictive Value or Precision** is also defined as the proportion of positive cases that were correctly identified.

In other words, it determines the proportion of **predicted Positives**which is truly Positive

Precision is a valid choice of evaluation metric when we want to be very sure of our prediction.



***RECALL***

Similar to the Precision, it is a metric derived from confusion matrix.

**Sensitivity or Recall** is also defined as the proportion of actual positive cases which are correctly identified.

In other words, it determines the proportion of **actual Positives**is correctly classified

Recall is a valid choice of evaluation metric when we want to capture as many positives as possible.

***ROC AUC Score***

This is again one of the popular metrics used in the industry.

The ROC (**Receiver operating characteristic**) curve is the plot between **sensitivity** and (**1- specificity**). (**1- specificity**) is also known as **false positive rate** and **sensitivity** is also known as **True Positive rate**.

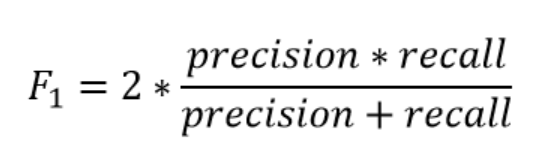
The biggest advantage of using ROC curve is that it is **independent** of the **change in proportion of responders.**

***F1 SCORE***

Similar to the Precision and Recall, it is a metric derived from confusion matrix.

The **F1 score** is a **number between 0** and **1** and is the **harmonic mean** of **precision** and **recall** values for a **classification** problem.

F1 score sort of maintains a balance between the precision and recall for your classifier. If your precision is low, the F1 is low and if the recall is low again your F1 score is low.



**1.8 Build a Random Forest Model on Train Dataset. Also showcase your model building approach**

**Random Forest**

Random forest is a **Supervised Machine Learning Algorithm** that is **used widely in Classification and Regression problems**. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

One of the most important features of the Random Forest Algorithm is that it can handle the data set containing **continuous variables** as in the case of regression and **categorical variables** as in the case of classification. It performs better results for classification problems.

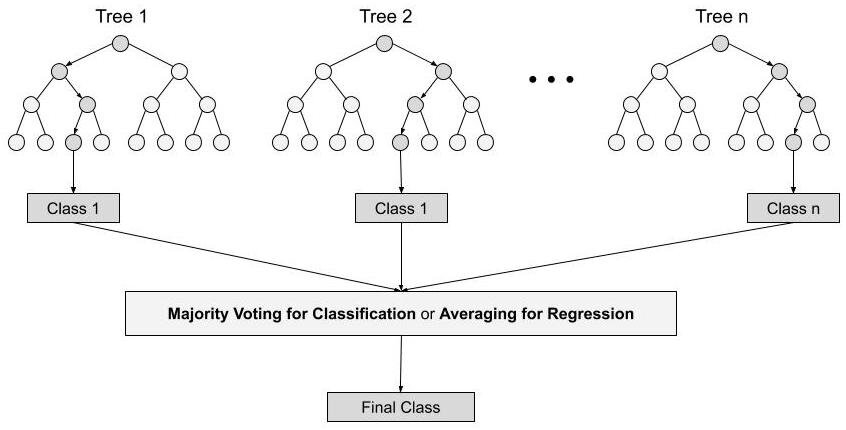
**Steps involved in random forest algorithm:**

Step 1: In Random forest n number of random records are taken from the data set having k number of records.

Step 2: Individual decision trees are constructed for each sample.

Step 3: Each decision tree will generate an output.

Step 4: Final output is considered based on ***Majority Voting or Averaging***for Classification and regression respectively.



# **Pros of Random Forest**

1. Robust to outliers.
2. Works well with non-linear data.
3. Lower risk of overfitting.
4. Runs efficiently on a large dataset.
5. Better accuracy than other classification algorithms.

**Cons of Random Forest**

* Random forests are found to be biased while dealing with categorical variables.
* Slow Training.
* Not suitable for linear methods with a lot of sparse features

*#Steps involved*

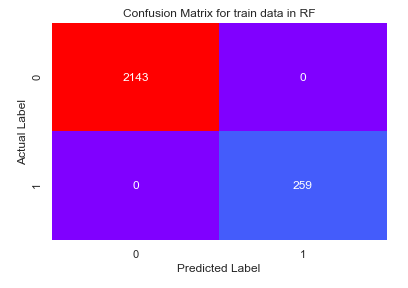
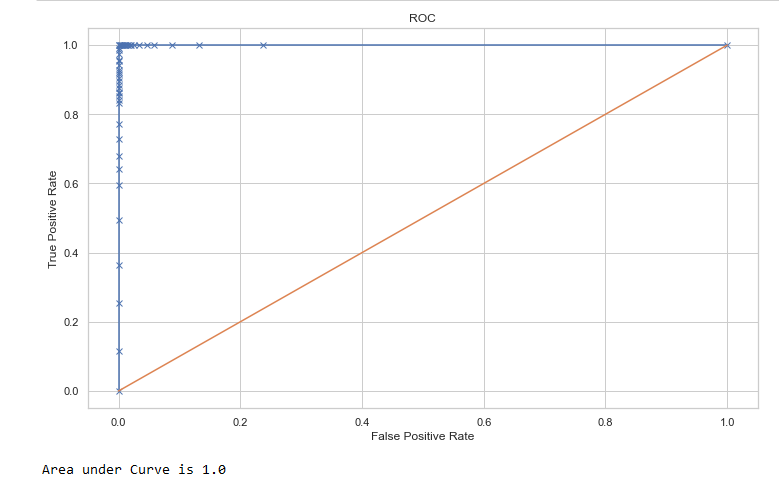
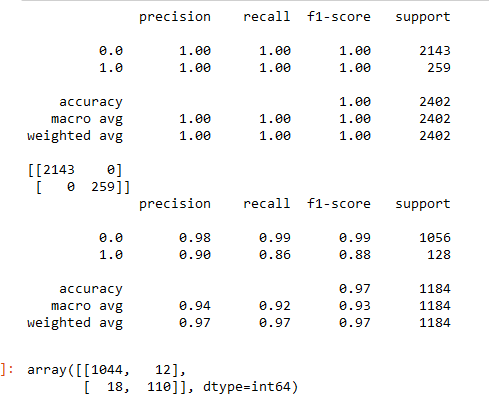
First, we have used the train dataset to create the RandomForestClassifier and to test on the test dataset.

For the Random Forest formation, we have imported the RandomForestClassifier module from the ‘sklearn’ package

With the help of afore mentioned package, Random Forest model is created, in order to fit the training data into this model.

A final model is thus being created which is being further used for model performance evaluation.

**1.Output from Base Random Forest Model on Train dataset (Without Grid Search and Smote)**



We can observe from here that performance scores for the base model are 1.00, which states the presence of overfitting within the train dataset.

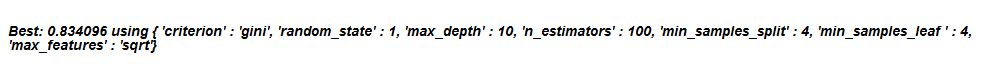
We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 1.00, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model shows no false prediction(false positive or false negative), which states the presence of overfitting within the train dataset.

**Using *GRID Search CV-***

For Random Forest, base estimator would be required to get a tuned model

We know that by using GRID Search CV, one can select the best parameters from the listed hyper parameters.

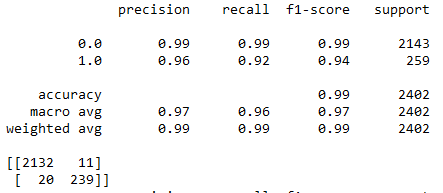


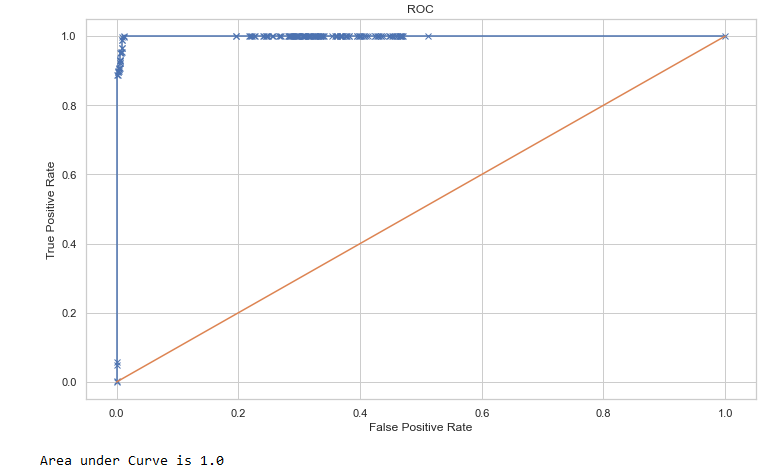
**2. Output from Random Forest Model on Train dataset(With Grid Search and without Smote)**

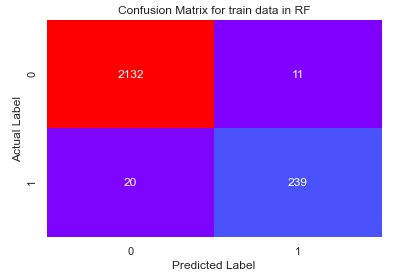
We can observe from here that performance scores for the base model are less than 1.00, which states the impact of overfitting within the train dataset is comparatively less.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 1.00, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model shows less no. of false prediction (false positive or false negative), which states the presence of overfitting within the train dataset.







***Random Forest with SMOTE data***

We have also tried to run the Random Forest with the balanced dataset (SMOTE based).

We also applied Grid Search on the model

But the key metrics based performance changed.

We will compare the performances of the metrics’ later.

The codes for the same is being attached with the Jupyter notebook (Machine Learning)

For the model evaluation and comparison, we have stated the same in the later half of the question, where model performances are being compared

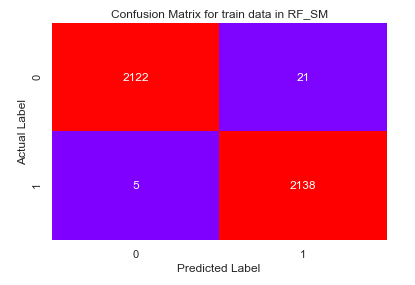
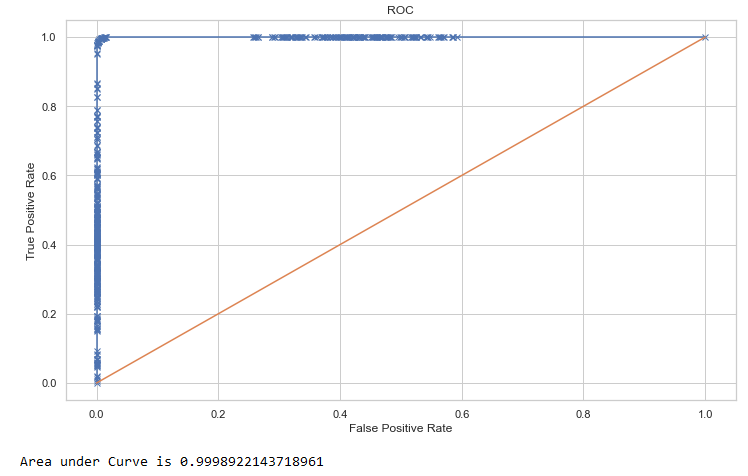
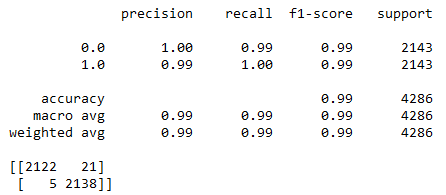
**Overfitting-** This are most prominent while using base model followed by smote in most of the metrices, although, overfitting issues is also marginally present in tuned model.

**3. Output from Random Forest Model on Train dataset (With Grid Search and Smote)**

We can observe from here that performance scores for the base model are less than 1.00, which states the impact of overfitting within the train dataset is comparatively less.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.99989, which states the presence of overfitting within the train dataset is present.

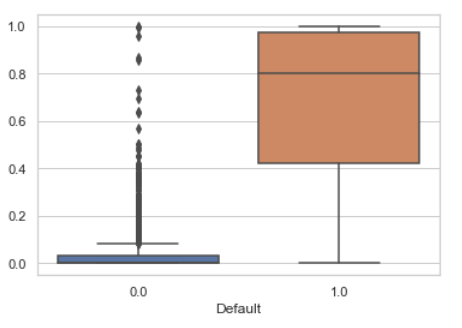
We can observe from here that confusion matrix for this model shows little false prediction(false positive or false negative), which states the impact of overfitting within the train dataset is less here as compared to the base model.



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

The model has been validated on the basis of various parameters taken into consideration.

First on the basis of the predicted values based on the train dataset, we have tried to use a boxplot to differentiate between the Default and Non-Default values .



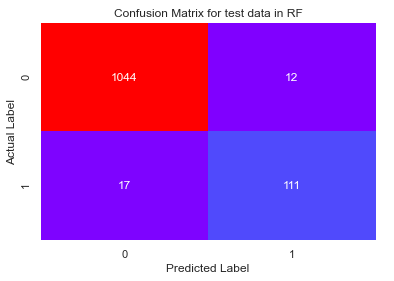
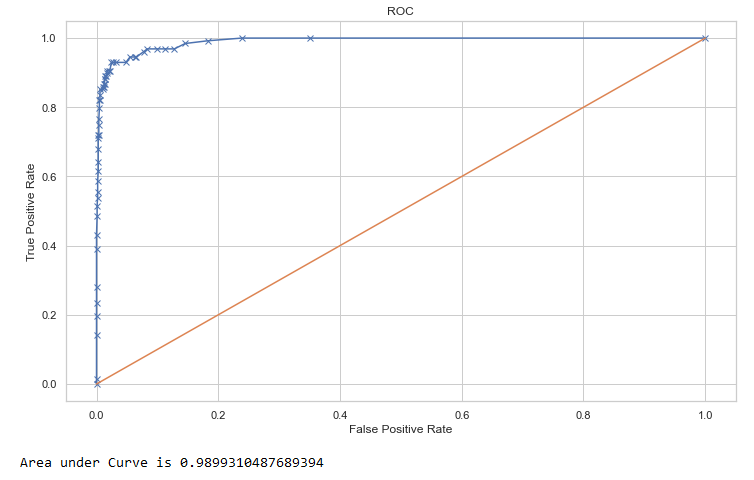
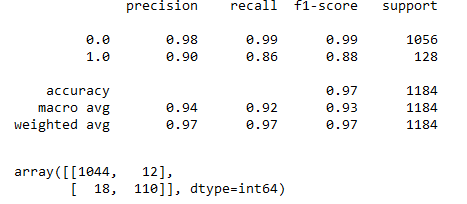
We have applied and validated the above models on the test datasets as well which are shown below

1. **Output from Random Forest Model on Test dataset (Without Grid Search and Smote)**

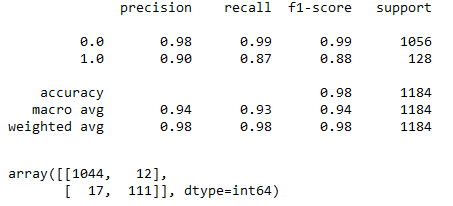
We can observe from here that performance scores for the base model are not 100% efficient, which means the presence of overfitting within the train dataset is has not impacted the results of the test dataset considerably

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.989931, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for this model shows little false prediction(false positive or false negative)



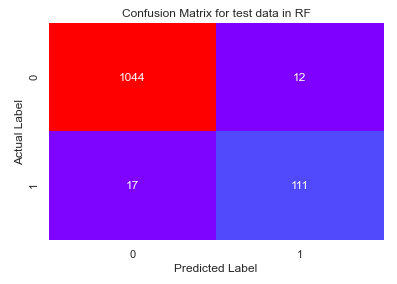
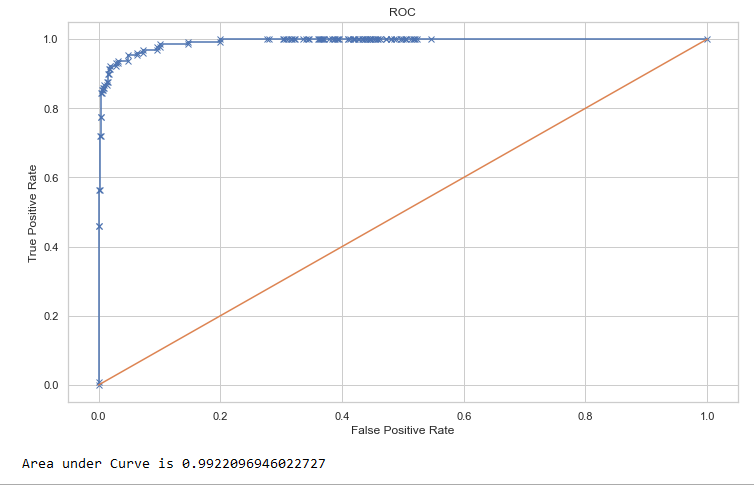
**2. Output from Random Forest Model on Test dataset (With Grid Search and without Smote)**



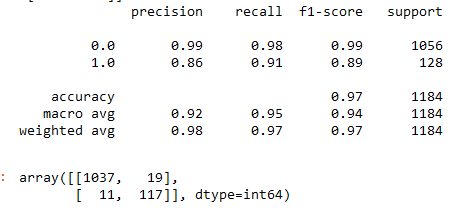
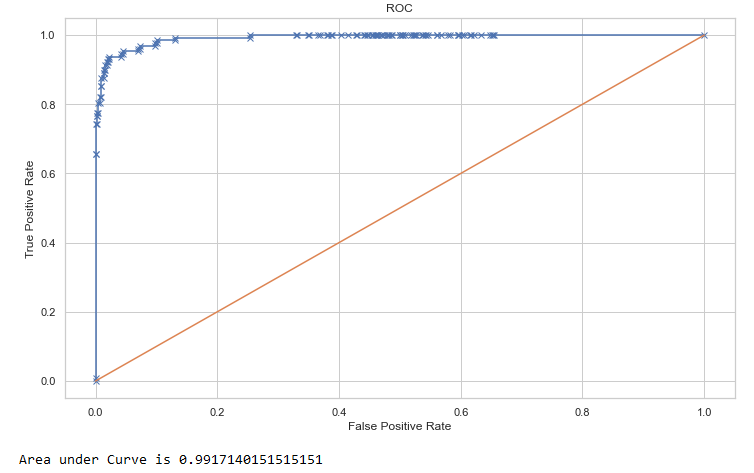
We can observe from here that performance scores for this model are less than 100, which states the impact of overfitting within the train dataset is comparatively less.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.9922, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model shows no false prediction(false positive or false negative), which states the presence of overfitting within the train dataset.



**3. Output from Random Forest Model on Test dataset (With Grid Search and Smote)**

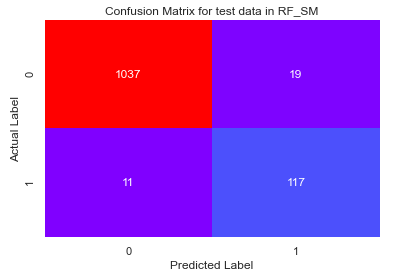


We can observe from here that performance scores for this model are less than 100%

1. Recall for default values-
2. Precision for default values
3. Accuracy -

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.99171, which is almost similar to train values.

We can observe from here that confusion matrix for this model shows little false prediction(false positive or false negative). Overall confusion matrix shows that True positives and negatives are being correctly determined in majority, but it has little no. of false positives as well as negatives



**1.10 Build a LDA Model on Train Dataset. Also showcase your model building approach**

**LDA**

**Linear Discriminant Analysis** or **Normal Discriminant Analysis** or **Discriminant Function Analysis** is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling differences in groups i.e. separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.

For example, we have two classes and we need to separate them efficiently. Classes can have multiple features. Using only a single feature to classify them may result in some overlapping as shown in the below figure. So, we will keep on increasing the number of features for proper classification.

Two criteria are used by LDA to create a new axis:

* Maximize the distance between means of the two classes.
* Minimize the variation within each class.

**Extensions to LDA:**

1. **Quadratic Discriminant Analysis (QDA):** Each class uses its own estimate of variance (or covariance when there are multiple input variables).
2. **Flexible Discriminant Analysis (FDA):** Where non-linear combinations of inputs are used such as splines.
3. **Regularized Discriminant Analysis (RDA):** Introduces regularization into the estimate of the variance (actually covariance), moderating the influence of different variables on LDA.

**Pros of LDA**

* It is simple, fast and portable algorithm. It still beats some algorithms (logistic regression) when its assumptions are met.

**Cons of LDA**

* It requires normal distribution assumption on features/predictors.
* Sometimes not good for few categories variables.

*#Steps involved*

First, we have used the train dataset to create the Linear Discriminants and to test on the test dataset.

For the LDA formation, we have imported the LinearDiscriminantAnalysis module from the ‘sklearn’ package

With the help of afore mentioned package, LDA model is created, in order to fit the training data into this model.

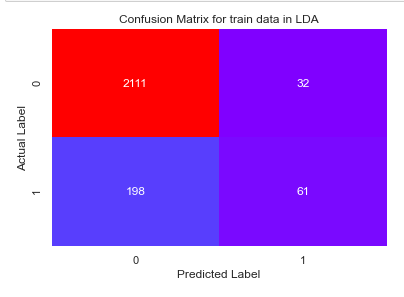
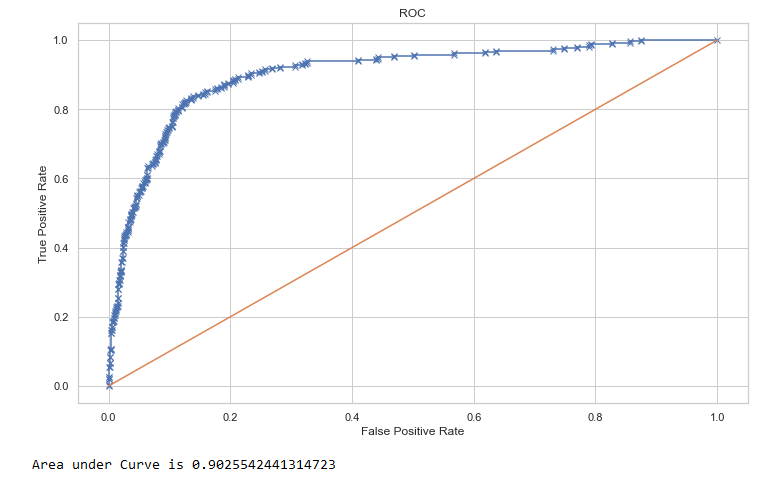
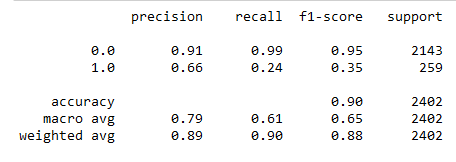
A LDA model is thus being created which is being further used for model performance evaluation.

**1.Output from Base LDA Model on Train dataset (Without Grid Search and Smote)**

We can observe from here that performance scores for the base model on train dataset is unable to work efficiently and correctly for the default values as compared to the Non-Default ones, even though Accuracy is high.

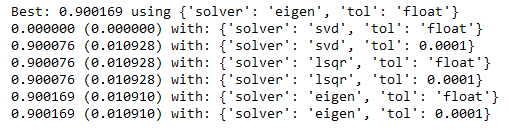
We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.90255, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model on train dataset shows that True positives and negatives are being correctly determined in majority, but it has high no. of false positives as well as negatives



**Using *GRID Search CV-***

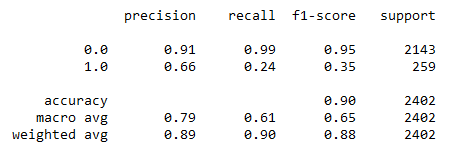
We know that by using GRID Search CV, one can select the best parameters from the listed hyper parameters.



Using the above mentioned GridSearchCV package, we have identified the best parameters and an Optimised Linear Discriminant Analysis model is being built after doing few iterations with the values we have received in each step. But even with this the model couldn’t generate better accuracy, precision and recall. We will discuss about the same in the next questions

**Note – Kindly refer the code file for the steps involved in the LDA formation.**

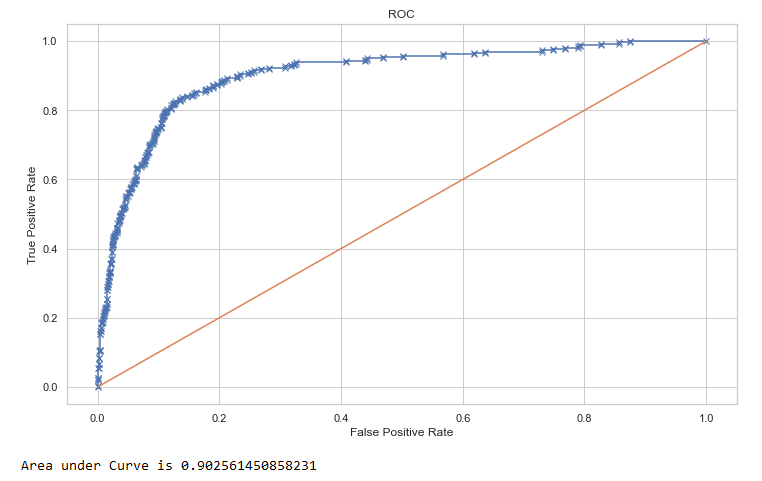
**2.Output from Base LDA Model on Train dataset (With Grid Search and without Smote)**

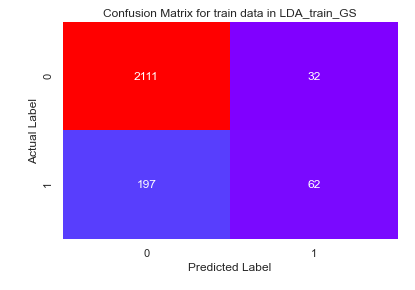


We can observe from here that performance scores for the base model on train dataset is unable to work efficiently and correctly for the default values as compared to the Non-Default ones, even though Accuracy is high.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.90255, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model on train dataset shows that True positives and negatives are being correctly determined in majority, but it has high no. of false positives as well as negatives





***LDA with SMOTE data***

We have also tried to run the LDA with the balanced dataset (SMOTE based).

We have also used Grid Search CV here.

But the key metrics based performance changed.

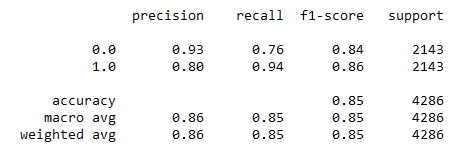
We will compare the performances of the metrics’ later in question 1.7.

The codes for the same is being attached with the Jupyter notebook (Machine Learning)

For the model evaluation and comparison, we have stated the same in the later half of the question, where model performances are being compared.

**Overfitting-** This are most prominent while using smote in most of the metrices, although, overfitting issues are more or less there even in the base and tuned model.

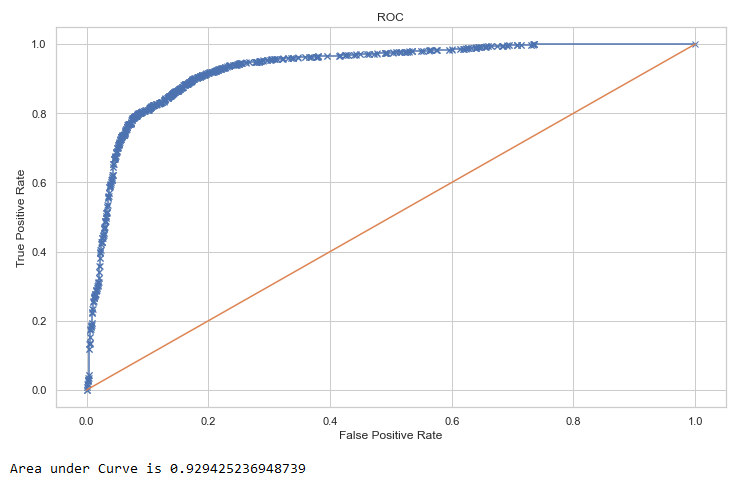
**3.Output from Base LDA Model on Train dataset (With Grid Search and Smote)**

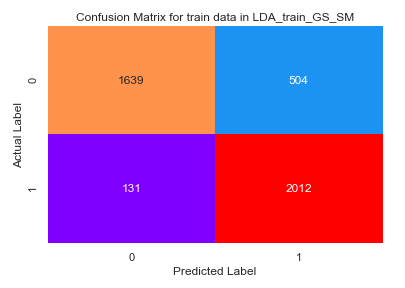


We can observe from here that performance scores for the base model on train dataset is unable to work efficiently and correctly for the default values as compared to the Non-Default ones, even though Accuracy is high.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.90255, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model on train dataset shows that True positives and negatives are being correctly determined in majority, but it has high no. of false positives as well as negatives

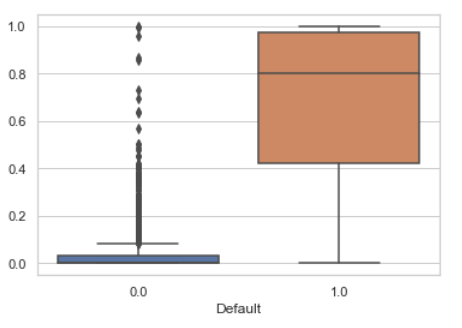




**1.11 Validate the LDA Model on test Dataset and state the performance matrices. Also state interpretation from the model**

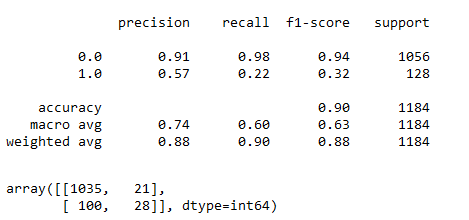
The model has been validated on the basis of various parameters taken into consideration.

First on the basis of the predicted values based on the train dataset, we have tried to use a boxplot to differentiate between the Default and Non-Default values .



We have applied and validated the above models on the test datasets as well which are shown below

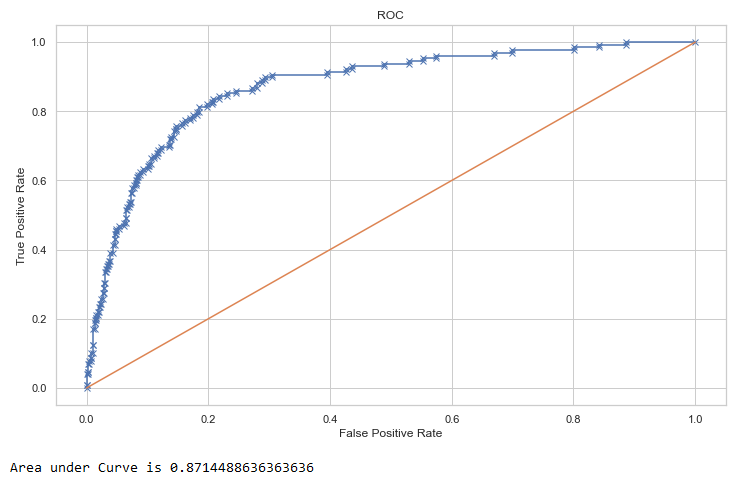
1. **Output from LDA Model on Test dataset (Without Grid Search and Smote)**

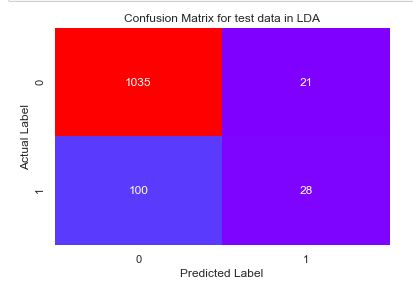


We can observe from here that performance scores for the base model on train dataset is unable to work efficiently and correctly for the default values as compared to the Non-Default ones, even though Accuracy is high.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.87144, which is low as compared to the Random Forest base model

We can observe from here that confusion matrix for the base model on test dataset shows that True positives and negatives are being correctly determined in majority, but it has high no. of false positives as well as negatives



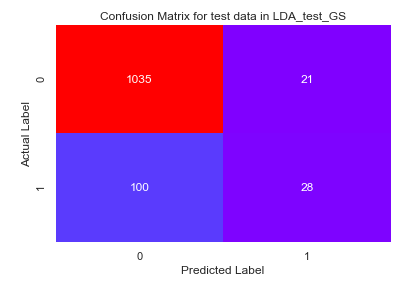
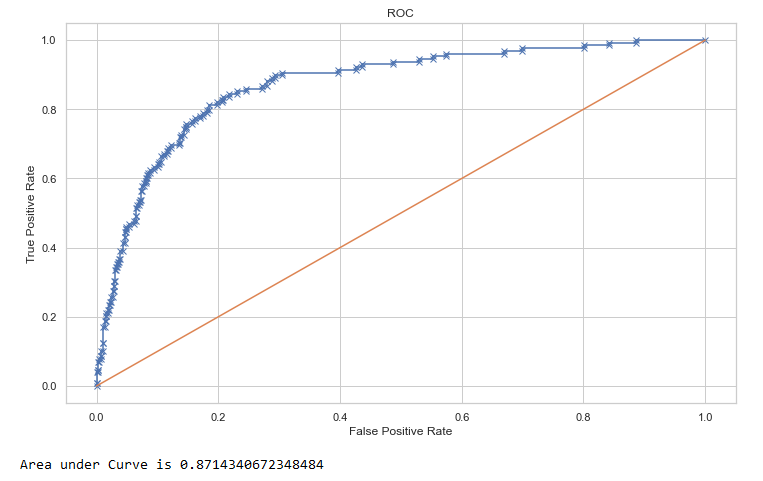
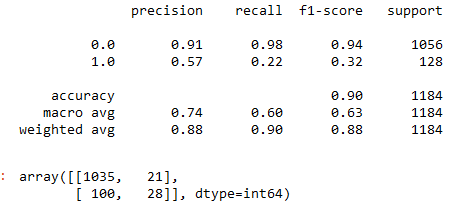


**2. Output from LDA Model on Test dataset (With Grid Search and without Smote)**

We can observe from here that performance scores for this model are less than 100, which states the impact of overfitting within the train dataset is comparatively less.

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 1.00, which states the presence of overfitting within the train dataset.

We can observe from here that confusion matrix for the base model on test dataset shows that True positives and negatives are being correctly determined in majority, but it has high no. of false positives as well as negatives



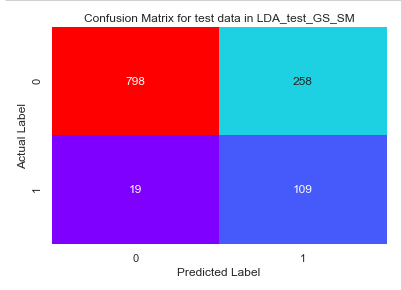
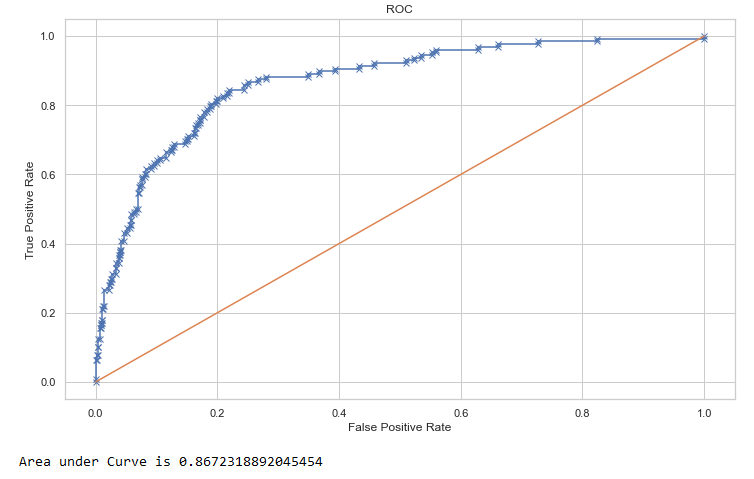
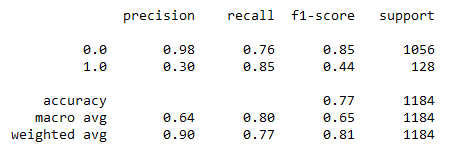
**3. Output from LDA Model on Test dataset (With Grid Search and Smote)**

We can observe from here that performance scores for this model are less than 100%

1. Recall for default values-
2. Precision for default values
3. Accuracy -

We can observe from here that AUC scores for the base model from the ROC\_AUC curve are 0.86723, which is almost similar to train values.

We can observe from here that confusion matrix for this model shows little false prediction (false positive or false negative). Overall confusion matrix shows that True positives and negatives are being correctly determined in majority, but it has high no. of false positives as well as negatives



* 1. **Compare the performances of Logistics, Radom Forest and LDA models (include ROC Curve)**

***#Train and Test accuracies for each model***

|  |  |  |  |
| --- | --- | --- | --- |
| **Model Types** | **Categories** | **Train Data Accuracy** | **Test Data Accuracy** |
| **Logistic Regression** | **Base Model** | **85.1%** | **85.6%** |
| **Grid Search CV** |  |  |
| **Smote** |  |  |
| **Random Forest** | **Base Model** | **100%** | **98%** |
| **Grid Search CV** | **99%** | **98%** |
| **Smote** | **99%** | **97%** |
| **LDA** | **Base Model** | **90%** | **90%** |
| **Grid Search CV** | **90%** | **90%** |
| **Smote** | **85%** | **77%** |

As per the model validation is concerned, we can observe that once after using the Hyper Tuning Parameters, through Grid Search Cross Validation, the models show generalisation, which means they are no longer Under fitted or Over fitted. Hence as per Accuracy metrics- Random Forest under Grid Search CV has shown the best performance so far in both train and test dataset

***#Train and Test Recall for each model***

|  |  |  |  |
| --- | --- | --- | --- |
| **Model Types** | **Categories** | **Train Data Recall** | **Test Data Recall** |
| **Logistic Regression** | **Base Model** | **95%** | **94.5%** |
| **Grid Search CV** |  |  |
| **Smote** |  |  |
| **Random Forest** | **Base Model** | **100%** | **86%** |
| **Grid Search CV** | **92%** | **87%** |
| **Smote** | **100%** | **91%** |
| **LDA** | **Base Model** | **24%** | **22%** |
| **Grid Search CV** | **24%** | **22%** |
| **Smote** | **94%** | **85%** |

As per the model validation is concerned, we can observe that once after using the Hyper

Tuning Parameters, through Grid Search Cross Validation, the models show generalisation, which means they are no longer Under fitted or Over fitted. Hence as per Recall metrics- Logistic regression has shown the best performance so far in both train and test dataset

|  |  |  |  |
| --- | --- | --- | --- |
| **Model Types** | **Categories** | **Train Data Precision** | **Test Data Precision** |
| **Logistic Regression** | **Base Model** | **41.6%** | **42.5%** |
| **Grid Search CV** |  |  |
| **Smote** |  |  |
| **Random Forest** | **Base Model** | **100%** | **91%** |
| **Grid Search CV** | **96%** | **90%** |
| **Smote** | **99%** | **97%** |
| **LDA** | **Base Model** | **66%** | **57%** |
| **Grid Search CV** | **66%** | **57%** |
| **Smote** | **80%** | **30%** |

***#Train and Test Precision for each model***

As per the model validation is concerned, we can observe that once after using the Hyper

Tuning Parameters, through Grid Search Cross Validation, the models show generalisation, which means they are no longer Under fitted or Over fitted. Hence as per Precision metrics- Random Forest under Grid Search CV and smote process together has shown the best performance so far in both train and test dataset

***#Train and Test F1 score for each model***

|  |  |  |  |
| --- | --- | --- | --- |
| **Model Types** | **Categories** | **Train Data Accuracy** | **Test Data Accuracy** |
| **Logistic Regression** | **Base Model** | **58.6%** | **60%** |
| **Grid Search CV** |  |  |
| **Smote** |  |  |
| **Random Forest** | **Base Model** | **100%** | **88%** |
| **Grid Search CV** | **94%** | **88%** |
| **Smote** | **99%** | **89%** |
| **LDA** | **Base Model** | **35%** | **32%** |
| **Grid Search CV** | **35%** | **32%** |
| **Smote** | **86%** | **44%** |

As per the model validation is concerned, we can observe that once after using the Hyper

Tuning Parameters, through Grid Search Cross Validation, the models show generalisation, which means they are no longer Under fitted or Over fitted. Hence as per F1 metrics- Random Forest under Grid Search CV and smote process together has shown the best performance so far in both train and test dataset

***#Train and Test Area under curve for each model***

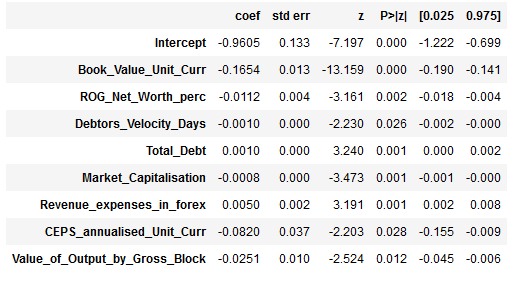
|  |  |  |  |
| --- | --- | --- | --- |
| **Model Types** | **Categories** | **Train Data Accuracy** | **Test Data Accuracy** |
| **Logistic Regression** | **Base Model** |  |  |
| **Grid Search CV** |  |  |
| **Smote** |  |  |
| **Random Forest** | **Base Model** | **100%** | **99.22%** |
| **Grid Search CV** | **99.92%** | **99.14%** |
| **Smote** | **99.96%** | **99.18%** |
| **LDA** | **Base Model** | **90.25%** | **87.14%** |
| **Grid Search CV** | **90.25%** | **87.14%** |
| **Smote** | **92.94%** | **86.72%** |

As per the model validation is concerned, we can observe that once after using the Hyper

Tuning Parameters, through Grid Search Cross Validation, the models show generalisation, which means they are no longer Under fitted or Over fitted. Hence as per AUC metrics- Random Forest under Grid Search CV and smote process together has shown the best performance so far in both train and test dataset

**1.13 State Recommendations from the above models**

From the information received after comparing all the above models based on their performance metrices, we can say that Random Forest model after Grid Search CV on SMOTE based dataset is the best model. We will also consider the coefficients derived from the best logit model which was built using Stats Model to derive insights.



1. **Book\_Value\_Unit\_Curr** – Book value adjusted to reflect asset's true fair market value, which being negative means higher chances of default and also indicates that the Net worth for the next year will be negative as well.

1. **ROG\_Net\_Worth\_perc** - Rate of Growth – Networth, which being negative means Net worth growth has decreased over time
2. **Debtors\_Velocity\_Days** - Average days required for receiving the payments, here the average days are quite low w.r.t. payment to be received
3. **Total\_Debt**- The sum of money borrowed by the company and is due to be paid. This being positive indicates, chances of default, but it’s lower

1. **Market\_Capitalisation -** Product of the total number of a company's outstanding shares and the current market price of one share, this being low, shows the company’s position and it’s hold in the current market

1. **Revenue\_expenses\_in\_forex -** Expenses due to foreign currency transactions. This being high, should be handled considering its high significance
2. **CEPS\_annualised\_Unit\_Curr -**Cash Earnings per Share, profitability ratio that measures the financial performance of a company by calculating cash flows on a per share basis. This being negative, indicates higher chances of default
3. **Value\_of\_Output\_by\_Gross\_Block -** Ratio of Value of Output (market value) to Gross Block- this is also negative

***Observations***

As observed above while comparing the models, accuracy of the model i.e. %overall correct predictions **has increased from 85.6% to 98% while using Rnadom Forest model**

**Recommendations**

In Financial Industry or Healthcare industry, it is to be taken into consideration that the value of False negatives is low, which means a test result which wrongly indicates that a particular condition or attribute is absent. So, in the model building process, the false negative values were quite low compared to its true counterpart in majority of the models.

To increase the accuracy and to determine the classes in a better way, it is recommended to clarify the values at the data entry level and check accordingly as there can be mismatch in data which can lead to the increase in the no. of false positives and negatives.

**Topic: Market Risk**

**Problem Statement**

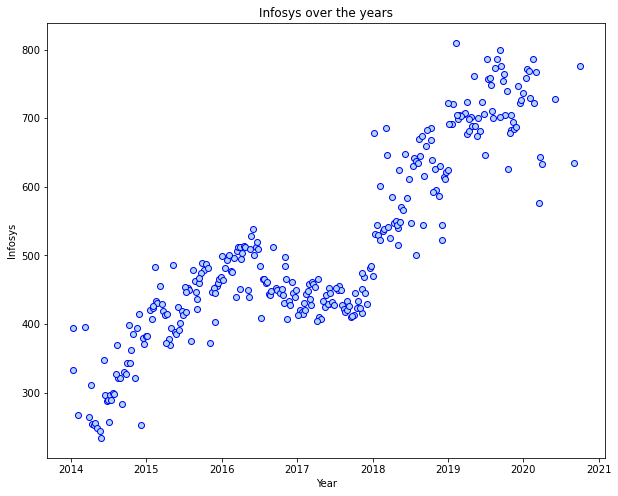
The dataset contains 6 years of information (weekly stock information) on the stock prices of 10 different Indian Stocks. Calculate the mean and standard deviation on the stock returns and share insights.

**2.1 Draw Stock Price Graph(Stock Price vs Time) for any 2 given stocks with inference**

The two price graphs of two stocks out of the total stocks are being done and shown below-

We have taken into consideration – **Infosys** and **Mahindra & Mahindra**

***Stock Price graph of Infosys***

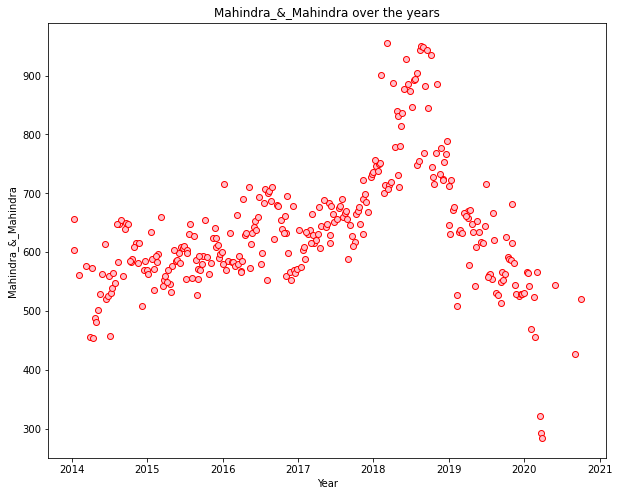


***Inference***- For this stock we can observe that the stock prices though fluctuated on a particular interval, but it has been growing from 2014 till 2021.

During the period between 2016-2018, there has been a period of stagnant price move or rather there was a dip observed, which ultimately kept on increasing beyond 2018.

The highest price is observed during first quarter of 2019, where it has reached beyond 800 units, while the lowest has been during the second quarter of 2014, where it has reached below 300 units

***Stock Price graph of Mahindra & Mahindra***

****

***Inference***- For this stock, we can observe that the stock prices though fluctuated on a particular interval, but it has overall been on similar price range between 2014 to 2021, except for an increase in price during 2018-2019 and gradually it decreased.

During the period between 2018-2019, the increase is price for this stock has reached to beyond 900 units(highest), while during the first quarter of 2020, it has reached its lowest ie beyond 300 units

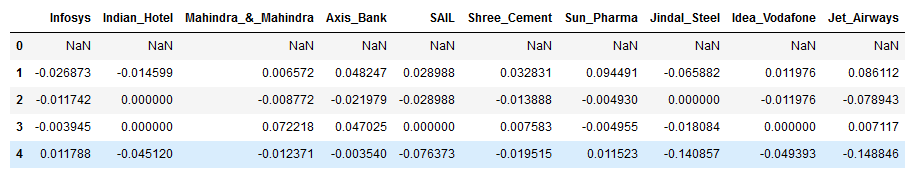
**2.2 Calculate Returns for all stocks with inference**

**Stock Returns-** A return, also known as a financial return, in its simplest terms, is the money made or lost on an investment over some period of time.

A return can be expressed nominally as the change in dollar value of an investment over time. A return can also be expressed as a percentage derived from the ratio of profit to investment.

In this particular problem, the stock returns are being calculated using the difference in Log of prices at t time period and at (t-1) time period.

**The Stock return output table**



***Inference***- As weekly stock information are given for the 10 stocks, we can observe over here the top 5 rows of them

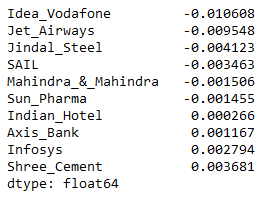
**2.3 Calculate Stock Means and Standard Deviation for all stocks with inference**

**Stock means -** Mean return, in securities analysis, is the expected value, or mean, of all the likely returns of investments comprising a portfolio.

A mean return is also known as an expected return and can refer to how much a stock returns on a monthly basis. In capital budgeting, a mean return is the mean value of the probability distribution of possible returns.

Stock means of the returns of all the stocks are being calculated using the averaging formula

**The Stock mean of return output table**

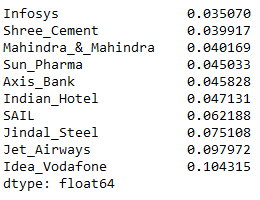


***Inference***- We can observe from here that the Idea\_Vodafone has generated lowest return means of all the stocks, while Shree\_Cement has generated highest returns means.

**Standard Deviation**- Standard deviation is the statistical measure of market volatility, measuring how widely prices are dispersed from the average price. If prices trade in a narrow trading range, the standard deviation will return a low value that indicates low volatility.

Standard deviation of the returns of all the stocks are being calculated using the Standard Deviation Function.

**The Standard Deviation of return output table**

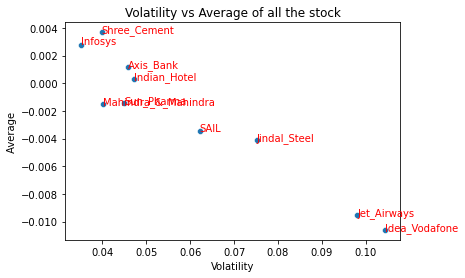


***Inference***- We can observe from here that the Idea\_Vodafone has generated lowest Standard deviation among all the stocks, while Infosys has generated highest Standard deviation.

**2.4 Draw a plot of Stock Means vs Standard Deviation and state your inference**

The Stock Means(Average) and Standard Deviation(Volatility) of each of the companies are being taken into consideration to create the scatter plot shown below-

**Volatility vs Average of Stocks- Scatter plot**



***Inference-*** From the above scatter plot we can observe that the stock- ‘**Infosys’** has the highest Average while also having the lowest volatility and it is being followed by ‘**Shree** **Cement’.**

But on the other hand , ‘**Idea Vodafone**’ has the highest Volatility while having the lowest Standard deviation followed by ‘**Jet Airways’**

This is one useful graph to find how risk and returns helps in understanding and picking right stocks to invest

**Volatility vs Average of Stocks- Tabular representation**



***Inference-*** We can also check from the table above how the comparison from the scatter plots is relevant here.

**2.5 Conclusion and Recommendations**

***Conclusion -***

* Stocks with high volatility or Standard deviation and lower Average or Mean returns are not to be taken into consideration for creating a portfolio. This is because higher risk is involved without generating higher returns
* Thus, in this dataset we have come across the following criteria for selection of stocks-
  + Stock with highest return
  + Stock with lowest risk

**Return Basis-**

From this standpoint, we can consider stocks like **Shree Cement**, **Infosys** and **Axis Bank**

**Risk Basis-**

From this standpoint, we can consider stocks like **Infosys**, **Shree** **Cement** and **Mahindra & Mahindra**

***Recommendation-***

* The Average vs Volatility scatter plot or the Return to Risk plot can provide some idea for the basis of stock selection as they help to access the risk-reward ratio.
* High Volatility in many stocks can provide the short-term returns, but that may not be sustainable for a long-term horizon
* Similarly low Volatility in many stocks may not provide the short-term gains, but that may generate good gains for a long-term horizon.
* So, it is important to look out for the background of the company as checking the fundamental and technical analysis involved with the same, before any investment in the stock
* It is also recommended to check with the industry and the global scenario while investing in the stocks.

**References**

* *https://adataanalyst.com/data-analysis-resources/visualise-categorical-variables-in-python/#:~:text=Visualise%20Categorical%20Variables%20in%20Python%20using%20Bivariate%20Analysis,a%20pre%2Ddefined%20significance%20level.*
* *https://www.investopedia.com/terms/v/variance-inflation-factor.asp#:~:text=Variance%20inflation%20factor%20(VIF)%20is,only%20that%20single%20independent%20variable.*
* *https://www.investopedia.com/terms/r/return.asp*
* *https://www.investopedia.com/terms/m/meanreturn.asp*
* *https://www.fidelity.com/learning-center/trading-investing/technical-analysis/technical-indicator-guide/standard-deviation#:~:text=Standard%20deviation%20is%20the%20statistical,value%20that%20indicates%20low%20volatility.*

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***