greatlearning
Power Ahead

# FRA

# MILESTONE - 2

**Pooja Kabadi**
PGP-DSBA Online
Batch- A4
18-06-2022

## Table of Contents:



## List of Tables:



## List of Figures:

## Problem Statement:

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

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

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

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

**Hints**:

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

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

**Market Risk**

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.

**Data Dictionary:**

| Sl.no | Field Name | Description |
|---|---|---|
| **1** | Co_Code | Company Code |
| **2** | Co_Name | Company Name |
| **3** | Networth Next Year | Value of a company as on 2016 - Next Year(difference between the value of total assets and total liabilities) |
| **4** | Equity Paid Up | Amount that has been received by the company through the issue of shares to the shareholders |
| **5** | Networth | Value of a company as on 2015 - Current Year |
| **6** | Capital Employed | Total amount of capital used for the acquisition of profits by a company |
| **7** | Total Debt | The sum of money borrowed by the company and is due to be paid |
| **8** | Gross Block | Total value of all of the assets that a company owns |
| **9** | Net Working Capital | The difference between a company's current assets (cash, accounts receivable, inventories of raw materials and finished goods) and its current liabilities (accounts payable). |

| | | |
|---|---|---|
| **10** | Current Assets | All the assets of a company that are expected to be sold or used as a result of standard business operations over the next year. |
| **11** | Current Liabilities and Provisions | Short-term financial obligations that are due within one year (includes amount that is set aside cover a future liability) |
| **12** | Total Assets/Liabilities | Ratio of total assets to liabilities of the company |
| **13** | Gross Sales | The grand total of sale transactions within the accounting period |
| **14** | Net Sales | Gross sales minus returns, allowances, and discounts |
| **15** | Other Income | Income realized from non-business activities (e.g. sale of long term asset) |
| **16** | Value Of Output | Product of physical output of goods and services produced by company and its market price |
| **17** | Cost of Production | Costs incurred by a business from manufacturing a product or providing a service |
| **18** | Selling Cost | Costs which are made to create the demand for the product (advertising expenditures, packaging and styling, salaries, commissions and travelling expenses of sales personnel, and the cost of shops and showrooms) |
| **19** | PBIDT | Profit Before Interest, Depreciation & Taxes |
| **20** | PBDT | Profit Before Depreciation and Tax |
| **21** | PBIT | Profit before interest and taxes |
| **22** | PBT | Profit before tax |
| **23** | PAT | Profit After Tax |
| **24** | Adjusted PAT | Adjusted profit is the best estimate of the true profit |
| **26** | CP | Commercial paper , a short-term debt instrument to meet short-term liabilities. |
| **27** | Revenue earnings in forex | Revenue earned in foreign currency |
| **28** | Revenue expenses in forex | Expenses due to foreign currency transactions |
| **29** | Capital expenses in forex | Long term investment in forex |
| **30** | Book Value (Unit Curr) | Net asset value |
| **31** | Book Value (Adj.) (Unit Curr) | Book value adjusted to reflect asset's true fair market value |
| **32** | Market Capitalisation | Product of the total number of a company's outstanding shares and the current market price of one share |
| **33** | 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 |
| **34** | Cash Flow From Operating Activities | Use of cash from ongoing regular business activities |
| **35** | Cash Flow From Investing Activities | Cash used in the purchase of non-current assets–or long-term assets– that will deliver value in the future |
| **36** | Cash Flow From Financing Activities | Net flows of cash that are used to fund the company (transactions involving debt, equity, and dividends) |
| **37** | ROG-Net Worth (%) | Rate of Growth - Networth |
| **38** | ROG-Capital Employed (%) | Rate of Growth - Capital Employed |
| **39** | ROG-Gross Block (%) | Rate of Growth - Gross Block |

| | | |
|---|---|---|
| **40** | ROG-Gross Sales (%) | Rate of Growth - Gross Sales |
| **41** | ROG-Net Sales (%) | Rate of Growth - Net Sales |
| **42** | ROG-Cost of Production (%) | Rate of Growth - Cost of Production |
| **43** | ROG-Total Assets (%) | Rate of Growth - Total Assets |
| **44** | ROG-PBIDT (%) | Rate of Growth- PBIDT |
| **45** | ROG-PBDT (%) | Rate of Growth- PBDT |
| **46** | ROG-PBIT (%) | Rate of Growth- PBIT |
| **47** | ROG-PBT (%) | Rate of Growth- PBT |
| **48** | ROG-PAT (%) | Rate of Growth- PAT |
| **49** | ROG-CP (%) | Rate of Growth- CP |
| **50** | ROG-Revenue earnings in forex (%) | Rate of Growth - Revenue earnings in forex |
| **51** | ROG-Revenue expenses in forex (%) | Rate of Growth - Revenue expenses in forex |
| **52** | ROG-Market Capitalisation (%) | Rate of Growth - Market Capitalisation |
| **53** | Current Ratio[Latest] | Liquidity ratio, company's ability to pay short-term obligations or those due within one year |
| **54** | Fixed Assets Ratio[Latest] | Solvency ratio, the capacity of a company to discharge its obligations towards long-term lenders indicating |
| **55** | Inventory Ratio[Latest] | Activity ratio, specifies the number of times the stock or inventory has been replaced and sold by the company |
| **56** | Debtors Ratio[Latest] | Measures how quickly cash debtors are paying back to the company |
| **57** | Total Asset Turnover Ratio[Latest] | The value of a company's revenues relative to the value of its assets |
| **58** | Interest Cover Ratio[Latest] | Determines how easily a company can pay interest on its outstanding debt |
| **59** | PBIDTM (%)[Latest] | Profit before Interest Depreciation and Tax Margin |
| **60** | PBITM (%)[Latest] | Profit Before Interest Tax Margin |
| **61** | PBDTM (%)[Latest] | Profit Before Depreciation Tax Margin |
| **62** | CPM (%)[Latest] | Cost per thousand (advertising cost) |
| **63** | APATM (%)[Latest] | After tax profit margin |
| **64** | Debtors Velocity (Days) | Average days required for receiving the payments |
| **65** | Creditors Velocity (Days) | Average number of days company takes to pay suppliers |
| **66** | Inventory Velocity (Days) | Average number of days the company needs to turn its inventory into sales |
| **67** | Value of Output/Total Assets | Ratio of Value of Output (market value) to Total Assets |
| **68** | Value of Output/Gross Block | Ratio of Value of Output (market value) to Gross Block |

*Table 1. Data dictionary*

**Objective of the report:**
The goal is to analyse the data available, which includes information from the financial statements of the companies for the year 2015 (prior year) as well as information on the company's net worth for the year 2016. We aim to provide suggestions to help potential investors determine which company to invest in in order to improve the company's performance in the future and attract investors to profit from the funds.

We'll utilise **Linear Discriminant Analysis (LDA) and Random Forest (RF)**, which is a Supervised Machine Learning approach for binary classification of one dependent variable (Net Worth Next Year) and one or more independent variables like Co_Code, Capital Employed, Equity Paid Up, and Net Working, Market Capitalisation, etc. It gives discrete outputs, either **"Default"** or "**Non-Default"**.

Here's a brief overview of the supervised machine learning approaches we'll be employing to determine the best optimum model for the analysis:

**1. Linear Discriminant Analysis (LDA)** is a linear model for classification and dimensionality reduction that predicts the class in the response variable of a given observation using a linear combination of independent variables.

**2. Random forests (RF)** is used for both classification and regression. It generates decision trees from randomly selected data samples, receives predictions from each tree, and votes on the best option.

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

The dataset is divided into train and test groups in a 67-33 ratio, with training data accounting for 67 percent and test data for 33 percent. The number 42 represents the random situation indicated below.

- Train dataset has 2402 records (67% of the total dataset)
- Test dataset has 1184 records (33% of the total dataset)

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

We have created a new column named default to predict with output as 0 or 1.
We have transformed the value as follows,

- **1 - If the Net Worth of 2016 is less than or 0 for the company**
- **0 - If the Net Worth Next of 2016 is greater than 0 for the company**

**The proportion of the dependent variable are as follows:**

```
0  -  3199  -  0.89208  -  89%
1  -  387   -  0.10792  -  11%
```

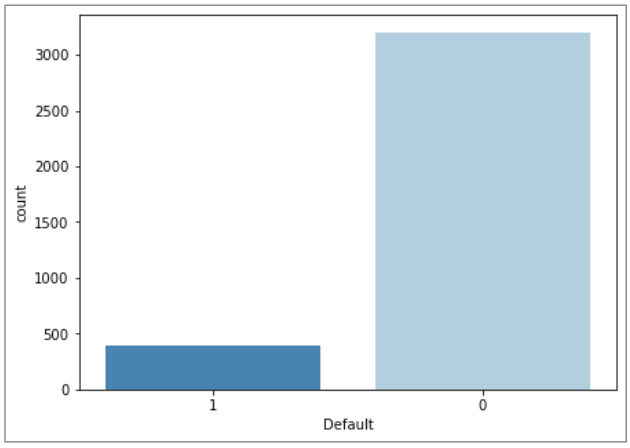


*Figure 1. Bar plot of dependent variable- Default*

**Below is the output of dataset after appending the new variable 'Default' which is derived from the Net worth next year:**

| APATM_perc_Latest | Debtors_Vel_Days | Creditors_Vel_Days | Inventory_Vel_Days | Value_of_Output_to_Total_Assets | Value_of_Output_to_Gross_Block | Default |
|---|---|---|---|---|---|---|
| 0.00 | 0.0 | 0.0 | 45.0 | 0.00 | 0.00 | 1 |
| -87.18 | 29.0 | 101.0 | 2.0 | 0.31 | 0.24 | 1 |
| -7961.51 | 97.0 | 210.5 | 0.0 | -0.03 | -0.26 | 1 |
| -51.58 | 93.0 | 63.0 | 2.0 | 0.24 | 1.90 | 1 |
| 274.79 | 253.0 | 210.5 | 0.0 | 0.01 | 0.05 | 1 |

*Figure 2.Head of dataset after appending default variable*

By doing the VIF check, we have cleaned up our data, dealt with outliers, imputed missing values, and eliminated non-significant variables. We divided the data into train and test sets after completing these procedures. Moving further, we developed our Logistic Regression model and fine-tuned it with the aid of optimum threshold to achieve the best results in the milestone 1. Now, we will be working on Random Forest model and see how it performs on our given dataset.

**Random Forest Model:**

Random Forest is based on the concept of ensemble learning, which is a process of combining multiple models to solve a complex problem and to improve the performance of the model. It is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the majority to improve the classification accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

**Model Building – Random Forest (RF) :**

A forest is comprised of trees (we can say RF is an extension of Decision Trees). It is said that the more trees it has, the more robust a forest is. It works in four steps:

1. Select random samples from a given dataset.
2. Construct a decision tree for each sample and get a prediction result from each decision tree.
3. Perform a vote for each predicted result.

4. Select the prediction result with the most votes as the final prediction.

Initially, we fit the train data and labels in the **Random Forest model**, based on the model performance the model is tunned using **Grid search,** the best parameters are used and the model is re-built and model performance is calculated which includes Classification report of accuracy, recall, precision and F1 score for both train and test data.

**Hyperparameter Tuning:**

- *'max_depth': [4,5,6,7,8]*
- *'max_features': [3,4,5, 6,7]*
- *'min_samples_leaf': [20,30, 40,50,60]*
- *'min_samples_split': [60,150, 120,150,180]*
- *'n_estimators': [301, 501]*
- *Cross validation (cv) = 3*

To hyper tune the random forest trees to the best height, we take the values of max depth as 4,5,6, 7 and 8 in our grid search to get optimum results. Max depth represents the depth of each tree in the forest.

Max_features in the algorithm is the maximum number of features random forest model is allowed to try in an individual tree. There are many ways to take max_features. SQRT is one of them. Here we have 10 variables and taking 2 as max_features doesn't make much sense. For grid search we keep 3,4,5,6 and 7. Let us see what we get as our best parameter.

Min samples leaf should be 1% to 3% of the total records. 1% of 3000 is 30. We take 20, 30,40,50 and 60 in our case to see which fits perfectly. Also, min samples split is approximately 3 times of Min samples life. So we take 60, 120,150 and 180 as our inputs for grid search. Also, cross validation (CV) given for this is equal to 3.

N_estimators is the number of trees you want to build before taking the maximum voting or averages of predictions. Higher number of trees give you better performance but makes your code slower. We check for 301 and 501 in our grid search. Let's check for the best out of these in our model.

We fit the grid on the train data set. Once the grid is fit, we check the best parameters that are given to us from the grid search algorithm.

**The final best parameters are:**

- *Max depth is equal to 5*
- *Max_features is equal to 7*
- *Min samples leaf is 20*
- *Min_samples spit is 60*
- *n_estimators is 501*

**Feature importance:**

We can check for the importance of each variable in our model built obtained from the grid search best parameters.

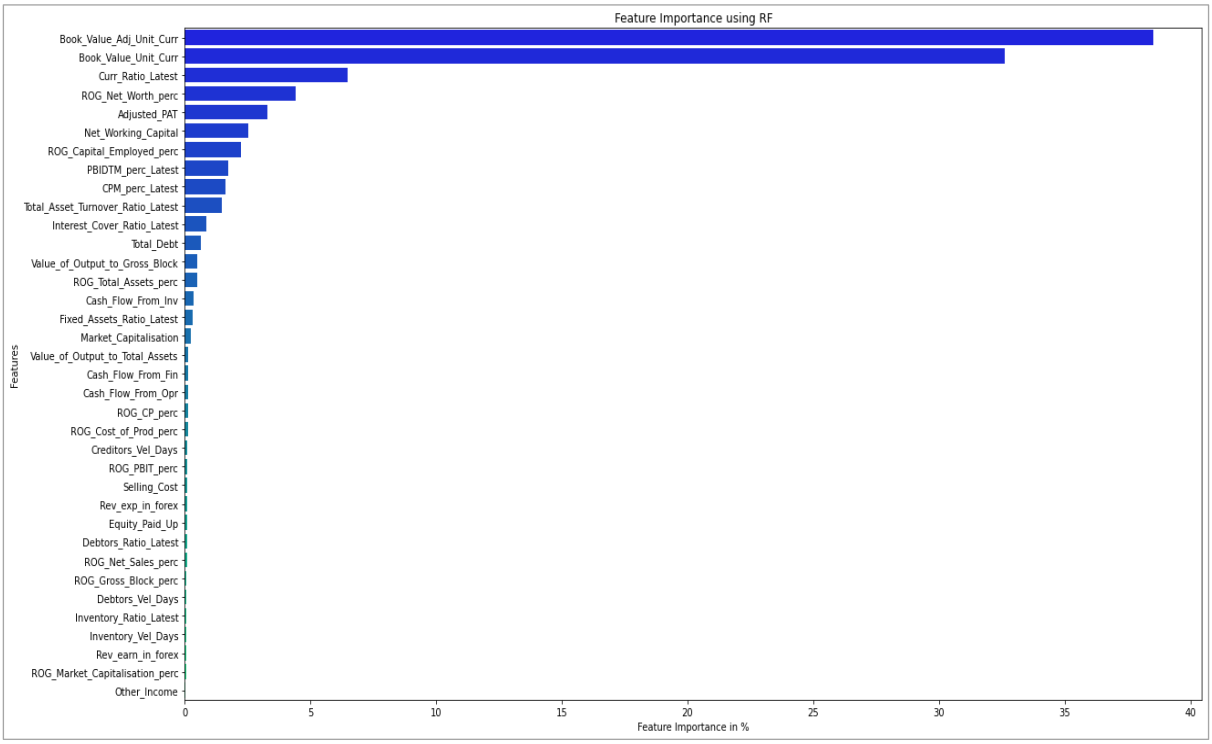


*Figure 3. Feature importance for Random Forest*

Below is the table of feature importance values of all variables.

| Variable | Feature Importance |
|---|---|
| **Book_Value_Adj_Unit_Curr** | 30.163905 |
| **Book_Value_Unit_Curr** | 28.358341 |
| **Curr_Ratio_Latest** | 5.169612 |
| **ROG_Net_Worth_perc** | 4.081462 |
| **Adjusted_PAT** | 2.941480 |
| **Net_Working_Capital** | 2.599456 |
| **PBIDTM_perc_Latest** | 2.323307 |
| **ROG_Capital_Employed_perc** | 2.130456 |
| **CPM_perc_Latest** | 2.111027 |
| **Total_Asset_Turnover_Ratio_Latest** | 1.917800 |
| **Interest_Cover_Ratio_Latest** | 1.462328 |
| **Value_of_Output_to_Gross_Block** | 1.179198 |

| | |
|---|---|
| **Total_Debt** | 1.116918 |
| **ROG_Total_Assets_perc** | 0.976471 |
| **Value_of_Output_to_Total_Assets** | 0.949656 |
| **Market_Capitalisation** | 0.861308 |
| **Fixed_Assets_Ratio_Latest** | 0.783726 |
| **ROG_PBIT_perc** | 0.727333 |
| **Cash_Flow_From_Fin** | 0.721987 |
| **Equity_Paid_Up** | 0.701849 |
| **ROG_Gross_Block_perc** | 0.661112 |
| **ROG_Cost_of_Prod_perc** | 0.655478 |
| **Cash_Flow_From_Opr** | 0.636198 |
| **Cash_Flow_From_Inv** | 0.628674 |
| **ROG_Market_Capitalisation_perc** | 0.628640 |
| **Inventory_Ratio_Latest** | 0.608278 |
| **ROG_CP_perc** | 0.580188 |
| **Selling_Cost** | 0.575043 |
| **Creditors_Vel_Days** | 0.549292 |
| **Debtors_Ratio_Latest** | 0.523847 |
| **Inventory_Vel_Days** | 0.523109 |
| **Other_Income** | 0.487337 |
| **Debtors_Vel_Days** | 0.478173 |
| **ROG_Net_Sales_perc** | 0.446102 |
| **Rev_exp_in_forex** | 0.420935 |
| **Rev_earn_in_forex** | 0.319975 |

*Table 2. Feature Importance for Random Forest*

From Table 2 above we can say that, Book_Value_Adj_Unit_Curr, Book_Value_Unit_Curr , Curr_Ratio_Latest, ROG_Net_Worth_perc are the most relevant features towards default, follo wed by other variables. Capital_expenses_in_forex, Creditors_Velocity_Days, ROG_CP_perc, Debtors_Velocity_Days and Other_Income have almost null contribution which is very low importance, we can choose to remove these features if required.

We have built the Random Forest model on the variables, by computing the VIF scores of all variables in the dataset, we can determine which variables are significant.

A VIF score of 5 or above indicates that the variable is not relevant in model construction. As a result, we'll eliminate variables with high VIF scores one by one until we're left with variables with VIF values of less than 5.

**The top 10 important variables on which our final model is built are as follows:**

1) Market Capitalisation
2) Total Debt
3) Equity Paid Up
4) Cash Flow from Opr
5) ROG Net Worth perc
6) Book Value Unit Curr
7) Value of Output to gross Block
8) CPM perc Latest
9) Curr Ratio Latest
10) Debtors Ratio Latest

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

Model performance helps to understand how good the model that we have trained using the dataset is so that we have confidence in the performance of the model for future predictions. We evaluate our models' performance on train and test datasets once they've been constructed. We try to determine if the model is underfitting or overfitting by checking for accuracy, precision, and other factors. We have specific scores and matrices for our model's performance. Following are the methods used to evaluate the model performance:

- **Confusion Matrix**
- **Classification Report**
  - **Accuracy**
  - **Precision**
  - **Recall**
  - **F1 Score**
- **ROC curve**
- **AUC score**

1. **Confusion Matrix:**

This gives us how many zeros (0s) i.e. (class = No claim) and ones (1s) i.e. (class = Yes claim) were correctly predicted by our model and how many were wrongly predicted.

| | Predicted Class | | |
|---|---|---|---|
| **Actual class** | | Class = No | Class = Yes |
| | Class = No | True Negative | False Positive |
| | Class = yes | False Negative | True Positive |

True positive and true negatives are the observations that are correctly predicted and therefore shown in green. We want to minimize false positives and false negatives so they are shown in red color.

2. **Accuracy :**

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.

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

3. **Precision:**

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

Precision = TP/(TP + FP)

4. Recall **(Sensitivity):**

Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes.

Recall = TP/(TP + FN)

5. **F1 Score:**

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. That is, a good F1 score means that you have low false positives and low false negatives, so you're correctly identifying real threats and you are not disturbed by false alarms. An F1 score is considered perfect when it's 1 , while the model is a total failure when it's 0

F1 score = 2 x [(Precision x Recall) / (Precision + Recall)]

6. **ROC Curve:**

ROC curve is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: True Positive Rate. False Positive Rate.

7. **AUC Score:**

AUC score gives the area under the ROC curve built. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative.

Before optimizing or applying the grid search the model performance on train data set was 100% leading to overfit. Below is the model performance of the Random Forest of the model built using the best parameters.

# Model performance of Random Forest

## Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.99 | 0.99 | 0.99 | 2157 |
| 1 | 0.92 | 0.87 | 0.89 | 245 |
| accuracy | | | 0.98 | 2402 |
| macro avg | 0.95 | 0.93 | 0.94 | 2402 |
| weighted avg | 0.98 | 0.98 | 0.98 | 2402 |

*Figure 7. Classification report of RF for train*

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.99 | 0.99 | 0.99 | 1042 |
| 1 | 0.93 | 0.90 | 0.91 | 142 |
| accuracy | | | 0.98 | 1184 |
| macro avg | 0.96 | 0.95 | 0.95 | 1184 |
| weighted avg | 0.98 | 0.98 | 0.98 | 1184 |

*Figure 6.Classification report of RF for test*

## Confusion Matrix:

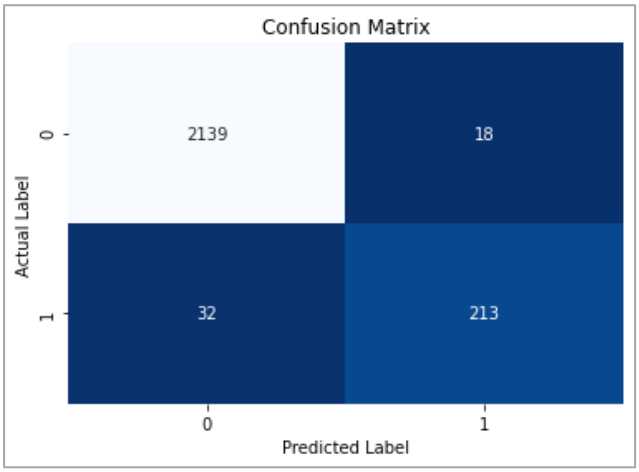


*Figure 5. Confusion Matrix of RF for train*

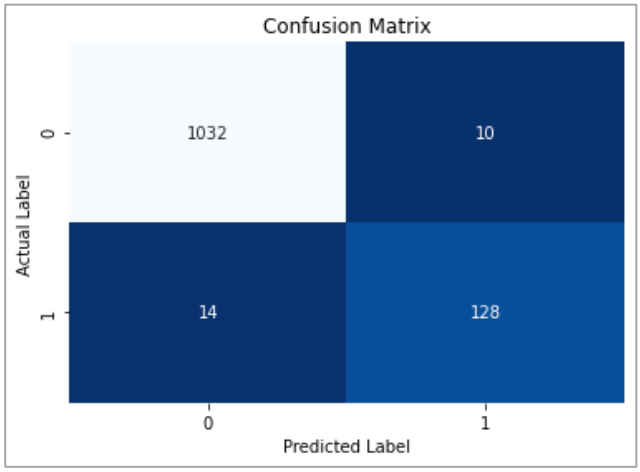


*Figure 4. Confusion Matrix of RF for test*

## ROC Curve and ROC_AUC score

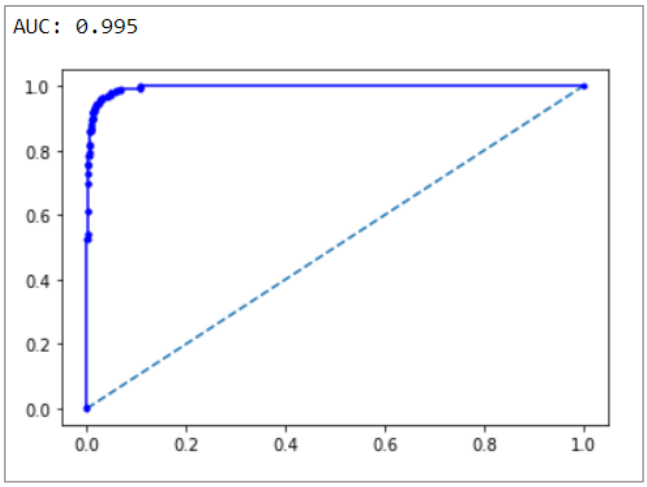


*Figure 9. ROC curve of RF for train*

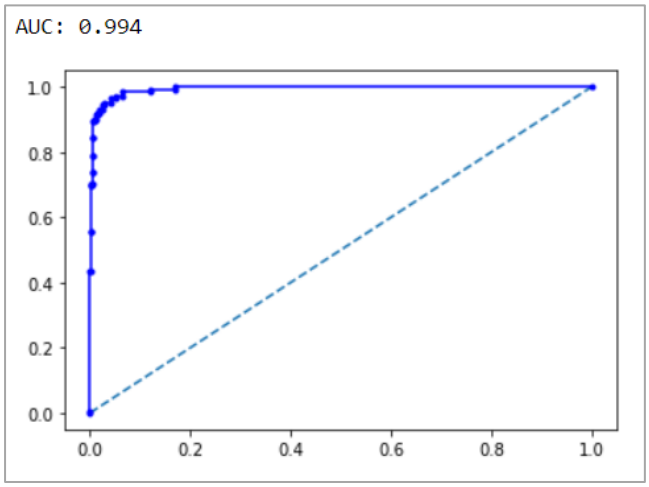


*Figure 8.ROC curve of RF for test*

| Model Performance of Random Forest | | | | | |
|---|---|---|---|---|---|
| **Sl. No** | | **Train Data** | | **Test Data** | |
| **1.** | True Positive | 213 | | 128 | |
| **2.** | True Negative | 2139 | | 1032 | |
| **3.** | False Positive | 18 | | 10 | |
| **4.** | False Negative | 32 | | 14 | |
| **6.** | Accuracy | 98% | | 98% | |
| | | **Defaulter (1)** | **Non-Defaulters (0)** | **Defaulter (1)** | **Non-Defaulters (0)** |
| **7.** | Precision | 92% | 99% | 93% | 99% |
| **8.** | Recall | 87% | 99% | 90% | 99% |
| **9.** | F1 score | 89% | 99% | 91% | 99% |

*Table 3.Model performance table for RF*

The accuracy of the test dataset is same as that of train dataset, indicating that the dataset is not overfitting. The model is a good classification model. Furthermore, the derived metrics have high values and are well-fitting.

**Interpretation from the model:**

- Based on the Random Forest model's performance measures, it appears that this model is doing well and providing 98 percent correct predictions.
- The train correctness is similarly about 98 percent, indicating that the model is not overfitting.
- We also have a high recall value, which is one of the most important criteria to consider.
- in the case of defaulters
- The ROC curve is also virtually perfect.
- The area under the curve is 99.4 percent.

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

## Linear discriminant analysis model:

Linear Discriminant Analysis uses linear combination of independent variables to predict the class in the response variable of a given observation. The prediction is made simply by the use of Bayes' Theorem which estimated the probability of the output class given the input. It also makes use of the probability of each class and also the data belonging to the class. The class which has the highest probability is considered as the output class and the model makes the prediction. The LDA model is built using the sklearn. discriminant analysis package and then fit in the training data. Using this fitted model, the predictions are made on the testing data.

LDA works when the measurements made on independent variables for each observation are continuous quantities. When dealing with categorical independent variables, the equivalent technique is discriminant correspondence analysis. On the train data set, we fit our Linear Discriminant model. By default, LDA uses a cut-off probability of 0.5. So, initially, we'll create our LDA model with a default probability of 0.5 and see how it performs, then we'll see how it performs with multiple cut-off probabilities to see which one performs the best.

**To build a Linear discriminant analysis model:**

- Fitting the linear discriminant analysis model from Sklearn discriminant analysis.
- Predicting on Training and Testing dataset.
- Getting the Predicted Classes and Probabilities and creating a data frame.

Normally, the LDA method uses a probability cut-off of 0.5. We will compute the optimal threshold probability using the ROC curve function in the sklearn library package instead of testing the performance on the default probability of 0.5.

When we calculate the threshold probability, we get it as 0.082. As a result, any probability more than or equal to 0.08 will result in prediction 1 (default), whereas any probability less than or equal to 0.08 will result in prediction 0. (non-defaulter). This is how we receive our LDA model's final predictions.

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

**Model performance of LDA:**

**Classification report:**

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.99 | 0.84 | 0.90 | 2157 |
| 1 | 0.38 | 0.89 | 0.54 | 245 |
| accuracy | | | 0.84 | 2402 |
| macro avg | 0.68 | 0.87 | 0.72 | 2402 |
| weighted avg | 0.92 | 0.84 | 0.87 | 2402 |

*Figure 10. Classification report of LDA for train*

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.99 | 0.78 | 0.87 | 1042 |
| 1 | 0.37 | 0.93 | 0.53 | 142 |
| accuracy | | | 0.80 | 1184 |
| macro avg | 0.68 | 0.86 | 0.70 | 1184 |
| weighted avg | 0.91 | 0.80 | 0.83 | 1184 |

*Figure 11. Classification report of LDA for test*

**Confusion Matrix:**

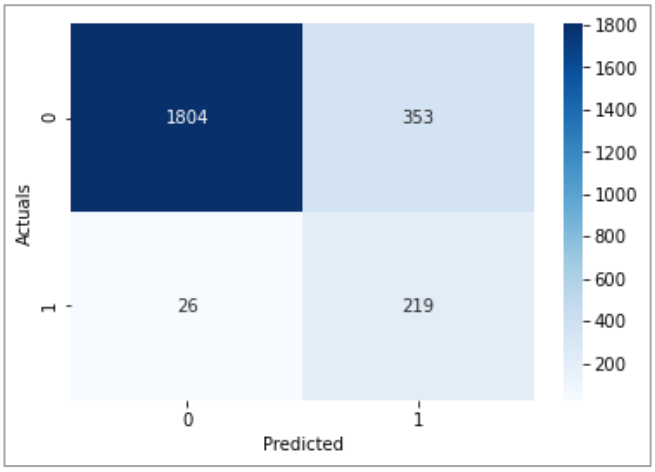


*Figure 13. Confusion Matrix of LDA for train*

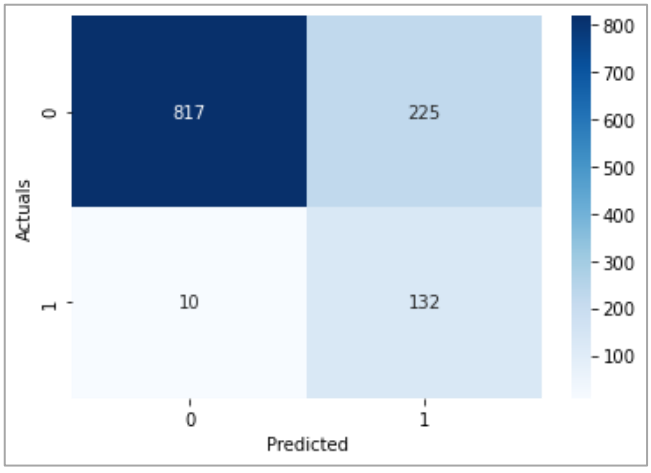


*Figure 12. Confusion Matrix of LDA for test*

## ROC Curve and ROC_AUC score:

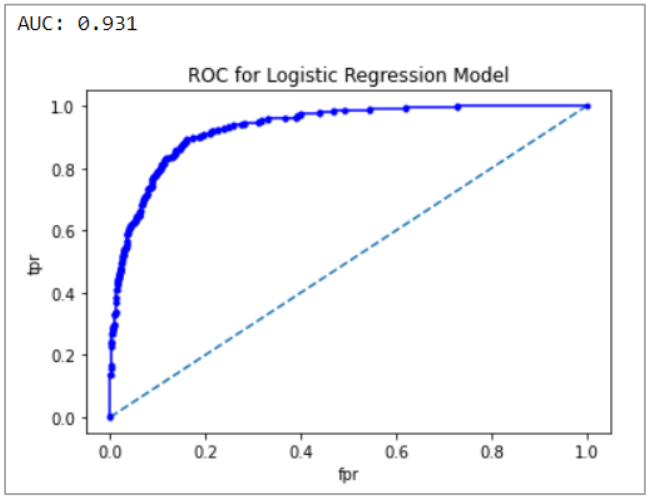


*Figure 14. ROC curve of LDA for train*

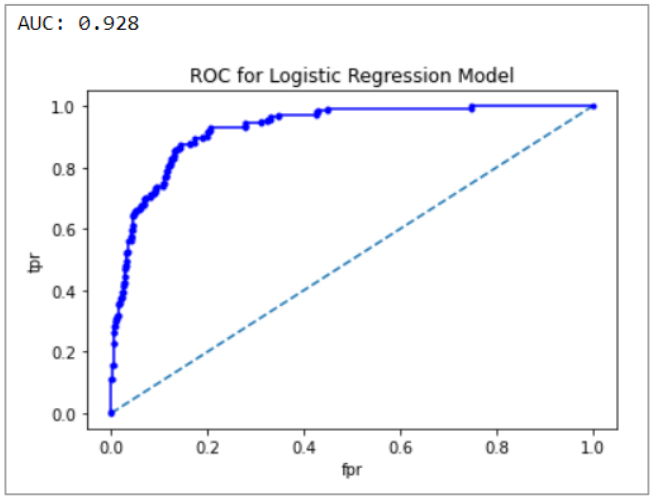


*Figure 15. ROC Curve of LDA for test*

| Model Performance of Random Forest | | | | | |
|---|---|---|---|---|---|
| **Sl. No** | | **Train Data** | | **Test Data** | |
| **1.** | True Positive | 219 | | 132 | |
| **2.** | True Negative | 1804 | | 817 | |
| **3.** | False Positive | 353 | | 225 | |
| **4.** | False Negative | 26 | | 10 | |
| **6.** | Accuracy | 84% | | 80% | |
| | | **Defaulter (1)** | **Non-Defaulters (0)** | **Defaulter (1)** | **Non-Defaulters (0)** |
| **7.** | Precision | 38% | 99% | 37% | 99% |
| **8.** | Recall | 89% | 84% | 93% | 78% |
| **9.** | F1 score | 54% | 90% | 53% | 87% |

*Table 4. Model performance table for LDA*

The accuracy of the test dataset is less than 10% of that of the train dataset, indicating that the dataset is not overfitting. The model is a decent classification model in general. Furthermore, the derived metrics have high values and are well-fitting.

**Interpretation from the Linear Discriminant Analysis model:**

- Looking at the LDA model's performance data, it appears that it is functioning normally and making predictions that are 80 percent correct.
- The train accuracy is about 84 percent, indicating that the model is not overfitting.
- We also have a high recall value, which is one of the most important criteria for defaulters
- However, we can observe that the Precision value in this model is significantly lower. That suggests the number of False Positives is extremely high, which is not a good indicator.
- The ROC curve seems decent as well.
- The area under the curve is 92.8 percent.

### 1.12 Compare the performances of Logistics, Radom Forest and LDA models (include ROC Curve)

We've finished building our three various models:

1) Regression Logistic
2) Forest of Chance
3) Linear Discriminant Analysis (LDA)

We also evaluated each model's performance using performance metrics such as the confusion matrix, classification report, and ROC curve, and others.

Now we'll put these models up against one another. To begin, created a table with all of the models and their results against each other for a better comparison.

| Models | Accuracy - Train | Accuracy - Test | Recall (1) -Train | Recall (1) - Test | Precision (1) - Train | Precision (1) - Test |
|---|---|---|---|---|---|---|
| Logistic Regression | 0.94 | 0.93 | 0.90 | 0.93 | 0.67 | 0.63 |
| Random Forest | 0.98 | 0.98 | 0.87 | 0.90 | 0.92 | 0.93 |
| LDA | 0.84 | 0.80 | 0.89 | 0.93 | 0.38 | 0.37 |

*Table 5. Model performance comparison*

According to the table above, the random forest model has the highest accuracy and precision values, while Logistic Regression has the highest recall. The recall of Logistic regression and Random Forest aren't that distinct. Out of the three models, random forest appears to be the most accurate prediction model. Out of the three models, the LDA model has the lowest performance.

To have a better understanding, we may look at the ROC curves of the three models.

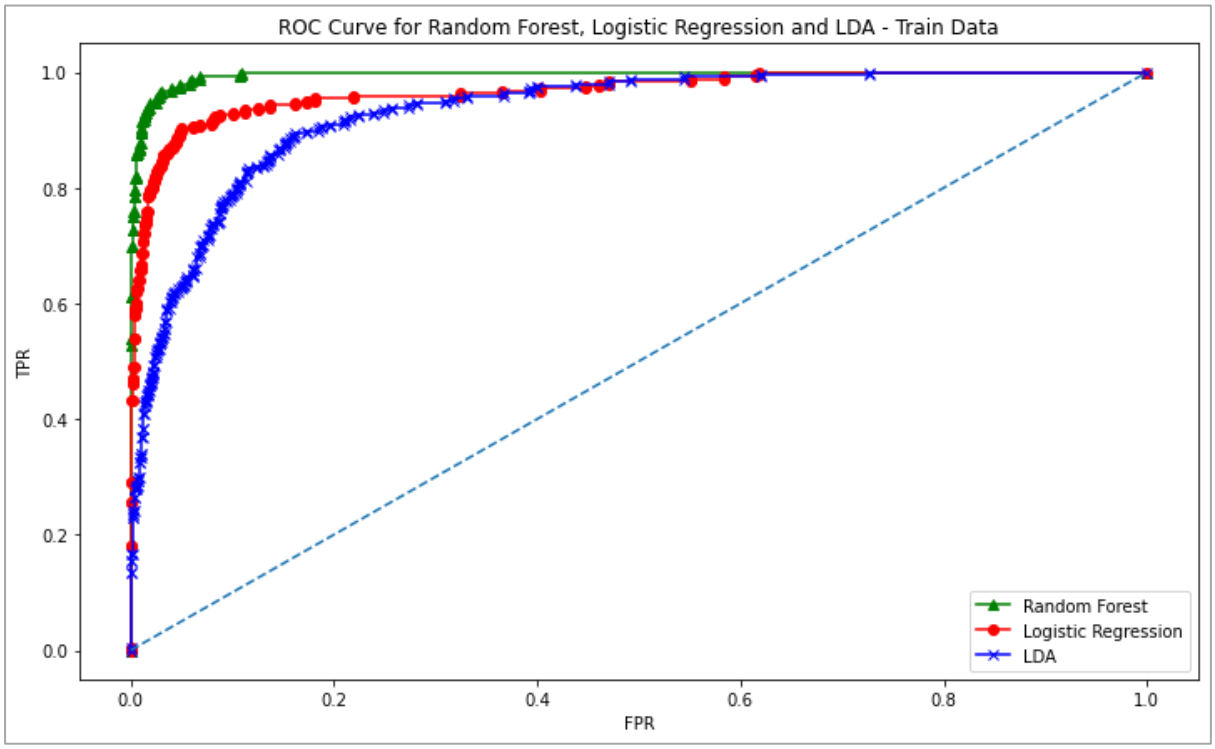


*Figure 16. ROC curves for Model comparisons for train*

```
Area under the curve for Random Forest Model is 1.0
Area under the curve for Logistic Regression Model is 0.97
Area under the curve for LDA Model is 0.93
```

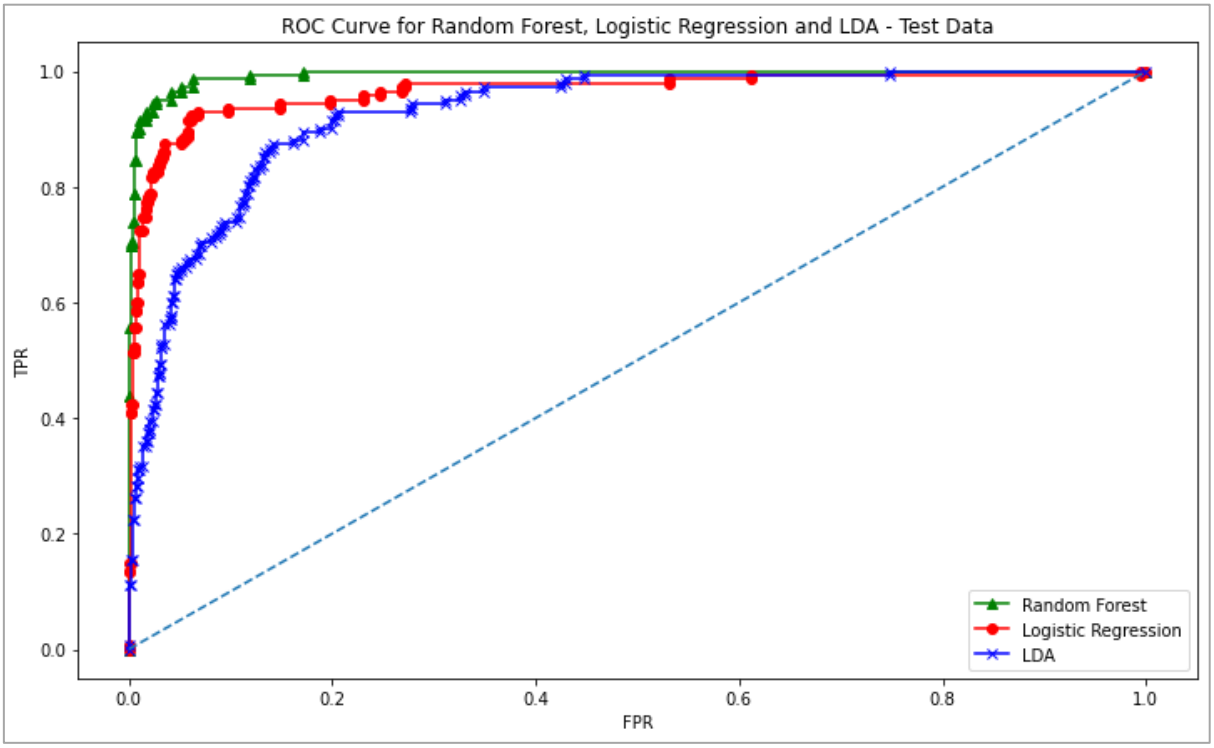


*Figure 17. ROC curves for Model comparisons for test*

```
Area under the curve for Random Forest Model is 0.99
Area under the curve for Logistic Regression Model is 0.97
Area under the curve for LDA Model is 0.93
```

We may infer from the above graph that the Random Forest model has the highest area under the curve, followed by Logistic Regression, and finally LDA. Random Forest is the best model from the ROC curve for our dataset.

Noticeably, looking at the AUC_ROC Curve of both Train & Test Data for all the three models Random Forest is the best fit model as it has cleanly predicted the Defaulters and Non-Defaulters without any False Predictions

To summarise, all three models are good and would provide a reasonable prediction for defaulters, but the **Random Forest model is the best**. It would provide the most accurate forecasts for defaulters.

### 1.13 State Recommendations from the above models

- Based on the three models we've built, we can make the following recommendations:
- Random Forest is the best model for predicting defaulters.
- On the test dataset, we can get an accuracy of 98 percent and a recall of 90 percent using the Random Forest model.
- The LDA model is also a good one, although it is the least accurate of the three.
- The most significant variables from our model are:
    - Market Capitalisation
    - Total Debt o Equity Paid Up

    - Cash Flow from Opr
    - ROG Net Worth perc
    - Book Value Unit Curr 10
    - Value of Output to gross Block
    - CPM perc Latest
    - Curr Ratio Latest
    - Debtors Ratio Latest
- Coefficients of all variables indicate the weightage of that variable in predictions of Defaulters and non-defaulters.
- Positive coefficient means, higher value of the variable will lead to higher likelihood for default.
- Negative coefficient means, higher value of this variable will lead to lower likelihood for default.
- The variables listed above are key in distinguishing defaulters from non-defaulters.
- For example, if a company's total debt is big, there's a good possibility it'll default. As a result, things like these must be kept under check.
- Checking the sign of the coefficient is also a good idea. If any variable in the logistic model (statsmodel) summary has a negative coefficient, that suggests it is negatively related to default probability.
- The default assumption is based on Networth next year. If Networth next year is positive, company does not default. If it is negative, it defaults.
- To keep track of Networth next year, try to come up with a sensible structure. So that you may quickly determine how much and how bad the company's net worth will be. It would be good to know this ahead of time for investors.
- Lower the Book_value_unit_curr i.e. Net assets, higher is the chance of a default, which would mean the net worth next year for this company is expected to be negative.
- Lower the CEPS_annualised_Unit_Curr i.e. Cash earnings per share, higher is the change of a default.
- Higher the Curr_Ratio_Latest, i.e. the companies' ability to pay short term dues, lower are its chances of defaulting or having a negative net worth in the next year.
- Higher the Interest_Cover_Ratio_Latest lower the chances of default. Which means easier the company is able to pay the interest on its outstanding debt, lower are its chances to default.
- Curr_Ratio_Latest is most important criteria amongst the above parameters, while Interest_Cover_Ratio_Latest is the least important when considering only these 4 parameters. However all these 4 parameters remain important compared to the other variables in the data set.
- **False positives (FP)** are datapoints that are not default but are anticipated to be such. This is sometimes referred to as a type 1 error. We must raise the model's accuracy in order to decrease type 1 error (among the points identified as positives by the model how many are actually positive).
  In this case study, type 1 error indicates the model classified the data point as 1 instead of 0. Companies that have been mistakenly designated as defaulting may attempt to review their net value, modify their working capital, or adjust their strategy as needed. They might sell their non-profitable investments instead. This might not be of priority

for our case study, since predicting the actual non-defaulters as default might have minimal impact than false negatives tend to pose.

- **False negative (FN)** data points are those that are truly defaulters but are expected to be non-defaulters. This is referred to as a type 2 error. To lessen the likelihood of a type 2 error. We need to improve recall (how many actual true data points are identified as true by the model) Type 2 errors occur when the model incorrectly classifies a data point as 0 rather than 1.

  In such event, the firm will continue on its current course, making it a risky investment for stockholders. The importance of focusing on this sort of inaccuracy cannot be overstated.

---

## Problem Statement -2

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. Objective of the report

**Objective of the report:**
The report's goal is to examine the previous six years of weekly stock price data for 10 different stocks and compute the mean and standard deviation of stock returns and provide insights.

Below is the list of stocks provided in the dataset:

**Infosys, Indian Hotel, Mahindra & Mahindra, Axis Bank, SAIL, Shree Cement, Sun Pharma, Jindal Steel, Idea Vodafone and Jet Airways.**

**Exploratory data analysis:**

**Read and view data:**

Reading the dataset from the excel file and checking the head () of the dataset i.e., the first 5 rows of the dataset. We will be replacing any space in column names with underscore and we will also be removing any dots if there are in column names. The first five rows of the imported dataset look like below:

| | Date | Infosys | Indian_Hotel | Mahindra_&_Mahindra | Axis_Bank | SAIL | Shree_Cement | Sun_Pharma | Jindal_Steel | Idea_Vodafone | Jet_Airways |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 31-03-2014 | 264 | 69 | 455 | 263 | 68 | 5543 | 555 | 298 | 83 | 278 |
| 1 | 07-04-2014 | 257 | 68 | 458 | 276 | 70 | 5728 | 610 | 279 | 84 | 303 |
| 2 | 14-04-2014 | 254 | 68 | 454 | 270 | 68 | 5649 | 607 | 279 | 83 | 280 |
| 3 | 21-04-2014 | 253 | 68 | 488 | 283 | 68 | 5692 | 604 | 274 | 83 | 282 |
| 4 | 28-04-2014 | 256 | 65 | 482 | 282 | 63 | 5582 | 611 | 238 | 79 | 243 |

*Figure 18. Market risk dataset*

**Checking for number of rows and columns:**

```
The number of rows (observations) is 314
The number of columns (variables) is 11
```

**Checking data type of data features:**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 314 entries, 0 to 313
Data columns (total 11 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Date                 314 non-null    object
 1   Infosys              314 non-null    int64
 2   Indian_Hotel         314 non-null    int64
 3   Mahindra_&_Mahindra  314 non-null    int64
 4   Axis_Bank            314 non-null    int64
 5   SAIL                 314 non-null    int64
 6   Shree_Cement         314 non-null    int64
 7   Sun_Pharma           314 non-null    int64
 8   Jindal_Steel         314 non-null    int64
 9   Idea_Vodafone        314 non-null    int64
 10  Jet_Airways          314 non-null    int64
```

- The dataset contains 6 years of information (weekly stock information) on the stock prices of 10 different Indian Stocks
- Data on stock prices is available from March 31, 2014, through March 30, 2020.
- Categorical and numerical variables are included in the data.
- There are ten distinct stocks in the dataset, as well as one date column.
- The dates are split weekly and go back up to the last six years.
- As a result, there are 314 observations in all.
- There are no missing or null values in the data.
- There are no duplicate values in the data.

**Descriptive Statistics:**

Descriptive statistics help describe and understand the features of a specific data set by giving short summaries about the sample and measures of the data. The most recognized types of descriptive statistics are measures of centre: the mean, median, and mode, which are used at almost all levels of math and statistics.

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Infosys** | 314.0 | 511.340764 | 135.952051 | 234.0 | 424.00 | 466.5 | 630.75 | 810.0 |
| **Indian_Hotel** | 314.0 | 114.560510 | 22.509732 | 64.0 | 96.00 | 115.0 | 134.00 | 157.0 |
| **Mahindra_&_Mahindra** | 314.0 | 636.678344 | 102.879975 | 284.0 | 572.00 | 625.0 | 678.00 | 956.0 |
| **Axis_Bank** | 314.0 | 540.742038 | 115.835569 | 263.0 | 470.50 | 528.0 | 605.25 | 808.0 |
| **SAIL** | 314.0 | 59.095541 | 15.810493 | 21.0 | 47.00 | 57.0 | 71.75 | 104.0 |
| **Shree_Cement** | 314.0 | 14806.410828 | 4288.275085 | 5543.0 | 10952.25 | 16018.5 | 17773.25 | 24806.0 |
| **Sun_Pharma** | 314.0 | 633.468153 | 171.855893 | 338.0 | 478.50 | 614.0 | 785.00 | 1089.0 |
| **Jindal_Steel** | 314.0 | 147.627389 | 65.879195 | 53.0 | 88.25 | 142.5 | 182.75 | 338.0 |
| **Idea_Vodafone** | 314.0 | 53.713376 | 31.248985 | 3.0 | 25.25 | 53.0 | 82.00 | 117.0 |
| **Jet_Airways** | 314.0 | 372.659236 | 202.262668 | 14.0 | 243.25 | 376.0 | 534.00 | 871.0 |

*Figure 19. Descriptive statistics*

**Insights from Descriptive Statistics:**

Generally, people would be interested in the returns given by the respective stocks and not the price. Hence, descriptive statistics of stock_price might not be very useful but still listing down few observations:

- Amongst these 10 stocks Shree_Cement is the highest priced stock overall while SAIL and Idea_Vodafone are the lowest priced stocks.
- Infosys: On an average stock price of Infosys is 466.5 with a minimum Stock price of 234.0 and a maximum of 810.0.
- Indian_Hotel: On an average stock price of Indian_Hotel is 115 with a minimum Stock price of 64.0 and a maximum of 157.0.
- Mahindra_&Mahindra: On an average stock price of Mahindra&_Mahindra is 625.0 with a minimum Stock price of 284.0 and a maximum of 808.0.
- Axis_Bank: On an average stock price of Axis_Bank is 528.0 with a minimum Stock price of 263.0 and a maximum of 956.0.

- SAIL: On an average stock price of SAIL is 21.0 with a minimum Stock price of 263.0 and a maximum of 104.0.
- Shree_Cement: On an average stock price of Shree_Cement is 16018.5 with a minimum Stock price of 5543.0 and a maximum of 24806.0.
- Sun_Pharma: On an average stock price of Sun_Pharma is 614.0 with a minimum Stock price of 338.0 and a maximum of 1089.0.
- Jindal_Steel: On an average stock price of Jindal_Steel is 142.5 with a minimum Stock price of 53.0 and a maximum of 338.0.
- Idea_Vodafone: On an average stock price of Idea_Vodafone is 53.0 with a minimum Stock price of 3.0 and a maximum of 117.0.
- Jet_Airways: On an average stock price of Jet_Airways is 376.0 with a minimum Stock price of 14.0 and a maximum of 871.0.

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

We will be checking the stock prices over the period of time for any two stocks given in the dataset.

**Infosys stock vs Time**

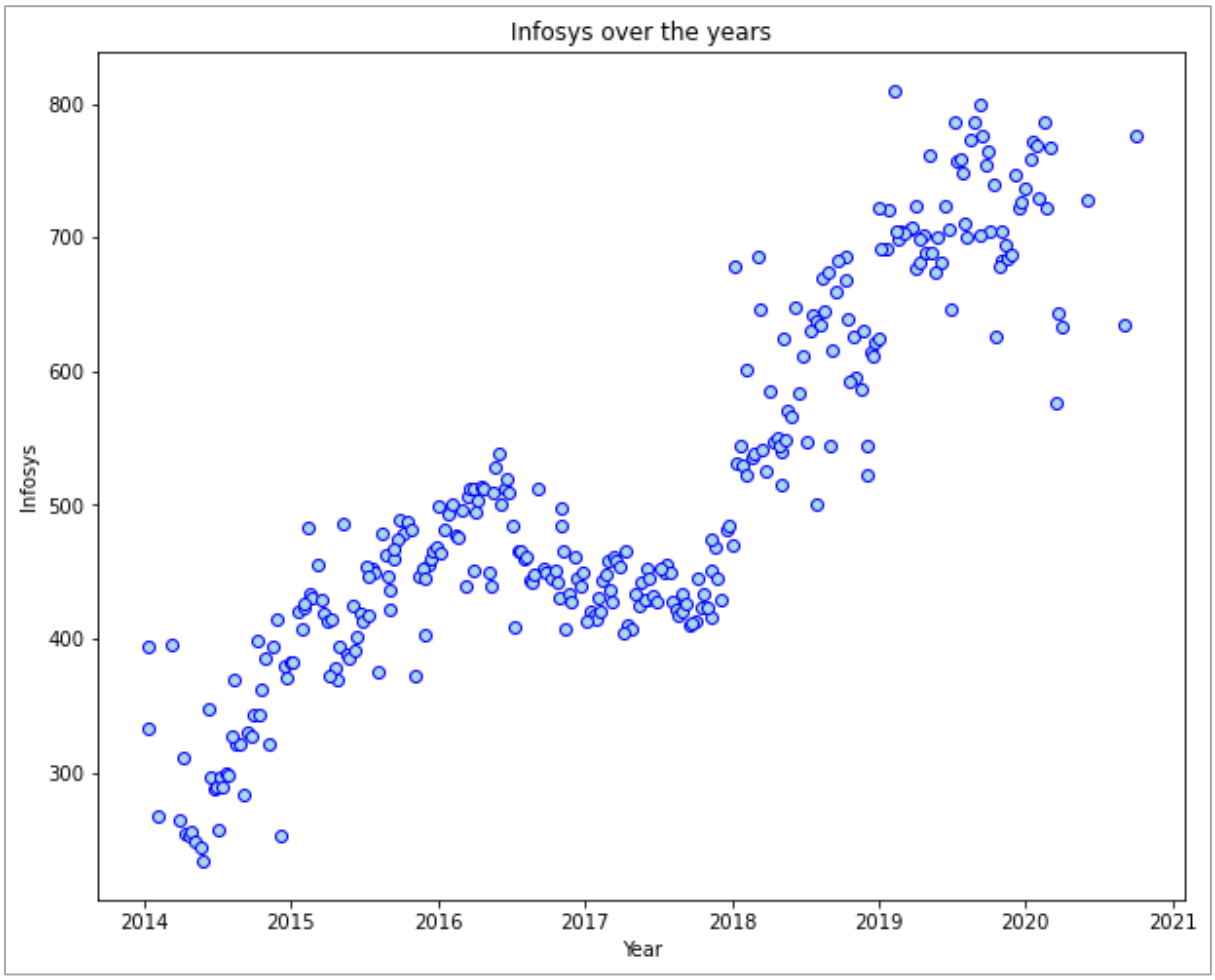


*Figure 20. Infosys stocks over the years*

**Inferences:**

- Over the years, Infosys has done an excellent job with stock pricing. The graph shows that prices have been rising steadily throughout the years.
- There was a little drop in 2017 and 2018, but it rebounded quickly after that and significantly boosted pricing.
- The stock prices were at 250 in 2014, and by 2021, they were around 800.
- As a result, the stock price of Infosys increased significantly.

**Mahindra_&Mahindra stock vs Time**

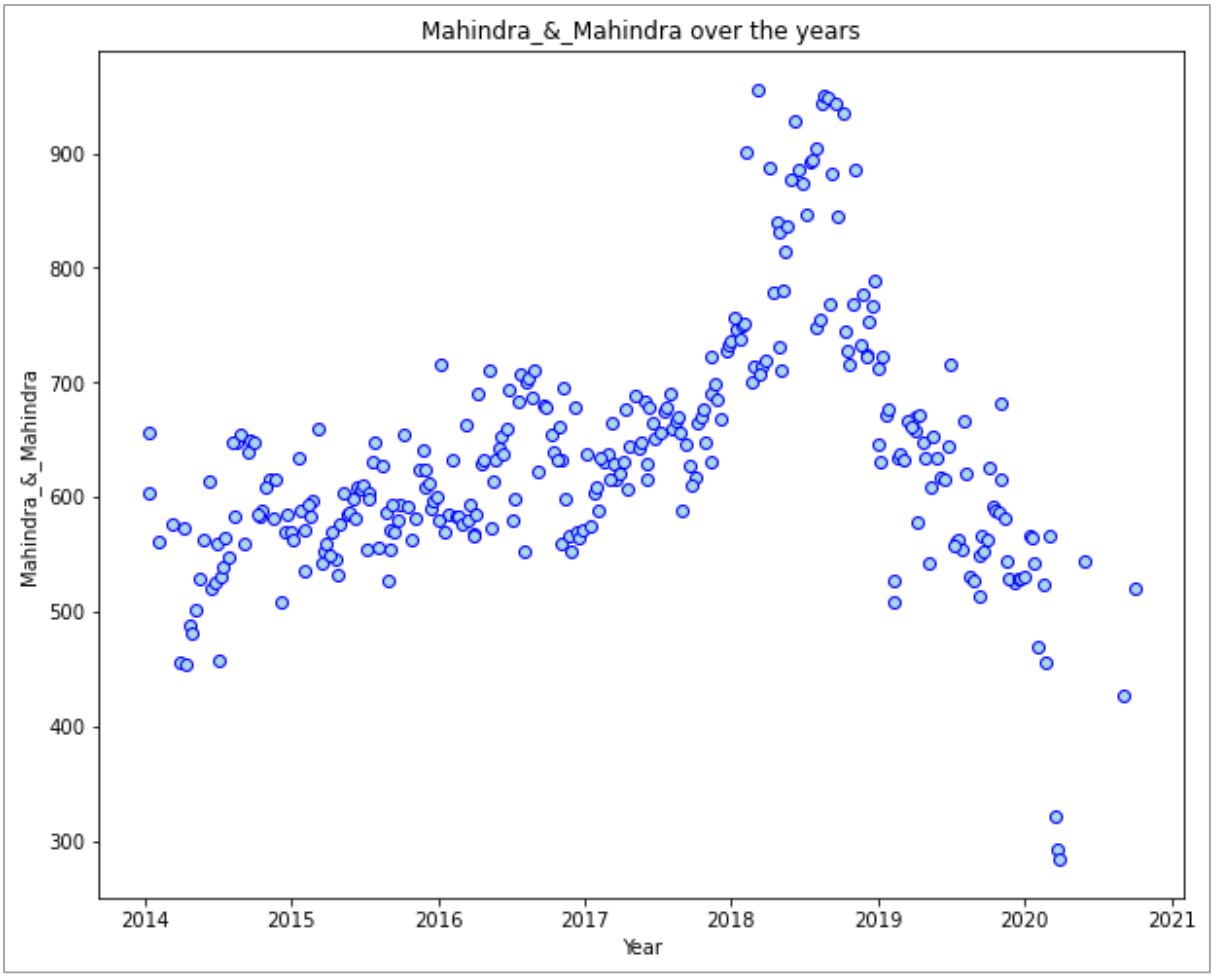


*Figure 21. Mahindra & Mahindra stocks over the years*

**Inferences:**

- Between 2014 and 2017 year-end though the stock price is increasing but not much of a fluctuation can be observed.
- Sharp Increasing trend can be observed from Year 2017 till mid 2018 end followed by a constant dip in stock price.
- Stock price has gone as low as 284.0 and has seen a high of 808.0.

## 2.2 Calculate Returns for all stocks with inference

As stated earlier, we are more concerned about stock returns. As a result, we'll dig further into the results of these stocks. These returns can be calculated in two ways:

- Taking Logarithms
- Taking Differences

We'll apply the logarithm approach to look at the returns on these stock prices, which are the difference between two consecutive days' prices. We'll calculate the difference between two consecutive weeks' prices because data is collected on a weekly basis.

1. Below are the steps followed to calculate the stock returns:
2. Take the stock price dataset and remove the date column.
3. Take logarithm of all stock prices in the dataset
4. Take difference of stock price from the previous period stock price

**Checking top 5 rows of stock returns:**

| | Infosys | Indian_Hotel | Mahindra_&_Mahindra | Axis_Bank | SAIL | Shree_Cement | Sun_Pharma | Jindal_Steel | Idea_Vodafone | Jet_Airways |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| 1 | -0.026873 | -0.014599 | 0.006572 | 0.048247 | 0.028988 | 0.032831 | 0.094491 | -0.065882 | 0.011976 | 0.086112 |
| 2 | -0.011742 | 0.000000 | -0.008772 | -0.021979 | -0.028988 | -0.013888 | -0.004930 | 0.000000 | -0.011976 | -0.078943 |
| 3 | -0.003945 | 0.000000 | 0.072218 | 0.047025 | 0.000000 | 0.007583 | -0.004955 | -0.018084 | 0.000000 | 0.007117 |
| 4 | 0.011788 | -0.045120 | -0.012371 | -0.003540 | -0.076373 | -0.019515 | 0.011523 | -0.140857 | -0.049393 | -0.148846 |

*Figure 22. Stock returns*

Noticeably, first row contains NaN values as the value at index [0] does not have a previous value to be converted into a return and thereafter every other value has been converted to a logarithmic return.

**Checking the summary of stock returns:**

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| Infosys | 313.0 | 0.002794 | 0.035070 | -0.167300 | -0.014514 | 0.004376 | 0.024553 | 0.135666 |
| Indian_Hotel | 313.0 | 0.000266 | 0.047131 | -0.236389 | -0.023530 | 0.000000 | 0.027909 | 0.199333 |
| Mahindra_&_Mahindra | 313.0 | -0.001506 | 0.040169 | -0.285343 | -0.020884 | 0.001526 | 0.019894 | 0.089407 |
| Axis_Bank | 313.0 | 0.001167 | 0.045828 | -0.284757 | -0.022473 | 0.001614 | 0.028522 | 0.127461 |
| SAIL | 313.0 | -0.003463 | 0.062188 | -0.251314 | -0.040822 | 0.000000 | 0.032790 | 0.309005 |
| Shree_Cement | 313.0 | 0.003681 | 0.039917 | -0.129215 | -0.019546 | 0.003173 | 0.029873 | 0.152329 |
| Sun_Pharma | 313.0 | -0.001455 | 0.045033 | -0.179855 | -0.020699 | 0.001530 | 0.023257 | 0.166604 |
| Jindal_Steel | 313.0 | -0.004123 | 0.075108 | -0.283768 | -0.049700 | 0.000000 | 0.037179 | 0.243978 |
| Idea_Vodafone | 313.0 | -0.010608 | 0.104315 | -0.693147 | -0.045120 | 0.000000 | 0.024391 | 0.693147 |
| Jet_Airways | 313.0 | -0.009548 | 0.097972 | -0.458575 | -0.052644 | -0.005780 | 0.036368 | 0.300249 |

*Figure 23. Stock returns summary*

**Inferences -**

- Infosys, Indian Hotel, Axis Bank, and Shree Cement all have positive average returns. In general, these stocks' returns have risen over time.
- Mahindra & Mahindra, SAIL, Sun Pharma, Jindal Steel, Idea Vodafone, and Jet Airways all have negative average returns. Overall, these stocks' returns have dropped over time.
- Infosys has the highest average return.
- Idea Vodafone has the lowest average return.
- Infosys posed the fewest risks (in terms of standard deviation of returns).
- With Jet Airways, the risks (standard deviation of returns) were the highest.

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

We now look at Means & Standard Deviations of these stock returns which is also known as the "Volatility" of a stock

**Stock Means:** These are the average weekly returns on the stock. The expected value, or mean, of all the potential returns of assets in a portfolio is the mean return in securities analysis. A mean return, also known as an anticipated return, refers to the amount of money a stock earns on a weekly basis (in this case).

**Standard Deviation:** It's a measure of volatility, which means the more a stock's returns differ from its average return, the more volatile it is. In other words, the more unstable a stock is, the higher the risk associated with it.

Stock means and standard deviations have been calculated below:

**Stock Means:**

```
Infosys                 0.002794
Indian_Hotel            0.000266
Mahindra_&_Mahindra    -0.001506
Axis_Bank               0.001167
SAIL                   -0.003463
Shree_Cement            0.003681
Sun_Pharma             -0.001455
Jindal_Steel           -0.004123
Idea_Vodafone          -0.010608
Jet_Airways            -0.009548
dtype: float64
```

*Figure 24. Stock Means*

**Stock Standard Deviation:**

```
Infosys                 0.035070
Indian_Hotel            0.047131
Mahindra_&_Mahindra     0.040169
Axis_Bank               0.045828
SAIL                    0.062188
Shree_Cement            0.039917
Sun_Pharma              0.045033
Jindal_Steel            0.075108
Idea_Vodafone           0.104315
Jet_Airways             0.097972
dtype: float64
```

*Figure 25. Stock standard deviation*

Putting both of stock returns and stock standard deviation in a data-frame to get averages and volatility in one place.

| | Average | Volatility |
|---|---|---|
| **Infosys** | 0.002794 | 0.035070 |
| **Indian_Hotel** | 0.000266 | 0.047131 |
| **Mahindra_&_Mahindra** | -0.001506 | 0.040169 |
| **Axis_Bank** | 0.001167 | 0.045828 |
| **SAIL** | -0.003463 | 0.062188 |
| **Shree_Cement** | 0.003681 | 0.039917 |
| **Sun_Pharma** | -0.001455 | 0.045033 |
| **Jindal_Steel** | -0.004123 | 0.075108 |
| **Idea_Vodafone** | -0.010608 | 0.104315 |
| **Jet_Airways** | -0.009548 | 0.097972 |

*Figure 26. Stock mean and volatility data frame*

**Insights:**

- A good stock to invest is the one which has high average returns and has low risks involved with it.
- On the other hand, a bad stock to invest is the one which has low average returns and has high risks involved with it.
- The above calculations once again tell us that Infosys is the best stock among the given ones to invest while jet airways and idea Vodafone are the worst choices
- Idea_Vodafone has the lowest return and also the highest volatility which in turn means higher risk.
- Shree_Cement on the other hand has the highest returns and one of the least risky stocks.
- Infosys and Shree_Cement are the least risky stocks and the return are also giving positive returns

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

A scatter plot is the best approach to plot a graph between Stock Means and Standard Deviation. We may get a better grasp of the better stock by plotting volatility (standard deviation) on the x-axis and means on the y-axis.

The approach is designed to Maximizing means and minimizing risk.

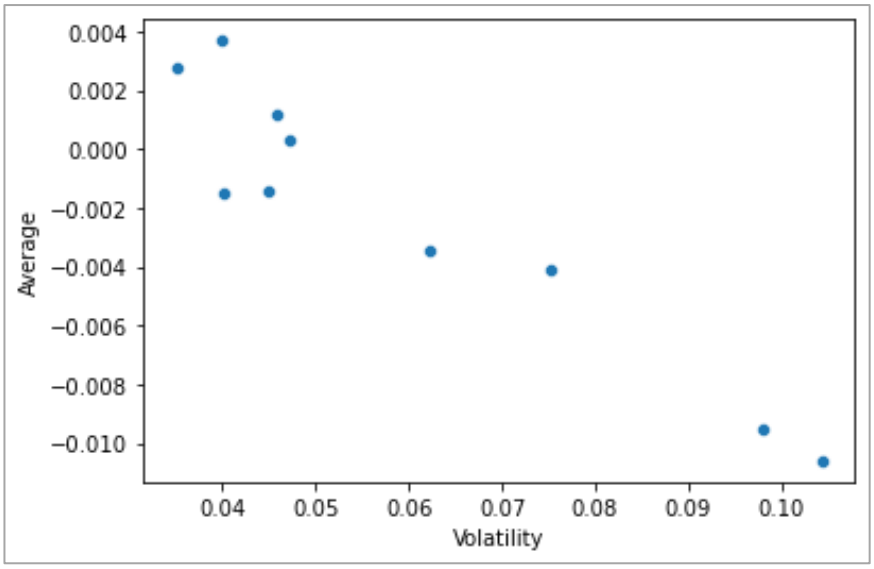


*Figure 27. Stock means vs volatility plot*

**Inferences:**

- We can see from the graph above that the two stocks on the bottom right have the largest risks and the lowest average. These are the stocks we should stay away from.
- The two stocks in the upper left corner of the graph have the lowest risk and the highest averages. As a result, these are the ones we should choose.
- The stocks in the middle are dispersed throughout. Low volatility medium averages are found in some, whereas average volatility is found in others. Average volatility is also found in the middle.

Let us set a threshold for mean and volatility to observe which stocks are above threshold and which are not. For that purpose, let us keep the risk threshold to be 0.05 and stock mean threshold to be 0.00.

Plotting the same on our scatter plot with threshold lines, we observe the following:

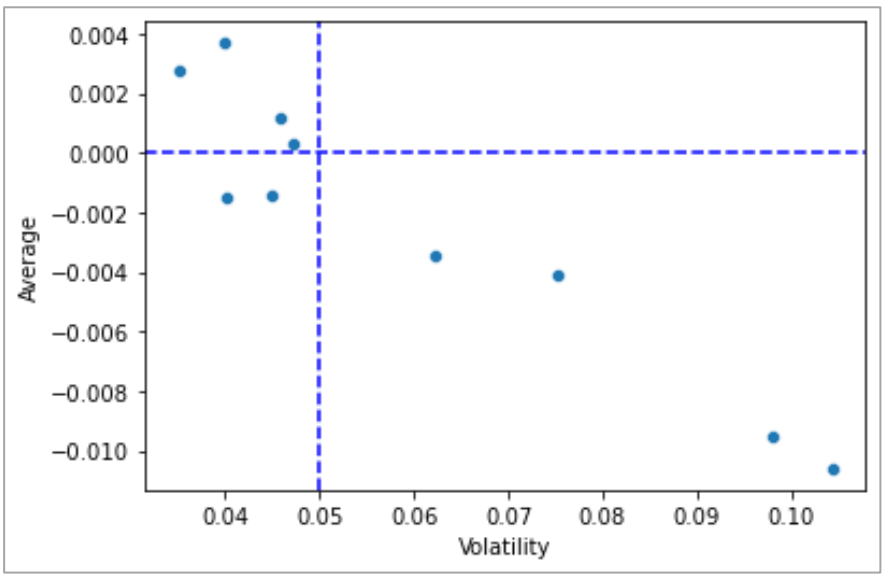


*Figure 28. stock mean and volatility with thresholds*

The threshold cut-offs for Averages and volatility are shown in blue dashed lines. The stocks to the left of the vertical blue dashed line are good to go, and the stocks to the right of the horizontal blue dashed line are okay to go for.

We can see that four stocks meet the criterion for being ready to trade, while the others do not. Let's take a closer look at these four companies and rank them in order of increasing volatility. So, the stock with the least volatility/risk is at the top of the table, while the one with the most volatility/risk is at the bottom.

| | Average | Volatility |
|---|---|---|
| **Infosys** | 0.002794 | 0.035070 |
| **Shree_Cement** | 0.003681 | 0.039917 |
| **Axis_Bank** | 0.001167 | 0.045828 |
| **Indian_Hotel** | 0.000266 | 0.047131 |

*Figure 29. Top 4 stocks*

From the above table we can infer that **Infosys is the one which has the lowest risk whereas 2nd highest stock average.**

## 2.5 Conclusion and Recommendations

**Insights and Conclusion:**

- Stock with a lower mean & higher standard deviation do not play any role in a portfolio with a competing stock giving more returns & less risk.
- When plotted against time, four stocks show an upward trend, they are Infosys, Shree cement, Axis Bank, and Indian hotel are among them.
- When plotted against time, 6 stocks show a declining trend. Mahindra & Mahindra, SAIL, Sun Pharma, Jindal Steel, Idea Vodafone, and Jet Airways are among the companies.
- Infosys has the lowest volatility.
- Shree Cement has the highest stock mean.
- Idea Vodafone has the highest stock return value of 0.69 during the full time period, but it does not have a decent average return value when the complete time period is evaluated.

**Recommendations:**

- For the given data, stock recommendations could be listed as:
    - One with highest return and lowest risk
    - One with lowest risk and highest return.
- Shree Cement, followed by Infosys, might be deemed attractive companies to buy in just on the basis of "Return."
- Considering the Plot: Stock Means vs Standard Deviation to assess the risk to reward ratio. More volatile stock might give short term gains but might not be a good

investment in long term. Whereas a low volatile stock might not be a good investment in short term, but might give a good return in long term.

- Infosys, Shree Cement, Axis Bank, and Indian Hotel are the top four stocks that can be considered for investment.
- Jet Airways and Idea Vodafone are two stocks that should not be considered for investing. After few years re checking on Jet Airways stock. The stock market may increase once again, as it did in 2017. And maybe there is drop in stocks because of covid situations
- Compare these stock prices and values to those of their competitors to see if they are performing well in their respective fields
- While IT sector is growing at a rapid pace, therefore speculating on Infosys won't be risky unless and until something unusual occurs.
- As a result, recommendations may be made depending on the amount and type of investment that an investor is interested in, and a conclusion can be drawn to establish a diversified portfolio for the investor.

---